教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

信息论、推理与学习算法

翻译版

INFORMATION THEORY, INFERENCE, AND LEARNING ALGORITHMS

[英]·DAVID J. C. MACKAY 著
肖明波 席 斌
许 芳 王建新 译

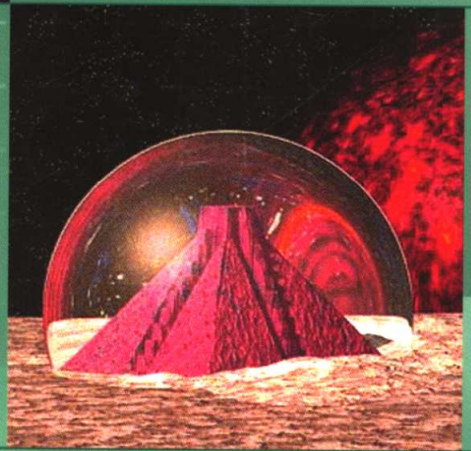




高等教育出版社
Higher Education Press



查询其他教学用书，敬请登录
www.hep-st.com.cn



ISBN 7-04-019641-7
9 787040 196412 >
定价 59.00 元



教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

信息论、推理与学习算法

翻译版

INFORMATION THEORY,INFERENCE, AND LEARNING ALGORITHMS

[英]·DAVID J.C.MACKAY 著
肖明波 席 斌
许 芳 王建新 译



高等教育出版社
Higher Education Press

图字:01－2005－2303 号

Information Theory, Inference and Learning Algorithms

David J. C. MacKay

© Cambridge University Press 2003

内容提要

本书是英国剑桥大学卡文迪许实验室的著名学者 David J. C. MacKay 博士总结多年教学经验和科研成果,于 2003 年推出的一部力作。本书作者不仅透彻地论述了传统信息论的内容和最新编码算法,而且以高度的学科驾驭能力,匠心独具地在一个统一框架下讨论了贝叶斯数据建模、蒙特卡罗方法、聚类算法、神经网络等属于机器学习和推理领域的主题,从而很好地将诸多学科的技术内涵融会贯通。本书注重理论与实际的结合,内容组织科学严谨,反映了多门学科的内在联系和发展趋势。同时,本书还包含了丰富的例题和近 400 道习题(其中许多习题还配有详细的解答),便于教学或自学,适合作为信息科学与技术相关专业高年级本科生和研究生教材,对相关专业技术人员也不失为一本有益的参考书。

图书在版编目(CIP)数据

信息论、推理与学习算法:翻译版/(英)麦凯(Mackay,D. J. C.)著;肖明波等译. -北京.高等教育出版社,2006.7

书名原文:Information Theory,Inference and Learning Algorithms

ISBN 7-04-019641-7

Ⅰ.信… Ⅱ.①麦… ②肖… Ⅲ.信息论－高等学校－教材 Ⅳ.G201

中国版本图书馆 CIP 数据核字(2006)第 060464 号

出版发行	高等教育出版社	**购书热线**	010－58581118
社　　址	北京市西城区德外大街 4 号	**免费咨询**	800－810－0598
邮政编码	100011	**网　　址**	http://www.hep.edu.cn
总　　机	010－58581000		http://www.hep.com.cn
		网上订购	http://www.landraco.com
经　　销	蓝色畅想图书发行有限公司		http://www.landraco.com.cn
印　　刷	北京民族印刷厂	**畅想教育**	http://www.widedu.com
开　　本	787×1092　1/16	**版　　次**	2006 年 7 月第 1 版
印　　张	47.75	**印　　次**	2006 年 7 月第 1 次印刷
字　　数	1 030 000	**定　　价**	59.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有　侵权必究

物料号　19641-00

前　言

20世纪末，以计算机和通信技术为代表的信息科学和技术对世界经济、科技、军事、教育和文化等产生了深刻影响。信息科学技术的迅速普及和应用，带动了世界范围信息产业的蓬勃发展，为许多国家带来了丰厚的回报。

进入21世纪，尤其随着我国加入WTO，信息产业的国际竞争将更加激烈。我国信息产业虽然在20世纪末取得了迅猛发展，但与发达国家相比，甚至与印度、爱尔兰等国家相比，还有很大差距。国家信息化的发展速度和信息产业的国际竞争能力，最终都将取决于信息科学技术人才的质量和数量。引进国外信息科学和技术优秀教材，在有条件的学校推动开展英语授课或双语教学，是教育部为加快培养大批高质量的信息技术人才采取的一项重要举措。

为此，教育部要求由高等教育出版社首先开展信息科学和技术教材的引进试点工作。同时提出了两点要求，一是要高水平，二是要低价格。在高等教育出版社和信息科学技术引进教材专家组的努力下，经过比较短的时间，第一批引进的20多种教材已经陆续出版。这套教材出版后受到了广泛的好评，其中有不少是世界信息科学技术领域著名专家、教授的经典之作和反映信息科学技术最新进展的优秀作品，代表了目前世界信息科学技术教育的一流水平，而且价格也是最优惠的，与国内同类自编教材相当。

这项教材引进工作是在教育部高等教育司和高教社的共同组织下，由国内信息科学技术领域的专家、教授广泛参与，在对大量国外教材进行多次遴选的基础上，参考了国内和国外著名大学相关专业的课程设置进行系统引进的。其中，John Wiley公司出版的贝尔实验室信息科学研究中心副总裁Silberschatz教授的经典著作《操作系统概念》，是我们经过反复谈判，做了很多努力才得以引进的。William Stallings先生曾编写了在美国深受欢迎的信息科学技术系列教材，其中有多种教材获得过美国教材和学术著作者协会颁发的计算机科学与工程教材奖，这批引进教材中就有他的两本著作。留美中国学者Jiawei Han先生的《数据挖掘》是该领域中具有里程碑意义的著作。由达特茅斯学院的Thomas Cormen和麻省理工学院、哥伦比亚大学几位学者共同编著的经典著作《算法导论》，在经历了11年的锤炼之后于2001年出版了第二版。目前任教于美国Massachusetts大学的James Kurose教授，曾在美国三所高校先后10次获得杰出教师或杰出教学奖，由他主编的《计算机网络》出版后，以其体系新颖、内容先进而备受欢迎。在努力降低引进教材售价方面，高等教育出版社做了大量和细致的工作。这套引进的教材体现了权威性、系统性、先进性和经济性等特点。

教育部也希望国内和国外的出版商积极参与此项工作，共同促进中国信息技术

教育和信息产业的发展。我们在与外商的谈判工作中，不仅要坚定不移地引进国外最优秀的教材，而且还要千方百计地将版权转让费降下来，要让引进教材的价格与国内自编教材相当，让广大教师和学生负担得起。中国的教育市场巨大，外国出版公司和国内出版社要通过扩大发行数量取得效益。

在引进教材的同时，我们还应做好消化吸收，注意学习国外先进的教学思想和教学方法，提高自编教材的水平，使我们的教学和教材在内容体系上，在理论与实践的结合上，在培养学生的动手能力上能有较大的突破和创新。

目前，教育部正在全国35所高校推动示范性软件学院的建设和实施，这也是加快培养信息科学技术人才的重要举措之一。示范性软件学院要立足于培养具有国际竞争力的实用性软件人才，与国外知名高校或著名企业合作办学，以国内外著名IT企业为实践教学基地，聘请国内外知名教授和软件专家授课，还要率先使用引进教材开展教学。

我们希望通过这些举措，能在较短的时间，为我国培养一大批高质量的信息技术人才，提高我国软件人才的国际竞争力，促进我国信息产业的快速发展，加快推动国家信息化进程，进而带动整个国民经济的跨越式发展。

教育部高等教育司

二〇〇二年三月

译 者 序

1948 年，香农发表了他的划时代论文 *The Mathematical Theory of Communication*，宣告了信息论这一崭新学科的诞生。信息论不仅给出有效通信的极限，而且明确地指出了实现有效而可靠通信的必由之路。通过 50 多年的努力，人们不仅在理论上发展了香农信息论，而且实际通信系统也在逐步逼近香农极限。信息论是通信理论的重要基石，同时也正迅速渗透到其他相关的自然科学和社会科学领域，显示出强大的生命力和不可估量的应用前景。所有这一切在本书中都得到了绝佳体现，我们很高兴能将它翻译出来推荐给大家。

本书是英国著名的卡文迪许实验室的著名学者 David J. C. MacKay 博士总结多年教学经验和科研成果，于 2003 年推出的一部力作。它不仅包含传统信息论的内容（如香农定理和通信系统理论基础），还对 LDPC 等最新的编码算法进行了论述。此外，本书作者以高度的学科驾驭能力，匠心独具地在一个统一框架下讨论了贝叶斯数据建模、蒙特卡罗方法、聚类算法、神经网络等属于机器学习和推理领域的主题。从而很好地将通信、信号处理、数据挖掘、机器学习、模式识别、计算神经学、生物信息学和密码学等诸多学科的技术内涵融会贯通。本书注重理论与实际的结合，内容组织科学严谨，反映了多门学科的内在联系和发展趋势。同时，本书还包含了丰富的例题和近 400 道习题（其中许多习题还配有详细的解答），便于教学或自学，适合作为信息科学与技术相关专业高年级本科生和研究生教材，对相关专业技术人员也不失为一本有益的参考书。虽然信息论相关技术的发展日新月异，我们相信本书的内容不会很快过时；虽然国内外关于信息论的专著和教材已经出版很多，但本书有其显著特色和优势，有望成为一本素质教育教材。

本书主体部分包含六大部分，共 50 章。第一部分至第三部分为传统信息论的内容，讨论了信源编码定理和数据压缩、噪声信道编码定理和纠错码以及散列码等专题。第四部分涉及概率与推理，着重论述贝叶斯推理、聚类算法、蒙特卡罗方法以及决策论等内容，并指出了基于抽样理论的传统推理方法所存在的不足。第五部分神经网络，讨论了单神经元分类器、神经网络学习推理和 Hopfield 网络，还着重推导了神经网络的信息容量。第六部分稀疏图码，介绍 LDPC 码、Turbo 码、重复累加码和数字喷泉码等性能优越的“后现代”信道编码算法。

本书由肖明波、席斌、许芳和王建新等人翻译。具体分工为：肖明波负责前言、第 1 章至第 3 章、第 27 章至第 30 章和附录部分；王建新负责第 4 章至第 7 章、第 12 章至第 19 章和第 31 章至第 33 章；许芳负责第 8 章至第 11 章、第 34 章至第 37 章和第 47 章至第 50 章；席斌负责第 20 章至第 26 章和第 38 章至第 46 章。肖明波教授对全

书进行了审校。参加本书初稿翻译、校对、讨论、文字整理和公式录入等工作的还有杨光松、张静、卢菲恬、喻继森、林碧娇、苏杰、林小兰等人，在此对他们认真出色的工作和无私奉献的精神表示衷心感谢。

由于本书涉及领域广泛，而译者水平有限，加上时间紧迫，译文中难免存在不妥乃至错误之处，敬请读者和同行专家指正。

译 者

2006 年 2 月

序　言

本书为工程、科学、数学和计算等学科的高年级本科生和研究生而作。本书假设学生们熟悉本科一、二年级所学的微积分、概率论和线性代数等科学和工程数学课程的内容。

传统信息论课程不仅涵盖了优美的香农理论思想，还涵盖了通信问题的一些实际解决方案。本书则更进一步，论述了贝叶斯数据建模、蒙特卡罗方法、变参法、聚类算法和神经网络。

为什么要将信息论和机器学习统一起来呢？因为它们本来是一件事的两个方面。在20世纪60年代，一个叫控制论的领域中聚集了众多的信息论专家、计算机科学家和神经科学家，对共同的课题展开研究。信息论和机器学习至今仍然归划为一体。大脑是终极的压缩和通信系统。而数据压缩和纠错码的现代算法所使用的工具与机器学习中的相同。

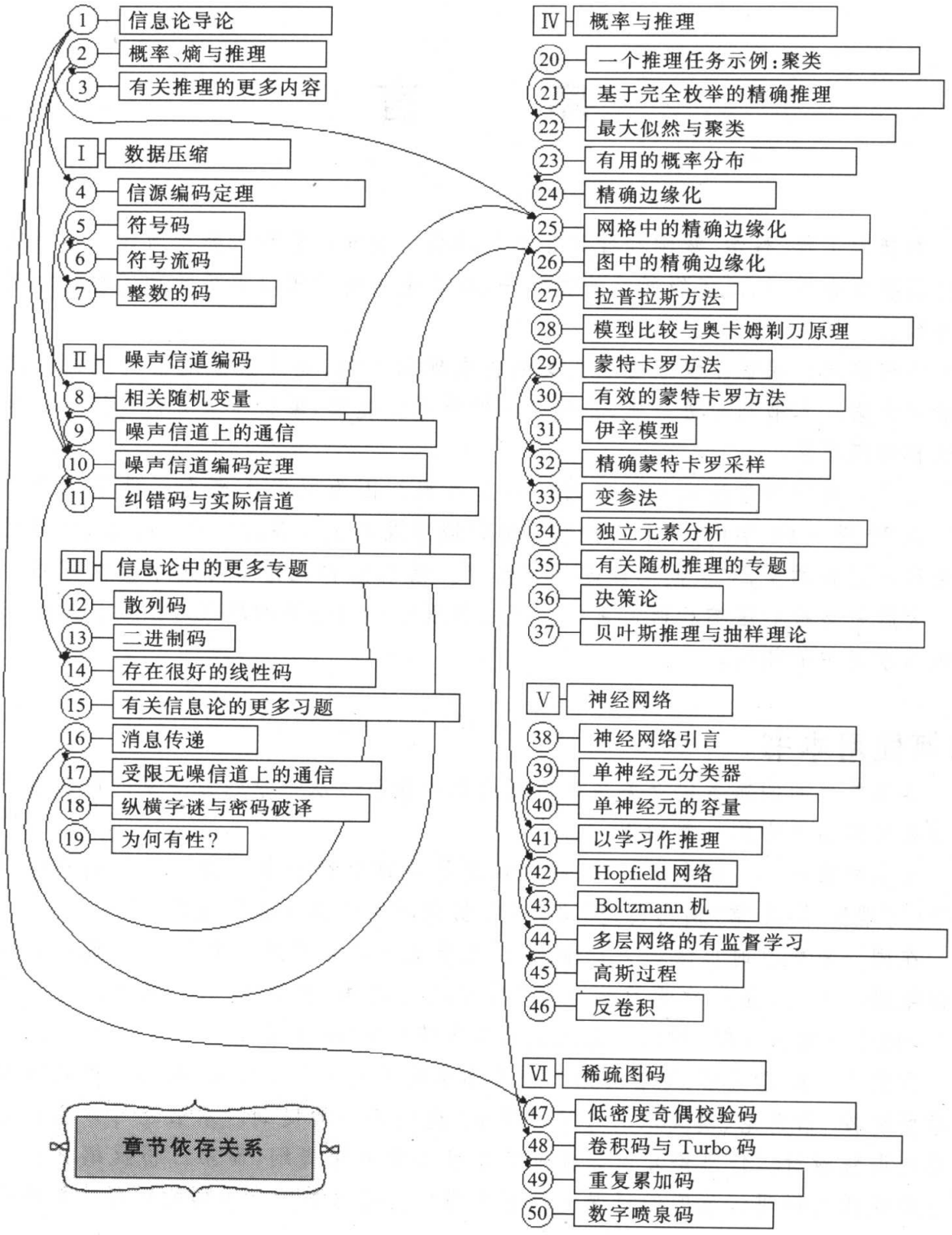
如何使用本书

各章节之间的基本依存关系如后面的图中所示。从一章到另一章的箭头表明后者需要用到前者中的一些知识。

在本书第一、二、四和五部分，高级或选学的章节都放在后面。第三部分的全部章节（可能除第16章“消息传递”之外）在初读时都可作为选学内容。

在同一章内有时也做如下安排：最后几节往往是关于高级主题的，因此在初读时可以略过。例如，在初次学习第4章“信源编码定理”和第10章“噪声信道编码定理”这两个关键章节时，可以分别绕过第4.5节和第10.4节。

在接下来的几页中，笔者将给出使用本书的几种方式。首先，给出了本人在剑桥大学所讲授“信息论、模式识别与神经网络”这门课程的提纲。本书还可作为传统的信息论课程教材。信息论导论课程可参考第二份授课提纲，而第三份提纲旨在让学生理解现代纠错码。第四份授课提纲则表明如何将本书用于传统的机器学习课程。


1 信息论导论
2 概率、熵与推理
3 有关推理的更多内容
Ⅰ 数据压缩
4 信源编码定理
5 符号码
6 符号流码
7 整数的码
Ⅱ 噪声信道编码
8 相关随机变量
9 噪声信道上的通信
10 噪声信道编码定理
11 纠错码与实际信道
Ⅲ 信息论中的更多专题
12 散列码
13 二进制码
14 存在很好的线性码
15 有关信息论的更多习题
16 消息传递
17 受限无噪信道上的通信
18 纵横字谜与密码破译
19 为何有性?
Ⅳ 概率与推理
20 一个推理任务示例:聚类
21 基于完全枚举的精确推理
22 最大似然与聚类
23 有用的概率分布
24 精确边缘化
25 网格中的精确边缘化
26 图中的精确边缘化
27 拉普拉斯方法
28 模型比较与奥卡姆剃刀原理
29 蒙特卡罗方法
30 有效的蒙特卡罗方法
31 伊辛模型
32 精确蒙特卡罗采样
33 变参法
34 独立元素分析
35 有关随机推理的专题
36 决策论
37 贝叶斯推理与抽样理论
Ⅴ 神经网络
38 神经网络引言
39 单神经元分类器
40 单神经元的容量
41 以学习作推理
42 Hopfield 网络
43 Boltzmann 机
44 多层网络的有监督学习
45 高斯过程
46 反卷积
Ⅵ 稀疏图码
47 低密度奇偶校验码
48 卷积码与 Turbo 码
49 重复累加码
50 数字喷泉码
章节依存关系

1 信息论导论
2 概率、熵与推理
3 有关推理的更多内容

Ⅰ 数据压缩
4 信源编码定理
5 符号码
6 符号流码
7 整数的码

Ⅱ 噪声信道编码
8 相关随机变量
9 噪声信道上的通信
10 噪声信道编码定理
11 纠错码与实际信道

Ⅲ 信息论中的更多专题
12 散列码
13 二进制码
14 存在很好的线性码
15 有关信息论的更多习题
16 消息传递
17 受限无噪信道上的通信
18 纵横字谜与密码破译
19 为何有性?

Ⅳ 概率与推理
20 一个推理任务示例:聚类
21 基于完全枚举的精确推理
22 最大似然与聚类
23 有用的概率分布
24 精确边缘化
25 网格中的精确边缘化
26 图中的精确边缘化
27 拉普拉斯方法
28 模型比较与奥卡姆剃刀原理
29 蒙特卡罗方法
30 有效的蒙特卡罗方法
31 伊辛模型
32 精确蒙特卡罗采样
33 变参法
34 独立元素分析
35 有关随机推理的专题
36 决策论
37 贝叶斯推理与抽样理论

Ⅴ 神经网络
38 神经网络引言
39 单神经元分类器
40 单神经元的容量
41 以学习作推理
42 Hopfield 网络
43 Boltzmann 机
44 多层网络的有监督学习
45 高斯过程
46 反卷积

Ⅵ 稀疏图码
47 低密度奇偶校验码
48 卷积码与 Turbo 码
49 重复累加码
50 数字喷泉码

我在英国剑桥大学开设的“信息论、模式识别和神经网络”课程提纲


1 信息论导论
2 概率、熵与推理
3 有关推理的更多内容
Ⅰ 数据压缩
4 信源编码定理
5 符号码
6 符号流码
7 整数的码
Ⅱ 噪声信道编码
8 相关随机变量
9 噪声信道上的通信
10 噪声信道编码定理
11 纠错码与实际信道
Ⅲ 信息论中的更多专题
12 散列码
13 二进制码
14 存在很好的线性码
15 有关信息论的更多习题
16 消息传递
17 受限无噪信道上的通信
18 纵横字谜与密码破译
19 为何有性?
“信息论”短期课程提纲
Ⅳ 概率与推理
20 一个推理任务示例:聚类
21 基于完全枚举的精确推理
22 最大似然与聚类
23 有用的概率分布
24 精确边缘化
25 网格中的精确边缘化
26 图中的精确边缘化
27 拉普拉斯方法
28 模型比较与奥卡姆剃刀原理
29 蒙特卡罗方法
30 有效的蒙特卡罗方法
31 伊辛模型
32 精确蒙特卡罗采样
33 变参法
34 独立元素分析
35 有关随机推理的专题
36 决策论
37 贝叶斯推理与抽样理论
Ⅴ 神经网络
38 神经网络引言
39 单神经元分类器
40 单神经元的容量
41 以学习作推理
42 Hopfield 网络
43 Boltzmann 机
44 多层网络的有监督学习
45 高斯过程
46 反卷积
Ⅵ 稀疏图码
47 低密度奇偶校验码
48 卷积码与 Turbo 码
49 重复累加码
50 数字喷泉码


1 信息论导论
2 概率、熵与推理
3 有关推理的更多内容
I 数据压缩
4 信源编码定理
5 符号码
6 符号流码
7 整数的码
II 噪声信道编码
8 相关随机变量
9 噪声信道上的通信
10 噪声信道编码定理
11 纠错码与实际信道
III 信息论中的更多专题
12 散列码
13 二进制码
14 存在很好的线性码
15 有关信息论的更多习题
16 消息传递
17 受限无噪信道上的通信
18 纵横字谜与密码破译
19 为何有性?
IV 概率与推理
20 一个推理任务示例:聚类
21 基于完全枚举的精确推理
22 最大似然与聚类
23 有用的概率分布
24 精确边缘化
25 网格中的精确边缘化
26 图中的精确边缘化
27 拉普拉斯方法
28 模型比较与奥卡姆剃刀原理
29 蒙特卡罗方法
30 有效的蒙特卡罗方法
31 伊辛模型
32 精确蒙特卡罗采样
33 变参法
34 独立元素分析
35 有关随机推理的专题
36 决策论
37 贝叶斯推理与抽样理论
V 神经网络
38 神经网络引言
39 单神经元分类器
40 单神经元的容量
41 以学习作推理
42 Hopfield 网络
43 Boltzmann 机
44 多层网络的有监督学习
45 高斯过程
46 反卷积
VI 稀疏图码
47 低密度奇偶校验码
48 卷积码与 Turbo 码
49 重复累加码
50 数字喷泉码
"信息论与编码"高级课程

1 信息论导论
2 概率、熵与推理
3 有关推理的更多内容

Ⅰ 数据压缩
4 信源编码定理
5 符号码
6 符号流码
7 整数的码

Ⅱ 噪声信道编码
8 相关随机变量
9 噪声信道上的通信
10 噪声信道编码定理
11 纠错码与实际信道

Ⅲ 信息论中的更多专题
12 散列码
13 二进制码
14 存在很好的线性码
15 有关信息论的更多习题
16 消息传递
17 受限无噪信道上的通信
18 纵横字谜与密码破译
19 为何有性?

Ⅳ 概率与推理
20 一个推理任务示例:聚类
21 基于完全枚举的精确推理
22 最大似然与聚类
23 有用的概率分布
24 精确边缘化
25 网格中的精确边缘化
26 图中的精确边缘化
27 拉普拉斯方法
28 模型比较与奥卡姆剃刀原理
29 蒙特卡罗方法
30 有效的蒙特卡罗方法
31 伊辛模型
32 精确蒙特卡罗采样
33 变参法
34 独立元素分析
35 有关随机推理的专题
36 决策论
37 贝叶斯推理与抽样理论

Ⅴ 神经网络
38 神经网络引言
39 单神经元分类器
40 单神经元的容量
41 以学习作推理
42 Hopfield 网络
43 Boltzmann 机
44 多层网络的有监督学习
45 高斯过程
46 反卷积

“贝叶斯推理与机器学习”课程提纲

Ⅵ 稀疏图码
47 低密度奇偶校验码
48 卷积码与 Turbo 码
49 重复累加码
50 数字喷泉码

有关习题

只有通过亲自参与对某一主题的创造，你才能真正理解它。习题在本书中起到了关键作用。为了便于指导，我们对每个题目都标出了等级，用从1到5之间的某个数字表明其难度，这和参考文献Knuth(1968年)所采用的评级类似。



此外，对于作者特别推荐的习题，在页边用“鼓励鼠”标记出来。有些需要使用计算机的习题，则用“*C*”标记出来。

书中给出了许多习题的答案。请善用之。在给出解答的地方，还给出了原题的难度等级。

许多其他习题的答案可供本书授课教师使用，如需要请发电子邮件至 solutions@cambridge.org。



习题标记汇总

特别推荐

▷　推荐

C　需要使用计算机的部分

[1]　简单(一分钟左右可解出)

[2]　中等难度(15分钟左右可解出)

[3]　较难

[4]　难

[5]　可作为研究课题

因特网资源

网站 http://www.inference.phy.cam.ac.uk/mackay/itila 包含了以下几种资源。

1. 软件：在讲课时我所使用的教学软件、交互式软件和研究软件，是用 Perl、octave、tcl、C 和 gnuplot 编写的。还包括一些动画。

2. 本书的勘误信息。提前感谢你将它们用电子邮件发给我。

3. 本书的电子版。本书以 postscript、pdf 和 djvu 三种格式供读者在计算机上阅读。纸质图书的版权限制同样适用于这些电子书。

关于这一版本

原著是第一版的第三次印刷。在第二次印刷时，本书的设计做了一些小调整。全书的页码基本维持不变，只是在第1章、第6章和第28章中将一些段落、图和公式做了移动。所有公式、章节和习题序号都未改变。在第三次印刷时，将第8章的名字

由“相关联(Correlated)随机变量”改为“相关随机变量”,因为前者不够严谨。

致谢

非常感谢在我创作期间提供资助的机构:英国皇家协会与达尔文学院,在早年为我提供了优厚的研究基金;英国剑桥大学;美国加州大学旧金山分校 Keck 中心,我在这里度过了一个多产的学术假期;Gatsby 慈善基金的资助使我获得充分的写作自由,而幸免于陷入 Escher 阶梯的怪圈之中。

我的工作有赖于免费软件作者们的慷慨。写作本书时我所使用的是 LATEX 2ε。再三感谢 Donald Knuth 和 Leslie Lamport! 我们的计算机所运行的是 GNU/Linux 操作系统。我每天都使用 emacs、Perl 和 gnuplot 等软件。谢谢 Richard Stallman,谢谢 Linus Torvalds,感谢诸位。

许多读者都将他们对本书的意见反馈给我,在此无法一一列出,我要向他们致以诚挚的谢意。特别要感谢过去 9 年中在英国剑桥大学研读“信息论与机器学习”这门课程的所有学生和同事们。

推理研究小组的成员给了我巨大的支持,我要感谢他们在过去 10 年中所表现出的慷慨和耐心:Mark Gibbs, Michelle Povinelli, Simon Wilson, Coryn Bailer-Jones, Matthew Davey, Katriona Macphee, James Miskin, David Ward, Edward Ratzer, Seb Wills, John Barry, John Winn, Phil Cowans, Hanna Wallach, Matthew Garrett, 特别还有 Sanjoy Mahajan。还要感谢 Graeme Mitchison、Mike Cates 和 Davin Yap。

最后,向我个人心目中的英雄——我的导师们表示感激,我从他们那里学到了许多东西。他们是 Yaser Abu-Mostafa, Andrew Blake, John Bridle, Peter Cheeseman, Steve Gull, Geoff Hinton, John Hopfield, Steve Luttrell, Robert MacKay, Bob McEliece, Radford Neal, Roger Sewell 以及 John Skilling。

谨以此书献给

反军火贸易运动

www. caat. org. uk

和平不能靠武力维持。它只能通过理解来达到。

——Albert Einstein

目　录

第 1 章　信息论导论 …… 4
第 2 章　概率、熵与推理 …… 26
第 3 章　有关推理的更多内容 …… 57

第一部分　数据压缩 …… 77
第 4 章　信源编码定理 …… 79
第 5 章　符号码 …… 105
第 6 章　符号流码 …… 128
第 7 章　整数的码 …… 151

第二部分　噪声信道编码 …… 157
第 8 章　相关随机变量 …… 158
第 9 章　噪声信道上的通信 …… 166
第 10 章　噪声信道编码定理 …… 185
第 11 章　纠错码与实际信道 …… 202

第三部分　信息论中的更多专题 …… 219
第 12 章　散列码：用于有效信息检索的码 …… 221
第 13 章　二进制码 …… 235
第 14 章　存在很好的线性码 …… 263
第 15 章　有关信息论的更多习题 …… 267
第 16 章　消息传递 …… 277
第 17 章　受限无噪信道上的通信 …… 286
第 18 章　纵横字谜与密码破译 …… 301
第 19 章　为何有性？信息获取与进化 …… 312

第四部分　概率与推理 …… 325
第 20 章　一个推理任务示例：聚类 …… 328
第 21 章　基于完全枚举的精确推理 …… 339
第 22 章　最大似然与聚类 …… 346

第 23 章 有用的概率分布 …… 359
第 24 章 精确边缘化 …… 371
第 25 章 网格中的精确边缘化 …… 377
第 26 章 图中的精确边缘化 …… 389
第 27 章 拉普拉斯方法 …… 397
第 28 章 模型比较与奥卡姆剃刀原理 …… 400
第 29 章 蒙特卡罗方法 …… 416
第 30 章 有效的蒙特卡罗方法 …… 450
第 31 章 伊辛模型 …… 464
第 32 章 精确蒙特卡罗采样 …… 480
第 33 章 变参法 …… 490
第 34 章 独立元素分析与隐含变量建模 …… 506
第 35 章 有关随机推理的专题 …… 514
第 36 章 决策论 …… 522
第 37 章 贝叶斯推理与抽样理论 …… 529

第五部分 神经网络 …… 541
第 38 章 神经网络引言 …… 542
第 39 章 单神经元分类器 …… 545
第 40 章 单神经元的容量 …… 557
第 41 章 以学习作推理 …… 567
第 42 章 Hopfield 网络 …… 582
第 43 章 玻耳兹曼机 …… 600
第 44 章 多层网络的有监督学习 …… 606
第 45 章 高斯过程 …… 616
第 46 章 反卷积 …… 630

第六部分 稀疏图码 …… 637
第 47 章 低密度奇偶校验码 …… 639
第 48 章 卷积码与 Turbo 码 …… 660
第 49 章 重复累加码 …… 671
第 50 章 数字喷泉码 …… 680

第七部分 附录 …… 691
附录 A 记号 …… 692
附录 B 一些物理知识 …… 695
附录 C 一些数学知识 …… 699

英汉词汇表 …… 709
参考文献 …… 729

关于第 1 章

阅读第 1 章，要求你熟悉二项分布。若想求解本章的习题（笔者强烈建议你做一下），你还需要知道阶乘函数的 Stirling 近似：$x! \simeq x^x e^{-x}$，并能够将它应用于公式 $\binom{N}{r} = \frac{N!}{(N-r)!\,r!}$。[①] 下面会复习一下这些内容。

二项分布

例 1.1 假设抛掷一枚变形的硬币时，正面朝上的概率为 f。将这枚硬币抛 N 次，出现 r 次正面的概率分布是什么？次数 r 的均值和方差各为多少？

解 出现正面的次数服从二项分布（如图 1.1 所示）

$$P(r|f,N) = \binom{N}{r} f^r (1-f)^{N-r} \quad (1.1)$$

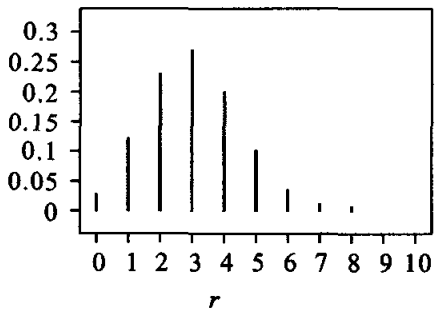


图 1.1 二项分布 $P(r|f=0.3, N=10)$

该分布的均值 $\varepsilon[r]$ 和方差 var$[r]$ 定义为

$$\varepsilon[r] \equiv \sum_{r=0}^{N} P(r|f,N)\, r \quad (1.2)$$

$$\begin{aligned} \mathrm{var}[r] &\equiv \varepsilon[(r-\varepsilon[r])^2] \quad (1.3) \\ &= \varepsilon[r^2] - (\varepsilon[r])^2 \\ &= \sum_{r=0}^{N} P(r|f,N) r^2 - (\varepsilon[r])^2 \quad (1.4) \end{aligned}$$

求均值和方差最简单的方法是利用 r 为 N 个独立随机变量之和这一事实，而并非在式(1.2)和式(1.4)中对 r 直接求和。具体地说，这 N 个独立随机变量分别是，在第一次抛掷时出现正面的次数（或为 0 或为 1），在第二次抛掷时出现正面的次数，等等。通常有

$$\begin{aligned} \varepsilon[x+y] &= \varepsilon[x] + \varepsilon[y] \quad \text{对任何随机变量 } x \text{ 和 } y \\ \mathrm{var}[x+y] &= \mathrm{var}[x] + \mathrm{var}[y] \quad \text{如果 } x \text{ 和 } y \text{ 相互独立} \end{aligned} \quad (1.5)$$

因此，r 的均值等于所有这些随机变量的均值之和，而 r 的方差等于所有这些随机变量的方差之和。在一次抛掷中出现正面的平均次数为 $f \times 1 + (1-f) \times 0 = f$，在一次抛掷中出现正面的方差为

$$[f \times 1^2 + (1-f) \times 0^2] - f^2 = f - f^2 = f(1-f) \quad (1.6)$$

于是 r 的均值和方差为

---

① 如果不熟悉这些记号，请参看附录 A。

$$\varepsilon[r] = Nf \quad 和 \quad \mathrm{var}[r] = Nf(1-f) \tag{1.7}$$

□

对 $x!$ 和 $\binom{N}{r}$ 的近似

下面通过一种非常规的途径来推导 Stirling 近似。由均值为 λ 的泊松分布(如图 1.2 所示)入手:

$$P(r|\lambda) = \mathrm{e}^{-\lambda}\frac{\lambda^r}{r!}, \quad r \in \{0,1,2,\cdots\} \tag{1.8}$$

对于大的 λ 值,该分布可以用均值和方差均为 λ 的高斯分布来很好地近似:

$$\mathrm{e}^{-\lambda}\frac{\lambda^r}{r!} \simeq \frac{1}{\sqrt{2\pi\lambda}}\mathrm{e}^{-\frac{(r-\lambda)^2}{2\lambda}} \tag{1.9}$$

至少在 $r \simeq \lambda$ 附近如此。将 $r = \lambda$ 代入上式,得

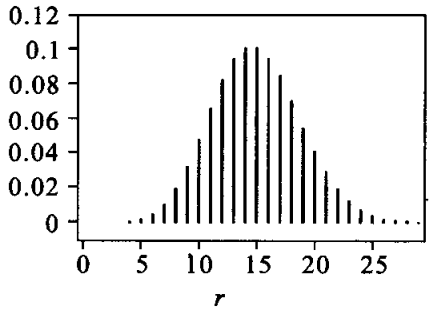


图 1.2 泊松分布 $P(r|\lambda = 15)$

$$\mathrm{e}^{-\lambda}\frac{\lambda^\lambda}{\lambda!} \simeq \frac{1}{\sqrt{2\pi\lambda}} \tag{1.10}$$

$$\Rightarrow \lambda! \simeq \lambda^\lambda \mathrm{e}^{-\lambda}\sqrt{2\pi\lambda} \tag{1.11}$$

这就是阶乘函数的 Stirling 近似:

$$x! \simeq x^x \mathrm{e}^{-x}\sqrt{2\pi x} \Leftrightarrow \ln x! \simeq x\ln x - x + \frac{1}{2}\ln 2\pi x \tag{1.12}$$

这里不仅推导出了主阶特性 $x! \simeq x^x\mathrm{e}^{-x}$,而且无需任何代价地得到了下一阶的修正项 $\sqrt{2\pi x}$。下面将 Stirling 近似应用于 $\ln\binom{N}{r}$:

$$\ln\binom{N}{r} \equiv \ln\frac{N!}{(N-r)!\,r!} \simeq (N-r)\ln\frac{N}{N-r} + r\ln\frac{N}{r} \tag{1.13}$$

因为这个公式中的所有项都是对数,所以结果可以重写成以任意数为底数的形式。我们将用“ln”表示自然对数($\log_\mathrm{e}$),用“log”表示以 2 为底的对数($\log_2$)。①

若引入*以 2 为底的熵函数*(如图 1.3 所示)

$$H_2(x) \equiv x\log\frac{1}{x} + (1-x)\log\frac{1}{(1-x)} \tag{1.14}$$

则可将近似式(1.13)重写为

$$\log\binom{N}{r} \simeq NH_2(r/N) \tag{1.15}$$

---

① 回顾 $\log_2 x = \frac{\log_\mathrm{e} x}{\log_\mathrm{e} 2}$,并注意 $\frac{\partial \log_2 x}{\partial x} = \frac{1}{\log_\mathrm{e} 2}\frac{1}{x}$。

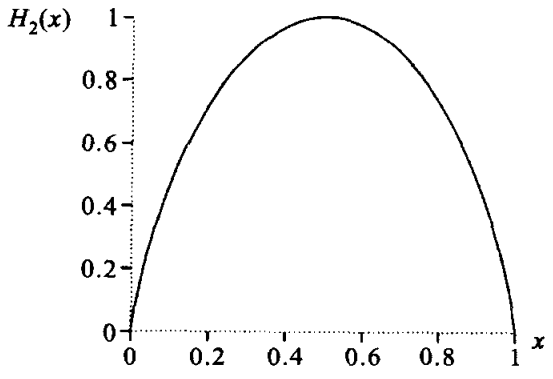


图 1.3 以 2 为底的熵函数(Binary Entropy Function)

或等价地得到

$$\binom{N}{r} \simeq 2^{NH_2(r/N)} \tag{1.16}$$

如果需要一个更精确的近似,则可以包含 Stirling 近似式(1.12)中的下一阶的项:

$$\log\binom{N}{r} \simeq NH_2(r/N) - \frac{1}{2}\log\left[2\pi N \frac{N-r}{N}\frac{r}{N}\right] \tag{1.17}$$

第 1 章

信息论导论

在某一点上精确地或者近似地复制出在另一个点上所选择的消息，这是根本的问题。

(Claude Shannon, 1948)

在本书的前半部分，我们研究如何度量信息的内容，学习如何压缩数据，并学习如何在非理想的通信信道上进行理想的通信。

首先对这一问题有一个大致的了解。

1.1 如何在非理想的噪声信道上实现理想的通信

以下是噪声通信信道的几个例子。

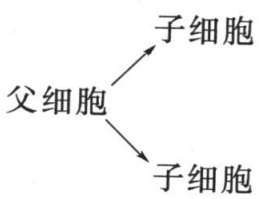
- 模拟电话线，两个调制解调器通过这根电话线进行数字信息的通信。

  调制解调器→电话线→调制解调器

- 从在木星轨道上运行的伽利略号太空船到地球的无线通信链路。

  伽利略号→无线电波→地球

- 复制细胞，其中子细胞的 DNA 包含来自父细胞的信息。



- 磁盘驱动器。

  计算机内存→磁盘驱动器→计算机内存

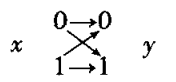
最后一个例子表明通信不必涉及从一个地方到另一个地方的信息。如果在磁盘驱动器上写入一个文件，我们将从同一位置读取它，只不过是在晚一些的时刻。

这些信道上都存在噪声。一根电话线会与其他线路发生串音；线路上的硬件会使发送信号发生失真，并在其中加入噪声。收听来自伽利略号太空船微弱发送信号的外层太空网络，会接收到来自地面和太空干扰源的背景辐射。DNA 可能遭受突变和损坏。磁盘驱动器通过将一小块磁介质校准到两个方向之一来写入一个二进制数（一个 1 或 0，也称为一个比特）。但是，这小块介质材料可能自发地改变了磁化方向，使磁盘驱动器后来可能读不出所存储的二进制数；或者背景噪声中出现一个短时

脉冲干扰，它可能导致数据读取电路所给出的是二进制数的错误值；或者由于受到相邻比特的干扰，写磁头可能一开始就没有感应出磁化。

在所有这些实例中，如果在信道上发送数据（比如一个比特串），在一定的概率下，所接收到的消息会不同于被发送的消息。我们希望得到一个理想的通信信道，即上述概率为0，或非常接近于0以至于在实际应用时无法将该值同0区分开。

现在考虑一个有噪声的磁盘驱动器，它正确发送一个比特的概率为$(1-f)$，而错误发送的概率为f。这个模型通信信道叫做二进制对称信道（图1.4）。



$$P(y=0|x=0)=1-f; \quad P(y=0|x=1)=f;$$
$$P(y=1|x=0)=f; \quad P(y=1|x=1)=1-f$$

图1.4 二进制对称信道。其中，发送符号为x，接收符号为y。噪声水平（即一个比特数值翻转的概率）为f。

作为例子，试考虑$f=0.1$，即有10%的比特数值发生翻转的情形（图1.5）。一个可用的磁盘驱动器在其使用寿命期内都不会出现比特数值翻转。如果想在十年内的每一天读/写1 GB的数据，我们要求比特误码率在10^{-15}这个数量级或更低。有两种途径可以达到这一目标。

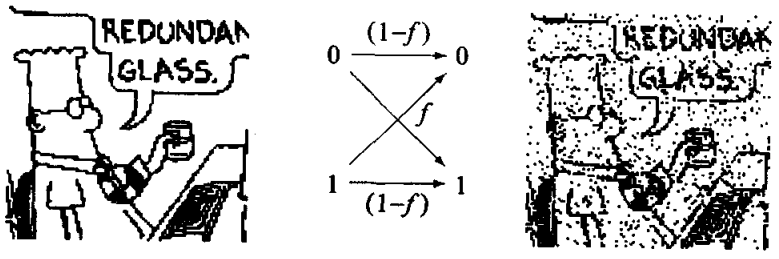


图1.5 在噪声水平$f=0.1$的二进制对称信道上所传输的一个长度为10 000的二进制数据序列。[Dilbert图像版权©1997，由United Feature Syndicate公司授权使用。]

物理级解决方案

物理级解决方案是指提高通信信道的物理特性，以降低其差错概率。可以通过如下方法来提高磁盘驱动器的性能。

1. 在电子线路中使用更可靠的元器件。
2. 从磁盘密封盒中将空气抽掉，以消除会使读磁头偏离磁道的扰动。
3. 使用一片稍大的磁介质来表示一个比特。
4. 使用高功率信号或为电子线路降温，以便降低热噪声。

这些物理方面的改进通常会提高通信信道的成本。

“系统级”解决方案

信息论与编码理论给出了另一条（更有意思的）途径：我们依原样接受给定的噪声信道，再将通信系统加在其上，以便能够检测并纠正信道所引入的差错。如图1.6

所示，我们在信道前添加了*编码器*，并在信道后添加了*译码器*。编码器通过在源消息 s 中以某种方式加入冗余信息，将它编码为*发送消息* t。信道则在发送消息上叠加噪声，产生被接收消息 r。译码器利用编码系统所引入的已知冗余信息，推断出源信号 s 和被叠加的噪声。

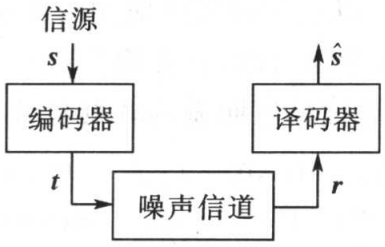


图 1.6 在噪声信道上获得可靠通信的“系统级”解决方案。编码系统在发送向量 t 中引入系统化冗余信息。译码系统利用这个已知的冗余信息，从接收向量 r 中推断出原来的源信号向量和信道所引入的噪声。

物理级解决方案要提高信道性能只能使成本越来越高，而系统级解决方案能够将噪声信道变成可靠通信信道，其唯一代价是要求编码器和译码器具有一定的计算能力。

信息论所关心的是这些系统的理论极限和潜能。“我们能达到的最佳纠错性能是什么？”

编码理论所关心的则是创建实用的编码和译码系统。

1.2 二进制对称信道的纠错码

现在考察编码与译码系统的几个例子。要在传输中加入有用的冗余信息，最简单的办法是什么呢？[为了使游戏规则更清楚，做如下约定：我们要求能检测并纠正差错；不允许重传。因此，我们只有一次机会对信息进行编码、传输和译码。]

重复码

一个直接的想法是让消息中的每一个比特都重复事先设置的次数，比如 3 次，如表 1.7 所示。我们称之为*重复码*“R_3”。

表 1.7 重复码 R_3

源序列 s	发送序列 t
0	000
1	111

假设要在一个噪声水平 $f=0.1$ 的二进制对称信道上，用这个重复码发送源消息

$$s=0010110$$

可以将信道描述为在发送向量上“叠加”一个稀疏的噪声向量 $\boldsymbol{n}$，这里的“叠加”是在模 2 算术意义下的，即满足 $1+1=0$ 的二进制代数。一个可能的噪声向量 $\boldsymbol{n}$ 和接收向量 $\boldsymbol{r}=\boldsymbol{t}+\boldsymbol{n}$ 如图 1.8 所示。

s	0	0	1	0	1	1	0
t	000	000	111	000	111	111	000
n	000	001	000	000	101	000	000
r	000	001	111	000	010	111	000

图 1.8 使用 R_3 的一个发送示例

应该如何对接收向量进行译码呢？最优算法是每次查看 3 个接收比特并采用“服从多数”的策略（算法 1.9）。

接收序列 r	似然比 $\frac{P(r\mid s=1)}{P(r\mid s=0)}$	译码序列 $\hat{s}$
000	γ^{-3}	0
001	γ^{-1}	0
010	γ^{-1}	0
100	γ^{-1}	0
101	γ^{1}	1
110	γ^{1}	1
011	γ^{1}	1
111	γ^{3}	1

算法 1.9 R_3 的“服从多数”译码算法。假设信道为二进制对称信道，表中所示还有式(1.23)所定义的似然比。其中 $\gamma\equiv(1-f)/f$。

尽管有对显而易见的事实做解释之嫌，这里还是给出这个结果的证明如下。最优译码判决（在错误判决概率最小的意义下的最优）是：对于给定的 $\boldsymbol{r}$，找出最有可能的 s 值。假设一个比特 s 被编码为 $\boldsymbol{t}(s)$ 并给出 3 个接收比特 $\boldsymbol{r}=r_1r_2r_3$，现在考虑它的译码。由贝叶斯定理，s 的*后验概率*（posterior probability）为

$$P(s\mid r_1r_2r_3)=\frac{P(r_1r_2r_3\mid s)P(s)}{P(r_1r_2r_3)} \tag{1.18}$$

可以写出两种可替换方式的后验概率如下：

$$P(s=1\mid r_1r_2r_3)=\frac{P(r_1r_2r_3\mid s=1)P(s=1)}{P(r_1r_2r_3)} \tag{1.19}$$

$$P(s=0\mid r_1r_2r_3)=\frac{P(r_1r_2r_3\mid s=0)P(s=0)}{P(r_1r_2r_3)} \tag{1.20}$$

这个后验概率由两个因素决定：*先验概率*（prior probability）$P(s)$ 和数据相关项 $P(r_1r_2r_3\mid s)$。后者称为 s 的*似然函数*（likelihood）。根据最优译码判决，如果 $P(s=0\mid$

$\boldsymbol{r}) > P(s=1|\boldsymbol{r})$，则认为 $\hat{s}=0$，否则 $\hat{s}=1$。在确定最优判决值时，不必计算归一化常数 $P(r_1r_2r_3)$。

为了确定 $P(s=0|\boldsymbol{r})$ 和 $P(s=1|\boldsymbol{r})$，必须对两个假设 $s=0$ 和 $s=1$ 以及给定 s 时 $\boldsymbol{r}$ 的概率各做一个假定。假定先验概率相等，即 $P(s=0)=P(s=1)=0.5$，于是极大化后验概率 $P(s|\boldsymbol{r})$ 就等价于极大化似然函数 $P(\boldsymbol{r}|s)$。再假定信道为噪声水平 $f<0.5$ 的二进制对称信道，从而似然函数为

$$P(\boldsymbol{r}\mid s) = P(\boldsymbol{r}\mid t(s)) = \prod_{n=1}^{N} P(r_n \mid t_n(s)) \tag{1.21}$$

其中 $N=3$ 是我们所考虑的分组中发送比特的个数，并且有

$$P(r_n|t_n) = \begin{cases} (1-f), & \text{若 } r_n = t_n \\ f, & \text{若 } r_n \neq t_n \end{cases} \tag{1.22}$$

因此这两个假设的似然比为

$$\frac{P(\boldsymbol{r}|s=1)}{P(\boldsymbol{r}|s=0)} = \prod_{n=1}^{N} \frac{P(r_n|t_n(1))}{P(r_n|t_n(0))} \tag{1.23}$$

如果 $r_n=1$，每个因子 $\frac{P(r_n|t_n(1))}{P(r_n|t_n(0))}$ 等于 $\frac{(1-f)}{f}$；如果 $r_n=0$，该因子等于 $\frac{f}{(1-f)}$。比值 $\gamma \equiv \frac{(1-f)}{f}$ 大于 1，因为 $f<0.5$，于是获胜假设是得票最多的假设，每一票都计入似然比中的因子 γ。

因此，如果假设信道是二进制对称信道，而且两个可能的源消息 0 和 1 具有相同的先验概率，那么算法 1.9 中所示的“服从多数”译码器是最优译码器。

下面将“服从多数”译码器应用于图 1.8 中的接收向量。第一组的 3 个接收比特都是 0，因此将这一 3 - 元组译码为 0。在图 1.8 中的第二个 3 - 元组里，有两个 0 和一个 1，因此将这一 3 - 元组也译码为 0——此时纠正了所发生的差错。不过，并非所有差错都能得到纠正。如果在同一个分组不幸出现了两个差错，就像图 1.8 中的第五个 3 - 元组那样，译码规则就会给出错误的答案，如图 1.10 所示。

s	0	0	1	0	1	1	0
t	000	000	111	000	111	111	000
n	000	001	000	000	101	000	000
r	000	001	111	000	010	111	000
$\hat{s}$	0	0	1	0	0	1	0
纠正的错误		*					
未检测出的错误					*		

图 1.10 译码图 1.8 中的接收向量



习题 1.2[2] 通过计算 R_3 码在具有噪声水平 f 的二进制对称信道上的差错概率，证明使用 R_3 码会降低差错概率。①

差错概率中占主导的是每组 3 个比特中两个出现数值翻转这种情况的概率，其数量级与 f^2 相同。对于 $f=0.1$ 的二进制对称信道，在译码之后，R_3 码的差错概率为每比特 $p_b \simeq 0.03$。图 1.11 所示为使用重复码在二进制对称信道上传输二进制图像的结果。

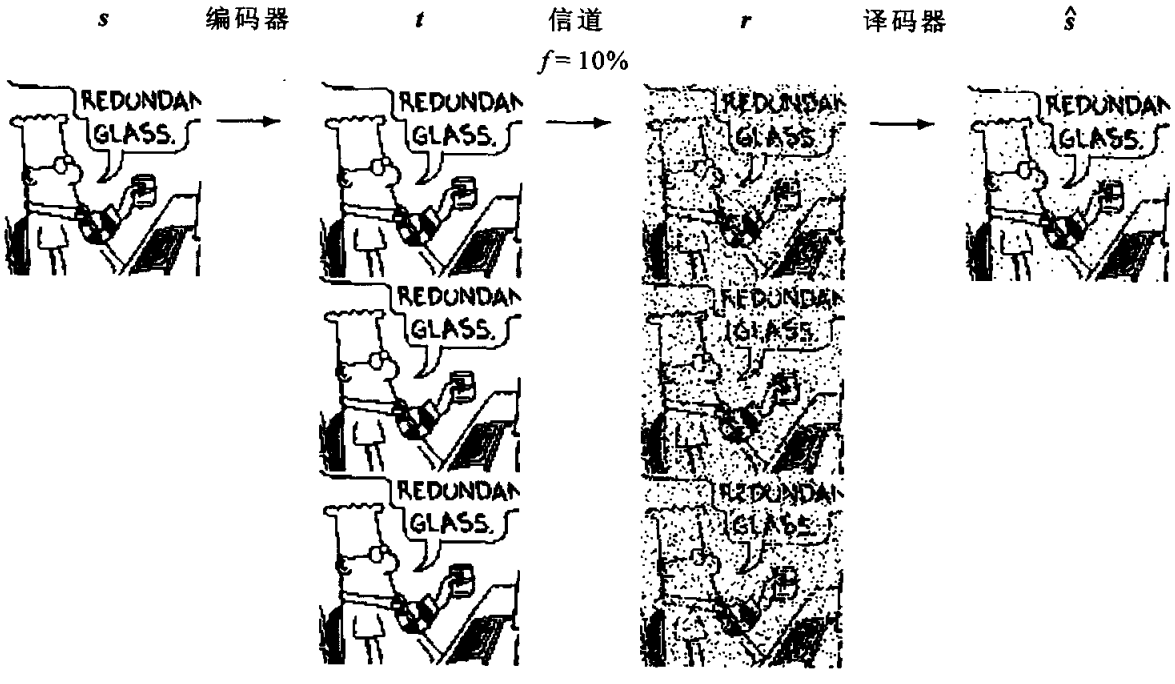


图 1.11 使用重复码和“服从多数”译码算法，在一个 $f=10\%$ 的二进制对称信道上发送 10 000个源比特。译码差错概率降至约 3%，码率降至 1/3。

因此，重复码 R_3 就如所期望的那样降低了差错概率。但是，我们同时也失去了一些东西：信息传输的速率降低为原来的三分之一。因此，如果使用重复码在电话线上传输数据，它会降低差错的频度，但也会降低通信速率。我们每次打电话都不得不付 3 倍的价钱。类似地，我们需要 3 个原来那种带噪声的千兆磁盘驱动器，才能相当于一个 $p_b=0.03$ 的千兆磁盘驱动器。

能否把差错概率降低到 10^{-15}（这是一个可售磁盘驱动器所需要的差错概率值）？通过使用具有更多重复的重复码，就可以达到更低的差错概率。



习题 1.3[2] （a）证明具有 N 次重复的重复码 R_N 的差错概率是

① 习题的评级，比如“[2]”，表明了它的难度级别：“1”级的习题最容易。页边上标有老鼠图案的习题是特别推荐的。

$$p_b = \sum_{n=(N+1)/2}^{N} \binom{N}{n} f^n (1-f)^{N-n} \tag{1.24}$$

其中 N 为奇数。

(b) 假设 $f=0.1$,和式中的哪一项最大?它比第二大的项大多少?

(c) 使用 Stirling 近似公式来近似求解最大项中的 $\binom{N}{n}$,并近似地求出具有 N 次重复的重复码的差错概率。

(d) 假设 $f=0.1$,试求需要重复多少次才能把差错概率降至 10^{-15}。[答案:约为 60。]

因此,要由 $f=0.1$ 的有噪声千兆磁盘驱动器构建一个满足可靠性要求的千兆磁盘驱动器,需要 60 个这样的有噪声磁盘驱动器。差错概率与重复码码率之间的折中如图 1.12 所示。

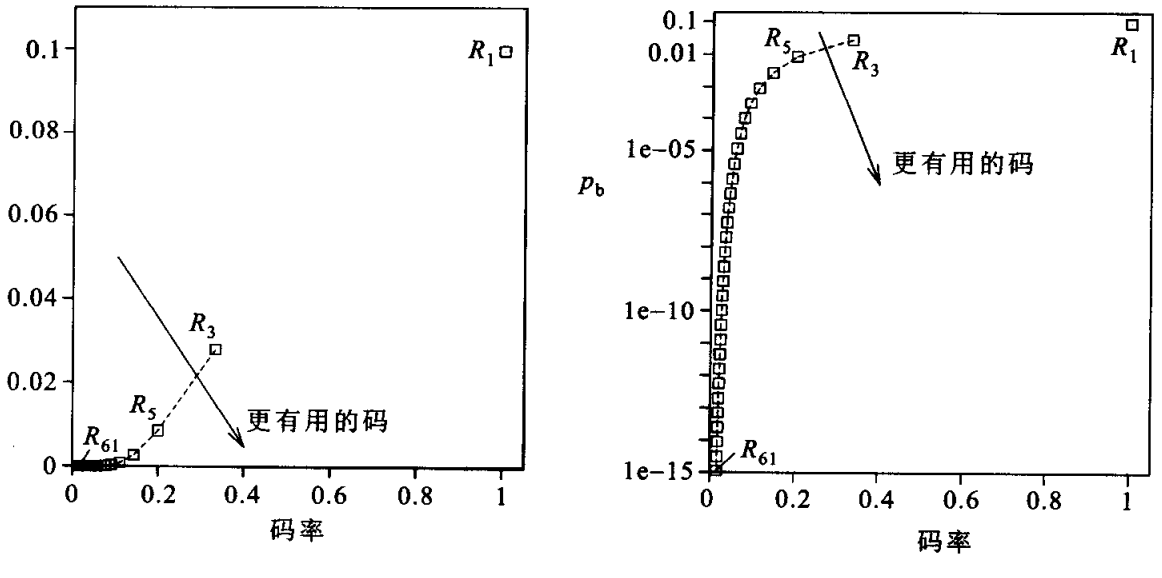


图 1.12 在 $f=0.1$ 二进制对称信道上的差错概率 p_b 与重复码的码率的关系。右半图所示为对数坐标上所表示的 p_b。我们希望有高的码率和低的 p_b。

分组码——(7,4)汉明码

我们希望所进行的通信有很小的差错概率并有较高的码率。能否改善重复码呢?如果在一组数据中加入冗余,而不是对单个比特进行编码,结果会怎样?下面来考察一种简单的分组码。

分组码是将长度为 K 的一个源比特序列 $\boldsymbol{s}$ 转换成长度为 N 比特的发送序列 $\boldsymbol{t}$ 的一条规则。在此规定 N 大于 K,以便加入冗余。在线性分组码中,多出来的 $N-K$ 比特是原有 K 比特的线性函数;这些多出来的比特就叫做奇偶校验比特。(7,4)汉明码是线性分组码的一个例子,它每收到 $K=4$ 个源比特,就发送 $N=7$ 个比特。

这种码的编码运算图形化如图 1.13 所示。我们将 7 个发送比特放入 3 个相交

的圆中。将前 4 个发送比特 $t_1t_2t_3t_4$ 设定为 4 个源比特 $s_1s_2s_3s_4$。设定奇偶校验比特 $t_5t_6t_7$ 使得每个圆中的奇偶校验为偶校验:第一个奇偶校验比特是前 3 个源比特的奇偶校验(即如果这些比特之和是偶数,则奇偶校验比特的取值为 0,否则为 1);第二个奇偶校验比特是后 3 个源比特的奇偶校验;而第三个奇偶校验比特是第一、第三和第四个源比特的奇偶校验。

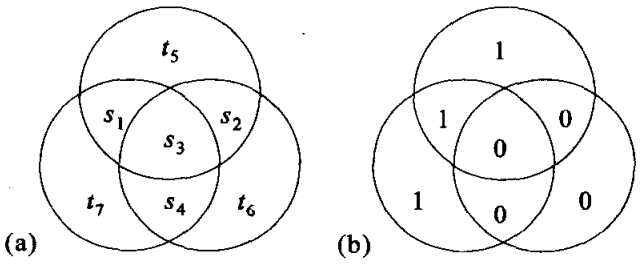


图 1.13 (7,4)汉明码编码的图形化表示

作为示例,如图 1.13(b)所示为 $s=1000$ 时的发送码字。如表 1.14 所示为 4 个源比特的 $2^4=16$ 种组合所生成的对应码字。这些码字有一种特性,每一对码字之间至少有 3 个比特不同。

表 1.14 (7,4)汉明码的 16 个码字$\{t\}$。每一对码字之间至少有 3 个比特不同。

s	t	s	t	s	t	s	t
0000	0000000	0100	0100110	1000	1000101	1100	1100011
0001	0001011	0101	0101101	1001	1001110	1101	1101000
0010	0010111	0110	0110001	1010	1010010	1110	1110100
0011	0011100	0111	0111010	1011	1011001	1111	1111111

因为汉明码是线性码,它可以用矩阵形式简洁地表示如下。要从源序列 s 得出发送码字 t,可以通过线性运算

$$t=G^{\mathrm{T}}s \tag{1.25}$$

其中矩阵 G 是该码的生成矩阵。

$$G^{\mathrm{T}}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\\1&1&1&0\\0&1&1&1\\1&0&1&1\end{bmatrix} \tag{1.26}$$

而且编码运算式(1.25)所用的是模 2 算术(1+1=0,0+1=1,等等)。

在编码运算式(1.25)中,假设 s 和 t 均为列向量。如果它们是行向量,这个公式应改写为

$$t = sG \tag{1.27}$$

其中

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{1.28}$$

我觉得使用右乘形式(1.25)比左乘形式(1.27)简便。然而,许多编码论的教材使用左乘形式(1.27~1.28)。

可以认为生成矩阵式(1.28)的行向量定义了七维二进制空间中的4个基向量。通过对这些向量进行所有可能的线性组合,可以得到这16个码字。

(7,4)汉明码的译码

当我们设计更复杂的编码器 $s \to t$ 时,对接收向量 r 进行译码的工作就不会那么简单了。记住任何比特(包括奇偶校验比特)都有可能会翻转。

如果假定信道为二进制对称信道且所有源向量等概率出现,那么最优译码器译出来的源向量 s 满足:与之对应的编码 $t(s)$ 同接收向量 r 之间所相差的比特数最少。[理由可参见似然函数(1.23)。]通过计算接收向量 r 和表1.14中16个码字中每个码字之间的距离,并选择最近的那一个,就可以解决译码问题。是否存在更有效的方法来找出最可能的源向量呢?

汉明码的校正子译码

基于编码图(图1.13),(7,4)汉明码的译码问题有一个图示解。

作为第一个示例,假设发送向量为 $t = 1000101$,而噪声使得第二个比特翻转,于是接收向量为 $r = 1000101 \oplus 0100000 = 1100101$。如图1.15(a)所示,将接收向量写入3个圆圈内,并检查每一个圆圈中的向量是否为偶校验。不是偶校验的圆圈如图1.15(b)中的虚线所示。译码的任务就是找出导致校验错误的翻转比特的最小集合。[奇偶校验错误的模式叫做校正子(Syndrome),并可以写成一个二进制向量,比如在图1.15(b)中,校正子为 $z = (1,1,0)$,因为前两个圆圈"不高兴"(校验为1)而第三个圆圈是"高兴的"(校验为0)。]

要完成译码任务,我们提出这样一个问题:是否可以找到一个唯一的比特,使之位于所有"不高兴"的圆圈之内,同时又位于所有"高兴的"圆圈之外?如果可以找到,那么那个比特的翻转就导致了所观察到的校正子。对于如图1.15(b)所示的情况,比特 r_2 位于两个"不高兴"的圆圈之内和那个"高兴的"圆圈之外;其他任何一个比特都不具备这一特点,所以 r_2 是可以解释校正子的唯一一个比特。

我们再来看几个例子。图1.15(c)所示为校验位之一 t_5 为噪声所翻转的情形。此时只有一个校验错误。只有 r_5 位于那个"不高兴"的圆圈之内和另两个"高兴的"

圆圈之外，因而就可以识别出 r_5 是能够解释校正子的唯一一个比特。

如果中间比特 r_3 被接收为翻转的，从图 1.15(d)可知所有 3 个校验都出现错误；而只有 r_3 位于所有 3 个圆圈之内，因而就认为 r_3 是那个可疑的比特。

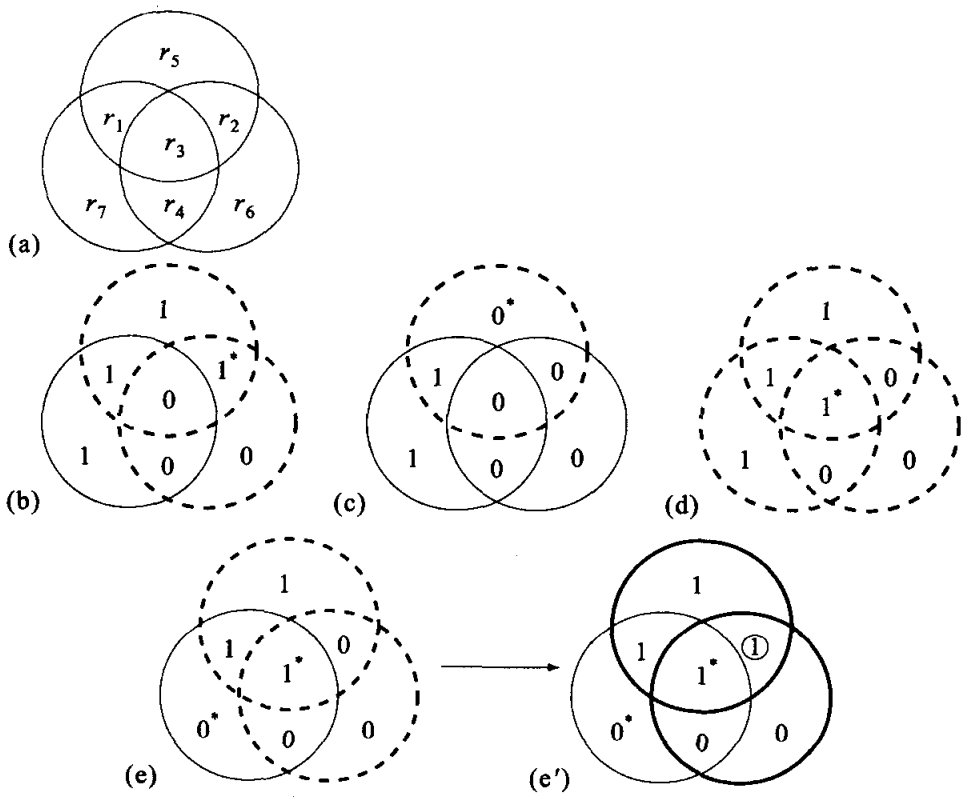


图 1.15 (7,4)汉明码译码过程的图形化表示。如图(a)所示，将接收向量写入示意图。假定发送向量如图 1.13(b)所示，图(b、c、d、e)所示分别为星号标志比特翻转时所对应的接收向量。奇偶校验错误用突出显示的虚线圆圈表示。对于每一个“校正子”(即错误和正确奇偶校验所组成的每一个模式)，7 个比特中的某一个最有可能予以解释。在例子(b)、(c)和(d)中，最可疑的比特就是翻转的比特。在例子(e)中，有两个翻转比特 s_3 和 t_7。此时，最可疑的比特是 r_2，这表示译码算法的输出，如图(e′)中的小圆圈所示。

如果试图翻转 7 个比特中的任何一个，每种情况都会有一个不同的校正子(共 7 个非零校正子，对应于每一比特)。此外只有一种校正子，即全零校正子。因此，如果信道是一个具有低噪声水平 f 的二进制对称信道，如算法 1.16 所示，最优译码器最多只回转一个比特，具体情况与校正子有关。每种校正子也可能是由其他噪声模式所引起的，不过产生相同校正子的任何其他噪声模式发生的可能性会低一些，因为这些噪声模式会涉及更多噪声事件。

校正子 z	000	001	010	011	100	101	110	111
回转该比特	无	r_7	r_6	r_4	r_5	r_1	r_2	r_3

算法 1.16 假设信道是一个具有低噪声水平 f 的二进制对称信道，最优译码器对(7,4)汉明码所采取的行动。校正子向量 z 表明依次对 r_5、r_6 和 r_7 比特进行校验检查时，每个奇偶校验是错误的(1)还是正确的(0)。

如果噪声确实导致了多于一个比特的翻转,情况又会如何呢? 如图 1.15(e)所示为两个接收比特 r_3 和 r_7 翻转的情况。校正子 110 使得我们怀疑单个比特 r_2 有问题,因此最优译码算法会将该比特翻转,从而得到一个具有 3 个错误比特的译码模式,如图 1.15(e′)所示。如果使用最优译码算法,任何 2 - 比特错误模式都会导致所译 7 - 比特向量中包含 3 个错误。

线性码译码的全视图:校正子译码

可以用矩阵来描述线性码的译码问题。前 4 个接收比特 $r_1r_2r_3r_4$ 称为 4 个源比特;接收比特 $r_5r_6r_7$ 称为源比特的校验,如生成矩阵 $\boldsymbol{G}$ 所定义。先求接收比特 $r_1r_2r_3r_4$ 的 3 个奇偶校验比特,并看它们是否和 3 个接收比特 $r_5r_6r_7$ 匹配。这两个三元组之间的(模 2)差称为接收向量的校正子。如果校正子为 0,即所有 3 个奇偶校验都是正确的,那么接收向量是一个码字,最有可能的译码就是直接读出其前 4 个比特。如果校正子非零,则这个分组的噪声序列是非零的,而且校正子会指示出最可能的错误模式。

校正子向量的计算是一个线性运算。如果定义维数为 3 ×4 的矩阵 $\boldsymbol{P}$,使得方程(1.26)中的矩阵为

$$\boldsymbol{G}^{\mathrm{T}}=\begin{bmatrix}\boldsymbol{I}_4\\ \boldsymbol{P}\end{bmatrix} \tag{1.29}$$

式中 $\boldsymbol{I}_4$ 为 4 × 4 的单位矩阵,那么校正子向量 $\boldsymbol{z}=\boldsymbol{Hr}$,其中奇偶校验矩阵 $\boldsymbol{H}=[-\boldsymbol{P}\quad \boldsymbol{I}_3]$;在模 2 算术中,$-1\equiv 1$,于是有

$$\boldsymbol{H}=[\boldsymbol{P}\quad \boldsymbol{I}_3]=\begin{bmatrix}1&1&1&0&1&0&0\\0&1&1&1&0&1&0\\1&0&1&1&0&0&1\end{bmatrix} \tag{1.30}$$

该码的所有码字 $\boldsymbol{t}=\boldsymbol{G}^{\mathrm{T}}\boldsymbol{s}$ 都满足

$$\boldsymbol{Ht}=\begin{bmatrix}0\\0\\0\end{bmatrix} \tag{1.31}$$

▷ **习题 1.4**[✓] 通过对 3 ×4 矩阵 $\boldsymbol{HG}^{\mathrm{T}}$ 求值,证明以上关系成立。

因为接收向量 $\boldsymbol{r}$ 满足 $\boldsymbol{r}=\boldsymbol{G}^{\mathrm{T}}\boldsymbol{s}+\boldsymbol{n}$,校正子译码问题就是寻找满足方程

$$\boldsymbol{Hn}=\boldsymbol{z} \tag{1.32}$$

的最可能的噪声向量 $\boldsymbol{n}$。求解这个问题的一个译码算法叫做*最大似然译码器*。在后面的章节中,还会讨论这一类译码问题。

(7,4)汉明码性质小结

可能出现的长度为 7 的各种接收向量要么是一个码字,要么和一个码字之间相差一个比特。

因为共有三种校验约束，而其中的每一种都可能被违背也可能被满足，所以共有 $2\times2\times2=8$ 个不同的校正子。可以将这 8 个校正子分为 7 个非零校正子和 1 个全零校正子。每个非零校正子对应于一种 1 - 比特错误模式，而全零校正子对应于零噪声的情形。

如果所出现的是全零校正子，最优译码器不采取任何行动，否则它会利用非零校正子到 1 - 比特错误模式的映射将可疑的那个比特位回转。

如果 4 个译码后的比特 $\hat{s}_1,\hat{s}_2,\hat{s}_3,\hat{s}_4$ 和源比特 s_1,s_2,s_3,s_4 不完全匹配，就认为出现了*译码错误*。*分组差错概率*(probability of block error)p_B 是一个分组中的一个或多个译码后的比特同相应的源比特不相匹配的概率

$$p_B=P(\hat{s}\neq s) \tag{1.33}$$

误码率(probability of bit error)p_b 是一个译码比特同相应源比特不匹配的平均概率

$$p_b=\frac{1}{K}\sum_{k=1}^{K}P(\hat{s}_k\neq s_k) \tag{1.34}$$

对于这种汉明码而言，每当噪声导致 7 - 比特数据分组中的翻转比特数多于 1 个时，就会出现一次译码错误。因此，分组差错概率是一个分组中出现两个或两个以上翻转比特的概率。和重复码 R_3 的差错概率一样，此概率的数量级也是 $O(f^2)$。但应该注意到这种汉明码通信的码率要大一些，$R=4/7$。

如图 1.17 所示为使用(7,4)汉明码在二进制对称信道上传送的二进制图像。大约 7% 的译码比特出现错误。注意这些错误是相关的：经常是 2 个或 3 个相邻的译码比特都出现了翻转。

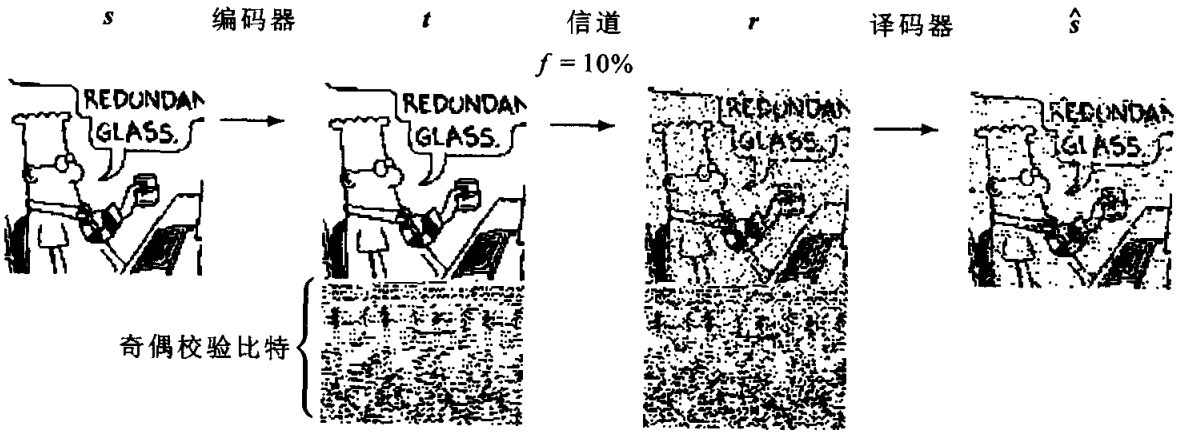


图 1.17 在 $f=10\%$ 的二进制对称信道上用(7,4)汉明码发送 10 000 个源比特。译码误码率约为 7%。



习题 1.5[1] 本题和接下来的 3 道题都是针对(7,4)汉明码的。试对下列接收到的比特串进行译码。

(a) $r=1101011$

(b) $r=0110110$

(c) $r=0100111$

(d) $r=1111111$



习题 1.6[e] (a) 计算以噪声水平 f 为自变量的(7,4)汉明码的分组差错概率 p_B,并证明按主导阶(leading order)该差错概率趋于 $21f^2$。

(b)[e] 证明按主导阶,误码率 p_b 趋于 $9f^2$。



习题 1.7[e] 试求出产生全零校正子的噪声向量(即那些使所有奇偶校验都得到满足的噪声向量)。这种噪声向量共有多少?

▷**习题 1.8**[e] 我在上文中断言:在单个分组中,若有两个或更多的比特发生翻转,则会出现分组译码错误。试证明事实确实如此。[从原则上讲,可能会存在这样一些错误模式,在译码后,它们只会损坏奇偶校验比特,而与此同时不会有源比特被错误地译码。]

编码性能小结

图 1.18 给出了重复码和汉明码的性能。该图还给出了一族叫做 BCH 码的线性分组码的性能,它们是对汉明码的推广。

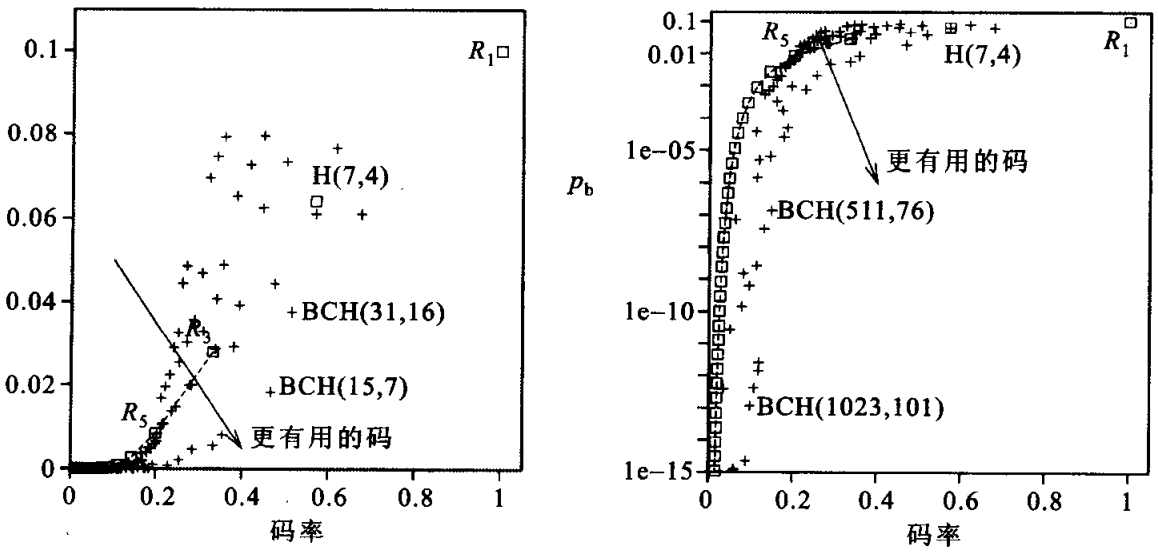


图 1.18 分组长达 1 023 的重复码、(7,4)汉明码和 BCH 码通过 $f=0.1$ 的二进制对称信道时,差错概率 p_b 与码率 R 的关系。右图所示为以对数坐标表示 p_b 的情形。

该图表明通过使用线性分组码,可以取得优于重复码的性能。不过,渐近情形(asymptotic situation)看来仍然不妙。



习题 1.9[ㄔ] 试设计一个纠错码和它的一种译码算法,估计它的差错概率,并添加到图 1.18 上。[如果你发现很难设计出比汉明码更好的码,或很难为你的码设计一个好的译码器,你也不必担心;这个习题的主要目的也就是为了让你体会到这一点。]



习题 1.10[ㄔ] 一个(7,4)汉明码可以纠正任意的单个错误;那么是否能找到一个(14,8)汉明码纠正任意两个错误?

附加选答题:该问题的答案是否取决于码是线性的还是非线性的?



习题 1.11[ㄔ] 设计一个除重复码之外的纠错码,使之能纠正大小为 N 的分组中的任意两个错误。

1.3 最佳码能达到何种性能?

在译码误码率 p_b 和码率 R 二者之间似乎要有所折中,我们希望减小前者而让后者保持一个较高的水平。如何来刻画这种折中呢? 在(R,p_b)平面上哪些点是可达到的呢? Claude Shannon(香农)在他 1948 年那篇开创性的论文中对此问题进行了论述。在这篇论文中,他开辟了信息论这个领域,并对其中的很多根本性问题进行了解答。

当时,人们普遍认为在(R,p_b)平面上的可达区域和不可达区域之间的边界是一条穿过原点(R,p_b) = (0,0)的曲线。要是果真如此,为了获得一个小到可以忽略的差错概率 p_b,必须相应地把码率减小到接近 0。也就是说"没有付出,没有收获"。

然而,香农证明了一个非同寻常的结果:可达区域和不可达区域之间的分界线和 R 轴相交于一个非零点 $R = C$,如图 1.19 所示。对于任意信道,存在某些编码,使得通信码率为非零值而差错概率 p_b 可以任意小。本书的前半部分(第一部分至第三部分)将重点讲述这一非同寻常的结果,即所谓的*噪声信道编码定理*。

例:$f = 0.1$

差错概率 p_b 可以任意小的最大通信码率称为信道的容量。具有噪声水平 f 的二进制对称信道的容量公式为

$$C(f) = 1 - H_2(f) = 1 - \left[f \log_2 \frac{1}{f} + (1-f) \log_2 \frac{1}{1-f} \right] \tag{1.35}$$

前面所讨论的噪声水平 $f = 0.1$ 的信道具有容量 $C \simeq 0.53$。现在来看看这对于有噪磁盘驱动器意味着什么。重复码 R_3 可以在这个信道上以码率 $R = 1/3$ 进行通信,并

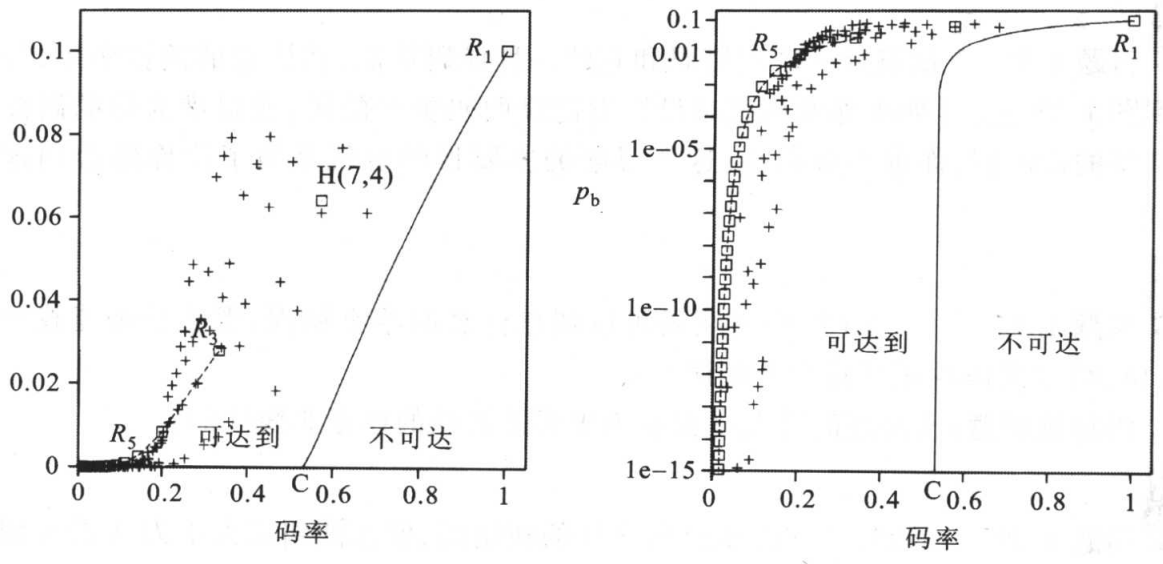


图 1.19 香农噪声信道编码定理。实线表示 $f=0.1$ 的二进制对称信道上 (R,p_b) 可达值的香农极限。不高于 $R=C$ 的任何速率都能以任意小的 p_b 达到。图中的点表示教材上的一些码(如图 1.18 所示)的性能。定义香农极限(实线)的方程是 $R=C/(1-H_2(p_b))$,其中 C 和 H_2 由方程(1.35)定义。

达到 $p_b=0.03$。因而我们知道了如何用 3 个有噪吉字节磁盘驱动器构造一个具有 $p_b=0.03$ 的吉字节磁盘驱动器。我们也知道了如何用 60 个有噪 1 GB 磁盘驱动器构造一个 $p_b \simeq 10^{-15}$ 的吉字节磁盘驱动器(习题 1.3)。现在假设香农经过这里,并注意到我们在摆弄磁盘驱动器和编码,就会说:

"你们想达到什么性能? 10^{-15}? 你并不需要 60 个磁盘驱动器,仅用两个磁盘驱动器就可以达到同样的性能(因为 1/2 小于 0.53)。而如果你想要的是 $p_b=10^{-18}$ 或 10^{-24} 或任何其他性能值,你也可以用两个磁盘驱动器达到!"

[严格地说,以上论断可能不完全正确,因为正如我们将要看到的,香农是通过研究一个分组长度递增的分组码序列来证明他的噪声信道编码定理的,而所要求的分组长度可能比 1 GB(这是我们的磁盘驱动器大小)还要大,在这种情况下,香农可能会说:"呃,你不能用这些小磁盘驱动器完成这一任务,但是如果你有两个太字节有噪磁盘驱动器,你可以用它们造出一个高质量的太字节磁盘驱动器。"]

1.4 小 结

(7,4)汉明码

在一个 7 比特的分组中包含 3 个奇偶校验比特,就有可能检测并纠正每个分组

中的任何单比特错误。

香农噪声信道编码定理

信息可以在一个噪声信道上以一个非零速率进行通信，并使得差错概率为任意小。

信息论对通信的极限和可能性这两方面都进行了论述。噪声信道编码定理表明以高于信道容量的任何速率进行可靠通信是不可能的，而以不高于此容量的任何速率进行可靠通信则是可能的。我们将在第 10 章给出噪声信道编码定理的证明。

接下来的几章将讨论如何度量信息量以及与之密切相关的一个主题——数据压缩。这将为噪声信道编码定理打下一定的基础。

1.5 进一步的习题

▷**习题 1.12**[⌀] 考虑重复码 R_9。考察这种码的一种方法是 R_3 码与 R_3 码的级联。先用 R_3 码对信源流进行编码，再对编码所得输出信息用 R_3 码进行编码。可以将这种码称作 R_3^2 码。这一思想启发我们得到另一种译码算法：使用 R_3 译码器每次译码 3 个比特，然后对第一个译码器译码后的比特再次使用 R_3 译码器进行译码。

估算这个译码器的差错概率，并将此概率与 R_9 的最佳译码器的差错概率进行比较。

R_3^2 码的级联编码器和译码器与 R_9 的相比是否具有优越性？

1.6 习题解答

习题 1.2 的解答 如果在一个 3 - 比特的分组中有两个或更多的比特出现翻转，R_3 就会出现错误译码。因此，R_3 的差错概率是下面两项之和：所有 3 个比特都出现翻转的概率 f^3 以及刚好两个比特出现翻转的概率 $3f^2(1-f)$。[如果觉得这些数学表达式并非显而易见的，请参看例 1.1：这两个表达式分别是 $P(r=3|f,N=3)$ 和 $P(r=2|f,N=3)$。]

$$p_b = p_B = 3f^2(1-f) + f^3 = 3f^2 - 2f^3 \tag{1.36}$$

对于小的 f 值，这一概率由 $3f^2$ 这一项占主导。

对于这一问题的进一步讨论，请参见习题 2.38。

习题 1.3 的解答 重复码 R_N 的差错概率中占主导的项是$\lceil N/2 \rceil$比特出现翻转

时的概率，它在N为奇数时取值为①

$$\binom{N}{\lceil N/2\rceil} f^{(N+1)/2}(1-f)^{(N-1)/2} \tag{1.37}$$

可以用以 2 为底的熵函数来近似$\binom{N}{K}$项：

$$\frac{1}{N+1}2^{NH_2(K/N)} \leqslant \binom{N}{K} \leqslant 2^{NH_2(K/N)} \Rightarrow \binom{N}{K} \simeq 2^{NH_2(K/N)} \tag{1.38}$$

其中这个近似引入一个阶次与$\sqrt{N}$相同的误差，如式(1.17)所示。因此

$$p_b = p_B \simeq 2^N (f(1-f))^{N/2} = (4f(1-f))^{N/2} \tag{1.39}$$

将该量设定为所需的数值 10^{-15}，可得 $N \simeq 2\dfrac{\log 10^{-15}}{\log 4f(1-f)} = 68$。这一答案稍微偏大，因为我们所用的近似值过大地估算了$\binom{N}{K}$，而且也没有区分$\lceil N/2\rceil$和 $N/2$。

下面给出一个略微细致一些的解答(显式计算除外)。将对$\binom{N}{K}$的近似值取到下一个阶次，有

$$\binom{N}{N/2} \simeq 2^N \frac{1}{\sqrt{2\pi N/4}} \tag{1.40}$$

这一近似可利用一个精确的 Stirling 近似式(1.12)来证明；或者考虑 $p=1/2$ 时的二项分布，并注意到

$$1 = \sum_K \binom{N}{K} 2^{-N} \simeq 2^{-N}\binom{N}{N/2} \sum_{r=-N/2}^{N/2} e^{-r^2/2\sigma^2} \simeq 2^{-N}\binom{N}{N/2}\sqrt{2\pi}\sigma \tag{1.41}$$

其中 $\sigma = \sqrt{N/4}$，由此即可推得式(1.40)。在这一项中，区分$\lceil N/2\rceil$和 $N/2$ 并不重要，因为$\binom{N}{K}$在 $K=N/2$ 处取最大值。

于是，在 N 为奇数时，这一差错概率的主阶次近似为

$$p_b \simeq \binom{N}{(N+1)/2} f^{(N+1)/2}(1-f)^{(N-1)/2} \tag{1.42}$$

$$\simeq 2^N \frac{1}{\sqrt{\pi N/2}} f[f(1-f)]^{(N-1)/2} \simeq \frac{1}{\sqrt{\pi N/8}} f[4f(1-f)]^{(N-1)/2} \tag{1.43}$$

方程 $p_b = 10^{-15}$可写做

$$(N-1)/2 \simeq \frac{\log 10^{-15} + \log \dfrac{\sqrt{\pi N/8}}{f}}{\log 4f(1-f)} \tag{1.44}$$

由此可递归求解 N。第一次递归从 $N_1 = 68$ 开始，有

① 记号：$\lceil N/2\rceil$表示大于或等于 $N/2$ 的最小整数。

$$(\hat{N}_2 - 1)/2 \simeq \frac{-15 + 1.7}{-0.44} = 29.9 \Rightarrow \hat{N}_2 \simeq 60.9 \tag{1.45}$$

可得知这一结果为稳定值，因此 $N \simeq 61$ 是使得 $p_b \simeq 10^{-15}$ 的分组长度。

习题 1.6 的解答

(a) 汉明码的分组差错概率是 6 项之和，即在一个分组中出现 2、3、4、5、6 或 7 个差错的概率。

$$p_B = \sum_{r=2}^{7} \binom{7}{r} f^r (1-f)^{7-r} \tag{1.46}$$

上式近似到主阶次，有

$$p_B \simeq \binom{7}{2} f^2 = 21 f^2 \tag{1.47}$$

(b) 汉明码的误码率要比分组差错概率小，因为一个分组差错很少会破坏译码后的分组中的所有比特。考虑最有可能出现的情况，即噪声向量权重为 2 时的结果，可得出其主阶次特性。译码器会错误地翻转第三个比特，使得改变后的接收向量(长度为 7)与发送向量之间相差 3 个比特。这意味着，如果对所有 7 个比特取平均值，一个随机选定的比特出现翻转的概率在主阶次上是分组差错概率的 3/7。现在，我们真正关心的是源比特出现翻转的概率。当噪声向量权重为 2 时，奇偶校验比特或源比特会更可能出现在这 3 个翻转比特中呢，还是被破坏的概率对于所有 7 个比特都相同呢？事实上，汉明码对于 7 个比特所提供的保护是完全对称的(假定是二进制对称信道的情况)。[这种对称性可以通过这一点进行证明：将一个奇偶校验比特和一个源比特的位置进行互换，所得的码仍是一个(7,4)汉明码；详见下文。]对于所有 7 个比特，任何一个比特被破坏的概率都是相同的。因此，误码率(对于源比特而言)只是分组差错概率的 3/7。

$$p_b \simeq \frac{3}{7} p_B \simeq 9 f^2 \tag{1.48}$$

(7,4)汉明码的对称性

为了证明(7,4)汉明码对于所有比特提供同等的保护，由如下奇偶校验矩阵入手

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{1.49}$$

如果用排列 $(t_1 t_2 t_3 t_4 t_5 t_6 t_7) \rightarrow (t_5 t_2 t_3 t_4 t_1 t_6 t_7)$ 对 7 个比特重新进行排序，就可以很容易看出 7 个发送比特中的对称性。于是可将矩阵 $\boldsymbol{H}$ 重写为

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{1.50}$$

现在，如果任取 t 所满足的两个奇偶校验约束，并将它们相加，会得到另一个奇偶校验约束。比如，第一行表明 $t_5 + t_2 + t_3 + t_1 =$ 偶数，而第二行表明 $t_2 + t_3 + t_4 + t_6 =$ 偶数，这两个约束之和是

$$t_5 + 2t_2 + 2t_3 + t_1 + t_4 + t_6 = \text{偶数} \tag{1.51}$$

可以省略项 $2t_2$ 和 $2t_3$，因为无论 t_2 和 t_3 取何值，这两项都为偶数；因此，我们推导出一个校验约束 $t_5 + t_1 + t_4 + t_6 =$ 偶数，可以将这个新导出的校验约束加入校验矩阵作为第四行。[满足 $\boldsymbol{Ht} = \mathbf{0}$ 的向量集合不会变化。]因而可定义

$$\boldsymbol{H}' = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{1.52}$$

第四行是最上面两行的(模2)和。注意第二、第三和第四行都是第一行的循环移位。如果在加入第四个冗余约束之后，去掉第一个约束，所获得的新校验矩阵 $\boldsymbol{H}''$ 为

$$\boldsymbol{H}'' = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{1.53}$$

对于所有码字，它仍然满足 $\boldsymbol{H}''\boldsymbol{t} = \mathbf{0}$，而且它看起来和式(1.50)中的起始 $\boldsymbol{H}$ 很相像，唯一差别只是将所有列往右做了一次移位，并让最右边的那一列重新出现在左边(即各列的循环排列)。

这就证实了7个比特之间的对称性。对以上程序再重复5次，可以得出同一原始码的7个不同的 $\boldsymbol{H}$ 矩阵，每一矩阵让每一比特都担任不同的角色。

还可以对这个码构造一个超级冗余的7行校验矩阵

$$\boldsymbol{H}''' = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{1.54}$$

此矩阵在如下意义下是“冗余”的：其行向量所张成的空间仅是三维的，而非七维的。

此矩阵也是一个循环矩阵。每一行都是最顶行的一个循环排列。

重复码：如果存在比特 $t_1 \cdots t_N$ 的一个排序，使得一个线性码具有一个循环校验矩阵，则称这个码为重复码。

这种码的码字也具有循环特性：一个码字的任何循环排列仍然是一个码字。

比如，比特排序如上的(7,4)汉明码，是由码字1110100和1011000以及码字0000000和1111111的7个循环移位所组成的。

重复码是代数法纠错码的一个基础。不过，在本书中不会使用它们，因为它们已

经被稀疏图码(第六部分)所取代。

习题 1.7 的解答 有 15 个非零噪声向量给出全零校正子;汉明码刚好也有 15 个非零码字。注意因为汉明码是线性码,任何两个码字之和也是一个码字。

对应于码的图

习题 1.9 的解答 在解答这个问题时,你可能会发现制造新码比获得其最优译码器要容易些。有多种设计码的方法,下面给出的只是一条可能的思路。笔者构造一个类似于(7,4)汉明码但是比它大一些的线性分组码。

许多码可以用图的形式方便地表示出来。在图 1.13 中,给出了(7,4)汉明码的一个图形化表示。那个图中每个大圈表示 4 个特定比特的奇偶校验是偶的,如果以与 4 个比特关联的一个"校验结点"取代大圈,就可得到(7,4)汉明码的一个二分图(bipartite graph)表示,如图 1.20 所示。图中的 7 个圆圈是 7 个发送比特。3 个方块是校验结点(不要同 3 个校验比特相混淆,后者是 3 个位于最外围的圆圈)。这个图是一个"二分"图,因为它的结点分为两类(比特和校验),而且只有不同类的结点之间才有边线相连。该图和码的校验矩阵(1.30)简单地相互关联:每一个校验结点对应于 $\boldsymbol{H}$ 的一行,而每个比特结点则对应于 $\boldsymbol{H}$ 的一列;对于 $\boldsymbol{H}$ 中的每一个 1,在相应的那对结点之间有一条边。

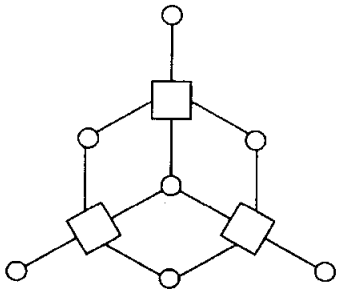


图 1.20 (7,4)汉明码的图。7 个圆圈表示比特结点,3 个方块表示校验结点。

注意到线性码和图之间的这种联系,就可以简单地利用二分图来构造线性码。例如,将 12 面体的顶点称作校验结点,在 12 面体的每条边上放一个发送比特,就可得到一个漂亮的二分图。这一构造定义一个校验矩阵,矩阵每一列权重为 2,每一行权重为 3。[一个二进制向量的权重是它所包含的元素 1 的个数。]

这个码有 $N=30$ 个比特。它看起来好像有 $M_{\text{apparent}}=20$ 个校验约束。实际上,只有 $M=19$ 个独立的约束;第 20 个约束是冗余的(也就是说,如果满足 19 个约束,则第 20 个约束自动满足);因此源比特的个数为 $K=N-M=11$。这个码是一个(30,11)码,如图 1.21 所示。

很难为这个码找到一种译码算法,但可以通过寻找它的最小权码字来估计它的差错概率。如果翻转围绕原 12 面体一面的所有比特,所有校验都会被满足;因此,这个码有 12 个权重为 5 的码字,每面有一个。因为最小权码字的权重为 5,我们说这个码的距离 $d=5$;(7,4)汉明码的距离为 3,可以纠正所有单个的比特翻转错误。一

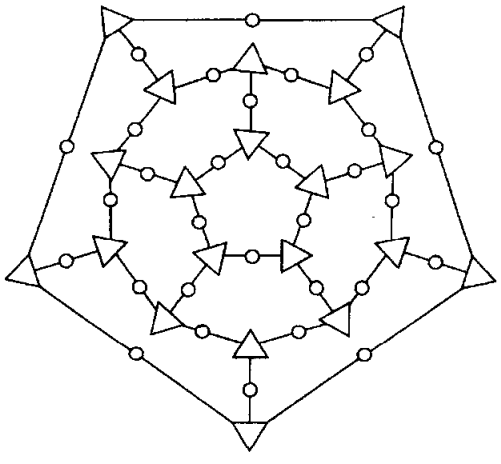


图 1.21 定义(30,11)12 面体码的图。圆圈表示 30 个发送比特,三角为 20 个校验。一个校验是冗余的。

个距离为 5 的码可以纠正所有成对的比特翻转错误,但是不能纠正某些 3 - 比特翻转错误。因此,假定二进制对称信道,至少对于低噪声水平 f,这个码的差错概率会被一个阶次为 f^3 的项所主导,可能类似于

$$12\binom{5}{3}f^3(1-f)^{27} \tag{1.55}$$

当然,也不必一定要像这个码那样,使构造码所对应的图能够在平面上表示出来;最好的线性码具有简单的图描述,但它们的图却是缠结在一起的。如图 1.22 所示的一个小型(16,4)码就是这样。

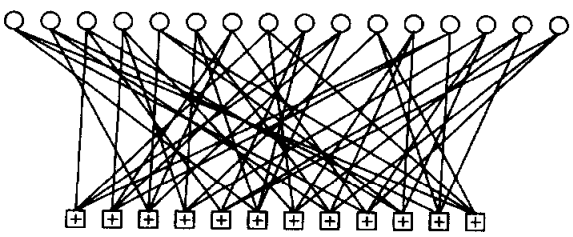


图 1.22 分组长度为 $N=16$、校验约束个数为 $M=12$、码率为 1/4 的低密度校验码(Gallager 码)的图。每个白色圆圈代表一个发送比特。每个比特出现在 $j=3$ 个约束中,约束用方块⊞表示。结点之间的边是随机配置的。(详情参见第 47 章。)

此外,也没有理由一定要坚持用线性码;确实有些非线性码(那些码字不能用类似于 $\boldsymbol{Ht}=\boldsymbol{0}$ 的线性方程来定义的码)具有很好的属性。不过,非线性码的编码和译码都更难办。

习题 1.10 的解答 首先假设将要构造的是一个线性码,并用校正子译码对它进行译码。如果有 N 个发送比特,那么权重不超过 2 的可能差错模式的个数为

$$\binom{N}{2}+\binom{N}{1}+\binom{N}{0} \tag{1.56}$$

对于 $N=14$,有 $91+14+1=106$ 种模式。现在,每个可分辨的错误模式一定会引起一个不同的校正子;而校正子由 M 比特构成,因此校正子的最大可能数目为 2^M。对

于一个(14,8)码,$M=6$,因此最多有 $2^6=64$ 个校正子。权重不超过 2 的可能差错模式的个数为 106,大于校正子数目 64,因此马上可以排除能够纠正 2 位错误的(14,8)码存在的可能性。

同样的计数论据也适用于非线性码。当译码器接收到 $\boldsymbol{r}=\boldsymbol{t}+\boldsymbol{n}$ 时,其目标是从 $\boldsymbol{r}$ 中导出 $\boldsymbol{t}$ 和 $\boldsymbol{n}$。如果发送方可以从大小为 S_t 的码中选择任何发送 $\boldsymbol{t}$,而信道可以从大小为 S_n 的一个集合中选择任意噪声向量,并且可以从接收到的比特串 $\boldsymbol{r}$(它是最多为 2^N 的可能串中的一个)中恢复出这两个选择,那么一定有

$$S_tS_n \leqslant 2^N \tag{1.57}$$

因此,对于一个能纠正 2 位错误的(N,K)码,无论是线性的还是非线性的,都满足

$$2^K\left[\binom{N}{2}+\binom{N}{1}+\binom{N}{0}\right] \leqslant 2^N \tag{1.58}$$

习题 1.11 的解答 构造能纠正多位错误的码存在着多种不同的策略,笔者强烈建议读者自己想出一种或两种方案。

如果你的方法所用的是线性码(比如一个具有 M 个校验的码),那么在预计可能需要多少个校验 M 时,记住上题给出的计数论据是很有帮助的。

能够纠正任意 2 位错误的码的例子包括前面习题解答中所给出的(30,11)12 面体码以及将在 13.13 节介绍的(15,6)五边形码。有关如何从只能纠正单个错误的码,构造出能纠正多位错误的码的一些简单思想,将在 13.7 节进一步进行介绍。

习题 1.12 的解答 R_3^2 码的差错概率在主导阶次上为

$$p_{\mathrm{b}}(\mathrm{R}_3^2) \simeq 3[p_{\mathrm{b}}(\mathrm{R}_3)]^2 = 3(3f^2)^2 + \cdots = 27f^4 + \cdots \tag{1.59}$$

但是 R_9 码的差错概率是由 5 次翻转的概率主导的,即

$$p_{\mathrm{b}}(\mathrm{R}_9) \simeq \binom{9}{5} f^5(1-f)^4 \simeq 126f^5 + \cdots \tag{1.60}$$

因此,R_3^2 译码过程是次优的,因为存在权重为 4 的噪声向量会使它发生译码错误。

不过,它具有要求较少计算资源的优势:只需记忆 3 个比特,计数不超过 3(而非计至 9)。

这个简单码阐明了一个重要的概念。在实际应用中,级联码被广泛采用,这是因为级联码允许用简单的编码和译码硬件来实现大的码。一些著名的实用码就是级联码。

第 2 章

概率、熵与推理

本章及其姊妹章第 8 章会用一定的篇幅说明所采用的记号。正如白骑士区分歌、歌名以及歌名的叫法(Carroll,1998),有时也需小心辨别随机变量、随机变量的值以及确定随机变量取某个特定值的命题。然而,无论在哪一章,笔者都会尽可能使用最简单、最友好的记号,哪怕这样难免会让思想单纯的读者感到不安。比如,如果一件事"以概率 1 为真",我就会简单地说它是"真"的。

2.1 概率与总体

总体(ensemble) X 是一个 3 - 元组$(x, \mathcal{A}_X, \mathcal{P}_X)$,这里结果 x 是随机变量的值,它在所有可能值组成的集合 $\mathcal{A}_X = \{a_1, a_2, \cdots, a_i, \cdots, a_I\}$ 上以概率 $\mathcal{P}_X = \{p_1, p_2, \cdots, p_I\}$ 取值,其中 $P(x = a_i) = p_i, p_i \geq 0$ 且 $\sum_{a_i \in \mathcal{A}_X} P(x = a_i) = 1$。

名称 $\mathcal{A}$ 是"字母表"(alphabet)的助记符。从一个英文文档中随机选取的一个字母就是总体的例子之一。这个总体如图 2.1 所示,共有 27 个可能的字母:"a" ~ "z"以及空格符" - "。

缩写 有时会使用简短些的记号。比如,$P(x = a_i)$ 可能写做 $P(a_i)$ 或 $P(x)$。

子集的概率 如果 T 是 $\mathcal{A}_X$ 的一个子集,则有

$$P(T) = P(x \in T) = \sum_{a_i \in T} P(x = a_i) \tag{2.1}$$

例如,如果定义 V 为图 2.1 中元音字母的集合,即 $V = \{\mathrm{a}, \mathrm{e}, \mathrm{i}, \mathrm{o}, \mathrm{u}\}$,则有

$$P(V) = 0.06 + 0.09 + 0.06 + 0.07 + 0.03 = 0.31 \tag{2.2}$$

联合总体 XY 是这样一种总体:它的每一个结果都是一个有序对 x, y,其中 $x \in \mathcal{A}_X = \{a_1, \cdots, a_I\}$ 和 $y \in \mathcal{A}_Y = \{b_1, \cdots, b_J\}$。

我们称 $P(x, y)$ 为 x 和 y 的联合概率。

在书写有序对时,逗号是可选的,也就是说 $xy \Leftrightarrow x, y$。

注意:联合总体 XY 中的两个变量不一定是相互独立的。

边缘概率 可以通过如下求和从联合概率 $P(x, y)$ 得到边缘概率 $P(x)$

$$P(x = a_i) \equiv \sum_{y \in \mathcal{A}_Y} P(x = a_i, y) \tag{2.3}$$

类似地,使用简短些的记号,y 的边缘概率为

$$P(y) \equiv \sum_{x \in \mathcal{A}_X} P(x, y) \tag{2.4}$$

i	a_i	p_i	
1	a	0.057 5	a
2	b	0.012 8	b
3	c	0.026 3	c
4	d	0.028 5	d
5	e	0.091 3	e
6	f	0.017 3	f
7	g	0.013 3	g
8	h	0.031 3	h
9	i	0.059 9	i
10	j	0.000 6	j
11	k	0.008 4	k
12	l	0.033 5	l
13	m	0.023 5	m
14	n	0.059 6	n
15	o	0.068 9	o
16	p	0.019 2	p
17	q	0.000 8	q
18	r	0.050 8	r
19	s	0.056 7	s
20	t	0.070 6	t
21	u	0.033 4	u
22	v	0.006 9	v
23	w	0.011 9	w
24	x	0.007 3	x
25	y	0.016 4	y
26	z	0.000 7	z
27	-	0.192 8	-



图 2.1 从一篇英文文档中随机选取的一个字母在 27 个结果上的概率分布(根据 *The Frequently Asked Questions Manual for Linux* 估算)。本图以白方块的面积来表示概率的大小。

条件概率

$$P(x=a_i|y=b_j)\equiv\frac{P(x=a_i,y=b_j)}{P(y=b_j)},\quad P(y=b_j)\neq 0 \tag{2.5}$$

[如果 $P(y=b_j)=0$,那么 $P(x=a_i|y=b_j)$ 是未定义的。]

$P(x=a_i|y=b_j)$ 读做"在给定 y 等于 b_j 的条件下,x 等于 a_i 的概率"。

例 2.1 在一篇英文文档中,两个相邻字母所组成的有序对 XY 是一个联合总体的例子。可能的结果是诸如 aa,ab,ac 和 zz 之类的有序对;在这些有序对中,也许能预计 ab 和 ac 出现的可能性要高于 aa 和 zz。如图 2.2 所示为两个相邻字母的联合概率分布的估计。

这一联合总体有一个特殊的性质:它的两个边缘概率分布 $P(x)$ 和 $P(y)$ 完全相同。这两个概率都等于如图 2.1 所示单字组(monogram)分布。

根据这个联合总体的 $P(x,y)$,可通过分别对行和列进行归一化得出条件分布 $P(y|x)$ 和 $P(x|y)$(图 2.3)。概率 $P(y|x=\mathrm{q})$ 是在给定第一个字母为"q"的前提下,第二个字母的概率分布。从图 2.3(a)可以看出,在给定第一个字母 x 为"q"的条件下,第二个字母 y 最可能的两个取值是"u"和" - "。(字母"q"后面经常是空格的原

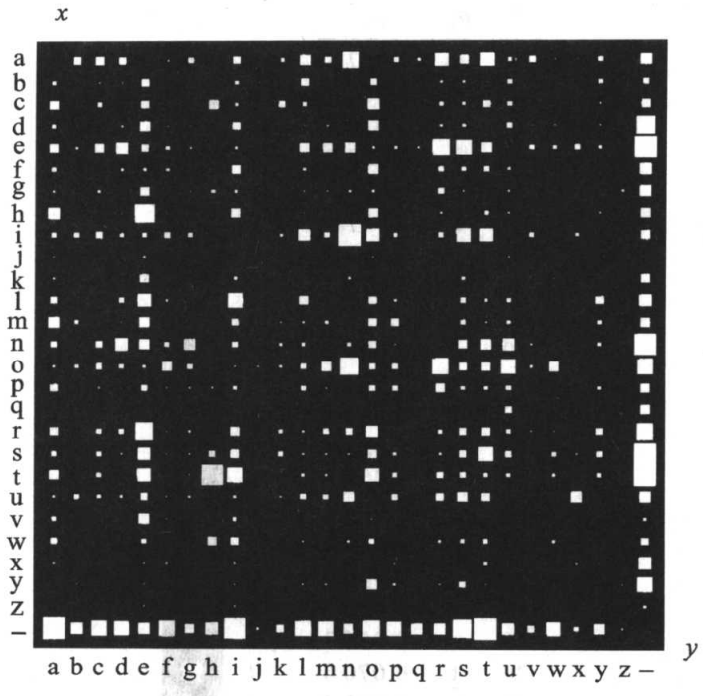


图 2.2 英文文档 *The Frequently Asked Questions Manual for Linux* 中出现的字母对在 27 × 27 个可能的双字组(bigram) xy 上的概率分布

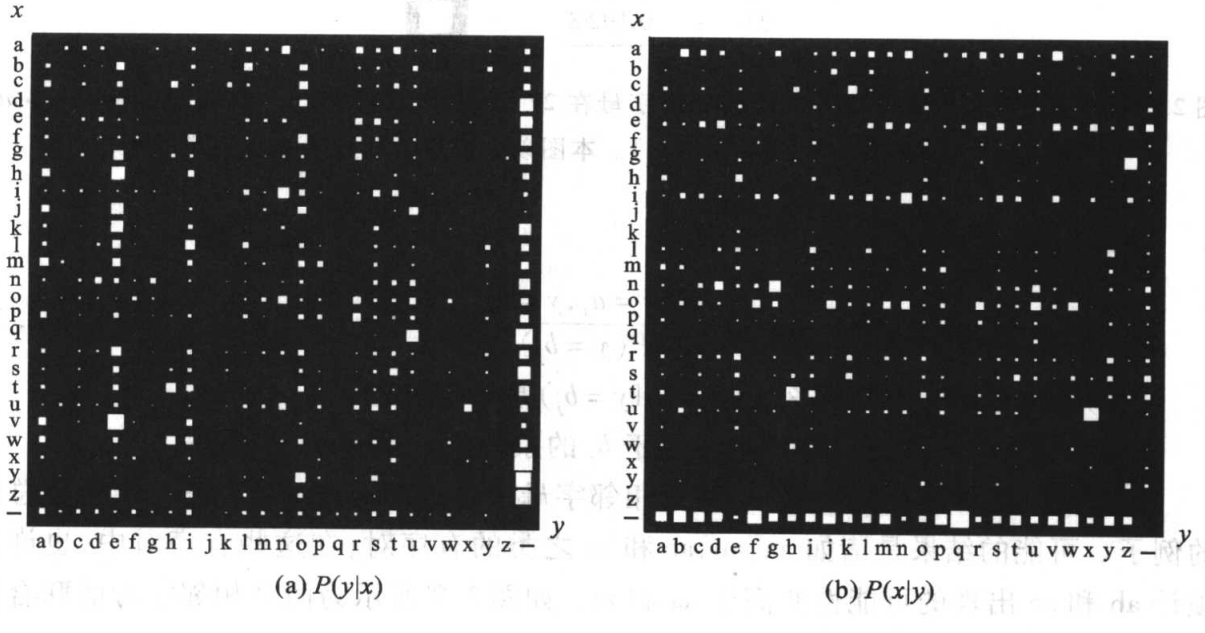


图 2.3 条件概率分布。(a) $P(y|x)$:每行表示双字组(bigram)的 xy 中给定第一个字母 x 后,第二个字母 y 的条件分布。(b) $P(x|y)$:每列表示给定第二个字母 y 后,第一个字母 x 的条件分布。

因在于源文档大量地使用了单词“FAQ”。)

概率 $P(x|y=\mathrm{u})$ 是在给定第二个字母 y 为“u”的前提下,第一个字母 x 的概率分

布。从图 2.3(b)可以看出,在给定第二个字母 y 为"u"的前提下,第一个字母 x 最可能的两个取值是"n"和"o"。

我们经常会以一组条件概率来定义一个总体,而非直接写出联合概率。以下概率论的法则有助于后续内容的理解。($\mathscr{H}$ 表示概率所基于的假设。)

乘法律 可由条件概率的定义得出

$$P(x,y|\mathscr{H}) = P(x|y,\mathscr{H})P(y|\mathscr{H}) = P(y|x,\mathscr{H})P(x|\mathscr{H}) \tag{2.6}$$

该法则又称为链式法则。

加法律 是边缘概率定义的另一种写法

$$P(x|\mathscr{H}) = \sum_y P(x,y|\mathscr{H}) \tag{2.7}$$

$$= \sum_y P(x|y,\mathscr{H})P(y|\mathscr{H}) \tag{2.8}$$

贝叶斯定理 可从乘法律得出

$$P(y|x,\mathscr{H}) = \frac{P(x|y,\mathscr{H})P(y|\mathscr{H})}{P(x|\mathscr{H})} \tag{2.9}$$

$$= \frac{P(x|y,\mathscr{H})P(y|\mathscr{H})}{\sum_{y'} P(x|y',\mathscr{H})P(y'|\mathscr{H})} \tag{2.10}$$

独立性 当且仅当

$$P(x,y) = P(x)P(y) \tag{2.11}$$

两个随机变量 X 和 Y 相互独立(有时写做 $X \perp Y$)。



习题 2.2[7] 图 2.2 所示联合总体中的随机变量 X 和 Y 相互独立吗?

笔者说过我们经常会以一组条件概率来定义一个总体。下例将阐明这一思想。

例 2.3 乔对于一种疾病做了检查。用变量 a 来表示乔的健康状态,用 b 来表示检查结果。

$$\begin{aligned} a=1, &\quad \text{乔得了这种病} \\ a=0, &\quad \text{乔没有得这种病} \end{aligned} \tag{2.12}$$

检查结果或者是阳性($b=1$)或者是阴性($b=0$)。假设检查结果是 95% 可靠的:得了这种病的人 95% 会得出阳性结果,未患这种病的人 95% 会得出阴性结果。最后一条背景信息是具有乔这个年龄和背景的人只有 1% 患这种病。

好,现在乔做了检查,并且检查结果呈阳性。那么,乔患这种病的概率是多少?

解答 先写下所提供的所有概率。检测可靠性给出了已知 a 的前提下 b 的条件概率

$$\begin{aligned} P(b=1|a=1) &= 0.95 \quad P(b=1|a=0) = 0.05 \\ P(b=0|a=1) &= 0.05 \quad P(b=0|a=0) = 0.95 \end{aligned} \tag{2.13}$$

这种疾病的流行程度告诉我们 a 的边缘概率

$$P(a=1) = 0.01 \quad P(a=0) = 0.99 \tag{2.14}$$

由边缘概率 $P(a)$ 和条件概率 $P(b|a)$,可以推得联合概率 $P(a,b) = P(a)P(b|a)$ 和

我们感兴趣的所有其他概率。比如，根据加法律，$b=1$ 的边缘概率（得出一个阳性结果的概率）是

$$P(b=1)=P(b=1|a=1)P(a=1)+P(b=1|a=0)P(a=0) \tag{2.15}$$

乔得到一个阳性的检查结果 $b=1$，并想知道她患这种病（即 $a=1$）的可信度有多大。一个普通人可能会被这样一种说法所蒙骗：检查是95%可靠的，因此乔的阳性结果意味着乔得了这种病的可能性是95%。但是，这是不对的。推理问题的正确答案可以利用贝叶斯定理得出。

$$P(a=1|b=1)=\frac{P(b=1|a=1)P(a=1)}{P(b=1|a=1)P(a=1)+P(b=1|a=0)P(a=0)} \tag{2.16}$$

$$=\frac{0.95\times0.01}{0.95\times0.01+0.05\times0.99} \tag{2.17}$$

$$=0.16 \tag{2.18}$$

因此，尽管检查结果是阳性的，乔确实患这种病的概率只有16%。 □

2.2 概率的涵义

概率可以有两种使用方式。

概率可以描述随机试验中结果出现的频度，不过要对术语“频度”和“随机”给出一个非循环的定义殊非易事。比如说，掷硬币出现正面的频度是1/2究竟是什么意思呢？如果说这个频度是在一个长的序列中正面所占的平均比重，就必须定义“平均”；很难不使用一个概率的同义词来定义“平均”！笔者不会去尝试揭开这个哲学纽结。

概率也可以更广义地用来描述未涉及随机变量的命题的置信度：比如，给定现有证据，S先生是谋杀他太太凶手的概率（他要么是，要么不是，而评定他是谋杀犯的可能性是陪审团的事）；托马斯·杰弗逊和他的一个奴仆生了一个孩子的概率；莎士比亚的戏剧为弗朗西斯·培根所创作的概率；或者，举一个当代的例子，一张特定支票上的一个特定签名是真实签名的概率。

一个普通人会乐意使用这两种方式，但是有些概率书籍将概率限于可重复随机试验的结果的频度。

不过，如果置信度满足一组称作 Cox 公理（Cox，1946）的简单的一致性规则（图2.4），它们就可以被映射为概率（如图2.4所示）。因此概率可以用来描述假设，也可以描述给定这些假设后所能做的推理。概率的法则保证：如果两个人做同样的假设，得到相同的数据，那么他们就会得到相同的结论。更广义地使用概率来量化置信则被称为贝叶斯观点。它也称作概率的主观解释，因为概率取决于假设。支持采用贝叶斯方法进行数据建模和模式识别的人不认为这种主观性是一种缺陷，因为在他们看来，

不做假设就无法进行推理。

记号 将“命题 x 的置信度”记做 $B(x)$。x 的否定(非 x)记做 $\bar{x}$。条件命题“假定命题 y 成立时 x 成立”的置信度记做 $B(x|y)$。

公理 1 可对置信度进行排序。如果 $B(x)$“大于”$B(y)$,而 $B(y)$ 又“大于”$B(z)$,则 $B(x)$“大于”$B(z)$。

[**推论** 置信度可被映射到实数上。]

公理 2 命题 x 及其否定命题 $\bar{x}$ 的置信度是相关联的。存在函数 f 使得 $B(x)=f[B(\bar{x})]$。

公理 3 联合命题 x,y(x AND y)的置信度、条件命题 $x|y$ 的置信度和命题 y 的置信度是相关联的。存在函数 g,使得 $B(x,y)=g[B(x|y),B(y)]$。

图 2.4 Cox 公理。如果一组置信度满足这些公理,那么就可将它们映射成概率,并满足 $P(\text{FALSE})=0$,$P(\text{TRUE})=1$,$0\leqslant P(x)\leqslant 1$ 以及概率法则 $P(x)=1-P(\bar{x})$ 和 $P(x,y)=P(x|y)P(y)$。

在本书中,不时地会默认贝叶斯方法是合理的,不过要提醒读者注意:这并非一个获得一致认可的观点;在 20 世纪的大部分时间里,非贝叶斯方法在统计学领域占统治地位,那里概率只能描述随机变量。这两种方法之间的一大差别在于,贝叶斯方法也采用概率来描述推理。

2.3 前向概率与反向概率

概率计算往往属于如下两大类之一:*前向概率*与*反向概率*。下面是前向概率问题的一个例子。



习题 2.4[2] 一个罐子中装着 K 个球,其中 B 个为黑色,$W=K-B$ 个为白色。弗雷德随机地从这个罐子中取出一个球,然后再放回去,如此反复 N 次。

(a) 取到黑球的次数 n_B 服从什么概率分布?

(b) n_B 的期望值是多少? n_B 的方差是多少? n_B 的标准方差是多少? 假定 $B=2$,$K=10$,对 $N=5$ 和 $N=400$ 这两种情况分别给出数值答案。

前向概率问题涉及发生模型。该模型描述的是假定要产生一些数据的过程;任务是计算取决于数据的某个数量的概率分布或期望值。下面是前向概率问题的另一个例子。



习题 2.5[2] 一个罐子中装着 K 个球,其中 B 个为黑色,$W=K-B$ 个为白色。定义比重 $f_B=B/K$。与习题 2.4 完全相同,弗雷德从罐子中取球 N 次,共 n_B 次抽得黑球,并计算数量

$$z=\frac{(n_B-f_BN)^2}{Nf_B(1-f_B)} \tag{2.19}$$

z 的期望值是多少？在 $N=5$ 和 $f_B=1/5$ 的情况下，z 的概率分布是怎样的？$z<1$ 的概率是多少？[提示：将 z 和上题中所计算的数量进行比较。]

如同前向概率问题一样，反向概率问题也涉及一个过程的发生模型，但是不用计算该过程产生的某个量的概率分布，而是计算在给定观察变量的前提下，过程中一个或多个未观察变量的条件概率。这类问题无一例外地要用到贝叶斯定理。

例 2.6 共有 11 个罐子，标记为 $u\in\{0,1,2,\cdots,10\}$，每个罐子里装着 10 个球。罐子 u 中装着 u 个黑球和 $10-u$ 个白球。弗雷德随机地选定罐子 u，并从这个罐子中有放回地取球 N 次，结果 n_B 次抽到黑球，$N-n_B$ 次抽到白球。弗雷德的朋友比尔在一旁观看。如果 $N=10$ 次后抽到 $n_B=3$ 次黑球，那么从比尔的角度来看，弗雷德所选罐子是 u 的概率是多少？（注意：比尔并不知道 u 的数值。）

解答 随机变量 u 和 n_B 的联合概率分布可写做

$$P(u,n_B|N)=P(n_B|u,N)P(u) \tag{2.20}$$

由 u 和 n_B 的联合概率，可以得到给定 n_B 的前提下，u 的条件分布

$$P(u|n_B,N)=\frac{P(u,n_B|N)}{P(n_B|N)} \tag{2.21}$$

$$=\frac{P(n_B|u,N)P(u)}{P(n_B|N)} \tag{2.22}$$

对于所有 u，u 的边缘概率为 $P(u)=\frac{1}{11}$。解答了习题 2.4，就可写出给定 u 和 N 时 n_B 的概率 $P(n_B|u,N)$（这是强烈推荐类的习题）。如果定义 $f_u\equiv u/10$，则

$$P(n_B|u,N)=\binom{N}{n_B}f_u^{n_B}(1-f_u)^{N-n_B} \tag{2.23}$$

分母 $P(n_B|N)$ 是怎样的呢？这是 n_B 的边缘概率，可利用加法律求得

$$P(n_B|N)=\sum_u P(u,n_B|N)=\sum_u P(u)P(n_B|u,N) \tag{2.24}$$

因此，给定 n_B 时 u 的条件概率为

$$P(u|n_B,N)=\frac{P(u)P(n_B|u,N)}{P(n_B|N)} \tag{2.25}$$

$$=\frac{1}{P(n_B|N)}\frac{1}{11}\binom{N}{n_B}f_u^{n_B}(1-f_u)^{N-n_B} \tag{2.26}$$

可以通过对图 2.5 中第 3 列进行归一化来得到这个条件分布，结果如图 2.6 所示。归一化常数是 n_B 的边缘概率 $P(n_B=3|N=10)=0.083$。后验概率式(2.26)对于所有 u 均成立，在端点 $u=0$ 和 $u=10$ 处（此时分别有 $f_u=0$ 和 $f_u=1$）也是如此。在 $n_B=3$ 时，$u=0$ 的后验概率等于 0，因为如果弗雷德是从罐子 0 中取球，则不可能取到黑球。$u=10$ 时的后验概率也等于 0，因为这个罐子中没有白球。对于其他假设 $u=1,u=2,\cdots,u=9$，后验概率均为非零值。 □

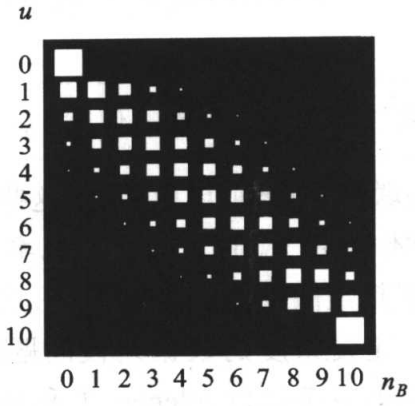
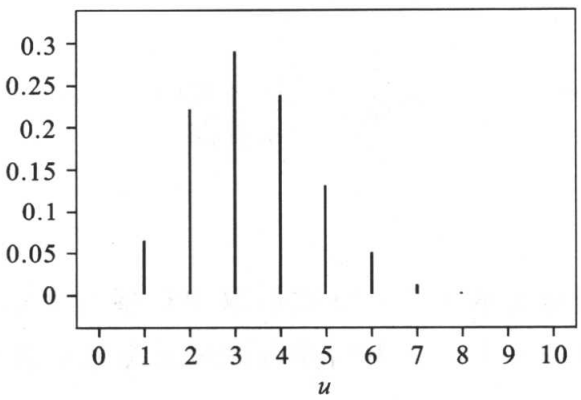


图 2.5 抽取 $N=10$ 次后，比尔和弗雷德罐子问题中 u 与 n_B 的联合概率。



u	$P(u\|n_B=3,N=10)$
0	0
1	0.063
2	0.22
3	0.29
4	0.24
5	0.13
6	0.047
7	0.009 9
8	0.000 86
9	0.000 009 6
10	0

图 2.6 在 $n_B=3$ 和 $N=10$ 时 u 的条件概率

反向概率中的术语

为了方便反向概率问题的讨论，首先为贝叶斯定理中所出现的各个概率命名。在公式(2.25)中，将边缘概率 $P(u)$ 称作 u 的先验概率，$P(n_B|u,N)$ 称作 u 的似然。特别需要注意的是，似然和概率这两个术语涵义不同。$P(n_B|u,N)$ 这个量是 n_B 和 u

的一个函数。对于固定的 u，$P(n_B|u,N)$ 定义了 n_B 的一个概率。对于固定的 n_B，$P(n_B|u,N)$ 定义了 u 的似然。

> 不要说“数据的似然”，而要说“参数的似然”。似然函数不是概率分布。

（如果要提到与似然函数相关联的数据，可以说“给定这些数据时参数的似然……”。）

条件概率 $P(u|n_B,N)$ 称作给定 n_B 时 u 的**后验概率**。归一化常数 $P(n_B|N)$ 与 u 无关，因此如果只是评估一下其他假设 u 的相对概率，这个归一化常数的具体数值并不重要。但是，在大多数有一定复杂度的数据建模问题中，这个数值变得重要起来，并有不同的名称：$P(n_B|N)$ 叫做**显著性**（evidence，亦称**证据**）或**边缘似然**。

如果未知参数记做 θ，数据记做 D，整个假设空间记做 $\mathcal{H}$，那么通用公式

$$P(\theta|D,\mathcal{H}) = \frac{P(D|\theta,\mathcal{H})P(\theta|\mathcal{H})}{P(D|\mathcal{H})} \tag{2.27}$$

写做

$$\text{后验} = \frac{\text{似然} \times \text{先验}}{\text{显著性}} \tag{2.28}$$

反向概率与预测

例 2.6（续） 再假定比尔在 $N=10$ 次抽取中观察到黑球出现了 $n_B=3$ 次，让弗雷德从同一罐子中再抽取一个球。下面抽得的球是黑球的概率是多少？［要用到图 2.6 中的后验概率。］

解答 根据加法律，有

$$P(\text{球 } N+1 \text{ 是黑球} \mid n_B,N) = \sum_u P(\text{球 } N+1 \text{ 是黑球} \mid u,n_B,N)P(u \mid n_B,N) \tag{2.29}$$

因为球是从给定的罐子中取出并放回的，所以无论 n_B 和 N 的值是什么，概率 $P(\text{球 } N+1 \text{ 是黑球}|u,n_B,N)$ 等于 $f_u=u/10$。于是有

$$P(\text{球 } N+1 \text{ 是黑球} \mid n_B,N) = \sum_u f_u P(u \mid n_B,N) \tag{2.30}$$

利用图 2.6 中给出的概率 $P(u|n_B,N)$ 的值，可得

$$P(\text{球 } N+1 \text{ 是黑球} | n_B=3,N=10) = 0.333 \qquad \square(2.31)$$

讨论 注意利用概率论所得出的这个预测与在统计学中普遍使用的预测之间的不同。统计学中所用的预测首先选取看上去最可能的假设（在这里，最可能的罐子是罐子 $u=3$），然后假设该假设为真，并做出预测（这会得出这样一个结果：下一个抽得的球是黑球的概率为 0.3）。正确的预测会通过在假设 u 的所有可能值上进行边缘化（marginalize），把不确定性考虑进去。这里的边缘化会给出一个更合适的、不那么极端的预测。

以反向概率做推理

现在来考虑下面这个有点简单科学调查风格的题目。

例 2.7 比尔将一个变形硬币抛掷了 N 次，得到一个正面/反面序列。假设这个硬币掷出正面的概率是 f_H，但 f_H 的数值未知。假设在 N 次抛掷中掷出了 n_H 次正面，那么 f_H 的概率分布是什么？（比如，N 可能是 10，而 n_H 可能是 3；或者，在抛掷了多次之后，可能有 $N=300$ 而 $n_H=29$。）在 N 次中掷出了 n_H 次正面的前提下，第 $N+1$ 次抛掷的结果为正面的概率是多少？

与例 2.6 不同，这个题目中有一个主观因素。假定在狭义的概率定义下，即“概率是随机变量的频度”，这个例子和 11 个罐子的例子有所不同。罐子 u 是一个随机变量，但通常不称这个硬币的偏倚（bias）f_H 为随机变量。它只是我们感兴趣的一个固定的未知参数。然而，难道例 2.6 和例 2.7 这两个例子从本质上看不是有些类似吗？[特别是当 $N=10$ 和 $n_H=3$ 时！]

要解答例 2.7，必须假设这个硬币的偏倚 f_H 可能是多大。f_H 的先验概率分布 $P(f_H)$①对应于 11 个罐子问题中 u 的先验概率。在那个例子中，有帮助的问题定义给出了 $P(u)$。在实际应用中，必须做些假设才能确定先验（prior）概率；这些假设是主观性的，而答案则取决于它们。对于广义模型中的其他概率，也可以做出完全相同的论断。比如，假定球是从一个罐子中相互独立地抽取的；然而是否可能因为弗雷德的取球动作并非完全随机，而造成这个序列中存在一定的相关性？事实上这是有可能的，因此我们所用的似然函数也取决于所做的假设。在实际数据建模问题中，先验概率带有主观色彩，因此也是似然函数。

现在用 $P(\ )$ 表示连续变量的概率密度、离散变量的概率以及逻辑命题的概率。一个连续变量 v 位于数值 a 和 b（这里 $b>a$）之间的概率定义为 $\int_a^b \mathrm{d}vP(v)$。$P(v)\mathrm{d}v$ 是无量纲的（dimensionless）。密度 $P(v)$ 是有量纲的，其量纲和 v 的量纲相反，这一点和离散概率不同，离散概率是无量纲的。如果看到概率密度大于 1，也不要感到奇怪。只要对于任意区间 (a,b) 都满足 $\int_a^b \mathrm{d}vP(v)<1$，这都是正常的，并非出了什么差错。

条件和联合概率密度的定义方式与条件和联合概率完全相同。

▷**习题 2.8**[2] 假设 f_H 的先验概率服从均匀分布，$P(f_H)=1$，试求解例 2.7 所提出的问题。画出 f_H 的后验分布简图，并计算以下情况下第 $N+1$ 个结果是正面的概率。

(a) $N=3, n_H=0$；

(b) $N=3, n_H=2$；

(c) $N=10, n_H=3$；

① 这里 $P(f)$ 表示概率密度而非概率分布。

(d) $N=300, n_H=29$。

在解答过程中,可能会发现以下贝塔(Beta)积分有帮助:

$$\int_0^1 \mathrm{d}p_a p_a^{F_a}(1-p_a)^{F_b} = \frac{\Gamma(F_a+1)\Gamma(F_b+1)}{\Gamma(F_a+F_b+2)} = \frac{F_a!F_b!}{(F_a+F_b+1)!} \tag{2.32}$$

你可能会觉得回顾一下例 2.6 和式(2.31)是有益的。

人们有时会把为诸如 f_H 之类的未知参数指定先验分布和对这个参数的数值做初始猜测混同起来。然而,f_H 的先验概率 $P(f_H)$ 并非像"开始,我猜 $f_H=1/2$"这样一个简单的命题。先验概率是 f_H 的概率密度,它给定了 f_H 位于区间 $(f, f+\delta f)$ 上的先验置信度。f_H 的先验概率关于 1/2 可能是对称的,因此在此先验概率下,f_H 的均值是 1/2。在这种情况下,对第一掷 x_1 的预测分布确实是

$$P(x_1=\text{正面}) = \int \mathrm{d}f_H P(f_H)P(x_1=\text{正面}\mid f_H) = \int \mathrm{d}f_H P(f_H) f_H = 1/2 \tag{2.33}$$

不过,对随后抛掷硬币的预测会取决于整个先验分布,而不只是其均值。

数据压缩与反向概率

考虑如下一项任务。

例 2.9 编写一个计算机程序,可以压缩像下面这类二进制文件。

```
0000000000000000000010010001000000100000010000000000000000000000000000000000000101000000000000011000010000000001000010000000001000000000000000000000010000000000000000010000000001100000100000001100010000000001001000000000100010000000000000000001100000000000000000000000000000100000000000000001000000000
```

这里给出的串中包含了 $n_1=29$ 个 1 和 $n_0=271$ 个 0。

直观上,利用文件的可预测性做压缩是可行的。在这种情况下,该文件的源看起来更可能产生 0 而不是 1。压缩此文件的一个数据压缩程序必须隐含地或明确地解答这样一个问题:这个文件中接下来的一个字符是 1 的概率是多少?

你是否认为这个问题和例 2.7 在本质上颇为类似?我认为如此。本书的主题之一就是数据压缩和数据建模是相同的,而且它们二者都应该采用反向概率来解决,就像例 2.6 中的罐子问题一样。例 2.9 在第 6 章有解答。

似然原理

请解答下面两个习题。

例 2.10 罐子 A 中装有 3 个球(如图 2.7 所示):一个黑的和两个白的;罐子 B 中装有 3 个球:两个黑的和一个白的。随机挑选一个罐子,并从这个罐子中抽取一个球。取出的球是黑色的。所选的罐子是罐子 A 的概率是多少?

例 2.11 罐子 A 中装有 5 个球(如图 2.8 所示):一个黑的,两个白的,一个绿的和一个粉的;罐子 B 中装有 500 个球:200 个黑的,100 个白的,50 个黄的,40 个青的,30 个赭色的,25 个绿的,25 个银色的,20 个金色的以及 10 个紫的。[罐子 A 中五分之一的球是黑的,罐子 B 中五分之二的球是黑的。]随机地挑选一个罐子,并从这个罐子中抽取一个球。取出的球是黑的。所选的罐子是罐子 A 的概率是多少?

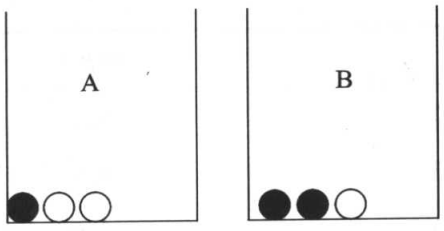


图 2.7 例 2.10 的罐子

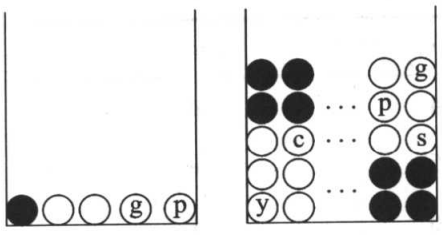


图 2.8 例 2.11 的罐子

从你的答案中,你能看出什么? 每个答案是否和每个罐子中所装球的具体情况有关?

其他可能的结果和它们的概率的详细情况是无关紧要的。关键是在不同的假设下,实际出现结果(这里,就是抽出的球是黑的)的概率。我们只需知道似然,即出现的数据的概率如何随着假设而变化。关于推理的这个简单法则称作*似然原理*。

> **似然原理** 已知参数 θ,$P(d|\theta)$,假设有一个数据 d 的生成模型,并已观察到一个特定结果 d_1,那么所有推理和预测应该只取决于函数 $P(d_1|\theta)$。

尽管这个原理很简单,许多经典的统计方法却违背了它。

2.4 熵和相关函数的定义

结果 x 的香农信息量定义为

$$h(x) = \log_2 \frac{1}{P(x)} \tag{2.34}$$

它用比特来量度。["比特"一词也用于表示一个具有 0 或 1 两种数值的变量;笔者希望上下文能够很清楚地表明用到比特时想表达的究竟是两个意思中的哪一个。]

在接下来的几章中,将确定香农信息量 $h(a_i)$ 确实是事件 $x = a_i$ 的信息量的一个自然量度。到那时候,将把这个量简短地称为"信息量"。

表 2.9 中第 4 列表示,当我们从一篇英文文档中挑出一个随机字符时的 27 种可能结果的香农信息量。结果 x = z 具有香农信息量 10.4 比特,而 x = e 具有香农信息量 3.5 比特。

表 2.9 结果 a ~ z 的香农信息量

i	a_i	p_i	$h(p_i)$
1	a	0.057 5	4.1
2	b	0.012 8	6.3
3	c	0.026 3	5.2

续表

i	a_i	p_i	$h(p_i)$
4	d	0.028 5	5.1
5	e	0.091 3	3.5
6	f	0.017 3	5.9
7	g	0.013 3	6.2
8	h	0.031 3	5.0
9	i	0.059 9	4.1
10	j	0.000 6	10.7
11	k	0.008 4	6.9
12	l	0.033 5	4.9
13	m	0.023 5	5.4
14	n	0.059 6	4.1
15	o	0.068 9	3.9
16	p	0.019 2	5.7
17	q	0.000 8	10.3
18	r	0.050 8	4.3
19	s	0.056 7	4.1
20	t	0.070 6	3.8
21	u	0.033 4	4.9
22	v	0.006 9	7.2
23	w	0.011 9	6.4
24	x	0.007 3	7.1
25	y	0.016 4	5.9
26	z	0.000 7	10.4
27	-	0.192 8	2.4

总体 X **的熵**定义为一个结果的平均香农信息量

$$H(X) \equiv \sum_{x \in \mathcal{A}_X} P(x) \log \frac{1}{P(x)} \tag{2.35}$$

对 $P(x)=0$ 约定 $0 \times \log 1/0 \equiv 0$，因为 $\lim_{\theta \to 0^+} \theta \log 1/\theta \equiv 0$。

与信息量类似，熵也用比特来度量。

在方便的时候，也可能将 $H(X)$ 写做 $H(\boldsymbol{p})$，其中 $\boldsymbol{p}$ 为向量 $(p_1, p_2, \cdots, p_I)$。X 的

熵的又叫做 X 的不确定性。

例 2.12 假设从一篇英文文档中随机选取的一个字母的概率如表 2.9 所示，它的熵约为 4.11 比特。通过在概率分布 p_i（如第 3 列所示）下对 $\log 1/p_i$（如第 4 列所示）做平均，可以得出这一数值。

下面来看看熵函数的几个性质。

- $H(X) \geqslant 0$ 当且仅当对一个 i 有 $p_i = 1$ 时取等号。
- 如果 $\boldsymbol{p}$ 为均匀的，熵最大化：

$$H(X) \leqslant \log(|\mathcal{A}_X|) \text{ 当且仅当对所有 } i \text{ 都有 } p_i = 1/|\mathcal{A}_X| \text{ 时取等号。} \tag{2.36}$$

记号 竖杠“$|\cdot|$”有两种涵义。如果 $\mathcal{A}_X$ 是一个集合，$|\mathcal{A}_X|$ 表示 $\mathcal{A}_X$ 中元素的个数；如果 x 是一个数，那么 $|x|$ 表示 x 的绝对值。

冗余量度的是 $H(X)$ 和它的最大可能值 $\log(|\mathcal{A}_X|)$ 之间的百分差。

X 的冗余是

$$1 - \frac{H(X)}{\log|\mathcal{A}_X|} \tag{2.37}$$

在本书中，不会用到“冗余”，因此笔者没有为它设定一个记号。

X 和 Y 的联合熵是

$$H(X,Y) = \sum_{xy \in \mathcal{A}_X \mathcal{A}_Y} P(x,y) \log \frac{1}{P(x,y)} \tag{2.38}$$

对于独立的随机变量，熵具有可加性：

$$H(X,Y) = H(X) + H(Y) \Leftrightarrow P(x,y) = P(x)P(y) \tag{2.39}$$

我们对信息量的定义到目前为止只适用于有限集合 $\mathcal{A}_X$ 上的离散概率分布。这些定义可以推广到无限集合，不过熵可能也是无穷大了。第 11.3 节将对连续集合上概率密度的情况进行讨论。关于熵的进一步重要定义和习题将在第 8.1 节给出。

2.5 熵的可分解性

熵函数满足在计算熵时非常有用的递归性质。为了方便起见，将对我们的记号进行扩张，以便将 $H(X)$ 写做 $H(\boldsymbol{p})$，其中 $\boldsymbol{p}$ 为对应于总体 X 的概率向量。

先通过一个例子来阐明这一性质。设想构造一个随机变量 $x \in \{0,1,2\}$：首先翻转一枚公平硬币来确定是否有 $x = 0$；如果 x 不为 0，就第二次翻转一枚公平硬币来确定 x 是 1 还是 2。x 的概率分布为

$$P(x=0) = \frac{1}{2}; P(x=1) = \frac{1}{4}; P(x=2) = \frac{1}{4} \tag{2.40}$$

X 的熵是多少呢？可以用蛮力计算

$$H(X) = 1/2\log 2 + 1/4\log 4 + 1/4\log 4 = 1.5 \tag{2.41}$$

或者也可用如下分解法来计算，此时 x 的值会逐渐地显现出来。设想首先弄清是否

有 $x=0$,然后,如果 x 不为0,再弄清它是哪个非零值。揭示是否存在 $x=0$ 意味着对一个具有概率分布 $\{1/2,1/2\}$ 的二进制变量的揭示。这一揭示过程所具有的熵为 $H(1/2,1/2)=\frac{1}{2}\log 2+\frac{1}{2}\log 2=1$ 比特。如果 x 不为0,要弄清翻转第二枚硬币的取值情况。这也是一个具有概率分布 $\{1/2,1/2\}$ 的二进制变量,其熵为1比特。不过,只需在一半的时候里经历第二次揭示过程,因此这个熵就可以写做

$$H(X)=H(1/2,1/2)+1/2H(1/2,1/2) \tag{2.42}$$

推而广之,对任意概率分布 $\boldsymbol{p}=(p_1,p_2,\cdots,p_I)$ 的熵进行观察,得

$$H(\boldsymbol{p})=H(p_1,1-p_1)+(1-p_1)H\left(\frac{p_2}{1-p_1},\frac{p_3}{1-p_1},\cdots,\frac{p_I}{1-p_1}\right) \tag{2.43}$$

令人遗憾的是,若将这个性质写成公式形式,它看上去很丑陋;不过,它却是一条简单的性质,而且也是你应该利用的一条性质。

做进一步的推广,对于任意 m,熵具有性质

$$\begin{aligned}H(\boldsymbol{p})=&H[(p_1+p_2+\cdots+p_m),(p_{m+1}+p_{m+2}+\cdots+p_I)]\\&+(p_1+\cdots+p_m)H\left(\frac{p_1}{(p_1+\cdots+p_m)},\cdots,\frac{p_m}{(p_1+\cdots+p_m)}\right)\\&+(p_{m+1}+\cdots+p_I)H\left(\frac{p_{m+1}}{(p_{m+1}+\cdots+p_I)},\cdots,\frac{p_I}{(p_{m+1}+\cdots+p_I)}\right)\end{aligned} \tag{2.44}$$

例 2.13 一个源从符号表 $\mathcal{A}=\{0,1,\cdots,9,a,b,\cdots,z\}$ 中产生一个字符 x;x 是一个数字(0,1,…,9)的概率是1/3;x 是一个元音(a,e,i,o,u)的概率是1/3;是另外的21个辅音之一的概率也是1/3。所有数字都是等概率地出现,元音和辅音也是如此。估算一下 X 的熵。

解答

$$\log 3+\frac{1}{3}(\log 10+\log 5+\log 21)=\log 3+\frac{1}{3}\log 1\,050\simeq\log 30 \text{ 比特} \qquad \square$$

2.6 Gibbs 不等式

定义在同一个字母表 $\mathcal{A}_X$ 上的两个概率分布 $P(x)$ 和 $Q(x)$ 的**相对熵或 Kullback-Leibler**[①] **散度**(divergence)为

$$D_{\mathrm{KL}}(P\parallel Q)=\sum_x P(x)\log\frac{P(x)}{Q(x)} \tag{2.45}$$

相对熵满足 Gibbs 不等式

$$D_{\mathrm{KL}}(P\parallel Q)\geqslant 0 \tag{2.46}$$

只有当 $P=Q$ 时才能取到等号。注意:一般而言,相对熵对于分布 P 和 Q 的互换是不

① Leibler 中的“ei”发音和 heist 中的“ei”相同。

对称的，即 $D_{KL}(P \| Q) \neq D_{KL}(Q \| P)$，因此尽管 D_{KL} 有时也称作“KL 距离”，严格来说它并非距离。相对熵在模式识别、神经网络和信息论中都很重要。

Gibbs 不等式可能是本书中最重要的不等式。可以利用凸性的概念来证明它以及许多其他不等式。

2.7 凸函数的 Jensen 不等式

词汇“凸 ⌣”和“凹⌢”可以读做“凸—微笑”和“凹—皱眉”。这个术语提供了有益的冗余：人们可能会忘记“凸”和“凹”是对准哪个方向，但不太可能混淆微笑和皱眉。

凸 ⌣函数 函数 $f(x)$ 在 (a,b) 上是凸 ⌣的，如果该函数的每一条弦都位于该函数之上，如图 2.10 所示；也就是说，对于所有 $x_1, x_2 \in (a,b)$ 和 $0 \leqslant \lambda \leqslant 1$，都有

$$f(\lambda x_1 + (1-\lambda)x_2) \leqslant \lambda f(x_1) + (1-\lambda) f(x_2) \tag{2.47}$$

如果对于所有 $x_1, x_2 \in (a,b)$，上式中的等号只在 $\lambda = 0$ 和 $\lambda = 1$ 处成立，则函数 f 是严格凸 ⌣的。

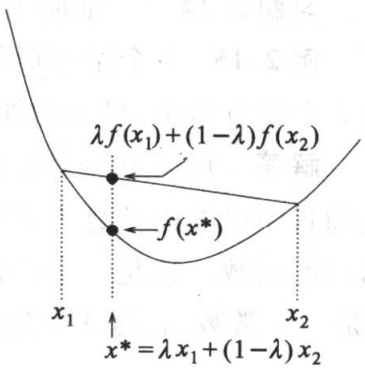


图 2.10 凸性的定义

类似定义适用于凹⌢和严格凹⌢函数。

下面是一些严格凸 ⌣函数的例子（如图 2.11 所示）。

- 对于所有 x, x^2, e^x 和 e^{-x}；
- 对于所有 $x > 0$，$\log(1/x)$ 和 $x\log x$。

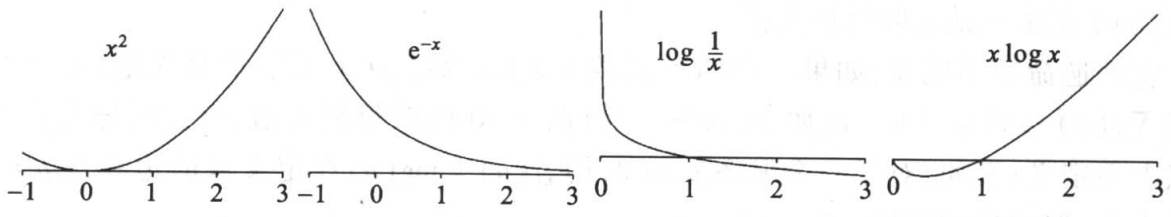


图 2.11 凸 ⌣函数

Jensen 不等式 如果 f 是一个凸 ⌣函数，x 是一个随机变量，则有

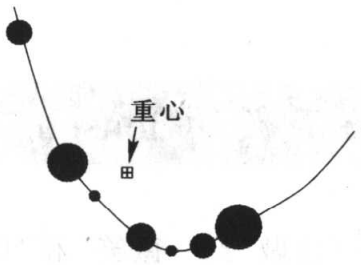
$$\mathscr{E}[f(x)] \geqslant f(\mathscr{E}[x]) \tag{2.48}$$

其中 $\mathscr{E}$ 表示期望值。如果 f 是一个严格凸 ⌣函数且 $\mathscr{E}[f(x)] = f(\mathscr{E}[x])$，那么这个随机变量 x 是一个常数。

对凹⌢函数也有 Jensen 不等式，只是不等式的方向要反过来。

Jensen 不等式的物理意义可表述如下：

如果把一组具有质量 p_i 的质点放置在凸 $\smile$ 函数曲线 $f(x)$ 的 $(x_i, f(x_i))$ 位置上，那么这些质点的重心，即 $(\mathcal{E}[x], \mathcal{E}[f(x)])$ 会位于曲线之上。



如果这不能让你信服，你不妨做做下面这道习题。

习题 2.14[21] 证明 Jensen 不等式成立。

例 2.15 3 个正方形的平均面积 $\bar{A} = 100\ \mathrm{m}^2$。它们的平均边长为 $\bar{l} = 10\ \mathrm{m}$。在这 3 个正方形中，最大的那个正方形的大小怎样？[利用 Jensen 不等式。]

解答 设 x 为一个正方形的边长，并设 x 取长度 l_1, l_2, l_3 的概率为 1/3, 1/3, 1/3。从题目中已知的信息为 $\mathcal{E}[x] = 10$ 和 $\mathcal{E}[f(x)] = 100$，其中 $f(x) = x^2$ 是将边长映射为面积的函数。这是一个严格凸 $\smile$ 函数。注意到等式 $\mathcal{E}[f(x)] = f(\mathcal{E}[x])$ 成立，因此 x 是一个常数，且这 3 个边长必须都相等。最大正方形的面积为 $100\ \mathrm{m}^2$。 □

凸性和凹性也和最大化有关

如果 $f(\boldsymbol{x})$ 是凹 $\frown$ 且存在一点使得

$$\frac{\partial f}{\partial x_k} = 0, \text{对所有 } k \text{ 成立} \tag{2.49}$$

那么 $f(\boldsymbol{x})$ 在这一点上取得最大值。

它的逆命题不成立：如果一个凹 $\frown$ 函数 $f(\boldsymbol{x})$ 在某个点 $\boldsymbol{x}$ 上取得最大值，此处的梯度 $\nabla f(\boldsymbol{x})$ 未必等于 0。比如，$f(x) = -|x|$ 在 $x = 0$ 处取得最大值，但它的导数在这一点上无定义；又如，对于一个概率 $p \in (0,1)$，$f(p) = \log(p)$ 在定义域的边界点 $p = 1$ 上最大化，但此处的梯度 $\mathrm{d}f(p)/\mathrm{d}p = 1$。

2.8 习 题

随机变量之和



习题 2.16[21] (a) 掷面上标着 1, …, 6 的两个普通骰子。两个数值之和的概率

分布是什么？两个数值之差的绝对值的概率分布是什么？①

（b）掷 100 个普通骰子。粗略地说，这些数值之和的概率分布是什么？画出概率分布的草图，并估算其均值和标准差。

（c）怎样用数字{0,1,2,3,4,5,6}来标注上述两个骰子，使得所掷得的两个数值之和服从整数 1～12 的均匀概率分布。

（d）是否存在对 100 个骰子进行标记的一种方式，使得所掷得的数值之和服从均匀概率分布？

推理问题



习题 2.17[2] 如果 $q=1-p$ 且 $a=\ln p/q$，证明

$$p=\frac{1}{1+\exp(-a)} \tag{2.50}$$

画出该函数的草图，并找出它和双曲正切函数 $\tanh(u)=\frac{e^{u}-e^{-u}}{e^{u}+e^{-u}}$ 之间的关系。

熟练地使用以 2 为底的对数也很有用。如果 $b=\log_2 p/q$，以 p 为自变量的函数 b 是怎样的？

▷ **习题 2.18**[2] 令 x 和 y 为相关的随机变量，其中 x 是一个取值于 $\mathcal{A}_X=\{0,1\}$ 的二进制变量。运用贝叶斯定理证明给定 y 时 x 的对数后验概率比为

$$\log\frac{P(x=1|y)}{P(x=0|y)}=\log\frac{P(y|x=1)}{P(y|x=0)}+\log\frac{P(x=1)}{P(x=0)} \tag{2.51}$$

▷ **习题 2.19**[2] 令 x、d_1 和 d_2 为随机变量，且在给定二进制变量 x 时 d_1 和 d_2 是条件独立的。运用贝叶斯定理证明已知 $\{d_i\}$ 时 x 的后验概率比为

$$\frac{P(x=1|\{d_i\})}{P(x=0|\{d_i\})}=\frac{P(d_1|x=1)}{P(d_1|x=0)}\frac{P(d_2|x=1)}{P(d_2|x=0)}\frac{P(x=1)}{P(x=0)} \tag{2.52}$$

高维空间中的生存状态

在高维空间中，概率分布和体积会有一些出人意料的性质。



习题 2.20[2] 考虑 N 维实空间中一个半径为 r 的球体。试证明半径为 $r-\varepsilon\sim r$（其中 $0<\varepsilon<r$）的球面的球体积所占比重为

$$f=1-\left(1-\frac{\varepsilon}{r}\right)^{N} \tag{2.53}$$

---

① 本练习的目的是帮助读者对中心极限定理进行思考。该定理说：如果独立随机变量 $x_1,x_2,\cdots,x_N$ 都具有均值 μ_n 和有限方差 σ_n^2，那么在大 N 的极限下，和数 $\sum_n x_n$ 的分布趋于正态（高斯）分布，其均值为 $\sum_n \mu_n$，方差为 $\sum_n \sigma_n^2$。

对于 $N=2, N=10$ 和 $N=1\,000$ 三种维度，分别对下面两种情况计算 f 值。

(a) $\varepsilon/r=0.01$；

(b) $\varepsilon/r=0.5$。

隐含的意思：当 N 很大时，均匀分布在 N 维球体中的点很可能出现在靠近表面的一个薄层里。

期望与熵

你也许熟悉如何计算关于 x 的函数的期望值

$$\mathcal{E}[f(x)] = \langle f(x) \rangle = \sum_x P(x) f(x) \tag{2.54}$$

你可能不是很习惯对函数 $f(x)$ 与概率 $P(x)$ 有关的这种情况计算期望值。下面几个例子就是针对这种情况的。



习题 2.21[1] 设 $p_a=0.1, p_b=0.2, p_c=0.7$，并设 $f(a)=10, f(b)=5, f(c)=10/7$。请问 $\mathcal{E}[f(x)]$ 是多少？ $\mathcal{E}[1/P(x)]$ 是多少？



习题 2.22[2] 对于任意总体，$\mathcal{E}[1/P(x)]$ 是多少？

▷ **习题 2.23**[1] 设 $p_a=0.1, p_b=0.2, p_c=0.7$，并设 $g(a)=0, g(b)=1, g(c)=0$。请问 $\mathcal{E}[g(x)]$ 是多少？

▷ **习题 2.24**[1] 设 $p_a=0.1, p_b=0.2, p_c=0.7$。请问 $P(x)\in[0.15, 0.5]$ 的概率是多少？概率

$$P\left(\left|\log\frac{P(x)}{0.2}\right|>0.05\right)$$

是多少？



习题 2.25[2] 证明：$H(X)\leqslant\log(|\mathcal{A}_X|)$，并且当且仅当对所有 i 都有 $p_i=1/|\mathcal{A}_X|$ 时取等号。($|\mathcal{A}_X|$ 表示集合 $\mathcal{A}_X$ 中元素的个数。)[提示：使用 Jensen 不等式(2.48)；如果开始试图用 Jensen 不等式行不通，请记住 Jensen 不等式涉及随机变量和函数，你在选择它们时有很大的自由度；想想所选函数 f 是凸的还是凹的。]

▷ **习题 2.26**[2] 证明相对熵(式(2.45))满足 Gibbs 不等式 $D_{\mathrm{KL}}(P\|Q)\geqslant 0$，当 $P=Q$ 时取等号。

▷ **习题 2.27**[2] 证明熵确实如式(2.43)～(2.44)所描述的那样是可分解的。

▷ **习题 2.28**[2] 按如下方式选择随机变量 $x\in\{0,1,2,3\}$：抛掷一个偏倚为 f 的变形硬币来确定结果是在 $\{0,1\}$ 中还是 $\{2,3\}$ 中；然后根据情况分别抛掷偏倚为 g(第二枚)或 h(第三枚)的变形硬币。写出 x 的概率分布。利用熵的可分解性(式(2.44))求 X 的熵。[如果使用二进制熵函数 $H_2(x)$，而非显式地写出 4 - 项熵，注意所得表

达式是否变得简洁。]求出 $H(X)$ 关于 f 的导数。[提示:$\mathrm{d}H_2(x)/\mathrm{d}x = \log((1-x)/x)$。]

▷**习题 2.29**[2] 重复抛掷一枚无偏倚的硬币,直至掷出一个正面。抛掷次数这一随机变量 $x \in \{1,2,3,\cdots\}$ 的熵是多少?对于概率为 f 的偏倚硬币重复以上计算。[提示:用直接法和熵的可分解性(式(2.43))求解这个问题。]

2.9 进一步的习题

前向概率

▷**习题 2.30**[1] 一个罐子装有 w 个白球和 b 个黑球。先后从罐子中取出两个球而不放回罐中。试证明第一个球是白色的概率等于第二个球是白色的概率。

▷**习题 2.31**[2] 将一个直径为 a 的圆形硬币投在小格面积为 $b \times b$ 的方形栅格上($a < b$)。该硬币完全落在一个方格内的概率是多少?[答案:$(1-a/b)^2$]

▷**习题 2.32**[2] **Buffon 针**。将一根长度为 a 的针投在一个画有间隔为 b 的平行线的平面上。这根针会和一条线相交的概率是多少?[答案:如果 $a < b$,此概率为 $2a/(\pi b)$。][推广的 Buffon 针:平均而言,一个长度为 A 的随机曲线会和这组平行线相交 $2A/(\pi b)$ 次。]

习题 2.33[2] 在长度为 1 的线段上随机选取两个点。这两个点所分成的 3 条短线段能构成一个三角形的概率是多少?

习题 2.34[2] 抛掷一枚无偏倚的硬币直至掷出一个正面。预计掷出的反面和正面的次数各是多少?弗雷德不知道这枚硬币是无偏倚的,于是他就用公式 $\hat{f} \equiv h/(h+t)$ 来估计偏倚,其中 h 和 t 分别为投出的正面和反面的次数。计算 $\hat{f}$ 的概率分布,并画出其草图。

注意:这是一个前向概率问题,一个抽样理论问题,而非推理问题。不要使用贝叶斯定理。



习题 2.35[2] 弗雷德每秒掷一次一枚无偏倚的 6 边骰子,并记下掷出第 6 点的情形。

(a) 从一个第 6 点到下一个第 6 点要抛掷的平均次数是多少?

(b) 在两次抛掷之间,时钟敲了一下。到下一个第 6 点所需抛掷的平均次数是多少?

(c) 现在再从时钟敲响的时刻回顾。碰到上一次时钟敲响之前所抛掷的平均次数是多少?

(d) 从时钟敲响前掷出的那个第 6 点到下一个第 6 点所需抛掷的平均次数是

多少?

(e) 请问(d)的答案和(a)的答案是否不同? 试解释之。

本题也可以这样描述:弗雷德在 Poissonville 的一个汽车站候车。在该车站,汽车独立地随机到达(泊松过程),平均每 6 分钟来一辆车。在弗雷德到达车站后,他等到下一班车的平均候车时间是多少? [6 分钟。]弗雷德刚错过的那班车和他赶上的这班车之间的平均时间是多少? [12 分钟。]试解释这个明显的矛盾。在 Clockville 这个地方,汽车的间隔时间恰好为 6 分钟,请注意此时的情况和前面的不同。可以验证,在 Clockville 汽车站的平均候车时间是 3 分钟,而刚错过的那班车和下一班车之间的时间间隔恰好为 6 分钟。

条件概率

▷ **习题 2.36**[2] 假设弗雷德告诉你,他有两个兄弟:阿尔夫和鲍勃。

弗雷德比鲍勃年龄大的概率是多少?

如果弗雷德告诉你他比阿尔夫年龄大,那么弗雷德比鲍勃年龄大的概率是多少? (也就是说,在 $F>A$ 的前提下,$F>B$ 的条件概率是多少?)

▷ **习题 2.37**[2] 一个岛上的居民三分之一的时候讲真话。他们撒谎的概率为 2/3。

有一次,一个居民做了一段陈述之后,你问另一个居民:"那个陈述是真的吗?"得到的回答为"是真的"。

那么这个陈述确实为真的概率是多少?

▷ **习题 2.38**[2] 计算重复码 R_3 在二进制对称信道的假设下的差错概率有两种方法(在习题 1.2 做过一次),对这两种方法进行比较,并证实它们给出的结果是相同的。

二项分布方法 将所有 3 个比特都翻转的概率加上正好有两个比特出现翻转的概率。

加法律方法 运用加法律,计算 $\boldsymbol{r}$ 取 8 个可能值中每一个的边缘概率 $P(\boldsymbol{r})$。[$P(\boldsymbol{r})=\sum_s P(s)P(\boldsymbol{r}|s)$。]然后对 8 个 $\boldsymbol{r}$ 值中的每一个计算 s 的后验概率。① [事实上,根据对称性,只需考虑 $\boldsymbol{r}=(000)$ 和 $\boldsymbol{r}=(001)$ 这两种情形作为范例。]注意:在确定某些推断出的比特时,把握要大一些。由后验概率 $P(s|\boldsymbol{r})$,可得各种情形的差错概率 $P(\text{差错}|\boldsymbol{r})$,即最可能的假设不正确的概率。运用如下加法律计算平均差错概率。

$$P(\text{差错})=\sum_{\boldsymbol{r}} P(\boldsymbol{r})P(\text{差错}\mid \boldsymbol{r}) \tag{2.55}$$

▷ **习题 2.39**[2,p87] 在英语中,第 n 个最可能出现的单词的频度 p_n 可以大致地近似为

$$p_n \simeq \begin{cases} \dfrac{0.1}{n}, & n\in 1,\cdots,12\ 367 \\ 0, & n>12\ 367 \end{cases} \tag{2.56}$$

[这一非凡的 $1/n$ 法则叫做 Zipf 法则,它适用于许多语言的单词频度估计(Zipf,

① 式(1.18)给出了在已知接收向量 $\boldsymbol{r}$ 的前提下,输入 s 的后验概率。

1949)。]如果假设英语文章是根据这一分布随机挑选单词构成的,那么英语(每个单词)的熵是多少?[该计算出现在 C. E. 香农的论文"Prediction and entropy of printed English"(*Bell Syst. Tech. J.* 30, pp. 50—64 (1950))中。不过令人费解的是,在这篇论文中,这位伟人做数值计算时出了很多错。]

2.10 习题解答

习题 2.2 的解答 不,它们不是独立的。否则,不管 x 为何值,所有条件分布 $P(y|x)$ 都会是关于 y 的同一个函数(参见图 2.3)。

习题 2.4 的解答 定义分式 $f_B = B/K$。

(a) 黑球的数目服从二项分布

$$P(n_B|f_B,N) = \binom{N}{n_B} f_B^{n_B}(1-f_B)^{N-n_B} \tag{2.57}$$

(b) 这个分布的均值和方差为

$$\mathcal{E}[n_B] = Nf_B \tag{2.58}$$

$$\mathrm{var}[n_B] = Nf_B(1-f_B) \tag{2.59}$$

这些结果是在例 1.1 中推导出来的。n_B 的标准差为 $\sqrt{\mathrm{var}[n_B]} = \sqrt{Nf_B(1-f_B)}$。

当 $B/K = 1/5$ 和 $N = 5$ 时,n_B 的期望和方差为 1 和 4/5。标准差为 0.89。

当 $B/K = 1/5$ 和 $N = 400$ 时,n_B 的期望和方差为 80 和 64。标准差为 8。

习题 2.5 的解答 可看出下列量

$$z = \frac{(n_B - f_B N)^2}{Nf_B(1-f_B)}$$

的分子为 $(n_B - \mathcal{E}[n_B])^2$;分母等于 n_B 的方差(式(2.59)),由定义可知这是分子的期望。因此 z 的期望是 1。[像 z 这样一个随机变量有时称作 χ^2 随机变量,它所度量的是一个数据与其期望值之间的偏差。]

当 $N = 5$ 和 $f_B = 1/5$ 时,有 $Nf_B = 1$ 和 $\mathrm{var}[n_B] = 4/5$。这个分子有 5 个可能值,其中只有一个是小于 1 的:$(n_B - f_B N)^2 = 0$ 的概率为 $P(n_B = 1) = 0.409\,6$;因此 $z < 1$ 的概率为 0.409 6。

习题 2.14 的解答 已知性质

$$f(\lambda x_1 + (1-\lambda)x_2) \leqslant \lambda f(x_1) + (1-\lambda)f(x_2) \tag{2.60}$$

如果 $\sum p_i = 1$ 且 $p_i \geqslant 0$,欲证

$$\sum_{i=1}^{I} p_i f(x_i) \geqslant f\left(\sum_{i=1}^{I} p_i x_i\right) \tag{2.61}$$

从右边入手进行递归。(这个证明中未考虑 $p_i = 0$ 的情况;这类细节就留给那些有学究气的读者处理。)

$$
\begin{aligned}
f\Big(\sum_{i=1}^{I} p_i x_i\Big) &= f\Big(p_1 x_1 + \sum_{i=2}^{I} p_i x_i\Big) \\
&\leqslant p_1 f(x_1) + \Big[\sum_{i=2}^{I} p_i\Big]\Big[f\Big(\sum_{i=2}^{I} p_i x_i \Big/ \sum_{i=2}^{I} p_i\Big)\Big] \\
&\leqslant p_1 f(x_1) + \Big[\sum_{i=2}^{I} p_i\Big]\left[\frac{p_2}{\sum_{i=2}^{I} p_i} f(x_2) + \frac{\sum_{i=3}^{I} p_i}{\sum_{i=2}^{I} p_i} f\Big(\sum_{i=3}^{I} p_i x_i \Big/ \sum_{i=3}^{I} p_i\Big)\right]
\end{aligned}
\tag{2.62}
$$

上式中，第一行用到 $\lambda = \dfrac{p_1}{\sum_{i=1}^{I} p_i} = p_1$ 时凸性的定义（式(2.60)）；在第二行，$\lambda = \dfrac{p_2}{\sum_{i=2}^{I} p_i}$；如此等等。 □

习题 2.16 的解答

（a）对于结果$\{2,3,4,5,6,7,8,9,10,11,12\}$，概率

$$
\mathcal{P} = \left\{\frac{1}{36}, \frac{2}{36}, \frac{3}{36}, \frac{4}{36}, \frac{5}{36}, \frac{6}{36}, \frac{5}{36}, \frac{4}{36}, \frac{3}{36}, \frac{2}{36}, \frac{1}{36}\right\}
$$

（b）一个骰子的面值具有均值 3.5 和方差 35/12。因此，100 个骰子之和具有均值 350 和方差 3 500/12≃292，根据中心极限定理，其概率分布大致为具有这个均值和方差的高斯分布（不过要局限在整数上）。

（c）为了获得一个具有均匀分布的和，开始选定的一些随机变量必须具有尖峰分布，即概率质量（probability mass）集中在两个极值上。本题的唯一解是一个面值为 6、6、6、0、0、0 的普通骰子。

（d）是的，可用多种方法来产生一个均匀分布，比如，可以用数字$\{0,1,2,3,4,5\} \times 6^r$ 来标记第 r 个骰子。①

习题 2.17 的解答

$$
a = \ln \frac{p}{q} \quad \Rightarrow \quad \frac{p}{q} = e^a \tag{2.63}
$$

由 $q = 1 - p$，有

$$
\frac{p}{1-p} = e^a \tag{2.64}
$$

$$
\Rightarrow \quad p = \frac{e^a}{e^a + 1} = \frac{1}{1 + \exp(-a)} \tag{2.65}
$$

双曲正切函数为

$$
\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \tag{2.66}
$$

① 思考：这个均匀分布和中心极限定理相矛盾吗？

因此有

$$f(a) \equiv \frac{1}{1+\exp(-a)} = \frac{1}{2}\left(\frac{1-e^{-a}}{1+e^{-a}}+1\right)$$

$$= \frac{1}{2}\left(\frac{e^{a/2}-e^{-a/2}}{e^{a/2}+e^{-a/2}}+1\right) = \frac{1}{2}(\tanh(a/2)+1) \tag{2.67}$$

当 $b = \log_2 p/q$ 时，用 2 取代 e，重复步骤式(2.63) ~ (2.65)，可得

$$p = \frac{1}{1+2^{-a}} \tag{2.68}$$

习题 2.18 的解答

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)} \tag{2.69}$$

$$\Rightarrow \frac{P(x=1|y)}{P(x=0|y)} = \frac{P(y|x=1)}{P(y|x=0)}\frac{P(x=1)}{P(x=0)} \tag{2.70}$$

$$\Rightarrow \log\frac{P(x=1|y)}{P(x=0|y)} = \log\frac{P(y|x=1)}{P(y|x=0)} + \log\frac{P(x=1)}{P(x=0)} \tag{2.71}$$

习题 2.19 的解答 在已知 x 的前提下，d_1 和 d_2 是条件独立的，即

$$P(x,d_1,d_2) = P(x)P(d_1|x)P(d_2|x) \tag{2.72}$$

据此，可将后验概率比分解成一系列因子(每个数据点一个)乘以先验概率比。

$$\frac{P(x=1|\{d_i\})}{P(x=0|\{d_i\})} = \frac{P(\{d_i\}|x=1)}{P(\{d_i\}|x=0)}\frac{P(x=1)}{P(x=0)} \tag{2.73}$$

$$= \frac{P(d_1|x=1)}{P(d_1|x=0)}\frac{P(d_2|x=1)}{P(d_2|x=0)}\frac{P(x=1)}{P(x=0)} \tag{2.74}$$

高维空间中的情况

习题 2.20 的解答 事实上，N 维空间中一个半径为 r 的超球体的体积为

$$V(r,N) = \frac{\pi^{N/2}}{(N/2)!}r^N \tag{2.75}$$

不过你并不需要知道这个公式。对于这个问题，只需知道体积与 r 的相关性，即 $V(r,N) \propto r^N$。因此在 $(r-\varepsilon, r)$ 中的体积所占比重为

$$\frac{r^N-(r-\varepsilon)^N}{r^N} = 1-\left(1-\frac{\varepsilon}{r}\right)^N \tag{2.76}$$

对于提问的几种情况，球壳的体积所占比重为

N	2	10	1 000
$\varepsilon/r = 0.01$	0.02	0.096	0.999 96
$\varepsilon/r = 0.5$	0.75	0.999	$1-2^{-1\,000}$

注意：无论 ε 多么小，只要 N 足够大，几乎所有概率质量都在厚度为 ε 的球面壳内。

习题 2.21 的解答 $p_a=0.1, p_b=0.2, p_c=0.7$。$f(a)=10, f(b)=5, f(c)=10/7$。

$$\mathcal{E}[f(x)]=0.1\times10+0.2\times5+0.7\times10/7=3 \tag{2.77}$$

对于每个 x, $f(x)=1/P(x)$, 因此

$$\mathcal{E}[1/P(x)]=\mathcal{E}[f(x)]=3 \tag{2.78}$$

习题 2.22 的解答 对于一般的 X, 有

$$\mathcal{E}[1/P(x)]=\sum_{x\in\mathcal{A}_X}P(x)1/P(x)=\sum_{x\in\mathcal{A}_X}1=|\mathcal{A}_X| \tag{2.79}$$

习题 2.23 的解答 $p_a=0.1, p_b=0.2, p_c=0.7$。$g(a)=0, g(b)=1, g(c)=0$。

$$\mathcal{E}[g(x)]=p_b=0.2 \tag{2.80}$$

习题 2.24 的解答

$$P(P(x)\in[0.15,0.5])=p_b=0.2 \tag{2.81}$$

$$P\left(\left|\log\frac{P(x)}{0.2}\right|>0.05\right)=p_a+p_c=0.8 \tag{2.82}$$

习题 2.25 的解答 这类问题可以用两种方法来求解。一种方法是先对要最大化的函数求导, 得出最大值, 再证明它是全局最大值; 这种方法多少有点风险, 因为函数的最大可能值位于空间的边界上, 或位于导数不为 0 的地方。另一种方法是, 仔细选择一个不等式来给出答案。第二种方法比第一种方法要漂亮得多。

通过求导来证明(不推荐采用这种方法) 因为求导 $\ln 1/p$ 比 $\log_2 1/p$ 要稍微简单一些, 暂时定义 $H(X)$ 是用自然对数来度量的, 因此将它按比例缩小了一个因子 $\log_2 e$。

$$H(X)=\sum_i p_i\ln\frac{1}{p_i} \tag{2.83}$$

$$\frac{\partial H(X)}{\partial p_i}=\ln\frac{1}{p_i}-1 \tag{2.84}$$

在 $\sum_i p_i=1$ 的约束条件下进行最大化。这个约束条件可以用如下拉格朗日乘子来保证:

$$G(\boldsymbol{p})\equiv H(X)+\lambda\left(\sum_i p_i-1\right) \tag{2.85}$$

$$\frac{\partial G(\boldsymbol{p})}{\partial p_i}=\ln\frac{1}{p_i}-1+\lambda \tag{2.86}$$

在最大值处有

$$\ln\frac{1}{p_i}-1+\lambda=0 \tag{2.87}$$

$$\Rightarrow\ln\frac{1}{p_i}=1-\lambda \tag{2.88}$$

因此所有的 p_i 都相等。这一极值确实为最大值, 因为曲率

$$\frac{\partial^2 G(\boldsymbol{p})}{\partial p_i \partial p_j} = -\frac{1}{p_i}\delta_{ij} \tag{2.89}$$

为负定的(negative definite)。 □

用 Jensen 不等式证明(推荐方法) 先回顾一下这个不等式。

如果 f 是一个凸 $\smile$ 函数,x 是一个随机变量,则有

$$\mathscr{E}[f(x)] \geqslant f(\mathscr{E}[x])$$

如果 f 是一个严格凸 $\smile$ 函数且 $\mathscr{E}[f(x)] = f(\mathscr{E}[x])$,那么随机变量 x(以概率 1)为常数。

利用 Jensen 不等式进行证明的秘密在于,选择一个正确的函数和一个正确的随机变量。可以定义

$$f(u) = \log\frac{1}{u} = -\log u \tag{2.90}$$

(它是一个凸函数),并将 $H(X) = \sum p_i \log\frac{1}{p_i}$ 当做 $f(u)$ 的均值,其中 $u = P(x)$,但这不会得出所需要的结论,而只是给出一个反向的不等式。如果转而定义

$$u = 1/P(x) \tag{2.91}$$

则有

$$H(X) = -\mathscr{E}[f(1/P(x))] \leqslant -f(\mathscr{E}[1/P(x)]) \tag{2.92}$$

由习题 2.22 知 $\mathscr{E}[1/P(x)] = |\mathscr{A}_X|$,因此有

$$H(X) \leqslant -f(|\mathscr{A}_X|) = \log|\mathscr{A}_X| \tag{2.93}$$

只有当随机变量 $u = 1/P(x)$ 为常数时,也就是说,对于所有 x 都有 $P(x)$ 为常数时,等号才成立。 □

习题 2.26 的解答

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_x P(x)\log\frac{P(x)}{Q(x)} \tag{2.94}$$

利用 Jensen 不等式来证明 Gibbs 不等式。令 $f(u) = \log 1/u$ 和 $u = \frac{Q(x)}{P(x)}$。于是有

$$D_{\mathrm{KL}}(P \parallel Q) = \mathscr{E}[f(Q(x)/P(x))] \tag{2.95}$$

$$\geqslant f\left(\sum_x P(x)\frac{Q(x)}{P(x)}\right) = \log\left(\frac{1}{\sum_x Q(x)}\right) = 0 \tag{2.96}$$

只有当 $u = \frac{Q(x)}{P(x)}$ 为常数,即 $Q(x) = P(x)$ 时,才有等式成立。 □

另一种解法 在上面的证明中,是对概率分布 $P(x)$ 所求的期望。而这第二种解法则对 $Q(x)$ 运用 Jensen 不等式。定义 $f(u) = u\log u$ 和 $u = \frac{P(x)}{Q(x)}$,则有

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_x Q(x)\frac{P(x)}{Q(x)}\log\frac{P(x)}{Q(x)} = \sum_x Q(x) f\left(\frac{P(x)}{Q(x)}\right) \tag{2.97}$$

$$\geqslant f\left(\sum_x Q(x)\frac{P(x)}{Q(x)}\right) = f(1) = 0 \tag{2.98}$$

只有当 $u=\frac{P(x)}{Q(x)}$ 为常数，即 $Q(x)=P(x)$ 时，才有等式成立。 □

习题 2.28 的解答

$$H(X)=H_2(f)+fH_2(g)+(1-f)H_2(h) \tag{2.99}$$

习题 2.29 的解答 掷出 $x-1$ 个反面之后才有 1 个正面（因此在第 x 次抛掷才第一次掷出正面）的概率是

$$P(x)=(1-f)^{x-1}f \tag{2.100}$$

如果第一次抛掷为反面，未来情形的概率分布看上去和抛掷第一次之前完全一样。因此，得到熵的一个递归表达式

$$H(X)=H_2(f)+(1-f)H(X) \tag{2.101}$$

重新组合，得

$$H(X)=H_2(f)/f \tag{2.102}$$

习题 2.34 的解答 反面的次数 t 服从概率分布

$$P(t)=\left(\frac{1}{2}\right)^t\frac{1}{2},\quad t\geqslant 0 \tag{2.103}$$

由题目定义，正面的期望次数为 1。反面的期望次数为

$$\mathcal{E}[t]=\sum_{t=0}^{\infty}t\left(\frac{1}{2}\right)^t\frac{1}{2} \tag{2.104}$$

可用多种方式证明上式等于 1。比如，因为在抛掷出一个反面后的情形和未抛之前的情形是等价的，则可写出如下递归关系

$$\mathcal{E}[t]=\frac{1}{2}(1+\mathcal{E}[t])+\frac{1}{2}0\Rightarrow\mathcal{E}[t]=1 \tag{2.105}$$

在已知 $f=1/2$ 的前提下，“估计器”$\hat{f}=1/(1+t)$ 的概率分布如图 2.12 所示。$\hat{f}$ 的概率就是相应 t 值的概率。

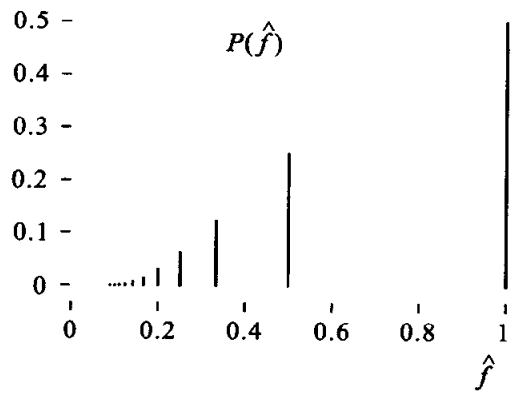


图 2.12 在已知 $f=1/2$ 的前提下，“估计器”$\hat{f}=1/(1+t)$ 的概率分布。

习题 2.35 的解答

本题示意图如图 2.13 所示。

(a) 从一个第 6 点到下一个第 6 点要抛掷的平均次数为 6（假设在抛掷出前两

个第6点之后开始对次数进行计数)。下一个第6点在第 r 次抛掷出现的概率等于 $r-1$ 次未抛出第6点的概率乘以接下来抛出一个第6点的概率:

$$P(r_1=r)=\left(\frac{5}{6}\right)^{r-1}\frac{1}{6},\quad r\in\{1,2,3,\cdots\} \tag{2.106}$$

抛掷次数 r 的概率分布可叫做指数分布,因为

$$P(r_1=r)=\mathrm{e}^{-\alpha r}/Z \tag{2.107}$$

其中 $\alpha=\ln(6/5)$,Z 是归一化常数。

(b) 到下一个第6点的平均抛掷数为6。

(c) 时间回溯到最近一个第6点的平均抛掷数为6。

(d) 从敲钟前的一个第6点到敲钟后的一个第6点的平均抛掷数是(b)和(c)答案之和减1,即11。

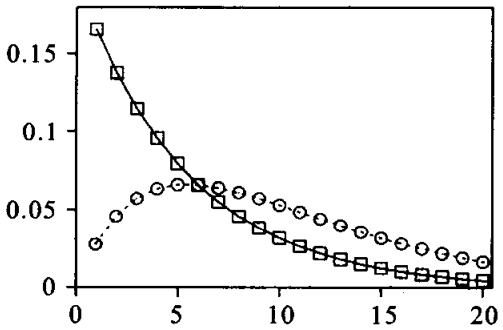


图2.13 从一个第6点到下一个第6点要抛掷的次数 r_1 的概率分布(下降的实线)$P(r_1=r)=\left(\frac{5}{6}\right)^{r-1}\frac{1}{6}$。在下午一点钟之前的一个第6点到下一个第6点要抛掷的次数 r_{tot} 的概率分布(虚线)$P(r_{\mathrm{tot}}=r)=r\left(\frac{5}{6}\right)^{r-1}\left(\frac{1}{6}\right)^2$。概率 $P(r_1>6)$ 约为1/3;概率 $P(r_{\mathrm{tot}}>6)$ 约为2/3。r_1 的均值为6,而 r_{tot} 的均值为11。

(e) 再给一个提示,而不是直接解释(a)和(d)之间的差别。设想 Poissonville 的汽车独立地随机到达(泊松过程),平均每6分钟来一辆车。设想乘客以均匀速率出现在汽车站,并且无延迟地被汽车运走,因此两辆汽车之间的间距保持恒定。间隔大于6分钟之后到达的汽车会变得过于拥挤。乘客中的代表抱怨,有三分之二的乘客坐上的是过于拥挤的汽车。而汽车运营商则声称:"不,不,我们的汽车只有三分之一是过于拥挤的"。有没有可能两方面的说法都是正确的?

习题2.38的解答

二项分布法 由习题1.2的解答有 $p_B=3f^2(1-f)+f^3$。

加法律方法 $\boldsymbol{r}$ 的8个值的边缘概率说明如下:

$$P(\boldsymbol{r}=000)=1/2(1-f)^3+1/2f^3 \tag{2.108}$$

$$P(\boldsymbol{r}=001)=1/2f(1-f)^2+1/2f^2(1-f)=1/2f(1-f) \tag{2.109}$$

后验概率表示为

$$P(s=1\,|\,\boldsymbol{r}=000)=\frac{f^3}{(1-f)^3+f^3} \tag{2.110}$$

和

$$P(s=1|r=001)=\frac{(1-f)f^2}{f(1-f)^2+f^2(1-f)}=f \tag{2.111}$$

因此,在这些有代表性的场景下,差错概率为

$$P(\text{差错}|r=000)=\frac{f^3}{(1-f)^3+f^3} \tag{2.112}$$

和

$$P(\text{差错}|r=001)=f \tag{2.113}$$

注意:R_3 的平均差错概率约为 $3f^2$,但是任一特定比特出错(已知 r 的前提下)的概率要么约为 f^3 要么为 f。

运用加法律,平均差错概率为①

$$P(\text{差错})=\sum_r P(r)P(\text{差错}|r)$$

$$=2[1/2(1-f)^3+1/2f^3]\frac{f^3}{(1-f)^3+f^3}+6[1/2f(1-f)]f$$

因此

$$P(\text{差错})=f^3+3f^2(1-f)$$

习题 2.39 的解答 熵为每单词 9.7 比特。

① 下式中的前两项是针对 $r=000$ 和 111 这两种情形;其余 6 个是针对其他结果,后者具有相同的发生概率和相同的差错概率 f。

关于第3章

如果你迫切地想了解信息论、数据压缩以及噪声信道的内容，你可以直接阅读第4章。然而，数据压缩和数据建模是紧密相关的，所以当你读到第6章时，可能需要返回到本章来。在阅读第3章之前，最好看看下面这些习题。

▷ **习题3.1**[2] 随机地从两个有20面的骰子中选取一个骰子，这两个骰子上所写的符号1～10是频度不均的，如下表所示。

符号	1	2	3	4	5	6	7	8	9	10
骰子A的面数	6	4	3	2	1	1	1	1	1	0
骰子B的面数	3	3	2	2	2	2	2	2	1	1

随机选取的骰子被抛掷7次，结果如下：

5,3,9,3,8,4,7

所选骰子是骰子A的概率有多大？

▷ **习题3.2**[2] 假定存在第三个有20面的骰子C，在它的面上，符号1～20各出现一次。与上题一样，每次从这3个骰子中随机地选取一个，并抛掷7次，得出的结果如下：

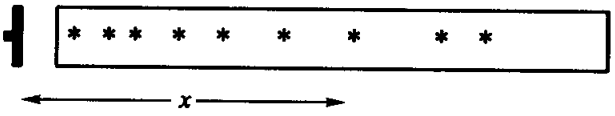
3,5,4,8,3,9,7

所选骰子是(a) 骰子A，(b)骰子B，(c)骰子C的概率各是多大？



习题3.3[2] **推测衰变常数**

不稳定的粒子从一个源被释放出来，并在距离 x 处衰变，其中实数 x 是服从特征长度 λ 的指数概率分布。只有发生在 $x=1$ cm 到 $x=20$ cm 的这个窗口内的衰变事件才能被观测到。假定在位置 $\{x_1,\cdots,x_N\}$ 处观测到了 N 次衰变，λ 为多少？



▷ **习题3.4**[2] **司法证据**

两个人在一个犯罪现场留下了他们自己的血迹。对一个叫奥利弗的嫌疑人进行检测，并得出结果是“O”型血。两处血迹的血型查出来一处是“O”型（这种血型在

当地非常普遍，频度为60%），另一处是“AB”型（这种血型非常罕见，频度为1%）。这些数据（在犯罪现场发现的“O”型血和“AB”型血）是否支持“奥利弗是两个出现在犯罪现场的人之一”的推测？

第 3 章

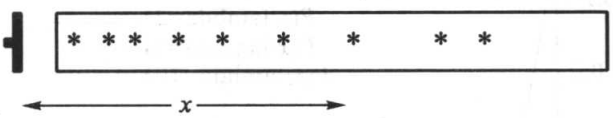
有关推理的更多内容

正如第 1 章所使用的那样，对于有噪信道上的通信消息的推理，贝叶斯定理为其描述提供了一种正确的语言，这一点是确定无疑的。但很奇怪的是，对于其他推理问题，贝叶斯定理的运用就不那么广泛了。

3.1 第一个推理问题

笔者在剑桥大学读本科时，有幸得到 Steve Gull 的指导。坐在他位于圣约翰学院的那间零乱的办公室里，我问他应如何回答一道古老的 Tripos 问题（习题 3.3）：

不稳定的粒子从一个源被释放出来，并在距离 x 处发生衰变，其中实数 x 是服从特征长度 λ 的指数概率分布。衰变事件只有发生在 $x = 1$ cm 到 $x = 20$ cm 的这个窗口内时，才能被观测到。假定在位置 $\{x_1, \cdots, x_N\}$ 处观测到了 N 次衰变，λ 为多少？



我为这个问题已冥思苦想了一段时间。根据我所受的教育，我可以用两种方法来求解这类推理问题：构造关于未知参数的“估计器”；或者根据原始数据或经过处理的数据对模型进行“拟合”。

因为一个无约束的指数分布随机变量的均值为 λ，考察样本均值 $\bar{x} = \sum_n x_n / N$，并看看能否由此得出一个估计器 $\hat{\lambda}$，这样做似乎有些道理。显然，估计器 $\hat{\lambda} = \bar{x} - 1$ 对于 $\lambda << 20$ cm 的情况是合适的，但当分布的右截断不可忽略时就不再适用了。用些技巧并引入一些特别的自组器，对于 $\lambda >> 20$ cm 的情况也可以构造出很好的估计器。但是，几乎不存在一种在所有条件下都适用的估计器。

而根据数据柱状图来拟合概率密度 $P(x|\lambda)$，也无法由此找出一种令人满意的方法。我被难住了。

这个问题以及其他类似问题的通解是怎样的？是否在遇到一个新的推理问题时，都必须在黑暗中摸索出那些合适的“估计器”并为寻求“最佳”估计器而费心（无论其具体涵义是什么）？

Steve 写出了已知 λ 时一个数据点的概率

$$P(x|\lambda)=\begin{cases}\dfrac{1}{\lambda}e^{-x/\lambda}/Z(\lambda), & 1<x<20\\ 0, & \text{其他}\end{cases}\tag{3.1}$$

其中

$$Z(\lambda)=\int_1^{20}dx\,\frac{1}{\lambda}e^{-x/\lambda}=(e^{-1/\lambda}-e^{-20/\lambda})\tag{3.2}$$

这看起来非常显而易见。他然后写出贝叶斯定理

$$P(\lambda|\{x_1,\cdots,x_N\})=\frac{P(\{x\}|\lambda)P(\lambda)}{P(\{x\})}\tag{3.3}$$

$$\propto\frac{1}{(\lambda Z(\lambda))^N}\exp\Big(-\sum_1^N x_n/\lambda\Big)P(\lambda)\tag{3.4}$$

已知一个假设 λ 时，定义数据的概率是一种简明易懂的分布 $P(\{x_1,\cdots,x_N\}|\lambda)$。这里，它一下转而变成了已知数据时定义一个假设的概率。一幅简图可将不同 λ 值所对应的单点概率 $P(x|\lambda)$ 表示为 x 的一个常见函数（如图 3.1 所示）。其中每条曲线都是一条单纯的指数曲线，只是经过了归一化使其所围面积为 1。对于一个固定的 x 值，将同一个函数作为 λ 的函数画出来，一件不同凡响的事情发生了：出现了一个尖峰（如图 3.2 所示）。为了帮助读者理解一个函数的两种视角的情况，图 3.3 所示为以 x 和 λ 的函数所表示的 $P(x|\lambda)$ 的平面图。

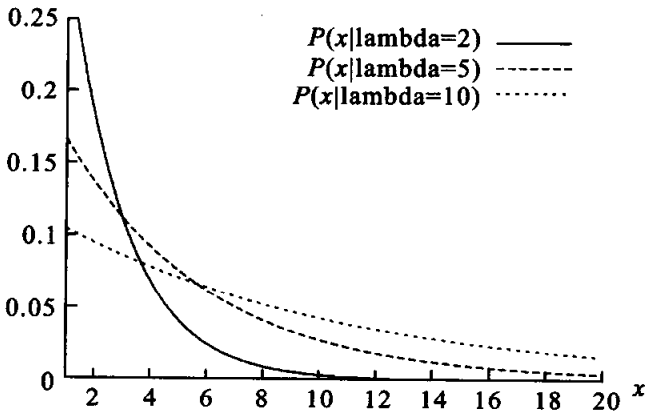


图 3.1　以 x 的函数所表示的概率密度 $P(x|\lambda)$

对由几个点所组成的一个数据集，例如这 6 个点 $\{x\}_{n=1}^N=\{1.5,2,3,4,5,12\}$，似然函数 $P(\{x\}|\lambda)$ 是 N 个 λ 函数 $P(x_n|\lambda)$ 的乘积（如图 3.4 所示）。

Steve 将贝叶斯定理总结为体现了如下事实。

> 在数据到达后，你所得知的 λ 是你所知道的[$P(\lambda)$]之前的值，而数据给你的只是[$P(\{x\}|\lambda)$]。

在这里概率用来对置信度进行量化。为了在一开始就消除可能的混淆，必须强调正确地描述这种情况的假设 λ 不是随机变量，贝叶斯使用了概率分布 P 也不意味着他就认为这个世界会在不同假设所描述的各种状态之间随机地改变其本性。他使

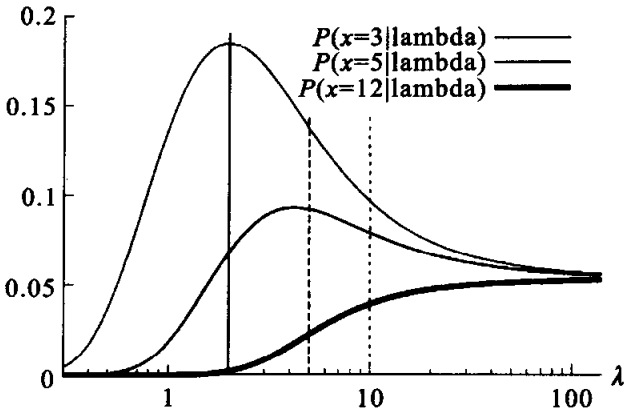


图 3.2 对于 3 个不同的 x 值，以 λ 的函数表示的概率密度 $P(x|\lambda)$。这样画图的时候，这个函数称为 λ 的似然函数。图中竖线标出了在前图中所用的 3 个 λ 值，$\lambda=2,5,10$。

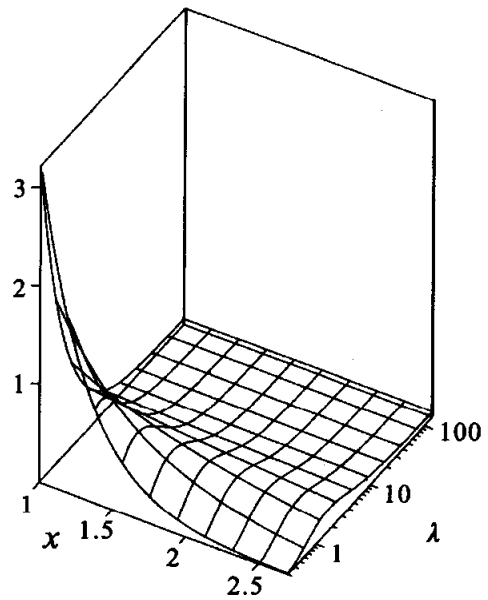


图 3.3 以 x 和 λ 的函数所表示的概率密度 $P(x|\lambda)$。
图 3.1 和图 3.2 所示都是该曲面的纵向剖面。

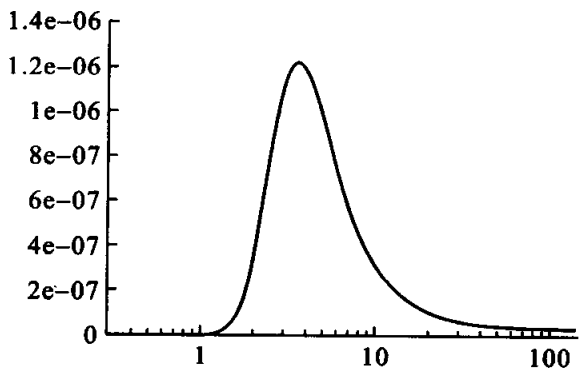


图 3.4 6 点数据集 $P(\{x\}=\{1.5,2,3,4,5,12\}|\lambda)$ 例子中，以 λ 为自变量的似然函数。

用概率记号所表示的只是他对各互斥的小假设(这里为 λ 的各个值)的置信度,其中这些小假设中只有一个是正确的。对于给定的假设条件,概率可用来表示置信度,在我看来是合情合理的。

后验概率分布式(3.4)表示这个问题的唯一完全解。我们无需去新创"估计器",也不必创建判据来对几种估计器进行比较。正统的统计学家们会提供二十种方法来求解一个问题,并用另外二十种不同的判据来确定这些方案中哪一种最佳,而贝叶斯统计学对于一个适定的问题只给出一个答案。①

推理中的假设

我们的推理是以我们所做假设为前提条件的[比如,先验概率 $P(\lambda)$]。批评者将这些先验概率视做一个障碍,因为它们是"主观性的",但是笔者看不出来除此之外还能怎样。人们又怎能不做假设而进行推理呢?笔者相信贝叶斯方法迫使人们明确地做出这些默认的假设,这一点本身就具有很大的价值。

首先,一旦做了这些假设,所进行的推理就是客观的和唯一的,而且对于拥有相同信息并做出了同样假设的任何人,都是完全可以重复的。比如,已知上述假设 $\mathscr{H}$ 和数据 D,大家都会同意衰变长度 λ 的后验概率是

$$P(\lambda|D,\mathscr{H})=\frac{P(D|\lambda,\mathscr{H})P(\lambda|\mathscr{H})}{P(D|\mathscr{H})} \tag{3.5}$$

其次,当我们以显式形式做出假设时,要对它们进行批评或修改都会更容易一些。事实上,还可以就我们的推理对具体假设的敏感度进行量化。比如,从图 3.2 所示的似然曲线中,可以看出单一数据点 $x=5$ 所对应的曲线尖峰不如 $x=3$ 的情况那么突出;样本均值 $\bar{x}$ 距离窗口中心 10.5 越近,先验概率 $P(\lambda)$ 的详情就变得越重要。当 $x=12$ 时,似然函数完全没有尖峰——这种数据仅仅排除掉很小的 λ 值,而未给出有关大 λ 值的相对概率方面的任何信息。因此,在这种情况下,在小 λ 值这一端,先验概率的详情并不重要,但是在大 λ 值那一端,先验概率就非常重要了。

第三,当我们不太确定各种假设中哪一种对于一个问题最合适时,可以把这个问题处理为另一个推理问题。因此,已知数据 D,可以用贝叶斯定理对于可供选择的几种假设 $\mathscr{H}$ 进行比较。

$$P(\mathscr{H}|D,I)=\frac{P(D|\mathscr{H},I)P(\mathscr{H}|I)}{P(D|I)} \tag{3.6}$$

其中 I 表示那些我们不怀疑问的最高假设。

第四,在进行后续的预测时,可以将我们对这些假设的不确定性纳入其中。我们并不选择一个特定的假设 $\mathscr{H}^*$,并算出关于某个量 t 的预测 $P(t|D,\mathscr{H}^*,I)$,而是运用如下加法律将我们关于 $\mathscr{H}$ 的不确定性考虑进去,从而得出预测值

① 如果你阅读本章时觉得理解上有困难,我建议你先解答好习题 3.1 和习题 3.2,再注意一下它们和习题 3.3 之间的相似之处。

$$P(t \mid D, I) = \sum_{\mathcal{H}} P(t \mid D, \mathcal{H}, I) P(\mathcal{H} \mid D, I) \tag{3.7}$$

这是同正统统计学相对的又一不同点。在正统统计学中，通常会“测试”一个默认的模型，如果在“很大程度上”测试“接受了这个模型”，然后就会单纯地使用这个模型做出预测。

因此，Steve 让我确信

> 概率论能抵达自组方法所不能达到的地方。

下面再来看几个有关简单推理问题的例子。

3.2 变形硬币

抛掷一枚变形硬币 F 次，观察到正面和反面（分别记为符号 a 和 b）所组成的一个序列 $\boldsymbol{s}$。我们想知道这枚硬币的偏倚，并预测下一次抛掷会出现正面的概率。在例 2.7 中，首次遇到这一问题，而且在第 6 章讨论自适应数据压缩时还会遇到它。它也是托马斯·贝叶斯在他发表于 1763 年的论文中最初讨论过的推理问题。

与在习题 2.8 中一样，假设一个均匀先验分布，并通过乘以似然函数得出后验分布。批评者可能会提出异议：“这种先验分布凭何而来？”笔者并不认为均匀先验分布是一个根本性的假设，而且事实上我们还会在后面给出非均匀先验分布的例子。先验分布是一种主观性的假设。本书的一个主要观点是：

> 不做假设就无法进行推理或数据压缩。

我们将我们的假设称作 $\mathcal{H}_1$。[稍后将引入另一组假设。]已知 p_a，F 次抛掷产生一个包含 $\{F_a, F_b\}$ 次正/反面序列 $\boldsymbol{s}$ 的概率为

$$P(\boldsymbol{s} \mid p_a, F, \mathcal{H}_1) = p_a^{F_a} (1 - p_a)^{F_b} \tag{3.8}$$

[例如，$P(\boldsymbol{s} = \text{aaba} \mid p_a, F = 4, \mathcal{H}_1) = p_a p_a (1 - p_a) p_a$。]第一个模型假设 p_a 的一个均匀先验分布

$$P(p_a \mid \mathcal{H}_1) = 1, \quad p_a \in [0, 1] \tag{3.9}$$

且有 $p_b \equiv 1 - p_a$。

对未知参数的推理

已知一个长度为 F 的字符串，其中包含 F_a 个 a 和 F_b 个 b，我们感兴趣的是：(a) 推断出 p_a 的数值；(b) 预测下一个字符是 a 还是 b。[预测一般表示为概率形式。因此，“预测下一个字符是否为 a”也就是计算下一个字符为 a 的概率。]

假设 $\mathcal{H}_1$ 为真，根据贝叶斯定理，已知一个长度为 F 且具有计数 $\{F_a, F_b\}$ 的字符串 $\boldsymbol{s}$，p_a 的后验概率为

$$P(p_a|s,F,\mathcal{H}_1)=\frac{P(s|p_a,F,\mathcal{H}_1)P(p_a|\mathcal{H}_1)}{P(s|F,\mathcal{H}_1)} \tag{3.10}$$

分式 $P(s|p_a,F,\mathcal{H}_1)$ 由式(3.8)给出,它是 p_a 的函数,叫做似然函数;先验概率 $P(p_a|\mathcal{H}_1)$ 由式(3.9)给出。因此所推出的 p_a 为

$$P(p_a|s,F,\mathcal{H}_1)=\frac{p_a^{F_a}(1-p_a)^{F_b}}{P(s|F,\mathcal{H}_1)} \tag{3.11}$$

其中的归一化常数由如下 Beta 积分给出。

$$P(s|F,\mathcal{H}_1)=\int_0^1 dp_a p_a^{F_a}(1-p_a)^{F_b}=\frac{\Gamma(F_a+1)\Gamma(F_b+1)}{\Gamma(F_a+F_b+2)}=\frac{F_a!+F_b!}{(F_a+F_b+1)!} \tag{3.12}$$



习题 3.5[⌐] 画出后验概率 $P(p_a|s=\text{aba},F=3)$ 的草图。最可能的 p_a 值(也就是使后验概率密度最大化的那个值)是多少?在此分布下,p_a 的均值是多少?

对于后验概率 $P(p_a|s=\text{bbb},F=3)$,解答同样的问题。

由推理到预测

对于下一抛掷的预测,即下一抛为 a 的概率,可以通过对 p_a 进行积分而得到。因此,在做预测时,可以将 p_a 的不确定性考虑进去。由加法律,有

$$P(a|s,F)=\int dp_a P(a|p_a)P(p_a|s,F) \tag{3.13}$$

已知 p_a 的条件下,出现一个 a 的概率就是 p_a,因此有

$$P(a|s,F)=\int dp_a p_a\frac{p_a^{F_a}(1-p_a)^{F_b}}{P(s|F)} \tag{3.14}$$

$$=\int dp_a\frac{p_a^{F_a+1}(1-p_a)^{F_b}}{P(s|F)} \tag{3.15}$$

$$=\left[\frac{(F_a+1)!\,F_b!}{(F_a+F_b+2)!}\right]\Big/\left[\frac{F_a!\,F_b!}{(F_a+F_b+1)!}\right]=\frac{F_a+1}{F_a+F_b+2} \tag{3.16}$$

以上公式称为拉普拉斯法则。

3.3 变形硬币和模型比较

设想有一位科学家对我们的数据引入了另一种理论。他宣称我们的数据源并非一个真正的变形硬币,而是一个形状完美的骰子,其中一面漆成正面("a"),其他 5 个面都漆成反面("b")。因此,在原有模型 $\mathcal{H}_1$ 中的可以在 0 与 1 之间取任何值的参数 p_a,在新假设 $\mathcal{H}_0$ 之下,不再是一个自由参数,而是等于 1/6。[这个假设记为 $\mathcal{H}_0$,这样每个模型的下标表示其自由参数的个数。]

怎样才能对数据的这两种模型进行比较呢？我们希望推断出相对于 $\mathcal{H}_0$ 而言模型 $\mathcal{H}_1$ 有多大的可能性是正确的。

以模型比较做推理

为了进行模型比较，重写贝叶斯定理，不过这一次出现在左边的自变量不同了。我们想知道，已知这些数据，模型 $\mathcal{H}_1$ 有多大的可能性是正确的。由贝叶斯定理，有

$$P(\mathcal{H}_1 \mid s, F) = \frac{P(s \mid F, \mathcal{H}_1) P(\mathcal{H}_1)}{P(s \mid F)} \tag{3.17}$$

类似地，$\mathcal{H}_0$ 的后验概率是

$$P(\mathcal{H}_0 \mid s, F) = \frac{P(s \mid F, \mathcal{H}_0) P(\mathcal{H}_0)}{P(s \mid F)} \tag{3.18}$$

上面两种情况中的归一化常数都是获取观测数据的总概率 $P(s \mid F)$。如果所考虑的只是模型 $\mathcal{H}_1$ 和 $\mathcal{H}_0$，这一概率可由加法律给出

$$P(s \mid F) = P(s \mid F, \mathcal{H}_1) P(\mathcal{H}_1) + P(s \mid F, \mathcal{H}_0) P(\mathcal{H}_0) \tag{3.19}$$

要计算这些假设的后验概率，需要对先验概率 $P(\mathcal{H}_1)$ 和 $P(\mathcal{H}_0)$ 进行赋值；这里可将它们都赋值为 1/2。而且，还要计算与数据相关的项 $P(s \mid F, \mathcal{H}_1)$ 和 $P(s \mid F, \mathcal{H}_0)$。可以为这些量命名。$P(s \mid F, \mathcal{H}_1)$ 度量了数据对 $\mathcal{H}_1$ 的依赖程度，因此称它为模型 $\mathcal{H}_1$ 的证据（evidence）。在式(3.10)中已经遇到过这个量：它在我们所做的第一个推理（即已知数据而推断 p_a）中是以归一化常数的形式出现的。

如何进行模型比较 一个模型的证据通常是早先的贝叶斯推理的归一化常数。

在式(3.12)中，计算了模型 $\mathcal{H}_1$ 的归一化常数。模型 $\mathcal{H}_0$ 的证据非常简单，因为这个模型的所有参数都不需要做推断。定义 p_0 为 1/6，有

$$P(s \mid F, \mathcal{H}_0) = p_0^{F_a} (1 - p_0)^{F_b} \tag{3.20}$$

因此，模型 $\mathcal{H}_1$ 和模型 $\mathcal{H}_0$ 的后验概率比为

$$\frac{P(\mathcal{H}_1 \mid s, F)}{P(\mathcal{H}_0 \mid s, F)} = \frac{P(s \mid F, \mathcal{H}_1) P(\mathcal{H}_1)}{P(s \mid F, \mathcal{H}_0) P(\mathcal{H}_0)} \tag{3.21}$$

$$= \frac{F_a!\ F_b!}{(F_a + F_b + 1)!} \Big/ p_0^{F_a} (1 - p_0)^{F_b} \tag{3.22}$$

表 3.5 列出了这个后验概率比的一些数值。前 5 行表明有些结果倾向于一个模型，而有些结果则倾向于另一个模型。没有哪个结果是和其中任何一个模型完全不相容的。如果数据量较小（比如，抛掷 6 次），通常不会出现两个模型之一相对另一个模型处于压倒性的优势这种情况。但是，随着数据量的增大，由比值 $F_a : F_b$ 不等于 1:5的任何数据集所提供的不倾向于 $\mathcal{H}_0$ 的证据会增加。你无法事先预测要很有把握地确定那个理论正确，到底需要多少数据。这还是取决于 p_0。

表 3.5 “变形硬币”模型 $\mathcal{H}_1$ 和 $\mathcal{H}_0$ 之间的模型比较结果。模型 $\mathcal{H}_0$ 规定 $p_a=1/6, p_b=5/6$。

F	数据 (F_a, F_b)	$\frac{P(\mathcal{H}_1 \mid s,F)}{P(\mathcal{H}_0 \mid s,F)}$	
6	(5,1)	222.2	
6	(3,3)	2.67	
6	(2,4)	0.71	$=1/1.4$
6	(1,5)	0.356	$=1/2.8$
6	(0,6)	0.427	$=1/2.3$
20	(10,10)	96.5	
20	(3,17)	0.2	$=1/5$
20	(0,20)	1.83	

因为比较简单的模型 $\mathcal{H}_0$ 没有可调节的参数，它可以以最大的边界值失利。对它不利的几率可能是几百分之一。更复杂的模型不可能以一个大的边界值失利；已知模型 $\mathcal{H}_1$，不会有哪个数据集在实际中是不可能的。

▷ **习题 3.6**[2] 试证明在抛掷了 F 次之后，如果 $\mathcal{H}_1$ 更有可能性，对数证据比。

$$\log \frac{P(s \mid F, \mathcal{H}_1)}{P(s \mid F, \mathcal{H}_0)} \tag{3.23}$$

的最大值会随 F 以一定比例线性增长；如果倾向于 $\mathcal{H}_0$，对数证据的增长速率至多是 $\log F$。

▷ **习题 3.7**[2] 使用你的抽样理论，假设尚未测量 F_a，作为 F 和真值 p_a 的函数，计算对数证据比可能的取值范围，并对 $p_a=p_0=1/6$，$p_a=0.25$ 和 $p_a=1/2$ 这三种情况画出它作为 F 的函数的草图。[提示：画出以随机变量 F_a 为自变量的对数证据的草图，并计算出 F_a 的均值和标准差。]

证据的典型特性

图 3.6 所示是在一系列仿真试验中，以抛掷次数 F 的函数所表示的对数证据比。在左半图所对应的试验中，$\mathcal{H}_0$ 为真。在右半图所对应的试验中，$\mathcal{H}_1$ 为真，而且 p_a 的取值为 0.25 或 0.5。

在后续章节中，将对模型比较进行更多的论述。

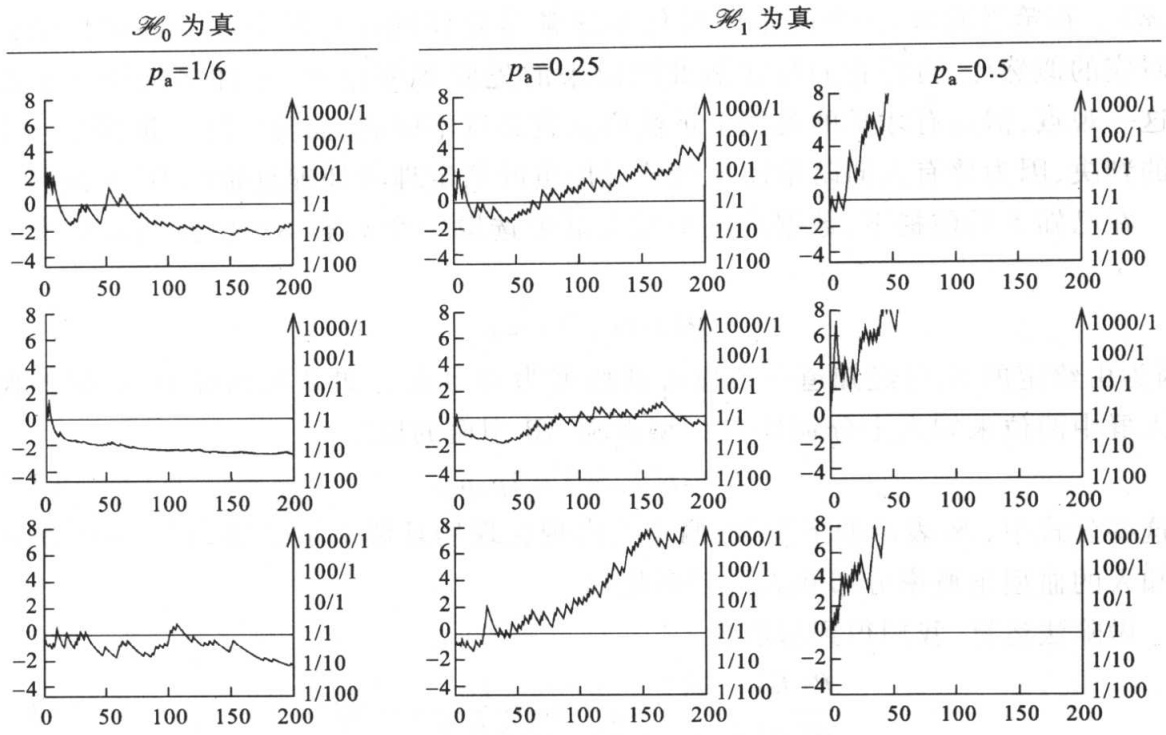


图 3.6 当变形硬币抛掷次数在 3 个不同条件下增加时，倾向于 $\mathscr{H}_1$ 的证据的典型特性。横坐标轴是抛掷次数 F。左图的纵坐标轴为 $\ln \frac{P(s|F,\mathscr{H}_1)}{P(s|F,\mathscr{H}_0)}$；右图的纵坐标轴表示 $\frac{P(s|F,\mathscr{H}_1)}{P(s|F,\mathscr{H}_0)}$ 的值。（也可参见图 3.8。）

3.4 一个司法证据的例子

下例表明贝叶斯推理可以给出先验概率以外的更多信息。

两个人在一个犯罪现场留下了他们自己的血迹。一个嫌疑犯奥利弗接受了测试，并得知其血型为“O”型。两处血迹的血型查出来分别是“O”型（当地人中一种常见的血型，出现的频度为 60%）和“AB”型（一种罕见血型，出现的频度为 1%）。试问所发现的信息（“O”型和“AB”型血迹出现在现场）是否有助于提供证据证明，奥利弗是出现在犯罪现场的两人之一？

一位粗心的律师可能会宣称：因为嫌疑犯的血型和犯罪现场所出现的血型相同，这为此人出现在现场提供了有力证据。但事实并非如此。

将命题“这个嫌疑犯和另一个未知人士出现在现场”记做 S。而另一命题 $\bar{S}$ 表示“两个未知人士出现在现场”。这个问题的预先知识是命题 S 和 $\bar{S}$ 的先验概率比。这个量对于做出最后的决断很重要，而且在这个案件中它要基于所有其他可获得的

信息。这里，我们的任务是估算数据 D 所做出的贡献，即似然比 $P(D|S,\mathscr{H})/P(D|\bar{S},\mathscr{H})$。在笔者看来，一个陪审团的任务通常是先仔细计算每个独立的可接纳证据所对应的似然比，再将它们与仔细推理出来的先验概率相乘。[许多统计学家都支持这一观点，但是有才学的英国上诉法庭法官最近不同意，而且实际上推翻了一个案子的判决，因为曾有人向陪审团成员传授用贝叶斯定理来处理复杂的 DNA 证据。]

在已知 S 的前提下，数据的概率是人群中选取一个具有 AB 型血的未知人的概率

$$P(D|S,\mathscr{H}) = p_{AB} \tag{3.24}$$

(因为由给定的 S，已经知道一条血迹的血型为 O。)在已知 $\bar{S}$ 的前提下，数据的概率是人群中两位未知人士分别具有 O 型血和 AB 型血的概率

$$P(D|\bar{S},\mathscr{H}) = 2p_O p_{AB} \tag{3.25}$$

在这些公式中，$\mathscr{H}$ 表示如下假设：两个人出现在现场且留下了血迹，而且一个解释中未知人的血型组概率分布和人群频率相同。

做除法运算，我们得到似然比

$$\frac{P(D|S,\mathscr{H})}{P(D|\bar{S},\mathscr{H})} = \frac{1}{2p_O} = \frac{1}{2\times 0.6} = 0.83 \tag{3.26}$$

因此，数据实际上给出了一个较弱的证据证明奥利弗不在现场。

这一结果可能看起来有点让人吃惊，那么就从不同的角度来考察一下。首先考虑这种情形：另一嫌疑犯阿尔伯特的血型为 AB。凭直觉，相对于空假设 $\bar{S}$ 而言，数据确实提供了有利于理论 S' 的证据，即该嫌疑犯在现场。而且，事实上，这种情况的似然比是

$$\frac{P(D|S',\mathscr{H})}{P(D|\bar{S},\mathscr{H})} = \frac{1}{2p_{AB}} = 50 \tag{3.27}$$

现在稍微改变一下场景：设想 99% 的人是 O 型血，其他人的血型为 AB 型。假设人们的血型只有这两种。案发现场的数据和以前相同。重新考虑一下这些数据如何影响我们对血型为 O 的嫌疑犯奥利弗和血型为 AB 的嫌疑犯阿尔伯特的认识。直观上讲，我们仍然相信，发现了较罕见 AB 型血为阿尔伯特出现在现场提供了正面的证据。但是，在现场检测出 O 型血这一事实是否有利于证实奥利弗曾经出现的假设呢？如果真是如此，那就意味着不管谁是嫌疑犯，这些数据都会使他们曾在现场出现的可能性更大；而每个人都有更大的嫌疑，这显然是荒谬的。数据可以和具有任意一种血型的嫌疑犯出现在现场的假设相兼容，但是如果它们为一些理论提供正面的证据，那么它们必定要为另外一些理论提供反面的证据。

下面是有关这个问题的另一种思考方法：设想现场有 10 个人的血迹而非两个，在当地的 100 位居民中，90 位是 O 型血嫌疑犯，10 位是 AB 型血嫌疑犯。考虑一个特定的 O 型血嫌疑犯奥利弗：没有任何其他信息，在取得血液测试结果之前，他出现在案发现场的几率是十分之一，因为这 100 个嫌疑犯中只有 10 人在现场出现过。现在得到了血液测试结果，得知 10 个血迹中有 9 个是 AB 型的，1 个血迹是 O 型的。这

是否使奥利弗出现在现场的可能性更大了呢？没有，他出现在现场的几率变成了九十分之一，因为我们知道出现在案发现场的人里只有一人是 O 型血。

写出一般情况下的公式，可能会对我们的直觉有所助益。假设 n_O 个 O 型血的人和 n_{AB} 个 AB 型血的人出现在案发现场，总共有 N 个未知人，他们来自一个很大的人群，其血型为 O 型和 AB 型的概率分别为 p_O 和 p_{AB}。（还可能存在其他血型的人。）任务是计算下面两个假设的似然比：假设 S 为“O 型血嫌疑犯（奥利弗）和 $N-1$ 个未知人在案发现场留下了 N 道血痕”；假设 $\bar{S}$ 为“N 个未知人在案发现场留下了 N 道血痕”。假设 $\bar{S}$ 下的数据的概率就是从人群中随机抽取 N 个人并发现其中有 n_O 个人为 O 型血和 n_{AB} 个人为 AB 型血的概率

$$P(n_O, n_{AB} | \bar{S}) = \frac{N!}{n_O!\ n_{AB!}} p_O^{n_O} p_{AB}^{n_{AB}} \tag{3.28}$$

在假设 S 的情况下，需要其他 $N-1$ 个人的分布

$$P(n_O, n_{AB} | S) = \frac{(N-1)!}{(n_O-1)!\ n_{AB}!} p_O^{n_O-1} p_{AB}^{n_{AB}} \tag{3.29}$$

似然比为

$$\frac{P(n_O, n_{AB} | S)}{P(n_O, n_{AB} | \bar{S})} = \frac{n_O/N}{p_O} \tag{3.30}$$

这一结果很能说明问题。该似然比，即这些数据对奥利弗是否出现在案发现场的问题的贡献，仅仅取决于他的血型在观测数据中出现的频度和人群中背景频度一番比较。而不取决于在现场所发现的其他血型的数目，或它们在人群中出现的频度。如果所发现的 O 型血血痕的个数多于在假设 $\bar{S}$ 下期望的平均数，数据所给出的证据倾向于奥利弗在案发现场。相反，如果所发现的 O 型血血痕的个数少于在假设 $\bar{S}$ 下期望的平均数，那么数据会降低他出现在案发现场这一假设的概率。在 $n_O/N = p_O$ 这一特殊情况下，无论数据是否与假设 S 相兼容，数据对哪一种假设都不会提供有力证据。

3.5 习 题



习题 3.8[2] **三扇门，正常规则。**

在一场游戏秀中，参赛者被告知如下规则：

有三扇门，编号为 1，2，3。在其中的某扇门后藏有一份奖品。你选取一扇门。刚开始，游戏主持人不会打开你所选中的这扇门，而是打开另外两扇门中的一扇，并且他选择打开的是后面没有奖品的那扇门。比如，如果你首先选取 1 号门，他就会打开 2 号门和 3 号门中的一扇，而且保证他所打开的门后没有奖品。

这时，允许你重新做一次选择：你可以坚持当初的选择，也可以换成关闭着的那

扇门。然后,打开所有的门,而你获得你最后选取的门后面的东西,无论是否有奖品。

假设参赛者首先选取1号门;然后游戏主持人打开3号门,并如约展示给你看后面没有东西。参赛者应该(a)坚持1号门,或者(b)换选2号门,还是(c)换不换都一样?



习题3.9[♂] **三扇门,地震情况。**

假设重新做这个游戏,并且正当游戏主持人要打开其中的一扇门时,一场强烈的地震让大楼摇晃并猛地打开了三扇门中的一扇。这扇门恰好是3号门,而它后面又恰好没有奖品。参赛者最初选取的是1号门。

主持人理了一下他的假发,说:"这样吧,因为你最初选取的是1号门,而根据游戏规则3号门也是我可以打开的门,那么我就选择让3号门继续开着。我们继续游戏,就像什么事也没有发生。"

参赛者应该坚持选择1号门,还是换选2号门,或是换不换都一样?假设奖品是随机地摆放在一扇门后,而游戏主持人自己也不知道奖品在哪里,再假设那扇门被震开是因为地震将门闩弄断了。

[另一个类似的场景是游戏主持人糊涂了,自己忘记了游戏规则,也不知道奖品摆在哪里了,只是随机地打开未被选中的两扇门之一。他打开3号门,并发现后面没有奖品。参赛者应该选择是1号门还是2号门后有东西呢?参赛者的最佳决策取决于参赛者认为游戏主持人是否糊涂吗?]

▷**习题3.10**[♂] **不强调先验概率的另一个例子。**你去拜访一家人,这家的3个孩子都在当地一所学校上学。你一点也不知道这3个孩子的性别。在尴尬地在他们家里走动时,你不小心闯入了3间没有门牌的卧室(3个孩子每人一间)中的一间,并发现里面放了女孩子的物品,数量多得足以让你相信这一间住的是女孩。后来,你瞄了一眼写给这家家长的一封信,上面写着:"这是校长来信:我们给有男孩子在我们学校上学的家长寄上这封信,通知他们如下关于男孩的事项……"。

这两个证据源说明3个孩子中至少有一个是女孩,至少有一个是男孩。(a)两个女孩一个男孩的概率是多少?(b)两个男孩一个女孩的概率是多少?

▷**习题3.11**[♂] S太太被发现让人捅死在自家的花园里。S先生在她死后行为异常,有杀人嫌疑。在警方调查和社会记录中发现,S先生至少有9次殴打他太太的前科。检方提交这些记录作为证据,倾向于S先生犯了谋杀罪的假设。S先生高价聘请的律师说:"哦,不!根据统计,一千个打老婆的男人中只有一个会狠心杀妻①。因此,殴打太太的记录根本不是一个有力的证据。事实上,仅仅根据殴打太太这一证

① 在美国,据估计每年有二百万妇女受到其配偶的虐待。1994年,4 739名妇女成为凶杀案的牺牲品,其中1 326名(占28%)是被其丈夫或男朋友所杀害。

(数据来源:http://www.umn.edu/mincava/papers/factoid.htm, http://www.gunfree.inter.net/vpc/womenfs.htm)

据,他极不可能是杀害他太太的凶犯——只有 1/1 000 的可能性。因此你们应宣判他无罪。”

这个律师所做的殴打太太的前科与 S 先生是谋杀犯这两件事不相关的推断正确吗?还是说这个律师是一个虚伪的诡辩家?如果是后者,他的辩驳错在哪里?

[在收到对上面这段话恼火的一位律师寄来的一封信后,我想在这里额外增加一道推理题:我说 S 先生的律师可能是一个虚伪的诡辩家是否隐喻我相信所有律师都是虚伪的诡辩家?(答案:否。)]

▷**习题 3.12**[⌀] 一个袋子中装有一个筹码,已知它要么是黑的要么是白的。放入一个白筹码,摇匀袋子,并从中取出一个筹码,证实它是白的。再将袋中剩下的筹码取出,请问这是白筹码的概率是多少?[注意:在这些动作后,袋子的状态与它以前的状态完全相同。]

▷**习题 3.13**[⌀] 你搬进一所新房子;装好电话后,你比较有把握电话号码是 740511,但没有十足的把握。作为试验,你提起话筒拨了 740511 这个号码,并听到忙音。你现在是不是对你的电话号码更有把握了呢?如果是,有多大把握?

▷**习题 3.14**[⌀] 在一个游戏中,抛掷两枚硬币。如果两枚硬币中有一枚是正面,你就赢得一份奖品。要兑现奖品,你必须指着其中是正面的一枚硬币说“看,那枚硬币是正面,我赢啦”。你看着弗雷德玩这个游戏。他抛掷了两枚硬币,并指着其中是正面的一枚硬币说“看,那枚硬币是正面,我赢啦”。另一枚硬币也是正面的概率是多少?

▷**习题 3.15**[⌀] 2002 年 1 月 4 日星期五的 The Guardian 上刊登了这样一条统计命题:

在用侧边转了 250 次后,一枚比利时 1 欧元硬币转出 140 次正面和 110 次反面。伦敦经济学院的一位统计学讲师 Barry Blight 说:“我看很可疑。如果这枚硬币真的是无偏倚的,得出这样一个极端结果的概率不会高于 7%。”

但是,这些数据是否为“这是一枚有偏倚的而非均匀的硬币”这种说法提供了证据?[提示:参见式(3.22)。]

3.6 习题解答

习题 3.1 的解答 令数据为 D。假设均等先验概率

$$\frac{P(A|D)}{P(B|D)}=\frac{1}{2}\frac{3}{2}\frac{1}{1}\frac{3}{2}\frac{1}{2}\frac{2}{2}\frac{1}{2}=\frac{9}{32} \tag{3.31}$$

且 $P(A|D)=9/41$。

习题 3.2 的解答 在已知各个假设的前提下,数据的概率为

$$P(D|A)=\frac{3}{20}\frac{1}{20}\frac{2}{20}\frac{1}{20}\frac{3}{20}\frac{1}{20}\frac{1}{20}=\frac{18}{20^7} \tag{3.32}$$

$$P(D|B)=\frac{2}{20}\frac{2}{20}\frac{2}{20}\frac{2}{20}\frac{2}{20}\frac{1}{20}\frac{2}{20}=\frac{64}{20^7} \tag{3.33}$$

$$P(D|C)=\frac{1}{20}\frac{1}{20}\frac{1}{20}\frac{1}{20}\frac{1}{20}\frac{1}{20}\frac{1}{20}=\frac{1}{20^7} \tag{3.34}$$

因此

$$P(A|D)=\frac{18}{18+64+1}=\frac{18}{83};\quad P(B|D)=\frac{64}{83};\quad P(C|D)=\frac{1}{83} \tag{3.35}$$

习题 3.5 的解答

(a) $P(p_a|s=\mathrm{aba},F=3)\propto p_a^2(1-p_a)$。最可能的 p_a 值(也就是使后验概率密度最大化的数值)为 2/3。p_a 的均值为 3/5。

参见图 3.7(a)。

(b) $P(p_a|s=\mathrm{bbb},F=3)\propto(1-p_a)^3$。最可能的 p_a 值(也就是使后验概率密度最大化的数值)为 0。p_a 的均值为 1/5。

参见图 3.7(b)。

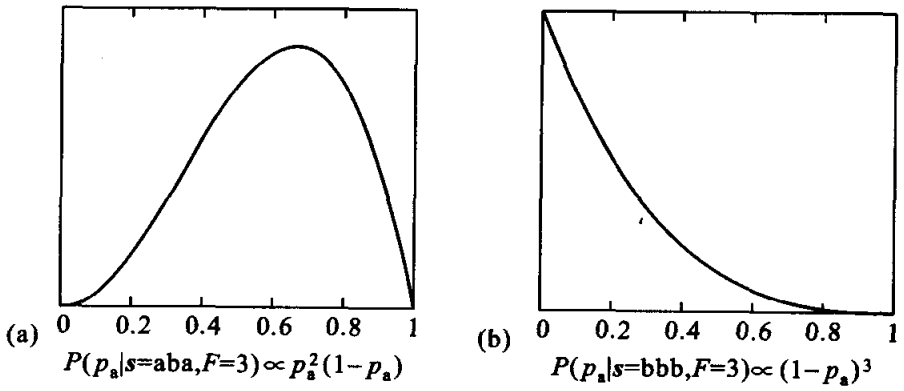


图 3.7 已知两组不同的数据集,一枚变形硬币的偏倚 p_a 的后验概率。

习题 3.7 的解答 先算出 F_a 的均值和标准差,再设定 F_a 为均值 ± 两倍标准差,得出 F_a 的一个 95% 可能范围,并计算出对数证据比的 3 个相应值。由此可得出图 3.8中的各条曲线。

习题 3.8 的解答 令 $\mathcal{H}_i$ 表示奖品在 i 号门后面这一假设。我们做出如下假设:3 个假设 $\mathcal{H}_1$、$\mathcal{H}_2$ 和 $\mathcal{H}_3$ 先验等概率,即

$$P(\mathcal{H}_1)=P(\mathcal{H}_2)=P(\mathcal{H}_3)=\frac{1}{3} \tag{3.36}$$

在选取 1 号门之后,所得数据是 $D=3$ 和 $D=2$ 之一(意思分别是 3 号门或 2 号门打开)。假设这两个可能的结果具有如下概率。如果奖品在 1 号门后面,那么主持人可以自由地做出选择;此时假设主持人在 $D=2$ 和 $D=3$ 之间随机进行选择。不然,主持人的选择是强制性的,且概率为 0 和 1。

$$\begin{vmatrix} P(D=2|\mathcal{H}_1)=1/2 & P(D=2|\mathcal{H}_2)=0 & P(D=2|\mathcal{H}_3)=1 \\ P(D=3|\mathcal{H}_1)=1/2 & P(D=3|\mathcal{H}_2)=1 & P(D=3|\mathcal{H}_3)=0 \end{vmatrix} \tag{3.37}$$

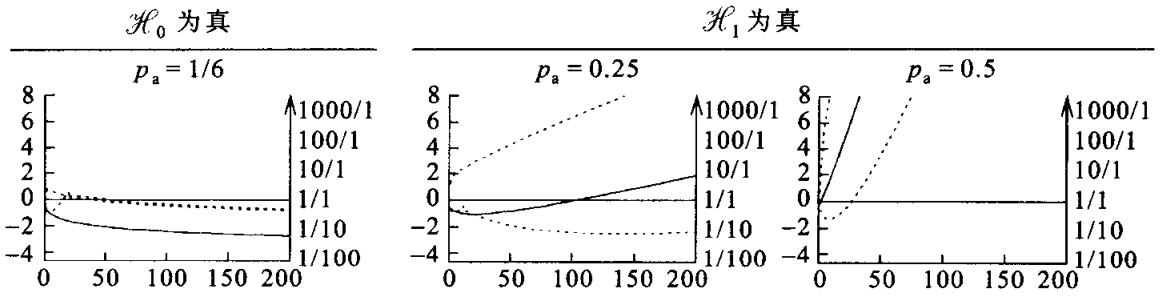


图 3.8 以 F 的函数所表示的倾向于 $\mathcal{H}_1$ 的对数证据可能值范围。左侧纵坐标为 $\log\frac{P(s|F,\mathcal{H}_1)}{P(s|F,\mathcal{H}_0)}$；右侧纵坐标为 $\frac{P(s|F,\mathcal{H}_1)}{P(s|F,\mathcal{H}_0)}$ 的值。实线所示为随机变量 F_a 取其均值 $F_a = p_a F$ 时的对数证据。点划线大致画出了若 F_a 在第 2.5 或第 97.5 个百分点上时的对数证据。

现在，使用贝叶斯定理，计算假设的后验概率：

$$P(\mathcal{H}_i|D=3) = \frac{P(D=3|\mathcal{H}_i)P(\mathcal{H}_i)}{P(D=3)} \tag{3.38}$$

$$\left|P(\mathcal{H}_1|D=3) = \frac{(1/2)(1/3)}{P(D=3)}\right| P(\mathcal{H}_2|D=3) = \frac{(1)(1/3)}{P(D=3)} \left| P(\mathcal{H}_3|D=3) = \frac{(0)(1/3)}{P(D=3)}\right| \tag{3.39}$$

分母 $P(D=3)=1/2$，因为它是这个后验分布的归一化常数。因此

$$|P(\mathcal{H}_1|D=3) = 1/3 | P(\mathcal{H}_2|D=3) = 2/3 | P(\mathcal{H}_3|D=3) = 0| \tag{3.40}$$

因此，参赛者应该换成 2 号门，以得到最大的获奖机会。

许多人觉得这个结果令人吃惊。有两种方法可以让它更直观。一种方法是找个朋友一起玩 30 次，并记下切换获奖的频度。另一种方法是，可以进行一个思维实验：游戏中有一百万扇门。现在的规则是参赛者选取一扇门，然后游戏主持人打开 999 998扇门而不暴露奖品的位置，只剩下参赛者所选取的门和另一扇门还关着。参赛者可以选择坚持或切换。假设参赛者面对一百万扇门，其中只有 1 号门和 234 598 号门没打开，而 1 号门是参赛者的初始选择。你认为奖品会在哪里呢？

习题 3.9 的解答 如果 3 号门是让地震震开的，做出的推理就不同了，尽管情况看起来一样。数据的性质和数据的概率二者都不同了。首先，可能的数据结果是任何序号的门都可能被震开。可以对 8 种可能的结果进行标号：

$$\boldsymbol{d} = (0,0,0),(0,0,1),(0,1,0),(1,0,0),(0,1,1),\cdots,(1,1,1)$$

第二，在地震震开一扇或多扇门后，奖品可能暴露出来。因此，数据 D 包括 $\boldsymbol{d}$ 的数值以及奖品是否已暴露的命题。很难说清这些结果的概率是多少，因为它们取决于我们对门闩可靠性的信心和地震的性质。不过，无需一一对每个 $\boldsymbol{d}$ 给出 $P(\boldsymbol{d}|\mathcal{H}_i)$ 的值，也可能得出期望的后验概率。有关系的只是实际出现的 D 值所对应的 $P(D|\mathcal{H}_1)$，$P(D|\mathcal{H}_2)$ 和 $P(D|\mathcal{H}_3)$ 的相对值。[这是在 2.3 节遇到过的似然原理。]实际

出现的 D 值是“$\boldsymbol{d}=(0,0,1)$，没暴露奖品”。首先，显然有 $P(D|\mathscr{H}_3)=0$，因为“没暴露奖品”这一数据和 $\mathscr{H}_3$ 不相容。现在，假设参赛者选取 1 号门，概率 $P(D|\mathscr{H}_1)$ 和 $P(D|\mathscr{H}_2)$ 相比如何？假设地震不受游戏参赛者决策的影响，由对称性可知，这两个量必须相等。我们不知道 3 号门有多大可能性会从门枢上掉落，但不管其可能性有多大，无论奖品在 1 号门后面还是 2 号门后面，这个可能性不会变。因此，如果 $P(D|\mathscr{H}_1)$ 和 $P(D|\mathscr{H}_2)$ 相等，有

$$\left|\begin{array}{c|c|c} P(\mathscr{H}_1|D)=\dfrac{P(D|\mathscr{H}_1)(1/3)}{P(D)} & P(\mathscr{H}_2|D)=\dfrac{P(D|\mathscr{H}_2)(1/3)}{P(D)} & P(\mathscr{H}_3|D)=\dfrac{P(D|\mathscr{H}_3)(1/3)}{P(D)} \\ =1/2 & =1/2 & =0 \end{array}\right| \tag{3.41}$$

此时，这两种可能的假设是等可能性的。

如果假设主持人知道奖品在哪里，并可能舞弊，那么答案还要做进一步修改，因为我们将主持人的话也看做数据的一部分了。

困惑了？确保你弄明白这两个游戏秀问题，这很值得。别担心，第一次碰到第二个问题时，我自己也摔了跟头。

当你遇到一个令人困惑的概率问题时，有一个通用规则很受用：

> 把全体事物的概率写出来。(Steve Gull)

由这个联合概率，可以机械地得到任何想要的推理(图 3.9)。

奖品的位置

地震打开了哪扇门	1号门	2号门	3号门
none	$\frac{p_{\text{none}}}{3}$	$\frac{p_{\text{none}}}{3}$	$\frac{p_{\text{none}}}{3}$
1			
2			
3	$\frac{p_3}{3}$	$\frac{p_3}{3}$	$\frac{p_3}{3}$
1,2			
1,3			
2,3			
1,2,3	$\frac{p_{1,2,3}}{3}$	$\frac{p_{1,2,3}}{3}$	$\frac{p_{1,2,3}}{3}$

图 3.9 对于第二个 3 扇门问题，假设刚发生了地震，全体事物的概率。这里，p_3 表示只有 3 号门被地震震开的概率。

习题 3.11 的解答 这个律师所引用的统计数据表明一个随机选定的殴打太太者同时也会谋杀他太太的概率。在已知妻子已被谋杀的前提下，这个丈夫是谋杀犯的概率则是一个完全不同的量。

要推导出后者，需要对这个妻子是被其他人谋杀的概率做进一步假设。如果她生活在一个任意谋杀屡见不鲜的社区，那么这个概率比较大，且在缺乏其他证据的条件下，她丈夫是谋杀犯的后验概率可能不会太大。但是在比较安全的社区，如果有人被发现谋杀，最可能的凶犯也许就在死者最亲近的人中。

下面借助习题 3.11 中的统计数据，计算出几个有说明性的数字。用 $m=1$ 表示一位妇女被谋杀这一命题；用 $h=1$ 表示她是被她丈夫谋杀这一命题；并用 $b=1$ 表示他在她被谋杀的前一年殴打过她这一命题。“她被其他人谋杀”这一命题就可记做 $h=0$。要计算后验概率 $P(h=1 \mid b=1, m=1)$，需要先定义 $P(h \mid m=1)$，$P(b \mid h=1, m=1)$ 和 $P(b=1 \mid h=0, m=1)$。由统计数据，可解读出 $P(h=1 \mid m=1)=0.28$。如果一亿妇女中有两百万遭到殴打，则有 $P(b=1 \mid h=0, m=1)=0.02$。最后，需要知道 $P(b \mid h=1, m=1)$ 的值，即如果一个男人谋杀了他的妻子，那么在此之前他从未向她下过毒手的可能性有多大？我想这是相当罕见的，因此 $P(b=1 \mid h=1, m=1)$ 可能是0.9或更大。

由贝叶斯定理，有

$$P(h=1 \mid b=1, m=1)=\frac{0.9 \times 0.28}{0.9 \times 0.28+0.02 \times 0.72} \simeq 95\% \tag{3.42}$$

揭穿习题 3.11 中那位律师虚伪性的一种办法，就是使用跟他相同的逻辑结构，构造一个明显荒谬的论据。比如，可以学着律师说“S 太太被谋杀了，而且是在下午 4:02 和 4:03 之间被谋杀的。从统计学角度来看，一百万个殴打太太者中也只有一位会在下午 4:02 和 4:03 之间真正谋杀他太太。因此，殴打太太这一事实完全算不上是有力的证据。事实上，仅仅提供殴打太太这一证据，他极不可能会这样谋杀他的太太——可能性只有 1/1 000 000。”

习题 3.13 的解答 存在两种假设。$\mathcal{H}_0$：你的号码是 740511；$\mathcal{H}_1$：它是其他号码。数据 D 为“我拨 740511 这个电话号码，听到的是忙音”。在已知每个假设的前提下，数据 D 的概率是多少？如果你的号码是 740511，我们肯定会听到忙音的概率是

$$P(D \mid \mathcal{H}_0)=1$$

而另一方面，如果 $\mathcal{H}_1$ 为真，拨打这个电话号码会听到忙音的概率小于 1，因为另外一些结果也是可能的（比如，振铃声或该号码无法接通的信号）。概率 $P(D \mid \mathcal{H}_1)$ 的值取决于像你自己电话号码的一个随机电话号码是有效号码的概率 α，及你拨打一个有效号码时会听到忙音的概率 β。

根据我的电话簿，我估计剑桥大约有 75 000 个有效电话号码，长度都是 6 位数。因此，一个随机的 6 位数字是一个有效电话号码的概率约为 $75\,000/10^6=0.075$. 如果从随机选择中排除以 0、1 和 9 开头的号码，概率 α 约为 $75\,000/700\,000 \simeq 0.1$。如

果假设电话号码是成组分配的，一个误记的号码要比一个随机选择号码更可能是有效号码；因此假定 $\mathscr{H}_1$ 为真，我们猜测的号码是有效号码的概率 α 可能比 0.1 大。无论如何，α 会是 0.1 ~ 1 之间的某个数。我们保持这一不确定概率，并看看它最后会产生多大影响。

你拨打一个有效号码听到忙音的概率 β，等于尝试打电话时你认为正在通话或没挂好电话所占的比例。这一比例在不同的城市会有所不同，而且在同一天的不同时段也会有所不同。在剑桥，我估计白天约有 1% 的电话在通话。在凌晨 4 点钟，这个比例可能是 0.1% 或更低。

概率 $P(D|\mathscr{H}_1)$ 等于 α 和 β 之积，大概是 $0.1 \times 0.01 = 10^{-3}$。根据我们的估计，当你拨打一个随机电话号码，听到忙音的可能性是千分之一；如果有效电话是成组分配的话，这个可能性增到百分之一；如果是在凌晨拨号的话，这个可能性降到万分之一。

这些数据怎样影响你对电话号码的置信度呢？后验概率比为似然比乘以先验概率比：

$$\frac{P(\mathscr{H}_0|D)}{P(\mathscr{H}_1|D)} = \frac{P(D|\mathscr{H}_0)P(\mathscr{H}_0)}{P(D|\mathscr{H}_1)P(\mathscr{H}_1)} \tag{3.43}$$

似然比约为 100 比 1 或 1 000 比 1，因此后验概率比对 $\mathscr{H}_0$ 的倾向程度在 100 或 1 000 的因子范围内变化。如果 $\mathscr{H}_0$ 的先验概率为 0.5，那么后验概率为

$$P(\mathscr{H}_0|D) = \frac{1}{1 + \dfrac{P(\mathscr{H}_1|D)}{P(\mathscr{H}_0|D)}} \simeq 0.99 \text{ 或 } 0.999 \tag{3.44}$$

习题 3.15 的解答 对模型 $\mathscr{H}_0$（硬币是均匀的）和 $\mathscr{H}_1$（硬币是偏倚的）进行比较，这里偏倚的先验概率设定为均匀分布 $P(p|\mathscr{H}_1) = 1$。[使用均匀先验在我看来是合理的，因为我知道有些硬币，比如美国分币，在用侧边转动时会产生严重的偏倚；因此出现像 $p = 0.01$ 或 $p = 0.1$ 或 $p = 0.95$ 之类的情况，也就不足为奇。]

在我提到 $\mathscr{H}_0$（硬币是均匀的）时，学究们可能会说："想一想硬币是均匀的就让人觉得荒谬！任何一枚硬币都必定或多或少存在着偏倚"。当然我完全赞成。因此学究们可以很好地将 $\mathscr{H}_0$ 理解为"硬币在千分之一的允许范围内是均匀的，即 $p \in 0.5 \pm 0.001$"。

似然比为

$$\frac{P(D|\mathscr{H}_1)}{P(D|\mathscr{H}_0)} = \frac{\dfrac{140!\ 110!}{251!}}{1/2^{250}} = 0.48 \tag{3.45}$$

因此数据几乎没有为任何一方提供有利证据；事实上它们给出了微弱证据（2 对 1）倾向于 $\mathscr{H}_0$！如图 3.10 所示。

偏倚的拥戴者会提出反对意见："不，不，你那个愚蠢的均匀先验概率并不代表我对硬币偏倚程度的看法，我觉得只有较小的偏倚"。为了对 $\mathscr{H}_1$ 尽量宽容，让我们来看看如果有预知性地设定先验概率之后，情况会如何。假定先验概率具有形式

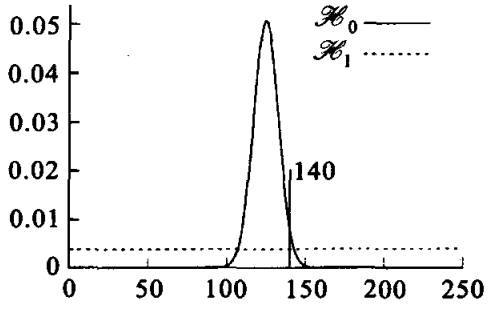


图 3.10 在已知两种假设(即硬币是均匀的,及硬币是偏倚的且偏倚的先验分布是均匀的)的前提下,正面次数的概率分布。结果($D=140$ 次正面)提供了微弱证据倾向于 $\mathscr{H}_0$,即硬币是均匀的这一假设。

$$P(p|\mathscr{H}_1,\alpha)=\frac{1}{Z(\alpha)}p^{\alpha-1}(1-p)^{\alpha-1},\quad Z(\alpha)=\frac{\Gamma(\alpha)^2}{\Gamma(2\alpha)} \tag{3.46}$$

(Beta 分布,若设 $\alpha=1$,则退化成原来的均匀先验情形)。通过调节 α,$\mathscr{H}_1$ 对 $\mathscr{H}_0$ 的似然比

$$\frac{P(D|\mathscr{H}_1,\alpha)}{P(D|\mathscr{H}_0)}=\frac{\Gamma(140+\alpha)\Gamma(110+\alpha)\Gamma(2\alpha)2^{250}}{\Gamma(250+2\alpha)\Gamma(\alpha)^2} \tag{3.47}$$

会增大一点。如图 3.11 所示为几个不同 α 值的情形。哪怕是选择最有利的 $\alpha(\alpha\simeq 50)$,所得的似然比也只是 2 比 1 地倾向于 $\mathscr{H}_1$。

α	$\frac{P(D\|\mathscr{H}_1,\alpha)}{P(D\|\mathscr{H}_0)}$
0.37	0.25
1.0	0.48
2.7	0.82
7.4	1.3
20	1.8
55	1.9
148	1.7
403	1.3
1 096	1.1

图 3.11 选择不同先验分布超参数 α 时的似然比。

总之,数据不是“非常可疑”。可以将它们构造成至多给出证据,以 2 比 1 倾向于两个假设中的这一个或另一个。

这些虚弱的似然比是否归咎于先验概率受到过多的限制呢?是否有办法得出一个“非常可疑”的结论?根据似然比,与数据匹配得最好的先验概率是以概率 1 将 p

设定为$f\equiv140/250$的先验概率。将这个模型称为$\mathscr{H}_*$。似然比为$P(D|\mathscr{H}_*)/P(D|\mathscr{H}_0)=2^{250}f^{140}(1-f)^{110}=6.1$。因此，数据可以构造成反对无偏假设的最强证据只是6比1。

注意到“采样理论”统计学所得出的荒谬误导性答案，比如在习题中刚求解的7%的p值，让我们更进一步。如果对数据集做一个小小的改动，将250次抛掷中出现正面的次数从140增加到141，可以看到p值会降到神秘值0.05之下(p值为0.049 7)。采样理论统计学者会很高兴地叫嚷：“得出一个像141个正面这种极端情况的概率低于0.05，因此可以在重要水平5%上拒绝空假设”。对几个α值的正确答案如图3.12所示。在这张表中值得强调的数值，首先是$\mathscr{H}_1$使用标准均匀先验时的似然比，它是1:0.61，倾向于空假设$\mathscr{H}_0$。其次，在$\mathscr{H}_1$看来，选择最有利的α值，也只能得出一个约为2.3:1的似然比倾向于$\mathscr{H}_1$。

α	$\frac{P(D'\|\mathscr{H}_1,\alpha)}{P(D'\|\mathscr{H}_0)}$
0.37	0.32
1.0	0.61
2.7	1.0
7.4	1.6
20	2.2
55	2.3
148	1.9
403	1.4
1 096	1.2

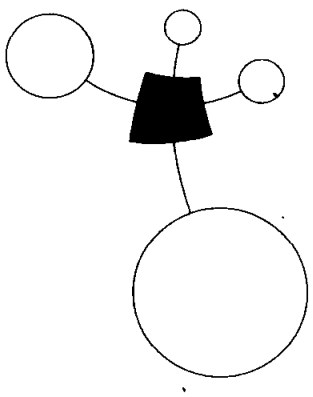
图3.12 当数据是250次试验中出现$D'=141$次正面时，选择不同的先验分布超参数α时的似然比。

当心！一个大小为0.05的p值往往被解读成以筹码为20比1的几率不利于空假设。而在这种情况下的真实情况是证据或者稍微有利于空假设，或者至多以2.3比1不利于它，具体情况视所选先验分布而定。

要非常谨慎地对待经典统计学中的p值和“重要水平”。避开它们！训诫到此结束。

第一部分

数据压缩



关于第 4 章

在本章中，讨论如何度量一次随机实验结果所包含的信息。

本章中有一些难点。如果你觉得一些数学细节很难懂，请跳过这些难点，继续往下学，没有这些工具，你仍然可以弄懂第 5 章和第 6 章中的内容。

在阅读第 4 章之前，应该先阅读第 2 章并完成习题 2.21 ~ 2.25 和习题 2.16 以及下面的习题 4.1。

下面的习题是为了帮助你思考如何度量所含信息。



习题 4.1[2] 请在阅读第 4 章之前解答此题。

给你 12 个球，其中 11 个球重量均等，而另一个球要么比其他球重，要么比其他球轻。你可以使用一架 2 - 托盘的天平。每次使用天平时，可将这些球中的一部分放在左托盘上，并取相同数目的球放在右托盘上，然后撳按钮来开始称重。有三种可能的结果：重量相等，左边球重，或者左边球轻。你的任务是设计一种方案，用尽量少的称量次数找出哪一个球不同于其他球，并确定它是比其他球重还是轻。

思考这个问题时，可能会觉得从以下几个方面来考虑会有所帮助。

(a) 怎么才能度量信息？

(b) 当你确定了哪一个是异常球，并知道了它比其他球重还是轻时，你获得了多少信息？

(c) 一旦你设计好一种方案，针对每一种可能出现的称重结果，画出一个树状图，以表明接下来你要怎样称重。在树的每一个结点处，你已经从结果获得多少信息量以及还可进一步获得多少信息量？

(d) 当获知以下情况时，你获得了多少信息：(i) 一个抛掷硬币的状态；(ii) 两个抛掷硬币的状态；(iii) 一个 4 面体骰子的转动结果？

(e) 在第一步称量时，若用其中 6 个球和另外 6 个球一起称重，能获得多少信息？若最初用其中 4 个球和另外 4 个球一起称重，并留下 4 个球，又能获得多少信息？

记　号

$x \in \mathcal{A}$　x 是集合 $\mathcal{A}$ 的一个元素

$\mathcal{S} \subset \mathcal{A}$　$\mathcal{S}$ 是集合 $\mathcal{A}$ 的一个子集

$\mathcal{S} \subseteq \mathcal{A}$　$\mathcal{S}$ 是集合 $\mathcal{A}$ 的一个子集，或者等于 $\mathcal{A}$

$\mathcal{V} = \mathcal{B} \cup \mathcal{A}$　$\mathcal{V}$ 是集合 $\mathcal{B}$ 和 $\mathcal{A}$ 的并集

$\mathcal{V} = \mathcal{B} \cap \mathcal{A}$　$\mathcal{V}$ 是集合 $\mathcal{B}$ 和 $\mathcal{A}$ 的交集

$|\mathcal{A}|$　集合 $\mathcal{A}$ 中元素的个数

第 4 章

信源编码定理

4.1 如何度量一个随机变量的信息量?

在接下来的几章中,将讲述概率分布和随机变量。大多数情况我们采用一些随意的记号,但偶尔也需要采用精确的记号。以下是第 2 章中确定曾使用的记号。

总体 X 是一个 3 - 元组$(x,\mathcal{A}_X,\mathcal{P}_X)$,其中结果 x 是一个随机变量的值,它在可能值集合 $\mathcal{A}_X=\{a_1,a_2,\cdots,a_i,\cdots,a_I\}$ 上取值,其概率 $\mathcal{P}_X=\{p_1,p_2,\cdots,p_I\}$,其中 $P(x=a_i)=p_i,p_i\geqslant 0$ 且有 $\sum_{a_i\in\mathcal{A}_X}P(x=a_i)=1$。

如何来度量这样一个总体中的一个结果 $x=a_i$ 的信息量呢? 在这一章中,考察如下论断

1. 香农信息量

$$h(x=a_i)\equiv\log_2\frac{1}{p_i} \tag{4.1}$$

是结果 $x=a_i$ 所含信息量的一种合适度量。

2. 总体的熵

$$H(X)=\sum_i p_i\log_2\frac{1}{p_i} \tag{4.2}$$

是总体平均信息量的一种合适度量。

如图 4.1 所示是概率为 p 的一个结果的香农信息量,它是一个关于 p 的函数。一个结果的概率越小,其香农信息量越大。图 4.1 也给出了二进制熵函数

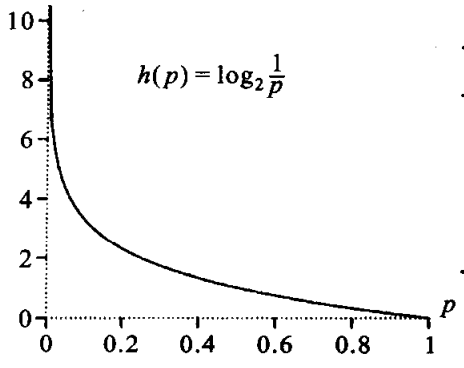
$$H_2(p)=H(p,1-p)=p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p} \tag{4.3}$$

它是符号集和概率分布分别为 $\mathcal{A}_X=\{a,b\}$ 和 $\mathcal{P}_X=\{p,(1-p)\}$ 的总体 X 的熵。

独立随机变量的信息量

为什么信息量会与 $\log 1/p_i$ 有关呢? 而不是关于 p_i 的其他函数呢? 稍后会分析这个问题,但在此之前,注意到这个特定函数 $h(x)=\log 1/p(x)$ 的一个良好特性。

假设想知道两个相互独立的随机变量 x 和 y 的值。这里,相互独立的定义是概



p	h(p)	$H_2(p)$
0.001	10.0	0.011
0.01	6.6	0.081
0.1	3.3	0.47
0.2	2.3	0.72
0.5	1.0	1.0

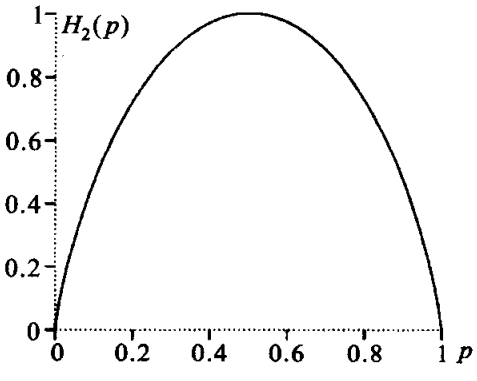


图 4.1 香农信息量 $h(p)=\log_2\frac{1}{p}$ 和以 p 的函数所表示的二进制熵函数。

$$H_2(p)=H(p,1-p)=p\log_2\frac{1}{p}+(1-p)\log_2\frac{1}{1-p}。$$

率分布可以分解成一个乘积

$$P(x,y)=P(x)P(y) \tag{4.4}$$

从直观上讲，我们可能希望“所获得的信息量”的度量具有可加性，也就是说，对于独立随机变量 x 和 y，从二者所获得的信息量应等于单独从 x 和 y 所获信息量之和。

结果 x 和 y 的香农信息量是

$$h(x,y)=\log\frac{1}{P(x,y)}=\log\frac{1}{P(x)P(y)}=\log\frac{1}{P(x)}+\log\frac{1}{P(y)} \tag{4.5}$$

所以，它确实满足

$$h(x,y)=h(x)+h(y)\quad x\text{ 和 }y\text{ 相互独立} \tag{4.6}$$



习题 4.2[1] 如果 x 和 y 相互独立，结果 x 和 y 的熵满足

$$H(X,Y)=H(X)+H(Y) \tag{4.7}$$

简而言之，对于独立变量，熵具有可加性。

现在先用一些例子来阐述这些思想；然后在第 4.4 节、第 5 章和第 6 章，证明香农信息量和熵与描述实验结果所需的比特数有关。

称重问题：设计能提供信息的实验

你是否已经解答了称重问题（习题 4.1）？你确定吗？注意在天平的 3 次使用中（其读数为“左边重”、“右边重”或“平衡”），可以设想到的结果有 $3^3=27$ 种；但是，可能出现的现实状态有 24 种：异常球可能是 12 个球中的任意一个，它可能比其他球重，也可能比其他球轻。因此在原理上，这个问题可以通过 3 次称量（但不是两次）来解决，因为 $3^2<24$。

如果你知道如何通过 3 次称重确定异常球并且获知它比其他球重还是轻，可继

续阅读以下内容了。如果你尚未找到一种策略总能用 3 次称量来解决这个问题,笔者建议你再思考习题 4.1。

为什么你的策略是最优的?能让你尽快获得有用信息的系列称量是怎样的?答案是在优化方案中的每一步骤中,尽可能让 3 种可能结果("左边重"、"右边重"和"平衡")等概率出现。一个最优解如图 4.2 所示。

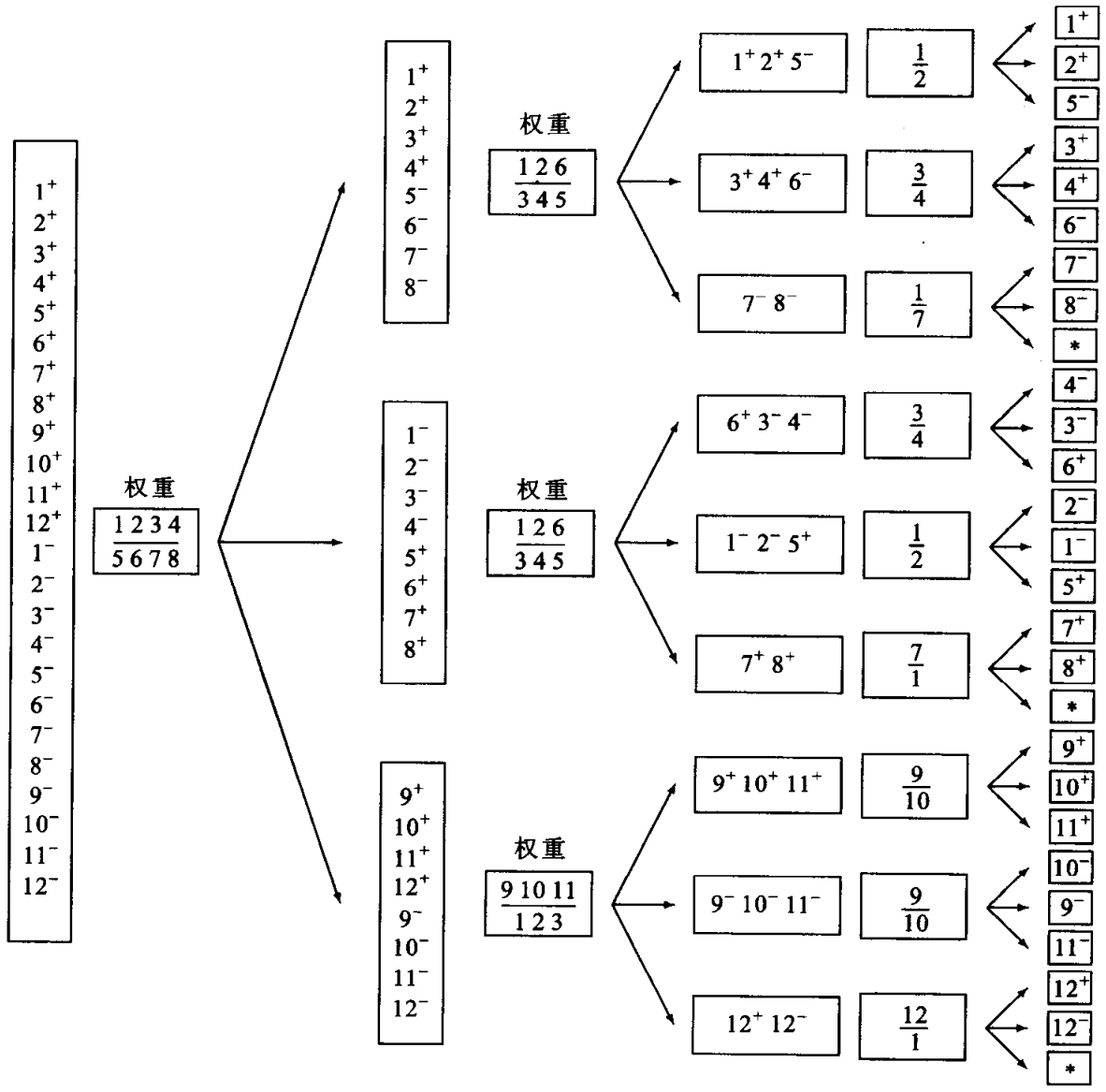


图 4.2 称重问题的一个最优解。在每一步中有两个盒子:左边盒子给出哪一种假设仍然有可能出现;右边盒子给出下一次称重所涉及的球。24 种假设分别用 $1^+,\cdots,12^-$ 表示,比如,1^+ 表示 1 号球是异常球而且它比其他球重。通过列出两个托盘中球的号码,并用一条横线隔开,来表示一次称重;例如,在第一次称重中,1、2、3 和 4 号球放在左边托盘中,5、6、7 和 8 号球放在右边托盘中。3 个一组的箭头中,上面箭头指向左边重的情况,中间箭头指向右边重的情况,下面箭头指向两边平衡的情况。3 个标有 * 的点表示不可能发生的情况。

次优策略不能在每一步中都等概率地得到结果;例如在第一步中,若用1~6号球和7~12号球来进行称重,则这两组球不可能平衡,只能得到"左边重"或者"右边重"这两种可能的结果。这样一个二进制结果只能排除一半的可能假设,因此使用这些结果的方法有时必须花费更长的时间才能得到正确答案。

"结果应该尽可能等概率出现"这一洞察有助于人们更容易地找到最优策略。第一次称重必须将24种假设分成3组,每组8种。第二次称重必须将假设按照3:3:2的比例分开。

因此可以得出如下结论:

> 如果一个随机实验在所有结果上的概率分布为均匀分布,可以保证它可提供的信息量最大。

这个结论和习题2.25所证明的熵特性相吻合:当所有结果等概率地分布,即$p_i = 1/|\mathcal{A}_X|$时,总体X的熵最大。

猜谜游戏

在"20个问题"游戏中,一个人想好一件物品,由另一个人通过询问他关于这件物品的一些问题来猜出谜底。这些问题只能用"是"和"否"来回答,例如,"它是有生命的吗?"或者"是人类吗?"目标是为了用尽可能少的问题来确定谜底。玩这类游戏最好的方法是什么?为了简单起见,假设我们玩的是"20个问题"游戏中相当笨拙的一种称为"63"的游戏。

例4.3 "63"游戏。最少需要问多少个"是/否"问题,才能确定一个0~63之间的整数?

从直观上看,最好的问题是连续将64种可能性平均分成相同大小的集合。那么,6个问题就足够了。一种合理的方法是按照如下顺序来提问。

1: $x \geq 32$ 吗?

2: $x \bmod 32 \geq 16$ 吗?

3: $x \bmod 16 \geq 8$ 吗?

4: $x \bmod 8 \geq 4$ 吗?

5: $x \bmod 4 \geq 2$ 吗?

6: $x \bmod 2 = 1$ 吗?

[记号$x \bmod 32$读做"x模32",表示当x被32除时的余数。例如,$35 \bmod 32 = 3$,$32 \bmod 32 = 0$。]

如果将{是,不是}翻译成{1,0},这些问题的答案就给出了x的二进制扩展,例如$35 \Rightarrow 100011$。 □

这个例子中,这些结果的香农信息量是多少呢?若假定所有x值的可能性相等,则这些问题的答案是相互独立的,并且每个答案的香农信息量为$\log_2 1/0.5 = 1$比特;所获得的总香农信息量是6比特。此外,从这些问题中得到的数x是一个6比特二

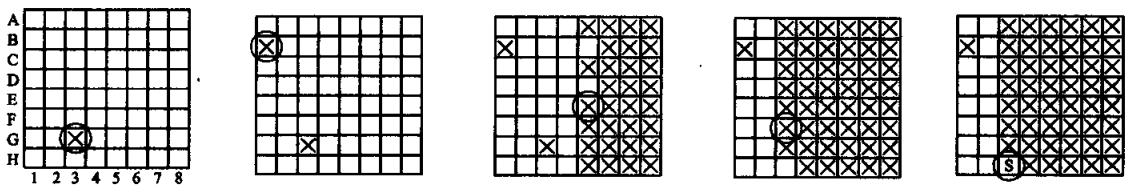
进制数。我们提问的策略也就定义了一种将随机变量 x 编码为二进制文件的方法。

迄今为止,香农信息量看来合乎逻辑:它度量了对 x 进行编码的二进制文件的长度。但是,我们尚未研究结果具有不等概率的总体。香农信息量在这种情况下也合乎逻辑吗?

潜艇游戏:一个比特可以传达多少比特的信息?

在战舰游戏中,游戏双方都把舰队隐藏在用方格表示的海域中。每一轮游戏,一方朝对方海域中的一个方格开火,试图击中对方的船只。对于一个选中的方格(如"G3")所做出的反应是"没有击中"、"击中"或者"击中且摧毁"。

"潜艇"是一种比较无趣的战舰游戏,双方在一个 8×8 的网格中,只藏匿一艘潜艇。图 4.3 给出了做这个游戏过程中的一些图片:圆圈表示被瞄准的方格;×表示没被击中的方格,x = n;潜艇在第 49 次射击尝试时被击中(结果为 x = y,图中用符号 s 表示)。



步数	1	2	32	48	49
问题	G3	B1	E5	F3	H3
结果	x = n	x = n	x = n	x = n	x = y
$P(x)$	$\frac{63}{64}$	$\frac{62}{63}$	$\frac{32}{33}$	$\frac{16}{17}$	$\frac{1}{16}$
$h(x)$	0.022 7	0.023 0	0.044 3	0.087 4	4.0
总信息	0.022 7	0.045 8	1.0	2.0	6.0

图 4.3 潜艇游戏。潜艇在第 49 次被击中。

某一玩家的每一次射击定义一个总体。两个可能出现的结果是{y,n},对应于"击中"和"未被击中",其概率依赖于棋盘上的状态。刚开始,P(y) = 1/64 且 P(n) = 63/64。如果第一次射击未击中,第二次射击时有 P(y) = 1/63 且 P(n) = 62/63。如果前两次都未击中,第三次射击时有 P(y) = 1/62 且 P(n) = 61/62。

从一个结果 x 中所获得的香农信息量是 $h(x) = \log(1/P(x))$。如果很幸运的话,第一次就击中潜艇,则

$$h(x) = h_{(1)}(\mathrm{y}) = \log_2 64 = 6 \text{ 比特} \tag{4.8}$$

现在,看起来可能会觉得有点奇怪,一个二进制结果可以传达 6 比特信息。但是,我们得知了隐藏位置,而它可以是 64 个方格中的任意一个;因此,通过这一个幸运的二进制问题,确实得到了 6 比特信息。

如果第一次射击未击中，又怎样呢？从这个结果中所得的香农信息是

$$h(x)=h_{(1)}(\mathrm{n})=\log_2\frac{64}{63}=0.022\ 7\ \text{比特} \tag{4.9}$$

这是否合理呢？答案就不是那么显而易见了。让我们继续讲解。如果第二次射击也没有击中，第二次结果的香农信息量为

$$h_{(2)}(\mathrm{n})=\log_2\frac{63}{62}=0.023\ 0\ \text{比特} \tag{4.10}$$

如果连续32次都未射中（每次都射向一个新的方格），所获得的总香农信息是

$$\begin{aligned}&\log_2\frac{64}{63}+\log_2\frac{63}{62}+\cdots+\log_2\frac{33}{32} \\ &=0.022\ 7+0.023\ 0+\cdots+0.043\ 0=1.0\ \text{比特}\end{aligned} \tag{4.11}$$

为什么是这个整数呢？还有，从中我们学到了些什么？现在知道潜艇不在所射过的32个方格中；得知这一事实与玩一个“63”游戏（例4.3）一样，第一个问题“x是我射击方格的32个数之一吗？”这个问题的答案就会排除一半的假设，所以它提供了1比特的信息。

在连续48次不成功的射击之后，所获得的信息便是2比特：未知位置越来越少，只有最初假设空间的四分之一了。

如果在第49次射击时击中了潜艇，此时还剩下16个方格，情况又如何呢？这个结果的香农信息量为

$$h_{(49)}(\mathrm{y})=\log_2 16=4.0\ \text{比特} \tag{4.12}$$

则所有结果的总香农信息量是

$$\begin{aligned}&\log_2\frac{64}{63}+\log_2\frac{63}{62}+\cdots+\log_2\frac{17}{16}+\log_2\frac{16}{1} \\ &=0.022\ 7+0.023\ 0+\cdots+0.087\ 4+4.0=6.0\ \text{比特}\end{aligned} \tag{4.13}$$

因此，一旦知道潜艇在哪里，所获得的总香农信息量为6比特。

无论何时击中潜艇，这个结果始终成立。如果在还剩n个方格时击中潜艇（比如式(4.13)中的n为16），则所获得的总信息量为

$$\begin{aligned}&\log_2\frac{64}{63}+\log_2\frac{63}{62}+\cdots+\log_2\frac{n+1}{n}+\log_2\frac{n}{1} \\ &=\log_2\left\lceil\frac{64}{63}\times\frac{63}{62}\times\cdots\times\frac{n+1}{n}\times\frac{n}{1}\right\rceil=\log_2\frac{64}{1}=6\ \text{比特}\end{aligned} \tag{4.14}$$

到目前为止，我们从这里例子中学到了什么？笔者认为潜艇例子颇有说服力地表明了香农信息量是一个有意义的信息量度量。“63”数字游戏表明香农信息量与随机实验结果的编码文件的大小紧密相关，因此也暗示了它与数据压缩有一定关联。

如果你还不太信服，就再看一个例子。

Wenglish 语言

Wenglish是与英语很相似的一种语言。Wenglish句子由Wenglish字典中随机抽

取的单词组成,字典中有 $2^{15}=32\ 768$ 个单词,每个单词的长度都为 5 个字符。Wenglish 字典中的每个单词由从 a 到 z 随机抽取的 5 个字母组成,而这些字母所服从的概率分布如图 2.1 所示。

如图 4.4 所示为字典中按照字母次序排列的一些词条。注意字典中的单词数目(32 768)远少于 5 个字母所组成单词的总数($26^5 \simeq 12\ 000\ 000$)。

由于字母 z 的概率大约是 1/1 000,字典中只有 32 个单词以字母 z 开头。与之相对,字母 a 的概率是 0.062 5,有 2 048个单词以字母 a 开头;其中 2 个以 az 开头,128 个以 aa 开头。

假设正在阅读一篇 Wenglish 文档,让我们来讨论在获得这些字符时它们所包含的香农信息量。如果一次得到一个单词,则每 5 个字符的单词的香农信息量是log32 768 = 15 比特,因为 Wenglish 文档等概率地使用所有单词。因此每个字符的平均信息量是 3 比特。

现在来讨论如果我们阅读文档时一次看一个字符时的信息量。例如,一个单词的第一个字母是 a,香农信息量为 log1/0.062 5≃4比特,若第一个字母是 z,香农信息量是 log1/0.001 ≃10 比特。因此从第一个字母所获信息量的变化就很大。而一个单词中 5 个字符的总信息量恰是 15 比特;因此紧跟着第一个字母 z 的字母与紧跟着第一个字母 a 的字母相比,所含平均信息量要低一些。一个罕见的词首字母,如 z,比一个常用的词首字母所传达的信息量确实大得多。

1	aaail
2	aaaiu
3	aaald
⋮	⋮
129	abati
⋮	⋮
2 047	azpan
2 048	aztdn
⋮	⋮
⋮	⋮
16 384	odrcr
⋮	⋮
⋮	⋮
32 737	zatnt
⋮	⋮
32 768	zxast

图 4.4 Wenglish 字典

类似地,在英语中,如果一个单词以稀有字符开头(如xyl…),则我们经常能够立即确定整个单词;而那些以常用字母开头的单词(如 pro…),则需要更多的字符才能识别。

4.2 数据压缩

前述例子证明一个结果的香农信息量是其信息量的自然度量。不太可能的结果比可能的结果所传达的信息量大。现在通过考虑描述实验结果需要多少比特来讨论信源的信息量。

如果能够表明可将特定源数据压缩成每个信息符号为 L 比特的文件,并且能够可靠地恢复数据,那么就说这个信源的平均信息量最多为每个符号 L 比特。

例:文本文件压缩

一个文件由一系列字节组成。一个字节由8比特组成[这里用"比特"表示"具有两个值的符号",不要和信息量的单位相混淆。],可以取从0到255的十进制数。典型的文本文件由ASCII字符集(十进制数从0到127)组成。字符集仅仅用到一个字节中的7比特。

▷**习题4.4**[1] 假定文件是ASCII码文件,文件大小可被压缩多少?这种缩减是如何实现的?

因为甚至在我们浏览该文件之前,已经知道每8比特中的1个比特,从直观上看,似乎可以断定:该ASCII文件包含与其同等大小的任意文件的7/8倍信息量。这是一个简单的冗余例子。大多数数据源具有更大的冗余性:英文文本文件对ASCII字符的使用是不均等的;某些字母对的使用频率更高;在特定的上下文中,通过对文章的语义理解可以预测完整的单词。

定义信息量度量的一些简单数据压缩方法

度量随机变量信息量的一种方法是简单地计算可能结果的数目,$|\mathcal{A}_X|$(集合$\mathcal{A}$中的元素数用$|\mathcal{A}|$表示)。如果对每个结果给定一个二进制码字,每个码字的长度为$\log_2|\mathcal{A}_X|$比特(如果$|\mathcal{A}_X|$恰巧是2的幂)。因此,可以进行如下定义。

*X*的**原始比特内容**为

$$H_0(X) = \log_2|\mathcal{A}_X| \tag{4.15}$$

$H_0(X)$是总能从总体X中确定结果的二进制问题数目的下界。这是一个具有可加性的量:一个有序对x,y的原始比特内容有$|\mathcal{A}_X||\mathcal{A}_Y|$种可能的结果,满足

$$H_0(X,Y) = H_0(X) + H_0(Y) \tag{4.16}$$

这种对信息量的度量不包括任何不确定元素,与之对应的编码规则不"压缩"信源数据,它仅仅是将每一种结果映射为定长的二进制字符串。



习题4.5[2] 是否存在一种压缩器,可以将结果x映射成一个二进制码$c(x)$,而其解压器又能将c映射回x,使得每种可能的结果都被压缩成长度小于$H_0(X)$比特的二进制码?

尽管一个简单的计数论据表明:能减小所有文件大小的可逆压缩器不存在,但一些业余的压缩爱好者还是经常宣布他们已经发明了一种程序能做到这一点——事实上,他们能够将已经压缩过的数据多次通过压缩器,来做进一步的压缩。奇怪的是,这些现代炼金术竟然获得了专利。要做进一步的了解,可以查看comp. compression的常见问题。①

"压缩器"压缩文件的方式实际上只有两种。

① http://sunsite.org.uk/public/usenet/news-faqs/comp.compression/

1. 有损压缩器压缩一些文件，但将有些不同的文件映射成相同的编码。假设用户要求源文件得到完好的恢复，所以一旦出现易混淆的文件，将导致出错（尽管在图像压缩这类应用中，有损压缩被认为是令人满意的）。用 δ 表示源符号串是一个可混淆文件的概率，因此有损压缩器的失效概率为 δ。如果可将 δ 减得很小，有损压缩器还是可能有实用价值的。

2. 无损压缩器将所有文件映射成不同的编码；如果它缩短了某些文件，就必定增长了其他文件。我们试图设计压缩器，以使文件变长的概率很小，而使文件缩短的概率很大。

在这一章中，讨论一种简单的有损压缩器。在随后的章节中讨论无损压缩方法。

4.3 以有损压缩形式定义信息量

无论构建哪一种压缩器，都必须考虑出现不同结果的概率。设想比较两个文本文件的信息量，其中一个文件等概率地使用所有 128 个 ASCII 码；另一个文件中字符的使用概率与英文文本中的使用概率相同。是否可以定义一种信息量的度量方式来区分这两个文件呢？从直觉上说，后一个文件中每个字符所包含的信息量较小，因为它更容易预测。

我们已知某些符号使用概率较低，利用这一信息的一种简单方法是：设想用一个较小的字母表来对观察结果重新编码（因而丧失了对不太可能出现的一些符号进行编码的能力），然后测量新字母表的原始比特内容。例如，在压缩英文文本时，可以冒险地猜测最不常见的字符将不会出现，并得出一个简化的 ASCII 码，它省略了字符 {!,@,#,%,^,*,~,<,>,/,\,_,{,},[,],|}，因此将字母表规模减少了 17 个字符。愿意冒的风险越大，所得的最终字母表就越小。

在使用这种压缩方法时，引进一个参数 δ 来描述风险：δ 是一个结果 x 没有对应码字的概率。

例 4.6 设

$$\mathcal{A}_X = \{a,b,c,d,e,f,g,h\}$$

$$\mathcal{P}_X = \{\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{3}{16},\frac{1}{64},\frac{1}{64},\frac{1}{64},\frac{1}{64}\} \tag{4.17}$$

这个总体的原始比特内容是 3 比特，对应于 8 个二进制码字。但注意到 $P(x \in \{a,b,c,d\}) = 15/16$，因此愿意冒没有码字的风险的大小为 $\delta = 1/16$，可以只要 4 个码字，这是每个 $x \in \mathcal{A}_X$ 都有一个码字时所需码字个数的一半。

表 4.5 给出了当 $\delta = 0$ 和 $\delta = 1/16$ 时，可以给不同结果的二进制码字。当 $\delta = 0$ 时，需要 3 个比特对结果进行编码；当 $\delta = 1/16$ 时，只需要 2 个比特就够了。

表 4.5 对于两个失效概率 δ，各种结果所对应的二进制码字

$\delta=0$		$\delta=1/16$	
x	$c(x)$	x	$c(x)$
a	000	a	00
b	001	b	01
c	010	c	10
d	011	d	11
e	100	e	
f	101	f	
g	110	g	
h	111	h	

现在对这种思想进行严格描述。要得出这种冒险概率为 δ 的压缩策略，就要构造最小可能子集 S_δ，使得 x 不属于 S_δ 的概率小于或等于 δ，即 $P(x\notin S_\delta)\leqslant\delta$。对于每一个 δ 值，定义一个新的信息量度量方式——最小子集 S_δ 大小的对数。[在几个元素具有相同概率的总体中，可能存在几个包括不同元素的最小子集，但需要关注的只是它们的大小(而这里是相等的)，因此本章不在这种不明确的情况上花费更多时间。]

最小 δ - 充分子集 S_δ 是满足

$$P(x\in S_\delta)\geqslant 1-\delta \tag{4.18}$$

的最小 $\mathscr{A}_X$ 子集。可以按如下方法构造子集 S_δ：将 $\mathscr{A}_X$ 中的元素按照概率递减顺序排列，从概率最大的元素开始连续地加入元素，直到总概率 $\geqslant(1-\delta)$。

可以给最小充分子集中的每一个元素分配一个二进制码字，就构造了一个数据压缩码。这种压缩方法促使我们得到信息量度量的如下方式。

X 的**基本比特内容**为

$$H_\delta(X)=\log_2|S_\delta| \tag{4.19}$$

注意 $H_0(X)$ 是 $H_\delta(X)$ 中 $\delta=0$ 的特例(若对所有 $x\in\mathscr{A}_X$ 都有 $P(x)>0$)。[注意：不要把 $H_0(X)$ 和 $H_\delta(X)$ 与图 4.1 所示的 $H_2(p)$ 函数相混淆。]

图 4.6 所示为例 4.6 中总体的 $H_\delta(X)$，它是 δ 的函数。

扩展总体

如果所压缩的是信源的符号块，这种压缩方法是否仍然有用呢？

现在来看例子：结果 $\boldsymbol{x}=(x_1,x_2,\cdots,x_N)$ 是来自一个总体 X 的 N 个独立相同分布随机变量。将总体 $(X_1,X_2,\cdots,X_N)$ 记为 X^N。记住熵对于独立变量而言具有可加性(习题 4.2)，因此 $H(X^N)=NH(X)$。

例 4.7 考虑一枚变形硬币的 N 次翻转所得到的字符串 $\boldsymbol{x}=(x_1,x_2,\cdots,x_N)$，其中 $x_n\in\{0,1\}$，概率为 $p_0=0.9$，$p_1=0.1$。最可能的字符串 $\boldsymbol{x}$ 是那些有最多 0 的串。如果 $r(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 中 1 的个数，则

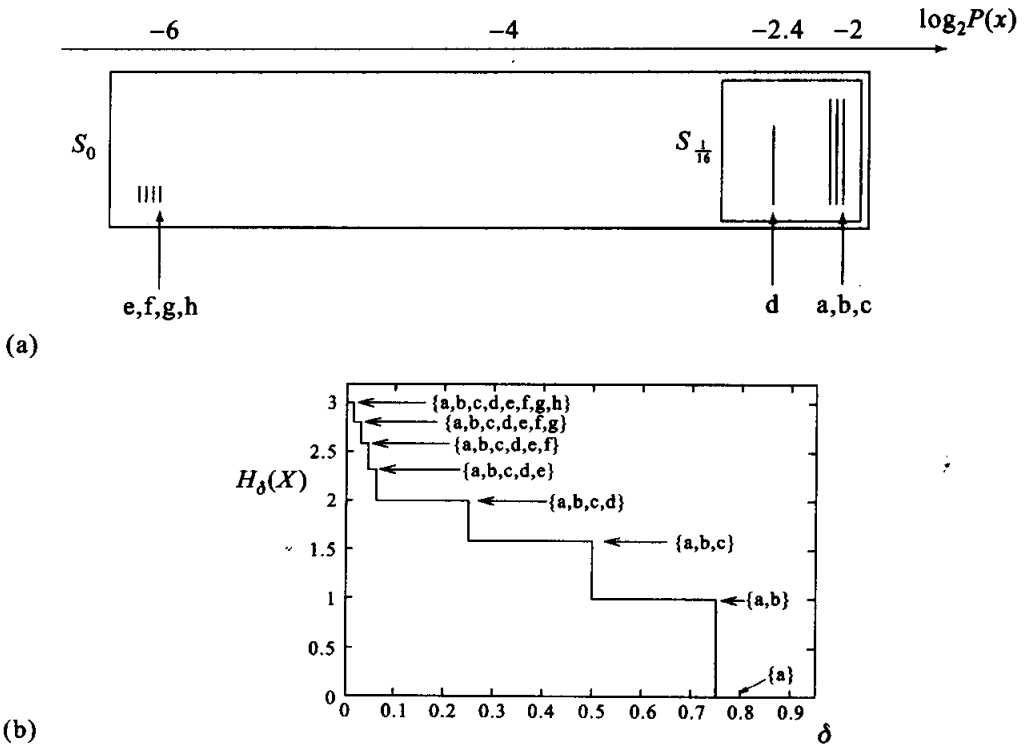


图 4.6 （a）按照概率大小，对 X 的结果（例 4.6）进行排序。（b）基本比特内容 $H_\delta(X)$。图中符号表明最小充分集是 δ 的函数。注意：$H_0(X)=3$ 比特，$H_{1/16}(X)=2$ 比特。

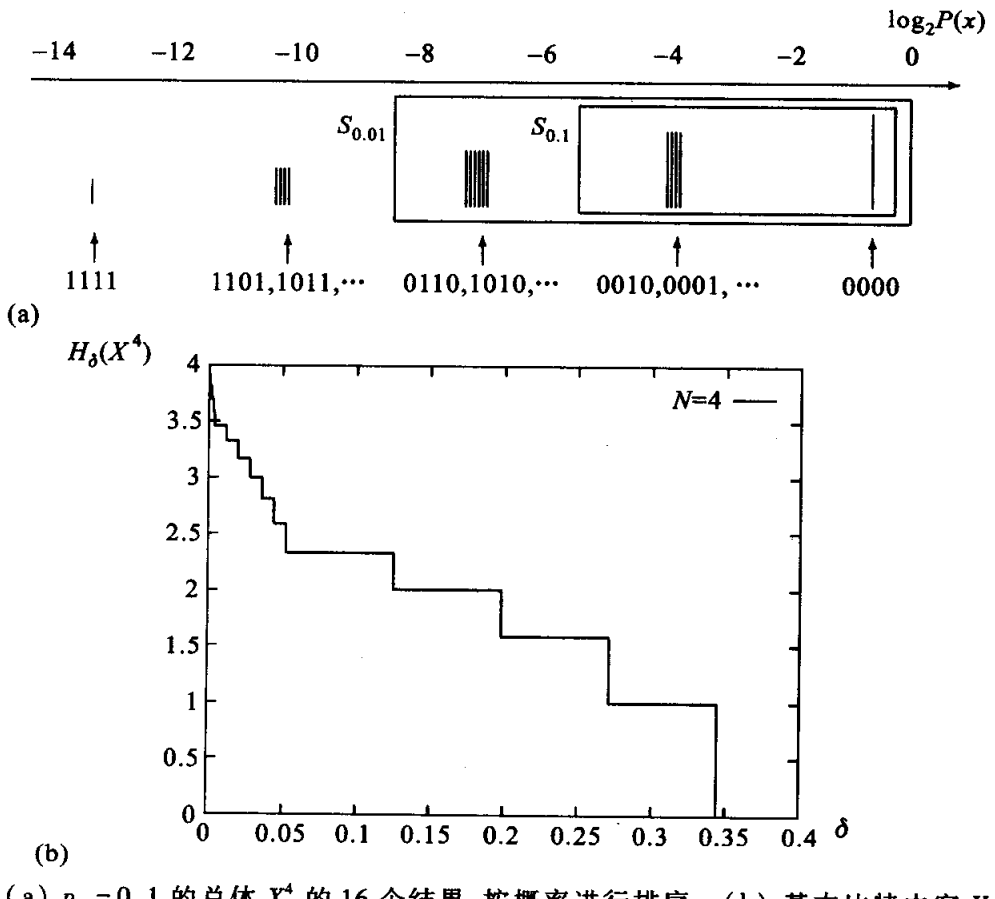


图 4.7 （a）$p_1=0.1$ 的总体 X^4 的 16 个结果，按概率进行排序。（b）基本比特内容 $H_\delta(X^4)$。上部的示意图用竖直线的长度表示字符串的概率（没有完全按比例）。

$$P(\boldsymbol{x}) = p_0^{N-r(\boldsymbol{x})} p_1^{r(\boldsymbol{x})} \tag{4.20}$$

为了计算 $H_\delta(X^N)$，必须找到最小的充分子集 S_δ。这个子集包含 $r(\boldsymbol{x})=0,1,2$ 直到 $r_{\max}(\delta)-1$ 的所有 $\boldsymbol{x}$，以及 $r(\boldsymbol{x})=r_{\max}(\delta)$ 的某些 $\boldsymbol{x}$。图 4.7 和图 4.8 给出了在 $N=4$ 和 $N=10$ 时，以 δ 为横轴的 $H_\delta(X^N)$ 的图形。步长是当 $|S_\delta|$ 改变 1 时所对应的 δ 值，阶梯斜率发生变化的尖端是 $r_{\max}$ 变化 1 的那些点。

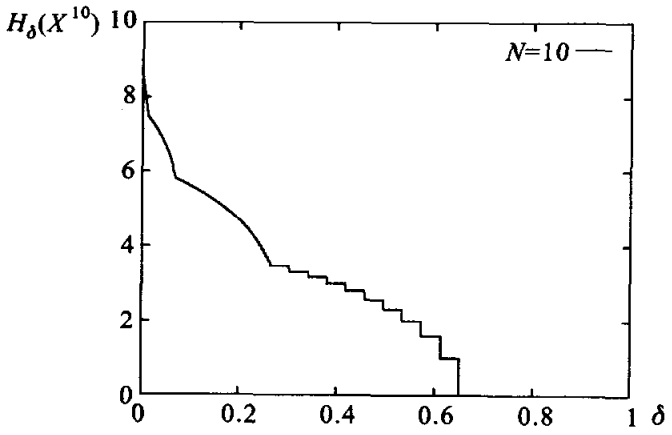


图 4.8 $N=10$ 个 $p_1=0.1$ 的二进制变量所对应的 $H_\delta(X^N)$

习题 4.8[⌕] 尖端点之间曲线的数学形状是怎样的？

图 4.6 ~ 图 4.8 所示的例子中，$H_\delta(X^N)$ 很大程度上依赖于 δ 的值，因此它看起来可能不是一个基本的或有用的信息量定义。但我们将考虑当 X^N 中独立变量个数 N 增加时会发生什么情况。我们会发现一个明显的结果：$H_\delta(X^N)$ 变得几乎独立于 δ，且对于所有 δ，它与 $NH(X)$ 非常接近，这里 $H(X)$ 是其中一个随机变量的熵。

图 4.9 给出了例 4.7 中的二进制总体的渐近趋势。随着 N 的增大，$\frac{1}{N}H_\delta(X^N)$ 变成了一个越来越平坦的函数，只有接近于 $\delta=0$ 和 $\delta=1$ 的尾部例外。只要允许存在

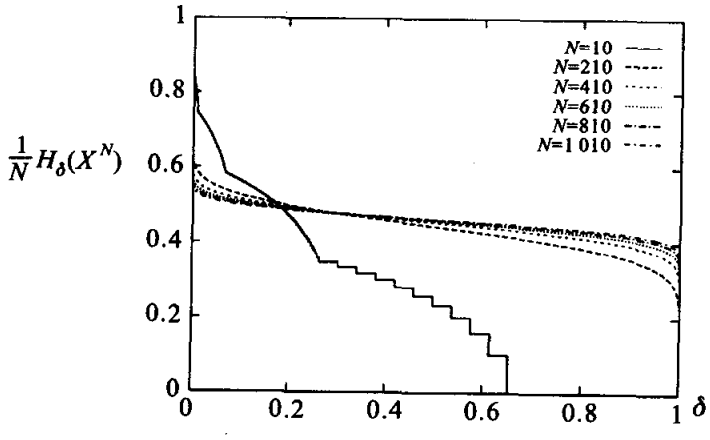


图 4.9 $N=10,210,\cdots,1\,010$ 个 $p_1=0.1$ 的二进制变量所对应的 $\frac{1}{N}H_\delta(X^N)$

一个很小的差错概率 δ,压缩到 NH 比特是可能实现的。即使允许存在一个大的差错概率,还是只能压缩成 NH 比特。这就是信源编码定理。

定理 4.1(香农信源编码定理) 设 X 为熵 $H(X)=H$ 比特的一个总体。若 $\varepsilon>0$ 且 $0<\delta<1$,则存在一个正整数 N_0,当 $N>N_0$ 时,有

$$\left|\frac{1}{N}H_\delta(X^N)-H\right|<\varepsilon \tag{4.21}$$

4.4 典 型 性

为什么增加 N 有所帮助呢?现在来考察 X^N 中的长字符串。图 4.10 给出了当 $N=100$,$p_1=0.1$ 时 X^N 中的 15 个样本。字符串 $\boldsymbol{x}$ 包含 r 个 1 和 $N-r$ 个 0 的概率为

$\boldsymbol{x}$	$\log_2(P(\boldsymbol{x}))$
...1.....1.....1....1.1.....1.....1.....1.....1.....11...	−50.1
.....1.....1....1.....1....1.....1.....1.....1....	−37.3
.....1....1..1...1....11..1.1.....11.....1...1.1..1..1.....1.	−65.9
1.1...1.....1.....11.1..1.....1.....1.....1..1.11.....	−56.4
...11.....1...1.....1.1.....1.....1....1..1.....1.....1.....	−53.2
.....1.....1.....1.1.....1.....1.....1...1.....1.....	−43.7
.....1.....1.....1...1.....1.....1.....1.....1..11.....	−46.8
.....1..1..1.....111.....1.....1.....1.1...1...1.....1	−56.4
.....1.....1.....1.....1.....1....1.....1...	−37.3
.....1.....1.....1.....1..1.1.1..1.....1.	−43.7
1.....1.....1...1.....1....1....1.....1..11..1.1...1.....	−56.4
.....11.1.....1.....1.....1.....1.....	−37.3
.1.....1...1.1.....1.....11.....1.1...1.....1.....11.....	−56.4
.....1...1..1.....1..11.1.1.1...1.....1.....1.....1..1.....	−59.5
.....11.1.....1....1..1.....1.....1.....1.....1.....	−46.8
.....	−15.2
111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111	−332.1

图 4.10 顶上 15 个字符串是从 X^{100} 中采得的样本,其中 $p_1=0.1$,$p_0=0.9$。底部两个是这个总体中最可能和最不可能的串。最后一列表示随机字符串的对数概率,可以将它们同熵 $H(X^{100})=46.9$ 比特进行对比。

$$P(\boldsymbol{x})=p_1^r(1-p_1)^{N-r} \tag{4.22}$$

包含 r 个 1 的字符串数目为

$$n(r)=\binom{N}{r} \tag{4.23}$$

因此,1 的个数 r 服从二项分布

$$P(r)=\binom{N}{r}p_1^r(1-p_1)^{N-r} \tag{4.24}$$

这些函数如图 4.11 所示。r 的均值为 Np_1，其标准差为 $\sqrt{Np_1(1-p_1)}$（第 1 章）。当 N 为 100 时，有

$$r \sim Np_1 \pm \sqrt{Np_1(1-p_1)} \simeq 10 \pm 3 \tag{4.25}$$

若 $N = 1\ 000$，则

$$r \sim 100 \pm 10 \tag{4.26}$$

注意，随着 N 值的增加，r 的概率分布变得越来越集中，即当 r 可能值的范围依 N 增大时，r 的标准差依 $\sqrt{N}$ 增大。r 很可能落入一个小的取值范围，这意味着结果 $\boldsymbol{x}$ 也很可能相应地落在结果的一个小子集中，而这个子集称之为典型集。

典型集的定义

针对字母表 $\mathscr{A}_X$ 的任意总体 X 定义典型性。关于典型字符串的定义会牵涉到字符串的概率。N 个符号所组成的长字符串通常包括第一个符号约 p_1N 次，第二个符号 p_2N 次，等等。因此这个字符串的概率可以粗略估算为

$$P(\boldsymbol{x})_{\text{typ}} = P(x_1)P(x_2)P(x_3)\cdots P(x_N) \simeq p_1^{(p_1N)} p_2^{(p_2N)} \cdots p_I^{(p_IN)} \tag{4.27}$$

于是，一个典型字符串的信息量为

$$\log_2 \frac{1}{P(\boldsymbol{x})} \simeq N \sum_i p_i \log_2 \frac{1}{p_i} \simeq NH \tag{4.28}$$

因此，随机变量 $\log_2 \dfrac{1}{P(\boldsymbol{x})}$ 是 $\boldsymbol{x}$ 的信息量，其值很接近于 NH。基于这一发现，我们对典型性进行定义。

定义 $\mathscr{A}_X^N$ 的典型元素是那些概率接近 2^{-NH} 的元素。（注意典型集不同于最小充分子集，它不包括 $\mathscr{A}_X^N$ 中最可能出现的元素，但是我们将证明这些最可能出现的元素所提供的概率几乎可以忽略。）

引入一个参数 β，它定义了一个元素的概率多接近 2^{-NH} 时才算“典型的”。称典型元素的集合为典型集 $T_{N\beta}$（如图 4.12 所示）：

$$T_{N\beta} = \left\{ \boldsymbol{x} \in \mathscr{A}_X^N : \left| \frac{1}{N} \log_2 \frac{1}{P(\boldsymbol{x})} - H \right| < \beta \right\} \tag{4.29}$$

将要证明无论选用怎样的 β 值，随着 N 的增大，典型集包含几乎所有的概率。

这个重要结论有时也称作“渐近均分”原理。

“渐近均分”原理 对于 N 个独立同等分布（i. i. d.）随机变量的总体 $X^N \equiv (X_1, X_2, \cdots, X_N)$，当 N 足够大时，结果 $\boldsymbol{x} = (x_1, x_2, \cdots, x_N)$ 几乎肯定属于一个只有 $2^{NH(X)}$ 个成员的子集，而每个成员的概率都“接近”$2^{-NH(X)}$。

注意：若 $H(X) < H_0(X)$，则 $2^{NH(X)}$ 是可能出现的结果数 $|\mathscr{A}_X^N| = |\mathscr{A}_X|^N = 2^{NH_0(X)}$ 的很小一部分。

选用“均分”这个术语所描述的思想是：典型集中所有成员具有大致相等的概率。[不能过分咬文嚼字，因此对“渐近均分”加了引号。]

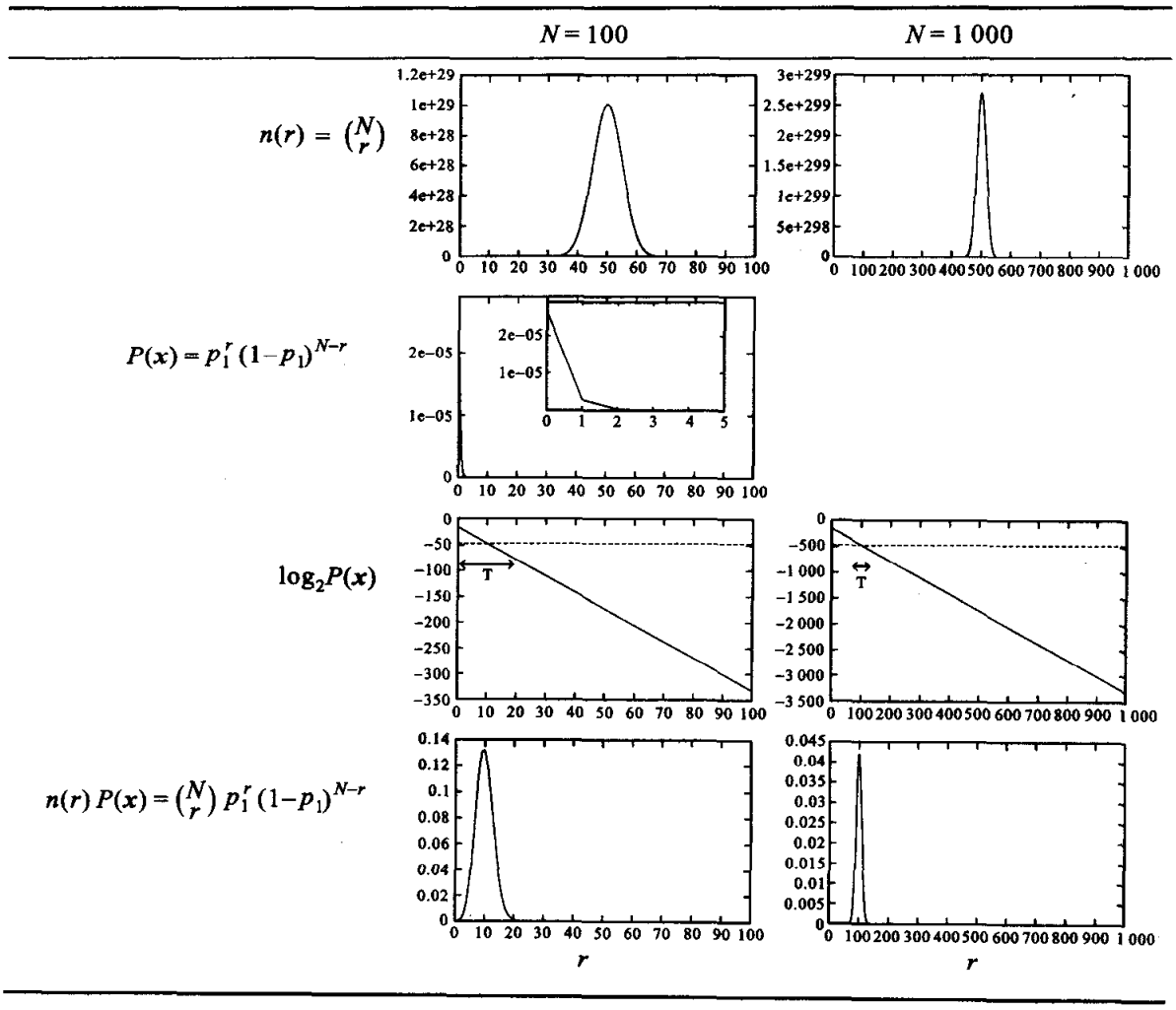


图 4.11 典型集 T 的解剖图。对于 $p_1 = 0.1$ 和 $N = 100$ 及 $N = 1\,000$ 的情形，这些图表示出了包含 r 个 1 的字符串的个数 $n(r)$；一个包含 r 个 1 的字符串的概率 $P(x)$；在对数坐标上的同一概率；所有包含 r 个 1 的字符串的总概率 $n(r)P(x)$。横坐标轴为个数 r。在 $\log_2 P(x)$ 的图上还用虚线表示出 $\log_2 P(x) = -NH_2(p_1)$ 的均值，当 $N = 100$ 时它等于 -46.9，而当 $N = 1\,000$ 时它等于 -469。典型集只包括 $\log_2 P(x)$ 接近于这个数值的那些字符串。用 T 所标出的范围表示集合 $T_{N\beta}$（其定义见第 4.4 节），左边对应于 $N = 100, \beta = 0.29$ 的情形，右边对应于 $N = 1\,000, \beta = 0.09$ 的情形。

均分的第二层涵义：在热物理学中，它所表示的是一个经典系统的每一个自由度具有同样的平均能量，$\frac{1}{2}kT$。这里所讨论的并非针对这一层涵义。

“渐近均分”原理等价于如下定理。

香农信源编码定理（口头表述） 当 $N \to \infty$ 时，熵为 $H(X)$ 的 N 个独立同等分布随机变量，可被压缩成多于 $NH(X)$ 个比特，使得信息丢失的风险可以忽略不计；反过来，如果压缩后的比特数少于 $NH(X)$，几乎可以肯定会丢失信息。

这两个定理是等价的，因为可定义一个压缩算法对典型集中的每个 x 赋予一个

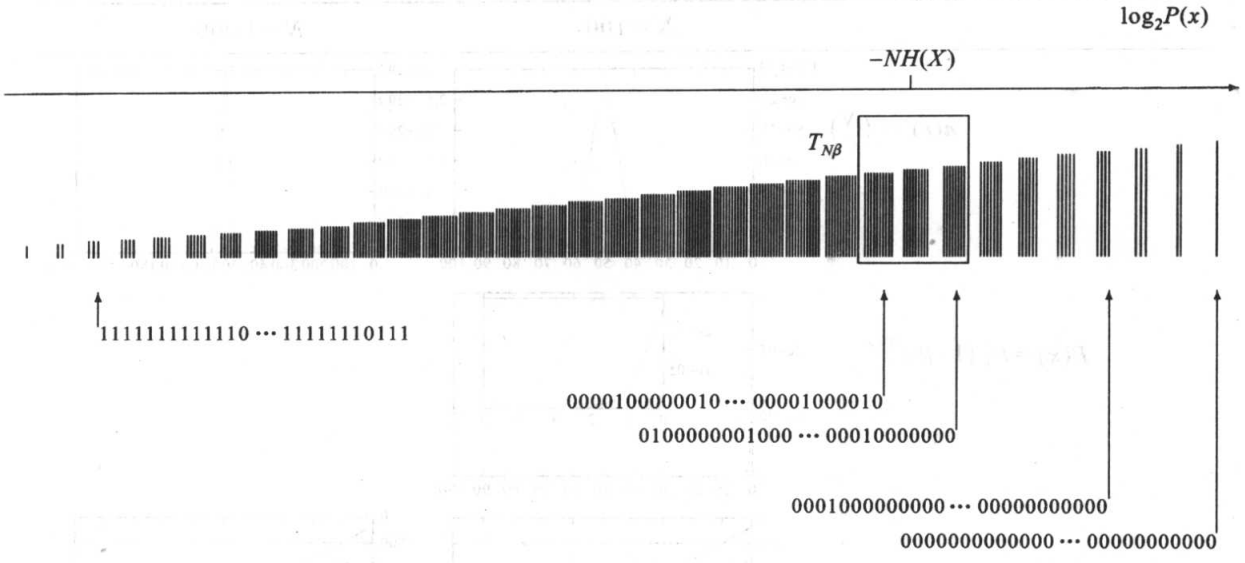


图 4.12 表示总体 X^N 中所有字符串(按其概率排序)和典型集 $T_{N\beta}$ 的示意图

长度为 $NH(X)$ 的不同码字。

4.5 证 明

如果觉得这一节太难,可以跳过。

大数定律

我们对信源编码定理的证明用到大数定律。

一个实数随机变量的**均值和方差**为 $\mathscr{E}[u] = \bar{u} = \sum_u P(u)u$ 和 $\mathrm{var}(u) = \sigma_u^2 = \mathscr{E}[(u-\bar{u})^2] = \sum_u P(u)(u-\bar{u})^2$。

技术注释:严格地讲,这里假设 u 是有限离散总体 X 中一个样本 x 的函数 $u(x)$。于是,和式 $\sum_u P(u)f(u)$ 应该写成 $\sum_x P(x)f(u(x))$。这也就意味着 $P(u)$ 是 δ 函数的一个有限和。这个约束保证了 u 的均值和方差确实存在,但一般的 $P(u)$ 却未必如此。

契比雪夫(Chebyshev)不等式 1 设 t 为一个非负实数随机变量,α 是一个正实数,则有

$$P(t \geqslant \alpha) \leqslant \frac{\bar{t}}{\alpha} \tag{4.30}$$

证明 $P(t \geqslant \alpha) = \sum_{t \geqslant \alpha} P(t)$。将每一项都乘以 $t/\alpha \geqslant 1$,得 $P(t \geqslant \alpha) \leqslant \sum_{t \geqslant \alpha} P(t)t/\alpha$。加上(非负的)缺失项,得 $P(t \geqslant \alpha) \leqslant \sum_t P(t)t/\alpha = \bar{t}/\alpha$。 □

契比雪夫不等式 2 如果 x 是一个随机变量,α 是一个正实数,则有

$$P((x-\bar{x})^2 \geqslant \alpha) \leqslant \sigma_x^2/\alpha \tag{4.31}$$

证明 设 $t=(x-\bar{x})^2$，应用前一命题即可得证。 □

弱大数定律 有 N 个独立随机变量 $h_1,h_2,\cdots,h_N$，它们有相同的均值 $\bar{h}$ 和相同的方差 σ_h^2：$x=\frac{1}{N}\sum_{n=1}^{N}h_n$。若 x 为这些随机变量的平均值，则有

$$P((x-\bar{h})^2\geqslant\alpha)\leqslant\sigma_h^2/\alpha N \tag{4.32}$$

证明 给出 $\bar{x}=\bar{h}$ 以及 $\sigma_x^2=\sigma_h^2/N$，即可得证。 □

我们所关心的是非常接近均值（α 很小）的那些 x。无论 σ_h^2 多大，无论所需的 α 多小，且无论 $(x-\bar{h})^2\geqslant\alpha$ 的期望概率多小，总是能够通过选用足够大的 N 值得到这个结果。

定理 4.1 的证明

将大数定律应用于为 $\boldsymbol{x}$ 所定义的随机变量 $\frac{1}{N}\log_2\frac{1}{P(\boldsymbol{x})}$，这里 $\boldsymbol{x}$ 是从总体 X^N 中抽取的。这个随机变量可写成 N 个信息量 $h_n=\log_2(1/P(x_n))$ 的平均值，其中每个信息量是均值为 $H=H(X)$、方差为 $\sigma^2\equiv\mathrm{var}[\log_2(1/P(x_n))]$ 的随机变量。（项 h_n 是第 n 次结果的香农信息量。）

再用参数 N 和 β 定义典型集如下：

$$T_{N\beta}=\left\{\boldsymbol{x}\in\mathcal{A}_X^N:\left[\frac{1}{N}\log_2\frac{1}{P(\boldsymbol{x})}-H\right]^2<\beta^2\right\} \tag{4.33}$$

对于所有 $\boldsymbol{x}\in T_{N\beta}$，$\boldsymbol{x}$ 的概率满足

$$2^{-N(H+\beta)}<P(\boldsymbol{x})<2^{-N(H-\beta)} \tag{4.34}$$

根据大数定律

$$P(\boldsymbol{x}\in T_{N\beta})\geqslant 1-\frac{\sigma^2}{\beta^2 N} \tag{4.35}$$

这样就证明了“渐进均分”原理。随着 N 的增加，对于任意 β，$\boldsymbol{x}$ 落在 $T_{N\beta}$ 中的概率趋近于 1。这个结果与信源编码有何关系？

必须把 $T_{N\beta}$ 与 $H_\delta(X^N)$ 联系起来。将要证明对于任何已知的 δ，都有一个充分大的 N，使得 $H_\delta(X^N)\simeq NH$。

第一部分：$\frac{1}{N}H_\delta(X^N)<H+\varepsilon$

集合 $T_{N\beta}$ 不是用于压缩的最佳子集。所以 $T_{N\beta}$ 的大小给出了 H_δ 的上界。通过计算 $T_{N\beta}$ 可能有多大，可证明 $H_\delta(X^N)$ 必须有多小。可以自由地将 β 设定为任何方便计算的值。一个 $T_{N\beta}$ 元素的最小可能概率为 $2^{-N(H+\beta)}$，且 $T_{N\beta}$ 所包含的所有元素的总概率不能超过 1。因此，有

$$|T_{N\beta}|2^{-N(H+\beta)}<1 \tag{4.36}$$

也就是说，典型集的大小所受约束为

$$|T_{N\beta}|<2^{N(H+\beta)} \tag{4.37}$$

如果设定 $\beta=\varepsilon$ 和 N_0，使得 $\frac{\sigma^2}{\varepsilon^2 N}\leqslant\delta$，那么 $P(T_{N\beta})\geqslant 1-\delta$，并且集合 $T_{N\beta}$ 满足 $H_\delta(X^N)\leqslant \log_2|T_{N\beta}|<N(H+\varepsilon)$。

第二部分：$\frac{1}{N}H_\delta(X^N)>H-\varepsilon$

假设有人说这第二部分不是这样的——也就是对于任何 N，存在比上面不等式所允许还要小的最小 δ - 充分子集 S_δ。可以用典型集来证明他们一定是搞错了。记住，可以自由地将 β 设定为所选择的任意值。设 $\beta=\varepsilon/2$，从而任务是证明不存在子集 S' 满足 $|S'|\leqslant 2^{N(H-2\beta)}$ 和 $P(x\in S')\geqslant 1-\delta$（对那些大于指定 N_0 值的 N）。

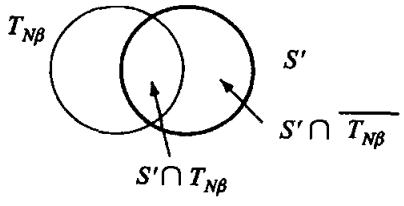
所以，考虑落在这个更小的对手子集 S' 中的概率。子集 S' 的概率是

$$P(x\in S')=P(x\in S'\cap T_{N\beta})+P(x\in S'\cap \overline{T_{N\beta}}) \tag{4.38}$$

其中 $\overline{T_{N\beta}}$ 表示补集 $\{x\notin T_{N\beta}\}$。假如 $S'\cap T_{N\beta}$ 包含 $2^{N(H-2\beta)}$ 个最大概率为 $2^{-N(H-\beta)}$ 的结果，就找到了第一项的最大值。第二项的最大值是 $P(x\notin T_{N\beta})$。因此有

$$P(x\in S')\leqslant 2^{N(H-2\beta)}2^{-N(H-\beta)}+\frac{\sigma^2}{\beta^2 N}=2^{-N\beta}+\frac{\sigma^2}{\beta^2 N} \tag{4.39}$$

现设 $\beta=\varepsilon/2$ 和 N_0，使得 $P(x\in S')<1-\delta$，这表示 S' 不满足充分子集 S_δ 的定义。因而任何大小满足 $|S'|\leqslant 2^{N(H-\varepsilon)}$ 的子集 S' 的概率小于 $1-\delta$，所以根据 H_δ 的定义，有 $H_\delta(X^N)>N(H-\varepsilon)$。



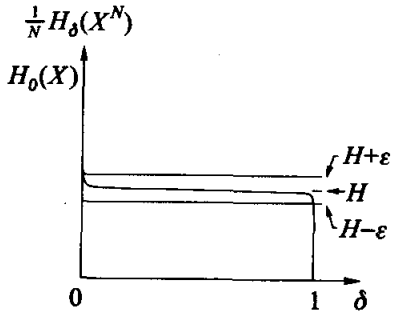


图 4.13 定理两个部分的示意图。已知任意 δ 和 ε，证明对于充分大的 N，$\frac{1}{N}H_\delta(X^N)$ 位于(1) 线 $H+\varepsilon$ 之下；(2) 线 $H-\varepsilon$ 之上。

因此对于充分大的 N，函数 $\frac{1}{N}H_\delta(X^N)$ 基本上是 δ 的常函数，其中 $0<\delta<1$，如图 4.9 和图 4.13 所示。 □

4.6 评　论

信源编码定理包含两部分，$\frac{1}{N}H_\delta(X^N) < H+\varepsilon$ 和 $\frac{1}{N}H_\delta(X^N) > H-\varepsilon$。两个结论都很有意思。

第一部分告诉我们，即使差错概率 δ 非常小，以逐渐减小的差错概率指定一个 N 符号长串 $\boldsymbol{x}$ 所需的每符号比特数 $\frac{1}{N}H_\delta(X^N)$ 也不必超过 $H+\varepsilon$ 比特。只需很小的可容忍误差，所需比特数就从 $H_0(X)$ 显著地降至 $(H+\varepsilon)$。

如过可以容忍更多的压缩误差，又会发生什么情况？第二部分告诉我们，即使 δ 非常接近 1，以至于大多数时间都会发生误差，指定 $\boldsymbol{x}$ 所需的每个符号的平均比特数仍然至少是 $H-\varepsilon$ 比特。这两个极端情况告诉我们，不管人们能容忍的误差具体是多少，指定 $\boldsymbol{x}$ 所需的每个符号的比特数是 H 比特；不多也不少。

关于“渐进均分”的注意点

将“渐进均分”这个词加上引号，因为非常重要的一点是，不要认为典型集 $T_{N\beta}$ 中的元素真的大致有同样的概率。说它们具有相似的概率，只是说它们的 $\log_2\frac{1}{P(\boldsymbol{x})}$ 值都在彼此的 $2N\beta$ 范围内。现在，随着 β 的减小，假如保持典型集的概率质量界 $P(\boldsymbol{x}\in T_{N\beta}) \geqslant 1-\frac{\sigma^2}{\beta^2 N}$ 恒定，N 会如何增加？N 必须随着 $1/\beta^2$ 的增加而增加，所以，如果将 β 表示成 N 的形式，如 $\alpha/\sqrt{N}$（对某个常数 α），那么在典型集中最有可能的字符串将比典型集中最不可能的字符串大 $2^{\alpha\sqrt{N}}$ 倍。随着 β 的减少，N 增加，且这一比值 $2^{\alpha\sqrt{N}}$ 按指数级增加。因此只是在弱的意义上使用“均分”的概念！

为何引入典型集？

对于分组压缩，所选择的最佳子集（根据定义）是 S_δ，而非典型集。那为什么还要引入典型集？答案是，可以对典型集计数。我们知道它所有的元素都有“几乎相同”的概率（2^{-NH}），并且知道整个集合的概率几乎为 1，所以典型集一定约有 2^{NH} 个元素。如果不借助典型集（它与 S_δ 非常相似），将很难计算 S_δ 中有多少元素。

4.7 习 题

称重问题

▷ **习题 4.9**[1] 当第一次遇到12个球和3种结果天平的称重问题(习题4.1)时,有人会认为先用6个球对6个球进行第一次称重是一种好办法,而其他人说"不,称6个球对6个球,根本不会传达任何信息"。向第二组人解释为什么他们既是对的又是错的。计算"哪个球是异常"所获得的信息量,及"哪个球是异常的并判断它是重是轻"所获得的信息量。

▷ **习题 4.10**[2] 求解存在39个球,已知其中有一个球是异常时的称重问题。

▷ **习题 4.11**[2] 给定16个球,只有一个球不知是重是轻,其他所有球的重量都一样。假设有一个奇怪的两盘天平,它只能报出两种结果:"两边平衡"或"两边不平衡"。设计一种方法,用尽可能少的天平称量次数来确定哪个球是异常的。

▷ **习题 4.12**[2] 有一个两盘的天平;任务是要称出重量为整数1~40磅的袋装面粉。需要称几次?[允许把砝码放在任何一个托盘内,但一次只允许在天平上放一袋面粉。]

▷ **习题 4.13**[1] (a) 能否使用3次固定的称重顺序来求解习题4.1(12个球和3种结果天平的称重问题),使得第二次称重所选择的球不再依赖于第一次的结果,而第三次称重又不依赖于第一次或第二次的结果?

(b) 找出 N 个球中有一个异常球的一般 N - 球称重问题的解决方案。证明通过 W 次称重,可从 $N=(3^W-3)/2$ 个球中识别那个异常球。

习题 4.14[3] 假设给定习题4.1中的12个球和3种结果的天平;这次两个球是异常的;每个异常球可能重也可能轻,我们并不知道是哪一种。我们想识别这个异常球及其异常方向(是重是轻)。

(a) 估计使用最优策略所需的称重次数。假如有3个球是异常的,需要称重几次?

(b) 假如知道所有的规则球的重量为100 g,轻球的重量为99 g,重球的重量为110 g,那么你的答案会变化吗?

带有失真度 δ 的有损压缩器的信源编码

▷ **习题 4.15**[2] 设 $\mathcal{P}_X=\{0.2,0.8\}$,当 N = 1,2 和 1 000 时,画出 $\frac{1}{N}H_\delta(X^N)$ 作为 δ 函数的曲线图。

▷ **习题 4.16**[2] 设 $\mathcal{P}_Y=\{0.5,0.5\}$,画出当 N = 1,2,3 和 100 时,$\frac{1}{N}H_\delta(Y^N)$ 作为 δ 函

数的曲线图。

▷ **习题 4.17**[2] （对于物理系学生） 讨论‘渐进均分’原理的证明和 Boltzmann 熵与 Gibbs 熵的等价性(对于大系统而言)之间的关系。

不服从大数定律的分布

这章所使用的大数定律,证明了 N 个独立同等分布的随机变量集合的均值所服从的概率分布随着 N 的增加,宽度 $\propto 1/\sqrt{N}$ 变窄。可是,我们只对离散随机变量做了证明,也就是说,对于有限的可能值集合中所取的实数,这个性质是成立的。虽然许多具有连续概率分布的随机变量也满足大数定律,还是有些重要分布不满足。有些连续分布并不存在均值或方差。

▷ **习题 4.18**[2] 画出如下柯西(Cauchy)分布的草图

$$P(x)=\frac{1}{Z}\frac{1}{x^2+1},\quad x\in(-\infty,\infty) \tag{4.40}$$

归一化常数 Z 是什么？你能计算它的均值或方差吗？

考虑和式 $z=x_1+x_2$,其中 x_1 和 x_2 是服从柯西分布的独立随机变量。那么 $P(z)$ 是多少？x_1 和 x_2 的均值 $\bar{x}=(x_1+x_2)/2$ 的概率分布是什么？服从柯西分布的 N 个样本的均值的概率分布是什么？

其他渐近特性

习题 4.19[2] Chernoff 界 设不等式 $P(t\geqslant\alpha)\leqslant\bar{t}/\alpha$ 中的随机变量 t 是我们感兴趣的随机变量 x 的函数 $t=(x-\bar{x})^2$,从契比雪夫不等式(4.30)推导出弱大数定律。

可以使用其他函数获得其他一些有用的不等式。可以通过令 $t=\exp(sx)$ 获得 Chernoff 界,它对于限制一个分布的尾部是相当有用的。

证明

$$P(x\geqslant a)\leqslant e^{-sa}g(s),\quad \text{对于所有 } s>0 \tag{4.41}$$

和

$$P(x\leqslant a)\leqslant e^{-sa}g(s),\quad \text{对于所有 } s<0 \tag{4.42}$$

其中 $g(s)$ 是 x 的矩生成函数

$$g(s)=\sum_x P(x)e^{sx} \tag{4.43}$$

与 $p\log 1/p$ 相关的奇特函数

习题 4.20[3] 这道题没有任何目的;将它放在这里只是为了满足那些喜欢数学者的好奇心。

画出 $x\geqslant 0$ 时函数

$$f(x) = x^{x^{x^{x^{x^{\cdots}}}}} \tag{4.44}$$

的草图。提示:计算 f 的反函数,也就是函数 $g(y)$ 满足:如果 $x = g(y)$,则有 $y = f(x)$,它与 $p\log 1/p$ 密切相关。

4.8 习题解答

习题 4.2 的解答 设 $P(x,y) = P(x)P(y)$,则有

$$H(X,Y) = \sum_{xy} P(x)P(y)\log\frac{1}{P(x)P(y)} \tag{4.45}$$

$$= \sum_{xy} P(x)P(y)\log\frac{1}{P(x)} + \sum_{xy} P(x)P(y)\log\frac{1}{P(y)} \tag{4.46}$$

$$= \sum_{x} P(x)\log\frac{1}{P(x)} + \sum_{y} P(y)\log\frac{1}{P(y)} \tag{4.47}$$

$$= H(X) + H(Y) \tag{4.48}$$

习题 4.4 的解答 一个 ASCII 文件可以压缩到 7/8 大小。这种压缩通过如下一个分组码获得,这个分组码通过把 56 个信息比特复制成 7 个字节并忽略每个字节的最后一位,而将 8 字节的分组码映射成 7 字节。

习题 4.5 的解答 鸽巢原理表明:在没有两次使用一个笼子的情况下,你不能把 16 只鸽子放进 15 个笼子里。

类似地,你不能给出 $\mathcal{A}_X$ 个结果,它是长度 l 小于 $\log_2|\mathcal{A}_X|$ 比特的唯一二进制码字,因为只有 2^l 个这样的二进制码字,而 $l < \log_2|\mathcal{A}_X|$ 意味着 $2^l < |\mathcal{A}_X|$,所以至少两个不同的输入压缩器的文件被压缩成同一输出文件。

习题 4.8 的解答 在尖点之间,所有概率变化都是相等的,T 中的元素数每一步变化 1 个,所以 H_δ 是随 $(-\delta)$ 对数级地变化的。

习题 4.13 的解答 这个解答是 Dyson 和 Lyness 于 1946 年发现,而 John Conway 于 1999 年给出以下比较优雅的形式。注意:符号 A、B 和 C 用来命名球、天平的托盘、结果,并命名异常球的可能状态!

(a) 顺序标识 12 个球

AAB ABA ABB ABC BBC BCA BCB BCC CAA CAB CAC CCA 并在

第 1 次称重,放 AAB ABA ABB ABC 在盘 A,BBC BCA BCB BCC 在盘 B。

第 2 次称重,放 AAB CAA CAB CAC 在盘 A,ABA ABB ABC BBC 在盘 B。

第 3 次称重,放 ABA BCA CAA CCA 在盘 A,AAB ABB BCB CAB 在盘 B。

在一次给定的称重中,一个托盘最终会处在:

- 标准位置(C),是在两个盘平衡时取得;
- 向上情况(A);
- 向下情况(B)。

所以 3 次称重决定了每个盘是这 3 个字母的一个序列。

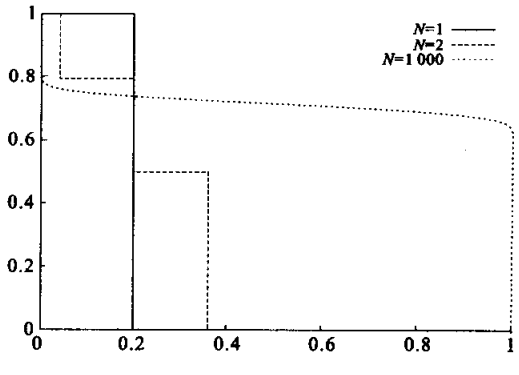
假如两个序列都是 CCC,那么就没有异常球了。否则,对于两个盘中的任何一个,字母序列在前述 12 个中,并根据异常球的重量是重于或轻于正常球,以 A 或 B 对其盘命名。

(b) 在 W 次称重中,可用同样的方法从

$$N=(3^W-3)/2 \tag{4.49}$$

个球中识别出异常球,具体方法:用 W 个字母 A、B 或 C(它们的首次变化是 A 到 B,或者 B 到 C,或者 C 到 A)的非常量序列来标识它们,并在第 w 次称重时,把第 w 个字母是 A 的,放在盘 A,第 w 个字母是 B 的,放在盘 B。

习题 4.15 的解答 当 $N=1,2,1\ 000$ 时,$\frac{1}{N}H_\delta(X^N)$ 作为 δ 的函数,其曲线如图 4.14 所示。注意 $H_2(0.2)=0.72$ 比特。



$N=1$		
δ	$\frac{1}{N}H_\delta(X)$	$2^{H_\delta(X)}$
0~0.2	1	2
0.2~1	0	1

$N=2$		
δ	$\frac{1}{N}H_\delta(X)$	$2^{H_\delta(X)}$
0~0.04	1	4
0.04~0.2	0.79	3
0.2~0.36	0.5	2
0.36~1	0	1

图 4.14 对 $N=1,2,1\ 000$ 个 $p_1=0.4$ 的二进制变量,$\frac{1}{N}H_\delta(X)$(纵轴)对 δ(横轴)。

习题 4.17 的解答 Gibbs 熵是 $k_B\sum_i p_i\ln\frac{1}{p_i}$,其中 i 取遍系统的所有状态。这个熵等于(除了相差一个系数 k_B 外)总体的香农熵。

然而可以对任何总体定义 Gibbs 熵,而 Boltzmann 熵只定义在微正则总体上,它的概率分布在可达状态集上是均匀的。Boltzmann 熵的定义为 $S_B=k_B\ln\Omega$,其中 Ω 是微正则总体的可达状态数。这等价于受限集合的完全信息量 H_0(除了相差系数 k_B 之外)。微正则总体的 Gibbs 熵简单地等于 Boltzmann 熵。

现在考虑热噪声分布(正则总体),其中状态 x 的概率是

$$P(x)=\frac{1}{Z}\exp\left(-\frac{E(x)}{k_BT}\right) \tag{4.50}$$

可将正则总体与相应的微正则总体联系起来,后一总体的总能量固定在正则总体的

平均能量上(固定精度限于 ε 内)。现在,将总能量固定在精度 ε 上,这等价于将 $\ln\frac{1}{P(x)}$ 的值固定在精度 $\varepsilon k_B T$ 上。我们所定义的典型集 $T_{N\beta}$ 正是,在正则总体下,由 $\log P(x)$ 值非常接近 $\log P(x)$ 的均值 $-NH(X)$ 的所有元素组成。因此,微正则总体等价于定义在正则总体的典型集上的均匀分布。

对"渐近均分"原理所做的证明也证明了:对于一个能量被分割成独立项之和的系统,如果将微正则总体的能量限制成等于正则总体的平均能量,微正则总体的 Boltzmann 熵(对于 N 很大的情况)非常接近正则总体的 Gibbs 熵。

习题 4.18 的解答 柯西分布

$$P(x)=\frac{1}{Z}\frac{1}{x^2+1}$$

的归一化常数是

$$Z=\int_{-\infty}^{\infty}\mathrm{d}x\frac{1}{x^2+1}=\left[\tan^{-1}x\right]_{-\infty}^{\infty}=\frac{\pi}{2}-\frac{-\pi}{2}=\pi \tag{4.51}$$

这个分布的均值和方差都未定义。(分布关于 0 是对称的,但这并不意味着它的均值为 0。它的均值是一个发散积分的值。)若 x_1 和 x_2 都服从柯西分布,和数 $z=x_1+x_2$ 的概率密度则由如下卷积给出。

$$P(z)=\frac{1}{\pi^2}\int_{-\infty}^{\infty}\mathrm{d}x_1\frac{1}{x_1^2+1}\frac{1}{(z-x_1)^2+1} \tag{4.52}$$

使用标准方法,经过一番繁杂的计算后,得

$$P(z)=\frac{1}{\pi^2}2\frac{\pi}{z^2+4}=\frac{2}{\pi}\frac{1}{z^2+2^2} \tag{4.53}$$

可以看出它服从宽度参数为 2 的柯西分布(原分布的宽度参数为 1)。这意味着两个点的平均值 $\bar{x}=(x_1+x_2)/2=z/2$ 服从宽度参数为 1 的柯西分布。推而广之,服从柯西分布的 N 个样值的均值与单个样值具有相同参数的柯西分布。均值的概率分布不会按 $1/\sqrt{N}$ 变窄。

中心极限定理并不适用于柯西分布,因为它不具有有限的方差。

另一种求得式(4.53)的简洁方法使用柯西分布的傅里叶变换,它是双边指数函数 $\mathrm{e}^{-|\omega|}$。在实域的卷积,相应于傅里叶域的乘积,所以 z 的傅里叶变换就是 $\mathrm{e}^{-|2\omega|}$。经过反向变换,得到式(4.53)。

习题 4.20 的解答 函数 $f(x)$ 有反函数

$$g(y)=y^{1/y} \tag{4.54}$$

注意

$$\log g(y)=(1/y)\log y \tag{4.55}$$

以 y 为垂直轴,$g(y)$ 为水平轴,画出 $g(y)$,就得到 $f(x)$ 的粗略图。从所得图可知,$f(x)$ 是 $x\in(0,1)$ 上的单值函数,而令人惊奇的是,它看上去很规范和普通;对于 $x\in(1,\mathrm{e}^{1/\mathrm{e}})$,$f(x)$ 是双值函数。$f(\sqrt{2})$ 同时等于 2 和 4。对于 $x>\mathrm{e}^{1/\mathrm{e}}$(约为 1.44),$f(x)$ 是

无穷的。可是，有人可能会反驳说：假如我们定义 f 为函数序列 $x, x^x, x^{x^x}, \cdots$ 的极限，这种勾勒 $f(x)$ 草图的方法只是部分正确的；这种序列对于 $0 \leqslant x \leqslant (1/\mathrm{e})^{\mathrm{e}} \simeq 0.07$ 没有极限，因为在 $x = (1/\mathrm{e})^{\mathrm{e}}$ 处出现了二值分叉。对于 $x \in (1, \mathrm{e}^{1/\mathrm{e}})$，这个序列的极限是单值——如图所示为两个值中较小的那个。

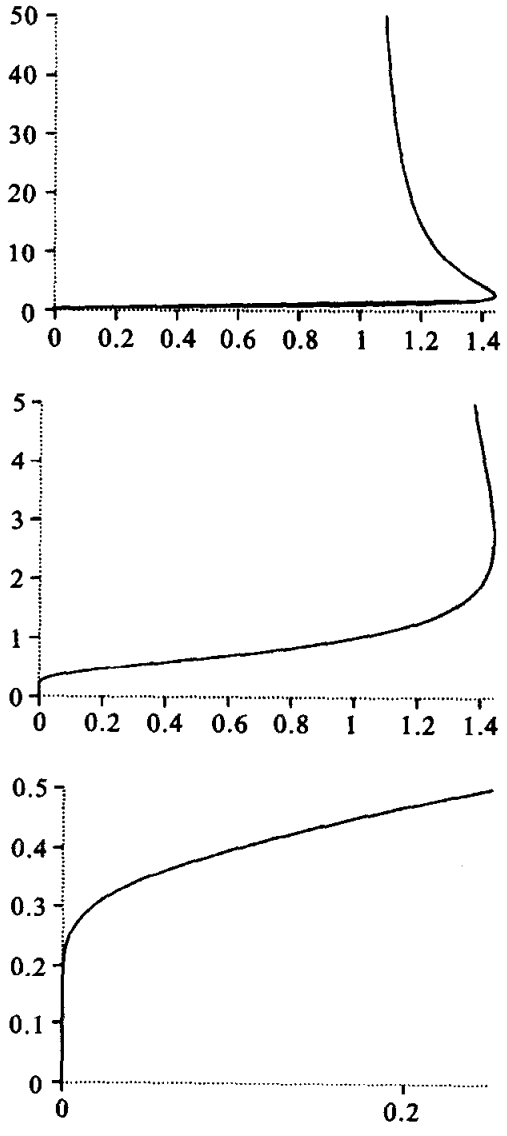


图 4.15 用 3 种不同的尺度画出 $f(x) = x^{x^{x^{x^{\cdots}}}}$

关于第5章

在上一章，看到了熵作为平均信息量度量处于关键地位的证据。通过使用固定长度分组码定义了一种数据压缩方法，并证明随着 N 的增加，可将 N 个独立同等分布的变量 $\boldsymbol{x}=(x_1,\cdots,x_N)$ 编码成 $N(H(X)+\varepsilon)$ 比特的码块，并使差错概率逐渐消失，然而假如试图将 X^N 编码成 $N(H(X)-\varepsilon)$ 比特，它的差错概率几乎为1。

因此，我们验证了数据压缩的可能性，但是证明中所定义的分组编码没有给出实际的算法。在本章和下一章中，将学习实际的数据压缩算法。上一章的压缩算法使用固定长度的大码块，并且是有损的，在下一章讨论可变长度的压缩算法，它对应于分组长度较短的实用码，并且是无损的。

设想一个装满水的橡皮手套。如果挤压这个手套中的两个手指，手套的其他部分就要膨胀，因为水的总体积是恒定的。（水基本上是不可压缩的。）类似地，当压缩某些结果的码字时，如果所用方法是无损的，其他码字必定要变长。在本章中，将发现信息论中能够和水量类比的内容。

在阅读第5章之前，应该先完成习题2.26。

将使用如下记号表示区间：

$x\in[1,2)$ 表示 $x\geqslant 1$ 且 $x<2$。

$x\in(1,2]$，表示 $x>1$ 且 $x\leqslant 2$。

第 5 章

符 号 码

在这一章,将讨论可变长符号码,它每次只对一个信源符号进行编码,而不是对 N 个信源符号的巨大字符串进行编码。这些码是无损的:与上一章的分组码不同,它们肯定能毫无差错地压缩和解压缩;但是有可能存在这样一种情况,就是编码后的字符串有时比原本的信源串还要长。

基本思想是,通过给更有可能出现的结果分配短些的码字,给不太可能出现的结果分配长的码字,从而在平均意义上达到压缩的目的。

关键问题是:

假如符号码是无损的,会有什么影响? 假如一些码字可以更短,那么其他码字必须增至多长?

使压缩切实可行。如何保证符号码易于译码?

最优符号码。为了获得最佳的压缩,应如何分配码长? 可实现的最佳压缩是什么?

通过证明如下定理,可再次证实香农信息量和熵处于关键地位。

信源编码定理(符号码) 存在总体 X 的可变长度编码 C,使得已编码符号的平均长度 $L(C,X)$ 满足 $L(C,X) \in [H(X),H(X)+1)$。

只有当每个结果的码长都等于它的香农信息量时,这个平均长度才等于熵 $H(X)$。

还将定义一个能产生最优符号码的富有建设性的算法:霍夫曼(Huffman)编码算法。

符号集的记号 $\mathcal{A}^N$ 表示由集合 $\mathcal{A}$ 的元素所组成的有序 N - 元组集合,即所有字符串的长度都为 N。符号 $\mathcal{A}^+$ 表示由集合 $\mathcal{A}$ 中元素所构成的所有有限长度字符串的集合。

例 5.1 $\{0,1\}^3 = \{000,001,010,011,100,101,110,111\}$

例 5.2 $\{0,1\}^+ = \{0,1,00,01,10,11,000,001,\cdots\}$

5.1 符 号 码

总体 X 的(**二进制**)**符号码** C 是从 x 的取值范围 $\mathcal{A}_X = \{a_1,a_2,\cdots,a_I\}$ 到 $\{0,1\}^+$ 的一个映射。$c(x)$ 表示相对于 x 的码字,$l(x)$ 表示它的长度,即 $l_i = l(a_i)$。

扩展码 C^+ 是从 $\mathcal{A}_X^+$ 到 $\{0,1\}^+$ 的一个映射,可以通过对如下相应码字进行无标

点级联得到。

$$c^+(x_1x_2\cdots x_N)=c(x_1)c(x_2)\cdots c(x_N) \tag{5.1}$$

[这里的“映射”与“函数”是同义词。]

例 5.3 总体 X 的符号码 C_0定义为

$$\begin{aligned}\mathscr{A}_X&=\{a,b,c,d\}\\ \mathscr{P}_X&=\{1/2,1/4,1/8,1/8\}\end{aligned} \tag{5.2}$$

该码如下图所示。

	a_i	$c(a_i)$	l_i
	a	1000	4
C_0:	b	0100	4
	c	0010	4
	d	0001	4

使用扩展码,可以把 acdbac 编码成

$$c^+(\text{acdbac})=100000100001010010000010 \tag{5.3}$$

对实用的符号码有一些基本要求。首先,任何已编码的字符串必须有唯一的译码。第二,符号码必须易于译码。第三,编码应取得尽可能多的压缩。

任何编码字符串必须有唯一的译码

码字 $C(X)$可唯一译码 如果在扩展码 C^+下,两个不同的字符串不会有相同的编码,也就是

$$\forall x,y\in\mathscr{A}_X^+,x\neq y\Rightarrow c^+(x)\neq c^+(y) \tag{5.4}$$

上面所定义的码字 C_0是可唯一译码的一个例子。

符号码必须易于译码

如果在符号码到达时马上就可识别出码字的结尾,那么这个码就很容易译码,这意味着没有一个码字是另一码字的前缀。[如果存在一个尾串 t,使得级联 ct 与 d 相同,就说码字 c 是另一码字 d 的前缀。例如,1 是 101 的前缀,10 也是 101 的前缀。]

稍后将要证明,如果限定我们的符号码为前缀码,不会使性能有任何折损。

符号码称作前缀码 如果没有码字是其他任何码字的前缀,则称符号码为前缀码。

前缀码也叫做即时码或自标码,因为无需往前查看后续码字,就可以从左到右对一个已编码字符串进行译码。码字的结尾是可立即被识别出来。前缀码是可唯一译码的。

前缀码同时也叫做“无前缀码”或“前缀条件码”。

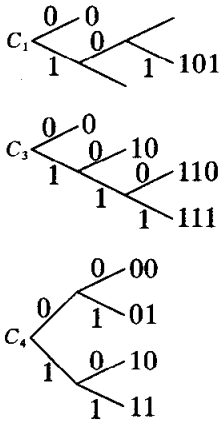
前缀码和树结构相对应。

例 5.4 码字 $C_1=\{0,101\}$ 是前缀码，因为 0 不是 101 的前缀，101 也不是 0 的前缀。

例 5.5 $C_2=\{1,101\}$。这个码不是前缀码，因为 1 是 101 的前缀。

例 5.6 码字 $C_3=\{0,10,110,111\}$ 是一个前缀码。

例 5.7 码字 $C_4=\{00,01,10,11\}$ 是一个前缀码。



前缀码可以用二叉树来表示。完全的前缀码对应于一个没有未用分支的二叉树。C_1 是一个非完全的前缀码。



习题 5.8[7] C_2 是可唯一译码的码吗？

例 5.9 考虑习题 4.1 和图 4.2。任何识别异常球并判定孰轻孰重的称重策略，都可视做是对 24 种可能状态中的每一种所分配的一个三位码。这个码是一个前缀码。

码字应尽可能地做到压缩

对于总体 X，符号码 C 的**期望长度** $L(C,X)$ 是

$$L(C,X)=\sum_{x\in\mathscr{A}_X}P(x)l(x) \tag{5.5}$$

也可以把这个量写为

$$L(C,X)=\sum_{i=1}^{I}p_il_i \tag{5.6}$$

其中 $I=|\mathscr{A}_X|$。

例 5.10 设

$$\begin{aligned}\mathscr{A}_X&=\{a,b,c,d\}\\ \mathscr{P}_X&=\{1/2,1/4,1/8,1/8\}\end{aligned} \tag{5.7}$$

并考虑码字 C_3。X 的熵为 1.75 比特，此码的期望长度 $L(C_3,X)$ 也是 1.75 比特。符号序列 $\boldsymbol{x}=(\text{acdbac})$ 可以编码成 $c^+(\boldsymbol{x})=0110111100110$。$C_3$ 是一个前缀码，也是可

唯一译码的码。注意:码字长度满足 $l_i = \log_2(1/p_i)$,或等价地有 $p_i = 2^{-l_i}$。

C_3:

a_i	$c(a_i)$	p_i	$h(p_i)$	l_i
a	0	1/2	1.0	1
b	10	1/4	2.0	2
c	110	1/8	3.0	3
d	111	1/8	3.0	3

例 5.11 考虑同一总体 X 的固定长度码 C_4。期望长度 $L(C_4, X)$ 为 2 比特。

	C_4	C_5
a	00	0
b	01	1
c	10	00
d	11	11

例 5.12 考虑 C_5。期望长度 $L(C_5, X)$ 为 1.25 比特,它比 $H(X)$ 小。但是码字不是可唯一译码的。序列 $\boldsymbol{x}$ = (acdbac) 编码成 000111000,这也可被译码成(cabdca)。

例 5.13 考虑码字 C_6。这个码的期望长度 $L(C_6, X)$ 为 1.75 比特。符号序列 $\boldsymbol{x}$ = (acdbac) 被编码成 $c^+(\boldsymbol{x})$ = 0011111010011。

C_6是一个前缀码吗? 不是,因为 c(a) = 0 也是 c(b)和 c(c)的一个前缀。

C_6是可唯一译码的码吗? 答案不是很显而易见的。如果你认为它可能不是可唯一译码的码,可试着通过找出具有相同编码的两个字符串 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 来证明之。[可唯一译码性的定义由式(5.4)给出。]

C_6:

a_i	$c(a_i)$	p_i	$h(p_i)$	l_i
a	0	1/2	1.0	1
b	01	1/4	2.0	2
c	011	1/8	3.0	3
d	111	1/8	3.0	3

C_6肯定不易译码。当接收到“00”时,$\boldsymbol{x}$ 可能是从“aa”、“ab”或“ac”开始的。一旦接收到“001111”,第二个符号仍然是含糊不定的,因为 $\boldsymbol{x}$ 可能是“abd…”或者

“acd…”。但是一旦编码流中出现下一个0时,那么唯一的译码就最终确定了。

事实上 C_6 是可唯一译码的。与前缀码 C_3 进行比较,可以发现 C_6 的码字是 C_3 码字的逆。C_3 是可唯一译码的,这也证明了 C_6 是可唯一译码的,因为 C_6 中的任何字符串与 C_3 中倒着读的字符串相同。

5.2 可唯一译码性施加了哪些限制?

现在要问,已知一列正整数 $\{l_i\}$,是否存在可唯一译码的码,用这些整数作为它的码字长度? 在现阶段,我们忽略不同符号的概率;一旦能够更好地理解可唯一译码性,我们将再次引入概率,并讨论如何构造最优的可唯一译码的符号码。

在上面的例子中,已经观察到,如果采用诸如{00,01,10,11}之类的码,并缩短其中的一个码字,例如00→0,那么只有当我们加长其他码字时,才可以保持其可唯一译码性。因此,能够花费在码字上的预算看来是受限的,而越短的码字就越昂贵。

现在讨论这种预算的本质。如果单纯从长度 $l=3$ 的码字来构造一个码字,要保持可唯一译码性,可以有多少码字? 答案是 $2^l=8$。一旦选择所有这8个码字,那么是否有办法在码字中加入其他长度的码字,并保持码的唯一可译性? 好像不能。

如果要包含一个长度为1的码字“0”,而其他码字的长度为3,情况又如何? 我们可以有多少个长度为3的码字? 如果仔细观察前缀码,可以得到4个长度为3的码字,{100,101,110,111}。那么其他码字呢? 有其他选择长度为3码字的方法,可让我们获得更多码字吗? 直观地,我们认为这是不可能的。看来,一个长度为3的码字所需付出的代价比一个长度为1的码字会小 2^2 倍。

让我们定义可花费在码字上的总预算为1。如果设定长度为 l 的码字的费用为 2^{-l},那么就得到一个适合上面所讨论例子的定价系统。长度为3的码字各花费1/8,长度为1的码字各花费1/2。可以将预算花费在任何码字上。如果超出了预算,那么码字肯定不会是可唯一译码的。另一方面,如果

$$\sum_i 2^{-l_i} \leqslant 1 \tag{5.8}$$

那么码也许是可唯一译码的。这个不等式就是Kraft不等式。

Kraft不等式 对于任何定义在二元符号集{0,1}上的可唯一译码的码 $C(X)$,其码长必须满足

$$\sum_{i=1}^{I} 2^{-l_i} \leqslant 1 \tag{5.9}$$

其中 $I=|\mathcal{A}_X|$。

完全性 如果一个可唯一译码的码,其Kraft不等式的等号成立,那么它叫做完全码。

我们想要可唯一译码的码;而前缀码是一个可唯一译码的码,也易于译码。所以

如果将注意力集中在前缀码，那么情况会简单些。幸运地是，对于任何信源，都存在亦为前缀码的最优符号码。

Kraft 不等式和前缀码 已知一组码字长度满足 Kraft 不等式，那么就存在一个可唯一译码的前缀码具有这样的码字长度。

Kraft 不等式更准确的称谓可能是 Kraft - McMillan 不等式：Kraft 证明如果不等式被满足，那么存在一个具有给定码字长度的前缀码。McMillan(1956)证明了其反命题，即可唯一译码性意味着不等式成立。

Kraft 不等式的证明 定义 $S=\sum_i 2^{-l_i}$。考虑量

$$S^N=\left[\sum_i 2^{-l_i}\right]^N=\sum_{i_1=1}^{I}\sum_{i_2=1}^{I}\cdots\sum_{i_N=1}^{I}2^{-(l_{i_1}+l_{i_2}+\cdots+l_{i_N})} \tag{5.10}$$

指数中的量$(l_{i_1}+l_{i_2}+\cdots+l_{i_N})$是字符串$\boldsymbol{x}=a_{i_1}a_{i_2}\cdots a_{i_N}$的编码长度。对于每个长度为 N 的字符串 $\boldsymbol{x}$，在上面的和式中都有一个对应项。引入数组 A_l 表示有多少个编码长度为 l 的字符串 $\boldsymbol{x}$。然后，定义 $l_{\min}=\min_i l_i$ 和 $l_{\max}=\max_i l_i$：

$$S^N=\sum_{l=Nl_{\min}}^{Nl_{\max}}2^{-l}A_l \tag{5.11}$$

现在假设 C 是可唯一译码的，于是对于所有 $\boldsymbol{x}\neq\boldsymbol{y}$，都有 $c^+(\boldsymbol{x})\neq c^+(\boldsymbol{y})$。集中考虑编码长度为 l 的 $\boldsymbol{x}$。共有 2^l 个长度为 l 的不同比特串，因此它必须满足 $A_l\leqslant 2^l$，于是有

$$S^N=\sum_{l=Nl_{\min}}^{Nl_{\max}}2^{-l}A_l\leqslant\sum_{l=Nl_{\min}}^{Nl_{\max}}l\leqslant Nl_{\max} \tag{5.12}$$

因而对于所有 N，有 $S^N\leqslant l_{\max}N$。现在假设 S 大于 1，那么随着 N 的增加，S^N 会是以指数级递增的函数，并且当 N 足够大时，指数函数总会超过多项式函数，如 $l_{\max}N$。但是对于所有 N，我们的结果($S^N\leqslant l_{\max}N$)都成立。因此 $S\leqslant 1$。 □

▷ **习题 5.14**[扩] 证明上述结果，即对任何满足 Kraft 不等式的码字长度集合$\{l_i\}$，都存在具有这些码长的前缀码。

Kraft 不等式的示意图也许可以帮助你求解这道题。假设选择码字来构成一个符号码。我们可以在“超市”画出所有候选码字集，并用盒子的面积来表示码字的“价格”(图 5.1)。总的可用预算为“1”，也就是出现在 Kraft 不等式右边的那个数，如图的一侧所示。第 5.1 节所讨论的一些码如图 5.2 所示。注意那些前缀码 C_0、C_3 和 C_4 具备这样一种性质：在任何选定码字的右边，不存在任何其他选定的码字，因为前缀码所对应的是树结构。注意完全前缀码对应于没有未用分支的完全树。

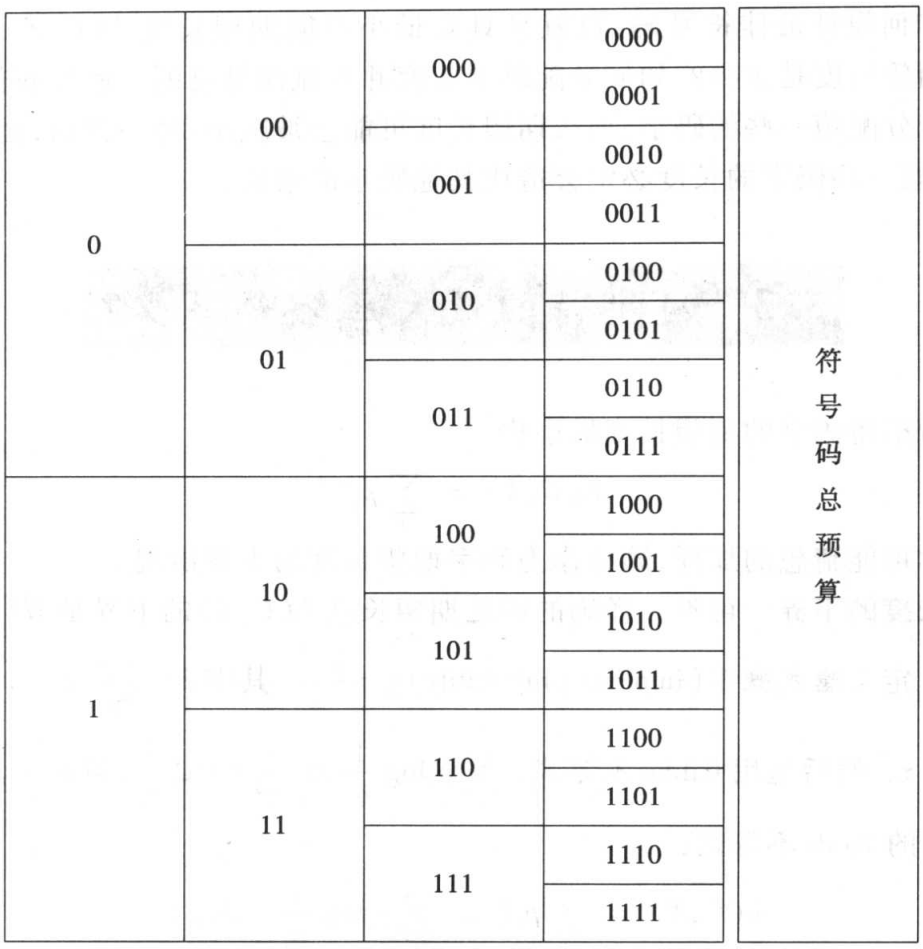


图 5.1 符号编码预算。将每个码字写入一个格子,长度为 l 的码字的"代价"2^{-l}用相应格子的大小来表示。在构造一个可唯一译码的码时,总的可用预算为 1。

可以将这个示意图想象为一个*码字超市*:码字按其长度成列排放,每个码字的费用以其所在货架上格子的大小来表示。如果你所选择的码字超出了预算,那么你构造的码就不是可唯一译码的。

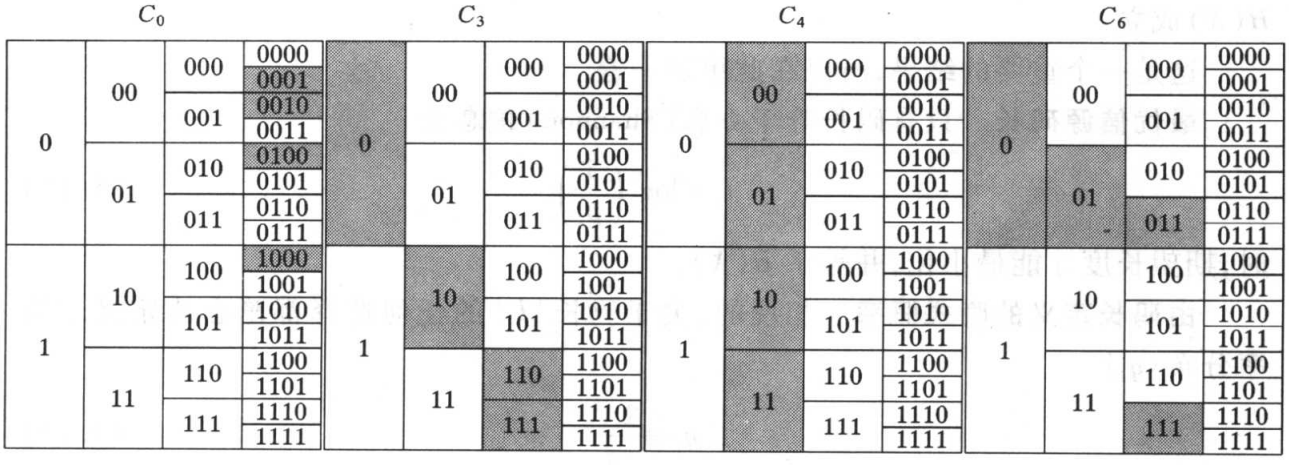


图 5.2 第 5.1 节中的码 C_0、C_3、C_4和 C_6所选择的码字

现在准备将符号概率$\{p_i\}$放回来。已知一组符号概率(例如,图 2.1 中的英语语

言概率),如何设计最佳符号码,也就是具有最小可能期望长度 $L(C,X)$ 的符号码?最小可能期望长度是多少?如何分配码字长度并非显而易见的。假如我们给更可能出现的符号分配短一些的码字,那么期望长度可能会被减小;另一方面,由 Kraft 不等式可知,缩短一些码字的长度必定会造成其他码字的增长。

5.3 可期待的最大压缩是多少?

我们希望将码字的期望长度最小化

$$L(C,X) = \sum_i p_i l_i \tag{5.13}$$

正如你可能猜想的那样,熵将作为码字期望长度的下界出现。

期望长度的下界 可唯一译码的码之期望长度 $L(C,X)$ 的下界是 $H(X)$。

证明 定义隐式概率(implicit probability)$q_i = \frac{2^{-l_i}}{z}$,其中 $z = \sum_{i'} 2^{-l_{i'}}$,于是有 $l_i = \log\frac{1}{q_i} - \log z$。然后运用 Gibbs 不等式,$\sum_i p_i \log\frac{1}{q_i} \geqslant \sum_i p_i \log\frac{1}{p_i}$,当 $q_i = p_i$ 时等式成立;$z \leqslant 1$ 时的 Kraft 不等式:

$$L(C,X) = \sum_i p_i l_i = \sum_i p_i \log\frac{1}{q_i} - \log z \tag{5.14}$$

$$\geqslant \sum_i p_i \log\frac{1}{p_i} - \log z \tag{5.15}$$

$$\geqslant H(X) \tag{5.16}$$

只有当 Kraft 等式成立,即满足 $z = 1$,且码长满足 $l_i = \log\left(\frac{1}{p_i}\right)$ 时,等式 $L(C,X) = H(X)$ 成立。 □

这是一个重要的结果,所以在此重述一遍。

最优信源码长 只有码长等于香农(Shannon)信息量

$$l_i = \log_2\left(\frac{1}{p_i}\right) \tag{5.17}$$

时,期望长度才能最小化,并等于 $H(X)$。

由码长定义的隐式概率 相反的,关于码长 $\{l_i\}$ 的任何选择都隐含地定义了概率分布 $\{q_i\}$

$$q_i \equiv \frac{2^{-l_i}}{z} \tag{5.18}$$

在此概率分布下,这些码长将是最优的。假如码是完全的,那么 $z = 1$,并且隐式概率由 $q_i = 2^{-l_i}$ 表示。

5.4 能够压缩多少?

所以,不能压缩到低于熵的水平。能期望有多接近熵呢?

定理 5.1(符号码的信源编码定理) 对于一个总体 X,存在一个前缀码 C,其期望长度满足

$$H(X) \leqslant L(C,X) < H(X) + 1 \tag{5.19}$$

证明 设定码长为略大于最优长度的整数

$$l_i = \left\lceil \log_2\left(\frac{1}{p_i}\right) \right\rceil \tag{5.20}$$

其中$\lceil l^* \rceil$表示大于或等于 l^* 的最小整数。[我们没有断言最优码一定要使用这些长度,我们选择这些长度只是因为能用它来证明定理。]

通过确认是否满足 Kraft 不等式,来检查是否存在具有这些码字长度的前缀码。

$$\sum_i 2^{-l_i} = \sum_i 2^{-\left\lceil \log_2\left(\frac{1}{p_i}\right)\right\rceil} \leqslant \sum_i 2^{-\log_2\left(\frac{1}{p_i}\right)} = \sum_i p_i = 1 \tag{5.21}$$

于是确认

$$L(C,X) = \sum_i p_i \left\lceil \log\left(\frac{1}{p_i}\right) \right\rceil < \sum_i p_i \left(\log\left(\frac{1}{p_i}\right) + 1\right) = H(X) + 1 \tag{5.22}$$

□

使用错误码长的代价

如果使用一个码,其码长不等于最优码长,那么平均消息长度将会比熵大。

如果真实概率为$\{p_i\}$,且使用长度为 l_i 的完全码,可认为这些长度定义了隐式概率 $q_i = 2^{-l_i}$。继续运用式(5.14),得平均长度

$$L(C,X) = H(X) + \sum_i p_i \log \frac{p_i}{q_i} \tag{5.23}$$

也就是说,它超过熵的部分为相对熵 $D_{\mathrm{KL}}(\boldsymbol{p} \parallel \boldsymbol{q})$。

5.5 最优的符号码信源编码:霍夫曼码

已知一组概率 $\mathscr{P}$,如何才能设计一个最优前缀码呢?例如,对于图 5.3 中的英语语言总体,最好的符号码是什么?所谓“最优”,是假定目标是对期望长度 $L(C,X)$ 最小化。

怎么会不能实现

有人可能会试着大致将集合 $\mathscr{A}_X$ 分成两部分，并继续将每个子集分成两部分，从而从根开始定义一棵二叉树。与称重问题一样，这种构造在思路上是好的，但它未必是最优的；它会得到 $L(C,X) \leqslant H(X)+2$。

霍夫曼编码算法

现在给出一种优美而简单的算法（见算法5.4），来寻找最优前缀码。诀窍是从码字的尾部开始往后构造码字；即从树叶开始生成二叉树。

> 1. 从符号集中选取两个具有最小概率的符号。对这两个符号分配最长的码字，二者长度相同，只有最后一位数字不同。
>
> 2. 将这两个符号合并成一个符号，并继续重复下去。

算法5.4 霍夫曼编码算法

因为每一步都会将符号表（alphabet）的大小减少1，这种算法会在 $|\mathscr{A}_X|-1$ 步之后为所有符号分配好字符串。

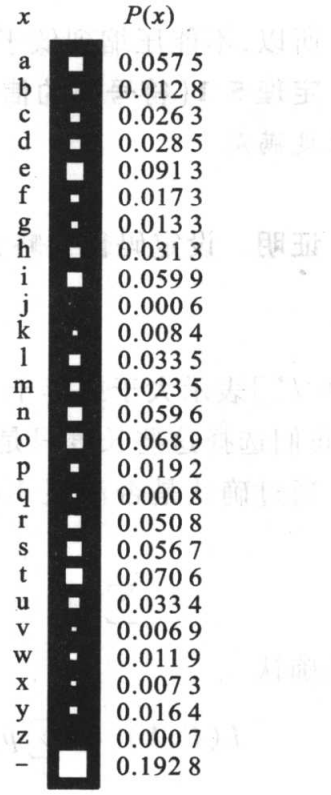
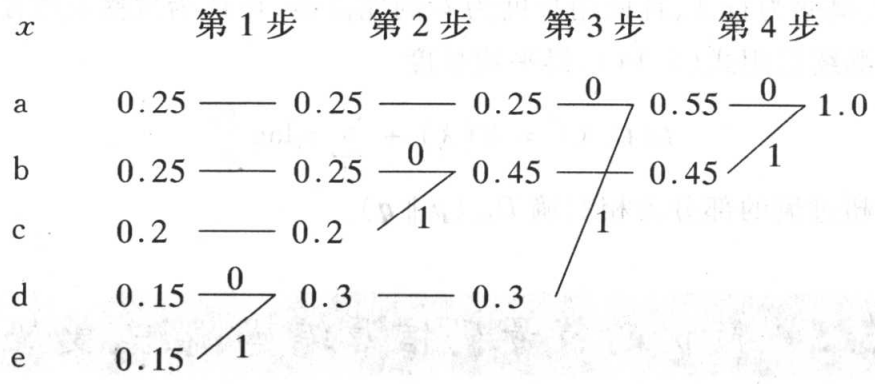


图5.3 需要符号码的一个总体

例5.15 设 $\mathscr{A}_X=\{a,b,c,d,e\}$ 且 $\mathscr{P}_X=\{0.25,0.25,0.2,0.15,0.15\}$



以逆序连接这些二进制数字就得到了码字：$C=\{00,10,11,010,011\}$。霍夫曼算法（表5.5中的第4列）所选择的码长有时比理想长度要长，有时比理想长度要短，香农信息量为 $\log_2 \frac{1}{p_i}$（第3列）。码的期望长度为 $L=2.30$ 比特，而熵为 $H=2.285\ 5$ 比特。 □

表 5.5 用霍夫曼算法构造的码

a_i	p_i	$h(p_i)$	l_i	$c(a_i)$
a	0.25	2.0	2	00
b	0.25	2.0	2	10
c	0.2	2.3	2	11
d	0.15	2.7	3	010
e	0.15	2.7	3	011

若任意点处有不止一种方法来选择两个具有最小概率的符号,则可以任意选择——码的期望长度与这种选择无关。

习题 5.16[小] 证明:对于一个信源而言,没有比霍夫曼码更好的符号码。

例 5.17 现在可以针对图 2.1 所介绍的符号集上的概率分布,来构造一个霍夫曼码。结果如图 5.6 所示。这个码的期望长度为 4.15 比特,总体的熵为 4.11 比特。观察所分配的码长与理想码长 $\log_2 \frac{1}{p_i}$ 之间的偏差。

a_i	p_i	$\log_2 \frac{1}{p_i}$	l_i	$c(a_i)$
a	0.057 5	4.1	4	0000
b	0.012 8	6.3	6	001000
c	0.026 3	5.2	5	00101
d	0.028 5	5.1	5	10000
e	0.091 3	3.5	4	1100
f	0.017 3	5.9	6	111000
g	0.013 3	6.2	6	001001
h	0.031 3	5.0	5	10001
i	0.059 9	4.1	4	1001
j	0.000 6	10.7	10	1101000000
k	0.008 4	6.9	7	1010000
l	0.033 5	4.9	5	11101
m	0.023 5	5.4	6	110101
n	0.059 6	4.1	4	0001
o	0.068 9	3.9	4	1011
p	0.019 2	5.7	6	111001
q	0.000 8	10.3	9	110100001
r	0.050 8	4.3	5	11011
s	0.056 7	4.1	4	0011
t	0.070 6	3.8	4	1111
u	0.033 4	4.9	5	10101
v	0.006 9	7.2	8	11010001
w	0.011 9	6.4	7	1101001
x	0.007 3	7.1	7	1010001
y	0.016 4	5.9	6	101001
z	0.000 7	10.4	10	1101000001
–	0.192 8	2.4	2	01

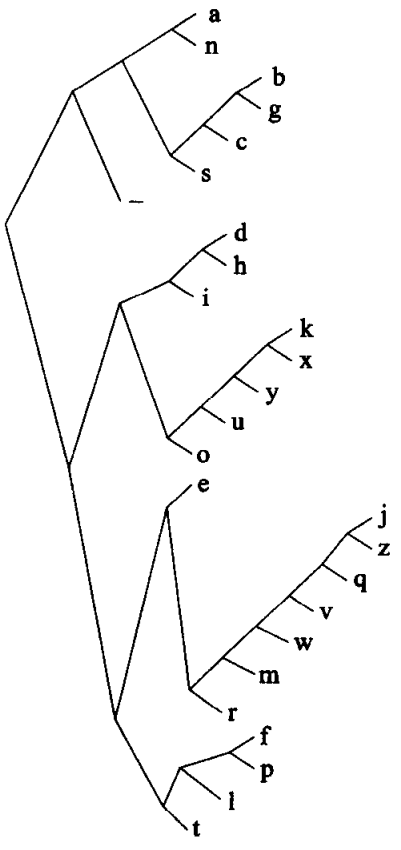


图 5.6 英语语言总体(字母组合统计)的霍夫曼码

自顶向下构建二叉树是次优的

在前面几章中,研究了称重问题,其中构建了三叉树或二叉树。我们注意到平衡树(在每个步骤中,使两种可能的结果尽可能等概率的树)似乎描述了一个最有效的实验。这样从直觉上启发我们以熵作为信息量的一种度量。

在自顶向下的方法中,符号表被连续地分割成尽可能相近的子集,以便子集之间是等概率的。但是,采用这种贪婪的方法并非总能构造出最优码。

例 5.18 对如下总体求解最优二进制符号码。

$$\mathscr{A}_X = \{a,b,c,d,e,f,g\}$$
$$\mathscr{P}_X = \{0.01,0.24,0.05,0.20,0.47,0.01,0.02\} \tag{5.24}$$

注意:自顶向下的贪婪方法可以将其分裂成两个子集{a,b,c,d}和{e,f,g},二者概率均为1/2,{a,b,c,d}又可分成{a,b}和{c,d}两个子集,其概率均为1/4;于是,自顶向下的贪婪方法就给出如表5.7中第3列所示的码,其期望长度为2.53比特。第4列是用霍夫曼码算法所产生的码,它的期望长度是1.97比特。 □

表 5.7 用贪婪法所构造的码与霍夫曼码之对比

a_i	p_i	贪婪法	霍夫曼法
a	0.01	000	000000
b	0.24	001	01
c	0.05	010	0001
d	0.20	011	001
e	0.47	10	1
f	0.01	110	000001
g	0.02	111	00001

5.6 霍夫曼码的缺点

霍夫曼算法为一个总体产生了一个最优符号码,但事情并未到此了结。术语"总体"(ensemble)和"符号码"(symbol Code)在使用时需要格外小心。

变化的总体

若希望对于不变总体中的一个结果序列进行通信,则霍夫曼码可能是方便的。但相关的总体往往会变化。例如,如果所压缩的是文本,那么符号频率会随上下文而变化:英文中的字母u跟在字母q之后的概率比它跟在字母e之后的概率要大得多

(如图 2.3 所示)。还有,关于这些上下文相关符号使用频率的知识,也会随着我们对源文本统计特性的认知而改变。

霍夫曼码并不能很好地处理总体概率变化的情形。一种愚笨的方法是,符号的概率一旦发生变化,就重新计算霍夫曼码。另一种态度是否认这种适应性选择,取而代之的是提前检查整个文件,并计算出一个好的概率分布,而在传输过程中此概率分布会始终保持不变。在这种情况下,还必须针对码本身进行通信。这种方法不仅繁琐受限,而且还是次优的,因为用于确定码的初始消息和文档本身都是部分冗余的。因此,这种方法浪费了一些比特。

额外比特

霍夫曼码中一个同样重要的问题是看似无伤大雅的"额外比特"问题,即同理想平均长度 $H(X)$ 相比,霍夫曼码所达到的长度满足 $H(X) \leqslant L(C,X) < H(X)+1$(这已在定理 5.1 中证明)。因此,霍夫曼码会导致每个符号有 0 ~ 1 比特的开销。若 $H(X)$ 很大,则这种开销所占份额并不重要。但在很多应用中,熵可能只是每个符号 1 比特或更小,因此开销 $L(C,X)-H(X)$ 可能在编码文件长度中占主导地位。考虑英文文本:在某些上下文中,长的字符串也许是高度可预测的。例如,在上下文"strings _ of _ ch"中,可以以 0.99 的概率预测到,下面的 9 个符号为"aracters _"。传统的霍夫曼码必须对每个字符至少使用 1 比特,于是共要消耗 9 比特却几乎没有传送任何信息(确切地说,共传达了 0.13 比特)。给定一个好的模型,英文的熵也就大约每个字符 1 比特(Shannon, 1948),因此一个霍夫曼码有可能效率很低。

霍夫曼码的一种传统扩充可用来压缩符号块,例如在第 4 章所讨论的"扩展信源" X^N。每个块的额外开销至多 1 比特,确保每个符号的额外开销至多 $\frac{1}{N}$ 比特。对于足够大的块,额外比特的问题不再存在,但要为此付出一些代价:(a) 失去了简单霍夫曼码所具有的即时译码能力;(b) 不得不计算所有相关字符串的概率,以构建与之关联的霍夫曼树。人们最后要对大量字符串显式地计算概率和码,而其中多数的串在实际中却永远不会出现。(见习题 5.29。)

符号码之外的问题

因此,尽管霍夫曼码被大肆鼓吹为"最优",但在实际应用中仍然存在很多缺陷。它们*确实是最优符号码*,但在实际应用中*并不需要符号码*。

霍夫曼码中的缺陷可以用*算术编码*来矫正,后者缺少一种约束,即不必将每个符号都翻译成整数个比特。下一章将重点讲述算术编码。

5.7 小 结

Kraft 不等式 如果一个码是可唯一译码的，它的码长必定满足

$$\sum_i 2^{-l_i} \leqslant 1 \tag{5.25}$$

对于满足 Kraft 不等式的任意一组长度，总存在一个具有这种长度的前缀码。

总体的最优信源码长等于香农信息量

$$l_i = \log_2 \frac{1}{p_i} \tag{5.26}$$

反之，任意一组码长都定义了一种隐式概率

$$q_i = \frac{2^{-l_i}}{z} \tag{5.27}$$

相对熵 $D_{\mathrm{KL}}(\boldsymbol{p} \parallel \boldsymbol{q})$ 所度量的是当总体的真实概率分布为 $\boldsymbol{p}$，而使用隐式概率为 $\boldsymbol{q}$ 的码时，每个符号所浪费的比特数。

符号码的信源编码定理 对一个总体 X 而言，存在一个前缀码，其期望长度满足

$$H(X) \leqslant L(C,X) < H(X) + 1 \tag{5.28}$$

霍夫曼编码算法迭代地产生一个最优符号码。在每次迭代过程中，两个最不可能出现的符号被组合。

5.8 习 题

▷ **习题 5.19**[✍] 码 $\{00,11,0101,111,1010,100100,0110\}$ 是否是可唯一译码的？

▷ **习题 5.20**[✍] 3 - 元码 $\{00,012,0110,0112,100,201,212,22\}$ 是否是可唯一译码的？



习题 5.21[✍] 给出 X^2、X^3 和 X^4 的霍夫曼码，其中 $\mathscr{A}_X = \{0,1\}$，$\mathscr{P}_X = \{0.9, 0.1\}$。计算它们的期望长度，并与熵 $H(X^2)$、$H(X^3)$ 和 $H(X^4)$ 进行比较。

当 $\mathscr{P}_X = \{0.6,0.4\}$ 时，对 X^2 和 X^4 重复此题。



习题 5.22[✍] 求一个概率分布 $\{p_1,p_2,p_3,p_4\}$，使 4 个符号有两种最优码，而为它们所分配的长度 $\{l_i\}$ 不同。

习题 5.23[✍] （习题 5.22 的延续） 假设 4 个概率 $\{p_1,p_2,p_3,p_4\}$ 按照 $p_1 \geqslant p_2 \geqslant p_3 \geqslant p_4 \geqslant 0$ 的顺序排列。设 $\mathscr{Q}$ 为具有两种不同长度最优码的所有概率向量 $\boldsymbol{p}$ 的集合。给出 $\mathscr{Q}$ 的一个完整描述。求解 $\mathscr{Q}$ 的凸包，即计算 3 个概率向量 $\boldsymbol{q}^{(1)}$、$\boldsymbol{q}^{(2)}$、$\boldsymbol{q}^{(3)}$，

使得任意 $\boldsymbol{p}\in\mathcal{O}$ 都可以写成

$$p=\mu_1\boldsymbol{q}^{(1)}+\mu_2\boldsymbol{q}^{(2)}+\mu_3\boldsymbol{q}^{(3)} \tag{5.29}$$

其中 $\{\mu_i\}$ 均为正。

▷ **习题 5.24**[1] 写出一个短小的评论，讨论如何玩“20 个问题”游戏是最优的。[在这 20 个问题中，玩游戏的人想好一个谜底，而另一个人用尽可能少的二进制问题去猜测，问题的数目最好不多于 20 个。]

▷ **习题 5.25**[2] 证明：如果每个概率 p_i 等于 2 的整数次幂，则存在期望长度等于熵的信源编码。

▷ **习题 5.26**[2] 构造一个总体，使得熵和霍夫曼码期望长度之间相差尽可能大。

▷ **习题 5.27**[2] 信源 X 的符号表由 11 个字符组成

$$\{a,b,c,d,e,f,g,h,i,j,k\}$$

每一个字符都以 1/11 等概率出现。

求解该信源的一个可唯一译码的最优符号码。此最优码的期望长度比 X 的熵大多少？

▷ **习题 5.28**[2] 考虑符号表大小为 I 的总体 X 的最优符号码，其中所有符号具有相同概率 $p=1/I$，I 不是 2 的幂。

证明：所分配码长等于

$$l^+\equiv\lceil\log_2 I\rceil \tag{5.30}$$

的部分在 I 中所占比例 f^+ 满足

$$f^+\equiv 2-\frac{2^{l^+}}{I} \tag{5.31}$$

最优符号码的期望长度为

$$L=l^+-1+f^+ \tag{5.32}$$

对相对于 I 的额外长度 $\Delta L\equiv L-H(X)$ 微分，证明额外长度的上界为

$$\Delta L\leqslant 1-\frac{\ln(\ln 2)}{\ln 2}-\frac{1}{\ln 2}=0.086 \tag{5.33}$$



习题 5.29[2] 考虑 $\mathcal{P}_X=\{0.99,0.01\}$ 的稀疏二进制信源。讨论如何利用霍夫曼码有效地压缩该信源。估计你所提出的方案需要多少码字。

▷ **习题 5.30**[2] Scientific American 杂志 1975 年刊出如下谜题。

有毒的玻璃杯 “数学家们是好奇的鸟，”警署专员对他的妻子说，“你看，我们在酒店餐桌上把所有这些半满的玻璃杯排成行。其中只有一杯装的是毒药，我们想知道究竟是哪一杯，然后再去搜索那个杯子上的指纹。实验室能够测试每个玻璃杯中的液体，但测试既花时间又花钱，所以我们想同时测试由其中某些玻璃杯中的少量液体样本所制成的混合物，以尽可能地减少测试次数。某大学派一位数学教授来帮助我们。他数了数玻璃杯，然后笑着说：

“你随便挑一杯吧，长官。我们先来测试它。”

我问:“这样做会不会浪费一次测试?”

他说:“不会,这是最佳方案的一部分。我们可以首先测试一个玻璃杯。具体是哪个都没有关系。”

“开始时共有多少个玻璃杯呢?”警官的妻子问。

“我不记得了,大概100到200之间吧。”

试问到底有多少个玻璃杯。

求解这个谜题,并解释为什么这位教授实际上是错误的,而指挥官是对的。实际上,确定哪一个玻璃杯装毒药的最优过程是怎样的?相对于最优过程而言,若采用该教授的方法,预期要浪费多少次测试?解释其与符号编码的关系。



习题5.31[♪] 假设采用C_3码对本章开头介绍的总体X中某一符号序列进行压缩。若从二进制编码序列$\boldsymbol{c}=c(x_1)c(x_2)c(x_3)\cdots$中随机选取1比特,这个比特为1的概率是多少?

C_3:

a_i	$c(a_i)$	p_i	$h(p_i)$	l_i
a	0	1/2	1.0	1
b	10	1/4	2.0	2
c	110	1/8	3.0	3
d	111	1/8	3.0	3

▷**习题5.32**[♪] 对编码符号表中有q个符号的情况,如何修改二进制霍夫曼编码方案才能得出最优符号码?(也称为“q基”问题。)

混合码

一种很有吸引力的想法是,根据几种符号码(为各个符号分配一个不同长度的码字)来构建一个“元码”,然后从一个码切换到另一个码,并选择为当前符号分配最短码字的那个码。非常清楚,这样做需要付出些代价。如果希望在两个码中做出选择,则必须用某种方法加长消息,以指示现在所用的是哪个码。如果只通过单一前导比特来表示这种选择,那么所得出的编码会是次优的,因为它是不完全的。(也就是说,它不满足Kraft等式。)



习题5.33[♪] 证明这种元码是不完全码,并解释为什么这种混合码是次优的。

5.9 习题解答

习题5.8的解答 是,$C_2=\{1,101\}$是可唯一译码的,尽管它不是一个前缀码,

因为两个不同的字符串不会映射成同一个字符串；只有码字 $c(a_2)=101$ 包含符号 0。

习题 5.14 的解答 我们想证明对于满足 Kraft 不等式的码字长度集合为 $\{l_i\}$ 的任何集合，总存在一个具有这种长度的前缀码。将码字想像为如图 5.8 所示的“码字超市”中用大小表示价格的一个码字，不难得出证明。设想用图 5.8 右侧的预算，从最短码字（也就是最贵的采购）开始，一次购买一个码字。从码字超市一侧，比如说上方开始，购买与所需长度相一致的第一个码字。继续沿超市往下走 2^{-l} 的距离，并购买下一个与所需长度相一致的码字，如此等等。因为码字长度越来越长，相应的间距越来越短，我们所购买的码字总可与上一次购买的码字相邻，因而不会浪费预算。所以，在第 I 个码字，就沿着超市前进 $\sum_{i=1}^{I} 2^{-l_i}$ 的距离；若 $\sum_{i=1}^{I} 2^{-l_i} \leqslant 1$，就可以购买到所有码字而不致超支。

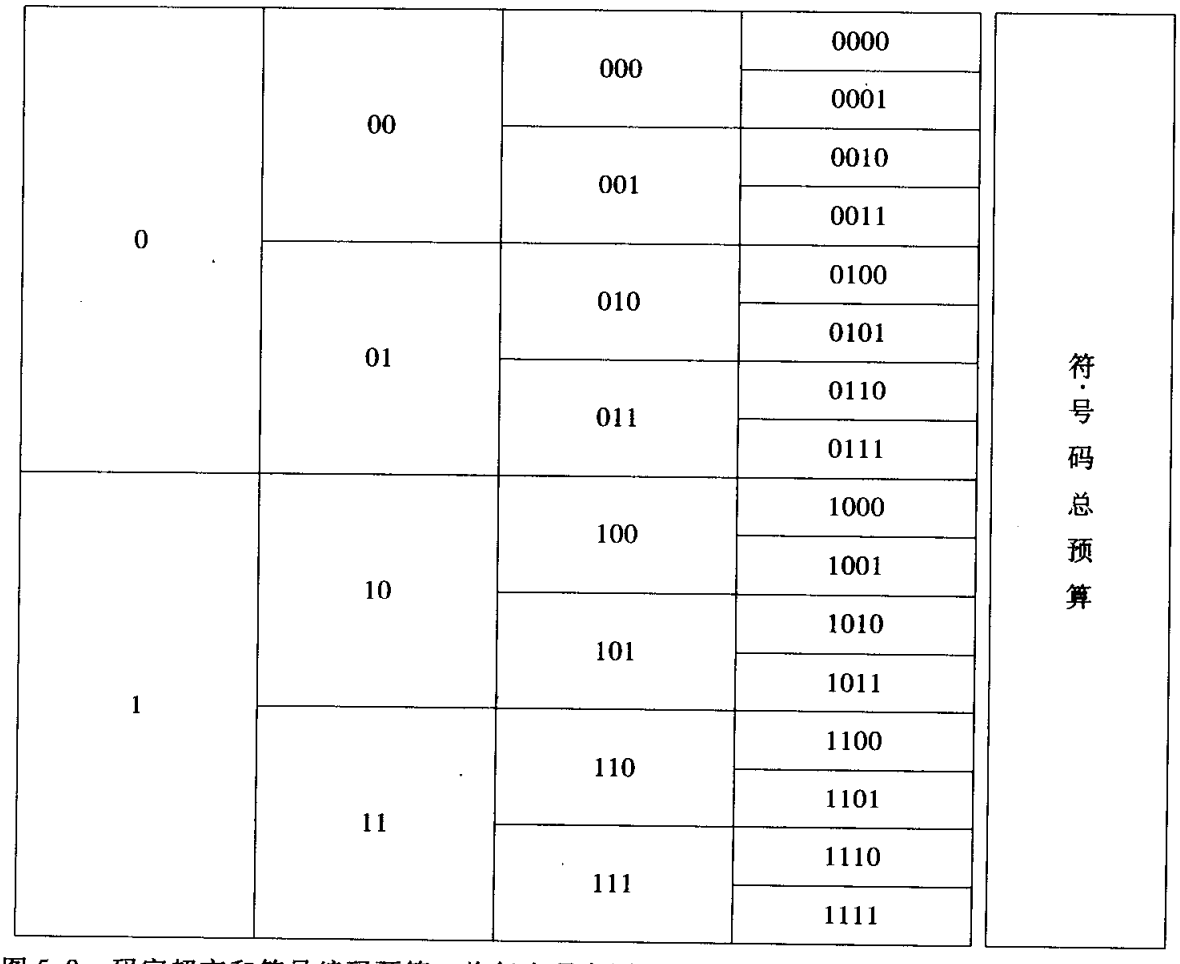


图 5.8 码字超市和符号编码预算。将每个码字写入一个格子，长度为 l 的码字的“费用”2^{-l} 用其相应的格子的大小来表示。在构造一个可唯一译码的码时，总的可用预算为 1 。

习题 5.16 的解答 对霍夫曼编码是最优编码的证明，依赖于算法中的一个关键步骤（即决定对具有最小概率的两个符号分配相等的编码长度），其所导致的期望长度不会比其他任何码大。可以采用反证法来证明这一点。

设具有最小概率的两个符号 a 和 b，霍夫曼算法给它们分配相等长度的码字，假

定在任何最优符号码中它们的码字长度不等。于是,最优符号码是另一种对手码,这两个码字具有不等的长度 l_a 和 l_b,且 $l_a < l_b$。不失一般性,假定这另一种码是完全前缀码,因为任何可唯一译码的码所对应的码长可以由前缀码实现。

在这种对手码中,必须有另外一个符号 c,其概率 p_c 比 p_a 大,其对手码中的长度大于或等于 l_b,因为符号 b 的码必须有一个长度相等或更大的邻近码字——完全前缀码永远不可能有一个最大长度的单独码字。

考虑交换码字 a 和 c(如图 5.9 所示),以便用比 c 更长的码字对 a 进行编码,而 c 比 a 更有可能得到更短的码字。显然,这样减小了码的期望长度。期望长度的变化是 $(p_a - p_c)(l_c - l_a)$。这样假定对手码是最优的说法出现了矛盾。因此对两个具有最小概率的符号分配相等编码长度的办法是有效的。霍夫曼编码产生最优符号码。 □

符号	概率	霍夫曼码字	对手码码字	改进的对手码
a	p_a	$c_H(a)$	$c_R(a)$	$c_R(c)$
b	p_b	$c_H(b)$	$c_R(b)$	$c_R(b)$
c	p_c	$c_H(c)$	$c_R(c)$	$c_R(a)$

图 5.9 证实霍夫曼编码给出一个最优符号码。假设一个对手码(据说也是最优码)对两个具有最小概率的符号 a 和 b 分配不等长度的码字。通过互换对手码的码字 a 和 c(这里符号 c 的对手码长与 b 相同),我们可以构造比对手码更好的一个码。这表明这个对手码不是最优的。

习题 5.21 的解答 当 $\mathcal{A}_X = \{0,1\}$,$\mathcal{P}_X = \{0.9, 0.1\}$ 时,X^2 的一个霍夫曼码是 $\{00,01,10,11\} \rightarrow \{1,01,000,001\}$。此码的 $L(C,X^2) = 1.29$,然而,熵 $H(X^2)$ 是 0.938。

X^3 的霍夫曼码是

$$\{000,100,010,001,101,011,110,111\} \rightarrow \{1,011,010,001,00000,00001,00010,00011\}$$

它的期望长度 $L(C,X^3) = 1.598$,而熵 $H(X^3)$ 为 1.406 9。

X^4 的霍夫曼码将 16 个信源字符串映射成以下码长

$$\{0000,1000,0100,0010,0001,1100,0110,0011,0101,1010,1001,1110,1101,1011,0111,1111\} \rightarrow \{1,3,3,3,4,6,7,7,7,7,7,9,9,9,10,10\}$$

它具有期望长度 $L(C,X^4) = 1.970\ 2$,而熵 $H(X^4)$ 为 1.876。

若 $\mathcal{P}_X = \{0.6, 0.4\}$,$X^2$ 的霍夫曼码的长度为 $\{2,2,2,2\}$;期望长度为 2 比特,熵为 1.94 比特。X^4 的霍夫曼码如表 5.10 所示。其期望长度为 3.92 比特,熵为 3.88 比特。

表 5.10 当 $p_0=0.6$ 时，X^4 的霍夫曼码。第 3 列表示所分配的码字长度，而第 4 列为码字。有些具有相同概率的字符串，例如第 4 个和第 5 个，得到不同的码字长度。

a_i	p_i	l_i	$c(a_i)$
0000	0.129 6	3	000
0001	0.086 4	4	0100
0010	0.086 4	4	0110
0100	0.086 4	4	0111
1000	0.086 4	3	100
1100	0.057 6	4	1010
1010	0.057 6	4	1100
1001	0.057 6	4	1101
0110	0.057 6	4	1110
0101	0.057 6	4	1111
0011	0.057 6	4	0010
1110	0.038 4	5	00110
1101	0.038 4	5	01010
1011	0.038 4	5	01011
0111	0.038 4	4	1011
1111	0.025 6	5	00111

习题 5.22 的解答 一组概率 $\{p_1,p_2,p_3,p_4\}=\left\{\frac{1}{6},\frac{1}{6},\frac{1}{3},\frac{1}{3}\right\}$ 导致了两个不同的最优码长集合，因为在霍夫曼编码算法的第二步中，可以选用 3 个可能配对中的任何一个。可以将它们放置在一个等长码 $\{00,01,10,11\}$ 或者码 $\{000,001,01,1\}$ 中。两种码的期望长度均为 2。

另一种答案是 $\{p_1,p_2,p_3,p_4\}=\left\{\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{2}{5}\right\}$。

第三种答案是 $\{p_1,p_2,p_3,p_4\}=\left\{\frac{1}{3},\frac{1}{3},\frac{1}{3},0\right\}$。

习题 5.26 的解答 设 p_{max} 为 $p_1,p_2,\cdots,p_I$ 中的最大概率。期望长度 L 与熵 H 之间的差别不会大于 $\max(p_{max},0.086)$（Gallager，1978）。

参见习题 5.27、习题 5.28，以理解奇怪的 0.086 从何而来的。

习题 5.27 的解答 长度 - 熵 = 0.086。

习题 5.31 的解答 有两种方法正确回答这一问题，而一种常见的做法回答不正确。先来看看不正确的答案。

错误答案 “首先选取概率为 p_i 的随机信源符号 x_i，然后从 $c(x_i)$ 中随机选取 1 比特，这样就随机地选取了 1 比特。若定义 f_i 为 $c(x_i)$ 中比特为 1 部分所占的比重，可以得到

$$P(\text{比特为 1}) = \sum_i p_i f_i \tag{5.34}$$

$$= \frac{1}{2} \times 0 + \frac{1}{4} \times \frac{1}{2} + \frac{1}{8} \times \frac{2}{3} + \frac{1}{8} \times 1 = \frac{1}{3} \tag{5.35}$$

C_3：

a_i	$c(a_i)$	p_i	l_i
a	0	$\frac{1}{2}$	1
b	10	$\frac{1}{4}$	2
c	110	$\frac{1}{8}$	3
d	111	$\frac{1}{8}$	3

这个答案是错的，因为犯了习题 2.35 所介绍的公交车站的谬论：如果公交车随机到达，我们感兴趣的是两辆车之间的平均间隙时间，必须区分两个可能的平均值：(a) 从随机选取的公交车到下一辆车到来之间的平均时间；(b) 从刚错过的一辆车到下一辆车到来之间的平均时间。第二种“平均”是第一种的两倍，因为在一个随机时刻等车时，你将客车选择倾向于那些间隔大的车了。你不太可能赶上一辆在前一趟车开走 10 秒钟后就到达的车。类似地，符号 c 和 d 的二进制字符串编码长度比 a 的编码长度大，因此当从已压缩的字符串中随机选取 1 比特时，它更可能是一个属于 c 或 d 的比特，而不是由期望式(5.34)中的概率 p_i 所给定。所有概率都必须扩大 l_i 倍，并重新归一化。

同一风格的正确答案 每次对符号 x_i 进行编码时，要增加 l_i 比特到二进制字符串中，该串有 $f_i l_i$ 个 1。在每个符号中加入 1 的个数的期望值为

$$\sum_i p_i f_i l_i \tag{5.36}$$

每个符号中加入的比特总数的期望值为

$$\sum_i p_i l_i \tag{5.37}$$

因此，传输字符串中为 1 的部分所占比例是

$$P(\text{比特为 1}) = \frac{\sum_i p_i f_i l_i}{\sum_i p_i l_i} \tag{5.38}$$

$$= \frac{\frac{1}{2} \times 0 + \frac{1}{4} \times 1 + \frac{1}{8} \times 2 + \frac{1}{8} \times 3}{\frac{7}{4}} = \frac{\frac{7}{8}}{\frac{7}{4}} = \frac{1}{2}$$

对于一般的符号码和一般的总体，期望值公式(5.38)都给出了正确答案。但在这种情况下，可以运用更有力的论据。

信息论答案 被编码的字符串 c 是一个最优压缩器的输出，该压缩器将 X 中的样本压缩到期望长度 $H(X)$ 比特。不能再期待对这些数据做进一步的压缩了。但是，如果概率 P(比特为 1)不等于$\frac{1}{2}$，则有可能将二进制字符串进一步压缩(比如，采用分组压缩码)。因此 P(比特为 1)必须等于$\frac{1}{2}$；事实上，压缩流中任意 l 比特序列取任何特定值的概率必须是 2^{-l}。完美压缩器的输出总是完全随机的比特。

换言之，如果概率 P(比特为 1)不等于$\frac{1}{2}$，则压缩字符串中每比特的信息量最多为 $H_2(P(1))$，它比 1 小；但这与能从 c 中恢复原始数据的事实相矛盾，因此，压缩字符串中每比特的信息量必须是 $H(X)/L(C,X)=1$。

习题 5.32 的解答 编码符号表有 q 个符号时的通用霍夫曼编码算法与二进制情况有一个不同点。将 q 个符号合并为 1 个符号的过程减少了 $q-1$ 个符号。因此，若从 A 个符号开始，只有当 $A \bmod (q-1)$ 等于 1 时，才会得到一个完全的 q 叉树。否则，无论采用哪种前缀码，必定是一棵所缺叶子数在模 $(q-1)$ 下等于 $A \bmod (q-1)-1$ 的不完全树。例如，若为 8 个符号创建一棵三叉树，则不可避免地要丢失一片叶子。

将这些额外的叶子放置在树的最远分支上，以构造出最优 q 元码。可以通过在原先的符号源集中增加适量的符号来实现，所有这些额外符号的概率均为 0。对于某个整数 r，叶子总数等于 $r(q-1)+1$。与在二进制霍夫曼编码算法中类似，选取 q 个具有最小概率的符号，用单一符号替代它们，从而可以不断地对符号进行合并。

习题 5.33 的解答 我们希望证明贪婪元码(它选取给出最短编码的码)实际上是次优的，因为它违反了 Kraft 不等式。

假定每个候选码 C_k 为每个符号 x 分配长度 $l_k(x)$。设有 K 个可选码，而且可以用长为 $\log K$ 比特的分组头来对所用的码进行编码。于是，元码分配长度 $l'(x)$ 为

$$l'(x) = \log_2 K + \min_k l_k(x) \tag{5.39}$$

计算 Kraft 和

$$S = \sum_x 2^{-l'(x)} = \frac{1}{K}\sum_x 2^{-\min_k l_k(x)} \tag{5.40}$$

将集合 $\mathscr{A}_X$ 划分成不重叠的子集 $\{\mathscr{A}_k\}_{k=1}^K$，使得子集 $\mathscr{A}_k$ 包含元码通过码 k 所发送的所有符号 x。于是有

$$S = \frac{1}{K}\sum_k \sum_{x\in\mathscr{A}_k} 2^{-l_k(x)} \tag{5.41}$$

如果一个子码 k 满足 Kraft 等式 $\sum_{x\in\mathscr{A}_X} 2^{-l_k(x)} = 1$，则必定为

$$\sum_{x\in\mathscr{A}_k} 2^{-l_k(x)} \leqslant 1 \tag{5.42}$$

只有当所有符号 x 都在 $\mathcal{A}_k$ 中时，才取得等号，这意味着仅使用了 K 个码之一。于是有

$$S \leqslant \frac{1}{K}\sum_{k=1}^{K} 1 = 1 \tag{5.43}$$

只有当式(5.42)对所有码 k 都取等号时，上式才取等号。但所有符号都在所有不重叠子集 $\{\mathcal{A}_k\}_{k=1}^{K}$ 中是不可能的，故式(5.42)不可能对所有 k 都取等号，因此 $S<1$。

看到混合码为次优码的另一种方法是，考虑它所定义的二叉树。想想两个码的特殊情况。所发送的第一个比特表明我们采用的是哪个码。在完全码中，随后的二进制字符串是一个有效串。但一旦知道用的是什么，比如码 A，也就知道了紧随其后的只能是对应于符号 x 的码字(符号 x 用码 A 编码后的长度比用码 B 时短)。因此，有些串是无效的延续，且混合码是不完全的和次优的。

对这个问题及其与概率建模关系的进一步讨论可参阅 28.3 节和 Frey(1998)中的“比特返回编码”部分。

关于第6章

在阅读第6章之前，应该已经阅读了上一章，并且已经完成了其中的大部分习题。

我们还将使用习题2.8等问题中的贝叶斯建模的思想。

第 6 章

符 号 流 码

在这一章，讨论两种数据压缩方法。

*算术编码*是一种优美的方法，它体现了压缩信源数据需要建立该信源的概率建模这样一种思想。截止到 1999 年，最好的文本文件压缩方法使用算术编码，而几种先进的图像压缩系统也使用它。

Lempel - Ziv *编码*是一种“通用”的方法，其设计思想是：用一种压缩算法以较好的性能来处理任何信源。事实上，对于许多实际的信源，这种算法的普遍适用特性只有当数据量大到难以处理时才成立，尽管如此，Lempel - Ziv 压缩仍然被广泛采用，而且一般还比较有效。

6.1 猜 字 游 戏

为了明白使用这两种压缩方法的动机，考虑典型英文文本文件的冗余度。这种文件在若干等级上存在冗余：例如，文件所包含的 ASCII 字符是不等概率；出现特定的连续字符对的可能性比其他的要大；给定上下文以及对文本的语义理解，可以预测出整个单词。

为了说明英文的冗余度以及它的一种奇特压缩方法，可以设想一个猜字游戏。在这个游戏中，一个说英语的人重复地试着去预测文本文件中的下一个字符。

为了简便起见，假设所允许的字母表包含 26 个大写字母 A，B，C，…，Z 和空格符“ - ”。这个游戏重复地请参与者猜测下一个字符，而唯一反馈信息是这种猜测是对还是错，直到正确地猜出这个字符。正确地猜出之后，我们记录猜出这个字符共用了多少次猜测，并且通过同样的方式请参与者猜测下一个字符。

在让一个人猜一个句子的时候，一句话给出如下结果。猜测次数列在每个字符的下面。

T H E R E - I S - N O - R E V E R S E - O N - A - M O T O R C Y C L E -

1 1 1 5 1 1 2 1 1 2 1 1 15 1 17 1 1 1 2 1 3 2 1 2 2 7 1 1 1 1 4 1 1 1 1 1

注意，在许多情况下，经过一次猜测后，下一个字母就立即被猜出。在其他情况下，特别是在音节开始处，则需要更多的猜测。

这个游戏和这些结果为我们提供了什么信息呢？首先，它们从一个说英语人的角度说明了英文的冗余性。其次，这个游戏可以用在如下数据压缩算法中。

上面所列的数字串'1,1,1,5,1,…'是通过将文本提交给游戏参与者的方式获得的。对于一个给定的字母,游戏参与者最多做 27 次猜测。于是,游戏参与者为我们所完成的工作就是,将 27 个字母{A,B,C,…,Z,-}时变地映射为 27 个数字{1,2,3,…,27},可以将后者视为新符号表中的符号。符号总数并未减少,但是由于一些符号(如 1 和 2)比另外一些用得更频繁,因此,压缩这个新符号串应该会容易些。

数字序列"1,1,1,5,1,…"的*无压缩*(uncompression)将如何进行呢?在无压缩期间,将没有原始的字符串"THERE…",我们只有已编码的序列。设想对于同一问题,那个游戏参与者有一个完全相同的双胞胎也在和我们一起玩猜字游戏,而我们已经知道信源文本。每当他的猜测次数达到给定的数目,我们让他停下来,然后让他猜出正确的字母,于是我们就说"对,你猜对了"并跳到下一个字母。如果没有双胞胎,在只有一个人帮助的情况下,也可按如下方式来设计一个压缩系统。我们选择一个窗口长度 L,它等于展示给这个人看的上下文中的字符数。对于长度为 L 的 27^L 种可能字符串中的每一个,我们问道,"你预测下一个字符是什么?","如果预测是错的,那么你的下一次猜测又将是什么?"在把他们对这 26×27^L 个问题的答案列成表之后,可以在编码器和译码器中使用两个巨大的表格来代替这两个双胞胎。这种语言模型就叫做 L 阶马尔科夫模型。

这些系统对于实际压缩而言,显然是不切实际的,但是它们说明了将要在下面使用的一些原理。

6.2 算术码

当我们讨论可变长度符号码以及用于构造它们的最优霍夫曼算法时,以指出霍夫曼码的两个实际和理论问题(第 5.6 节)作为结束。

这些缺点可以通过*算术码*来弥补,算术码是由 Elias、Rissanen、Pasco 发明的;后来,Witten 等人将其实用化(1987 年)。在算术码中,概率建模工作很明确地与编码操作分开来了。这个系统与猜字游戏非常相似。用信源的*概率模型*取代了人类预测器。随着信源产生每一个符号,概率模型提供下一个符号所有可能值的*预测分布*,即一列和为 1 的正数$\{p_i\}$。如果我们选择将信源建模成产生独立同等分布的且分布已知的符号,那么每次的预测分布都是相同的;但是算术编码也能同样简便地处理复杂自适应模型,这些模型能产生上下文相关的预测分布。预测模型一般用计算机程序实现。

编码器通过模型的预测来产生二进制符号串。译码器使用相同的模型(如同在猜字游戏中那样)来翻译二进制字符串。

设信源符号表为 $\mathcal{A}_X = \{a_1, \cdots, a_I\}$,让第 I 个符号 a_I 具有"传输结束"这一特别含义。信源发出序列 $x_1, x_2, \cdots, x_n, \cdots$。信源所产生的符号未必是独立同等分布的。我们将假设编码器使用计算机程序,在给定已发生序列的前提下,来分配 a_i 的预测

概率分布 $P(x_n = a_i | x_1, \cdots, x_{n-1})$。接收机有相同的程序，产生同样的预测概率分布 $P(x_n = a_i | x_1, \cdots, x_{n-1})$。

理解算术编码所需的概念

区间的记号 区间[0.01,0.10)是0.01和0.10之间的所有数，包括 $0.01\dot{0} \equiv 0.01000\cdots$，但不包括 $0.10\dot{0} \equiv 0.10000\cdots$。

二进制传输在实轴上定义了从0到1的一个区间(如图6.1所示)。例如，串01被翻译为二进制实数0.01…，它在二进制上对应于区间[0.01,0.10)，也就是十进制区间[0.25,0.50)。

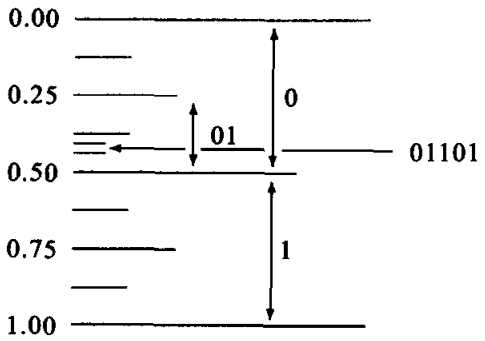


图6.1 二进制字符串定义了实线段[0,1)内的实区间。在第5章讨论符号码超市时，第一次碰到这样的图

长些的串01101对应于更小的区间[0.01101,0.01110)。因为01101有第一个字符串01作为前缀，新区间是区间[0.01,0.10)的子区间。因而，可以认为一个1 MB的二进制文件(2^{23}比特)指定了一个0～1之间的数字，该数字精确到十进制小数点后约200万位，因为每个字节翻译成十进制数时比二位数稍大。

现在，还可将实线[0,1)划分成 I 个长度等于概率 $P(x_1 = a_i)$ 的区间，如图6.2所示。

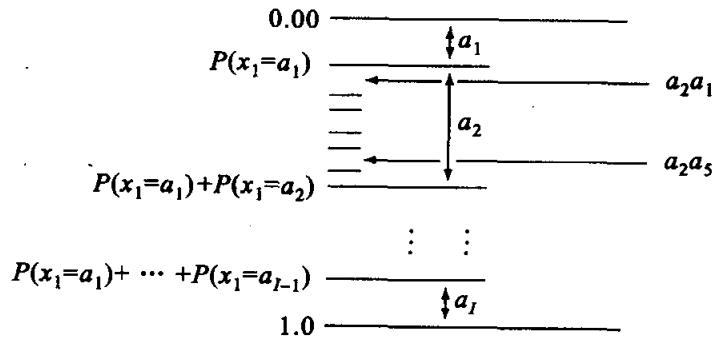


图6.2 概率模型定义了实线[0,1)内的实区间

可以将每个区间 a_i 进一步划分成子区间，并用 $a_ia_1, a_ia_2, \cdots, a_ia_I$ 表示，以使 a_ia_j 的长度与 $P(x_2 = a_j | x_1 = a_i)$ 成正比。事实上，区间 a_ia_j 的长度将正好是联合概率

$$P(x_1 = a_i, x_2 = a_j) = P(x_1 = a_i)P(x_2 = a_j | x_1 = a_i) \tag{6.1}$$

重复这个过程，区间[0,1)将被划分成区间序列，对应于所有可能的有限长度字符串 $x_1x_2\cdots x_N$，使得一个区间的长度等于在我们模型下的对应字符串的概率。

描述算术编码的公式

图 6.2 所描述的过程可以明确地通过如下公式写出。通过下累积概率和上累积概率来定义区间。

$$Q_n(a_i \mid x_1, \cdots, x_{n-1}) \equiv \sum_{i'=1}^{i-1} P(x_n = a_{i'} \mid x_1, \cdots, x_{n-1}) \tag{6.2}$$

$$R_n(a_i \mid x_1, \cdots, x_{n-1}) \equiv \sum_{i'=1}^{i} P(x_n = a_{i'} \mid x_1, \cdots, x_{n-1}) \tag{6.3}$$

随着第 n 个符号的到达，我们在 Q_n 和 R_n 所定义的各点上进一步划分第 $n-1$ 个区间。例如，由第一个字符开始，区间“a_1”、“a_2”和“a_I”为

$$a_1 \leftrightarrow [Q_1(a_1), R_1(a_1)) = [0, P(x_1 = a_1)) \tag{6.4}$$

$$a_2 \leftrightarrow [Q_1(a_2), R_1(a_2)) = [P(x = a_1), P(x = a_1) + P(x = a_2)) \tag{6.5}$$

和

$$a_I \leftrightarrow [Q_1(a_I), R_1(a_I)) = [P(x_1 = a_1) + \cdots + P(x_1 = a_{I-1}), 1.0) \tag{6.6}$$

算法 6.3 描述了一般的步骤。

```
u := 0.0
v := 1.0
p := v − u
for n = 1 to N {
    计算累计概率 Q_n 和 R_n (6.2,6.3)
    v := u + pR_n(x_n | x_1, ···, x_{n-1})
    u := u + pQ_n(x_n | x_1, ···, x_{n-1})
    p := v − u
}
```

算法 6.3 算术编码。为字符串 $x_1x_2\cdots x_N$ 寻找区间$[u,v)$的迭代过程。

要对字符串 $x_1x_2\cdots x_N$ 进行编码，找到对应于 $x_1x_2\cdots x_N$ 的区间，并发送位于那个区间内的二进制字符串。下面将说明，这种编码能够实时进行。

例子：对掷偏倚硬币所用次数进行压缩

设想将一枚偏倚（变形）硬币抛掷数遍，对其进行观察（参看例 2.7 和第 3.2 节）。硬币被抛掷的两个结果用 a 和 b 表示。第三种可能性是实验终止，用“文件结束”符号“□”来表示这一事件。因为硬币是变形的，我们期望结果 a 和 b 的概率是不相等的，虽然我们事先并不知道哪一结果更有可能发生。

编码

设信源字符串为“bbba□”。我们发出字符串的方式是一次只发送一个符号，并用模型计算在给定当前串这一前提下，下一个符号的概率分布。设这些概率为

上下文(当前序列)	下一个符号的概率		
	$P(a)=0.425$	$P(b)=0.425$	$P(□)=0.15$
b	$P(a\|b)=0.28$	$P(b\|b)=0.57$	$P(□\|b)=0.15$
bb	$P(a\|bb)=0.21$	$P(b\|bb)=0.64$	$P(□\|bb)=0.15$
bbb	$P(a\|bbb)=0.17$	$P(b\|bbb)=0.68$	$P(□\|bbb)=0.15$
bbba	$P(a\|bbba)=0.28$	$P(b\|bbba)=0.57$	$P(□\|bbba)=0.15$

图 6.4 所示为对应的区间。区间 b 是[0,1)的中间 0.425。区间 bb 是区间 b 的中间 0.567，等等。

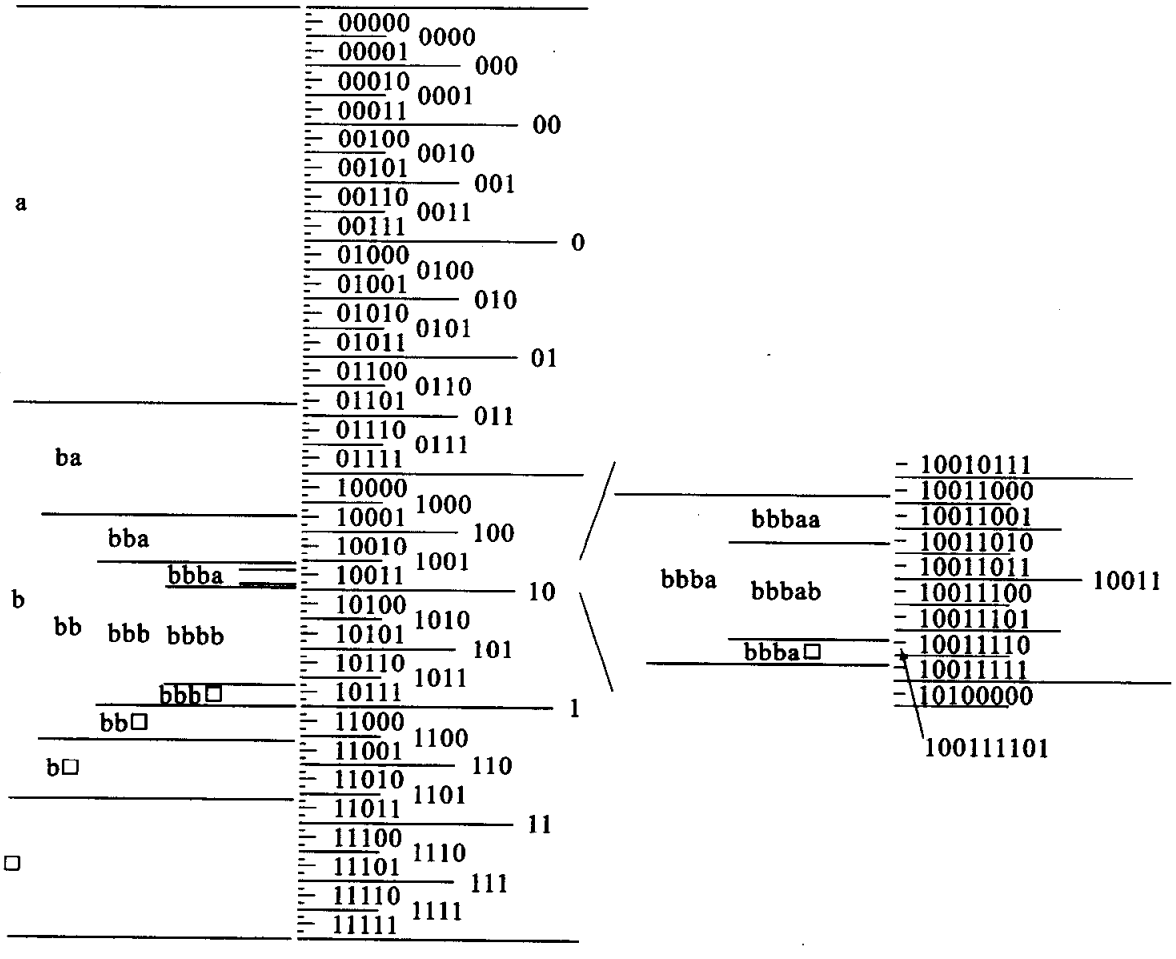


图 6.4 发送序列“bbba□”时，算术编码过程示意图。

在观察第一个符号“b”时，编码器知道已编码字符串将以“01”、“10”或“11”开头，但是不知道具体是哪个。编码器暂时不会做任何记录，而是去检查下一个符号“b”。区间“bb”全部位于区间“1”内，所以编码器可以写出第一个比特：“1”。第三个符号“b”使得区间又变窄一点，但还没有窄到可以完全放进区间“10”内。只有从信源读出下一个“a”之后，才能发送更多的比特。区间“bbba”完全位于区间“1001”内，所以编码器将“001”加入已写的“1”。最终当“□”到达时，需要终止编码的一个过程。放大区间“bbba□”（如图 6.4 右边所示），注意到标出来的区间“100111101”完全被 bbba□包含，所以可通过添加“11101”来完成编码。



习题 6.1[编] 证明相对于给定概率模型 $\mathcal{H}$ 的理想消息长度 $h(\boldsymbol{x}\mid\mathcal{H})=\log[1/P(\boldsymbol{x}\mid\mathcal{H})]$ 而言，终止一条消息所需的额外开销决不会超过 2 比特。

这是一个很重要的结果。算术编码非常接近最优。消息长度永远不会超过整个信源字符串的香农信息量 2 比特，所以期望消息长度也不会超过整个消息的熵 2 比特。

译码

译码器接收到字符串“100111101”，每次译出一个符号。首先使用与编码器相同的程序，计算概率 $P(\mathrm{a})$、$P(\mathrm{b})$ 和 $P(□)$，并推断区间“a”、“b”和“□”。一旦检测出最先的 2 比特“10”，就可以确定原字符串必定以“b”开始，因为区间“10”完全位于区间“b”内。那么译码器可以用模型来计算 $P(\mathrm{a}\mid\mathrm{b})$、$P(\mathrm{b}\mid\mathrm{b})$、$P(□\mid\mathrm{b})$，并且推导出区间“ba”、“bb”和“b□”的边界。继续工作，一旦收到“1001”，就译出第二个 b；而收到“100111”就译出第三个 b，等等。一旦读入整个二进制字符串，就可以毫不含糊地确定“bbba□”。由约定，“□”表示消息的结束，译码器由此得知可停止译码了。

多文件的传输

怎样才可能用算术编码在二进制信道中传输几个不同的文件？一旦发送了□字符，就假设译码器被复位成它的初始状态，不会将第一份文件的已知统计特性转换到第二份文件。但是，如果确实相信将要压缩的文件之间存在某种关系，可以用不同方法定义符号表，引入第二个文件结束符号来标识文件的结束，并且指示编码器和译码器以继续使用相同的概率模型。

全景图

注意，要传输 N 个字母所组成的字符串，编码器和译码器都只需计算 $N|\mathcal{A}|$ 个条件概率（每个可能的字母在实际遇到的每种上下文中的概率），就如同在猜字游戏中一样。可以将这里的开销与使用大分组长度的霍夫曼码时的开销进行对比。使用大分组长度是为了减少可能的每符号 1 比特的额外开销，如第 5.6 节所述。在霍夫

曼码中,必须考虑所有可能出现的分组序列,并且计算它们的概率。

注意算术编码有多么灵活:它能用于任何信源符号表和任何已编码字符。信源符号表和已编码字符表的大小可以随时间而变化。算术编码能使用任何概率分布,甚至允许概率分布随上下文彻底发生变化。

此外,如果喜欢使用非均等频度的编码符号表(如 0 和 1),也很容易安排,只要根据所要求的频度按比例地划分右边区间即可。

概率模型如何进行预测

算术编码技术并不强迫人们用任何特定的方法来生成预测概率,但是用贝叶斯模型可以很自然地产生预测分布。

图 6.4 是使用简单模型来产生的,这个模型总是给□分配概率 0.15,并根据拉普拉斯规则将剩下的 0.85 分配给 a 和 b,即按照与概率成正比的方式来划分

$$P_L(\mathrm{a}|x_1,\cdots,x_{n-1}) = \frac{F_a + 1}{F_a + F_b + 2} \tag{6.7}$$

其中,$F_a(x_1,\cdots,x_{n-1})$表示到目前为止 a 已经出现的次数,F_b 表示 b 的出现次数。这些预测对应于简单贝叶斯模型,它预测并自动适应文件中信源符号 a 和 b 的非均等使用频度。

图 6.5 所示是对应于长度不超过 5 的字符串的一些区间。注意,如果到目前为止的字符串中已包含大量的 b,那么 b 相对于 a 的概率要增加;而与之相反,如果出现了很多 a,那么就会让 a 变得更有可能。记住,越大的区间进行编码所需的比特越少。

贝叶斯模型的详情

已经强调了,可以使用任何模型(算术编码没有和任何特定概率集合锁定在一起),让我们解释一下前面例子所用的简单自适应概率模型;在习题 2.8 中首次遇到这个模型。

假设

这个模型将使用定义如下参数 $p_□$、p_a 和 p_b 来描述;不要将它们与特定上下文中的预测概率,如 $P(\mathrm{a}|s = \mathrm{baa})$,相混淆。用 a 和 b 表示一枚变形硬币的两面,将这枚硬币抛掷 l 次,次数事先并不为人所知。当抛掷硬币时,出现 a 的概率为 p_a,且 $p_b = 1 - p_a$;参数 p_a 和 p_b 事先也是不知道的。信源字符串 s = baaba□表明 l 为 5,并且结果序列为 baaba。

1. 假设字符串长度 l 服从指数概率分布

$$P(l) = (1 - p_□)^l p_□ \tag{6.8}$$

这个分布对应于如下假设:每个字符的终止符号为“□”的概率等于常量 $p_□$。

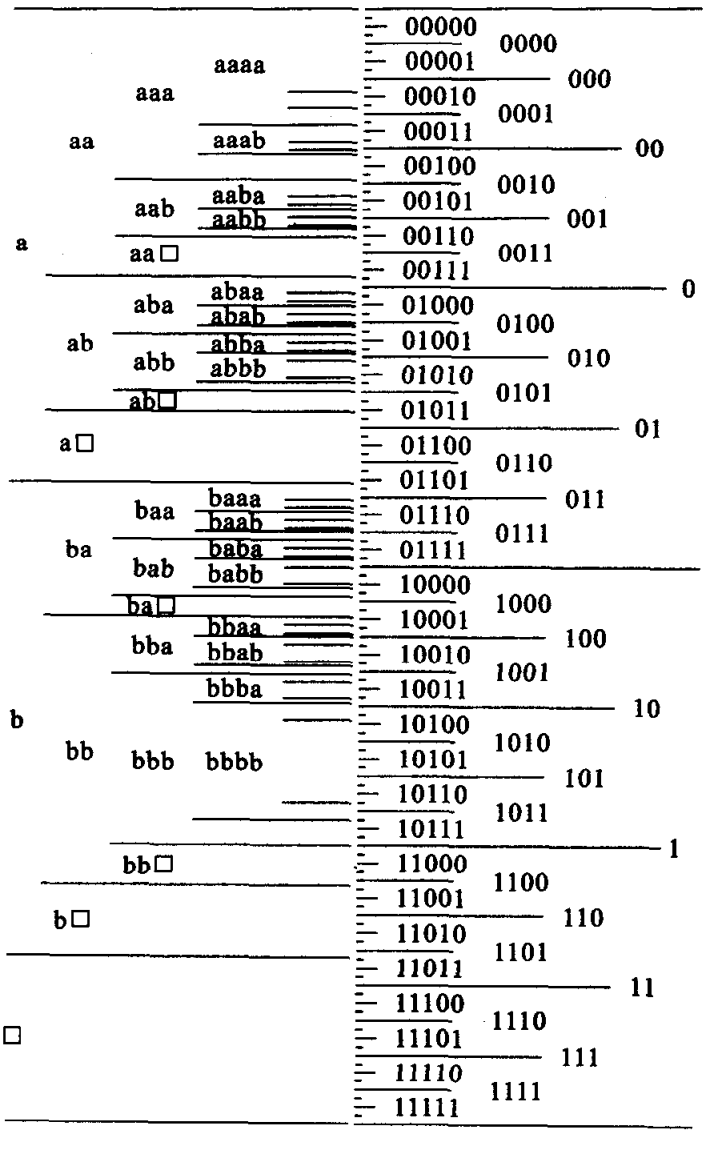


图 6.5 用一个简单贝叶斯概率模型定义区间的示意图。区间大小同相应字符串的概率成正比。这个模型预计信源可能会偏向于 a 和 b 之一，因此，有很多 a 或很多 b 的序列同具有相同长度的 a 和 b(50:50)的序列相比，具有更大的区间。

2. 假设字符串中的非终止字符从概率为 $\mathcal{P}=\{p_a,p_b\}$ 的总体中独立地随机选择；概率 p_a 在整个字符串中都是某个固定的未知值，它可以是 0 ~ 1 之间的任何值。给定 p_a(如果知道它的值)，下一个出现的符号是 a 的概率为 $(1-p_\square)p_a$。已知 p_a，若长度为 F 的未终止字符串是一个给定串 s，它包含的两个结果数为 $\{F_a,F_b\}$，那么出现该串的概率服从伯努利分布

$$P(s|p_a,F)=p_a^{F_a}(1-p_a)^{F_b} \tag{6.9}$$

3. 假设 p_a 的先验分布是均匀的

$$P(p_a)=1,\quad p_a\in[0,1] \tag{6.10}$$

并定义 $p_b \equiv 1 - p_a$。假设 p_a 的其他先验也很容易，用 Beta 分布最容易处理。

在第 3.2 节，已经研究过这个模型。我们所需的关键结果是，给定当前的字符串 s，下一符号的预测分布。下一符号为 a 或 b（假设不是“□”）的概率，在式(3.16)中推导过，而这正好就是拉普拉斯法则(6.7)。

▷ **习题 6.2**[p] 用如下三种方法对 ASCII 文件进行压缩，比较所期待的消息长度。

带头部的霍夫曼码 读取整个文件，发现每个符号的经验频度，对于这些频度构造一个霍夫曼码，通过发送码长来发送霍夫曼码字，然后用霍夫曼码发送文件。（不需要发送实际码字，因为给定码长，就可以用确定性方法来构建相应的树。）

使用拉普拉斯模型的算术码

$$P_L(a \mid x_1, \cdots, x_{n-1}) = \frac{F_a + 1}{\sum_{a'} (F_{a'} + 1)} \tag{6.11}$$

使用狄利克雷模型的算术码 这个模型的预测概率为

$$P_D(a \mid x_1, \cdots, x_{n-1}) = \frac{F_a + \alpha}{\sum_{a'} (F_{a'} + \alpha)} \tag{6.12}$$

其中 α 是一个固定数，如 0.01。一个小的 α 值所对应的拉普拉斯模型更加容易做出响应；字符的概率预计将会更加不均匀；$\alpha = 1$ 时，就变成了拉普拉斯模型。

注意你的霍夫曼码的信息头是自己划定界限的。值得考虑的特殊情况是(a)只包含几百个字符的短文件；(b) 有些字符从来都未曾使用过的大文件。

6.3 算术编码的进一步应用

随机样本的有效生成

算术编码不仅提供了一种方法，压缩据信是来自某给定模型的字符串；它同时也给出了由一个模型生成随机字符串的一种方法。设想将一根针随机地插入一个单位区间，而该区间已经被划分成与概率 p_i 成正比的子区间；你的针位于区间 i 的概率是 p_i。

所以要从模型生成一个样本，所需做的仅是将普通随机比特填充进该模型的算术译码器。一个无限随机比特序列对应于从线[0,1)上随机选取一个点，所以译码器将从假设的分布中随机地选择字符串。这种算术方法保证使用非常接近于最少的随机比特数来进行选择，这一点在随机数非常昂贵的场合很重要！[这不是一个玩笑。大量的金钱就花费在用软件和硬件生成随机比特上。随机数是昂贵的。]

使用这种方法的一个简单例子是，生成服从非均匀分布 $\{p_0, p_1\}$ 的随机比特。



习题 6.3[2] 比较下面两种生成服从非均匀分布 $\{p_0, p_1\} = \{0.99, 0.01\}$ 的随机

符号的方法。

(a) 标准方法:使用标准随机数生成器来产生 $1 \sim 2^{32}$ 之间的一个整数。再将该整数按比例缩放到(0,1)。测试这个均匀分布随机变量是否小于 0.99,并相应地发送 0 或 1。

(b) 使用正确模型进行算术编码,并输入标准随机比特。

要从这种稀疏分布中生成一千个样本,每种方法大致需要用到多少随机比特?

有效的数据输入器件

当我们将文本输入计算机时,需要做出某些动作:可能是敲击键盘,或用光笔涂写,或用鼠标点击;一个有效的文本输入系统要求:输入一个给定文本串所需的动作数很小。

书写可以看做数据压缩的逆过程。在数据压缩中,目标是将给定的文本串映射成数目较少的比特。在文本输入时,我们想要使用一系列较少的动作来产生所需要的文本。

通过逆转算术编码器,可以获得一个信息有效的文本输入器件,它由连续的指向动作来驱动(Ward 等,2000)。在这个叫做 Dasher 的系统中,使用者放大一个单元区间以找到与他们所期望的字符串对应的区间,这些字符串与图 6.4 具有相同的格式。一个语言模型(与文本压缩中所用的完全相同)控制区间的大小,使得概率高的字符串能被快速、容易地识别出来。经过一个小时的练习,一个初学者能用一个手指驱动 Dasher,每分钟写出 25 个单词,这大致是他们用 10 个手指在规则键盘上正常打字速度的一半。它甚至有可能无需用手,而对驱动器 Dasher 定向凝视,每分钟写出 25 个单词(Ward 和 Mackay,2002)。Dasher 是在各种平台上都可使用的免费软件。①

6.4 Lempel – Ziv 编码

Lempel – Ziv 算法广泛应用于数据压缩(如 compress 和 gzip 命令),它在思路上与算术编码不同。它没有将建模与编码分开,也没有机会进行显式建模。

基本 Lempel – Ziv 算法

这种压缩方法用指向先前出现的相同子串的指针来取代子串。例如,如果字符串为 1011010100010…,我们将它分解成一个有序的子串字典,其中子串是从未出现过的,如:λ,1,0,11,01,010,00,10,…。在字典中包含了空子串 λ 作为第一个子串,在字典中按照从信源出现的顺序来对子串进行排序。每个逗号之后,查看输入序列

① http://www.inference.phy.cam.ac.uk/dasher/

的下一部分直到我们已经读到先前没有被标识的子串。略加思索，就能确认这个子串比先前在字典中出现的一个子串长 1 比特。这意味着可这样对每个子串进行编码：给前缀的先前出现分配指针，并且发送字典中新子串不同于原有子串的额外比特。如果在第 n 比特，已列出了 $s(n)$ 个子串，那么可以用$\lceil \log_2 s(n) \rceil$比特为指针赋值。上面序列的编码如下表第 4 行所示（为清楚起见，包含了标点），最上几行表示信源字符串和 $s(n)$ 的值

信源子串	λ	1	0	11	01	010	00	10
$s(n)$	0	1	2	3	4	5	6	7
$s(n)_{二进制}$	000	001	010	011	100	101	110	111
（指针，比特）		(,1)	(0,0)	(01,1)	(10,1)	(100,0)	(010,0)	(001,0)

注意到所发送的第一个指针是空的，因为，假设在字典中只有一个子串，即符号串 λ，则不需要任何比特来传达“选择”该子串作为前缀的信息。已编码串为 10001110110000010000010。在这种简单情况下，所编成的码实际上是比信源字符串更长的串，因为信源字符串中不存在明显的冗余。

▷ **习题 6.4**[2] 证明：对于任何从$\{0,1\}^+$到$\{0,1\}^+$的可唯一译码的码，如果要使一些字符串变短，就必须使另外一些字符串变长。

上述算法使许多字符串变长的一个原因是因为它是低效的，它发送了不必要的比特；换言之，它的码是不完备的。一旦字典中的一个子串已经通过它的两个子串加入里面，那么可以确信它是不必要的（除了可能是我们协议中终止消息的一部分）；所以在那一点上，可以从子串字典中丢弃它，并且沿一个前缀重做全部子串，因而减短了随后指针消息的长度。等价地，可以在先前由父串所占用的点上把第二个前缀写入字典中。另一个不必要的额外开销是在这些情况下发送新的比特：一个前缀第二次被使用时，就能确知下一比特的身份。

译码

译码器在译码端又涉及相同的过程，它边对数据进行译码边构造子串字典。

▷ **习题 6.5**[2] 使用上述基本 Lempel - Ziv 算法对字符串

000000000000100000000000

进行编码。

▷ **习题 6.6**[2] 用基本 Lempel - Ziv 算法对已编码的字符串

001010111011001001000110101010000011

进行译码。

实用性

在前面的讲解中，尚未讨论终止字符串的方法。

Lempel - Ziv 算法有很多变种，都采用字典管理等同样的思想，但使用了不同的步骤。所得程序运行快，但是它们虽然有用，压缩英文文本的性能却赶不上算术编码中所设置的标准。

理论性质

与前 3 章所讨论的分组码、霍夫曼码和算术编码方法相比，Lempel - Ziv 算法无需知道任何的信源概率模型。但是，给定一个遍历的信源（也就是，在充分长的时间尺度上，无记忆的信源），可以证明 Lempel - Ziv 算法会渐近地将它压缩到信源熵。这就是为什么称它为“通用”压缩算法的原因。这个性质的证明，见 Cover 和 Thomas (1991 年)。

只要记住已经出现的子串，以便在它下次出现时有一个较短的名字，就可以实现压缩。对于大多数信源，实现这个通用性能所需的渐近时间尺度长得不切实际，因为需要记忆的典型子串数可能是巨大的。在实际应用中，算法的实用性能基于这样一个事实：许多文件包含特定的短字符序列的多次重复，这是很适合我们算法的一种冗余形式。

共同基础

我们已经强调算术编码和 Lempel - Ziv 编码在原理上的不同。尽管如此，它们还是有共同基础：从原理上说，能够设计自适应概率模型，因此能设计具有“通用性”的算术码，也就是能将某类中的任何信源渐近地压缩到其熵的一定因子内（最好是 1）。可是，在实际应用中，笔者认为只有信源的种类严格受限制时，才能构造出这种通用模型。能够发现任何信源的概率分布的通用压缩器，是一种通用的人工智能！而通用的人工智能现在尚不存在。

6.5 示 例

研究算术编码的一种交互式辅助工具 dasher. tcl 可供人们使用。①

由 Radford Neal② 编写的算术编码演示软件包由编码和译码模块组成，而用户可加入一个定义概率模型的模块。需要强调的是，不存在一种通用的算术编码压缩器；对于各种类型的信源，必须给出一种新的模型。Radford Neal 的软件包包含一个简单的自适应模型，它与 6.2 节中的贝叶斯模型很相似。使用拉普拉斯模型所得的结果应被视为基准，因为它是最简单的可能概率模型——它简单地假设文件中的字符独

① http://www. inference. phy. cam. ac. uk/mackay/itpr nn/softwareI. html

② ftp://ftp. cs. toronto. edu/pub/radford/www/ac. software. html

立取自一个固定总体。在读文件时,对符号$\{a_i\}$的计数$\{F_i\}$进行了缩放和舍入,使所有的计数都在1~256之间。

可用于包含文本和图像的文档的一种最新压缩器是DjVu,它使用了算术编码。① 它使用了一种对二进制符号表精心设计的近似算术编码器,称作Z-编码器(Bottou等,1998年)。DjVu比上述算术编码软件要快得多。Z-编码器的一个诀窍在于:自适应模型只是偶尔地进行自适应(节省计算机时间),而何时进行自适应的决定,则由算术编码器是否发出1比特来伪随机地控制。

二进制图像的JBIG图像压缩标准使用具有上下文相关模型的算术编码,它使用与拉普拉斯法则相似的规则来进行自适应。PPM(Teahan,1995年)是文本压缩的主要方法,它也使用算术编码。

存在很多基于Lempel-Ziv的程序。gzip是基于一种称作"LZ77"(Ziv和Lempel,1977年)的Lempel-Ziv版本。compress是基于"LZW"(Welch,1984年)的。根据笔者的经验,最好的是gzip,而compress,对于多数文件性能都要差一些。

bzip是一个块排序文件压缩器,它使用了一种称作Burrows-Wheeler变换(Burrows和Wheeler,1994年)的灵巧方法。这种方法不基于显式概率模型,且仅对大小超过几千字符的文件有较好效果;但实际上,对于字符上下文是该字符很好的预测器的文件,它也是非常有效的压缩器。②

文本文件的压缩

当压缩程序应用于包含本章文本的大小为20 942 B的LATEX文件时,表6.6给出了所用的机时(以秒计)和所得压缩。

表6.6 应用于文本文件的压缩算法的比较

方法	压缩时间/s	压缩大小(占20 942的百分比)	未压缩时间/s
拉普拉斯模型	0.28	12 974(61%)	0.32
gzip	0.10	8 177(39%)	**0.01**
compress	0.05	10 816(51%)	0.05
bzip		7 495(36%)	
bzip2		7 640(36%)	
ppmz		**6 800(32%)**	

① http://www.djvuzone.org/

② 网上有很多关于Burrows-Wheeler变换的信息。http://dogma.net/DataCompression/BWT.shtml

稀疏文件的压缩

有趣的是，gzip 并非总有这么好的效果。将这些程序应用于一个包含 10^6 个字符的文本文件，这些字符要么是 0 要么是 1，而且是 0 的概率为 0.99，是 1 的概率为 0.01。表 6.7 给出了所实现的压缩情况。拉普拉斯模型同这种信源非常匹配，于是标准的算术编码器实现了很好的性能，compress 紧跟其后，而 gzip 是最差的。该信源的理想模型是把文件压缩成 $10^6 H_2(0.01)/8 \simeq 10\ 100$ B。拉普拉斯模型压缩器未能达到这种性能，因为它只用 8 比特精度来实现。ppmz 压缩器达到最好的压缩，但是所需的计算机时间要多很多。

表 6.7　应用于 10^6 个字符（其中 99% 为 0，1% 为 1）的随机文件时，压缩算法的比较

方法	压缩时间/s	压缩大小/B	未压缩时间/s
拉普拉斯模型	0.45	14 143(1.4%)	0.57
gzip	0.22	20 646(2.1%)	0.04
gzip—最好 *	1.63	15 553(1.6%)	0.05
compress	0.13	14 785(1.5%)	0.03
bzip	0.30	10 903(1.09%)	0.17
bzip2	0.19	11 260(1.12%)	0.05
ppmz	533	**10 447(1.04%)**	535

6.6 小　结

在前面 3 章中，研究了三种类型的数据压缩码。

固定长度的分组码（第 4 章）　这是从固定数目的信源符号到固定长度的二进制消息的映射。只对很小一部分信源串进行编码。这些码对于将熵识别为可压缩性的度量是有趣的，但是它们没有什么实用价值。

符号码（第 5 章）　对于信源符号表中的每一个符号，符号码采用变长码，码长是由它的符号概率所确定的整数长度。对于一组已知的符号概率，霍夫曼算法构造了最优符号码。

每个信源字符串都有一个可唯一译码的编码，如果信源符号服从假设的分布，那么符号码将压缩到位于区间 $[H, H+1)$ 内的期望长度 L。信源的统计波动可能使实际长度比这个平均长度更长或更短。

如果信源不能与假设分布很好地匹配，那么在信源分布和码的隐含分布之间，平均长度将会增加一个相对熵 D_{KL}。对于熵值小的信源，每个信源符号至少要发出 1

比特;只能通过将信源数据分成码块这种繁笨的过程,才可实现低于每信源符号1比特的压缩。

符号流码 与符号码相比,符号流码的独特点在于,它们没有限于对从信源读入的每个符号都至少发出1比特。所以大量信源符号可以被编码成很少的几个比特。如果信源流因某种原因被分成块,只能通过使用符号码来获得这种性质。

- 算术码将概率模型和一种编码算法相结合,该算法可以通过[0,1)的子区间来识别每个字符串,子区间长度等于这个模型下字符串的概率。字符串 $\boldsymbol{x}$ 的压缩长度同给定概率模型下 $\boldsymbol{x}$ 的香农信息量匹配良好,在这种意义下,算术码几乎是最优的。算术码符合好的压缩需要*数据建模*这一基本思想,这里数据建模可具有自适应贝叶斯模型的形式。
- Lempel-Ziv码能记忆已出现过的字符串,在这种意义下,它们是自适应的。它们建立在这样一个基本思想上:我们并不知道信源的概率分布如何,并且我们需要一种压缩算法,该算法无论信源的分布如何,都能工作得出色。

如果压缩文件中的任何一个比特发生变化,算术码和Lempel-Ziv码都将无法正确译码。所以,如果所压缩文件要在有噪媒介上存储或传送,纠错码会很关键。在不可靠信道上进行可靠通信是第二部分的主题。

6.7 有关符号流码的习题



习题6.7[2] 描述一个算术编码算法,对长度为 N、权重为 K(即 K 个1和 $N-K$ 个0)的随机比特串进行编码,其中 N 和 K 是已知的。

对于 $N=5, K=2$ 的情况,详细给出所有长度1~5的信源子串所对应的区间。

▷**习题6.8**[2] 要确定从 N 个对象中选择 K 个对象,需要多少比特?(假设 N 和 K 已知,并且 K 个对象的选择未排序。)在不浪费随机比特的情况下,如何随机地做出这样的选择?

▷**习题6.9**[2] 二进制信源 X 发出独立同等分布的符号,其概率分布为 $\{f_0, f_1\}$,其中 $f_1=0.01$。为信源的3个连续样本所组成的字符串 $\boldsymbol{x}=x_1x_2x_3$ 找出一种最优的可唯一译码的符号码。

估算这种最优码的期望长度比3比特字符串 $\boldsymbol{x}$ 的熵大多少倍(精确到十进制小数点后一位)。[$H_2(0.01)\simeq 0.08$,其中 $H_2(x)=x\log_2(1/x)+(1-x)\log_2(1/(1-x))$。]

用一个算术码来压缩由信源 X 的1 000个样本所组成的字符串。估算压缩文件长度的均值和标准差。

▷**习题6.10**[2] 描述一种算术编码算法,生成长度为 N、密度为 f(即每个比特为1的概率是 f)的随机比特串,其中 N 是已知的。

习题 6.11[2] 使用 6.4 节所讨论过的一种改进 Lempel - Ziv 算法，通过将新的前缀写入不再需要的前缀所占用的空间，来删改前缀字典。当一个前缀的两个子结点都已加入前缀字典中时，就可识别出这种前缀。（你可以忽略编码终止的问题。）使用这种算法对字符串 0100001000100010101000001 进行编码。特别注意在第二次所用前缀之后的比特。（如前所述，这些比特可被忽略。）

习题 6.12[2] 证明改进 Lempel - Ziv 码仍然是不“完备”的，也就是说，存在一些不是任何字符串的编码的二进制字符串。

▷ **习题 6.13**[3] 举几个例子说明，有些简单信源的熵值很低，却不能用 Lempel - Ziv 算法很好地进行压缩。

6.8 关于数据压缩的进一步习题

对于想尽快学习噪声信道的读者，可以跳过下列习题。



习题 6.14[2] 考虑 N 维高斯分布

$$P(\boldsymbol{x}) = \frac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\frac{\sum_n x_n^2}{2\sigma^2}\right) \tag{6.13}$$

定义点 $\boldsymbol{x}$ 的半径为 $r = (\sum_n x_n^2)^{1/2}$。估算半径的平方 $r^2 = (\sum_n x_n^2)$ 的均值和方差。

你可能会觉得积分

$$\int \mathrm{d}x \frac{1}{(2\pi\sigma^2)^{1/2}} x^4 \exp\left(-\frac{x^2}{2\sigma^2}\right) = 3\sigma^4 \tag{6.14}$$

有所帮助，虽然没有它，你应该也能计算出所要求的量。

假设 N 很大，证明几乎所有高斯概率都包含在一个半径为 $\sqrt{N}\sigma$ 的薄圆壳中。求出这个圆壳的厚度。

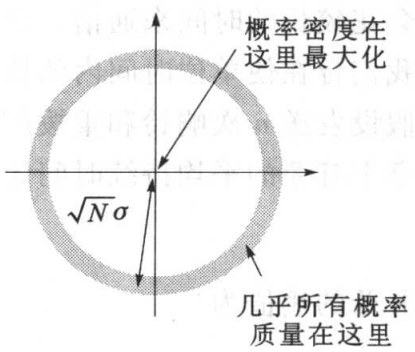


图 6.8 N 维高斯分布的典型集示意图

计算此薄圆壳内的一点和原点 $\boldsymbol{x} = \boldsymbol{0}$ 处的概率密度式(6.13)，并对其进行比较。以 $N = 1\,000$ 的情况为例。

注意：几乎所有概率质量均位于最高概率密度所在区域的不同空间部分（如图 6.8 所示）。



习题 6.15[2] 解释最优二进制符号码意味着什么。找出如下总体的最优二进制符号码。

$$\mathcal{A} = \{\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d},\mathrm{e},\mathrm{f},\mathrm{g},\mathrm{h},\mathrm{i},\mathrm{j}\}$$

$$\mathcal{P}=\left\{\frac{1}{100},\frac{2}{100},\frac{4}{100},\frac{5}{100},\frac{6}{100},\frac{8}{100},\frac{9}{100},\frac{10}{100},\frac{25}{100},\frac{30}{100}\right\}$$

并计算这个码的期望长度。



习题 6.16[2] 字符串 $y=x_1x_2$ 由来自如下总体的两个独立样本组成。

$$X:\mathcal{A}_X=\{a,b,c\};\mathcal{P}_X=\left\{\frac{1}{10},\frac{3}{10},\frac{6}{10}\right\}$$

字符串 y 的熵是多少？构建字符串 y 的最优二进制符号码，并计算它的期望长度。



习题 6.17[2] 使用与总体相匹配的算术码，对概率为 $\mathcal{P}=\{0.1,0.9\}$ 的总体中的 N 个独立样本的字符串进行压缩。对于 $N=1\,000$ 的情况，估算这个被压缩字符串长度的均值和标准差[$H_2(0.1)\simeq 0.47$]。



习题 6.18[2] **可变长度符号的信源编码**

在关于信源编码的章节中，我们假设编码成二进制符号表{0,1}，其中两个符号的使用频率应相同。在本题中，将探讨如果采用不同次数发送符号，如何使用编码符号表。

一个贫困的学生用电话铃声与朋友进行免费通信。他从{1,2,3,…}中选择一个整数 n 来决定朋友的电话铃响的次数，并在第 n 次时挂断电话。重复这个过程，并得到一个符号串 $n_1n_2n_3\cdots$。请问通信的最佳方式是什么？如果选择大整数 n，那么消息会花较长的时间来通信。如果使用小整数 n，那么每个符号所携带的信息量较小。我们旨在使单位时间内的信息传送速率最大。

假设发送 n 次响铃和重拨所花费的时间为 l_n 秒。考虑 n 的一个概率分布 $\{p_n\}$，定义每个符号的平均持续时间为

$$L(\boldsymbol{p})=\sum_n p_n l_n \tag{6.15}$$

每个符号的熵值为

$$H(\boldsymbol{p})=\sum_n \boldsymbol{p}_n\log_2\frac{1}{p_n} \tag{6.16}$$

证明如果要使每秒的平均信息率最大化，则符号必须使用如下概率形式

$$p_n=\frac{1}{Z}2^{-\beta l_n} \tag{6.17}$$

其中 $Z=\sum_n 2^{-\beta l_n}$，β 满足隐式方程

$$\beta=\frac{H(\boldsymbol{p})}{L(\boldsymbol{p})} \tag{6.18}$$

也就是说，β 是通信速率。证明这两个方程(6.17,6.18)意味着必须设置 β 以满足

$$\log Z=0 \tag{6.19}$$

假设信道具有性质

$$l_n = n \text{ 秒} \tag{6.20}$$

求出最优分布 $\boldsymbol{p}$，并证明最大信息速率为每秒 1 比特。

如果将 $\boldsymbol{p}$ 设为 $(1/2,1/2,0,0,0,0,\cdots)$，即只选择符号 $n=1$ 和 $n=2$，并且它们具有相同的概率，所达信息速率同最大信息速率相比，情况如何？

讨论上面所得结果式(6.17,6.19)和信源编码理论中 Kraft 不等式之间的关系。

要在方程(6.20)所定义的信道上传输随机二进制信源，如何有效地将此信源编码成符号序列 $n_1 n_2 n_3 \cdots$？

▷ **习题 6.19**[1] 洗一副扑克牌需要多少比特？

▷ **习题 6.20**[2] 在桥牌比赛中，4 个玩家各得 52 张牌中的 13 张，并且通过查看自己手上的牌和叫牌来开始游戏。合法的叫牌按升序排列为：1 梅花、1 方块、1 红桃、1 黑桃、1 无将、2 梅花、2 方块、…、7 红桃、7 黑桃、7 无将。连续的叫牌必须按照这个顺序进行；比如，叫了 2 红桃后，下家只能按照更高的顺序叫牌，如 2 黑桃、3 梅花或 7 无将。（我们忽略“双重”叫牌。）

玩家叫牌有多种目的。目的之一是使两个搭档尽可能地让对方了解自己手中抓到了哪些牌。

现在集中考虑这个任务。

(a) 发玩牌后，北方要把自己手中牌的信息传递给南方，需要多少比特？

(b) 假设东方和西方完全没有叫牌，那么北方和南方在叫牌时所能传递的最大总信息量是多少？假设北方开始叫牌，且只要北方或者南方停止叫牌，则叫牌过程结束。

▷ **习题 6.21**[2] 我的旧“阿拉伯”微波炉有 11 个按键用于设置烹饪时间，而我的新“罗马”微波炉只有 5 个按键（如图 6.9 所示）。“罗马”微波炉的按键分别标为“10 分钟”、“1 分钟”、“10 秒钟”、“1 秒钟”和“开始”；我将这 5 行简化为符号 M、C、X、I、□。要输入 1 分 23 秒(1:23)，“阿拉伯”的按键序列是

$$123\square \tag{6.21}$$

而“罗马”的按键序列是

$$\text{CXXIII}\square \tag{6.22}$$

每个按键面板定义一个编码，将 0:01 到 59:59 之间的 3 599 个烹饪时间映射为一串符号。

(a) 哪些时间可以用 2 个或 3 个符号表示？（比如，0:20 在每一编码中都可以用 3 个符号表示：XX□和 20□。）

(b) 两个编码是完备的吗？给出详细的答案。

(c) 对于每个编码，找出一个烹饪时间，它可以用 4 个符号表示，而另一种编码方式则做不到。

(d) 讨论时间上的隐式概率分布，它跟每种码都最匹配。

(e) 设想一个实际用户可能使用的时间上的可能概率分布，并对每一编码粗略

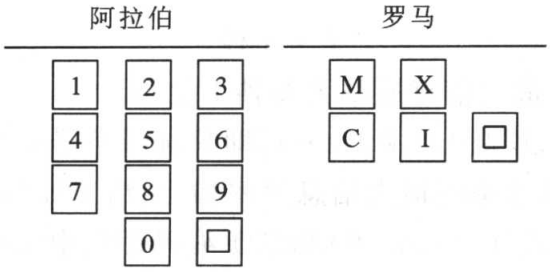


图 6.9 微波炉的两种不同按键面板

地计算所期望的符号数和所需的最大符号数。讨论每一编码方式效率的高低。

(f) 为微波炉设计一种更有效的烹饪时间编码系统。

习题 6.22[2] 正整数的标准二进制表示(如 $c_b(5)=101$)是不是可唯一译码的码?

为正整数设计一个可唯一译码的二进制码,即从 $n\in\{1,2,3,\cdots\}$ 映射到 $c(n)\in\{0,1\}^+$。尽量使所设计的码为前缀码,且满足 Kraft 等式 $\sum_n 2^{-l_n}=1$。

目的:以文件字符的特殊结尾结束的任何数据文件可被映射为一个整数,所以一个整数的前缀码也可用做文件的自定界编码。大型文件对应于大整数。而且,"通用"编码方案(即适用于各种信源的编码方案)的要素之一就是整数编码能力。最后,在微波炉中,烹饪时间是正整数!

讨论用于比较整数的不同编码(或者等价地,文件的自定界编码)的判据。

6.9 习题解答

习题 6.1 的解答 最坏情形出现在所表示的区间正好位于一个二进制区间内时。在这种情况下,可以选择如图 6.10 所示两种二进制区间中的任何一种。这些二进制区间不小于 $P(\boldsymbol{x}\mid\mathscr{H})/4$,所以二进制编码的长度不大于 $\log_2 1/P(\boldsymbol{x}\mid\mathscr{H})+\log_2 4$,比理想消息长度多 2 比特。

习题 6.3 的解答 标准方法对每个生成符号使用 32 个随机比特,因此生成 1 000个样本需要 32 000 比特。

算术编码对每个生成符号平均使用 $H_2(0.01)=0.081$ 比特,所以需要 83 比特来生成 1 000 个样本(假设结束过程约需 2 比特的额外开销)。

1 的数量波动将在均值附近引起标准差为 21 的变化。

习题 6.5 的解答 编码是 01010011001011000 1100,它来自如下分解

$$0,00,000,0000,001,00000,000000 \tag{6.23}$$

由其编码得

$$(0),(1,0),(10,0),(11,0),(010,1),(100,0),(110,0) \tag{6.24}$$

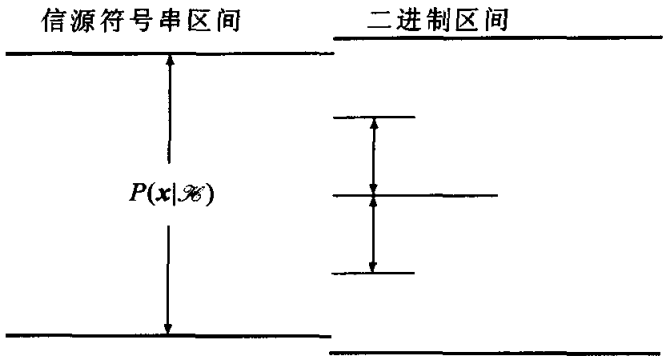


图 6.10 算术编码在最坏情况下终止，其中有 2 比特的额外开销。在右边标出的两个二进制区间中的任何一个都可能被选中。这些二进制区间不小于 $P(x|\mathscr{M})/4$。

习题 6.6 的解答 译码是 010000100010000101010000001。

习题 6.8 的解答 此题和习题 6.7 等价。

从 N 个物体中选择 K 个物体，需要$\lceil \log_2 \binom{N}{K} \rceil$比特$\simeq NH_2(K/N)$比特。可用算术编码来进行这种选择。这种选择对应于一个长度为 N 的二进制串，其中比特 1 表示哪个物体被选上。起初，1 的概率是 K/N，0 的概率是 $(N-K)/N$。然后，假定到目前为止已产生的字符串长度为 n，包含 k 个 1，那么出现 1 的概率是 $(K-k)/(N-n)$，出现 0 的概率是 $1-(K-k)/(N-n)$。

习题 6.12 的解答 改进的 Lempel - Ziv 码仍然是不“完备”的，因为，比如，在已经收集了 5 个前缀之后，指针可以是如下串之一：000，001，010，011，100，但不会是 101、110 或 111。因此有些二进制串不能由编码产生。

习题 6.13 的解答 熵值较低且不能很好地用 Lempel - Ziv 算法进行压缩的信源包括：

(a) 有些符号具有远距离相关性并夹杂了随机无用数据，这样的符号组成了一定的信源。理想模型必须捕捉到相关性并对其进行压缩。Lempel - Ziv 算法只能通过记录所有夹杂的无用数据来压缩相关性。作为一个简单例子，考虑这样一个电话簿，它的每行都包含(旧号码，新号码)对：

285 - 3820:572 - 5892□

258 - 8302:593 - 2010□

每行的字符数为 18，这些字符来自 13 - 字符的符号表{0，1，… ，9，-，:，□}。字符“-”、“:”和“□”以可预测的顺序出现，所以，假设所有电话号码是 7 位长，并假设是随机序列，那么每行的真实信息量为 14 ban。(每个 ban 等于 0 ~ 9 的随机整数所含信息量。)有限状态语言模型能轻易捕捉这些数据中所存在的规律性。可是，Lempel - Ziv 算法需要花费较长的时间才能将这样一个文件压缩到每行 14 ban，因为它必须查看所有这些字符串，才“知道”对于任意 3 位数字 ddd，字符串 ddd 后面总是

跟着字符“-”。所以,只有读取文件中的上千行内容之后,才能取得接近最优的压缩。

(b) 具有远距离相关性的信源,例如,用像素序列逐行表示的二维图像,使得垂直相邻像素在信源符号流中的距离为 w,其中 w 是图像宽度。例如,考虑图像的传真传输,其中每行都与上一行非常相似(图 6.11)。真实的熵仅仅是每像素 $H_2(f)$,其中 f 是一个像素同其前面出现的像素不同的概率。当所有长度为 $2^w = 2^{400}$ 的字符串都出现过,并且它们的后续串被记住时,Lempel-Ziv 算法才能压缩到熵值。宇宙中只有 2^{300} 个粒子,所以我们能相当有把握地说 Lempel-Ziv 码将永远不可能捕捉到这样一张图像的冗余度。

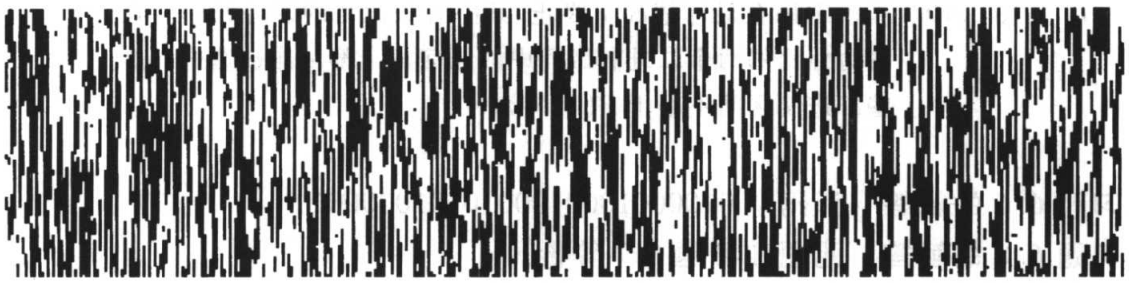


图 6.11 熵值较低且不能很好地用 Lempel-Ziv 算法进行压缩的信源。从左至右读取比特序列。每一行和它的上一行只有 $f = 5\%$ 的比特不同。该图形的宽度为 400 像素。

另一个高度冗余的结构如图 6.12 所示。这幅图通过在平面上随机地扔下水平和竖直方向的别针来获得。它同时包含远距离的垂直相关和远距离的水平相关。如果逐个像素地扫描这幅图像,并运用 Lempel-Ziv 算法,则不存在切实可行的方法来捕捉这些相关性。

生物计算系统能很容易地识别出这些图像甚至更复杂图像中的冗余;因此我们预计最好的数据压缩算法将源自人工智能方法的发展。

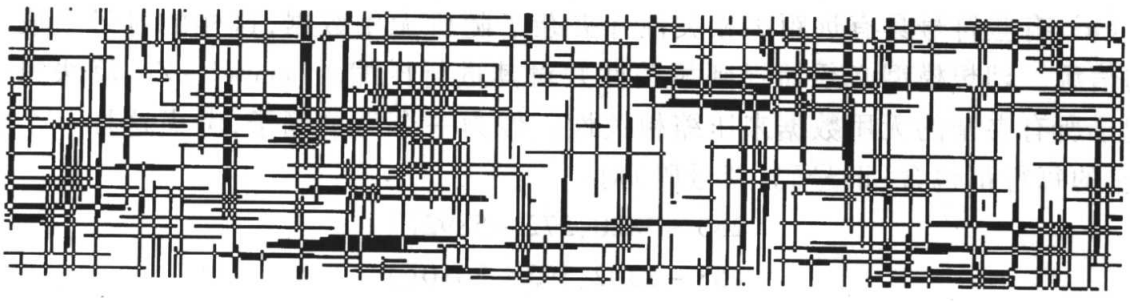


图 6.12 在平面上随机地扔下水平和竖直方向的别针所获得的结构图

(c) 具有复杂冗余度的信源,如计算机所生成的文件。例如,编码成 PostScript 文件的 LATEX 文件。这两个文件的信息量近似地等于 LATEX 文件单独的信息量。

(d) Mandelbrot 集的图像。图像的信息量等于指定所研究的复平面范围所需的比特数、像素大小和所使用的着色规则。

(e) 如图 6.13 所示为受抑的反铁磁性伊辛模型的基状态图,将在第 31 章讨论这种模型。与图 6.12 类似,这个二进制图像在两个方向上具有有趣的相关性。

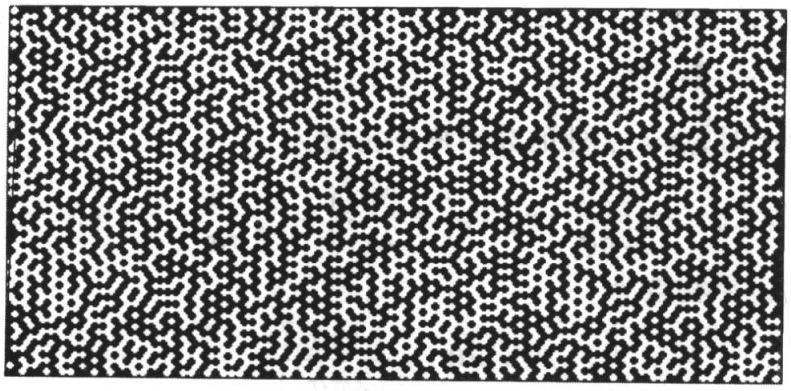


图 6.13 处于一种基状态的受抑三角伊辛模型

(f) 细胞自动机——图 6.14 所示为 400 个细胞的细胞自动机的 100 步状态史。随机地选择更新规则,根据这些规则,每个细胞的新状态依赖于前 5 个细胞的状态。信息量等于边界的信息量(400 比特)和传播规则,后者在这里可用 32 比特描述。一个最优压缩器所给出的压缩文件长度,几乎是一个独立于图像的垂直高度的常量。一旦细胞自动机进入周期性的极限环中,而这轻易就要花费 2^{100} 次迭代,那么 Lempel – Ziv 码才会给出零代价的压缩。

与之相反,JBIG 压缩方法却能很好地压缩这些图像,它建立了给定周边环境时像素概率的模型,并使用算术编码。

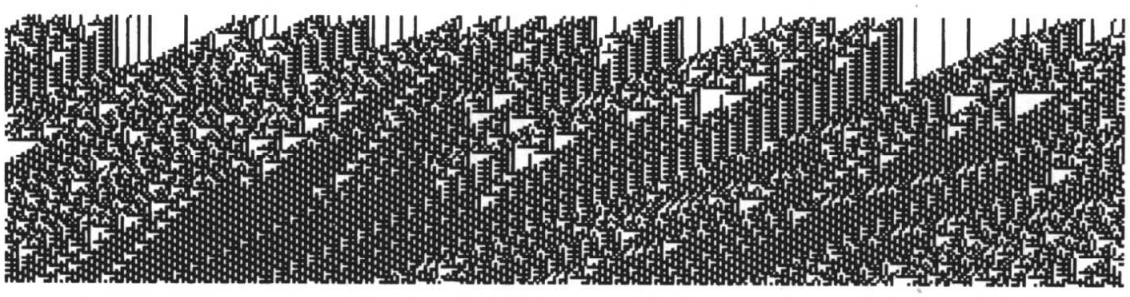


图 6.14 400 个细胞的细胞自动机的 100 步时间史

习题 6.14 的解答 一维高斯变量 x 的方差 $\mathscr{E}[x^2]$ 是 σ^2。由于 $\boldsymbol{x}$ 的元素是独立随机变量,所以 N 维时 r^2 的均值为

$$\mathscr{E}[r^2] = N\sigma^2 \tag{6.25}$$

类似地,r^2 的方差为 x^2 方差的 N 倍,其中 x 是一维高斯变量。

$$\mathrm{var}(x^2) = \int \mathrm{d}x \frac{1}{(2\pi\sigma^2)^{1/2}} x^4 \exp\left(-\frac{x^2}{2\sigma^2}\right) - \sigma^4 \tag{6.26}$$

可求得积分部分等于 $3\sigma^4$(式(6.14)),所以 $\mathrm{var}(x^2) = 2\sigma^4$,因此 r^2 的方差为 $2N\sigma^4$。

对于 N 很大的情形,中心极限定理表明 r^2 为高斯分布,它的均值为 $N\sigma^2$,标准差

为 $\sqrt{2N\sigma^2}$，所以 r 的概率密度必须近似集中于 $r \simeq \sqrt{N}\sigma$。

通过将 r^2 的标准差变换成 r 的标准差，可以给出这个圆壳的厚度：对于小的 $\delta r/r$，$\delta \log r = \delta r/r = (1/2)\delta \log r^2 = (1/2)\delta(r^2)/r^2$ 所以设 $\delta(r^2) = \sqrt{2N\sigma^2}$，$r$ 有标准差 $\delta r = (1/2)r\delta(r^2)/r^2 = \sigma/\sqrt{2}$。

在当 $r = \sqrt{N}\sigma$ 时，点 $\boldsymbol{x}_{\text{shell}}$ 处的高斯概率密度为

$$P(\boldsymbol{x}_{\text{shell}}) = \frac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\frac{N\sigma^2}{2\sigma^2}\right) = \frac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\frac{N}{2}\right) \tag{6.27}$$

但是在原点处的概率密度为

$$P(\boldsymbol{x}=0) = \frac{1}{(2\pi\sigma^2)^{N/2}} \tag{6.28}$$

因此 $P(\boldsymbol{x}_{\text{shell}})/P(\boldsymbol{x}=0) = \exp(-N/2)$。在典型半径的概率密度比在原点的密度小 $e^{-N/2}$ 倍。假如 $N = 1\,000$，那么在原点的概率密度会大 e^{500} 倍。

第 7 章

整 数 的 码

这一章是可选的，跳过此章不会影响对其他部分的理解。

习题 6.22 的解答

为了讨论整数的编码，需要给出一些定义。

正整数 n 的标准二进制表示记做 $c_b(n)$，如 $c_b(5)=101$，$c_b(45)=101101$。

正整数 n 的标准二进制长度为字符串 $c_b(n)$ 的长度，记做 $l_b(n)$。如 $l_b(5)=3$，$l_b(45)=6$。

因为无法确定一个整数何时结束，故标准二进制表示 $c_b(n)$ 不是整数的可唯一译码的信源码。例如，$c_b(5)c_b(5)$ 与 $c_b(45)$ 相同。如果在收到之前，就确知每个整数的标准二进制长度，则它是可唯一译码的。

注意到所有正整数都有一个始于 1 的标准二进制表示，还可以定义另一种表示。

正整数 n 的无头二进制表示记做 $c_B(n)$，例如，$c_B(5)=01$，$c_B(45)=01101$，$c_B(1)=\lambda$（其中 λ 表示空字符串）。

若知道整数的长度 $l_b(n)$，这种表示将是可唯一译码的。

那么，如何得出整数的可唯一译码的码呢？有两种突出的策略。

1. **自定界符号码**。首先传送整数的长度 $l_b(n)$，它也是一个正整数；然后用 $c_B(n)$ 传送原整数 n 本身。

2. **带“文件结束符”的码**。将整数编码成长度为 b 比特的块，并预留 2^b 个符号之一来表示“文件结束”这一特殊含义。将整数编码成块，使得已预留的符号不再用于其他用途。

最简单的可唯一译码的整数码是一元码，可以将它看做带文件结束符的码。

一元码。对整数 n 的编码方式：先发送 $n-1$ 个 0，紧接着发送一个 1。

n	$c_u(n)$
1	1
2	01
3	001
4	0001
5	00001
⋮	
45	000000000000000000000000000000000000000000001

一元码的长度为 $l_U(n)=n$。

若 n 的概率分布是 $p_U(n)=2^{-n}$，则一元码是整数的最优码。

自定界符号码

可以采用一元码对 n 的二进制编码长度进行编码，并生成一个自定界码。

C_α 码。用一元码发送 $l_b(n)$，后接 n 的无头二进制表示。

$$c_\alpha(n)=c_U[l_b(n)]c_B(n) \tag{7.1}$$

表 7.1 给出了一些整数所对应的码。加上划线将每个字符串对应的 $c_U[l_b(n)]$ 和 $c_B(n)$ 部分区分开。也可以等价地认为，$c_\alpha(n)$ 是 $(l_b(n)-1)$ 个 0 的字符串后跟 n 的标准二进制表示，$c_b(n)$。

表 7.1 C_α 码

n	$c_b(n)$	$l_b(n)$	$c_\alpha(n)$
1	1	1	$\overline{1}$
2	10	2	$\overline{01}0$
3	11	2	$\overline{01}1$
4	100	3	$\overline{00}100$
5	101	3	$\overline{00}101$
6	110	3	$\overline{00}110$
⋮			
45	101101	6	$\overline{00000}101101$

码字 $c_\alpha(n)$ 的长度为 $l_\alpha(n)=2l_b(n)-1$。

C_α 码中 n 的隐式概率分布可分离成两部分的乘积：即长度 l 的概率分布

$$P(l)=2^{-l} \tag{7.2}$$

与具有该长度的整数的均匀分布

$$P(n|l)=\begin{cases}2^{-l+1}, & l_b(n)=l\\ 0, & \text{其他}\end{cases} \tag{7.3}$$

现在，对于上述码而言，传送长度的头部和整数的标准二进制表示总是占据同样多的比特数（即给一个取一个）。若遇到很大的整数（较大的文件），就会导致所有文件的大小是未编码的原整数大小的两倍，这种表示看起来是次优的。也可以使用 C_α 而非一元码对长度 $l_b(n)$ 进行编码。

C_β 码 采用 C_α 发送长度 $l_b(n)$，紧跟着 n 的无头二进制表示。如表 7.2 所示。

$$c_\beta(n)=c_\alpha[l_b(n)]c_B(n) \tag{7.4}$$

重复此过程，可以定义一列码。

C_γ 码 $c_\gamma(n)=c_\beta[l_b(n)]c_B(n)$ (7.5)

C_δ 码

$$c_\delta(n) = c_\gamma[l_b(n)]c_B(n) \tag{7.6}$$

表 7.2 C_β 和 C_γ

n	$c_\beta(n)$	$c_\gamma(n)$
1	$\overline{1}$	$\overline{1}$
2	$\overline{0100}$	$\overline{01000}$
3	$\overline{0101}$	$\overline{01001}$
4	$\overline{01100}$	$\overline{010100}$
5	$\overline{01101}$	$\overline{010101}$
6	$\overline{01110}$	$\overline{010110}$
⋮		
45	$\overline{0011001101}$	$\overline{0111001101}$

具有文件结束符号的码

也可以采用基于字节的表示。(这里灵活地使用术语“字节”来表示任意固定长度的比特串,而不一定是长度为 8 的比特串。)若对某种基(如十进制数)上的数进行编码,则可以用字节代表每一位数字。为了用一个字节表示从 0 到 9 的数字,需要 4 个比特。因为 $2^4 = 16$,多出来 6 个 4 比特符号,{1010,1011,1100,1101,1110,1111},没有十进制数与之对应。可以利用这些来作为文件结束标志,以表示正整数的结束。

很显然,不止一个文件结束符号的做法是冗余的,因此一种更高效的码是将整数编码成十五进制数,第 16 个符号 1111 用做标点符号。将这种思想进行推广,可以类似地得出其他基于字节的整数码,比如基于三进制和七进制,甚至基于形式 $2^n - 1$ 的任意进制整数码。如表 7.3 所示。

表 7.3 两种带文件结束符的码 C_3 和 C_7。其中包含了空格,以表明字节的边界

n	$C_3(n)$	$C_7(n)$
1	01 11	001 111
2	10 11	010 111
3	01 00 11	011 111
⋮		
45	01 10 00 00 11	110 011 111

这些码几乎是完备的。(回忆一下,如果满足 Kraft 不等式中的等号,则这种码是“完备”的)。这些码具有对整数零和空字符串进行编码的能力,而二者都不是必

需的,这是它们仍然存在的一点低效性。

▷ **习题 7.1**[⌀] 考虑具有文件结束符的码所对应整数的隐式概率分布。

(a) 若码块为 8 比特(即整数编码基于 255 进制),在隐式分布下,整数的平均长度是多少比特?

(b) 若希望用具有文件结束符的码对期望大小约为 100 KB 的二进制文件进行编码,则最优的分块大小是多少?

对一个微型文件进行编码

为了说明已讨论过的码,用它们依次对如下这个仅有 14 个字符的小文件进行编码。

Claude Shannon

- 若将 ASCII 符号映射成 7 比特符号(如,用十进制数表示,C = 67,l = 108,等等),这个 14 字符文件对应于整数

$$n = 167\ 987\ 786\ 364\ 950\ 891\ 085\ 602\ 469\ 870(\text{十进制数})$$

- n 的一元码包括这么多 0(至少一个)再跟一个 1。如果所有海洋都变成墨水,且每立方毫米墨水可以写 100 比特,才可能有足够的墨水将 $c_{\mathrm{U}}(n)$ 写出来。

- n 的标准二进制表示是长度为 98 的比特序列

$$c_{\mathrm{b}}(n) = 10000111101100110000111101011100100110010101000001010011110100011000011101110110111011011111101110$$

▷ **习题 7.2**[⌀] 写出或描述上述数 n 的如下自定界表示法:$c_\alpha(n)$,$c_\beta(n)$,$c_\gamma(n)$,$c_\delta(n)$,$c_3(n)$,$c_7(n)$ 以及 $c_{15}(n)$。其中哪个编码最短?[答案:c_{15}。]

对这些码进行比较

在回答问题"两个码中哪个更好?"时,人们可以说的话形如:"对于 $n > k$ 的情况,码 1 更好些,对于 $n < k$,码 2 更好些"。但是,我认为这样的回答忽略了一个要点:任何一个完备码对应于一个先验概率分布,在此分布下它是最优的;所以,不应该说其他码比该码更好。其他码在其他先验概率下也可以是最优码。应该考虑到这些隐式先验分布,以便得到某个应用的最优码。

注意:不能无代价地从一种码切换到另一种码,选择较短码。如果要这样做,则必须以某种方法加长消息,以指示当前正在使用的是哪一种码。若用单一前导比特来实现该功能,所得的码是次优码,因为它不满足 Kraft 等式(如习题 5.33 所讨论)。

另一种比较整数码的方法是考虑一系列概率分布,如 $n \geq 1$ 上的单调概率分布,并根据它们对其中任意分布进行编码的好坏情况,对码进行排序。如果对给定类中的任意分布,平均编码长度处于理想平均长度的某个因子范围之内,则称这种码为"通用"码。

再换种说法说一次。不像上面提倡的那样找出整数的一个好的先验分布，而是采用另一种世界观：许多理论家研究了构造码的问题，即所构造的码对一大类先验分布中的任何一个都是很好的码。这里所考虑的先验分布通常是指：(a) 对整数分配单调递减概率的分布；(b) 具有有限熵的分布。

在上面讨论的码中，有好几种都是通用码。另一种优美地超越自定界码序列的码是 Elias 的“整数通用码”(Elias，1975 年)，它从所有码($C_\alpha, C_\beta, \cdots$)中有效地进行选择。它的工作过程如下：发送一系列消息，而每个消息还对下一个消息的长度进行编码，并用单个比特指示该消息是否是最后一个整数(采用标准二进制表示法)。因为长度是一个正整数，而所有正整数都以“1”开始，因此所有起始位上的 1 都可被省略。

```
写“0”
循环{
    若⌊log n⌋=0 终止
    预先考虑 c_b(n)给要写的字符串
    n:=⌊log n⌋
    }
```

算法 7.4　整数 n 的 Elias 编码器

如算法 7.4 所示为 C_ω 编码器。编码从右到左地进行。表 7.5 给出了所得码字。

表 7.5　Elias 的整数“通用”码。从 1 到 1 025 的例子。

n	$c_\omega(n)$	n	$c_\omega(n)$	n	$c_\omega(n)$	n	$c_\omega(n)$
1	0	9	1110010	31	10100111110	256	1110001000000000
2	100	10	1110100	32	101011000000	365	1110001011011010
3	110	11	1110110	45	101011011010	511	1110001111111110
4	101000	12	1111000	63	101011111110	512	11100110000000000
5	101010	13	1111010	64	1011010000000	719	11100110110011110
6	101100	14	1111100	127	1011011111110	1 023	11100111111111110
7	101110	15	1111110	128	10111100000000	1 024	111010100000000000
8	1110000	16	10100100000	255	10111111111110	1 025	111010100000000010

▷ **习题 7.3**[ℰ]　证明对一个预计每个整数都很大的先验分布，Elias 码实际上并不是最好的码。(构造另一种码，指定该码要求 n 必须有多大，可以给出一个比 Elias 码更短的码。)

习题解答

习题 7.1 的解答　在表示基于 q 进制整数的码中，使用文件结束标识，对应于这

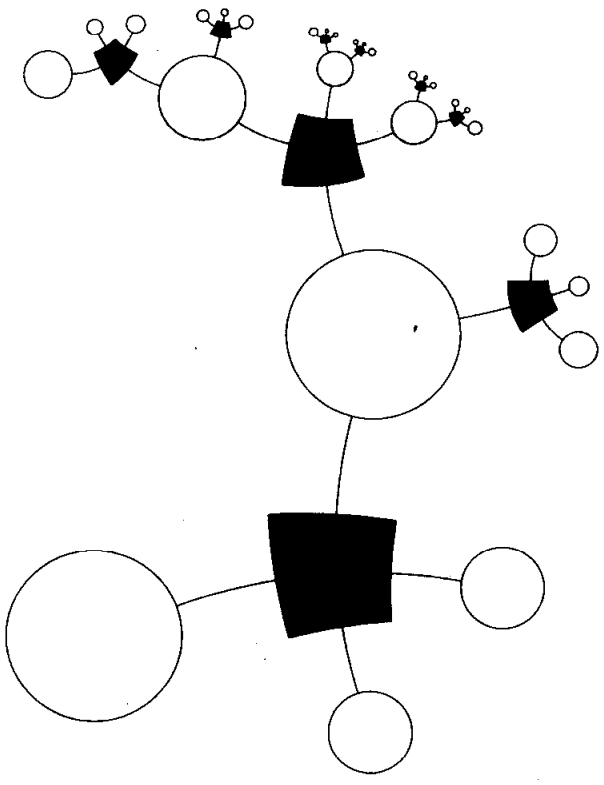
样一种信念:当前字符是数中最后一个字符的概率为$(1/(q+1))$。因此,与这个码相匹配的先验分布是为整数长度所分配的指数先验分布。

(a) 所期望的字符数是$q+1=256$,因此整数的期望长度为$256\times 8\simeq 2\ 000$比特。

(b) 希望找到满足$q\log q\simeq 800\ 000$比特的q值。介于2^{15}和2^{16}之间的一个q值满足这种约束,因此假定有一个文件结束字符,16-比特块大概具有最优大小。

第二部分

噪声信道编码



第 8 章

相关随机变量

在前面 3 章讨论数据压缩时，集中讨论了随机向量 $\boldsymbol{x}$ 服从一个极其简单的概率分布，即每个分量 x_n 都是彼此独立的可分离分布。

在这一章中，将考虑相关随机变量的联合总体。考虑这些内容有两个目的。首先，现实世界中的数据具有一些有趣的相关性，所以为了更好地进行数据压缩，需要知道如何处理包含相关性的模型。其次，在一个输入为 x 和输出为 y 的噪声信道定义了一个联合总体，其中 x 和 y 是相关的——如果它们是独立的，则在该信道上不可能进行通信——所以噪声信道上的通信（第 9 章 ~ 第 11 章的主题）是用联合总体的熵来描述的。

8.1 关于熵的更多内容

这一节继续第 2.4 节的讨论，给出与熵有关的一些定义和习题。

X、Y **的联合熵**是

$$H(X,Y) = \sum_{xy \in \mathcal{A}_X \mathcal{A}_Y} P(x,y) \log \frac{1}{P(x,y)} \tag{8.1}$$

独立随机变量的熵是可加的：

$$H(X,Y) = H(X) + H(Y) \Leftrightarrow P(x,y) = P(x)P(y) \tag{8.2}$$

已知 $y = b_k$ **时** X **的条件熵**是概率分布 $P(x \mid y = b_k)$ 的熵。

$$H(X \mid y = b_k) \equiv \sum_{x \in \mathcal{A}_X} P(x \mid y = b_k) \log \frac{1}{P(x \mid y = b_k)} \tag{8.3}$$

已知 Y **时** X **的条件熵**是已知 y 时 X 的条件熵在 y 上的平均值。

$$\begin{aligned} H(X \mid Y) &\equiv \sum_{y \in \mathcal{A}_Y} P(y) \left[\sum_{x \in \mathcal{A}_X} P(x \mid y) \log \frac{1}{P(x \mid y)} \right] \\ &= \sum_{xy \in \mathcal{A}_X \mathcal{A}_Y} P(x,y) \log \frac{1}{P(x \mid y)} \end{aligned} \tag{8.4}$$

这度量了在 y 为已知时关于 x 仍存在的平均不确定性。

X 的**边缘熵**是 X 的熵 $H(X)$ 的另一种叫法，是相对于上述条件熵而言的。

信息量的链式法则 由概率的乘法法则，即式(2.6)，得到

$$\log \frac{1}{P(x,y)} = \log \frac{1}{P(x)} + \log \frac{1}{P(y \mid x)} \tag{8.5}$$

所以

$$h(x,y)=h(x)+h(y|x) \tag{8.6}$$

也就是说，x 和 y 的信息量是 x 的信息量加上已知 x 前提下 y 的信息量。

熵的链式法则 联合熵、条件熵和边缘熵之间的关系是：

$$H(X,Y)=H(X)+H(Y|X)=H(Y)+H(X|Y) \tag{8.7}$$

即，X 和 Y 的不确定性是 X 的不确定性加上已知 X 前提下 Y 的不确定性。

X 和 Y 之间的互信息是

$$I(X;Y)\equiv H(X)-H(X|Y) \tag{8.8}$$

并满足 $I(X;Y)=I(Y;X)$ 和 $I(X;Y)\geqslant 0$。它所度量的是得知 y 值后 x 不确定性的平均减小幅度；或与之相对，x 所传递的关于 y 的平均信息量。

已知 $z=c_k$ **时 X 和 Y 之间的条件互信息**是在联合总体 $P(x,y|z=c_k)$ 中随机变量 X 和 Y 的互信息

$$I(X;Y|z=c_k)=H(X|z=c_k)-H(X|Y,z=c_k) \tag{8.9}$$

已知 Z **时 X 和 Y 之间的条件互信息**是上述条件互信息在 z 上的平均值

$$I(X;Y|Z)=H(X|Z)-H(X|Y,Z) \tag{8.10}$$

不会再定义其他的“三项熵”。例如，诸如 $I(X;Y;Z)$ 和 $I(X|Y;Z)$ 之类的表达式是非法的。但是你可以在表达式 $I(X;Y|Z)$ 的 3 个部分中放置任意数量的变量联合——例如，$I(A,B;C,D|E,F)$ 是很好的：它所度量的是在假设 e 和 f 已知时，c 和 d 平均传递了多少关于 a 和 b 的信息。

图 8.1 显示了一个联合总体的总熵 $H(X,Y)$ 是如何被分解的。**这个图很重要。**

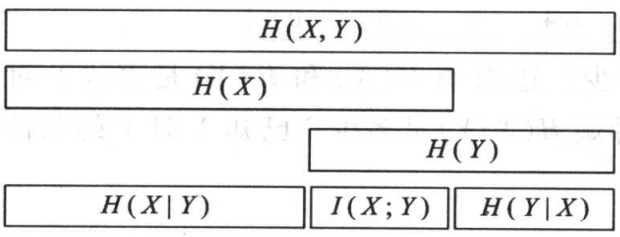


图 8.1 联合信息、边缘熵、条件熵和互信息熵之间的关系

8.2 习 题

▷ **习题 8.1**[1] 考虑 3 个独立随机变量 u、v、w，它们的熵分别为 H_u、H_v、H_w。设 $X\equiv(U,V)$ 和 $Y\equiv(V,W)$。试问 $H(X,Y)$ 是多少？$H(X|Y)$ 是多少？$I(X;Y)$ 是多少？

▷ **习题 8.2**[2] 参考条件熵的定义式(8.3～8.4)，举例证明 $H(X|y=b_k)$ 可能会大于 $H(X)$，但是平均而言，$H(X|Y)$ 要小于 $H(X)$。所以数据还是有帮助的——在平均意义下，它们不会增加不确定性。

▷ **习题 8.3**[2] 证明熵的链式法则(8.7)[$H(X,Y)=H(X)+H(Y|X)$]。



习题 8.4[M] 证明互信息 $I(X;Y) \equiv H(X) - H(X|Y)$ 满足 $I(X;Y) = I(Y;X)$ 且 $I(X;Y) \geq 0$。

[提示:参见习题 2.26 并注意到

$$I(X;Y) = D_{\mathrm{KL}}(P(x,y) \parallel P(x)P(y))] \tag{8.11}$$

习题 8.5[M] 两个随机变量之间的"熵距离"可被定义为它们的联合熵和它们的互信息之间的差值

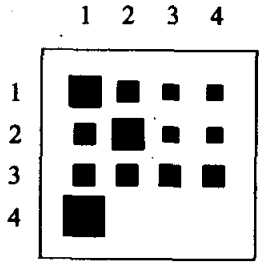
$$D_H(X,Y) \equiv H(X,Y) - I(X;Y) \tag{8.12}$$

证明熵距离满足距离公理——$D_H(X,Y) \geq 0, D_H(X,X) = 0, D_H(X,Y) = D_H(Y,X)$ 和 $D_H(X,Z) \leq D_H(X,Y) + D_H(Y,Z)$。[顺便提一下,不太可能再次看到$D_H(X,Y)$,但它是一个很好的函数,可用来证明不等式。]



习题 8.6[M] 联合总体 XY 具有如下联合分布。

$P(x,y)$		x			
		1	2	3	4
	1	1/8	1/16	1/32	1/32
y	2	1/16	1/8	1/32	1/32
	3	1/16	1/16	1/16	1/16
	4	1/4	0	0	0



联合熵 $H(X,Y)$ 是多少? 边缘熵 $H(X)$ 和 $H(Y)$ 是多少? 对于每个 y 值,条件熵 $H(X|y)$ 是多少? 条件熵 $H(X|Y)$ 是多少? 已知 X 时 Y 的条件熵是多少? X 和 Y 之间的互信息是多少?



习题 8.7[M] 考虑总体 XYZ,其中 $\mathcal{A}_X = \mathcal{A}_Y = \mathcal{A}_Z = \{0,1\}$。$x$ 和 y 是独立的,$\mathcal{P}_X = \{p, 1-p\}$,$\mathcal{P}_Y = \{q, 1-q\}$,且

$$z = (x + y) \bmod 2 \tag{8.13}$$

(a) 如果 $q = 1/2$,$\mathcal{P}_Z$ 是多少? $I(Z;X)$ 是多少?

(b) 对于一般的 p 和 q,$\mathcal{P}_Z$ 是多少? $I(Z;X)$ 是多少? 注意这个总体与二进制对称信道有关,x 为输入,y 为噪声,z 为输出。

三项熵

习题 8.8[M] 很多教材都将图 8.1 画成文氏图的形式(图 8.2)。讨论为什么这个图是熵的一个误导性表示。提示:考虑具有 3 个变量的总体 XYZ,其中 $x \in \{0,1\}$ 和 $y \in \{0,1\}$ 是独立的二进制变量,而 $z \in \{0,1\}$ 定义为 $z = (x + y) \bmod 2$。

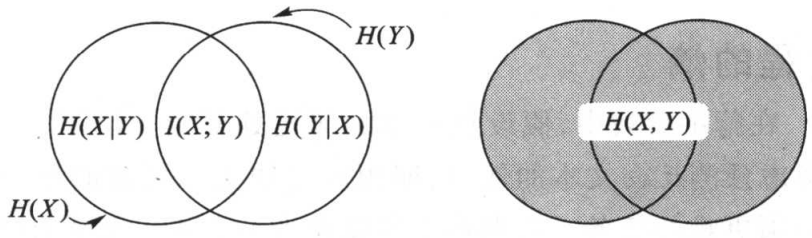


图 8.2 熵的一个误导性表示(与图 8.1 对照)

8.3 进一步的习题

数据处理定理

数据处理定理表明:数据处理只会破坏信息。



习题 8.9[♪] 用下述方法证明该定理。考虑总体 WDR,其中 w 是现实世界的状态,d 是所采集到的数据,r 是处理后的数据,使得这 3 个变量组成一个马尔科夫链

$$w \rightarrow d \rightarrow r \tag{8.14}$$

即概率 $P(w,d,r)$ 可以写成

$$P(w,d,r) = P(w)P(d|w)P(r|d) \tag{8.15}$$

证明 R 所传递的关于 W 的平均信息 $I(W;R)$ 少于或等于由 D 所传递的关于 W 的平均信息 $I(W;D)$。

这个定理对于定义“信息”和进行数据处理都是一个警示。

干扰和信息度量



习题 8.10[♪] 3 张卡片问题。

(a) 一张卡片的两面都是白色;一张卡片的两面都是黑色;一张卡片的一面是白色,另一面是黑色。对这 3 张卡片进行洗牌,且假设它们的方向是任意的。抽出一张卡片放在桌上。朝上的一面是黑色。问朝下的一面是什么颜色?(解决干扰问题。)

(b) 看到正面的颜色是否传递了关于反面颜色的信息?讨论这种情况下的信息量和熵。将上面颜色的值记为 u,反面颜色的值记为 l。假设随机抽到一张卡片并知道 u 和 l 的值。u 的熵 $H(U)$ 是多少?l 的熵 $H(L)$ 是多少?U 和 L 之间的互信息 $I(U;L)$ 是多少?

马尔科夫过程的熵

▷ **习题 8.11**[𝒫] 在猜字游戏中，假设从文本的起始处开始预测下一个字母，直到文本的结尾处。考虑预测反向文本的任务，即预测已知文本之前的那个字母。大多数人都认为这是一项更难的任务。假设将语言建模为一个 $N-$元模型（也就是说，下一个字符出现的概率仅依赖于其前的 $N-1$ 个字符）。反向语言和正向语言的平均信息量之间有什么差别吗？

8.4 习题解答

习题 8.2 的解答 关于 $H(X|y)$ 大于 $H(X)$ 的一个例子（设 $y=3$），请参见习题 8.6。

可以将表达式转换为相对熵的形式（使用贝叶斯定理）并用 Gibbs 不等式（习题 2.26）来证明不等式 $H(X|Y)\leqslant H(X)$：

$$\begin{aligned} H(X\mid Y) &\equiv \sum_{y\in\mathcal{A}_Y} P(y)\Big[\sum_{x\in\mathcal{A}_X} P(x\mid y)\log\frac{1}{P(x\mid y)}\Big] \\ &= \sum_{xy\in\mathcal{A}_X\mathcal{A}_Y} P(x,y)\log\frac{1}{P(x\mid y)} \qquad (8.16) \\ &= \sum_{xy} P(x)P(y\mid x)\log\frac{P(y)}{P(y\mid x)P(x)} \qquad (8.17) \\ &= \sum_{x} P(x)\log\frac{1}{P(x)} + \sum_{x} P(x)\sum_{y} P(y\mid x)\log\frac{P(y)}{P(y\mid x)} \qquad (8.18) \end{aligned}$$

最后一个表达式是分布 $P(y|x)$ 和 $P(y)$ 的相对熵之和。因此

$$H(X|Y)\leqslant H(X)+0 \qquad (8.19)$$

只有当对所有 x 和 y，$P(y|x)=P(y)$ 时（即，仅当 X 和 Y 独立时），等式成立。

习题 8.3 的解答 从一个联合概率的分解，得到熵的链式法则为

$$\begin{aligned} H(X,Y) &= \sum_{xy} P(x,y)\log\frac{1}{P(x,y)} \qquad (8.20) \\ &= \sum_{xy} P(x)P(y\mid x)\Big[\log\frac{1}{P(x)}+\log\frac{1}{P(y\mid x)}\Big] \qquad (8.21) \\ &= \sum_{x} P(x)\log\frac{1}{P(x)} + \sum_{x} P(x)\sum_{y} P(y\mid x)\log\frac{1}{P(y\mid x)} \qquad (8.22) \\ &= H(X)+H(Y\mid X) \qquad (8.23) \end{aligned}$$

习题 8.4 的解答 互信息的对称性：

$$I(X;Y) = H(X)-H(X\mid Y) \qquad (8.24)$$

$$= \sum_{x} P(x)\log\frac{1}{P(x)} - \sum_{xy} P(x,y)\log\frac{1}{P(x\mid y)} \tag{8.25}$$

$$= \sum_{xy} P(x,y)\log\frac{P(x\mid y)}{P(x)} \tag{8.26}$$

$$= \sum_{xy} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \tag{8.27}$$

这个表达式关于 x 和 y 是对称的，所以

$$I(X;Y) = H(X) - H(X\mid Y) = H(Y) - H(Y\mid X) \tag{8.28}$$

可以用两种方法来证明互信息是正的。一是从下式出发：

$$I(X;Y) = \sum_{x,y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \tag{8.29}$$

它是一个相对熵，使用了 Gibbs 不等式（第 2 章证明过），可以证明这个相对熵非负，且仅当 $P(x,y) = P(x)P(y)$（即 X 和 Y 相互独立）时等式才成立。

另一种方法是对下式使用 Jensen 不等式。

$$-\sum_{x,y} P(x,y)\log\frac{P(x)P(y)}{P(x,y)} \geqslant -\log\sum_{x,y}\frac{P(x,y)}{P(x,y)}P(x)P(y) = \log 1 = 0 \tag{8.30}$$

习题 8.7 的解答 $z = x + y \bmod 2$。

（a）如果 $q = 1/2$，$\mathscr{P}_Z = \{1/2, 1/2\}$，$I(Z;X) = H(Z) - H(Z\mid X) = 1 - 1 = 0$。

（b）对于一般的 p 和 q，$\mathscr{P}_Z = \{pq + (1-p)(1-q), p(1-q) + q(1-p)\}$。互信息为 $I(Z;X) = H(Z) - H(Z\mid X) = H_2(pq + (1-p)(1-q)) - H_2(q)$。

三项熵

习题 8.8 的解答 用文氏图来描述熵容易引起误解，至少有两个原因。

第一，人们习惯性地认为文氏图是描述集合的；但图 8.2 所描述的“集合”$H(X)$ 和 $H(Y)$ 是什么？这些集合的元素又是哪些对象？笔者认为这个图会误导初学者进行不恰当的类推。比如，一些学生会认为随机输出 (x,y) 也许与图中的某一点相对应，因此将熵同概率相混淆。

第二，根据文氏图的描述方法容易误导人们相信，所有的区域都对应于正数。在两个随机变量的特殊情况，$H(X\mid Y)$，$I(X;Y)$ 和 $H(Y\mid X)$ 的确是正数。但当处理 3 个变量的总体时，所得到的图中看起来为正的区域实际上也许对应的是负数。图 8.3 正确地表明了如下关系

$$H(X) + H(Z\mid X) + H(Y\mid X,Z) = H(X,Y,Z) \tag{8.31}$$

但是它给人以错误印象，即条件互信息 $I(X;Y\mid Z)$ 比互信息 $I(X;Y)$ 要小。实际上，图中标为 A 的区域可对应于一个负值。考虑联合总体 (X,Y,Z)，其中 $x \in \{0,1\}$ 和 $y \in \{0,1\}$ 是独立的二进制变量，$z \in \{0,1\}$ 被定义为 $z = x + y \bmod 2$。于是，显然有 $H(X) = H(Y) = 1$ 比特。同样 $H(Z) = 1$ 比特。由于两个变量是独立的，所以 $H(Y\mid X) =$

$H(Y)=1$。所以 X 和 Y 之间的互信息为零，$I(X;Y)=0$。但是，如果观测到 z，X 和 Y 就会变成相关的——给定 z 的情况下。已知 x，就能知道 y 是多少：$y=z-x \bmod 2$。所以，$I(X;Y|Z)=1$ 比特。于是，正确答案是图中标为 A 的区域一定对应于 -1 比特。

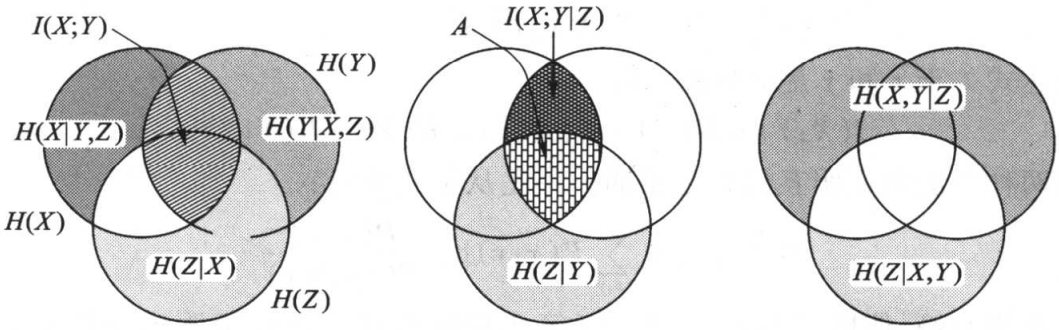


图 8.3 熵的误导性表示(续)

上面的例子完全不是一个古怪或例外的示例。输入为 X、噪声为 Y、输出为 Z 的二进制对称信道就是如此：$I(X;Y)=0$（输入和噪声是独立的），但是 $I(X;Y|Z)>0$（一旦得知了输出，则未知的输入和未知的噪声就密切关联起来了）。

只有当你意识到正区域可能代表负值时，文氏图表示法才是有效的。只有紧记附带条件，从集合的角度来解释熵才是有帮助的（Yeung，1991 年）。

习题 8.9 的解答 对于任意联合总体 XYZ，如下互信息的链式法则成立。

$$I(X;Y,Z)=I(X;Y)+I(X;Z|Y) \tag{8.32}$$

现在，在 $w\rightarrow d\rightarrow r$ 的情形下，已知 d 时 w 和 r 是独立的，所以 $I(W;R|D)=0$。两次使用链式法则，有

$$I(W;D,R)=I(W;D) \tag{8.33}$$

和

$$I(W;D,R)=I(W;R)+I(W;D|R) \tag{8.34}$$

所以

$$I(W;R)-I(W;D)\leqslant 0 \tag{8.35}$$

关于第9章

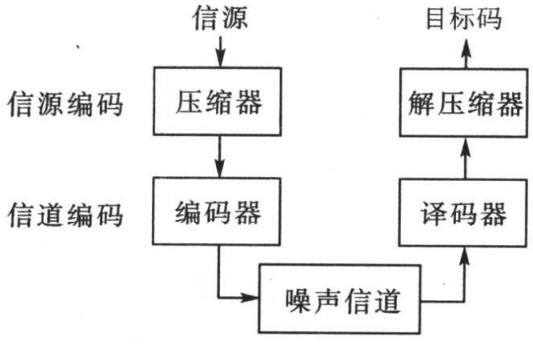
在阅读第9章之前，你应该读过第1章并完成了习题2.26和习题8.2～习题8.7。

第 9 章

噪声信道上的通信

9.1 全 景 图

在第 4 章 ~ 第 6 章中，讨论了信源编码中的分组码、符号码以及符号流码。我们隐含地假设从压缩器到解压缩器之间的信道是无噪声的。实际信道是有噪声的。接下来将用两章内容来讨论噪声信道编码这一主题——在噪声信道上进行无差错通信的基本可行性和限制条件。信道编码的目的就是使噪声信道变得像一个无噪信道那样。我们将假设被传输的数据已经通过了一个很好的压缩器，所以比特流中不包含明显的冗余。所设计的信道编码，将会重新加入一些特殊的冗余，使有噪声的接收信号可以得到正确的译码，从而实现传输。



假设在噪声信道上每秒传输 1 000 比特，$p_0 = p_1 = \frac{1}{2}$，比特翻转概率为 $f = 0.1$。信息的传输速率是多少？大家也许会猜测减去每秒预期的错误数就可以得到每秒 900 比特的速率。但这是不对的，因为接收端并不知道哪里发生了错误。考虑这样一种情形，噪声太大，以至于接收符号与发送符号彼此独立。因为一半的接收符号由于运气的原因是正确的，所以这对应于 $f = 0.5$ 的噪声级。但是当 $f = 0.5$ 时，根本就没有传输任何信息。

由已经学到的关于熵的知识，一种看上去合理的传输信息度量是：信源和接收信号之间的互信息，即信源的熵减去给定接收信号时信源的条件熵。

现在将复习条件熵和互信息的定义。然后将考察是否可能使用这样的噪声信道来进行可靠的通信。将证明对于任意信道 Q，存在一个非零速率，即容量 $C(Q)$，使得不大于该速率的信息可以在任意小的差错概率下传输。

9.2 概率和信息的复习

以习题8.6中的联合分布 XY 为例进行讨论。边缘分布 $P(x)$ 和 $P(y)$ 在表格的边缘列出。

$P(x,y)$		x				$P(y)$
		1	2	3	4	
y	1	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{32}$	$\frac{1}{4}$
	2	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{32}$	$\frac{1}{32}$	$\frac{1}{4}$
	3	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{4}$
	4	$\frac{1}{4}$	0	0	0	$\frac{1}{4}$
$P(x)$		$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$	

联合熵 $H(X,Y)=27/8$ 比特。边缘熵 $H(X)=7/4$ 比特和 $H(Y)=2$ 比特。

对于每个 y 值，可以计算 x 的条件分布以及每个条件分布的熵。

$P(x\mid y)$		x				$H(X\mid y)$/比特
		1	2	3	4	
y	1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{7}{4}$
	2	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{7}{4}$
	3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	2
	4	1	0	0	0	0

$$H(X\mid Y)=\frac{11}{8}$$

注意到虽然 $H(X\mid y=4)=0$ 小于 $H(X)$，但 $H(X\mid y=3)$ 却比 $H(X)$ 大。所以在有些场合中，已知 y 反倒会增加关于 x 的不确定性。同时还要注意到，尽管 $P(x\mid y=2)$ 与 $P(x)$ 是不同的分布，但条件熵 $H(X\mid y=2)$ 却与 $H(X)$ 相等。所以知道 y 为2可以改变我们对 x 的认识，但是并不能减小 x 的不确定性（用熵来度量）。尽管平均而言，已知 y 确实会传递有关 x 的一些信息，因为 $H(X\mid Y)<H(X)$。

也许还可以计算出 $H(Y\mid X)=13/8$ 比特。互信息为 $I(X;Y)=H(X)-H(X\mid Y)=3/8$ 比特。

9.3 噪声信道

离散无记忆信道 Q 由一个输入符号集 $\mathcal{A}_X$、一个输出符号集 $\mathcal{A}_Y$ 和一组对每一

个 $x \in \mathscr{A}_X$ 的条件概率分布 $P(y|x)$ 来描述。

这些转移概率可以写成矩阵形式

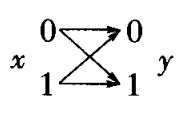
$$Q_{j|i} = P(y = b_j | x = a_i) \tag{9.1}$$

我通常用输出变量 j 作为矩阵的行号，用输入变量 i 作为矩阵的列号，所以 $\boldsymbol{Q}$ 的每一列是一个概率向量。使用这种约定，可通过右乘一个输入的概率分布 $\boldsymbol{p}_X$，得到一个输出的概率 $\boldsymbol{p}_Y$：

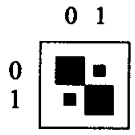
$$\boldsymbol{p}_Y = \boldsymbol{Q}\boldsymbol{p}_X \tag{9.2}$$

一些有用的信道模型包括以下几种。

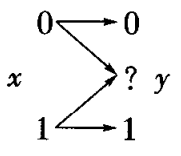
二进制对称信道　$\mathscr{A}_X = \{0,1\}$。$\mathscr{A}_Y = \{0,1\}$。



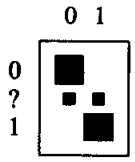
$$P(y=0|x=0) = 1-f; P(y=0|x=1) = f;$$
$$P(y=1|x=0) = f; \quad P(y=1|x=1) = 1-f$$



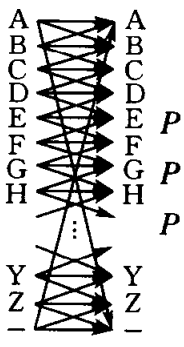
二进制删除信道　$\mathscr{A}_X = \{0,1\}$。$\mathscr{A}_Y = \{0,?,1\}$。



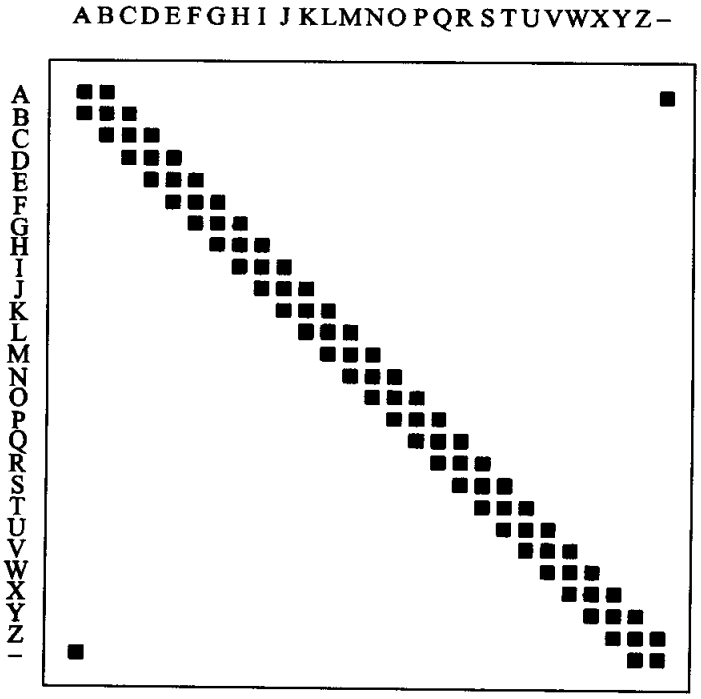
$$P(y=0|x=0) = 1-f; \quad P(y=0|x=1) = 0;$$
$$P(y=?|x=0) = f; \quad P(y=?|x=1) = f;$$
$$P(y=1|x=0) = 0; \quad P(y=1|x=1) = 1-f$$



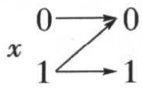
有噪声的打字机　$\mathscr{A}_X = \mathscr{A}_Y = \{A,B,\cdots,Z,-\}$ 是27个字母所组成的集合。字母被排列成一个圆圈，当打字员输入B时，打出来的可能是A、B或C，且每个均以1/3的概率出现；而输入为C时，输出为B、C或D；依此类推，最后一个字母“-”与第一个字母A相邻。



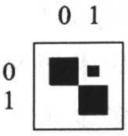
$$\vdots$$
$$P(y=F|x=G) = 1/3;$$
$$P(y=G|x=G) = 1/3;$$
$$P(y=H|x=G) = 1/3;$$
$$\vdots$$



Z信道　$\mathscr{A}_X = \{0,1\}$。$\mathscr{A}_Y = \{0,1\}$。



$$P(y=0|x=0)=1;P(y=0|x=1)=f;$$
$$P(y=1|x=0)=0;P(y=1|x=1)=1-f$$



9.4 已知输出时推断输入

如果假设从总体 X 中选择一条信道的输入 x，则得到一个联合总体 XY，其中随机变量 x 和 y 服从联合分布：

$$P(x,y)=P(y|x)P(x) \tag{9.3}$$

现在如果接收到一个特定符号 y，那么输入符号 x 是什么？通常，我们不会很肯定地知道。可以用贝叶斯定理写出输入的后验分布：

$$P(x\mid y)=\frac{P(y\mid x)P(x)}{P(y)}=\frac{P(y\mid x)P(x)}{\sum_{x'}P(y\mid x')P(x')} \tag{9.4}$$

例 9.1 考虑一个差错概率为 $f=0.15$ 的二进制对称信道。输入总体为 $\mathscr{P}_X:\{p_0=0.9,p_1=0.1\}$。假设观察到 $y=1$。

$$\begin{aligned}P(x=1\mid y=1)&=\frac{P(y=1\mid x=1)P(x=1)}{\sum_{x'}P(y\mid x')P(x')}\\&=\frac{0.85\times0.1}{0.85\times0.1+0.15\times0.9}\\&=\frac{0.085}{0.22}=0.39\end{aligned} \tag{9.5}$$

因此，得出“$x=1$”的可能性还是要比“$x=0$”小，尽管可能性不像原来那样小。



习题 9.2[↗] 现在假设观察到 $y=0$。计算已知 $y=0$ 时 $x=1$ 的概率。

例 9.3 考虑一个差错概率 $f=0.15$ 的 Z 信道。输入总体为 $\mathscr{P}_x:\{p_0=0.9,p_1=0.1\}$。假设观察到 $y=1$。

$$P(x=1|y=1)=\frac{0.85\times0.1}{0.85\times0.1+0\times0.9}=\frac{0.085}{0.085}=1.0 \tag{9.6}$$

所以在给定输出 $y=1$ 时就可以确定输入。



习题 9.4[↗] 再假设观察到 $y=0$。计算 $P(x=1|y=0)$。

9.5 一个信道所传递的信息

现在考虑一个信道可以传递多少信息。从实用的角度出发，想要找到使用信道

的这样一些方式，使得所有通信比特能够以一个可忽略的差错概率得以恢复。用数学术语来表达就是，假设一个特定的输入总体 X，可通过互信息来度量输出中传递了多少关于输入的信息量：

$$I(X;Y) \equiv H(X) - H(X|Y) = H(Y) - H(Y|X) \tag{9.7}$$

我们的目的是建立这两种思想之间的联系。让我们来计算上面一些信道的 $I(X;Y)$。

关于计算互信息的提示

我们倾向于将 $I(X;Y)$ 当做 $H(X) - H(X|Y)$，即当看到输出 Y 时关于输入 X 的不确定性被减少了多少。但是从计算的角度来看，求解 $H(Y) - H(Y|X)$ 的值往往更容易。如图 9.1 所示。

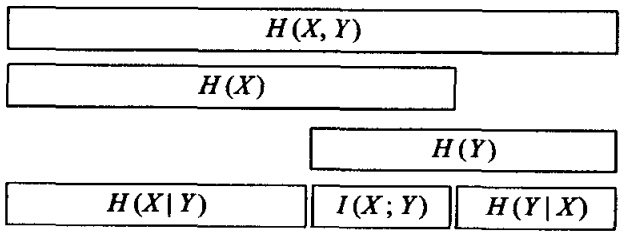


图 9.1 联合信息、边缘熵、条件熵和互信息熵之间的关系。这个图很重要，所以在此再次给出。

例 9.5 再考虑二进制对称信道，其中 $f = 0.15$ 和 $\mathscr{P}_X: \{p_0 = 0.9, p_1 = 0.1\}$。已经在上面隐含地计算了边缘概率 $P(y): P(y=0) = 0.78; P(y=1) = 0.22$。互信息为

$$I(X;Y) = H(Y) - H(Y|X)$$

$H(Y|X)$ 是多少？它被定义为 $H(Y|x)$ 对所有 x 的加权和；但是对于每个 x 值，$H(Y|x)$ 都是相等的：$H(Y|x=0)$ 是 $H_2(0.15)$，$H(Y|x=1)$ 是 $H_2(0.15)$。所以

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y|X) \\ &= H_2(0.22) - H_2(0.15) \\ &= 0.76 - 0.61 = 0.15 \text{ 比特} \end{aligned} \tag{9.8}$$

这也许可以与信源的熵 $H(X) = H_2(0.1) = 0.47$ 比特进行比较。

注意：这里使用了二进制熵函数 $H_2(p) \equiv H(p, 1-p) = p\log\frac{1}{p} + (1-p)\log\left(\frac{1}{1-p}\right)$。[①]

例 9.6 现在考虑 Z 信道，$\mathscr{P}_X$ 同上。$P(y=1) = 0.085$。

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y|X) \\ &= H_2(0.085) - [0.9H_2(0) + 0.1H_2(0.15)] \\ &= 0.42 - (0.1 \times 0.61) = 0.36 \text{ 比特} \end{aligned} \tag{9.9}$$

① 在全书中，log 表示 $\log_2$。

和前面一样，信源的熵是 $H(X)=0.47$ 比特。注意到 Z 信道的互信息 $I(X;Y)$ 在 f 相同的情况下比二进制对称信道的互信息要大。Z 信道是一个更可靠的信道。



习题 9.7[1] 当输入分布为 $\mathcal{P}_X=\{p_0=0.5,p_1=0.5\}$ 时，计算 $f=0.15$ 二进制对称信道的 X 和 Y 之间的互信息。



习题 9.8[2] 当输入分布为 $\mathcal{P}_X:\{p_0=0.5,p_1=0.5\}$ 时，计算 $f=0.15$ 的 Z 信道的 X 和 Y 之间的互信息。

互信息最大化

在上面的例子中，注意到输入与输出之间的互信息取决于所选择的输入总体。

假设希望通过选择最好的可能输入总体，来对信道所传递的互信息进行最大化。定义信道的容量为其最大互信息。

信道 Q 的**容量**是

$$C(Q)=\max_{\mathcal{P}_X} I(X;Y) \tag{9.10}$$

取得最大值的分布 $\mathcal{P}_X$ 称作**最优输入分布**，记做 $\mathcal{P}_X^*$。[也许会有多个最优输入分布，它们达到同样的 $I(X;Y)$ 值。]

在第 10 章，将证明容量确实度量了单位时间内信道可传送的最大无差错信息量。

例 9.9 考虑 $f=0.15$ 的二进制对称信道。上面考虑了 $\mathcal{P}_X=\{p_0=0.9,p_1=0.1\}$，并得到 $I(X;Y)=0.15$ 比特。还可以改善多少？由对称性，最优的输入分布为 $\{0.5,0.5\}$，且容量为

$$C(Q_{\text{BSC}})=H_2(0.5)-H_2(0.15)=1.0-0.61=0.39 \text{ 比特} \tag{9.11}$$

在后面将证明上面关于对称性的讨论是正确的。如果对于对称性的讨论有任何疑虑，总可以采用显式最大化互信息 $I(X;Y)$ 的方法

$$I(X;Y)=H_2((1-f)p_1+(1-p_1)f)-H_2(f),\quad \text{如图 9.2 所示} \tag{9.12}$$

例 9.10 有噪声的打字机。最优输入分布是 x 的一个均匀分布，由此可得 $C=\log_2 9$ 比特。

例 9.11 考虑 $f=0.15$ 的 Z 信道。确定最优输入分布并不容易。对 $\mathcal{P}_X=\{p_0,p_1\}$ 直接求 $I(X;Y)$ 值。首先，需要计算 $P(y)$。$y=1$ 的概率很容易写出

$$P(y=1)=p_1(1-f) \tag{9.13}$$

于是，互信息为

$$\begin{aligned} I(X;Y)&=H(Y)-H(Y|X)\\ &=H_2(p_1(1-f))-(p_0H_2(0)+p_1H_2(f))\\ &=H_2(p_1(1-f))-p_1H_2(f) \end{aligned} \tag{9.14}$$

这是 p_1 的一个非平凡函数，如图 9.3 所示。当 $f=0.15$ 时，在 $p_1^*=0.445$ 处取得最大

值。求得 $C(Q_Z)=0.685$。注意到最优输入分布不是$\{0.5,0.5\}$。如果让输入符号0的使用频度高于输入符号1,就可以稍微提高通信的信息量。

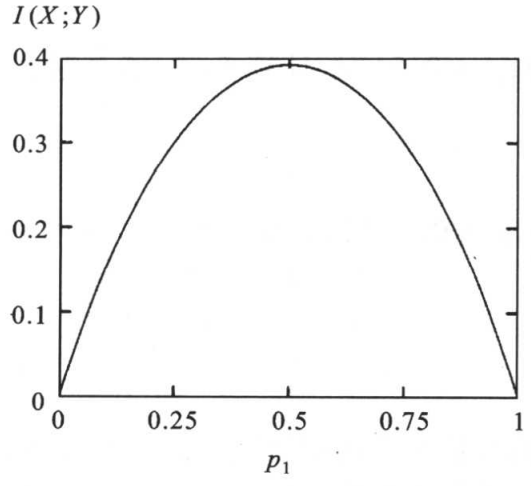


图9.2 $f=0.15$的二进制对称信道的互信息$I(X;Y)$,它是输入分布的函数。

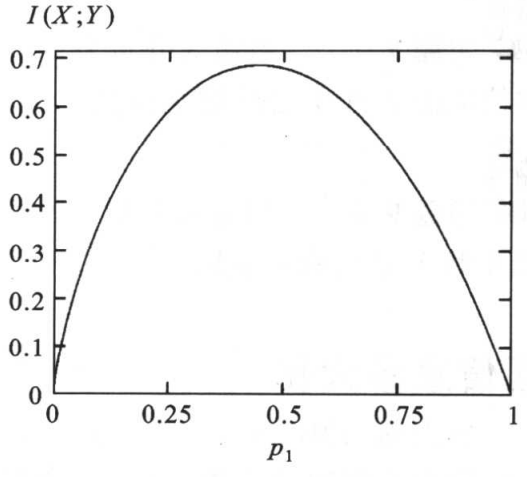


图9.3 $f=0.15$的Z信道的互信息$I(X;Y)$,它是输入分布的函数。



习题9.12[1] f取一般值时,二进制对称信道的容量是多少?



习题9.13[2] 证明当$f=0.15$时二进制删除信道的容量为$C_{\mathrm{BEC}}=0.85$。f取一般值时的容量是多少?对所得结果进行讨论。

9.6 噪声信道编码定理

所定义的"容量"似乎可以作为信道所传递信息的一种度量;有一点不是显而易见的(同时也是下一章将要证明的):容量确实可以度量*以任意小的差错概率*在信道上通信的数据块速率。

做如下定义。

信道Q的(N,K)**分组码**是一列个数为$S=2^K$的码字

$$\{\boldsymbol{x}^{(1)},\boldsymbol{x}^{(2)},\cdots,\boldsymbol{x}^{(2^K)}\},\boldsymbol{x}^{(s)}\in\mathcal{A}_X^N$$

每个码字的长度为N。使用这个码可将信号$s\in\{1,2,3,\cdots,2^K\}$编码为$\boldsymbol{x}^{(s)}$。[码字S的数量是一个整数,而选择一个码字所规定的比特数$K\equiv\log_2 S$不一定是一个整数。]

码*率*为每次信道使用$R=K/N$比特。

[将对任意信道使用码率的这个定义,而不仅针对有二进制输入的信道;然而,注意,有时将有q个输入符号的信道的码率定义为$K/(N\log q)$更为方便。]

一个(N,K)分组码的**译码器**是将信道输出的一组长度为N的串$\mathcal{A}_Y^N$映射为一

个码字标号 $\hat{s} \in \{0,1,2,\cdots,2^K\}$。额外的符号 $\hat{s}=0$ 可用来表示一个“错误”。

对于一个已知信道，和一个编码信号 $P(s_{in})$ 上的已知概率分布，一个编码及其译码器的分组差错概率为

$$p_B = \sum_{s_{in}} P(s_{in})P(s_{out} \neq s_{in} \mid s_{in}) \tag{9.15}$$

最大分组差错概率为

$$p_{BM} = \max_{s_{in}} P(s_{out} \neq s_{in} \mid s_{in}) \tag{9.16}$$

一个信道编码的**最优译码器**是一个能够对分组差错概率进行最小化的译码器。它将输出 y 译码为具有如下最大后验概率 $P(s|y)$ 的输入 s。

$$P(s \mid y) = \frac{P(y \mid s)P(s)}{\sum_{s'} P(y \mid s')P(s')} \tag{9.17}$$

$$\hat{s}_{\text{最优}} = \operatorname{argmax} P(s \mid y) \tag{9.18}$$

通常假设 s 的先验分布是均匀的，在这种情况下，最优译码器同时也是最大似然译码器，即译码器将输出 y 译码为具有最大似然 $P(y|s)$ 的输入 s。

比特差错概率 p_b 的定义：假设用长度为 K 比特的二进制向量 s 来表示码字数 s；它是 s_{out} 中有一个比特与 s_{in} 的对应比特不相等的平均概率（对所有 K 比特求平均值）。

Shannon 噪声信道编码定理（第一部分） 对于每个离散无记忆信道，存在一个具有如下性质的非负数 C（称为信道容量）。对于任意 $\varepsilon > 0$ 和 $R < C$，当 N 足够大时，存在一个长度为 N 且码率 $\geqslant R$ 的分组码以及一个译码算法，使得最大分组差错概率 $< \varepsilon$。如图 9.4 所示。

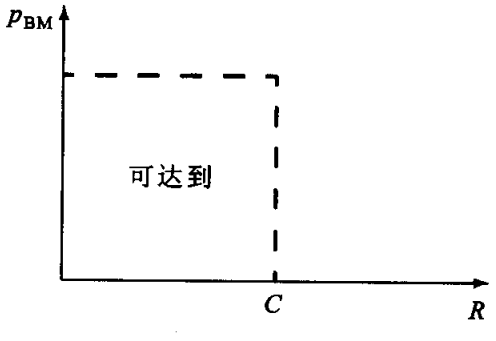


图 9.4 由 Shannon 噪声信道编码定理第一部分可达到的 R 和 p_{BM} 平面区域

定理验证（针对有噪声打字机信道）

对于有噪声打字机的情形，可以很容易地验证该定理，因为可使用长度 $N=1$ 的分组码来产生一个完全无误的通信策略：只使用字母 B，E，H，…，Z，即每隔 3 个字母取一个。这些字母构成输入符号表的一个不易混淆子集（如图 9.5 所示）。任何输出都可唯一译码。在不易混淆子集中所输入的个数是 9，所以这个系统的无误信息

速率是 $\log_2 9$ 比特，它等于容量 C，在例 9.10 中曾计算过它的值。

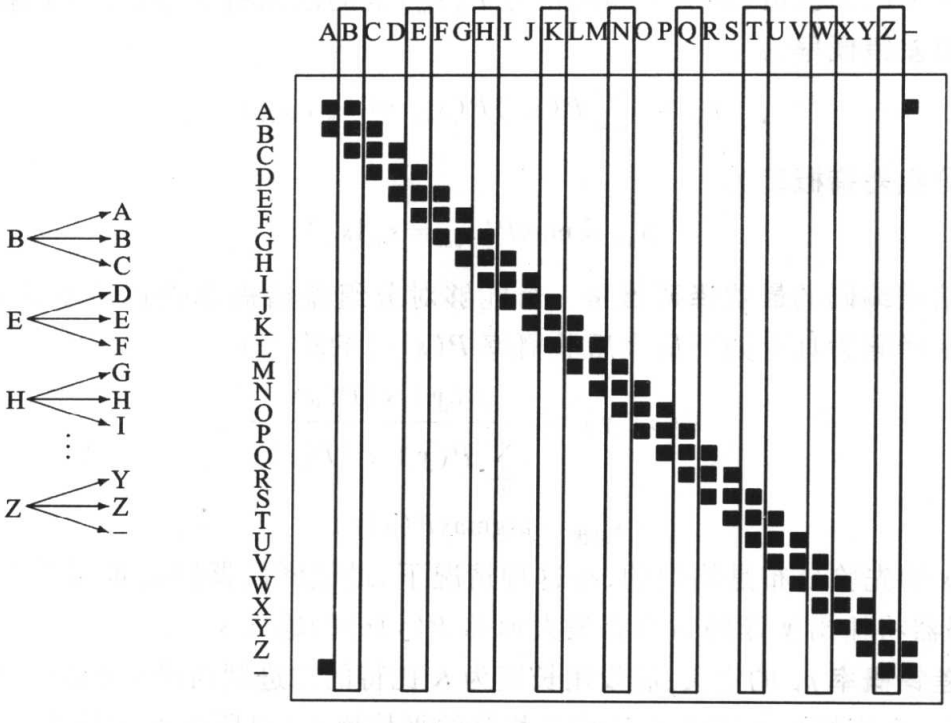


图 9.5　有噪声打字机的一个不易混淆的输入子集

如何将以上阐述转换为定理的形式？下列表格进行了解释。

定理	它如何应用于有噪声打字机
对应于每个离散的无记忆信道，存在一个非负数 C。	容量 C 为 $\log_2 9$。
当 N 足够大，对于任意 $\varepsilon>0$ 和 $R<C$，存在一个长度为 N 的分组码，其码率 $\geq R$。	不管 ε 和 R 是多少，将分组长度 N 设为 1。分组码为 $\{B,E,\cdots,Z\}$。K 的值由 $2^K=9$ 给出，所以 $K=\log_2 9$，这个码的码率为 $\log_2 9$，比要求的 R 值要大。
和一个译码算法，	译码算法将所接收到的字母映射为码中距离最近的字母。
这样最大的分组差错概率 $<\varepsilon$。	最大分组差错概率为零，小于已知的 ε 值。

9.7　证明的直接预观

扩展信道

要在任意给定信道条件下证明该定理，考虑 N 次信道使用所对应的扩展信道。扩展信道有 $|\mathscr{A}_X|^N$ 个可能的输入 $\boldsymbol{x}$ 和 $|\mathscr{A}_Y|^N$ 个可能的输出。当 $N=2$ 和 $N=4$ 时，

如图 9.6 和图 9.7 所示是由一个二进制对称信道和一个 Z 信道所得扩展信道。

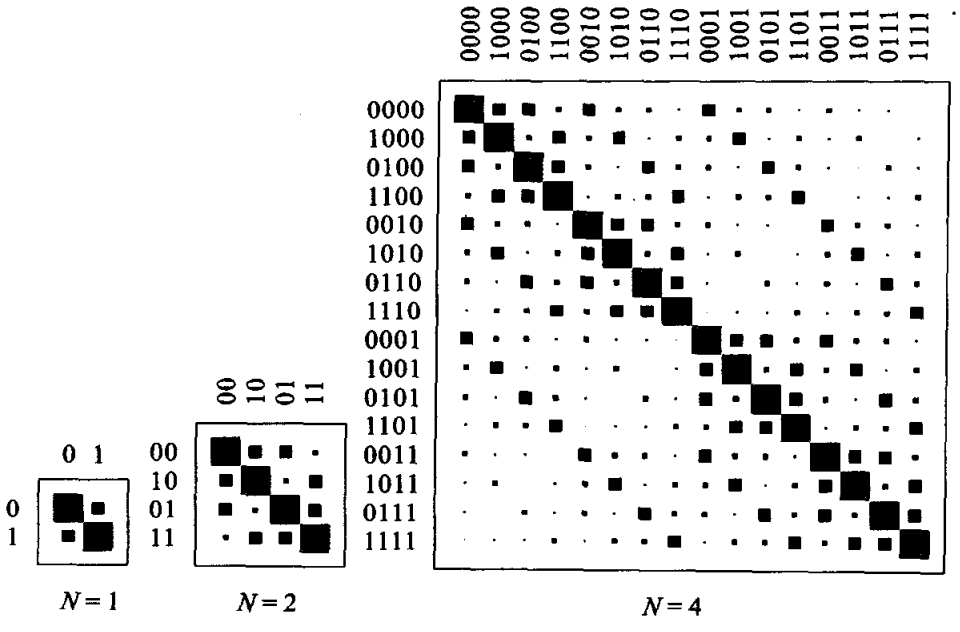


图 9.6 从一个转移概率为 0.15 的二进制对称信道所得的扩展信道

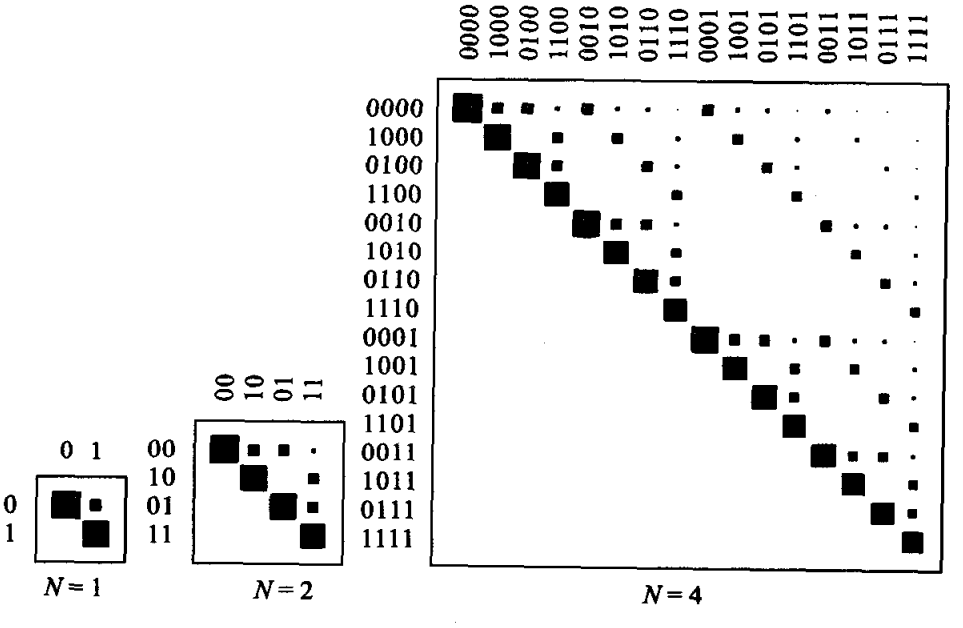


图 9.7 由一个转移概率为 0.15 的 Z 信道获得的扩展信道。每一列对应一个输入，而每一行对应一个不同的输出。



习题 9.14[*] 当 $N=2$ 时，求解由删除概率为 0.15 的二进制删除信道所推导出的扩展信道的转移概率矩阵 $\boldsymbol{Q}$。

通过选择转移概率矩阵的两列，可以为此信道定义一个码长为 $N=2$、码率为1/2

的码。这两列的最佳选择是什么？其译码算法是什么？

为了证明噪声信道的编码定理，使用大的分组长度 N。一种直观的想法是，如果 N 很大，一个扩展信道看起来非常像一个有噪声打字机。任何特殊输入 $\boldsymbol{x}$ 所产生的输出很可能属于输出字符表的一个小子空间里——已知输入的典型输出集合。所以可找到输入的一个不易混淆的子集来产生基本上不相交的输出序列。对于一个已知的 N，考虑生成输入的这种不易混淆子集的一种方法，并计算它包含了多少个不同的输入。

假设从总体 X^N 中抽取扩展信道的一个输入序列 $\boldsymbol{x}$，其中 X 是输入符号集的一个任意总体。回忆第4章中的信源编码定理，并考虑可能输出序列 $\boldsymbol{y}$ 的数目。典型输出序列 $\boldsymbol{y}$ 的总数为 $2^{NH(Y)}$，它们都具有相似的概率。对于任意特定的典型输入序列 $\boldsymbol{x}$，存在大约 $2^{NH(Y|X)}$ 个可能的序列。$\mathscr{A}_Y^N$ 中的部分子集如图 9.8(a)中的圆圈所示。

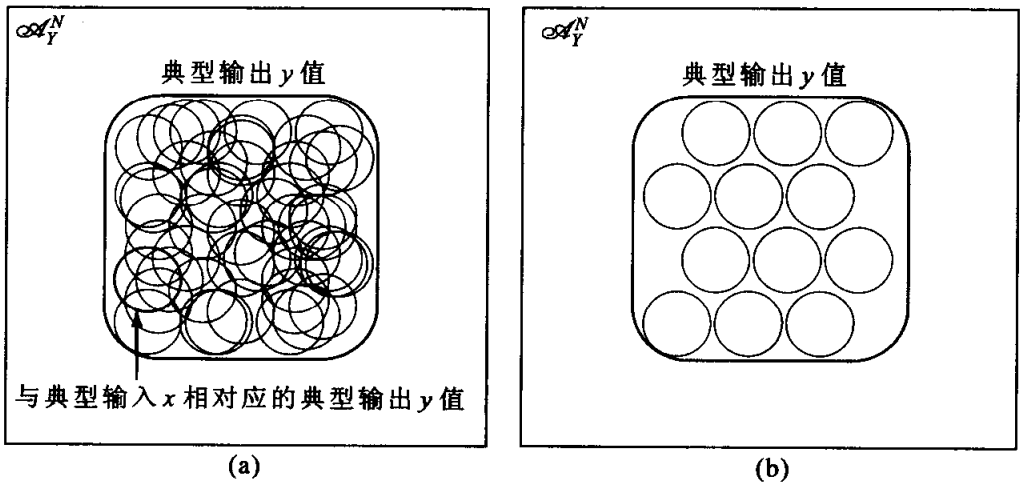


图 9.8 (a) $\mathscr{A}_Y^N$ 中对应于典型输入 $\boldsymbol{x}$ 的一些典型输出。(b)图(a)所示典型集的一个互不相交的子集。可以将这个图与图 9.5 所示有噪声打字机的解答做比较。

现在假设将自己限制在典型输入 $\boldsymbol{x}$ 的一个子集中，使得与之对应的典型输出集合彼此不相交，如图 9.8(b)所示。将典型集 $\boldsymbol{y}$ 的大小 $2^{NH(Y)}$ 除以每个已知 $\boldsymbol{x}$ 后典型 $\boldsymbol{y}$ 的大小 $2^{NH(Y|X)}$，就可以限制不易混淆输入的数量。如果不易引起混淆的输入是从典型输入 $\boldsymbol{x} \sim X^N$ 中选择的，则其个数 $\leqslant 2^{NH(Y)-NH(Y|X)} = 2^{NI(X;Y)}$。

如果 X 是使 $I(X;Y)$ 最大化的总体，那么就可得到这个界限的最大值，在这种情况下，不易混淆的输入的个数 $\leqslant 2^{NC}$。因此，渐近地最多可以用每周期 C 比特的速率，进行零差错概率通信。

我们并未严格证明这个概括性的结论：确实可以进行可靠的通信——那是下一章的任务。

9.8 进一步的习题



习题 9.15[p] 参考 $f=0.15$ 时 Z 信道的容量计算。

(a) 为什么 p_1^* 小于 0.5？有人认为它的输入为 0 好，因为它可以进行无差错的传输。有人认为输入为 1 好，因为它通常给出高回报的输出 1，这将对输入进行确凿的判决！试给出一个令人信服的结论。

(b) 在 f 取一般值的情形，证明最优输入分布是

$$p_1^* = \frac{1/(1-f)}{1+2^{(H_2(f)/(1-f))}} \tag{9.19}$$

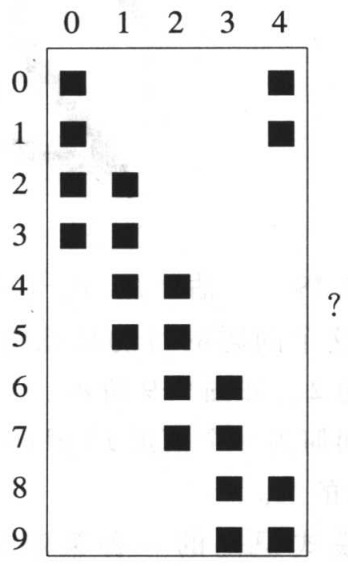
(c) 如果噪声水平 f 非常接近 1，p_1^* 将会怎样？



习题 9.16[p] 绘制作为 f 函数的 Z 信道、二进制对称信道和二进制删除信道的容量的草图。

▷ **习题 9.17**[p] 一个 5 - 输入、10 - 输出的信道具有如下转移概率矩阵。它的容量是多少？

$$\begin{bmatrix} 0.25 & 0 & 0 & 0 & 0.25 \\ 0.25 & 0 & 0 & 0 & 0.25 \\ 0.25 & 0.25 & 0 & 0 & 0 \\ 0.25 & 0.25 & 0 & 0 & 0 \\ 0 & 0.25 & 0.25 & 0 & 0 \\ 0 & 0.25 & 0.25 & 0 & 0 \\ 0 & 0 & 0.25 & 0.25 & 0 \\ 0 & 0 & 0.25 & 0.25 & 0 \\ 0 & 0 & 0 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0.25 & 0.25 \end{bmatrix} \tag{9.20}$$



?



习题 9.18[p] 考虑一个具有二进制输入 $x \in \{-1, +1\}$ 和实输出符号集 $\mathcal{A}_Y$ 的高斯信道，其转移概率密度为

$$Q(y|x,\alpha,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x\alpha)^2}{2\sigma^2}} \tag{9.21}$$

其中 α 为信号的幅度。

(a) 假设两个输入是等概率的,计算已知 y 时 x 的后验概率。将你的答案表达为如下形式

$$P(x=1|y,\alpha,\sigma)=\frac{1}{1+e^{-a(y)}} \tag{9.22}$$

并以 y 的函数绘制 $P(x=1|y,\alpha,\sigma)$ 值的草图。

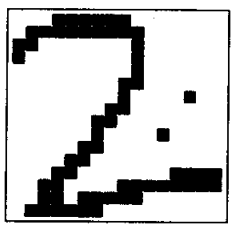
(b) 假设传输了单个比特。最优译码器是什么? 它的差错概率是多少? 用信噪比 α^2/σ^2 和误差函数(高斯分布的累积概率函数)来表示你的答案。

$$\Phi(z)\equiv\int_{-\infty}^{z}\frac{1}{\sqrt{2\pi}}e^{-\frac{z^2}{2}}dz \tag{9.23}$$

[注意误差函数 $\Phi(z)$ 的这个定义可能与其他人的定义不同。]

将模式识别作为噪声信道

可以将很多模式识别问题以通信信道的形式来考虑。考虑识别手写数字的情形(诸如信封上的邮政编码)。数字的作者希望传递集合 $\mathcal{A}_X=\{0,1,2,3,\cdots,9\}$ 中一条消息;这个被选择的消息是信道的输入。信道的输出是一个写在纸上的墨水图案。如果这个墨水图案用 256 个二进制像素来表示,则信道 Q 的输出是一个随机变量 $y\in\mathcal{A}_Y=\{0,1\}^{256}$。这一组符号的一个元素的例子如下图所示。





习题 9.19[2] 估计在 $\mathcal{A}_Y$ 中有多少模式可被识别为字符"2"。[这个问题的目的是试图说明有尽可能多的模式可被识别为 2。]如图 9.9 所示。

讨论如何为一个信道 $P(y|x=2)$ 建模。估计概率分布 $P(y|x=2)$ 的熵。

进行模式识别的一种策略是:为输入 $x=\{0,1,2,3,\cdots,9\}$ 的每个值生成 $P(y|x)$ 的一个模型,然后用贝叶斯定理推断已知 y 时的 x。

$$P(x|y)=\frac{P(y|x)P(x)}{\sum_{x'}P(y|x')P(x')} \tag{9.24}$$

这个策略被称为*全概率建模或生成式建模*。目前的语音识别系统基本上就是这样工作的。除了信道模型 $P(y|x)$ 之外,还可以使用一个先验概率分布 $P(x)$,在字符识别和语

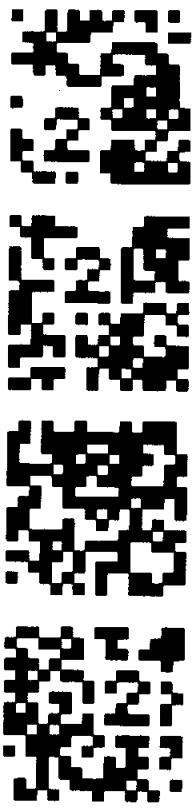


图 9.9 更多的图案"2"

音识别这两种情况下，先验概率分布是一个语言模型，在已知上下文及语法和该语言统计规律的条件下，它给出了下一个字符/单词的概率。

随机编码



习题 9.20[2] 在一个房间里有 24 个人，至少有两个人有相同生日（即同月同日）的概率是多少？有相同生日的人的对数的期望是多少？这两个问题中的哪一个更容易解决？哪一个答案更能说明问题的实质？你也许会觉得使用诸如 A = 一年中的天数 = 365 和 S = 人数 = 24 之类的记号会有助于解决这些问题。

▷ **习题 9.21**[2] 生日的问题可以与编码方案联系起来。假设希望给外人传递一个信息，来识别 24 个人中的一个。如果事先约定从人到数字的一个映射，可以简单地传递 $\mathcal{A}_S = \{1,2,\cdots,24\}$ 中的一个数字 s；或者可以传递 $\mathcal{A}_X = \{1,2,\cdots,365\}$ 中的一个数字，来识别被选中者生日在一年中的这一天（向闰年出生的人道歉）。[假设接收者知道所有人的生日。]假设将这个通信方案用于单次传输，它的差错概率大概是多少？通信信道的容量是多少？方案所能达到的通信速率又是多少？

▷ **习题 9.22**[2] 现在假设一幢楼中有 K 个房间，每个房间中有 q 个人。（你可以将 $K = 2$ 和 $q = 24$ 视为一个例子。）目的是通过发送 K 天（从 $\mathcal{A}_X$ 中选出）的有序序列，来传递从每个房间选一个人的选择信息。比较下面两种方案的差错概率。

（a）和原来一样，每个房间发送被选中者的生日。

（b）对于从每个房间选择一人所组成的每个 K - 人组，赋予从 $\mathcal{A}_X$ 中随机选择的日期所构成的一个有序 K - 元组（这个 K - 元组与他们的生日无关）。接收者已知这列巨大的符号串（共 $S = q^K$ 个）。当大楼里的每个房间已选择一个特定的人，就发送对应于 K - 人组的有序日期字符串。

当 $q = 364$ 和 $K = 1$ 时，差错概率是多少？当 $q = 364$ 和 K 很大时，如 $K = 6\ 000$ 时，差错概率又是多少？

9.9 习题解答

习题 9.2 的解答 假设观察到 $y = 0$，有

$$P(x = 1 \mid y = 0) = \frac{P(y = 0 \mid x = 1)P(x = 1)}{\sum_{x'} P(y \mid x')P(x')} \tag{9.25}$$

$$= \frac{0.15 \times 0.1}{0.15 \times 0.1 + 0.85 \times 0.9} \tag{9.26}$$

$$= \frac{0.015}{0.78} = 0.019 \tag{9.27}$$

习题 9.4 的解答 如果观察到 $y=0$,有

$$P(x=1|y=0)=\frac{0.15\times0.1}{0.15\times0.1+1.0\times0.9} \tag{9.28}$$

$$=\frac{0.015}{0.915}=0.016 \tag{9.29}$$

习题 9.7 的解答 $y=1$ 的概率为 0.5,所以互信息为

$$I(X;Y)=H(Y)-H(Y|X) \tag{9.30}$$

$$=H_2(0.5)-H_2(0.15) \tag{9.31}$$

$$=1-0.61=0.39 \text{ 比特} \tag{9.32}$$

习题 9.8 的解答 再用 $I(X;Y)=H(Y)-H(Y|X)$ 来计算互信息。$y=0$ 的概率为 0.575,且有

$$H(Y|X)=\sum_x P(x)H(Y|x)=P(x=1)H(Y|x=1)+P(x=0)H(Y|x=0)$$

所以互信息为

$$I(X;Y)=H(Y)-H(Y|X) \tag{9.33}$$

$$=H_2(0.575)-[0.5\times H_2(0.15)+0.5\times0] \tag{9.34}$$

$$=0.98-0.30=0.679 \text{ 比特} \tag{9.35}$$

习题 9.12 的解答 由对称性,最优输入分布为{0.5,0.5},则容量为

$$G=I(X;Y)=H(Y)-H(Y|X) \tag{9.36}$$

$$=H_2(0.5)-H_2(f) \tag{9.37}$$

$$=1-H_2(f) \tag{9.38}$$

你是否想不借助对称性就求得最优输入分布?在一般情况下,当输入总体为$\{p_0,p_1\}$时,可以通过计算如下互信息来求解。

$$I(X;Y)=H(Y)-H(Y|X) \tag{9.39}$$

$$=H_2(p_0f+p_1(1-f))-H_2(f) \tag{9.40}$$

唯一依赖于 p 的项是第一项 $H_2(p_0f+p_1(1-f))$,它在自变量为 0.5 时最大化。通过设置 $p_0=1/2$,可给出该值。

习题 9.13 的解答 **答案 1**:由对称性,最优输入分布为{0.5,0.5}。将互信息写为 $I(X;Y)=H(X)-H(X|Y)$,很容易计算出容量。条件熵 $H(X|Y)$ 是 $\sum_y P(y)H(X|y)$;当 y 为已知时,仅在 $y=?$ 时 x 是不确定的,此时的概率为 $f/2+f/2$,所以条件熵 $H(X|Y)$ 是 $fH_2(0.5)$。

$$C=I(X;Y)=H(X)-H(X|Y) \tag{9.41}$$

$$=H_2(0.5)-fH_2(0.5) \tag{9.42}$$

$$=1-f \tag{9.43}$$

二进制删除信道所损失时间的比例为 f。它的容量正好是 $1-f$,它是信道可靠所占的时间比例。这个结果看起来非常合理,但距离如何将信息进行编码以使其在此信道上进行可靠通信还相差甚远。

答案 2：或者，不使用上面所假设的对称性，可以从输入总体$\{p_0, p_1\}$开始。当$y=?$时，概率为$p_0 f + p_1 f = f$；当接收到$y=?$时，x的后验概率与先验概率是相等的，所以

$$I(X;Y) = H(X) - H(X|Y) \tag{9.44}$$

$$= H_2(p_1) - fH_2(p_1) \tag{9.45}$$

$$= (1-f)H_2(p_1) \tag{9.46}$$

当$p_1 = 1/2$时，这个互信息达到它的最大值$(1-f)$。

习题 9.14 的解答 如图 9.10 所示为扩展信道。$N=2$时这个信道的最优码可以通过选择重叠最小的两列来获得，例如，列 00 和 11。如果扩展信道的输出属于上面 4 个，则译码算法将返回“00”，如果它属于下面 4 个，则返回“11”。如果输出是“??”，则放弃。

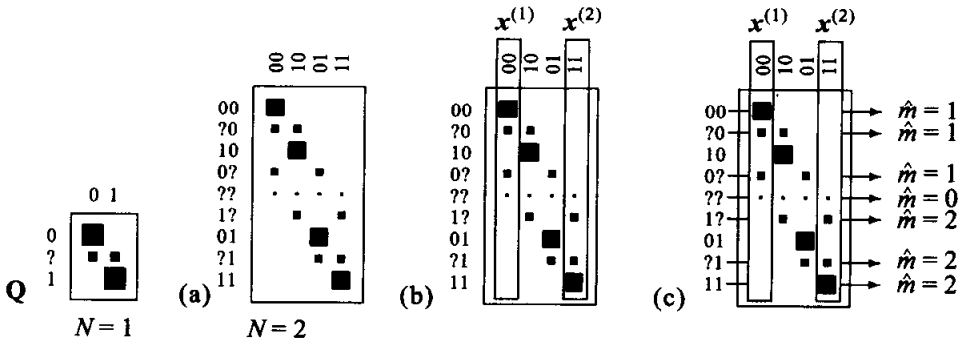


图 9.10 （a）从删除概率为 0.15 的二进制删除信道所获得的扩展信道（$N=2$）。（b）包括两个码字 00 和 11 的一个分组码。（c）这个码的最优译码器。

习题 9.15 的解答 在例 9.11 中，证明了 Z 信道的输入与输出之间的互信息为

$$I(X;Y) = H(Y) - H(Y|X)$$

$$= H_2(p_1(1-f)) - p_1 H_2(f) \tag{9.47}$$

上式对p_1求微分，注意不要将$\log_2$与$\log_e$相混淆：

$$\frac{\mathrm{d}}{\mathrm{d}p_1} I(X;Y) = (1-f)\log_2 \frac{1-p_1(1-f)}{p_1(1-f)} - H_2(f) \tag{9.48}$$

将这个导数设为零，利用习题 2.17 给出的技巧重新整理，得到

$$p_1^*(1-f) = \frac{1}{1+2^{H_2(f)/(1-f)}} \tag{9.49}$$

于是最优输入分布为

$$p_1^* = \frac{1/(1-f)}{1+2^{(H_2(f)/(1-f))}} \tag{9.50}$$

当噪声水平f趋向于 1 时，这个表达式趋向于 1/e（用罗必塔法则可以证明）。

对于f的所有值，p_1^*比 1/2 要小。为什么输入 1 比输入 0 用得要少？一个大致的直觉是当输入为 1 时，噪声信道将熵插入接收字符串中；然而当输入为 0 时，噪声

具有零熵。因此从 $p_1=1/2$ 开始，使 p_1 变小的扰动会线性地减少条件熵 $H(Y|X)$，而 $H(Y)$ 的一阶项保持不改变。$H(Y)$ 仅随 $(p_1-1/2)$ 二次地减少。

习题 9.16 的解答 三种信道的容量如图 9.11 所示。对于任意 $f<0.5$，二进制删除信道(BEC)具有最高的容量，而二进制对称信道(BSC)具有最低的容量。

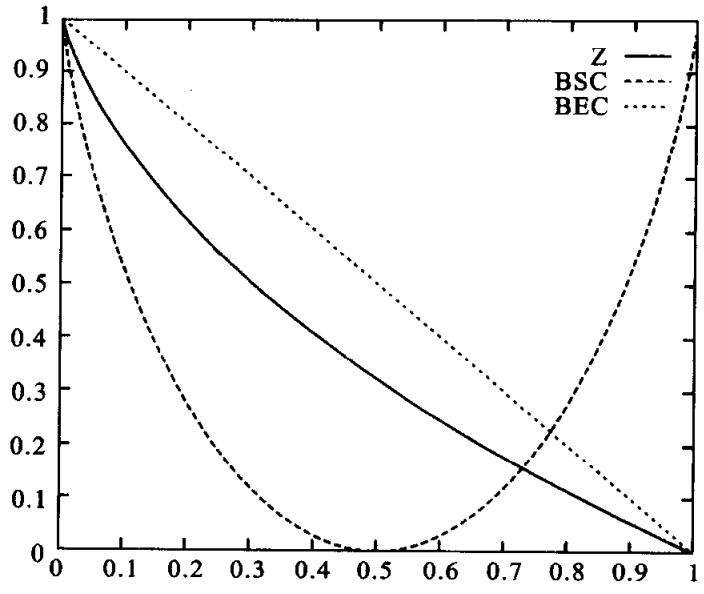


图 9.11 Z 信道、二进制对称信道和二进制删除信道的容量

习题 9.18 的解答 已知 y 时，后验概率比的对数为

$$a(y)=\ln\frac{P(x=1|y,\alpha,\sigma)}{P(x=-1|y,\alpha,\sigma)}=\ln\frac{Q(y|x=1,\alpha,\sigma)}{Q(y|x=-1,\alpha,\sigma)}=2\frac{\alpha y}{\sigma^2} \tag{9.51}$$

使用从习题 2.17 获得的技巧，将其重写为如下形式

$$P(x=1|y,\alpha,\sigma)=\frac{1}{1+\mathrm{e}^{-a(y)}} \tag{9.52}$$

最优译码器选择最有可能的假设；仅通过观察 $a(y)$ 的符号就很容易完成。即如果 $a(y)>0$，则译码为 $\hat{x}=1$。

差错概率为

$$p_{\mathrm{b}}=\int_{-\infty}^{0}\mathrm{d}yQ(y|x=1,\alpha,\sigma)=\int_{-\infty}^{-x\alpha}\mathrm{d}y\frac{1}{\sqrt{2\pi\sigma^2}}\mathrm{e}^{-\frac{y^2}{2\sigma^2}}=\Phi\left(-\frac{x\alpha}{\sigma}\right) \tag{9.53}$$

随机编码

习题 9.20 的解答 $S=24$ 个人，每个人的生日从 $A=365$ 天中随机地抽取，他们具有不同生日的概率为

$$\frac{A(A-1)(A-2)\cdots(A-S+1)}{A^q} \tag{9.54}$$

两个(或多个)人同一天生日的概率是 1 减去这个数，对于 $S=24$ 和 $A=365$，答案约

为0.5。回答这个问题的精确方法并不能提供更多信息，这是因为并不清楚什么样的 S 值会使概率从接近0的值变为接近1的值。

成对的数目是 $S(S-1)/2$，一对特定的人生日相同的概率是 $1/A$，所以巧合的期望次数为

$$\frac{S(S-1)}{2}\frac{1}{A} \tag{9.55}$$

这个答案更有意义。如果 $S \ll \sqrt{A}$，巧合的期望次数很小；如果 $S \gg \sqrt{A}$，则巧合的期望次数很大。

对于小的 S 值，还可以近似计算所有生日都不同的概率，于是有

$$\begin{aligned}\frac{A(A-1)(A-2)\cdots(A-S+1)}{A^S} &= (1)(1-1/A)(1-2/A)\cdots(1-(S-1)/A) \\ &\simeq \exp(0)\exp(-1/A)\exp(-2/A)\cdots\exp(-(S-1)/A) \qquad (9.56) \\ &\simeq \exp\left(-\frac{1}{A}\sum_{i=1}^{S-1} i\right) = \exp\left(-\frac{S(S-1)/2}{A}\right) \qquad (9.57)\end{aligned}$$

关于第 10 章

在阅读第 10 章之前，你应该已经阅读了第 4 章和第 9 章。特别推荐习题 9.14。

记号表

Q	噪声信道
C	信道容量
X^N	用来产生随机码的一个总体
$\mathscr{C}$	一个随机码
N	码字的长度
$\boldsymbol{x}^{(s)}$	一个码字，在码中是第 s 个
s	一个被选中码字的序号（为了帮助记忆：信源选择 s 表示）
$S = 2^K$	一个码中码字的总数
$K = \log_2 S$	从 S 中选择一个码字所传递的比特数，假设码字是以均匀概率选择的
s	数 s 的二进制表示
$R = K/N$	码率，以每次信道使用所传递的比特数为单位（有时也称为 R'）
$\hat{s}$	译码器关于 s 的估计

第 10 章

噪声信道编码定理

10.1 定　理

定理分为 3 个部分，两个是肯定的，一个是否定的。主要的肯定结论是第一个。

1. 对于每个离散的无记忆信道，信道容量

$$C = \max_{\mathscr{P}_X} I(X;Y) \tag{10.1}$$

具有如下性质。对于任意 $\varepsilon > 0$ 和 $R < C$，当 N 足够大时，存在一个长度为 N 的码，码率 $\geqslant R$ 和一个译码算法，使得最大分组差错概率 $< \varepsilon$。

2. 如果误码率 p_b 是可接受的，则不高于 $R(p_b)$ 的码率是可达的，其中

$$R(p_b) = \frac{C}{1 - H_2(p_b)} \tag{10.2}$$

3. 对于任意 p_b，高于 $R(p_b)$ 的码率是不可达的。如图 10.1 所示。

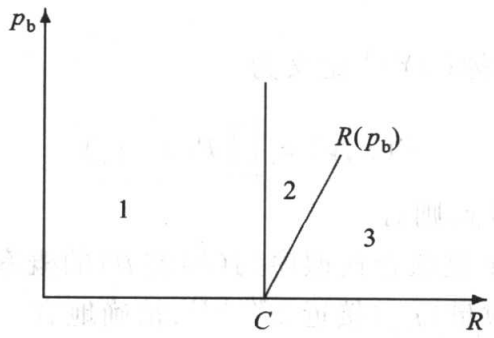


图 10.1　在 R, p_b 平面上，可证明区域(1,2)可达，而区域(3)不可达。

10.2 联合典型序列

将上一章中的直觉预证进行形式化描述。

定义码字 $\boldsymbol{x}^{(s)}$ 来自总体 X^N，并考虑随机选择一个码字和一个对应的信道输出 $\boldsymbol{y}$，由此定义一个联合总体 $(XY)^N$。如果 $\boldsymbol{x}^{(s)}$ 和 $\boldsymbol{y}$ 是联合典型的(不久将定义这个术语)，我们将使用一个典型集译码器，将一个接收到的信号 $\boldsymbol{y}$ 译码为 s。

证明将集中于确定以下事件的概率:(a) 当真实的输入码字与输出序列不是联合典型的;(b) 当一个错误的输入码字与输出是联合典型的。将要证明,对于很大的 N,在码字小于 2^{NC} 个和总体 X 是最优输入分布的条件下,这两个概率都趋于零。

联合典型性 一对长度为 N 的序列 $\boldsymbol{x}$ 和 $\boldsymbol{y}$,被定义为关于分布 $P(\boldsymbol{x},\boldsymbol{y})$ 是联合典型的(容差为 β),如果满足

$$\boldsymbol{x} \text{ 是 } P(\boldsymbol{x}) \text{ 的典型,即} \left| \frac{1}{N}\log\frac{1}{P(\boldsymbol{x})} - H(X) \right| < \beta$$

$$\boldsymbol{y} \text{ 是 } P(\boldsymbol{y}) \text{ 的典型,即} \left| \frac{1}{N}\log\frac{1}{P(\boldsymbol{y})} - H(Y) \right| < \beta$$

$$\boldsymbol{x},\boldsymbol{y} \text{ 是 } P(\boldsymbol{x},\boldsymbol{y}) \text{ 的典型,即} \left| \frac{1}{N}\log\frac{1}{P(\boldsymbol{x},\boldsymbol{y})} - H(X,Y) \right| < \beta$$

联合典型集 $J_{N\beta}$ 是长度为 N 的所有联合典型序列对的集合。

例 对于总体 $P(x,y)$,其中 $P(x)$ 有 $(p_0,p_1)=(0.9,0.1)$,且 $P(y|x)$ 对应于噪声水平为 0.2 的二进制对称信道,存在一个长度 $N=100$ 的联合典型对。

$\boldsymbol{x}$ 1111111111000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000

$\boldsymbol{y}$ 0011111111000000000000000000000000000000000000000000000000000000000000000000000000000000000000000011111111111111111

注意到 $\boldsymbol{x}$ 有 10 个 1,所以是概率 $P(\boldsymbol{x})$ 的一个典型序列(对于任意容差 β);而 y 有 26 个 1,所以它是 $P(\boldsymbol{y})$ 的一个典型序列(因为 $P(y=1)=0.26$);$\boldsymbol{x}$ 和 $\boldsymbol{y}$ 相差 20 个比特,这是该信道的典型翻转数。

联合典型性定理 总体 $(XY)^N$ 定义为

$$P(\boldsymbol{x},\boldsymbol{y}) = \prod_{n=1}^{N} P(x_n,y_n)$$

从总体 $(XY)^N$ 中选择 $\boldsymbol{x}$ 和 $\boldsymbol{y}$,则有

1. 当 $N\to\infty$ 时,$\boldsymbol{x}$ 和 $\boldsymbol{y}$ 是联合典型序列(容差 β)的概率趋向于 1。
2. 联合典型序列的数量 $|J_{N\beta}|$ 接近 $2^{NH(X,Y)}$,精确地有

$$|J_{N\beta}| \leqslant 2^{N(H(X,Y)+\beta)} \tag{10.3}$$

3. 如果 $\boldsymbol{x}' \sim X^N$ 和 $\boldsymbol{y}' \sim Y^N$,即 $\boldsymbol{x}'$ 和 $\boldsymbol{y}'$ 是与 $P(\boldsymbol{x},\boldsymbol{y})$ 具有相同边缘分布的独立样本,则 $(\boldsymbol{x}',\boldsymbol{y}')$ 属于联合典型集的概率约为 $2^{-NI(X;Y)}$。更精确的表示为

$$P((\boldsymbol{x}',\boldsymbol{y}') \in J_{N\beta}) \leqslant 2^{-N(I(X;Y)-3\beta)} \tag{10.4}$$

证明 使用大数定律对第 1 部分和第 2 部分所进行的证明类似于第 4 章中的信源编码定理。对于第 2 部分,以一对 x、y 饰演信源编码定理中 x 的角色,并用概率分布 $P(x,y)$ 代替那里的 $P(x)$。对于第三部分,

$$P((\boldsymbol{x}',\boldsymbol{y}') \in J_{N\beta}) = \sum_{(\boldsymbol{x},\boldsymbol{y}) \in J_{N\beta}} P(\boldsymbol{x})P(\boldsymbol{y}) \tag{10.5}$$

$$\leqslant |J_{N\beta}| 2^{-N(H(X)-\beta)} 2^{-N(H(Y)-\beta)} \tag{10.6}$$

$$\leqslant 2^{N(H(X,Y)+\beta)-N(H(X)+H(Y)-2\beta)} \tag{10.7}$$

$$=2^{-N(I(X;Y)-3\beta)} \qquad \square(10.8)$$

联合典型集的一个卡通图如图 10.2 所示。两个独立的典型向量是联合典型的概率为

$$P((\boldsymbol{x}',\boldsymbol{y}')\in J_{N\beta})\simeq 2^{-N(I(X;Y))} \qquad (10.9)$$

因为独立典型对的总数是虚线矩形的面积 $2^{NH(X)}2^{NH(Y)}$，且联合典型对的数量约为 $2^{NH(X,Y)}$，所以出现一个典型对的概率约为

$$2^{NH(X,Y)}/2^{NH(X)+NH(Y)}=2^{-NI(X;Y)} \qquad (10.10)$$

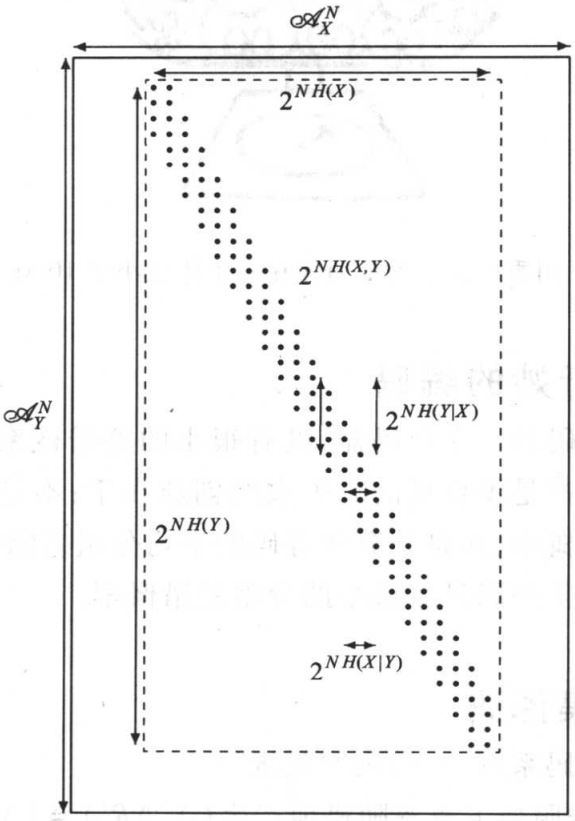


图 10.2 联合典型集。水平方向表示 $\mathscr{A}_X^N$，是所有长度为 N 的输入字符串的集合。垂直方向表示 $\mathscr{A}_Y^N$，是所有长度为 N 的输出字符串的集合。外面的盒子包含所有可能的输入—输出对。每个点代表序列 $(\boldsymbol{x},\boldsymbol{y})$ 的一个联合典型对。联合典型序列的总数约为 $2^{NH(X,Y)}$。

10.3 噪声信道编码定理的证明

类比

假如希望证明在 100 个婴儿所组成的班级里有一个婴儿的体重小于 10 kg。单个婴儿很难抓住并称重。完成这个任务的香农(Shannon)方法是召集所有孩子并用

大的称重设备为他们同时称重。如果发现他们的平均体重小于 10 kg,则至少有一个婴儿的体重小于 10 kg——而实际上肯定会有很多! 香农方法并不保证能表明存在一个体重偏轻的孩子,因为还取决于在班级里是否存在少数超重婴儿。但是如果使用他的方法并且得到总体重数小于 1 000 kg,那么任务就完成了。如图 10.3 所示。

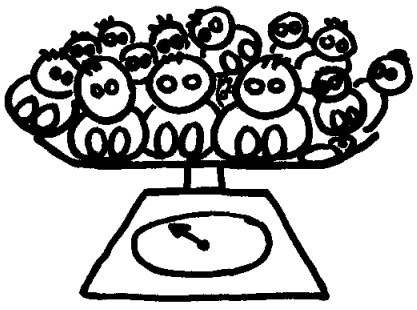


图 10.3 用香农方法来证明存在一个体重小于 10 kg 的婴儿

从瘦弱的孩子到奇妙的编码

希望证明存在一个码和一个译码器,具有很小的差错概率。计算任意特定编码和译码系统的差错概率都是不容易的。香农的创新在于:不是构造一个好的编码和译码系统并计算其差错概率,而是计算所有码的平均分组差错概率,并证明这个平均值很小。那么一定存在某些码具有很小的分组差错概率。

随机编码与典型集译码

考虑如下编码—译码系统,它的码率是 R'。

1. 固定 $P(x)$,并按照如下概率随机地产生 $(N,NR')=(N,K)$ 码 $\mathscr{C}$ 中的 $S=2^{NR'}$ 个码字。

$$P(\boldsymbol{x})=\prod_{n=1}^{N}P(x_n) \tag{10.11}$$

如图 10.4(a)所示为一个随机码字的示意图。

2. 发送方和接收方都知道该码字。

3. 从 $\{1,2,\cdots,2^{NR'}\}$ 中选择消息 s,并传输 $\boldsymbol{x}^{(s)}$。接收信号为 $\boldsymbol{y}$,满足

$$P(\boldsymbol{y}|\boldsymbol{x}^{(s)})=\prod_{n=1}^{N}P(y_n|x_n^{(s)}) \tag{10.12}$$

4. 信号由典型集译码法来译码。

典型集译码 如果 $(\boldsymbol{x}^{(\hat{s})},\boldsymbol{y})$ 是联合典型的且没有其他 s' 使得 $(\boldsymbol{x}^{(s')},\boldsymbol{y})$ 是联合典型的,则 $\boldsymbol{y}$ 被译码为 $\hat{s}$;否则认为译码失败 $(\hat{s}=0)$。

这不是最优译码算法,但它是足够好的,并且易于分析。典型集译码器如图 10.4(b)所示。

5. 如果 $\hat{s}\neq s$,则出现了一个译码错误。

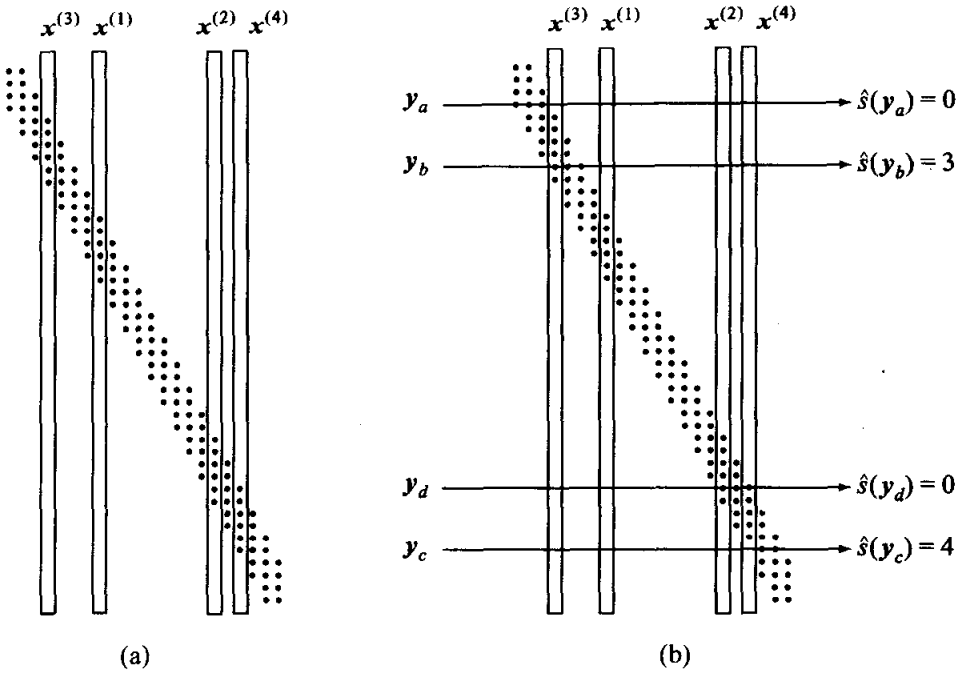


图 10.4 (a) 一个随机码。(b) 用典型集译码器进行译码的例子。诸如 y_a 之类的序列与任何码字都不是联合典型的,它被译码为 $\hat{s}=0$。只与码字 $x^{(3)}$ 的联合典型序列 y_b 被译码为 $\hat{s}=3$。同样地,y_c 被译码为 $\hat{s}=4$。若一个序列(如 y_d)与多个码字是联合典型的,则被译码为 $\hat{s}=0$。

可以区分 3 种差错概率。

第一,一个特定码 $\mathcal{C}$ 的分组差错概率,即

$$p_B(\mathcal{C}) \equiv P(\hat{s} \neq s \mid \mathcal{C}) \tag{10.13}$$

对于任意已知码而言,这都是一个很难计算的数量。

第二,这个分组差错概率对所有码取平均值

$$\langle p_B \rangle \equiv \sum_{\mathcal{C}} P(\hat{s} \neq s \mid \mathcal{C}) P(\mathcal{C}) \tag{10.14}$$

幸运的是,计算这个量要比计算第一个量 $P(\hat{s} \neq s \mid \mathcal{C})$ 容易得多。$\langle p_B \rangle$ 正是上述第 5 步出现译码错误的概率。

第三,码 $\mathcal{C}$ 的最大分组差错概率

$$p_{BM}(\mathcal{C}) \equiv \max_s P(\hat{s} \neq s \mid s, \mathcal{C}) \tag{10.15}$$

是最让人感兴趣的量:希望证明存在一个码 $\mathcal{C}$,具有所要求的码率,并且它的最大分组差错概率很小。

要获得这个结果,首先求出平均分组差错概率 $\langle p_B \rangle$。一旦证明了这可以比一个期望的小数还要小,就可立即推导出至少存在一个码 $\mathcal{C}$,它的分组差错概率也小于这个小数。最后,证明这个码的分组差错概率可以小到令人满意的程度,但是它的最大分组差错概率却是未知的(且可以想象是非常大的),可以将这个码修改为一个码率

略低的码，同时保证它的最大分组差错概率也很小。可通过将其码字中最差的50%丢弃，来修改该码。

因此现在可以着手寻找平均分组差错概率。

典型集译码器的差错概率

当使用典型集进行译码时，存在两种错误来源。它们分别是：(a) 输出 $\boldsymbol{y}$ 与被传输的码字 $\boldsymbol{x}^{(s)}$ 不是联合典型的，(b) $\mathscr{C}$ 中有其他码字与 $\boldsymbol{y}$ 是联合典型的。

由于码字构造的对称性，对所有码做平均的平均差错概率并不依赖于所选择的 s 值；不失一般性，可以假设 $s=1$。

(a) 根据联合典型定理的第一部分(10.2节)，输入 $\boldsymbol{x}^{(1)}$ 和输出 $\boldsymbol{y}$ 不是联合典型的概率趋于零。给这个概率的上限命名为 δ，它满足当 $N\to\infty$ 时 $\delta\to 0$；对于任意期望中的 δ，都可以找到一个分组长度 $N(\delta)$，使得 $P((\boldsymbol{x}^{(1)},\boldsymbol{y})\notin J_{N\beta})\leqslant\delta$。

(b) 根据第三部分，对于一个已知的 $s'\neq 1$，$\boldsymbol{x}^{(s')}$ 和 $\boldsymbol{y}$ 是联合典型的概率 $\leqslant 2^{-N(I(X;Y)-3\beta)}$。但有 $(2^{NR'}-1)$ 个候选 s' 值需要考虑。

于是，平均差错概率 $\langle p_{\mathrm{B}}\rangle$ 满足

$$\langle p_{\mathrm{B}}\rangle\leqslant\delta+\sum_{s'=2}^{2^{NR'}}2^{-N(I(X;Y)-3\beta)} \tag{10.16}$$

$$\leqslant\delta+2^{-N(I(X;Y)-R'-3\beta)} \tag{10.17}$$

不等式(10.16)通过各种足以引起差错的事件 s' 的概率 $P_{s'}$ 之和，给出总差错概率 P_{TOT} 的上界

$$P_{\mathrm{TOT}}\leqslant P_1+P_2+\cdots$$

这个上界叫做“联合界”。如果引发错误的不同事件不会在同一时间发生，则这个不等式中的等号就成立了。

如果下述不等式成立

$$R'<I(X;Y)-3\beta \tag{10.18}$$

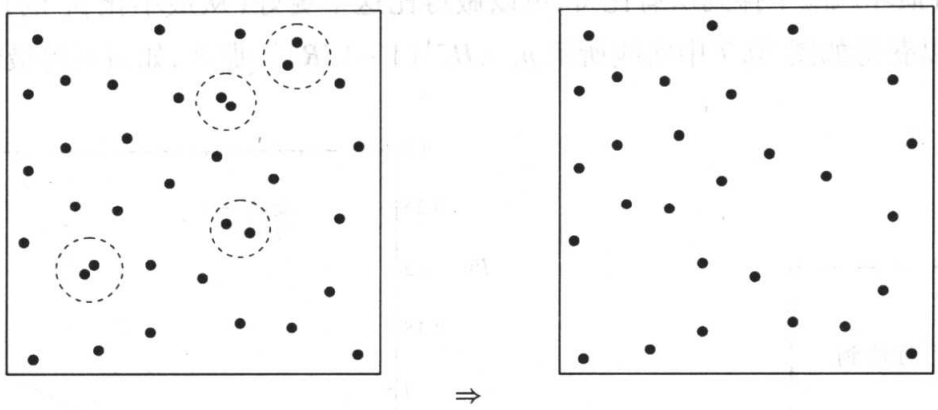
则可通过增加 N 来使式(10.17)所给出的平均差错概率小于 2δ。几乎可以满足上式。做如下3处改动。

1. 在证明中选择 $P(x)$ 为信道的最优输入分布。于是，条件 $R'<I(X;Y)-3\beta$ 变为 $R'<C-3\beta$。

2. 由于所有码的平均差错概率 $<2\delta$，所以必定存在一个码，它的分组差错概率均值 $p_{\mathrm{B}}(\mathscr{C})<2\delta$。

3. 为了证明不仅平均值而且最大差错概率 p_{BM} 也可以很小，对这个码进行修改：将码字中最差的一半删除掉，它们最容易产生错误。那些剩余部分的条件差错概率都必定小于 4δ。使用这些保留的码字来定义一个新码。这个新码有 $2^{NR'-1}$ 个码字，即，将码率从 R' 降为 $R'-1/N$(如果 N 很大，这只是一个可忽略的降低)，并得 $p_{\mathrm{BM}}<4\delta$。这种技巧称为删余，如图10.5所示。最后所得的码从码率和长度来讲也许不是

最优码,但它依然足以满足要求:证明噪声信道编码定理。



(a) 一个随机码 ●●● (b) 删除后

图 10.5 如何进行删除。(a) 在一个典型随机码中,一小部分码字会卷入冲突——成对的码字彼此非常接近,以至于当任何一个码字被传输时,差错概率都不是很小。通过删除所有这些易引起混淆的码字,可从一个随机码获得一个新码。(b) 所得的码有略少些的码字,所以有略低些的码率,它的最大差错概率大大降低了。

总之,可以"构造"一个码率为 $R'-1/N$ 的码,其中 $R'<C-3\beta$,最大差错概率 $<4\delta$。通过设置 $R'=(R+C)/2$,$\delta=\varepsilon/4$,$\beta<(C-R')/3$,且 N 足够大以保证其他条件成立,即可得到所述定理。于是定理的第一部分得证。 □

10.4 超出容量的通信(存在差错)

对于任意离散无记忆信道,已证明如图 10.6 所示 R 和 p_b 平面区域的可达性。证明了可将任意噪声信道转换成一个基本无噪声二进制信道,其码率最高可达每周期 C 比特。现在扩展非零差错概率的可达区域的右边界。[这称为*数据率失真理论*。]

使用一种新的技巧来解决此问题。由于我们知道可将噪声信道转换为一个码率略低的完美信道,只需考虑在一个*无噪声*信道上进行有差错的通信就足够了。如果允许出现差错,在一个无噪声信道上能够以多快的速率进行通信?

考虑一个无噪声二进制信道,并且假设我们迫使通信速率比它 1 比特的容量大。例如,如果要求发送者试图以每周期 $R=2$ 比特的速率通信,则他必须有效地将信息的一半扔掉。如果目标是要达到最小可能的误比特概率,那么采取什么方法最好呢?一个简单的策略是以信源比特的 $1/R$ 来进行通信,并将其他部分忽略掉。接收方随机地猜测所丢失的 $1-1/R$ 部分,且平均误比特概率为

$$p_b=\frac{1}{2}(1-1/R) \tag{10.19}$$

与这个策略相对应的曲线如图 10.7 中的虚线所示。

通过将损坏风险平摊到所有比特，可以做得比这个更好（从最小化 p_b 的角度而言）。实际上，可以获得如图 10.7 中实线所示 $p_b = H_2^{-1}(1 - 1/R)$。那么，如何获得最优性能？

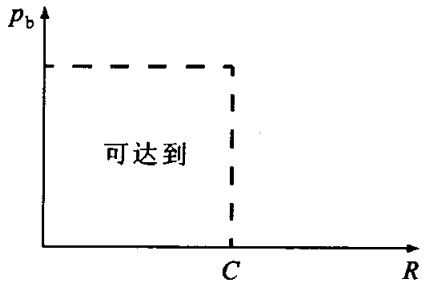


图 10.6 在定理的第一部分证明为可达的 R, p_b 平面区域。[已经证明可以使最大分组差错概率 p_{BM} 任意小，而比特差错概率 p_b 一定比 p_{BM} 小，所以也可使 p_b 任意小。]

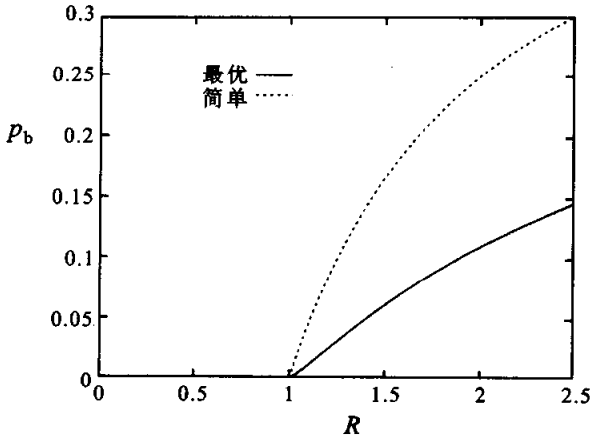


图 10.7 可达点 (R, p_b) 的一个简单上界和香农界

再次使用刚开发出来的一种工具，即噪声信道的 (N, K) 码，并且掉过头来，用*译码器*来定义一个有损耗的压缩器。具体而言，将一个优秀的 (N, K) 码看做一个二进制对称信道。假设这个码的码率为 $R' = K/N$，且能够对转移概率为 q 的二进制对称信道所引入的差错进行纠正。渐近地，存在码率为 R' 的码，其中 $R' \simeq 1 - H_2(q)$。回忆一下，如果将这些长度为 N 且可达容量的码中的一个与二进制对称信道联系起来，则（a）输出的概率分布接近均匀分布，因为输出的熵等于信源的熵 (NR') 加上噪声的熵 $(NH_2(q))$；（b）在这种情况下，码的最优译码器通常会将一个长度为 N 的接收向量映射成一个与接收向量相差 qN 比特的发送向量。

将希望发送的信号分成长度为 N 的分组（是的，是 N 而非 K）。将每个分组送往*译码器*并获得一个长度为 K 比特的较短信号，再将它送到无噪声信道上进行通信。为了对传输信息进行译码，将 K 比特的信息送到原始码的*编码器*。重新构造的信息将在某些比特上与原始信息有所不同——典型情况下有 qN 个。所以误比特概率会是 $p_b = q$。这个有损耗的压缩器的码率是 $R = N/K = 1/R' = 1/(1 - H_2(p_b))$。

现在，将这个有损耗的压缩器与那个容量为 C 的无误通信器相连，就证明了不高于下式定义的 (p_b, R) 曲线的通信可达性。

$$R = \frac{C}{1 - H_2(p_b)} \qquad \square \quad (10.20)$$

关于数据率失真理论的进一步讨论请参见 Gallager（1968 年），p. 451，或

McEliece(2002 年),p. 75。

10.5 不可达区域(定理的第三部分)

信源、编码器、噪声信道和译码器($s \to \boldsymbol{x} \to \boldsymbol{y} \to \hat{s}$)定义了一个马尔科夫(Markov)链

$$P(s,\boldsymbol{x},\boldsymbol{y},\hat{s}) = P(s)P(\boldsymbol{x}|s)P(\boldsymbol{y}|\boldsymbol{x})P(\hat{s}|\boldsymbol{y}) \tag{10.21}$$

数据处理不等式(习题 8.9)一定适用于这个链:$I(s;\hat{s}) \leqslant I(\boldsymbol{x};\boldsymbol{y})$。此外,由信道容量定义,有 $I(\boldsymbol{x},\boldsymbol{y}) \leqslant NC$,因此 $I(s;\hat{s}) \leqslant NC$。

假设一个系统达到速率 R 并具有比特差错概率 p_b;则互信息满足 $I(s;\hat{s}) \geqslant NR(1-H_2(p_b))$。但是 $I(s;\hat{s}) > NC$ 是不可达的,所以 $R > \dfrac{C}{1-H_2(p_b)}$ 也是不可达的。

习题 10.1[p] 补充前面讨论的细节。如果 $\hat{s}$ 和 s 之间的比特差错是独立的,那么有 $I(s;\hat{s}) = NR(1-H_2(p_b))$。如果在这些比特差错之间存在复相关性,情况会怎样?为什么不等式 $I(s;\hat{s}) \geqslant NR(1-H_2(p_b))$ 会成立?

10.6 计算容量[①]

已经证明了信道容量是实现可靠通信的最大速率。怎样才能计算一个已知离散无记忆信道的容量?需要找出它的最优输入分布。一般地,使用互信息关于输入概率的导数,可通过计算机搜索来找到最优输入分布。

▷ **习题 10.2**[p] 对于一个条件概率为 $Q_{j|i}$ 的信道,求 $I(X;Y)$ 关于输入概率 p_i 的导数 $\partial I(X;Y)/\partial p_i$。

习题 10.3[p] 证明 $I(X;Y)$ 是输入概率向量 $\boldsymbol{p}$ 的凹⌒函数。

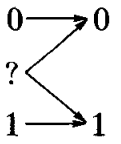
由于 $I(X;Y)$ 是输入分布 $\boldsymbol{p}$ 中的凹⌒函数,所以使 $I(X;Y)$ 静止不变的任何概率分布 $\boldsymbol{p}$ 一定取得 $I(X;Y)$ 的全局最大值。所以一种诱人的方法是将 $I(X;Y)$ 的导数放入寻找 $I(X;Y)$ 局部极大值的程序,也就是使下式成立的输入分布 $P(x)$。

$$\frac{\partial I(X;Y)}{\partial p_i} = \lambda, \quad \text{对于所有 } i \tag{10.22}$$

其中,λ 是与约束条件 $\sum_i p_i = 1$ 相关的拉格朗日乘数。但是,这种方法也许不能找出正确答案,因为使 $I(X;Y)$ 最大化的分布也许会使某些输入的 $p_i = 0$。一个简单的例子是三进制混淆信道。

三进制混淆信道 $\mathcal{A}_X = \{0,?,1\}$。$\mathcal{A}_Y = \{0,1\}$。

① 第 10.6 节 ~ 第 10.8 节包含一些较深的题材。建议初读者略过第 10.9 节



$$P(y=0|x=0)=1;P(y=0|x=?)=1/2;P(y=0|x=1)=0;$$
$$P(y=1|x=0)=0;P(y=1|x=?)=1/2;P(y=1|x=1)=1$$

无论何时使用输入“?”,输出都是随机的;其他输入都是可靠的输入。不使用输入“?”,可获得1比特的最大信息速率。

▷ **习题 10.4**[⌐] 绘制这个信道的互信息,以输入分布 $\boldsymbol{p}$ 的函数表示。选择 $\boldsymbol{p}$ 的一个简便的二维表示。

因此,在用最优化子程序向上搜索 $I(X;Y)$ 时,一定要考虑可能会遇到不等式约束 $p_i \geq 0$。

▷ **习题 10.5**[⌐] 与式(10.22)类似,描述使 $I(X;Y)$ 取得最大值的点所要满足的条件,并描述一个用来求信道容量的计算机程序。

对于寻找最优输入分布也许有所帮助的结果

1. 所有输出都要使用。
2. $I(X;Y)$ 是信道参数的一个凸⌣函数。①
3. 也许存在多个最优输入分布,但是对于输出来讲,它们看起来都是一样的。

▷ **习题 10.6**[⌐] 证明不存在未被一个最优输入分布使用的输出 y,除非它是不可达的,即对于所有 x,都有 $Q(y|x)=0$。

习题 10.7[⌐] 证明 $I(X;Y)$ 是 $Q(y|x)$ 的一个凸⌣函数。

习题 10.8[⌐] 证明一个信道的所有最优输入分布具有相同的输出概率分布 $P(y)=\sum_x P(x)Q(y|x)$。

这些结果以及 $I(X;Y)$ 是一个输入概率向量 $\boldsymbol{p}$ 的凹⌢函数这一事实,一起证明了采用对称性来求解对称信道容量是正确的。如果一个信道在一组对称运算下(例如,交换输入符号和交换输出符号)是不变的,则给定任意不对称的最优输入分布,即在这些运算下会变化,就可以通过平均这个最优输入分布和它所有的特定排列形式来产生另外一个输入分布,而这些特定排列形式都是通过对原最优输入分布应用对称运算而得到的。由对称性,排列后的分布与原分布一定具有相同的 $I(X;Y)$,但是由于 I 的凹性,通过平均所产生的新输入分布所具有的 $I(X;Y)$ 一定大于或等于原分布。

对称信道

为了使用对称论点,定义一个对称信道将会有所帮助。笔者喜欢 Gallager(1968

① 提醒:术语“凸⌣”的意思是“凸”,而术语“凹⌢”的意思是“凹”;运用小的“微笑”和“皱眉”符号只是为了提醒你凸和凹的含义。

年)的定义。

一个离散无记忆信道是一个对称信道 如果可以将输出集合划分为子集,使得每个子集的转移概率矩阵满足如下性质:每一行(如果多于1行)是其他行的一个转置,且每一列都是其他列的一个转置。

例 10.9 这个信道

$$\begin{aligned} &P(y=0|x=0)=0.7;P(y=0|x=1)=0.1;\\ &P(y=?|x=0)=0.2;P(y=?|x=1)=0.2;\\ &P(y=1|x=0)=0.1;P(y=1|x=1)=0.7 \end{aligned} \tag{10.23}$$

是一个对称信道,因为它的输出可被划分为(0,1)和"?",于是矩阵可被重写为

$$\begin{array}{ll} P(y=0|x=0)=0.7; & P(y=0|x=1)=0.1;\\ P(y=1|x=0)=0.1; & P(y=1|x=1)=0.7;\\ \hline P(y=?|x=0)=0.2; & P(y=?|x=1)=0.2 \end{array} \tag{10.24}$$

对称性是一种很有用的性质,因为线性码可以在对称信道上达到通信容量,这将在后续章节中看到。

习题 10.10[2] 证明对于一个具有任意多个输入的对称信道,均匀的输入分布是一个最优输入分布。

▷**习题 10.11**[2] 是否存在非对称信道,其最优输入分布是均匀的?找出一个例子,或证明这样的信道不存在。

10.7 其他编码定理

在这一章中证明的噪声信道编码定理具有较强的通用性,可适用于任何离散无记忆信道;但是它不是很具体。定理只是说明可通过使用分组长度 N 足够大的码,来实现差错概率为 ε、速率为 R 的可靠通信。定理并未说明需要多大的 N 才能获得给定的 R 和 ε。

大概 ε 越小、R 越接近 C,N 就要越大。

噪声信道编码定理——显式 N - 相关版

对于一个离散无记忆信道、一个分组长度 N 和一个码率 R,存在长度为 N 的分组码,其平均差错概率满足

$$p_B \leqslant \exp[-NE_r(R)] \tag{10.25}$$

其中,$E_r(R)$ 是信道的随机编码指数,它在 $0 \leqslant R < C$ 上是 R 的一个正的递减凸⌣函数。随机编码指数也叫做可靠性函数。[根据删余的论点,也可以证明存在分组码使得最大差错概率 p_{BM} 也随着 N 以指数级减小。]

$E_r(R)$的定义由 Gallager(1968 年),p. 139 给出。当 $R \to C$ 时,$E_r(R)$趋近于零;这个函数的典型特性如图 10.8 所示。计算感兴趣的信道的随机编码指数是一项有挑战性的任务,研究者已经为此做出了很大努力。甚至对于诸如二进制对称信道之类的简单信道,$E_r(R)$也不存在简单的表达式。

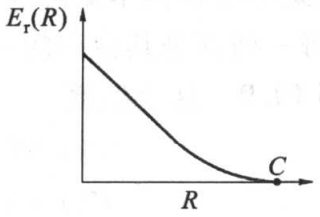


图 10.8 一个典型的随机编码指数

作为分组长度函数的差错概率下界

上述定理表明,存在一些码具有比 $\exp[-NE_r(R)]$更小的 p_B。但是差错概率究竟能有多小?它是否还可以更小?

对离散无记忆信道上分组长度为 N 的任意码,假设所有信源都被等概率地使用,差错概率满足

$$p_B \gtrsim \exp[-NE_{sp}(R)] \tag{10.26}$$

其中,函数 $E_{sp}(R)$是信道的填球指数,它在 $0 \leqslant R < C$ 上是 R 的一个正递减凸⌣函数。

关于这个结果的准确表述和进一步的参考资料,请参阅 Gallager(1968 年),p. 157。

10.8 噪声信道编码定理与编码实践

假设客户希望为一个噪声信道购买一个纠错编码和译码器。上述结果允许人们提供如下服务:如果他告诉我们信道特性、所要求的码率 R 以及想要得到的差错概率 p_B,在计算出相关函数 C、$E_r(R)$和 $E_{sp}(R)$之后,就可以告诉他使用一个特定分组长度 N 可求解他的问题;实际上,几乎任意随机挑选的具有那个分组长度的码都可以胜任这项工作。不幸的是,在实际中尚未找到如何实现这些编码器和译码器的方法;要实现一个 N 很大的随机码的编码器和译码器,其开销会随 N 指数级地增长。

此外,从实用的角度来看,客户不大可能确切知道他要使用的信道。所以 Berlekamp(1980 年)建议:解决纠错问题的合理方法是设计编码—译码系统,并通过信道噪声水平函数绘制它们在不同理想化信道上的性能。可将这些图表(图 47.17 给出了其中之一)展示给客户看,他就可以在这些供出售的系统中做出选择,而无需详细说明他认为自己的信道实际上是什么样子。采用这种态度来对待实际问题,函数 $E_r(R)$和 $E_{sp}(R)$的重要性就降低了。

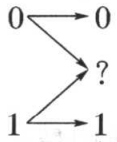
10.9 进一步的习题



习题 10.12[2] 一个输入为 x、输出为 y 的二进制删除信道具有如下转移概率

矩阵

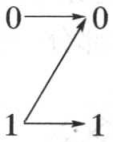
$$\boldsymbol{Q}=\begin{bmatrix}1-q & 0\\ q & q\\ 0 & 1-q\end{bmatrix}$$



对于一般输入分布$\{p_0,p_1\}$，求输入和输出之间的互信息$I(X;Y)$，并证明这个信道的容量是$C=(1-q)$比特。

一个Z信道具有如下转移概率矩阵

$$\boldsymbol{Q}=\begin{bmatrix}1 & q\\ 0 & 1-q\end{bmatrix}$$



试证明：使用一个(2,1)码，Z信道的**两次**使用可用来模拟删除信道的**一次**使用，并说明那个删除信道的删除概率。由此证明Z信道的容量C_Z满足$C_Z\geqslant\frac{1}{2}(1-q)$比特。

解释为什么结果$C_Z\geqslant\frac{1}{2}(1-q)$是一个不等式而非一个等式。

习题10.13[p] 一条横跨大西洋的电缆包括$N=20$根不易区分的电线。你的工作是：判断哪一根电线是哪一根，即，在电线的每个末端都做一个一致的标记。你的唯一办法是将电线以两根或多根为一组彼此连接，并用一个关联性测试仪来测试连通性。你至少需要进行多少次横跨大西洋的“旅行”，具体如何做？

对于N很大的情况，比如$N=1\ 000$，如何解决这个问题？

作为一个例子，如果N是3，则可以用两个步骤来解决这个任务：将一根电线的一端标注为a，将另外两根连接在一起，横跨大西洋，测试哪两根电线连到了一起，将它们标为b和c，而将没有连接的标为a，然后将b和a连接在一起并再次横跨大西洋，此时b和c是断开的，就可以推断出b和c分别是哪一根了。

这个问题可通过持续的搜索来完成，但在这一章中提出它的原因是，也可以通过基于最大化已获信息的贪婪方法来解决。令未知的电线排列为x。已经选择了电线在一端的一组连接$\mathcal{C}$，于是可在另一端进行测试，而这些测试y传递了关于x的信息。有多少信息？什么样的一组连接会使y所传递的关于x的信息最大化？

10.10 习题解答

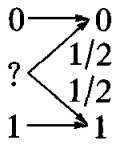
习题10.4的解答 如果输入分布$\boldsymbol{p}=(p_0,p_?,p_1)$，则互信息为

$$I(X;Y)=H(Y)-H(Y|X)=H_2(p_0+p_?/2)-p_? \tag{10.27}$$

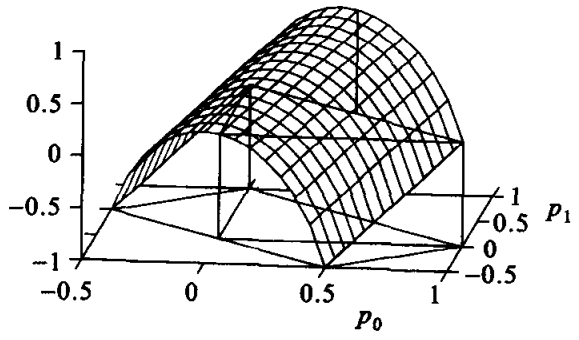
可以用两种方法绘制这个函数的简图：通过仔细观察函数，或通过考查特殊情况。

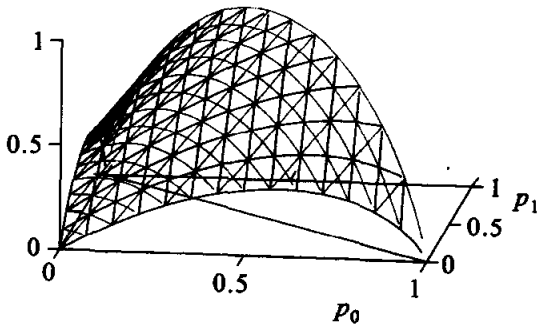
为了方便绘图，将使用$\boldsymbol{p}$的二维表示，即将p_0和p_1作为独立变量，使得

$$\boldsymbol{p}=(p_0,p_?,p_1)=(p_0,(1-p_0-p_1),p_1)$$



通过观察　如果使用量 $p_* \equiv p_0 + p_?/2$ 和 $p_?$ 作为两种自由度，互信息将变得非常简单：$I(X;Y) = H_2(p_*) - p_?$。换算回 $p_0 = p_* - p_?/2$ 和 $p_1 = 1 - p_* - p_?/2$，得到下面左边的简图。这个函数像一条隧道，随着 p_0 和 p_1 的增加而上升。为了获得所要求的 $I(X;Y)$ 的图形，必须去掉这条隧道位于概率的可行单纯形之外的部分；通过重画表面，只显示 $p_0 > 0$ 和 $p_1 > 0$ 的部分，可以完成这项任务。函数的全图如右图所示。





特殊情形　在 $p_? = 0$ 的特殊情形，信道是一个无噪声的二进制信道，并且 $I(X;Y) = H_2(p_0)$。

在 $p_0 = p_1$ 的特殊情形，$H_2(p_0 + p_?/2)$ 项等于 1，所以 $I(X;Y) = 1 - p_?$。

在 $p_0 = 0$ 的特殊情形，信道是 Z 信道，其差错概率为 0.5。从前面的章节可知如何描绘它的简图（图 9.3）。

这些特殊情形使我们能构建一个轮廓，如图 10.9 所示。

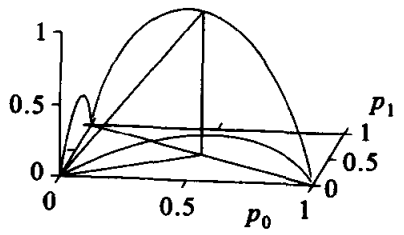


图 10.9　三进制混淆信道互信息的轮廓

习题 10.5 的解答　使 $I(X;Y)$ 最大化的 $\boldsymbol{p}$ 的充分必要条件是

$$\left.\begin{aligned} \frac{\partial I(X;Y)}{\partial p_i} &= \lambda, \quad p_i > 0 \\ \frac{\partial I(X;Y)}{\partial p_i} &\leqslant \lambda, \quad p_i = 0 \end{aligned}\right\} \quad \text{对于所有 } i \tag{10.28}$$

其中 λ 是一个与容量有关的常数，满足关系式 $C = \lambda + \log_2 e$。

这个结果可以用于计算导数的计算机程序中，并根据这些导数的差值按比例地增加或减小概率 p_i。

这个结果也有助于懒惰的寻找容量者，他们擅长猜测。在猜到最优输入分布之后，就可以简单地验证式（10.28）成立。

习题 10.11 的解答　由于设计信道时有 $I(J-1)$ 个自由度，而最优输入分布仅有 $(I-1)$ - 维，所以预期肯定存在具有均匀最优输入分布的非对称信道；所以在围绕

对称信道的 $I(J-1)$ - 维摄动空间中，预期存在一个 $I(J-1)-(I-1)=I(J-2)+1$ 维的摄动子空间，它保持最优输入分布不变。

下面是一个明显的例子，有点像一个 Z 信道。

$$\boldsymbol{Q}=\begin{bmatrix}0.958\,5 & 0.041\,5 & 0.35 & 0.0\\ 0.041\,5 & 0.958\,5 & 0.0 & 0.35\\ 0 & 0 & 0.65 & 0\\ 0 & 0 & 0 & 0.65\end{bmatrix} \tag{10.29}$$

习题 10.13 的解答 仅用两次旅行即可解决任意 $N>2$ 的做标记问题，每个方向穿越一次大西洋。

求解这个问题的信息论方法的关键步骤是，写下一个划分的信息量，其中的组合对象是连到一起的电线子集。如果 N 根电线被分组成大小为 1 的 g_1 个子集，大小为 2 的 g_2 个子集，……，那么这种划分的数量为

$$\Omega=\frac{N!}{\prod_r (r!)^{g_r} g_r!} \tag{10.30}$$

并且这种划分的信息量是上述数量的对数。在贪婪策略中，选择第一划分来对这个信息量进行最大化。

可以将数量 g_r 视为实数，并在 $\sum_r g_r r=N$ 的约束条件下，用该数量来对这个信息量进行最大化。为了约束引入一个拉格朗日因子 λ，导数为

$$\frac{\partial}{\partial g_r}\Big(\log\Omega+\lambda\sum_r g_r r\Big)=-\log r!\ -\log g_r+\lambda r \tag{10.31}$$

令这个导数为零，得到一个相当漂亮的表达式

$$g_r=\frac{\mathrm{e}^{\lambda r}}{r!} \tag{10.32}$$

最优 g_r 正比于一个泊松分布！可以通过将 g_r 插入约束条件 $\sum_r g_r r=N$ 来求解拉格朗日因子，它给出了一个隐式方程：

$$N=\mu\mathrm{e}^{\mu} \tag{10.33}$$

其中 $\mu\equiv\mathrm{e}^{\lambda}$ 是为方便起见，对拉格朗日因子所进行的一个再参量化。图 10.10(a) 所示为 $\mu(N)$ 的一个图形；图 10.10(b) 所示为 $\mu=2.2$ 时所推导的非整数赋值 g_r，及其附近的整数 $g_r=\{1,2,2,1,1\}$，这启发我们将第一划分设置为 (a)(bc)(de)(fgh)(ijk)(lmno)(pqrst)。这个划分在另一端产生一个随机划分，其信息量为 $\log\Omega=40.4$ 比特，远远大于我们推断横跨大西洋的排列所需获得的总信息量 $\log 20!\simeq 61$ 比特的一半。[相反，如果所有电线成对地连接在一起，所产生的信息量仅有约 29 比特。]如何选择第二个划分将留给读者来完成。使用一个香农式方法是适当的，在另一端选取一个随机划分，使用相同的 $\{g_r\}$；你需要确保两个划分应尽可能不同。

如果 $N\neq 2,5$ 或 9，则标记问题有一个特别易于实现的解决方案，称作 Knowlton - Graham 划分：用两种方法将 $\{1,\cdots,N\}$ 划分为不相交的集合 A 和 B，并满足对于任

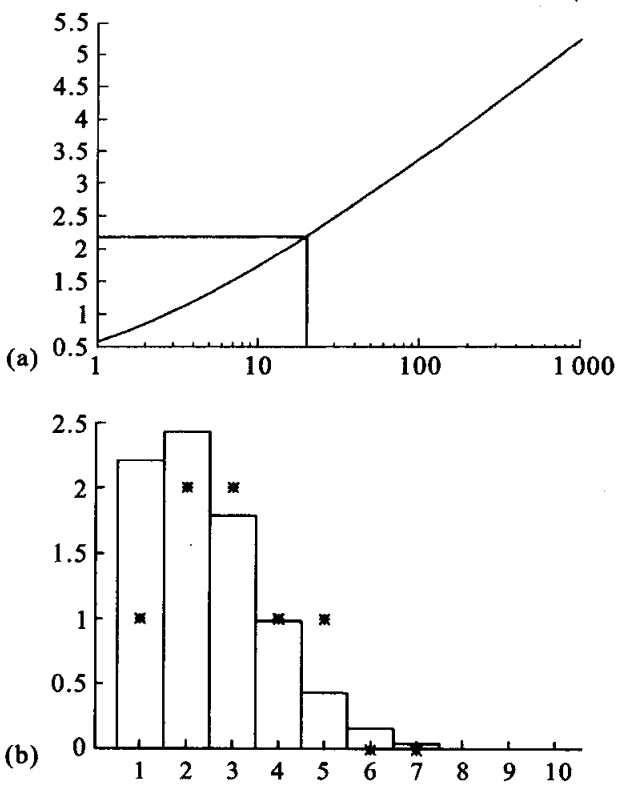


图 10.10 使用拉格朗日因子近似求解电线标记问题。(a) 参数 μ 是 N 的函数；$\mu(20)=2.2$ 的值是突出显示的。(b) 实线表示函数 $g_r=\mu^r/r!$ 的非整数值，用交叉号表示由那些非整数值推导出的 g_r 的整数值。

意 j 和 k，最多有一个元素同时出现在基为 j 的 A 集和基为 k 的 B 集中(Graham，1966 年；Graham 与 Knowlton，1968 年)。

关于第 11 章

在阅读第 11 章之前，你应该已经阅读了第 9 章和第 10 章。

同时你还需要熟悉高斯分布。

一维高斯分布 如果一个随机变量 y 服从高斯分布，且具有均值 μ 和方差 σ^2，并写成

$$y \sim \text{Normal}(\mu,\sigma^2) \text{ 或 } P(y) = \text{Normal}(y;\mu,\sigma^2) \tag{11.1}$$

则 y 的分布为

$$P(y|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp[-(y-\mu)^2/2\sigma^2] \tag{11.2}$$

[用符号 P 来表示概率密度和概率。]

有时称方差的倒数 $\tau \equiv 1/\sigma^2$ 为高斯分布的精度。

多维高斯分布 如果 $\boldsymbol{y} = (y_1, y_2, \cdots, y_N)$ 服从多元高斯分布，则

$$P(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{A}) = \frac{1}{Z(\boldsymbol{A})}\exp\left(-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{x})^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{y}-\boldsymbol{x})\right) \tag{11.3}$$

其中，$\boldsymbol{x}$ 是分布的均值，$\boldsymbol{A}$ 是方差 - 协方差矩阵的逆，而归一化常数是 $Z(\boldsymbol{A}) = (\det(\boldsymbol{A}/2\pi))^{-1/2}$。

这个分布满足这样一个性质：y_i 的方差 Σ_{ii} 以及 y_i 和 y_j 的协方差 Σ_{ij} 由

$$\Sigma_{ij} \equiv \mathcal{E}[(y_i - \bar{y}_i)(y_j - \bar{y}_j)] = A_{ij}^{-1} \tag{11.4}$$

给出，其中 $\boldsymbol{A}^{-1}$ 是矩阵 $\boldsymbol{A}$ 的逆。

一个分量 y_i 的边缘分布 $P(y_i)$ 是高斯的；分量的任意子集的联合边缘分布是多元高斯的；且在给定另一个子集之值时，任意子集的条件密度，例如 $P(y_i|y_j)$，也是高斯的。

第 11 章

纠错码与实际信道

已经证明的噪声信道编码定理表明，对于任意噪声信道都存在可靠的纠错码。在这一章中，探讨两个问题。

第一，很多实际信道的输入和输出是实数而非离散的。关于这些连续信道，香农能告诉我们什么呢？应该怎样将数字信号映射为模拟波形，及如何做反向转换？

第二，怎样构造实用的纠错码，及相对于香农定理所证明的可能性，在实际中可达到什么水平？

11.1 高斯信道

最普遍的实输入—实输出信道模型是高斯信道。

高斯信道 具有一个实输入 x 和一个实输出 y。已知 x 时 y 的条件分布是一个高斯分布

$$P(y|x)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp[-(y-x)^2/2\sigma^2] \tag{11.5}$$

这个信道具有连续的输入和输出，但在时间上是离散的。后面将证明，某些连续时间信道等价于离散时间高斯信道。

这个信道有时被称为添加高斯白噪声(AWGN)信道。

与离散信道相同，将要讨论在这个信道上可达到多高的无误信息传输速率。

关于连续时间信道的一个动机

考虑输入和输出在时间上连续的一个物理信道(比如，电信道)。输入 $x(t)$，输出 $y(t)=x(t)+n(t)$。

我们的传输有一个功率代价。长度为 T 的传输的平均功率可能受限，于是有

$$\int_0^T \mathrm{d}t[x(t)]^2/T\leqslant P \tag{11.6}$$

假设接收信号与 $x(t)$ 相差一个添加噪声 $n(t)$(例如 Johnson 噪声)，将噪声建模为高斯白噪声。这个噪声的幅度用噪声谱密度 N_0 来量化。

如何用这样的信道来传输信息？考虑用一个周期为 T 的信号传输 N 个实数所组成的集合 $\{x_n\}_{n=1}^N$，该信号由一组标准正交基函数 $\phi_n(t)$ 的加权组合构成(如图 11.1所示)

$$x(t)=\sum_{n=1}^{N}x_n\phi_n(t) \tag{11.7}$$

其中 $\int_0^T \mathrm{d}t\phi_n(t)\phi_m(t)=\delta_{nm}$。于是接收方可以计算标量

$$y_n\equiv\int_0^T \mathrm{d}t\phi_n(t)y(t)=x_n+\int_0^T \mathrm{d}t\phi_n(t)n(t) \tag{11.8}$$

$$\equiv x_n+n_n \tag{11.9}$$

其中，$n=1,\cdots,N$。如果没有噪声，则 y_n 将等于 x_n。高斯白噪声 $n(t)$ 将标量噪声 n_n 加到估计值 y_n 上。这个噪声是高斯的：

$$n_n\sim\mathrm{Normal}(0,N_0/2) \tag{11.10}$$

其中 N_0 是上面介绍的谱密度。因此这样使用的一个连续信道等价于式(11.5)所定义的高斯信道。功率约束 $\int_0^T \mathrm{d}t[x(t)]^2\leqslant PT$ 定义了对于信号幅度 x_n 的一个约束

$$\sum_n x_n^2\leqslant PT\Rightarrow\overline{x_n^2}\leqslant\frac{PT}{N} \tag{11.11}$$

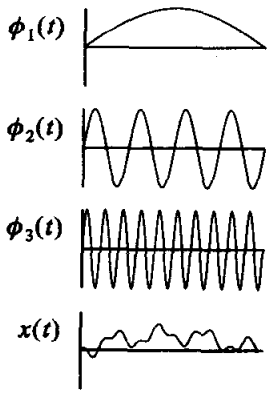


图 11.1 3 个基函数以及它们的一个加权组合，$x(t)=\sum_{n=1}^{N}x_n\phi_n(t)$，其中 $x_1=0.4, x_2=-0.2, x_3=0.1$。

在返回高斯信道这个主题之前，定义连续信道的带宽(用赫兹来度量)为

$$W=\frac{N^{\max}}{2T} \tag{11.12}$$

其中 $N^{\max}$ 是在一个长度为 T 的时间间隔内所能产生的标准正交函数的最大数量。设想由最大频率为 W 的标准正交余弦和正弦曲线产生一个周期为 T 的带限信号，由此可以得到这个定义的一个动机。标准正交函数的数量为 $N^{\max}=2WT$。这个定义与奈奎斯特(Nyquist)采样定理有关：如果一个信号中所出现的最高频率为 W，则信号可以由一串相距奈奎斯特间隔 $\Delta t=1/2W$ 秒的离散样值点完全地确定。

所以使用一个具有带宽 W、噪声谱密度 N_0 和功率 P 的实连续信道，等价于每秒 $N/T=2W$ 次使用一个噪声水平为 $\sigma^2=N_0/2$ 并满足信号功率限制 $\overline{x_n^2}\leqslant P/2W$ 的高斯信道。

E_b/N_0 的定义

假设高斯信道 $y_n=x_n+n_n$ 与一个编码系统一起使用，以每次信道使用 R 比特的速率来传输二进制信源比特。应如何比较两个具有不同通信速率 R 并使用不同功率 $\overline{x_n^2}$ 的编码系统？用大速率 R 进行发送是一件好事，使用小的功率也是一件好事。

通常使用每信源比特的功率 $E_b=\overline{x_n^2}/R$ 与噪声谱密度 N_0 的比值来度量补偿速率

信噪比①

$$E_b/N_0 = \frac{\overline{x_n^2}}{2\sigma^2 R} \tag{11.13}$$

E_b/N_0 是用来比较高斯信道编码方法的度量之一。

11.2 推断一个实际信道的输入

“脉冲的最佳检测”

在 1944 年,香农写了一份关于两类已知形状的脉冲之间最大差异问题的备忘录(Shannon,1993 年),这两个脉冲分别用向量 $\boldsymbol{x}_0$ 和 $\boldsymbol{x}_1$ 表示(如图 11.2 所示),并假设其中之一已经在一个噪声信道上传输。这是一个模式识别问题。假设噪声是高斯随机变量,其概率密度为

$$P(\boldsymbol{n}) = \left[\det\left(\frac{\boldsymbol{A}}{2\pi}\right)\right]^{1/2} \exp\left(-\frac{1}{2}\boldsymbol{n}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{n}\right) \tag{11.14}$$

其中 $\boldsymbol{A}$ 是噪声方差—协方差矩阵的逆,它是一个对称正定矩阵。(如果 $\boldsymbol{A}$ 是一个单位矩阵的倍数,$\boldsymbol{I}/\sigma^2$,则噪声是“白的”。对于更为一般的 $\boldsymbol{A}$,噪声是“有色的”。)已知信源信号是 s(0 或 1),则接收向量 $\boldsymbol{y}$ 的出现概率是

$$P(\boldsymbol{y}|s) = \left[\det\left(\frac{\boldsymbol{A}}{2\pi}\right)\right]^{1/2} \exp\left(-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{x}_s)^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{y}-\boldsymbol{x}_s)\right) \tag{11.15}$$

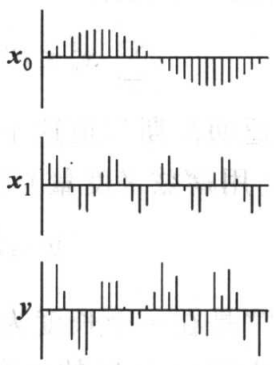


图 11.2 用 31 维向量所表示的两个脉冲 $\boldsymbol{x}_0$ 和 $\boldsymbol{x}_1$ 以及其中之一的噪声形式 $\boldsymbol{y}$。

最优检测器基于后验概率比

$$\frac{P(s=1|\boldsymbol{y})}{P(s=0|\boldsymbol{y})} = \frac{P(\boldsymbol{y}|s=1)P(s=1)}{P(\boldsymbol{y}|s=0)P(s=0)} \tag{11.16}$$

$$= \exp\left(-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{x}_1)^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{y}-\boldsymbol{x}_1) + \frac{1}{2}(\boldsymbol{y}-\boldsymbol{x}_0)^{\mathrm{T}}A(\boldsymbol{y}-\boldsymbol{x}_0) + \ln\frac{P(s=1)}{P(s=0)}\right)$$

$$= \exp(\boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{x}_1-\boldsymbol{x}_0) + \theta) \tag{11.17}$$

其中 θ 是独立于接收向量 $\boldsymbol{y}$ 的常数

$$\theta = -\frac{1}{2}\boldsymbol{x}_1^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}_1 + \frac{1}{2}\boldsymbol{x}_0^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}_0 + \ln\frac{P(s=1)}{P(s=0)} \tag{11.18}$$

如果检测器必须做出一个判决(即,猜测 $s=1$ 或 $s=0$),则使差错概率最小化的

① E_b/N_0 是无量纲的,但是它通常以分贝为单位来计量;已知数值为 $10\log_{10}E_b/N_0$。

判决是猜测最有可能的假设。可以用如下判别函数的形式写出最优判决。

$$a(\boldsymbol{y}) \equiv \boldsymbol{y}^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{x}_1 - \boldsymbol{x}_0) + \theta \tag{11.19}$$

其判决为

$$\begin{aligned} &a(\boldsymbol{y}) > 0 \rightarrow \text{猜测 } s = 1 \\ &a(\boldsymbol{y}) < 0 \rightarrow \text{猜测 } s = 0 \\ &a(\boldsymbol{y}) = 0 \rightarrow \text{猜测二者中的任意一个。} \end{aligned} \tag{11.20}$$

注意 $a(\boldsymbol{y})$ 是接收向量的一个线性函数

$$a(\boldsymbol{y}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{y} + \theta \tag{11.21}$$

其中 $\boldsymbol{w} \equiv \boldsymbol{A}(\boldsymbol{x}_1 - \boldsymbol{x}_0)$，如图 11.3 所示。

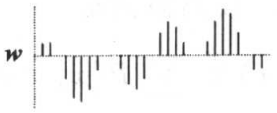


图 11.3 用来区分 $\boldsymbol{x}_0$ 和 $\boldsymbol{x}_1$ 的加权向量 $\boldsymbol{w} \propto \boldsymbol{x}_1 - \boldsymbol{x}_0$。

11.3 高斯信道的容量

直到现在仅仅度量了离散变量的联合、边缘和条件熵。为了定义由连续变量所传递的信息，必须解决两个问题——实线的无限长度和实数的无限精度。

无限输入

使用一次高斯信道可传输多少信息？如果允许将任意实数 x 输入高斯信道中，则通过设置 $x = d_1 d_2 d_3 \cdots d_N 000\cdots000$，可以传输 N 位数字 $d_1 d_2 d_3 \cdots d_N$ 的巨大字符串。可以通过增加 N 使得在仅仅一次通信中无误传输的信息量变得任意大，并通过增加 x 后面零的个数，使得通信的可靠性任意高。但是，与大的输入相关联的通常是某个功率代价，更不必提及接收方可接受的动态范围等实际限制。因此按照惯例，为每一个输入 x 引入一个成本函数 $v(x)$，并且限制码的平均成本 $\bar{v}$ 小于或等于某个最大值。可以证明一个通用信道编码定理，包括输入的一个成本函数——参见 McEliece(1977 年)。结果是信道容量 $C(\bar{v})$，它是可允许成本的一个函数。对于高斯信道，将要假设一个成本

$$v(x) = x^2 \tag{11.22}$$

使得输入的“平均功率”$\overline{x^2}$ 是受约束的。我们将对实际电信道的情形给出上述成本函数的一个动机：在这个电信道中，物理功率损耗确实是 x 的二次函数。约束条件 $\overline{x^2} = \bar{v}$ 使得它不可能在高斯信道的一次使用中传输无限信息。

无限精度

通过简单地用积分代替求和，来定义实变量的联合、边缘和条件熵，这看上去很诱人，但是这并非一个定义得好的方法。随着我们将一个时间间隔划分为越来越小的间隔，离散分布的熵逐渐发散(以粒度的对数)(如图 11.4 所示)。同时，也不允许计算诸如概率密度 $P(x)$(它的维数是 $[x]^{-1}$)之类维度相关量的对数。

但是有一个信息度量具有良好的极限，即互信息——这才是问题的关键，因为它度量了一个变量关于另一个变量所传递信息量的大小。在离散情形

$$I(X;Y) = \sum_{x,y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \tag{11.23}$$

由于 log 函数的自变量是同一空间上两个概率的比值，所以将 $P(x,y)$、$P(x)$ 和 $P(y)$ 作为概率密度不会有问题，并用一个积分来代替求和

$$I(X;Y) = \int \mathrm{d}x\mathrm{d}y P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \tag{11.24}$$

$$= \int \mathrm{d}x\mathrm{d}y P(x)P(y|x)\log\frac{P(y|x)}{P(y)} \tag{11.25}$$

现在可以就高斯信道提出这些问题：(a) 什么样的概率分布 $P(x)$ 可以最大化互信息（满足约束条件 $\overline{x^2} = v$）？(b) 最大的互信息是否还可以度量这个实际信道的最大无误通信速率，如同在离散信道上所做的？

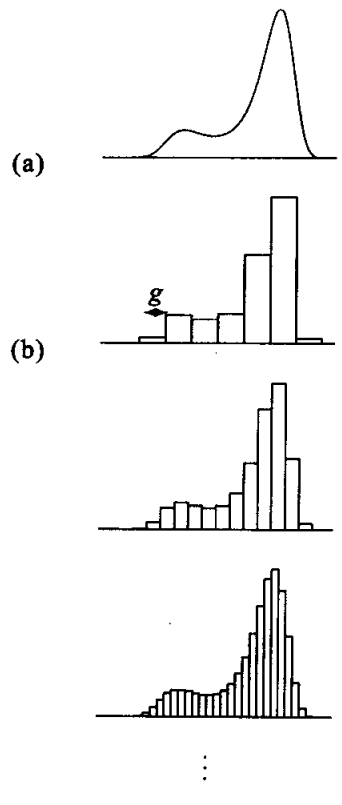


图 11.4 (a) 一个概率密度 $P(x)$。问题：能定义这个概率密度的“熵”吗？(b) 随着粒度大小 g 的减少，可以计算一个概率分布序列的熵，但这些熵趋向于 $\int P(x)\log\frac{1}{P(x)g}\mathrm{d}x$，它与 g 有关：每当 g 减半时，熵就增加 1 比特，所以 $\int P(x)\log\frac{1}{P(x)}\mathrm{d}x$ 是一个不合理的积分。

习题 11.1[求] 证明最大化互信息（满足约束条件 $\overline{x^2} = v$）的概率分布 $P(x)$ 是一个均值为零、方差为 v 的高斯分布。

▷ **习题 11.2**[求] 在这种最优化分布的情形下，证明互信息 $I(X;Y)$ 为

$$C = \frac{1}{2}\log\left(1 + \frac{v}{\sigma^2}\right) \tag{11.26}$$

这是一个重要的结论。将看到高斯信道的容量是信噪比 v/σ^2 的函数。

已知高斯输入分布时的推理

如果 $P(x) = \mathrm{Normal}(x;0,v)$ 并且 $P(y|x) = \mathrm{Normal}(y;x,\sigma^2)$，则 y 的边缘分布为 $P(y) = \mathrm{Normal}(y;0,v+\sigma^2)$，已知输出为 y，输入的后验概率是

$$P(x|y) \propto P(y|x)P(x) \tag{11.27}$$

$$\propto \exp(-(y-x)^2/2\sigma^2)\exp(-x^2/2v) \tag{11.28}$$

$$= \mathrm{Normal}\left(x;\frac{v}{v+\sigma^2}y,\left(\frac{1}{v}+\frac{1}{\sigma^2}\right)^{-1}\right) \tag{11.29}$$

[通过计算指数的平方项,可完成式(11.28)~(11.29)的步骤。]这个公式值得仔细琢磨。后验概率的均值$\frac{v}{v+\sigma^2}y$可视做与输出最为吻合的值$x=y$和与先验最为吻合的值$x=0$的加权组合

$$\frac{v}{v+\sigma^2}y=\frac{1/\sigma^2}{1/v+1/\sigma^2}y+\frac{1/v}{1/v+1/\sigma^2}0 \tag{11.30}$$

权重$1/\sigma^2$和$1/v$是式(11.28)中相乘在一起的两个高斯的精度:先验分布和似然分布。

后验分布的精度是这两个精度之和。这是一个通用的性质:两个独立的信源通过高斯分布为一个未知变量提供信息时,精度总是相加的。[这与众所周知的关系"当独立变量相加时,它们的方差也相加"是相对偶的。]

高斯信道的噪声信道编码定理

已经计算了一个最大互信息。它是否对应于无误信息传输的最大可能速率?证明它正确的一种方法是定义输入和输出数量渐增时的离散信道序列,其中所有这些信道都是从高斯信道推导而来,且证明这些信道的最大互信息趋向所宣称的C。将离散信道的噪声信道编码定理应用于每个导出信道中,这样可以得到连续信道的一个编码定理。或者,可以从直觉上对专门用于高斯信道的编码定理进行论证。

噪声信道编码定理的几何观点:填球

考虑输入的一个序列$\boldsymbol{x}=(x_1,\cdots,x_N)$和相应的输出$\boldsymbol{y}$,定义了一个$N$维空间中的两个点。对于大的$N$,噪声功率很可能(部分地)非常接近$N\sigma^2$。因此输出$\boldsymbol{y}$非常接近中心位于$\boldsymbol{x}$、半径为$\sqrt{N\sigma^2}$的一个球面。类似地,如果原信号$\boldsymbol{x}$是在满足平均功率约束$\overline{x^2}=v$的条件下随机产生的,则$\boldsymbol{x}$很可能靠近一个中心位于原点、半径为$\sqrt{Nv}$的球面;并且由于$\boldsymbol{y}$的总平均功率是$v+\sigma^2$,接收信号$\boldsymbol{y}$很可能靠近中心位于原点、半径为$\sqrt{N(v+\sigma^2)}$的一个球面。

半径为r的N维球的体积为

$$V(r,N)=\frac{\pi^{N/2}}{\Gamma(N/2+1)}r^N \tag{11.31}$$

现在考虑基于不易混淆的输入$\boldsymbol{x}$所构建的一个通信系统,即,输入的球彼此不会明显地重叠。不易混淆的输入的最大数量S由如下方法给出:用可能的$\boldsymbol{y}$球体积除以已知$\boldsymbol{x}$时的$\boldsymbol{y}$球体积:

$$S\leqslant\left(\frac{\sqrt{N(v+\sigma^2)}}{\sqrt{N\sigma^2}}\right)^N \tag{11.32}$$

因此容量被界定为

$$C = \frac{1}{N}\log M \leqslant \frac{1}{2}\log\left(1 + \frac{v}{\sigma^2}\right) \tag{11.33}$$

使用类似上一章节中所用的那个更详细的论证，可以建立等式关系。

回到连续信道

回忆一下，使用一个带宽为 W、噪声谱密度为 N_0 且功率为 P 的实际连续信道，等效于每秒使用 $N/T = 2W$ 次 $\sigma^2 = N_0/2$ 并满足约束条件 $\overline{x_n^2} \leqslant P/2W$ 的高斯信道。将结果代入高斯信道的容量中，可以发现连续信道的容量为

$$C = W\log\left(1 + \frac{P}{N_0 W}\right) \quad \text{比特/秒} \tag{11.34}$$

这个公式深刻地说明实际通信中所存在的权衡问题。假设有一个固定的功率限制。利用那个功率的最佳带宽是什么？引入 $W_0 = P/N_0$，即，信噪比是 1 的带宽，图 11.5 表示 $C/W_0 = W/W_0\log(1 + W_0/W)$ 与 W/W_0 之间的函数关系。容量增加到 $W_0\log e$ 的渐近线。在一个大带宽上以一个低信噪比传输，比在一个窄带宽上以高的信噪比传输要好得多（从固定功率的容量这个角度来说）；这是在 3G 移动电话中使用“直接序列扩频”方法之类的宽带通信方法的原因之一。当然，你并不是孤立的，而且如果使用一个大的带宽，你的“电磁邻居”也许会不高兴，所以出于“社交”原因，工程师通常使用高功率、窄带宽的发射机。

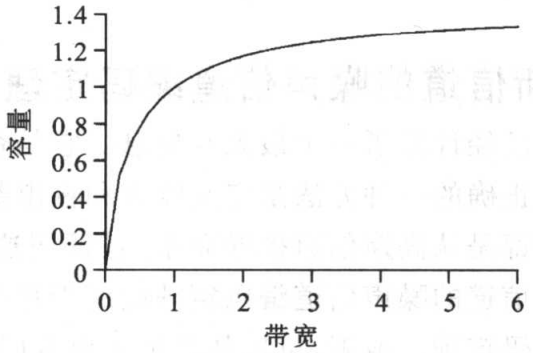


图 11.5 一个实际信道的容量与带宽之间的关系曲线：$C/W_0 = W/W_0\log(1 + W_0/W)$ 以 W/W_0 的函数表示。

11.4 实际纠错码的容量是什么？

几乎所有码都是好的，但是几乎所有码在实际编码器和译码器实现时都需要指数次地查表—它是分组长度 N 的指数。而且编码定理要求 N 足够大。

当人们说一个纠错码是*实用的*，意思是一个码在合理的时间内可以被编码和译码，例如其时间是一个分组长度 N 的多项式函数——最好是线性的。

香农极限在实际中是达不到的

噪声信道编码定理的非构造性证明表明，任何噪声信道都有好的分组码，而且事实上几乎所有分组码都是好的。但是写出一个显式的、实用的编码器和译码器且像

香农所许诺的那样好,则依然是一个未解决的问题。

非常好的码 已知一个信道,以低于信道容量的任意速率通信都可以获得任意小的差错概率的一族分组码被称为那个信道"非常好"的码。

好码 是以低于某最大值的非零速率通信可获得任意小的差错概率的一族码,其中最大通信速率比已知信道的容量要小。

坏码 是不能达到任意小的差错概率的一族码,或只有将信息速率降为零才能获得任意小的差错概率的码。循环码是坏码族的一个例子。(从实用的角度来说,坏码未必是无用的。)

实用码 是编码和译码工作在时间和空间上的开销为分组长度的多项式的一族码。

大多数已经确定的码是线性码

现在回顾一下分组码的定义,然后再增加线性分组码的定义。

信道 Q 的**一个** (N,K) **分组码**是一列 $S=2^K$ 的码字 $\{\boldsymbol{x}^{(1)},\boldsymbol{x}^{(2)},\cdots,\boldsymbol{x}^{(2^K)}\}$,每个长度为 N:$\boldsymbol{x}^{(s)}\in\mathcal{A}_X^N$。被编码的信号 s 来自大小为 2^K 的符号表,并被编码为 $\boldsymbol{x}^{(s)}$。

一个线性 (N,K) **分组码**是一个分组码,其中的码字 $\{\boldsymbol{x}^{(s)}\}$ 构成 $\mathcal{A}_X^N$ 的一个 K 维子空间。编码操作可以用一个 $N\times K$ 的二进制矩阵 $\boldsymbol{G}^{\mathrm{T}}$ 来表示,使得如果被编码的信号要使用二进制记号 s(一个长度为 K 比特的向量),则编码后的信号是 $\boldsymbol{t}=\boldsymbol{G}^{\mathrm{T}}\boldsymbol{s}$ 模 2。

码字 $\{\boldsymbol{t}\}$ 可被定义为满足 $\boldsymbol{Ht}=\boldsymbol{0}$ 模 2 的一组向量,其中 $\boldsymbol{H}$ 是码的奇偶校验矩阵。

例如,第 1.2 节中的(7,4)汉明码使用 $K=4$ 个信息比特 $\boldsymbol{s}$,再在它们后面接着传输 3 个奇偶校验比特。$N=7$ 的被传输符号由 $\boldsymbol{G}^{\mathrm{T}}\boldsymbol{s}$ 模 2 给出。

$$\boldsymbol{G}^{\mathrm{T}}=\begin{bmatrix}1&\cdot&\cdot&\cdot\\ \cdot&1&\cdot&\cdot\\ \cdot&\cdot&1&\cdot\\ \cdot&\cdot&\cdot&1\\ 1&1&1&\cdot\\ \cdot&1&1&1\\ 1&\cdot&1&1\end{bmatrix}$$

汉明的工作开创了编码理论。他发明了一族实用的纠错码,其中每一个都可以在分组长度为 N 的码中纠正一个错误,这其中的循环码 R_3 和(7,4)汉明码最为简单。在此之后确定的码大多是汉明码的推广,聊举数例:Bose - Chaudhury - Hocquenhem 码、Reed - Müller 码、Reed - Solomon 码和 Goppa 码。

卷积码

另一族线性码是卷积码,它并不是将信源流分成块,而是连续地读取和传输比特。被传输的比特是过去信源比特的线性函数。通常生成被传输比特的规则包括将

当前的信源比特送入一个长度为 k 的线性反馈移位寄存器，并在每次迭代时传输一个或多个移位寄存器状态的线性函数。最终要传输的比特流是信源流和一个线性滤波器的卷积。这个滤波器的脉冲响应函数可以是有限或无限时间的，这取决于反馈移位寄存器的选择。

将在第 48 章讨论卷积码。

线性码“好”吗？

也许有人会问，香农极限在实际中不可达的原因是否是因为线性码本身不如随机码好？答案是否定的，对于线性码而言，仍然可以证明噪声信道编码定理，至少在某些信道上如此（参见第 14 章），尽管如香农证明随机码一样，证明也是非构造性的。

线性码在编码端很容易完成。一个线性码的译码是否也同样容易？未必。一般的译码问题（求方程 $\boldsymbol{G}^{\mathrm{T}}\boldsymbol{s}+\boldsymbol{n}=\boldsymbol{r}$ 的最大似然解 $\boldsymbol{s}$）实际上是 NP - 完全问题（Berlekamp 等，1978 年）。[NP - 完全问题是计算问题，它们都具有同等难度并被广泛地认为一般需要指数级的计算机时间来求解。]所以注意力集中在具有快速译码算法的一类码上。

级联

用实用的译码器来构造码的一种技巧是级联的思想。

$$\mathscr{C}' \rightarrow \underbrace{\mathscr{C} \rightarrow Q \rightarrow \mathscr{D}}_{Q'} \rightarrow \mathscr{D}'$$

一个编码器—信道—译码器系统 $\mathscr{C} \rightarrow Q \rightarrow \mathscr{D}$ 可被视做定义了一个差错概率更小的超级信道 Q'，并且在它的错误间存在复杂的相关性。可以为这个超级信道 Q' 创建一个编码器 $\mathscr{C}'$ 和一个译码器 $\mathscr{D}'$。这个码由外码 $\mathscr{C}'$ 和紧随其后的内码 $\mathscr{C}$ 构成，被称为*级联码*。

某些级联码运用了交织的思想。分组地读取数据，每个分组的大小可以比组成码 $\mathscr{C}$ 和 $\mathscr{C}'$ 的分组长度要大。使用码 $\mathscr{C}'$ 对一个分组数据进行编码后，重新对分组内的各比特进行排序，使得当分组被送到第 2 个码 $\mathscr{C}$ 时，相邻比特彼此间被分离开。交织的一个简单例子是一个*矩形码*或*乘积码*，它将数据排列为 $K_2 \times K_1$ 分组，并用一个 (N_1, K_1) 线性码水平地编码，然后在垂直方向使用一个 (N_2, K_2) 线性码来编码。

▷ **习题 11.3**[$\mathscr{A}$] 证明这两个码中的任意一个都可被视做内码或外码。

作为一个例子，图 11.6 所示为一个乘积码，首先用循环码 R_3（也叫 $H(3,1)$ 汉明码）水平地编码，然后用 $H(7,4)$ 垂直地编码。级联码的分组长度是 27。每个码字的信源比特数是 4，用小的矩形表示。

在某些序列中，可以为每个子码使用单独的译码器来方便地（尽管不是最优地）进行译码。最合理的做法是，首先对具有最低码率并因此有最强纠错能力的码进行

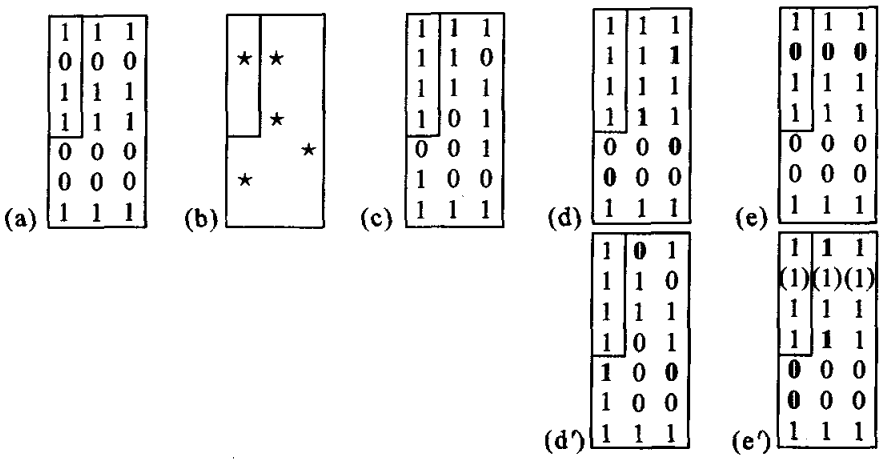


图 11.6 一个乘积码。(a) 用两个汉明码 $H(3,1)$ 和 $H(7,4)$ 所组成的级联码对字符串 1011 进行编码。(b) 使 5 比特翻转的一个噪声模式。(c) 接收向量。(d) 使用水平(3,1)译码器译码后的情况。(e) 随后使用垂直(7,4)译码器后的情况。译码后的向量与初始的相匹配。(d',e')用其他次序译码后,还剩下 3 个错误。

译码。

当接收到图 11.6(a)的码字有某些错误时(如图 11.6 所示,有 5 个比特被翻转),首先应用 $H(3,1)$ 的译码器,然后用 $H(7,4)$ 的译码器,所得结果如图 11.6(c ~ e)所示。第 1 个译码器纠正了 3 个错误,但是错误地改变了第 2 行的第 3 个比特,其中第 2 行中有 2 个比特错误。然后(7,4)译码器可以纠正所有 3 个错误。

图 11.6(d' ~ e')显示了如果采用其他次序来对两个码进行译码,会发生什么。在第 1 列和第 2 列有两个错误,所以(7,4)译码器引入了两个额外错误。它纠正了第 3 列的一个错误。(3,1)译码器将 4 个错误全部清除,但却错误地推断了第 2 个比特。

交织

交织的目的是将一个码中的相邻比特扩散开,使得人们能够忽略由内码所产生的错误之间的复杂相关性。也许内码会将整个码字都弄乱;但是这个码字被分散在几个外码码字中,每次 1 比特。所以可将内码所导致的错误视做独立的。

其他信道模型

除了二进制对称信道和高斯信道外,编码理论家也注意到了一些更复杂的信道。

*突发错误信道*在实际中是很重要的模型。Reed - Solomon 码使用包含大量元素(例如 2^{16})的 Galois 域(见附录 C.1)作为它的输入符号集,并且因此自动地获得一定程度的突发错误容忍度,因为即使在 17 个连续比特被破坏的情形下,在 Galois 域表示法中仅有两个连续符号被破坏。级联和交织可以为抵抗突发错误提供更进一步的保护。

在数字 CD 上所用的级联 Reed – Solomon 码可以纠正长度为 4 000 比特的突发错误。

▷ **习题 11.4**[2] 交织技术使人们可以将突发错误视做独立的,它已经得到广泛的应用,但是从需要冗余度的角度来看,这种对付突发错误以保护数据的方法在理论上却不算好方法。使用随后的突发错误信道作为例子,解释为什么交织不算好方法。时间被分为长度 $N=100$ 个时钟周期的块;在每块时间内,存在一个概率为 $b=0.2$ 的突发错误;在一个突发错误发生时,信道是一个 $f=0.5$ 的二进制对称信道。如果没有突发错误,信道是一个无误的二进制信道。计算这种信道的容量,与使用交织并将错误视做独立错误时可获得的最大通信速率进行比较。

*衰落信道*与高斯信道一样,是实际信道,但有一点不同:它假设接收功率是随时间变化的。不停变化的移动电话是这方面的一个重要例子。到来的无线信号被附近的物体所反射,所以存在干扰模式,且电话所接收到的信号强度随着它的位置而改变。当电话的天线移动一个与无线信号波长(几厘米)相当的距离时,接收功率可以轻易地变化 10 个分贝(一个大小为 10 的因子)。

11.5 最新技术水平

在高斯信道上进行通信的最为著名的码是什么? 所有实用码都是线性码,并且要么是基于卷积码,要么是基于分组码。

卷积码及基于其上的码

经典的卷积码 用于卫星通信纠错码的"事实标准"是一个约束长度为 7 的卷积码。卷积码将在第 48 章中讨论。

级联卷积码 上述卷积码可被用做一个级联码的内码,该级联码的外码是一个具有 8 比特符号的 Reed – Solomon 码。这种码被用在诸如航天宇宙飞船之类的太空通信系统中。要进一步学习关于 Reed – Solomon 码的内容,请参见 Lin 和 Costello(1983 年)。

Galileo 码 这是由喷气推进实验室(Jet Propulsion Laboratory, JPL)开发出来(Swanson, 1988 年)的一个码。它使用了相同格式,但是它用了一个具有更长约束长度 15 的卷积码和一个更大的 Reed – Solomon 码。这个码的详细资料没有发布到 JPL 之外,并且只能在一个装满专用硬件设备的房间里才能进行译码。在 1992 年,这是已知的码率为 1/4 的最佳码。

Turbo 码 在 1993 年,Berrou、Glavieux 和 Thitimajshima 发表了关于 Turbo 码的研究成果。一个 Turbo 码的编码器基于两个卷积码的编码器(如图 11.7 所示)。信源比特被送入每个编码器,并发送被随机排列的信源比特的顺序以及源自每个组成码的奇偶校验比特。

译码算法包括使用它的标准译码算法来迭代地对每个组成码进行译码,然后将译码器的输出作为另一个译码器的输入。这种译码算法叫做*和 – 积算法*,它是消息

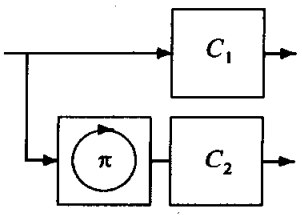


图 11.7 一个 Turbo 码的编码器。C_1 和 C_2 的每个方块里包含一个卷积码。在信源比特被送入 C_2 之前，用一个排列 π 进行重新排序。通过对两个卷积码的输出进行级联或交织，可获得被传输的码字。在设计码时选择随机排列，并从此将此排列次序固定下来。

传递算法的一个实例。

Turbo 码将在第 48 章中讨论，消息传递将在第 16 章、第 17 章、第 25 章和第 26 章中讨论。

分组码

Gallager 低密度奇偶校验码（如图 11.8 所示）。高斯信道上的最优分组码由 Gallager 在 1962 年发明，但是很快就被编码理论界的大多数人遗忘了。它们在 1995 年被重新发现，并且被证明具有优异的理论和实用特性。类似 Turbo 码，它们也是通过消息传递算法来译码的。

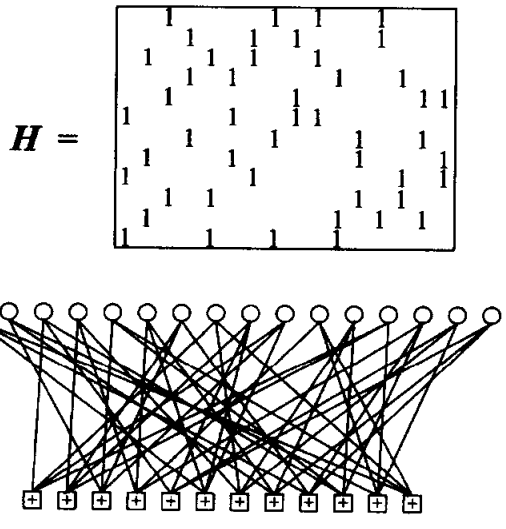


图 11.8 一个低密度奇偶校验矩阵和与之对应的码率为$\frac{1}{4}$的低密度奇偶校验码的图，其分组长度为 $N=16$，且有 $M=12$ 个约束。每个白圈代表一个传输比特。每个比特受 $j=3$ 个约束，用田方块表示。每个约束迫使与它相连的 $k=4$ 个比特的和为偶数。这个码是(16,4)码。当分组长度增加至 $N\simeq 10\,000$ 时，可获得优异的性能。

我们将在第47章讨论这些简单优美的码。

上述码在高斯信道上的性能比较如图47.17所示。

11.6 小 结

随机码是好码,但是它们需要指数级的资源来进行编码和译码。

非随机码在很多方面不如随机码那么好。对于一个非随机码,编码也许很简单,但是即使对于定义简单的线性码,译码问题依然很难解决。

最好的实用码(a) 使用很大的分组长度;(b) 基于半-随机码结构;(c) 使用基于概率的译码算法。

11.7 非线性码

大多数实际使用的码是线性的,但并非全都如此。数字声道以二进制形式被编码在电影胶片上。影响胶片的可能错误包括污迹和划痕,它们分别产生大量的1和0。我们希望没有哪个码字看上去全部是1或全部是0,以便使污迹和划痕所产生的错误很容易被检测出来。用在数字电影音响系统的一个码是非线性(8,6)码,它包括64个权重为4的$\binom{8}{4}$二进制模式。

11.8 除了噪声以外的错误

接收方的另一个不可靠性来自被传输信号$x(t)$的定时不确定性。在普通的编码理论和信息论中,假设发射方的时间t和接收方的时间u完全同步。但是如果接收方接收了一个信号$y(u)$,其中接收方的时间u是关于发射方时间t的一个不完全已知的函数$u(t)$,那么这个信道的通信容量就减小了。与迄今为止已经讨论过的同步化信道相比,关于这种信道的理论尚不完整。具有同步误差的信道的容量尚属未知(Levenshtein,1966年;Ferreira等,1997年);为了在具有同步误差的信道上可靠通信所进行的编码仍然是一个活跃的研究领域(Davey和MacKay,2001年)。

进一步的阅读

有关扩频方法的历史性回顾,请参见Scholtz(1982年)。

11.9 习 题

高斯信道

▷**习题 11.5**[2] 考虑一个有实输入 x 和信噪比 v/σ^2 的高斯信道。

(a) 容量 C 是多少?

(b) 如果输入被限定为二进制的,$x \in \{\pm\sqrt{v}\}$,这个受限信道的容量 C' 是多少?

(c) 如果除此之外,还要求信道的输出使用如下映射来阈值化

$$y \to y' = \begin{cases} 1, & y > 0 \\ 0, & y \leqslant 0 \end{cases} \tag{11.35}$$

所得信道的容量 C'' 是多少?

(d) 绘制以上 3 个容量与 v/σ^2 的函数关系从 0.1 到 2 范围之内的曲线。[要使用数字积分来计算 C'。]

▷**习题 11.6**[2] 对于大的整数 K 和 N,长度为 N 且码率为 $R = K/N$ 的所有二进制纠错码中有多大部分是线性码?[答案将取决于你是否选择将码定义为 2^K 个码字的一个有序排列,即,从 $s \in \{1,2,\cdots,2^K\}$ 到 $\boldsymbol{x}^{(s)}$ 的一个映射,还是将码定义为一个无序序列,使得由相同码字所组成的两个码是相同的。使用后一种定义:一个码是一组码字的集合,编码器如何工作并不是码的定义的一部分。]

删除信道

▷**习题 11.7**[2] 为二进制删除信道设计一个码及译码算法,并计算其差错概率。[为删除信道设计一个好码是一个活跃的研究领域(Spielman,1996 年;Byers 等,1998 年);也可参见第 50 章。]

▷**习题 11.8**[2] 为 q 元删除信道设计一个码,它的输入 x 是从 $0,1,2,3,\cdots,(q-1)$ 选取,而它的输出 y 以概率 $(1-f)$ 等于 x,否则等于?。[这个删除信道为在因特网上传输的分组提供了一个好模型,在网上分组要么被可靠地接收,要么被丢弃。]

习题 11.9[2] 独立磁盘冗余阵列(RAID)①是如何工作的?大约 10 个磁盘驱动器组成一个信息存储系统,当其中任意两个或 3 个不工作,而其他都工作时,还是可以重建任意被请求的文件。对于要求解的问题,它们使用了什么码,这些系统距离香农极限有多远?你应如何设计一个更好的 RAID 系统?在习题解答部分提供了一些信息。参见 http://www.acnc.com/raid2.html;也可参见第 50 章。

① 有人说 RAID 表示"便宜磁盘的冗余阵列",但是笔者认为这个观点是十分荒谬的——即使磁盘很昂贵,RAID 仍然是一个好主意。

11.10 习题解答

习题 11.1 的解答 为功率约束引入一个拉格朗日因子 λ，并为 $P(x)$ 的归一化约束引入另一个因子 μ。

$$F = I(X;Y) - \lambda \int \mathrm{d}x P(x) x^2 - \mu \int \mathrm{d}x P(x) \tag{11.36}$$

$$= \int \mathrm{d}x P(x) \left[\int \mathrm{d}y P(y|x) \ln \frac{P(y|x)}{P(y)} - \lambda x^2 - \mu \right] \tag{11.37}$$

对 $P(x^*)$ 进行函数求导。

$$\frac{\delta F}{\delta P(x^*)} = \int \mathrm{d}y P(y|x^*) \ln \frac{P(y|x^*)}{P(y)} - \lambda x^{*2} - \mu - \int \mathrm{d}x P(x) \int \mathrm{d}y P(y|x) \frac{1}{P(y)} \frac{\delta P(y)}{\delta P(x^*)} \tag{11.38}$$

利用 $P(y) = \int \mathrm{d}x P(x) P(y|x)$，可求得最后的因子 $\delta P(y)/\delta P(x^*)$ 为 $P(y|x^*)$，于是整个最后一项立刻变为 1，且可将它并入 μ 的那一项。

代入 $P(y|x) = \exp(-(y-x)^2/2\sigma^2)/\sqrt{2\pi\sigma^2}$，并设导数为零：

$$\int \mathrm{d}y P(y|x) \ln \frac{P(y|x)}{P(y)} - \lambda x^2 - \mu' = 0 \tag{11.39}$$

$$\Rightarrow \int \mathrm{d}y \frac{\exp(-(y-x)^2/2\sigma^2)}{\sqrt{2\pi\sigma^2}} \ln [P(y)\sigma] = -\lambda x^2 - \mu' - \frac{1}{2} \tag{11.40}$$

对于所有 x，$\ln[P(y)\sigma]$ 必须满足这个条件。

写出 $\ln[P(y)\sigma] = a + by + cy^2 + \cdots$ 的泰勒级数展开式，仅有二次函数 $\ln[P(y)\sigma] = a + cy^2$ 满足约束条件(11.40)。(任意高阶项 y^p，$p > 2$ 将产生一个 x^p 项，而右边并未出现这一项。)因此 $P(y)$ 是高斯的。可通过使用一个高斯输入分布 $P(x)$ 来获得这个最优输出分布。

习题 11.2 的解答 已知一个服从方差为 v 的高斯输入分布，由于 x 和噪声是独立随机变量，而独立随机变量的方差相加，于是输出分布是 Normal$(0, v+\sigma^2)$ 的高斯分布。互信息为

$$I(X;Y) = \int \mathrm{d}x \mathrm{d}y P(x) P(y|x) \log P(y|x) - \int \mathrm{d}y P(y) \log P(y) \tag{11.41}$$

$$= \frac{1}{2} \log \frac{1}{\sigma^2} - \frac{1}{2} \log \frac{1}{v+\sigma^2} \tag{11.42}$$

$$= \frac{1}{2} \log \left(1 + \frac{v}{\sigma^2} \right) \tag{11.43}$$

习题 11.4 的解答 信道的容量是 1 减去它加入的噪声所产生的信息量。每块

的信息量是，关于是否为突发块的选择的熵 $H_2(b)$，加上概率为 b 时被翻转的 N 个比特的熵，加起来每块的熵为 $H_2(b)+Nb$（这是粗略的解；如果 N 很大就比较精确）。所以，对于 $N=100$，每个比特的容量是

$$C=1-\left(\frac{1}{N}H_2(b)+b\right)=1-0.207=0.793 \tag{11.44}$$

与之相对，交织将突发错误视做独立的，使得信道可作为 $f=0.2\times0.5=0.1$ 的二进制对称信道来处理，其容量约为 0.53。

交织将关于差错相关性的有用信息也丢弃了。从理论上说，如果使用一个码和译码器明确地将突发错误作为突发错误对待，应该可以将通信速率加快约 $(0.79/0.53)\simeq1.6$ 倍。

习题 11.5 的解答

（a）将习题 11.1 和习题 11.2 的结果放在一起，推导出一个具有实输入 x 和信噪比 v/σ^2 的高斯信道的容量为

$$C=\frac{1}{2}\log\left(1+\frac{v}{\sigma^2}\right) \tag{11.45}$$

（b）如果输入被限定为二进制，$x\in\{\pm\sqrt{v}\}$，以相等的概率使用这两个输入能够达到容量。这个容量降低了，且可以用下面这个有点混乱的积分表示

$$C''=\int_{-\infty}^{\infty}\mathrm{d}yN(y;0)\log N(y;0)-\int_{-\infty}^{\infty}\mathrm{d}yP(y)\log P(y) \tag{11.46}$$

其中，$N(y;x)\equiv(1/\sqrt{2\pi})\exp[(y-x)^2/2]$，$x\equiv\sqrt{v}/\sigma$，$P(y)\equiv[N(y;x)+N(y;-x)]/2$。这个容量比没有受限制的容量(11.45)要小，但是对于小的信噪比，两个容量在数值上很接近。

（c）如果输出被阈值化，则高斯信道变成转移概率由第 9.8 节定义的误差函数 Φ 所给出的一个二进制对称信道。其容量为

$$C''=1-H_2(f),\quad f=\Phi(\sqrt{v}/\sigma) \tag{11.47}$$

相应图形如图 11.9 所示。

习题 11.9 的解答 存在几种 RAID 系统。其中最容易理解的一种由 7 个磁盘驱动器构成，它们用一个(7,4)汉明码以码率 4/7 存储数据：用这个码对每个连续的 4 比特进行编码，并将 7 个码字比特分别写到其中一个磁盘上。当两个也许还是 3 个磁盘驱动器停止工作，其他磁盘驱动器还是可以恢复数据。由于假设已经知道一个磁盘何时不工作，所以这里的有效信道模型是一个二进制删除信道。

选择某 3 个不工作的磁盘时，将不可能恢复数据；你能说出为什么吗？

▷ **习题 11.10**[p] 举出如果 3 个磁盘驱动器损坏将导致上述 RAID 系统失效的例子，和 3 个磁盘驱动器损坏却不会导致上述 RAID 系统失效的例子。

习题 11.10 的解答 (7,4)汉明码包含权重为 3 的码字。如果对应这些码字的任意一组 3 个磁盘驱动器被损坏，则另外 4 个磁盘仅仅可以恢复 4 个信源比特中的 3 比特信息；第 4 个比特会丢失。[比较习题 13.13，其中 $q=2$：没有二进制 MDS 码。

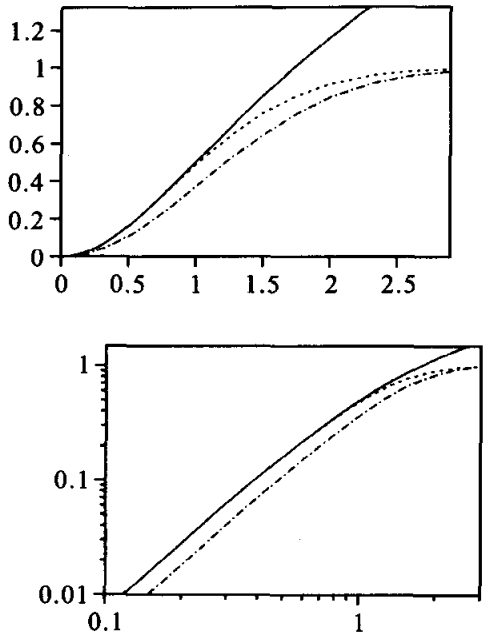


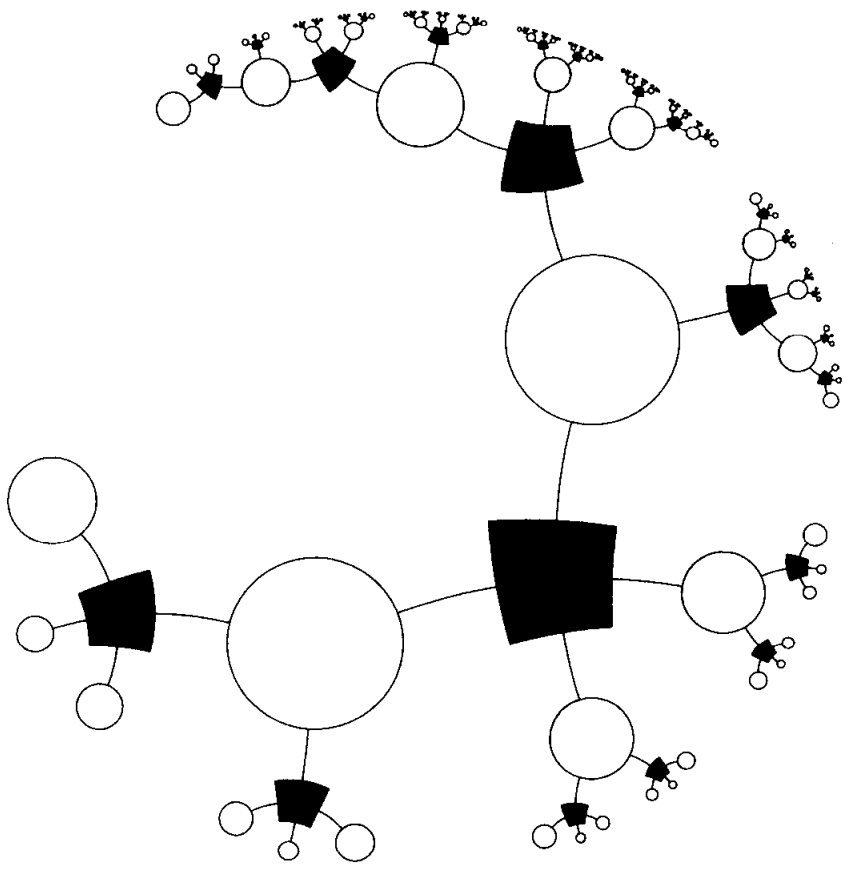
图 11.9 容量(每个图中从上到下依次为)C、C'和 C''与信噪比($\sqrt{v}/\sigma$)之间的关系,下方的图是用对数坐标画出的。

这个不足之处将在第 13.11 节详细讨论。]

其他任意组的 3 个磁盘驱动器损坏则不会有问题,因为生成矩阵所对应的 4×4 子矩阵是可逆的。一个更好的码是数字喷泉码——参见第 50 章。

第三部分

信息论中的更多专题



关于第 12 章

在第 1 章～第 11 章，集中讨论了信息理论和编码理论中的两方面内容：信源编码——信息压缩，以便充分利用数据传输和存储信道；信道编码——信息的冗余编码，以便能够检测和纠正通信错误。

在这两个领域中，先忽略实际需要考虑的因素，集中精力考虑理论极限和编码的可能性。然后，讨论实际的信源编码和信道编码方案，将重点转移到计算的可行性上。但是，比较编码方案的最主要评判标准仍然是用所要求的信道资源表示编码的效率：最好的信源编码能实现最大的压缩；最好的信道编码在一个已知差错概率下能以最高速率进行通信。

本章将稍许改变视角，转而将如何更容易地检索信息作为首要目标。结果证明，在研究信道编码时很有理论价值的随机码，对于快速信息检索也很有用。

有效的信息检索似乎是大脑可以毫不费力解决的问题之一；内容可寻址的存储是我们考虑神经网络时所要研究的主题之一。

第 12 章

散列码:用于有效信息检索的码

12.1 信息检索问题

信息检索问题的一个简单例子是实现电话号码簿服务的任务。在这项任务中,收到一个人的姓名时,回应(a) 那个人是否被列在目录中的确认信息;(b) 这个人的电话号码和其他详细情况。可以对这个问题进行严格的数学描述如下。令 S 为必须存储在电话号码簿中的姓名的个数。

给定一列长度为 N 比特的 S 个二进制字符串,即 $\{\boldsymbol{x}^{(1)},\cdots,\boldsymbol{x}^{(s)}\}$,其中 S 远小于可能字符串的总数 2^N。称 $\boldsymbol{x}^{(s)}$ 中的上标"s"为字符串的记录号。基本思想是按照用户加入电话号码簿的顺序用 s 快速地查寻客户,$\boldsymbol{x}^{(s)}$ 是客户 s 的姓名。为了简单起见,假设所有人具有相同长度的姓名。姓名的长度可以是,比如,$N=200$ 比特,并且可能希望保存一千万个客户的详细信息,于是 $S\simeq 10^7\simeq 2^{23}$(如图 12.1 所示)。我们将忽略两个客户有相同姓名的可能情况。

字符串长度	$N\simeq 200$
字符串的数量	$S\simeq 2^{23}$
可能字符串的数量	$2^N\simeq 2^{200}$

图 12.1 记号表

任务是构造从 s 到 $\boldsymbol{x}^{(s)}$ 的逆映射,即构造这样一个系统,对于给定的一个字符串 $\boldsymbol{x}$,如果存在 s 值使得 $\boldsymbol{x}=\boldsymbol{x}^{(s)}$,则系统返回 s 值,否则报告不存在这样的 s。(一旦得到记录号,就可以访问另辟为存放全部电话号码的内存区,在存储单元 s 中找到所需要的电话号码)。解决这个任务时,目的是使用最少的计算资源,即用来存储从 $\boldsymbol{x}$ 到 s 的逆映射所需的内存大小和计算这个逆映射所需的时间。而且还希望逆映射最好可以这样实现:使用少量的计算机时间就可将新的字符串加入电话号码簿中。

一些标准的解决方案

信息检索问题的最简单和最笨拙的解决方案是查找表和原始列表。

查找表是一块大小为 $2^N\log_2 S$ 的内存区,$\log_2 S$ 是存储 $1\sim S$ 之间的一个整数所需的内存大小。对于 2^N 存储单元中的每一个,都将写入零,只有在存储单元 $\boldsymbol{x}$ 与字符

串 $x^{(s)}$ 相对应时,才会写入 s 的值。

只有具备足够的空间存储表格,并且在存储器中查找条目的成本与存储器的大小无关,查找表才不失为一种简单、快速的解决方案。但是在任务定义中,假设 N 约为 200 比特或者更大,所以所要求的存储器容量大小会是 2^{200};这种方案是完全不可能实现的。请记住,太阳系中粒子的数目也仅仅约为 2^{190} 个。

原始列表是根据 s 值排列的一列简单有序对 $(s, x^{(s)})$。从 x 到 s 的映射可以这样得到:从顶部开始搜索字符串列表,并将输入的字符串 x 与每条记录 $x^{(s)}$ 进行比较,直至找到一个匹配。这个系统很容易维护,并且所用的存储空间小,约为 SN 比特,但是由于平均需要做 500 万次成对的比较,所以使用起来相当慢。

▷ **习题 12.1**[2] 假设最初的姓名是随机选择的,证明在一个原始列表中寻找所要求的字符串平均花费时间约为 $S+N$ 次二进制比较。(注意:你无需比较长度为 N 的整个字符串,因为只要发生错配,就可以中止比较行动;证明每次不正确的字符串匹配平均需要两次二进制比较。)将此结果与最坏情况的搜索时间相比较——假设魔鬼选择字符串的集合和搜索键。

创建电话号码簿的标准方法使用*按照字母顺序排列的列表*来改进查找表和原始列表。

按照字母顺序排列的列表 字符串 $\{x^{(s)}\}$ 是按照字母顺序排列的。由于我们利用了分类法的长处,所以现在查找一个条目所需的时间通常小于在原始列表中所需的时间;例如,可以从电话号码簿的中间页打开,并将在那里找到的姓名同目标字符串进行比较;如果目标字符串"大于"中间的字符串,则知道所要求的字符串,如果它存在,将在按字母顺序排列的电话号码簿的后半部分被找到。否则,就查找前半部分。以这种在中间分开的过程进行迭代,可以在 $\lceil \log_2 S \rceil$ 次字符串比较之内,找到目标字符串,或确定字符串不在列表中。每个字符串比较中的二进制比较的期望次数随着搜索的进行而倾向于增加,但是所需要的二进制比较总次数不会大于 $\lceil \log_2 S \rceil N$。

该方法所需要的内存大小与原始列表的相同。

在数据库中增加新的字符串需要我们将其插入列表中的正确位置。为了找到正确位置,需要进行 $\lceil \log_2 S \rceil$ 次二进制比较。

是否可以改进业已形成的按字母顺序排列的列表?从一些新的视角来考虑我们的任务。

这个任务就是构造从 N 比特到 $\log_2 S$ 比特的一个映射 $x \rightarrow s$。这是一个伪可逆映射,因为对于映射到非零 s 的任意 x,用户数据库都会包含数据对 $(s, x^{(s)})$,允许我们映射回去。在这之前,在哪里碰到过从 N 比特映射到 M 比特的思想呢?

曾经两次遇到这种思想:第一次,在信源编码中,研究分组码,它是从 N 个符号所组成的字符串到列表中一次标记选择的映射。信息检索的任务与构造典型集压缩码的编码器的任务(这是从未真正解决的问题)相似。

第二次将比特串映射到其他维数的比特串是在研究信道编码时。那里,考虑将 K 比特映射到 N 比特的码,其中 N 大于 K,通过使用随机码获得了理论上的进展。

在散列码中,将这两个观点结合在一起,将研究从 N 比特到 M 比特映射的随机码,其中 M 小于 N。

思路是这样的,将原来的高维空间向下映射到一个低维空间;在低维空间中,刚才否定的愚笨查表方法却是可行的。修正后的记号表如图 12.2 所示。

字符串长度	$N \simeq 200$
字符串的数量	$S \simeq 2^{23}$
散列函数的大小	$M \simeq 30$ 比特
散列表的大小	$T = 2^M \simeq 2^{30}$

图 12.2 修正后的记号表

12.2 散 列 码

首先,将描述散列码是如何工作的,然后研究理想化散列码的性质。散列码为信息检索问题提供了一种解决方案,即借助于一个被称为*散列函数*的伪随机函数,提供从 $\boldsymbol{x}$ 到 s 的一个映射。散列函数将 N 比特的字符串 $\boldsymbol{x}$ 映射为 M 比特的字符串 $\boldsymbol{h}(\boldsymbol{x})$,其中 M 小于 N。通常选择 M 使得"表格大小" $T \simeq 2^M$ 略大于 S,比如大 10 倍。例如,如果希望 S 大约为 100 万,则需要将 $\boldsymbol{x}$ 映射成 30 比特的散列 $\boldsymbol{h}$(不管每项 $\boldsymbol{x}$ 的大小 N 是多少)。散列函数是一个固定的确定性函数,它在理想情况下与一个固定的随机码应不可区分。为了实用起见,散列函数必须可以快速计算出来。

散列函数的两个简单例子如下。

除法方法 表格大小 T 是一个素数,最好是一个不接近 2 的幂的数。散列值是整数 x 被 T 除时的余数。

可变字符串添加方法 这种方法假设 $\boldsymbol{x}$ 是一个由字节所组成的字符串,并且表格大小 T 为 256。对 $\boldsymbol{x}$ 的字符做模 256 的加法。该散列函数有一个缺点,即它将彼此以颠倒字母顺序所构成的字符串映射为相同的散列。

该方法可以通过如下方法进行改进:在加上每个字符之后,将运行总数置于一个固定的伪随机排列上。在表格大小≤65 536 的*可变字符串异或方法*中,用这种方法将字符串散列两次,初始运行总数分别被设为 0 和 1(算法 12.3)。结果是一个 16 比特散列。

```
unsigned char Rand8[256];          //该数组包含了从 0..255 到 0..255 的一个随机排列
int Hash(char * x){                //x 是指向第一个字符的指针
    int h;                         // * x 是第一个字符
    unsigned char h1,h2;
    if( * x = =0)return 0;         //空字符串的特殊处理
    h1 = * x;h2 = * x +1;          //初始化两个散列
```

```
    x + +;                                  //进行到下一个字符
    while( * x) {
        h1 = Rand8[h1^ * x];                //两个散列相异或
                                            //并让它们通过随机发生器
        h2 = Rand8[h2^ * x];
        x + +;
    }                                       //当 * x = 0 时,到达了字符串尾部
    h = ((int)(h1) <<8) |                   //把 h1 左移 8 比特并加上 h2
        (int)h2;
    return h;                               //散列是 h1 和 h2 的级联
}
```

算法 12.3　可变字符串异或方法的 C 代码实现,它从一个字符串 x 中创建一个范围在 0 ~ 65 535 的散列 h。作者:Thomas Niemann。

已经选择了一个散列函数 $h(x)$,采用如下方法实现信息检索(如图 12.4 所示)。

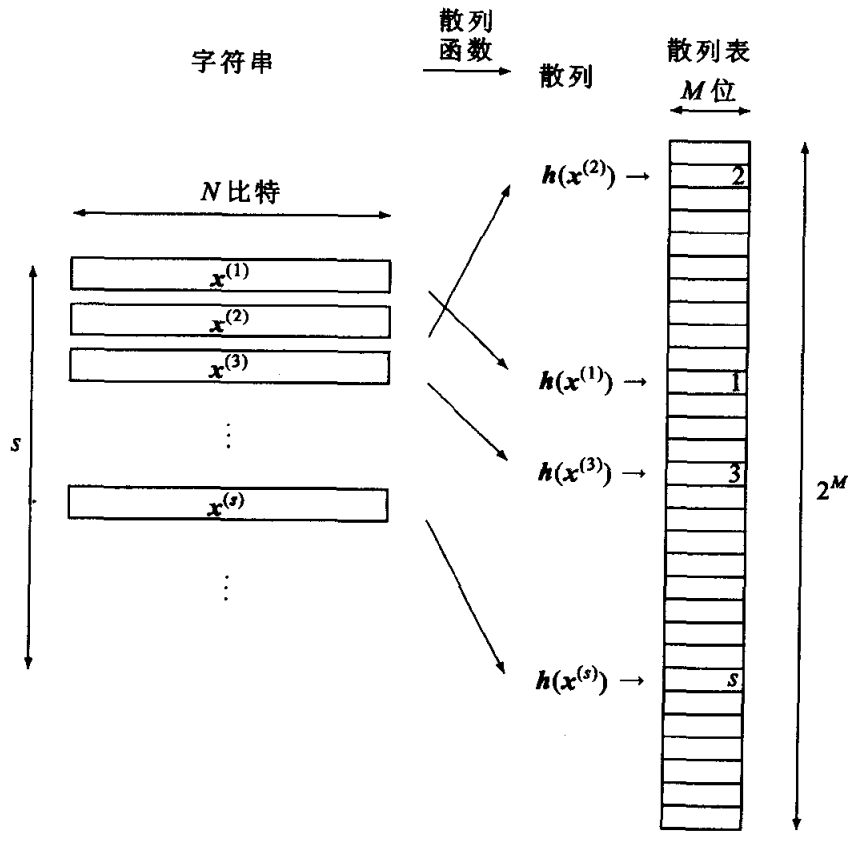


图 12.4　用散列函数进行信息检索。对于每个字符串 $x^{(s)}$,计算散列 $h = h(x^{(s)})$,并将 s 的值写入散列表的第 h 行。散列表中空行的值为零。表的大小为 $T = 2^M$。

编码　创建散列表,它是由 $2^M b$ 个存储单元所构成的一段内存,其中 b 是表示从 0 到 S 之间的整数所需的存储容量。开始时,这个表格全部置为零。将每个存储器

$\boldsymbol{x}^{(s)}$输入散列函数，并在所得向量 $\boldsymbol{h}^{(s)} = \boldsymbol{h}(\boldsymbol{x}^{(s)})$对应的散列表存储单元写入整数 s——除非散列表中那一项已经被占用，在这种情况下，$\boldsymbol{x}^{(s)}$与早先出现过的 $\boldsymbol{x}^{(s')}$发生了碰撞，因为它们恰好有相同的散列码。碰撞可以采用不同的方法来处理——稍后将讨论一些方法——但是首先应完成基本图景的描绘。

译码 为了检索与一个目标向量 $\boldsymbol{x}$ 相对应的一段信息，计算 $\boldsymbol{x}$ 的散列 $\boldsymbol{h}$，并在散列表中相应的存储单元中查找。如果结果为 0，立即可以知道字符串 $\boldsymbol{x}$ 不在数据库中。得出这个答案的代价是一次散列函数计算和在大小为 2^M 的表中的一次查找。另一方面，如果在表中有非零项 s，则存在两种可能性：或者向量 $\boldsymbol{x}$ 确实等于 $\boldsymbol{x}^{(s)}$；或者向量 $\boldsymbol{x}^{(s)}$是恰好与目标 $\boldsymbol{x}$ 具有相同散列码的另一个向量。（第三种可能性是这个非零项也许与将要讨论的冲突解决系统有关。）

为了检查 $\boldsymbol{x}$ 是否确实等于 $\boldsymbol{x}^{(s)}$，取试探性的答案 s，在原前向数据库中查找 $\boldsymbol{x}^{(s)}$，并将它与 $\boldsymbol{x}$ 逐比特进行比较；如果相匹配，则报告说 s 是想得到的答案。这个成功的检索所付出的全部代价包括一次散列函数计算，在大小为 2^M 表中的一次查找，在大小为 S 表中的另一次查找，及 N 次二进制比较——这种方法的开销比第 12.1 节介绍的简单方法更小。

▷ **习题 12.2**[⇗] 如果用 $\boldsymbol{x}$ 检查 $\boldsymbol{x}^{(s)}$的前几个比特，并发现它们是相等的，那么正确项被检索出来的概率是多少？如果另一个假设是 $\boldsymbol{x}$ 实际上不在数据库中，则上述问题的概率是多少？假设原信源字符串是随机的，而且散列函数也是一个随机散列函数。如果以 10 亿比一的几率来检索正确项，需要多少次二进制计算？

如果散列函数 $\boldsymbol{h}(\boldsymbol{x})$可被用于任意长度的字符串，则信息检索中的散列方法也适用于任意长度的字符串 $\boldsymbol{x}$。

12.3 冲突解决

我们将研究两种解决冲突的方法：在表后添加，在其他地方存储。

在表后添加

在编码时，如果出现了一次冲突，那么将继续往散列表下面走，并将 s 值写入内存中的下一个有效存储单元，它目前包含一个零。如果在搜索到一个零之前已到达表底，就继续从表头开始搜索。

在译码时，如果计算 $\boldsymbol{x}$ 的散列码，并发现表中所包含的 s 并未指向与 $\boldsymbol{x}$ 匹配的一个 $\boldsymbol{x}^{(s)}$，则继续搜索散列表直至我们或者得到一个 s，使得它的 $\boldsymbol{x}^{(s)}$确实与 $\boldsymbol{x}$ 相匹配，这种情况下就完成了；或者遇到一个零，在这种情况下可知 $\boldsymbol{x}$ 不在数据库中。

对于这种方法，很重要的一点是表格大小要比 S 大得多。如果 $2^M < S$，编码规则将会受阻，找不到地方来输入最后的字符串。

存储在其他地方

一个更鲁棒和灵活的方法是使用指针,指向存储冲突字符串的附加内存段。有很多方法可以做到这一点。举个例子,可以在散列表中的存储单元 h 中存储一个指针(它与有效的记录号 s 必须是可区分的)指向一个“存储桶”,其中具有散列码 h 的所有字符串都按有序表存储。创建散列表和存储桶之后,编码器就在每个存储桶中按照字母顺序排列字符串。

译码器只需简单地访问并查看相关的存储桶,再通过一个简短的字母序搜索,就可查找那里的字符串短列表。

这种在存储桶中存储字符串的方法允许人们将散列表变得非常小,这也许会有现实意义。可以将它做得很小,使得几乎所有字符串都包含于冲突中,于是所有的存储桶都包含少量的字符串。仅需进行少量的二进制比较,就可识别出存储桶中的哪个字符串与 x 相匹配。

12.4　冲突规划:生日问题



习题 12.3[2]　如果希望使用一个输出为 M 比特的散列函数来存储 S 个条目,假设散列函数是一个理想的随机函数,期望发生多少次冲突呢?如果希望冲突的期望数小于 1,则所需散列表的大小 M 是多少?

如果希望冲突的期望数在 S 中占一小部分,比如 1%,散列表的大小 M 是多少?

[注意这个问题与习题 9.20 之间的相似之处。]

12.5　散列码的其他用途

检验计算结果

如果想检查手算的加法是否正确,可能会觉得*弃九法*是一种有效的方法。在弃九法中,将所有位的数字模 9 相加所得的和,与答案所有位的数字模 9 相加所得的和进行比较。[稍加练习,计算这些和比原来的全位相加要快得多。]

例 12.4　在如下所示的加法计算 189 + 1254 + 238 中,其中各位数字的模 9 和为 7,而原答案中各位数字的模 9 和,即 1 + 6 + 8 + 1 所给出的结果也为 7。因此,这个计算通过了弃九法的检测。

$$\begin{array}{r} 189 \\ +1254 \\ +\ 238 \\ \hline 1681 \end{array}$$

弃九法给出了散列函数的一个简单例子。对于任何形如 $a+b+c+\cdots$ 的加法表示，其中 $a,b,c,\cdots$ 是十进制数，用下式定义 $h\in\{0,1,2,3,4,5,6,7,8\}$：

$$h(a+b+c+\cdots)=a,b,c\text{ 中所有数字的模 }9\text{ 和} \tag{12.1}$$

于是十进制算术具有如下良好性质，即如果

$$a+b+c+\cdots=m+n+o+\cdots \tag{12.2}$$

则散列 $h(a+b+c+\cdots)$ 与 $h(m+n+o+\cdots)$ 也相等。

▷ **习题 12.5**[小] 一个正确的弃九法匹配给出怎样的证据来支持加法计算是正确的这一假设？

朋友之间的错误检测

两个文件是否相同？如果文件位于同一台计算机中，就可以逐比特地比较它们。但是如果两个文件位于不同的机器上，最好有一种确定方法，不需要将文件中的一个从 A 传送到 B 就能确定两个文件是相同的。[即使传送了其中的一个文件，还是想要一种方法来确认所收到的文件是否未经改动！]

这个问题可以通过散列码来解决。设艾丽丝和鲍勃是两个文件的持有者；艾丽丝发送文件给鲍勃，并且他们希望确认该文件被无差错地接收。如果艾丽丝计算她文件的散列值并将其发送给鲍勃，而鲍勃也用同一 M 比特的散列函数来计算其文件的散列值，并发现两个散列值匹配，则鲍勃就几乎可以肯定地推断两个文件是一样的。

例 12.6 一个假负性(false negative)的概率是多少，即假设两个文件确实不同，而两个散列值却相同的概率是多少？

如果假设散列函数是随机的，并且导致文件不同的过程对于散列函数一无所知，则一个假负性的概率是 2^{-M}。

一个 32 比特散列所给出的假负性概率约为 10^{-10}。通常使用一个称作 32 位循环冗余校验的线性散列函数来检测文件中的错误。(一个循环冗余校验是一个 32 奇偶校验比特集合，类似于(7,4)汉明码的 3 个奇偶校验比特。)

> 如果假负性是由噪声所引起的，则 $M=32$ 比特足以使假负性率低于 10 亿分之一。

▷ **习题 12.7**[小] 这种简单的奇偶校验码仅能检测错误；但是没有帮助进行纠错。既然存在纠错码，为什么不使用它们中的一个来获得某些纠错能力呢？

篡改检测

如果两个文件的不同之处不仅是简单的“噪声”,而是对手引入的;这个对手是一个叫菲奥纳的聪明的伪造者,她将原始文件修改成一件伪造品,并声称这是艾丽丝的文件,情况又会如何?艾丽丝如何在文件中加入数字签名从而使鲍勃确认没人篡改文件?而且我们该如何防止菲奥纳通过监听得到艾丽丝的签名,并将此签名加在其他文件上?

假设艾丽丝为该文件计算出一个散列函数并安全地发给鲍勃。如果艾丽丝为文件计算出一个简单散列函数,如同线性循环冗余校验那样,而菲奥纳也知道是用这种方法验证文件的真实性。于是,菲奥纳就可以对文件做选择性的修改,然后非常容易地辨认出(通过线性代数)一个约32单一比特,当它们翻转时,将文件的散列函数恢复为原始值。*线性散列函数对于伪造者而言是没有安全性的。*

因此,必须要求散列函数*很难求逆*,这样就没有人可以构造一件伪造品而不影响散列函数。但是,我们还是希望散列函数易于计算,这样鲍勃就不必花费数小时来确认所接收的每个文件。这样一个易于计算但很难求逆的散列函数叫做*单向散列函数*。寻找这种函数是密码学的一个活跃研究邻域。

自由软件界广泛使用的确保两个文件没有不同的一个散列函数是MD5,它产生一个128比特的散列。它如何工作的详情非常复杂,包括复杂“异或”、“如果”以及“与”逻辑。①

即使有一个好的单向散列函数,如果菲奥纳可以取到散列函数,上面描述的数字签名还是容易被攻击。菲奥纳拿到经篡改的文件,并猎寻更进一步的小修改,使得它的散列值与艾丽丝文件的原散列值相匹配。这需要一些时间——平均来说,如果散列函数有32比特,需要约2^{32}次尝试——但是最终菲奥纳可以找到一个经篡改的文件同所给散列相匹配。为了能安全地防止伪造,数字签名必须有足够的比特数,从而使这样的随机搜索需要太长的时间,或者要对散列函数本身进行保密。

> 菲奥纳要进行诈骗必须散列2^M个文件。2^{32}次文件修改并不算很多,所以32位的散列函数用于防止伪造还是不够大。

另外一个想要伪造文件的人可能是艾丽丝自己。例如,她可能对一次赛马的结果下赌注,并且不希望公开公布她的预测;对她来讲,一种下注方法是将她下注的散列值发送给赛马赌注者鲍勃。稍后,她可以发送其下注的详细内容给鲍勃。每个人都可以确定她的赌注与前面公开化的散列有关。[当Isaac Newton和Robert Hooke希望确立科学思想的优先权而不暴露它们时,他们使用了这种秘密发表的方法。Hooke的散列函数是按照字母顺序排列的,可表示为UT TENSIO,SIC VIS向颠倒字

① http://www.freesoft.org/CIE/RFC/1321/3.htm

母顺序的 CEIIINOSSSTTUV 的转换。]这种协议基于如下假设:艾丽丝没法在事后改变她的赌注而不使散列出错。需要使用多大的散列函数来保证艾丽丝不能作弊?这个答案与上面击败菲奥纳所需散列的大小不同,因为艾丽丝是两个文件的作者。艾丽丝可以通过搜寻具有相同散列的两个文件来作弊。例如,如果她想将两个赌注押在一号价格上,她可对押一号做出多个(N_1 个)版本(彼此之间仅在小细节上有些差别),对押二号也做出多个(N_2 个)版本,并且将它们全部进行散列。如果不同类型的两个赌注的散列发生冲突,她就可以提交一个公共散列,她从而得到机会使所下赌注可以押在两者中的任一个上。

例 12.8 如果散列有 M 个比特,需要多大的 N_1 和 N_2 才能使艾丽丝有较大的机会找到具有相同散列的两个不同赌注?

这个问题与习题 9.20 的生日问题类似。如果宴会上有 N_1 个叫 Montague 的人和 N_2 个叫 Capulet 的人,并且为每个人指派 M 比特的"生日",则一个 Montague 和一个 Capulet 的生日发生冲突的期望次数为

$$N_1 N_2 2^{-M} \tag{12.3}$$

因此,为了使要散列的文件数 $N_1 + N_2$ 最小化,艾丽丝应该使 N_1 和 N_2 相等,并将需要散列大约 $2^{M/2}$ 个文件才能找到两个匹配。 □

> 艾丽丝必须散列 $2^{M/2}$ 个文件来作弊。[这是菲奥纳需要散列的次数的平方根。]

如果艾丽丝在 $T = 10$ 年内用了 $C = 10^6$ 台计算机,而每台计算机需要用 $t = 1$ ns 的时间来计算一个散列,只要 $M >> 2\log_2 CT/t \simeq 160$ 比特,赌博通信系统就可安全地防止艾丽丝的不诚实。

更多的阅读

关于散列码的经典之作是 Knuth(1968 年)第三卷。笔者强烈推荐阅读 Doug McIlroy 关于 spell 程序的故事;在 *Programming Pearl*(Bentley,2000 年)的第 13.8 节中可以找到它。这个奇特的软件使用了一个 64 KB 的数据结构,来存储一本包含 75 000个单词的词典中所有单词的拼写。

12.6 进一步的习题



习题 12.9[1] 如果收信人唯一,那么一封典型国际信件上的最短地址应该是什么?(假设允许使用的字符是[A ~ Z,0 ~ 9]。典型电子邮件地址有多长?



习题 12.10[2] 你确信以前没有人写过的一个字符串会是多长的一段文本?一

个此前没有人谱过的新曲子中有多少音符?

▷**习题 12.11**[$\mathcal{S}$] **分子的模式识别。**

在一个细胞中产生的某些蛋白质具有调节功能。一个调节型蛋白质在染色体中控制着特殊基因的复制。这种控制通常涉及将蛋白质捆绑到调节基因附近的一个特定 DNA 序列中去。这种被捆绑的蛋白质可以促进或抑制基因复制。

(a) 根据信息论的理论获得在整个人类基因组中充当一个基因调节器的典型蛋白质的大小的下界。假定基因组是 3×10^9 个核苷酸的一个序列,其字符表{A,C,G,T}由 4 个字符组成;一个蛋白质是一个氨基酸序列,其字符表由 20 个字符组成。[提示:确定可识别 DNA 序列要多长才能够保证序列在一个基因附近是唯一的,将其他基因组视为一个随机序列。然后讨论蛋白质必须要多大才可以唯一地识别那个长度的一个序列。]

(b) DNA 绑定的调节蛋白质所识别的某些序列包含一个重复两次或多次的子序列,例如,序列

$$\underline{\text{GCCCCCC}}\text{ACCCCT }\underline{\text{GCCCCC}} \tag{12.4}$$

是在人类 alpha-actin 基因的前面所找到的一个绑定点。某些绑定点包括一个重复子序列的事实是否会影响你关于问题(a)的答案?

12.7 习题解答

习题 12.1 的解答 首先设想比较字符串 $\boldsymbol{x}$ 和另一个随机字符串 $\boldsymbol{x}^{(s)}$。两个字符串的第 1 个比特相匹配的概率为 1/2。第 2 个比特相匹配的概率为 1/2。假设一旦遇到第 1 次失配就停止比较,匹配的期望数为 1,所以比较的期望数为 2(习题 2.34)。

假设正确的字符串随机地出现在原始列表中,平均将需要比较 $S/2$ 个字符串才能找到它,这需要 $2S/2$ 次二进制比较;如果字符串是随机选择的,给定 $S+N$ 次二进制比较的总期望值,则对正确字符串进行比较需要 N 次二进制比较。

在最坏的情况下(实际中确实存在),其他字符串与搜索键非常相似,于是要寻找一个失配,需要一个冗长的比较序列。最坏的情况是正确字符串位于列表的最后,并且所有的其他字符串仅是最后 1 个比特有所差别,则需要 SN 次二进制比较。

习题 12.2 的解答 由数据"$\boldsymbol{x}^{(s)}$ 和 $\boldsymbol{x}$ 的第 1 个比特相等"所给出的两个假设 $\mathcal{H}_0:\boldsymbol{x}^{(s)}=\boldsymbol{x}$ 和 $\mathcal{H}_1:\boldsymbol{x}^{(s)}\neq\boldsymbol{x}$ 的似然比为

$$\frac{P(\text{数据}\mid\mathcal{H}_0)}{P(\text{数据}\mid\mathcal{H}_1)}=\frac{1}{1/2}=2 \tag{12.5}$$

若前面的 r 个比特全部匹配,则似然比是 2^r 比 1。假设从相等几率开始,如果找到 30 个比特相匹配,则有利于 $\mathcal{H}_0$ 的几率是 10 亿比 1。[对于一个完整的答案,应该计算由先验信息所提供的证据,在表格中 $\boldsymbol{h}(\boldsymbol{x})$ 处找到散列项 s。这个事实为支持 $\mathcal{H}_0$ 提

供了更多的证据。]

习题 12.3 的解答 设散列函数有一个大小为 $T=2^M$ 的输出字母表。如果 M 等于 $\log_2 S$,则刚好有足够多的比特使得每一项都有它自己唯一的散列。在一个随机散列函数下,一对特定项发生冲突的概率为 $1/T$。总的对数为 $S(S-1)/2$。因此相互发生冲突的对数的期望值正好是

$$S(S-1)/(2T) \tag{12.6}$$

如果希望这个值小于 1,则需要 $T>S(S-1)/2$,因此

$$M>2\log_2 S \tag{12.7}$$

满足每一项有唯一名字需要的比特数为 $\log_2 S$,这里所需的比特数则两倍于此。

如果不介意偶尔发生冲突,涉及名字 S 中份额为 f 的一部分,则需要 $T>S/f$(因为某个特定名字发生冲突的概率为 $f\simeq S/T$)所以

$$M>\log_2 S+\log_2[1/f] \tag{12.8}$$

这就意味着对于 $f\simeq 0.01$,需要在 $\log_2 S$ 的基础上额外增加 7 个比特。

值得重点注意的是在两种情况下(12.7,12.8)T 随 S 变化的关系。如果希望散列函数是无冲突的,则必须有 T 大于 $\sim S^2$。如果不介意有较少频率的冲突,则 T 只需处于 S 这个数量级就可以了。

习题 12.5 的解答 两种假设,$\mathcal{H}_+$ =“计算正确”和 $\mathcal{H}_-$ =“计算不正确”的后验概率比是先验概率比 $P(\mathcal{H}_+)/P(\mathcal{H}_-)$ 与似然比 $P(\text{匹配}|\mathcal{H}_+)/P(\text{匹配}|\mathcal{H}_-)$ 的乘积。第 2 个因子就是本题的答案。分子 $P(\text{匹配}|\mathcal{H}_+)$ 等于 1。分母的值取决于所选择的误差模型。如果知道人类“计算器”在乘 10 时易于出错,或者造成相邻数字的错位,但两者都不会影响散列值,则 $P(\text{匹配}|\mathcal{H}_-)$ 也可能等于 1,所以正确的匹配并未给出有利于 $\mathcal{H}_+$ 的证据。但是如果假设错误“从散列函数的角度来看是随机的”,那么一个假正性的概率是 $P(\text{匹配}|\mathcal{H}_-)=1/9$,且正确匹配以 9:1 给出有利于 $\mathcal{H}_+$ 的证据。

习题 12.7 的解答 通过增加一个很小的 $M=32$ 个额外散列比特到一个巨大的 N - 比特文件,可以获得相当好的误码检测性能——差错未检测的概率是 2^{-M},小于 10 亿分之一。为了进行纠错,需要更多的校验比特,具体数目取决于期望的损坏类型以及文件大小。例如,如果在一个 1 MB 的文件中仅有 8 个随机比特出错,它需要使用约 $\log_2\binom{2^{23}}{8}\simeq 23\times 8\simeq 180$ 个比特来确定出错的比特是哪几个;根据习题 1.10(解答)所给出的计数论点,一个成功的纠错码所用的奇偶校验比特数至少应该是这个数。

习题 12.10 的解答 我们想知道一个字符串的长度 L,以便使它非常不可能与人类的全部著作的任何部分相匹配。现在估计这些著作的总数大约是每个活着的人一部,并且每本书包含 200 万个字符(200 页,每页 10 000 个字符)——即 10^6 个字符,这里的字符都是从 37 个字符所组成的字母表中抽取的。

一个随机选择的长度为 L 的字符串与人类著作总集在某点上相匹配的概率为

$1/37^L$。所以期望的匹配数为 $10^{16}/37^L$,如果 $L \geq 16/\log_{10} 37 \simeq 10$,则该值会变得极小。因为人类著作中存在冗余和重复,所以 $L \simeq 10$ 有可能是一个过高的估计。

所以如果你想写出独一无二的东西,坐下来并书写一个包含 10 个字符的字符串。但是不要写 gidnebinzz,因为我已经想到那个字符串。

因此对于一首新的乐曲,如果集中关注音符序列,忽略持续时间和重音,并允许每个音符最多有 8 度的跳跃,则每个音符的选择数是 23。第一个音符的基音是任意的。在这个相当难听的 Schönbergian 乐曲集里,长度为 r 个音符的旋律数为 23^{r-1};例如,对于长度 $r=5$ 有 250 000。限制所允许的间隔会减小这个数字;包括持续时间和重音在内又将增加这个数字。[如果限制重奏和全音程或半音程所允许的间隔,则减小得特别显著;这是"欢乐颂"的旋律听起来如此乏味的原因吗?]记录下来的乐曲总数可能小于 100 万。如果你在生命中的每个星期学习 100 段新的乐曲,则到你 50 岁时就学习了 250 000 段乐曲。根据玩"猜乐曲"游戏的经验①,笔者觉得尽管许多旋律中都会有共同的 4 - 音符序列,5 - 音符序列之间的冲突数量就会很少——那些最著名的 5 - 音符序列都是独一无二的。

习题 12.11 的解答 (a) 设 DNA 绑定蛋白质能认出一个长度为 L 的核苷酸序列。即,它优先绑定那个 DNA 序列,而不是整个基因组中的其他 DNA 序列。(实际上,识别的序列可能包含某些通配符,例如 TATAA * A 中的 * 可以表示"A,C,G 和 T"中的任意一个;所以,为了精确起见,假设所识别的序列包含 L 个非通配符。)

假定基因组的其余部分是"随机的",即,序列由核苷酸,A,C,G 和 T 以等概率形式随机地组成——这显然是不符合实际的,但是它不会对计算有太大的影响——在整个基因组中,长度为 N 的目标核苷酸序列不会再次出现的几率约为

$$(1-(1/4)^L)^N \simeq \exp\left(-N(1/4)^L\right) \tag{12.9}$$

只有当下式成立时,它才接近 1。

$$N4^{-L} \ll 1 \tag{12.10}$$

即

$$L > \log N/\log 4 \tag{12.11}$$

采用 $N=3\times10^9$,要求被识别的序列长于 $L_{\min}=16$ 个核苷酸。

这意味着多大尺寸的蛋白质呢?

- 假定蛋白质序列自身的信息量比它想绑定的核苷酸序列的信息量大(上面已经讨论过,至少要 32 比特),这样就可以得到一个弱下界。这给出了一个最小蛋白质长度,$32/\log_2(20) \simeq 7$ 个氨基酸。
- 从实际角度考虑,蛋白质对 DNA 序列的识别大概会使蛋白质同目标序列中

① 在"猜乐曲"游戏中,一个参与者选择一段曲子,并逐渐唱出更多的音符,而与此同时另一个参与者试图猜出整段曲子。

Parsons 码是一个与乐曲相关的散列函数:每对相邻的音符被编码为 U("上升")如果第 2 个音符高于第一个,为 R("重复")如果音调相同,其他情况为 D("下降")。你可以在 www.name-this-tune.com 发现这个散列函数的效果。

所有 16 个核苷酸接触。如果蛋白质是一个单体,它必须足够大才能同时与 DNA 的 16 个核苷酸接触。DNA 的一个螺旋状的旋转包括 10 个长度为 3.4 nm 的核苷酸,所以 16 个核苷酸的一个相邻序列的长度为 5.4 nm。因此,蛋白质的直径必须不小于 5.4 nm。蛋白溶解酵素是一种小球状蛋白质,其长度为 129 个氨基酸,直径约为 4 nm。假设体积与序列长度成正比,并且体积与直径成立方关系,一个直径为 5.4 nm 的蛋白质必须具有长度为 $2.5 \times 129 \simeq 324$ 个氨基酸的序列。

(b) 然而,如果一个目标序列由一个两次重复的子序列组成,所得出的结果可以是一个小许多的蛋白质,它只能识别子序列,只有在能形成一个二聚体时它才会与 DNA 紧密结合,它的两个半边都与所识别序列相结合。将蛋白质的直径减半,现在仅需要一个长度大于 $324/8 = 40$ 的氨基酸蛋白质。长度小于它的蛋白质本身无法成为一个基因特定的调节型蛋白质,因为它太小,无法充分地进行特定匹配——其可用的表面没有包含足够的信息量。

关于第 13 章

在第 8 章 ~ 第 11 章中，已经建立了具有任意输入、输出符号表的一般信道的香农噪声信道编码定理。编码理论大多集中在具有二进制输入的信道这种特殊情况上。将自己限定在这些信道上可以使问题简化，并将我们带入一个异常丰富多彩的世界，而在本书中只是稍做涉猎。

本章的目的之一是指出以下两个目标之间的差异：在一个噪声信道上获得可靠通信的香农目标，编码理论界的很多明显目标。许多编码理论家认为他们要解决的根本问题是，在一个 N 维空间中填充尽可能多的、半径尽可能大的球体，且没有出现球体重叠。荣誉属于那些发现能够多挤进几个球的填充的人。虽然这是一个奇妙的数学问题，但是可以看到，将一个码的码字之间的距离最大化这一目标，与香农的可靠通信目标之间的关系甚微。

第 13 章

二 进 制 码

我们已经建立了具有任意输入、输出字母的一般信道的香农噪声信道编码定理。编码理论大多集中在具有二进制输入的信道这种特殊情况上，而首当其冲的选择就是二进制对称信道。

给定一个二进制对称信道，一个码的最优译码器寻找在汉明距离下最接近接收向量的码字。两个二进制向量之间的汉明距离是这两个向量互异的坐标数。① 如果噪声使得被传输的码字 t 变为接收向量 r，而 r 又与另一个码字更近，则会发生译码错误。因此，码字之间的距离与译码差错概率有关。

13.1 一个码的距离特性

一个码的距离是指它的两个码字之间的最小距离。

例 13.1 (7,4)汉明码的距离 $d=3$。它的所有码字对之间至少有 3 比特互异。它所能纠正的最大错误数是 $t=1$；一般来说，一个距离为 d 的码可以纠正$\lfloor (d-1)/2 \rfloor$个错误。

关于距离的一个更准确的术语是码的最小距离。一个码的距离通常记做 d 或 $d_{\min}$。

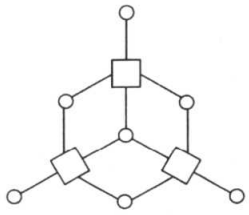
现在集中考虑线性码。在一个线性码中，因为所有码字具有相同的距离特性，所以可通过计算全零码字与其他码字之间的距离来概括码中的所有码字之间的距离。

一个码的权重枚举函数 $A(w)$ 的定义是：码中具有权重 w 的码字的个数。权重枚举函数也叫做码的距离分布。

例 13.2 (7,4)汉明码和十二面体码的权重枚举函数如图 13.1 和图 13.2 所示。

---

① 例如，00001111 和 11001101 之间的汉明距离为 3。



w	$A(w)$
0	1
3	7
4	7
7	1
总数	16

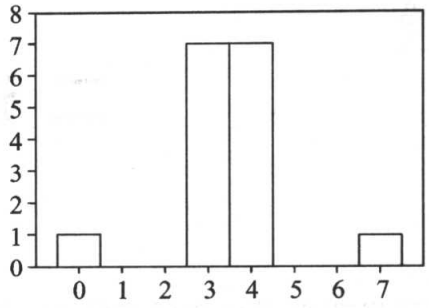
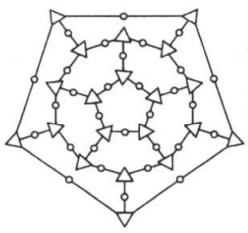


图 13.1 (7,4) 汉明码的图及其权重枚举函数



w	$A(w)$
0	1
5	12
8	30
9	20
10	72
11	120
12	100
13	180
14	240
15	272
16	345
17	300
18	200
19	120
20	36
总数	2 048

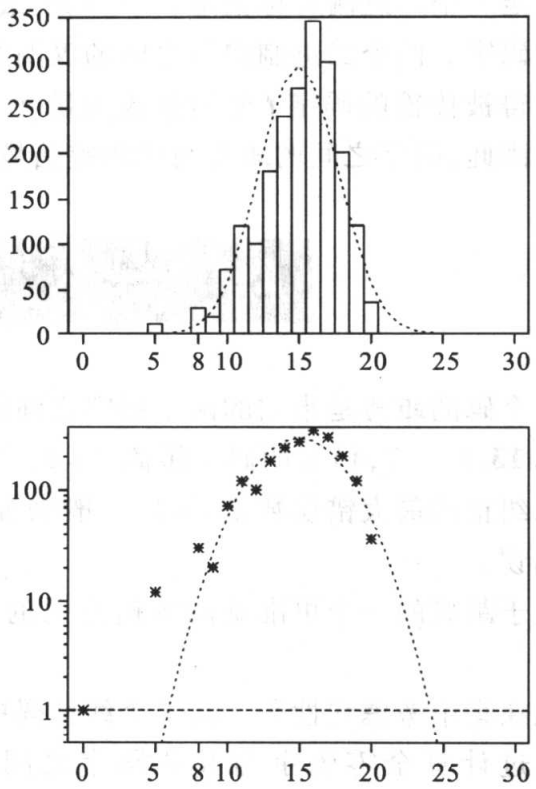


图 13.2 定义(30,11)十二面体码(圆圈是30个传输比特,三角形是20个奇偶校验比特,其中之一是冗余的)的图及其权重枚举函数(实线)。虚线表示具有相同大小生成式矩阵的所有随机线性码的平均权重枚举函数,稍后会计算它。下部的图用对数坐标表示同样的函数。

13.2 对距离的执著

因为一个码可确保纠正的最大错误数 t 与它的距离 d 之间的关系为 $t = \lfloor (d -$

1)/2」,[1]所以很多编码理论家将重点放在一个码的距离上,致力于寻找给定大小的且具有最大可能距离的码。很多实用编码理论所集中讨论的译码器,可以对权重不大于它们码距一半 t 的所有错误模式给出最优的译码。

一个距离受限的译码器是这样一种译码器:如果所接收的二进制向量 $\boldsymbol{r}$ 与那个码字之间的距离小于或等于 t,译码器将返回与接收向量 $\boldsymbol{r}$ 最接近的一个码字;否则它返回一条失败信息。

当多于 t 个错误发生时,就不再试图译码的原则也许是"我们不能保证可以纠正多于 t 个错误,所以就不必劳神尝试了——谁会对一个纠正权重大于 t 的某些错误模式却不能纠正另一部分错误的译码器感兴趣呢?"这种失败主义的态度是最坏情况论的一个例子,这是本书试图治愈的一种普遍的精神障碍。

事实上,距离受限译码器不能达到二进制对称信道的香农极限;只有一个经常可纠正多于 t 个错误的译码器才可以达到。纠错码的最新发展情况是译码器的工作范围大于码的最小距离。

好的和坏的距离属性的定义

已知分组长度 N 递增的一族码,其码率趋近一个极限 $R>0$,可以将这一族码归入以下类别,这有点类似于早先定义的"好的"码和"坏的"码(第 11.4 节)。

具有"好"距离的码序列 如果 d/N 趋于一个大于 0 的常数。

具有"坏"距离的码序列 如果 d/N 趋于 0。

具有"极坏"距离的码序列 如果 d 趋于一个常数。

例 13.3 一个低密度生成式矩阵码是一个线性码(如图 13.3 所示),在它的 $K\times N$维生成式矩阵 $\boldsymbol{G}$ 中,不管 N 有多大,每行都有少量的(d_0 个)1。这种码的最小距离至多为 d_0,所以低密度生成式矩阵码有"很坏的"距离。

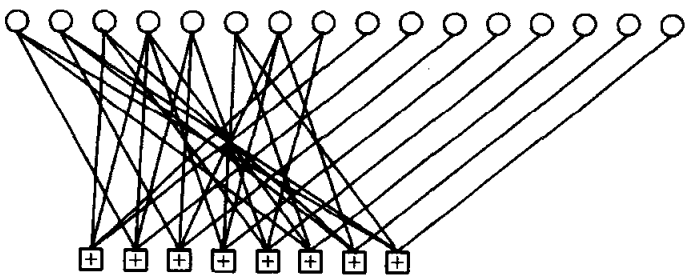


图 13.3 码率为 1/2 的低密度生成式矩阵码的图。最右边 M 个传输比特都和一个不同的奇偶约束相连。

尽管有大的距离不是坏事,后面将会看到,为什么看重距离会具有误导性。

① 若 d 是奇数,则 $d=2t+1$;若 d 是偶数,则 $d=2t+2$。

13.3 完 美 码

与 x 的汉明距离小于或等于 t 的点的集合，是汉明空间中以点 x 为球心的一个 t - 球体（或者一个半径为 t 的球体）。

(7,4)汉明码具有一个优美的特性：如果为它的16个码字各放置一个1 - 球体，则这些球体可以完美地填充汉明空间而没有重叠。正如在第1章所看到的那样，每个长度为7的二进制向量都在汉明码的一个码字的 $t=1$ 距离之内。

一个码是一个完美的 t - 纠错码 以该码的码字为中心的 t - 球体的集合可以无重叠地填充汉明空间（如图13.4所示）。

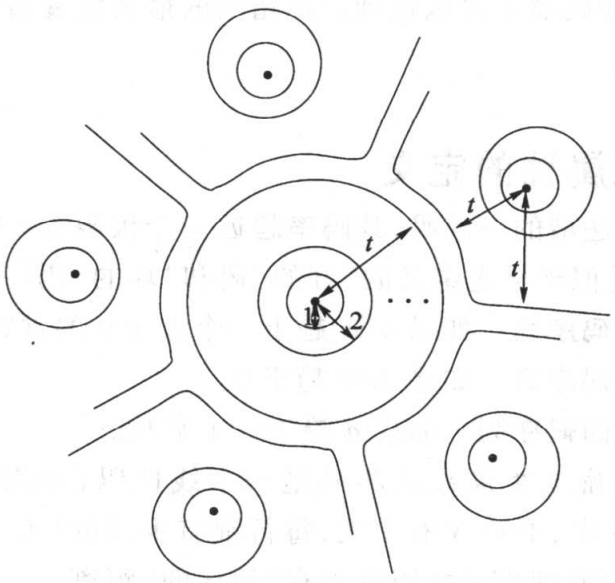


图13.4 以一个完美码的码字为中心的 t 个球体进行完美填充的部分汉明空间示意图

再来简要复述一下记号表。码字的数量为 $S=2^K$。整个汉明空间的点数为 2^N，半径为 t 的汉明球体中的点数为

$$\sum_{w=0}^{t}\binom{N}{w} \tag{13.1}$$

要使具有这些参数的一个码成为完美码，要求 t - 球体内点数的 S 倍等于 2^N：

$$\text{对于一个完美码，}2^K\sum_{w=0}^{t}\binom{N}{w}=2^N \tag{13.2}$$

$$\text{或者，等价地，}\sum_{w=0}^{t}\binom{N}{w}=2^{N-K} \tag{13.3}$$

对于一个完美码而言，一个球体中噪声向量的个数必须等于可能的校正子的个数。(7,4)汉明码满足这种数值逻辑条件，因为

$$1+\binom{7}{1}=2^3 \tag{13.4}$$

使用完美码有多么令人满意

如果存在大量的完美码可供选择，分组长度和码率都有一个大的范围，那么这是香农问题的完美解答。例如，可以在噪声水平为 f 的一个二进制对称信道中用这种方式通信：选择分组长度为 N 且 $t=f^*N$ 的一个完美 t - 纠错码，其中 $f^*=f+\delta$，并且选择 N 和 δ 使得噪声翻转大于 t 比特的概率小到令人满意的程度。

然而，*几乎不存在完美码*。仅有的非平凡完美二进制码是以下几种形式。

1. 汉明码，是 $t=1$ 且分组长度 $N=2^M-1$ 的完美码，其定义见下文；随着分组长度 N 的增加，汉明码的码率接近 1；

2. 分组长度 N 为奇数时的重复码，是 $t=(N-1)/2$ 的完美码；重复码的码率随 $1/N$ 趋近于 0；

3. 一个值得注意的能纠正 3 个错误的纠错码叫做二进制 Golay 码，其分组长度 $N=23$，有 2^{12} 个码字。[另一个 Golay 码能纠正 2 个错误，其分组长度 $N=11$，具有 3 - 元符号集，是由芬兰的一名叫 Juhani Virtakallio 的橄榄球球迷于 1947 年发明的。]

不存在其他的二进制完美码。为什么完美码这么少呢？难道是因为像式 (13.4) 汉明码和 Golay 码的参数所满足的那些精确的数值逻辑

$$1+\binom{23}{1}+\binom{23}{2}+\binom{23}{3}=2^{11} \tag{13.5}$$

很少出现巧合吗？是否存在足够多的“几乎完美”的码，使得 t - 球体可以几乎充满整个空间？

没有。实际上，以码字为中心的汉明球几乎充满汉明空间的图 (图 13.5) 也具有

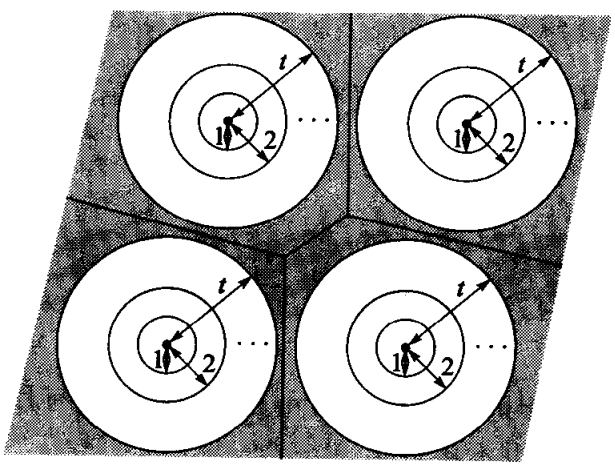


图 13.5 没有完全被以一个码的码字为中心的 t - 球体填充的汉明空间的示意图。灰色区域表示到任意码字的汉明距离都大于 t 的点。这个图容易使人产生误解，因为对于高维空间中有大 t 的任意码，球体之间的灰色区域占据了几乎所有的汉明空间。

误导性:对于很多码而言,无论它们是好码还是坏码,几乎所有汉明空间都被 t - 球体之间的空间所占据(如图 13.5 中灰色区域所示)。

描绘了这幅令人沮丧的图景之后,花一些时间对上述完美码的性质做一点补充。

汉明码

(7,4)汉明码可以定义为线性码,它的 3 × 7 奇偶校验矩阵包含所有 7($=2^3-1$)个长度为 3 的非零向量作为列。因为这 7 个向量都不相同,所以任意单一比特的翻转都会产生一个不同的校正子,所以所有单一比特错误都能被检测并得以纠正。

可以用 $M=3$ 的奇偶约束条件来推广这个码。汉明码是单比特纠错码,由选取的奇偶校验约束条件的数量 M 来定义;分组长度 N 为 $N=2^M-1$;奇偶校验矩阵包含所有 N 个长度为 M 比特的非零向量作为它的列。

最前面的几个汉明码的码率如下:

校验数 M	(N,K)	$R=K/N$	
2	(3,1)	1/3	循环码 R_3
3	(7,4)	4/7	(7,4) 汉明码
4	(15,11)	11/15	
5	(31,26)	26/31	
6	(63,57)	57/63	



习题 13.4[2] 当用于噪声密度为 f 的二进制对称信道时,(N,K) 汉明码的分组差错概率的主导阶是多少?

13.4 完美是达不到的——第一次证明

将用几种方法证明有用的完美码根本不存在(这里,“有用”是指“具有大的分组长度 N,且码率既不接近 0 也不接近 1”)。

香农证明了给定具有任意噪声水平 f 的一个二进制对称信道,存在具有大分组长度 N 和任意接近 $C(f)=1-H_2(f)$ 的码,并可用它以任意小的差错概率进行通信。对于大的 N,每个分组中的差错数量通常约为 fN,所以香农的这些码是“几乎可以确定的 fN - 纠错码”。

选择$f\in(1/3,1/2)$的噪声信道特例。是否能找到一个大的完美码实现fN-纠错呢？好，先假定已经找到了这样一个码，并只查看它的3个码字。（记住，这个码的码率应为$R\simeq 1-H_2(f)$，所以它的码字数量应该是巨大的（2^{NR}个）。）不失一般性，选择码字中的一个全零码字，并定义其他两个码字与它有如图13.6所示的重叠。第2个码字在它的坐标$u+v$部分与第1个码字有所不同。第3个码字在坐标$v+w$部分与第1个码字有所不同，与第2个码字在坐标$u+w$部分有所不同。在所有3个码字中都有比例为x的部分坐标是零值。现在，如果码是fN-纠错码，则它的最小距离必定大于$2fN$，因此

$$u+v>2f, v+w>2f, u+w>2f \tag{13.6}$$

将这3个不等式相加并被2除，有

$$u+v+w>3f \tag{13.7}$$

因此如果$f>1/3$，就能够推断$u+v+w>1$，于是$x<0$，而这是不可能的。这样的一个码是不存在的。所以码不可能有3个码字，更不用说2^{NR}个了。

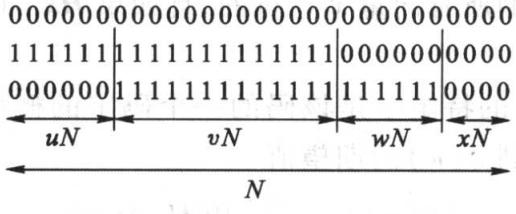


图13.6 3个码字

由此得出结论：尽管香农证明了存在很多码可用来在$f>1/3$的二进制对称信道上进行通信，但是没有完美码可以做到这一点。

现在来研究一种更具一般性的论点，以表明对于一般的码率（除1和0之外），不存在大的完美线性码。通过寻找一个随机线性码的典型距离来证明。

13.5 随机线性码的权重枚举函数

假设通过在$M\times N$奇偶校验矩阵$\boldsymbol{H}$（如图13.7所示）中随机选用二进制元素来构造一个码。应该期望得到什么样的权重枚举函数呢？

一个具有奇偶检验矩阵$\boldsymbol{H}$的特定码的权重枚举式$A(w)_{\boldsymbol{H}}$是重量为w的码字的数量，它可以写为

$$\overbrace{\begin{bmatrix}
1010101001001101000010110\\
0011101111000110011010 00\\
1011101110010110001101 00\\
0000101111001011010010 00\\
0000001100111101000001 00\\
1100100011111000001011 10\\
1011111000101000010011 10\\
1100101100011010111010 10\\
1000111001010000101111 01\\
0100010000101010011010 10\\
0101111101111111101110 10\\
1011101010010011010000 11
\end{bmatrix}}^{N}\Bigg\}M$$

图 13.7 一个随机二进制奇偶校验矩阵

$$A(w)_H = \sum_{x:|x|=w} 1[\boldsymbol{Hx}=0] \tag{13.8}$$

其中求和是对重量为 w 的所有向量 $\boldsymbol{x}$ 进行的，且如果 $\boldsymbol{Hx}=0$，真值函数 $1[\boldsymbol{Hx}=0]$ 等于 1，否则为 0。

通过计算重量 $w>0$ 的特定字是该码的一个码字的概率（对全集中所有二进制线性码求平均），可以得到 $A(w)$ 的期望值

$$\langle A(w)\rangle = \sum_{H} P(\boldsymbol{H})A(w)_H \tag{13.9}$$

$$= \sum_{x:|x|=w}\sum_{H} P(\boldsymbol{H})\,1[\boldsymbol{Hx}=0] \tag{13.10}$$

由对称性，这个概率仅取决于字的重量 w，而不取决于字的细节。整个校正子 $\boldsymbol{Hx}$ 为零的概率等于校正子的 M 个比特中的每一个为零的概率之积。校正子中的每个比特 z_m 是 w 个随机比特的和（模 2），因此 $z_m=0$ 的概率为 1/2。因此 $\boldsymbol{Hx}=0$ 的概率为

$$\sum_{H} P(\boldsymbol{H})1[\boldsymbol{Hx}=0] = (1/2)^M = 2^{-M} \tag{13.11}$$

与 w 无关。

权重为 w 的字的期望数(13.10)是对所有权重为 w 的码字而言的，是将每个字是一个码字的概率进行求和得到的。权重为 w 的字的数量为 $\binom{N}{w}$，因此

$$\langle A(w)\rangle = \binom{N}{w}2^{-M}, \quad \text{对于任意 } w>0 \tag{13.12}$$

对于大的 N，可以用 $\log\binom{N}{w}\simeq NH_2(w/N)$ 和 $R\simeq 1-M/N$ 来写出

$$\log_2\langle A(w)\rangle \simeq NH_2(w/N) - M \tag{13.13}$$

$$\simeq N[H_2(w/N)-(1-R)], \quad \text{对于任意 } w>0 \tag{13.14}$$

作为一个具体例子，图 13.8 表示了 $N=540, M=360$，码率为 1/3 的随机线性码的期望权重枚举函数。

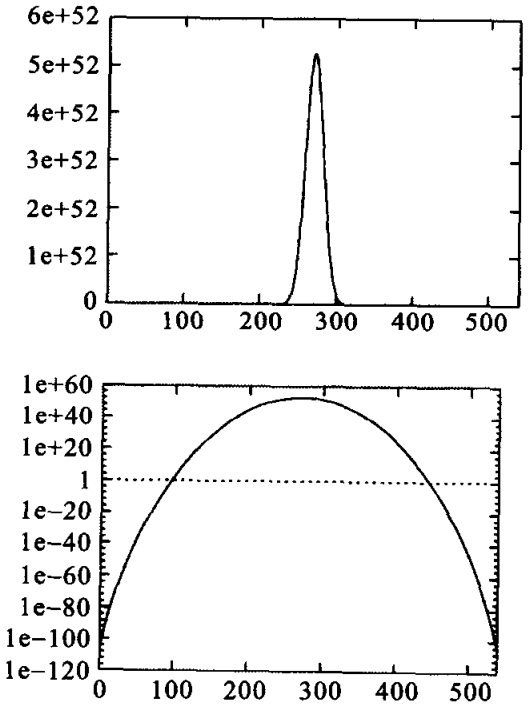


图 13.8 一个 $N=540, M=360$ 的随机线性码的期望权重枚举函数 $\langle A(w)\rangle$。下方的图形是用对数坐标表示的 $\langle A(w)\rangle$。

Gilbert – Varshamov 距离

对于满足 $H_2(w/N) < (1-R)$ 的权重 w，$A(w)$ 的期望值小于 1；对于满足 $H_2(w/N) > (1-R)$ 的权重 w，期望值大于 1。因此希望对于大的 N 值，一个随机线性码的最小距离将接近距离 d_{GV}，其定义式如下

$$H_2(d_{GV}/N) = (1-R) \tag{13.15}$$

定义 距离 $d_{GV} \equiv NH_2^{-1}(1-R)$ 是码率为 R、分组长度为 N 的 Gilbert – Varshamov 距离。

被广泛相信的 Gilbert – Varshamov 假设声称（对于大的 N）不可能创建最小距离比 d_{GV} 大很多的二进制码。

定义 假定 Gilbert – Varshamov 假设成立，则 Gilbert – Varshamov 速率 R_{GV} 是你可以用一个距离受限译码器（第 13.2 节定义）进行可靠通信的最大速率。

为什么填球是一种坏的观点，并且执著于距离是不适当的

如果使用距离受限译码器，则最大噪声容限将会翻转所占比重为 $f_{bd}=\frac{1}{2}d_{min}/N$ 的比特。因此，假设 d_{min} 等于 Gilbert 距离 d_{GV}，则有

$$H_2(2f_{bd})=(1-R_{GV}) \tag{13.16}$$

$$R_{GV}=1-H_2(2f_{bd}) \tag{13.17}$$

这就是关键所在：香农所说可实现的是什么？他说通信的最大可能速率就是容量

$$C=1-H_2(f) \tag{13.18}$$

因此，对于一个给定速率 R，根据香农定理，最大噪声容限（如图 13.9 所示）由下式给出。

$$H_2(f)=(1-R) \tag{13.19}$$

结论是：假设选取一个速率为 R 的好码；式（13.16）和式（13.19）分别定义了一个距离受限译码器和香农译码器的最大噪声容限 f_{bd} 和 f。

$$f_{bd}=f/2 \tag{13.20}$$

距离受限译码器可以容忍的噪声水平永远只是香农证明可容忍水平的一半！

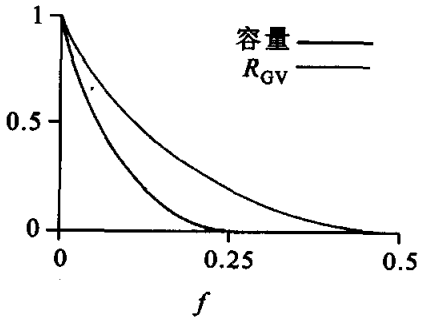


图 13.9 作为噪声水平 f 的函数，香农信道容量 C 和 Gilbert 速率 R_{GV} 之间的对比。R_{GV} 是使用一个距离受限译码器所能获得的最大通信速率。对于任意给定的速率 R，香农的最大噪声容限是一个“最坏情况者”所用距离受限译码器的最大噪声容限的两倍。

这与完美码有什么关系呢？一个码是完美码，如果它的码字周围存在 t - 球体，并且这些球体无重叠地填充汉明空间。但当一个典型随机线性码被用在二进制对称信道中以接近香农极限的速率进行通信时，典型的比特翻转数是 fN，且码字之间的最小距离也是 fN；如果比香农极限低一点，距离可以更大一点。因此，码字周围的 fN - 球体彼此充分地重叠，以至于每个球几乎包含相邻球体的中心！这种重叠不是灾难性的原因是，在高维情况下，如图 13.10 中的阴影所示，重叠部分所占的体积只是任一球体的一小部分而已，因此，落在这部分的概率也非常小。

这个故事的寓意是“最坏情况论调”对你没好处，它将你容忍噪声的能力减半。

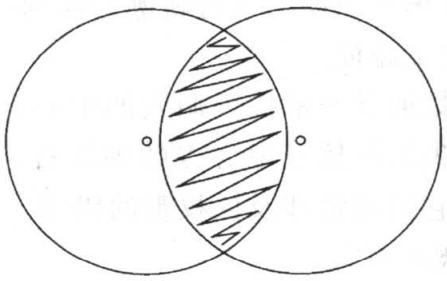


图 13.10 两个重叠的球体，它们的半径几乎与球中心之间的距离一样大。

因此，你必须能够对一个码的最小距离以外的错误进行译码，才可能达到香农极限。

然而，一个码的最小距离在实际中却是令人感兴趣的，因为在某些条件下，最小距离主导着一个码所产生的错误。

13.6 Berlekamp 蝙蝠

一个瞎的蝙蝠居住在一个洞穴里。它在洞穴的中心附近盘旋，它对应于一个码字，它到中心的典型距离由一个活跃参数 f 控制。（蝙蝠到中心的位移对应于噪声向量。）洞穴的边界由钟乳石组成，指向洞穴的中心（如图 13.11 所示）。每个钟乳石类似于本地码字与另一码字之间的边界。钟乳石就如同图 13.10 中的阴影部分，但是被重塑以表达它是一块体积很小的区域这种思想。

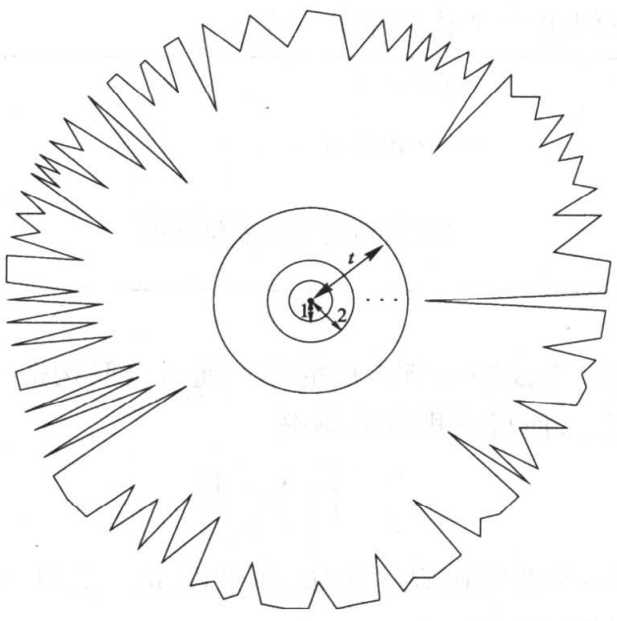


图 13.11 一个码字附近的汉明空间的 Berlekamp 示意图。锯齿状的实线围绕着与该码字距离最近的所有点。码字周围的 t - 球体占据这个空间的一小部分。

对错误进行译码对应于蝙蝠预期的飞行轨迹会穿越一个钟乳石。可能在与中心不同距离的地方与钟乳石发生碰撞。

如果活跃程度很低,蝙蝠通常非常接近洞穴的中心;碰撞将很少发生,并且一旦发生,它们通常只涉及那些尖端最接近中心点的钟乳石。类似地,在低噪声条件下,对错误的译码会很少,并且它们通常涉及低权重的码字。在低噪声条件下,一个码的最小距离与(很小的)差错概率有关。

如果活跃程度较高,蝙蝠就会经常在超出安全距离 t 以外的地方飞行,其中 t 是最长钟乳石开始的地方。蝙蝠将会与更远处的钟乳石发生更频繁的碰撞,因为那里钟乳石数量更多。因为在最小距离内只有很少量的钟乳石,所以相对而言它们不可能引起错误。同样地,一个实际纠错码的错误依赖于权重枚举函数的特性。

如果活跃程度非常高,蝙蝠总在距离洞穴中心很远的地方飞行,并且几乎所有碰撞都包括与远处钟乳石的接触。在这些条件下,蝙蝠的碰撞频率与中心到最近钟乳石之间的距离无关。

13.7 汉明码的级联

在图 11.6 中,初次涉及汉明码的级联这一概念,对它多做些讨论很有意义,这是因为我们可以深入了解"好码"的概念以及码的最小距离的相关性或其他一些问题。

可以为噪声密度为 f 的二进制对称信道构造一个级联码,它由几个汉明码连续编码而成。

下表简要重述了汉明码的关键特性,以约束数量 M 为索引。所有汉明码都具有最小距离 $d=3$ 并可以纠正 N 个错误中的一个。

$N=2^M-1$	分组长度
$K=N-M$	信源比特数
$p_B=\frac{3}{N}\binom{N}{2}f^2$	到主导阶的分组差错概率

如果将 M 递增的 C 个汉明码所组成的序列通过级联构造一个乘积码,可以适当地选择那些参数 $\{M_c\}_{c=1}^{C}$,使得乘积码的码率

$$R_C=\prod_{c=1}^{C}\frac{N_c-M_c}{N_c} \tag{13.21}$$

随着 C 的增加而趋于一个非零极限。例如,如果设 $M_1=2, M_2=3, M_3=4$,等等,则渐近码率为 0.093(如图 13.12 所示)。

分组长度 N 是 C 的一个快速增长函数,所以这些码都有些不切实际。这些码的另一个弱点是它们的最小距离不很好(如图 13.13 所示)。每个汉明码分量的最小

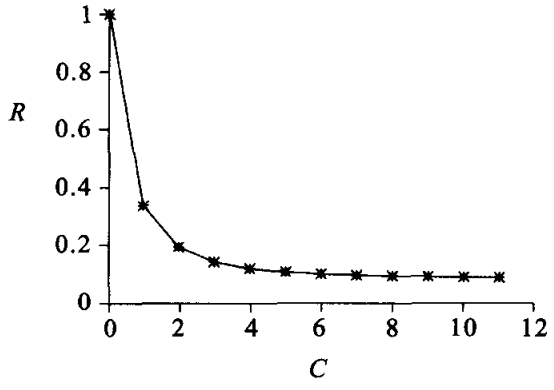


图 13.12 级联汉明码的码率 R,以级联数 C 的函数表示

距离为 3,所以第 C 次乘积后的最小距离为 3^C。分组长度 N 比 3^C 增长得快,所以随着 C 的增长,比值 d/N 趋近于零。相反的,对于典型随机码,比值 d/N 趋近于一个常数,使得 $H_2(d/N) = 1 - R$。因此,级联汉明码具有"坏的"距离。

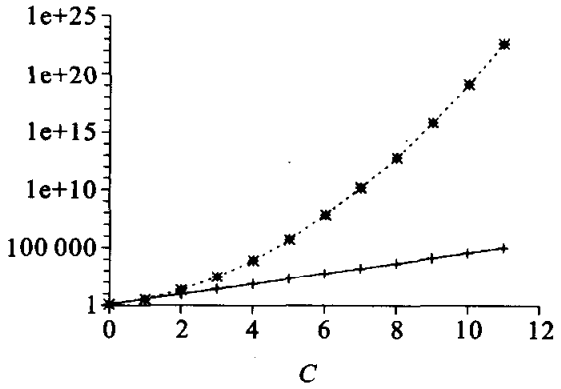


图 13.13 作为级联数 C 的函数的级联汉明码的分组长度 N_C(上面的曲线)和最小距离 d_C(下面的曲线)

然而,结果表明,这个简单的码序列会在某些信道上产生好码——但不是非常好的码(参见第 11.4 节,回顾术语"好的"和"非常好的"定义)。在此不证明这个结果,而只是简单地做些数值探究。

图 13.14 给出了级联码的误比特概率 p_b,假设对分量码进行串行译码,如第 11.4节所述。[在那里已经看到,这种一次一码的译码是次优的。] 横坐标轴表示码的码率。随着级联数的增加,码率降为 0.093,而差错概率下降至接近于 0。假设图中的信道是 $f = 0.058\ 8$ 的二进制对称信道。这是采用级联码所能容忍的最高噪声水平。

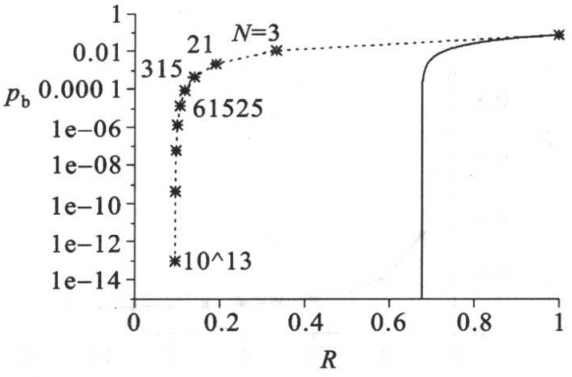


图 13.14 在 $f=0.058\ 8$ 的二进制对称信道中，级联汉明码的误比特概率与码率 R 的关系曲线图。点旁边的标志表示了分组长度 N。实线表示该信道的香农极限。当码率趋近于 0.093 时，误比特概率降为 0，所以级联汉明码是一族“好的”码。

这个故事传递的信息是距离并非一切。一个码的最小距离，尽管被编码理论家广泛推崇，但对于香农希望在噪声信道上获得可靠通信的使命而言，并不具有根本的重要性。

▷ **习题 13.5**[p] 证明存在一族码，有“坏的”距离却是“很好”的码。

13.8 距离不是一切

针对一个二进制对称信道的特殊情况，让我们对一个码的最小距离的影响有一个定量认识。

与一个低权重码字相关的差错概率

设一个二进制码的分组长度为 N 且只有两个码字，它们之间有 d 处不同。为了简单起见，假设 d 为偶数。如果将这个码用在噪声水平为 f 的二进制对称信道上，它的差错概率是多少？

仅在两个码字不同的地方出现比特翻转才是相干的。差错概率由这些比特中 $d/2$ 个被翻转的概率主导。其他比特发生了什么情况是无关紧要的，因为最优译码器会忽略它们。

$$P(\text{分组差错}) \simeq \binom{d}{d/2} f^{d/2}(1-f)^{d/2} \tag{13.22}$$

与单个权重为 d 的码字有关的差错概率如图 13.15 所示。利用式(1.16)的二项式系数的近似值，可以进一步近似

$$P(\text{分组差错}) \simeq [2f^{1/2}(1-f)^{1/2}]^d \tag{13.23}$$

$$\equiv [\beta(f)]^d \tag{13.24}$$

其中 $\beta(f) = 2f^{1/2}(1-f)^{1/2}$ 被称为信道的 Bhattacharyya 参数。

现在,考虑一个距离为 d 的一般线性码。它的分组差错概率必须至少是 $\binom{d}{d/2} f^{d/2}(1-f)^{d/2}$,与码的分组长度 N 无关。正是由于这个原因,在任意二进制对称信道中,具有递增分组长度 N 和常数距离 d(即"很坏的"距离)的码序列不可能具有趋近于 0 的分组差错概率。如果对构造差错概率极小的超级纠错码感兴趣,就可能要避开具有坏距离的码。然而,为了注重实效,应更仔细地观察图 13.15。在第 1 章中,讨论了用于磁盘驱动器的码,它要求差错概率大约小于 10^{-18}。如果磁盘驱动器的原始差错概率约为 0.001,则与距离 $d=20$ 的码字所对应的差错概率小于 10^{-24}。如果磁盘驱动器的原始差错概率约为 0.01,则与距离 $d=30$ 的码字有关的差错概率小于 10^{-20}。因此,出于实用的目的,一个码是否有好的距离并不重要。例如,分组长度为 10 000 的码,已知具有很多权重为 32 的码字,却能够以很小的差错概率纠正权重为 320 的错误。

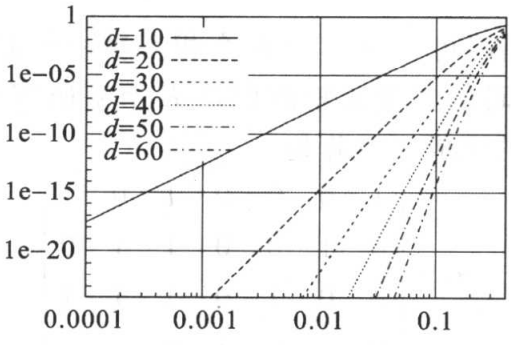


图 13.15 与单个权重为 d 的码字有关的差错概率 $\binom{d}{d/2} f^{d/2}(1-f)^{d/2}$,它是 f 的函数。

笔者不希望你们认为我在推荐使用具有坏距离的码;在第 47 章中,将讨论我所钟爱的低密度奇偶校验码,它同时具有优异的性能和好的距离。

13.9 联 合 界

在二进制对称信道中,一个码的差错概率可以根据它的权重枚举函数加上一个与单一码字相关的差错概率(式(13.24))的适当倍数来定界。

$$P(\text{分组差错}) \leqslant \sum_{w>0} A(w)[\beta(f)]^w \tag{13.25}$$

该不等式是联合界的一个例子,它对低噪声水平 f 来说是精确的,但对于高噪声水平就不再准确了,因为它高估了那些同时导致多个码字混淆的差错的影响力。

▷ **习题 13.6**[f] 穷人的噪声信道编码定理

假设联合界(13.25)是准确的,并用式(13.14)(第13.5节)中的一个随机线性码的平均权重枚举函数作为 $A(w)$,估算在二进制对称信道中通信的最大速率 $R_{\mathrm{UB}}(f)$。

或者,从更积极的角度来看,使用联合界(13.25)作为不等式,证明在二进制对称信道中,不高于 $R_{\mathrm{UB}}(f)$ 的通信速率是可以达到的。

在下一章,通过分析一个二进制线性码的*校正子译码*的差错概率,并使用联合界,将证明香农的噪声信道编码定理(在二进制对称信道中),并因此证明了*存在非常好的线性码*。

13.10 对偶码

编码定理中的一个比较重要的概念是线性纠错码的对偶,尽管在这本书中并未直接用到它。

一个 (N,K) 线性纠错码可以视做由 K 个相互独立的基码字的所有组合相加而生成的一组 2^K 个码字。码的生成式矩阵由这 K 个基码字组成,通常将其写成行向量。例如,(7,4)汉明码的生成式矩阵是

$$\boldsymbol{G}=\begin{bmatrix}1&0&0&0&1&0&1\\0&1&0&0&1&1&0\\0&0&1&0&1&1&1\\0&0&0&1&0&1&1\end{bmatrix} \tag{13.26}$$

而且它的16个码字在表1.14中列出。这个码的码字是4个向量[1 0 0 0 1 0 1],[0 1 0 0 1 1 0],[0 0 1 0 1 1 1],[0 0 0 1 0 1 1]的线性组合。

一个 (N,K) 码也可以根据 $M\times N$ 奇偶校验矩阵(其中 $M=N-K$)来描述为一组满足

$$\boldsymbol{Ht}=\boldsymbol{0} \tag{13.27}$$

的向量 $\{\boldsymbol{t}\}$。思考这个等式的一种方法是:如果 $\boldsymbol{t}$ 是一个码字,它就必须与 $\boldsymbol{H}$ 中每一个行向量正交。

> 生成式矩阵指定 K 个向量,通过它们可以构建所有码字,奇偶校验矩阵指定一组 M 个向量,所有码字都与它们正交。
>
> 码的对偶是通过交换生成式矩阵和奇偶校验矩阵得到的。

定义 与一个码 $\mathscr{C}$ 中的所有码字都正交、长度为 N 的所有向量的集合称作该码的对偶 $\mathscr{C}^{\perp}$。

若 $\boldsymbol{t}$ 与 $\boldsymbol{h}_1$ 和 $\boldsymbol{h}_2$ 正交,则它与 $\boldsymbol{h}_3\equiv\boldsymbol{h}_1+\boldsymbol{h}_2$ 也是正交的;所以所有码字与 $\boldsymbol{H}$ 中 M 行的任意线性组合都是正交的。因此,奇偶校验矩阵中行的所有线性组合集都是对

偶码。

对于(7,4)汉明码,奇偶校验矩阵是

$$\boldsymbol{H}=[\boldsymbol{P}\quad \boldsymbol{I}_3]=\begin{bmatrix}1&1&1&0&1&0&0\\0&1&1&1&0&1&0\\1&0&1&1&0&0&1\end{bmatrix} \tag{13.28}$$

(7,4)汉明码的对偶 $\mathscr{H}_{(7,4)}$ 如表 13.16 中的码所示。

表 13.16 (7,4)汉明码的对偶的 8 个码字[与表 1.14 进行比较]

0000000	0101101	1001110	1100011
0010111	0111010	1011001	1110100

这一对码有一个意想不到的特性是,对偶 $\mathscr{H}_{(7,4)}^{\perp}$ 包含在 $\mathscr{H}_{(7,4)}$ 自身中:对偶码中的每个码字都是原(7,4)汉明码中的一个码字。这种关系可以用集合符号写为

$$\mathscr{H}_{(7,4)}^{\perp}\subset\mathscr{H}_{(7,4)} \tag{13.29}$$

如果将向量视为实向量,则对偶向量的集合与码字向量的集合相重叠的可能性是违反直觉的——一个向量如何与它自己正交呢?但是工作在模 2 算术中时,很多非零向量的确与它们自身正交!

▷**习题 13.7**[✓] 给出一个简单的规则来辨别二进制向量是否与其自身正交,并检验如下 3 个向量:[1 1 1 0 1 0 0],[0 1 1 1 0 1 0]和[1 0 1 1 0 0 1]。

更多的对偶

通常一个码有一个系统的生成式矩阵

$$\boldsymbol{G}=[\boldsymbol{I}_K|\boldsymbol{P}^{\mathrm{T}}] \tag{13.30}$$

其中 $\boldsymbol{P}$ 是一个 $K\times M$ 矩阵,则它的奇偶校验矩阵为

$$\boldsymbol{H}=[\boldsymbol{P}|\boldsymbol{I}_M] \tag{13.31}$$

例 13.8 重复码 R_3 的生成式矩阵为

$$\boldsymbol{G}=[1\quad 1\quad 1] \tag{13.32}$$

它的奇偶校验矩阵为

$$\boldsymbol{H}=\begin{bmatrix}1&1&0\\1&0&1\end{bmatrix} \tag{13.33}$$

两个码字是[1 1 1]和[0 0 0]。

对偶码的生成式矩阵为

$$\boldsymbol{G}^{\perp}=\boldsymbol{H}=\begin{bmatrix}1&1&0\\1&0&1\end{bmatrix} \tag{13.34}$$

或等价地,通过行相加将 $\boldsymbol{G}^{\perp}$ 改为系统形式

$$\boldsymbol{G}^{\perp}=\begin{bmatrix}1&0&1\\0&1&1\end{bmatrix} \tag{13.35}$$

称这种对偶码是*简单奇偶码* P_3；这种码有一位奇偶校验比特，它等于两个信源比特之和。对偶码的4个码字是[1 1 0]，[1 0 1]，[0 0 0]和[0 1 1]。

在这种情况下，码与其对偶码的唯一共同向量是全零码字。

对偶码的好坏

若一个码序列是"好的"，那么它们的对偶也是好的吗？所有情况都可以构造出例子：好码有好的对偶（随机线性码）；坏码有坏的对偶；好码有坏的对偶。最后一类尤为重要：很多最新的码都具有这种特性：其对偶是坏的。经典的例子是低密度奇偶校验码，它的对偶是低密度生成式矩阵码。

▷ **习题 13.9**[2] 证明低密度生成式矩阵码是坏的。一族低密度生成式矩阵码由两个参数 j,k 来定义，它们分别是 $\boldsymbol{G}$ 中所有行和列的列权重和行权重。这些权重是固定的，与 N 无关；例如，$(j,k)=(3,6)$。[提示：先证明码有低权重码字，然后使用第13.8节的论点。]

习题 13.10[2] 证明低密度奇偶校验码是好码且具有好的距离。（求解时，参见Gallager（1963年）和MacKay（1999b）。）

自对偶码

(7,4)汉明码具有其对偶码也包含在自身之中的特性。如果一个码包含在它的对偶码中，则这个码是*自正交的*。例如，(7,4)汉明码的对偶是一个自正交码。看到这一点的一种方法是 $\boldsymbol{H}$ 中任意两行之间的重叠是偶数。包含其对偶的码在量子纠错中很重要（Calderbank 和 Shor，1996年）。

尽管不一定有用，但这些自对偶码看着就让人着迷。如果码的对偶码与该码相同，则码 $\mathcal{C}$ 是自对偶的。

$$\mathcal{C}^{\perp}=\mathcal{C} \tag{13.36}$$

可以推导出自对偶码的一些特性。

1. 如果一个码是自对偶的，那么它的生成式矩阵也是它的奇偶校验矩阵。
2. 自对偶码的码率为1/2，即 $M=K=N/2$。
3. 所有码字的权重都是偶数。

▷ **习题 13.11**[2] 如果具有生成式矩阵 $\boldsymbol{G}=[\boldsymbol{I}_K|\boldsymbol{P}^{\mathrm{T}}]$ 的码是自对偶码，则矩阵 $\boldsymbol{P}$ 必须满足什么性质？

自对偶码的例子

1. 重复码 R_2 是一个自对偶码的简单例子。

$$\boldsymbol{G}=\boldsymbol{H}=[1 \quad 1] \tag{13.37}$$

2. 最小非平凡自对偶码是下面的(8,4)码。

$$G = [I_4 | P^{\mathrm{T}}] = \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{array}\right] \tag{13.38}$$

▷ **习题 13.12**[⌀] 求上述(8,4)码与(7,4)汉明码之间的关系。

对偶和图

假设用图来表示码,图中有两种类型的结点,奇偶校验约束和等式约束,由代表码中比特的边线相连接(并非所有比特都需要传输)。

对偶码的图是将所有奇偶校验结点用等式结点代替而得到的,反之亦然。这种图被 Forney(2001 年)称为正则图。

进一步的阅读

在编码理论中,对偶码是重要的,因为有关一个码的函数(例如码字的后验分布)可以通过傅里叶变换得到其对偶码的函数。关于有限群的傅里叶分析的入门介绍可以参考 Terras(1999 年)。也可以参考 MacWilliams 和 Sloane(1977 年)。

13.11 将完美性推广到其他信道

放弃了对二进制对称信道中完美码的搜索,可以通过改变信道得到些安慰。如果码能够恢复任意 u 个(但决不会超过 u 个)已删除的比特,则称这个码为“二进制删除信道中的一个完美 u - 纠错码”。不过,习惯术语不用“完美”这个词,而是称这种码为“最大可分隔距离码”或 MDS 码。①

正如在习题 11.10 中所述,(7,4)汉明码不是一种 MDS 码。它能恢复某些 3 个已删除比特的集合而非全部。如果对应于权重为 3 的码字的任意 3 比特被删除,那么信息的 1 个比特就无法恢复。这就是为什么(7,4)码对于 RAID 系统是一种拙劣选择的原因。

一个最大可分隔距离码的一个小例子就是简单的奇偶校验码 P_3,它的奇偶校验矩阵是 $\boldsymbol{H} = [1 \quad 1 \quad 1]$。这个码有 4 个码字,都是偶校验。所有码字彼此之间的距离为 2。对于任意单个被删除比特,可以通过将它设定为其他两个比特的奇偶校验值来恢复。重复码也是最大可分隔距离码。

▷ **习题 13.13**[⌀] 你是否可以为 q 进制删除信道构造一个有 $M = N - K$ 个奇偶符号的 (N,K) 码,使得在一个长度为 N 的分组中有任意 M 个符号被删除时,译码器还可

① 对于二进制删除信道中的一个完美 u - 纠错码,冗余比特数一定是 $N - K = u$。

以恢复码字？[例：对于 $q=4$ 个符号的信道，存在一个 $(N,K)=(5,2)$ 码，它能够纠正任意 $M=3$ 的删除。]

对于 $q>2$ 的 q 进制删除信道，存在大量 MDS 码，其中最著名同时也是应用最广泛的是 Reed－Solomon 码。只要域的大小 q 大于分组长度 N，都可以找到任意码率的 MDS 分组码。（进一步的阅读材料可以参考 Lin 和 Costello(1983 年)。）

13.12 小 结

二进制对称信道中的香农码几乎总是可以纠正 fN 个错误，但它们不是 fN－纠错码。

码距不甚相干的原因

1. 香农极限表明，最好的码所能处理的最大噪声水平必须是一个距离受限译码器可处理的噪声水平的两倍。

2. 如果二进制对称信道的噪声水平 $f>1/4$，则根本不存在带距离受限译码器的码可以用于通信；但香农认为对于这种信道，还是存在好码的。

3. 级联表明即使距离是坏的，还是可以得到好的性能。

整个权重枚举函数与一个码是否是好码的问题关系密切。

好码和距离特性的关系将在习题 13.14 中进一步讨论。

13.13 进一步的习题

习题 13.14[3] 从一个线性 (N,K) 码 $\mathcal{C}$ 中选取码字 $\boldsymbol{t}$，并在噪声信道中传送它；接收信号是 $\boldsymbol{y}$。假设信道是诸如高斯信道之类的无记忆信道。给定一个假设的信道模型 $P(\boldsymbol{y}|\boldsymbol{t})$，存在两个译码问题。

码字译码问题的任务是给定接收信号时，推断所传输的是哪个码字 $\boldsymbol{t}$。

逐比特译码问题的任务是推断每一个传输比特 t_n 有多大可能是一个 1 而不是一个 0。

考虑这两种译码问题的最优译码器。通过证明以下定理，来证明最优逐比特译码器的差错概率与最优码字译码器的差错概率密切相关。

定理 13.1 *如果一个二进制线性码具有最小距离 $d_{\min}$，则对于任意已知信道，最优逐比特译码器的码字误比特概率 p_b、最大似然译码器的分组差错概率 p_B 之间的关系为*

$$p_B \geqslant p_b \geqslant \frac{1}{2}\frac{d_{\min}}{N}p_B \tag{13.39}$$

习题 13.15[1] (15,11)汉明码和(31,26)汉明码的最小距离各是多少?

▷ **习题 13.16**[2] 设一个码率为 1/3、$N=540$ 且 $M=360$ 的随机线性码的平均权重枚举函数是 $A(w)$,用基本原理估计 $A(w)$ 在 $w=1$ 处的值。

习题 13.17[2C] **最小距离大于 d_{GV} 的码** 由这个生成式矩阵生成一个相当好的(15,5)码,它基于测量所有 $\binom{5}{3}=10$ 个 3 - 元源比特组的奇偶性。

$$G=\begin{bmatrix}
1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & \cdot & \cdot & 1 & 1 & \cdot & 1\\
\cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & 1 & \cdot\\
\cdot & \cdot & 1 & \cdot & \cdot & 1 & \cdot & \cdot & 1 & 1 & \cdot & 1 & \cdot & 1 & 1\\
\cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & 1 & 1 & \cdot & 1 & \cdot & 1\\
\cdot & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & \cdot & \cdot & 1 & 1 & \cdot & 1 & \cdot
\end{bmatrix} \tag{13.40}$$

求解这个码的最小距离和权重枚举函数。

习题 13.18[2C] 求解“五边形”低密度奇偶校验码(如图 13.17 所示)的最小距离,其奇偶校验矩阵为

$$H=\left[\begin{array}{ccccc:ccccc:ccccc}
1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
1 & 1 & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot\\
\hdashline
\cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & 1\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & 1 & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1
\end{array}\right] \tag{13.41}$$

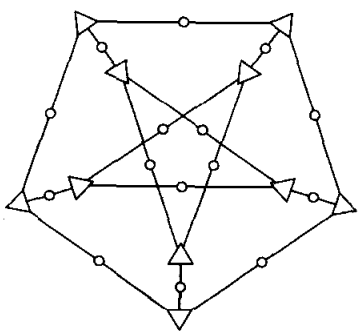


图 13.17 有 15 比特结点(圆圈)和 10 个奇偶校验结点(三角形)的五边形低密度奇偶校验码示意图

证明10行中的9行是独立的，所以码具有参数 $N=15, K=6$。利用计算机求出它的权重枚举函数。

▷**习题 13.19**[mC] 重复用于得出图13.12的计算。检验命题：可纠正的最高噪声水平为0.058 8。研究其他可选级联码序列。是否可以找到更好的一种级联码序列——这里"更好"的意思是指它具有更高的渐近码率 R 或可以容忍一个更高的噪声水平 f？



习题 13.20[s] 使用以下计数论点，研究用线性分组码达到香农极限的可能性。假设一个大的分组长度为 N、码率为 $R=K/N$ 的线性码。码的奇偶校验矩阵 $\boldsymbol{H}$ 有 $M=N-K$ 行。假设码的最优译码器（它求解校正子译码问题 $\boldsymbol{Hn}=\boldsymbol{z}$）允许以翻转概率 f 在二进制对称信道上进行可靠通信。

存在多少个"典型"噪声向量 $\boldsymbol{n}$？

大概有多少个不同的校正子 $\boldsymbol{z}$？

既然 $\boldsymbol{n}$ 是由最优译码器从 $\boldsymbol{z}$ 中可靠地推导而来的，校正子的数目必须大于或等于典型噪声向量的数目。由此可得知关于已知 f 时最大可能码率 R 值的何种信息？

▷**习题 13.21**[s] 线性二进制码使用等概率的输入符号0和1，隐含地将信道视做一个对称信道。如果实际上信道是一个高度非对称信道，研究这种假设会导致多大的通信速率损失。以Z信道为例。采用对称输入时最大可能通信速率比信道的容量小多少？[答案：约为6%。]

习题 13.22[s] 证明具有"很坏"距离的码是"坏"码，如第11.4节所定义。

习题 13.23[s] 通过删余可以从一个线性码获得另一个码。删余的意思是从每个码字中删除一个已定义的比特集合。删余使得 (N,K) 码变成 (N',K) 码，其中 $N'<N$。

从旧码生成新的线性码的另一种方法是缩短。缩短的意思是强制一个已定义比特集合为0，然后将它们从码字中删除。典型地，如果缩短1比特，就会丢失一半的码字。缩短通常会使一个 (N,K) 码变成一个 (N',K') 码，其中 $N-N'=K-K'$。

从两个旧码中产生一个新的线性码的另外一种方法是取两个码的交集：如果一个码字在两个旧码中都出现，该码字在新码中才被保留下来。

讨论删余、缩短以及交集对一个码的距离的影响。通过这三种处理中的每一种，是否能将具有坏距离的一族码转换成具有好距离的一族码，或将具有好距离的一族码转换成具有坏距离的一族码。

习题 13.24[s] **Todd Ebert的"帽子难题"**

3个游戏参与者进入一个房间，并且每个人的头上戴着一顶红色或蓝色的帽子。每顶帽子的颜色是由抛掷硬币决定的，且硬币的一次抛掷结果对其他的没有影响。每个人都可以看到其他人所戴帽子的颜色，但是看不到自己的。

除了小组进入房间之前的策略协商以外，不允许任何类型的通信。一旦他们有机会看到别人帽子的颜色，游戏参与者必须同时猜出他们自己帽子的颜色或者放弃。

如果至少一位游戏参与者猜对并且没有人猜错时，这个小组就可以共享300万美元的奖金。①

同样的游戏也可以由任意多的人来参与。一个一般性的问题是为小组找到一种策略来使它赢得奖金的机会最大化。求组大小为**3**和**7**时的最佳策略。

[提示：如果你能找到**3**个和**7**个的最佳策略，也许可以求出**15**个人的最佳策略。]

习题13.25[♂] 估计有多少种二进制低密度奇偶校验码具有自正交的对偶码。[注意：既然几乎所有低密度奇偶校验码是“好的”，而一个包含其对偶码的低密度奇偶校验码一定是“坏的”，所以预计这个数目不会太大。]

习题13.26[☙] 在图13.15中，绘制了权重为d的单一码字对应的差错概率和二进制对称信道的噪声水平f的函数关系图。对于高斯信道情况，也绘制了一张类似的图，表示权重为d的单一码字对应的差错概率和速率补偿信噪比E_b/N_0的函数关系图。因为E_b/N_0依赖于码率，所以你必须选择一个码率。选择$R=1/2,2/3,3/4$或$5/6$。

13.14 习题解答

习题13.4的解答 分组差错概率的主导阶次项为$p_B=\frac{3}{N}\binom{N}{2}f^2$。

习题13.7的解答 如果一个二进制向量具有偶权重，即有偶数个1，则该向量与其自身正交。

习题13.11的解答 自对偶码有两个等价的奇偶校验矩阵，$\boldsymbol{H}_1=\boldsymbol{G}=[\boldsymbol{I}_K|\boldsymbol{P}^{\mathrm{T}}]$和$\boldsymbol{H}_2=[\boldsymbol{P}|\boldsymbol{I}_K]$；它们通过行加法必须是彼此等价的，即存在一个矩阵$\boldsymbol{U}$使得$\boldsymbol{UH}_2=\boldsymbol{H}_1$，所以

$$[\boldsymbol{UP}|\boldsymbol{UI}_K]=[\boldsymbol{I}_K|\boldsymbol{P}^{\mathrm{T}}] \tag{13.42}$$

由这个等式的右边，有$\boldsymbol{U}=\boldsymbol{P}^{\mathrm{T}}$，所以左边变为

$$\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}=\boldsymbol{I}_K \tag{13.43}$$

因此，如果生成式矩阵为$\boldsymbol{G}=[\boldsymbol{I}_K|\boldsymbol{P}^{\mathrm{T}}]$的一个码是自对偶的，则$\boldsymbol{P}$是一个模2的正交矩阵，反之亦然。

习题13.12的解答 (8,4)和(7,4)码是密切相关的。(8,4)码的奇偶校验矩阵为

① 如果你已经知道“帽子难题”，可以尝试使用“苏格兰版本”规则：只有所有选手都猜对了，这个小组才能获奖。在“改良苏格兰版本”规则中，所有选手都必须猜对，且可以猜两轮。在第一轮做出猜测的选手离开房子，其余人在第二轮必须做出猜测。一支队伍应该采用什么策略才能使其获奖机会最大化？

$$\boldsymbol{H}=[\boldsymbol{P}|\boldsymbol{I}_4]=\left[\begin{array}{cccc|cccc}0&1&1&1&1&0&0&0\\1&0&1&1&0&1&0&0\\1&1&0&1&0&0&1&0\\1&1&1&0&0&0&0&1\end{array}\right] \tag{13.44}$$

可通过下面两种方法来获得该码:(a) 添加一个额外的奇偶校验比特,它是(7,4)汉明码中所有7个比特的奇偶校验值;(b) 对前面4个比特重新排序。

习题13.13的解答 如果有 $M=N-K$ 个奇偶符号的(N,K)码具有这样的特性:译码器在长度为 N 的分组中任意 M 个符号被删除时,若能恢复码字,则该码叫做最大可分隔距离码(MDS)。

除了重复码和简单的奇偶码,不存在MDS二进制码。当 $q>2$ 时,可以找到某些MDS码。

作为一个简单的例子,这里给出用于八进制删除信道的(9,2)码。这个码根据附录C.1中 $GF(8)$ 的乘法和加法规则定义。输入符号表中的元素为$\{0,1,A,B,C,D,E,F\}$,且码的生成式矩阵为

$$\boldsymbol{G}=\left[\begin{array}{ccccccccc}1&0&1&A&B&C&D&E&F\\0&1&1&1&1&1&1&1&1\end{array}\right] \tag{13.45}$$

所得的64个码字为

000000000	011111111	0AAAAAAAA	0BBBBBBBB	0CCCCCCCC	0DDDDDDDD	0EEEEEEEE	0FFFFFFFF
101ABCDEF	110BADCFE	1AB01EFCD	1BA10FEDC	1CDEF01AB	1DCFE10BA	1EFCDAB01	1FEDCBA10
A0ACEB1FD	A1BDFA0EC	AA0EC1BDF	AB1FD0ACE	ACE0AFDB1	ADF1BECA0	AECA0DF1B	AFDB1CE0A
B0BEDFC1A	B1AFCED0B	BA1CFDEB0	BB0DECFA1	BCFA1B0DE	BDEB0A1CF	BED0B1AFC	BFC1A0BED
C0CBFEAD1	C1DAEFBC0	CAE1DC0FB	CBF0CD1EA	CC0FBAE1D	CD1EABF0C	CEAD10CBF	CFBC01DAE
D0D1CAFBE	D1C0DBEAF	DAFBE0D1C	DBEAF1C0D	DC1D0EBFA	DD0C1FAEB	DEBFAC1D0	DFAEBD0C1
E0EF1DBAC	E1FE0CABD	EACDBF10E	EBDCAE01F	ECABD1FE0	EDBAC0EF1	EE01FBDCA	EF10EACDB
F0FDA1ECB	F1ECB0FDA	FADF0BCE1	FBCE1ADF0	FCB1EDA0F	FDA0FCB1E	FE1BCF0AD	FF0ADE1BC

习题13.14的解答 **定理快速而粗略的证明** 设 $\boldsymbol{x}$ 表示重构码字与传输码字之间的差异。对于任意已知信道输出 $\boldsymbol{r}$,存在 $\boldsymbol{x}$ 的一个后验分布。仅当向量 $\boldsymbol{x}$ 属于该码时,这个后验分布才是正的;下面的求和都是在码字 $\boldsymbol{x}$ 上进行的。分组差错概率为

$$p_{\mathrm{B}}=\sum_{\boldsymbol{x}\neq 0}P(\boldsymbol{x}|\boldsymbol{r}) \tag{13.46}$$

对码字中所有比特取平均,所得的平均误比特概率为

$$p_{\mathrm{b}}=\sum_{\boldsymbol{x}\neq 0}P(\boldsymbol{x}|\boldsymbol{r})\frac{w(\boldsymbol{x})}{N} \tag{13.47}$$

其中 $w(\boldsymbol{x})$ 是码字 $\boldsymbol{x}$ 的权重。非零码字的权重满足

$$1\geqslant\frac{w(\boldsymbol{x})}{N}\geqslant\frac{d_{\min}}{N} \tag{13.48}$$

将不等式(13.48)代入定义式(13.46,13.47),得到

$$p_{\mathrm{B}}\geqslant p_{\mathrm{b}}\geqslant\frac{d_{\min}}{N}p_{\mathrm{B}} \tag{13.49}$$

右边的项比式(13.39)所给结果大两倍。为了使证明更加严密,将结果弱化了一点。

仔细的证明 该定理将最优分组译码算法和最优逐比特译码算法的性能联系起来。

可以引入另一对译码算法，称为分组猜测译码器和比特猜测译码器。这两种算法的思想类似于最优分组译码器和最优逐比特译码器，但是分析起来更容易。

现在定义这些译码器。设 $\boldsymbol{x}$ 表示推导出来的码字，对于任意给定的码，相应译码器的定义如下。

最优分组译码器返回使后验概率 $P(\boldsymbol{x}|\boldsymbol{r})$ 最大化的码字 $\boldsymbol{x}$，这个概率与似然 $P(\boldsymbol{r}|\boldsymbol{x})$ 成正比。

这个译码器的差错概率记做 p_{B}。

最优比特译码器对 N 个比特中的每个比特 x_n，返回使后验概率 $P(x_n = a \mid \boldsymbol{r}) = \sum_{\boldsymbol{x}} P(\boldsymbol{x} \mid \boldsymbol{r})\, 1[x_n = a]$ 最大化的 a 值。

这个译码器的差错概率记做 p_{b}。

分组猜测译码器返回一个随机码字 $\boldsymbol{x}$，其概率分布由后验概率 $P(\boldsymbol{x}|\boldsymbol{r})$ 给出。

这个译码器的差错概率记做 $p_{\mathrm{B}}^{\mathrm{G}}$。

比特猜测译码器对 N 个比特中的每个比特 x_n，返回从概率分布 $P(x_n = a|\boldsymbol{r})$ 中得到的一个随机比特。

这个译码器的差错概率记做 $p_{\mathrm{b}}^{\mathrm{G}}$。

定理表明最优误比特概率 p_{b} 的上界为 p_{B}，下界为 p_{B} 的一个给定乘数——式(13.39)。

式(13.39)左边的不等式显然成立——如果一个分组是正确的，则组成该分组的所有比特也是正确的；所以，如果最优分组译码器性能比最优比特译码器的性能要好，就可以从分组译码器中得到一个更好的比特译码器。

通过确定下列公式来证明右边的不等式。

(a) 比特猜测译码器与最优比特译码器几乎一样好。

$$p_{\mathrm{b}}^{\mathrm{G}} \leqslant 2p_{\mathrm{b}} \tag{13.50}$$

(b) 比特猜测译码器的差错概率与分组猜测译码器的差错概率通过下式相关联。

$$p_{\mathrm{b}}^{\mathrm{G}} \geqslant \frac{d_{\min}}{N} p_{\mathrm{B}}^{\mathrm{G}} \tag{13.51}$$

因为 $p_{\mathrm{B}}^{\mathrm{G}} \geqslant p_{\mathrm{B}}$，于是有

$$p_{\mathrm{b}} > \frac{1}{2} p_{\mathrm{b}}^{\mathrm{G}} \geqslant \frac{1}{2}\, \frac{d_{\min}}{N} p_{\mathrm{B}}^{\mathrm{G}} \geqslant \frac{1}{2}\, \frac{d_{\min}}{N} p_{\mathrm{B}} \tag{13.52}$$

现在来证明两个引理。

猜测的近似最优性 首先考虑单一比特情况，其后验概率为 $\{p_0, p_1\}$。最优比特译码器的差错概率为

$$P^{\text{最优}} = \min(p_0, p_1) \tag{13.53}$$

猜测译码器从 0 和 1 中选取。事实也服从同样的概率分布。猜测与事实相匹配的概

率为 $p_0^2+p_1^2$；两者失配的概率就是猜测差错概率

$$P^{\text{猜测}}=2p_0p_1\leqslant 2\min(p_0,p_1)=2P^{\text{最优}} \tag{13.54}$$

因为 p_b^G 是许多这种差错概率 $P^{\text{猜测}}$ 的平均值，且 p_b 是相应的最优差错概率 $P^{\text{最优}}$ 的平均值，所以可得 p_b^G 和 p_b 二者之间的关系式(13.50)。 □

误比特概率与分组差错概率之间的关系 比特猜测和分组猜测译码器可以在一个系统中联合使用：可以通过从联合分布 $P(x_n,\boldsymbol{x}|\boldsymbol{r})$ 中提取一个样本 $(x_n,\boldsymbol{x})$ 并丢弃 $\boldsymbol{x}$ 值，从而得到边缘分布 $P(x_n|\boldsymbol{r})$ 的一个样本 x_n。

能够区分这两种情况：被丢弃的 $\boldsymbol{x}$ 值是或不是一个正确的码字。比特猜测译码器的误比特概率可以被写成两项之和

$$\begin{aligned}p_b^G&=P(\boldsymbol{x}\text{ 正确})P(\text{比特差错}|\boldsymbol{x}\text{ 正确})\\&\quad+P(\boldsymbol{x}\text{ 不正确})P(\text{比特差错}|\boldsymbol{x}\text{ 不正确})\\&=0+p_B^GP(\text{比特差错}|\boldsymbol{x}\text{ 不正确})\end{aligned}$$

现在，只要所猜测的 $\boldsymbol{x}$ 不正确，则真的 $\boldsymbol{x}$ 必须至少与它相差 d 比特，所以这些情况下的误比特概率至少是 d/N。所以

$$p_b^G\geqslant\frac{d}{N}p_B^G$$

证毕。 □

习题 13.20 的解答 “典型”噪声向量 $\boldsymbol{n}$ 的数量约为 $2^{NH_2(f)}$。不同校正子 $\boldsymbol{z}$ 的数量为 2^M。所以可靠通信意味着

$$M\geqslant NH_2(f) \tag{13.55}$$

或者，写成码率 $R=1-M/N$ 的形式有

$$R\leqslant 1-H_2(f) \tag{13.56}$$

这是与信道容量完全相符的一个边界。

这个论点在下一章会变成一个证明。

习题 13.24 的解答 对于3名游戏参与者的情况，小组有3/4的时候可能取胜。

有3/4的时候，两个人戴着同样颜色的帽子而第3个人的帽子是相反的颜色。小组使用以下策略才能够在每次出现这种情况时都获胜：每个人看其他两个人帽子的颜色。如果这两顶帽子的颜色不同，则放弃。如果它们颜色相同，则猜测自己的帽子是相反的颜色。

用这种方法，每当帽子的颜色分布为2和1时，一个人将猜测正确，而其他人则放弃，那么小组就会在游戏中获胜。但是当所有帽子都是同一种颜色时，所有3个人都会猜错，小组也就失败了。

当其中任何一个特定的人猜测一种颜色时，他仅有50:50的机会猜对。小组75%的时候能获胜的原因在于，他们的策略保证了当游戏参与者猜错时，很多人都猜错了。

对于有更多人参与游戏的情况，目标是确保大多数时间没有人出错，偶尔所有人都同时猜错。在7人参与的游戏中，有一种可以使得每8次游戏中小组赢7次的策

略。在有 15 人参与的游戏中,小组可以每玩 16 次就赢 15 次。如果你还没有计算出 7 人组和 15 人组的获胜策略,笔者建议你从下面这个角度仔细思考一下 3 个人玩游戏时的解答:分析获胜和失败状态在三维立方体上的位置,然后再横向思考一下。

如果参与游戏的人数 N 为 2^r-1,则可使用长度为 N 的一个汉明码来定义最优策略,赢得奖金的概率是 $N/(N+1)$。每人用一个数 $n\in 1,\cdots,N$ 来定义。两种颜色被映射为 0 和 1。他们帽子的任意状态可以被视做从一个二进制信道出来的一个接收向量。长度为 N 的随机二进制向量是汉明码中的一个码字的概率为 $1/(N+1)$,或者刚好与一个码字相差 1 比特。每个人看到其他所有比特,并考虑他的比特是否可被设为一种颜色,使得该状态为一个码字(使用汉明码的译码器可以推断出来)。如果可以,则这个人就猜测他的帽子是另外一种颜色。如果此时的状态实际上是一个码字,则所有参与游戏的人都会做出猜测并且全部猜错。如果此时的状态不是一个码字,则只有一个人会猜测,并且他的猜测是正确的。如果采用(7,4)汉明码的循环表示,则训练 7 个参与者遵循最优策略是很容易的。

关于第 14 章

在这一章中，把到目前为止遇到过的几种思想用一个简短的证明很好地统一在一起。将同时证明香农噪声信道编码定理（对二进制对称信道）和他的信源编码定理（对二进制信源）。尽管这个证明与这本书中前面很多章节有联系，但是并不要求一定要阅读那些章节的所有材料。

在噪声信道编码方面，我们的证明比第 10 章所给出的证明更具建设性。那里，已证明了几乎任意随机码都是“很好的”。这里将证明几乎任意线性码都是“很好的”。我们将利用典型集合的概念（第 4 章和第 10 章），并借用上一章对随机线性码的权重枚举函数的计算（第 13.5 节）。

在信源编码方面，我们的证明将表明随机线性散列函数可用于压缩可压缩的二进制信源，从而与第 12 章关联起来。

第 14 章

存在很好的线性码

在这一章，用一个计算来同时证明二进制对称信道的信源编码定理和噪声信道编码定理。

顺便提一下，这个证明适用于更一般的信道模型，而不仅仅适用于二进制对称信道。例如，可以改写证明，使之适用于非二进制输出的信道、时变信道和存储信道，只要它们的二进制输入满足一个对称特性，与第 10.6 节比较。

14.1 信源编码和噪声信道编码定理的同时证明

考虑一个线性纠错码，其二进制奇偶校验矩阵 $\boldsymbol{H}$ 有 M 行 N 列。在稍后的证明中，将会增大 N 和 M 的值，但保持 $M \propto N$。该码的码率满足

$$R \geqslant 1 - \frac{M}{N} \tag{14.1}$$

如果 $\boldsymbol{H}$ 的所有行独立，则这是一个等式，$R = 1 - M/N$。接下来，将假设等式成立。热心的读者可以计算出一个随机二进制矩阵 $\boldsymbol{H}$ 的期望秩（它很接近于 M），并考察差值（M - 秩）对这个证明的其他部分的影响（它是可忽略的）。

选择满足如下条件的一个码字 $\boldsymbol{t}$。

$$\boldsymbol{Ht} = \boldsymbol{0} \bmod 2 \tag{14.2}$$

一个二进制对称信道加入噪声 $\boldsymbol{x}$，并给出接收信号

$$\boldsymbol{r} = \boldsymbol{t} + \boldsymbol{x} \bmod 2 \tag{14.3}$$

接收方的目的是采用校正子译码方法从 $\boldsymbol{r}$ 中推断 $\boldsymbol{t}$ 和 $\boldsymbol{x}$。校正子译码首先在第 1.2 节中做过介绍。接收方计算校正子

$$\boldsymbol{z} = \boldsymbol{Hr} \bmod 2 = \boldsymbol{Ht} + \boldsymbol{Hx} \bmod 2 = \boldsymbol{Hx} \bmod 2 \tag{14.4}$$

校正子仅取决于噪声 $\boldsymbol{x}$，而译码的问题是求出满足如下条件的最可能的 $\boldsymbol{x}$。

$$\boldsymbol{Hx} = \boldsymbol{z} \bmod 2 \tag{14.5}$$

从 $\boldsymbol{r}$ 中减去这个噪声向量的最佳估计 $\hat{\boldsymbol{x}}$，就给出了对 $\boldsymbol{t}$ 的最好猜测。目标是要证明，只要 $R < 1 - H(x) = 1 - H_2(f)$，其中 f 是二进制对称信道的翻转概率，对于随机的 $\boldsymbol{H}$，这个校正子译码问题的最优译码器随着 N 的增加会有一个逐渐降为零的差错概率。

通过研究求解该译码问题的次优策略，来证明这个结论。无论是最优译码器还

是这种典型集译码器都不容易实现，但是典型集译码器更容易分析。[①] 典型集译码器检查噪声向量的典型集 T，即满足 $\log 1/P(\boldsymbol{x}') \simeq NH(X)$ 的那些噪声向量 $\boldsymbol{x}'$ 的集合，并查看这些典型向量 $\boldsymbol{x}'$ 中是否有向量满足所观察到的校正子

$$\boldsymbol{H}\boldsymbol{x}' = z \tag{14.6}$$

如果恰好有一个典型向量 $\boldsymbol{x}'$ 满足条件，那么典型集译码器就给出那个向量作为假设的噪声向量。如果没有典型向量匹配所观察到的校正子，或者不止一个向量匹配，那么典型集译码器报告出现了一个错误。

对于一个已知矩阵 $\boldsymbol{H}$，典型集译码器的差错概率可以写成两项之和

$$P_{\text{TS}|\boldsymbol{H}} = P^{(I)} + P^{(II)}_{\text{TS}|\boldsymbol{H}} \tag{14.7}$$

其中 $P^{(I)}$ 是真实噪声向量 $\boldsymbol{x}$ 本身不典型的概率，$P^{(II)}_{\text{TS}|\boldsymbol{H}}$ 是真实噪声 $\boldsymbol{x}$ 典型但至少还有一个典型向量与之冲突的概率。正如第一次研究典型集时所证明的那样（第 4 章），随着 N 的增加，第一个概率趋近于零。集中考虑第 2 个概率。先扼要概述一下：假设一个真实的噪声向量 $\boldsymbol{x}$；如果任意一个与 $\boldsymbol{x}$ 不同的典型噪声向量 $\boldsymbol{x}'$ 满足 $\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0$，则出现了一个错误。使用真值函数

$$\mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0] \tag{14.8}$$

如果命题 $\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0$ 为真，则它的值为 1，否则为 0。

当噪声为 $\boldsymbol{x}$，可以限定类型 II 所产生的错误数

$$[\text{已知 } \boldsymbol{x} \text{ 和 } \boldsymbol{H} \text{ 的错误数}] \leqslant \sum_{\boldsymbol{x}':\substack{\boldsymbol{x}'\in T\\ \boldsymbol{x}'\neq\boldsymbol{x}}} \mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0] \tag{14.9}$$

错误的数量要么为 0 要么为 1；当几个典型噪声向量具有同一校正子时，右边的和可能超过 1。[②]

现在可以通过在 $\boldsymbol{x}$ 上取平均值，写出第 II 类错误的概率

$$P^{(II)}_{\text{TS}|\boldsymbol{H}} \leqslant \sum_{\boldsymbol{x}\in T} P(\boldsymbol{x}) \sum_{\boldsymbol{x}':\substack{\boldsymbol{x}'\in T\\ \boldsymbol{x}'\neq\boldsymbol{x}}} \mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0] \tag{14.10}$$

现在，可以通过在 $\boldsymbol{H}$ 上取平均值，来求所有线性码中的第 II 类差错概率的平均值。将要证明第 II 类错误的平均概率趋近于零，并进而证明存在具有趋近于零的差错概率的线性码，事实上，几乎所有线性码都是很好的码。

用 $\langle\cdots\rangle_{\boldsymbol{H}}$ 表示在所有二进制矩阵 $\boldsymbol{H}$ 上取平均。第 II 类错误的平均概率为

$$\overline{P}^{(II)}_{\text{TS}} = \sum_{\boldsymbol{H}} P(\boldsymbol{H}) P^{(II)}_{\text{TS}|\boldsymbol{H}} = \langle P^{(II)}_{\text{TS}|\boldsymbol{H}} \rangle_{\boldsymbol{H}} \tag{14.11}$$

$$= \Big\langle \sum_{\boldsymbol{x}\in T} P(\boldsymbol{x}) \sum_{\boldsymbol{x}':\substack{\boldsymbol{x}'\in T\\ \boldsymbol{x}'\neq\boldsymbol{x}}} \mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0] \Big\rangle_{\boldsymbol{H}} \tag{14.12}$$

$$= \sum_{\boldsymbol{x}\in T} P(\boldsymbol{x}) \sum_{\boldsymbol{x}':\substack{\boldsymbol{x}'\in T\\ \boldsymbol{x}'\neq\boldsymbol{x}}} \langle \mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0] \rangle_{\boldsymbol{H}} \tag{14.13}$$

---

① 我们将忽略使典型集合定义严密的 ε 和 β。鼓励有兴趣的读者复习第 4.4 节并将细节补充到这个证明中。

② 式(14.9)是一个联合界。

在计算随机线性码的期望权重枚举函数时(第 13.5 节),已经得到了变量 $\langle \mathbb{1}[\boldsymbol{H}(\boldsymbol{x}'-\boldsymbol{x})=0]\rangle_{\boldsymbol{H}}$:对任意非零二进制向量$\boldsymbol{v}$,在所有矩阵 $\boldsymbol{H}$ 上取平均时 $\boldsymbol{Hv}=0$ 的概率为 2^{-M}。所以

$$\overline{P}_{\mathrm{TS}}^{(II)}=\Big(\sum_{x\in T}P(\boldsymbol{x})\Big)(|T|-1)2^{-M} \tag{14.14}$$

$$\leqslant |T|2^{-M} \tag{14.15}$$

其中$|T|$表示典型集的大小。回顾第 4 章的有关内容,可得知典型集约包括 $2^{NH(X)}$ 个噪声向量。因此

$$\overline{P}_{\mathrm{TS}}^{(II)}\leqslant 2^{NH(X)}2^{-M} \tag{14.16}$$

差错概率的这种边界随着 N 的增加要么趋近于零,要么呈指数级上升(记住,当 N 增加时,保持 M 与 N 成正比)。如果

$$H(X)<M/N \tag{14.17}$$

则这个概率将趋近于零。代入 $R=1-M/N$,可以得证二进制对称信道的噪声信道编码定理:对任意满足

$$R<1-H(X) \tag{14.18}$$

的码率 R,存在非常好的线性码。其中 $H(X)$是每比特信道噪声的熵。

习题 14.1[ㄗ] 对一个更一般的信道,重新进行证明。

14.2 用线性散列码进行数据压缩

刚刚做过的译码游戏也可视做一个解压缩游戏。物理世界从噪声源 $P(\boldsymbol{x})$中产生一个二进制噪声向量 $\boldsymbol{x}$。噪声具有冗余度(如果翻转概率不是 0.5)。用一个线性压缩器对它进行压缩,该压缩器将 N 比特的输入 $\boldsymbol{x}$(噪声)映射为 M 比特的输出 $\boldsymbol{z}$(校正子)。解压缩任务是从输出 $\boldsymbol{z}$ 中恢复输入 $\boldsymbol{x}$。压缩器的压缩率为

$$R_{\text{压缩器}}\equiv M/N \tag{14.19}$$

[这里,对压缩率定义中存在线性冗余的可能性并不关心。]刚得到了这样的结果:随着差错概率的减小,对于几乎任意 $\boldsymbol{H}$,只要满足 $H(X)<M/N$,就可以解决译码问题。由此,立即可以证明一个信源编码定理:

假定一个熵为 $H(X)$的二进制信源 X 和一个所要求的压缩率 $R>H(X)$,存在一个线性压缩器 $\boldsymbol{x}\to\boldsymbol{z}=\boldsymbol{Hx} \bmod 2$,其压缩率 M/N 等于所要求的压缩率 R,及一个相关的基本无损的解压缩器。

这个定理不仅对独立同等分布的符号的信源成立,而且对于定义典型集的任意信源都成立:例如,有存储器的信源和时变的信源;唯一的要求是,信源必须是各态历经的。

注意

证明码是好码的这种方法可以应用到其他线性码上,例如低密度奇偶校验码

(MacKay,1999b;Aji 等,2000 年)。对于每一个码,所需的只是它的期望权重枚举函数的一个近似。

第 15 章

有关信息论的更多习题

一直到本章结束，最令人兴奋的习题将进一步向你介绍关于信息论的思想。

关于信源编码与噪声信道的热身习题

▷ **习题 15.1**[⌀] 设 X 是 $\mathscr{A}_X=\{0,1\}$ 且 $\mathscr{P}_X=\{0.995,0.005\}$ 的一个总体。考虑采用分组码 X^{100} 的信源编码，为每个包含 3 个或更少个 1 的 $\boldsymbol{x}\in X^{100}$ 都指定一个不同的码字，并忽略其他 $\boldsymbol{x}$。

（a）如果指定的所有码字具有相同长度，为上面这个集合提供不同码字所需的最小长度是多少？

（b）计算得到一个将要忽略的 $\boldsymbol{x}$ 的概率。

▷ **习题 15.2**[⌀] 设 X 是 $\mathscr{P}_X=\{0.1,0.2,0.3,0.4\}$ 的一个总体。采用符号码 $\mathscr{C}=\{0001,001,01,1\}$ 对该总体进行编码。考虑与 $\boldsymbol{x}\in X^N$ 对应的码字，其中 N 很大。

（a）计算第 4 个传输比特的熵。

（b）已知第 3 个比特，计算第 4 个比特的条件熵。

（c）估计第 100 个比特的熵。

（d）已知第 99 比特，估计第 100 个比特的条件熵。



习题 15.3[⌀] 艾丽丝转动两个公平骰子，并记录所得的和。鲍勃的任务是询问一系列可以用“是”或“不是”来回答的问题，以便找出这个数。详细设计一个策略，达到可能的最小的平均问题数。

▷ **习题 15.4**[⌀] 如何通过一枚硬币在 3 个人中抽签？

▷ **习题 15.5**[⌀] 在一个魔术表演中，有 3 个参与者：魔术师、助手和志愿者。声称具有超常能力的助手在一个隔音的房间里。魔术师给志愿者 6 张空白卡片，其中 5 张白色，1 张蓝色。魔术师看着志愿者在每张卡片上写一个 1 ~ 100 之间的不同整数。志愿者自己保留蓝色卡片。魔术师将 5 张白色卡片按照某个顺序排列并传递给助手。然后助手就宣布在蓝色卡片上的数字。

这个魔术是怎样玩出来的？

▷ **习题 15.6**[⌀] 这个魔术是怎样完成的？

“有一副普通扑克牌，洗成随机顺序。按照你的意愿，从这副牌中任意选择 5 张。不要给我看这些牌。不，不要把它们给我：将这 5 张牌递给我的助手 Esmerelda。她可以看这些牌。

“现在,Esmerelda,给我看其中的4张牌。嗯…黑桃9,梅花6,红心4,方块10。那么剩下的那张牌肯定是黑桃皇后!”

可以用52张纸牌来做上面所描述的魔术。使用信息论求出可以完成这个魔术的纸牌张数的上限。

▷ **习题 15.7**[✍] 求概率序列 $\boldsymbol{p}=(p_1,p_2,\cdots)$,使得 $H(\boldsymbol{p})=\infty$。

▷ **习题 15.8**[✍] 考虑 $\mathcal{A}_X=\{\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d}\}$ 且 $\mathcal{P}_X=\{1/2,1/4,1/8,1/8\}$ 的离散无记忆信源。用这4个字母可以组成 $4^8=65\,536$ 个有8个字母的单词。当 $N=8$ 且 $\beta=0.1$ 时,求出典型集 $T_{N\beta}$(式(4.29))中这种单词的总数。

▷ **习题 15.9**[✍] 考虑信源 $\mathcal{A}_S=\{\mathrm{a},\mathrm{b},\mathrm{c},\mathrm{d},\mathrm{e}\}$,$\mathcal{P}_S=\{1/3,1/3,1/9,1/9,1/9\}$,信道的转移概率矩阵为

$$Q=\begin{bmatrix}1&0&0&0\\0&0&2/3&0\\0&1&0&1\\0&0&1/3&0\end{bmatrix} \tag{15.1}$$

注意信源符号表中有5个符号,但是信道符号表中 $\mathcal{A}_X=\mathcal{A}_Y=\{0,1,2,3\}$ 只有4个符号。假设信源以信道接收符号速率的3/4的速率产生符号。对于一个已知(很小的)$\varepsilon>0$,解释如何设计一个系统,使得该系统以每个信源符号的平均差错概率小于 ε 的可靠度,在信道上传输信源的输出。尽可能地明确。特别地,不要使用香农的噪声信道编码定理。

▷ **习题 15.10**[✍] 考虑一个二进制对称信道和一个码 $C=\{0000,0011,1100,1111\}$;假设4个码字的使用概率为 $\{1/2,1/8,1/8,1/4\}$。

将译码差错概率最小化的译码规则是什么?[最优译码规则取决于二进制对称信道的噪声水平 f。对于0~1/2之间的 f,给出每个范围的 f 值的译码规则。]



习题 15.11[✍] 对于具有如下转移概率

$$Q=\begin{bmatrix}1&0&0\\0&2/3&1/3\\0&1/3&2/3\end{bmatrix} \tag{15.2}$$

的3-输入、3-输出信道,求信道容量和最优输入分布。

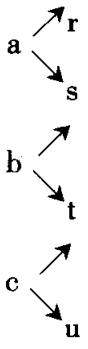


习题 15.12[✍] 信道 Q 的输入是一个大小为8比特的字。输出也是一个8比特的字。每次使用信道时,信道刚好翻转1个传输比特,但是接收方并不知道是哪一比特。其他7比特无误接收。所有8个比特都等可能地是那个被翻转的比特。推导这个信道的容量。

明确地描述一个编码器和译码器,以证明有可能在这个信道上可靠地(即,以零差错概率)以每周期5个比特进行通信。

▷ **习题 15.13**[♂]　输入 $x \in \{a,b,c\}$、输出 $y \in \{r,s,t,u\}$ 的信道具有条件概率矩阵

$$Q = \begin{bmatrix} 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 \end{bmatrix}$$



它的容量是多少？

▷ **习题 15.14**[♂]　一本书的封面上称为 ISBN 的 10 位数字包含了一个纠错码。这个数包括 9 个信源数字 $x_1, x_2, \cdots, x_9$（其中 $x_n \in \{0,1,\cdots,9\}$）和第 10 个用于校验的数字，其值由下式给出。

$$x_{10} = \left(\sum_{n=1}^{9} n x_n \right) \bmod 11$$

这里 $x_{10} \in \{0,1,\cdots,9,10\}$。如果 $x_{10} = 10$，那么第 10 位数字用罗马数 X 表示。

证明一个有效 ISBN（如表 15.1 所示）满足

$$\left(\sum_{n=1}^{10} n x_n \right) \bmod 11 = 0$$

假设通过一个不可靠的人与人之间的信道传输 ISBN，它有时会更改数字，有时会重排数字。

表 15.1　一些有效的 ISBN。[采用半字线是为了提高易读性。]

0 - 521 - 64298 - 1
1 - 010 - 00000 - 4

证明这个码可用来检测（但不能纠正）所有因为 10 位数字中任意一位被改变而引起的错误。（例如，1 - 010 - 00000 - 4 被改成了 1 - 010 - 00080 - 4）。

证明这个码可用来检测所有因 10 位中的任意相邻两位数字顺序颠倒而引起的错误（例如，1 - 010 - 00000 - 4 被改成了 1 - 100 - 00000 - 4）。

还能检测出哪些非相邻数字对颠倒顺序的错误呢？

如果将第 10 位数字定义为

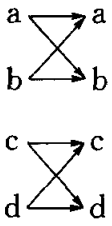
$$x_{10} = \left(\sum_{n=1}^{9} n x_n \right) \bmod 10$$

为什么所得的码不能像 ISBN 那样工作呢？（讨论检测单一数字错误和数字颠倒顺序错误。）



习题 15.15[♂]　一个信道的输入 x 和输出 y 之间的转移概率矩阵为

$$Q=\begin{bmatrix} 1-f & f & 0 & 0 \\ f & 1-f & 0 & 0 \\ 0 & 0 & 1-g & g \\ 0 & 0 & g & 1-g \end{bmatrix}$$



假设一个输入分布具有形式

$$\mathcal{P}_X=\left\{\frac{p}{2},\frac{p}{2},\frac{1-p}{2},\frac{1-p}{2}\right\}$$

写出输出的熵 $H(Y)$，及已知输入时输出的条件熵 $H(Y|X)$。

证明最优输入分布由下式给出。

$$p=\frac{1}{1+2^{-H_2(g)+H_2(f)}}$$

其中 $H_2(f)=f\log_2\frac{1}{f}+(1-f)\log_2\frac{1}{(1-f)}$。①

写出在 $f=1/2, g=0$ 时的最优输入分布和信道容量，并对你的答案进行论述。

▷ **习题 15.16**[ρ] 一个检测码（它能够可靠地检测出数据的一个分组被破坏）和一个纠错码（能够检测并纠正错误）所需的冗余度有什么不同？

来自信息论的更多传奇故事

以下习题使你有机会自己得出信息论中一些更令人吃惊的结论。

习题 15.17[ρ] **来自相关信源的信息的通信** 设想要通过单向无噪声信道（图 15.2(a)）从两个数据源 $X^{(A)}$ 和 $X^{(B)}$ 传输数据到一个中心点 C。信号 $x^{(A)}$ 和 $x^{(B)}$ 是强相关的，所以它们的联合信息量只比它们其中任一个的边缘信息量大一点。例如，C 是一个气象校对员，他希望接收一组报告，说 Allerton 是否在下雨（$x^{(A)}$），以及 Bognor 是否在下雨（$x^{(B)}$）。$x^{(A)}$ 和 $x^{(B)}$ 的联合概率可能是

$P(x^{(A)},x^{(B)})$：

$x^{(B)}$ \ $x^{(A)}$	0	1
0	0.49	0.01
1	0.01	0.49

(15.3)

天气校对员希望确切地知道 $x^{(A)}$ 和 $x^{(B)}$ 的 N 个连续值，但是，因为他要为所接收到的每一比特信息付费，所以他对如何避免从信源 A 购买 N 比特再从信源 B 购买 N 比特的可能性感兴趣。假设从这个分布中重复地生成变量 $x^{(A)}$ 和 $x^{(B)}$，是否能以码率 R_A

① 记住 $\frac{\mathrm{d}}{\mathrm{d}p}H_2(p)=\log_2\frac{1-p}{p}$。

和 R_B 对它们进行编码，使得 C 可以重构所有变量，而在每个周期里从两条路径上传输的信息速率之和不超过 2 比特？

应该给出如图 15.2 所示的答案。在两个信源 $X^{(A)}$ 和 $X^{(B)}$ 相关的一般情况下，只要 $X^{(A)}$ 的速率 R_A 超过 $H(X^{(A)}|X^{(B)})$，$X^{(B)}$ 的速率 R_B 超过 $H(X^{(B)}|X^{(A)})$，且总的信息速率 R_A+R_B 超过联合熵 $H(X^{(A)},X^{(B)})$（Slepian 和 Wolf，1973 年），就能在两个发送端找到编码以实现 $X^{(A)}$、$X^{(B)}$ 与 C 之间的可靠通信。

因此对于上述 $x^{(A)}$ 和 $x^{(B)}$ 的情况，每个发送端必须以大于 $H_2(0.02)=0.14$ 比特的速率传输，且总速率 R_A+R_B 必须大于 1.14 比特，例如 $R_A=0.6$，$R_B=0.6$。存在可以达到这些速率的码。你的任务是证明为什么会这样。

如果要求其中一个信源以纯文本方式发送，$t^{(B)}=x^{(B)}$，而另一个信源被编码，试给出显式解。

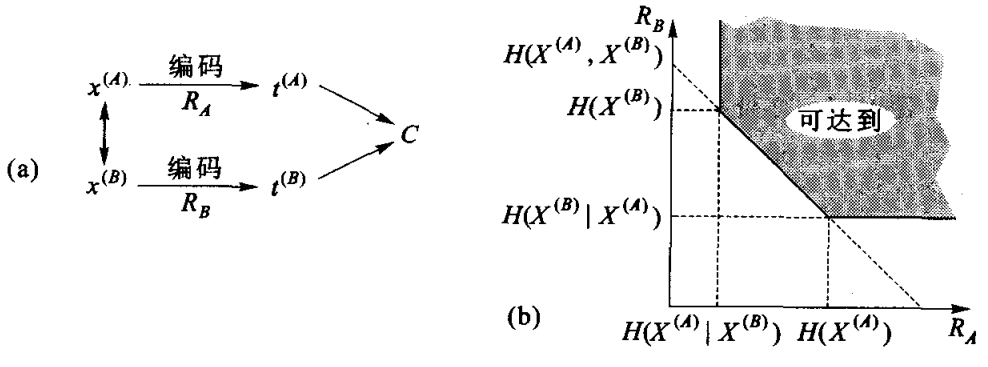


图 15.2　对来自相关信源的信息进行通信。(a) $x^{(A)}$ 和 $x^{(B)}$ 是相关信源（相关性用虚线箭头表示）。使用码率为 R_A 和 R_B 的码，将每个变量的一串数值编码成 $t^{(A)}$ 和 $t^{(B)}$，并在无噪声信道上传输给接收方 C。(b) 可达到的码率区域。即使 $R_A<H(X^{(A)})$ 和 $R_B<H(X^{(B)})$，还是可以无误地传输两串数值。

习题 15.18[小] 多址接入信道。考虑有两组输入和一组输出的信道——例如，一段共享的电话线（如图 15.3(a)所示）。具有两个二进制输入 $x^{(A)}$、$x^{(B)}$ 和一个三进制输出 y 的简单模型系统，其中 y 等于两个输入的算术和（取值为 0、1 或 2）。不存在噪声。用户 A 和 B 之间不能相互通信，并且他们听不到信道的输出。如果输出是一个 0，则接收方能确定两个输入均为 0；如果输出是一个 2，则接收方也能确定两个输入均为 1。但如果输出是 1，输入端的状态就可能是(0,1)或(1,0)。用户 A 和 B 应该如何使用信道才可从接收信号中推断出他们所发送的消息呢？A 和 B 的通信速率有多快？

显然，从 A 和 B 到接收方的总信息速率不可能是 2 比特。另一方面，很容易达到 1 比特的总信息速率 R_A+R_B。当 $R_A+R_B>1$ 时，是否可实现速率为 (R_A,R_B) 的可

靠通信呢?

答案如图 15.3 所示。

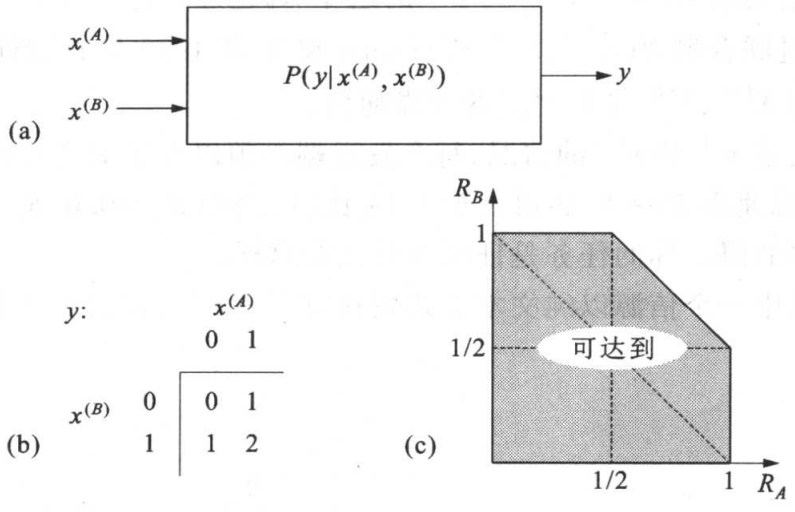


图 15.3 多址接入信道。(a) 2 发送方 - 1 接收方的一般多址信道。(b) 输出等于两输入之和的二进制多址信道。(c) 可达区域。

Ratzer 和 MacKay(2003 年)给出了多用户信道的一些实用码。

习题 15.19[25] **广播信道** 一个广播信道(如图 15.4 所示)由一个发送方和两个或多个接收方组成。用一个条件分布 $Q(y^{(A)},y^{(B)}|x)$ 定义信道的属性。(假设信道是无记忆的。)任务是增加一个编码器和两个译码器,以便两个接收方能可靠地接收速率为 R_0 的共同消息,接收方 A 能接收速率为 R_A 的单独信息,接收方 B 能接收速率为 R_B 的单独信息。广播信道的容量区域是可达速率三元组 (R_0,R_A,R_B) 的集合的凸包。

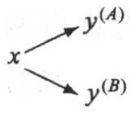


图 15.4 广播信道。x 为信道输入,$y^{(A)}$ 和 $y^{(B)}$ 为输出。

这种信道的一个简单基准是分时(时分信令),如图 15.5 所示。如果单独考虑两个信道的容量分别为 $C^{(A)}$ 和 $C^{(B)}$,则通过将传输时间的 ϕ_A 部分用于信道 A,$\phi_B = 1-\phi_A$ 用于信道 B,可以得到 $(R_0,R_A,R_B)=(0,\phi_A C^{(A)},\phi_B C^{(B)})$。

然而我们可以做得更好。作为一个类比,设想同时与一个美国人和一个白俄罗斯人交谈;你的英语和白俄罗斯语都很流利,但是你的两个谈话对象彼此不懂对方的

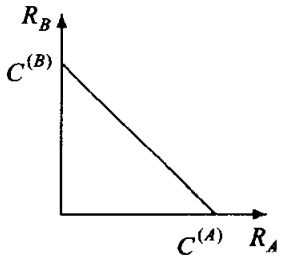


图 15.5　通过简单分时可达的速率

语言。如果每个谈话对象能够区分一个单词是否属于他们自己的语言，那么可以向两个谈话对象传输一个额外的二进制文件，通过使用它的比特来确定下一个被传输的字是来自美国源文本还是白俄罗斯源文本。每个谈话对象将他们所理解的单词串接起来以接收他们各自的信息，并能恢复二进制字符串。

一个广播信道的例子是具有共同输入的两个二进制对称信道。这个信道的两部分的翻转概率分别为 f_A 和 f_B。假设 A 有性能较好的子信道，即 $f_A < f_B < 1/2$。[与之密切相关的一个信道是“性能降低的”广播信道，它的条件概率使得随机变量具有马尔科夫链的结构

$$x \to y^{(A)} \to y^{(B)} \tag{15.4}$$

即，$y^{(B)}$ 是 $y^{(A)}$ 的性能降低版。]在这个特例中，结果证明与接收方 B 通信的任何信息都可在接收方 A 中恢复。因此在 R_0 和 R_B 之间不存在差别：任务是求速率对 (R_0, R_A) 的容量区域，其中，R_0 是信息同时到达 A 和 B 的速率，R_A 是到达 A 的额外信息的速率。

下一个习题与此等价，它的解答如图 15.8 所示。

习题 15.20[♪]　**具有未知噪声水平的信道的可变速率纠错码**　在实际生活中，在安装编码器之前，信道特征有时可能并未清楚地描述出来。作为这种情况的一个模型，假设信道是噪声水平为 f_A 或 f_B 的二进制对称信道。设 $f_B > f_A$，并设二者的容量为 C_A 和 C_B。

那些喜欢冒险的人也许会安装一个噪声水平设计为 f_A 的、速率 $R_A \simeq C_A$ 的系统；在实地噪声水平为 f_B 时，由香农理论方面的经验，我们预计在可靠信息通信方面会出现一个灾难性的失败（如图 15.6 中的实线所示）。

保守的方法是为最坏情况设计编码系统，安装一个速率为 $R_B \simeq C_B$ 的码（如图 15.6 中的虚线所示）。如果实地噪声水平是较低的 f_A，系统仍能工作正常，但业务主管们会为浪费了容量差 $C_A - R_B$ 而感到遗憾。

是否可能构建一个系统，它不但可以在任何噪声水平下以某速率 R_0 可靠地传输，而且在噪声水平较低时还能可靠地传输一些额外“较低优先级”比特？如图 15.7 所示。这种码在 $f_A \sim f_B$ 的所有噪声水平下都能可靠地传输高优先级比特，在噪声水平为 f_A 或更低时，还能传输低优先级比特。

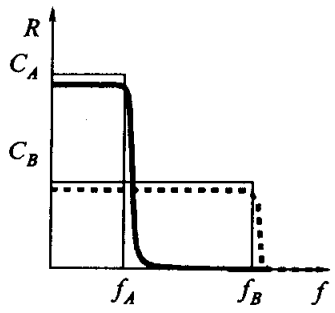


图 15.6 设计运行于噪声水平 f_A(实线)和 f_B(虚线)的香农码的可靠通信速率 R,以噪声水平 f 的函数表示。

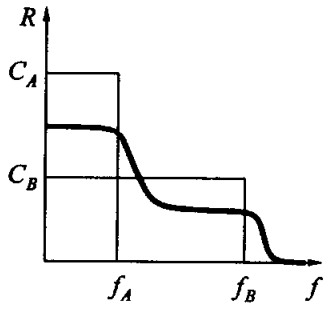


图 15.7 理想可变速率码的可靠通信速率 R,以噪声水平 f 的函数表示。

这个问题在数学上等价于前一个问题,即性能降低的广播信道。那里较低的通信速率被称为 R_0,如果噪声水平低时,低优先级比特的通信速率则被称为 R_A。

对于 $f_A = 0.01$ 和 $f_B = 0.1$ 的情况,说明性的答案如图 15.8 所示。(该图同时给出了两个子信道噪声水平分别为 $f_A = 0.01$ 和 $f_B = 0.1$ 的广播信道的可达区域。)笔者承认发现简单分时方案和灵巧方案之间的性能差距小得令人失望。

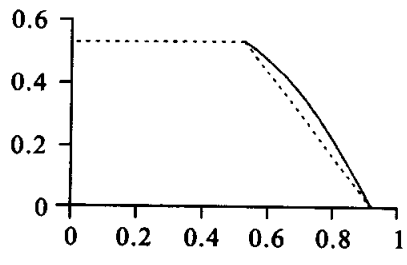


图 15.8 具有未知噪声水平的信道的可达区域。假设两个可能的噪声水平为 $f_A = 0.01$ 和 $f_B = 0.1$,虚线表示利用一个简单的分时方法就可达到的速率 R_A 和 R_B,实线表示用一个更巧妙的方法达到的速率。

在第 50 章所讨论的编码问题是针对一类特殊的广播信道，即删除信道，其中每个符号或者被无误接收或者被删除。这些码有一个良好的特性即*无码率特性*——因为无论信道的删除统计特性如何，都可实时确定被传输的符号的数目以实现可靠通信。

习题 15.21[24]　**多终端信息网络**在实际应用和理论吸引力两方面都很重要。考虑一个双向二进制信道的例子（如图 15.9(a)、(b)所示）：两人都希望在一个信道上通话，并且他们都想听到另一个人说些什么；但是只有在你发送一个 0 时，才能听到另一个人所发送的信号。从 A 到 B 和从 B 到 A 同时通信能达到的信息速率是多少，又如何达到呢？在日常生活中，这类网络的实例包括船舶上使用的 VHF 信道和计算机以太网（其中若有两台或更多的设备同时广播，则所有的设备都不能听到任何东西）。

显然，通过简单的分时，可以在两个方向上达到 1/2 的速率。但是这两个信息速率是否能更大一些呢？求解通用双向信道的容量仍然是一个未解决的问题。然而，如图 15.9(c)所示，对于上述简单二进制信道，可以获得有关可达点的有趣结论。存在码可以达到不超出所示边界的速率。也许还存在更好的码。

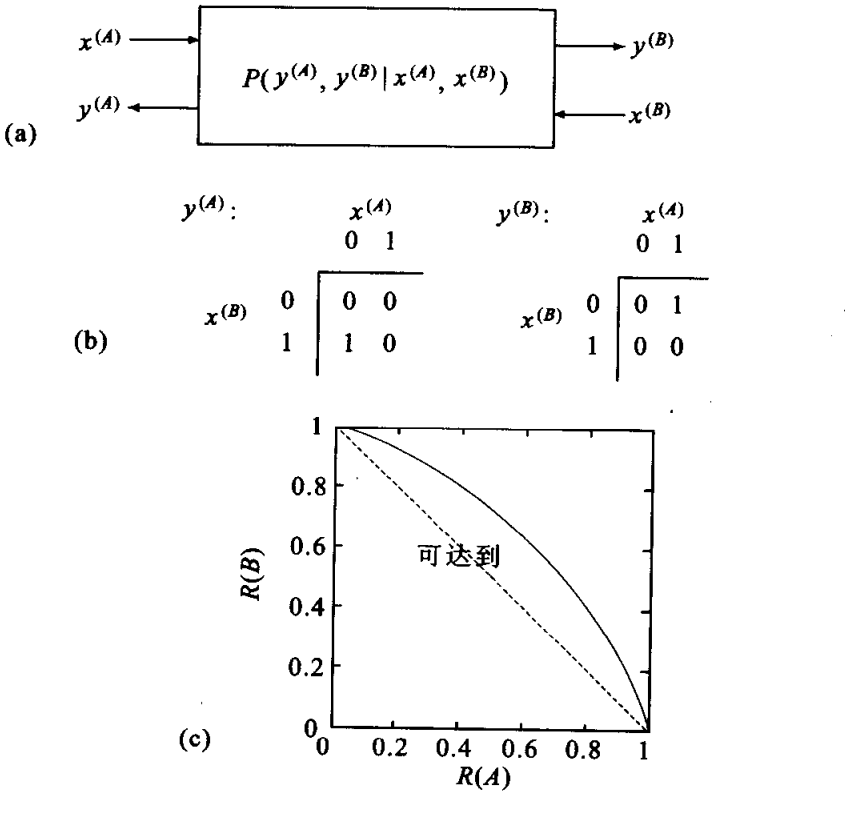


图 15.9　(a) 通用双向信道。(b) 一个二进制双向信道的规则。两个表格表示了每个输入状态所得的输出 $y^{(A)}$ 和 $y^{(B)}$。(c) 二进制双向信道的可达区域。实线以下的速率是可达的。虚线表示通过简单分时“显然可达”的区域。

习题解答

习题 15.12 的解答 $C(Q)=5$ 比特。

最后部分的提示:存在一个涉及简单(8,5)码的解答。

第 16 章

消息传递

本书的主题之一是使用简单的分布式硬件来进行复杂计算的思想。结果证明，消息传递算法可以解决很多有趣的问题，这种算法在各简单处理器之间进行本地消息传递，经过一段时间后，它们的运算会给出一个全局问题的解。

16.1 计 数

举个例子，考虑在雾中行走的一列士兵。指挥官希望进行复杂计算以计数队伍中的士兵。这个问题可以用两种方法来解决。

首先是一个采用昂贵硬件的解决方案：指挥官和他的部下需要大声说话。指挥官可以大声叫道："所有士兵必须在一分钟之内向我报到！"，然后他就仔细地听士兵回应"Molesworth 到！"，"Fotherington - Thomas 到！"，等等。这种方法依靠几种昂贵的"硬件"：指挥官与士兵之间必须有可靠的通信渠道；指挥官一定要能听清所有发给他的消息——哪怕是有上百名士兵——还必须能够计数；所有士兵都必须营养好，以便有力气大声回应可能处于远处的指挥官。

第二种求解士兵数目这一全局函数的方法不要求全程通信"硬件"、高智商或好的饮食；只需每个士兵和队列中的两个相邻士兵用单个整数进行通信，而且士兵能在一个数字上加 1。每个士兵遵循以下规则（算法 16.1），图示如图 16.2 所示。

1. 队列中最前面的士兵对其后的士兵说"1"。
2. 队列中最后一个士兵对其前面的士兵说"1"。
3. 如果你前面的（或者你后面的）士兵告诉你一个数字，你就将数字加 1 之后报给后面的（或者前面的）士兵。

算法 16.1　消息传递规则集 *A*

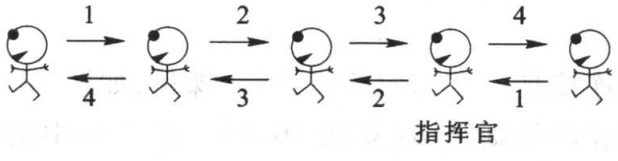


图 16.2　采用消息传递规则 *A* 计数的一列士兵。指挥官可以从他前面的士兵加"3"，从他后面的士兵加"1"，再加上自己，得出总共有 5 个士兵。

如果聪明的指挥官不仅会在一个数的基础上加1，还会将两个数相加，就可以简单地把如下数字加在一起以求得士兵的总数。

他前面的士兵所告诉他的数字（等于他前面的士兵总数）+他后面的士兵所告诉他的数字（等于他后面的士兵数）+1（算上指挥官自己）

这种解决方案只需要本地通信的硬件条件和简单的计算（存储和整数加法）。

隔离法

这种聪明的技巧利用了士兵总数的一个深刻特性：它可以写成某点前面的士兵数与其后面士兵数的总和，而这两个数值可以分别计算，因为这列士兵被指挥官分成了两组。

如果士兵并非排成队列而是成群地行进，就不太容易用这种方法将他们分成两组了。图16.3中的游击队员不能采用上述消息传递规则集 A 进行计数，因为尽管游离队伍中确实有邻居（用线来表示），但并不清楚谁在“前”谁在“后”；此外，因为游击队员的连接图中出现了回路，在回路中的游击队员（比如Jim）就无法将这群游击队员分成“前面的”和“后面的”两组。

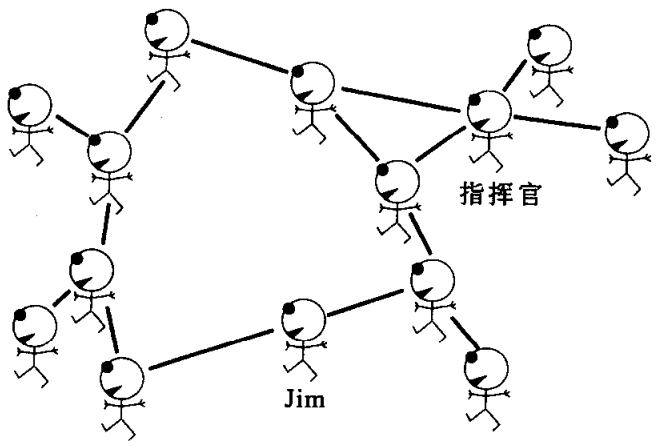


图16.3 成群的游击队员

如果将这群游击队员排成不包含回路的图，就可以用一个改进的消息传递算法进行计数。

如果游击队员的连接是一个无回路图，即一棵树，如图16.4所示，则存在一种计算其数目的消息传递算法规则集 B（算法16.5）。游击队中的任意成员可以从他所接收的消息中推断树中结点的总数。

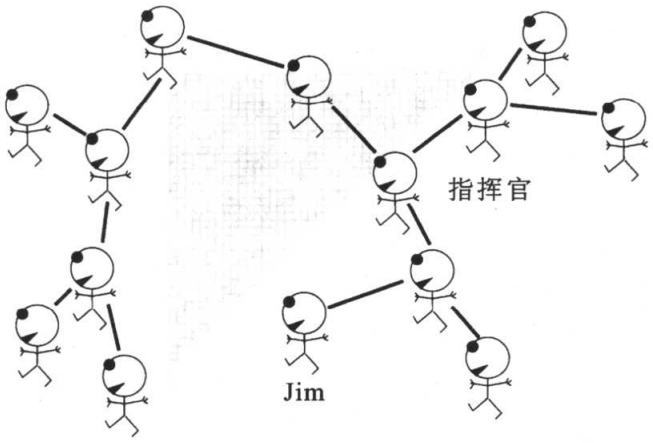


图 16.4 一群游击队员,其连接形成一棵树。

1. 计算你的邻居数 N。

2. 保存你从邻居所获得的消息数 m 以及这些消息中的值 $v_1, v_2, \cdots, v_N$。设 V 为你所接收的动态消息总数。

3. 如你所接收的消息数 m 等于 $N-1$,找出没有给你发送消息的那个邻居,并将数字 $V+1$ 告诉他。

4. 如果你所接收的消息数等于 N,则

(a) $V+1$ 就是希望得到的总数。

(b) 对于每个邻居 n|

将数 $V+1-v_n$ 告诉邻居 n。

}

算法 16.5 消息传递规则集 B

16.2 路径计数

比成员计数更有意义的一项任务是计算通过一个网格的路径数,并求出其中有多少条路经穿过网格中的一个给定点。

图 16.6 给出了一个方形网格和穿过网格、连接点 A 和 B 的一条路径。一条有效路径是从点 A 出发向右、向下移向 B 的路径。我们的问题如下。

1. 从 A 到 B 有多少条这样的路径?

2. 如果随机选择从 A 到 B 的一条路径,则它经过网格中一个特定结点的概率是多少?[我们说“随机”,是指所有路径具有被选中的相同概率。]

3. 如何选取从 A 到 B 的一条随机路径?

似乎无法直接对从 A 到 B 的所有路径进行计数。预计路径数会很大——即使

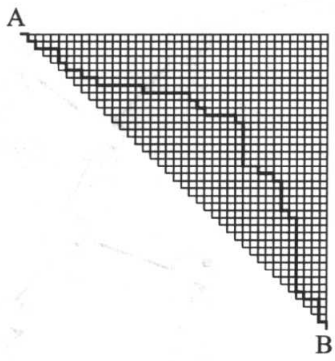


图 16.6 41×41 的三角形网格。从 A 到 B 有多少条路径？图中给出了一条路径。

所允许的网格是一个只有 3 个结点宽的斜对角条，仍然会有约 $2^{N/2}$ 条可能的路径。

计算上的突破来自这样一种认识：无需明确地列举出所有路径，就可以求出路径数。从网格中选取点 P，并考虑从 A 到 P 的路径数。从 A 到 P 的每条路径在经过上游的邻接点后，都到达点 P（“上游”意味着上面或左侧）。因此从 A 到 P 的路径数等于从 A 到每个相邻结点的路径数之和。如图 16.7 所示。

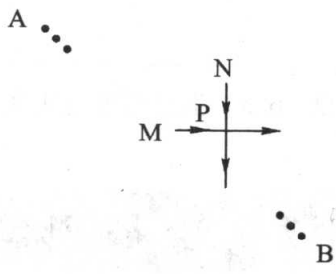


图 16.7 从 A 到 P 的每条路径都通过 P 的一个上游相邻结点 M 或 N；所以可以将从 A 到 M 的路径数和从 A 到 N 的路径数相加得到从 A 到 P 的路径数。

如图 16.8 所示的消息传递算法，所针对的是由 12 条有向边连接 10 个顶点组成的一个简单网格。开始时，从点 A 发送“1”消息。每个结点收到来自所有上游结点的消息之后，就将这些消息之和送至相邻的下游结点。在点 B，出现了数字 5：我们没有列举全部路径就计算出了从 A 到 B 之间的路径数。作为验证，图 16.9 给出了从 A 到 B 的 5 条不同路径。

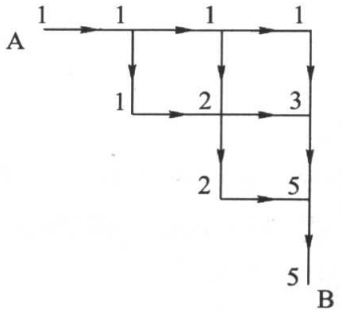


图 16.8 用前向传递发送的消息

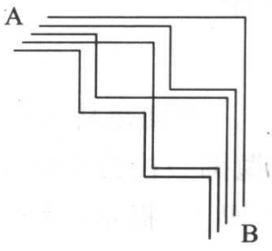


图 16.9 5 条路径

计数完所有路径,继续探讨一些更有挑战性的问题:计算一条随机路径穿过一个给定点的概率,并创建一条随机路径。

经过一个结点的概率

通过构建后向传递和前向传递,可以推断出有多少条路径穿过每一结点;若将它除以总路径数,得到一条随机选择路径经过某结点的概率。如图 16.10 所示,表中右下角是后向传递的消息,左上角是原来的前向传递消息。在某一已知顶点处,将这两个数相乘,得到穿过该顶点的总路径数。例如,4 条路径穿过中心顶点。

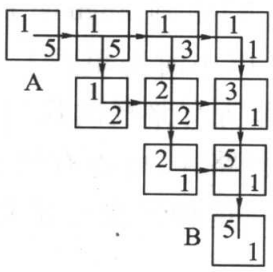


图 16.10 消息通过前向和后向路径传递

如图 16.11 所示是对 41 ×41 的三角形网格这样求解所得到的结果。每个黑点的面积与经过相应结点的概率成比例。

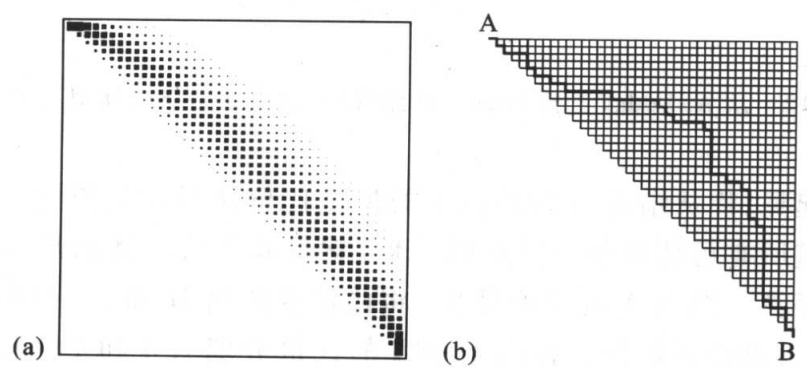


图 16.11 (a) 经过每一结点的概率。(b) 随机选择的一条路径。

随机路径抽样



习题 16.1[↗] 若创建从 A 到 B 的一条“随机”路径，在有两个方向可选择的每个连接结点处，抛掷一枚公平硬币来决定移动方向，所得路径是所有路径集合中的一个均匀随机样本吗？[提示：设想用图 16.8 中的网格试验一下。]

这里有一个漂亮的深刻见解，笔者希望你能对发现它感到满意。



习题 16.2[↗] 在网格的点 A 和点 B 之间运行了前向算法和后向算法之后，如何均匀地画出从 A 到 B 的一条随机路径？（如图 16.11 所示。）

用来计算到 B 的路径数的消息传递算法是*和 - 积算法*的一个例子。在每个结点上对来自它前驱结点的消息进行求和；这里没有提到求“积”，但可以把它看做权重都为 1 的项的加权和。

16.3 求解代价最低的路径

设想你希望尽快从 Ambridge(A)走到 Bognor(B)。图 16.12 给出了不同的路线以及穿越图中每条边所需的代价(以小时为单位)。例如，路线 A - I - L - N - B 要花费 8 个小时。希望不要明确计算所有路径的代价，就求得代价最低的路径。通过求解从 A 点到每个结点的最低代价路径，可以有效地解答这个问题。可以从结点 A 开始用消息传递算法计算这些量。这种消息传递算法也称为*最小和算法*或 Viterbi *算法*。

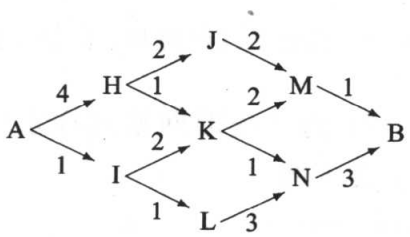


图 16.12 从 Ambridge 走到 Bognor 的路线图，给出了每条边所对应的代价。

为了简短起见，称从结点 A 到结点 x 的最低代价路径的代价为“x 的代价”。每个结点一旦知道所有可能的前驱结点的代价，就将其代价广播给后续结点。现在对算法逐步进行手算。结点 A 的代价是 0。将此消息传给 H 和 I。当消息途经图中的各条边时，就加上该边的代价。得到 H 和 I 的代价分别为 4 和 1(如图 16.13(a)所示)。类似地，J 和 L 的代价分别为 6 和 2，但 K 的代价是多少呢？从边 H—K 传出的

消息是,存在从 A 经 H 到 K 的代价为 5 的路径;由边 I—K 可知,另一条路径的代价为 3 (如图 16.13(b)所示)。最小和算法将 K 的代价设置为其中的最小值,并记录通往 K 的哪一条路径代价最小:只保留边 I—K,去掉其他通向 K 的边(如图 16.13(c)所示)。图 16.13(d)和图 16.13(e)所示为算法的最后两次迭代,表明从 A 到 B 存在一条代价为 6 的路径。[如果最小和算法遇到一个平局,即有多条通往某个结点的路径具有最小代价,那么该算法可以随机地选取其中的任意一条路线。]

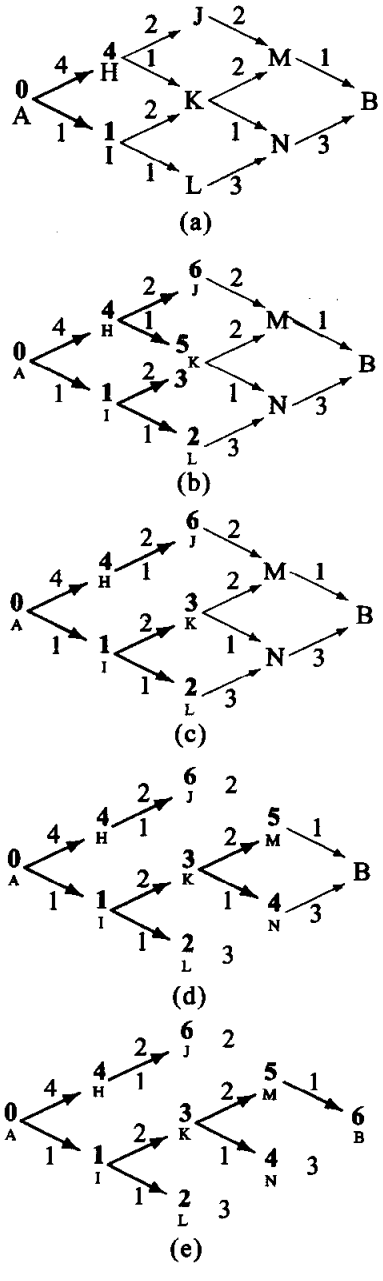


图 16.13 最小和消息传递算法,它可求出到达每一结点的代价,从而得出从 A 到 B 的最小代价路径。

可以通过从点 B 顺着保留下来的边原路返回 A 来恢复这条最小代价路径。可以推导出最低代价路径为 A—I—K—M—B。

最小和算法的其他应用

假设你要管理从原材料到大量操作的产品生产过程。你希望识别处理过程中的关键路径，即支持生产的操作子集。如果关键路径上的任何操作完成得早一点，就可减少从原材料到成品的时间。

可以利用最小和算法求出一组操作的关键路径。

在第25章中，将利用最小和算法对纠错码进行解码。

16.4 小结及相关思想

一些全局函数具有可分离特性。例如，从A到P的路径数分离成从A到M（点P左侧的点）的路径数和从A到N（点P上方的点）的路径数之和。这些函数可通过消息传递有效地计算出来。其他函数不具有这种可分离特性，如下所示。

1. 军队中生日相同的士兵对数。

2. 一大群士兵中身高相同（四舍五入至最近的厘米数）的人所组成的最大一组的大小。

3. 访问军队里每个士兵的旅行推销员所需要走的最短路程的长度。

机器学习的挑战之一就是找到这类问题的最低代价解。可以用神经网络中的方法来求解寻找一大子集近似相等变量的问题（Hopfield和Brody，2000年；Hopfield和Brody，2001年）。旅行推销员问题的神经网络方法将在第42.9节中讨论。

16.5 进一步的习题

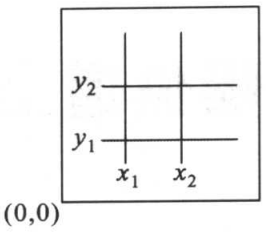
▷ **习题16.3**[⌕] 对于一个边长和高均为N的三角形网格，描述图16.11(a)所画概率的渐近特性。

▷ **习题16.4**[⌕] 在图像处理中，从图像$f(x,y)$（其中x和y均为像素坐标）得出的集总图像$I(x,y)$定义为

$$I(x,y) \equiv \sum_{u=0}^{x} \sum_{v=0}^{y} f(u,v) \tag{16.1}$$

证明集总图像$I(x,y)$可以通过消息传递有效地求得。

证明从集总图像中可以获得一些简单的图像函数。例如，给出从(x_1,y_1)到(x_2,y_2)的矩形区域内所有(x,y)的图像强度之和$f(x,y)$的表达式。



16.6 习题解答

习题 16.1 的解答 因为图 16.8 中有 5 条穿过网格的路径，它们的概率肯定都是$\frac{1}{5}$。但基于公平投币策略得出的路径的概率是$\frac{1}{2}$的幂。

习题 16.2 的解答 为了得出均匀随机游动，需在每个连接点处，用不同的偏倚硬币选择走动的每个前向步骤，所选择的偏倚值与从这两个候选方向发出的后向消息成正比。例如，离开 A 后的第一次选择，有一个消息“3”从东边来，一个消息“2”从南边来，因此去东边的概率为$\frac{3}{5}$，去南边的概率为$\frac{2}{5}$。图 16.11 中的路径就是这样产生的。

第 17 章

受限无噪信道上的通信

在这一章中，研究在受限无噪信道上进行有效通信的任务——在受限信道上，并非所有来自输入字母表的字符串都可以被传输。

利用第 16 章介绍的思想：可以通过一个本地消息传递算法计算图的全局特性。

17.1 受限二进制信道的 3 个例子

受限信道可以通过定义所允许传输字符串的规则来定义。

例 17.1 在信道 A 中，每个 1 后面必须至少跟着一个 0。[①] 该信道上的一个有效串是

$$00100101001010100010 \tag{17.1}$$

作为使用这个模型的一个动机，考虑用电磁能量脉冲表示 1 的信道。产生这些脉冲的器件在生成一个脉冲之后需要一个时钟周期的恢复时间，才能生成另一个脉冲。

例 17.2 信道 B 规定所有 1 必须两个或多个成组地出现，所有 0 也必须两个或多个成组地出现。[②]

对于这个信道，一个有效的串是

$$00111001110011000011 \tag{17.2}$$

作为使用这个模型的一个动机，考虑一个磁盘驱动器，把连续比特写到沿磁盘表面的磁道的邻近点上；0 和 1 的值由两个相反的磁方向表示。字符串 101 和 010 是被禁止的，因为被反向磁域围绕的单一孤立磁域是不稳定的，例如，101 可能会转变成 111。

例 17.3 信道 C 规定最大允许游程是 2，也就是每个符号最多可重复 1 次。[③]

这个信道的一个有效串是

$$10010011011001101001 \tag{17.3}$$

使用这个模型的物理动机是一个不能精确得知转速的磁盘驱动器。在这个磁盘驱动器上，辨别两个 1 的串和 3 个 1 的串很困难，它们分别表示成周期为 2τ 和 3τ 的有向磁化，其中 τ 是经过 1 比特所需要的时间（该值未知）；为了避免出现混乱并导

---

① 信道 A：子串 11 是禁用的。

② 信道 B：子串 101 和 010 是禁用的。

③ 信道 C：子串 111 和 000 是禁用的。

致发送端和接收端失去同步,我们禁止出现3个连1串和3个连0串。

所有这3个信道是游程受限信道的例子。规则限制了连续1和连续0的最小和最大数目。

信道	1的游程		0的游程	
	最小值	最大值	最小值	最大值
非受限	1	∞	1	∞
A	1	1	1	∞
B	2	∞	2	∞
C	1	2	1	2

在信道A中,连0串可以是任意长度,但是连1串的长度被限制为1。在信道B中,所有的连串都必须长度为2或更多。在信道C中,所有连串的长度必须是1或2。

非受限二进制信道的容量是每次信道使用1比特。这3个受限信道的容量是多少?[为了公平起见,至今尚未定义这种信道的容量;请把"容量"理解为每次信道使用所能可靠传输的比特数。]

受限信道的一些码

现在集中考虑信道A,它的连0串可以任意长,但是连1串的长度限制为1。尽可能有效地在这种信道上传输随机二进制文件。

以(2,1)码作为一个简单的起点,它把每个信源比特映射成两个发送比特C_1。这是一个码率为1/2的码,它遵守信道A的约束,所以信道A的容量至少为0.5。我们可以做得更好吗?

C_1是冗余的,因为如果两个接收比特中的第1个为0,我们知道第2个比特也将是0。可以通过使用忽略C_1中冗余0的一个码来获得更小的平均发送长度。

码	C_1
s	t
0	00
1	10

C_2是这样一种可变长度码。如果信源符号以等频率被使用,那么每信源比特的平均发送长度为

$$L=\frac{1}{2}1+\frac{1}{2}2=\frac{3}{2} \tag{17.4}$$

所以平均通信速率为

$$R=\frac{2}{3} \tag{17.5}$$

而且信道 A 的容量必须至少为 2/3。

码	C_2
s	t
0	0
1	10

我们能做得比 C_2 更好吗？有两种方法可以说明信息率可被增加至 $R=2/3$ 以上。

第一种方法假设我们对于用熵来度量信息量的做法能够接受。这个思想是，从码 C_2 开始，通过减少 1 在发送中所占的比重，可以减少平均信息长度，而不会太多地减少所发送信息的熵。假设在送入 C_2 的一串比特中，1 出现的频率为 f。[这种串可以从任意二进制文件中获得，具体方法是将信源文件输入算术码（对压缩密度为 f 的二进制符号串而言是最优码）的译码器。]所达到的信息率 R 是信源熵 $H_2(f)$ 除以平均发送长度。

$$L(f)=(1-f)+2f=1+f \tag{17.6}$$

因此（如图 17.1 所示）

$$R(f)=\frac{H_2(f)}{L(f)}=\frac{H_2(f)}{1+f} \tag{17.7}$$

不带预处理器的原码 C_2 对应于 $f=1/2$。如果让 f 受到扰动使之变得更小，设定

$$f=\frac{1}{2}+\delta \tag{17.8}$$

其中 δ 是小负数，会有什么变化？在 $f=1/2$ 附近，分母 $L(f)$ 随着 δ 线性变化。相反的，分子 $H_2(f)$ 同 δ 之间只存在一个二阶的依存关系。

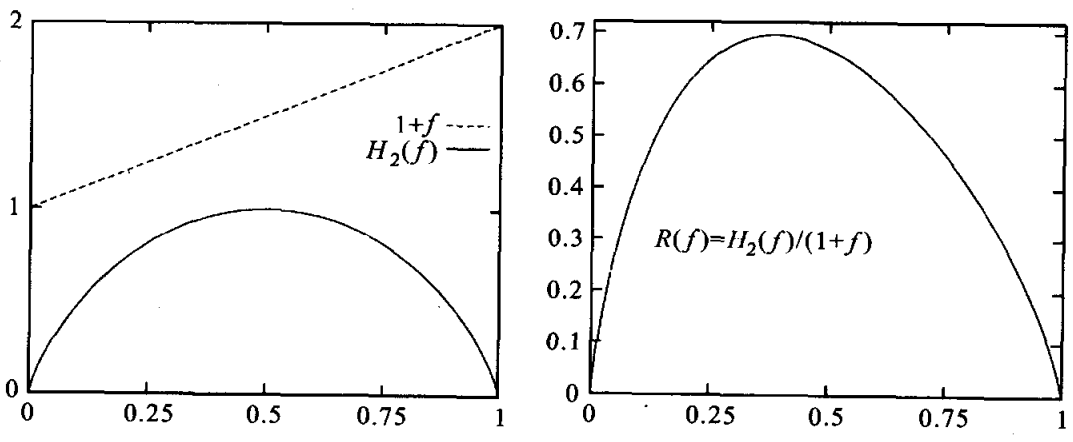


图 17.1 上半图：每个源符号的信息量和每个信源符号的平均传输长度，以源密度的函数表示。下半图：每个传输符号的信息量（用比特表示），以 f 的函数表示。

▷ **习题 17.4**[1] 求出作为 δ 函数的 $H_2(f)$ 的泰勒展开式，要求展开到阶 δ^2。

对于第一阶，$R(f)$随着δ的减少而线性增加。一定有可能通过减少f来增加R。图17.1画出了这些函数；$R(f)$确实随着f的减小而增加了，并在$f\simeq 0.38$处达到最大值，为每次信道使用约0.69比特。

通过这种论证，可知信道A的容量至少为$\max_f R(f)=0.69$。

▷**习题17.5**[⌀] 如果通过C_2发送一个包含1的比率为$f=0.5$的文件，那么发送流中1的比例是多少？

如果使用密度为$f=0.38$的稀疏信源来驱动码C_2，那么发送比特中1的比例又是多少？

另一种也是更加基本的方法，是计算有多少个长度为N的有效序列S_N。通过给每个有效序列取名，可以在N个信道周期中传输$\log S_N$比特。

17.2 受限无噪信道的容量

我们通过输入和输出之间的互信息来定义噪声信道的容量，然后证明了这个数（信道容量）与N次信道使用能够可靠传输的不同消息数$S(N)$之间的关系

$$C=\lim_{N\to\infty}\frac{1}{N}\log S(N) \tag{17.9}$$

对于受限无噪信道，也可采用此式作为信道容量的定义。可是，名称s要有不同的用途：在为噪声信道编码时（第9.6节），表示消息$s=1,2,\cdots,S$，而这里将用于标记信道状态；因此在这一章中，将用M_N表示长度为N的不同消息数，并定义信道容量为

$$C=\lim_{N\to\infty}\frac{1}{N}\log M_N \tag{17.10}$$

一旦求出信道容量，将回来完成原来的任务，即为那个信道构造实际的码。

17.3 计算可能的消息数

首先引入受限信道的一些表示方法。在状态图中，发射方的状态可以用标注状态名的圆圈来表示。从一个状态到另一个状态的有向边表示允许发射方从第1个状态转到第2个状态，而那条边上标号表示进行转移时所发送的符号。如图17.2(a)所示为信道A的状态图。它有两个状态，0和1。当转移到状态0时，就发送0；当转移到状态1时，就发送1；不能从状态1转移到状态1。

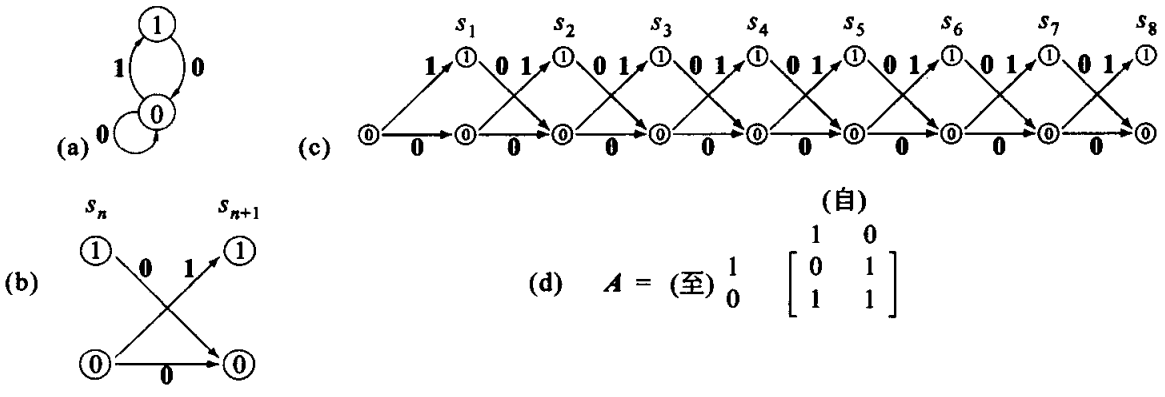


图 17.2　(a) 信道 A 的状态图。(b) 网格段。(c) 网格图。(d) 连接矩阵。

也可以通过网格段来表示状态图，它表示两个连续水平位置上的两个相继状态(图 17.2(b))。发送端在时间 n 的状态称作 s_n。这组可能的状态序列可用图 17.2(c)所示的网格图表示。一个有效序列对应于通过网格图的一条路径，而有效序列数就是路径数。为了计算网格图中有多少条路径，忽略边线上的标号，并用连接矩阵 A 来概括网格段，其中 $A_{ss'}=1$，前提条件是从状态 s 到 s' 存在一条边；否则，$A_{ss'}=0$(图 17.2(d))。图 17.3 表示信道 B 和 C 的状态图、网格段和连接矩阵。

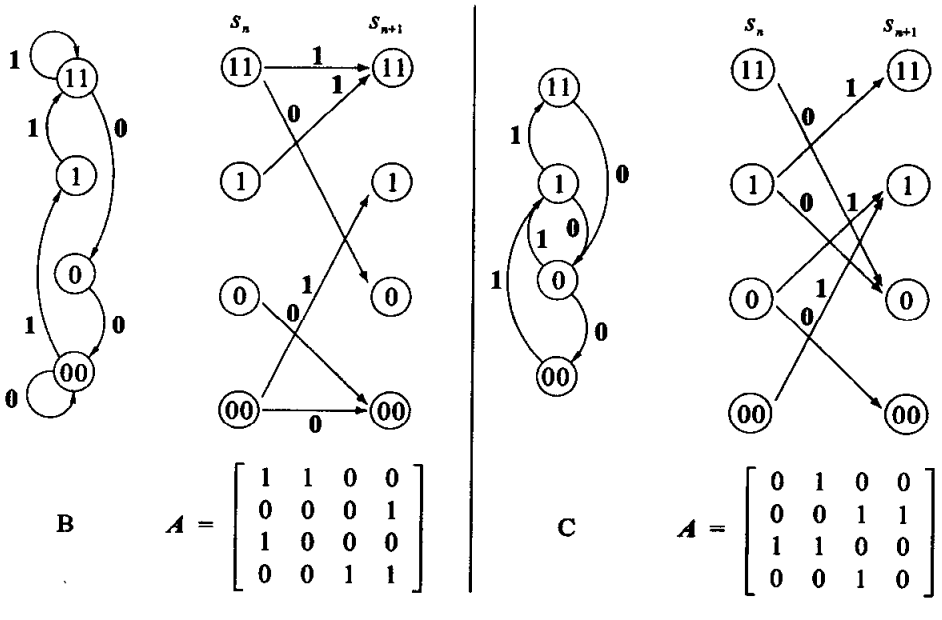


图 17.3　信道 B 和 C 的状态图、网格段和连接矩阵

可以通过在网格图中进行消息传递来计算信道 A 所对应的路径数(如图 17.6 所示)。图 17.4 给出了这个计数过程的前几步，图 17.5(a)所示为 n 步后终止于每个状态的路径数($n=1,2,\cdots,8$)。长度为 n 的路径的总数 M_n 由最上面的一行表示。可以看出 M_n 为斐波纳契数列。

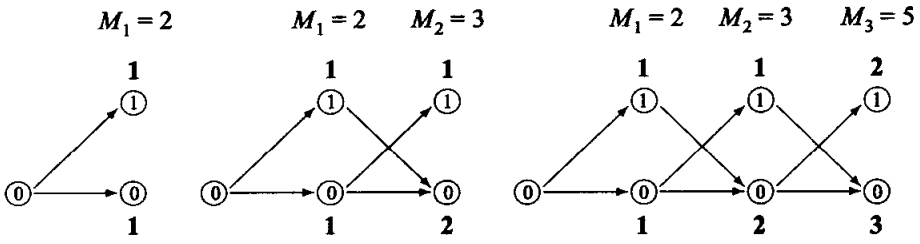
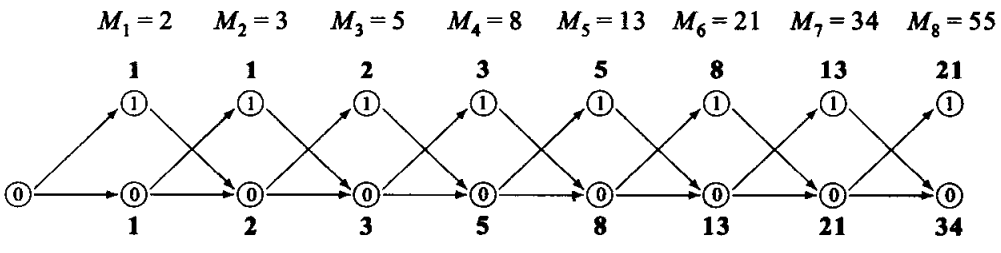
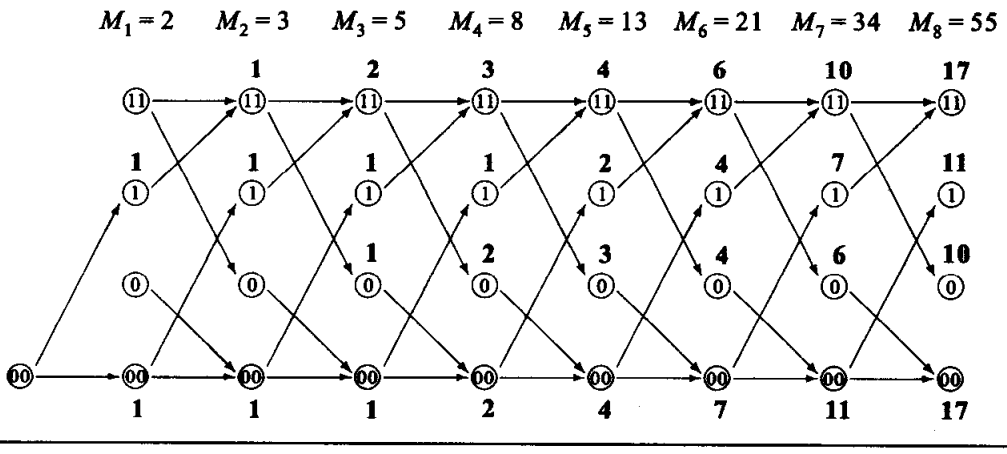


图 17.4 计算信道 A 网格图中的路径数。从左到右通过网格，并累加得出结点旁所示的计数。

(a) 信道 A



(b) 信道 B



(c) 信道 C

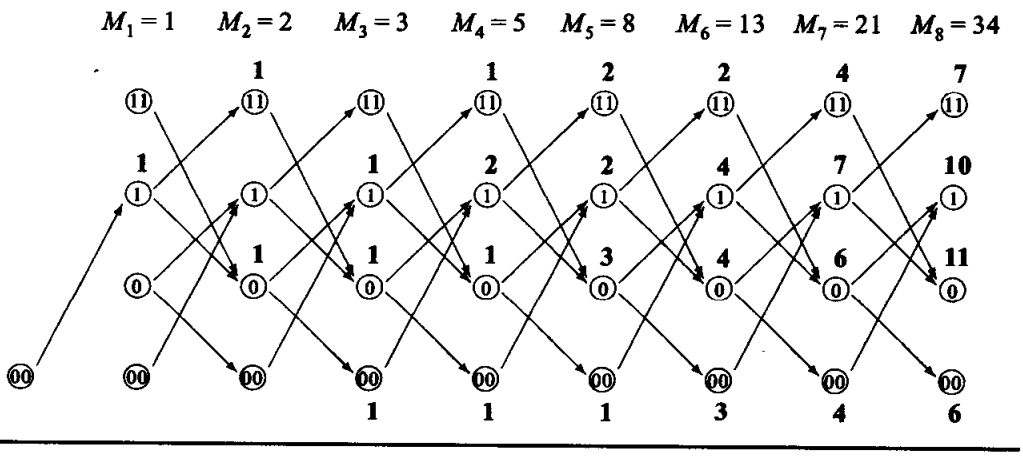


图 17.5 对信道 A、B、C 的网格图中的路径进行计数。假设开始的第一个比特之前是 00，于是对于信道 A 和 B，初始字符可以是任意的，而对于信道 C，初始字符必须是 1。

n	M_n	M_n/M_{n-1}	$\log_2 M_n$	$\frac{1}{n}\log_2 M_n$
1	2		1.0	1.00
2	3	1.500	1.6	0.79
3	5	1.667	2.3	0.77
4	8	1.600	3.0	0.75
5	13	1.625	3.7	0.74
6	21	1.615	4.4	0.73
7	34	1.619	5.1	0.73
8	55	1.618	5.8	0.72
9	89	1.618	6.5	0.72
10	144	1.618	7.2	0.72
11	233	1.618	7.9	0.71
12	377	1.618	8.6	0.71
100	9×10^{20}	1.618	69.7	0.70
200	7×10^{41}	1.618	139.1	0.70
300	6×10^{62}	1.618	208.5	0.70
400	5×10^{83}	1.618	277.9	0.69

图 17.6 对信道 A 的网格图中的路径进行计数

▷ **习题 17.6**[✓] 证明斐波纳契数列中连续项的比值趋近于黄金分割点

$$\gamma \equiv \frac{1+\sqrt{5}}{2} = 1.618 \tag{17.11}$$

因此，在一常数范围之内，随着 $N\to\infty$，M_N 按 $M_N \sim \gamma^N$ 增长，所以信道 A 的容量为

$$C = \lim \frac{1}{N}\log_2[\text{常数}\cdot\gamma^N] = \log_2\gamma = \log_2 1.618 = 0.694 \tag{17.12}$$

如何描述我们刚才所做的事情？路径数的计数是向量 $\boldsymbol{c}^{(n)}$；可由下式从 $\boldsymbol{c}^{(n)}$ 得到 $\boldsymbol{c}^{(n+1)}$。

$$\boldsymbol{c}^{(n+1)} = \boldsymbol{A}\boldsymbol{c}^{(n)} \tag{17.13}$$

所以

$$\boldsymbol{c}^{(N)} = \boldsymbol{A}^N\boldsymbol{c}^{(0)} \tag{17.14}$$

其中 $\boldsymbol{c}^{(0)}$ 是在任何符号发送之前的状态计数。在图 17.5 中，假设 $\boldsymbol{c}^{(0)} = [0,1]^{\mathrm{T}}$，即开始时，允许两个符号中的任意一个。总路径数为 $M_n = \sum_s c_s^{(n)} = \boldsymbol{c}^{(n)}\cdot\boldsymbol{n}$。在极限情况下，$\boldsymbol{c}^{(N)}$ 为 $\boldsymbol{A}$ 的主右特征向量所主导。

$$\boldsymbol{c}^{(N)} \to \text{常数}\cdot\lambda_1^N\boldsymbol{e}_{\mathrm{R}}^{(0)} \tag{17.15}$$

这里，λ_1 是 $\boldsymbol{A}$ 的主特征值。

因此，为了求出任意受限信道的容量，只需求出连接矩阵的主特征值 λ_1。于是

$$C = \log_2\lambda_1 \tag{17.16}$$

17.4 重新讨论模型信道

比较图 17.5(a)、图 17.5(b)和图 17.5(c),信道 B 和 C 同信道 A 看起来好像具有相同的容量。3 个网格的主特征值相同(在本书附录中,表 C.4 的底部给出了信道 A 和 B 的特征向量)。这些信道确实密切相关。

信道 A 和 B 的等价性

如果让信道 A 的任意有效串 $\boldsymbol{s}$ 通过一个累加器(如图 17.7 所示),并获得定义如下的 $\boldsymbol{t}$

$$\begin{aligned} t_1 &= s_1 \\ t_n &= t_{n-1} + s_n \bmod 2, \quad n \geqslant 2 \end{aligned} \tag{17.17}$$

于是,所得结果串是信道 B 的有效串,因为在 $\boldsymbol{s}$ 中没有 11,所以 $\boldsymbol{t}$ 中没有单独的数字。累加器是可逆算子。所以,类似地,信道 B 的任意有效串 $\boldsymbol{t}$ 也可以通过二进制差分器映射到信道 A 的有效串 $\boldsymbol{s}$ 上。

$$\begin{aligned} s_1 &= t_1 \\ s_n &= t_n - t_{n-1} \bmod 2, \quad n \geqslant 2 \end{aligned} \tag{17.18}$$

因为 + 和 - 在模 2 算术中是等价的,差分器也是一个模糊器,其功能是将信源流与滤波器(1,1)进行求卷积。

信道 C 也与信道 A 和 B 密切相关。

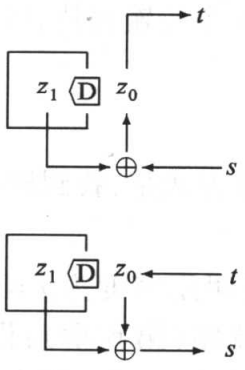


图 17.7 一个累加器和一个差分器

▷ **习题 17.7**[ㄥ] 信道 C 和信道 A、信道 B 之间的关系是什么?

17.5 受限信道上的实际通信

好了,那么在实际中要怎样做? 因为所有 3 个信道是等价的,所以可集中考虑信道 A。

固定长度解

首先来明确地列出这些码。在表 17.8 中,码的码率为 3/5 = 0.6。

表 17.8 信道 A 的游程受限码

s	$c(s)$
1	00000
2	10000
3	01000
4	00100
5	00010
6	10100
7	01010
8	10010

▷**习题 17.8**[↗] 类似地,列出所有长度为 8 并以 0 状态结束的字符串(共 34 个)。于是,证明可以映射 5 个比特(32 个信源串)到 8 个传输比特,并达到码率 5/8 = 0.625。

映射整数个信源比特到 $N = 16$ 个传输比特,可达到的码率是多少?

最优可变长度解

在受限信道上传递信息的最佳方式是,找到网格中所有点的最优转移概率 $Q_{s'|s}$,并以这些概率进行转移。

在对信道 A 进行讨论时,已证明用密度为 $f = 0.38$ 的稀疏信源驱动码 C_2,可以达到容量。我们知道如何制作稀疏器(第 6 章):设计一个算术码,对稀疏信源进行最佳压缩;于是它所对应的译码器就给出了从稠密串(即随机二进制串)到稀疏串的最优映射。

求最优概率的任务以习题形式给出。

习题 17.9[↗] 证明用如下方法可以求得最优转移概率 Q。

求 $\boldsymbol{A}$ 的主右特征向量和主左特征向量,即 $\boldsymbol{A}\boldsymbol{e}^{(R)} = \lambda \boldsymbol{e}^{(R)}$ 和 $\boldsymbol{e}^{(L)\mathrm{T}}\boldsymbol{A} = \lambda \boldsymbol{e}^{(L)\mathrm{T}}$ 的解,其中 λ 为最大特征值。然后构造矩阵 $\boldsymbol{Q}$,它的不变分布与 $e_i^{(R)} e_i^{(L)}$ 成正比,即

$$Q_{s'|s} = \frac{e_{s'}^{(L)} A_{s's}}{\lambda e_s^{(L)}} \tag{17.19}$$

[提示:也许习题 16.2 可在这里给出有益的"交叉培育"启示。]

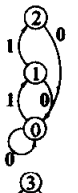
▷ **习题 17.10**[$\mathcal{J}$] 证明用最优转移概率矩阵(17.19)生成序列时,所得序列的熵渐近地是每个符号 $\log_2\lambda$。[提示:假定前一个符号的分布是不变分布,考虑在给定前一个符号时,一个符号的条件熵。]

实际上,或许可以使用最优可变长度解的有限精度近似值。有些人可能不喜欢可变长度解,因为在任何特定场合下,所得的实际编码长度是不可预知的。或许在一些应用中,我们想保证一个大小为 N 比特的信源文件的编码长度小于某个给定长度,比如 $N/(C+\varepsilon)$。举个例子,如果知道所有 512 B 的块在磁盘上占据同样大小的空间,这种磁盘驱动器就更容易控制。对于一些受限信道,可以对变长编码进行简单的修改,以便提供这种保证。找到两个码,即从二进制串到变长编码的两个映射,对于任意信源串 $\boldsymbol{x}$ 具有以下性质:如果 $\boldsymbol{x}$ 在第 1 个码下的编码短于平均值,那么 $\boldsymbol{x}$ 在第 2 个码下的编码会长于平均值,反之亦然。那么,要传送字符串 $\boldsymbol{x}$,同时用两个码对整个串进行编码,并发送那个编码长度最短的码,预先给出一个适当编码的比特来传送正在使用的那个码的信息。

▷ **习题 17.11**[$\mathcal{JC}$] 对于如图 17.9 所示的游程受限信道,有多少长度为 8 并以 0 开头的有效序列?这些信道的容量是多少?

对于信道 A 以及如图 17.9 所示的两个游程受限信道,使用计算机,求解用来生成穿过各网格图中随机路径的矩阵 $\boldsymbol{Q}$。

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

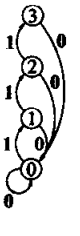


图 17.9 符号 1 的最大游程等于 2 和 3 时,信道的状态图和连接矩阵。

▷ **习题 17.12**[$\mathcal{J}$] 考虑一个游程受限信道,其中 0 的游程长度可以是任意的,而 1 的最大游程长度是一个很大的数 L,比如 9 或 90。

估计该信道的容量。(给出包含 L 的级数展开式中的前两项。)

为了生成一条随机路径以穿过这个信道的网格图,最优矩阵 $\boldsymbol{Q}$ 大概是什么形式?集中考虑元素 $Q_{1|0}$ 的值(已知前面一个为 0、生成一个 1 的概率)和 $Q_{L|L-1}$ 的值(已知前面是 $L-1$ 个连续的 1、下一个产生的也是 1 的概率)。对 1 的最大游程长度为 9 的信道,进行显式计算并验证你的答案。

17.6 可变的符号周期

假设所发送的符号具有不同的周期，且我们的目标是以最大可能的单位时间速率进行通信，就可以对受限信道中的通信任务进一步进行补充。这种信道可以有两种情况：非受限的和受限的。

符号周期可变的非受限信道

在习题 6.18 中，遇到过符号周期可变的非受限无噪信道。解决了那个问题，你就完成了该主题的内容。目标是在已知周期时，确定应以什么样的最优频率使用符号。

这项任务和设计一个最优符号码的任务（第 4 章）可以形成一个很好的类比。为一个有不等概率 p_i 的信源构造二进制符号码时，最优消息长度为 $l_i^* = \log_2 \frac{1}{p_i}$，因此

$$p_i = 2^{-l_i^*} \tag{17.20}$$

类似地，当信道中的符号周期为 l_i（以某种时间单位），那些符号应该被使用的最优概率为

$$p_i^* = 2^{-\beta l_i} \tag{17.21}$$

其中，β 是信道容量，其单位为：每单位时间的比特数。

符号周期可变的受限信道

一旦你掌握了本章前面的几个主题，你应能指出如何定义并求出这些最难处理的受限信道的容量。

习题 17.13[✍] 具有可变符号周期的受限信道的一个经典例子是 Morse 信道，它的符号如下。

点	d
划线	D
短空格（在 Morse 码中用于字母之间）	s
长空格（用于字之间）	S

约束条件是空格后面只能跟着点或者划线。假设（a）所有 4 个符号的周期相等；（b）符号周期分别为 2、4、3 和 6 个时间单位。分别求这个信道的容量（以“比特/时间单位”表示）。

习题 17.14[✍] 将 Morse 码用于英语（假设具有图 2.1 中的概率分布）时，其设计效果如何？

习题 17.15[JC] 将 DNA 放入狭窄的细管中有多难？

对于一个信息理论家，与受限信道相关的熵说明了它能传递多少信息量。在统计物理学中，同样的计算却是为着不同的目的：例如，为了预测聚合体的热力学特性。

举一个小例子，考虑长度为 N 的聚合体，它可以放在一个宽度为 L 的约束细管中，也可以放在外面不受任何约束。在外面，聚合体所处的状态是从一维随机游动（比如，每步有 3 个可能方向）集合中随机抽取的。这种走动的熵是每步 $\log 3$，也就是总熵为 $N\log 3$［聚合体的自由能定义为 $-kT$ 和它的乘积，其中 T 是温度。］而在细管中，聚合体的一维走动也有 3 个方向，除非细管壁挡住，因此，对于 $L=10$ 的例子而言，连接矩阵为

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ & & & & & \ddots & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

这个聚合体的熵是多少？当聚合体进入细管后，熵的变化是多少？若有可能，求它以 L 的函数所表示的式子。利用计算机求出用 L 的一个特定值（如 20）走动时的熵，画出聚合体在细管中横向位置的概率密度。

注意：两种信道在容量上的差异（一个是受限信道，一个是非受限信道）同将 DNA 拖进细管所需的力成正比。如图 17.10 所示。



图 17.10 将 DNA 挤进一个窄管中的模型。DNA 有从细管中弹出的趋势，因为在细管外面，其随机游动有更大的熵。

17.7 习题解答

习题 17.5 的解答 平均来看，C_2 所传输的文件中包括 1/3 的 1 和 2/3 的 0。若 $f=0.38$，1 所占比例为 $\dfrac{f}{1+f}=\dfrac{\gamma-1.0}{2\gamma-1.0}=0.276\,4$。

习题 17.7 的解答 可以从信道 A 的一个有效串得出信道 C 的一个有效串：首

先将它逆转[1→0;0→1],然后让所得串通过累加器。这些操作是可逆的,故 C 中的任何有效串也可映射成 A 中的有效串。这里唯一的附加条件就是边界效应。如果假定信道 C 中传输的第一个字符前面是一串 0,那么第一个字符只能是一个 1(如图 17.5(c)所示),于是只要假定信道 A 的第一个字符为 0,这两个信道就完全等价。

习题 17.8 的解答 对于 $N=16$ 个传输比特,可被编码的信源比特的最大整数为 10,因此当 $N=16$ 时,一个固定长度码的最大速率为 0.625。

习题 17.10 的解答 设不变分布为

$$P(s)=\alpha e_s^{(L)}e_s^{(R)} \tag{17.22}$$

其中 α 是归一化常数。假设 S_{t-1} 来自不变分布,则已知 S_{t-1} 时 S_t 的熵为①

$$H(S_t|S_{t-1})=-\sum_{s,s'}P(s)P(s'|s)\log P(s'|s) \tag{17.23}$$

$$=-\sum_{s,s'}\alpha e_s^{(L)}e_s^{(R)}\frac{e_{s'}^{(L)}A_{s's}}{\lambda e_s^{(L)}}\log\frac{e_{s'}^{(L)}A_{s's}}{\lambda e_s^{(L)}} \tag{17.24}$$

$$=-\sum_{s,s'}\alpha e_s^{(R)}\frac{e_{s'}^{(L)}A_{s's}}{\lambda}[\log e_{s'}^{(L)}+\log A_{s's}-\log\lambda-\log e_s^{(L)}] \tag{17.25}$$

$A_{s's}$ 要么是 0 要么是 1,故与 $A_{s's}\log A_{s's}$ 成正比的所有项的贡献均为 0。因此有

$$H(S_t|S_{t-1})=\log\lambda+-\frac{\alpha}{\lambda}\sum_{s'}\Big(\sum_s A_{s's}e_s^{(R)}\Big)e_{s'}^{(L)}\log e_{s'}^{(L)}+$$

$$\frac{\alpha}{\lambda}\sum_s\Big(\sum_{s'}e_{s'}^{(L)}A_{s's}\Big)e_s^{(R)}\log e_s^{(L)} \tag{17.26}$$

$$=\log\lambda-\frac{\alpha}{\lambda}\sum_{s'}\lambda e_{s'}^{(R)}e_{s'}^{(L)}\log e_{s'}^{(L)}+\frac{\alpha}{\lambda}\sum_s\lambda e_s^{(L)}e_s^{(R)}\log e_s^{(L)} \tag{17.27}$$

$$=\log\lambda \tag{17.28}$$

习题 17.11 的解答 两个信道的连接矩阵的主特征值为 1.839 和 1.928。容量($\log\lambda$)为 0.879 比特和 0.947 比特。

习题 17.12 的解答 该信道类似于非受限二进制信道;若 L 很大,则游程大于 L 的情况很少见,因此我们预计和这个信道只存在微弱的差别;这些差异在游程接近于 L 的环境中会体现出来。信道容量非常接近 1 比特。

考虑此信道中简单的长度可变码:用 111…10 代替最大游程字符串 111…1,且不对信源文件做其他改变,就可以得到容量的下界。此码的平均码率为 $\frac{1}{1+2^{-L}}$,因为在不变分布下,2^{-L} 部分时间会碰上“额外增加一个 0”的状态。

重新使用习题 6.18 中长度可变信道的解答。容量是满足下列方程的 β 值。

$$Z(\beta)=\sum_{l=1}^{L+1}2^{-\beta l}=1 \tag{17.29}$$

和式中的 $L+1$ 个项对应于 $L+1$ 个可忽略的字符串:0,10,110,…,$\dot{1}1\cdots10$。这个和

① 和第 4 章一样,这里 S_t 表示随机变量为状态 s_t 的总体。

由下式精确给出。

$$Z(\beta)=2^{-\beta}\frac{(2^{-\beta})^{L+1}-1}{2^{-\beta}-1} \tag{17.30}$$

[这里用到了 $\sum_{n=0}^{N} ar^n = \frac{a(r^{N+1}-1)}{r-1}$。]

我们预计 β 应该比 1 小一点，以使 $Z(\beta)$ 等于 1。整理并使用 $\ln(1+x)\simeq x$ 求出 β 的近似解

$$Z(\beta)=1 \tag{17.31}$$

$$\Rightarrow \beta \simeq 1-2^{-(L+2)}/\ln 2 \tag{17.32}$$

前面习题中计算了 $L=2$ 和 $L=3$ 的真实容量。对于一些选定的 L 值，下表对近似容量 β 和真实容量进行了比较。

L	β	真实容量
2	0.910	0.879
3	0.955	0.947
4	0.977	0.975
5	0.988 7	0.988 1
6	0.994 4	0.994 2
9	0.999 3	0.999 3

因为在非受限二进制信道中有 $Q_{110}=\frac{1}{2}$，元素 Q_{110} 近似于 1/2（只大一点比特）。当遇到长度为 $L-1$ 的游程时，实际上面临输出 10 或 0 的选择。设选取 10 的概率为 f。假定矩阵 $\boldsymbol{Q}$ 的其余部分被设置成最优值，估算符号流中剩余 N 个字符的熵，以 f 的函数表示。符号流中的后 N 个字符的熵是第一个比特的熵 $H_2(f)$ 加上其余字符的熵；其余字符约为 $N-1$ 个比特（若选 0 为第一个比特）或者 $N-2$ 个比特（若选 1 为第一个比特）。更准确地，如果 C 为信道容量（大概为 1），有

$$\begin{aligned} H(\text{后 } N \text{ 个符号}) &\simeq H_2(f)+[(N-1)(1-f)+(N-2)f]C \\ &= H_2(f)+NC-fC \simeq H_2(f)+N-f \end{aligned} \tag{17.33}$$

求微分并设为 0，以求最优的 f，可得

$$\log_2\frac{1-f}{f}\simeq 1 \Rightarrow \frac{1-f}{f}\simeq 2 \Rightarrow f\simeq\frac{1}{3} \tag{17.34}$$

随着发送 1 的个数的增加，发出一个 1 的概率从大约 0.5 下降到大约 $\frac{1}{3}$。

因此最优矩阵为

$$\begin{bmatrix} 0 & 0.3334 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.4287 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.4669 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4841 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4923 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4963 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4983 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4993 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4998 \\ 1 & 0.6666 & 0.5713 & 0.5331 & 0.5159 & 0.5077 & 0.5037 & 0.5017 & 0.5007 & 0.5002 \end{bmatrix} \tag{17.35}$$

因此,我们粗略的理论是正确的。

第 18 章

纵横字谜与密码破译

在这一章，将对同语言建模相关的一些问题进行不甚深入的讨论。

18.1 纵横字谜

可以将制作纵横字谜的规则想象为定义一个受限信道。可以制作很多有效字谜的事实证明了这类受限信道的容量远大于零。

有两种原型的字谜格式（如图 18.1 所示）。在“A 类”（或称美国式）字谜中，各行各列由一组长度为 2 以上的单词组成，中间由一个或更多的空格隔开。“B 类”（或称英国式）字谜中，各行各列由一些单词和一些单个字符混合而成（由一个或更多的空格隔开），且每个字符至少出现在一个水平方向或垂直方向的单词中。在 A 类字谜中，每个字母同时出现在一个水平方向的单词和一个垂直方向的单词中，然而在 B 类的典型字谜中，只有一半的字母是这样，而另一半字母只在一个单词中出现。

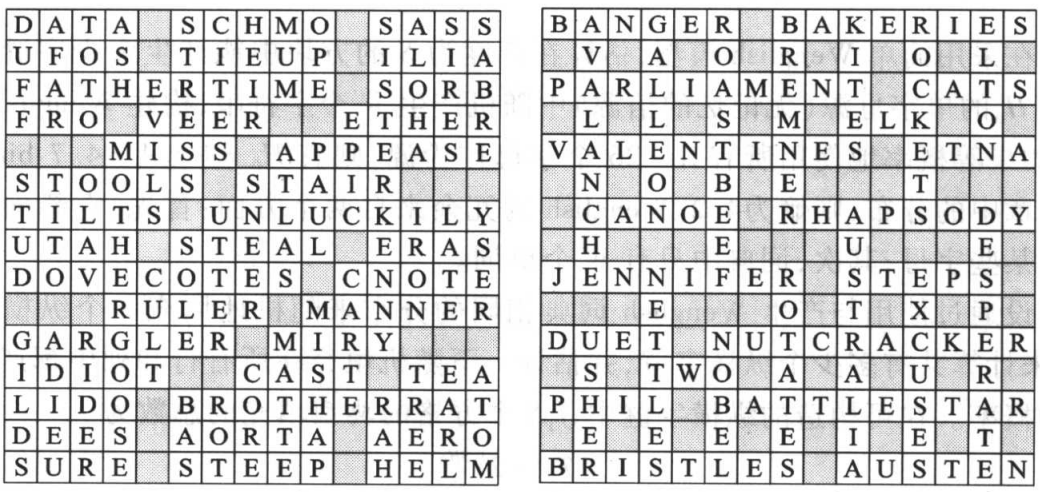


图 18.1　A 类（美国式）和 B 类（英国式）纵横字谜

因为有不允许存在单个字符的限制，故设计一个 A 类字谜比设计一个 B 类字谜更难。B 类字谜通常较难解答，因为每个字符中的限定更少。

为什么字谜可能存在？

如果一种语言没有冗余，则写在一个格子里的任意一个字母都可形成一个有效

纵横字谜。反之,如果语言的冗余度很高,就很难构造一个纵横字谜了(也许有一小部分很明显的除外)。一种语言里能形成纵横字谜的可能性也证实了这种语言的冗余度的界。纵横字谜通常不是用纯正的英语写成的,而是用"单词英语"写成的。所谓"单词英语"是由词典中的单词串加上空格所组成的一种语言。

▷**习题 18.1**[✓] 估计"单词英语"的容量(单位为 bit/字符)。[提示:将"单词英语"想象成定义一个受限信道(见第 17 章),然后参考习题 6.18。]

可以构造许多纵横字谜的事实,也致使"单词英语"的熵的界更低。

为了简单起见,将单词英语建模为 Wenglish,在第 4.1 节介绍过这种语言,它由 W 个长度均为 L 的单词组成。包括单词之间的空格,这种语言中每个字符的熵为

$$H_W \equiv \frac{\log_2 W}{L+1} \tag{18.1}$$

我们将得出结论:要依赖于 H_W 值,但对 L 值并不十分敏感。考虑区域中有 S 个方格的一个大字谜。设单词数量为 $f_w S$,字母所占用的方格为 $f_1 S$。对由长度为 L 的单词所构成的典型 A 类和 B 类字谜,两个因子 f_w 和 f_1 的大概数值如表 18.2 所示。

表 18.2 单词数和字母方格数分别比方格总数小的因子 f_w 和 f_1

	A	B
f_w	$\frac{2}{L+1}$	$\frac{1}{L+1}$
f_1	$\frac{L}{L+1}$	$\frac{3}{4}\ \frac{L}{L+1}$

现在采用简单 Wenglish 模型,估计在长度为 S 的方块中能产生多少字谜。假设从熵为 H_0 的单字母源(无记忆的信源)中随机产生 W 个字符串,构建 Wenglish 语言。比如,如果等概率地使用所有 $A=26$ 个字母作为源,则有 $H_0 = \log_2 A = 4.7$ bit。若改用第 2 章中的分布,则熵为4.2。Wenglish 的冗余来自两个方面:首先,它趋向于更多地采用某些字母;其次,词典中只有 W 个单词。

假设通过使用与产生 Wenglish 词典相同的分布来随机地填充一个纵横字谜的方格,来计算共有多少个纵横字谜,并估计一下随机填充在所有行和列中都产生有效单词的概率。在可构造的纵横字谜中,$f_1 S$ 个方格的典型填充的总数为

$$|T| = 2^{f_1 S H_0} \tag{18.2}$$

长度为 L 的单词能进行有效填充的概率为

$$\beta = \frac{W}{2^{LH_0}} \tag{18.3}$$

由 $f_w S$ 个单词所组成的整个字谜用单一典型填充进行有效填充的概率大约为①

$$\beta^{f_w S} \tag{18.4}$$

① 这个计算低估了有效 Wenglish 字谜的个数,因为只计算了用"典型"字符串所填充的字谜。如果单字母分布是非均匀的,则真实的计数是由"非典型"填充所主导的,其中适合于字谜的单词的出现频度较高。

因此大小为 S 的有效纵横字谜数的对数，可估计为

$$\log\beta^{f_w S}|T| = S[(f_1 - f_w L)H_0 + f_w \log W] \tag{18.5}$$

$$= S[(f_1 - f_w L)H_0 + f_w (L+1)H_W] \tag{18.6}$$

如果下式成立，它是一个关于 S 的递增函数。

$$(f_1 - f_w L)H_0 + f_w (L+1)H_W > 0 \tag{18.7}$$

因此只要 Wenglish 词典中有足够的单词，使得

$$H_W > \frac{(f_w L - f_1)}{f_w (L+1)} H_0 \tag{18.8}$$

可以制造出任意多的纵横字谜。插入表 18.2 中的 f_1 和 f_w，所得如下。

字谜类型	A	B
字谜条件	$H_W > \frac{1}{2}\frac{L}{L+1}H_0$	$H_W > \frac{1}{4}\frac{L}{L+1}H_0$

如果设 $H_0 = 4.2$ bit，并假设一般讲英语的人的词典中有 $W = 4\,000$ 个单词，所有单词的长度 $L = 5$，则可看到 B 类纵横字谜的条件完全满足，但 A 类纵横字谜的条件只是刚刚满足。这与我们的经验相符，即 A 类字谜中往往有更多晦涩的单词。

进一步的阅读

香农(1948 年)首先进行了关于这些纵横字谜的观察；我从参考文献 Wolf 和 Siegel(1998 年)中读到它们。这个主题与二维受限信道的容量问题关系密切。邮包上的二维条形码就是二维受限信道的一个例子。

习题 18.2[小] 一个二维信道有如下约束定义：在 $N \times N$ 的矩形网格中，每个内部像素都有 8 个相邻像素，其中 4 个必须是黑色，4 个必须是白色。(边界像素周围的黑色像素和白色像素数目不受此限制)。如图 18.3 所示的二进制图案满足这种限制。当 N 很大时，这种信道的容量(以每像素的比特数为单位)是多少？

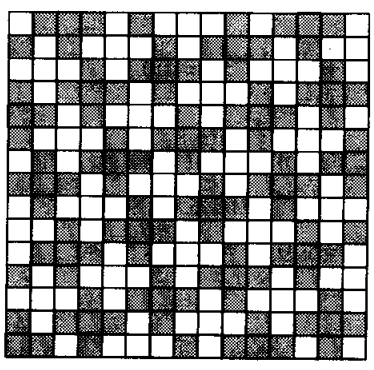


图 18.3 每个像素都与 4 个黑像素和 4 个白像素相邻的二进制图案

18.2 简单的语言模型

Zipf – Mandelbrot 分布

一种最简陋的语言模型是单字组模型,每个连续单词都是从单词的分布中独立抽取的。然而单词的分布特性又是怎样的呢?

Zipf 定理(Zipf,1949 年)断定:一种语言中第 r 个最可能出现的单词的概率近似为

$$P(r) = \frac{\kappa}{r^{\alpha}} \tag{18.9}$$

其中指数 α 的值接近 1,而 κ 是一个常数。根据 Zipf 定理,单词频度与排序的对数坐标曲线应该是一条斜率为 $-\alpha$ 的直线。

Mandelbrot(1982 年)修改了 Zipf 定理,他引入了第 3 个参数 v,认为概率是

$$P(r) = \frac{\kappa}{(r+v)^{\alpha}} \tag{18.10}$$

对于某些资料,如简·奥斯汀的《爱玛》,与 Zipf – Mandelbrot 分布拟合得很好(如图 18.4 所示)。

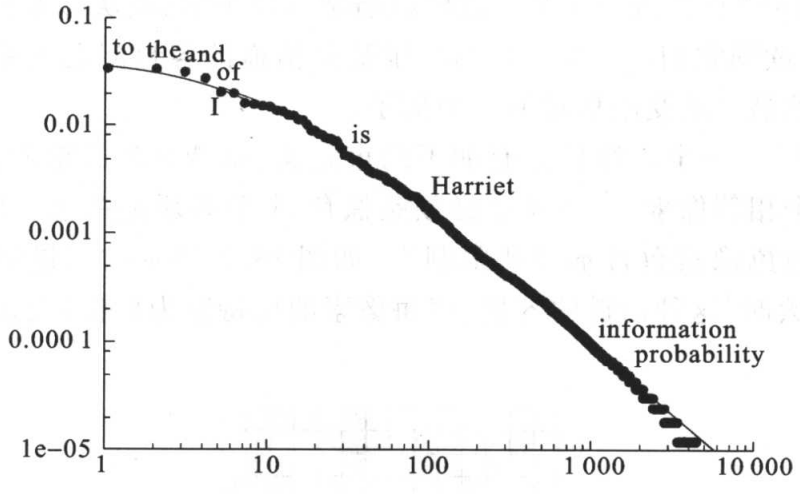


图 18.4 式(18.10)的 Zipf – Mandelbrot 分布(曲线)与简·奥斯汀的《爱玛》中单词的经验频度(点)相符。拟合参数是 $\kappa = 0.56; v = 8.0; \alpha = 1.26$。

其他资料所给出的分布与 Zipf – Mandelbrot 分布不一致。图 18.5 所示是本书英文版的 Latex 源文件中的频度与排序关系。从定量上看,图形类似于一条直线,但明显有些弯曲。为了公平起见,源文件不用纯粹的英文书写,而是英语、数学符号如“x”与 Latex 命令的混合体。

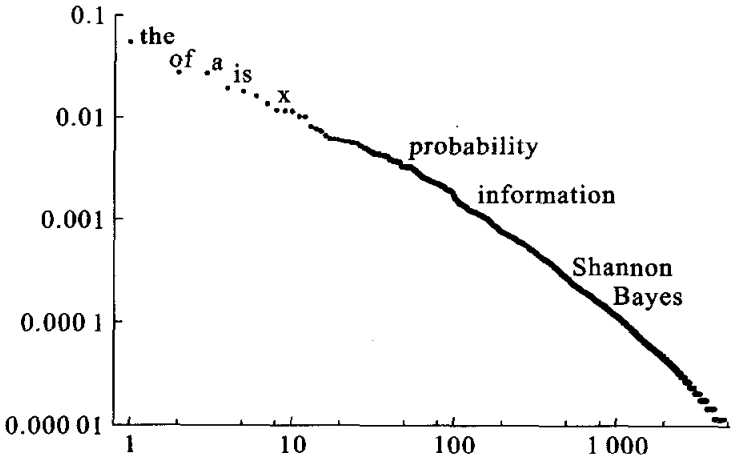


图 18.5 本书英文版 Latex 文件中单词的频度与排序的对数关系图

Dirichlet 过程

假设我们对语言的单字组模型感兴趣,应采用哪一种模型呢? 对语言进行建模的困难之一在于词汇量无限大。语言的样本越大,所遇到的单词越多。一种语言的生成模型应能模拟这种性能。如果问到"最新发现的莎士比亚著作中的下一个单词是什么?",单词的概率分布必须包含以前从未用过的单词的非零概率。而我们的语言生成式单字组模型还必须满足一致性原则,称作可交换性。假设由生成式模型生成一种新的语言,并产生一个不断增大的文本体,文本的所有统计特性是均匀的:在文本流中,在已知位置找到某特定单词的概率应该和在其他地方找到的概率相同。

Dirichlet 过程模型是一个符号流模型(将它当做"单词"),满足可交换性原则且允许词汇表无限制地增大。该模型有一个参数 α。产生字符流时,每个新字符由一个唯一整数 w 识别。当已知了一个长度为 F 的符号流时,就可以依据这些已经出现符号的计数 $\{F_w\}$ 定义下一个符号的概率,因而:下一个字符是一个新字符(以前从未出现过)的概率是

$$\frac{\alpha}{F+\alpha} \tag{18.11}$$

下一个字符是字符 w 的概率是

$$\frac{F_w}{F+\alpha} \tag{18.12}$$

图 18.6 给出了包含百万符号的文件的 Zipf 图(即符号频度与排序曲线),这些符号由 Dirichlet 过程产生,对应 α 值的范围是从 1 到 1 000。

很明显,Dirichlet 过程并不是大致遵循 Zipf 定律的观察分布的一个充分模型。

然而使用一个小技巧,Dirichlet 过程就可以产生相当好的 Zipf 图。假设通过一个具有相当小的参数值 α 的 Dirichlet 过程,生成一种由基本符号所组成的语言,使得

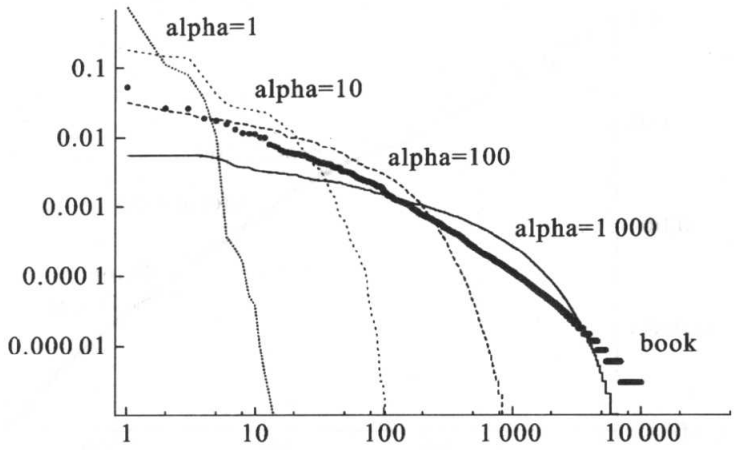


图 18.6 从 Dirichlet 过程随机产生的 4 种“语言”的 Zipf 图，其参数 α 的范围是从 1 到 1 000。同时还画出了本书的 Zipf 图。

合理地频繁出现的符号数目大约为 27。若再断定其中一个符号(现在称为“字符”而非单词)是空格符，我们就能确定空格符之间的字符串是一个“单词”。用这种方法生成一种语言，则单词出现的频度经常是很好的 Zipf 曲线，如图 18.7 所示。选择哪个字符作为空格决定了 Zipf 曲线的斜率——空格的概率越小，所生成的语言越丰富，而且曲线的斜率也越小。

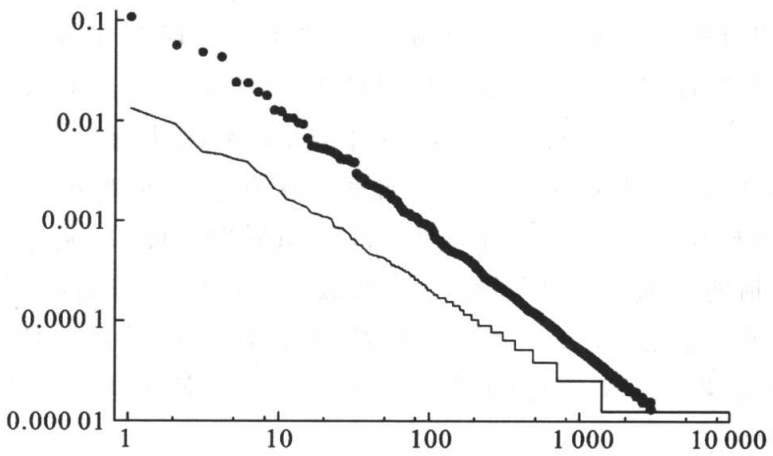


图 18.7 用 $\alpha = 2$ 的 Dirichlet 过程所创建连续字符生成的两种“语言”中单词的 Zipf 图，宣称其中一个字符是空格符。两种空格符的不同选择所对应的两条曲线。

18.3 信息量单位

一个概率为 $P(x)$ 的结果 x 的信息量定义为

$$h(x) = \log \frac{1}{P(x)} \tag{18.13}$$

总体的熵是一个平均信息量

$$H(X) = \sum_x P(x)\log\frac{1}{P(x)} \tag{18.14}$$

当针对数据对各种假设进行比较时，在另外的假设下，比较数据概率的对数通常会比较方便。

$$\mathscr{H}_i \text{ 的对数证据} = \log P(D|\mathscr{H}_i) \tag{18.15}$$

或者，在刚好比较两个假设的情况下，计算“对数几率”

$$\log\frac{P(D|\mathscr{H}_1)}{P(D|\mathscr{H}_2)} \tag{18.16}$$

已将它称为“有利于 $\mathscr{H}_1$ 的证据权重”。一种假设的对数证据 $\log P(D|\mathscr{H}_i)$ 是数据 D 的信息量的负值：如果在已知一种假设的前提下，数据含有很大的信息量，则表示该假设令人惊讶；如果某个其他的假设并不觉得数据令人惊讶，那么那个假设就变得更加可能。“信息量”、“惊奇值”和对数似然或对数证据都是一回事。

所有这些量都是概率的对数值，或者是概率对数的加权和，因此都可以用相同单位来度量。单位取决于所选择的对数底。

如表 18.8 所示为这些单位的名称。

表 18.8 度量信息量的单位

单位	具有这些单位的表达式
比特	$\log_2 P$
奈特	$\log_e P$
班	$\log_{10} P$
分班(db)	$10\log_{10} P$

比特是本书中用得最多的一个单位。因为单词“比特”还有其他含义，该单位的另外一个名称是“香农”。一个字节包含 8 比特。1MB 有 $2^{20} \simeq 10^6$B。如果使用自然对数法，信息量和证据权重都以奈特(nat)为单位来度量。最有趣的单位是班(ban)和分班(deciban)。

班的历史

先讲述一下为什么概率中一个大小为 10 的因子称为班(ban)。当图灵和在 Bletchley 公园的其他译码者破译每新的一天的 Enigma 码时，他们面临的任务是一个非常巨大的推理问题：给定当天要破译的密文，要推断那一天在 Enigma 机下是哪 3 个轮子？他们的初始位置在哪里？在 steckerboard 上还使用了哪些字母替换？以及至少要知道最初的德语消息是什么？这些推理都采用贝叶斯方法来进行(当然是!)，所选择的单位是分班(deciban)或者半分班(half-deciban)，而分班被认为是人类可辨别的最轻量级的证据。将支持特定假设的证据记录在纸上，这种纸是在距离 Bletchley 30 英里的一个叫 Banbury 的小镇印刷的。这个推理任务就叫做 Banburis-

mus,而在进行这种推理的过程中,所用的单位就以那个镇的名字命名为班(ban)。

18.4 浅探 Banburismus

Bletchley 公园密码破译方法的细节被保密了很长一段时间,但 Banburismus 的一些侧面还是可以被拼凑起来的。但愿以下对 Banburismus 的一小部分描述不会太不准确。①

到底需要多少信息? Enigma 机的可能设置数大约是 8×10^{12}。如 Good 所述,为了推断机器的状态,"必须从某个地方找到近 129 分班"。Banburismus 的目标不是推导出机器的所有状态,而是指出所使用的是哪几个轮子;这些填充了明文假设(crib)的基于逻辑的"炸弹"被用来破译轮子是怎样设置的。

一旦定好了轮子和插销的位置,Enigma 机就可以实现连续变化的排列密码,该密码在 26^3 种排列所组成的状态空间中确定性地移动。由于每天都要发送大量消息,无论一个机器在发送一个消息的一个字符时处于何种状态,很可能有另一台机器在发送另一个消息中的某一特定字符时处于同样的状态下。由于机器状态的演化是确定的,两台机器在进行余下部分的传送时将保持在同一状态。这些成对的机器的输出之间所存在的相关性提供了一点信息量,从这些信息中,图灵和他的同事每天抽取 129 分班。

如何检测两个消息来自具有相同状态序列的机器?

$\mathscr{H}_0$ 是一个空假设,声称机器处于不同状态,且两个明文消息也是不相关的;与之"匹配"的假设 $\mathscr{H}_1$ 声明机器处于相同状态,且两个明文消息是无关的。对任何机器所处的状态不做推断。所提供的数据是两个密文 $\boldsymbol{x}$ 和 $\boldsymbol{y}$;假设二者长度均为 T,且字母表的大小为 A(Enigma 机中为 26)。在已知两个假设的条件下,数据的概率是多少?

首先,考虑空假设。这个假设断定两个密文由

$$\boldsymbol{x}=x_1x_2x_3\cdots=c_1(u_1)c_2(u_2)c_3(u_3)\cdots \tag{18.17}$$

和

$$\boldsymbol{y}=y_1y_2y_3\cdots=c_1'(v_1)c_2'(v_2)c_3'(v_3)\cdots \tag{18.18}$$

给出,其中编码 c_t 和 c_t'是字母表中两个无关的时变排列,$u_1u_2u_3\cdots$和 $v_1v_2v_3\cdots$是明文消息。一个数据$(\boldsymbol{x},\boldsymbol{y})$概率的精确计算取决于明文的语言模型和 Enigma 机的内部模式,但是如果假设每个 Enigma 机都是一个理想的随机时变排列,那么两个密文的概率分布是均匀的。所有密文都是等可能的。

① 对笔者帮助最大的描述来自 Tony Sale(http://www.codesandciphers.org.uk/lectures/)和曾与图灵在 Bletchley 公园共事过的 Jack Good(1979 年)。

$$P(\boldsymbol{x},\boldsymbol{y}\mid\mathscr{H}_0)=\left(\frac{1}{A}\right)^{2T},\text{对于所有长度为 } T \text{ 的 } \boldsymbol{x}、\boldsymbol{y} \tag{18.19}$$

假设 $\mathscr{H}_1$ 情况又怎样？这个假设断定

$$\boldsymbol{x}=x_1x_2x_3\cdots=c_1(u_1)c_2(u_2)c_3(u_3)\cdots \tag{18.20}$$

和

$$\boldsymbol{y}=y_1y_2y_3\cdots=c_1(v_1)c_2(v_2)c_3(v_3)\cdots \tag{18.21}$$

二者存在单一的时变排列 c_t。数据$(\boldsymbol{x},\boldsymbol{y})$的概率是多少？必须对明文语言做一些假设。如果明文语言是完全随机的，则 $u_1u_2u_3\cdots$和 $v_1v_2v_3\cdots$的概率也是均匀分布的，而 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 也是如此，因此概率 $P(\boldsymbol{x},\boldsymbol{y}\mid\mathscr{H}_1)$必定等于 $P(\boldsymbol{x},\boldsymbol{y}\mid\mathscr{H}_0)$，两个假设 $\mathscr{H}_0$ 和 $\mathscr{H}_1$ 将不可区分。

假设明文并非完全随机的，可取得一些成果。两段明文都用同一种语言书写，且该语言中存在冗余。比如，假设一些特定明文字母的使用概率比其他字母的使用概率大。因此，尽管两个明文消息是无关的，它们彼此采用一些相同字母的可能性也稍微大了一些；如果 $\mathscr{H}_1$ 为真，两个密文中的两个同步字母相同可能性也稍微大了一些。类似地，如果一种语言频繁采用特殊的双字母组和三字母组，则两个明文消息偶尔会彼此同时包含相同的双字母组和三字母组；如果 $\mathscr{H}_1$ 为真，这会导致包含 2 个或 3 个相同字母的小尖峰。表 18.9 给出了两个无关明文消息（除非都用英语写成）中一种这样的巧合。

表 18.9 **两段对齐的英语明文 u 和v，其中匹配字母用 * 号标记。注意：总共有 12 种匹配，包括一个连续 6 个的匹配，而完全随机的长度为 $T=74$ 字符串中所期望的匹配数约为 3。来自两个处于相同状态的机器的对应密文也会有 12 种匹配。**

u	LITTLE – JACK – HORNER – SAT – IN – THE – CORNER – EATING – A – CHRISTMAS – PIE – – HE – PUT – IN – H
v	RIDE – A – COCK – HORSE – TO – BANBURY – CROSS – TO – SEE – A – FINE – LADY – UPON – A – WHITE – HORSE
匹配	. *.... *.. * * * * * * . *..... *..... *..... *.....

密码破译者通过计算匹配单字母、双字母组和三字母组等的个数，在成对的消息中寻找那些可能彼此相似的消息对。这种方法最早为波兰密码破译者 Rejewski 所采用。

现在来看单字母组语言模型的简单例子，并估计需要多长一条消息才能判断出两台机器是否处于相同状态。假设源语言是单字母英语，其中连续字母是从图 2.1 所示概率分布$\{p_i\}$中独立同等分布地抽取出来的。$\boldsymbol{x}$ 和 $\boldsymbol{y}$ 的概率是非均匀的：考虑两个单一字符 $x_t=c_t(u_t)$和 $y_t=c_t(v_t)$；二者相同的概率为

$$\sum_{u_t,v_t}P(u_t)P(v_t)\,\mathbb{1}[u_t=v_t]=\sum_i p_i^2\equiv m \tag{18.22}$$

称该“匹配概率”为 m；对于英语和德语，m 约为 2/26 而非 1/26（后者是用于完全随机语言的数值）。假设 c_t 是一个理想的随机排列，则由对称性得 x_t 和 y_t 的概率为

$$P(x_t, y_t \mid \mathcal{H}_1) = \begin{cases} \dfrac{m}{A}, & x_t = y_t \\ \dfrac{(1-m)}{A(A-1)}, & x_t \neq y_t \end{cases} \tag{18.23}$$

已知一对长度为 T 的密文 $\boldsymbol{x}$ 和 $\boldsymbol{y}$，二者有 M 处相匹配，有 N 处不匹配，有利于假设 $\mathcal{H}_1$ 的对数证据为

$$\log \frac{P(\boldsymbol{x}, \boldsymbol{y} \mid \mathcal{H}_1)}{P(\boldsymbol{x}, \boldsymbol{y} \mid \mathcal{H}_0)} = M \log \frac{m/A}{1/A^2} + N \log \frac{\dfrac{(1-m)}{A(A-1)}}{1/A^2} \tag{18.24}$$

$$= M \log mA + N \log \frac{(1-m)A}{A-1} \tag{18.25}$$

每一匹配有利于假设 $\mathcal{H}_1$ 的贡献为 $\log mA$；每一不匹配有利于假设 $\mathcal{H}_0$ 的贡献为 $\log \dfrac{A-1}{(1-m)A}$。

单字母组英语的匹配概率	m	0.076
偶然匹配概率	$1/A$	0.037
$\mathcal{H}_1$ 中每个匹配的对数证据	$10 \log_{10} mA$	3.1db
$\mathcal{H}_1$ 中每个不匹配的对数证据	$10 \log_{10} \dfrac{(1-m)A}{(A-1)}$	-0.18db

比如，若长度为 $T=51$ 的一对密文中有 $M=4$ 个匹配，$N=47$ 个不匹配，有利于假设 $\mathcal{H}_1$ 的证据权重将增加 4 个分班，或者以似然比 2.5∶1 有利于假设 $\mathcal{H}_1$。

一行长度为 $T=20$ 个字符的文本的期望证据权重是式(18.25)的期望值，它取决于 $\mathcal{H}_1$ 为真还是 $\mathcal{H}_0$ 为真。如果 $\mathcal{H}_1$ 为真，则期望以速率 m 出现匹配，期望证据权重为每 20 字符 1.4 分班。如果 $\mathcal{H}_0$ 为真，则期望以速率 $1/A$ 出现虚假匹配，期望证据权重为每 20 字符 -1.1 分班。在典型情况下，大约需要检验 400 字符，才可以得到大于 100∶1（20 分班）的证据权重以支持某一假设。

因此两个英文明文比两个随机字符串有更多的匹配。此外，由于英文中的连续字符不是独立的，英文中的双字母组和三字母组的统计是非均匀的，匹配往往是突发的连续匹配。[同样的观察也适用于德语。]采用更好的语言模型，可以更加精确地计算所出现的匹配所提供的证据。这样一个评分系统是由图灵发现、由 Good 精化的。将正面结果传给自动的和人工的密码破译者。据 Good 所述，最长的假正性结果是，在实际上处于无关状态的两台机器上出现的 8 个连续匹配的字符串。

进一步的阅读

关于图灵和 Bletchley 公园的进一步阅读材料可参见参考文献 Hodges(1983 年)

和 Good(1979 年)。关于密码学的深入阅读材料,笔者极力推荐 Schneier(1996 年)。此书可读性强,论述清晰,也很有趣。

18.5 习 题

▷ **习题 18.3**[♂] Enigma 机想尽量模拟一个完美的随机时变排列,但它的设计中存在另一缺陷:它从不将一个字母映射到其自身。当按下 Q 时,所出现的总是一个不同于 Q 的字母。这种设计缺陷会使每个字符的信息量泄露多少呢?需要多长的译文才能确保译文与密文完全对应?又需要多长的译文才能确保识别出正确密钥呢?

[一个译文就是对明文的一种猜想。假设英国人知道有一位德国要人将从柏林前往亚琛,并且他们窃听到发往亚琛的 Enigma 编码消息。一个很好的猜想是,一个或多个原始明文消息包含字符串 OBERSTURMBANNFUEHRERXGRAFXHEINRICHXVONXWEIZSAECKER,即这位要人的名字。可以在蛮力法中通过一个译文来寻找正确的 Enigma 密钥(将接收消息输入所有可能的 Enigma 机,看这些解码后的文本中是否有与以上明文相匹配的)。这个问题集中在可以用一种低廉得多的方式使用译文这一思想:沿着所有编码消息移动明文译文,直至找到译文和编码消息之间的完全不匹配;如果正确,这种对准就能告诉你关于这个密钥的许多信息。]

第 19 章

为何有性？信息获取与进化

地球上的进化过程在过去 10^9 年间一直进行。毋庸置疑，在进化过程中获取了信息。感谢“盲眼钟表匠”不知疲倦的工作，一些细胞携带了成为杰出蜘蛛所需的所有信息，而另一些细胞携带了成为上等章鱼所需的所有信息。这些信息从何而来？

整个地球上的生物体蓝图形成于一个教学过程，其中的教师是自然选择：适者昌盛，这里的适应性是由本地环境（包括其他生物体）定义的。教导信号只是每一个体中少许的几个比特：一个个体具有少量或大量的子孙，取决于该个体的适应性。“适应性”是一个广义概念，包括如下能力。

- 一头羚羊具有比其他羚羊跑得快的能力，因而避免了被狮子吃掉。
- 一头狮子有能力伪装得足够好，并且跑得足够快，因而每天都能捕捉到一头羚羊。
- 一只雄孔雀吸引一只雌孔雀与之交配的能力。
- 一只雌孔雀同时孵很多幼孔雀的能力。

一种生物体的适应性很大程度上取决于其本身的 DNA——包括编码区或者基因以及非编码区（在调节基因的复制中起着重要作用）。我们将适应性视做 DNA 序列和环境的一个函数。

DNA 如何决定适应能力，基因组又是如何从自然选择中获取信息的呢？如果对一头羚羊的蛋白质进行编码的基因存在缺陷，则这只羚羊可能在只有两个（而非 40 个）后代的时候就被狮子吃掉了。自然选择的信息量完全包含在后代存活下来并繁衍新的后代这一指标中，每一个后代最多只有 1 比特的信息量。教导信号不会与生态系统进行通信，即关于某个体中存在不完美性致使其拥有较少后代的描述。教导信号的比特是高度冗余的，因为在一个物种中，彼此相似的不适应环境的个体会因为相似的原因而没有后代。

那么，通过自然选择，一个物种作为总体，每一代能获得多少比特？自从 4 000 000年前，南方古猿与猿猴分道扬镳以来，自然选择成功地传递了多少比特到生命树的人类分枝上呢？假设对于繁殖，10 年为一代，则从与猿分离开始人类祖先已繁衍了 400 000 代。假设人口为 10^9，每个人从自然选择中获得 2 比特信息，负责将公元 400 万年前的基因修改为当今人类基因的信息比特总数约为 8×10^{14} 比特。然而，像我们注意到的那样，自然选择并不善于对分散到人群中的信息进行比较，因而那些信息中存在大量冗余。如果人口规模增大至两倍，信息比特也会增大至两倍吗？不会，因为自然选择经常会简单地将同一缺陷纠正两次。

John Maynard Smith 认为一个物种获取信息的速率与群体大小无关，且在每代 1 比特这个数量级上。这种观点表明在猿与人类之间只有 400 000 比特的差别，远小

于人类基因组本身的总大小 6×10^9 比特。[一个人类基因组大约包含 3×10^9 个核苷酸。]猿与人类的基因组之间肯定存在巨大的重叠,但差别会那么小吗?

有两个缺陷的基因比有一个缺陷的基因更容易导致生理缺陷,一个具有两个缺陷基因的生物体比只具有一个缺陷基因的生物体更不适应环境。基于以上假设,本章将建立一个粗略的模型来描述通过进化获取信息的过程。毫无疑问,该模型只是一个粗略的模型,因为真实的生物学系统像具有复杂相互作用的巴罗克建筑。然而,我们坚持采用简单的模型,因为它容易得出显著的结果。

从这个简单模型中,可以发现以下几点。

1. 对一个无性生殖群体而言,John Maynard Smith 的每代 1 比特的值是正确的。

2. 相反的,如果物种进行的是有性繁殖,每代获得信息的速率可高达每代$\sqrt{G}$比特,其中 G 是基因组的大小。

同时,还可以发现一些与物种能够容忍的最大突变速率有关的有趣结果。

19.1 模　型

研究一个基因组大小为 G 比特、有 N 个个体的繁殖群体的简单模型:通过突变或重组(即交配)产生变异,并通过截断选择在每一代的后代中选取 N 个最合适的个体做下一代的双亲。具有重组的群体与无重组的群体之间存在着十分显著的差异。

每个个体的基因型是有 G 比特的一个向量 $\boldsymbol{x}$,每个元素都有好状态 $x_g=1$ 和坏状态 $x_g=0$。一个个体的适应性函数 $F(\boldsymbol{x})$ 简单地为其各个比特之和

$$F(\boldsymbol{x})=\sum_{g=1}^{G}x_g \tag{19.1}$$

基因组中的比特可以看做对应于好等位基因($x_g=1$)或坏等位基因($x_g=0$),或者看做对应于基因的核苷酸。我们主要采用后一种解释。假设适应性的本质特性为:它大致是一个基因组的局部线性函数,也就是说,基因组可以有多种可能的改变,每一种对适应性都有较小的影响,这些影响近似地以线性关系组合在一起。

定义归一化的适应性为 $f(\boldsymbol{x})\equiv F(\boldsymbol{x})/G$。

我们在两个变异模型下通过自然选择来考虑进化。

突变变异　模型假定每代都是离散的。在每一代 t 中,每一个体生产两个孩子。孩子的基因型由于随机突变而不同于其父母。自然选择在子群体中选取最适应环境的 N 个后裔继续繁殖,于是新的一代就开始了。

[在每一代中选择 N 个最适应环境的个体叫做截断选择。]

最简单的突变模型是后代的比特$\{x_g\}$是独立的。每一比特都有一个小的翻转概率。将这些比特粗略地看做与一些核苷酸对应,可以假设翻转概率是与 x_g 无关的常数 m。[如果在另一方面认为比特对应于基因,可以将发现一个好基因的概率 $P(x_g=0\to x_g=1)$ 建模为小于好基因发生恶性突变的概率 $P(x_g=1\to x_g=0)$。]

基因重组变异（交叉或交配） 我们的生物体是单倍体而非二倍体的。他们通过交配实现重组。可以将群体中的 N 个个体随机地配对成 $M=N/2$ 对，而每一对有 C 个后代——$C=4$ 个后代是标准假设，以便像以前那样使群体翻倍再在每一代减半。已知父母的基因型后，C 个后代的基因型相互独立。每个后代的基因型 $\boldsymbol{z}$ 由父母的基因型 $\boldsymbol{x},\boldsymbol{y}$ 随机交叉得到。最简单的重组模型没有联动，所以

$$z_g=\begin{cases}x_g, & \text{概率为 } 1/2\\ y_g, & \text{概率为 } 1/2\end{cases} \tag{19.2}$$

生产了 MC 个后代后，父母去世，通过自然选择得到最适应环境的 N 个后代，新的一代就开始了。

接下来详细地研究这两个变异模型。

19.2 适应性的增长率

突变理论

假设归一化适应性为 $f=F/G$ 的个体的基因型经历比特翻转概率为 m 的突变。首先证明，如果群体的平均归一化适应性 f 大于 $1/2$，则最优突变速率很小，信息获取速率最多是每代 1 比特这个量级。

因为通过简单突变就很容易得到归一化适应性 $f=1/2$，假设 $f>1/2$，并以额外的归一化适应性 $\delta f\equiv f-1/2$ 的形式进行处理。如果额外归一化适应性为 δf 的个体有一个后代，且突变速率 m 很小，后代的额外归一化适应性概率分布的均值为

$$\overline{\delta f}_{\text{child}}=(1-2m)\delta f \tag{19.3}$$

方差为

$$\frac{m(1-m)}{G}\simeq\frac{m}{G} \tag{19.4}$$

如果父母群体的均值为 $\delta f(t)$，方差为 $\sigma^2(t)\equiv\beta m/G$，则选择之前子群体的均值为 $(1-2m)\delta f(t)$，方差为 $(1+\beta)m/G$。自然选择会选择这种分布的上半部分，因此下一代的平均适应性和适应性方差为

$$\delta f(t+1)=(1-2m)\delta f(t)+\alpha\sqrt{(1+\beta)}\sqrt{\frac{m}{G}} \tag{19.5}$$

$$\sigma^2(t+1)=\gamma(1+\beta)\frac{m}{G} \tag{19.6}$$

其中 α 是与均值之间的均差，以标准偏差度量，γ 是后代分布因选择作用而导致方差减少的因子。α 和 γ 的数量级都是 1。对于高斯分布情况而言，$\alpha=\sqrt{2/\pi}\simeq0.8$，$\gamma=(1-2/\pi)\simeq0.36$。如果假设方差处于动态平衡，也就是 $\sigma^2(t+1)\simeq\sigma^2(t)$，则有

$$\gamma(1+\beta)=\beta,\text{因此}(1+\beta)=\frac{1}{1-\gamma} \tag{19.7}$$

且式(19.5)中的因子 $\alpha\sqrt{1+\beta}$ 等于 1,若将此结果应用到高斯分布中,当适应性的离散性变得重要,即当 m 较小时,近似效果最差。因此,归一化适应性的增长率为

$$\frac{\mathrm{d}f}{\mathrm{d}t}\simeq -2m\,\delta f+\sqrt{\frac{m}{G}} \tag{19.8}$$

假定 $G(\delta f)^2>>1$,则它取最大值的点为

$$m_{\mathrm{opt}}=\frac{1}{16G(\delta f)^2} \tag{19.9}$$

在该点上有

$$\left(\frac{\mathrm{d}f}{\mathrm{d}t}\right)_{\mathrm{opt}}=\frac{1}{8G(\delta f)} \tag{19.10}$$

因此适应性 $F=fG$ 的增长率最多为

$$\frac{\mathrm{d}F}{\mathrm{d}t}=\frac{1}{8(\delta f)}(\text{每一代}) \tag{19.11}$$

对于具有低适应性($\delta f<0.125$)的群体而言,适应性的增长率可能超过每代 1 个单位。事实上,如果 $\delta f\leqslant 1/\sqrt{G}$且 $m=1/2$,,则增长率在$\sqrt{G}$ 数量级;这种最初的爆发所能持续的阶在$\sqrt{G}$ 数量级。对于 $\delta f>0.125$,每一代的适应性增长率小于 1。当适应性接近 G 时,最优突变速率趋近 $m=1/(4G)$,从而平均每个基因型有 1/4 比特的翻转,适应性的增长率也等于 1/4;获得信息的速率大约是每一代 0.5 比特。大概要过 $2G$ 代才能使群体中所有个体的基因型达到完美。

对于固定的 m,适应性为

$$\delta f(t)=\frac{1}{2\sqrt{mG}}(1-c\mathrm{e}^{-2mt}) \tag{19.12}$$

但受到$\delta f(t)\leqslant 1/2$ 的条件约束,其中 c 是一个积分常数,在$f(0)=1/2$ 时,$c=1$。如果每个基因型翻转的平均比特数 mG 大于 1,则适应性 F 接近平衡值 $F_{\mathrm{eqm}}=(1/2+1/(2\sqrt{mG}))G$。

这个理论在某种程度上并不准确,因为实际的适应性概率分布是非高斯的和不对称的,且要量化到整数值。尽管如此,该理论的预测与下述仿真结果并无太大的差别。

有性繁殖理论

采用如下两个近似使对有性群体的分析变得易于处理:首先,假设基因池能足够快地混合,使得基因之间的关联性可被忽略;第二,假设同质性,即比特 g 处在好状态的部分 f_g,对于所有 g 都是相同的 $f(t)$。

已知上述假设,如果适应性为 $F=fG$ 的双亲进行交配,其后代适应性概率分布的均值等于父母的适应性 F;有性生殖所产生的变异并不减小平均适应性。孩子的适应性标准方差在 $\sqrt{Gf(1-f)}$数量级。因为经过选择之后,适应性的增长与这个标准方差成正比,因此每代的适应性按照基因组大小的平方根$\sqrt{G}$增长(如图 19.1 所示)。如图 19.2 所示,平均适应性 $\bar{F}=fG$ 的进化满足如下微分方程

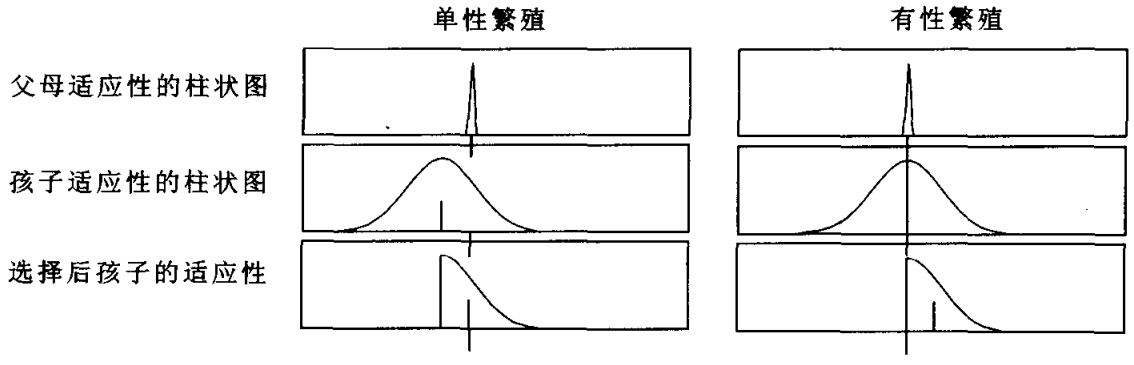


图 19.1 为什么有性繁殖优于无性繁殖。如果利用突变产生后代之间的变异，则孩子的平均适应性低于父母的适应性是不可避免的；变异越大，二者的平均偏差越大。选择会将适应性提到平均值之上。相反的，重组产生变异但不降低平均适应性。典型的变异程度在 $\sqrt{G}$ 数量级，其中 G 是基因组的大小，因此经过选择之后，平均适应性增长 $O(\sqrt{G})$。

$f(t+1)$ 如何依赖于 $f(t)$？首先假设一个孩子的父母刚好都有 $f(t)G$ 个好比特，根据同质性假设，这些比特是 G 个比特的独立随机子集。父母双方都好的比特数目大致为 $f(t)^2G$，而只有单亲好的比特数大约是 $2f(t)(1-f(t))G$，因此孩子的适应性是 $f(t)^2G$ 加上 $2f(t)(1-f(t))G$ 次公平硬币抛掷的总和，它是一个均值为 $f(t)(1-f(t))G$、方差为 $\frac{1}{2}f(t)(1-f(t))G$ 的二项分布。因而孩子的适应性大致分布为

$$F_{\text{child}} \sim \text{正态分布}\left(\text{均值}=f(t)G, \text{方差}=\frac{1}{2}f(t)(1-f(t))G\right)$$

与突变情况下的分布不同，这个分布的重要特性是适应性均值等于父母的适应性；由有性繁殖所产生的变异并未减小平均适应性。

如果考虑父母的群体的方差，即 $\sigma^2(t)=\beta(t)\frac{1}{2}f(t)(1-f(t))G$，孩子群体的适应性分布为

$$F_{\text{child}} \sim \text{正态分布}\left(\text{均值}=f(t)G, \text{方差}=\left(1+\frac{\beta}{2}\right)\frac{1}{2}f(t)(1-f(t))G\right)$$

自然选择将选择位于分布上半部分的孩子。适应性的平均增长为

$$\bar{F}(t+1)-\bar{F}(t)=[\alpha(1+\beta/2)^{1/2}/\sqrt{2}]\sqrt{f(t)(1-f(t))G}$$

生存下来的孩子的适应性方差为

$$\sigma^2(t+1)=\gamma(1+\beta/2)\frac{1}{2}f(t)(1-f(t))G$$

其中 $\alpha=\sqrt{2/\pi}$，$\gamma=(1-2/\pi)$。如果存在动态平衡 $[\sigma^2(t+1)=\sigma^2(t)]$，则式(19.2)中的因子为

$$\alpha(1+\beta/2)^{1/2}/\sqrt{2}=\sqrt{\frac{2}{(\pi+2)}}\simeq 0.62$$

定义常数 $\eta\equiv\sqrt{2/(\pi+2)}$，可以得出结论：在有性繁殖和自然选择的条件下，群体平均适应性的增长率与基因大小的平方根成正比。

$$\frac{\mathrm{d}\bar{F}}{\mathrm{d}t}\simeq\eta\sqrt{f(t)(1-f(t))G}\text{比特(每代)}$$

图 19.2 有性繁殖理论的详情

$$\frac{\mathrm{d}\bar{F}}{\mathrm{d}t} \simeq \eta \sqrt{f(t)(1-f(t))G} \tag{19.13}$$

其中 $\eta \equiv \sqrt{2/(\pi+2)}$，该方程的解为

$$f(t) = \frac{1}{2}\left[1 + \sin\left(\frac{\eta}{\sqrt{G}}(t+c)\right)\right], \quad t+c \in \left(-\frac{\pi}{2}\sqrt{G}/\eta, \frac{\pi}{2}\sqrt{G}/\eta\right) \tag{19.14}$$

其中 c 是一个积分常数，$c = \sin^{-1}(2f(0)-1)$。所以这种理想化系统在有限时间 $(\pi/\eta)\sqrt{G}$ 代之内就可以达到完美的优生状态($f=1$)。

仿真

图 19.3(a)所示为有性群体的适应性情况，其中个体数量为 $N=1\,000$，基因大小为 $G=1\,000$，起始于归一化适应性为 0.5 的随机初始状态。同时也画出了式(19.14)给定的理论曲线 $f(t)G$，二者匹配得非常好。

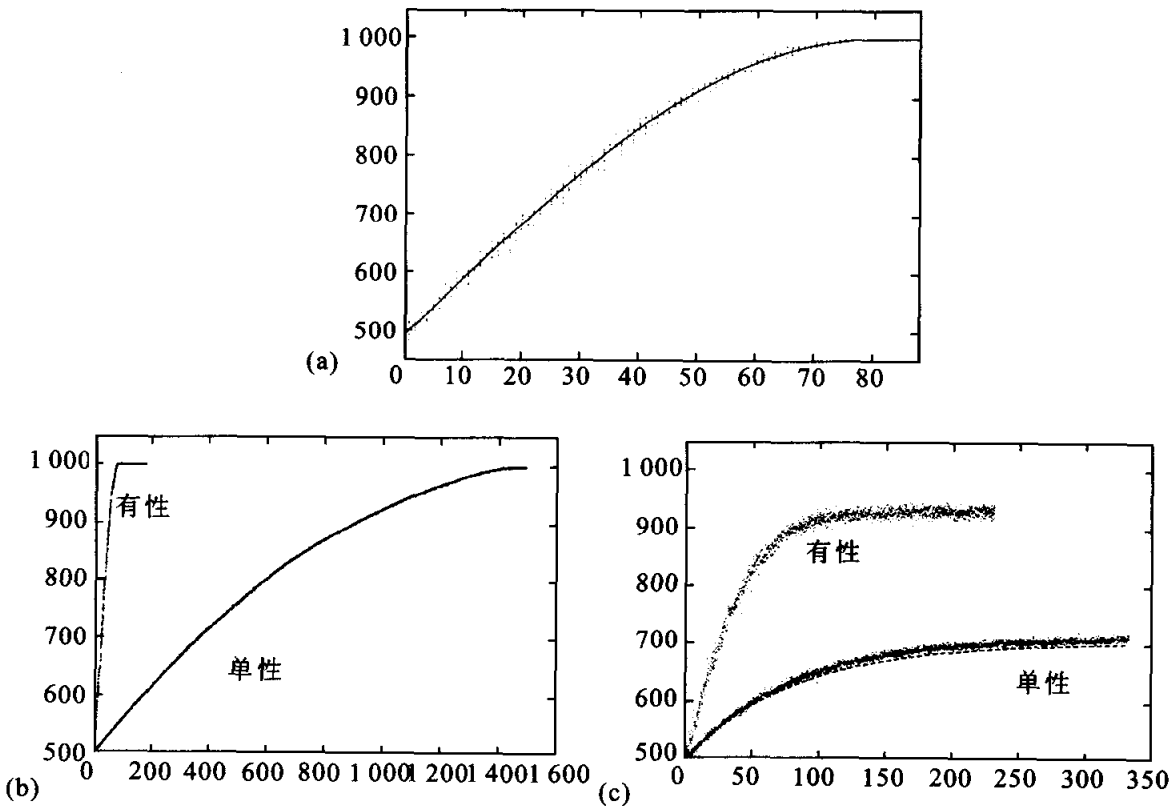


图 19.3 适应性与时间函数的关系。基因组的大小为 $G=1\,000$，圆点表示从每一代新生群体中随机选取的 6 个个体的适应性。初始群体数量为 $N=1\,000$，以 $f(0)=0.5$(准确值)随机地生成基因。(a) 由有性所产生的变异。实线表示无限同质群体的理论曲线，即式(19.14)的曲线。(b)和(c)中的变异是在有性和无性繁殖下由突变产生的，突变速率分别为 $mG=0.25$(b)或 6(c)比特(每基因)。虚线表示的是式(19.12)的曲线。

与之相对，图 19.3(b)和图 19.3(c)分别给出了以速率 $m=0.25/G$ 和 $m=6/G$

由突变产生变异时进化的适应性。注意：与图19.3(a)横坐标的比例不同。

习题19.1[♪] **与群体大小相关** 一个有性群体对群体大小的依赖结果如何？预测到有性繁殖理论准确的条件是群体大小必须大于一个最小值。然而与 G 相关的最小群体大小是多少？

习题19.2[♪] **与交叉机制相关** 在简单的性模型中，每个比特都是随机从双亲之一获得的，也就是，允许任意两个相邻核苷酸之间发生交叉的概率为50%。(a) 如果交叉概率小于50%时；(b) 如果每 d 个比特基因中，交叉仅发生在热点位置时；模型影响如何？

19.3 最大可容忍突变速率

如果将两个变异模型结合起来，会出现什么情况？一个有性物种可容忍的最大突变速率是多少？如图19.4所示。

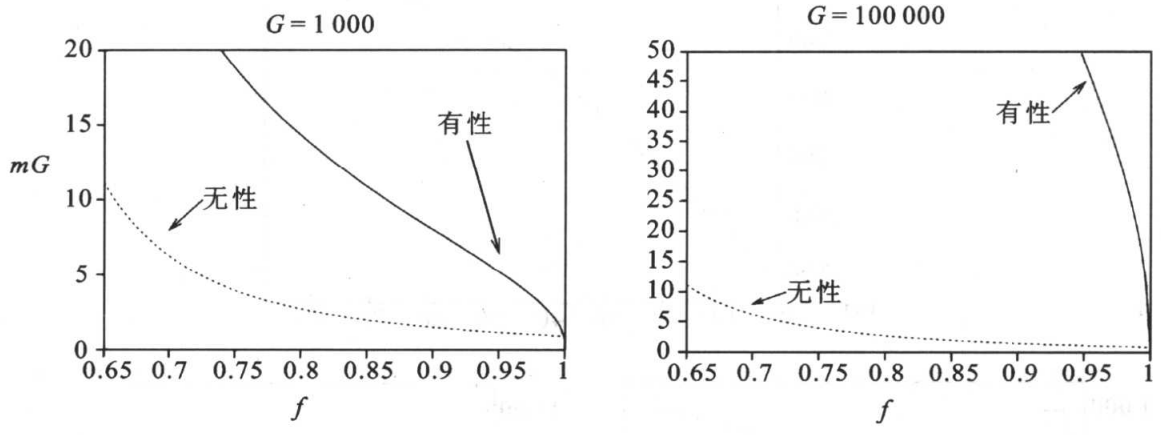


图19.4 最大可容忍突变速率，以每个基因(mG)的错误数量与归一化适应性 $f = F/G$ 之比表示。左图：基因大小为 $G = 1\,000$，右图：$G = 100\,000$。基因大小彼此独立，单性生殖(或无性)只能接受每代每个基因错误为1次幂的范围；使用重组(有性繁殖)的物种能够容忍大得多的突变速率。

适应性的增长率由下式给定。

$$\frac{\mathrm{d}f}{\mathrm{d}t} \simeq -2m\delta f + \eta\sqrt{2}\sqrt{\frac{m + f(1-f)/2}{G}} \tag{19.15}$$

若突变速率满足条件

$$m < \eta\sqrt{\frac{f(1-f)}{G}} \tag{19.16}$$

适应性增长率为正。现在将这个速率与无性物种的结果进行比较。从式(19.8)得到无性时最大可容忍的突变速率为

$$m < \frac{1}{G}\frac{1}{(2\delta f)^2} \tag{19.17}$$

可见,有性繁殖可容忍的突变速率比无性繁殖可容忍的突变速率大$\sqrt{G}$倍的量级!

单性繁殖(无性)物种可以通过增加它后代的个数,来试着避开这个突变速率限。但如果突变平均翻转 mG 比特,一个基因中没有翻转比特的概率大概是 e^{-mG},故大约 e^{mG}个后代中才能有一个与其母亲具有相同的适应性。因此,如果一个无性物种要延续,无性物种后代的数量必须是 mG 的幂(当 mG 大于 1 时)。

因此对于无性物种而言,最大可容忍突变速率十分接近 $1/G$,然而对于可重组的物种而言,它是一个量级为$1/\sqrt{G}$的较大的数。

反过来,对于已知的固定突变速率 m,可以预测最大可能的基因组大小。对于单性生殖物种,最大的基因组大小是 $1/m$ 量级,而对于有性繁殖物种而言,最大的基因组大小是 $1/m^2$ 量级。设每一代中每个核苷酸的突变速率为 $m = 10^{-8}$(EyreWalker 和 Keightley, 1999 年),一次最多允许繁殖 20 000 个后代(也就是 $mG \simeq 10$),可以预测具有多于 $G = 10^9$ 个编码核苷酸的所有物种,至少偶尔要使用重组。如果生育的规模是 12,则这个数字降至 $G = 2.5 \times 10^8$。

19.4 适应性的增加与信息获取

对于简单模型,可以将适应性增加与信息获取联系起来。

如果随机设置比特,适应性大致为 $F = G/2$。如果进化导致某个群体中的所有个体具有最大适应性 $F = G$,则物种就获得了 G 比特的信息,即对于每个比特 x_g,物种能分辨出两种状态中哪一种较好。

定义一个中等适应性上所获信息为:从基因池中选择完美状态所需进行的选择总数(用比特度量)。设 $x_g = 1$ 的群体所占比例为 f_g。因为 $\log_2(1/f)$ 是在一个包含黑球与白球的比例为 f:1 $-f$ 的容器中找到一个黑球所需的信息,定义获取的信息为

$$I = \sum_g \log_2 \frac{f_g}{1/2} \text{ 比特} \tag{19.18}$$

如果所有比例 f_g 等于 F/G,则

$$I = G\log_2 \frac{2F}{G} \tag{19.19}$$

它可用下式很好地近似。

$$\tilde{I} \equiv 2(F - G/2) \tag{19.20}$$

因而信息获取速率大约是群体适应性增长率的两倍。

19.5 讨 论

这些结果从数量上解释了为何物种要进行有性繁殖的著名论点，即重组允许有用突变在物种中更快地传播，允许有害突变更快地从群体中清除（Maynard Smith，1978年；Felsenstein，1985年；Maynard Smith，1988年；Maynard Smith和Száthmary，1995年）。通过重组来繁殖的群体从自然选择中获取信息的速率比单性繁殖群体快$\sqrt{G}$倍的量级，而且所能忍受的突变速率是后者的$\sqrt{G}$倍（量级）。对于大小为$G \simeq 10^8$个编码核苷酸的基因组，$\sqrt{G}$这个因子是很有分量的。

有性繁殖所具有的巨大优势以前在Kondrashov（1988年）中提到过，但是这个meme（Kondrashov对“确定性突变假设”的称呼）好像未能在进化研究领域得到传播，因为至今仍有许多论文将有性繁殖的流行看做有待通过精确机理进行解释的神秘事物。

“雄性的代价”：有性繁殖和单性繁殖中基因的稳定性

为什么人们宣称有性繁殖是神秘的呢？主要是因为一个被称为“雄性的代价”的思想。有人认为有性繁殖相对于无性繁殖而言是不利的，因为有性繁殖的每两个后代中，有一个（平均而言）是无用的雄性，不能生孩子，只有一个是能生产的雌性。同时，一个单性母亲可以生育两个雌性体。从另一方面考虑，单性繁殖的优越性，就个体而言，在于上一代的基因可以100%传给下一代，而不只是50%。因此，如果物种有两个版本，一个是有性繁殖，另一个是无性繁殖，则单亲母亲的基因优越性要超过其有性繁殖的堂兄弟姐妹。这种简单模式至今没有包括性别或者从有性繁殖到无性繁殖转换的能力，但是可以很容易地对它进行修改。

修改模型以便基因中G比特中的一个个体决定是愿意单性繁殖（$x=1$）还是有性繁殖（$x=0$）。结果依赖于一个单性生殖母亲所能生育孩子的数量K_p以及有性伴侣所能生育孩子的数量K_s。（$K_p=2, K_s=4$）和（$K_p=4, K_s=4$）两者都属于合理模型。前者（$K_p=2, K_s=4$）似乎最适合单细胞生物体，其中父母双方的细胞质都进入孩子细胞中。后者（$K_p=4, K_s=4$）适合于后代单独由父母中的一方供养，因此单亲母亲和有性父母具有相同数量的后代。我们主要讨论后一模型，因为它给出了单性繁殖的最大优势，按照推测，预计它会比有性群体有更多的后代。因为单性繁殖每一代有4个孩子，他们的最大可容忍突变速率是用$K_p=2$推导出来的表达式（19.17）之值的两倍。如果适应性很大，最大可容忍突变速率为$mG \simeq 2$。

开始时，基因组随机设置，其中$F=\frac{G}{2}$且一半成员具有单性繁殖基因。图19.5表示结果。在进化的“学习”阶段，适应性增长很快，其中短暂地出现了单性繁殖，但

是在几代之后，他们的堂兄们在适应性方面超过他们，于是单性繁殖就消失了。然而，一旦群体达到最高适应性，如果突变速率足够低（$mG = 1$），单性繁殖又可以重新赶上。

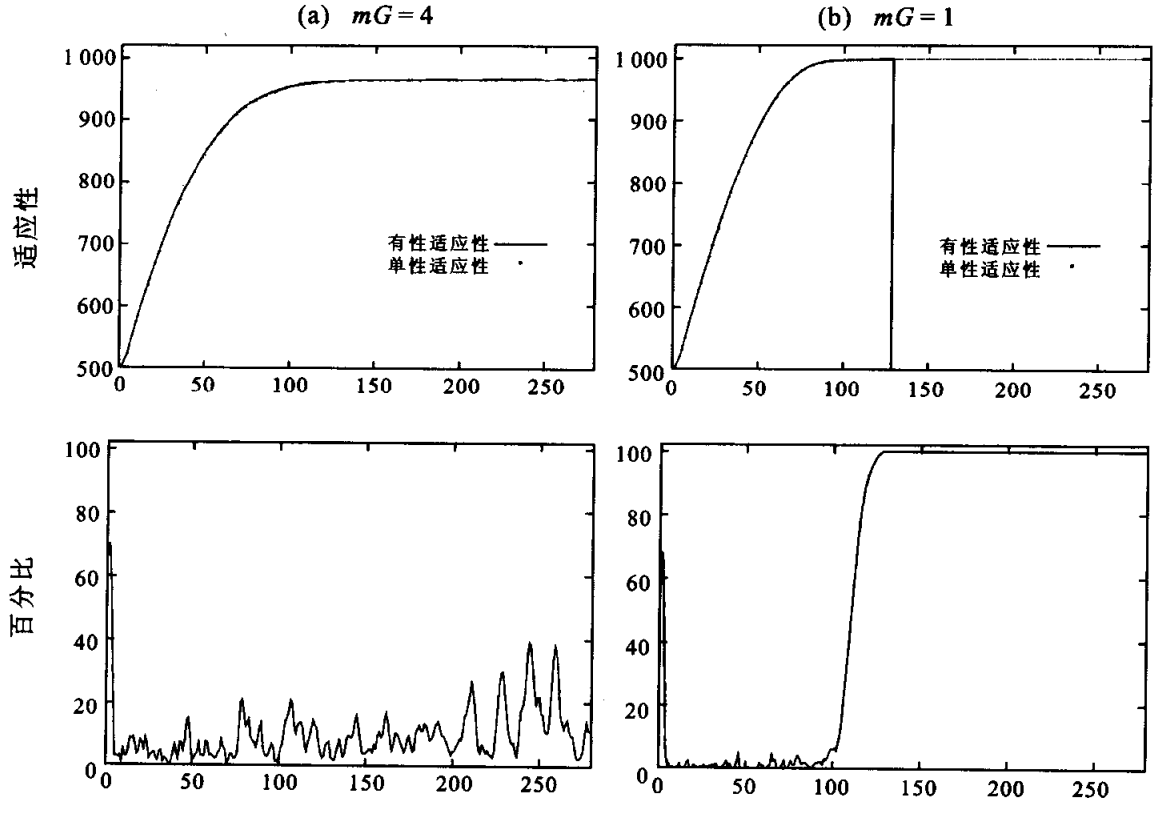


图 19.5　一个基因进行单性繁殖，没有杂交，且单亲母亲生育的孩子和有性伴侣生育的孩子一样多时的结果。$G = 1\,000, N = 1\,000$。(a) $mG = 4$；(b) $mG = 1$。纵轴表示两个子群的适应性和单性繁殖群体的百分比。

然而当出现较高突变速率（$mG = 4$）时，单性繁殖不可能赶上。有性群体适应性宽度的量级为$\sqrt{G}$，因此在单性群体的适应性略高于平均值时，它将持续大约$\sqrt{G}/(mG) = 1/(m\sqrt{G})$代，然后它的适应性再落到有性堂兄们的后面。只要有性群体中某些个体的数量足够多时，使其在此时能够生存，有性繁殖就不会消失。

在一个很不稳定的环境中，其中适应性函数连续变化，单性繁殖总是落后于有性繁殖。这些结果与 Haldane 和 Hamilton（2002 年）关于性在寄生虫的军备竞赛中有所帮助的论点是一致的。寄生虫定义了一种有效的适应性函数，它随时间而变化，而且有性群体总是能更快地提高当前适应性函数。

可加的适应性函数

当然，我们的结果依赖于所假设的适应性函数和所选择的模型。将适应性建模为到第 1 阶的独立项之和，是否合理呢？比如，Maynard Smith（1968 年）就主张：所拥

有的好基因越多，所处的啄序就越高。定向选择模型被广泛用于种群遗传的理论研究中（Bulmer，1985年）。我们可能期望实际的适应性函数应包含相互作用，在此情况下，交叉将会减少平均适应性。然而，因为基因重组使加性适应性函数的物种具有最大优势，可以预测进化会有利于这样一些物种：他们所用的基因表示对应于只有弱交互作用的适应性函数。而且即使存在交互作用，适应性仍将包括这些交互项的总和，这些项的数量是基因组大小 G 的一部分，这种假设似乎也是合理的。

习题 19.3[μC] 如果有性物种和无性物种的适应性函数有交互作用，研究其进化能有多快。例如，设适应性是比特对的异或之和；将此进化适应性、那些有简单加性适应性函数的有性物种和无性物种的适应性进行对比。

此外，如果适应性函数是基因型的高阶非线性函数，则也许可以用 Baldwin 效应将它变得更平滑或使其成为局部线性的。Baldwin 效应（Baldwin，1896年；Hinton 和 Nowlan，1987年）作为一种通过学习引导进化的机制得到了广泛研究，它也可以在基因转录和转译这一层次起作用。考虑为了新用途而进行的缩氨酸序列进化。假设缩氨酸的效用是序列的高阶非线性函数，也许存在被同样差的序列“海洋”所包围的一个好序列“小孤岛”。在一个基因转录和转译机制准确无误的生物体中，适应性同样是 DNA 序列的一个非线性函数，进化也只能通过随机游动方式在海洋中蜿蜒，并逐渐向孤岛移动。相反的，如果生物体具有相同的 DNA 序列，但从 DNA 到 RNA 的转录或者从 RNA 到蛋白质的转译偶尔会出现错误，即出现误转译或误转录，偶然产生一种工作酶；尤其是其 DNA 序列接近好序列时，这种概率会更大。一个细胞可能从一个 mRNA 序列中产生 1 000 个蛋白质，其中 999 个蛋白质是没有酶效应的，只有一个蛋白质有酶效应。这个起作用的催化剂足以保证相对于 DNA 序列更远离好序列孤岛的竞争者，该细胞具有增加的适应性。出于这个原因，我们推测，至少在进化的早期，也许现在还是，基因码并非完美无缺地而是有噪声地完成，对一些可能的氨基酸分布进行密码子的编码。在一个生物体中，通过多种氨基的 tRNA 合成酶，这个噪声编码甚至可以在各个细胞之间打开或关断，其中一些比另一些更可靠。

尽管我们的模型假设基因的比特不会相互影响，忽略了信息是冗余表示的这一事实，假设了显型适应性与基因型之间存在直接联系，假设了重组中的交叉概率较高，笔者相信在采用更复杂的适应性模型和交叉模型时，这些定性的结果仍然成立：有性繁殖的相对益处仍然以量级 $\sqrt{G}$ 增长。只有在数量少、近亲交配的群体中，有性繁殖的益处才有可能减少。

总之：为什么会有性？因为性有益于你的比特！

进一步的阅读

高信息量自复制系统到底是怎么出现的？对于生命起源和关于进化的其他玄妙问题这一广泛领域，笔者极力推荐：Maynard Smith 和 Száthmary（1995年），Maynard Smith 和 Száthmary（1999年），Kondrashov（1988年），Maynard Smith（1988年），Ridley

(2000 年),Dyson(1985 年),Cairns-Smith(1985 年),Hopfield(1978 年)。

19.6 进一步的习题

习题 19.4[考] 已知哺乳动物未在很久以前完全灭绝,则 DNA 复制中的纠错机制必须有多好?估计每次细胞分裂中核苷酸替换概率。[参见附录 C.4。]

习题 19.5[考] 已知 DNA 复制是通过 35℃生化浓汁中不断的布朗运动和普通热力学完成的,DNA 复制的差错概率能达到大约每复制一个核苷酸有 10^{-9}个错误这种水平是很惊人的。已知一个正确的基对与一个错误对之间的能量差异仅仅只是一两个氢键,其热能 kT 只是比与一个氢键相关的自由能量还小 1/4 的一个因子,这个可靠性如何达到?如果普通热力学有利于正确基配对,基对的错误频率应该大约为

$$f=\exp(-\Delta E/kT) \tag{19.21}$$

其中 ΔE 是自由能量差异,即 $f\simeq 10^{-4}$的错误频率吗?DNA 复制是如何骗过热力学的?

这种情况与蛋白质合成的情况同样令人困惑,后者根据基因代码,把一个 mRNA 序列转译成多肽。防止如下两个特别的化学反应错误:把一个 tRNA 分子绑定到氨基酸上,及在核糖体中产生多肽,它和 DNA 复制一样,也涉及基配对。而且其保真度很高(错误率约为 10^{-4}),不能要求"正确"的最终状态具有特别低的能量来达到这个保真度。我们不认为正确多肽序列的能量比其他序列低很多。细胞如何纠错呢?(参考 Hopfield(1974 年), Hopfield (1980 年))。

习题 19.6[考] 遗传基因通过自然选择获取信息的速率为每代几个比特,但是你的大脑获取信息的速率要快很多。

估计你的大脑中能够以什么速率将新信息存入长期记忆。比如,想想学习一门新语言中的单词。

19.7 习题解答

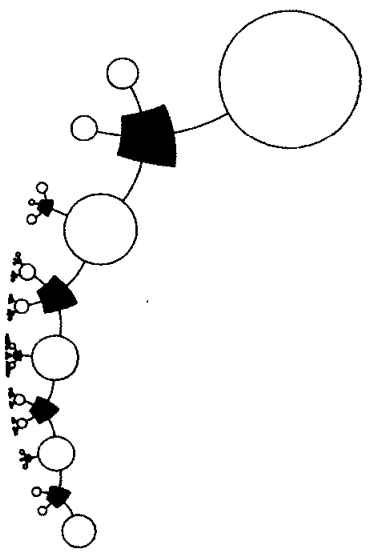
习题 19.1 的解答 对于足够小的 N,尽管群体的平均适应性在增长,一些运气不好的比特还是固定在坏状态。(这些坏基因有时叫做搭便车者。)这就破坏了同质性假设。最后,所有个体具有相同的基因型,它们大多是 1 比特,也有些是 0 比特。预计群体越小,会有越多的比特固定在 0。如果有性理论是准确的,那么群体大小 N 可能有多小呢?

通过实验发现,只要群体尺寸 N 小于 $\sim\sqrt{G}$,基于同质性假设的理论就不太合适。如果 N 远远小于$\sqrt{G}$,就不可能以$\sqrt{G}$ 这样大的速率获取信息,因为"盲眼钟表匠"的

决定所包含的信息量不可能超过每代 $2N$ 比特，这也是指定 $2N$ 个孩子中哪一个进行繁殖所需的比特数。Baum 等人（1995 年）通过分析一个类似的模型证明：要使搭便车者不太可能出现，群体大小 N 应该约为 $\sqrt{G}(\log G)^2$。

第四部分

概率与推理



关于第四部分

可以(也许是应该)采用贝叶斯推理方法来解决的推理问题的数目是巨大的。例如,在本书中,讨论了纠错码的解码问题、从数据中推理聚类的任务、在有噪声的数据中进行插值的任务、已知标记示例进行模式分类的任务。用于解决这些问题的大多数方法可归入以下两类。

精确方法直接计算所需的量。只有一些有趣问题有直接解,但是精确方法是解决大问题中的子任务的重要工具。推理问题的精确解的求法是第21章、第24章、第25章和第26章的主题。

近似方法可以细分为以下几种。

1. **确定性的近似方法** 包括最大似然方法(第22章)、拉普拉斯方法(第27章和第28章)和变参法(第33章)。

2. **蒙特卡罗方法** 这是随机数起主要作用的方法,将在第29章、第30章和第32章讨论。

本书的这一部分并非一维独立的。相反的,本章的思想构成了一张网,网线是互相关联的并将在后续章节中重新组合。

作为这一部分的荣誉章节,第3章讨论了一系列推理问题的简单例子及其贝叶斯解。

为了进一步阐明这一部分将要讨论的推理问题工具箱的动机,第20章讨论聚类问题;随后几章讨论将聚类问题概率地解释为混合建模。

第21章讨论是否能够通过完全枚举所有假设的方法来处理概率分布。作为避免完全枚举所连带的巨大开销的一种方法,第22章介绍最大化方法的思想,并指出最大似然方法不够好的原因。第23章回顾贝叶斯推理中最常用的概率分布。第24章、第25章和第26章提出另一种避免完全枚举所产生开销的方法:边缘化。第25章以纠错码解码为例,讨论适合于图模型的消息传递方法。第26章将第16章和第17章中消息传递的概念结合在一起。这些章节是理解高级纠错码的必备基础。

第27章讨论了确定性的近似方法,包括拉普拉斯方法。这一章是理解学习算法的复杂度控制的必备基础,第28章对学习算法的复杂度控制在普遍意义上进行了讨论。

第29章讨论蒙特卡罗方法。第30章详细描述了蒙特卡罗方法的最新发展状况。

第31章引入伊辛模型作为概率方法的试验台。展示了精确的消息传递方法和蒙特卡罗方法。研究伊辛模型的一个原因是它和几个神经网络模型关系密切。第

32 章描述了“精确”蒙特卡罗方法并说明它在伊辛模型中的应用。

第 33 章讨论变参法及其在伊辛模型和包括聚类在内的简单统计推理问题中的应用。这一章会帮助读者理解 Hopfield 网络(第 42 章)和 EM 算法,这是隐含变量建模中的重要方法。第 34 章讨论一个叫做独立元素分析的特别简单的隐含变量模型。

第 35 章是多种推理主题的集成。第 36 章讨论决策论的一个简单例子。第 37 章讨论抽样理论和贝叶斯方法之间的差别。

一个主题:推理的概念

一个普遍的错误概念:推理的目标是找到一些数据的最可能的解释。虽然人们也许会对这个最有可能的假设感兴趣,一些推理方法也确实能够找到它,这个假设只是概率分布的峰值,而人们感兴趣的是整个分布。正如在第 4 章所见,一个信源中最可能的结果往往不是信源的典型结果。同样,已知某些数据时最有可能的假设在整个看似合理的假设集合中也许是非典型的。

关于第 20 章

在阅读下一章之前,建议完成习题 2.17 并阅读第 11.2 节(推理高斯信道的输入)。

第 20 章

一个推理任务示例:聚类

人类的大脑很擅长在数据中发现规律。表达规则的一种方式是把一个目标集合分成彼此相似的组。例如,生物学家发现自然界中的大部分事物可以归入两类:棕色且可以跑动的,绿色且不能移动的。第一类称为动物,第二类称为植物。把这种将事物组合在一起的做法称作聚类。如果生物学家更进一步将植物类划分成子类,我们称之为"递阶聚类";但是本书不准备讨论递阶聚类。在这一章中,只讨论把 N 个物体的集合分成 K 类的方法。

研究聚类有几方面的原因。首先,一个好的聚类具备预测能力。当早期生物学家遇见他从未见过的绿色事物时,他已有的关于植物和动物的模型可以对这个绿色事物的属性做出一些预测:它不大可能跳到他的身上来吃他;如果接触它,有可能被擦伤或刺痛;如果吃它,有可能会生病。所以这些预测尽管是不确定的,但仍然有用,因为它能帮助生物学家较好地投入资源(例如,花在观察食肉动物上的时间)。人们进行聚类,是因为相信潜在的聚类标记是有意义的:它能使我们对数据进行更有效的描述,能帮助我们采取更好的行动。这种类型的聚类有时叫做"混合密度建模",度量预测模型工作好坏情况的目标函数是数据的信息量 $\log 1/P(\{x\})$。

其次,聚类允许有损压缩,因而有助于通信。生物学家能这样为一个朋友指路:"走到右边的第 3 棵树,然后向右转"(而不是"经过有红色浆果的大的绿东西,然后经过有荆棘的大的绿东西,然后……")。简短的类名称"树"是有用的,因为它足以用于辨认一个物体。同样地,在有损图像压缩中,目标是用尽可能少的传输比特来合理地再现图像;完成这一任务的一种方法是把图像分成 N 个小片,并对每个小片在 K 个图像模板所组成的符号表中寻找一个最接近的匹配;然后通过发送一列匹配模板的标记 $k_1, k_2, \cdots, k_N$,发送与图像最接近的匹配。构建一个好的图像模板库的任务等价于选择一组聚类中心。这种类型的聚类有时称作"矢量量化"。

我们对矢量量化器进行正式描述。分配规则 $\boldsymbol{x} \rightarrow k(\boldsymbol{x})$ 把数据点 $\boldsymbol{x}$ 分配给 K 个代码名之一,重构规则 $k \rightarrow \boldsymbol{m}^{(k)}$ 的目标是选择函数 $k(\boldsymbol{x})$ 和 $\boldsymbol{m}^{(k)}$ 以使期望失真最小化,其定义可以是

$$D = \sum_{\boldsymbol{x}} P(\boldsymbol{x}) \frac{1}{2} [\boldsymbol{m}^{(k(\boldsymbol{x}))} - \boldsymbol{x}]^2 \tag{20.1}$$

[理想的目标函数应该是对心理感知的图像失真进行极小化。因为很难量化人所感知的失真,矢量量化和有损压缩不像数据建模和无损压缩那样是能定义清楚的问题。] 在矢量量化中,不必认为模板 $\{\boldsymbol{m}^{(k)}\}$ 有任何自然意义,它们只是完成任务的工具。注意将矢量量化的分配规则(编码器)的相似性传递到对一个纠错码进行译码

时的译码问题。

利用聚类模型的第三个原因是聚类模型的失败可能会突显一些值得特别关注的有趣目标。如果已经训练一个矢量量化器来完成海洋表面卫星图像的压缩,则那些不能被矢量量化器很好地压缩的图像块中可能包含有船! 如果生物学家发现一个绿色物体在跑动(或在游动),这种和他的聚类模型(说明绿色物体不能移动)的不吻合也会提醒他特别注意。一个人不可能用一生的精力来关注许多事情;聚类模型有助于一个人从所面对的大量事物中筛选出那些值得关注的事物。

使用聚类算法的第四个原因是它可以是神经系统学习过程的模型。现在讨论的聚类算法,如 K-均值算法,是竞争学习算法的一个例子。通过使 K 个类相互竞争以赢得数据点的归属权来运行算法。

20.1 K-均值聚类

K-均值算法将 I 维空间中的 N 个数据点归入 K 个类中。每个类用一个称为其均值的向量 $\boldsymbol{m}^{(k)}$ 参数化。①

数据点用 $\boldsymbol{x}^{(n)}$ 表示,其中上标 n 的范围是从 1 到数据点数 N(如图 20.1 所示)。每个向量 $\boldsymbol{x}$ 有 I 个分量 x_i。假设 $\boldsymbol{x}$ 所在空间是实空间,并且存在定义两点之间距离的一种度量,例如

$$d(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{2}\sum_i (x_i - y_i)^2 \tag{20.2}$$

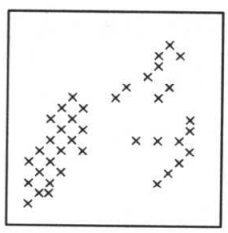


图 20.1 $N=40$ 个数据点

为了启动 K-均值算法(算法 20.2),要以某种方式初始化 K 个均值 $\{\boldsymbol{m}^{(k)}\}$,例如赋上随机值。K-均值算法是一个迭代式两步算法。在赋值步骤,每个数据点 n 被赋值给最近的均值。在更新步骤,调整均值以便更好地匹配它所负责的数据点的样本均值。

① **关于这个名称** 据笔者所知,K-均值中的"K"只表示选择聚类的数目。如果 Newton 也采用同样的命名方法,也许在学校里所学到的会是关于"变量 x 的微积分"。这是一个愚蠢的名称,但我们还是坚持使用它。

初始化 将 K 个均值 $\{\boldsymbol{m}^{(k)}\}$ 设置为随机值。

赋值步骤 每个数据点 n 被赋予最接近的均值。用 $\hat{k}^{(n)}$ 表示对数据点 $\boldsymbol{x}^{(n)}$ 所属类 $k^{(n)}$ 的猜测。

$$\hat{k}^{(n)} = \underset{\hat{k}}{\operatorname{argmin}}\{d(\boldsymbol{m}^{(k)}, \boldsymbol{x}^{(n)})\} \tag{20.3}$$

另一种将数据点分配到聚类的等价表示由“责任”给出,它是一个指标变量 $r_k^{(n)}$。在赋值步骤中,如果数据点 $\boldsymbol{x}^{(n)}$ 最接近均值 k,则设 $r_k^{(n)}$ 为 1,否则为 0。

$$r_k^{(n)} = \begin{cases} 1, & \hat{k}^{(n)} = k \\ 0, & \hat{k}^{(n)} \neq k \end{cases} \tag{20.4}$$

出现平局怎么办? 我们不希望一个数据点到两个均值有相同的距离,但是如果发生了平局,$\hat{k}^{(n)}$ 总是设为最小的获胜 $\{k\}$。

更新步骤 调整模型参数即均值,以便更好地匹配它所负责数据点的样本均值。

$$\boldsymbol{m}^{(k)} = \frac{\sum_n r_k^{(n)} \boldsymbol{x}^{(n)}}{R^{(k)}} \tag{20.5}$$

其中 $R^{(k)}$ 是均值 k 的总责任

$$R^{(k)} = \sum_n r_k^{(n)} \tag{20.6}$$

没有责任的均值是怎么回事? 如果 $R^{(k)} = 0$,则维持均值 $\boldsymbol{m}^{(k)}$ 在原处。

重复赋值和更新步骤 直到赋值不再变化。

算法 20.2 K - 均值聚类算法

使用图 20.3 所示的简化二维数据集,对 K - 均值算法进行了演示,其中使用了两个均值。将点赋值给两个类通过两种点形式表示,两个均值用圆圈表示。算法经过 3 次迭代后就收敛了,此时数据点的赋值并未改变,所以均值在更新过程中不会移动。K - 均值算法总是收敛到一个不动点。

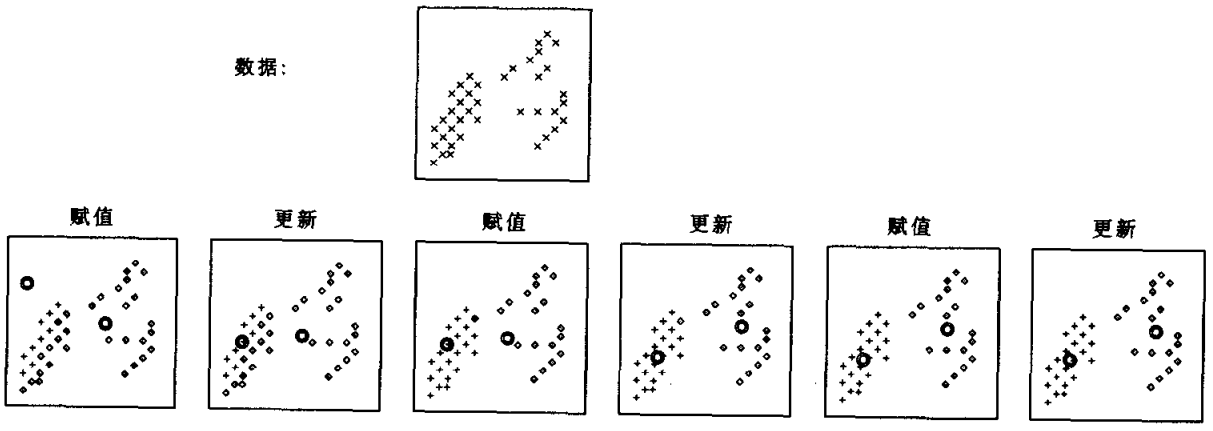


图 20.3 应用于 40 个点的数据集的 K - 均值算法。$K = 2$ 个均值在 3 次迭代后演化为稳定位置。

习题 20.1[✓] 试证 K-均值算法总是收敛的。

[提示:找一个类似的物理系统和一个相关的 Lyapunov 函数。]

[一个 Lyapunov 函数是算法状态的函数,当状态发生变化时,函数值下降并有下界。如果一个系统具有 Lyapunov 函数,则其动态收敛。]

如图 20.4 所示为具有较多均值(共 4 个)的 K-均值算法。算法的结果依赖于初始条件。在第一种情况下,经过 5 次迭代找到了稳定状态,数据点相当均匀地分布在 4 个类中。在第二种情况,经过 6 次迭代,有一半的数据点落在一个类中,剩下的点在其他 3 个类中。

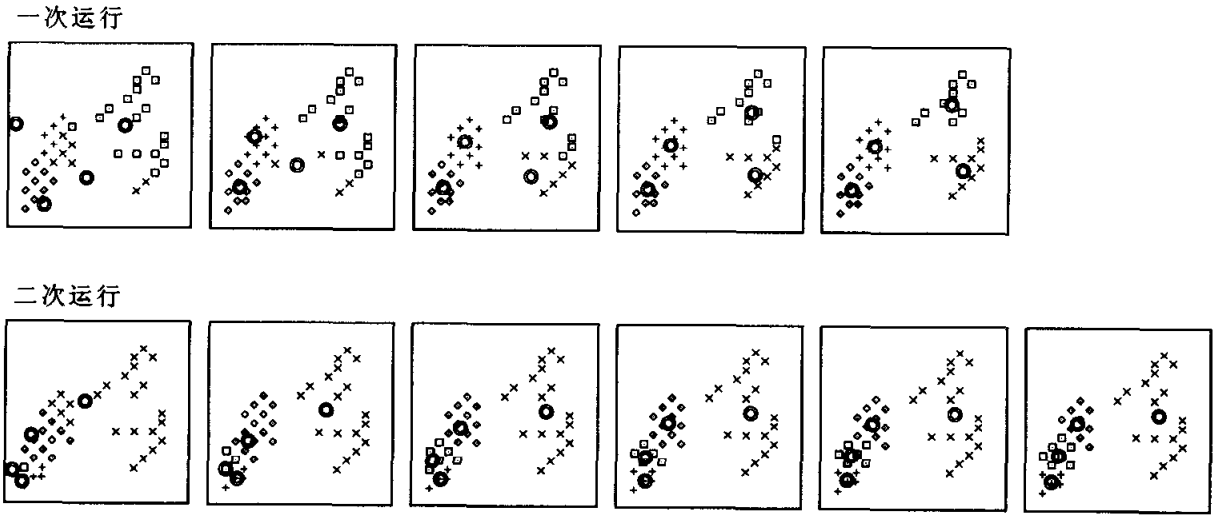


图 20.4 应用于 40 个点的数据集的 K-均值算法。两次独立运行,都是 $K=4$ 个均值,但得到不同的解。每一帧所示都是连续的赋值步骤。

关于此算法的一些问题

K-均值算法有几个特别的特征。为什么在更新步骤中要把“均值”设置为已赋值点的平均值?距离 d 从何而来?如果采用 $\boldsymbol{x}$ 和 $\boldsymbol{m}$ 的一个不同的距离度量会怎样?如何才能选择到“最佳”距离?[在向量量化中,距离函数是问题定义的一部分;但是,笔者假设我们所感兴趣的是数据建模而非向量量化。]如何选择 K?如果对于已知 K 找到了多种可替换的聚类方式,如何在它们中间做出选择?

K-均值可能被视做失败的情况

当寻找算法使用效果欠佳的情况时(与普通人所谓的“聚类”相比),就会出现更进一步的问题。如图 20.5(a)所示是由两个高斯分布混合所产生的 75 个数据点集合。右边的高斯具有较小权重(只有 1/5 的数据点),是较窄的类。如图 20.5(b)所示为利用具有 $K=2$ 个均值的 K-均值聚类的结果。4 个属于大类的数据点被分配到小类,而两个均值都移到了聚类真实中心的左边。K-均值算法只考虑均值和数

据点之间的距离;每个类的权重或宽度都没有得到表示。因此,实际上属于宽类的数据点被错误地划分到了窄类。

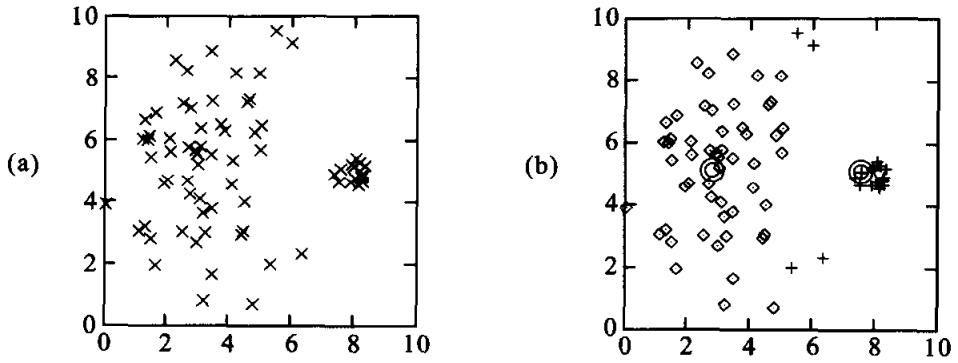


图 20.5 两个不相似聚类情形的 K - 均值算法。(a)“小”与“大”数据。(b)一组稳定的赋值和均值。注意有 4 个属于较宽聚类的点被错划到较窄聚类中了。(赋值给右边聚类的点用加号表示。)

图 20.6 所示为 K - 均值表现欠佳的另一个例子。数据点显然属于两个延长类。但是 K - 均值算法的唯一稳定状态如图 20.6(b)所示:两个类被分成两半! 这两个例子表明 K - 均值算法中的距离 d 存在一些问题。K - 均值算法没有办法表示类的大小或形状。

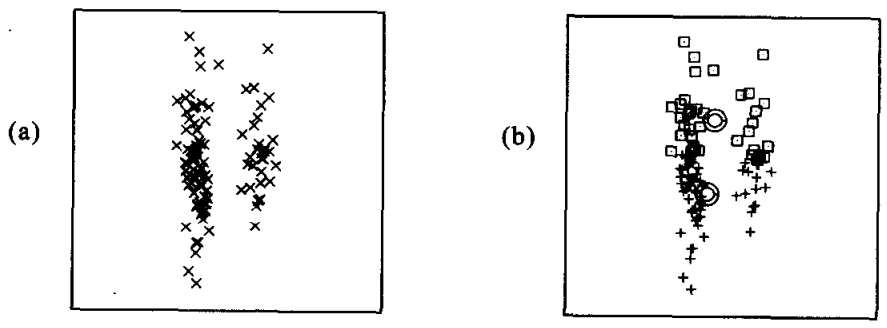


图 20.6 两个延长的类以及用 K - 均值算法得到的稳定解。

K - 均值算法的最后一个不足之处在于,它是一个“硬”的而不是“软”的算法:点只被分配到一个类中,分到一个类的所有点对于这个类而言都是等价的。但是如前所述,在决定点似应被分配到的所有类的位置时,位于两个或多个类的边界上的点应只起到部分作用。在 K - 均值算法中,每个边界线上的点都被拉入一个类中,与该类中的其他点享有同样的发言权,在其他类中则没有发言权。

20.2 软 K - 均值聚类

K - 均值的不足之处启发人们提出了“软 K - 均值算法”——算法 20.7。算法有一个参数 β,可以称之为*倔强系数*。

注意软 K - 均值算法和硬 K - 均值算法 20.2 的相似性。更新步骤是相同的;唯一差别在于责任 $r_k^{(n)}$ 可以取 0 和 1 之间的值。然而 K - 均值算法中的赋值 $\hat{k}^{(n)}$ 是在距离中找“最小”,分配责任的规则是“软最小”(算法 20.7)。

赋值步骤 对于每个均值,为每个数据点 $\boldsymbol{x}^{(n)}$ 都赋予一个软性“赋值度”。把 $\boldsymbol{x}^{(n)}$ 赋值到类 k 的度叫做*责任* $r_k^{(n)}$(类 k 对于数据点 n 的责任)。

$$r_k^{(n)} = \frac{\exp(-\beta d(\boldsymbol{m}^{(k)}, \boldsymbol{x}^{(n)}))}{\sum_{k'} \exp(-\beta d(\boldsymbol{m}^{(k')}, \boldsymbol{x}^{(n)}))} \tag{20.7}$$

第 n 个点的 K 个责任之和为 1。

更新步骤 调整模型参数均值,使它能够与所负责的数据点的样本均值相匹配。

$$\boldsymbol{m}^{(k)} = \frac{\sum_n r_k^{(n)} \boldsymbol{x}^{(n)}}{R^{(k)}} \tag{20.8}$$

其中 $R^{(k)}$ 是均值 k 的总责任。

$$R^{(k)} = \sum_n r_k^{(n)} \tag{20.9}$$

算法 20.7 软 K - 均值算法(版本 1)

▷ **习题 20.2**[ㄗ] 证明当倔强系数 β 变成 ∞ 时,除了未赋值数据点的均值行为方式之外,软 K - 均值算法和原先的硬 K - 均值算法相同。描述这些不再是静止不动的均值如何变化。

从量纲上讲,倔强系数 β 是长度平方的倒数,所以可以有一个长度尺度 $\sigma \equiv 1/\sqrt{\beta}$ 与之关联。软 K - 均值算法如图 20.8 所示。长度尺度如围绕 4 个均值的圆的半径所示。每个子图显示用不同长度尺度 σ 值所达到的最后不动点。

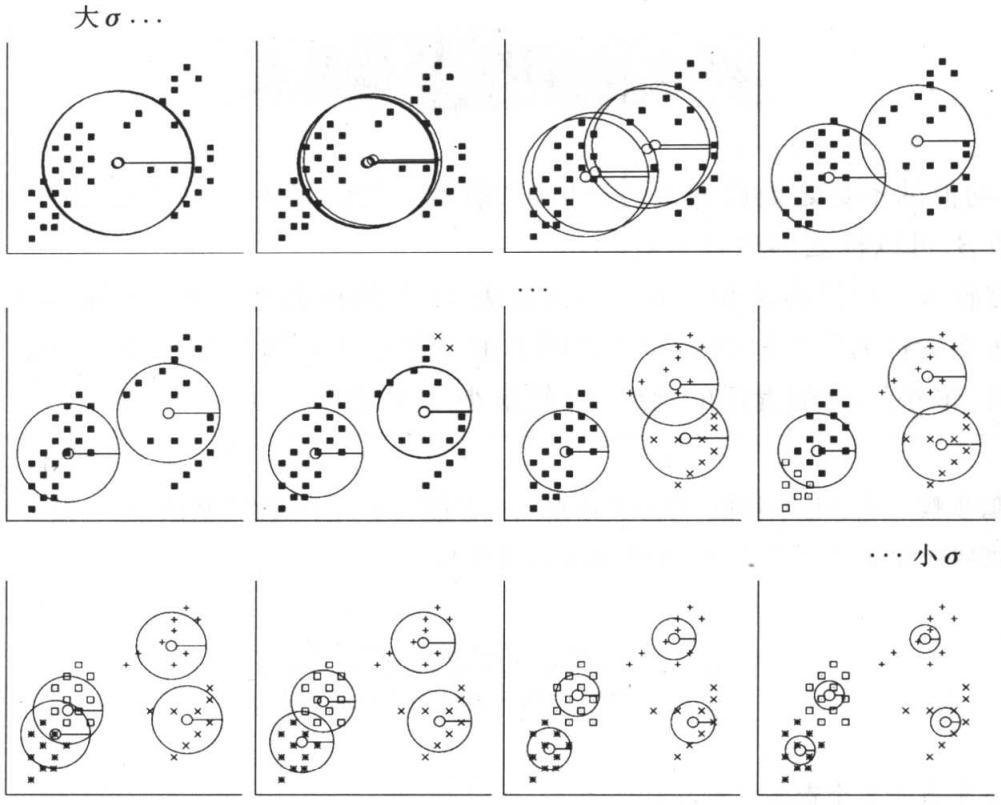


图 20.8 应用于 40 个数据点集合的软 K - 均值算法(版本 1)。$K=4$。隐含的长度尺度参数 $\sigma=1/\beta^{1/2}$ 的数值从大变到小。经过算法的几十次迭代运行,每幅图显示了所有 4 个均值的状态,隐含的长度尺度用 4 个圆的半径表示。在最大的长度尺度上,所有 4 个均值都准确地收敛到数据均值。然后 4 个均值两两分到两组中。在较短的长度尺度上,每一对自己又分成了两个子组。

20.3 结 论

到目前为止,通过引入一个额外的复杂性控制参数 β,也许已经解决了原 K - 均值算法中存在的一些问题。但是应如何设置 β? 如何处理延长类的问题? 如何处理权重和宽度不等的类? 增加一个倔强系数参数 β 并不能解决所有这些问题。

我们将在后面章节中重新研究这些问题,那时将以混合密度建模的观点来看待聚类问题。

进一步的阅读

关于聚类的向量量化方法参见参考文献(Luttrell,1989 年;Luttrell,1990 年)。

20.4 习 题

▷**习题 20.3**[研] 假设数据点$\{\boldsymbol{x}\}$来自单一可分的二维高斯分布，其均值为零，方差为$(\mathrm{var}(x_1),\mathrm{var}(x_2))=(\sigma_1^2,\sigma_2^2),\sigma_1^2>\sigma_2^2,$。探讨软 K - 均值算法（版本 1）的特性。设 $K=2$，假设 N 很大，当 β 发生变化时，研究算法的不动点。

［提示：假设 $\boldsymbol{m}^{(1)}=(m,0)$ 和 $\boldsymbol{m}^{(2)}=(-m,0)$。］

▷**习题 20.4**[研] 考虑应用于大量一维数据的软 K - 均值算法，数据来自两个权重相等的高斯分布，其均值为 $\mu=\pm 1$，标准差为 σ_P，如 $\sigma_P=1$。证明用 $K=2$ 的硬 K - 均值算法所得到的解是比实际均值分得更远的两个均值。讨论用其他 β 值会发生什么情况，找出用软算法能把两个均值放在正确位置的 β 值。



20.5 习题解答

习题 20.1 的解答 可以在每个点 $\boldsymbol{x}^{(n)}$ 和与它相关的均值之间连接一个弹簧，使得 K - 均值算法的状态具有“能量”。弹簧的能量正比于它的长度的平方，即 $\beta d(\boldsymbol{x}^{(n)},\boldsymbol{m}^{(k)})$，其中 β 是弹簧的倔强系数。所有弹簧的总能量就是算法的 Lyapunov 函数。因为（a）赋值步骤只能减少能量——如果弹簧的长度减小，点就改变它的忠诚度；（b）更新步骤只能减小能量——把 $\boldsymbol{m}^{(k)}$ 移到均值是使弹簧能量极小化的方式；（c）能量存在下界——这是 Lyapunov 函数的第二个条件。由于算法具有 Lyapunov 函数，所以它收敛。

习题 20.3 的解答 如果均值被初始化为 $\boldsymbol{m}^{(1)}=(m,0)$ 和 $\boldsymbol{m}^{(1)}=(-m,0)$，对于位于 x_1,x_2 的点，点的赋值步骤是

$$r_1(\boldsymbol{x})=\frac{\exp(-\beta(x_1-m)^2/2)}{\exp(-\beta(x_1-m)^2/2)+\exp(-\beta(x_1+m)^2/2)} \tag{20.10}$$

$$=\frac{1}{1+\exp(-2\beta m x_1)} \tag{20.11}$$

更新的 m 是

$$m' = \frac{\int dx_1 P(x_1) x_1 r_1(x)}{\int dx_1 P(x_1) r_1(x)} \tag{20.12}$$

$$= 2\int dx_1 P(x_1) x_1 \frac{1}{1 + \exp(-2\beta m x_1)} \tag{20.13}$$

现在 $m=0$ 是一个不动点,问题是它是否稳定? 对于很小的 m(即 $\beta\sigma_1 m << 1$),可以进行泰勒级数展开

$$\frac{1}{1 + \exp(-2\beta m x_1)} \simeq \frac{1}{2}(1 + \beta m x_1) + \cdots \tag{20.14}$$

所以

$$m' \simeq \int dx_1 P(x_1) x_1 (1 + \beta m x_1) \tag{20.15}$$

$$= \sigma_1^2 \beta m \tag{20.16}$$

对于小的 m,取决于 $\sigma_1^2\beta$ 是大于还是小于 1,m 在这样的映射下以指数级增长或衰减。如果

$$\sigma_1^2 \leqslant 1/\beta \tag{20.17}$$

则不动点 $m=0$ 是稳定的。否则不稳定。如图 20.9 所示。[巧合的是,通过推导可证明这个结果是普遍性的:对于方差为 σ_1^2 的任何真实概率分布 $P(x_1)$ 都是适用的,而不仅限于高斯分布。]

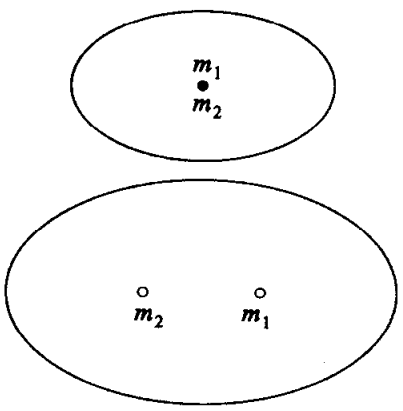


图 20.9 当最大数据方差 σ_1 从小于 $1/\beta^{1/2}$ 增加到大于 $1\beta^{1/2}$ 以上时出现分岔的示意图。数据方差用椭圆表示。

如果 $\sigma_1^2 > 1/\beta$,则出现一个分岔,且在不稳定的不动点 $m=0$ 附近会有两个稳定的不动点。为了描绘这个分岔,图 20.10 显示了对于具有标准差 σ_1 的一维数据点,在不同 σ_1 值下运行 $\beta=1$ 软 K-均值算法所得的结果。图 20.11 显示了从另一角度来看的二分岔,其中数据的标准差 σ_1 是固定的,算法的长度尺度 $\sigma = 1/\beta^{1/2}$ 在水平轴方向变化。

基于对级数的继续展开,现在给出在分岔点以后对拟合参数 $\pm m$ 如何行动来建

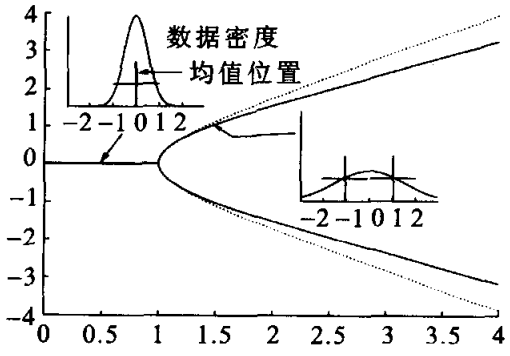


图 20.10 对于恒定的 β 值，稳定的均值位置，以 σ_1 的函数表示，得出的数值结果（粗线）和近似结果(20.22)（细线）。

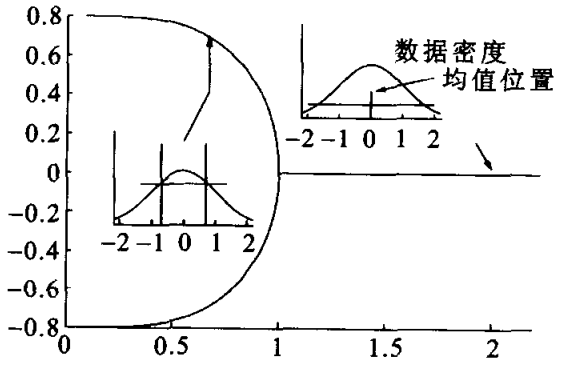


图 20.11 对于恒定的 σ_1，稳定均值的位置，以 $1/\beta^{1/2}$ 的函数表示。

立模型的简单理论。这个级数的连续性是相当可疑的，因为在分岔点以后，级数不一定会收敛，但是无论如何，理论的符合程度还是很高的。

将解析方法在级数展开中多应用一项：

$$\frac{1}{1+\exp(-2\beta m x_1)} \simeq \frac{1}{2}\left(1+\beta m x_1 - \frac{1}{3}(\beta m x_1)^3\right) + \cdots \tag{20.18}$$

然后就可以解出分岔形状的主导阶，它依赖于以下分布的四阶矩。

$$m' \simeq \int \mathrm{d}x_1 P(x_1) x_1 \left(1+\beta m x_1 - \frac{1}{3}(\beta m x_1)^3\right) \tag{20.19}$$

$$= \sigma_1^2 \beta m - \frac{1}{3}(\beta m)^3 3\sigma_1^4 \tag{20.20}$$

[在式(20.20)中，利用 $P(x_1)$ 是高斯分布的事实来求四阶矩。] 这个映射在 m 处有不动点，从而

$$\sigma_1^2 \beta (1 - (\beta m)^2 \sigma_1^2) = 1 \tag{20.21}$$

即

$$m = \pm \beta^{-1/2} \frac{(\sigma_1^2 \beta - 1)^{1/2}}{\sigma_1^2 \beta} \tag{20.22}$$

如图 20.10 中的细线所示为理论上的近似。图 20.10 表示对于固定 β,分岔以 σ_1 的函数表示的形式;图 20.11 是对于固定 σ_1,分岔以 $1/\beta^{1/2}$ 的函数表示的形式。

▷ **习题 20.5**[⌀] 为什么图 20.11 中的二分叉在 $1/\beta^{1/2}\to 0$ 时趋于值 ~ ±0.8? 给出渐近线的解析表达式。

习题 20.5 的解答 渐近线是修正高斯分布的均值

$$\frac{\int_0^\infty \text{Normal}(x,1)x\ \mathrm{d}x}{1/2} = \sqrt{2/\pi} \simeq 0.798 \tag{20.23}$$

第 21 章

基于完全枚举的精确推理

通过讨论蛮力推理方法(即对所有假设完全枚举,并估计它们的概率),来开启概率处理方法工具箱。这是一种准确的方法,实现它的困难将激发后面章节中的更灵巧的准确方法和近似方法的动力。

21.1 防盗警报

贝叶斯概率论有时称为“增强的常识”。在考虑下面的问题时,请让你的常识告诉你答案;然后来看贝叶斯方法如何证实你日常生活中的直觉。

例 21.1 弗雷德住在洛杉矶,每天往返乘车 60 英里去上班。在上班时,他接到邻居打来的电话,说他家的防盗警报响了。那么他家今天来了盗贼的概率是多少?就在他驱车回家的路上,弗雷德听到广播里说当日在他家附近发生了一次小地震。“哦”,他感到一阵放松,“可能是地震触发了警报”。如图 21.1 所示。现在他房间里有一个盗贼的概率又是多少?(来自 Pearl,1988 年)。

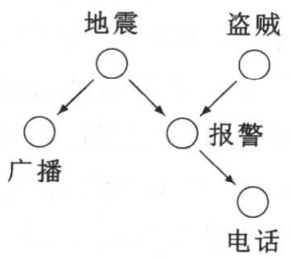


图 21.1 盗贼报警问题的置信网络

引入变量 b(一个盗贼今天出现在弗雷德家里),a(报警器响了),p(弗雷德接到邻居打来的报警电话),e(今天弗雷德家附近发生了小地震)和 r(弗雷德听到了广播里关于地震的报道)。所有这些变量都发生的概率可以做以下因式分解

$$P(b,e,a,p,r) = P(b)P(e)P(a|b,e)P(p|a)P(r|e) \tag{21.1}$$

这些概率的可能值如下。

1. 盗贼概率

$$P(b=1) = \beta, P(b=0) = 1-\beta \tag{21.2}$$

比如,$\beta = 0.001$ 表示 3 年出现一次盗窃案的平均案发率。

2. 地震概率

$$P(e=1)=\varepsilon, P(e=0)=1-\varepsilon \tag{21.3}$$

比如 $\varepsilon=0.001$；我们声称地震和盗窃是独立发生的，即 b 和 e 的先验概率是 $P(b,e)=P(b)P(e)$，这看起来是合理的，除非考虑有盗贼趁机在地震后立即行窃。

3. 报警器鸣叫的概率：假设如果以下三件事情中的任何一件发生，都会引起报警器鸣叫：(a) 盗贼进入房间，触发了报警器（假设报警的可靠性是 $\alpha_b=0.99$，即99%的盗贼都会触发报警器）；(b) 发生地震并且触发了报警器（也许只有 $\alpha_e=1\%$ 的报警是由地震触发的?）；或者(c) 其他一些事件引起误报警；假设误报警率 f 是0.001，也就是弗雷德每3年才有一次是非地震情况下的报警。[a 对 b 和 e 的这种依存关系类型叫做“有噪-或”。] 于是，已知 b 和 e 时 a 的概率为

$$P(a=0|b=0,e=0)=(1-f), \quad P(a=1|b=0,e=0)=f$$
$$P(a=0|b=1,e=0)=(1-f)(1-\alpha_b), \quad P(a=1|b=1,e=0)=1-(1-f)(1-\alpha_b)$$
$$P(a=0|b=0,e=1)=(1-f)(1-\alpha_e), \quad P(a=1|b=0,e=1)=1-(1-f)(1-\alpha_e)$$
$$P(a=0|b=1,e=1)=(1-f)(1-\alpha_b)(1-\alpha_e),$$
$$P(a=1|b=1,e=1)=1-(1-f)(1-\alpha_b)(1-\alpha_e)$$

或者以数字表示为

$$P(a=0|b=0,e=0)=0.999, \quad P(a=1|b=0,e=0)=0.001$$
$$P(a=0|b=1,e=0)=0.009\,99, \quad P(a=1|b=1,e=0)=0.990\,01$$
$$P(a=0|b=0,e=1)=0.989\,01, \quad P(a=1|b=0,e=1)=0.010\,99$$
$$P(a=0|b=1,e=1)=0.009\,890\,1, \quad P(a=1|b=1,e=1)=0.990\,109\,9$$

假设邻居在没有报警的情况下是不会打电话的：[$P(p=1|a=0)=0$]；广播也是值得信赖的报告者：[$P(r=1|e=0)=0$]；要回答以上问题，无需指定概率 $P(p=1|a=1)$ 或 $P(r=1|e=1)$，因为结果 $p=1$ 和 $r=1$ 分别让人确知 $a=1$ 和 $e=1$。

已知一些现有的信息，可以通过计算所有假设的后验概率来回答有关盗贼的两个问题。首先提醒自己，在观测到 p 或 r 之前，出现盗贼的概率是 $P(b=1)=\beta=0.001$，地震发生的概率是 $P(e=1)=\varepsilon=0.001$，且这两个事件是独立的。

首先，当 $p=1$ 时，我们知道报警器在鸣叫 $a=1$。b 和 e 的后验概率变为

$$P(b,e|a=1)=\frac{P(a=1|b,e)P(b)P(e)}{P(a=1)} \tag{21.4}$$

分子的4个可能值是

$$P(a=1|b=0,e=0)\times P(b=0)\times P(e=0)=0.001\times 0.999\times 0.999=0.000\,998$$
$$P(a=1|b=1,e=0)\times P(b=1)\times P(e=0)=0.990\,01\times 0.001\times 0.999=0.000\,989$$
$$P(a=1|b=0,e=1)\times P(b=0)\times P(e=1)=0.010\,99\times 0.999\times 0.001=0.000\,010\,979$$
$$P(a=1|b=1,e=1)\times P(b=1)\times P(e=1)=0.990\,109\,9\times 0.001\times 0.001=9.9\times 10^{-7}$$

归一化常数是4个数之和，$P(a=1)=0.002$，后验概率是

$$\begin{aligned} P(b=0,e=0|a=1)&=0.499\,3\\ P(b=1,e=0|a=1)&=0.494\,7\\ P(b=0,e=1|a=1)&=0.005\,5\\ P(b=1,e=1|a=1)&=0.000\,5 \end{aligned} \tag{21.5}$$

为了回答“那里出现盗贼的概率是多少?”这一问题,对地震变量 e 进行边缘化。

$$\begin{aligned} P(b=0|a=1) &= P(b=0,e=0|a=1)+P(b=0,e=1|a=1)=0.505 \\ P(b=1|a=1) &= P(b=1,e=0|a=1)+P(b=1,e=1|a=1)=0.495 \end{aligned} \tag{21.6}$$

所以有接近 50% 的几率会出现盗贼。很重要的一点是注意变量 b 和 e 是先验独立的,但现在是非独立的。后验分布(21.5)不是 b 和 e 的可分函数。只要研究得知 $e=1$ 后所产生的效应就可以很清楚地说明这个事实。

当得知 $e=1$ 后,b 的后验概率由下式给出

$$P(b|e=1,a=1)=P(b,e=1|a=1)/P(e=1|a=1)$$

即把式(21.5)中的下面两行除以它们的和 $P(e=1|a=1)=0.006\,0$。b 的后验概率是

$$\begin{aligned} P(b=0|e=1,a=1) &= 0.92 \\ P(b=1|e=1,a=1) &= 0.08 \end{aligned} \tag{21.7}$$

因此根据现在的信息,盗贼身处弗雷德家中的概率为 8%。这和日常的感觉是一致的,$b=1$(引起报警的可能原因)的概率在弗雷德知道发生地震之后减小了,就是因为出现了对于报警的另一种解释。

辩解

一些数据(这里是 $a=1$)的一个可能起因($b=1$)的可能性变小时,另一个起因($e=1$)的可能性变大,即使在先验上这两种原因是独立变量,这种现象叫做*辩解*。辩解是正确推理的重要特征,也是任何人工智能都应该具备的。

如果相信邻居和广播服务是不可靠的或反复无常的,以至于人们不能确定是否真的发生了报警或地震,计算会变得更复杂,但是辩解效应仍然存在;收到地震报道 r 的同时,更加可能真的发生报警,同时盗贼出现的可能性更小。

总之,可以通过枚举变量 (b,e) 的所有 4 个假设来解决盗贼推理问题,找到它们的后验概率和边缘化,得到所需的关于 b 的推理。

▷ **习题 21.2**[2] 在弗雷德接到盗窃报警的电话之后,但在他听到广播报道之前,根据他自己的判断,今天发生小地震的概率有多大?

21.2 对连续假设空间的精确推理

要考虑的许多假设空间可以自然地看做连续的。例如,第 3.1 节中未知的衰减长度 λ 存在于连续一维空间中;高斯过程的未知均值 μ 和标准差 σ 存在于连续二维空间中。但是,在许多实际的计算机实现中,有必要把这些连续空间离散化,使之变成可以枚举的(从原理上讲)——例如是参数值的一个网格。在图 3.2 中,通过计算在一系列细密的点上的似然值,画出了衰减长度的似然函数,以 λ 的函数表示。

一个双参数模型

现在以高斯分布作为具有二维假设空间的模型的例子。一维高斯分布以均值 μ 和标准差 σ 被参数化。

$$P(x|\mu,\sigma)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)\equiv\text{Normal}(x;\mu,\sigma^2) \tag{21.8}$$

如图 21.2 所示是对一维高斯分布均值和标准差的 100 个假设的枚举。这些假设均匀地分布在 10×10 的网格上，包含了 10 个 μ 值和 10 个 σ 值。每个假设用关于 x 的概率密度图形来表示。现在来看已知数据点 $x_n, n=1,2,\cdots,N$ 时对 μ 和 σ 的推理，假设数据点 x_n 独立地从这个概率密度中产生。

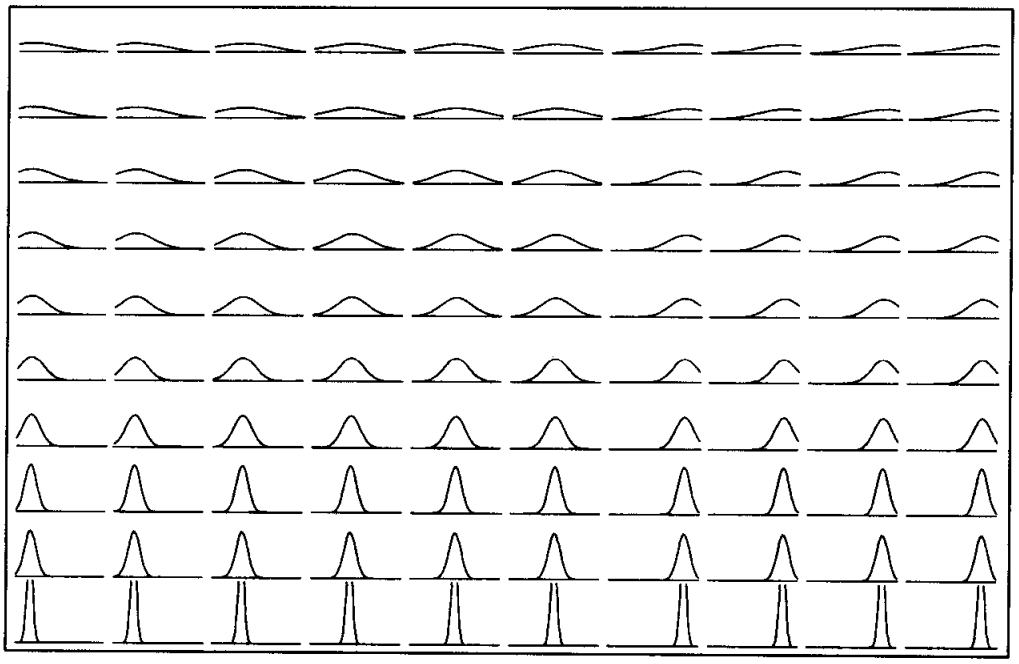


图 21.2 具有参数 μ（水平轴）和 σ（垂直轴）的整个高斯（离散化）假设空间的枚举。

设想我们得到了数据，比如如图 21.3 所示的 5 个数据点。通过计算 100 个子假设中的每一个假设的似然，即 $P(\{x_n\}_{n=1}^5|\mu,\sigma)$ 的值，就可以算出每一个子假设的后验概率。似然值在图 21.4 中通过线的粗细程度对似然值编码来表示。删除了似然小于最大似然 e^{-8} 倍的子假设。

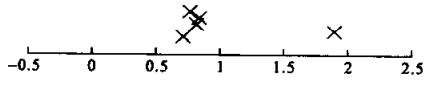


图 21.3 5 个数据点 $\{x_n\}_{n=1}^5$。横坐标为数据 x_n 的值，纵坐标没有意义。

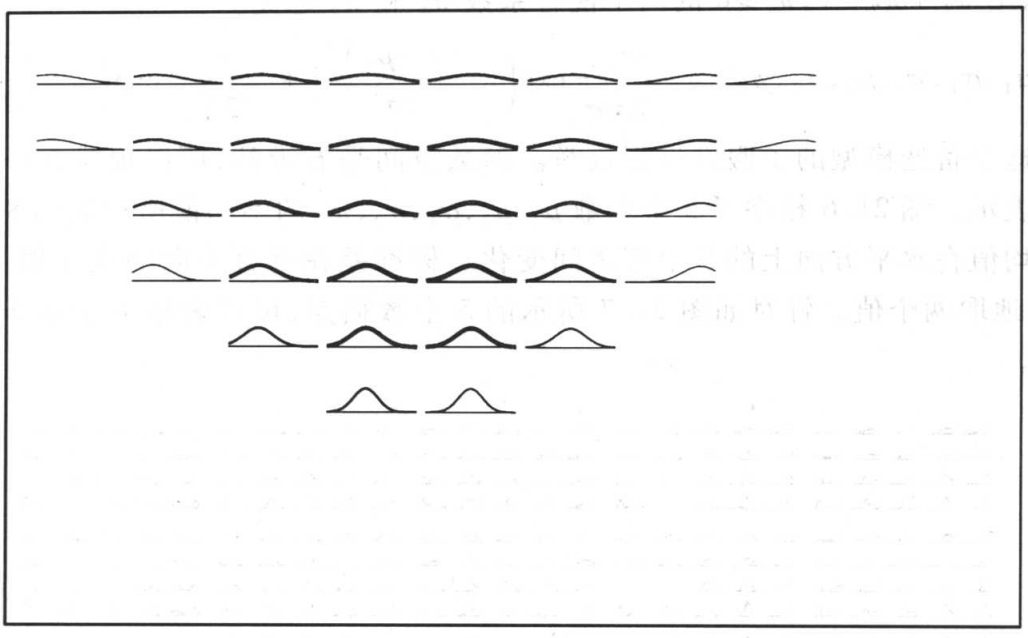


图 21.4 用线的粗细程度所表示的已知图 21.3 中数据的似然函数。没有显示似然值小于 e^{-8} 倍最大似然的子假设。

使用更细的网格,可以通过画出似然(以 μ 和 σ 的函数表示)的曲面图或等值图(如图 21.5 所示)来表示相同信息。

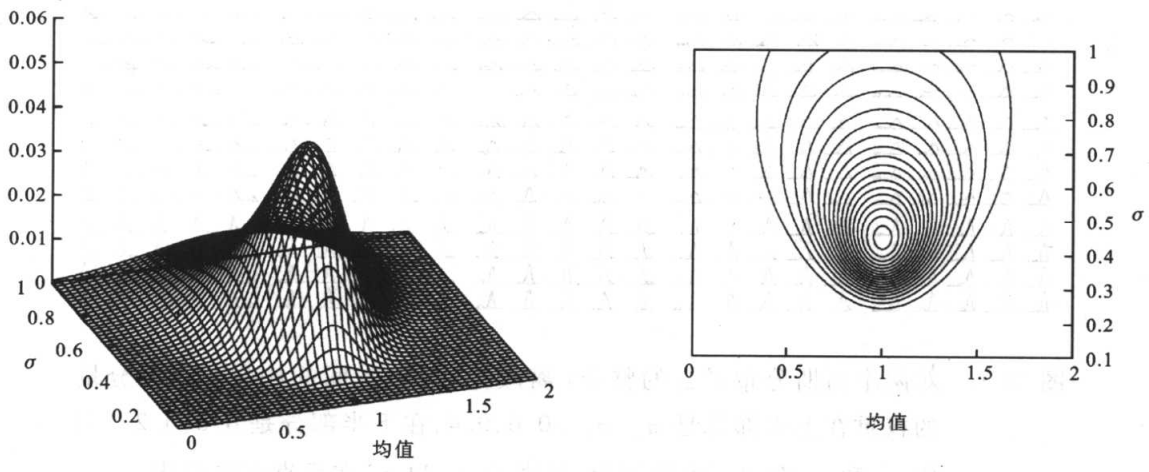


图 21.5 一个高斯分布的参数的似然函数。以 μ 和 σ 的函数表示的对数似然的曲面图和等值图。数据集中的 $N=5$ 个点有均值 $\bar{x}=1.0$ 和 $S^2 = \sum (x-\bar{x})^2 = 1.0$。

一个 5 - 参数混合模型

对数据(如图 21.3 所示)进行目测,你也许会同意这种说法:它们似乎并非来自单一高斯分布,而是来自两个高斯分布的混合,这两个高斯分布由两个均值、两个标

准差、满足 $\pi_1+\pi_2=1,\pi_i\geqslant 0$ 的两个混合系数 π_1 和 π_2 定义。

$$P(x\mid\mu_1,\sigma_1,\pi_1,\mu_2,\sigma_2,\pi_2)=\frac{\pi_1}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{(x-\mu_1)^2}{2\sigma_1^2}\right)+\frac{\pi_2}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{(x-\mu_2)^2}{2\sigma_2^2}\right)$$

现在对这个备选模型的子假设进行枚举。参数空间是五维的，所以很难在一个页面上进行表示。图21.6枚举了5个参数 $\mu_1,\mu_2,\sigma_1,\sigma_2,\pi_1$ 的不同值所对应的800个子假设。均值在水平方向上的5个值之间变化。标准差在垂直方向的4个值上变化。π_1 垂直地取两个值。针对如图21.7所示的5个数据点，可以表示关于这5个参数的推理。

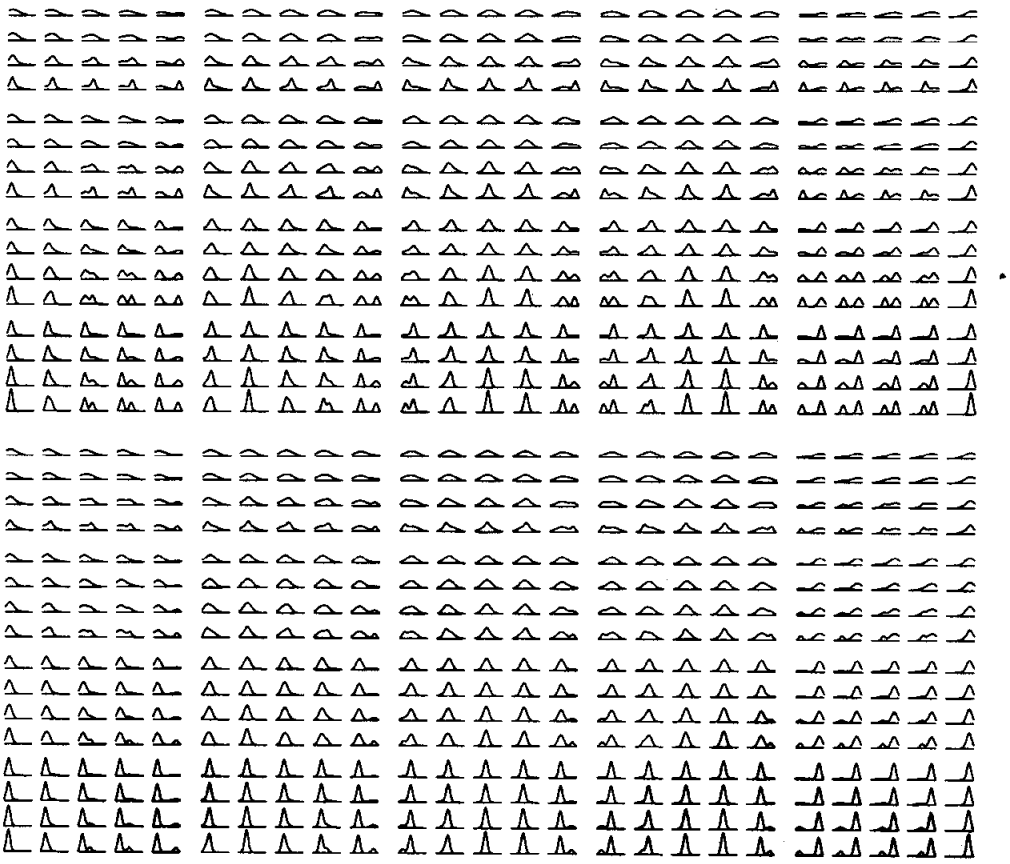


图21.6 对两个高斯分布混合的整个（离散化）假设空间的枚举。混合分量的权重在上半部分是 $\pi_1,\pi_2=0.6,0.4$，在下半部分是0.8,0.2。均值 μ_1 和 μ_2 在水平方向变化，标准差 σ_1 和 σ_2 在垂直方向变化。

如果希望将一个高斯模型和两个模型的混合进行比较，可以通过计算每个模型 $\mathscr{H}$，$P(\{x\}\mid\mathscr{H})$ 的边缘似然或显著性来找到模型的后验概率。显著性是通过对参数 $\boldsymbol{\theta}$ 进行积分得到的；可以通过对 $\boldsymbol{\theta}$ 的不同枚举值求和来实现数值积分。

$$P(\{x\}\mid\mathscr{H})=\sum_{\boldsymbol{\theta}}P(\boldsymbol{\theta})P(\{x\}\mid\boldsymbol{\theta},\mathscr{H}) \tag{21.9}$$

其中 $P(\boldsymbol{\theta})$ 是在参数值网格上的先验分布，笔者采用了均匀分布。

对于两个高斯分布的混合，这是一个五维积分；如果想精确地计算它，网格点必

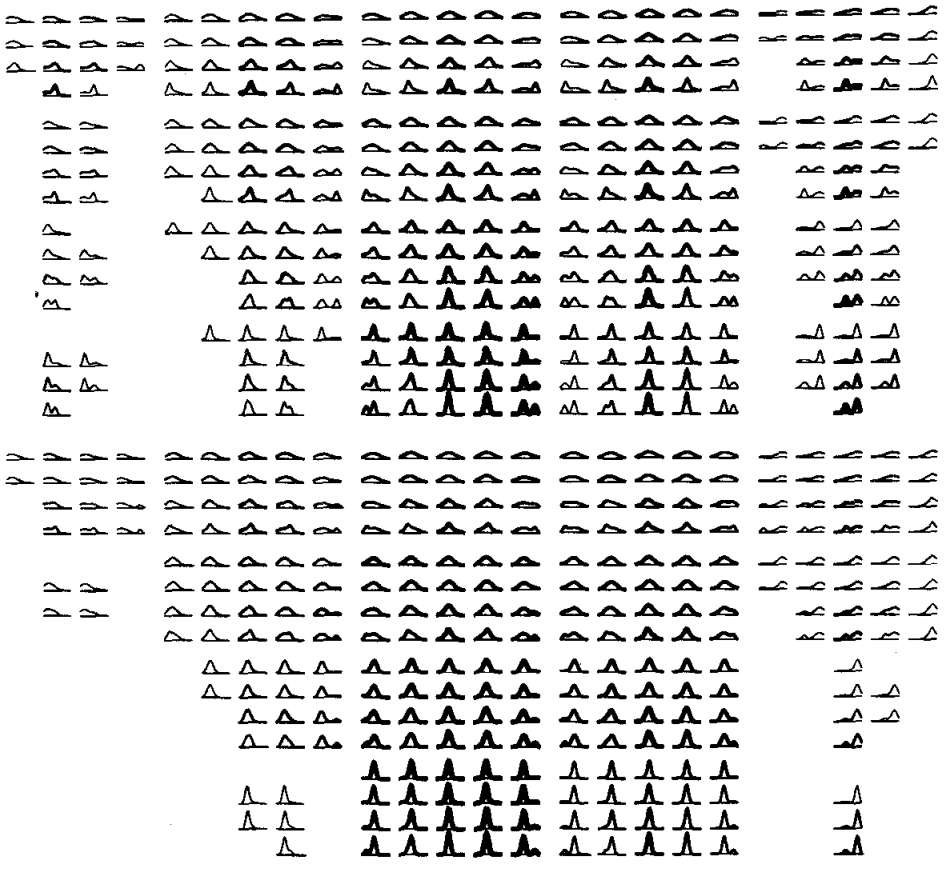
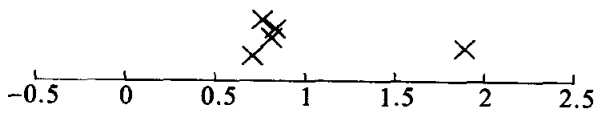


图 21.7 对两个高斯混合的推理。已知图 21.3 的数据，似然函数用线的粗细来表示。假设空间与图 21.6 所示的相同。没有显示似然值小于最大似然的 e^{-8} 倍的子假设，因此空白区域对应于数据被排除掉的假设。



须比图上显示的要细密许多才行。如果通过对数据进行观察，K 个参数中的每一个的不确定性都减少 10 倍，则蛮力积分需要网格上至少有 10^K 个点。计算量随模型大小而呈指数级增长，这是完全枚举法很少成为可行的计算策略的原因。



习题 21.3[小] 设想用 10 个高斯分布的混合来对二十维空间中的数据进行拟合。通过枚举参数值的网格来实现这个模型的推理。估计其计算代价。

第 22 章

最大似然与聚类

与其列举所有的假设——这可能是按指数级增长的——不如把注意力放在能够很好地拟合数据的一个好假设上，从而可以节省大量时间。这就是最大似然方法背后的依据，这种方法通过极大化似然 $P(\text{数据}|\boldsymbol{\theta},\mathscr{H})$ 来辨识参数向量 $\boldsymbol{\theta}$ 的设置。

对于有些模型，最大似然参数可以从数据中立即辨识出来；对于更复杂的模型，可能需要迭代算法才能得到最大似然参数。

对于任何模型，处理对数似然而非似然函数本身通常是最简单的，因为似然函数是许多数据点概率的乘积，一般都非常小。似然函数做乘法，对数似然做加法。

22.1 一个高斯的最大似然

作为第一个例子，重新讨论高斯分布。假设有数据点 $\{x_n\}_{n=1}^N$。对数似然函数为

$$\ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma) = -N\ln(\sqrt{2\pi}\sigma) - \sum_n (x_n-\mu)^2/(2\sigma^2) \tag{22.1}$$

似然函数可以用数据的两个函数来表示，即样本均值

$$\bar{x} \equiv \sum_{n=1}^N x_n/N \tag{22.2}$$

以及方差之和

$$S \equiv \sum_n (x_n-\bar{x})^2 \tag{22.3}$$

$$\ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma) = -N\ln(\sqrt{2\pi}\sigma) - [N(\mu-\bar{x})^2+S]/(2\sigma^2) \tag{22.4}$$

因为似然函数只通过 $\bar{x}$ 和 S 而依赖于数据，所以这两个量被称为*充分统计*。

例 22.1 求对数似然关于 μ 的导数，并证明：如果已知标准差为 σ，则对于任何 σ 值，高斯分布的最大似然均值 μ 等于样本均值 $\bar{x}$。

解

$$\frac{\partial}{\partial\mu}\ln P = -\frac{N(\mu-\bar{x})}{\sigma^2} \tag{22.5}$$

$$=0, \quad \mu=\bar{x} \qquad \square \tag{22.6}$$

如果把对数似然在最大值附近进行泰勒级数展开，可以定义最大似然参数的近似误差限：可用二次逼近来估计可以偏离最大似然参数的设置多远，似然才会下降一个标准因子，比如 $e^{1/2}$ 或 $e^{4/2}$。对于似然是参数的高斯函数这一特殊情况，二次逼近是精确的。如图 22.1 所示。

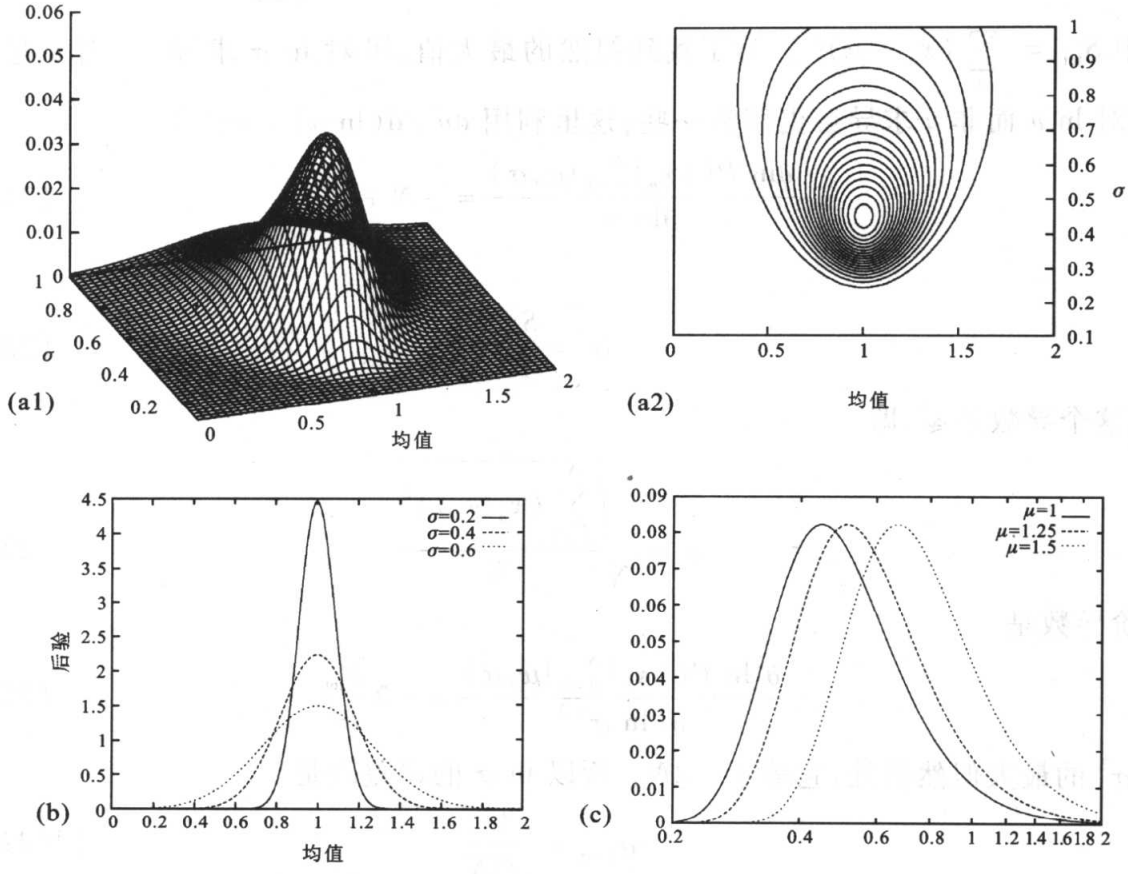


图 22.1 一个高斯分布参数的似然函数。(a1,a2) μ 和 σ 的函数的对数似然的曲面图和等势图。$N=5$ 个点的数据集具有均值 $\bar{x}=1.0$ 和 $S^2=\sum(x-\bar{x})^2=1.0$。(b) 各种 σ 值下 μ 的后验概率。(c) 各种固定 μ 值下 σ 的后验概率(表示为 $\ln\sigma$ 上的密度)。

例 22.2 已知数据和 σ,求对数似然关于 μ 的二阶导数以及关于 μ 的误差限。

解

$$\frac{\partial^2}{\partial\mu^2}\ln P=-\frac{N}{\sigma^2} \qquad \square(22.7)$$

把这个曲率和具有标准差 σ_μ 的高斯分布 $\exp(-\mu^2/(2\sigma_\mu^2))$ 的对数关于 μ 的曲率 $1/\sigma_\mu^2$ 进行比较,可以得出 μ 的误差限(从似然函数中导出)是

$$\sigma_\mu=\frac{\sigma}{\sqrt{N}} \qquad (22.8)$$

误差限具有这样的特性:在两个点 $\mu=\bar{x}\pm\sigma_\mu$ 上,似然比其最大值小一个因子 $e^{1/2}$。

例 22.3 针对数据 $\{x_n\}_{n=1}^N$,求已知均值为 μ 的高斯分布的最大似然标准差 σ。求对数似然关于 $\ln\sigma$ 的二阶导数以及关于 $\ln\sigma$ 的误差限。

解 似然与 σ 的依存关系是

$$\ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma) = -N\ln(\sqrt{2\pi}\sigma) - \frac{S_{\text{tot}}}{(2\sigma^2)} \tag{22.9}$$

其中 $S_{\text{tot}} = \sum_n (x_n - \mu)^2$。为了找到似然的最大值，可对 $\ln\sigma$ 求导。[当 u 是标量时，对 $\ln u$ 而非 u 求导会更简洁一些；这里利用 $\mathrm{d}u^n/\mathrm{d}(\ln u) = nu^n$。]

$$\frac{\partial \ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma)}{\partial \ln \sigma} = -N + \frac{S_{\text{tot}}}{\sigma^2} \tag{22.10}$$

当

$$\sigma^2 = \frac{S_{\text{tot}}}{N} \tag{22.11}$$

时，这个导数是零，即

$$\sigma = \sqrt{\frac{\sum_{n=1}^N (x_n - \mu)^2}{N}} \tag{22.12}$$

二阶导数是

$$\frac{\partial^2 \ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma)}{\partial(\ln\sigma)^2} = -2\frac{S_{\text{tot}}}{\sigma^2} \tag{22.13}$$

在 σ^2 的最大似然值处，它等于 $-2N$。所以 $\ln\sigma$ 的误差限是

$$\sigma_{\ln\sigma} = \frac{1}{\sqrt{2N}} \qquad \square \tag{22.14}$$

▷**习题 22.4**[1] 证明联合最大化似然的 μ 和 $\ln\sigma$ 值是 $\{\mu,\sigma\}_{\text{ML}} = \{\bar{x},\sigma_N = \sqrt{S/N}\}$，其中

$$\sigma_N \equiv \sqrt{\frac{\sum_{n=1}^N (x_n - \bar{x})^2}{N}} \tag{22.15}$$

22.2 混合高斯的最大似然

现在推导用混合高斯来拟合一维数据的算法。实际上，理解这个算法非常重要，因而作为普通读者的你应该自己推导这个算法。请完成下面的习题。



习题 22.5[2] 假设随机变量 x 的概率分布是两个高斯分布的混合

$$P(x \mid \mu_1,\mu_2,\sigma) = \left[\sum_{k=1}^2 p_k \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x-\mu_k)^2}{2\sigma^2}\right)\right] \tag{22.16}$$

其中两个高斯分别标记为 $k=1$ 和 $k=2$；标记为第 k 类的先验概率是 $\{p_1 = 1/2, p_2 = 1/2\}$；$\{\mu_k\}$ 是两个高斯分布的均值；二者的标准差均为 σ。为了简单起见，用 $\boldsymbol{\theta} \equiv$

$\{\{\mu_k\},\sigma\}$来表示这些参数。

数据集包含 N 个点$\{x_n\}_{n=1}^N$，假设是来自这个分布的独立样本。用 k_n 表示第 n 个点的未知类标记。

假设$\{\mu_k\}$和 σ 是已知的，证明第 n 个点的类标记 k_n 的后验概率可写成

$$P(k_n=1|x_n,\boldsymbol{\theta})=\frac{1}{1+\exp[-(w_1x_n+w_0)]}$$
$$P(k_n=2|x_n,\boldsymbol{\theta})=\frac{1}{1+\exp[+(w_1x_n+w_0)]} \tag{22.17}$$

并给出 w_1 和 w_0 的表达式。

现在假设均值$\{\mu_k\}$未知，希望从数据$\{x_n\}_{n=1}^N$中将它推断出来。（标准差 σ 是已知的。）在这个问题的余下部分中，将导出对如下似然进行极大化的$\{\mu_k\}$值的迭代算法。

$$P(\{x_n\}_{n=1}^N|\{\mu_k\},\sigma)=\prod_n P(x_n|\{\mu_k\},\sigma) \tag{22.18}$$

令 L 表示似然的自然对数。证明对数似然关于 μ_k 的导数是

$$\frac{\partial}{\partial\mu_k}L=\sum_n p_{k|n}\frac{(x_n-\mu_k)}{\sigma^2} \tag{22.19}$$

其中 $p_{k|n}\equiv P(k_n=k|x_n,\boldsymbol{\theta})$已经在式(22.17)中出现过。

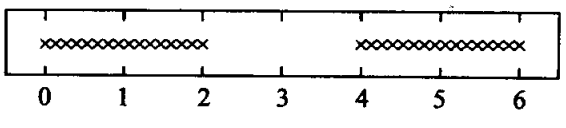
忽略$\frac{\partial}{\partial\mu_k}P(k_n=k|x_n,\boldsymbol{\theta})$中的项，证明二阶导数近似为

$$\frac{\partial^2}{\partial\mu_k^2}L=-\sum_n p_{k|n}\frac{1}{\sigma^2} \tag{22.20}$$

因此证明从初始状态 μ_1,μ_2，近似 Newton - Raphson 步骤把这些参数更新为 μ'_1,μ'_2，其中

$$\mu'_k=\frac{\sum_n p_{k|n}x_n}{\sum_n p_{k|n}} \tag{22.21}$$

[极大化 $L(\mu)$的 Newton - Raphson 法将 μ 更新为 $\mu'=\mu-\left[\frac{\partial L}{\partial\mu}\Big/\frac{\partial^2 L}{\partial\mu^2}\right]$。]



假设 $\sigma=1$，对上述数据集，画出以 μ_1 和 μ_2 函数表示的似然函数的等势图。数据集由 32 个点组成。描述你的草图中的峰值并指出其宽度。

注意你所推导的最大化似然算法与第 20.4 节的软 K - 均值算法相同。既然已经清楚可以将聚类看成是混合密度建模，则可以推导改进的 K - 均值算法，从而对前面提到的问题加以修正。

22.3 对软 K-均值算法的改进

算法 22.2 表示软 K-均值算法的版本相应于这样一种建模假设：每个类都是有自己宽度的球状高斯变量（每个类都有自己的 $\beta^{(k)}=1/\sigma_k^2$）。算法更新本身的长度尺度 σ_k。算法也包括可以自行更新的聚类权重参数 $\pi_1,\pi_2,\cdots,\pi_k$，可以对来自不等权重聚类中的数据进行精确建模。图 22.3 是对我们已经见到过的两个数据集的算法演示。第二个例子表明要花费很长的时间才能收敛，但是算法最终还是能区分小聚类和大聚类。

赋值步骤 责任是

$$r_k^{(n)}=\frac{\pi_k\frac{1}{(\sqrt{2\pi}\sigma_k)^I}\exp\left(-\frac{1}{\sigma_k^2}d(\boldsymbol{m}^{(k)},\boldsymbol{x}^{(n)})\right)}{\sum_{k'}\pi_k\frac{1}{(\sqrt{2\pi}\sigma_{k'})^I}\exp\left(-\frac{1}{\sigma_{k'}^2}d(\boldsymbol{m}^{(k')},\boldsymbol{x}^{(n)})\right)} \tag{22.22}$$

其中，I 是 $\boldsymbol{x}$ 的维度。

更新步骤 调整每个聚类的参数 $\boldsymbol{m}^{(k)}$、π_k 和 σ_k^2 来匹配它所负责的数据点。

$$\boldsymbol{m}^{(k)}=\frac{\sum_n r_k^{(n)}\boldsymbol{x}^{(n)}}{R^{(k)}} \tag{22.23}$$

$$\sigma_k^2=\frac{\sum_n r_k^{(n)}(\boldsymbol{x}^{(n)}-\boldsymbol{m}^{(k)})^2}{IR^{(k)}} \tag{22.24}$$

$$\pi_k=\frac{R^{(k)}}{\sum_k R^{(k)}} \tag{22.25}$$

其中，$R^{(k)}$ 是均值 k 的总责任

$$R^{(k)}=\sum_n r_k^{(n)} \tag{22.26}$$

算法 22.2 软 K-均值算法，第 2 版。

第 2 版的软 K-均值算法[①]是用球状高斯分布混合拟合数据的最大似然算法——“球状”的意思是高斯过程的方差在所有方向都相同。这个算法仍然不能很好地对如图 20.6 所示的雪茄状聚类进行建模。如果希望通过方差可能不等的轴向高斯对聚类进行建模，就用算法 22.4 中的规则（22.27）和（22.28）来代替赋值规则（22.22）和方差更新规则（22.24）。

① 此算法确实能使似然最大化的证明请参见第 33.7 节。

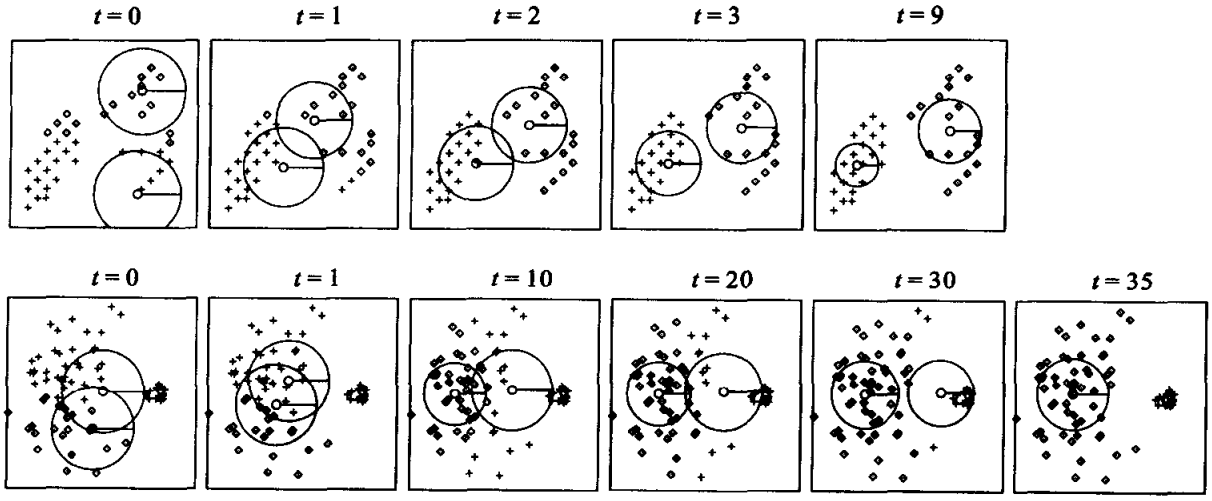


图 22.3 $K=2$ 的软 K - 均值算法，应用在(a) 图 20.3 的 40 个点的数据集；(b) 图 20.5 的“小”与“大”数据集。

$$r_k^{(n)} = \frac{\pi_k \dfrac{1}{\prod_{i=1}^{I}\sqrt{2\pi}\sigma_i^{(k)}} \exp\left(-\sum_{i=1}^{I}(m_i^{(k)} - x_i^{(n)})^2/2(\sigma_i^{(k)})^2\right)}{\sum_{k'}(\text{分子，用 } k' \text{ 代替 } k)} \tag{22.27}$$

$$\sigma_i^{2(k)} = \frac{\sum_n r_k^{(n)}(x_i^{(n)} - m_i^{(k)})^2}{R^{(k)}} \tag{22.28}$$

算法 22.4 软 K - 均值算法，第 3 版，对应于轴向高斯模型。

第 3 版的软 K - 均值算法应用于图 20.6 中的“双雪茄”数据集时的结果如图 22.5 所示。经过 30 次迭代，算法正确地定出了两个聚类。图 22.6 显示将同一算法应用于“小”与“大”数据集，再次得到聚类的正确位置。

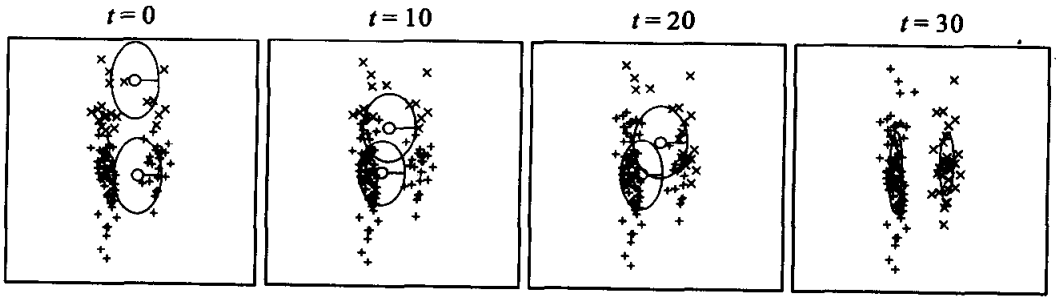


图 22.5 应用于包含两个雪茄状聚类的数据的软 K - 均值算法，第 3 版。$K=2$(同图 20.6 比较)。

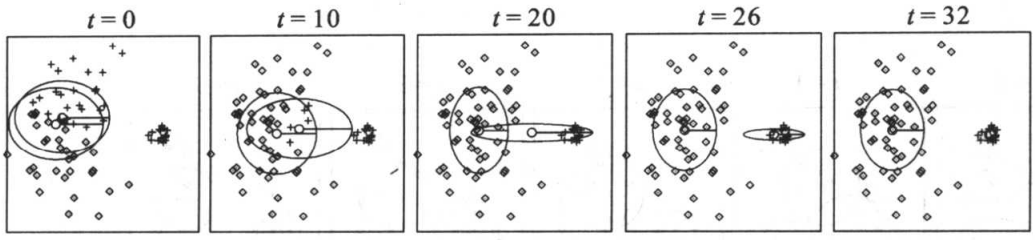


图 22.6 应用于“小”与“大”数据集的软 K - 均值算法，第 3 版。$K=2$。

22.4 最大似然法的致命缺陷

最后，图 22.7 给我们一个警示：当采用 $K=4$ 个均值拟合第一个简单数据集时，有时发现会形成只包含一两个数据点的非常小的聚类。这是第 2 版和第 3 版软 K - 均值聚类算法的病态特性。

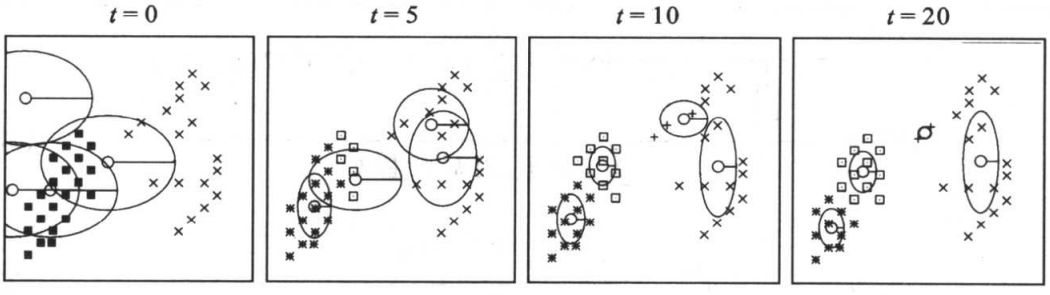


图 22.7 应用于 40 点数据集的软 K - 均值算法。$K=4$。注意：在算法收敛时，在两个数据点之间形成了一个很小的聚类。

▷ **习题 22.6**[2] 如果一个均值 $\boldsymbol{m}^{(k)}$ 恰巧设定在一个数据点上，探讨会发生什么情况？证明如果方差 σ_k^2 足够小，则不可能返回：σ_k^2 会变得更小。

发散！

软 K - 均值算法可能会发散。将一个聚类只放在一个数据点上，使其方差变为零——可有任意大的似然函数！如果所找的是一个和部分数据完全拟合的高度调节模型，则最大似然法可能会失败。这个现象就叫做超拟合。人们对这些具有巨大似然的解不感兴趣，因为虽然这些参数设置可能会有很大的后验概率密度，但只是在参数空间中一个非常小的体积上有大的密度。所以和这个似然尖峰相对应的概率质量通常是很小的。

我们的结论是，最大似然方法不能为数据建模问题提供令人满意的通解：最大似然在某些参数设置下可能有无穷大的值。即使在似然没有无穷大尖峰的情况下，在

高维问题中,最大似然解往往也不具备代表性。

即使在低维问题中,最大似然解也可能不具备代表性。正如从基本统计知识中所知道的那样,高斯分布标准差 σ_N 的最大似然估计器(22.15)是有偏估计器。这是第24章要讨论的问题。

最大后验方法(MAP)

最大化似然方法的一种常见替代方法是对参数的贝叶斯后验概率密度进行最大化。然而,用一个先验乘以似然,再对后验最大化并未解决上述问题;后验概率往往也有无限大的尖峰,后验概率密度的最大值对整个后验分布而言往往不具备代表性。回顾在第4章遇见过的典型性概念:在高维空间中,大部分概率质量所在的典型集和具有最大概率密度的点的特性非常不一样。具有最大概率密度的点是非典型的。

不喜欢最大后验解的另一个原因是它和基有关。如果把基从参数 θ 非线性地变换到参数 $u=f(\theta)$,则 θ 的概率密度函数被转换成

$$P(u)=P(\theta)\left|\frac{\partial\theta}{\partial u}\right| \tag{22.29}$$

最大密度 $P(u)$ 和最大密度 $P(\theta)$ 通常并不一致。(下一章可以看到基的非线性变换的示意图。)答案随表示方式的改变而改变的方法似乎也是不理想的。

进一步的阅读

自动分类软件包 AutoClass(Hanson 等,1991b;Hanson 等,1991a)的核心就是软 K-均值算法。

22.5 进一步的习题

最大似然法可能会有用武之地的习题

习题 22.7[3] 构造一个新的 K-均值算法,将数据建模为 K 个任意高斯分布的混合,即高斯分布不必限制为轴向的。

▷ **习题 22.8**[2] (a) 把一个光子计数器指向一个遥远星体达一分钟,以便推断星体的亮度,即每分钟到达计数器的光子速率 λ。假设所收集的光子数 r 服从均值为 λ 的泊松分布

$$P(r|\lambda)=\exp(-\lambda)\frac{\lambda^r}{r!} \tag{22.30}$$

已知 $r=9$,λ 的最大似然估计是多少?求 $\ln\lambda$ 的误差限。

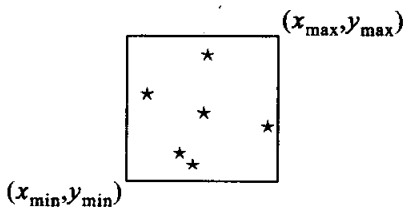
(b) 同样情况,但是现在假设计数器不仅检测从星体来的光子,而且还有“背

景"光子。背景光子的速率已知每分钟 $b=13$ 个光子。假设所收集的光子数 r 服从均值为 $\lambda+b$ 的泊松分布。现已知每分钟检测到 $r=9$ 个光子，λ 的最大似然估计是什么？对这个答案加以讨论，并讨论贝叶斯后验分布以及抽样理论中的"无偏估计器"$\hat{\lambda}\equiv r-b$。

习题 22.9[♪] 抛掷一枚变形硬币 N 次，得到 N_a 次正面和 N_b 次反面。假设正面出现的概率 p 服从均匀分布，并有 beta 先验分布。求 p 的最大似然和最大后验值，然后求出对数位 $a\equiv\ln[p/(1-p)]$ 的最大似然和最大后验值。与预测分布（即下一次抛掷出现正面的概率）进行比较。

▷ **习题 22.10**[♪] 两个人从监狱栅栏往外看；一个人看到了星星，另外一个人试图推断出窗户框的位置。

从房子的另一端，你通过窗户看到星星的位置为 $\{(x_n,y_n)\}$。因为除了星星以外全是黑的，你不能看到窗户的边框。假设窗户是长方形的，可见星星的位置是独立随机分布的。按照最大似然原理，$(x_{\min},y_{\min},x_{\max},y_{\max})$ 的推断值是什么？对于固定的 $x_{\min},y_{\min}$ 和 $y_{\max}$，画出似然以 $x_{\max}$ 的函数所表示的草图。



▷ **习题 22.11**[♪] 一个水手通过测量在海图上标出的 3 个位置为 (x_n,y_n) 浮标的方位角来推断自己的位置 (x,y)。设浮标的真实方位角为 θ_n，假设它所测量到的每个方位角 $\tilde{\theta}_n$ 叠加了一个具有小标准差 σ 的高斯噪声，由最大似然法所推断的位置是什么？

根据水手的经验规律，船的位置可以取为"三角帽"的中心，即由所测方位线相交而成的三角形的中心（图 22.8）。你能说服他对于最大似然的解答更好一些吗？

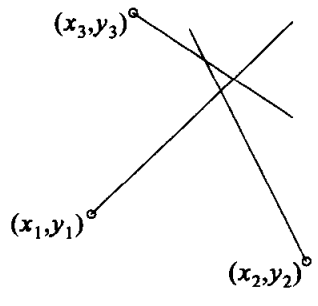


图 22.8 在一张海图上画 3 条稍有不同的方位线的标准方法产生一个三角形，这称为三角帽。水手的位置在哪里？

▷ **习题 22.12**[♪] **对一族指数模型的最大似然拟合。**

假设变量 x 来自具有以下形式的概率分布。

$$P(x|w) = \frac{1}{Z(w)}\exp\Big(\sum_k w_k f_k(x)\Big) \tag{22.31}$$

其中函数 $f_k(x)$ 已知，参数 $w = \{w_k\}$ 未知。给出包含 N 个点的数据集 $\{x^{(n)}\}$。

通过对对数似然函数求导，证明最大似然参数 w_{ML} 满足

$$\sum_x P(x|w_{ML}) f_k(x) = \frac{1}{N}\sum_n f_k(x^{(n)}) \tag{22.32}$$

其中左边是对所有 x 求和，右边是对数据点求和。这个结果的一种简便记法是在拟合模型下的每个函数平均必须等于由数据得出的函数平均

$$\langle f_k\rangle_{P(x|w_{ML})} = \langle f_k\rangle_{Data} \tag{22.33}$$

▷ **习题 22.13**[ρ] **满足约束的模型“最大熵”拟合。**

当所遇到的概率分布 $P(x)$ 只有少数几个事实已知时，最大熵原理（maxent）提供了选择满足这些约束的分布的一条规则。根据最大熵原理，应该选择在满足约束的前提下使熵

$$H = \sum_x P(x)\log 1/P(x) \tag{22.34}$$

最大化的 $P(x)$。假设约束确保已知某些函数 $f_k(x)$ 的平均，即

$$\langle f_k\rangle_{P(x)} = F_k \tag{22.35}$$

通过引入拉格朗日因子（每个约束一个，包括归一化），证明最大熵分布的形式是

$$P(x)_{Maxent} = \frac{1}{Z}\exp\Big(\sum_k w_k f_k(x)\Big) \tag{22.36}$$

其中参数 Z 和 $\{w_k\}$ 被设定为满足约束(22.35)。

因此最大熵方法所给出的结果和一族指数模型的最大似然拟合所给出的结果相同（前一习题）。

最大熵方法有时也被推荐用于在贝叶斯建模中分配先验分布。尽管最大熵方法的结果有时是有趣和富于启发性的，笔者并不赞成用这种方法实现先验分布的选择。

最大熵方法有时也被提议为解决推理问题的方法——例如，“已知这个非均匀的六面骰子的平均点数是2.5，试问 $(p_1, p_2, p_3, p_4, p_5, p_6)$ 的概率分布是什么？”笔者认为用最大熵求解这个问题并不妙，它会给出不合理的答案。解决这个推理问题的正确方法是贝叶斯定理。

最大似然和 MAP 法会遇到困难的习题

▷ **习题 22.14**[ρ] 这个习题要研究以下思想：最大化概率密度不是寻找密度函数中有代表性的点的好方法。考虑一个 k 维空间中的高斯分布

$$P(w) = (1/\sqrt{2\pi}\sigma_w)^k \exp\Big(-\sum_1^k w_i^2/2\sigma_w^2\Big)$$

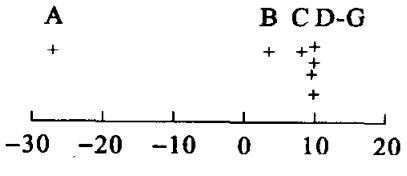
证明几乎所有高斯概率质量都集中在半径为 $r = \sqrt{k}\sigma_w$、厚度正比于 $r/\sqrt{k}$ 的薄球壳中。例如，在 1 000 维空间中，对于 $\sigma_w = 1$ 的高斯分布，90% 的质量都在半径为 31.6、厚度

为 2.8 的壳中。但是在原点的概率密度比集中了大部分质量的球壳中的密度大 $e^{k/2} \simeq 10^{217}$ 倍。

现在考虑 1 000 维空间中的两个高斯密度，它们在半径 σ_W 上仅相差 1%，且有相等的总概率质量。证明在具有较小半径 σ_W 的高斯分布中心处，最大概率密度所大出的倍数 $\sim \exp(0.01k) \simeq 20\,000$。

在病态问题中，一个典型的后验分布往往是具有可变均值和标准差的高斯的加权叠加，所以真实的后验分布有一个歪斜的高峰，概率密度的最大值出现在具有最小标准差的高斯分布的均值附近，并不是具有最大权重的那个高斯分布。

▷ **习题 22.15**[♂] **7 位科学家** 从 N 个分布中抽取 N 个数据点 $\{x_n\}$，这些分布都是高斯分布，具有相同的均值 μ，但是有不同的未知标准差 σ_n。已知数据，最大似然参数 μ 和 $\{\sigma_n\}$ 各是什么？例如，实验技巧悬殊较大的 7 位科学家(A,B,C,D,E,F,G)对 μ 进行测量。可以预计其中一些人做得很精确(即有小的 σ_n 值)，而有些人则交出极不准确的答案(即有很大的 σ_n)。如图 22.9 所示为他们的 7 个结果。μ 是多少？每个科学家的可靠性有多大？



科学家	x_n
A	−27.020
B	3.570
C	8.191
D	9.898
E	9.603
F	9.945
G	10.056

图 22.9 7 位科学家对参数 μ 的 7 个测量值 $\{x_n\}$，每人都有自己的噪声水平 σ_n。

笔者希望你同意这样的事实：从直观上说，A 和 B 是不熟练的测量，D ~ G 要好些，μ 的真值接近 10。但是你从最大化似然中得到什么启发？

▷ **习题 22.16**[♂] **MAP 方法的问题** 有一批小器具 $i = 1, \cdots, k$ 具有称为“wodge”的特性 w_i，在噪声水平为 $\sigma_\nu = 1.0$ 的噪声环境中对特性逐件进行测量。这些量的模型来自高斯先验 $P(w_i|\alpha) = \text{Normal}(0, 1/\alpha)$，其中 $\alpha = 1/\sigma_W^2$ 未知。关于这个方差的先验是 $\log \sigma_W$ 从 $\sigma_W = 0.1$ 到 $\sigma_W = 10$ 平坦分布。

场景 1 假设已对 4 个器具进行了测量，得到以下数据：$\{d_1, d_2, d_3, d_4\} = \{2.2, -2.2, 2.8, -2.8\}$，我们想推断这 4 个器具的特性 wodge。

(a) 求极大化后验概率 $P(\boldsymbol{w}, \log \alpha | \boldsymbol{d})$ 的 $\boldsymbol{w}$ 和 α 值。

(b) 在 α 上边缘化,并对已知数据求 $\boldsymbol{w}$ 的后验概率密度。[需要积分技巧。解答参见参考文献 MacKay(1999a)。]求 $P(\boldsymbol{w}|\boldsymbol{d})$ 的最大值。[答案:两个最大点——一个在 $\boldsymbol{w}_{\mathrm{MP}} = \{1.8, -1.8, 2.2, -2.2\}$,所有 4 个参数的误差限(由高斯分布近似后验而得)是 ± 0.9;另一个在 $\boldsymbol{w}'_{\mathrm{MP}} = \{0.03, -0.03, 0.04, -0.04\}$,误差限为 ± 0.1。]

场景 2 假设除了上面的 4 个测量值以外,现在知道另外还有 4 个器具,使用精度低很多的仪器进行测量,其中 $\sigma'_{\nu} = 100.0$。像在典型的病态问题中一样,现在既有良好定义的参数又有病态定义的参数。从这些测量所得出的数据是一串无信息价值的值,$\{d_5, d_6, d_7, d_8\} = \{100, -100, 100, -100\}$。

要求重新推断器具的 wodge 值。直观上,测量不佳器具的虚假信息对于测量良好的器具所做推断的影响应该忽略不计。但对于 MAP 方法,情况如何?

(a) 求使后验概率 $P(\boldsymbol{w}, \log \alpha | \boldsymbol{d})$ 极大化的 $\boldsymbol{w}$ 和 α 值。

(b) 求 $P(\boldsymbol{w}|\boldsymbol{d})$ 的最大点。[答案:只有一个最大点,$\boldsymbol{w}_{\mathrm{MP}} = \{0.03, -0.03, 0.03, -0.03, 0.000\ 1, -0.000\ 1, 0.000\ 1, -0.000\ 1\}$,所有 8 个参数的误差限都是 ± 0.11。]

22.6 习题解答

习题 22.5 的解答 图 22.10 显示了 32 个数据点的似然函数的等势图。峰值基本上集中于点(1,5)和(5,1),等势线基本上近乎是圆。每个峰值的宽度是一个标准差 $\sigma/\sqrt{16} = 1/4$。峰值在形状上近似高斯。

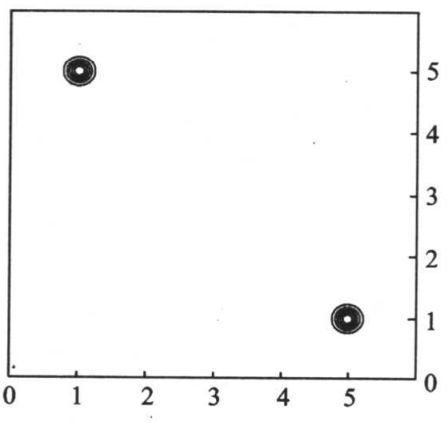


图 22.10 以 μ_1 和 μ_2 为自变量的似然函数

习题 22.12 的解答 对数似然是

$$\ln P(\{\boldsymbol{x}^{(n)}\}|\boldsymbol{w}) = -N \ln Z(\boldsymbol{w}) + \sum_n \sum_k w_k f_k(\boldsymbol{x}^{(n)}) \qquad (22.37)$$

$$\frac{\partial}{\partial w_k}\ln P(\{\boldsymbol{x}^{(n)}\}\mid \boldsymbol{w}) = -N\frac{\partial}{\partial w_k}\ln Z(\boldsymbol{w}) + \sum_n f_k(\boldsymbol{x}) \tag{22.38}$$

现在,对归一化常数的对数求导,会出现一些有趣的情况。

$$\frac{\partial}{\partial w_k}\ln Z(\boldsymbol{w}) = \frac{1}{Z(\boldsymbol{w})}\sum_{\boldsymbol{x}}\frac{\partial}{\partial w_k}\exp\Big(\sum_{k'} w_{k'}f_{k'}(\boldsymbol{x})\Big)$$

$$= \frac{1}{Z(\boldsymbol{w})}\sum_{\boldsymbol{x}}\exp\Big(\sum_{k'} w_{k'}f_{k'}(\boldsymbol{x})\Big)f_k(\boldsymbol{x}) = \sum_{\boldsymbol{x}} P(\boldsymbol{x}\mid \boldsymbol{w})f_k(\boldsymbol{x}) \tag{22.39}$$

于是有

$$\frac{\partial}{\partial w_k}\ln P(\{\boldsymbol{x}^{(n)}\}\mid \boldsymbol{w}) = -N\sum_{\boldsymbol{x}} P(\boldsymbol{x}\mid \boldsymbol{w})f_k(\boldsymbol{x}) + \sum_n f_k(\boldsymbol{x}) \tag{22.40}$$

在似然函数取最大值时,有

$$\sum_{\boldsymbol{x}} P(\boldsymbol{x}\mid \boldsymbol{w}_{\mathrm{ML}})f_k(\boldsymbol{x}) = \frac{1}{N}\sum_n f_k(\boldsymbol{x}^{(n)}) \tag{22.41}$$

第 23 章

有用的概率分布

在贝叶斯数据建模中，有一小组反复出现的概率分布。本章的目的就是介绍这些分布，以便在实际应用中遇到它们时不会感到吓人。

也许除了高斯分布之外，无需硬记它们；当然如果一个分布是足够重要的，人们将自然而然地记住它；不然，也非常容易查阅到。

23.1 整数上的分布

二项式分布，泊松分布，指数分布

在本书初始处已经遇到过二项式分布和泊松分布。

对于整数 r，具有参数 f（偏倚，$f\in[0,1]$）和 N（试验次数）的二项式分布是

$$P(r|f,N)=\binom{N}{r}f^r(1-f)^{N-r},\quad r\in\{0,1,2,\cdots,N\} \tag{23.1}$$

举例说明二项式分布是如何出现的。当投掷一枚偏倚为 f 的硬币 N 次，观察正面朝上的次数为 r 的分布。如图 23.1 所示。

具有参数 $\lambda>0$ 的泊松分布为

$$P(r|\lambda)=e^{-\lambda}\frac{\lambda^r}{r!},\quad r\in\{0,1,2,\cdots\} \tag{23.2}$$

比如，假定这个像素点的平均亮度与平均光子数 λ 相对应，当计数在固定时间间隔内到达像素点上的光子数 r 时，就得到一个泊松分布。如图 23.2 所示。

整数上的指数分布

$$P(r|f)=f^r(1-f),\quad r\in(0,1,2,\cdots,\infty) \tag{23.3}$$

出现在等待问题中。例如，当抛掷一枚公平的六面骰子时，需要等多久才能投出一次为 6 的面？答案：投掷次数 r 的概率分布是一个参数为 $f=5/6$ 的整数上的指数分布。这个分布也可以写成

$$P(r|f)=(1-f)e^{-\lambda r},\quad r\in(0,1,2,\cdots,\infty) \tag{23.4}$$

其中 $\lambda=\ln(1/f)$。

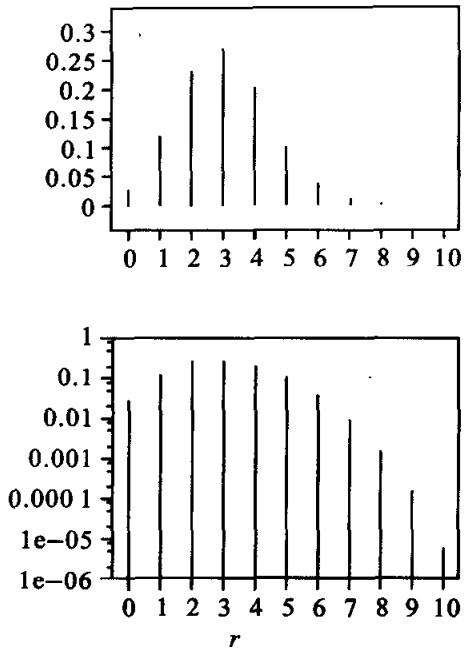


图 23.1 线性坐标(上半图)和对数坐标(下半图)上的二项式分布 $P(r|f=0.3,N=10)$。

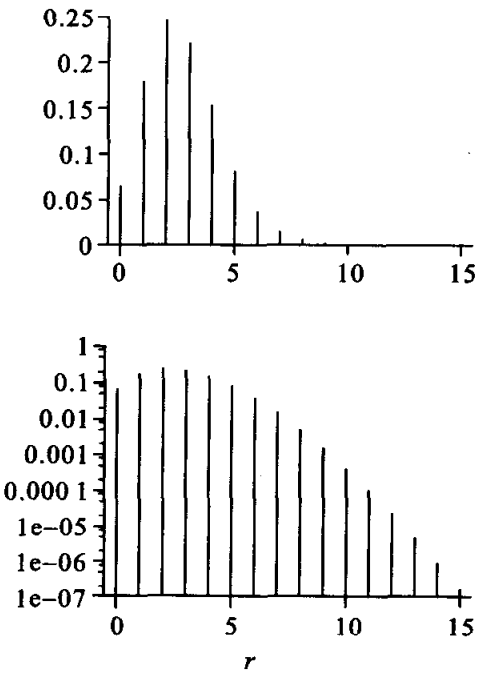


图 23.2 线性坐标(上半图)和对数坐标(下半图)上的泊松分布 $P(r|\lambda=2.7)$。

23.2 无界实数上的分布

高斯分布,学生分布,柯西分布,双边指数分布,反双曲余弦分布

一个均值为 μ、标准差为 σ 的高斯分布或称正态分布为

$$P(x|\mu,\sigma)=\frac{1}{Z}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right),\quad x\in(-\infty,\infty) \tag{23.5}$$

其中

$$Z=\sqrt{2\pi\sigma^2} \tag{23.6}$$

有时使用量 $\tau\equiv 1/\sigma^2$ 会更有所帮助,该参数叫做高斯分布的精度参数。

可以通过下面的计算产生一个来自标准单变量高斯分布的样本 z。

$$z=\cos(2\pi u_1)\sqrt{2\ln(1/u_2)} \tag{23.7}$$

其中,u_1 与 u_2 在(0,1)上均匀分布。于是,无额外开销就可以产生独立于第一个样本的第二个样本 $z_2=\sin(2\pi u_1)\sqrt{2\ln(1/u_2)}$。

高斯分布应用广泛,并往往被认为是客观世界中一种很普通的分布,但笔者对此表示怀疑。是的,单峰分布可能是常见的;但是高斯分布是一个特殊的、相当极端的单峰分布。其拖尾很小:其对数概率密度呈二次衰减。x 与 μ 之间的典型偏移量为 σ,但是 x 与 μ 之间的偏移量大于 2σ,3σ,4σ 和 5σ 的概率分别为 0.046,0.003,6×10^{-5} 和 6×10^{-7}。根据笔者的经验,偏移量为典型偏移量的 4~5 倍这种情况可能很罕见,但是不会罕见到概率为 6×10^{-5} 这个程度! 因此,笔者强烈要求谨慎使用高斯分布:如果一个变量实际上具有拖尾较大的分布,且使用高斯分布对它进行建模,那么为了降低外部的偏差,模型的其余部分会扭曲,就像被橡皮圈箍得变皱的纸张一样。

▷ **习题 23.1**[ㄅ] 选取一个概率分布函数被认为是钟形的变量,收集数据,并画出这个变量的经验分布图。将这个分布表示为对数坐标的柱状图,并研究其拖尾是否能用高斯分布很好地模拟出来。[可供研究的变量实例是语音信号的幅度。]

一种比高斯分布具有更大拖尾的分布是混合高斯分布。例如,一个由两个高斯分布混合而成的分布,可由两个均值、两个标准差和两个混合系数 π_1、π_2 定义,混合系数满足 $\pi_1+\pi_2=1$,$\pi_i\geqslant 0$。

$$P(x|\mu_1,\sigma_1,\pi_1,\mu_2,\sigma_2,\pi_2)=\frac{\pi_1}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{(x-\mu_1)^2}{2\sigma_1^2}\right)+\frac{\pi_2}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{(x-\mu_2)^2}{2\sigma_2^2}\right)$$

如果将无穷多个均值为 μ 的高斯分布通过适当的加权进行混合,将获得一个学生-t 分布

$$P(x|\mu,s,n)=\frac{1}{Z}\frac{1}{(1+(x-\mu)^2/(ns^2))^{(n+1)/2}} \tag{23.8}$$

其中

$$Z=\sqrt{\pi n s^2}\frac{\Gamma(n/2)}{\Gamma((n+1)/2)} \tag{23.9}$$

n 叫做自由度，而 Γ 是伽马函数。当 $n>1$ 时学生分布 (23.8) 存在一个均值，且等于 μ。当 $n>2$ 时，此分布还存在一个有限的方差，$\sigma^2=ns^2/(n-2)$。当 $n\to\infty$ 时，学生分布趋近于一个均值为 μ、标准差为 s 的正态分布。学生分布同时出现在经典统计学（作为某统计学的采样理论分布）和贝叶斯推理（作为来自标准差不确定的高斯分布的变量的概率分布）中。

对于 $n=1$ 的特例，学生分布又称为柯西分布。如图 23.3 所示。

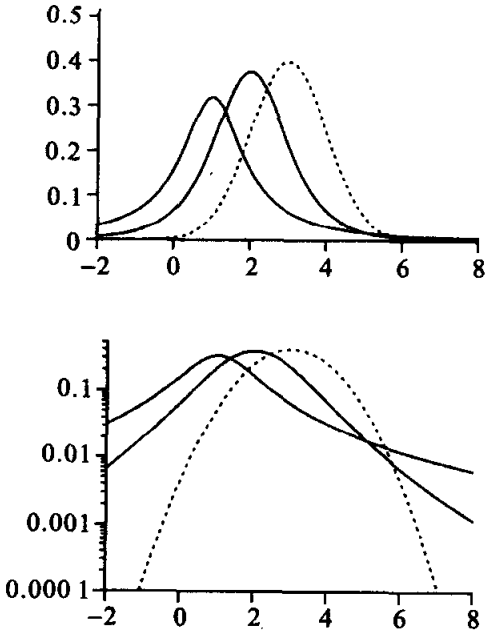


图 23.3 3个单峰分布。参数为 $(m,s)=(1,1)$（粗线）（柯西分布）和 $(2,4)$（细线）的两个学生分布，及均值为 $\mu=3$、标准差为 $\sigma=3$ 的高斯分布（虚线）。用线性纵向坐标系表示（上半图）和用对数纵向坐标系表示（下半图）。值得注意的是，在上面的“钟形曲线”中，柯西分布的重拖尾并不明显。

一个拖尾大小位于学生分布和高斯分布之间的分布是双边指数分布

$$P(x|\mu,s)=\frac{1}{Z}\exp\left(-\frac{|x-\mu|}{s}\right),\quad x\in(-\infty,\infty) \tag{23.10}$$

其中

$$Z=2s \tag{23.11}$$

反双曲余弦分布

$$P(x|\beta)\propto\frac{1}{[\cosh(\beta x)]^{1/\beta}} \tag{23.12}$$

是独立分量分析中的常用模型。在 β 很大的极限情况下，概率分布 $P(x|\beta)$ 变成一个

双边指数分布。在 $\beta\to 0$ 的极限情况下，$P(x|\beta)$ 趋于一个均值为 0、方差为 $1/\beta$ 的高斯分布。

23.3 正实数上的分布

指数分布，伽马分布，逆伽马分布，对数正态分布

指数分布

$$P(x|s)=\frac{1}{Z}\exp\left(-\frac{x}{s}\right),\quad x\in(0,\infty) \tag{23.13}$$

其中

$$Z=s \tag{23.14}$$

发生在等待问题中。已知汽车随机地独立到达，且平均每 s 分钟来一辆，在 Poissonville 多久才能等到一辆汽车？答案：你的等待时间 x 的概率分布是均值为 s 的指数分布。

伽马分布很像高斯分布，只是高斯分布的范围是从 $-\infty$ 到 $+\infty$，而伽马分布的范围是从 0 到 ∞。如同高斯分布有两个参数 μ 和 σ 来控制分布的均值和宽度一样，伽马分布也有两个参数。它是一个单参数的指数分布(23.13)和一个多项式 x^{c-1} 的乘积。多项式的指数 c 是第二个参数。

$$P(x|s,c)=\Gamma(x;s,c)=\frac{1}{Z}\left(\frac{x}{s}\right)^{c-1}\exp\left(-\frac{x}{s}\right),\quad 0\leqslant x<\infty \tag{23.15}$$

其中

$$Z=\Gamma(c)s \tag{23.16}$$

这是一个简单的尖峰分布，其均值为 sc，方差为 s^2c。

人们常常将一个正实变量 x 表示成它的对数 $l=\ln x$。l 的概率密度为

$$P(l)=P(x(l))\left|\frac{\partial x}{\partial l}\right|=P(x(l))x(l) \tag{23.17}$$

$$=\frac{1}{Z_l}\left(\frac{x(l)}{s}\right)^{c}\exp\left(-\frac{x(l)}{s}\right) \tag{23.18}$$

其中

$$Z_l=\Gamma(c) \tag{23.19}$$

[伽马分布是根据它的归一化常数命名的——在笔者看来，这是个奇怪的约定！]

图 23.4 显示了一对伽马分布，以 x 和 l 的函数表示。应该注意的是，原始伽马分布(23.15)在 $x=0$ 处可能会有一个“尖峰”，而在 l 上的分布从不会有这样一个尖峰。这个尖峰是选择一个坏基的人为产物。

在 $sc=1$，$c\to 0$ 的极限情况，得到关于尺度参数的无信息价值的先验，即 $1/x$ 先验。这个不适合的先验被称为无信息价值的，因为它没有长度尺度与之关联，没有 x 的特征值，所以倾向于使 x 的所有值相等。它在再参量化 $x=mx$ 下是不变的。如果

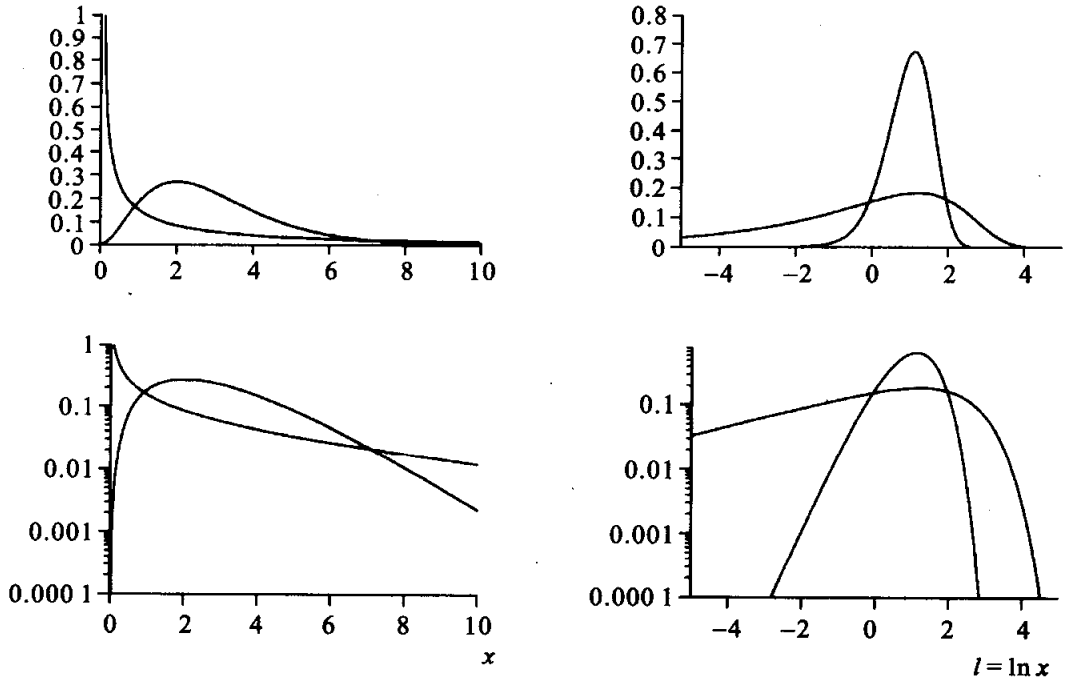


图 23.4 两个伽马分布，其参数为 $(s,c)=(1,3)$（粗线）和 $(10,0.3)$（细线），用线性纵坐标（上图）和对数纵坐标（下图）表示；且左边显示为 x 的函数(23.15)，右边显示为 $l=\ln x$ 的函数(23.18)。

将 $1/x$ 的概率密度转换为在 $l=\ln x$ 上的一个概率密度，那么可以发现后者的密度是均匀的。

▷ **习题 23.2**[1] 想象一下，对一个正变量 x 以其立方根的形式进行再参量化，得到 $u=x^{1/3}$。如果 x 的概率密度是不适当的分布 $1/x$，那么 u 的概率密度是什么？

在 $l=\ln x$ 上，伽马分布永远是一个单峰密度，由图中可以看见它是不对称的。如果 x 服从一个伽马分布，且我们决定考虑 x 的倒数形式 $v=1/x$，将获得一个新的分布，其密度在 l 上从左至右翻转：这个关于 v 的概率密度叫做*一个逆伽马分布*。

$$P(v|s,c)=\frac{1}{Z_v}\left(\frac{1}{sv}\right)^{c+1}\exp\left(-\frac{1}{sv}\right),\quad 0\leqslant v<\infty \tag{23.20}$$

其中

$$Z_v=\Gamma(c)/s \tag{23.21}$$

伽马分布与逆伽马分布（如图 23.5 所示）出现在许多从数据推断一个正数的推理问题中。例如从一些噪声样本中推断高斯噪声的方差，及从一些计数中推断一个泊松分布的速率参量。

伽马分布也自然地出现在泊松分布事件之间的等待时间分布中。已知一个速率为 λ 的泊松过程，则第 m 个事件的到达时间 x 的概率密度是

$$\frac{\lambda(\lambda x)^{m-1}}{(m-1)!}\mathrm{e}^{-\lambda x} \tag{23.22}$$

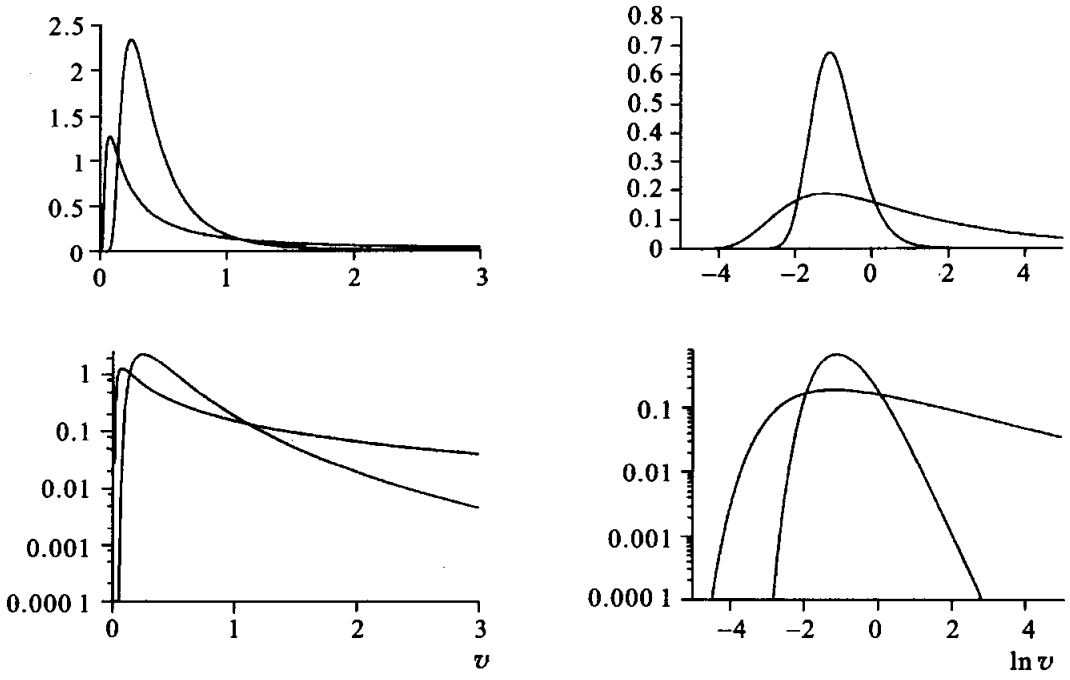


图 23.5 两个逆伽马分布,其参数为$(s,c)=(1,3)$(粗线)和$(10,0.3)$(细线),用线性纵坐标(上图)和对数纵坐标(下图)表示;且左边显示为 x 的函数,右边显示为 $l=\ln x$ 的函数。

对数正态分布

另一个在正实数 x 上的分布是对数正态分布(如图 23.6 所示),它是当 $l=\ln x$ 有一个正态分布时所得的分布。定义 m 为 x 的中值,s 为 $\ln x$ 的标准差。

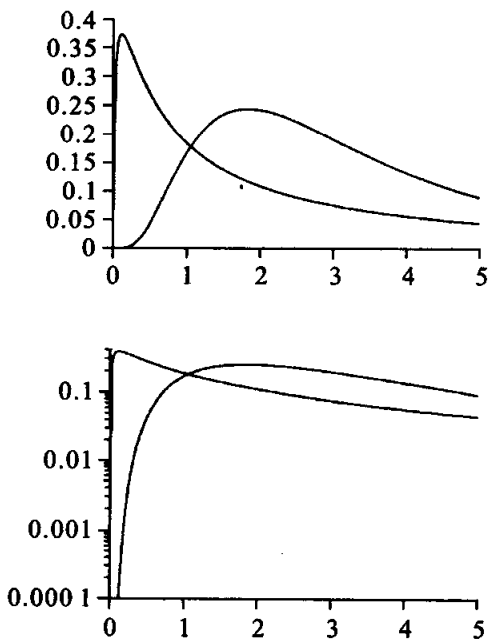


图 23.6 两个对数正态分布,其参数为$(m,s)=(3,1.8)$(粗线)和$(3,0.7)$(细线),用线性纵坐标(上图)和对数纵坐标(下图)表示。[是的,它们具有相同的中值 $m=3$。]

$$P(l|m,s)=\frac{1}{Z}\exp\left(-\frac{(l-\ln m)^2}{2s^2}\right),\quad l\in(-\infty,\infty) \tag{23.23}$$

其中

$$Z=\sqrt{2\pi s^2} \tag{23.24}$$

则

$$P(x|m,s)=\frac{1}{x}\exp\left(-\frac{(\ln x-\ln m)^2}{2s^2}\right),\quad x\in(0,\infty) \tag{23.25}$$

23.4 在周期变量上的分布

一个周期变量 θ 是在[0,2π]上的具有如下性质的实数:$\theta=0$ 和 $\theta=2\pi$ 是等价的。

与高斯分布对实变量所起作用相对应,周期变量有类似的分布,称为 Von Mises 分布。

$$P(\theta|\mu,\beta)=\frac{1}{Z}\exp(\beta\cos(\theta-\mu)),\quad \theta\in(0,2\pi) \tag{23.26}$$

归一化参数是 $Z=2\pi I_0(\beta)$,其中 $I_0(x)$是一个改进的 Bessel 函数。

来自圆周围的布朗扩散的一个分布是卷曲的高斯分布

$$P(\theta|\mu,\sigma)=\sum_{n=-\infty}^{\infty}\text{Normal}(\theta;(\mu+2\pi n),\sigma),\quad \theta\in(0,2\pi) \tag{23.27}$$

23.5 概率上的分布

Beta 分布,Dirichlet 分布,熵分布

Beta 分布是在变量 p 上的一个概率密度,其中概率 $p\in(0,1)$。

$$P(p|u_1,u_2)=\frac{1}{Z(u_1,u_2)}p^{u_1-1}(1-p)^{u_2-1} \tag{23.28}$$

参数 u_1,u_2可取任何正数值。其归一化常数是 Beta 函数

$$Z(u_1,u_2)=\frac{\Gamma(u_1)\Gamma(u_2)}{\Gamma(u_1+u_2)} \tag{23.29}$$

特例包括均匀分布——$u_1=1,u_2=1$;Jeffreys 先验分布——$u_1=0.5,u_2=0.5$;及不合适的拉普拉斯先验分布——$u_1=0,u_2=0$。如果将 Beta 分布转换为相应对数位形式 $l\equiv\ln p/(1-p)$的概率密度,那么可以发现它永远都是在 l 上的一个很好的钟形密度,p 上的密度可能会在 $p=0$ 和 $p=1$ 处出现奇异点(如图 23.7 所示)。

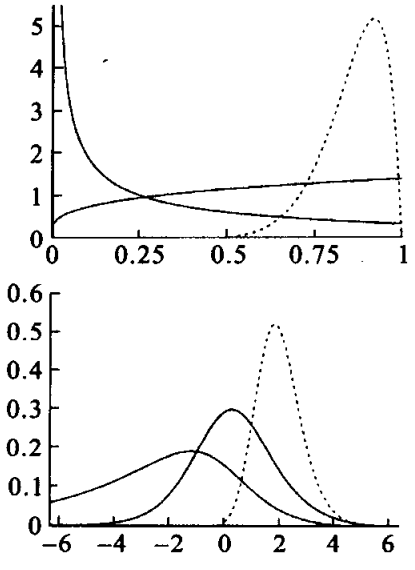


图 23.7 3 个 Beta 分布，其参数为 $(u_1, u_2) = (0.3, 1), (1.3, 1)$ 和 $(12, 2)$。上图表示 p 的函数 $P(p|u_1, u_2)$；下图表示相应对数位形式 $\ln \frac{p}{1-p}$ 的密度函数。注意：作为一个对数位的函数，该密度的特性有多良好。

更高维数

Dirichlet 分布是在 I 维向量 $\boldsymbol{p}$ 上的一个密度，其中 I 个分量都为正且其和为 1。Beta 分布是 Dirichlet 分布在 $I=2$ 时的特例。Dirichlet 分布是以度量 $\boldsymbol{u}$（所有系数 $u_i > 0$ 的一个向量）为参数的，在此将它写成 $\boldsymbol{u} = \alpha \boldsymbol{m}$，其中 $\boldsymbol{m}$ 是在 I 个分量上的归一化度量（$\sum m_i = 1$），且 α 是一个正数。

$$P(\boldsymbol{p} \mid \alpha \boldsymbol{m}) = \frac{1}{Z(\alpha \boldsymbol{m})} \prod_{i=1}^{I} p_i^{\alpha m_i - 1} \delta\left(\sum_i p_i - 1\right) \equiv \text{Dirichlet}^{(I)}(\boldsymbol{p} \mid \alpha \boldsymbol{m}) \quad (23.30)$$

函数 $\delta(x)$ 是 Dirac δ 函数，它将此分布限制为单纯形，以使 $\boldsymbol{p}$ 归一化，即 $\sum_i p_i = 1$。Dirichlet 分布的归一化常数是

$$Z(\alpha \boldsymbol{m}) = \prod_i \Gamma(\alpha m_i) / \Gamma(\alpha) \quad (23.31)$$

向量 $\boldsymbol{m}$ 是以下概率分布的均值。

$$\int \text{Dirichlet}^{(I)}(\boldsymbol{p} | \alpha \boldsymbol{m}) \boldsymbol{p} \mathrm{d}^I \boldsymbol{p} = \boldsymbol{m} \quad (23.32)$$

当使用概率向量 $\boldsymbol{p}$ 时，在“softmax 基”下使用往往是有帮助的。例如，一个三维概率 $\boldsymbol{p} = (p_1, p_2, p_3)$ 表示为满足 $a_1 + a_2 + a_3 = 0$ 的 3 个数 a_1, a_2, a_3，并且

$$p_i = \frac{1}{Z} \mathrm{e}^{a_i}, \text{其中 } Z = \sum_i \mathrm{e}^{a_i} \quad (23.33)$$

这个非线性转换类似于一个标量的变换 $\sigma \to \ln \sigma$，及一个单一概率的对数位变换 $p \to \ln \frac{p}{1-p}$。在 softmax 基上，Dirichlet 分布(23.30)的指数上那些丑陋的 -1 会消失，且密度函数变成

$$P(\boldsymbol{a} \mid \alpha \boldsymbol{m}) \propto \frac{1}{Z(\alpha \boldsymbol{m})} \prod_{i=1}^{I} p_i^{\alpha m_i} \delta\left(\sum_i a_i\right) \tag{23.34}$$

参数 α 的作用主要体现在两个方面。首先，α 度量了这个分布的对比度(如图 23.8 所示)；它测量了我们预计分布中典型样本 $\boldsymbol{p}$ 与其均值 $\boldsymbol{m}$ 相差多远，就像在高斯分布中精度 $\tau = 1/\sigma^2$ 度量了样本与它的均值偏差多远一样。一个大的 α 值会使 $\boldsymbol{p}$ 的分布在 $\boldsymbol{m}$ 附近急剧出现尖峰。通过绘制来自 Dirichlet$^{(I)}(\boldsymbol{p} \mid \alpha \boldsymbol{m})$ 分布的典型样本，其中 $\boldsymbol{m}$ 被设成均匀向量 $m_i = 1/I$，再作出一个 Zipf 图，即分量 p_i 值的等级图，就可以形象地看到 α 在高维情形中的影响。通常将 p_i(纵轴)和等级(横轴)画在对数坐标下，这样可以用直线来表示幂律关系。图 23.9 表示了一个样本的这些图，这个样本来自 $I = 100$ 和 $I = 1\,000$ 而 α 从 0.1 到 1 000 的总体。对于大的 α 值，许多分量的值相近，图变浅。对于小的 α 值，典型情况是，一个分量 p_i 得到一个占压倒性优势的概率，而其余的分量则共同得到一个小概率，另一分量 $p_{i'}$ 得到类似的大份额。当 α 趋于 0 时，图会趋于一个越来越陡的幂律。

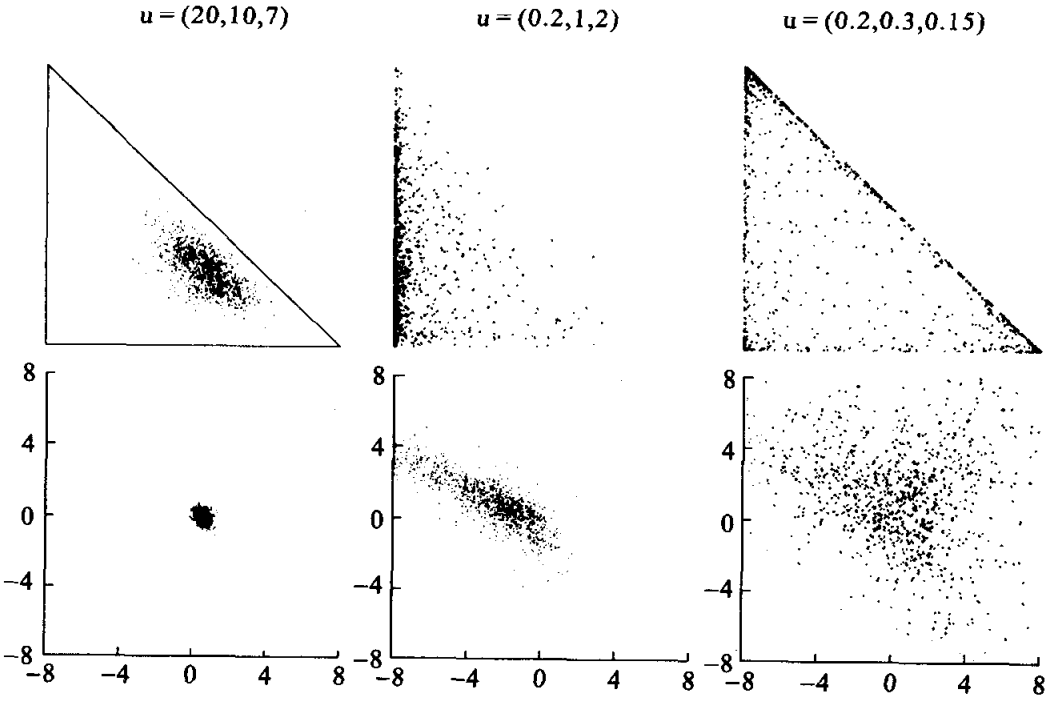


图 23.8 在一个三维概率向量(p_1, p_2, p_3)上的 3 个 Dirichlet 分布。上图表示从每个分布中抽取 1 000 个随机点，两个轴表示 p_1 和 p_2 的值。$p_3 = 1 - (p_1 + p_2)$。第一图中的三角形是合法概率分布的单纯形。下图表示了在 softmax 基(式(23.33))上的相同数据点。两轴表示 a_1 和 a_2。$a_3 = -a_1 - a_2$。

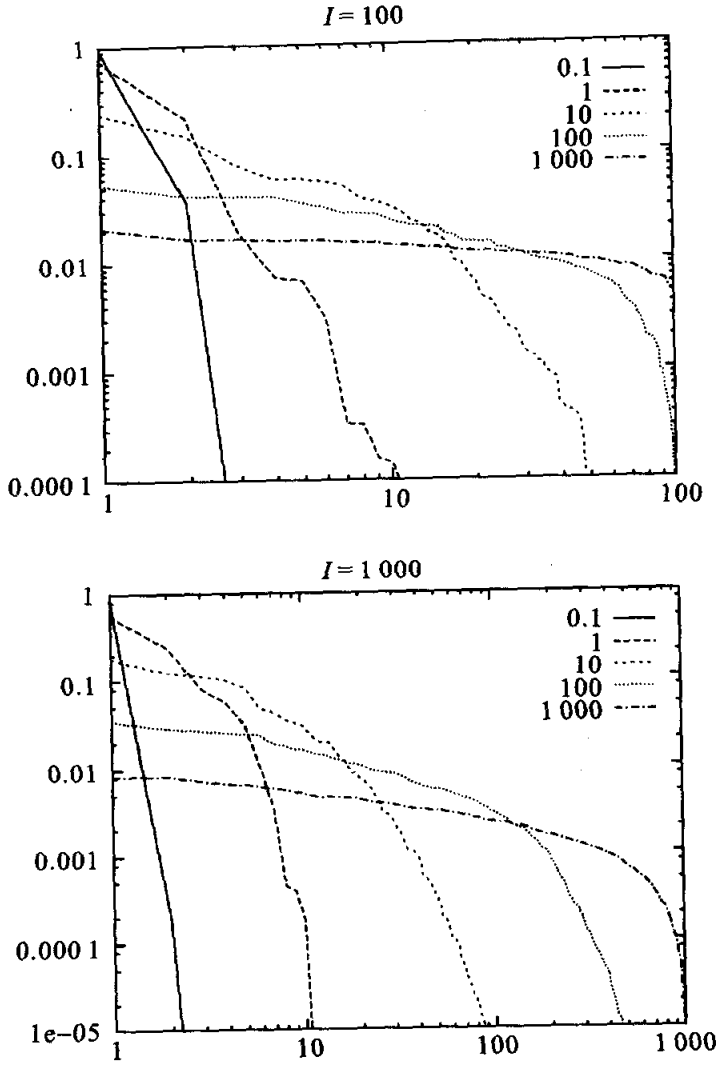


图 23.9 由具有不同 α 值($\alpha = 0.1, \cdots, 1\,000$)的 Dirichlet 分布产生的随机样本的 Zipf 图。对于 $I = 100$ 或 1 000 的每个值和每个 α,产生 Dirichlet 分布的一个样本 $\boldsymbol{p}$。这个 Zipf 图表示按幅度排列的概率 p_i 及其级别曲线。

其次,观察由 $\boldsymbol{p}$ 产生的样本并得到可能结果的计数 $\boldsymbol{F} = (F_1, F_2, \cdots, F_I)$,所得的预测分布可以刻画出 α 的另一个作用。α 值定义了 $\boldsymbol{p}$ 要产生的样本数,使得数据在预测时对先验占主导地位。

习题 23.3[小] Dirichlet 分布满足一个很好的可叠加性。假设一个有偏倚的六面骰子,有两面红色,4 面蓝色。这个骰子被抛掷 N 次,两个贝叶斯学者查看试验结果,以便推断出骰子的偏倚并做出预测。一个贝叶斯学者只能看到红色/蓝色结果,并且他推断出由 2 - 分量组成的概率向量(p_R, p_B)。另一个贝叶斯学者得到了所有完整结果:他可以看见六面中的哪一个面会出现,并可推断出一个 6 - 分量概率向量($p_1, p_2, p_3, p_4, p_5, p_6$),其中 $p_R = p_1 + p_2$ 和 $p_B = p_3 + p_4 + p_5 + p_6$。假设第二个贝叶斯学者为 $(p_1, p_2, p_3, p_4, p_5, p_6)$ 指定了一个带有超参数 $(u_1, u_2, u_3, u_4, u_5, u_6)$ 的 Dirichlet 分布,试证明,要使第一个贝叶斯学者的推断与第二个贝叶斯学者一致,第一个人的先验应

该是具有超参数$((u_1+u_2),(u_3+u_4+u_5+u_6))$的 Dirichlet 分布。

提示:计算以下积分是一种蛮力方法。

$$P(p_R,p_B)=\int \mathrm{d}^6\boldsymbol{p}P(\boldsymbol{p}\mid u)\delta(p_R-(p_1+p_2))\delta(p_B-(p_3+p_4+p_5+p_6))$$

一种更简便的方法是,计算已知任意数据$(F_1,F_2,F_3,F_4,F_5,F_6)$时的预测分布,同时找出两个预测分布可以匹配所有数据的条件。

一个概率向量$\boldsymbol{p}$的熵分布有时用于"最大熵"图像重构情形。

$$P(\boldsymbol{p}\mid\alpha,\boldsymbol{m})=\frac{1}{Z(\alpha,\boldsymbol{m})}\exp[-\alpha D_{KL}(\boldsymbol{p}\parallel\boldsymbol{m})]\delta(\sum_i p_i-1) \tag{23.35}$$

其中度量$\boldsymbol{m}$是一个正向量,且$D_{KL}(\boldsymbol{p}\parallel\boldsymbol{m})=\sum_i p_i\log p_i/m_i$。

进一步的阅读

关于 Dirichlet 分布的有趣资料可参看参考文献(MacKay 和 Peto,1995 年)。

23.6 进一步的习题

习题 23.4[2] 从未知参数s和c的伽马分布$P(x\mid s,c)=\Gamma(x;s,c)$中抽取$N$个数据点$\{x_n\}$。请问最大似然参数$s$和$c$是什么?

第 24 章

精确边缘化

如何才能避免对所有假设进行完全枚举所带来的指数代价呢？在我们着手研究近似方法之前，先探讨精确边缘化的两种方法：第一是通过积分在连续变量（有时称为麻烦参数）上进行边缘化；第二是通过消息传递在离散变量上求和。

在连续参数上进行精确边缘化是那些对定积分很熟练的人所乐意做的事。这一章利用伽马分布；正如前一章所解释的那样，伽马分布很像高斯分布，只是高斯分布是从 $-\infty$ 到 ∞，而伽马分布是从 0 到 ∞。

24.1 推断高斯分布的均值和方差

再次讨论参数是均值 μ 和标准差 σ 的一维高斯分布

$$P(x|\mu,\sigma)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)\equiv\text{Normal}(x;\mu,\sigma^2) \tag{24.1}$$

在推断这些参数时，必须指定它们的先验分布。这些先验让我们有机会给出关于 μ 和 σ 的特殊知识（例如，从独立实验或理论分析中所得的信息）。如果没有这样的知识，则可以构造一个适当的先验，以表明假设对它们一无所知。在第 21.2 节中，假设了在所画参数范围内的一个均匀先验。如果希望能够进行精确边缘化，考虑共轭先验也许有用；这些先验的泛函形式自然地和似然结合在一起，因而推理可以有方便的形式。

μ 和 σ 的共轭先验

均值 μ 的共轭先验是高斯分布，引入两个“超参数”μ_0 和 σ_μ 对 μ 的先验参数化，并写出 $P(\mu|\mu_0,\sigma_\mu)=\text{Normal}(\mu;\mu_0,\sigma_\mu)$。在 $\mu_0=0,\sigma_\mu\to\infty$ 的极限情况，得到的是一个位置参数的无信息价值的先验，即平坦先验。它是无信息价值的，因为分布在自然的重参量化 $\mu'=\mu+c$ 下是不变的。先验 $P(\mu)=$ 常数也不是一个合适的先验，即不是可归一化的。

标准差 σ 的共轭先验是伽马分布，有两个参数 b_β 和 c_β。定义逆方差（精度参数）$\beta=1/\sigma^2$ 的先验密度会带来很多方便。

$$P(\beta)=\Gamma(\beta;b_\beta,c_\beta)=\frac{1}{\Gamma(c_\beta)}\frac{\beta^{c_\beta-1}}{b_\beta^{c_\beta}}\exp\left(-\frac{\beta}{b_\beta}\right),\quad 0\leqslant\beta<\infty \tag{24.2}$$

这是一个简单的尖峰分布，均值为 $b_\beta c_\beta$，方差为 $b_\beta^2 c_\beta$。在 $b_\beta c_\beta = 1$，$c_\beta \to 0$ 的极限情况下，得到尺度参数 $1/\sigma$ 的无信息价值的先验分布。它是“无信息价值的先验分布”，因为它在重参量化 $\sigma' = c\sigma$ 下是不变的。如果考察 $\ln \sigma$ 或 $\ln \beta$ 上所得的密度，$1/\sigma$ 的先验不会显得那么怪模怪样，而是平坦的。[①] 这个先验表达了对 σ 的一无所知：“它可以是 10，也可以是 1，也可以是 0.1，……”。通常，σ 之类的尺度变量最好用其对数来表示。再次声明，无信息价值的 $1/\sigma$ 先验是非正常的。

在以下例子中，将使用 μ 和 σ 的非正常、无信息价值的先验。有些圈子中的人讨厌使用非正常的先验，所以我给自己一种借口：这样做是为了可读性；如果采用合适的先验，仍然可以完成计算，但是大量额外参数会使得要点变模糊。

最大似然与边缘化：σ_N 和 σ_{N-1}

大家都比较熟悉从 N 个样本点中推断高斯分布的均值和标准差，尽管并非每个人都理解计算器上 σ_N 和 σ_{N-1} 两个按键之间的差别。让我们重新给出公式，然后进行推导。

已知数据 $D = \{x_n\}_{n=1}^N$，μ 的“估计器”是

$$\bar{x} \equiv \sum_{n=1}^{N} x_n / N \tag{24.3}$$

σ 的两个估计器是

$$\sigma_N \equiv \sqrt{\frac{\sum_{n=1}^{N} (x_n - \bar{x})^2}{N}} \text{ 和 } \sigma_{N-1} \equiv \sqrt{\frac{\sum_{n=1}^{N} (x_n - \bar{x})^2}{N-1}} \tag{24.4}$$

在统计学中有两个原则性规范：抽样理论和贝叶斯推理。在抽样理论（也称为“频率主义”或正统统计）中，对于感兴趣的量设计估计器，然后通过使用一些判据来度量采样特性，并据此对这些估计器做出选择；没有明确的原理来确定该采用何种判据来度量一个估计器的性能，而且对大部分判据也没有构造最优估计器的系统化方法。相反的，在贝叶斯推理中，一旦对模型和数据做了所有清楚的假设，就能机械地进行推理。无论要解决什么问题，概率论的规则都给出唯一答案，能够一致地考虑所有已知信息。人为设计的估计器和置信区间在贝叶斯推理中不起作用；人的输入只会进入设计假设空间的重要任务（即确定模型及其所有概率分布），并给出在这个空间如何做实现推理的计算。问题的答案就是人们感兴趣的量的概率分布。当选择一个简单的假设空间并使用贝叶斯推理方法时，往往会发现抽样理论的估计器本身就是这些后验分布的峰值或均值。

在抽样理论中，使用以上所述估计器存在以下原因。$\bar{x}$ 是 μ 的无偏估计器，它在

① 注意：当用 σ 的一一对应函数把变量从 σ 变到 $l(\sigma)$ 后，概率密度函数从 $P_\sigma(\sigma)$ 变换到 $P_l(l) = P_\sigma(\sigma)\left|\frac{\partial \sigma}{\partial l}\right|$。这里的雅可比矩阵是 $\left|\frac{\partial \sigma}{\partial \ln \sigma}\right| = \sigma$。

μ 的所有可能的无偏估计器中有最小方差。(通过对假想试验的总体求平均来计算方差,在试验中假设数据样本来自一个未知高斯分布。)估计器 $(\bar{x},\sigma_N)$ 是 (μ,σ) 的最大似然估计器。然而,估计器 σ_N 是有偏倚的:已知 σ,对许多想象的试验求平均,σ_N 的期望不是 σ。



习题 24.1[21] 对于为什么 σ_N 是有偏倚估计器,给出一个直观解释。

这个偏倚启发人们在抽样理论中使用 σ_{N-1};可以证明这是一个无偏倚估计器。或者,更确切地说,σ^2_{N-1} 是 σ^2 的无偏倚估计器。

现在假设 μ 和 σ 的无信息价值的先验分布,来看这个问题的贝叶斯推理。因此,重点不在于先验,而在于(a) 似然函数,(b) 边缘化的概念。μ 和 σ 的联合后验概率与似然函数成正比,如图 24.1(a)中的等势图所示。对数似然是

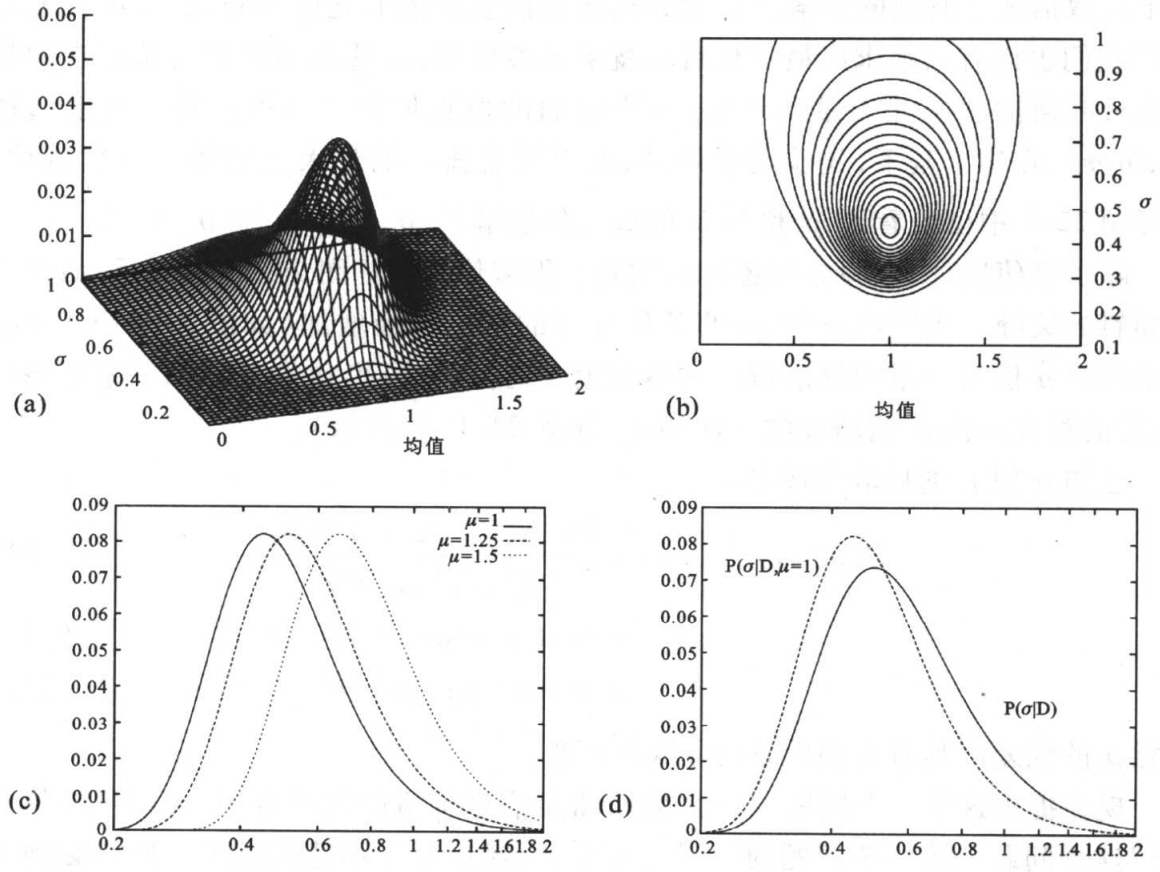


图 24.1 一个高斯分布参数的似然函数,这是对图 21.5 的重复。(a,b)是对数似然以 μ 和 σ 的函数表示的曲面图和等势图。$N=5$ 个点的数据集合,其均值 $\bar{x}=1.0$ 和 $S^2=\sum(x-\bar{x})^2=1.0$。注意:$\sigma$ 的最大值是倾斜的。标准差的两个估计器的值为 $\sigma_N=0.45$ 和 $\sigma_{N-1}=0.50$。(c) 在各种固定 μ 值下的 σ 后验概率(表示为 $\ln\sigma$ 上的密度)。(d) 假设 μ 具有平坦先验分布时 σ 的后验概率 $P(\sigma|D)$,这里先验分布是将(a)中的概率质量投影到 σ 轴上得出的。$P(\sigma|D)$ 的最大点位于 σ_{N-1},而 $P(\sigma|D,\mu=\bar{x})$ 的最大点位于 σ_N。(两个概率都显示为 $\ln\sigma$ 上的密度。)

$$\ln P(\{x_n\}_{n=1}^N \mid \mu,\sigma) = -N\ln(\sqrt{2\pi}\sigma) - \sum_n (x_n-\mu)^2/(2\sigma^2) \tag{24.5}$$

$$= -N\ln(\sqrt{2\pi}\sigma) - [N(\mu-\bar{x})^2 + S]/(2\sigma^2) \tag{24.6}$$

其中 $S \equiv \sum_n (x_n-\bar{x})^2$。已知高斯模型，似然函数可以用数据 $\bar{x}$ 和 S 的两个函数表示，所以这两个量叫做“充分统计”。利用非正常先验，μ 和 σ 的后验概率是

$$P(\mu,\sigma \mid \{x_n\}_{n=1}^N) = \frac{P(\{x_n\}_{n=1}^N \mid \mu,\sigma)P(\mu,\sigma)}{P(\{x_n\}_{n=1}^N)} \tag{24.7}$$

$$= \frac{\dfrac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\dfrac{N(\mu-\bar{x})^2+S}{2\sigma^2}\right)\dfrac{1}{\sigma_\mu}\dfrac{1}{\sigma}}{P(\{x_n\}_{n=1}^N)} \tag{24.8}$$

这个函数描述了问题的答案：“已知数据和无信息价值的先验分布，μ 和 σ 可能是多少？”我们也许有兴趣求出最大化后验概率的参数值，尽管必须强调后验概率的最大点在贝叶斯推理中并无重要地位，因为它们的位置依赖于基的选择。这里选择基 $(\mu,\ln\sigma)$，在这个基下，先验是平坦的，所以后验概率的最大点和最大似然点重合。在习题 22.4 中已经看到，μ 和 $\ln\sigma$ 的最大似然解是 $\{\mu,\sigma\}_{\mathrm{ML}} = \{\bar{x},\sigma_N = \sqrt{S/N}\}$。

除了峰值以外，后验分布还可提供其他很多信息。如图 24.1(a)所示，似然有一个偏斜的尖峰。当增加 σ 时，μ 的条件分布的宽度也会增加（如图 22.1(b)所示）。如果固定 μ 值为一系列逐渐偏离样本均值 $\bar{x}$ 的值，就可得到关于 σ 的一系列条件分布，它的最大点向 σ 值增加的方向移动（如图 24.1(c)所示）。

已知 σ 时 μ 的后验概率是

$$P(\mu \mid \{x_n\}_{n=1}^N,\sigma) = \frac{P(\{x_n\}_{n=1}^N \mid \mu,\sigma)P(\mu)}{P(\{x_n\}_{n=1}^N \mid \sigma)} \tag{24.9}$$

$$\propto \exp(-N(\mu-\bar{x})^2/(2\sigma^2)) \tag{24.10}$$

$$= \mathrm{Normal}(\mu;\bar{x},\sigma^2/N) \tag{24.11}$$

注意 μ 的误差限具有人们熟悉的 $\sigma/\sqrt{N}$ 尺度。

现在提出这样一个问题：“已知数据和无信息价值的先验分布，σ 的可能值是多少？”这个问题和第一个问题的不同之处在于，现在对 μ 不感兴趣了。所以必须将这个参数边缘化。σ 的后验概率是

$$P(\sigma \mid \{x_n\}_{n=1}^N) = \frac{P(\{x_n\}_{n=1}^N \mid \sigma)P(\sigma)}{P(\{x_n\}_{n=1}^N)} \tag{24.12}$$

数据相关项 $P(\{x_n\}_{n=1}^N \mid \sigma)$ 在前面作为归一化常数出现在式(24.9)中；这个量的一个名称是“显著性”，或 σ 的边缘似然。将 μ 积分掉，就可以得到 σ 的显著性；假设无信息价值的先验分布 $P(\mu)$ = 常数；称这个常数为 $1/\sigma_\mu$，这样就可以将先验看做宽度为 σ_μ 的至高先验。由高斯积分 $P(\{x_n\}_{n=1}^N \mid \sigma) = \int P(\{x_n\}_{n=1}^N \mid \mu,\sigma)P(\mu)\,\mathrm{d}\mu$ 得

$$\ln P(\{x_n\}_{n=1}^N|\sigma) = -N\ln(\sqrt{2\pi}\sigma) - \frac{S}{2\sigma^2} + \ln\frac{\sqrt{2\pi}\sigma/\sqrt{N}}{\sigma_\mu} \tag{24.13}$$

前两项是最佳拟合对数似然(即满足 $\mu=\bar{x}$ 的对数似然)。最后一项是 Occam 因子的对数,它对小的 σ 值进行惩罚。(将在第 28 章进一步讨论 Occam 因子。)当我们求对数显著性关于 $\ln\sigma$ 的导数,以发现最可能的 σ 值,额外的体积因子($\sigma/\sqrt{N}$)把最大值点从 σ_N 移到

$$\sigma_{N-1} = \sqrt{S/(N-1)} \tag{24.14}$$

从直观说,分母($N-1$)是包含在量 $S=\sum_n (x_n-\bar{x})^2$ 中有噪声的测量次数。和式中有 N 个残差平方,但是只有($N-1$)个有效的噪声测量;这是因为从数据中确定一个参数 μ 时,会在不可避免的超拟合中用掉一维噪声。在传统的统计学术语中,σ 的最佳贝叶斯猜测把 χ^2(定义为 $\chi^2 \equiv \sum_n (x_n-\hat{\mu})^2/\hat{\sigma}^2$ 的偏差度量)设定为自由度 $N-1$。

图 24.1(d)显示了 σ 的后验概率,它正比于边缘似然。可将其与固定 μ 为其最可能值 $\bar{x}=1$ 时的后验概率进行比较,如图 24.1(c)和图 24.1(d)所示。

我们想进行的最后推理可能是"已知数据,μ 是什么?"

▷ **习题 24.2**[β] 在 σ 上进行边缘化,并得到 μ 的后验边缘分布,它是学生 $-t$ 分布。

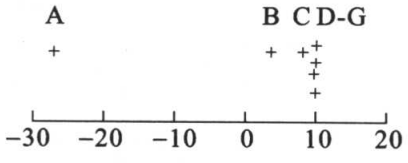
$$P(\mu|D) \propto 1/(N(\mu-\bar{x})^2+S)^{N/2} \tag{24.15}$$

进一步的阅读

关于精确边缘化的经典之作是 Bretthorst(1988 年)关于贝叶斯频谱分析和参数估计的书。

24.2 习 题

▷ **习题 24.3**[β] [这个习题需要极强的积分能力。]给出习题 22.15 的贝叶斯解,其中,7 位能力各异的科学家以各自的噪声水平 σ_n 测量 μ,我们所感兴趣的是对 μ 进行推理。设每个 σ_n 的先验是一个宽分布先验,例如是参数 $(s,c)=(10,0.1)$ 的伽马分布。求 μ 的后验分布。把它画出来,并对不同的数据集合探讨其特性,如对已知的这个和数据集 $\{x_n\}=\{13.01,7.39\}$。



[提示:首先求出已知 μ 和 x_n 时 σ_n 的后验分布 $P(\sigma_n|x_n,\mu)$。注意这个推理的归一化常数是 $P(x_n|\mu)$。在 σ_n 上进行边缘化,求出归一化常数,然后第二次利用贝叶斯定理求出 $P(\mu|\{x_n\})$。]

24.3 习题解答

习题 24.1 的解答 1. 数据点以均方差 σ^2 分布在真实均值的周围。2. 样本均值不太可能精确地等于真实均值。3. 样本均值是极小化数据点偏离 μ 的偏差的平方和的 μ 值。任何其他的 μ 值(特别是 μ 的真值)都有比 $\mu=\bar{x}$ 更大的偏差平方和。

所以,关于样本均值的期望均方差必然小于真正均值的均方差 σ^2。

第 25 章

网格中的精确边缘化

本章将讨论概率建模中所用的几种精确方法。以线性纠错码的译码任务为例进行讨论。大家将看到采用*消息传递算法*会使推理更加有效，该算法利用问题的图形结构，避免了不必要的重复计算（见第 16 章）。

25.1 译码问题

从线性 (N,K) 码 $\mathcal{C}$ 中选取码字 $\mathbf{t}$，并在噪声信道上传输；接收信号是 $\mathbf{y}$。在本章中，假设信道是无记忆信道，如高斯信道。已知一个假设的信道模型 $P(\mathbf{y}|\mathbf{t})$，存在两个译码问题。

码字译码问题的任务是对已知的接收信号推断哪个码字 $\mathbf{t}$ 被传输。

逐比特译码问题的任务是推断每个传输比特 t_n 是 1 而非 0 的可能性有多大。

以 (7,4) 汉明码作为一个具体的例子。在第 1 章中，假设信道为二进制对称信道，并对这种码的码字译码问题进行了讨论。我们没有讨论逐比特译码问题，也没有讨论如何处理更一般的信道模型，如高斯信道。

码字译码问题的求解

根据贝叶斯定理，码字 $\mathbf{t}$ 的后验概率为

$$P(\mathbf{t}|\mathbf{y}) = \frac{P(\mathbf{y}|\mathbf{t})P(\mathbf{t})}{P(\mathbf{y})} \tag{25.1}$$

似然函数 分子中的第一个因子 $P(\mathbf{y}|\mathbf{t})$ 是码字的似然，对于任何无记忆信道，它是可分函数

$$P(\mathbf{y}|\mathbf{t}) = \prod_{n=1}^{N} P(y_n|t_n) \tag{25.2}$$

例如，如果信道是一个高斯信道，所传输的是 $\pm x$ 和标准差 σ 的加性噪声，那么在 $t_n = 0, 1$ 这两种情况，接收信号 y_n 的概率密度为

$$P(y_n|t_n=1) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y_n-x)^2}{2\sigma^2}\right) \tag{25.3}$$

$$P(y_n|t_n=0) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(y_n+x)^2}{2\sigma^2}\right) \tag{25.4}$$

从译码的观点来看，起作用的是似然比，它对高斯信道这种情况是

$$\frac{P(y_n \mid t_n = 1)}{P(y_n \mid t_n = 0)} = \exp\left(\frac{2xy_n}{\sigma^2}\right) \tag{25.5}$$



习题 25.1[2] 证明从译码的角度来看，一个高斯信道等价于一个时变二进制对称信道，并具有取决于 n 的已知噪声水平 f_n。

先验 分子中的第二个因子是码字的先验概率 $P(\mathbf{t})$，一般假设为所有有效码字上的均匀分布。

式(25.1)中的分母是归一化常数

$$P(\mathbf{y}) = \sum_{\mathbf{t}} P(\mathbf{y} \mid \mathbf{t}) P(\mathbf{t}) \tag{25.6}$$

对码字译码问题的完全解是所有码字的列表及其概率，如式(25.1)所示。因为线性码中的码字数 2^K 往往是很大的，而我们也不想知道所有码字的具体概率，因此一般把注意力局限在简化的码字译码问题上。

MAP 码字译码问题的任务是，对于已知的接收信号，识别出最可能的码字 $\mathbf{t}$。

如果码字的先验概率是均匀的，那么这个任务与最大似然译码问题相同，即求出将 $P(\mathbf{y} \mid \mathbf{t})$ 最大化的码字。

例 在第 1 章中，对于(7,4)汉明码和一个二进制对称信道，我们讨论了由接收信号的校正子推导出最可能码字的方法，因而解决了那种情况下的 MAP 码字译码问题。我们想得到更一般的解。

可以通过在所有码字中搜寻最大化 $P(\mathbf{y} \mid \mathbf{t})P(\mathbf{t})$ 的那个码字，在指数时间(2^K 量级)内求解 MAP 码字译码问题。但是我们感兴趣的是比其更有效的方法。在第25.3节中，将讨论一种称为最小和算法的精确方法，它也许能更有效地解决码字译码问题；有效程度取决于码的性质。

需要强调的是，对于一般的线性码，已知 MAP 码字译码问题是 NP - 完全的(用外行的话说就是 MAP 码字译码问题的复杂性随分组长度呈指数级增长，除非计算机科学发生了一场革命)。所以把注意力集中在 MAP 译码问题上并未使问题简单多少；它只是让答案更易于记录。

逐比特译码问题的求解

正式地说，逐比特译码问题的精确解是通过对其他比特进行边缘化，由式(25.1)得出的。

$$P(t_n \mid \mathbf{y}) = \sum_{\{t_{n'}: n' \neq n\}} P(\mathbf{t} \mid \mathbf{y}) \tag{25.7}$$

也可以借助真值函数$\mathbb{1}[S]$来写出这个边缘概率。真值函数$\mathbb{1}[S]$：当属性 S 为真时其值为 1，否则为 0。

$$P(t_n = 1 \mid \mathbf{y}) = \sum_{\mathbf{t}} P(\mathbf{t} \mid \mathbf{y}) \, \mathbb{1}[t_n = 1] \tag{25.8}$$

$$P(t_n = 0 \mid \boldsymbol{y}) = \sum_{\boldsymbol{t}} P(\boldsymbol{t} \mid \boldsymbol{y}) \mathbb{1}[t_n = 0] \tag{25.9}$$

通过显式地对所有码字 $\boldsymbol{t}$ 求和来计算这些边缘概率，需要花费指数级的时间。但是对于有些码字，利用前向 - 后向算法解决逐比特译码问题效率会高很多。马上将要描述这种算法，它是和 - 积算法的一个特例。最小和算法与和 - 积算法都有广泛的重要性，它们在很多领域都得到多次开发应用。

25.2 码与网格

在第 1 章和第 11 章中，利用生成式矩阵及其奇偶校验矩阵表示线性 (N,K) 码。在系统化分组码的情况，对于大小为 N 的每个分组，前 K 个传输比特是源比特，余下的 $M = N - K$ 比特是奇偶校验比特。这意味着码的生成式矩阵可以写成

$$\boldsymbol{G}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{I}_K \\ \boldsymbol{P} \end{bmatrix} \tag{25.10}$$

奇偶校验矩阵可以写成

$$\boldsymbol{H} = [\boldsymbol{P} \quad \boldsymbol{I}_M] \tag{25.11}$$

其中 $\boldsymbol{P}$ 是 $M \times K$ 矩阵。

在本节，将要研究线性码的另一种表示，称作网格。这些网格所表示的码通常不是系统化的码，但是如果需要，通过对分组中的比特进行重新排序，就可以把它们映射成系统化的码。

网格的定义

我们定义的范围很窄。如果要更全面地了解网格，读者应阅读参考文献 Kschischang 和 Sorokine(1995 年)。

网格是一个由结点(也称为状态或顶点)和边所组成的图。将结点分组为垂直片段，称为时间；时间的排序方式是，每条边把一个时间中的一个结点连接到相邻时间的一个结点上。每条边用符号来标记。最左边和最右边的状态只包含一个结点。除了这两个端点结点以外，网格中的所有结点至少有一条边向左连，至少有一条边向右连。

一个有 $N+1$ 个时间的网格按如下方式定义了一个分组长度为 N 的码：通过选取一条从左到右穿越网格的路径，并读出路径所穿过的边上的符号，就得到了一个码字。每个穿过网格的有效路径都定义了一个码字。把最左边的时间编号为“时间 0”，最右边的时间为“时间 N”。把最左边的状态编号为“状态 0”，最右边的状态为“状态 I”，其中 I 是网格中的状态(顶点)总数。当我们从时间 $n-1$ 移动到时间 n 时，就产生了码字的第 n 个比特。

对于已知的时间，网格的宽度是那个时间的结点数。网格的最大宽度就可以顾

名思义了。

如果网格所定义的码是线性码，就将它叫做线性网格。从现在开始，只考虑线性网格，因为非线性网格是更为复杂的东西。为了简单起见，只讨论二进制网格，即网格的边只用 0 和 1 来标记。很容易把下面的方法推广到 q 元网格。

图 25.1(a ~ c)显示了 $(N,K)=(3,1)$ 的重复码 R_3 所对应的网格；$(N,K)=(3,2)$ 的奇偶码 P_3 所对应的网格；及(7,4)汉明码所对应的网格。

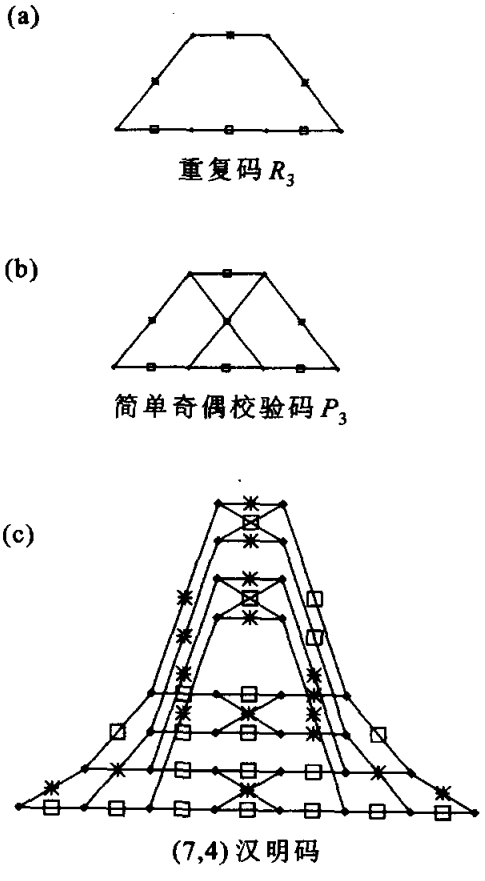


图 25.1 网格的例子，网格中的每条边被标记为 0(用方块表示)或 1(用叉子表示)。

▷ **习题 25.2**[2] 验证表 1.14 中所列的 16 个码字是由图 25.1(c)所示网格生成的。

关于线性网格的观察

对于任何线性码，最小网格是有最少结点数的网格。在一个最小网格中，每个结点最多只有两条进入它的边，最多只有两条离开它的边。在某个时间上的所有结点都有相同的左度数和相同的右度数。宽度总是 2 的一个幂。

对于线性 (N,K) 码，最小网格的宽度不会大于 2^K，因为每个结点至少有一个有效码字通过，而总共只有 2^K 个码字。此外，如果定义 $M=N-K$，最小网格宽度处处小于 2^M。将在第 25.4 节对此做出证明。

注意图 25.1 中的线性网格都是最小网格，K 是网格从左到右或从右到左时二进

制分支点所遇到的时间数。

将在第 25.4 节进一步讨论网格的构造。但是,现在已经有足够的知识来讨论译码问题了。

25.3 在网格上求解译码问题

可以把线性码的网格看成是对产生码字的概率过程的因果描述,其中时间是从左到右增加。每次遇到一个分叉,一个随机信源(通信的信息比特源)决定走哪条路。

在接收端,收到有噪版的边标记序列,并希望推断出所选的是哪条路径,或更确切地说:(a) 为了解决码字译码问题,要识别出最可能的路径;(b) 为了解决逐比特译码问题,要求出在时间点 n 所发送的符号是一个 0 或是一个 1 的概率。

例 25.3 考虑如图 25.1(c)所示的汉明(7,4)网格的单次传输情况。

设归一化的似然为(0.1,0.4,0.9,0.1,0.1,0.1,0.3),即似然比是

$$\frac{P(y_1 \mid x_1=1)}{P(y_1 \mid x_1=0)}=\frac{0.1}{0.9}, \frac{P(y_2 \mid x_2=1)}{P(y_2 \mid x_2=0)}=\frac{0.4}{0.6}, \text{等等} \tag{25.12}$$

如何对这些接收信号进行译码?

1. 如果以 0.5 为似然阈值,把信号转变成二进制接收向量,则有 $\boldsymbol{r}=(0,0,1,0,0,0,0)$,利用二进制对称信道的译码器(第 1 章),该向量被译码为 $\hat{\boldsymbol{t}}=(0,0,0,0,0,0,0)$。

这不是最优译码过程。最优推理总要通过贝叶斯定理获得。

2. 可以通过直接枚举所有 16 个码字,来求出码字的后验概率。后验概率如图 25.2 所示。当然,我们并非真的对这种蛮力解感兴趣,本章的目的是理解能用小于 2^K 单位的计算机时间获取同一信息的算法。

查看后验概率,注意到最可能的码字实际上是字符串 $\boldsymbol{t}=0110001$。它的可能性要比用阈值方法得到的答案 0000000 大两倍。

利用图 25.2 所示的后验概率,还可以计算每个比特的后验边缘分布。结果如图 25.3所示。注意:可以相当肯定地将第 1、4、5 和 6 比特推断为 0。第 2、3 和 7 比特的后验概率力度就不那么大了。 □

在上面的例子中,MAP 码字与逐比特译码是一致的。在逐比特译码时,利用后验边缘分布选择每个比特的最可能状态。但它们有时也可能不一致,下面的习题就说明这一点。



习题 25.4[2] 在归一化似然为(0.2,0.2,0.9,0.2,0.2,0.2,0.2)的情况,求最可能的码字。求出或估计出 7 比特中的每个比特的边缘后验概率,并进行逐比特

t	似然	后验概率
0000000	0.027 556 2	0.25
0001011	0.000 145 8	0.001 3
0010111	0.001 312 2	0.012
0011100	0.003 061 8	0.027
0100110	0.000 226 8	0.002 0
0101101	0.000 097 2	0.000 9
0110001	0.070 858 8	0.63
0111010	0.002 041 2	0.018
1000101	0.000 145 8	0.001 3
1001110	0.000 004 2	0.000 0
1010010	0.003 061 8	0.027
1011001	0.001 312 2	0.012
1100011	0.000 097 2	0.000 9
1101000	0.000 226 8	0.002 0
1110100	0.002 041 2	0.018
1111111	0.000 010 8	0.000 1

图 25.2 当接收向量 y 的归一化似然为(0.1,0.4,0.9,0.1,0.1,0.1,0.3)时，16 个码字的后验概率。

n	似然		后验边缘	
	$P(y_n\|t_n=1)$	$P(y_n\|t_n=0)$	$P(t_n=1\|y)$	$P(t_n=0\|y)$
1	0.1	0.9	0.061	0.939
2	0.4	0.6	0.674	0.326
3	0.9	0.1	0.746	0.254
4	0.1	0.9	0.061	0.939
5	0.1	0.9	0.061	0.939
6	0.1	0.9	0.061	0.939
7	0.3	0.7	0.659	0.341

图 25.3 在图 25.2 中的后验分布下，7 个比特的边缘后验概率。

译码。

[提示：集中考虑有最大概率的几个码字。]

现在讨论如何利用码的网格上所传递的消息来求解译码问题。

最小和算法

可以利用第 16.3 节介绍的最小和算法来求解 MAP 码字译码问题。码的每个码字都对应于穿过网格的一条路径。正如路程的代价是其组成步骤代价之和，一个码字的对数似然是逐比特对数似然之和。根据惯例，翻转对数似然的符号(希望对其最大化)，从代价这一角度考虑，这个代价是我们想要最小化的量。

为每一条边都关联一个代价——$\log P(y_n|t_n)$，其中 t_n 是和那条边相关的传输比特，而 y_n 是接收符号。第 16.3 节的最小和算法就可以在一组计算机操作中识别出最可能的码字，计算机操作数等于网格中的边数。这个算法也叫做 Viterbi 算法（Viterbi，1967 年）。

和 - 积算法

为了解决逐比特译码问题，可以对最小和算法做一些小的修改，使得传递在网格中的消息定义“到当前点为止的数据的概率”，而非“到这一点的最佳路径的代价”。通过似然本身 $P(y_n|t_n)$ 来代替边上的代价——$\log P(y_n|t_n)$。分别用“和”与“积”运算代替“最小和算法”中的“最小”与“和”运算。

用 i 标记结点/状态，$i=0$ 是初始状态的标记，$\mathscr{P}(i)$ 表示状态 i 的父状态集合，w_{ij} 是与从结点 j 到结点 i 的边相关联的似然。定义前向传递消息 α_i 为

$$\begin{aligned}\alpha_0 &= 1\\ \alpha_i &= \sum_{j\in\mathscr{P}(i)} w_{ij}\alpha_j\end{aligned} \tag{25.13}$$

这些消息可以从左向右依次地计算。

▷ **习题 25.5**[⌀] 证明对于时间坐标为 n 的结点 i，α_i 正比于码字路径通过结点 i 以及前 n 个接收符号为 $y_1,\cdots,y_n$ 的联合概率。

在网格末端结点处计算的消息 α_I 正比于数据的边缘概率。

▷ **习题 25.6**[⌀] 比例常数是多少？[答案：2^K]

以同样方式定义第二组后向传递消息 β_i。设结点 I 为末端结点。

$$\begin{aligned}\beta_I &= 1\\ \beta_j &= \sum_{i:j\in\mathscr{P}(i)} w_{ij}\beta_i\end{aligned} \tag{25.14}$$

这些消息可以从右至左反向地依次计算出来。

▷ **习题 25.7**[⌀] 证明对于时间坐标是 n 的结点 i，β_i 正比于给定码字路径通过结点 i 时顺序接收符号是 $y_{n+1},\cdots,y_N$ 的条件概率。

最后，为了求出第 n 个比特是 1 或 0 的概率，对前向和后向消息的乘积做两次加法。以 i 标记时间 n 上的结点，j 标记时间 $n-1$ 上的结点，并设 t_{ij} 为网格中与从结点 j 到结点 i 的边相关的 t_n 值。对于 $t=0/1$ 的每个值，计算

$$r_n^{(t)} = \sum_{i,j:j\in\mathscr{P}(i),t_{ij}=t} \alpha_j w_{ij}\beta_i \tag{25.15}$$

则 t_n 为 $t=0/1$ 的后验概率是

$$P(t_n=t|\mathbf{y}) = \frac{1}{Z}r_n^{(t)} \tag{25.16}$$

其中，归一化常数 $Z=r_n^{(0)}+r_n^{(1)}$ 应该等于前面所计算的最后前向消息 α_I。

习题 25.8[⌀] 验证上面的和 - 积算法确实计算出了 $P(t_n=t|\mathbf{y})$。

这里所述和 - 积算法的其他名称包括“前向 - 后向算法”、“BCJR 算法”和“置信

传播”。

表 25.4 P_3 中一个码字的逐比特似然

n	$P(y_n \mid t_n)$	
	$t_n = 0$	$t_n = 1$
1	$\frac{1}{4}$	$\frac{1}{2}$
2	$\frac{1}{2}$	$\frac{1}{4}$
3	$\frac{1}{8}$	$\frac{1}{2}$

▷ **习题 25.9**[2] 传送简单奇偶码 P_3 的一个码字，接收信号 $\boldsymbol{y}$ 具有如表 25.4 所示的关联似然。利用网格（如图 25.1 所示）的最小和算法与和 - 积算法，求解 MAP 码字译码问题和逐比特译码问题。通过列举所有码字（000，011，110，101）来验证你的答案。

［提示：用以 2 为底的对数，手工计算最小和。在用手工完成和 - 积算法时，用 3 种颜色的笔也许会有所帮助，一种颜色用于 α，一种颜色用于 w，一种颜色用于 β。］

25.4 关于网格的进一步讨论

现在讨论产生与码相对应的网格的几种不同方法。跳过这一节也没关系。

一个码字的间距是包含在码字第一个非零比特和最后一个非零比特之间（包括这两个比特）的比特集合。可以用如表 25.5 所示的二进制向量来表示一个码字的间距。

表 25.5 一些码字和它们的间距

码字	0000000	0001011	0100110	1100011	0101101
间距	0000000	0001111	0111110	1111111	0111111

如果生成式矩阵各行的间距在不同的列开始且在不同的列结束，那么生成式矩阵就具有*面向网格形式*。

如何从生成式矩阵产生网格

首先，通过类似于高斯消元法中的行运算，把生成式矩阵变成面向网格形式。例如，(7,4)汉明码可以由

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{25.17}$$

生成。但是,这个矩阵不具有面向网格形式——如第 1、3 和 4 行的间距都在同一列结束。通过从上一行中减去下一行,可以得到等价的生成式矩阵(即生成同样的码字集合的生成式矩阵)如下

$$\boldsymbol{G}=\begin{bmatrix}1&1&0&1&0&0&0\\0&1&0&0&1&1&0\\0&0&1&1&1&0&0\\0&0&0&1&0&1&1\end{bmatrix} \tag{25.18}$$

现在,可认为生成式矩阵的每一行都定义(N,K)码的一个$(N,1)$子码,此时也就是一个码有两个长度为 $N=7$ 的码字。对于第 1 行,码由两个码字 1101000 和 0000000 组成。第 2 行所定义的子码由 0100110 和 0000000 组成。很容易构造这些子码的最小网格,如图 25.6 的左列所示。

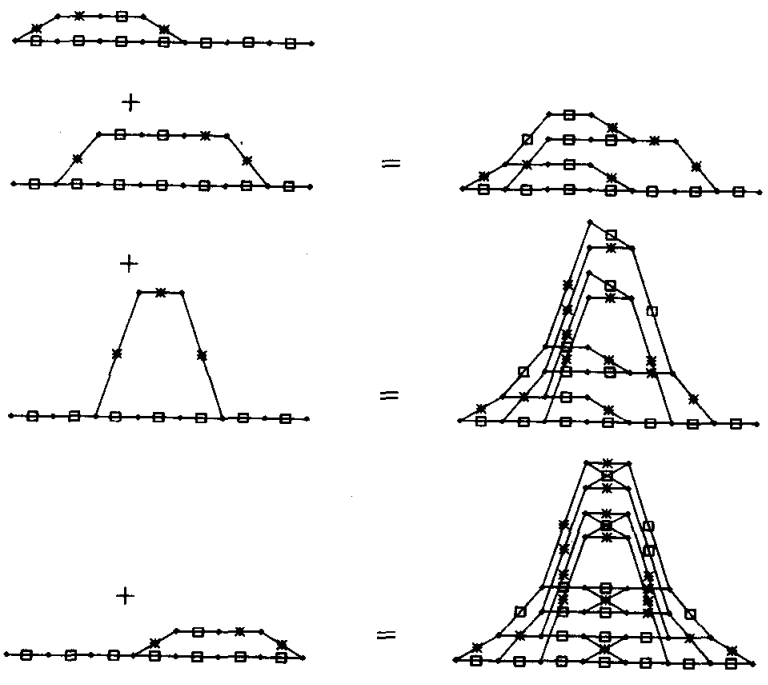


图 25.6 (7,4)汉明码的 4 个子码的网格(左列),及为构造(7,4)汉明码的网格而产生的网格序列(右列)。网格中的每条边都被标记为 0(用方块表示)或 1(用叉子表示)。

递增地构建网格,如图 25.6 所示。从生成式矩阵第 1 行产生的子码所对应的网格开始。然后每次加入一个子码。复制新子码间距之内的所有顶点。原先网格边上的符号保持不变;每当新的子码有一个 1 时,翻转网格第 2 部分的边上的符号,否则保持不变。

可以通过

$$\boldsymbol{G}=\begin{bmatrix}1&1&1&0&0&0&0\\0&1&1&1&1&0&0\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{bmatrix} \tag{25.19}$$

生成另一个(7,4)汉明码。由这个矩阵所生成的(7,4)汉明码和第 1 章以及上面系统化矩阵所生成的码只是比特排列次序不同。相应于这个排列的奇偶校验矩阵是

$$\boldsymbol{H}=\begin{bmatrix}1&0&1&0&1&0&1\\0&1&1&0&0&1&1\\0&0&0&1&1&1&1\end{bmatrix} \tag{25.20}$$

如图 25.7(a)所示为由式(25.19)给出的排列矩阵 $\boldsymbol{G}$ 得到的网格。注意这个网格中的结点数要小于前面图 25.1(c)中(7,4)汉明码的网格中的结点数。现在可以看出对码字中比特重新进行排序,有时会导致更小、更简单的网格。

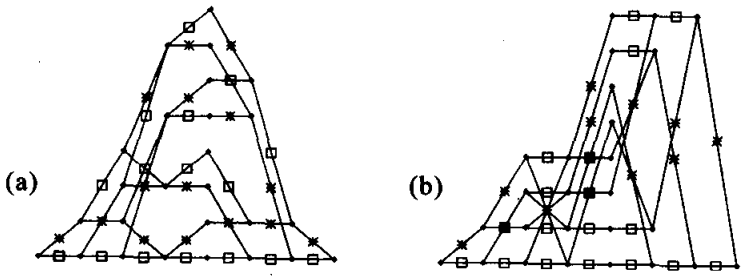


图 25.7 重新排列过的(7,4)汉明码的网格,(a) 通过图 25.6 的方法由生成式矩阵生成;(b) 通过下述方法由奇偶校验矩阵生成。网格中的每条边用 0(用方块表示)或 1(用叉子表示)来标记。

由奇偶校验矩阵所得的网格

看待网格的另一种方式是根据校正子。向量 $\boldsymbol{r}$ 的校正子定义为 $\boldsymbol{Hr}$,其中 $\boldsymbol{H}$ 是奇偶校验矩阵。只有在校正子为零的情况下,一个向量才是码字。在生成码字的时候,可以通过部分校正子(即 $\boldsymbol{H}$ 与到目前为止所生成码字比特的乘积)来描述当前状态。网格中的每个状态都是某一时刻的部分校正子。起始和终止状态都要满足零校正子的约束。状态中的每个结点代表了部分校正子的不同可能值。因为 $\boldsymbol{H}$ 是一个 $M\times N$ 矩阵,其中 $M=N-K$,校正子最多是 M 比特的向量,所以每个状态最多需要 2^M 个结点。可以从奇偶校验矩阵构造码的网格:从每个端点出发,生成可能的校正子序列的两棵树。这两棵树的交就定义了码的网格。

在由这样的构造所得的图中,可以让纵坐标代表校正子。于是,任何水平边就必须与一个 0 比特相关联(因为只有非零比特才改变校正子),而任何非水平边就与一个 1 比特相关联。(所以在这种表示中,不再需要对网格的边进行标记。)图 25.7(b)的网格对应于式(25.20)所给出的奇偶校验矩阵。

25.5 习题解答

表 25.8 习题 25.4 中码字的后验概率

t	似然	后验概率
0000000	0.026	0.300 6
0001011	0.000 41	0.004 7
0010111	0.003 7	0.042 3
0011100	0.015	0.169 1
0100110	0.000 41	0.004 7
0101101	0.000 10	0.001 2
0110001	0.015	0.169 1
0111010	0.003 7	0.042 3
1000101	0.000 41	0.004 7
1001110	0.000 10	0.001 2
1010010	0.015	0.169 1
1011001	0.003 7	0.042 3
1100011	0.000 10	0.001 2
1101000	0.000 41	0.004 7
1110100	0.003 7	0.042 3
1111111	0.000 058	0.000 7

习题 25.4 的解答 码字的后验概率如表 25.8 所示。最可能的码字是 0000000。所有 7 个比特的边缘后验概率如下。

n	似然		后验边缘	
	$P(y_n \mid t_n = 1)$	$P(y_n \mid t_n = 0)$	$P(t_n = 1 \mid \mathbf{y})$	$P(t_n = 0 \mid \mathbf{y})$
1	0.2	0.8	0.266	0.734
2	0.2	0.8	0.266	0.734
3	0.9	0.1	0.677	0.323
4	0.2	0.8	0.266	0.734
5	0.2	0.8	0.266	0.734
6	0.2	0.8	0.266	0.734
7	0.2	0.8	0.266	0.734

所以逐比特译码的结果是 0010000，它实际上并不是一个码字。

习题 25.9 的解答 MAP 码字是 101，它的似然是 1/8。和 - 积算法的归一化常数是 $Z = \alpha_7 = 3/16$。α_i 的中间值（从左向右）是 1/2，1/4，5/16，1/4；β_i 的中间值（从右向左）是 1/2，1/8，9/32，3/16。逐比特译码是 $P(t_1 = 1 \mid \boldsymbol{y}) = 3/4$；$P(t_1 = 1 \mid \boldsymbol{y}) = 1/4$；$P(t_1 = 1 \mid \boldsymbol{y}) = 5/6$。码字 000，011，110，101 的概率是 1/12，1/6，1/12，2/3。

第 26 章

图中的精确边缘化

现在通过更一般化的观点来看推理和边缘化的任务。在阅读本章之前，应该阅读第 16 章中的消息传递。

26.1 一般问题

假设将 N 个变量的集合 $\boldsymbol{x} \equiv \{x_n\}_{n=1}^N$ 的函数 P^* 定义为以下 M 个因式的乘积。

$$P^*(\boldsymbol{x}) = \prod_{m=1}^{M} f_m(\boldsymbol{x}_m) \tag{26.1}$$

每个因式 $f_m(\boldsymbol{x}_m)$ 是组成 $\boldsymbol{x}$ 的变量的子集 $\boldsymbol{x}_m$ 的函数。如果 P^* 是一个正函数，可能会对另一个归一化函数感兴趣。

$$P(\boldsymbol{x}) \equiv \frac{1}{Z} P^*(\boldsymbol{x}) = \frac{1}{Z} \prod_{m=1}^{M} f_m(\boldsymbol{x}_m) \tag{26.2}$$

其中归一化常数 Z 定义为

$$Z = \sum_{\boldsymbol{x}} \prod_{m=1}^{M} f_m(\boldsymbol{x}_m) \tag{26.3}$$

作为刚刚引入的记号的例子，考虑用以下 5 个因式所定义的 3 个二进制变量 x_1，x_2，x_3 的函数。

$$\begin{aligned}
f_1(x_1) &= \begin{cases} 0.1, & x_1 = 0 \\ 0.9, & x_1 = 1 \end{cases} \\
f_2(x_2) &= \begin{cases} 0.1, & x_2 = 0 \\ 0.9, & x_2 = 1 \end{cases} \\
f_3(x_3) &= \begin{cases} 0.9, & x_3 = 0 \\ 0.1, & x_3 = 1 \end{cases} \\
f_4(x_1, x_2) &= \begin{cases} 1, & (x_1, x_2) = (0,0) \text{或} (1,1) \\ 0, & (x_1, x_2) = (1,0) \text{或} (0,1) \end{cases} \\
f_5(x_2, x_3) &= \begin{cases} 1, & (x_2, x_3) = (0,0) \text{或} (1,1) \\ 0, & (x_2, x_3) = (1,0) \text{或} (0,1) \end{cases}
\end{aligned} \tag{26.4}$$

$$P^*(\boldsymbol{x}) = f_1(x_1) f_2(x_2) f_3(x_3) f_4(x_1, x_2) f_5(x_2, x_3)$$

$$P(\boldsymbol{x}) = \frac{1}{Z} f_1(x_1) f_2(x_2) f_3(x_3) f_4(x_1, x_2) f_5(x_2, x_3)$$

在一般函数(26.1)中,用 $\boldsymbol{x}_m$ 表示的 $\{x_1, x_2, x_3\}$ 的 5 个子集是 $\boldsymbol{x}_1 = \{x_1\}$, $\boldsymbol{x}_2 = \{x_2\}$, $\boldsymbol{x}_3 = \{x_3\}$, $\boldsymbol{x}_4 = \{x_1, x_2\}$ 和 $\boldsymbol{x}_5 = \{x_2, x_3\}$。

顺便提一下,可认为函数 $P(\boldsymbol{x})$ 是接收信号为 $\boldsymbol{r} = (1,1,0)$、二进制对称信道的翻转概率为 0.1 时,重复码中 3 个发送比特的后验概率分布(第 1.2 节)。因式 f_4 和 f_5 分别施加约束 x_1 和 x_2 必须相同以及 x_2 和 x_3 必须相同。因式 f_1, f_2, f_3 是 $\boldsymbol{r}$ 中每个分量所贡献的似然函数。

具有因式分解形式的函数(26.1)可以用一个因式图来描述,图中圆结点表示变量,方结点表示因式。只要函数 $f_m(\boldsymbol{x}_m)$ 依赖于变量 x_n,变量结点 n 和因式结点 m 之间就存在一条边。如图 26.1 所示是函数(26.4)的因式图。

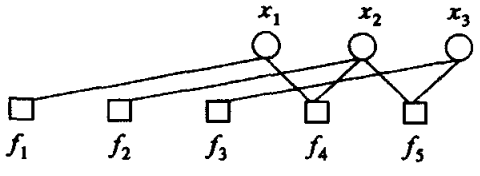


图 26.1 与函数 $P^*(\boldsymbol{x})$(26.4)相关的因式图

归一化问题

第一项需要完成的任务是计算归一化常数 Z。

边缘化问题

第二项需要完成的任务是计算任意变量 x_n 的边缘函数,它定义为

$$Z_n(x_n) = \sum_{\{x_{n'}\}, n' \neq n} P^*(\boldsymbol{x}) \tag{26.5}$$

例如,如果 f 是 3 个变量的函数,则对于 $n = 1$ 的边缘函数定义为

$$Z_1(x_1) = \sum_{x_2, x_3} f(x_1, x_2, x_3) \tag{26.6}$$

这种类型的求和(对"除了 x_n 之外的所有 $x_{n'}$"求和)非常重要,有必要赋予它一个专门的名称——"非和"或"求和式"。

需要解决的第三项任务是计算任意变量 x_n 的归一化边缘函数,它定义为

$$P_n(x_n) \equiv \sum_{\{x_{n'}\}, n' \neq n} P(\boldsymbol{x}) \tag{26.7}$$

[在 $P_n(x_n)$ 中保留了下标"n",这不同于本书其余部分的做法,那里都省略了。]

▷ **习题 26.1**[1] 证明归一化边缘函数和边缘函数 $Z_n(x_n)$ 的关系是

$$P_n(x_n) = \frac{Z_n(x_n)}{Z} \tag{26.8}$$

我们也许会对变量子集的边缘化感兴趣,如

$$Z_{12}(x_1, x_2) \equiv \sum_{x_3} P^*(x_1, x_2, x_3) \tag{26.9}$$

一般来说，所有这些任务都是不易求解的。即使每个因式只是3个变量的函数，计算 Z 和边缘函数的准确解的代价通常都会随变量 N 的个数呈指数级增长。

然而对于某些函数 P^*，通过 P^* 的因式分解来计算边缘函数非常有效。第16章的消息传递例子很好地说明了如何达到这种效率的思想。现在将要复习的和－积算法是对消息传递规则集B（第16.1节）的推广。与那里的情况一样，只有在图是树形时，和－积算法才是适用的。

26.2 和－积算法

记号

通过序数集 $\mathcal{N}(m)$ 识别第 m 个因式所依赖的变量集合 $\boldsymbol{x}_m$。例如对于函数(26.4)，集合是 $\mathcal{N}(1)=\{1\}$（因为 f_1 只是 x_1 的函数），$\mathcal{N}(2)=\{2\}$，$\mathcal{N}(3)=\{3\}$，$\mathcal{N}(4)=\{1,2\}$，$\mathcal{N}(5)=\{2,3\}$。同样可通过 $\mathcal{M}(n)$ 定义变量 n 所参与因式的集合。通过 $\mathcal{N}(m)\backslash n$ 表示从 $\mathcal{N}(m)$ 中排除变量 n 所得的集合。通过简写形式 $\boldsymbol{x}_m\backslash n$ 或 $\boldsymbol{x}_{m\backslash n}$ 表示从变量集合 $\boldsymbol{x}_m$ 中排除 x_n，即

$$\boldsymbol{x}_m\backslash n \equiv \{x_{n'}: n' \in \mathcal{N}(m)\backslash n\} \tag{26.10}$$

和－积算法将涉及在因式图沿边线传递的两类消息：从变量结点到因式结点的消息 $q_{n\to m}$，从因式结点到变量结点的消息 $r_{m\to n}$。沿着连接因式结点 f_m 到变量结点 x_n 的边线所传递的消息（无论是 q 类还是 r 类）都是变量 x_n 的一个函数。

下面是用于更新两个消息集合的两条规则。

从变量到因式：

$$q_{n\to m}(x_n) = \prod_{m'\in\mathcal{M}(n)\backslash m} r_{m'\to n}(x_n) \tag{26.11}$$

从因式到变量：

$$r_{m\to n}(x_n) = \sum_{\boldsymbol{x}_m\backslash n}\left(f_m(\boldsymbol{x}_m)\prod_{n'\in\mathcal{N}(m)\backslash n} q_{n'\to m}(x_{n'})\right) \tag{26.12}$$

如何把这些规则应用于因式图中的叶结点

只有一条边与其他结点相连的结点叫做叶结点。

图中的某些因式结点可能只和一个变量结点相连，在这种情况下，出现在因式消息更新(26.12)中的变量集合 $\mathcal{N}(m)\backslash n$ 是一个空集，而函数乘积 $\prod_{n'\in\mathcal{N}(m)\backslash n} q_{n'\to m}(x_{n'})$ 是空的乘积，它的值是1。所以这样的因式结点总是向它的一个邻居 x_n 广播消息 $r_{m\to n}(x_n)=f_m(x_n)$。如图26.2所示。

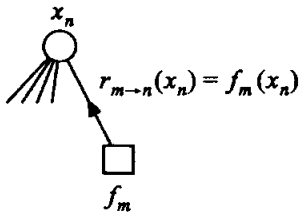


图 26.2 因式结点是永远传递消息 $r_{m\to n}(x_n)=f_m(x_n)$ 到它的邻居结点 x_n 的叶结点。

同样，可能有一个变量结点只连接到一个因式结点，所以式(26.11)中的集合 $\mathcal{M}(n)\backslash m$ 为空。这些结点始终广播消息 $q_{n\to m}(x_n)=1$。如图 26.3 所示。

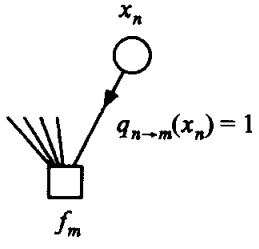


图 26.3 变量结点是永远传递消息 $q_{n\to m}(x_n)=1$ 的叶结点。

开始和结束，方法 1

可以有两种方法对算法进行初始化。如果图是树形的，则必定有叶结点。从一开始起，这些叶结点就可以把它们的消息广播给各自的邻居。

$$\text{对于所有叶变量结点 } n\text{：}\quad q_{n\to m}(x_n)=1 \tag{26.13}$$

$$\text{对于所有叶因式结点 } m\text{：}\quad r_{m\to n}(x_n)=f_m(x_n) \tag{26.14}$$

然后可以采用第 16 章的消息传递规则集 B 所使用的过程：只有一条消息所依赖的所有消息都出现，才按照规则(26.11,26.12)产生这条消息。例如在图 26.4 中，接收 f_4 到 x_1 的消息后，x_1 才会传递消息到 f_1；当消息 $r_{4\to 2}$ 和 $r_{5\to 2}$ 都被接收到以后，才可以传送从 x_2 到 f_2 的消息 $q_{2\to 2}$。

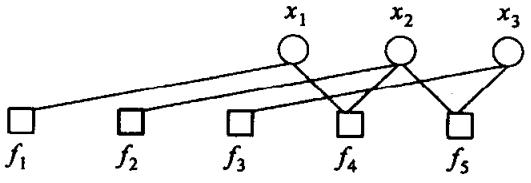


图 26.4 函数 $P^*(x)$(26.4)的模型因式图

消息就这样通过树传送，沿每条边一个方向可以有一条消息；经过等于图的直径的步骤数之后，所有消息就都产生出来了。

现在就可以得到所需答案了。通过把所有进入该结点的消息相乘就可以得到 x_n 的边缘函数。

$$Z_n(x_n) = \prod_{m \in \mathcal{M}(n)} r_{m \to n}(x_n) \tag{26.15}$$

对任何边缘函数求和，可以得到归一化常数 $Z = \sum_{x_n} Z_n(x_n)$，而归一化边缘函数为

$$P_n(x_n) = \frac{Z_n(x_n)}{Z}. \tag{26.16}$$

▷ **习题 26.2**[⌕] 对式(26.4)和图 26.1 所定义的函数应用和－积算法。检验归一化边缘函数是否和你所知道的重复码 R_3 相一致。

习题 26.3[⌕] 如果图是树形的，证明和－积算法是否能正确计算边缘函数 $Z_n(x_n)$。

习题 26.4[⌕] 描述如何利用和－积算法所计算的消息，来得到树形图中更为复杂的边缘函数，比如 $Z_{1,2}(x_1, x_2)$，两个变量 x_1 和 x_2 连接到一个共同的因式结点。

开始和结束，方法 2

或者，通过将来自各变量的所有初始消息都设置为 1，来对算法进行初始化。

$$\text{对于所有 } n, m: q_{n \to m}(x_n) = 1 \tag{26.17}$$

然后交替执行因式消息更新规则(26.12)和变量消息更新规则(26.11)。与方法 1 相比，这个较为懒惰的初始化方法会导致浪费计算的负荷，这种计算的结果会逐渐被方法 1 所得到的正确计算结果所代替。

经过与因式图直径的相等的迭代次数之后，算法将收敛到满足和－积关系(26.11，26.12)的消息集合。

习题 26.5[⌕] 对式(26.4)和图 26.1 所定义的函数应用第二种和－积算法。

采用这种懒惰方法的原因在于(和方法 1 不同)，它可以用在非树形图上。当和－积算法在有回路的图上运行时，算法不一定会收敛，一般来说是肯定计算不出正确的边缘函数的；但它仍然是一个很重要的实用算法，尤其是对稀疏图码的译码而言。

采用实时归一化的和－积算法

如果只是对归一化边缘函数感兴趣，那么另外一种和－积算法可能会有用。因式－变量消息 $r_{m \to n}$ 的计算同(26.12)完全一样，但是变量－因式消息就归一化为

$$q_{n \to m}(x_n) = \alpha_{nm} \prod_{m' \in \mathcal{M}(n) \backslash m} r_{m' \to n}(x_n) \tag{26.18}$$

其中标量 α_{nm} 的选择满足

$$\sum_{x_n} q_{n \to m}(x_n) = 1 \tag{26.19}$$

习题 26.6[⌕] 对式(26.4)和图 26.1 所定义的函数应用归一化的和－积算法。

从因式分解的观点解析和 - 积算法

看待和 - 积算法的一种方法是把原先 M 个因式的乘积 $P^*(\boldsymbol{x}) = \prod_{m=1}^{M} f_m(\boldsymbol{x}_m)$ 的分解函数重新表示为另一 $M+N$ 个因式乘积的分解函数

$$P^*(\boldsymbol{x}) = \prod_{m=1}^{M} \phi_m(\boldsymbol{x}_m) \prod_{n=1}^{N} \psi_n(x_n) \tag{26.20}$$

每个因式 ϕ_m 和一个因式结点 m 相关联，每个因式 $\psi_n(x_n)$ 和一个变量结点相关联。开始的时候有 $\phi_m(\boldsymbol{x}_m) = f_m(\boldsymbol{x}_m)$ 和 $\psi_n(x_n) = 1$。

每一次都传送因式 - 变量消息 $r_{m\to n}(x_n)$，于是分解就可更新为

$$\psi_n(x_n) = \prod_{m \in \mathcal{M}(n)} r_{m\to n}(x_n) \tag{26.21}$$

$$\phi_m(\boldsymbol{x}_m) = \frac{f(\boldsymbol{x}_m)}{\prod_{n \in \mathcal{N}(m)} r_{m\to n}(x_n)} \tag{26.22}$$

利用

$$r_{m\to n}(x_n) = \sum_{\boldsymbol{x}_m \backslash n} \left(\phi_m(\boldsymbol{x}_m) \prod_{n' \in \mathcal{N}(m)} \psi_{n'}(x_{n'}) \right) \tag{26.23}$$

就可以根据 ϕ 和 ψ 计算每条消息。它和赋值(26.12)的不同之处在于，它的乘积是针对所有 $n' \in \mathcal{N}(m)$。

习题 26.7[ρ] 验证更新规则(26.21 ~ 26.23)与和 - 积规则(26.11 ~ 26.12)等价。所以 $\psi_n(x_n)$ 最终变成边缘函数 $Z_n(x_n)$。

无论图是否是树形的，这种因式分解的观点都可以适用。

计算技巧

从计算的角度来看，实时归一化是一个好主意，因为如果 P^* 是许多因式的乘积，它的值可能会很大或很小。

另一个有用的计算技巧涉及传递消息 q 和 r 的对数，而非 q 和 r 本身；算法(26.11，26.12)中的乘积计算就被简单的加法所代替。式(26.12)中的和式当然会变得更难：为了对它们进行计算并给出对数结果，需要计算 softmax 函数

$$l = \ln(\mathrm{e}^{l_1} + \mathrm{e}^{l_2} + \mathrm{e}^{l_3}) \tag{26.24}$$

但是利用查表方法以及关于答案 l 值通常只是略大于 $\max_i l_i$ 的观察，可以更加有效地进行计算。如果把函数值

$$\ln(1 + \mathrm{e}^{\delta}) \tag{26.25}$$

存储在查找表中(对于负的 δ 值)，则经过几次查表以及根据和式中的项数按比例进行累加，就可以准确地计算出 l。如果查表和排序操作不如 exp( ) 复杂，那么这种方法的代价就会小于直接计算(26.24)。通过忽略 $\{l_i\}$ 中那些可忽略不计的最小贡献，可进一步减少操作次数。

第三个可用于某些纠错码的计算技巧是不传递消息，而是传递消息的傅里叶变换。这也会加快因式 - 变量消息的计算。第 47 章中的式(47.9)就是应用傅里叶变换技巧的一个简单例子。

26.3 最小和算法

和 - 积算法解决了对已知乘积 $P^*(\boldsymbol{x})$ 求边缘函数的问题。这与第 25.1 节中的求解逐比特译码问题相似。正如存在其他译码问题一样(例如，码字译码问题)，可以定义涉及 $P^*(\boldsymbol{x})$ 的其他任务，并利用修正的和 - 积算法来求解。例如，考虑同码字译码问题相似的下列任务。

最大化问题 找到最大化乘积 $P^*(\boldsymbol{x})$ 的 $\boldsymbol{x}$ 设置。

这个问题可以这样求解：把和 - 积算法中凡是出现“加法”和“乘法”两种运算的地方，都用满足分配律的另一对运算“最大化”和“乘法”来代替。如果用最大化来代替和(+ , $\sum$)，则注意到原来已知的归一化常数量

$$Z = \sum_{\boldsymbol{x}} P^*(\boldsymbol{x}) \tag{26.26}$$

就变成了 $\max_{\boldsymbol{x}} P^*(\boldsymbol{x})$。

因此和 - 积算法就变成了计算 $\max_{\boldsymbol{x}} P^*(\boldsymbol{x})$ 的最大积算法，由此可以推导出最大化问题的解。每个“边缘函数” $Z_n(x_n)$ 就列出了 $P^*(\boldsymbol{x})$ 对每个 x_n 所能达到的最大值。

实际上，最大积算法经常在负对数似然域中进行，其中“最大化”和“乘法”分别变成“最小化”和“加法”。最小和算法也称为 Viterbi 算法。

26.4 汇结树算法

如果所感兴趣的因式图不是树形的，我们该怎么办呢？

存在几种选择，分为精确方法和近似方法。对有回路的图进行边缘化处理的最常用精确方法叫做汇结树算法。这种算法把变量都聚在一起，直到聚集后的图中没有回路。你也许可以自己想出各种细节；边缘化的复杂性随着聚集变量的数目呈指数级增长。关于汇结树算法的更多内容请参见(Lauritzen，1996 年；Jordan，1998 年)。

有许多种近似方法，在后面几章将遇见到一些，略举两例：蒙特卡罗法和变参法。然而如前所述，处理和 - 积算法可能不适用的因式图，最有意思的方式是应用和 - 积算法！只要对图中的每个结点计算消息，就好像图真的是一棵树，迭代，然后就静候结果了。这种所谓的“回环式”消息传递，在纠错码的译码中起到很重要的作用，我们将在第六部分的第 33.8 节中重新研究它。

进一步的阅读

要进一步阅读因式图及和 - 积算法,参见 Kschischang 等(2001 年)、Yedidia 等(2000 年)、Yedidia 等(2001a)、Yedidia 等(2002 年)、Wainwright 等(2003 年)和 Forney(2001 年)。

也可参见 Pearl(1988 年)。一本关于图模型基本理论的好参考书是 Lauritzen(1996 年)。Jensen(1996 年)提供了一个可读性很强的贝叶斯网络导论。

与和 - 积算法效能不同的有趣消息传递算法包括置信传播(Minka,2001 年)和概观传播(Braunstein 等,2003 年)。也可参见第 33.8 节。

26.5 习 题

▷ **习题 26.8**[考] 将防盗警报和地震问题(例 21.1)的联合概率分布表示成一个因式图,并利用具有实时归一化的和 - 积算法,求出当每一则消息引起弗雷德注意时所有变量的边缘概率。

第 27 章

拉普拉斯方法

拉普拉斯近似的基本思想很简单。假设一个未归一化的概率密度 $P^*(x)$ 在点 x_0 处有一个尖峰，且令人感兴趣的参数是其归一化常数

$$Z_P \equiv \int P^*(x)\,\mathrm{d}x \tag{27.1}$$

在尖峰附近对 $P^*(x)$ 的对数进行泰勒级数展开

$$\ln P^*(x) \simeq \ln P^*(x_0) - \frac{c}{2}(x-x_0)^2 + \cdots \tag{27.2}$$

其中

$$c = -\frac{\partial^2}{\partial x^2}\ln P^*(x)\Big|_{x=x_0} \tag{27.3}$$

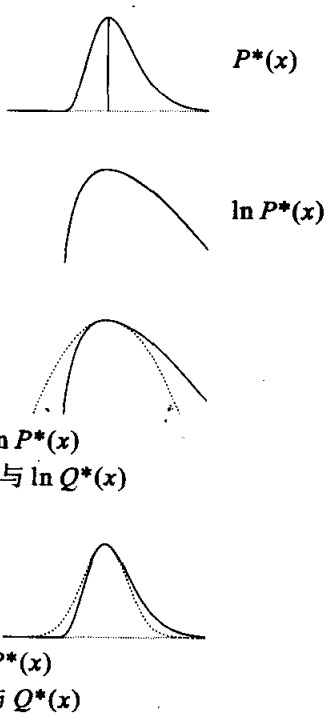
用非归一化的高斯函数

$$Q^*(x) \equiv P^*(x_0)\exp\left[-\frac{c}{2}(x-x_0)^2\right] \tag{27.4}$$

来近似 $P^*(x)$。然后通过该高斯函数的归一化常数

$$Z_Q = P^*(x_0)\sqrt{\frac{2\pi}{c}} \tag{27.5}$$

来近似归一化常数 Z_P。



可以将这一积分进行推广，来近似 K 维空间 $\boldsymbol{x}$ 上的密度 $P^*(\boldsymbol{x})$ 所对应的归一化常数 Z_P。如果 $-\ln P^*(\boldsymbol{x})$ 在最大点 $\boldsymbol{x}_0$ 处的二阶导数矩阵为 $\boldsymbol{A}$，其元素定义如下

$$A_{ij} = -\frac{\partial^2}{\partial x_i \partial x_j} \ln P^*(\boldsymbol{x}) \Big|_{x=x_0} \tag{27.6}$$

于是展开式(27.2)可推广为

$$\ln P^*(\boldsymbol{x}) \simeq \ln P^*(\boldsymbol{x}_0) - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_0)^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{x}-\boldsymbol{x}_0) + \cdots \tag{27.7}$$

那么归一化常数可以近似为

$$Z_P \simeq Z_Q = P^*(\boldsymbol{x}_0) \frac{1}{\sqrt{\det \frac{1}{2\pi}\boldsymbol{A}}} = P^*(\boldsymbol{x}_0) \sqrt{\frac{(2\pi)^K}{\det \boldsymbol{A}}} \tag{27.8}$$

可以利用近似 Q 来进行预测。物理学家也将这一广泛使用的近似称为鞍点近似。

可以对基向量 $\boldsymbol{u}$ 使用正交变换，将矩阵 $\boldsymbol{A}$ 变换成对角矩阵，由此不难证明一个高斯函数的归一化常数为

$$\int \mathrm{d}^K \boldsymbol{x} \exp\left[-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}} \boldsymbol{A} \boldsymbol{x}\right] = \sqrt{\frac{(2\pi)^K}{\det \boldsymbol{A}}} \tag{27.9}$$

于是前述积分就可分解成一维积分的乘积，其中每个一维积分的形式为

$$\int \mathrm{d}u_i \exp\left[-\frac{1}{2}\lambda_i u_i^2\right] = \sqrt{\frac{2\pi}{\lambda_i}} \tag{27.10}$$

特征值 λ_i 的乘积是 $\boldsymbol{A}$ 的行列式。

拉普拉斯近似与所选用的基有关：如果将 x 变换成一个非线性函数 $u(x)$，密度函数变换成 $P(u) = P(x)\,|\mathrm{d}x/\mathrm{d}u|$，那么一般来说，所得的近似归一化常数 Z_Q 会有所不同。可以将这一性质视做一个缺点，因为 Z_P 的真值与所选用的基无关。也可将它视做一个机会，因为可以寻求使得拉普拉斯近似最精确的基。

27.1 习 题



习题 27.1[2]（也可参见习题 22.8。） 将一个光子计数器对准一颗远星达一分钟，以推断出每分钟光子到达计数器的速率 λ。假设所收集到的光子数 r 服从均值为 λ 的泊松分布

$$P(r|\lambda) = \exp(-\lambda)\frac{\lambda^r}{r!} \tag{27.11}$$

并假设不合适的先验分布为 $P(\lambda) = 1/\lambda$。

(a) 对在 λ 上的后验分布进行拉普拉斯近似。

(b) 对在 $\log \lambda$ 上的后验分布进行拉普拉斯近似。[注意：不合适的先验分布变

换成 $P(\log \lambda) =$ 常数。]

▷ **习题 27.2**[$\mathscr{P}$] 使用拉普拉斯方法近似积分

$$Z(u_1,u_2) = \int_{-\infty}^{\infty} \mathrm{d}a f(a)^{u_1}(1-f(a))^{u_2} \tag{27.12}$$

其中 $f(a) = 1/(1+\mathrm{e}^{-a})$，$u_1$ 与 u_2 为正。对 $(u_1,u_2) = (1/2,1/2)$ 和 $(u_1,u_2) = (1,1)$ 两种情况，与精确解(23.29)进行比较，来确定近似的精度。以比特来度量误差 $(\log Z_P - \log Z_Q)$。

▷ **习题 27.3**[$\mathscr{P}$] **线性回归**。试验者选择各个 $x^{(n)}$，物理世界给出它的一个有噪线性函数

$$y(x) = w_0 + w_1 x \tag{27.13}$$

$$t^{(n)} \sim \mathrm{Normal}(y(x^{(n)}),\sigma_\nu^2) \tag{27.14}$$

并据此生成 N 个数据点 $\{(x^{(n)},t^{(n)})\}$。

假设 w_0 和 w_1 的高斯先验分布，对 w_0 和 w_1 的后验分布(它实际上是精确的)进行拉普拉斯近似，并对已知的 $x^{(N+1)}$ 给出下一个数据点 $t^{(N+1)}$ 的预测分布。

(进一步的阅读请参阅参考文献 MacKay(1992a)。)

第 28 章

模型比较与奥卡姆剃刀原理

28.1 奥卡姆剃刀

图 28.1 中有多少个箱子？特别地，在树的附近有多少个箱子？如果戴上 X 光眼镜，会看到树干后面是一个还是两个箱子（图 28.2）？（或者甚至更多？）奥卡姆剃刀所论述的是倾向于简单理论的原理。“接受同数据相符的最简单解释”。因此，按照奥卡姆剃刀原理，应该得出的结论是树后只有一个箱子。这仅仅是一个经验式方法呢，还是有充分的理由令人相信最可能的情况是树后只有一个箱子？可能你的直觉会支持这样一种论点“嗯，两个箱子的高度和颜色都相同是一个难得的巧合”。如果想获得一些可以正确解释数据的人工智能，我们必须将这种直观的感觉翻译成一个具体的理论。

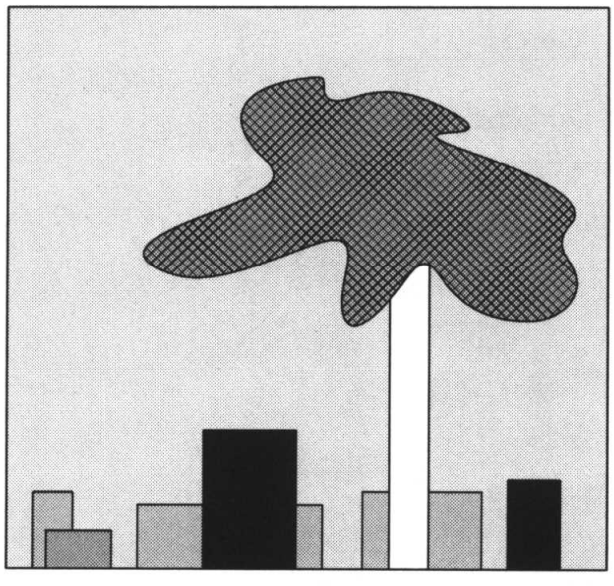


图 28.1　需要解释的一幅图。它包含了一棵树和一些箱子。

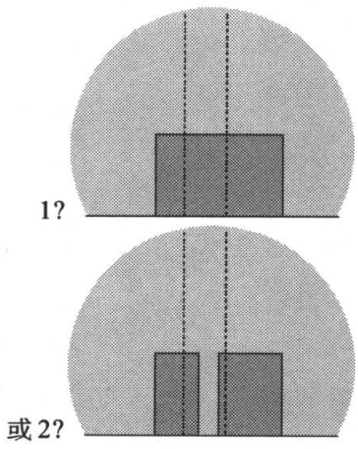


图 28.2　树后面有多少个箱子？

奥卡姆剃刀原理的动机

如果几种解释都和一组观测数据相容，奥卡姆剃刀原理建议采用最简单的那种。这一原理经常因为如下两个原因之一得到提倡：第一个原因是美学方面的考虑（“同

一个和实验数据相符的丑陋理论相比，一个在数学上优美的理论正确的可能性总是更大些”(Paul Dirac))；第二个原因是奥卡姆剃刀原理在过去取得了实际的成功。不过，对于奥卡姆剃刀原理还有一种不同的理由，即

一致性推理在数量上自动地体现了奥卡姆剃刀原理(正如贝叶斯概率所体现的那样)。

树后只有一个箱子的观点确实更合适，而且可以计算有一个箱子比有两个箱子的可能性大多少。

模型比较与奥卡姆剃刀原理

针对数据 D，评估两种理论 $\mathscr{H}_1$ 和 $\mathscr{H}_2$ 的可信度如下：使用贝叶斯定理，将已知数据时模型 $\mathscr{H}_1$ 的可信度 $P(\mathscr{H}_1|D)$、模型对数据所进行的预测 $P(D|\mathscr{H}_1)$ 和 $\mathscr{H}_1$ 的先验可信度 $P(\mathscr{H}_1)$ 联系起来。这会给出理论 $\mathscr{H}_1$ 和 $\mathscr{H}_2$ 的概率比如下

$$\frac{P(\mathscr{H}_1|D)}{P(\mathscr{H}_2|D)}=\frac{P(\mathscr{H}_1)}{P(\mathscr{H}_2)}\frac{P(D|\mathscr{H}_1)}{P(D|\mathscr{H}_2)} \tag{28.1}$$

右边第一个比值 $P(\mathscr{H}_1)/P(\mathscr{H}_2)$ 所度量的是初始置信度在多大程度上倾向于 $\mathscr{H}_1$ 而非 $\mathscr{H}_2$。第二个比值表示 $\mathscr{H}_1$ 相对于 $\mathscr{H}_2$ 对观察数据进行预测的好坏程度。

当模型 $\mathscr{H}_1$ 比 $\mathscr{H}_2$ 简单时，这和奥卡姆剃刀原理如何联系起来呢？如果我们愿意，第一个比值 $P(\mathscr{H}_1)/P(\mathscr{H}_2)$ 给了一次机会，可以基于美学或经验方面的考虑，预先插入一个有利于 $\mathscr{H}_1$ 的偏置。这相应于早先提到过的奥卡姆剃刀原理在美学或经验方面的动机。不过，这样一个先验偏置并非必要：第二个比值，即与数据相关的因素，自动地体现了奥卡姆剃刀原理。简单模型倾向于做出精确的预测。复杂模型本质上就会做出差别很大的预测(如图 28.3 所示)。因此，如果 $\mathscr{H}_2$ 是一个比较复杂的模型，它的预测型先验概率 $P(D|\mathscr{H}_2)$ 必然会在数据空间上分散得比 $\mathscr{H}_1$ 要薄。因此，在数据和两种理论都兼容的情况下，不需要我们表达对复杂模型的主观嫌恶，简单模型 $\mathscr{H}_1$ 出现的可能性也会比 $\mathscr{H}_2$ 大些。在主观上只需对简单性和复杂性预先设置相等的先验概率。概率论会允许观测数据表达它们的意愿。

先来看一个简单的例子。下面有一列数

$$-1,3,7,11$$

任务是预测下面两个数，并推断给出这个数列的潜在过程。这个问题的一个普遍接受的答案是：预测为“15,19”，而其解释为“在前一个数上加 4”。

另外一种答案“-19.9,1 043.8”的情况如何？它可由潜在规则“由前一个数 x 估算 $-x^3/11+9/11x^2+23/11$ 从而得到下一个数”给出。我假设这种预测看来是很不可能的。但是，第二种规则同规则“加 4”一样，都未和数据(-1,3,7,11)出现矛盾。那么为什么觉得第二种预测不太可能呢？先给下面两种一般性理论做标记。

$\mathscr{H}_a$——“加 n”等差数列，这里 n 是一个整数。

$\mathscr{H}_c$——由形如 $x\to cx^3+dx^2+e$ 的三次函数所生成的数列，其中 c、d 和 e 是分数。

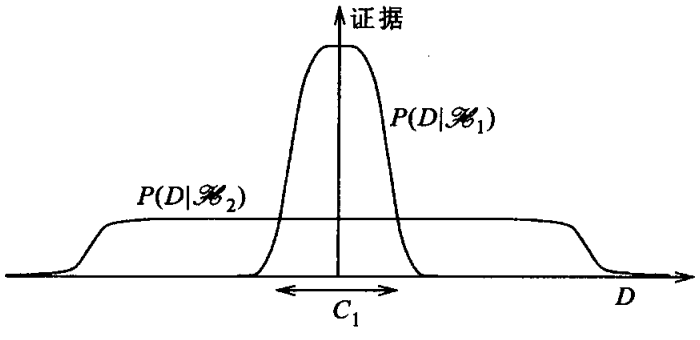


图 28.3 为什么贝叶斯推理体现了奥卡姆剃刀原理。本图给出了为什么复杂模型出现的可能性要小一些的基本直觉。横轴表示可能数据集 D 的空间。贝叶斯定理根据模型预测数据的能力，按比例“褒奖”模型。通过 D 上的一个归一化概率分布对这些预测进行量化。在已知模型 $\mathscr{H}_i$ 的前提下，数据的概率 $P(D|\mathscr{H}_i)$ 称作 $\mathscr{H}_i$ 的显著性。一个简单的模型 $\mathscr{H}_1$ 只能做出一些有限范围的预测，如 $P(D|\mathscr{H}_1)$ 所示；而一个更强大的模型 $\mathscr{H}_2$（比如，比 $\mathscr{H}_1$ 的自由参数要多），则可以预测一个更大范围的数据集。不过，这也意味着 $\mathscr{H}_2$ 对在 C_1 区域数据的预测不如 $\mathscr{H}_1$ 那么强。假设两种模型的先验概率相等。那么，如果数据集合落在区域 C_1，弱些的模型 $\mathscr{H}_1$ 将是更合适的模型。

觉得第二种解释 $\mathscr{H}_c$ 不太可能的原因大概在于，等差数列比三次函数更为常见。这可能会在式(28.1)中的先验概率比 $P(\mathscr{H}_a)/P(\mathscr{H}_c)$ 上加上一个偏置。不过，我们还是给两种理论赋予相等的先验概率，并专注于数据本身所表达的意思。每种理论对数据的预测情况如何？

要得到 $P(D|\mathscr{H}_a)$，必须说明每种模型中各参数的概率分布。首先，$\mathscr{H}_a$ 取决于加数 n 和数列中的第一个数。比如假定这些数可以是 $-50 \sim 50$ 之间的任何整数。因为只有 $\{n=4, \text{第一个数} = -1\}$ 这对数值会给出被观察数据 $D = \{-1, 3, 7, 11\}$，在已知 $\mathscr{H}_a$ 的前提下，数据的概率为

$$P(D|\mathscr{H}_a) = \frac{1}{101} \cdot \frac{1}{101} = 0.000\ 10 \tag{28.2}$$

要计算 $P(D|\mathscr{H}_c)$，必须类似地说明分数 c、d 和 e 可以取什么值。[我选择将这些数表示为分数而非实数，因为如果使用实数，相对于 $\mathscr{H}_a$，这个模型会给数据 D 赋予一个无穷小的概率。不过实参数是一种规范，在本章其余部分都会如此假设。]一个合理的先验假设可以表述为：分数的分子可以是 $-50 \sim 50$ 之间的任何整数，而分母则是 $1 \sim 50$ 之间的任何整数。而对于序列的初值，则让它的概率分布和 $\mathscr{H}_a$ 时相同。在这些先验假设下，有 4 种表示分数 c 的方式：$c = -1/11 = -2/22 = -3/33 = -4/44$；类似地，$d$ 和 e 也分别有 4 种和 2 种表示方式。因此，在已知 $\mathscr{H}_c$ 的前提下，被观察数据的概率为

$$P(D|\mathscr{H}_c) = \left(\frac{1}{101}\right)\left(\frac{4}{101} \cdot \frac{1}{50}\right)\left(\frac{4}{101} \cdot \frac{1}{50}\right)\left(\frac{2}{101} \cdot \frac{1}{50}\right)$$

$$=0.000\,000\,000\,002\,5=2.5\times10^{-12} \quad (28.3)$$

因此，将 $P(D|\mathscr{H}_c)$ 与 $P(D|\mathscr{H}_a)=0.000\,10$ 进行比较可知，即使假设 $\mathscr{H}_a$ 和 $\mathscr{H}_c$ 的先验概率相等，已知序列 $D=(-1,3,7,11)$，比值 $P(D|\mathscr{H}_a):P(D|\mathscr{H}_c)$ 有利于 $\mathscr{H}_a$ 而非 $\mathscr{H}_c$ 的几率几乎是 4 000 万比一。

这一答案与几个主观假设有关；特别是，两套理论中为自由参数 n,c,d 和 e 所赋予的概率。贝叶斯论者并不觉得这样有何不妥：不做假设而进行推理或预测，这种事是从来没有的！不过，先验概率的定量方面的细节对奥卡姆剃刀原理的定性效果并没有影响；复杂些的理论 $\mathscr{H}_c$ 总是会损失一个“奥卡姆因子”，因为它有更多的参数，并因此能预测更多种的数据集（如图 28.3 所示）。这只是一个小例子，数据点数也只有 4 个；当我们考虑更大、更复杂的问题时，奥卡姆因子的幅度通常会增大，而关于数量细节的主观假设对我们推理的影响程度也随之变小。

贝叶斯方法与数据分析

现在将上面所讨论的内容和数据分析中的实际问题联系起来。

在科学、统计和技术方面有无数这种问题：已知有限数据集，要求对具有不同复杂度的几种模型比较出优劣。比如，有两种描述行星运动的假设：宗教审判先生基于“本轮”的地心说模型，和哥白尼先生的简单模型，即太阳位于太阳系中心的模型。本轮模型与行星运动数据相符程度不亚于哥白尼模型，但这是建立在使用了更多参数的基础之上的。宗教审判先生对于每个行星使用的额外本轮参数中的两个，恰巧和太阳“绕地圆周运动”的周期与半径相同。根据直觉，我们觉得哥白尼先生的理论更合适。

贝叶斯剃刀机理：显著性与奥卡姆因子

在数据建模的过程中，经常可以区别出两级推理。如图 28.4 所示。在第一级推理中，假设一个特定模型是正确的，并根据数据对模型进行拟合，即已知数据，推断模型的自由参数应取什么值。这一级推理的结果经常可以归纳为最合适的参数值以及这些参数的误差限。这一分析对每个模型都要重复一次。第二级推理的任务是模型比较。在这里，希望根据数据对不同的模型进行比较，并对各个模型赋予某种偏好或等级。

注意两级推理都和决策论不同。推理的目标是，已知一个确定的假设空间和某特定数据集，为每种假设赋予概率值。而决策论通常根据这些概率在不同的动作中做出选择，以便最小化一个“损失函数”的期望值。本章只关心推理问题，没有涉及损失函数。在讨论模型比较时，不应将问题构造成隐含了模型选择的意思。理想的贝叶斯预测不涉及在模型之间进行选择；而是以各个模型的概率做加权求和认得出预测。

贝叶斯方法可以一致地在数量上完成两种推理任务。有这样一种流行的神话，

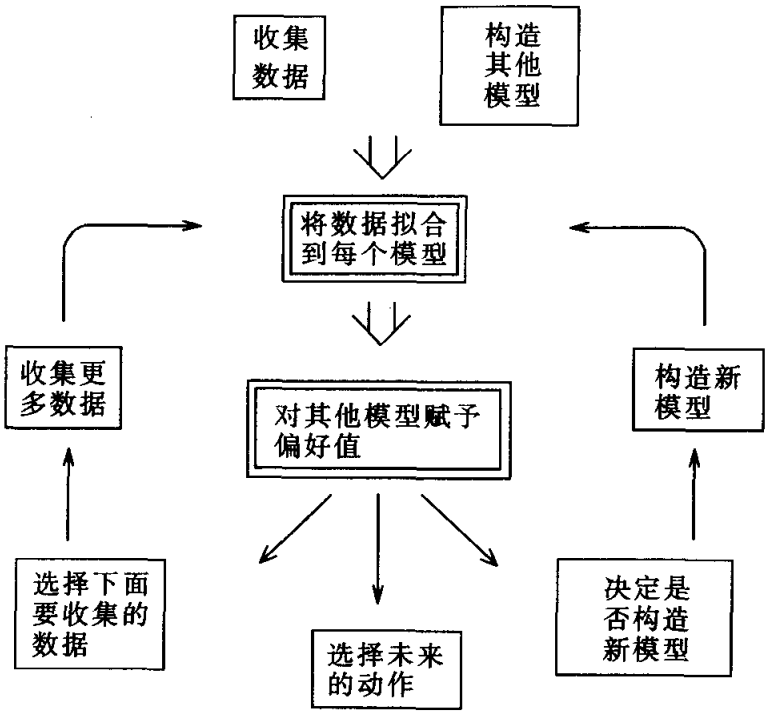


图 28.4 贝叶斯推理在数据建模过程中所处的位置。本图所示为科学过程中数据收集和建模部分的抽象框图。特别地，本图适用于模式分类、学习和内插等过程。图中的两个双线框表示涉及推理的两个步骤。只有在这两步中才能使用贝叶斯定理。比如，贝叶斯方法并未告诉你如何构造模型。第一个方框“将数据拟合到每个模型”所要完成的任务是，对于已知的模型和数据，推断出可能的模型参数。贝叶斯方法可用于寻找最可能的参数值以及这些参数的误差限。对这个问题运用贝叶斯方法所得结果通常和正统统计学所给出的答案差别不大。第二个推理任务，即针对数据进行模型比较，则显示贝叶斯方法是一组具有独特性的方法。这第二个推理问题需要一个定量的奥卡姆剃刀，对复杂模型施以惩罚。贝叶斯方法可以赋予其他模型以客观的偏好，使之自动地体现奥卡姆剃刀效应。

贝叶斯方法和正统统计方法之间的唯一差别在于前者包含了主观先验概率，而这些先验概率不容易得到，通常也不会对所得结论产生太大影响。在第一级推理中，贝叶斯方法的结果往往和正统统计方法的结果相差甚微，确实如此。而贝叶斯方法在第二级推理中的性能则尚未得到广泛认同；因此本章将集中讨论贝叶斯模型比较问题。

模型比较是一项艰巨的任务，因为不可能简单地选择同数据最相符的模型：复杂模型一般会和数据更吻合，因此最大似然模型选择必然会给出一些令人难以置信的、过参数化的模型，而这种模型很不容易推广。此时需要奥卡姆剃刀。

对前述两级推理写出贝叶斯定理，以便明显地看出贝叶斯模型比较如何工作。假设每个模型 $\mathcal{H}_i$ 有一个参数向量 $\boldsymbol{w}$。将一个模型定义为一组概率分布：一个“先验”分布 $P(\boldsymbol{w}|\mathcal{H}_i)$，说明期望模型参数取何值；一组条件分布，每个 $\boldsymbol{w}$ 值一个，定义模型对数据 D 所进行的预测 $P(D|\boldsymbol{w},\mathcal{H}_i)$。

1. **模型拟合** 在第一级推理中，假设一个模型（比如模型 i）为真，然后推断对于已知的数据 D，模型的参数 $\boldsymbol{w}$ 可能取什么值。运用贝叶斯定理，参数 $\boldsymbol{w}$ 的后验概率为

$$P(\boldsymbol{w}|D,\mathcal{H}_i)=\frac{P(D|\boldsymbol{w},\mathcal{H}_i)P(\boldsymbol{w}|\mathcal{H}_i)}{P(D|\mathcal{H}_i)} \tag{28.4}$$

即

$$\text{后验概率}=\frac{\text{似然度}\times\text{先验概率}}{\text{证据}}$$

通常忽略归一化常数 $P(D|\mathcal{H}_i)$，因为它与第一级推理（即 $\boldsymbol{w}$ 的推理）无关；但是它在第二级推理中很重要，可将它称为 $\mathcal{H}_i$ 的显著性。通常使用基于梯度的方法寻找最大后验概率 $\boldsymbol{w}_{\text{MP}}$，它所定义的是最可能的参数值；然后一般将后验分布归结为 $\boldsymbol{w}_{\text{MP}}$ 值和这些最佳拟合参数的误差限或置信区间。可以由后验概率的曲率得出误差限；计算在 $\boldsymbol{w}_{\text{MP}}$ 处的 Hessian 矩阵 $\boldsymbol{A}=-\nabla\nabla\ln P(\boldsymbol{w}|D,\mathcal{H}_i)|_{\boldsymbol{w}_{\text{MP}}}$，并使用 $\Delta\boldsymbol{w}=\boldsymbol{w}-\boldsymbol{w}_{\text{MP}}$ 对对数后验概率进行泰勒级数展开

$$P(\boldsymbol{w}|D,\mathcal{H}_i)\simeq P(\boldsymbol{w}_{\text{MP}}|D,\mathcal{H}_i)\exp(-1/2\Delta\boldsymbol{w}^{\text{T}}\boldsymbol{A}\Delta\boldsymbol{w}) \tag{28.5}$$

可以看出，后验概率可用一个协方差矩阵（与误差限等价）为 $\boldsymbol{A}^{-1}$ 的高斯分布来局部近似。[这个近似的好坏取决于所要解决的问题。事实上，后验概率的最大值和均值在贝叶斯推理中并不占重要地位，因为它们在非线性再参数化时都会改变。只有当式(28.5)之类的近似很好地总结了后验分布时，最大化后验概率才有用。]

2. **模型比较** 在第二级推理中，希望推断出，对于已知的数据，最有可能的是哪个模型。每个模型的后验概率为

$$P(\mathcal{H}_i|D)\propto P(D|\mathcal{H}_i)P(\mathcal{H}_i) \tag{28.6}$$

注意与数据相关的项 $P(D|\mathcal{H}_i)$ 是 $\mathcal{H}_i$ 的显著性，它以归一化常数的形式出现在式(28.4)中。第二项 $P(\mathcal{H}_i)$ 是定义在假设空间之上的主观先验概率，它所表示的是在数据到达之前我们所认为的各个模型可能性的大小。假设对各个模型选用相等的先验概率 $P(\mathcal{H}_i)$，通过计算显著性来对模型 $\mathcal{H}_i$ 进行排序。在式(28.6)中，忽略了归一化常数 $P(D)=\sum_i P(D|\mathcal{H}_i)P(\mathcal{H}_i)$，因为在数据建模过程中，比如当检测到第一类模型不尽如人意时，我们可能在数据到达后会开发新的模型。推理没有终结：继续寻找更合适的模型来解释我们所收集到的数据。

重复一下主要思想：为了对各个模型 $\mathcal{H}_i$ 进行排序，贝叶斯方法会计算显著性 $P(D|\mathcal{H}_i)$。这一概念非常通用：可以同样地对参数或“非参数”模型计算显著性；无论数据建模任务是回归问题、分类问题、概率密度估计问题或其他什么问题，显著性是比较不同模型时的一个可通用的量。在所有这些问题中，显著性自然地体现了奥卡姆剃刀。

计算显著性

下面更深入地研究显著性，以便更好地理解贝叶斯奥卡姆剃刀的工作原理。显

著性是式(28.4)中的归一化常数

$$P(D \mid \mathcal{H}_i) = \int P(D \mid w, \mathcal{H}_i) P(w \mid \mathcal{H}_i) \mathrm{d}w \tag{28.7}$$

对于许多问题,后验概率 $P(w \mid D, \mathcal{H}_i) \propto P(D \mid w, \mathcal{H}_i) P(w \mid \mathcal{H}_i)$ 在最可能的参数 w_{MP} 处会有一个很高的尖峰(如图 28.5 所示)。为了简单起见,考虑一维的情况:可以通过拉普拉斯方法,用被积函数 $P(D \mid w, \mathcal{H}_i) P(w \mid \mathcal{H}_i)$ 的尖峰高度乘以它的宽度 $\sigma_{w \mid D}$ 来近似显著性。

$$P(D \mid \mathcal{H}_i) \simeq \underbrace{P(D \mid w_{\mathrm{MP}}, \mathcal{H}_i)}_{\text{证据}} \times \underbrace{P(w_{\mathrm{MP}} \mid \mathcal{H}_i) \sigma_{w \mid D}}_{\text{奥卡姆因子}} \tag{28.8}$$

证据≃最佳拟合似然 × 奥卡姆因子

因此,可通过模型可达到的最佳拟合似然乘以奥卡姆因子来得出显著性。这里,奥卡姆因子是一个幅度小于 1 的项,它对 $\mathcal{H}_i$ 具有参数 w 进行惩罚。

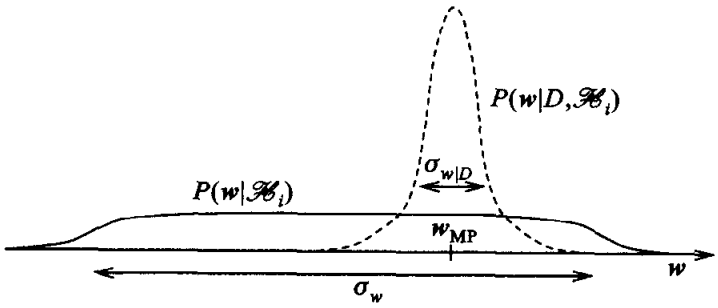


图 28.5 奥卡姆因子。本图所示为对具有单一参数 w 的假设 $\mathcal{H}_i$,决定奥卡姆因子的各个参量。参数 w 的先验分布(实线)的宽度为 σ_w。后验分布(虚线)在 w_{MP} 处有一个特征宽度为 $\sigma_{w \mid D}$ 的单峰。奥卡姆因子为

$$\sigma_{w \mid D} P(w_{\mathrm{MP}} \mid \mathcal{H}_i) = \frac{\sigma_{w \mid D}}{\sigma_w}$$

奥卡姆因子的解释

参量 $\sigma_{w \mid D}$ 是 w 的后验不确定性。为了简单起见,假设先验概率 $P(w \mid \mathcal{H}_i)$ 是某个大区间 σ_w 上的均匀分布,这里 σ_w 表示根据 $\mathcal{H}_i$,w 的先验可能值域(如图 28.5 所示)。于是 $P(w_{\mathrm{MP}} \mid \mathcal{H}_i) = 1/\sigma_w$,且有

$$\text{奥卡姆因子} = \frac{\sigma_{w \mid D}}{\sigma_w} \tag{28.9}$$

也就是说,奥卡姆因子等于 $\mathcal{H}_i$ 的参数空间的后验可达体积与先验可达体积之比,或 $\mathcal{H}_i$ 的假设空间在数据到达时会崩溃的因子。可以将模型 $\mathcal{H}_i$ 看做由一定数目的互斥子模型组成,在数据到达时,这些子模型中只有一个能存活。奥卡姆因子是那个数的倒数。奥卡姆因子的对数度量了在数据到达时,所获得的有关模型参数的信息量。

如果一个复杂模型具有许多参数,而每个参数都可以在一个大范围 σ_w 内自由变化,那么这样一个模型遭受来自奥卡姆因子的惩罚通常会比简单模型所受的惩罚要更强烈些。奥卡姆因子也惩罚那些需要精调才能和数据相符的模型,而偏爱那些

对参数 $\sigma_{w|D}$ 的精度要求很粗的模型。因此，奥卡姆因子是模型复杂度的一种度量；它和模型在数据空间中所做预测的复杂度相关。这不仅取决于模型中参数的个数，而且也取决于模型赋给它们的先验概率。哪个模型取得最大的显著性，要由最小化自然复杂度量和最小化数据失配二者之间的折中来决定。与模型复杂度的其他度量不同，一个模型的奥卡姆因子可以简单直接地计算出来：它仅仅取决于参数的误差限，而在对数据进行模型拟合时，已经计算出了这些值。

图 28.6 展示了一个完整的假设空间，以阐明在分析中所出现的各种概率。有 3 个具有相等先验概率的模型 $\mathcal{H}_1$，$\mathcal{H}_2$，$\mathcal{H}_3$。每个模型有一个参数 w（在横轴上表示），但是每个模型都对参数分配不同的先验范围 σ_w。$\mathcal{H}_3$ 是最为“灵活”或最为“复杂”的模型，具有最宽的先验范围。一个一维数据空间如图中的纵轴所示。每个模型都对数据和参数赋予一个联合概率分布 $P(D,w|\mathcal{H}_i)$，如图中的点云团所示。这些点表示来自全概率分布的随机样本。在 3 个模型子空间中，点的总数相同，因为对各个模型已赋予相等的先验概率。

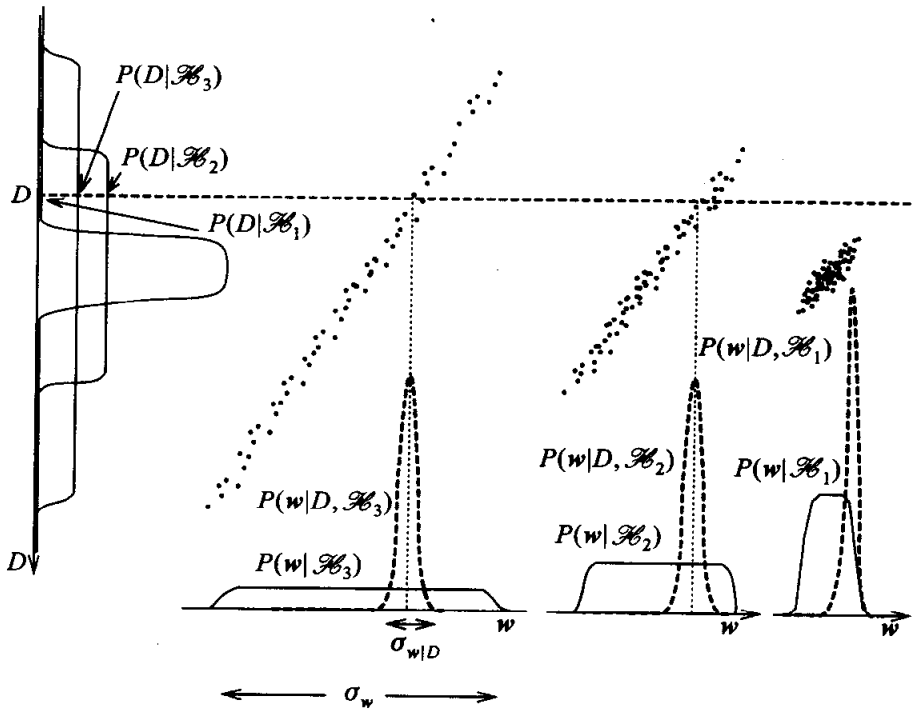


图 28.6 由 3 个互斥模型所组成的假设空间，其中每个模型都有一个参数 w 和一个一维数据集 D。“数据集”是单个测量值，它和参数 w 之间相差一个小幅度的叠加噪声。如图中小点所示为联合分布 $P(D,w,\mathcal{H})$ 的典型样本（注意：这些点不是数据点）。被观测“数据集”是如图水平虚线所示的单个特定 D 值。下面的虚线表示已知这个数据集时对应模型的 w 的后验概率（参看图 28.3）。通过在左边的 D 轴上进行边缘化，可以获得不同模型的显著性（参看图 28.5）。

当收到一个特定的数据集 D 时（水平线），我们读出沿该水平线的密度，并做归一化，就可针对一个模型（比如 $\mathcal{H}_3$）推断出 w 的后验分布。后验概率 $P(w|D,\mathcal{H}_3)$ 如

图底部的虚线所示。还画出了先验分布 $P(\boldsymbol{w}|\mathcal{H}_3)$(参看图 28.5)。[模型 $\mathcal{H}_1$ 和数据之间匹配很差,对于这种情况,后验分布的形状取决于先验概率 $P(\boldsymbol{w}|\mathcal{H}_1)$ 和似然 $P(D|\boldsymbol{w},\mathcal{H}_1)$ 的尾部详情;所示曲线对应于先验概率衰减更快的情形。]

将联合分布 $P(D,\boldsymbol{w}|\mathcal{H}_i)$ 边缘化到左边的 D 轴,可以得出图 28.3。对于水平虚线所示的数据集 D,更灵活模型 $\mathcal{H}_3$ 的显著性 $P(D|\mathcal{H}_3)$ 的数值比 $\mathcal{H}_2$ 的显著性小。这是因为 $\mathcal{H}_3$ 在那条线上投入了较少的预测概率(更少的点)。根据 $\boldsymbol{w}$ 的分布,模型 $\mathcal{H}_3$ 的显著性更小,因为 $\mathcal{H}_3$ 的奥卡姆因子 $\sigma_{w|D}/\sigma_w$ 比 $\mathcal{H}_2$ 的小。最简单的模型 $\mathcal{H}_1$ 在三者中具有最小的显著性,因为对数据 D 它所能取得的最佳拟合也很差。已知这个数据集,最可能的模型是 $\mathcal{H}_2$。

几个参数的奥卡姆因子

如果后验概率可以用高斯分布很好地近似,那么可以从相应协方差矩阵的行列式得出奥卡姆因子(参看式(28.8)和第 27 章):

$$P(D|\mathcal{H}_i) \simeq \underbrace{P(D|\boldsymbol{w}_{\mathrm{MP}},\mathcal{H}_i)}_{\text{最佳拟合似然}} \times \underbrace{P(\boldsymbol{w}_{\mathrm{MP}}|\mathcal{H}_i)\det{}^{-\frac{1}{2}}(\boldsymbol{A}/2\pi)}_{\text{奥卡姆因子}} \qquad (28.10)$$

证据 $\simeq$

其中 $\boldsymbol{A} = -\nabla\nabla \ln P(\boldsymbol{w}|D,\mathcal{H}_i)$ 是计算 $\boldsymbol{w}_{\mathrm{MP}}$ 的误差限时求出的 Hessian 矩阵(见式(28.5)和第 27 章)。随着所收集到的数据增多,可以预计高斯近似会变得越来越精确。

总而言之,贝叶斯模型比较是最大似然模型选择的一个简单扩展:*将最佳拟合似然乘以奥卡姆因子,可得显著性。*

如果高斯近似程度很好,只需 Hessian 矩阵 $\boldsymbol{A}$ 就可计算奥卡姆因子。因此,通过计算显著性来进行模型比较的贝叶斯方法在计算复杂度上,不会高于为每个模型寻找最佳拟合参数及其误差限这一任务。

28.2 例 子

现在回到本章开头给出的例子。在图 28.1 中,树后面有一个箱子还是两个箱子?为什么一些巧合会让我们变得狐疑?

假设树干和箱子周围的画面大小为 50 个像素,树干宽度为 10 个像素,可以分辨出 16 种不同颜色的箱子。如图 28.7 所示。宣称树干附近只有一个箱子的理论 $\mathcal{H}_1$ 有 4 个自由参数:定义箱子顶上 3 条边的 3 个坐标以及表示箱子颜色的 1 个参数。(如果箱子可以浮起来,就有 5 个自由参数。)

宣称树干附近有两个箱子的理论 $\mathcal{H}_2$ 有 8 个自由参数(两个 4 个),再加上第 9 个参数,一个表明两个箱子中哪个箱子更靠近观察者的二进制变量。

每个模型的显著性是多少?先来看看 $\mathcal{H}_1$。需要参数的先验概率来计算显著

性。为了方便起见,我们以像素为单位。对箱子的水平位置、宽度、高度和颜色赋予可分离的先验概率。比如,高度可以是20个可区分数值之一,宽度和位置也是如此。颜色可以是16种中的任何一种。对这些变量赋予均匀先验概率。忽略与图中其他物体相关的所有参数,因为它们同 $\mathscr{H}_1$ 和 $\mathscr{H}_2$ 之间的模型比较没什么关系。显著性是

$$P(D|\mathscr{H}_1)=\frac{1}{20}\cdot\frac{1}{20}\cdot\frac{1}{20}\cdot\frac{1}{16} \quad (28.11)$$

因为只有一组参数设置和数据相符,而且它完满地预测出了数据。

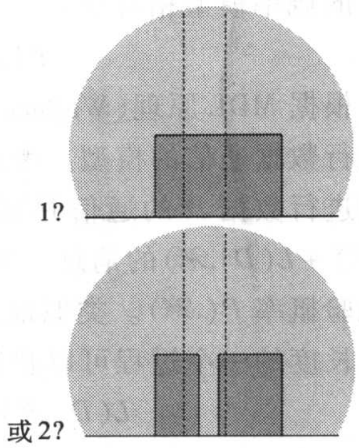


图28.7 树后有多少个箱子?

对于模型 $\mathscr{H}_2$,9个参数中的6个是完全确定的,而另外3个则部分地受数据约束。比如,如果左边的箱子离得最远,那么它的宽度至少是8个像素,最多是30个像素;如果它是两个箱子中离得较近的那个,那么它的宽度则在8~18个像素之间。(这里假设左边箱子的可见部分为8个像素宽。)为了得到显著性,需要将所有可成立假设的先验概率总结出来。要进行精确计算,必须有关于数据和先验概率的更具体信息,不过只要求得一个大致的答案就行了;并假设两个无约束的实变量有一半的值是已知的,而那个二进制变量的值则完全未确定。(作为练习,可以建立一个显式模型并求出精确解。)

$$P(D|\mathscr{H}_2)\simeq\frac{1}{20}\cdot\frac{1}{20}\cdot\frac{10}{20}\cdot\frac{1}{16}\cdot\frac{1}{20}\cdot\frac{1}{20}\cdot\frac{10}{20}\cdot\frac{1}{16}\cdot\frac{2}{2} \quad (28.12)$$

因此,假设先验概率相等,后验概率比为

$$\frac{P(D|\mathscr{H}_1)P(\mathscr{H}_1)}{P(D|\mathscr{H}_2)P(\mathscr{H}_2)}=\frac{1}{\frac{1\,1010\,1}{20202016}} \quad (28.13)$$

$$=20\times2\times2\times16\simeq1\,000/1 \quad (28.14)$$

因此,数据大约以1 000比1的比例倾向于简单些的假设。式(28.13)中的4个因子可以用奥卡姆因子予以解释。复杂些的模型另外还有4个参数:其中3个是关于大小的,一个是关于颜色的。两个箱子高度和颜色正好相同,这是非常可疑的巧合,因此它必须为此付出高昂的代价:两个大的奥卡姆因子(1/20和1/16);而另外两个不那么罕见的巧合是两个箱子刚好有一条边隐藏在树后或相互遮挡,为此所付出的奥卡姆因子也小一些。

28.3 最小描述长度(MDL)

如果将事件概率替换为接收方无损地进行事件通信所需消息的长度(以比特计算),那么就可得到贝叶斯模型比较的一个补充。消息长度 $L(\boldsymbol{x})$ 通过如下关系同事

件 x 的概率模型相对应。

$$P(x) = 2^{-L(x)}, L(x) = -\log_2 P(x) \tag{28.15}$$

根据 MDL 原理(Wallace 和 Boulton,1968 年),人们应该偏向于可以用最少比特数进行数据通信的模型。考虑一个包括两部分的消息:使用哪个模型 $\mathscr{H}$ 及通过这个模型进行数据 D 的通信并达到预定的精度 δD。这会产生一个长度为 $L(D,\mathscr{H}) = L(\mathscr{H}) + L(D|\mathscr{H})$ 的消息。不同 $\mathscr{H}$ 的长度 $L(\mathscr{H})$ 定义了一个相对于其他模型的隐式先验概率 $P(\mathscr{H})$。类似地,$L(D|\mathscr{H})$ 相应于一个概率密度 $P(D|\mathscr{H})$。因此,设定消息长度的一个过程可以映射为后验概率

$$L(D,\mathscr{H}) = -\log P(\mathscr{H}) - \log (P(D|\mathscr{H})\delta D) \tag{28.16}$$

$$= -\log P(\mathscr{H}|D) + 常数 \tag{28.17}$$

在原理上,MDL 总可以解释为贝叶斯模型比较,反之亦然。但是,这个简单讨论并没有解决如何实际计算关键的数据相关项 $L(D|\mathscr{H})$,它所对应的是 $\mathscr{H}$ 的显著性。通常,可以认为这个消息进一步被划分成参数段和数据段(如图 28.8 所示)。参数较少的模型只有一个较短的参数段,但是可能跟数据拟合得不太好,因此数据消息(一系列较大的残余信息)较长。随着参数个数的增加,参数段变长,数据消息也变短了。存在一个最优模型复杂度(图中的 $\mathscr{H}_2$)使得总长度最短。

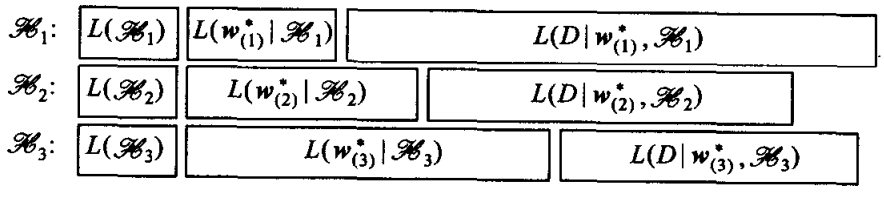


图 28.8 对模型比较的一个主流看法:最小描述长度。每个模型 $\mathscr{H}_i$ 发送模型标识、模型最佳拟合参数 w^* 以及与这些参数相关的数据,来完成数据 D 的通信。当进而考虑更复杂的模型时,参数消息的长度会增加。另一方面,数据消息的长度会减小,因为复杂模型可以更好地拟合数据,使得残余信息减少。在本例中,中等复杂的模型 $\mathscr{H}_2$ 在这两种趋势之间取得了最佳折中。

这张图略去了一些微妙的问题。我们没有指明参数 w 应该以何种精度进行发送。这个精度的影响很大(与取实数值的数据 D 的发送精度 δD 不同,后者假设 δD 比噪声水平低,仅仅引入一个叠加常数而已)。当降低 w 的发送精度时,参数消息变短了,但是数据消息通常会变长,因为截断后的参数同数据匹配得并不好。存在一个非平凡的最优精度。对于高斯这种简单情形,求解最优精度是可能的(Wallace 和 Freeman,1987 年),它和参数后验误差限 A^{-1} 密切相关,这里 $A = -\nabla\nabla \ln P(w|D,\mathscr{H})$。结果证明,最优参数消息长度基本上等于式(28.10)中的奥卡姆因子的对数。(在参数截断中涉及随机元素,这表明编码是略微有点次优的。)

因此,如果慎重一些,可以在 MDL 项中重复贝叶斯结果。尽管复杂模型比较方面一些最早期的工作涉及 MDL 框架(Patrick 和 Wallace,1982 年),MDL 同直接概率

方法相比,并没有明显的优势。

不过,作为教学工具,MDL 确实有用。描述长度的概念对于如何选择先验概率分布也很有帮助。而且,用模型将数据通信任务进行不同的分解,也有助于对建模过程提供深刻理解,下面会对此做进一步阐述。

在线学习与交叉校验

当数据由一系列的点 $D=t^{(1)},t^{(2)},\cdots,t^{(N)}$ 组成时,对数显著性可以分解成"在线"预测性能的和

$$\begin{aligned}\log P(D|\mathcal{H}) &= \log P(t^{(1)}|\mathcal{H}) + \log P(t^{(2)}|t^{(1)},\mathcal{H}) \\ &\quad + \log P(t^{(3)}|t^{(1)},t^{(2)},\mathcal{H}) + \cdots + \log P(t^{(N)}|t^{(1)}\cdots t^{(N-1)},\mathcal{H})\end{aligned} \tag{28.18}$$

这一分解可用于解释显著性和"留一个做交叉校验"作为预测能力度量之间的差别。交叉校验只在数据随机重排序的情况下,检查最后一项 $\log(t^{(N)}|t^{(1)},t^{(2)},\cdots,t^{(N-1)},\mathcal{H})$ 的平均值。而另一方面,显著性则从头开始总结模型对所有数据的预测情况。

"比特返回"编码方法

另一个 MDL 思想实验(Hinton 和 van Camp,1993 年)涉及将随机比特集成在我们的消息中。使用一个参数段和一个数据段进行数据的通信。所发送的参数向量是来自后验概率 $P(w|D,\mathcal{H})=P(D|w,\mathcal{H})P(w|\mathcal{H})/P(D|\mathcal{H})$ 的随机样本。用长度为 $L(w|\mathcal{H})=-\log[P(w|\mathcal{H})\delta w]$ 的消息,以任意小的粒度 δw 发送样本 w。以长度为 $L(D|w,\mathcal{H})=-\log[P(D|w,\mathcal{H})\delta D]$ 的消息相对于 w 对数据进行编码。一旦收到数据消息,接收机就可以推出用于从后验概率生成样本 w 的那些随机比特。这样恢复的比特数为 $-\log[P(w|D,\mathcal{H})\delta w]$。这些恢复的比特不必算入消息长度,因为也许可以使用其他一些经过最优编码的消息作为随机比特串,从而同时传送了那个消息。因此,净描述代价为

$$\begin{aligned}L(w|\mathcal{H}) + L(D|w,\mathcal{H}) - \text{"返回比特"} &= -\log\frac{P(w|\mathcal{H})P(D|w,\mathcal{H})\delta D}{P(w|D,\mathcal{H})} \\ &= -\log P(D|\mathcal{H}) - \log\delta D\end{aligned} \tag{28.19}$$

因此,这个思想试验得出了最优描述长度。对于用本征变量模型所建模的数据,比特返回编码已成为一种实用的压缩方法(Frey,1998 年)。

进一步的阅读

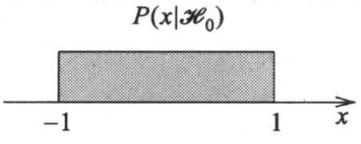
在(Jaynes,1983 年;Gull,1988 年;Loredo,1990 年)中,引入了贝叶斯方法,并将它与抽样理论统计学进行了比较。在(Gull,1988 年;MacKay,1992a)中,展示了如何将贝叶斯奥卡姆剃刀原理用于模型问题。有帮助的教材有(Box 和 Tiao,1973 年;Berger,1985 年)。

一个值得理解的争论是:是否允许在贝叶斯推理中使用不合适的先验概率(Dawid 等,1996 年)。如果想进行模型比较(像本章所讨论的那样),使用合适的先验概率至关重要,不然显著性和奥卡姆因子就毫无意义。只有当一个人无意进行模型比较时,使用不合适的先验概率才有可能是安全的;正如 Dawid 等所解释的,即便在这种情况下,还是可能出问题。笔者同意他们的建议:*总是使用合适的先验概率*,只是应鼓励在进行计算时保持开明的态度,以识别出进行近似的机会。

28.4 习 题

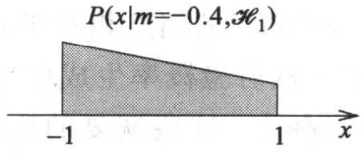
习题 28.1[扩] 随机变量 x 独立地出自一个概率分布 $P(x)$。根据模型 $\mathcal{H}_0$,$P(x)$ 是均匀分布

$$P(x|\mathcal{H}_0)=\frac{1}{2},\quad x\in(-1,1) \tag{28.20}$$



根据模型 $\mathcal{H}_1$,$P(x)$ 是具有未知参数 $m\in(-1,1)$ 的非均匀分布

$$P(x|m,\mathcal{H}_1)=\frac{1}{2}(1+mx),\quad x\in(-1,1) \tag{28.21}$$

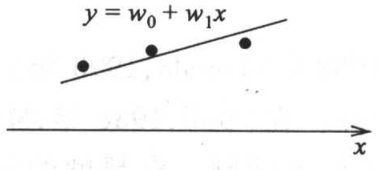


已知数据 $D=\{0.3,0.5,0.7,0.8,0.9\}$,$\mathcal{H}_0$ 和 $\mathcal{H}_1$ 的显著性是多少?

习题 28.2[扩] 数据点(x,t)据信出自于一条直线。试验者选择 x,而 t 服从均值为

$$y=w_0+w_1x \tag{28.22}$$

方差为 σ_ν^2 的高斯分布。根据模型 $\mathcal{H}_1$,直线是水平的,因此 $w_1=0$。根据模型 $\mathcal{H}_2$,w_1 是一个先验分布为正态(0,1)的参数。在两个模型中,w_0 的先验分布都是正态(0,1)。已知数据集 $D=\{(-8,8),(-2,10),(6,11)\}$,并假设噪声水平 $\sigma_\nu=1$,每个模型的显著性是多少?



习题 28.3[♪] 投掷一枚六边骰子 30 次，各面朝上的次数为 $F = \{3,3,2,2,9,11\}$。这个骰子是一枚完全公平的骰子（“$\mathcal{H}_0$”）的概率是多少？假设另一个假设 $\mathcal{H}_1$ 说这个骰子是一枚具有偏倚分布 $\boldsymbol{p}$ 的骰子，这里 $\boldsymbol{p}$ 的先验密度在单纯形 $p_i \geq 0$，$\sum_i p_i = 1$ 上是均匀的。

用两种方法求解这个问题：使用有助益的 Dirichlet 公式（23.30，23.31）进行精确求解；使用拉普拉斯方法进行近似求解。注意，你为拉普拉斯近似所选择的基很重要。有关本习题的讨论，可参看 MacKay（1998a）。

习题 28.4[♪] 美国竞选关于强迫接受谋杀犯死刑的影响得到了大量研究。如下 3 - 向表对 326 例宣判犯谋杀罪的被告进行了分类。3 个变量分别是被告的种族、被害人的种族以及被告是否被判处死刑。（数据来自 M. Radelet，“Racial characteristics and imposition of the death penalty”，American Sociological Review，46（1981），pp. 918 - 927。）

白种被告			黑种被告		
	死刑			死刑	
	是	否		是	否
白种被害人	19	132	白种被害人	11	52
黑种被害人	0	9	黑种被害人	6	97

看上去好像当被害人是白种人时，判死刑的情况要多些。当被害人是白种人时，14% 的被告被判死刑，但是当被害人是黑种人时，6% 的被告被判死刑。[碰巧，这些数据为辛普森悖论提供了一个例子。辛普森悖论：总的来说，白种被告人被判死刑的比例较高，但是在涉及黑种被害人时，黑种被告人被判死刑的比例较高，而在涉及白种被害人时，黑种被告人被判死刑的比例较高。]

量化如图 28.9 所示 4 种假设的显著性。应该提一下，笔者并不相信这些模型中

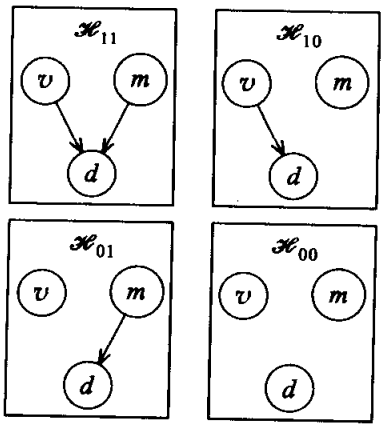


图 28.9 有关强制接受死刑 d 和被害人的种族 v 以及判刑的谋杀犯的种族 m 之间依存关系的四种假设。比如，$\mathcal{H}_{01}$ 断言接受死刑的概率跟谋杀犯的种族有关，但和被害人的种族无关。

的任何一个是理由充分的:另外还有好几个对于谋杀案很重要的变量,比如被害人和谋杀犯是否认识,谋杀犯是否有预谋,被告人是否有犯罪前科;这些变量在表中一个也没有考虑。因此,这只是模型比较方面的一道学术习题,而非有关佛罗里达州种族歧视的一项严肃研究。

几种假设以图形模型的方式表示,其中箭头表明变量 v(被害人的种族)、m(谋杀犯的种族)和 d(是否被判死刑)之间的依存关系。模型 $\mathscr{M}_{00}$ 只有一个自由参数,即接受死刑的概率;模型 $\mathscr{M}_{11}$ 有 4 个自由参数,变量 v 和 m 的每个状态各有一个。对这些变量赋予均匀先验概率。所得结论对先验概率的选择有多敏感?

关于第 29 章

最近两章假设对于我们感兴趣的概率分布，高斯近似就足够了。要是这种近似还不够呢？我们已经见过这样一个例子：聚类，此时似然函数是多峰的，而且在参数空间的某些地方还存在着无限高的尖峰这种麻烦；因此，最大化后验概率和拟合为高斯函数不会总是有用。使用拉普拉斯方法所遇到的困难是我们对蒙特卡罗方法感兴趣的原因之一。事实上，蒙特卡罗方法提供了一类通用的工具，可应用于贝叶斯数据建模和许多其他领域。

本章描述一系列方法：*重要性采样*、*舍弃采样*、*Metropolis 方法*、*Gibbs 采样*和*切片采样*。对于每种方法，都会讨论它是否可用于高维问题，比如用图模型进行推理的问题。[图模型是用图的结点表示变量，而用图的边表示变量之间的相关性和独立性关系。]顺便也介绍了马尔科夫链蒙特卡罗方法的有关术语。随后一章讨论了减小随机游动特性的一些高级方法。

关于蒙特卡罗方法、定理和证明以及较全面的参考文献列表，建议读者阅读参考文献 Neal(1993b)、Gilks 等(1996 年)和 Tanner(1996 年)。

在本章，笔者会在如下意义下使用“样本”一词：来自分布 $P(x)$ 的样本是概率分布为 $P(x)$ 的单一实现 x。这与统计学中的另一种用法不同，在那里“样本”指的是一组实现 $\{x\}$。

在讨论转移概率矩阵时，笔者会使用右乘规范：我喜欢矩阵对右边发生作用，也就是偏爱

$$\boldsymbol{u} = \boldsymbol{M}\boldsymbol{v} \tag{29.1}$$

而非

$$\boldsymbol{u}^{\mathrm{T}} = \boldsymbol{v}^{\mathrm{T}}\boldsymbol{M}^{\mathrm{T}} \tag{29.2}$$

转移概率矩阵 T_{ij} 或 $T_{i|j}$ 表示已知当前状态为 j，从 j 转移到 i 的概率。矩阵 $\boldsymbol{T}$ 的列是概率向量。在书写转移概率密度函数时，也采用同样的规范来表示变量的次序：$T(x';x)$ 表示从 x 到 x' 的转移概率密度。遗憾的是，你必须习惯从右往左读：序列 xyz 的概率为 $T(z;y)T(y;x)\pi(x)$。

第 29 章

蒙特卡罗方法

29.1 要求解的问题

蒙特卡罗方法是一类使用随机数的计算方法。蒙特卡罗方法的目标是求解下面一个或两个问题。

问题 1:由一个已知的概率分布 $P(x)$ 生成样本 $\{x^{(r)}\}_{r=1}^{R}$。

问题 2:在此分布下,估算一个函数的期望值,比如

$$\Phi = \langle \phi(x) \rangle \equiv \int \mathrm{d}^N x P(x) \phi(x) \tag{29.3}$$

我们称概率分布 $P(x)$ 为目标密度,它可能是来自统计物理的概率分布,也可能是来自数据建模的条件分布,例如,已知一些观测数据时模型参数的后验概率。通常假设 x 是一个具有实元素 x_n 的 N 维向量,但是有时也会考虑抽象空间。

对其期望值感兴趣的一些简单函数 $\phi(x)$ 包括想预测的量的一阶矩和二阶矩,由此可以计算出均值和方差;比如如果某变量 t 依赖于 x,可以计算出 t 在 $P(x)$ 意义下的均值和方差,只需求出函数 $\phi_1(x) = t(x)$ 和 $\phi_2(x) = (t(x))^2$ 的期望值。

$$\Phi_1 \equiv \varepsilon[\phi_1(x)] \text{和} \Phi_2 \equiv \varepsilon[\phi_2(x)] \tag{29.4}$$

然后使用

$$\bar{t} = \Phi_1 \text{和 } \mathrm{var}(t) = \Phi_2 - \Phi_1^2 \tag{29.5}$$

这里假设 $P(x)$ 足够复杂,无法用精确方法计算这些期望值;因此,我们对蒙特卡罗方法感兴趣。

集中考虑第一个问题(采样),因为如果我们解决了它,那么可以通过使用随机样本 $\{x^{(r)}\}_{r=1}^{R}$ 给出估计器

$$\hat{\Phi} \equiv \frac{1}{R} \sum_r \phi(x^{(r)}) \tag{29.6}$$

来解决第二个问题。如果向量 $\{x^{(r)}\}_{r=1}^{R}$ 是由 $P(x)$ 产生的,那么 $\hat{\Phi}$ 的期望就是 Φ。而且,随着样本数目 R 的增加,$\hat{\Phi}$ 的方差会降为 σ^2/R,这里 σ^2 是 ϕ 的方差。

$$\sigma^2 = \int \mathrm{d}^N x P(x) (\phi(x) - \Phi)^2 \tag{29.7}$$

这是蒙特卡罗方法的一个重要性质。

蒙特卡罗估计(29.6)的精度只取决于 ϕ 的方差,与样本空间的维度无关。更准确地说,$\hat{\Phi}$的方差会变为 σ^2/R。因此,无论 x 的维度是多少,十余个独立样本 $\{x^{(r)}\}$ 也许就足够给出 Φ 的满意估计。

不过,在后面将看到高维度问题会导致蒙特卡罗方法的其他一些困难。从已知的分布$P(x)$中得出独立样本通常不是很容易的。

为什么很难从分布 $P(x)$ 中采样?

假设想得到样本的密度 $P(x)$ 可以计算出来,至少是在一个常数因子的范围内可以计算出来;也就是说,可以计算一个函数 $P^*(x)$ 使得

$$P(x) = P^*(x)/Z \tag{29.8}$$

如果可以计算 $P^*(x)$,为什么不能轻易地求解问题 1 呢? 为什么一般很难由函数 $P(x)$ 得出样本? 这里存在着两大困难。首先,人们通常不知道归一化常数

$$Z = \int \mathrm{d}^N x P^*(x) \tag{29.9}$$

其次,即使真的知道 Z,从 $P(x)$ 中抽取样本的问题还是很难的,特别是在高维空间中,因为不把大部分或全部的可能状态列举出来,就没有什么好办法从 P 中抽取样本。由定义可知,从 P 中抽取正确样本一般来自 $P(x)$ 较大的 x 空间;如果不能计算 $P(x)$ 在各处的值,又怎能辨认出那些 $P(x)$ 较大的地方呢? 只有很少的几个高维密度函数,可以从中很容易地抽取样本,高斯分布是其中一个例子。

现在从一个简单的一维例子着手。假设要从密度函数 $P(x) = P^*(x)/Z$ 中抽取样本,这里

$$P^*(x) = \exp[0.4(x-0.4)^2 - 0.08x^4], \quad x \in (-\infty, \infty) \tag{29.10}$$

可以画出这个函数(如图 29.1(a)所示)。但是,这并不意味着可以从那里抽取样本。首先,并不知道归一化常数 Z。为了让问题简单些,可以对变量 x 离散化,离散概率分布定义在一个等间距有限点集 $\{x_i\}$ 上(如图 29.1(b)所示),要从这个分布上抽取样本。如何解决这个问题? 如果求出了每个点 x_i 上的概率值 $p_i^* = P^*(x_i)$,就可计算

$$Z = \sum_i p_i^* \tag{29.11}$$

以及

$$p_i = p_i^*/Z \tag{29.12}$$

并且接下来就可以使用基于随机比特源的不同方法(参见第 6.3 节),从概率分布 $\{p_i\}$ 中采样。但是这一过程的代价如何,这个代价随着空间维度 N 的增加会如何相应地变化? 先集中考虑用式(29.11)计算 Z 的初始代价。要计算 Z,必须对空间中的每个点逐一进行访问。在图 29.1(b)中,有 50 个点在一维空间上等距分布。如果系统是 N 维的,比如 $N = 1\,000$,那么相应的点数就会变成 $50^{1\,000}$,这是一个难以想象的大数,无法计算 P^*。即使每个元素 x_n 只取两个离散值,也要计算 $2^{1\,000}$ 次 P^* 值,这

个数目仍然是大得吓人。如果宇宙中的每个电子(约有 2^{266} 个)都是一个主频为 1 000 GHz的计算机,它每秒可以计算一万亿(2^{40})个状态的 P^* 值,让所有 2^{266} 台计算机同时运行一段长度等于宇宙年龄的时间(2^{58} s),它们也仅仅只能访问 2^{364} 个状态。还需再等 $2^{636} \sim 10^{190}$ 个宇宙年龄,才能等到所有 $2^{1\,000}$ 个状态都被访问到。

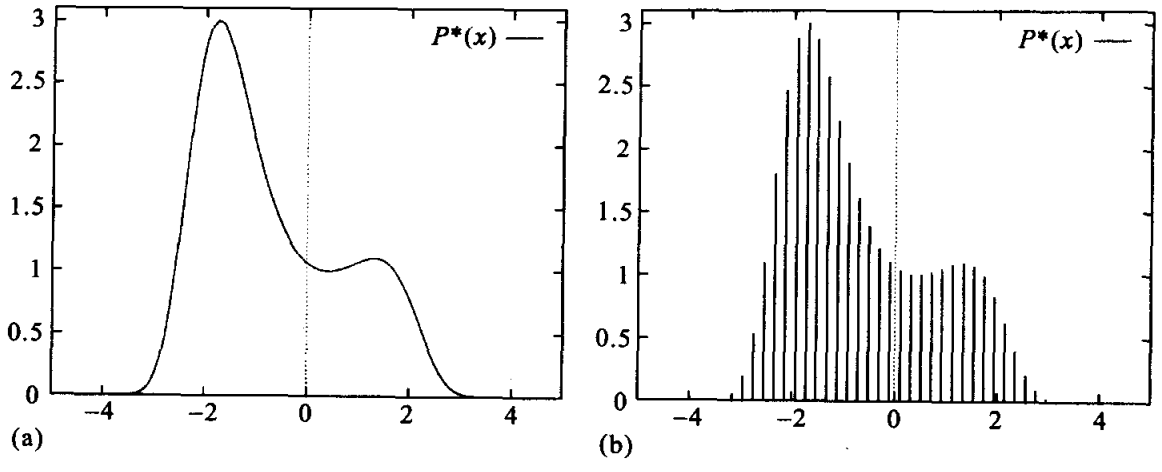


图 29.1 (a) 函数 $P^*(x) = \exp[0.4(x-0.4)^2 - 0.08x^4]$。如何从这个密度中抽取样本?(b) 定义在一个等间距离散点集 $\{x_i\}$ 上的函数 $P^*(x)$。如何从这个离散分布上抽取样本?

具有 $2^{1\,000}$ 个状态的系统是不稀罕的①。一个例子是:采集 1 000 个自旋,以至于一个伊辛模型的 30 × 30 个片段的概率分布正比于

$$P^*(x) = \exp[-\beta E(x)] \tag{29.13}$$

这里 $x_n \in \{\pm 1\}$,且有

$$E(x) = -\left[\frac{1}{2}\sum_{m,n} J_{mn}x_m x_n + \sum_n H_n x_n\right] \tag{29.14}$$

对于任何 x,都可以很容易地计算出能量函数 $E(x)$。但是如果想对所有状态 x 计算该函数值,所需的计算机时间就要计算 $2^{1\,000}$ 次函数值。

这种伊辛模型是一个简单模型,已出现了很长一段时间,但是从分布 $P(x) = P^*(x)/Z$ 生成样本,仍然是一个活跃的研究领域;从这个分布中所抽取的第一个"精确"样本由 Propp 和 Wilson(1996 年)在他们开创性的工作中完成,在第 32 章会对此进行描述。

一个有用的类比

设想这样一项任务:从一个湖里采集水样本,并得出浮游生物的平均浓度(如图 29.2 所示)。在 $x = (x, y)$ 处,湖的深度为 $P^*(x)$,并且假定浮游生物的浓度是 x 的

① 翻译成美国读者熟悉的说法:"这样的系统是一毛钱一打";碰巧,这一等效($10c = 6p$)表明了我们的货币之间的一个正确汇率 1 英镑 = 1.67 美元。

函数 $\phi(x)$(这样假设是为了使类比工作有效)。所要求的平均浓度是一个类似于式(29.3)的积分,具体而言就是

$$\Phi = \langle \phi(x) \rangle \equiv \frac{1}{Z} \int \mathrm{d}^N x P^*(x) \phi(x) \tag{29.15}$$

其中 $Z = \int \mathrm{d}x\mathrm{d}y P^*(x)$ 是湖的体积。你可以使用一条小船、一个卫星导航系统和一个铅锤仪。利用导航仪,可以把小船开到地图上的任何想去的位置 x;利用铅锤仪,可以测量该点处的 $P^*(x)$。还可以在那里测量浮游生物的浓度。

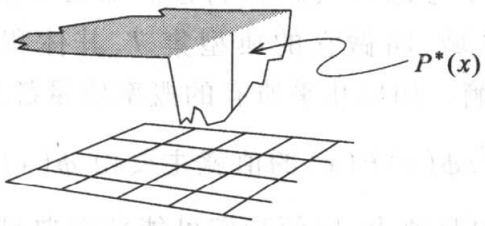


图 29.2 一个在点 $x=(x,y)$ 处深度为 $P^*(x)$ 的湖

问题 1 是从湖中随机地抽取 1 cm^3 的水样本,这样每个样本都是等概率地从湖中任何一点抽取的。问题 2 是求出浮游生物的平均浓度。

这些问题很难解答,因为在开始时对深度 $P^*(x)$ 一无所知。也许湖的很大一部分体积集中在狭窄深邃的水下峡谷(如图 29.3 所示),此时,要正确地在湖中进行采样并正确地估计 Φ,我们的方法必须隐含地找到这些峡谷,并求出它们的体积和湖的其他部分体积的相对值。这个问题很困难?是的;但是,可以看到聪明的蒙特卡罗方法可以解决它们。

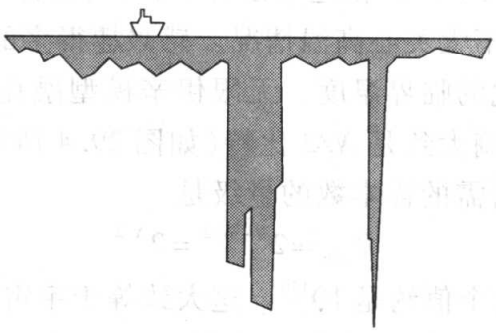


图 29.3 包含了一些峡谷的湖的切片图

均匀采样

接受了无法穷举遍历状态空间中的每一个点 x 这一事实,可以尝试解答第二个问题(估计函数 $\phi(x)$ 的期望值)。解决方法是,从状态空间中均匀地抽取随机样本 $\{x^{(r)}\}_{r=1}^{R}$,并计算这些点上的 $P^*(x)$。然后,可以引入定义如下的一个归一化

常数 Z_R

$$Z_R = \sum_{r=1}^{R} P^*(x^{(r)}) \tag{29.16}$$

并用下式估计 $\Phi = \int \mathrm{d}^N x \phi(x) P(x)$。

$$\hat{\Phi} = \sum_{r=1}^{R} \phi(x^{(r)}) \frac{P^*(x^{(r)})}{Z_R} \tag{29.17}$$

这种策略有何不妥？呃，这取决于函数 $\phi(x)$ 和 $P^*(x)$。先假设 $\phi(x)$ 是良性的、变化平缓的函数，并集中考虑 $P^*(x)$ 的特性。如第4章所知，高维分布通常会集中在一个小的状态空间区域，叫做它的典型集 T，其体积由 $|T| \simeq 2^{H(x)}$ 给出，这里 $H(x)$ 是概率分布 $P(x)$ 的熵。如果几乎所有的概率质量都集中在典型集上，且 $\phi(x)$ 是一个良性函数，$\Phi = \int \mathrm{d}^N x \phi(x) P(x)$ 的值就主要由 $\phi(x)$ 在典型集上的取值决定。因此，只有当样本数 R 取得足够大，以便我们可能落在典型集上一两次，均匀采样才有希望给出 Φ 的一个良好的估计。那么，需要多少个样本呢？再次以伊辛模型为例进行考虑。（严格地说，伊辛模型可能不是一个好例子，因为它不一定有一个如第4章所定义的典型集；典型集的定义是其中所有状态的对数概率都接近熵，对伊辛模型这意味着能量很接近平均能量；但是在相位转换附近，能量的方差（也叫热容）也可能发散，这表明一个随机状态的能量未必很靠近平均能量。）状态空间的总规模为 2^N，而典型集的大小为 2^H。因此，每个样本落在典型集上的机会是 $2^H/2^N$。所以，落在典型集上一次所需的样本数的量级是

$$R_{\min} \simeq 2^{N-H} \tag{29.18}$$

那么，H 是多少？在高温下，伊辛模型的概率分布趋向于均匀分布，熵接近 $H_{\max} = N$ 比特，也就是说 $R_{\min}$ 在量级1上。在这些条件下，均匀采样可能是估计 Φ 的一个令人满意的方法。但是，我们不太关心高温情况。感兴趣得多的是中等温度，比如伊辛模型从有序相到无序相融化的临界温度。无限伊辛模型融化的临界温度是 $\theta_c = 2.27$。在此温度下，伊辛模型的熵大致是 $N/2$ 比特（如图29.4所示）。对于这一概率分布，仅仅落在典型集上一次所需的样本数的量级是

$$R_{\min} \simeq 2^{N-N/2} = 2^{N/2} \tag{29.19}$$

对于 $N = 1\,000$ 的情况，这个值约是 10^{150}。这大致等于宇宙中所有粒子数的平方。因此，均匀采样对于研究一个中等规模的伊辛模型而言是完全没有用的。在多数高维问题中，如果分布 $P(x)$ 并非真正均匀，均匀采样也不太可能起到什么作用。

概述

前面已确定，即使 $P^*(x)$ 很容易计算，从高维分布 $P(x) = P^*(x)/Z$ 中抽取样本仍然是困难的。现在研究一系列更为复杂的蒙特卡罗方法：重要性采样、舍弃采样、Metropolis 方法、Gibbs 采样和分片采样。

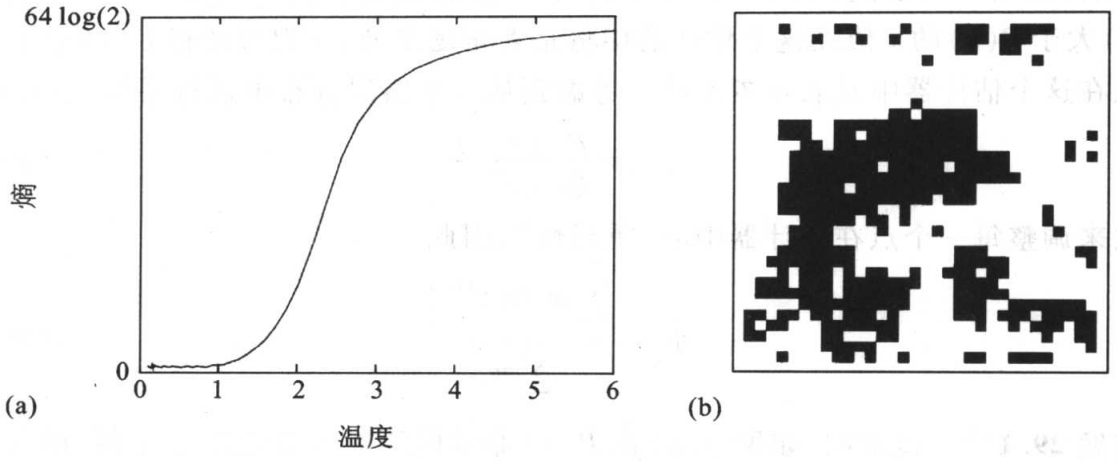


图 29.4 (a) 64 自旋伊辛模型的熵,以温度的函数表示。(b) 1 024 自旋伊辛模型的一个状态。

29.2 重要性采样

重要性采样不是关于如何从 $P(x)$ 生成样本(问题 1)的一种方法;它只是估计函数 $\phi(x)$ 的期望值(问题 2)的一种方法。可以将它看做是对均匀采样方法的推广。

为了便于阐述,设想目标分布是一个一维密度函数 $P(x)$。假设可以计算出该密度函数在任意选择点 x 处的值,至少是在一个常数因子的范围内可以计算出来;因此,可以计算函数值 $P^*(x)$ 使得

$$P(x) = P^*(x)/Z \tag{29.20}$$

但是,函数 $P(x)$ 过于复杂,不能直接从它那里抽取样本。现在假设有一个简单些的密度函数 $Q(x)$,可以供我们从中抽取样本,并可以在一个常数因子范围内计算函数值(也就是说,可以计算 $Q^*(x)$ 的值,而 $Q(x) = Q^*(x)/Z_Q$)。函数 P^*、Q^* 和 ϕ 的一个例子如图 29.5 所示。我们称 Q 为采样器密度。

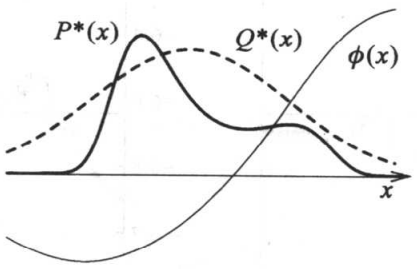


图 29.5 重要性采样所涉及的函数。希望在 $P(x) \propto P^*(x)$ 的意义下估计 $\phi(x)$ 的期望值。可以从更简单的分布 $Q(x) \propto Q^*(x)$ 中生成样本。在任何点上可以计算 Q^* 和 P^*。

在重要性采样中,从 $Q(x)$ 中生成 R 个样本 $\{x^{(r)}\}_{r=1}^{R}$。如果这些点是从 $P(x)$ 中

产生的，那么可以通过式(29.6)来估计 Φ。但是当我们由 Q 生成样本，那些使得 $Q(x)$ 大于 $P(x)$ 的 x 值在这个估计器中将是表示过度的，而那些使得 $Q(x)$ 小于 $P(x)$ 的点在这个估计器中是表示不足的。考虑到从一个错误分布中进行采样，引入权重

$$w_r \equiv \frac{P^*(x^{(r)})}{Q^*(x^{(r)})} \tag{29.21}$$

用它来调整每一个点在估计器中的“重要性”，因此

$$\hat{\Phi} \equiv \frac{\sum_r w_r \phi(x^{(r)})}{\sum_r w_r} \tag{29.22}$$

▷ **习题 29.1**[2] 试证明，如果 $Q(x)$ 在 $P(x)$ 非零的所有 x 处也取非零值，随着 R 值的增大，估计器 $\hat{\Phi}$ 收敛到 $\phi(x)$ 的均值 Φ。这个估计器的渐近方差是多少？[提示：分别考虑分子和分母的统计特性。]对于较小的 R 值，估计器 $\hat{\Phi}$ 是否是无偏倚估计器？

重要性采样的一个实际困难在于很难估计估计器 $\hat{\Phi}$ 有多可靠。该估计器的方差事先是未知的，因为它取决于一个涉及 $P^*(x)$ 的函数在 x 上的积分。$\hat{\Phi}$ 的方差难以估计，因为参量 w_r 和 $w_r\phi(x^{(r)})$ 的经验方差不一定对式(29.22)的分子和分母的真实方差有多好的指导意义。如果 $|\phi(x)P^*(x)|$ 在某个区域很大，而提议的密度 $Q(x)$ 在这个区域很小，那么很可能即使生成了多个样本点 $x^{(r)}$，仍然没有一个点会落在这个区域内。在这种情况下，Φ 的估计值会错得很厉害，经验方差也不会表明估计器 $\hat{\Phi}$ 的实际方差很大。

重要性采样的警戒性说明

对于用一维变量 x 建模氨基酸概率分布的一个简化问题，通过使用重要性采样计算一个感兴趣的参量。使用高斯采样器和柯西采样器所得结果如图 29.6 所示。横轴以对数坐标表示样本数。对于使用高斯采样器的情形，在计算了 500 个样本之

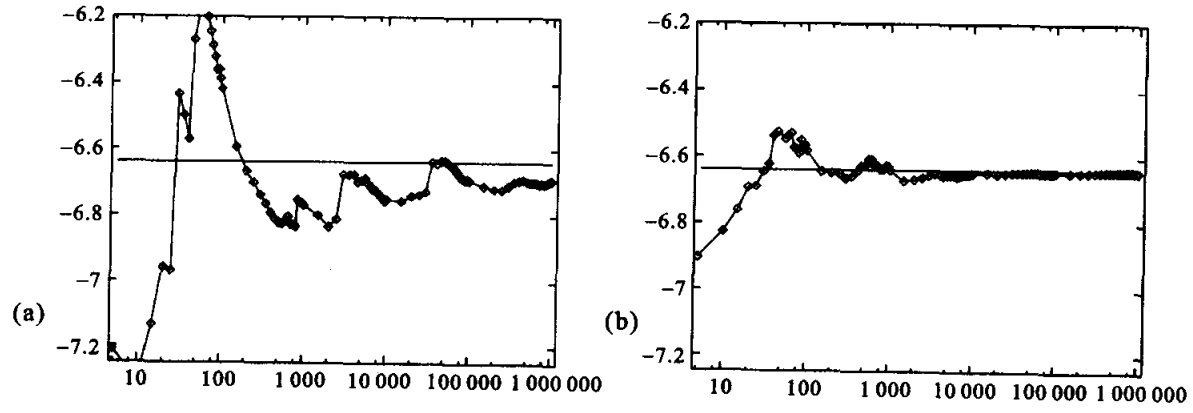


图 29.6 执行中的重要性采样：(a) 采用高斯采样器密度；(b) 采用柯西采样器密度。纵轴表示估计值 $\hat{\Phi}$。水平线表示 Φ 的真实值。横轴以对数坐标表示样本数。

后，人们可能会产生停止工作的念头；但是显然还有些不常出现的样本，会对$\hat{\Phi}$有较大的贡献，而在第 500 个样本处的估计值是错误的。即使在抽取 100 万个样本之后，估计值还是没有稳定在真实值附近。与此相反，柯西采样器不会出现这种闪烁不定的情况；在 5 000 个样本之后它就收敛了（在这里显示的幅度范围内）。

这个例子表明一个重要性采样器函数应该具有**大的拖尾**。



习题 29.2[2] 考虑 $P^*(x)$是多峰函数且包括几个相隔较远的峰的情况，如图 29.7 所示。（这样的概率分布经常出现在统计数据建模中。）如果利用和其中一个峰相拟合的单峰采样器分布函数 $Q(x)$，进行重要性采样，讨论采取这样的策略是否明智。$\phi(x)$是其均值 Φ 要被估计的函数，假设它是随 x 平缓变化的一个函数，就像函数 $mx+c$ 一样。描述一下估计器$\hat{\Phi}$作为样本数 R 的函数的典型演化过程。

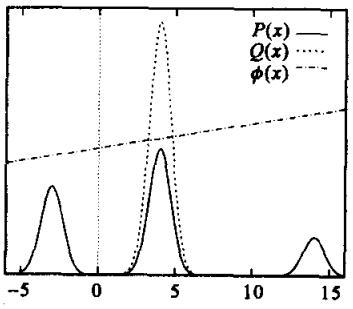


图 29.7 多峰分布 $P^*(x)$和单峰采样器 $Q(x)$

在多维空间中的重要性采样

已经观察到，解决一维重要性采样问题时需要很谨慎。那么在高维空间中，比如维度 $N=1\,000$的空间中，重要性采样是否是一种有用的方法？

考虑一个简单的案例：目标密度 $P(x)$是一个球内的均匀分布

$$P^*(x)=\begin{cases}1, & 0\leqslant\rho(x)\leqslant R_P\\ 0, & \rho(x)>R_P\end{cases} \tag{29.23}$$

其中 $\rho(x)\equiv\left(\sum_i x_i^2\right)^{1/2}$，提议的密度函数是中心在原点的高斯函数

$$Q(x)=\prod_i \text{Normal}(x_i;0,\sigma^2) \tag{29.24}$$

如果估计器$\hat{\Phi}$中有几个占主导地位的大权重 w_r，运用重要性采样方法时会遇到麻烦。权重 w_r的典型数值范围是什么？从本书第一部分所讨论的典型序列[比如，可以参见习题 6.14]可知，如果 ρ 是从原点到 Q 所产生的一个样本的距离，量 ρ^2大致服从高斯分布，其均值和标准方差为

$$\rho^2\sim N\sigma^2\pm\sqrt{2N}\sigma^2 \tag{29.25}$$

因此，几乎 Q 产生的所有样本都位于到原点距离很接近$\sqrt{N}\sigma$ 的一个典型集内。假设选择 σ 使得 Q 的典型集位于一个半径是 R_P的球体内。[如果不在里面，大数定理表明几乎所有由 Q 生成的样本都会落在 R_P之外，且具有零权值。]于是，可以知道 Q 所产生的大多数样本会有一个在如下范围内的 Q 值。

$$\frac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\frac{N}{2}\pm\frac{\sqrt{2N}}{2}\right) \tag{29.26}$$

因此，权重 $w_r = P^*/Q$ 典型的数值范围为

$$(2\pi\sigma^2)^{N/2}\exp\left(\frac{N}{2} \pm \frac{\sqrt{2N}}{2}\right) \tag{29.27}$$

那么，如果抽取 100 个样本，典型的权重范围是什么？可以通过将式(29.27)中的标准差加倍，从而大致估计出最大权重和中值权重之比。最大权重和中值权重的典型比值是

$$\frac{w_r^{\max}}{w_r^{\text{med}}} = \exp(\sqrt{2N}) \tag{29.28}$$

因此，在维度 $N = 1\,000$ 的空间里，经过 100 次采样之后，最大的权重可能比中值权重大约多出 10^{19} 倍。所以，高维问题的一个重要性采样估计很可能完全被几个具有巨大权重的样本所主导。

总之，高维空间中的重要性采样经常会遇到两个难题。首先，需要得到位于 P 的典型集内的样本，这可能需要很长一段时间，除非 Q 能够很好地近似 P。其次，即使得到了典型集中的样本，与这些样本相关联的权重有可能相差较大的因子，因为尽管典型集中各点的概率彼此类似，但其差别还是可能达到 $\exp(\sqrt{N})$ 这一量级，权重之间也会相差这么多，除非 Q 能够近乎完美地近似 P。

29.3 舍弃采样

再次假设一维密度函数 $P(x) = P^*(x)/Z$ 过于复杂，不能直接从它那里抽取样本。假设有一个简单些的提议密度函数 $Q(x)$，可以计算它的函数值(和以前一样，是在乘法因子 Z_Q 的范围之内)，并可从中生成样本。进一步假设已知常数 c 的值，从而

$$cQ^*(x) > P^*(x), \quad \text{对于所有 } x \tag{29.29}$$

这两个函数的示意图如图 29.8(a)所示。

我们生成两个随机数。第一个数 x 由提议密度 $Q(x)$ 生成。然后计算 $cQ^*(x)$，并在区间 $[0, cQ^*(x)]$ 上生成一个均匀分布随机变量 u。可以将这两个随机变量看做从一个二维平面上选择一个点，如图 29.8(b)所示。

现在计算 $P^*(x)$，并将 $P^*(x)$ 的值与 u 值做比较，决定究竟接收或舍弃样本 x。如果 $u > P^*(x)$，就舍弃 x；否则就接收它，这意味着我们将 x 添加到样本集 $\{x^{(r)}\}$。丢弃 u 值。

为什么这个过程会从 $P(x)$ 生成样本？如图 29.8(b)所示，被提出的点 (x, u) 以均匀概率出自曲线 $cQ^*(x)$ 下面的浅色阴影区域。舍弃规则抛弃所有位于曲线 $P^*(x)$ 之上的点。因此，所有被接收的点都均匀地分布在曲线 $P^*(x)$ 之下的深色阴影区域。这意味着所有被接收点的 x 坐标的概率密度必须正比于 $P^*(x)$，因此这些

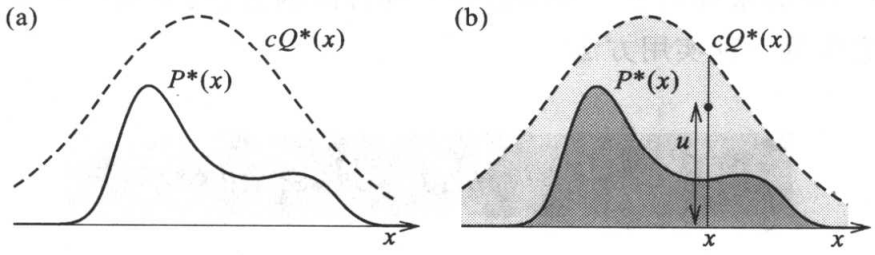


图 29.8 舍弃采样。(a) 舍弃采样所涉及的函数。希望得到 $P(x) \propto P^*(x)$ 产生的样本。可以从 $Q(x) \propto Q^*(x)$ 抽取样本，并且已知一个 c 值，对于所有 x 都满足 $cQ^*(x) > P^*(x)$。(b) 在曲线 $cQ^*(x)$ 下面的浅色阴影区随机地生成一个数据点 (x,u)。如果该点同时还落在 $P^*(x)$ 之下，就接收它。

样本必须和 $P(x)$ 相互独立。

如果 Q 是 P 的一个很好的近似，舍弃采样法最受用。如果 Q 和 P 很不一样，那么为了满足 cQ 在任何地方都大于 P，c 必须很大，因此舍弃频率也很高。

多维空间中的舍弃采样

在高维问题中，很可能 cQ^* 是 P^* 的上界这一要求会迫使 c 取巨大的值，而这又会使得接收事件变得非常稀少。求出这样一个 c 值可能也很难，因为在许多问题中，既不知道 P^* 的峰值在哪里，也不知道峰值有多高。

作为案例研究，考虑一对均值为零的 N 维高斯分布(如图 29.9 所示)。设想从其中一个标准差为 σ_Q 的分布生成样本，并采用舍弃采样来获得另一分布的样本，其标准差为 σ_P。假设这两个标准差在数值上很接近，比如 σ_Q 只比 σ_P 大 1%。[σ_Q 一定要比 σ_P 大，因为如果不是这样，就不存在 c 满足：对于所有 x 都有 cQ 大于 P。]那么，如果维度 $N = 1\,000$，要求 c 值多大？在原点处，$Q(x)$ 的密度是 $1/(2\pi\sigma_Q^2)^{N/2}$，因此为了满足 cQ 大于 P，需要设定

$$c = \frac{(2\pi\sigma_Q^2)^{N/2}}{(2\pi\sigma_P^2)^{N/2}} = \exp\left(N\ln\frac{\sigma_Q}{\sigma_P}\right) \tag{29.30}$$

对于 $N = 1\,000$ 和 $\dfrac{\sigma_Q}{\sigma_P} = 1.01$，得出 $c = \exp(10) \simeq 20\,000$。对于这样一个 c 值，接收率是多少？答案是可以直接得到的：因为接收率等于曲线 $P(x)$ 下方的体积和 $cQ(x)$ 下方的体积之比；而 P 和 Q 都是归一化的这一事实就意味着接收率将是 $1/c$，例如1/20 000。通常，c 会随着维度 N 呈指数级增长，因此可以预见接收率会随着 N 呈指数级降低。

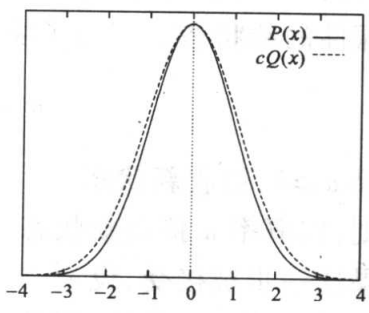


图 29.9 高斯函数 $P(x)$ 和一个比它略微宽一点的高斯函数 $Q(x)$，放大因子 c 倍后满足 $cQ(x) \geq P(x)$。

因此，虽然舍弃采样对于一维问题是一种有用的方法，别期望高维分布 $P(x)$ 生成样本会以它作为一种实用方法。

29.4 Metropolis - Hastings 方法

重要性采样和舍弃采样方法只有在提议密度 $Q(x)$ 和 $P(x)$ 类似时，才会起作用。在大型复杂问题中，很难构造具有这种性质的一个密度函数 $Q(x)$。

Metropolis - Hastings 算法则使用与当前状态 $x^{(t)}$ 有关的提议密度 Q。密度 $Q(x';x^{(t)})$ 可以是一个简单分布，比如以当前状态 $x^{(t)}$ 为中心的高斯函数。提议密度 $Q(x';x)$ 可以是任何可从中抽取样本的固定密度。与重要性采样和舍弃采样不同，$Q(x';x^{(t)})$ 完全不必同 $P(x)$ 类似，这种算法也可以实用。如图 29.10 所示是提议密度的一个例子；该图画出了两个不同状态 $x^{(1)}$ 和 $x^{(2)}$ 所对应的密度 $Q(x';x^{(t)})$。

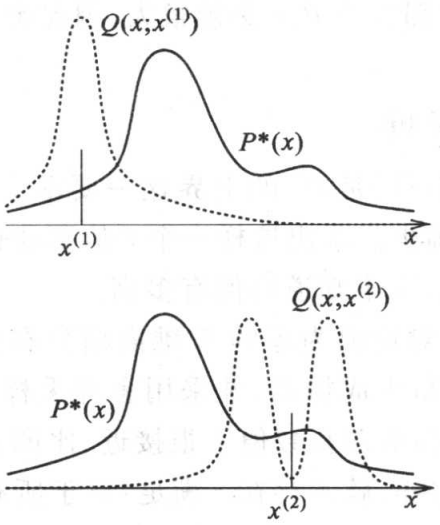


图 29.10 在一维空间中的 Metropolis - Hastings 方法。这里所示的提议分布 $Q(x';x)$ 会随着 x 的变化而改变形状，尽管在实际中所使用的提议密度通常并非如此。

与以前一样，假设对于任何 x 可以计算出 $P^*(x)$。由提议密度 $Q(x';x^{(t)})$ 生成一个试探性的新状态 x'。为了决定是否接收这个新状态，计算下面这个数量

$$a = \frac{P^*(x')}{P^*(x^{(t)})} \frac{Q(x^{(t)};x')}{Q(x';x^{(t)})} \tag{29.31}$$

如果 $a \geqslant 1$，接收新状态。

否则，以概率 a 接收新状态。

如果这一步被接受，设 $x^{(t+1)} = x'$。

如果这一步遭到拒绝，则设 $x^{(t+1)} = x^{(t)}$。

注意与舍弃采样不同：在舍弃采样中，被舍弃的点被抛弃，对我们所收集的这一列样本 $\{x^{(r)}\}$ 没有影响。这里，一次舍弃就会将当前状态重新写进样本列中。

记号:笔者使用上标 $r=1,\cdots,R$ 标识来自一个分布的独立样本的点,上标 $t=1,\cdots,T$ 标识马尔科夫链的状态序列。很重要的一点是,要注意 Metropolis – Hastings 所仿真的 T 次迭代并未从目标分布 P 产生 T 个独立样本。样本是相关的。

要计算接收概率(29.31),要能够计算概率比 $P(x')/P(x^{(t)})$ 和 $Q(x^{(t)};x')/Q(x';x^{(t)})$。如果提议密度是一个简单的对称密度,比如以当前点为中心的高斯函数,那么后一个因子是 1,Metropolis – Hastings 方法就只涉及比较目标密度在两个点处的值。这一特例有时也称作 Metropolis 方法。但是,要对 Hastings 抱歉的是,我要称非对称 Q 的通用 Metropolis – Hastings 方法为"Metropolis 方法",因为我相信重要的思想应配以较短的名称。

Metropolis 方法收敛到目标密度

可以证明,对于任何正的 Q(也就是说,对所有 x,x' 都有 $Q(x';x)>0$ 的任何 Q),当 $t\to\infty$ 时,$x^{(t)}$ 的概率分布趋向于 $P(x)=P^*(x)/Z$。[不应将这一命题理解为:Q 必须对每个点 x' 都赋予一个正概率,在后面讨论的例子中可以看到,对某些 x 和 x' 也可以有 $Q(x';x)=0$;还应注意,我们尚未对以多快的速度收敛到 $P(x)$ 做任何表示。]

Metropolis 方法是马尔科夫链蒙特卡罗方法(缩写为 MCMC)的一个例子。舍弃采样接收的点都来自理想分布的独立样本,与此不同,马尔科夫链蒙特卡罗方法涉及一个马尔科夫过程,该过程生成一系列状态 $\{x^{(t)}\}$,而每个样本 $x^{(t)}$ 的概率分布都和前一个值 $x^{(t-1)}$ 相关。因为相邻样本是相关的,可能需要让马尔科夫链运行相当长的一段时间,以便所生成的样本是来自 P 的、实际上独立的样本。

正如很难估计一个重要性采样估计器的方差,也同样很难评估一个马尔科夫链蒙特卡罗方法是否已经"收敛",并量化要等多久才能获得来自 P 的、实际上独立的样本。

Metropolis 方法的演示

Metropolis 方法在高维问题中得到了广泛的应用。Metropolis 方法的许多实现都采用了这样一种提议分布:相对于可能区域的最长长度尺度 L,其长度尺度 ε 较短(如图 29.11 所示)。选用小的长度尺度的一个原因是,对于大多数高维问题,从一个典型点(即来自 $P(x)$ 的样本)随机地跨一大步,很可能会进入一个具有很低概率的状态;这样的步长不太可能被接受。如果 ε 较大,只有当这样一个到低概率状态的转换被实际接受,或当随机地跨一大步碰巧落在另一个可能状态上时,状态空间中才有可能发生运动。因此,如果采用大步长,进展速率反而会慢。

另一方面,小步长的弊端在于:Metropolis 方法用随机游动探索概率分布,而随机游动要花比较长的时间才能到达其他地方,特别是当漫步的步长很小时。



习题 29.3[1] 考虑一个一维随机游动,每一步的状态以等概率随机地向左边或

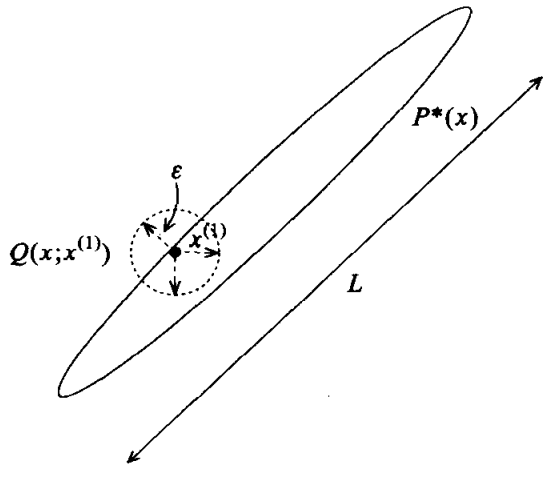


图 29.11 二维 Metropolis 方法，表明了有足够小步长 ϵ 的传统提议密度，其接受频率约为 0.5。

右边移动。证明 T 次大小为 ϵ 的漫步之后，状态可能运动的距离约为$\sqrt{T}\epsilon$。（计算途经距离的均方根。）

回顾一下，蒙特卡罗采样的第一个目标是从已知分布生成一些独立样本（比如，一打）。如果状态空间的最大长度尺度是 L，那么必须仿真随机游动 Metropolis 方法 $T \simeq (L/\epsilon)^2$ 这么长时间，才能期望得到一个和初始条件大致独立的样本——这里还假设每一步都被接受：如果平均而言，只有比例 f 这一部分被接受，那么这个时间还要增加一个因子 $1/f$。

经验法则：Metropolis 方法迭代次数的下限。如果可能状态的空间的最大长度尺度为 L，提议分布所生成的随机漫步的步长为 ϵ，这个 Metropolis 方法必须至少运行

$$T \simeq (L/\epsilon)^2 \tag{29.32}$$

次迭代，以获得一个独立样本。

这条经验法则只给出了一个下限；情况可能会坏得多，比如，若概率分布是由几个被低概率区域所分隔开的高概率的孤岛组成，情况便会很坏。

为了阐明随机游动探索状态空间有多慢，如图 29.12 所示为 Metropolis 算法的一个仿真，用以从如下分布生成样本。

$$P(x) = \begin{cases} 1/21, & x \in \{0,1,2,\cdots,20\} \\ 0, & \text{其他} \end{cases} \tag{29.33}$$

提议分布为

$$Q(x';x) = \begin{cases} 1/2, & x' = x \pm 1 \\ 0, & \text{其他} \end{cases} \tag{29.34}$$

因为目标分布 $P(x)$ 是均匀的，只有当提议将状态转到 $x' = -1$ 或 $x' = 21$ 时，才会发生舍弃。

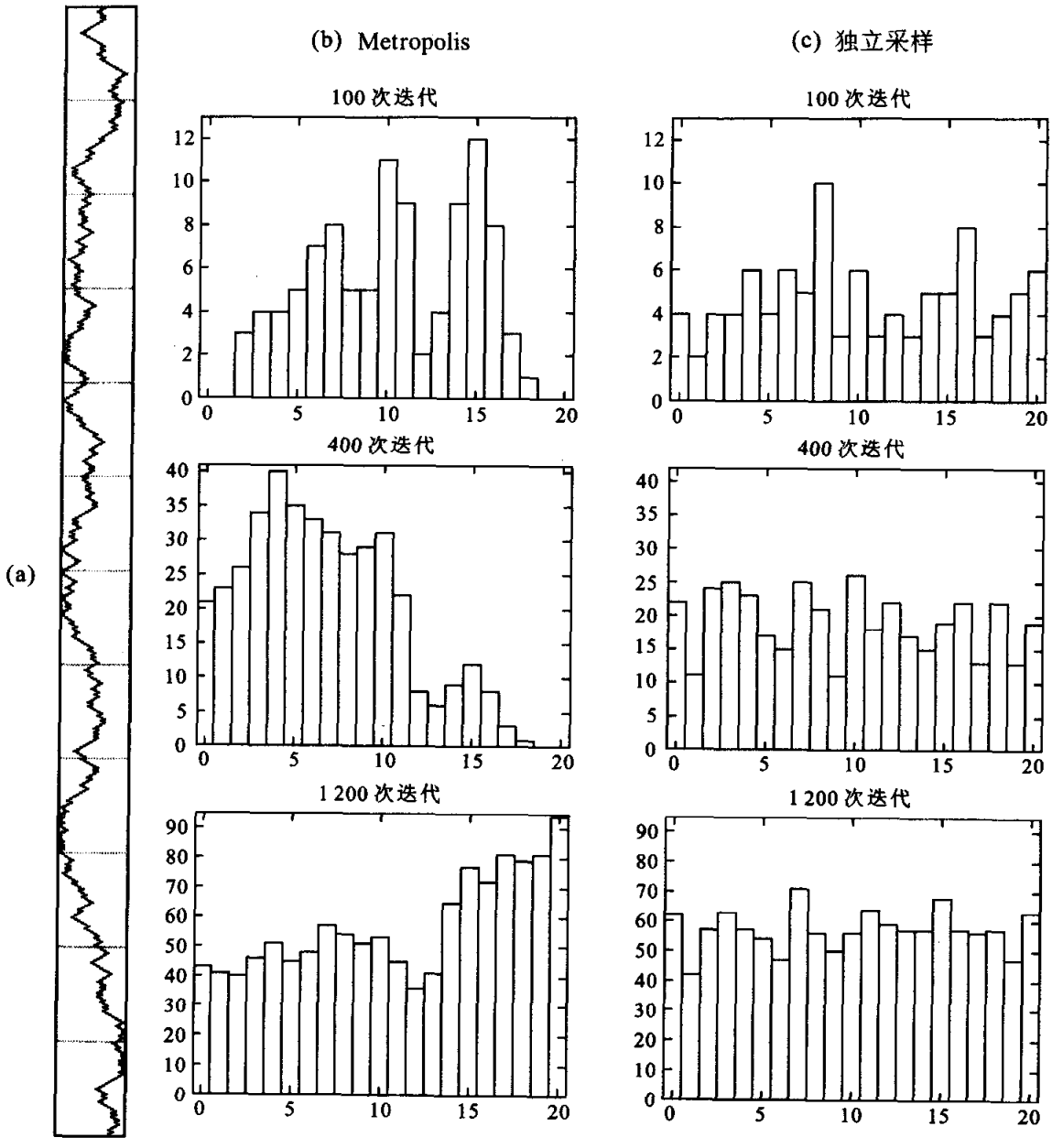


图 29.12 用于一个简化问题的 Metropolis 方法。(a) $t=1,\cdots,600$ 时的状态序列。水平方向 = 状态 0 到 20;垂直方向 = 时间 1 到 600;横隔线表示 50 次迭代的时间区间。(b) 在100、400 和 1 200 次迭代之后,状态占有情况直方图。(c) 为了便于比较,从目标分布中独立地抽取连续点时所得的直方图。

仿真起始于状态 $x_0=10$,其演化过程如图 29.12(a)所示。要过多久才能到达两个终端状态 $x=0$ 和 $x=20$ 之一呢? 因为距离是 10 步远,经验法则(29.32)预测通常需要 $T\simeq 100$ 次迭代才能到达一个终端状态。在现在这个例子中证实了这一点:第一次进入终端状态是在第 178 次迭代时。那么要多久才能访问到两个终端状态? 经验

法则预测大约需要400次迭代才能穿越整个状态空间;实际上,第一次碰到另一个终端状态是在第540次迭代时。对于每个独立样本而言,只有仿真约400次迭代才能生成实际独立的样本。

这个简单例子表明在蒙特卡罗方法中抛弃随机游动行为很关键。使用同样的步长,对这个简化状态空间$\{0,1,2,\cdots,20\}$做一次系统的探索,可以用20步而非400步就遍历所有状态。降低随机游动行为的方法将在下一章进行讨论。

高维空间中的 Metropolis 方法

经验法则(29.32)给出了随机游动 Metropolis 方法迭代次数的下限,它也适用于更高维的问题。考虑一种简单情况:目标分布为 N 维高斯分布,提议分布是在每个方向标准差为 ε 的球面高斯。不失一般性,可以假设可分离的分布和轴$\{x_n\}$对齐,在方向 n 上的标准差为 σ_n。令 $\sigma^{\max}$和 $\sigma^{\min}$分别是这些标准差中的最大值和最小值。假设调整 ε 使得接收频率接近1。在这个假设条件下,每个变量 x_n 独立于其他变量进行演化,执行一个步长约为 ε 的随机游动。从目标分布生成实际独立样本所需时间将由最大长度尺度 $\sigma^{\max}$所控制。在前一节,需要至少 $T\simeq(L/\varepsilon)^2$ 次迭代以获得独立样本,与之类似,这里需要 $T\simeq(\sigma^{\max}/\varepsilon)^2$ 次。

现在要问,ε 可以是多大?它越大,T 这个数就越小,但是如果 ε 太大(大于 $\sigma^{\min}$),那么接收率会急剧下降。看来好像最优 ε 必须和 $\sigma^{\min}$相仿。严格来说,这可能不正确;如果第二小的 σ_n 比 $\sigma^{\min}$大得多,那么在这一特殊情况下,最优 ε 可能会和第二小的 σ_n 接近。但是我们的粗略结论是:使用简单球面提议分布时,至少需要 $T\simeq(\sigma^{\max}/\sigma^{\min})^2$ 次迭代来获得一个独立样本,这里 $\sigma^{\max}$和 $\sigma^{\min}$是目标分布的最长和最短长度尺度。

这既是好消息又是坏消息。之所以是好消息,因为不像舍弃采样和重要性采样的情况,这种方法与维度 N 没有灾难性的依存关系。我们的计算机会在一个比宇宙年龄短的时间内给出有用的答案。而坏消息都是一样的,因为这个对长度尺度比的二次依存关系还是可能迫使我们进行长时间的仿真。

幸运的是,有些方法可以压制蒙特卡罗仿真中的随机游动,这将留到下一章进行讨论。

29.5 Gibbs 采样

通过一维例子已介绍了重要性采样、舍弃采样和 Metropolis 方法。Gibbs 采样,又叫*热浴方法*或"Glauber 动力学",是从至少二维的分布采样的一种方法。Gibbs 采样可以看做一种 Metropolis 方法,只是其中的一系列提议分布 Q 以联合分布 $P(x)$ 的条件分布定义。假设虽然 $P(x)$过于复杂,无法直接对它抽取样本,但它的条件分布

$P(x_i \mid \{x_j\}_{j\neq i})$则比较容易处理。对于许多图模型(但并非所有的),往往可以很容易对这些一维条件分布进行采样。例如,如果一些变量 $\boldsymbol{d}$ 服从高斯分布,其均值 $\boldsymbol{m}$ 未知,但 $\boldsymbol{m}$ 的先验分布是高斯分布,那么已知 $\boldsymbol{d}$ 时,$\boldsymbol{m}$ 的条件分布也是高斯的。如果条件分布满足一定的凸特性,即使它不具有标准形式,还是可能通过自适应舍弃采样进行采样(Gilks 和 Wild,1992 年)。

如图 29.13 所示是在一个双变量$(x_1, x_2) = x$ 情况下的 Gibbs 采样。在每一次迭代,从当前状态 $x^{(t)}$ 出发,将 x_2 固定在 $x_2^{(t)}$,从条件密度 $P(x_1 \mid x_2)$对 x_1 进行采样。然后使用新的 x_1 值,从条件密度 $P(x_2 \mid x_1)$对 x_2 进行采样。这会将我们代入一个新状态 $x^{(t+1)}$,并完成本次迭代。

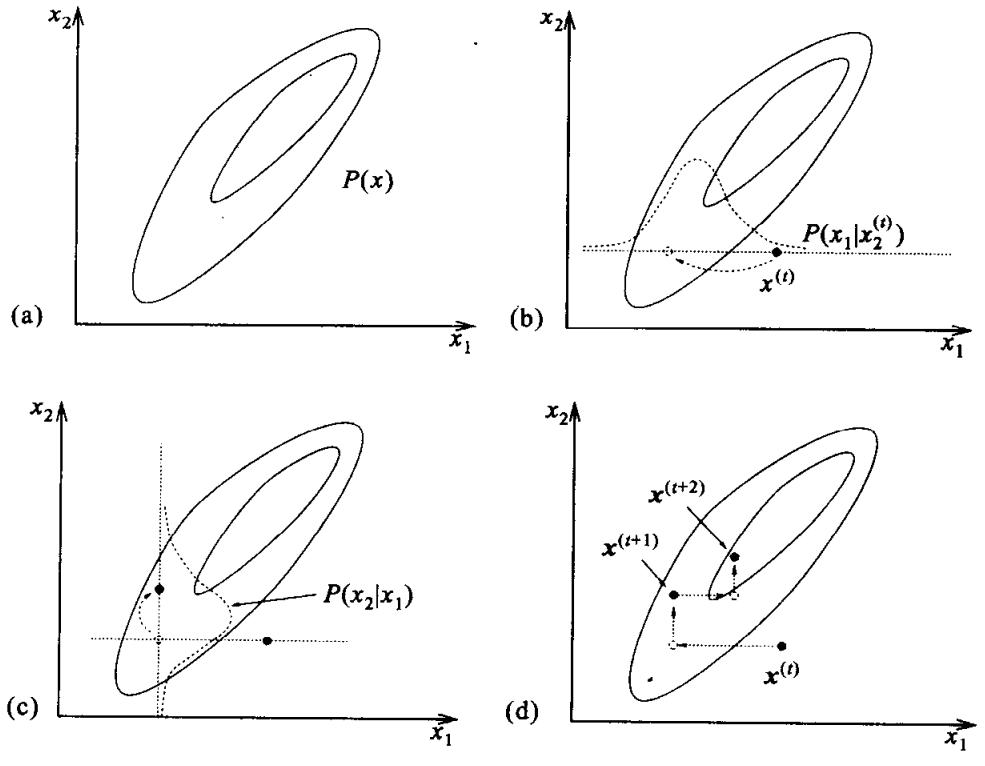


图 29.13 Gibbs 采样。(a) 需要采得样本的联合概率密度 $P(x)$。(b)从状态 $x^{(t)}$ 开始,从条件概率密度 $P(x_1 \mid x_2^{(t)})$中采样 x_1。(c) 然后从条件概率密度 $P(x_2 \mid x_1)$中构造一个样本。(d) Gibbs 采样的一对迭代。

对于一般情况,假设一个系统有 K 个变量,一次迭代只对一个参数进行采样:

$$x_1^{(t+1)} \sim P(x_1 \mid x_2^{(t)}, x_3^{(t)}, \cdots, x_K^{(t)}) \tag{29.35}$$

$$x_2^{(t+1)} \sim P(x_2 \mid x_1^{(t+1)}, x_3^{(t)}, \cdots, x_K^{(t)}) \tag{29.36}$$

$$x_3^{(t+1)} \sim P(x_3 \mid x_1^{(t+1)}, x_2^{(t+1)}, \cdots, x_K^{(t)}), \text{等等} \tag{29.37}$$

Gibbs 抽样到目标密度的收敛

▷**习题 29.4**[⇗] 证明 Gibbs 采样的一个单变量更新可以看做目标密度为 $P(x)$ 的 Metropolis 方法，而且该 Metropolis 方法具有这样一个特性：每个提议都会被接受。

因为 Gibbs 采样是一种 Metropolis 方法，只要 $P(x)$ 不具有病态特性，当 $t \to \infty$ 时，$x^{(t)}$ 的概率分布会趋向于 $P(x)$。

▷**习题 29.5**[⇗] 讨论(7,4)汉明码的校正子译码问题是否可以用 Gibbs 采样进行求解。如果要用蒙特卡罗方法求解校正子译码问题，要采样噪声向量 $\boldsymbol{n} = (n_1, \cdots, n_n, \cdots, n_N)$ 的后验分布

$$P(\boldsymbol{n} \mid \boldsymbol{f}, \boldsymbol{z}) = \frac{1}{Z} \prod_{n=1}^{N} f_n^{n_n} (1 - f_n)^{(1 - n_n)} \mathbb{1}[\boldsymbol{Hn} = \boldsymbol{z}] \tag{29.38}$$

其中，f_n 是第 n 个发送比特的归一化似然，而 $\boldsymbol{z}$ 是观测到的校正子。如果 $\boldsymbol{n}$ 有正确的校正子 $\boldsymbol{z}$，则因子 $\mathbb{1}[\boldsymbol{Hn} = \boldsymbol{z}]$ 为 1，否则它就是 0。

对于任意线性纠错码的校正子译码问题，情况如何呢？

高维空间中的 Gibbs 采样

Gibbs 采样也和简单 Metropolis 算法具有同样的缺陷，即要用缓慢的随机游动来探索状态空间；除非能选择一个偶然的参数化，将概率分布 $P(x)$ 变为可分离的。比如，如果两个变量 x_1 和 x_2 强相关，具有宽度为 L 的边缘密度和宽度为 ε 的条件密度，那么至少需要约 $(L/\varepsilon)^2$ 次迭代才能生成目标密度的一个独立样本。如图 30.3 所示为 $L >> \varepsilon$ 时 Gibbs 采样所取得的缓慢进展。

但是，Gibbs 采样不涉及可调节的参数，因此在需要一个可以快速运转的模型时，它还是一种颇有吸引力的策略。一个优秀的软件包 BUGS 可用来很容易地设置几乎任意的概率模型，并用 Gibbs 采样对它们进行仿真(Thomas 等，1992 年)。①

29.6 马尔科夫链蒙特卡罗方法的术语

下面花点时间来勾勒 Metropolis 方法和 Gibbs 采样所基于的理论。将马尔科夫链仿真器状态的概率分布记做 $p^{(t)}(x)$。(为了使这个分布更形象化，设想并行运行无数个相同仿真器的集合。)我们的目标是找到这样一条马尔科夫链，使得当 $t \to \infty$ 时，$p^{(t)}(x)$ 会趋向于理想分布 $P(x)$。

① http://www.mrc-bsu.cam.ac.uk/bugs/

可以通过初始概率分布 $p^{(0)}(x)$ 和转移概率 $T(x';x)$ 来说明一条马尔科夫链。在第 $(t+1)$ 次迭代时，这条马尔科夫链的状态概率分布 $p^{(t+1)}(x)$ 为

$$p^{(t+1)}(x') = \int \mathrm{d}^N x T(x';x) p^{(t)}(x) \tag{29.39}$$

▷**习题 29.6** 第 29.4 节(图 29.12)的 Metropolis 演示系统给出了一个马尔科夫链的例子，其转移概率为

$$T = \begin{bmatrix}
1/2 & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & 1/2
\end{bmatrix}$$

初始概率分布为

$$p^{(0)}(x) = [\,*\,*\,*\,*\,*\,*\,*\,*\,*\,*\,1\,*\,*\,*\,*\,*\,*\,*\,*\,*\,*\,] \tag{29.40}$$

对于 $t=0,1,2,3,5,10,100,200,400$，第 t 次迭代所处状态的概率分布 $p^{(t)}(x)$ 如图 29.14 所示；如图 29.15 所示为一个马尔科夫链的等价分布序列，它起始于初始状态 $x_0=17$。当 $t\to\infty$ 时，两条链都收敛到目标密度：均匀密度。

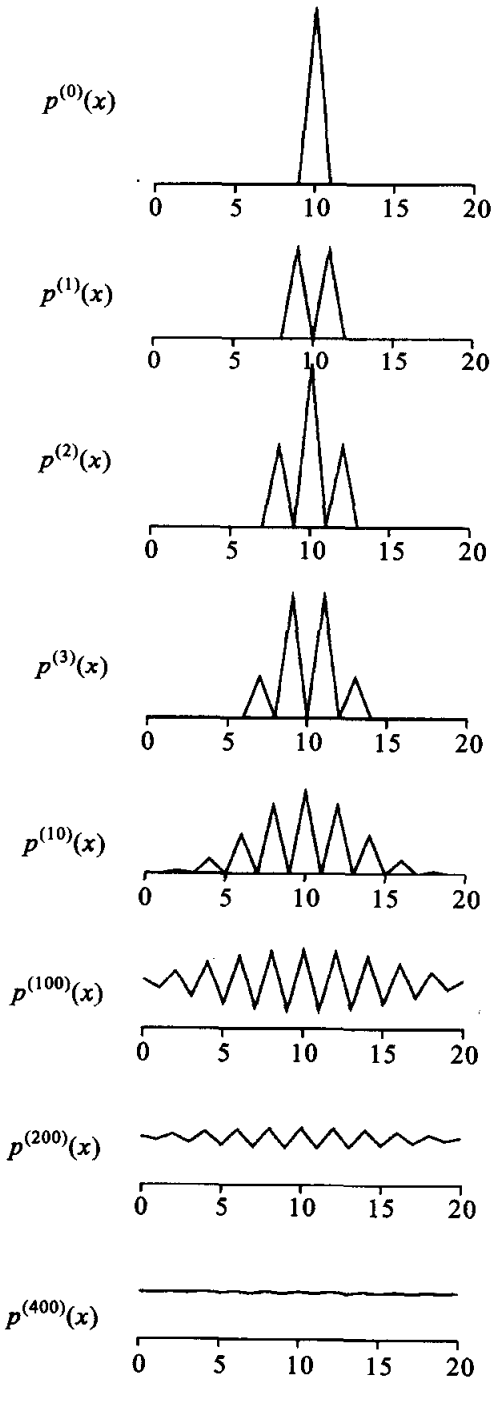


图 29.14 例 29.6 中马尔科夫链状态的概率分布

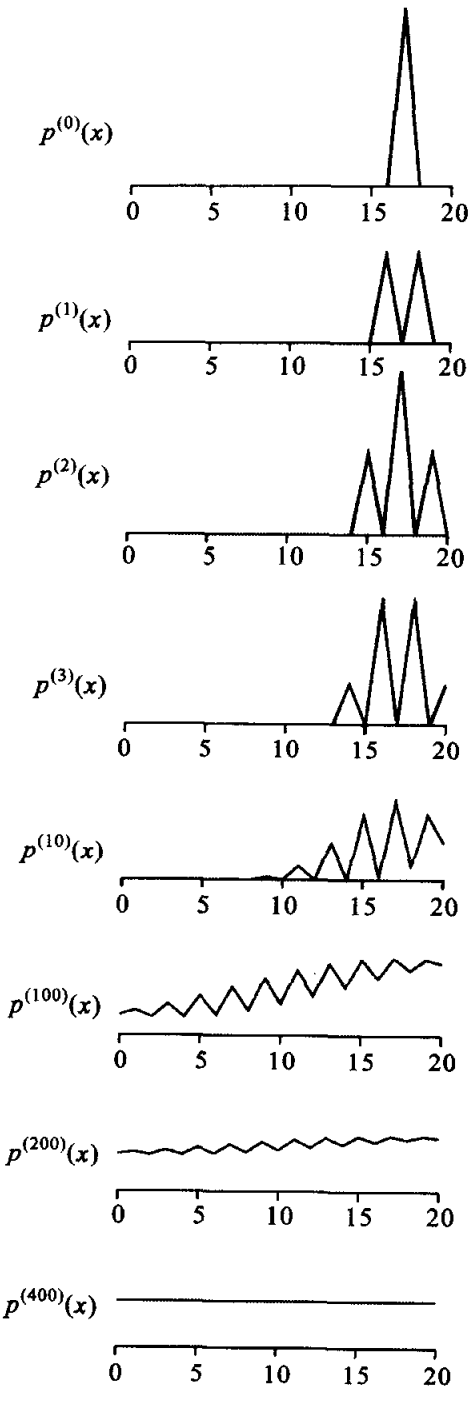


图 29.15 初始条件为 $x_0 = 17$ 时马尔科夫链状态的概率分布(例 29.6)

要求具有的特性

在设计一个马尔科夫链蒙特卡罗方法时,构造一条具有如下性质的链。

1. 理想的分布 $P(x)$是链的不变分布。一个分布 $\pi(x)$被称作转移概率 $T(x';x)$的不变分布,如果有

$$\pi(x') = \int \mathrm{d}^N x T(x';x)\pi(x) \tag{29.41}$$

一个不变分布是转移概率矩阵特征值 1 所对应的一个特征向量。

2. 这条链必须是各态历经的，也就是说

$$p^{(t)}(x) \to \pi(x), \quad \text{当 } t\to\infty \text{，对任意 } p^{(0)}(x) \text{ 成立} \tag{29.42}$$

一条链可能因为如下两个原因不具有各态历经性。

(a) 矩阵可能是可约的，也就是说状态空间包含两个或多个状态子集，其中一个子集的状态永远无法转移到另一个子集中去。这样一条链有很多不变分布；当 $t\to\infty$ 时，$p^{(t)}(x)$ 具体趋于哪一个不变分布，取决于初始条件 $p^{(0)}(x)$。

这样一条链的转移概率矩阵有多个等于 1 的特征值。

(b) 这条链可能有一个周期状态集，也就是说在某些初始条件下，$p^{(t)}(x)$ 不会趋向不变分布，而是进入一个周期性的极限环。

在 N 维超立方体上的随机游动就是具有这样一个性质的一条简单马尔科夫链。链 T 的状态从一个角到另一个随机选取的相邻角。这条链的唯一不变分布是在所有 2^N 个状态上的均匀分布，但是这条链不是各态历经的；它是周期性的，周期为 2：如果将状态分成奇数状态和偶数状态，注意到每一个奇数状态都处于偶数状态的包围中，反之亦然。因此，如果 $t=0$ 时的初始条件是偶数状态，那么在 $t=1$ 和所有其他奇数时刻，状态必定是奇数的，而在所有偶数时刻，状态必定是偶数的。

这样一条链的转移概率矩阵有多于一个幅度等于 1 的特征值。比如，立方体上的随机游动的特征值等于 +1 和 −1。

构造马尔科夫链的方法

对简单的基转移 B 进行混合或级联，通常可以很方便地构造马尔科夫链 T，这里所有 B 都满足：对于理想密度 $P(x)$ 有

$$P(x') = \int \mathrm{d}^N x B(x';x)P(x) \tag{29.43}$$

也就是说，它们都以理想密度作为不变分布。这些基转移本身不一定具有各态历经性。

T 是几个基转移 $B_b(x',x)$ 的混合，如果构造其状态转移的方式是：随机地选取其中的一个基转移，并允许它决定状态转移，即

$$T(x',x) = \sum_b p_b B_b(x',x) \tag{29.44}$$

其中，$\{p_b\}$是基转移上的概率分布。

T是两个基转移$B_1(x',x)$和$B_2(x',x)$的级联，如果使用B_1转移到一个中间状态x''，再使用B_2从状态x''转移到x'。

$$T(x',x) = \int \mathrm{d}^N x'' B_2(x',x'') B_1(x'',x) \tag{29.45}$$

详细平衡

许多有用的转移概率满足详细平衡特性

$$T(x_a;x_b)P(x_b) = T(x_b;x_a)P(x_a), \quad \text{对所有 } x_b \text{ 和 } x_a \text{ 成立} \tag{29.46}$$

这个方程表明，如果从目标密度P选取一个状态（通过魔法），并通过T转移到另一个状态，那么选取x_b并从x_b转移到x_a的可能性与选取x_a并从x_a转移到x_b是相同的。满足详细平衡特性的马尔科夫链称作*可逆*马尔科夫链。我们对详细平衡特性感兴趣的原因在于，详细平衡特性隐含了分布$P(x)$在马尔科夫链T下的不变性，而它又恰恰给出了链的概率分布会收敛到$P(x)$的必要条件。这种收敛性是进行MCMC仿真时想要具有的一个重要特性。

▷ **习题29.7**[2] 试证明详细平衡隐含了分布$P(x)$在马尔科夫链T下的不变性。

证明详细平衡成立通常是证明马尔科夫链蒙特卡罗仿真会收敛到理想分布的一个关键步骤。比如，Metropolis方法满足详细平衡。不过，详细平衡也不是一个必不可少的条件，而且在后面还会看到，不可逆马尔科夫链在实际中也很有用，因为它们可能会有不同的随机游动特性。

▷ **习题29.8**[2] 试证明，如果将两个满足详细平衡的基转移B_1和B_2级联在一起，式(29.45)所定义的T未必满足详细平衡。

▷ **习题29.9**[2] 如果几个变量都以一个确定的次序更新，Gibbs采样是否满足详细平衡？

29.7 切分采样

切分采样（Neal，1997a；Neal，2003年）是一种类似于舍弃采样、Gibbs采样和Metropolis方法的马尔科夫链蒙特卡罗方法。它可应用于任何Metropolis方法适用的场合，也就是说，适用于在任何点x处都可计算目标密度$P^*(x)$的任何系统；与简单Metropolis方法相比，它还具有独特的优势：对步长之类的参数选择更鲁棒。最简单的切分采样与Gibbs采样类似，它包含状态空间中的一维转移；但是它不要求可以很容易地对一维条件分布进行采样，也不像自适应舍弃采样那样要求它们具有任何凸特性。切分采样与舍弃采样类似，也是渐近地从$P^*(x)$曲线下的体积中抽取样本；但是它对上界函数没有要求。

笔者将这样描述切分采样：先给出一维采样算法的概貌，再给出形象化描述，包含那些让方法有效的细节。

切分采样的概貌

假设想对 $P(x) \propto P^*(x)$ 进行采样，其中 x 是一个实数。一个一维切分采样算法是这样一种方法：从曲线 $P^*(x)$ 下的一个二维点 (x,u) 转移到同一曲线下的另一个点 (x',u')，使得 (x,u) 的概率分布趋向于在曲线 $P^*(x)$ 下的区域内的均匀分布，而无论从什么初始点出发；这正如舍弃采样也产生曲线 $P^*(x)$ 下的区域内的均匀分布（第 29.3 节）。

一维切分采样算法的单次转移 $(x,u) \to (x',u')$ 包含如下步骤，其中的第 3 步和第 8 步还需要进一步细化。

```
1：计算 P*(x)
2：抽取纵坐标 u' ~ Uniform(0,P*(x))
3：构造一个包含 x 的水平区间(x_l,x_r)
4：loop{
5：      抽取 x' ~ Uniform(x_l,x_r)
6：      计算 P*(x')
7：      若 P*(x') > u',退出循环 4 ~ 9
8：      否则修改区间(x_l,x_r)
9：}
```

在第 3 步，有好几种构造区间 (x_l,x_r) 的方法；在第 8 步，也有好几种修改它的方法。关键是整个方法必须满足详细平衡，保证曲线 $P^*(x)$ 下方点 (x,u) 的均匀分布是不变的。

第 3 步的“迈出”法

用“迈出”法构造包含 x 的区间 (x_l,x_r) 时，以步长 w 迈出去，直至找到 P^* 小于 u 的端点 x_l 和 x_r。该算法如图 29.16 所示。

```
3a：抽取 r ~ Uniform(0,1)
3b：x_l := x - rw
3c：x_r := x + (1 - r)w
3d：while(P*(x_l) > u'){x_l := x_l - w}
3e：while(P*(x_r) > u'){x_r := x_r + w}
```

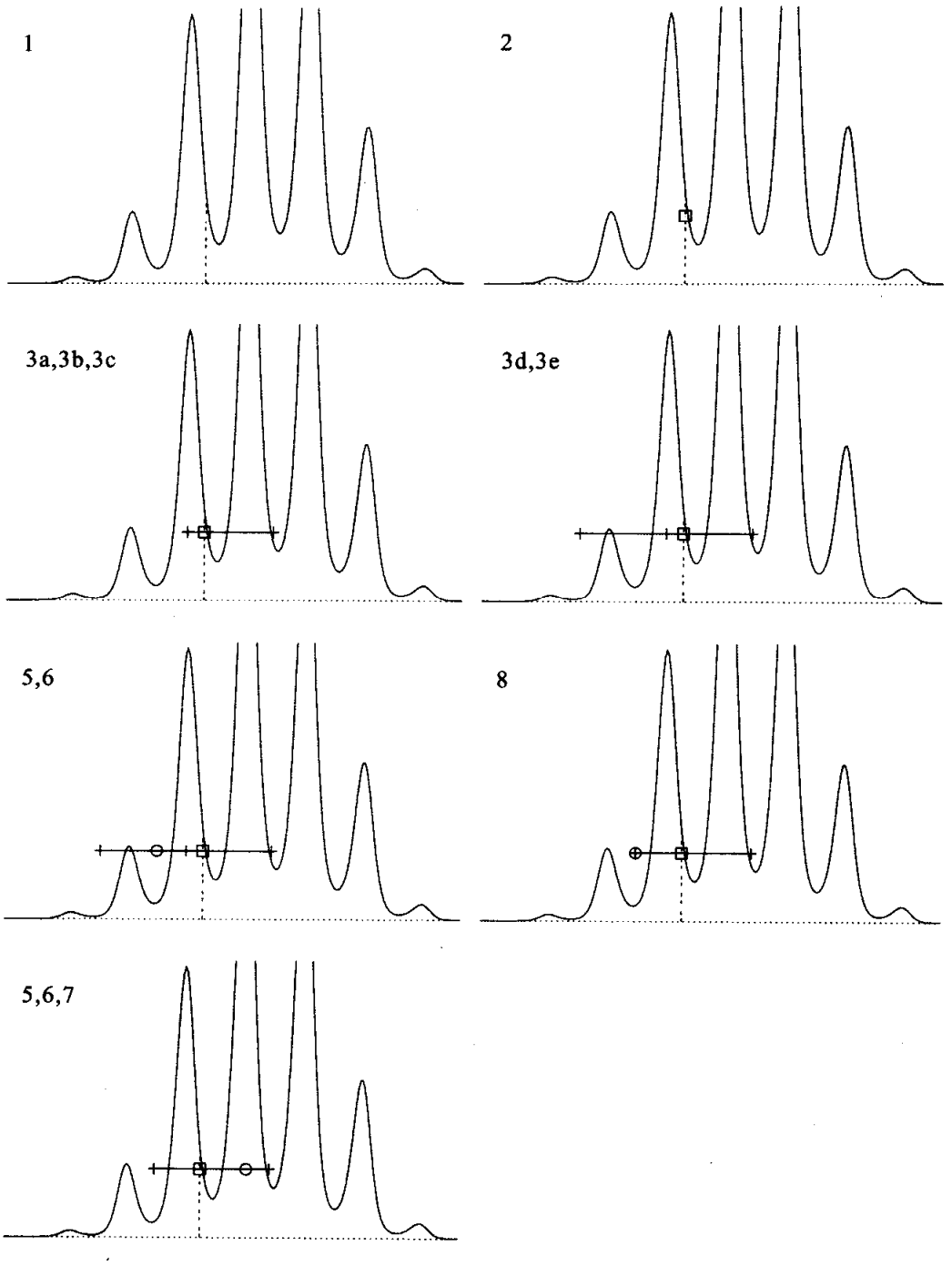


图 29.16 切分采样。每一栏都标上了正在执行算法的步骤号。在步骤 1,计算当前点 x 的 $P^*(x)$ 值。在步骤 2,给定图中小方块所示点 (x,u'),选择纵坐标。在步骤 3a ~ c,随机地构造一个大小为 w 且包含 (x,u') 的区间。在步骤 3d,计算 P^* 在区间左端点的值,并发现它大于 u',因此往左迈出大小为 w 的一步。在步骤 3e,计算 P^* 在区间右端点的值,并发现它小于 u',因此无需往右迈出。在重复步骤 3d 时,发现 P^* 小于 u',因此停止迈出。在步骤 5,从区间里抽出一个点,如图中小圆圈所示。步骤 6 确定该点在 P^* 上方,步骤 8 将区间缩小到被舍弃的点,并保证原来的点 x 仍然包含在区间内。在重复步骤 5 时,新坐标 x'(它在区间的右边)所给出的 P^* 值大于 u',因此点 x' 是第 7 步的结果。

第8步的“收缩”法

只要抽出的点 x' 使得 (x',u') 位于曲线 $P^*(x)$ 的上方，这样缩小区间：以 x' 为一个端点，并且保持原来的点 x 仍然包含在区间中。

```
8a:if(x' > x){x_r := x'}
8b:else{x_l := x'}
```

切分采样的性质

与标准 Metropolis 方法一样，切分采样也用随机游动在状态空间运动，但是在 Metropolis 方法中，步长的选择对仿真进展速率而言很关键，而在切分采样中，步长则是自行调节的。如果初始区间的大小 w 太小，与可能区域的宽度相比小了一个因子 f，那么迈出过程就会扩展区间的大小。迈出过程的开销只是 f 的线性函数，但是在 Metropolis 方法中，在步长太小时，计算机机时依 f 的平方增长。

如果所选择的 w 值太大，大出一个因子 F，那么算法将区间缩小到正确大小所花费的时间与 F 的对数成正比，因为每舍弃一个点时，区间大致会按因子 0.6 缩小。与此不同，Metropolis 算法对一个太大的步长的反应是舍弃几乎所有提议，因此仿真进展速率会依 F 的指数变坏。在切分采样中，不存在舍弃。严格地保持在原地的概率很小。

▷ **习题 29.10**[e] 考察将切分采样应用于如图 29.17 所示密度时的特性。x 是 0.0 ~ 11.0 之间的一个实变量。通常要多久切分采样才能将一个点 x 从顶峰区 $x \in (0,1)$ 转移到拖尾区 $x \in (1,11)$，反过来又是多久？验证这些转移的概率确实渐近地给出一个正确的概率密度。

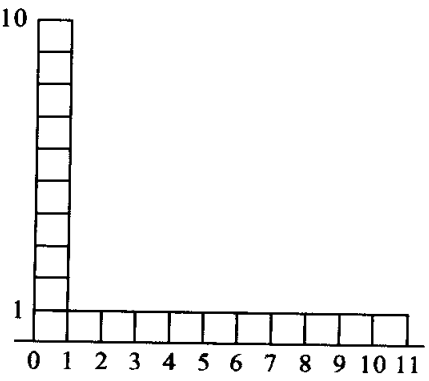


图 29.17 $P^*(x)$

在实际问题中如何使用切分采样

在上述一维切分采样方法的帮助下，可以对一个 N 维密度 $P(x) \propto P^*(x)$ 进行采样；具体做法是：选择一系列方向 $y^{(1)}, y^{(2)}, \cdots$ 并定义 $x = x^{(t)} + xy^{(t)}$。前面的函数 $P^*(x)$ 用 $P^*(x) = P^*(x^{(t)} + xy^{(t)})$ 替代。可以有多种选择方向的策略，比如，同 Gibbs 采样一样，方向可以选择坐标轴；或者，在使得整个过程满足详细平衡的前提下，随机地选择方向 $y^{(t)}$。

计算机友好的切分采样

概率模型中的实变量在计算机中总是用有限个比特来表示的。在下面这个由 Skilling 所给出的切分采样实现中，前面以浮点运算方式讨论的迈出、随机化和收缩等操作，都被代之以二进制和整数运算。

假设要做切分采样的变量 x 用 b 比特整数 X 表示，它取 $B = 2^b$ 个值 $0, 1, 2, \cdots, B-1$ 之一，且这些数值中的许多或全部对应于 x 的有效值。使用整数栅格来消除因浮点数的变量精度舍入而带来的详细平衡中的误差。从 X 到 x 的映射不必一定是线性的；如果它是非线性的，假设函数 $P^*(x)$ 由一个合适的转换函数替代，比如 $P^{**}(X) \propto P^*(x)\,|dx/dX|$。

假设可以使用如下针对 b 比特整数的运算符：

$X + N$	X 和 N 的模 B 算术和
$X - N$	X 和 N 的模 B 差
$X \oplus N$	X 和 N 的逐比特异或
$N := \text{randbits}(l)$	将 N 设置为一个 l 比特长的随机整数

于是，整数的切分采样过程如下。

已知：当前点 X 和高度 $Y = P^*(X) \times \text{Uniform}(0,1) \leqslant P^*(X)$

$1: U := \text{randbits}(b)$	定义二进制坐标系统的一个随机变换 U
$2:$ 为 l 赋值 $l \leqslant b$	设置初始 l 比特采样范围
3: do{	
4: $N := \text{randbits}(l)$	在宽度为 2^l 的当前区间内定义一个随机移动
5: $X' := ((X-U) \oplus N) + U$	随机化 X 的最低 l 比特（在变换后的坐标系统中）
6: $l := l - 1$	如果 X' 不可接受，减小 l，并用一个小一些的扰动 X 进行重试；
7: } until $(X' = X)$ 或 $(P^*(X') \geqslant Y)$	在 $l = 0$ 的当时或之前，结束。

引入变换 U 是为了避免永久性的明显边界；比如，不然的话，相邻二进制整数 0111111111 和 1000000000 可能就要永久性地处于不同区段，使得 X 很难从一个移动到另一个。

如图 29.18 所示是抽取新候选点的区间序列。首先,如图顶部水平线所示,从整个区间上抽取一点。在随后的抽取中,区间长度减半,但始终包含前一个点 X。

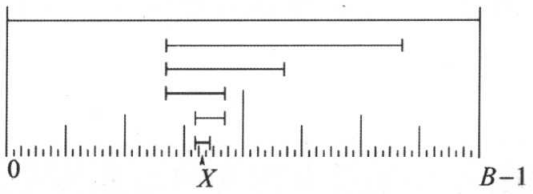


图 29.18 从中抽取新候选点的区间序列

如果初始范围要求一个预备迈出,上面的第二步可用如下类似的过程替代:

```
2a:设置 l 的值 l < b                    l 设置初始宽度
2b:do {
2c:    N: = randbits(l)
2d:    X′: = ((X − U) ⊕ N) + U
2e:    l: = l + 1
2f:  } until(l = b)或(P*(X′) < Y)
```

这些收缩和迈出方法每次计算会缩小或扩大一个两倍因子。一个变形是每次缩小或扩大高于一个比特,设 $l:=l\pm\Delta l$,其中 $\Delta l>1$。在每一步从任何预先指定的分布中取 Δl 的值(包括 $\Delta l=0$),可以有额外的灵活性。

▷ **习题 29.11**[✓] 在收缩阶段,产生了一个不可接受的 X'之后,可以允许根据切分的高度 Y 和 $P^*(X')$的值之间的差别选择 Δl,而不破坏算法的有效性。(证明之。)对于较大的 $Y-P^*(X')$,选择较大的 Δl 不失为良策。用理论或实验方法研究这一思想。

用整数所表示的一个特性:通过适当地扩大比特数,单个整数 X 可以表示两个或更多的实参数,比如,通过 Peano 曲线之类的空间填充曲线,可以将 X 映射到(x_1, x_2, x_3)。因此可以使用与一维相同的软件进行多维切分采样。

29.8 实 用 性

能预测一个马尔科夫链蒙特卡罗仿真要多久才能平稳下来吗? 通过考虑马尔科夫链蒙特卡罗仿真所涉及的随机游动,可以获得一个收敛所需时间的简单下界。但是更精确地预测这一个时间则是一个很难的问题,许多理论结果给出了收敛时间的上界,但是没有什么实用价值。第 32 章的精确采样方法对某些马尔科夫链就这个问题给出了一个解答。

在运行仿真时能诊断或检测收敛吗? 这也是一个很难的问题。有几种实用工具

可用,但是都算不上完美(Cowles 和 Carlin,1996 年)。

可以加速马尔科夫链蒙特卡罗方法的收敛时间和独立样本之间的时间吗? 如下章所述,这方面倒是有好消息。下章将描述哈密尔顿蒙特卡罗方法、超松弛以及模拟退火等方法。

29.9 进一步的实际问题

可以计算归一化常数吗?

如果目标密度 $P(x)$ 以非归一化密度 $P^*(x)$ 的形式给出:$P(x)=\frac{1}{Z}P^*(x)$,可能会对 Z 的值感兴趣。蒙特卡罗方法未能很好地给出这个量的估计,而寻找计算它的途径则是一个活跃的研究领域。计算 Z 的方法包括以下几种。

1. 重要性采样(Neal(1993 b)做了综述)和退火重要性采样(Neal,1998 年)。

2. 模拟退火过程中的"热动力综合"、"接受率"方法和"伞采样"(Neal(1993 b)做了综述)。

3. "可逆跳马尔科夫链蒙特卡罗"(Green,1995 年)。

但是,处理 Z 的一种方法也许是不要求计算 Z 值而直接完成求解任务。在贝叶斯数据建模中,如果不采用多个模型,可能可以避免计算 Z 的麻烦,不过它对于模型比较会很重要。可以使用一个递阶模型,比如有不同的连续超参数,其所完成的功能类似于不同模型所完成的(Neal,1996 年);而不是使用几个模型(比如在复杂度上有差别)并计算它们的相对后验概率。在提示不计算 Z 的可能性时,笔者并未对这种方法提供担保。归一化常数 Z 经常是问题中最重要的一个数,笔者认为应该尽最大的努力去计算它。

大模型的 Metropolis 方法

原来对 Metropolis 方法的描述涉及使用提议密度 $Q(x';x)$ 对所有变量做联合更新。对于大问题,使用几个提议分布 $Q^{(b)}(x';x)$,每一个只更新 x 的部分元素,这种方式可能更有效。对于每个提议,单独地决定是接受还是拒绝,提议分布依次重复运行。

▷ **习题 29.12**[2] 解释为什么当 B 个提议 $Q^{(1)},\cdots,Q^{(B)}$ 逐个依次考虑时,状态空间里的运动速率会高于单个提议 Q^* 时的速率,这里 Q^* 定义为 $Q^{(1)},\cdots,Q^{(B)}$ 的级联。假设每个提议分布 $Q^{(b)}(x';x)$ 的接收率 $f<1/2$。

在 Metropolis 方法中,提议密度 $Q(x';x)$ 通常有许多参数控制"宽度"之类的特性。这些参数通常用尝试法设定,经验法则所瞄准的舍弃频度约为 0.5。在仿真过

程中，根据仿真的历史动态地更新宽度参数，这样做是无效的。对提议密度的这样一种更改会违反详细平衡条件；该条件保证了马尔科夫链有正确的不变分布。

大模型的 Gibbs 采样

关于 Gibbs 采样的描述涉及一次采样一个参数，如式(29.35 ~ 29.37)所述。对于大型问题，联合地采样成组变量可能更有效，及使用几个提议分布

$$x_1^{(t+1)},\cdots,x_a^{(t+1)} \sim P(x_1,\cdots,x_a \mid x_{a+1}^{(t)},\cdots,x_K^{(t)}) \tag{29.47}$$

$$x_{a+1}^{(t+1)},\cdots,x_b^{(t+1)} \sim P(x_{a+1},\cdots,x_b \mid x_1^{(t+1)},\cdots,x_a^{(t+1)},x_{b+1}^{(t)},\cdots,x_K^{(t)}),\text{等等}$$

需要多少样本？

在本章开始处，已观察到估计器$\hat{\Phi}$的方差只依赖于独立样本数 R 和数值

$$\sigma^2 = \int \mathrm{d}^N x P(x)(\phi(x) - \Phi)^2 \tag{29.48}$$

已经讨论了由 $P(x)$生成样本的各种方法。那么应该瞄准多大的独立样本数 R？

在许多问题中，实际仅需要大约 12 个来自 $P(x)$的独立样本。设想 x 是未知向量，比如剑桥的 10 000 条地下管道上所出现的腐蚀个数，而 $\phi(x)$是修补这些管道的总费用。分布 $P(x)$所描述的是，在给定对一些管道所做的测试和关于腐蚀的物理特性所做假设的条件下，状态 x 的概率。量 Φ 是修补费用的期望值。量 σ^2 是费用的方差，σ 所度量的是预计实际费用和期望值 Φ 之间的差别。

那么，一个经理会想知道多精确的 Φ？我想说的是，精度范围小于约 $\sigma/3$ 的 Φ 形同虚设。毕竟，实际费用可能和 Φ 相差 $\pm\sigma$。如果获得 $R=12$ 个来自 $P(x)$的独立样本，对 Φ 的估计可以达到精度 $\sigma/\sqrt{12}$，它低于 $\sigma/3$。因此，12 个样本就足够了。

资源分配

假设已经决定所需的独立样本数 R，一个重要的问题是如何使用有限的计算机资源来获得这些样本。

一个典型的马尔科夫链蒙特卡罗实验会包括一个起始阶段，用来调整步长之类的仿真控制参数。接下来是一个“内燃”阶段，希望仿真在此阶段能“收敛”到所期望的分布。最后，随着仿真的继续，偶尔记录状态向量，以便得出一列状态$\{x^{(r)}\}_{r=1}^{R}$，希望它们是来自 $P(x)$的大致独立样本。

有几种可能的策略(如图 29.19 所示)。

1. 运行很长一次，从中获得所有 R 个样本。

2. 对于不同的初始条件，运行几个中等长度的仿真，并从每一个中获得一些样本。

3. 从不同的随机初始条件开始，运行 R 个短时间的仿真，并记录每一个仿真的最后状态。

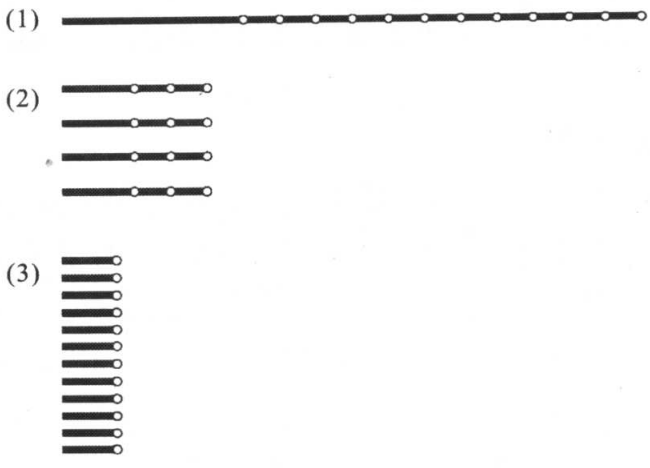


图 29.19 3 种可能的马尔科夫链蒙特卡罗策略,用以在固定计算机机时内获取 12 个样本。时间用水平线表示;样本用白色小圈表示。(1) 单次运行包含一次悠长的“内燃”期,然后是采样期。(2) 4 次具有不同初始条件的中等长度的运行以及一个中等长度的“内燃”期。(3) 12 个短时间的运行。

第一种策略达到“收敛”的机会最大。最后一种策略可能会有这样一种优势:记录的样本之间的相关性较小。中间这一种是马尔科夫链蒙特卡罗专家们使用最多的策略(Gilks 等,1996 年),因为它避免了因为抛弃多次运行中的内燃迭代而造成的效率低下,同时又允许我们检测未收敛问题,而这在单次运行中不会太明显。

最后,我想强调在估计中不必刻意去独立这些点。对于彼此相关的点进行平均是可以的,它不会在估计中导致任何偏倚。比如,当采用策略 1 或策略 2 时,如果你愿意,可以包括每次运行时从第一个样本到最后一个样本之间的所有点。当然,当这些点彼此相关时,估算估计值的精度会更困难些。

29.10 小 结

- 蒙特卡罗方法是一种强有力的工具,允许人们对可表示成 $P(x)=\frac{1}{Z}P^*(x)$ 这种形式的任何概率分布进行采样。
- 蒙特卡罗方法可以回答几乎所有关于 $P(x)$ 的查询,只需将查询表示成形式

$$\int \phi(x)P(x) \simeq \frac{1}{R}\sum_r \phi(x^{(r)}) \tag{29.49}$$

- 在高维问题中,唯一令人满意的解决方法是基于马尔科夫链的方法,比如 Metropolis 方法、Gibbs 采样和切分采样。Gibbs 采样是一种有吸引力的方法,因为它不需要可调整的参数,但是它的应用仅限于可以从条件分布生成样本的场合。切分采样具有吸引力,因为尽管它也有步长参数,但它的性能对步长不是很敏感。

• 尽管简单 Metropolis 算法和 Gibbs 采样算法的应用很广泛，但是它们的性能不佳，因为它们使用缓慢的随机游动来探索状态空间。下一章将讨论加速马尔科夫链蒙特卡罗仿真的方法。

• 切分采样没有避开随机游动行为，但是它自动地选择最大合适步长，因此和 Metropolis 方法使用小步长相比，它降低了随机游动的负面影响。

29.11 习 题



习题 29.13[2C] **关于重要性采样的一项研究**。在第 29.2 节，已经确定在高维问题中，重要性采样可能是无用的。本习题将进一步探讨一个警戒性的情景，说明即使在一维空间中使用友好的高斯分布，重要性采样还是可能会失效。

假设想知道一个函数 $\phi(x)$ 在分布 $P(x)$ 下的期望值

$$\Phi = \int \mathrm{d}x P(x)\phi(x) \tag{29.50}$$

并且用分布为 $Q(x)$ 的重要性采样来估计这个期望值。或者，也可能想要估计公式 $P(x) = P^*(x)/Z$ 中的归一化常数 Z，使用

$$Z = \int \mathrm{d}x P^*(x) = \int \mathrm{d}x Q(x)\frac{P^*(x)}{Q(x)} = \left\langle \frac{P^*(x)}{Q(x)} \right\rangle_{x\sim Q} \tag{29.51}$$

再设 $P(x)$ 和 $Q(x)$ 是均值为 0、标准差分别为 σ_p 和 σ_q 的高斯分布。从 Q 中所抽取的每一点 x 都会有一个相应的权重 $P^*(x)/Q(x)$。这些权重的方差是多少？[假设 $P^* = P$，即实际上 P 是已归一化的，而 $Z = 1$，尽管可以佯装不知。]当 $\sigma_q^2 \to \sigma_p^2/2$ 时，权重的方差会怎样？

通过在计算机上仿真这一重要性采样问题，来检查你的理论。



习题 29.14[2] 考虑将 Metropolis 算法应用于第 29.4 节的一维简化问题，并从 $\{0,1,\cdots,20\}$ 中进行采样。每当当前状态是终端状态之一时，式(29.34)所给出的提议密度会以 50% 的概率提议舍弃一个状态。

为了降低这种“浪费”，弗雷德修改了负责根据 Q 生成样本的那部分软件，使得当 $x = 0$ 时，提议密度在 $x' = 1$ 上是 100%，类似地，当 $x = 20$，则总是提议 $x' = 19$。

弗雷德设定实现接收规则的那部分软件，使得所有被提议的移动都为这些软件所接受。弗雷德修改后的软件生成样本的概率 $P'(x)$ 是什么？

为了获得来自 $P(x)$ 的样本，弗雷德的提议密度所对应的正确接收规则是什么？

▷ **习题 29.15**[2C] 在第 22.1 节，通过最大似然法研究了单个一维高斯变量的推理问题。现在请对这一推理问题，实现 Gibbs 采样。为 μ 赋予一个宽的高斯先验分布，为精度参数 $\beta = 1/\sigma^2$ 赋予一个宽的伽马先验分布(24.2)。对 μ 的每次更新都涉及一

个来自高斯分布的样本，而对 σ 的每次更新都涉及一个来自伽马分布的样本。



习题 29.16[JC] **用于聚类的 Gibbs 采样**。在第 22.2 节，通过最大似然法研究了 K 个一维高斯变量的混合的推理问题。现在请对这一推理问题，实现 Gibbs 采样。允许各个聚类有不同的标准差 σ_k。以与上题完全相同的方式，为均值和标准差赋予先验分布。要么将类 $\{\pi_k\}$ 的先验概率固定为均等，要么为参数 π 赋予均匀的先验分布并将它们包含于 Gibbs 采样中。

注意 Gibbs 采样和软 K-均值聚类算法（算法 22.2）之间的相似性。可以在给定参数 $\{\mu_k, \sigma_k\}$ 时交替地分配类标号 $\{k_n\}$，然后在给定类标号时更新参数。分配步骤涉及对责任（22.22）所定义的概率分布进行采样，而更新步骤通过使用以 K-均值算法的值（22.23，22.24）为中心的概率分布，来更新均值和方差。

你的实验是否证实蒙特卡罗方法绕过了第 22.4 节所讨论的最大似然方法的超拟合困难？

用 octave 写成的本题和上题的解答可以在网站 http://www.inference.phy.cam.ac.uk/mackay/itila/ 上找到。

▷ **习题 29.17**[JC] 对 7-**科学家**推理问题实现 Gibbs 采样，这个问题在习题 22.15 遇到过，而且你也可以通过精确边缘化求解（习题 24.3）[后者并非非做不可]。

▷ **习题 29.18**[2] 用 Metropolis 方法探索分布 $P(x)$，一个 1 000 维的球面高斯分布，其标准差在所有维度上都是 1。提议密度 Q 也是一个 1 000 维的球面高斯分布，但其标准差在所有维度上都是 ε。如果接收率为 0.5，步长 ε 大致是多少？假设这个 ε 值，回答以下问题

（a）大约要过多久，这种方法才能穿过该分布并生成一个独立于初始条件的样本？

（b）在一个典型步骤中，$\ln P(x)$ 会改变多少？当 x 是从 $P(x)$ 中抽取时，$\ln P(x)$ 应该改变多少？

（c）如果不采用 Metropolis 方法试图同时改变所有元素，而是用 Metropolis 的级联一次改变一个元素，情况又会怎样？

▷ **习题 29.19**[2] 在讨论 Metropolis 算法生成独立样本所用时间时，考虑使用一个步长为 ε 的提议分布，来探索最长空间长度尺度为 L 的一个分布。MCMC 方法必须探索的另一个维度是对数概率 $\ln P^*(x)$ 可能取值的范围。假设状态 x 所包含的独立随机变量的个数与 N 成正比，当样本来自 $P(x)$ 时，“渐近均分”原理告诉我们：$-\ln P(x)$ 的值可能接近 x 的熵，两边变动的标准差的等级与 $\sqrt{N}$ 相同。考虑一个具有对称提议密度的 Metropolis 方法，即提议密度满足 $Q(x;x') = Q(x';x)$ 的 Metropolis 方法。假设所接受的跳要么将 $\ln P^*(x)$ 增加一定量，要么减小一点点，例如减小 $\ln e = 1$（这是否是一个合理假设？），讨论必须等多久才能从 $P(x)$ 生成大致独立的样本。讨论 Gibbs 采样是否也有类似的特性。

习题 29.20[♪] 马尔科夫链蒙特卡罗方法不计算划分函数 Z，但是它们允许估计像 Z 之类参量的比值。例如，考虑状态空间中的随机游动 Metropolis 算法：在这个空间里，一个连通的可访问区域中的能量为 0，而在其他所有地方都是无穷大；并设想可将这个可访问空间分成两个区域，中间以一个或多个走廊状态进行连接。平衡时，在每个区域花费的时间所占比重与该区域的体积成正比。不允许测量体积，蒙特卡罗方法怎样才能做到这一点呢？

习题 29.21[♪] 哲学。

这些蒙特卡罗方法有一个古怪的缺陷：它们都是非贝叶斯的（O'Hagan，1987 年），这一缺陷被贝叶斯统计学家广泛使用。它们涉及计算机实验，从这些实验可以推导出感兴趣量的*估计器*。这些估计器依赖用于生成样本的提议分布以及我们随机数生成器所产生的随机数。与之相对，此问题的另一贝叶斯方法会用计算机实验的结果来推出目标函数 $P(x)$ 的特性，并对感兴趣量（比如 Φ）生成预测分布。这种方法给出的结果只依赖于 $P^*(x^{(r)})$ 在点 $\{x^{(r)}\}$ 处的计算值；结果同如何选择这些点无关。你能构造一个贝叶斯蒙特卡罗方法吗？（一个实际的尝试请参见 Rasmussen 和 Ghahramani（2003 年）。）

29.12 习题解答

习题 29.1 的解答 希望证明

$$\hat{\Phi} \equiv \frac{\sum_r w_r \phi(x^{(r)})}{\sum_r w_r} \tag{29.52}$$

收敛到 Φ 在 P 下的期望值。分别考虑分子和分母。首先看分母。考虑单一重要性权值

$$w_r \equiv \frac{P^*(x^{(r)})}{Q^*(x^{(r)})} \tag{29.53}$$

如果在点 $x^{(r)}$ 的分布 $Q = Q^*/Z_Q$ 下做平均，这个权值的期望是多少？

$$\langle w_r \rangle = \int \mathrm{d}x Q(x) \frac{P^*(x)}{Q^*(x)} = \int \mathrm{d}x \frac{1}{Z_Q} P^*(x) = \frac{Z_P}{Z_Q} \tag{29.54}$$

因此，分母的期望是

$$\left\langle \sum_r w_r \right\rangle = R \frac{Z_P}{Z_Q} \tag{29.55}$$

只要 w_r 的方差有限，随着 R 的增加，这个分母除以 R 之后会收敛到 Z_P/Z_Q。[事实上，即使方差无限，估计值也会收敛到正确答案，只要期望是定义明确的。]类似地，分子中一项的期望是

$$\langle w_r \phi(x) \rangle = \int \mathrm{d}x Q(x) \frac{P^*(x)}{Q^*(x)} \phi(x) = \int \mathrm{d}x \frac{1}{Z_Q} P^*(x) \phi(x) = \frac{Z_P}{Z_Q} \Phi \tag{29.56}$$

其中，Φ 是 ϕ 在 P 下的期望值。因此分子除以 R 后会随着 R 的增加收敛到 $\frac{Z_P}{Z_Q}\Phi$。因此 $\hat{\Phi}$ 收敛到 Φ。

分子和分母分别是 RZ_P/Z_Q 和 $RZ_P/Z_Q\Phi$ 的无偏倚估计器，但是对于有限的 R，它们的比值 $\hat{\Phi}$ 不一定是无偏倚估计器。

习题 29.2 的解答 当真实密度 P 是多峰函数时，使用具有拟合了一个峰值的采样器密度进行重要性采样是不明智的，因为当产生的点落在其他峰值上这种稀有事件发生时，与该点相对应的权值会变得巨大。估计值的方差会很大，但是在没有看见其他峰值上的点时，这个大方差对用户而言并不明显。

习题 29.5 的解答 从 Gibbs 采样的角度来看，校正子译码问题的后验分布是病态分布。因子 $\mathbb{1}[\boldsymbol{Hn}=\boldsymbol{z}]$ 在可能向量 $\boldsymbol{n}$ 的空间（即相应于有效码字的 2^K 个点）中只有很小一部分取 1。任何两个码字都不相邻，于是类似地，从一个可行状态 $\boldsymbol{n}$ 翻转任何一个比特，会将我们带入一个具有零概率的状态，因此，在 Gibbs 采样中状态永远不会移动。

一个通用码也有完全相同的问题。相应于有效码字的点在数量上相对较少，并且它们不相邻（至少任何有用的码都是如此）。因此，出于两种原因，Gibbs 采样对校正子译码没有用。首先，很难找到任何还算好的假设，而且只要状态不靠近一个有效码字，Gibbs 采样就帮不上忙，因为条件分布一个也没有定义；其次，一旦进入一个有效假设，Gibbs 采样无法把我们带出去。

可以试着使用原消息 $\boldsymbol{s}$ 的比特作为变量，进行 Gibbs 采样。这种方法不会像刚说的那样锁住，但是，对于一个好码，任何一个比特翻转都会极大地改变重构码字，因此如果已找到一个似然度适当大的状态，Gibbs 采样想从这个状态逃脱，所需的时间会大得不切实际。

习题 29.12 的解答 每个 Metropolis 提议会将状态的能量增加或减小一定量。当 B 个提议被级联起来时，总的能量改变是包含 B 步的随机游动的终点。这个漫步可能是零均值的，也可能有往上飘的趋势（如果大多数的移动增加能量，只有几个减少能量）。如果接收率 f 较小，通常后者会成立：任何一次移动的平均能量改变是某个 $\Delta E>0$，因而 B 步级联的接收概率的量级为 $1/(1+\exp(-B\Delta E))$，它大致地按照 f^B 变化。移动的均方距离的量级为 $f^B B\varepsilon^2$，其中 ε 是典型步长。与之相对，在各自移动时，移动的均方距离的量级为 $fB\varepsilon^2$。

习题 29.13 的解答 权重为 $w=P(x)/Q(x)$，且 x 是从 Q 中抽取。平均权重为

$$\int \mathrm{d}xQ(x)[P(x)/Q(x)] = \int \mathrm{d}xP(x) = 1 \tag{29.57}$$

并假设这个积分收敛。方差为

$$\mathrm{var}(w) = \int \mathrm{d}xQ(x)\left[\frac{P(x)}{Q(x)}-1\right]^2 \tag{29.58}$$

$$= \int \mathrm{d}x\,\frac{P(x)^2}{Q(x)} - 2P(x) + Q(x) \tag{29.59}$$

$$= \left[\int \mathrm{d}x \frac{Z_Q}{Z_P^2}\exp\left(-\frac{x^2}{2}\left(\frac{2}{\sigma_p^2}-\frac{1}{\sigma_q^2}\right)\right)\right]-1 \qquad (29.60)$$

其中，$Z_Q/Z_P^2=\sigma_q/(\sqrt{2\pi}\sigma_p^2)$。只有指数中 x^2 的系数为正，即

$$\sigma_q^2>\frac{1}{2}\sigma_p^2 \qquad (29.61)$$

式(29.60)中的积分才是有限的。如果满足这个条件，方差为

$$\mathrm{var}(w)=\frac{\sigma_q}{\sqrt{2\pi}\sigma_p^2}\sqrt{2\pi}\left(\frac{2}{\sigma_p^2}-\frac{1}{\sigma_q^2}\right)^{-\frac{1}{2}}-1=\frac{\sigma_q^2}{\sigma_p(2\sigma_q^2-\sigma_p^2)^{1/2}}-1 \qquad (29.62)$$

当 σ_q 趋近临界值(约为 $0.7\sigma_p$)时，方差变得无穷大。如图 29.20 所示为 $\sigma_p=1$ 时 σ_q 从 0.1 变为 1.5 时所出现的这种现象。对于所有的仿真，都使用了同一随机数种子，因此权重和估计值都是平滑的曲线。注意：R 个权重的经验标准差看起来可以很小，且行为正常(比如，在 $\sigma_q\simeq 0.3$ 处)，而实际的标准差却是无穷大。

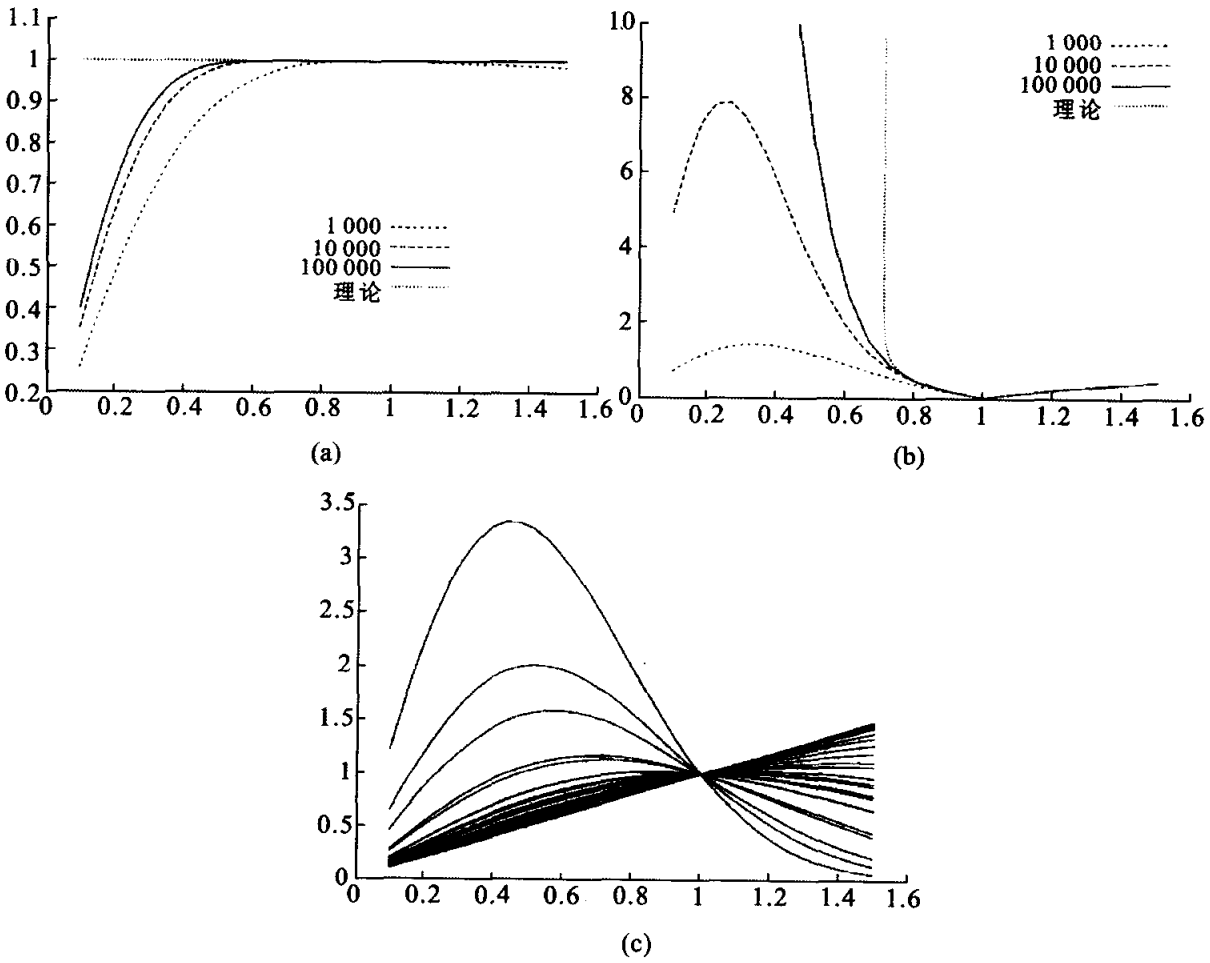


图 29.20 一维中的重要性采样。对于 $R=1\,000, 10^4$ 和 10^5，使用重要性采样(其采样器密度的标准差为 σ_q(横轴))来估计一个高斯分布的归一化常数(事实上已知为 1)。对于所有仿真运行，都使用了同一随机数种子。3 条曲线分别表示(a) 估计的归一化常数；(b) R个权重的经验标准差；(c) 权重中的 30 个。

第 30 章

有效的蒙特卡罗方法

本章讨论几种降低 Metropolis 方法中随机游动特性的方法。目标是降低获取有效独立样本所需的时间。为了简单起见，当要表达“实际上的独立样本”(Effective Independent Sample)这个意思时，本章采用“独立样本”的说法。

30.1 哈密尔顿 - 蒙特卡罗方法

哈密尔顿 - 蒙特卡罗方法是一种适用于连续状态空间的 Metropolis 方法，它使用梯度信息来降低随机游动特性。[出于历史的原因，哈密尔顿 - 蒙特卡罗方法原来叫做混合蒙特卡罗方法。]

对于许多系统，其概率 $P(\boldsymbol{x})$ 可写成形式

$$P(\boldsymbol{x}) = \frac{\mathrm{e}^{-E(\boldsymbol{x})}}{Z} \tag{30.1}$$

不仅 $E(\boldsymbol{x})$ 而且它关于 $\boldsymbol{x}$ 的梯度也很容易计算出来。在可获得梯度信息时，使用简单随机游动 Metropolis 方法看起来好像很浪费，因为梯度指明了朝哪个方向走才能找到具有较高概率的状态！

哈密尔顿 - 蒙特卡罗方法概述

在哈密尔顿 - 蒙特卡罗方法中，用动量变量 $\boldsymbol{p}$ 对状态空间 $\boldsymbol{x}$ 做了扩充，并且有两种可供选择的提议。第一种提议随机化动量变量，而不改变状态 $\boldsymbol{x}$。第二种提议使用模拟哈密尔顿动力系统，同时改变 $\boldsymbol{x}$ 和 $\boldsymbol{p}$。模拟哈密尔顿动力系统由哈密尔顿函数

$$H(\boldsymbol{x},\boldsymbol{p}) = E(\boldsymbol{x}) + K(\boldsymbol{p}) \tag{30.2}$$

定义，这里 $K(\boldsymbol{p})$ 是“动能”，比如 $K(\boldsymbol{p}) = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{p}/2$。这两个提议用于从联合密度(渐近地)产生样本

$$P_H(\boldsymbol{x},\boldsymbol{p}) = \frac{1}{Z_H}\exp[-H(\boldsymbol{x},\boldsymbol{p})] = \frac{1}{Z_H}\exp[-E(\boldsymbol{x})]\exp[-K(\boldsymbol{p})] \quad (30.3)$$

该密度是可分离的，因此，$\boldsymbol{x}$ 的边缘分布是理想分布 $\exp[-E(\boldsymbol{x})]/Z$。所以，简单地抛弃动量变量，可以获得一系列渐近地来自 $P(\boldsymbol{x})$ 的样本 $\{\boldsymbol{x}^{(t)}\}$。

哈密尔顿 - 蒙特卡罗方法的详情

第一种提议可以看做 Gibbs 采样更新，它从高斯密度 $\exp[-K(\boldsymbol{p})]/Z_K$ 中抽取新的动量。这一提议总是被接受。在第二种动态提议中，动量变量决定状态 $\boldsymbol{x}$ 往何处走，而 $E(\boldsymbol{x})$ 的梯度根据如下方程决定动量 $\boldsymbol{p}$ 如何变化

$$\dot{\boldsymbol{x}} = \boldsymbol{p} \quad (30.4)$$

$$\dot{\boldsymbol{p}} = -\frac{\partial E(\boldsymbol{x})}{\partial \boldsymbol{x}} \quad (30.5)$$

因为在每次动态提议过程中，$\boldsymbol{x}$ 在动量 $\boldsymbol{p}$ 的方向上持续运动，系统状态移动的距离倾向于随计算机机时线性增长，而不是按平方根增长。

第二种提议按 Metropolis 规则接受。如果对哈密尔顿动力系统的仿真在数值上是完美的，那么每次的提议都被接受，因为总能量 $H(\boldsymbol{x},\boldsymbol{p})$ 是该运动的一个常数，所以式(29.31)中的 a 等于 1。如果仿真不完美，比如受到有限步长的影响，那么有些动态提议就会被拒绝。拒绝法则利用 $H(\boldsymbol{x},\boldsymbol{p})$ 中的变化，在仿真完美时该变化为 0。偶尔的拒绝保证我们会渐近地获得来自所需联合密度 $P_H(\boldsymbol{x},\boldsymbol{p})$ 的样本 $(\boldsymbol{x}^{(t)},\boldsymbol{p}^{(t)})$。

算法 30.1 中的源代码描述了一个哈密尔顿 - 蒙特卡罗方法，它使用“蛙跳”算法来仿真函数 findE(x) 的动态特性，其梯度由函数 gradE(x) 求得。如图 30.2 所示，该算法从一个双变量高斯分布生成样本，它的能量函数为 $E(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}$，其中

$$\boldsymbol{A} = \begin{bmatrix} 250.25 & -249.75 \\ -249.75 & 250.25 \end{bmatrix} \quad (30.6)$$

相应于一个方差 - 协方差矩阵

$$\begin{bmatrix} 1 & 0.998 \\ 0.998 & 1 \end{bmatrix} \quad (30.7)$$

在图 30.2(a)中，从箭头所标识的状态开始，实线表示哈密尔顿动力系统相继产生的两条轨迹。小方块表示这两条轨迹的终点。每条轨迹包含 epsilon = 0.055 的 Tau =

19个“蛙跳”步。这些步骤以轨迹上的十字表示,插入时做了放大。在每条轨迹之后,动量被随机化。这里,两条轨迹都被接受;哈密尔顿函数上的误差也分别为+0.016和-0.06。

如图30.2(b)所示为4条轨迹的序列从初始条件开始如何收敛的情况,以箭头标出的初始条件并不靠近目标分布的典型集合。对于每条轨迹的轨迹参数Tau和epsilon,使用均值分别为19和0.055的均匀分布,做随机化。第一条轨迹将我们带入一个新状态(-1.5,-0.5),它在能量上和第一个状态类似。第二条轨迹恰好结束在靠近能量图底部的一个状态。这里,因为势能E小了一些,动能$K=p^2/2$必定比它在轨迹开始处大。当在第3条轨迹前进行动量随机化时,它的动能变得小了很多。在仿真第4条轨迹之后,状态看起来变成了目标密度的典型状态。

如图30.2(c)和图30.2(d)所示为随机游动Metropolis方法,分别从(a)和(b)的初始状态开始,使用高斯提议密度从同一高斯分布进行采样。在(c)中,调整步长,使得接收率为58%。提议数为38,因此所占用的总计算机机时与(a)类似。因为随机游动特性,移动的距离较小。在(d)中,使用随机游动Metropolis方法,从与(b)相同的初始状态开始,而所占用的计算机机时也类似。

```
g = gradE(x);                          #使用初始 x 设置梯度
E = findE(x);                          #还设置目标函数

for l = 1:L                            #循环 L 次
    p = randn(size(x));                #初始动量是 Normal(0,1)
    H = p' * p/2 + E;                  #计算 H(x,p)

    xnew = x;gnew = g;
    for tau = 1:Tau                    #做 Tau 次“蛙跳”步

      p = p - epsilon * gnew/2;        #p 跨半步
      xnew = xnew + epsilon * p;       #x 跨一步
      gnew = gradE(xnew);              #寻找新梯度
      p = p - epsilon * gnew/2;        #p 跨半步

    endfor

    Enew = findE(xnew);                #寻找新的 H 值
    Hnew = p' * p/2 + Enew;
    dH = Hnew - H;                     #决定是否接受

    if(dH < 0)                         accept = 1;
    elseif(rand() < exp(-dH))          accept = 1;
    else                               accept = 0;
    endif

    if(accept)
      g = gnew;  x = xnew;  E = Enew;
    endif
endfor
```

算法30.1 哈密尔顿-蒙特卡罗方法的Octave源代码

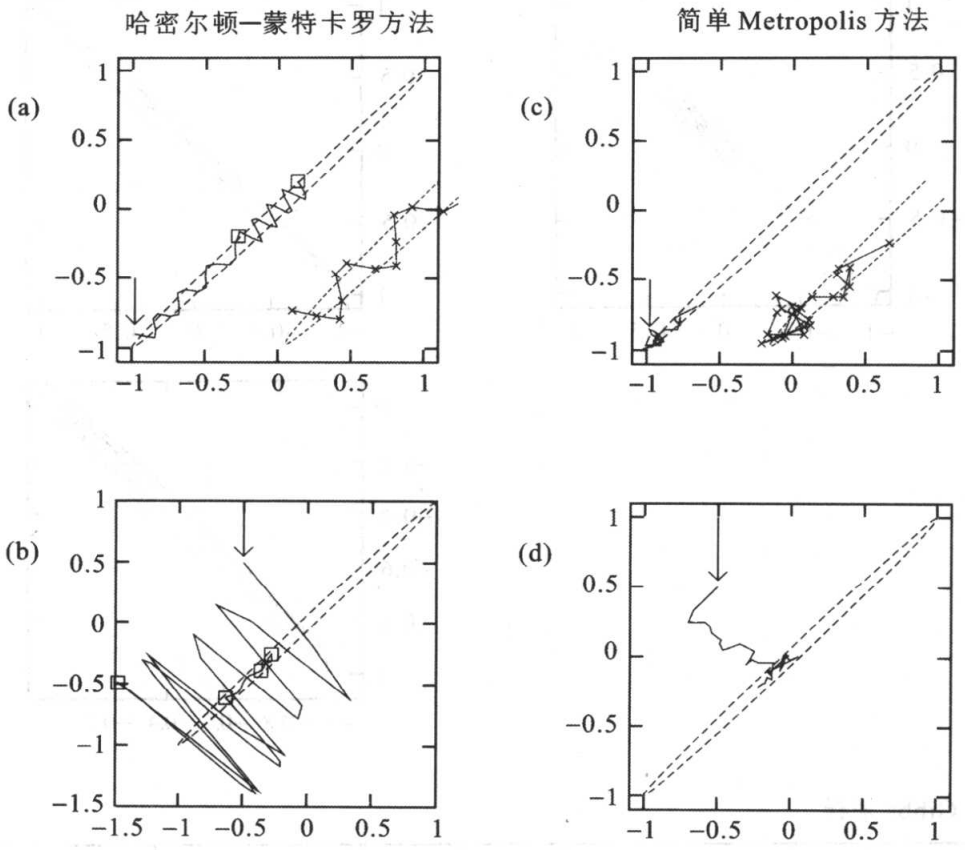


图 30.2 (a,b)用于从相关系数 $\rho = 0.998$ 的双变量高斯分布生成样本的哈密尔顿 - 蒙特卡罗方法。(c,d) 为了进行比较,已知计算机机时相等,简单随机游动 Metropolis 方法。

30.2 超 松 弛

超松弛法是 Gibbs 采样中降低随机游动特性的一种方法。超松弛法原来是针对所有条件分布都是高斯的系统提出的。

联合分布不是高斯而其条件分布都是高斯的一个例子是 $P(x,y) = \exp(-x^2y^2 - x^2 - y^2)/Z$。

高斯条件分布的超松弛

在普通 Gibbs 采样中,人们从当前变量 x_i 的条件概率中抽取新值 $x_i^{(t+1)}$,忽略旧值 $x_i^{(t)}$。如图 30.3 的左半图所示,当变量强烈相关时,状态会做漫长的随机游动。该图使用一个相关的高斯分布作为目标密度。

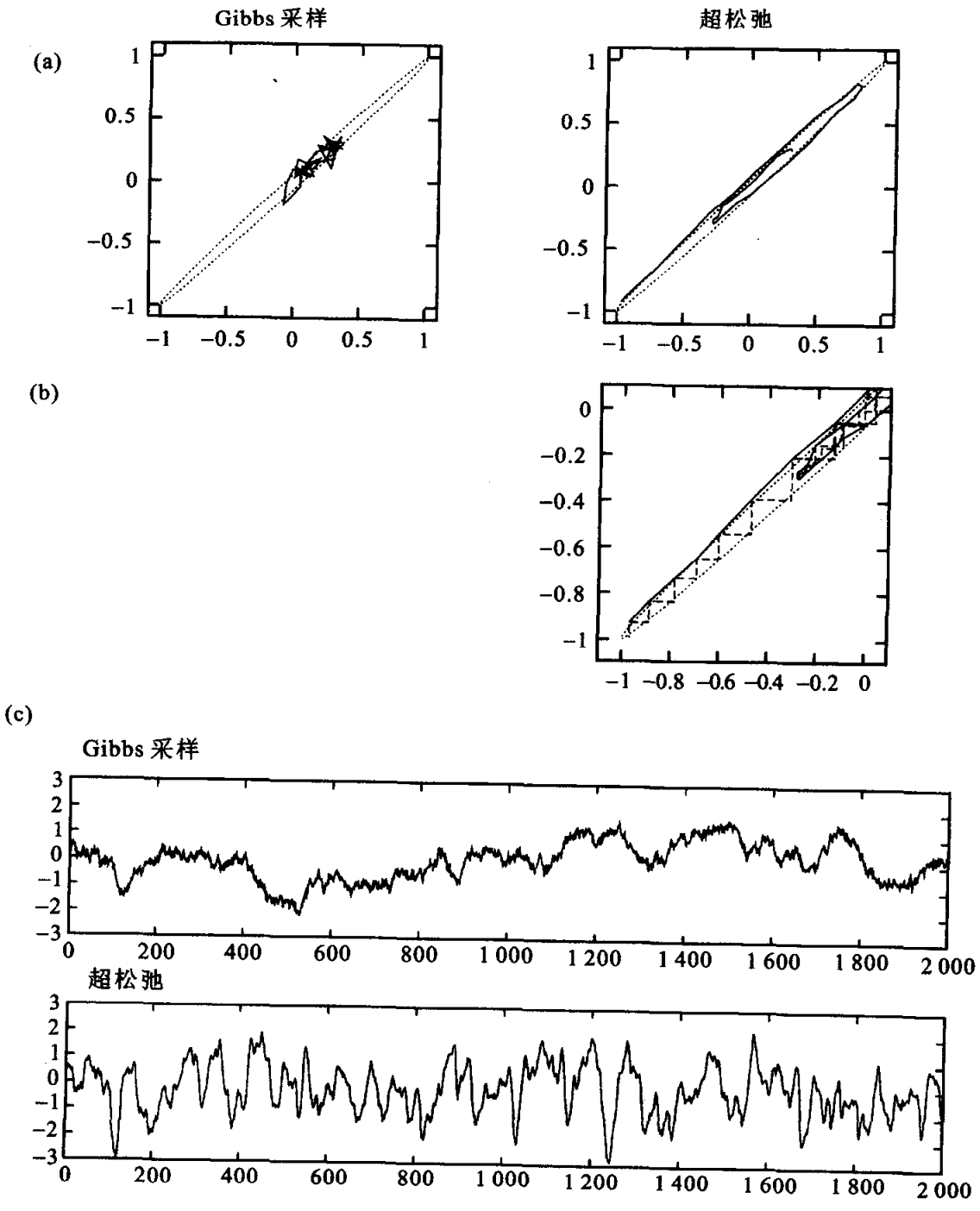


图 30.3 对于相关系数为 $\rho = 0.998$ 的双变量高斯分布，超松弛与 Gibbs 采样的比较。(a) 40 次迭代的状态序列，每次迭代包括对两个变量的一次更新。在超松弛方法中，$\alpha = -0.98$。(选择这个大的值就很容易看到超松弛方法如何减小随机游动特性。)虚线表示 $\boldsymbol{x}^{\mathrm{T}} \sum^{-1} \boldsymbol{x} = 1$ 的等势线。(b) 是(a)的详图，显示了产生每次迭代的两个步骤。(c) 两种方法中变量 x_1 在 2 000 次迭代中的时间过程。在超松弛方法中，$\alpha = -0.89$。(根据 Neal(1995 年)。)

在 Adler(1981 年)的超松弛法中，人们从一个偏向条件分布反面的高斯中抽取

样本 $x_i^{(t+1)}$。如果 x_i 的条件分布是 Normal(μ,σ^2)，x_i 的当前值是 $x_i^{(t)}$，Adler 方法会设定 x_i

$$x_i^{(t+1)} = \mu + \alpha(x_i^{(t)} - \mu) + (1-\alpha^2)^{1/2}\sigma\nu \tag{30.8}$$

其中，$\nu \sim$ Normal(0,1)，α 是在 -1 和 1 之间的参数，通常设为负值。（如果 α 为正，该方法称为低松弛法。）



习题 30.1[2] 证明这个单次转移会保持条件分布 $x_i \sim$ Normal(μ,σ^2)不变。

Adler 超松弛法的一次迭代，与一次 Gibbs 采样一样，会轮流更新每个变量，如式(30.8)所示。由所有变量以固定次序所进行的一次完整更新所定义的转移矩阵 $T(\boldsymbol{x}';\boldsymbol{x})$不满足详细平衡方程。刚才所描述的对一个坐标而言的单次转移确实满足详细平衡方程，因此整个链会给出一个有效的采样策略，保证到目标密度 $P(\boldsymbol{x})$的收敛；但是当通过一个固定次序逐一应用单次转移时，所形成的链不是可逆的。这种时间上的非对称性是为什么超松弛可能有益的关键所在。比如，如果两个变量是正相关的，那么它们（在短时间尺度上）会用直接的方式演化，而不是通过随机游动，如图 30.3 所示。这可能极大地减小获得独立样本所需的时间。

习题 30.2[2] 以固定次序进行的所有变量完整更新所定义的转移矩阵 $T(\boldsymbol{x}';\boldsymbol{x})$不满足详细平衡方程。如果更新次序是随机的，那么 T 会是对称的。对于简单二维高斯分布，研究命题：如果超松弛以随机次序进行更新，就丧失了超松弛的优势。

有序超松弛法

Neal(1995 年)对超松弛法做了推广，所得的*有序超松弛法*适用于任何可以使用 Gibbs 采样的系统。在有序超松弛法中，不是从条件分布 $P(x_i \mid \{x_j\}_{j\neq i})$ 中抽取一个样本，而是构造 K 个这样的样本 $x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(K)}$，其中 K 值可以设定在 20 左右。通常，生成 $K-1$ 个额外的样本，在得到第一个样本所需的初始计算的基础上，新增加的计算代价可以忽略。然后，对点 $\{x_i^{(k)}\}$ 进行数值分类，并在有序列中插入 x_i 的当前值，得到 $K+1$ 个点的队列。为这些点排序 $0,1,2,\cdots,K$。令 κ 为 x_i 当前值在队列中的序号。将队列中与另一端等距离的值（即序号为 $K-\kappa$ 的值）设为 x'_i。Adler 法中的参数 α 所扮演的角色，这里由参数 K 来扮演。当 $K=1$ 时，得到一个普通的 Gibbs 采样。从实用目的来看，Neal 估计有序超松弛法也许能以 10 倍或 20 倍的量级加速一个仿真。

30.3 模拟退火

加速收敛的第 3 种方法是*模拟退火*。在模拟退火中，引入一个“温度”参数，当这个参数较大时，系统可以进行在温度 1 时不太可能发生的转移。将温度设置为一

个大的值,然后逐渐降为1。采用这个过程是想降低仿真陷在一个无代表性的概率孤岛上的可能性。

假设希望对具有如下形式的分布进行采样。

$$P(\boldsymbol{x}) = \frac{\mathrm{e}^{-E(\boldsymbol{x})}}{Z} \tag{30.9}$$

其中 $E(\boldsymbol{x})$ 可以计算出来。在最简单的模拟退火方法中,转而对如下分布进行采样。

$$P_T(\boldsymbol{x}) = \frac{1}{Z(T)}\mathrm{e}^{-\frac{E(\boldsymbol{x})}{T}} \tag{30.10}$$

并将 T 逐步降低到1。

通常能量函数可以分成两项

$$E(\boldsymbol{x}) = E_0(\boldsymbol{x}) + E_1(\boldsymbol{x}) \tag{30.11}$$

其中第一项是"良性的"(比如,$\boldsymbol{x}$ 的一个可分离函数),而第二项是"恶性的"。在这些情况下,一个更好的模拟退火方法可能使用分布

$$P'_T(\boldsymbol{x}) = \frac{1}{Z'(T)}\mathrm{e}^{-E_0(\boldsymbol{x}) - E_1(\boldsymbol{x})/T} \tag{30.12}$$

其中 T 逐步降到1。采用这种方法,高温时的分布会返回由 E_0 所定义的性能良好的分布。

模拟退火经常用做最优化方法,其目标是求出最小化 $E(\boldsymbol{x})$ 的 $\boldsymbol{x}$,只是在这种情况下,温度降低到0而非1。

作为一种蒙特卡罗方法,如上所述,模拟退火没有对正确分布进行精确采样,因为并不能保证落在一个能量盆地内的概率就等于所有在那个盆地内的状态的总概率。与之密切相关的"模拟调和方法"(Marinari 和 Parisi,1992年)纠正了退火过程所引入的偏差,具体方法是:将温度本身变成一个随机变量,并在仿真过程中用 Metropolis 方式进行更新。Neal(1998年)的"退火重要性采样"方法通过对每个生成点计算重要性权值,来消除退火过程所引入的偏差。

30.4 Skilling 多状态蛙跳法

第4种加速蒙特卡罗仿真的方法是由 John Skilling 提出的,它和超松弛法在思想上有相通之处,但是应用在更高维度中。这种方法适用于对一个连续状态空间上的分布进行采样,而它的唯一要求就是能量 $E(\boldsymbol{x})$ 要容易计算。没有用到梯度。我们不打算单独使用蛙跳方法,而是与其他蒙特卡罗算符依次使用。

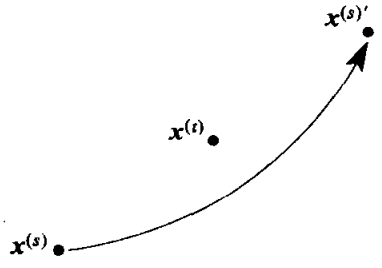
至今为止讨论过的所有蒙特卡罗方法,每次都只是在状态空间中移动一个状态向量 $\boldsymbol{x}$,而 Skilling 蛙跳法同时维护一组 S 个状态向量 $\{\boldsymbol{x}^{(s)}\}$,其中 S 可能是6或12。目标是让所有这 S 个向量都表示来自同一分布 $P(\boldsymbol{x})$ 的独立样本。

Skilling 蛙跳法从当前状态 $\boldsymbol{x}^{(s)}$ 跳到另一个状态向量 $\boldsymbol{x}^{(t)}$,并提出新的状态 $\boldsymbol{x}^{(s)'}$:

$$x^{(s)'} = x^{(t)} + (x^{(t)} - x^{(s)}) = 2x^{(t)} - x^{(s)} \tag{30.13}$$

并根据 Metropolis 方法接受或拒绝该状态。所有其他状态向量保持原位，因此接收概率只取决于 $x^{(s)}$ 的能量改变。

选择哪个向量 t 作为蛙跳事件的合作者，可以有多种方式。最简单的方法是从其他向量中随机地选择合作者。选择最靠近 $x^{(s)}$ 的那个 t 可能更好（这里可以用任何距离函数来度量），只要接下来使用一个可保证详细平衡的接受规则，这可以通过检查点 t 是否仍然最靠近新点 $x^{(s)'}$ 来完成。



为什么蛙跳是一个好主意？

设想目标密度 $P(x)$ 具有很强的相关性，比如，密度可能是针状高斯，其宽度为 ε，长度为 $L\varepsilon$，其中 $L \gg 1$。正如强调过的那样，对于这样一种密度使用标准方法会使得运动以缓慢的随机游动前进。

现在假设这 S 个点的集合刚开始处于这个密度下的一个合适位置，但是都位于一个大小为 ε 的不适当小球内。现在，使用 Skilling 蛙跳法，一个典型的第一步会将点稍稍移出当前的球，可能将它到球心的距离加倍。在所有点都得到一次移动机会之后，球的大小会增加；如果所有移动都被接受，球在所有维度上都会增加一倍左右。而有些移动被拒绝则意味着包含这些点的球可能在针长方向被拉长一个因子，比如 2。当所有点又经过一个周期，球在长方向上再次增长一个因子 2。因此，在长维度方向途经的典型距离随着迭代次数按指数级增长。

每次迭代增长一个因子 2 可能有点乐观了；但是即使这个球每次迭代只是增长一个小些的因子，比如 1.1，这种增长还是指数级的。探索长维度所需的迭代次数，需要正比于 $\log L/\log(1.1)$。

▷ **习题 30.3**[ø] 以相关的 N 维高斯分布为例，讨论 Skilling 方法的有效性随着维度的增加是如何变化的。假设马尔科夫链处于平衡状态，求一个关于舍弃概率的表达式。再讨论它随着高斯变量之间的相关程度增加是如何变化的。[**提示**：Skilling 方法在仿射变换下是不变的，因此可以通过查看可分离高斯的情况，来得出平衡状态的舍弃概率。]

这种方法与 Gilks 等（1994 年）所提出的“自适应方向采样”方法有些相似，但是蛙跳方法更简单而且可以适用于更多类型的分布。

30.5 作为通信信道的蒙特卡罗算法

在谈及加速蒙特卡罗方法时，考虑通信信息可能是一种有益的视角。当生成一个来自$P(\boldsymbol{x})$的样本时，发生了两次通信。

首先，从$P(\boldsymbol{x})$选择一个特定的$\boldsymbol{x}$，必定要求至少消耗$\log 1/P(\boldsymbol{x})$个随机比特。[回想一下，使用逆算术编码作为从已知分布生成样本的一种方法（第6.3节）。]

其次，样本的生成从可以计算$P^*(\boldsymbol{x})$的子程序（以及任何与$P^*(\boldsymbol{x})$的特性有关的其他子程序）传达了关于$P(\boldsymbol{x})$的信息。

比如，考虑一个哑Metropolis方法。在哑Metropolis方法中，提议$Q(\boldsymbol{x}';\boldsymbol{x})$与$P(\boldsymbol{x})$没有任何关系。只有在接受这一步骤，计算$P^*(\boldsymbol{x}')/P^*(\boldsymbol{x})$时，才会涉及$P(\boldsymbol{x})$的特性。因而，从真实分布$P(\boldsymbol{x})$到对计算$P(\boldsymbol{x})$特性感兴趣的用户之间的信道要通过一个瓶颈：所有关于P的信息是用一连串的接受和拒绝来表达的。如果用一个不同的分布$P_2(\boldsymbol{x})$代替$P(\boldsymbol{x})$，这一改变会产生影响的唯一方式是这个接受和拒绝串被改变。笔者没想到关于蒙特卡罗算法这种信息论观点会有多大用途，但是笔者认为这是一个有启发性的观点：如果目标是获得关于$P(\boldsymbol{x})$特性的信息，可以想像，认清信息流途经的信道并使信息传输速率最大化会很有帮助。

习题30.4 有一条广泛使用的经验法则：应该调整哑Metropolis方法的参数，使得接收率在1/2左右。信息论的观点为这条法则提供了一个简单的理由。称接受历史，即接受或拒绝决定的二进制字符串为$\boldsymbol{a}$。在算法运行了T步之后，学到的关于$P(\boldsymbol{x})$的信息会小于或等于$\boldsymbol{a}$所含信息量，因为关于P的所有信息都需要$\boldsymbol{a}$仲裁。$\boldsymbol{a}$的信息量的上界为$TH_2(f)$，其中f是接收率。设置$f=1/2$，所获取的关于P的信息界限会被最大化。

哑Metropolis方法的另一个有帮助的类比是进化方面的。每一个提议从当前状态$\boldsymbol{x}$生成一个后代$\boldsymbol{x}'$。然后，这两个个体会彼此竞争，而Metropolis方法使用一个有噪的适者生存法则。如果后代$\boldsymbol{x}'$比父母更适应环境（即$P^*(\boldsymbol{x}')>P^*(\boldsymbol{x})$，假设$Q/Q$因子取单位值），那么后代就取代其父母。生存法则有时也允许不适者取代其父母。因此，关于进化率的洞察可以运用于蒙特卡罗方法中。

习题30.5[3] 令$\boldsymbol{x}\in\{0,1\}^G$，设$P(\boldsymbol{x})$是一个可分离的分布

$$P(\boldsymbol{x})=\prod_g p(x_g) \tag{30.14}$$

且有$p(0)=p_0$和$p(1)=p_1$，比如$p_1=0.1$。假设哑Metropolis算法的提议密度Q涉及翻转状态$\boldsymbol{x}$的G个比特中的一部分：m。作为m的函数，分析链收敛到目标密度要花费多长时间。求出最优的m，并推断出Metropolis方法必须运行多久。

将这个结果与第19章关于在自然选择下人口演进的结果做一番比较。

一个标准Metropolis方法所能达到的最快速度，用信息论的术语来说，也就是每

次迭代约 1 比特，这一认识大大激发了我们加快算法速度的研究。本章已评述过几种降低随机游动行为的方法。这些方法是否也可以加快信息获取的速率呢？

习题 30.6[ㄔ] Gibbs 采样是提议分布确实与 $P(\boldsymbol{x})$ 有关的智能 Metropolis 方法，Gibbs 采样能否允许关于 $P(\boldsymbol{x})$ 的信息以高于每次迭代 1 比特的速率往外传？找出一个简例对这一问题进行精确研究。

习题 30.7[ㄔ] 哈密尔顿 - 蒙特卡罗法是另一种提议分布与 $P(\boldsymbol{x})$ 有关的智能 Metropolis 方法。哈密尔顿 - 蒙特卡罗法能否允许关于 $P(\boldsymbol{x})$ 的信息以高于每次迭代 1 比特的速率往外传？

习题 30.8[ㄙ] 在重要性采样中，权重 $w_r = P^*(\boldsymbol{x}^{(r)})/Q^*(\boldsymbol{x}^{(r)})$ 是一个浮点数，可以计算它并将它保存到计算结束。与此不同，在哑 Metropolis 方法中，比值 $a = P^*(\boldsymbol{x}')/P^*(\boldsymbol{x})$ 降为单个比特（“a 大于还是小于随机数 u？”）。因此，在原理上，重要性采样比哑 Metropolis 方法保有更多关于 P^* 的信息。你是否可以找到一个简例来表明：额外信息确实可用来使重要性采样的收敛速度比 Metropolis 方法快？你能否设计一个马尔科夫链蒙特卡罗算法，既可像 Metropolis 方法那样自适应地运行，又可像重要性采样那样保留更多关于 P^* 值的有用信息？

在第 19 章，注意到如果个体进行有性繁殖，N 个个体所组成的进化种群可以进化得更快。这一观察激发我们考察有多个进化和交互参数向量 $\boldsymbol{x}$ 的蒙特卡罗算法。

30.6 多状态方法

在一个多状态方法中，维护了多个参数向量 $\boldsymbol{x}$；每个参数向量都通过 Metropolis 和 Gibbs 之类的方法独自演进；这些向量之间也有交互。其目的是，所有向量 $\boldsymbol{x}$ 最终都是来自 $P(\boldsymbol{x})$ 的样本（如 Skilling 蛙跳法所述），或者与最终向量 $\boldsymbol{x}$ 关联的信息应该允许我们在 $P(\boldsymbol{x})$ 下逼近期望值（同重要性采样一样）。

基因方法

基因算法的支持者一般不把它描述成蒙特卡罗算法，但笔者认为这是一个正确的分类，而且可以证明理想的基因算法应该是一种收敛到特定密度的有效蒙特卡罗算法。

笔者将用 R 来表示种群中向量的个数。目标是满足 $P^*(\{\boldsymbol{x}^{(r)}\}_1^R) = \prod P^*(\boldsymbol{x}^{(r)})$。一个基因算法涉及两种或三种运动。

第一，一个状态向量受扰动的单个运动 $\boldsymbol{x}^{(r)} \to \boldsymbol{x}^{(r)'}$，可以使用已讨论过的任何一种蒙特卡罗方法完成。

第二，允许形如 $\boldsymbol{x}, \boldsymbol{y} \to \boldsymbol{x}', \boldsymbol{y}'$ 的交叉运动；在一个典型的交叉运动中，后代 $\boldsymbol{x}'$ 从它的单亲 $\boldsymbol{x}$ 获得一半的状态向量，从另一个单亲也获得一半的状态向量；基因算法成功的秘诀在于，参数 $\boldsymbol{x}$ 必须经过适当的编码，如果两个独立状态 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 都与 P^* 适应较

好,那么它们交换所产生的后代也会同样适应的几率比较高。这一约束条件在许多问题中很难满足,这也解释了为什么基因算法主要还是停留在谈论和炒作的层面,而严肃的专家却很少用到它。引入了交叉运动 $\boldsymbol{x},\boldsymbol{y}\rightarrow\boldsymbol{x}',\boldsymbol{y}'$ 之后,需要选择一个接受规则。获得一个有效算法的简易方法是,使用目标密度为 $P^*(\{\boldsymbol{x}^{(r)}\}_1^R)$ 的 Metropolis 规则来接受或拒绝交叉提议,这要用到如下比值对交叉前后的适应性做出比较。

$$\frac{P^*(\boldsymbol{x}')P^*(\boldsymbol{y}')}{P^*(\boldsymbol{x})P^*(\boldsymbol{y})} \tag{30.15}$$

如果交叉算符是可逆的,那么很容易证明这一过程满足详细平衡条件,因此是收敛到 $P^*(\{\boldsymbol{x}^{(r)}\}_1^R)$ 的马尔科夫链中一个有效部分。

▷ **习题 30.9**[求] 讨论上面两个算符:单个变异和使用 Metropolis 接受规则所进行的交叉,所给出的蒙特卡罗方法是否比只有一个状态向量、没有交叉的标准方法更有效。

第 19 章的有性社区比无性社区要更快获得信息,原因在于:交叉操作能够产生标准差为 $\sqrt{G}$ 的多样性,然后盲钟表匠可以通过杀掉不太适应的后代这种方式,来传达关于适应性函数的许多信息。和标准蒙特卡罗方法相比,因为没有用到杀灭,以上两种算符不能提供一个 $\sqrt{G}$ 的加速。要得到加速,需做到两件事:繁殖和死亡;而且至少其中之一必须有选择地进行。要么必须干掉不太适应的状态变量,要么必须允许更适应的状态向量生育更多的后代。勾画这些基本思想容易,要定义一种有效的方法去实现它却很困难。

习题 30.10[求] 设计一条初始规则和一条死亡规则,使得链能收敛到 $P^*(\{\boldsymbol{x}^{(r)}\}_1^R)$。

笔者相信这仍然是一个有待解决的研究问题。

粒子过滤器

粒子过滤器在涉及时间跟踪的推理问题中用得尤其多,它是一种混合了重要性采样和马尔科夫链蒙特卡罗思想的多状态方法。参见 Isard 和 Blake(1996 年)、Isard 和 Blake(1998 年)、Berzuini 等(1997 年)、Berzuini 和 Gilks(2001 年)以及 Doucet 等(2001 年)。

30.7 未必有帮助的方法

对于一个特定的马尔科夫链,通常会使用多个初始状态(如图 29.19 所示)。如果你担心是否能对一个复杂密度 $P(\boldsymbol{x})$ 采到好样本,是否能够通过确保初始点“在整个状态空间分布良好”这种方法,来保证仿真所产生的状态会在 $P(\boldsymbol{x})$ 的典型集合中

分布良好呢?

很遗憾,答案是否定的。比如,在递阶贝叶斯模型中,大量参数$\{x_n\}$可能会通过另一参数β(称为超参数)耦合在一些。例如,量$\{x_n\}$可能是独立的噪声信号,而β可能是噪声源的逆方差。β和$\{x_n\}$的联合分布可能是

$$P(\beta,\{x_n\}) = P(\beta)\prod_{n=1}^{N} P(x_n|\beta)$$

$$= P(\beta)\prod_{n=1}^{N} \frac{1}{Z(\beta)} e^{-\beta x_n^2/2}$$

其中$Z(\beta) = \sqrt{2\pi/\beta}$,$P(\beta)$是描述我们对噪声水平的忽视程度的宽分布。为了简单起见,忽略在实际问题中可能涉及的所有其他变量(诸如数据之类)。假设想通过Gibbs采样对$P(\beta,\{x_n\})$进行有效采样,也可以从条件分布$P(\beta|x_n)$采样β,然后从条件分布$P(x_n|\beta)$采样所有的x_n。[所得β的边缘分布应渐近地是宽分布$P(\beta)$。]

如果N很大,那么已知任何特定的$\{x_n\}$设置,β的条件分布会紧紧地集中在某个最可能的β值附近,其宽度与$1/\sqrt{N}$成正比。因此,会出现步长$\propto 1/\sqrt{N}$的缓慢随机游动在β轴上下的运动。

因此,对于初始化策略,是否可以通过使用"遍布状态空间"的初始条件,来巧妙地解决慢速收敛问题?令人伤心的是,答案是否定的。如果将点$\{x_n\}$分布得很广,实际上是在偏袒那些噪声水平$1/\beta$较大的初值。因而,参数为β的随机游动,在从$P(\beta|x_n)$抽取β之后,总会趋向于从β轴的一个端点开始运动。

进一步的阅读

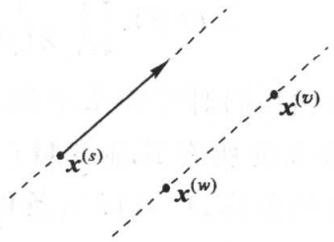
哈密尔顿-蒙特卡罗方法(Duane等,1987年)的综述见Neal(1993b)。这一优秀的论著对其他许多蒙特卡罗方法也做了综述,包括模拟退火和自由能量估计等相关主题。

30.8 进一步的习题

习题30.11[3] 哈密尔顿-蒙特卡罗方法的一个重要细节是,哈密尔顿动力学的仿真可以是不精确的,但必须是完全可逆的,也就是说,如果初始条件$(\boldsymbol{x},\boldsymbol{p})$运动到$(\boldsymbol{x}',\boldsymbol{p}')$,那么同一个仿真器必须能让初始条件$(\boldsymbol{x}', -\boldsymbol{p}')$运动到$(\boldsymbol{x}, -\boldsymbol{p})$,而且这个不精确的动力学必须能保持状态空间的体积。[算法30.1中的蛙跳法满足所有这些规则。]

解释为什么必须满足这些规则,并构造一个例子说明不满足时会出现什么问题。

习题 30.12[4] **切分采样的多状态思想**。探讨如下切分采样的多状态方法。与 Skilling 的多状态蛙跳法(第 30.4 节)一样,维护 S 个状态向量所组成的一个集合 $\{\boldsymbol{x}^{(s)}\}$。随机地选择其他两个状态向量 $\boldsymbol{x}^{(v)}$ 和 $\boldsymbol{x}^{(w)}$,并设 $\boldsymbol{y}=\boldsymbol{x}^{(v)}-\boldsymbol{x}^{(w)}$ 作为方向向量,在方向 $\boldsymbol{y}$ 上使用一维切分采样来更新一个状态向量 $\boldsymbol{x}^{(s)}$。对高度相关的多变量高斯分布之类的简化问题探讨这种方法。记住如果 $S-1$ 小于维数 N,那么这种方法本身不是各态历经的,因此它可能需要和其他方法混合使用。是否存在一类问题,用这种切分采样法比用标准方法(通过在坐标轴之间以循环方式来选择 $\boldsymbol{y}$ 或从一个高斯分布中随机地选择 $\boldsymbol{u}$ 求解的效果更好)?



30.9 习题解答

习题 30.3 的解答 考虑所有分量都有均值 0 和方差 1 的球面高斯分布。在第 n 维上,如果 $x_n^{(1)}$ 蛙跳到 $x_n^{(2)}$,所得提议坐标为

$$(x_n^{(1)})' = 2x_n^{(2)} - x_n^{(1)} \tag{30.16}$$

假设 $x_n^{(1)}$ 和 $x_n^{(2)}$ 都是服从分布 Normal(0,1)的高斯随机变量,$(x_n^{(1)})'$是服从分布 Normal $(0,\sigma^2)$的高斯随机变量,其中 $\sigma^2 = 2^2 + (-1)^2 = 5$。这个一维维度上的能量改变为

$$\frac{1}{2}[(2x_n^{(2)} - x_n^{(1)})^2 - (x_n^{(1)})^2] = 2(x_n^{(2)})^2 - 2x_n^{(2)}x_n^{(1)} \tag{30.17}$$

因此,能量上的典型变化是 $2\langle (x_n^{(2)})^2 \rangle = 2$。这种正的变化是一个坏消息。在 N 维维度上,进行蛙跳时,平衡态的典型能量变化为 $+2N$。这一移动的接收概率在量级上是

$$\mathrm{e}^{-2N} \tag{30.18}$$

这意味着 Skilling 方法,如所描述的那样,在高维问题上是无效的,至少没有收敛过一次。但是,它就有这么一个令人印象深刻的优势:它的收敛特性与变量之间的相关强度无关;这一特性就是哈密尔顿-蒙特卡罗和超松弛法都没有的。

关于第 31 章

将要遇到的一些神经网络模型与伊辛模型有关,伊辛模型是理想化的磁系统。理解伊辛模型的统计物理学原理对于理解这些神经网络问题并不是很关键,但笔者希望读者会觉得它们是有益的。

伊辛模型也和本书中其他几个主题有关。将使用如第 25 章所介绍的精确的基于树的计算方法来估算伊辛模型中感兴趣的特性。对于二进制图像,伊辛模型提供了粗略模型。而且伊辛模型和二维受限信道有关(比较第 17 章):一个二维条码,其中一个黑点可能没有完全被黑点包围,而一个白点也可能没有完全被白点包围,与低温下反铁磁体的伊辛模型相似。计算这种伊辛模型的熵等价于计算用于传递比特的受限信道的容量。

如果你想稍稍回顾一下关于统计物理学和热力学的知识,你会发现附录 B 是有所帮助的。我也推荐 Reif(1965 年)的著作。

第 31 章

伊辛模型

一个伊辛模型是一个相互磁耦合的自旋阵列(如,可处于状态 ±1 的原子)。我们说,在铁磁体模型中,如果一个自旋处于 +1 状态,那么它会在能量上有利于它的处于同一状态的近邻;在一个反铁磁体模型中,会出现与此相反的情况。本章讨论用于研究伊辛模型的两种计算方法。

设具有 N 个自旋的伊辛模型的状态 $\boldsymbol{x}$ 为一个向量,其中每个分量 x_n 取值 -1 或 +1。如果两个自旋 m 和 n 是邻居,则记 $(m,n)\in\mathcal{N}$。相邻自旋之间的耦合为 J。若 m 和 n 是相邻的,定义 $J_{mn}=J$,否则 $J_{mn}=0$。状态 $\boldsymbol{x}$ 的能量为

$$E(\boldsymbol{x};J,H)=-\left[\frac{1}{2}\sum_{m,n}J_{mn}x_mx_n+\sum_n Hx_n\right] \tag{31.1}$$

其中 H 是作用场。如果 $J>0$,则该模型是铁磁性的;如果 $J<0$,则它是反铁磁性的。加入因子 1/2,因为在第一个和式中每对状态被计算两次,一次是 (m,n),另一次是 (n,m)。在温度 T 的平衡态上,状态为 $\boldsymbol{x}$ 的概率是

$$P(\boldsymbol{x}|\beta,J,H)=\frac{1}{Z(\beta,J,H)}\exp[-\beta E(\boldsymbol{x};J,H)] \tag{31.2}$$

其中 $\beta=1/k_BT$,k_B 是玻尔兹曼常数,且

$$Z(\beta,J,H)\equiv\sum_{\boldsymbol{x}}\exp[-\beta E(\boldsymbol{x};J,H)] \tag{31.3}$$

伊辛模型的相关性

伊辛模型之所以是相关的,有以下几个重要原因。

首先,作为具有相变的磁系统模型,伊辛模型是非常重要的。统计物理学中的普适理论表明,所有具有相同维数(此处指二维)和相同对称性的系统都有等效的关键性质,即由其相变所表示的比例法则是相同的。所以通过研究伊辛模型,不仅可以找出关于磁的相变,而且可以找出关于许多其他系统的相变。

其次,如果把能量函数推广为

$$E(\boldsymbol{x};\boldsymbol{J},\boldsymbol{h})=-\left[\frac{1}{2}\sum_{m,n}J_{mn}x_mx_n+\sum_n h_nx_n\right] \tag{31.4}$$

其中耦合 J_{mn} 和应用场 h_n 并不是恒定不变的,我们得到一组模型,物理学家称之为“自旋玻璃”,神经网络界称之为“Hopfield 网”或“玻尔兹曼机”。在某些这类模型中,所有自旋都互为邻居,物理学家称这种系统为“无穷范围”自旋玻璃,而神经网络学者称之为“全连通”网络。

最后,伊辛模型本身作为统计模型也是非常有用的。

本章将用两种不同的计算方法来研究伊辛模型。

统计物理学中一些值得注意的关系

我们想从计算中得到尽可能多的信息。例如，考虑系统的热容，定义为

$$C \equiv \frac{\partial}{\partial T}\overline{E} \tag{31.5}$$

其中

$$\overline{E} = \frac{1}{Z}\sum_{x} \exp(-\beta E(x))E(x) \tag{31.6}$$

为了得出系统的热容，我们也许会天真地以为必须增加温度并测量能量的变化。然而，系统的热容与其在恒定温度下能量的波动关系密切。从配分函数

$$Z = \sum_{x} \exp(-\beta E(x)) \tag{31.7}$$

开始，通过对 β 求导可得平均能量

$$\frac{\partial \ln Z}{\partial \beta} = \frac{1}{Z}\sum_{x} -E(x)\exp(-\beta E(x)) = -\overline{E} \tag{31.8}$$

再次求导给出能量的方差

$$\frac{\partial^2 \ln Z}{\partial \beta^2} = \frac{1}{Z}\sum_{x} E(x)^2\exp(-\beta E(x)) - \overline{E}^2 = \langle E^2\rangle - \overline{E}^2 = \mathrm{var}(E) \tag{31.9}$$

但是，热容也是$\overline{E}$关于温度的导数

$$\frac{\partial \overline{E}}{\partial T} = -\frac{\partial}{\partial T}\frac{\partial \ln Z}{\partial \beta} = -\frac{\partial^2 \ln Z}{\partial \beta^2}\frac{\partial \beta}{\partial T} = -\mathrm{var}(E)(-1/k_B T^2) \tag{31.10}$$

所以，对于温度 T 处的任何系统，有

$$C = \frac{\mathrm{var}(E)}{k_B T^2} = k_B \beta^2 \mathrm{var}(E) \tag{31.11}$$

因此如果观测系统平衡态时能量的方差，就可以估算其热容。

笔者发现这几乎是一种荒谬的关系。考虑具有有限状态集的一个系统，并设想对它进行加热。在高温处，所有状态都是等可能的，所以平均能量基本上是一个常量且热容基本上是零。但从另一方面来说，所有状态都是等可能的，很显然能量必然有波动。那么，热容如何能和波动有关呢？答案就在于上面所说的“基本上是零”。在高温处，热容并不是完全为零的，它只不过趋于零。而且在高温时，它以$\frac{\mathrm{var}(E)}{k_B T^2}$趋于零，而量 $\mathrm{var}(E)$趋于一个常量。因此，有限系统在高温时热容的这种 $1/T^2$特性颇具普遍性。

可以把 $1/T^2$因子看做历史的偶然。要是用$\beta = \frac{1}{k_B T}$来定义温度刻度的话，那么热容的定义将是

$$C^{(\beta)} \equiv \frac{\partial \overline{E}}{\partial \beta} = \mathrm{var}(E) \tag{31.12}$$

所以热容和波动会是同一个量。

▷ **习题 31.1**[ℐ] [在统计物理学的章节中,我们宁愿把物理系统的熵称为 S 而非 H;设 $k_B=1$。]

在温度 $T=1/\beta$ 处,状态为 $\boldsymbol{x}$ 的系统的熵为

$$S=\sum p(\boldsymbol{x})[\ln 1/p(\boldsymbol{x})] \tag{31.13}$$

其中

$$p(\boldsymbol{x})=\frac{1}{Z(\beta)}\exp[-\beta E(\boldsymbol{x})] \tag{31.14}$$

(a) 证明

$$S=\ln Z(\beta)+\beta\overline{E}(\beta) \tag{31.15}$$

其中$\overline{E}(\beta)$是系统的平均能量。

(b) 证明

$$S=-\frac{\partial F}{\partial T} \tag{31.16}$$

其中自由能 $F=-kT\ln Z$ 且 $kT=1/\beta$。

31.1 伊辛模型——蒙特卡罗仿真

本节使用简单的 Gibbs 采样法研究二维平面伊辛模型。从某个初始状态开始,随机选取一个自旋 n,其他自旋的状态为已知且计算温度时,它应为 $+1$ 的概率

$$P(+1\,|\,b_n)=\frac{1}{1+\exp(-2\beta b_n)} \tag{31.17}$$

其中 $\beta=1/k_BT$ 且 b_n是局部场

$$b_n=\sum_{m:(m,n)\in\mathcal{N}} Jx_m+H \tag{31.18}$$

[在式(31.17)中出现了因子2,因为两个自旋状态是$\{+1,-1\}$而非$\{+1,0\}$。]设自旋 n 以此概率设为 $+1$,否则设为 -1;然后随机选取下一个要更新的自旋。经过足够多次迭代之后,这个过程收敛于平衡分布(31.2)。除了 Gibbs 采样公式(31.17),还可以采用 Metropolis 算法;在后者中,考虑给定的自旋由当前状态 x_n翻转所造成的能量改变

$$\Delta E=2x_nb_n \tag{31.19}$$

并把这个变化与概率联系在一起

$$P(\text{接受};\Delta E,\beta)=\begin{cases}1, & \Delta E\leqslant 0\\ \exp(-\beta\Delta E), & \Delta E>0\end{cases} \tag{31.20}$$

这个过程大致会使接受在能量上不利移动的概率翻倍,所以它可能是一个更有效的采样器;但在非常低的温度处,Gibbs 采样和 Metropolis 算法的相对优点就不那么明显了。

矩形几何

首先仿真一个如图 31.1 所示的具有矩形几何和周期边界条件的伊辛模型。两个自旋之间的连线表明它们是邻居。设外部场 $H=0$ 并考虑 $J=\pm 1$ 这两种情况，分别是铁磁体和反铁磁体。

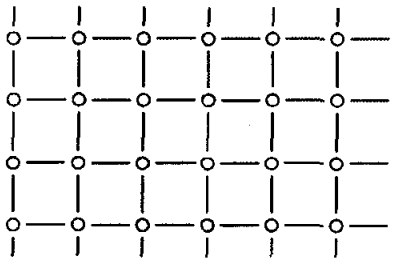


图 31.1 矩形伊辛模型

从一个很高的温度($T=33,\beta=0.03$)开始，并每 I 次迭代改变一次温度，首先把温度逐渐降到 $T=0.1,\beta=10$，然后再逐渐增加至一个高的温度。这个过程大致检查了在每个温度上是否“已经达到平衡状态”；如果没有达到，希望在我们画出的图中看到某些磁滞现象。如果假设采用递减和递增温度来进行的两个运行实际上相互独立，则该过程也给出了结果可重复性的概念。

在每一个温度点，笔者记录了每个自旋的平均能量和能量的标准差，及磁化 m 的均方值

$$m=\frac{1}{N}\sum_{n}x_n \tag{31.21}$$

必须做出的一个很需要技巧的决策是在确定一个新温度之后，过多久再开始进行这些测量；很难检测“平衡”——甚至是给出一个系统“处于平衡态”的清晰定义也不容易！[但在第 32 章中，将看到关于此问题的一个解答。]我的粗略策略是令每个温度点的迭代次数 I 为自旋次数 N 的几百倍，并且丢弃这些迭代中的前 1/3。当 $N=100$ 时，对于任一给定温度，笔者发现需要超过100 000次迭代才能达到平衡。

对于 $J=1$ 的较小 N 值的计算结果

仿真一个 $l\times l$ 网格，$l=4,5,\cdots,10,40,64$。快速思考一下预计会得到什么结果。在低温处，可以预料系统是处于基态的。$J=1$ 的矩形伊辛模型有两个基态(如图 31.2 所示)，全 +1 态和全 -1 态。任何一个基态的各个自旋的能量都是 -2。在高温处，自旋是独立的，所有状态是等可能的，并且预计能量将以正比于 $1/\sqrt{N}$ 的标准差围绕均值 0 上下波动。

让我们看一些结果。在所有图中，用 $k_B=1$ 来表示温度 T。基本的图片只有 16 个自旋(图 31.3 最上面的一个小图)：能量单调增加。当把自旋数增加到 100 时(图

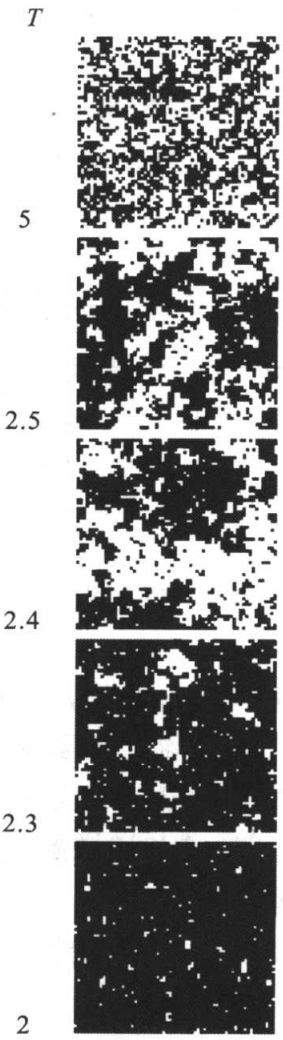


图 31.2 $J=1$ 的矩形伊辛模型在一列温度 T 下的样本状态

31.3 最下面的一个小图),会出现一些新的细节。首先,正如所预料的一样,在高温处,波动以 $1/\sqrt{N}$下降。其次,在中等温度时,波动相对较大。这是一个“集体现象”的信号,在这种情况下是相变。只有具有无穷 N 的系统才显示真正的相变,但当 $N=100$ 时,我们得到临界波动的迹象。图 31.5 显示了关于 $N=100$ 和 $N=4\ 096$ 时图的详情。图 31.2 显示了从 $N=4\ 096$ 个自旋在一组递减温度下的仿真中所得出的一系列典型状态。

与 Schottky 异常的对比

作为温度的函数,热容的峰值发生在任何具有有限能量级的系统中;峰值本身并非相变的证据。这种峰值在传统热力学中被视为异常,因为具有无穷能量级的“正常”系统(如盒子中的粒子),其热容是常数或温度的递增函数。相反的,具有有限级的系统在热容图中会产生小尖头(如图 31.4 所示)。

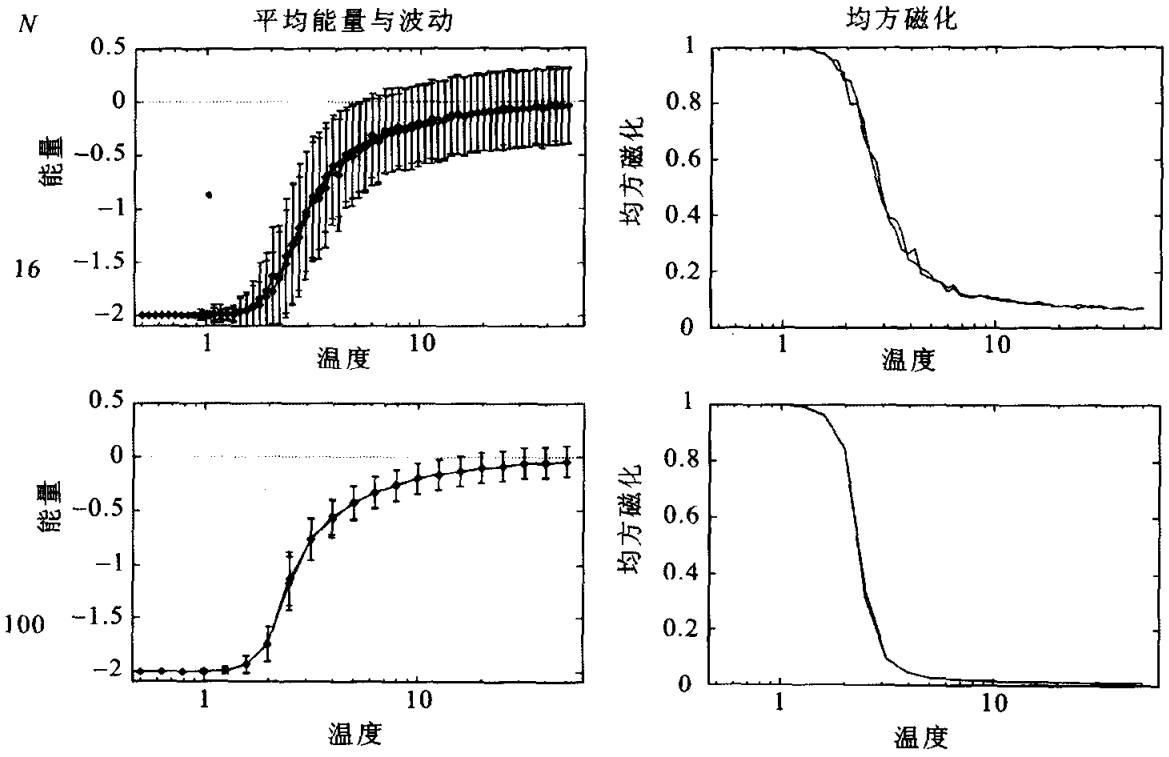


图 31.3 $J=1$ 的矩形伊辛模型的蒙特卡罗仿真。平均能量和能量波动以温度的函数表示(左边)。均方磁化以温度的函数表示(右边)。在顶排,$N=16$,而在底排,$N=100$。更大的 N 值请参见后面的图。

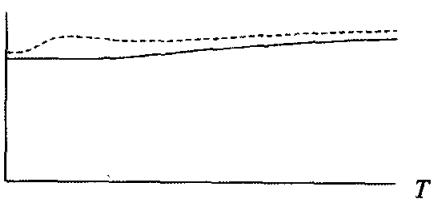


图 31.4 用于解释 Schottky 异常的示意图。曲线表示两种气体的热容与温度的函数关系。较低的曲线表示一个正常气体,其热容是温度的递增函数。较高的曲线有一个小的热容峰值,称为 Schottky 异常(至少在剑桥大学是这样说)。这个峰值由具有有限可达状态的磁自由度的气体所产生。

现在回顾这种系统的最简单情况,一个具有状态 $x=0$(能量 0)和 $x=1$(能量 ε)的二级系统。平均能量为

$$E(\beta)=\varepsilon\frac{\exp(-\beta\varepsilon)}{1+\exp(-\beta\varepsilon)}=\varepsilon\frac{1}{1+\exp(\beta\varepsilon)} \tag{31.22}$$

而它对 β 的导数为

$$\mathrm{d}E/\mathrm{d}\beta=-\varepsilon^2\frac{\exp(\beta\varepsilon)}{[1+\exp(\beta\varepsilon)]^2} \tag{31.23}$$

所以热容为

$$C=\mathrm{d}E/\mathrm{d}T=-\frac{\mathrm{d}E}{\mathrm{d}\beta}\frac{1}{k_{\mathrm{B}}T^2}=\frac{\varepsilon^2}{k_{\mathrm{B}}T^2}\frac{\exp(\beta\varepsilon)}{[1+\exp(\beta\varepsilon)]^2} \tag{31.24}$$

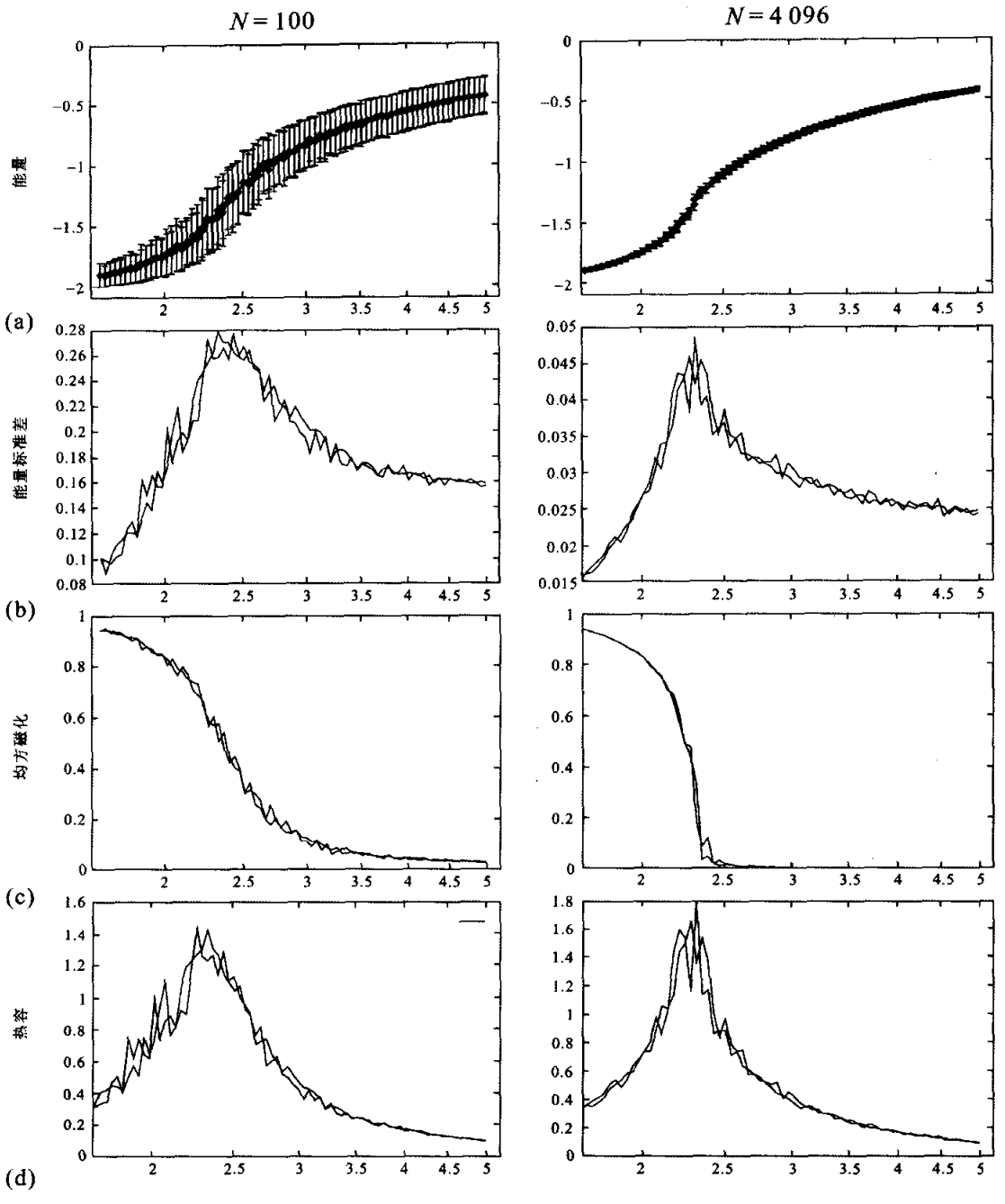


图 31.5 $J=1$ 的矩形伊辛模型的蒙特卡罗仿真详情。(a) 平均能量和能量的波动,以温度的函数表示。(b) 能量波动(标准差)。(c) 均方磁化。(d) 热容。

且能量波动由 $\mathrm{var}(E) = Ck_{\mathrm{B}}T^2 = -\mathrm{d}E/\mathrm{d}\beta$ 给出,该式在式(31.23)中计算。图 31.6 画出了热容和波动。此处的关键信息是,虽然 Schottky 异常在热容中有峰值,但在其波动中却没有峰值;能量方差只是随温度单调递增到一个与独立自旋数成正比的值。因此有意思的是波动的峰值,而非热容的峰值。伊辛模型在其波动中有一个这样的峰值,如图 31.5 第二行所示。

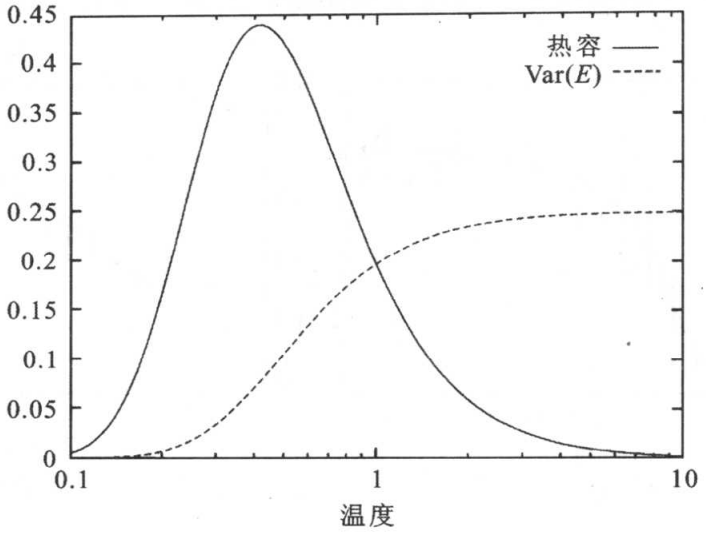


图 31.6 Schottky 异常——分隔为 $\varepsilon = 1$ 且 $k_B = 1$ 的二级系统的热容和能量波动，以温度的函数表示。

$J = -1$ 的矩形伊辛模型

在 $J = -1$ 情况，希望出现什么呢？一个无穷系统的基态是两种棋盘模式（如图 31.7 所示），且它们每个自旋的能量为 -2，和 $J = 1$ 模型的基态一样。是否可以使这种类比更进一步呢？稍微思考一下就可以肯定，两个系统在棋盘对称操作下是等价的。如果你取一个处于某状态中的 $J = 1$ 无穷系统，将处于无穷大的棋盘上所有黑色方块的自旋翻转，并设 $J = -1$（如图 31.8 所示），则不会改变能量。（当然，改变了磁化。）因此可以预料两个系统的所有热力学性质在零应用场的情况下是一样的。

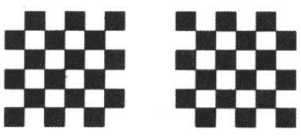


图 31.7 $J = -1$ 的矩形伊辛模型的两种基态

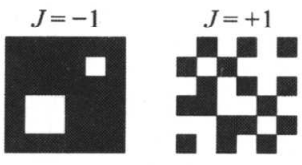


图 31.8 $J = \pm 1$ 的矩形伊辛模型中能量相同的两种状态

但是这里潜伏着一个精妙之处。你已经找到它了吗？我们所仿真的是具有周期边界条件的有限网格。如果网格的大小在任何方向都是奇数，则棋盘操作不再是相对于 $J = +1$ 到 $J = -1$ 对称操作，因为棋盘在边界处不匹配。这意味着对于奇数大小的系统，$J = -1$ 的系统基态会有大于 2 的衰退，且这些基态的能量将不会低到每自旋只有 -2。因此，在奇数大小的系统中，预计 $J = \pm 1$ 两种情形之间存在定性的差异。可以预料，在小系统中，这些差异特别明显。这些失败是由边界引入的，而边界长度以系统大小的平方根增加，因此能量和系统熵与边界相关的失败部分效应将以 $1/\sqrt{N}$ 减小。

图 31.9 比较了 $N=25$ 的铁磁体和反铁磁体模型的能量。这里的差异是显著的。

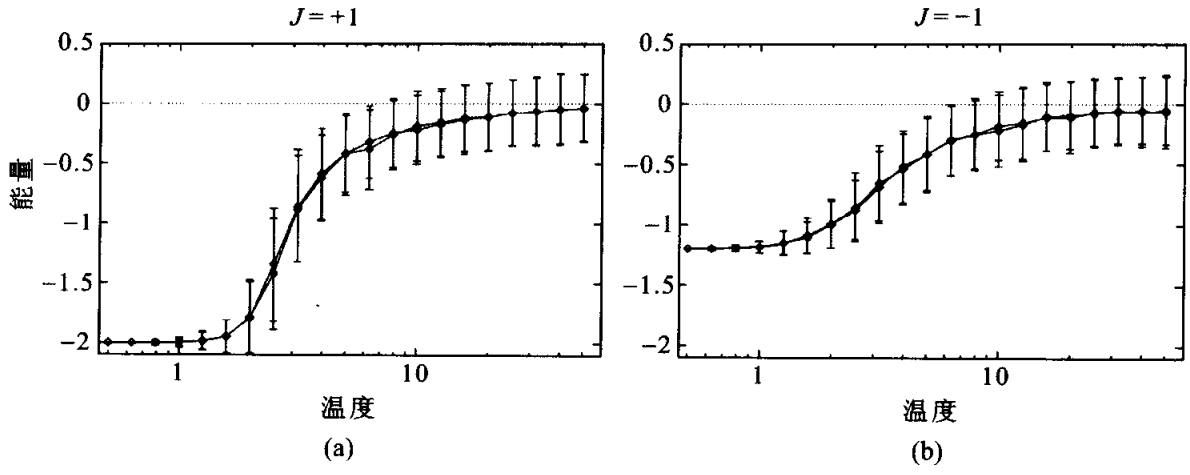


图 31.9 $J=\pm 1$ 且 $N=25$ 的矩形伊辛模型的蒙特卡罗仿真。平均能量和能量波动，以温度的函数表示。(a) $J=1$。(b) $J=-1$。

三角伊辛模型

可以对三角伊辛模型重复这些计算。我们预计 $J=\pm 1$ 的三角伊辛模型会显示出与矩形伊辛模型不同的物理性质吗？$J=1$ 的模型大概和与其相对应的矩形模型具有大体相似的性质。但是 $J=-1$ 的情况与以前的却有本质上的不同。想一想：不存在无失败的基态；在任何一个状态中，必定存在失败——具有相同符号的邻居对。与奇数大小的矩形模型不同，失败不是由周期边界条件引起的。每组 3 个互为邻居的自旋必处于一个失败的状态，如图 31.10 所示（实线表示为能量贡献了 $-|J|$ 的

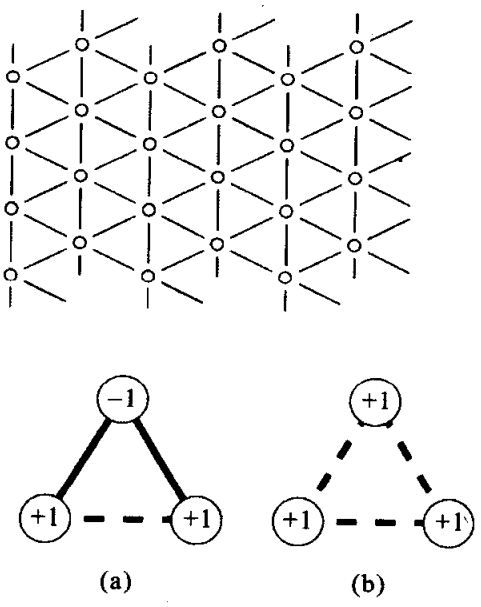


图 31.10 在一个反铁磁性三角伊辛模型中，任意 3 个相邻自旋都是失败的。3 - 自旋的 8 个可能配置中，6 个具有能量 $-|J|$(a)，而 2 个具有能量 $3|J|$(b)。

“开心”耦合;虚线表示贡献了$|J|$的“不开心”耦合。)因此,在低温处当然会期望得到不同的特性。实际上我们可能期望这个系统在绝对零度处具有非零熵。(“三角模型违反了热力学第三定律!”)

现在看一些结果。样本状态如图 31.12 所示,而图 31.11 显示了 $N=4\ 096$ 时的能量、波动和热容。注意,对于 $J=\pm 1$,结果有多么不同。在 $J=-1$ 的情况中,能量标准差中根本不存在峰值。这表明反铁磁性的系统无法使相变达到一个长期有序的状态。

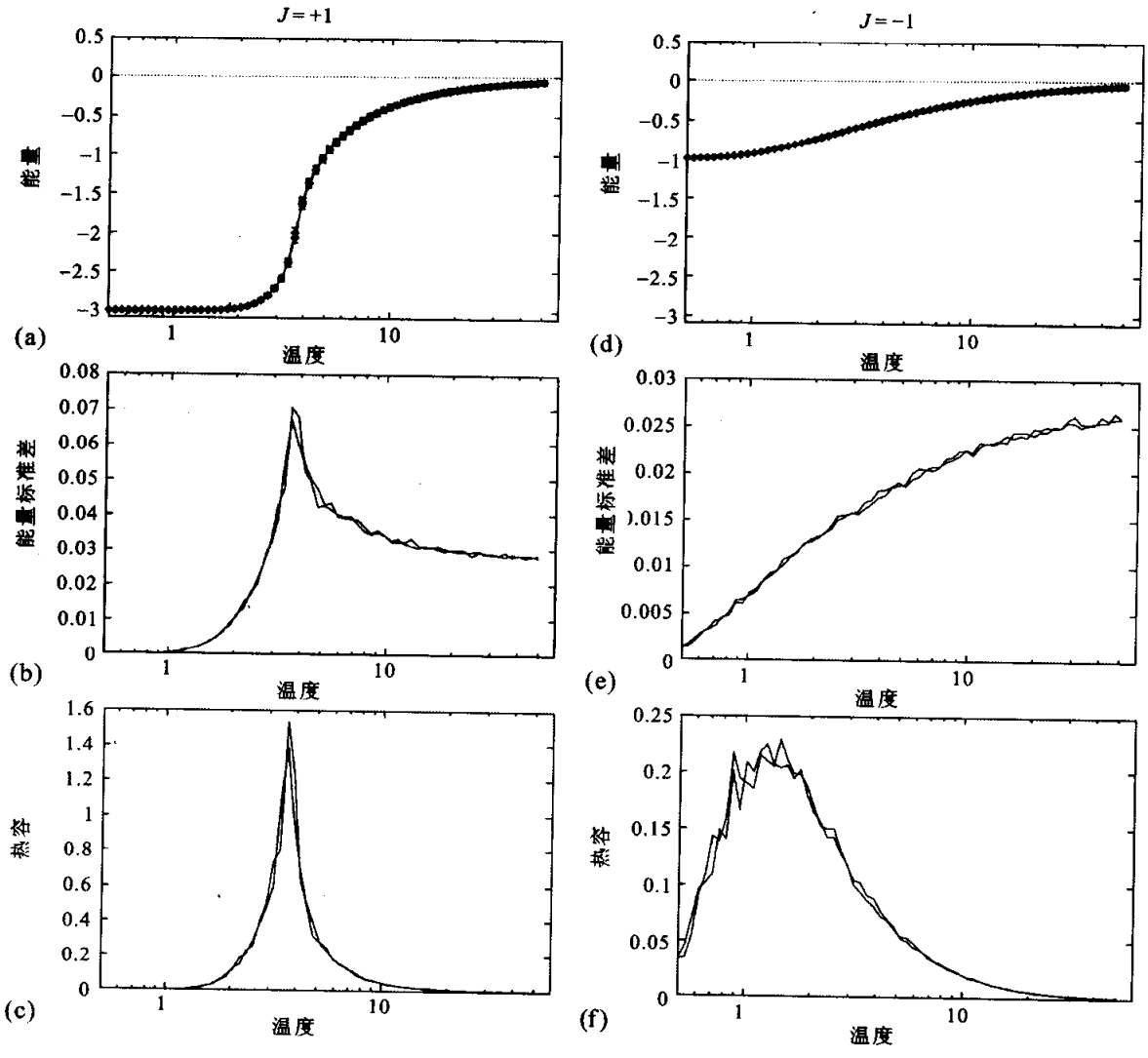


图 31.11 $J=\pm 1$ 且 $N=4\ 096$ 的三角伊辛模型的蒙特卡罗仿真。(a ~ c) $J=1$。(d ~ f) $J=-1$。(a,d) 以温度函数所表示的平均能量和能量的波动。(b,e) 能量中的波动(标准差)。(c,f) 热容。

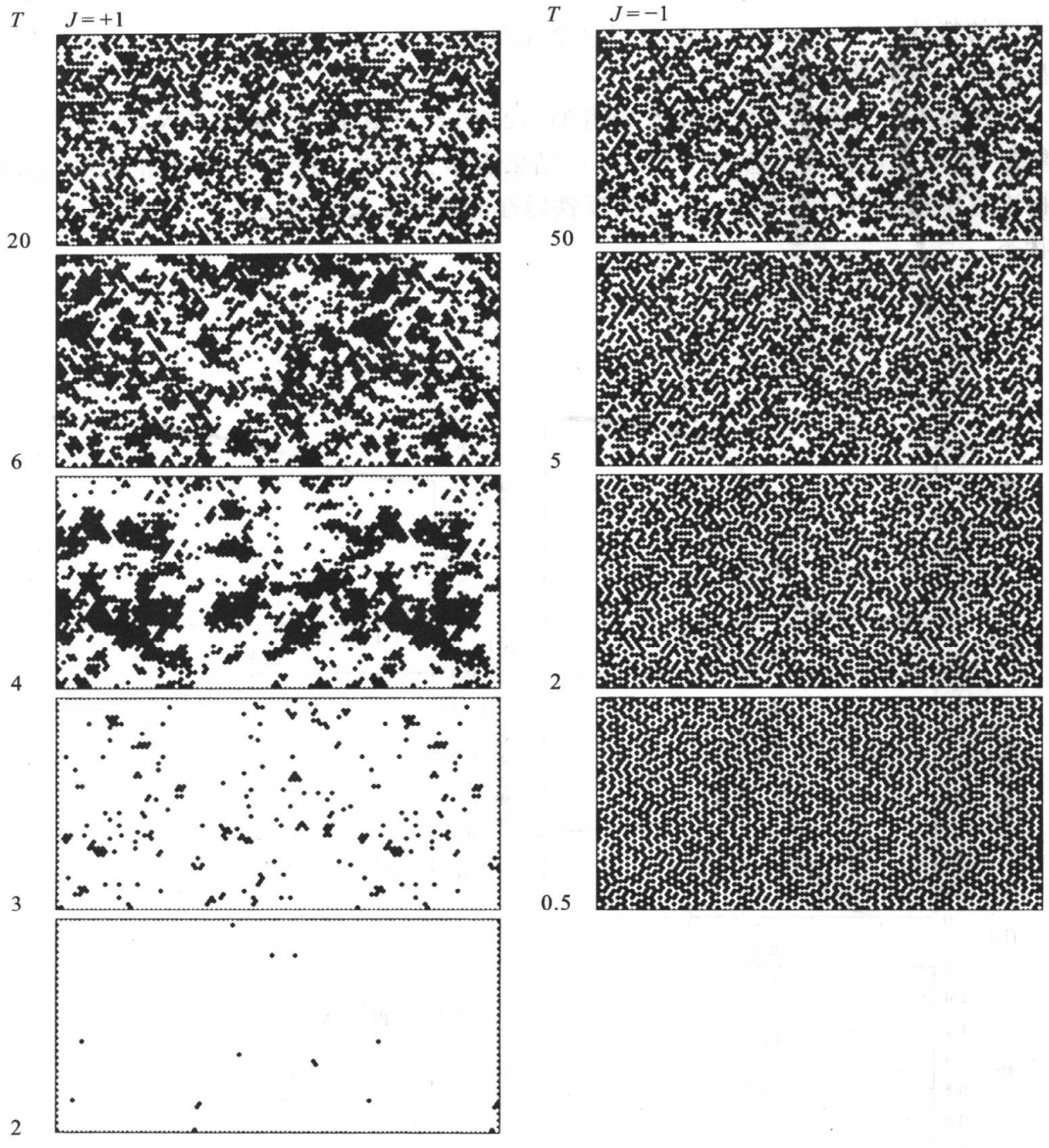


图 31.12 $J=1$ 和 $J=-1$ 三角伊辛模型的样本状态。顶部为高温，底部为低温。

31.2 伊辛模型划分函数的直接计算

现在考察一种全然不同的伊辛模型方法。*转移矩阵法*是一种精确而抽象的方法，它从如下划分函数中得到模型的物理性质

$$Z(\beta,\boldsymbol{J},\boldsymbol{b}) \equiv \sum_{x} \exp[-\beta E(\boldsymbol{x};\boldsymbol{J},\boldsymbol{b})] \tag{31.25}$$

其中求和是对所有状态 $\boldsymbol{x}$ 进行的，而温度的倒数为 $\beta = 1/T$。[与平常一样，设 $k_B = 1$。]自由能量由 $F = -\frac{1}{\beta}\ln Z$ 给出。状态数为 2^N，因此对于大的 N 值，直接计算划分函数是不可能的。为了避免显式地列举所有的全局状态，可以采用与第 25 章所讨论的和－积算法相似的技巧。集中考虑如下模型：具有宽度 W 的细长条的形式、在两个方向上都符合周期边界条件，并且沿着该模型的长度方向进行迭代，根据前 $l-1$ 个位置的部分划分函数，得到位置 l 处的一组部分划分函数。每次迭代都涉及对边界所有状态的求和。这种运算同条带宽度 W 成指数级关系。最后的技巧是注意：如果系统在其长度上具有平移不变性，那么只需做一次迭代就可以找到具有任意长度的系统的性质。

计算任务变成对一个 $S \times S$ 矩阵及其特征值的计算，其中 S 是在边界处需要考虑的微状态数。对于无穷长的细条，幅度最大的特征值给出了划分函数。

下面是更详细一些的解释。将 C 列细条的状态标记为 $s_1, s_2, \cdots, s_C$，每个 s 取 0 ~ $2^W - 1$之间的整数值。s_c的第 r 比特表示 r 行 c 列中的自旋是朝上还是朝下。划分函数为

$$Z = \sum_{x} \exp(-\beta E(\boldsymbol{x})) \tag{31.26}$$

$$= \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} \exp\Big(-\beta \sum_{c=1}^{C} \mathscr{E}(s_c, s_{c+1})\Big) \tag{31.27}$$

其中 $\mathscr{E}(s_c, s_{c+1})$是一个适当定义的能量，而且，如果想要周期的边界条件，可定义 s_{c+1} 为 s_1。对 $\mathscr{E}$的一个定义为

$$\mathscr{E}(s_c, s_{c+1}) = \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c, n\in c+1}} J x_m x_n + \frac{1}{4} \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c, n\in c}} J x_m x_n + \frac{1}{4} \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c+1, n\in c+1}} J x_m x_n \tag{31.28}$$

能量的这种定义（如图 31.13 所示）有一个很好的性质（对于矩形伊辛模型），它定义一个关于它的两个序数 s_c和 s_{c+1}对称的矩阵。需要 1/4 的因子，因为垂直链接被计数 4 次。定义

$$M_{ss'} = \exp(-\beta \mathscr{E}(s, s')) \tag{31.29}$$

然后继续式(31.27)

$$Z = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} \Big[\prod_{c=1}^{C} M_{s_c, s_{c+1}}\Big] \tag{31.30}$$

$$= \mathrm{Trace}[\boldsymbol{M}^C] \tag{31.31}$$

$$= \sum_{a} \mu_a^C \tag{31.32}$$

其中 $\{\mu_a\}_{a=1}^{2^W}$是 $\boldsymbol{M}$ 的特征值。当细条 C 的长度增加时，最大特征值 $\mu_{\max}$会主导 Z：

$$Z \to \mu_{\max}^C \tag{31.33}$$

因此在无穷细长条的极限情况下，每个自旋的自由能由下式给出。

$$f = -kT\ln Z/(WC) = -kTC\ln \mu_{\max}/(WC) = -kT\ln \mu_{\max}/W \tag{31.34}$$

可从矩阵 M 的最大特征值得出一个细长条的所有热力学性质，这真是太好了！

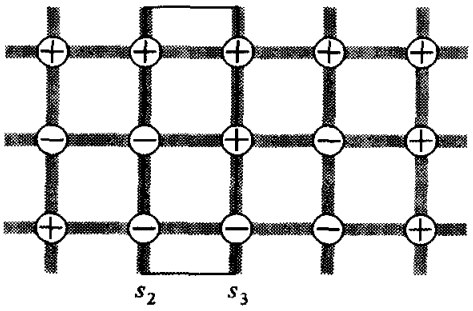


图 31.13 帮助解释定义(31.28)的示意图。$\mathscr{E}(s_2,s_3)$ 计算了矩形中对于能量的所有贡献。总能量通过遍历矩形来给出。在矩形内部，每个水平联结被计数一次；每个垂直联结有一半在矩形中(而另一半在相邻矩形中)，因此 $\mathscr{E}(s_2,s_3)$ 中只包含能量的一半；在第 2 项中出现 1/4 因子是因为 m 和 n 都遍历了列 c 中的所有结点，因此每个联结被访问了两次。

对于这里所显示的状态，假设在最顶和最底是周期边界条件，则 $s_2=(100)_2$，$s_3=(110)_2$，水平联结贡献了 $+J$ 给 $\mathscr{E}(s_2,s_3)$，且垂直联结对左边贡献了 $-J/2$，对右边贡献了 $-J/2$。因此 $\mathscr{E}(s_2,s_3)=0$。

计算

笔者计算了几何形状如图 31.14 所示的细长条伊辛模型的划分函数。

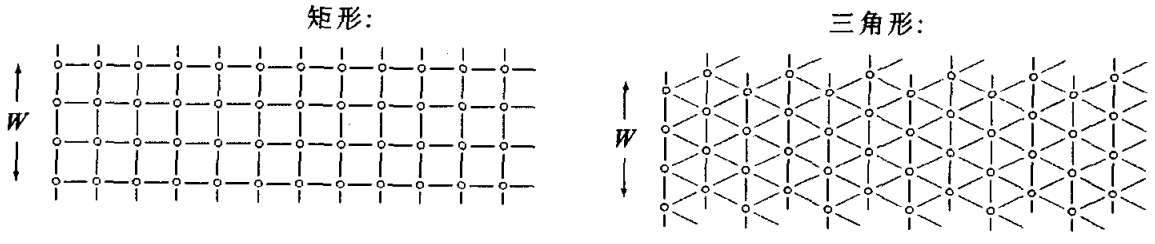


图 31.14 两个细长条伊辛模型。两个自旋之间的连线表示它们是邻居。带状宽度为 W，长度无穷。

和上一节一样，设置应用场 H 为零并考虑 $J=\pm1$ 的两种情况，它们分别是铁磁性和反铁磁性的。当 $H=0$，而宽度从 $W=2$ 到 8 变化时，笔者计算了以 β 函数表示的每自旋的自由能 $f(\beta,J,H)=F/N$。

计算思路

只需要最大特征值。有几条途径可以求得该值，例如，用一个初始向量对矩阵迭代相乘。因为矩阵是全正的，我们知道主特征向量也是全正的(Frobenius - Perron 定理)，因此一个合理的初始向量是 $(1,1,\cdots,1)$。这个迭代过程可能比直接计算所有特征值要快。笔者还是把它们都计算出来了，这样做的优点是可以通过式(31.32)计算有限长度线条以及无限长度线条的自由能。

对图的讨论

对于高温，所有伊辛模型都应该表现出相同的特性：自由能是由熵主导的，且每自旋熵为ln(2)。每自旋的平均能量趋于零。每自旋的自由能将趋于 $-\ln(2)/\beta$。自由能如图 31.15 所示。

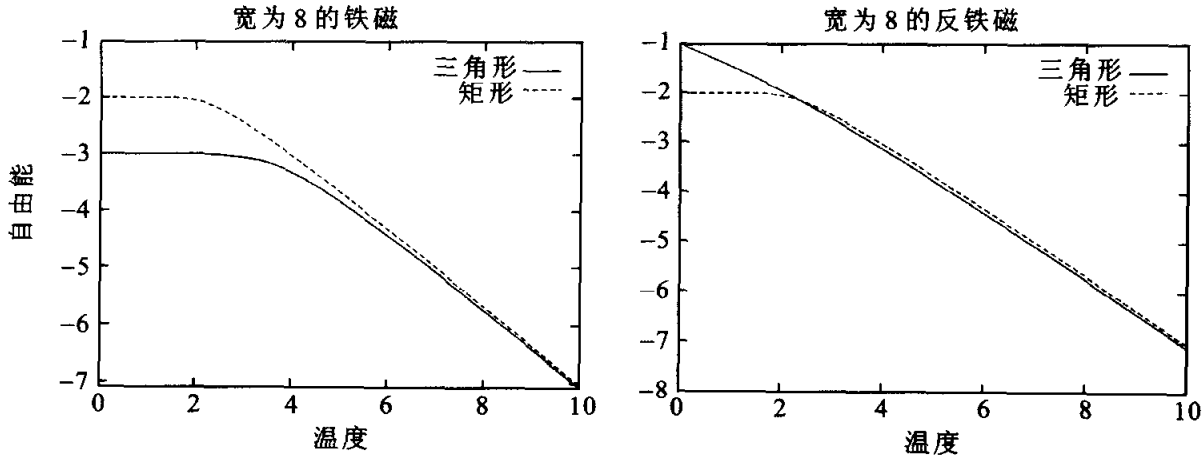


图 31.15 细长条伊辛模型的各自旋自由能。注意对于三角反铁磁性的情形，在 $T=0$ 处梯度非零。

可以从自由能中得到的一个有趣性质是基态的衰退。当温度降为零时，玻尔兹曼分布收敛于基态。如果基态是衰退的（即多个基态具有相同的能量），那么当 $T\to 0$ 时熵不为零。可以通过 $S=-\partial F/\partial T$ 从自由能中找到熵。

三角反铁磁体在绝对零度处的熵看来大约是 0.3，即，大约是其在最高温度处的熵值的一半（如图 31.16 所示）。平均能量以温度函数表示如图 31.17 所示。它是采用恒等式 $\langle E\rangle=-\partial\ln Z/\partial\beta$ 计算出来的。

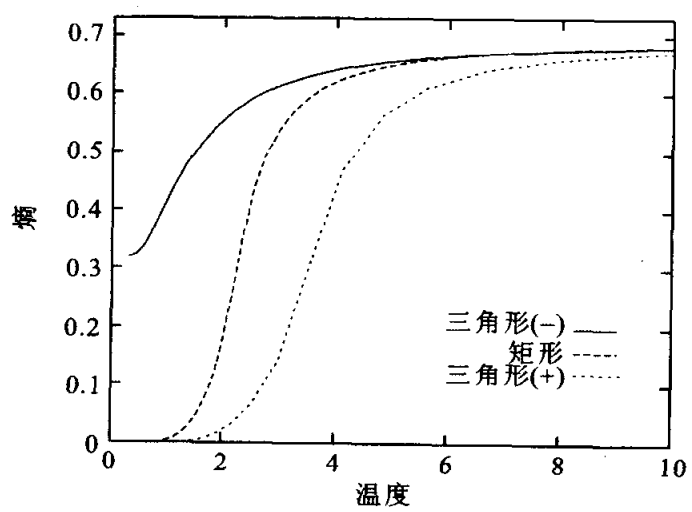


图 31.16 宽度为 8 的伊辛系统的熵（以 nats 为单位），以温度的函数表示，它是通过对图 31.15 中的自由能曲线求导得到的。矩形铁磁性物质和反铁磁性物质具有相同的热性质。对于三角系统，上部曲线（-）表示反铁磁体，下部曲线（+）表示铁磁体。

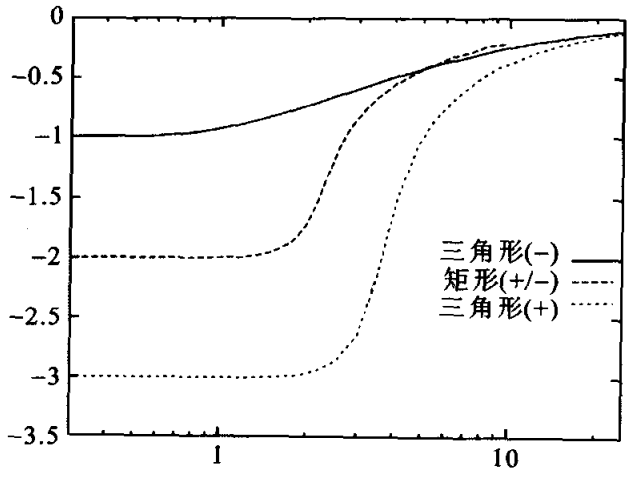


图 31.17 宽度为 8 的细长条伊辛模型的温度与平均能量。与图 31.3 进行对比。

对于宽度为 4 和 8 的三角模型，图 31.18 显示了估计热容，以温度的函数表示（大致地对平均能量取导数）。图 31.19 显示了能量波动，以温度的函数表示。所有这些图应该显示光滑曲线；曲线的粗糙之处是不准确的数值所造成的。任何相变的本性都是不明显的，但是曲线图似乎和如下断言相符：铁磁体表现出相变，而反铁磁体不表现出相变。

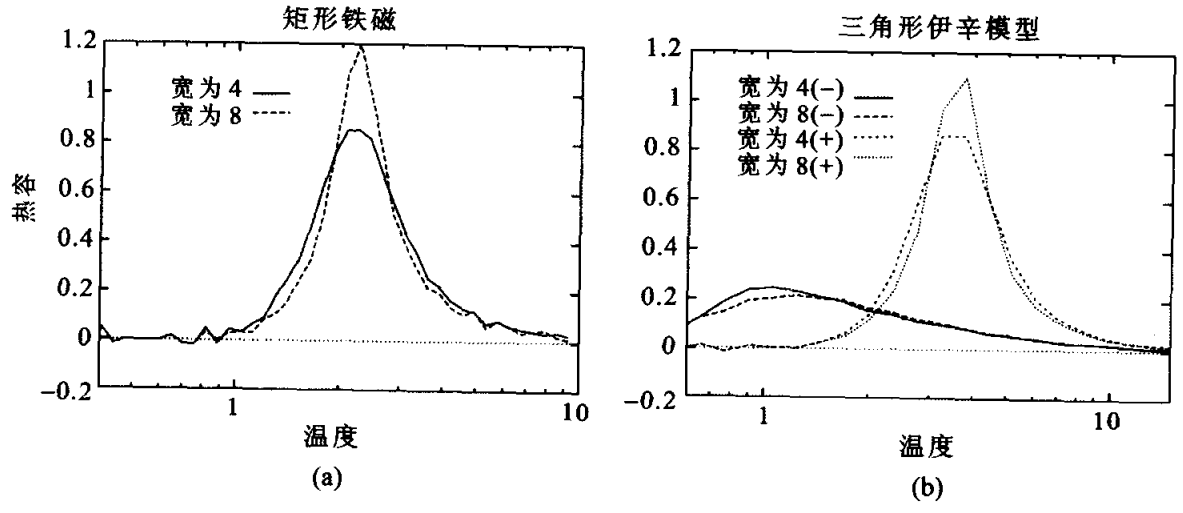


图 31.18 热容：(a) 矩形模型；(b) 具有不同宽度的三角模型，(+)和(−)分别表示铁磁体和反铁磁体。与图 31.11 进行比较。

图 31.15 中自由能的那幅图对于我们如何预见相变温度提供了一些更深入的理解。可以看出铁磁性系统的两种状态，每种状态都具有多么简单的自由能：对于高温状态，它是一条穿过 $F=0$、$T=0$ 的斜直线；而对于低温状态，它是一条水平直线。（每条直线的斜率显示了该状态各自旋的熵的大小）。大致而言，相变发生在这些直

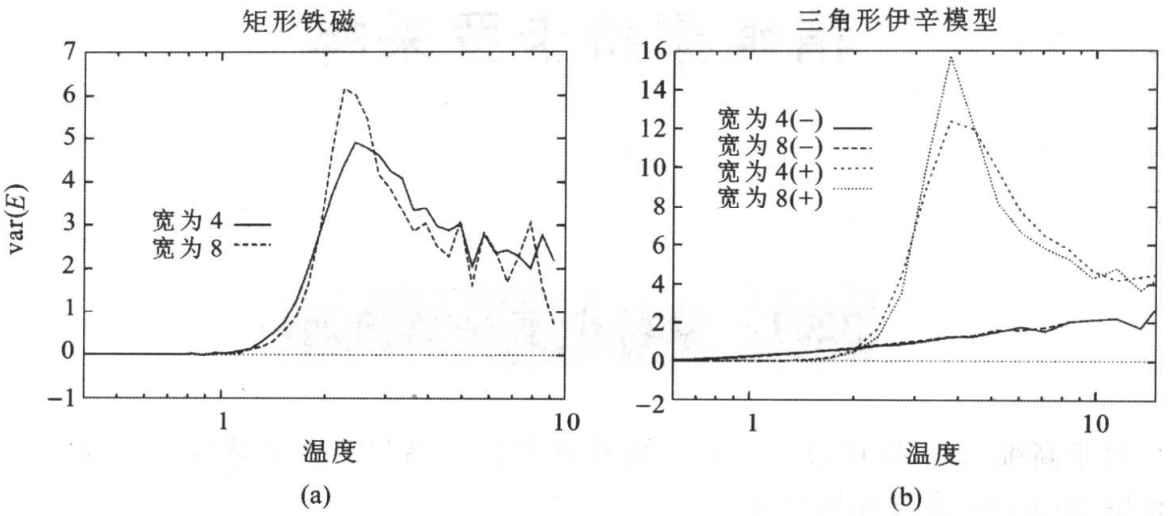


图 31.19 各自旋的能量方差:(a) 矩形模型;(b) 具有不同宽度的三角模型,(+)和(−)分别表示铁磁体和反铁磁体。与图 31.11 进行比较。

线的交点处。所以可以预测相变温度与基态能量是线性相关的。

与蒙特卡罗结果的比较

两个实验的结果非常一致。两个仿真系统(细长条和周期方阵)并不相同。找出宽度为 W 的带状矩阵的所有特征值,并计算 $\sum \lambda^W$ 以得到 $W \times W$ 块矩阵的划分函数,从而可以进行更详细的比较。

31.3 习 题

▷ **习题 31.2**[↗] 从蒙特卡罗仿真中求取熵的最佳途径是什么?从划分函数计算中得到熵和热容的最佳途径是什么?



习题 31.3[↗] 可将一个伊辛模型进行推广:在任一自旋 m 和 n 之间都有耦合 J_{mn},并且 J_{mn} 值对于每个 m 和 n 都可以不同。在所有耦合都为正的特例中,我们知道该系统有两个基态,全上状态和全下状态。对于一个更普遍的 J_{mn} 设置而言,可以想象会存在许多基态。

设想需要生成一个自旋系统,其局部最小值是一个已知的状态序列 $\boldsymbol{x}_{(1)}$, $\boldsymbol{x}_{(2)}, \cdots, \boldsymbol{x}_{(S)}$。你能否想出设置 $\boldsymbol{J}$ 的一种途径,以便所选状态都是低能状态?允许你将所有 $\{J_{mn}\}$ 调整到你所想要的任意值。

第 32 章

精确蒙特卡罗采样

32.1 蒙特卡罗法的问题

对于高维问题,应用最广泛的随机采样方法是马尔科夫链蒙特卡罗法,如 Metropolis 法、Gibbs 采样和切分采样。

所有这些方法都存在的问题是:没错,“只要链收敛于平衡分布”,就可以保证一个已知算法从目标密度 $P(\boldsymbol{x})$ 中渐进地产生样本”。但是,如果马尔科夫链的运行时间 T 过短,那么样本将来自另一分布 $P^{(T)}(\boldsymbol{x})$。马尔科夫链要运行多久才会“收敛”呢? 如第 29 章所述,这个问题一般很难解答。然而,Propp 和 Wilson(1996 年)的开创性工作使人们能够对于某种马尔科夫链解决这个难题;此外,Propp 和 Wilson 还说明了如何从目标密度函数得到“精确的”样本。

32.2 精确采样的概念

Propp 和 Wilson 的*精确采样法*(也称作“完美仿真”或“来自过去的耦合”)取决于三个概念。

耦合马尔科夫链的汇合

首先,如果初始条件各不相同的某些马尔科夫链共享一个随机数生成器,则它们在状态空间的轨迹可能汇合;并且,一旦发生汇合,将不会再分离。如果所有初始条件所导致的轨迹都汇合到一个单一轨迹中,则可确定马尔科夫链已经“忘掉”了它的初始条件。图 32.1(a-i)显示了 21 个与第 29.4 节所描述的马尔科夫链相同的链,它采用 Metropolis 算法(如图 29.12 所示)从 $\{0,1,\cdots,20\}$ 中采样;每条链的初始条件各不相同,但它们都由同一随机数生成器驱动;80 步之后这些链汇合到一起。图 32.1(a-ii)显示了同一马尔科夫链,但其随机数种子各不相同;在这种情况下,直到 400 步之后才会发生汇合(图中并未显示)。图 32.1(b)显示了类似的马尔科夫链,每个都有和第 29.4 节以及图 32.1(a)相同的提议密度;但在图 32.1(b)中,所有链在任一时间步都可通过同样的途径得到每一步的提议移动,“向左”或者“向右”,与当前状态无关。链的这种耦合改变了汇合的统计特性。因为两条相邻路径只在发生

舍弃时才会汇合，并且舍弃只发生在边界处（对于这种特定的马尔科夫链），只有当链都在最左边状态或最右边状态时才会发生汇合。

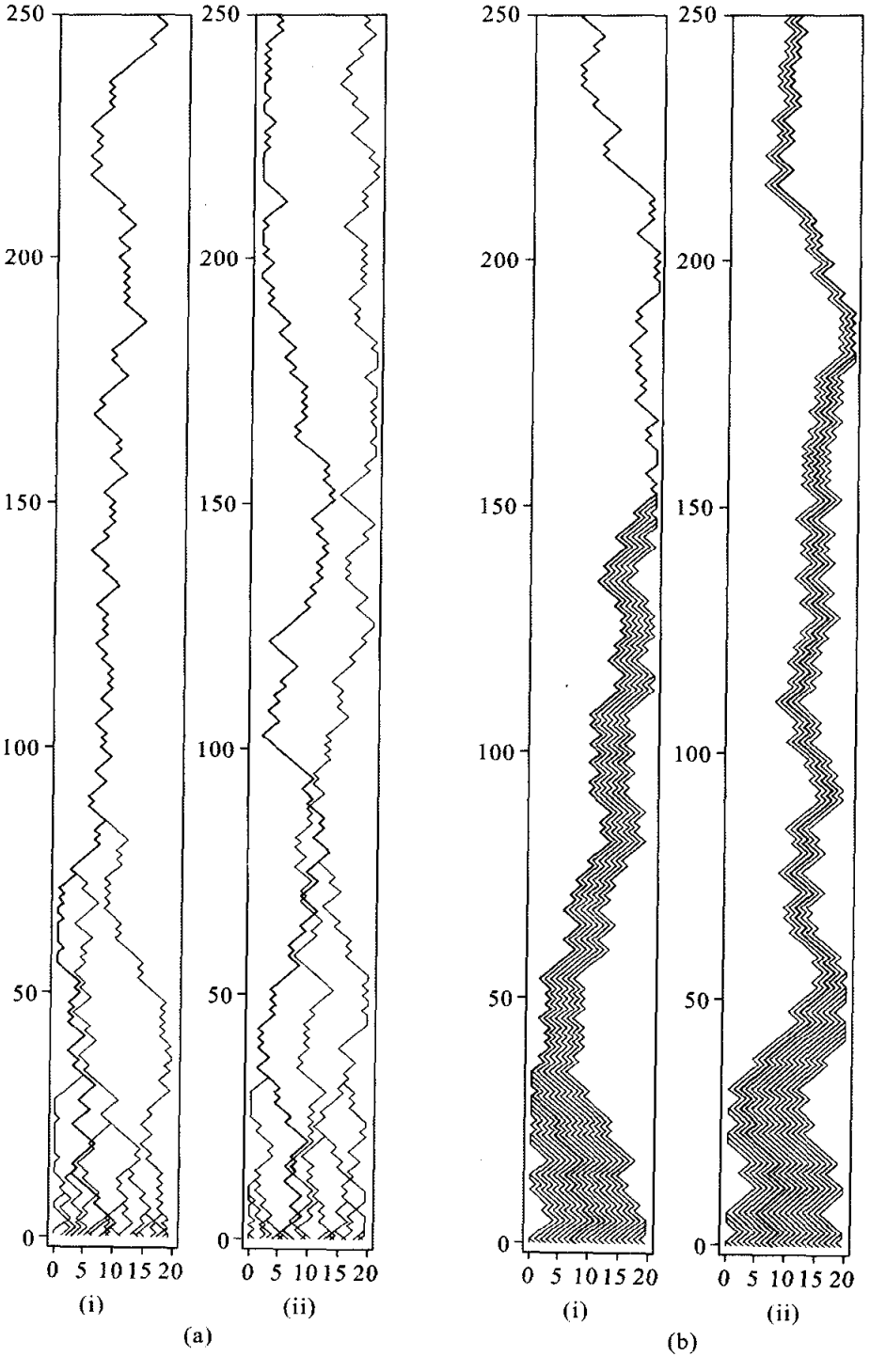


图 32.1 汇合，精确采样法中的第一种思想。时间自底向上增加。在最左边的图中，在 100 步之内发生汇合。根据每个状态使用向其所提供的随机数的方式，可得到不同的汇合特性。（a）Metropolis 仿真器的两次运行，其中确定提议步骤的随机比特取决于当前状态；在每种情况下，使用一个不同的随机数种子。（b）在这个仿真器中，所有状态的随机提议（“向左”或“向右”）都是相同的。在每幅图中，其中起始于位置 $x = 8$ 的那条路径都是突出显示的。

来自过去的耦合

我们如何利用汇合特性从马尔科夫链平衡分布中找到精确的样本呢？当完全汇合发生时，系统状态并不是来自平衡分布的一个正确样本；比如在图32.1(b)中，最终的汇合总是发生在状态碰到两个边界值之一时，因为轨迹只在边界处汇合。所以不断地向前采样直至汇合发生并不是一种正确的方法。

关于精确采样的第二个重要思想是：可以通过从过去的一个时间点 T_0 开始采样直到现在，来获得精确的样本。如果汇合已经发生，则当前样本是来自平衡分布的一个无偏倚样本；如果没有发生，可以重新使用相同的随机数，从过去的一个时间点 T_0 重新开始仿真。在一系列更早的时间点 T_0 处重复进行仿真，从一次运行到下一次将 T_0 加倍是一个方便的选择。当汇合发生在“当前”时间点之前时，可以将 $x(0)$ 记录为来自马尔科夫链平衡分布的一个精确采样。

图32.2显示了用这种方式产生的两个精确样本。在图32.2(a)最左边的图中，从 $T_0 = -50$ 处开始对所有可能初始条件，向前运行21条马尔科夫链。没有发生汇合。从 $T_0 = -100$ 处的所有可能初始条件开始，重新启动仿真，并以适当的方式重新设置随机数生成器，使得在每个时间点 t（特别地，从 $t = -50$ 至 $t = 0$）所生成的随机数与它在前一次运行中的值相等。注意，从 $T_0 = -100$ 所开始的运行，在 $t = -50$ 到 $t = 0$ 之间所产生的轨迹，和第一次从 $T_0 = -50$ 开始仿真的轨迹的子集相同。汇合还是没有发生，所以再次把 T_0 加倍到 $T_0 = -200$。这次所有轨迹汇合，而且得到了如箭头所示的一个精确样本。如果选取一个更早的时间点，比如 $T_0 = -500$，所有轨迹在 $t = 0$ 处必汇合到同一点，因为所有轨迹必将通过 $t = -200$ 的某个状态，而所有那些状态又都导致同一终点。所以如果从一个无限早的时间点开始运行马尔科夫链，从任一初始条件开始，它都将终止于同一状态。图32.2(b)显示了一个精确样本，其产生方式与图32.1(b)中的马尔科夫链相同。

这种叫做来自过去的耦合的方法非常重要，因为它使人们可以从平衡分布中得到精确样本；但是，如这里所言，它的实际应用价值不大，因为我们不得不从所有初始状态开始链的仿真。上面的示例中只有21个状态，但任何真实采样问题中都会有极多的状态；例如，想想包含1 000个二进制自旋的系统，就有 $2^{1\,000}$ 个状态。引入蒙特卡罗法的本意就是想避免访问这样一个系统的所有状态！

单调性

从过去时间点开始，对所有可能的状态进行仿真，就可以得到正确的采样。在确定了这一点之后，Propp 和 Wilson 的第3个诀窍（它使精确采样法成为在实际中有用的方法）是：对于某些马尔科夫链，无需仿真所有的轨迹就有可能检测出其汇合。比如，图32.1(b)中的链就满足如下性质：两条轨迹永远不会相交。所以如果简单地跟踪从最左和最右状态开始的两条轨迹，可知如果这两条轨迹汇合，那么所有轨迹都会

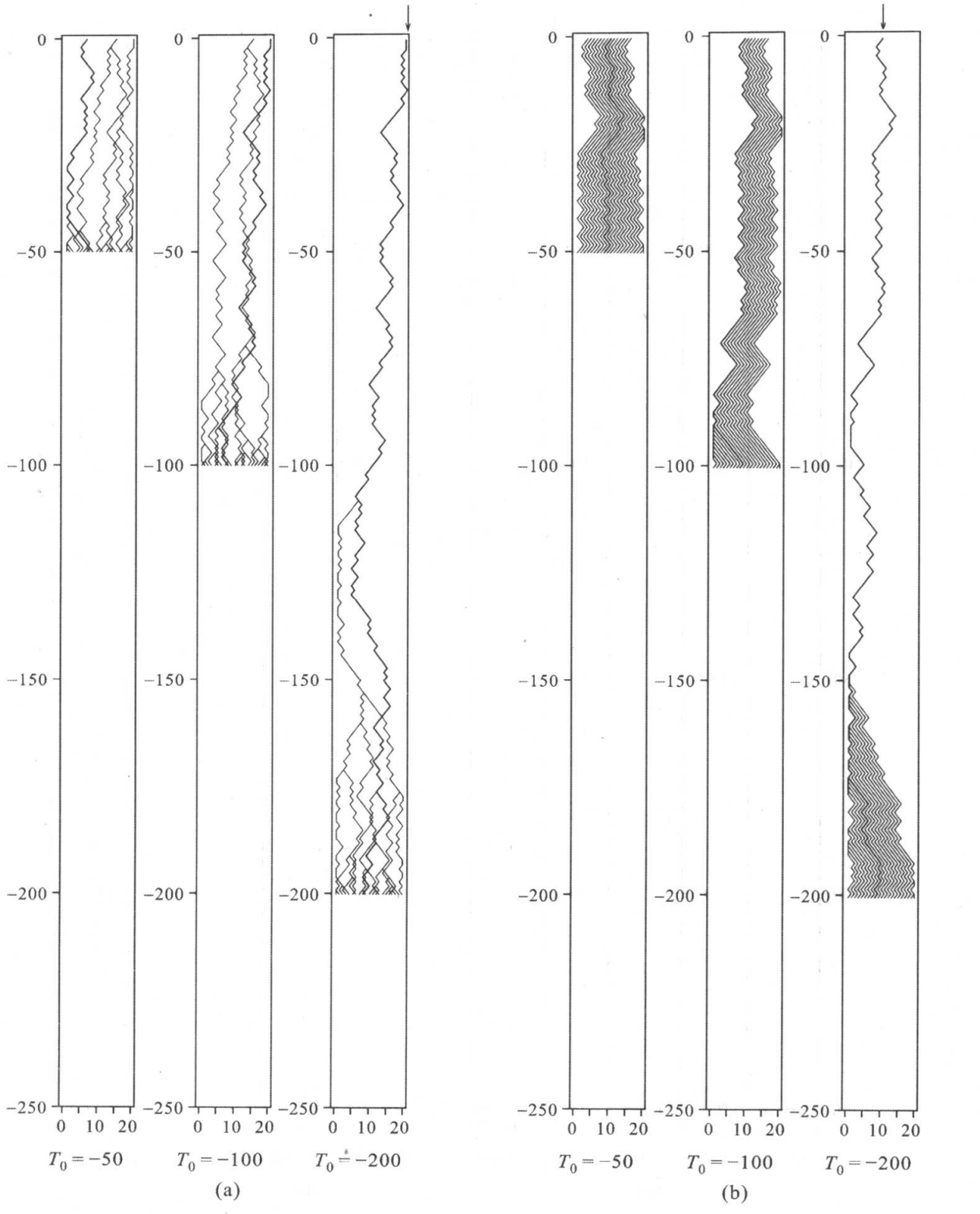


图 32.2 “来自过去的耦合”,精确采样法中的第二个思想。

汇合。图 32.3(a)只显示图 32.2(b)中最左和最右的轨迹,并以此阐明这种观点。图 32.3(b,c)显示了只从两个终端状态开始运行“来自过去的耦合”法所生成的另两个来自同一平衡分布的精确样本。在(b)中,从 $T_0 = -50$ 开始的两次运行汇合;在(c)中,必须将时间推移到 $T_0 = -1\,000$ 之外才能实现汇合。

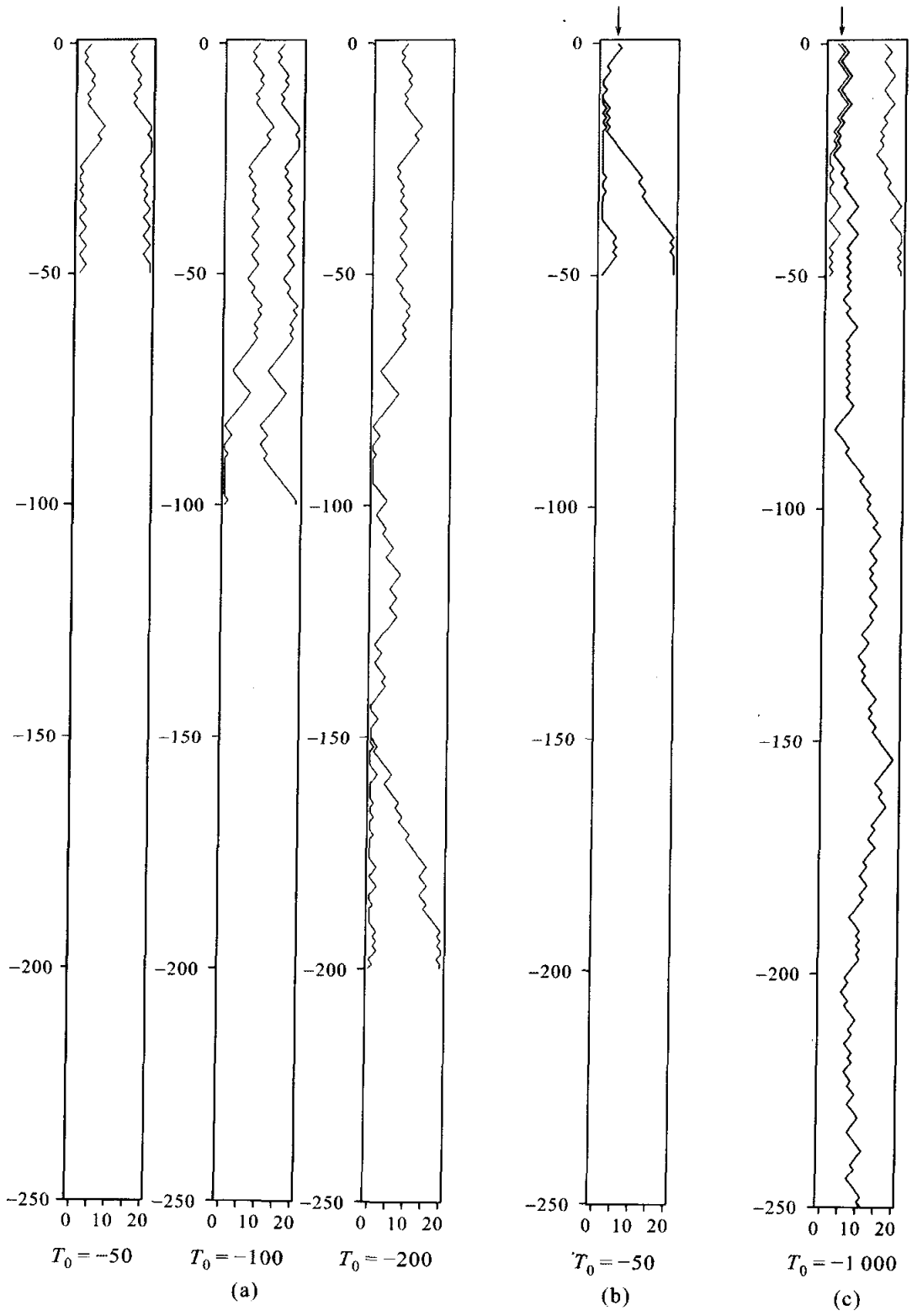


图 32.3 （a）状态的排序，精确采样法中的第 3 个概念。这里所示轨迹是图 32.2（b）中最左边和最右边的轨迹。为了建立在时间零点处的状态，只需从 $T_0 = -50$、$T_0 = -100$ 和 $T_0 = -200$ 处开始运行仿真，在 $T_0 = -200$ 之后发生点的汇合。

（b，c）来自目标密度的两个精确样本，也是用这种方法产生的，但随机数种子各不相同。所需的初始时间分别为 $T_0 = -50$ 和 $T_0 = -1\,000$。

32.3 对有趣分布进行精确采样

在所研究的简化问题中，可以将状态放入一维次序中，使得任何两条轨迹都不相交。许多有趣状态空间中的状态也可以放入偏序中，而且可以找到遵从该偏序的耦合马尔科夫链。[比如，双自旋的4种可能状态上的一个偏序如下：(+,+)>(+,−)>(−,−)；(+,+)>(−,+)>(−,−)；而且状态(+,−)和(−,+)是无序的。]对于这样的系统，只要验证对于初始状态为状态空间中“最大”和“最小”状态的所有历史轨迹都发生汇合，就可以证明已发生汇合。

作为一个实例，考虑将Gibbs采样法应用于铁磁伊辛自旋系统，其中状态的偏序是这样定义的：如果对于所有自旋 i，都有 $x_i \geq y_i$，则状态 $\boldsymbol{x}$“大于或等于”状态 $\boldsymbol{y}$。最大和最小状态是全上和全下的状态。如算法32.4所示，马尔科夫链耦合在一起。Propp和Wilson(1996年)证明这种系统可以生成精确样本，尽管如果伊辛模型低于其临界温度，则要很长时间才能找到精确样本，因为在这样的条件下，Gibbs采样法本身就是缓慢混合的。Propp和Wilson通过以下方式对伊辛模型改进了这种方法：使用一种叫做单联结热浴算法的马尔科夫链，从一种称为随机聚类模型的相关模型中进行采样；他们证明可以从随机聚类模型中快速得到精确样本，并可将它转换为来自伊辛模型的精确样本。他们开创性的论文给出了来自1 600万个自旋的伊辛模型在其临界温度处的精确样本。一个来自较小伊辛模型的样本如图32.5所示。

计算 $a_i := \sum_j J_{ij} x_j$
从 Uniform(0,1) 中抽取 u
If $u < 1/(1 + e^{-2a_i})$
  $x_i := +1$
Else
  $x_i := -1$

算法32.4 Gibbs采样耦合法。马尔科夫链通过将所有链在每个时间步骤都更新为相同的自旋 i，且所有链共用相同的随机数序列 u，而耦合在一起。

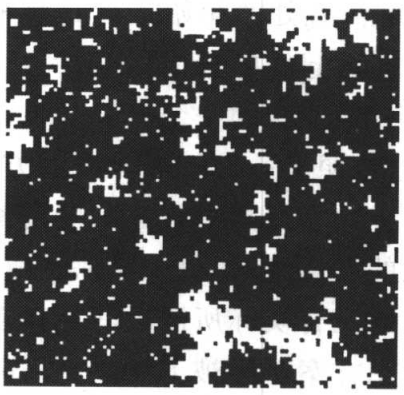


图32.5 由D. B. Wilson产生的来自伊辛模型临界温度点处的一个精确采样。在一台普通计算机上，几秒之内就可以用精确采样法产生这种样本。

将精确采样法推广到“无吸引力的”分布

上面简要描述了 Propp 和 Wilson 对伊辛模型所使用的方法，它只适用于他们称为“有吸引力的”概率分布。不定义这个术语，而是从实用的角度说明它的含义：这种方法可用于所有耦合都为正的自旋系统（如铁磁体），也可用于具有负耦合的一些特殊自旋系统（例如，如第 31 章所述，矩形铁磁体和反铁磁体是等价的）；但它不能用于其中某些耦合为负的一般自旋系统，因为在这种系统中，无法保证全上和全下状态所遵循的轨迹是所有轨迹集合的上界和下界。然而幸运的是，我们不必这么严格。可以用适当的方式重新表示 Propp 和 Wilson 算法，使之能推广到具有负耦合的自旋系统。概括状态形式的精确采样法的基本思想还是跟踪所有轨迹集合的边界，并且探测这些边界何时相等，以便找出精确采样。但是这些边界本身并不是实际的轨迹，而且它们也不一定是紧的边界。

我们不仿真两条在状态空间 $\{-1,+1\}^N$ 中移动的轨迹，而是在一个增广的状态空间 $\{-1,+1,?\}^N$ 中仿真一个轨迹包络，其中符号“?”表示“-1 或者 +1”。称这种增广系统的状态为“概括状态”。一个 6 - 自旋系统的示例概括状态为 + + - ? +?。这个概括状态是对如下状态集合的简记。

$$++-+++,++-++-,++--++,++--+-$$

马尔科夫链每一步的更新法则取一个自旋，列举相邻自旋的与当前概括状态相适应的所有可能状态，并对每一种局部情况，运用 Gibbs 采样（如算法 32.4 那样耦合到一个随机数 u）计算自旋的新值（+或 -）。如果所有这些新值都相符，则将概括状态中被更新自旋的新值设置为一致的值（+或 -）。否则，概括状态中的自旋新值为“?”。在时间点 T_0 处的初始条件是将概括状态中的所有自旋设置为“?”，它对应于考虑所有可能的起始配置。

对于具有正耦合的一个自旋系统，这个概括状态仿真将和最高状态及最低状态的仿真一样，按照 Propp 和 Wilson 的方式，在所有“?”符号都消失时发生汇合。概括状态法可用于具有任意耦合的一般自旋系统。这种方法唯一的缺陷是包络可能描述了一个过大的状态集，所以无法保证概括状态算法收敛；探测到汇合所需的时间可能比马尔科夫链汇合实际所需时间大得多。

概括状态方案已经被 Harvey 和 Neal（2000 年）应用于置信网络的精确采样，并被 Childs 等（2001 年）用于三角反铁磁体伊辛模型。概括状态法首先由 Huber（1998 年）引入，也称为夹心法和铰合链法。

进一步的阅读

为了进一步的阅读，获取来自其他分布的精确样本的令人印象深刻的图片，及对

精确采样法进行推广，可浏览完美随机采样的网址。①

要在你的 Web 浏览器中观看漂亮的精确采样的动画演示，请参见 Jim Propp 的网址。② 如图 32.6 所示。

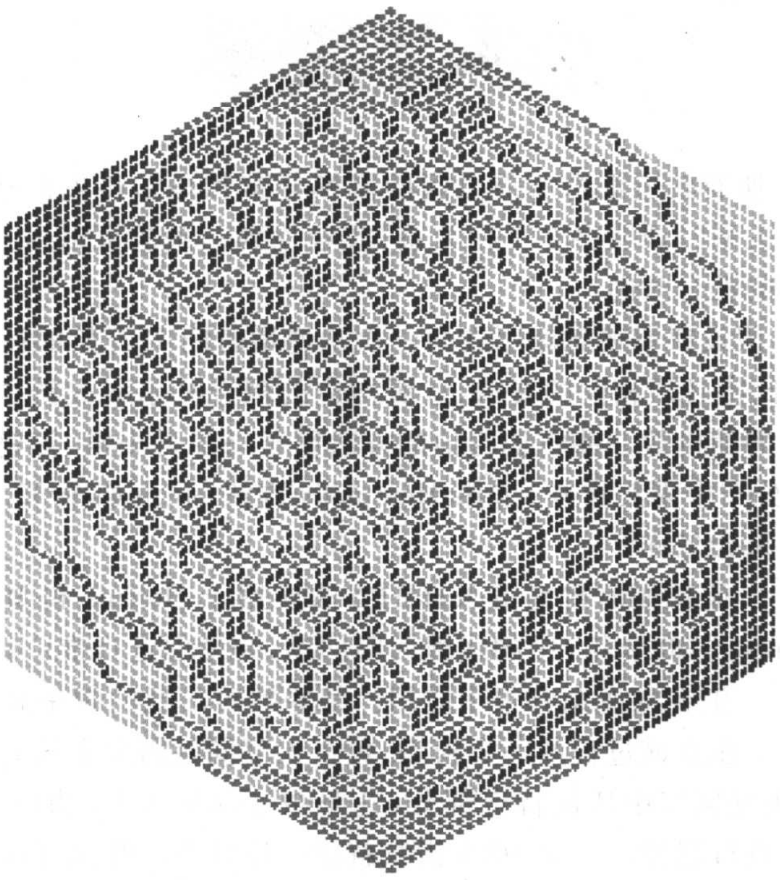


图 32.6 用菱形完成的一个六边形完美随机拼贴，由 J. G. Propp 和 D. B. Wilson 提供。

耦合的其他应用

通过共用一个随机数生成器把马尔科夫链耦合到一起的思想除了应用于精确采样中，还有其他的应用。应用估计器

$$\hat{\Phi}_P \equiv \frac{1}{T} \sum_t \phi(\boldsymbol{x}^{(t)}) \tag{32.1}$$

Pinto 和 Neal(2001 年)证明从马尔科夫链蒙特卡罗仿真(第 29.1 节所讨论的第 2 个问题)所得的估计的精确性可以通过对所感兴趣的链进行耦合而得到改善；这里，感兴趣的链收敛到 P，而与之耦合的是另一条链，它从第 2 个简单些的分布 Q 生成样本。必须以适当的方式设定耦合，使得两条链的状态是强相关的。基本思想是：首先用通常的方法(32.1)，估计所感兴趣的函数 ϕ 在 P 和 Q 下的期望，并把在 Q 下的估

① http://www.dbwilson.com/exact/

② http://www.math.wisc.edu/~propp/tiling/www/applets/

计 $\hat{\Phi}_Q$ 和 Q 下的期望真值 Φ_Q 进行比较，假设后者可被精确计算出来。如果 $\hat{\Phi}_Q$ 是超估计，那么 $\hat{\Phi}_P$ 也可能是超估计。于是，差值 $(\hat{\Phi}_Q - \Phi_Q)$ 就可用于修正 $\hat{\Phi}_P$。

32.4 习 题

▷ **习题 32.1**[2] 所有轨迹汇合所需时间的概率分布和马尔科夫链达到平衡所需时间之间是否存在关系？证明存在一种关系，或者找到一条链可以通过两种不同方式实现但具有不同的汇合时间。

▷ **习题 32.2**[2] 假设弗雷德忽略了在每一次从递增时间 T_0 开始的运行中，用于某一时间点 t 的随机比特必须相同的要求，且所构造的耦合马尔科夫链仿真器在每次 T_0 改变时都使用新的随机数。弗雷德将他的方法用于马尔科夫链，该链打算从状态 0、1 和 2 上的均匀分布中采集样本，采用 Metropolis 方法，并由如图·32.1(b)所示随机比特源驱动。结果会怎样？

习题 32.3[3] 研究如何将完美采样的方法应用于 Holmes 和 Mallick(1998 年)或 Holmes 和 Denison(2002 年)中的线性回归问题，并考虑进行推广。

习题 32.4[2] 汇合的概念有很多应用。某些姓氏出现的频率高于其他的，某些已经完全消失了。建立该过程的一个模型；每个人都同姓需要多长时间达到？

类似地，人类基因组中任何特定部分(它形成了法院 DNA 指纹识别基础)的可变性和姓氏一样进行遗传。一个 DNA 指纹就像一串姓氏。姓氏可以汇合，使得某些姓氏偶然地比其他姓氏更为普遍，这一事实是否应该影响 DNA 指纹证据的使用方式？

▷ **习题 32.5**[2] 你如何用一枚硬币来产生 3 人的一个随机名次？通过精确采样构造一种解法。例如，你可将精确采样应用于马尔科夫链，在该链中重复使用硬币交替地决定是否在第一和第二之间切换，然后是否在第二和第三之间切换。

习题 32.6[3] 找到概率分布的划分类函数 Z 是一个难题。许多马尔科夫链蒙特卡罗法在无需找到 Z 的情况下，就可从分布中产生有效的样本。

是否存在这样的概率分布和马尔科夫链，使得产生完美样本所需的时间或者产生完美样本所需的随机比特数与 Z 值有关？是否存在某些情形，达到汇合的时间传达了关于 Z 的信息？

32.5 习题解答

习题 32.1 的解答 也许令人惊讶，均衡时间和达到汇合的时间之间并不存在直接联系。可以通过整数 $\mathcal{A} = \{0,1,2,\cdots,20\}$ 上的均匀分布例子来证明这一点。只经

过一次迭代就收敛于此分布的马尔科夫链满足:对于所有已知的 s_t,状态 s_{t+1} 的概率是均匀分布的。这样一条链可以用两种方式耦合到一个随机数生成器中:(a) 可以抽取一个随机整数 $u \in \mathscr{A}$,不论 s_t 如何,都设置 s_{t+1} 等于 u;或者(b) 可以抽取一个随机整数 $u \in \mathscr{A}$,并设置 s_{t+1} 等于 $(s_t + u) \bmod 21$。方法(b)将产生一群锁定在一起的轨迹,除了不发生汇合外,它们与图 32.1 所示的轨迹相同。因此,尽管方法(a)和(b)的平衡时间都是 1,汇合时间却分别为 1 和无穷。

另一方面,汇合次数似乎给出了平衡时间的某种上界。

第 33 章

变 参 法

变参法是用于复杂概率分布近似的一种重要方法，可应用于统计物理学、数据建模及神经网络。

33.1 变参自由能最小化

在物理系统中，近似复杂分布的一种方法是*平均场理论*。平均场理论是将要研究的通用*变参自由能*方法的一种特例，后者由 Feynman 和 Bogoliubov 提出。理解这种方法所需的数学上的重点在于 Gibbs 不等式，在此重复一下这个不等式。①

在同一符号集 $\mathcal{A}_X$ 上定义的两个概率分布 $Q(x)$ 和 $P(x)$ 的**相对熵**为

$$D_{\mathrm{KL}}(Q \parallel P) = \sum_x Q(x) \log \frac{Q(x)}{P(x)} \tag{33.1}$$

相对熵满足 $D_{\mathrm{KL}}(Q \parallel P) \geqslant 0$（Gibbs 不等式），只有当 $Q = P$ 时，等式才成立。一般来说，$D_{\mathrm{KL}}(Q \parallel P) \neq D_{\mathrm{KL}}(P \parallel Q)$。

本章中，将把对数 log 替换成自然对数 ln，并以奈特为单位来度量发散性。

统计物理学中的概率分布

在统计物理学中，常常会遇到具有以下形式的概率分布。

$$P(\boldsymbol{x} \mid \beta, \boldsymbol{J}) = \frac{1}{Z(\beta, \boldsymbol{J})} \exp[-\beta E(\boldsymbol{x}; \boldsymbol{J})] \tag{33.2}$$

例如，其中状态向量为 $\boldsymbol{x} \in \{-1, +1\}^N$，且能量函数 $E(\boldsymbol{x}; \boldsymbol{J})$ 可以是

$$E(\boldsymbol{x}; \boldsymbol{J}) = -\frac{1}{2} \sum_{m,n} J_{mn} x_m x_n - \sum_n h_n x_n \tag{33.3}$$

划分函数（归一化常数）为

$$Z(\beta, \boldsymbol{J}) \equiv \sum_x \exp[-\beta E(\boldsymbol{x}; \boldsymbol{J})] \tag{33.4}$$

式(33.2)中的概率分布较复杂，但并非无法忍受的复杂，毕竟对于任意特定的 $\boldsymbol{x}$，可以在自旋数目的多项式时间内计算出 $E(\boldsymbol{x}; \boldsymbol{J})$。但是很难计算归一化常数 $Z(\beta, \boldsymbol{J})$，如第 29 章中看到的一样，而且描述该概率分布的性质也很难。例如，已知在一些任意点 $\boldsymbol{x}$ 上的 $E(\boldsymbol{x}; \boldsymbol{J})$ 值，并不能给出关于这个系统一般性质的任何有用信息。

---

① Gibbs 不等式首次出现在式(1.24)；也参见习题 2.26。

计算 $Z(\beta,\boldsymbol{J})$的值会特别受用，因为从 Z 可以推导出系统的所有热力学性质。

变参自由能最小化是一种通过可调参数 $\boldsymbol{\theta}$ 参数化的较简单总体 $Q(\boldsymbol{x};\boldsymbol{\theta})$来近似复杂分布$P(\boldsymbol{x})$的一种方法。我们调整这些参数，以使 Q 在某种意义上最好地近似 P。这种近似的一个副产品是 $Z(\beta,\boldsymbol{J})$的一个下界。

变参自由能

选择来衡量近似品质的目标函数是变参自由能

$$\beta\tilde{F}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{\theta}) \ln \frac{Q(\boldsymbol{x};\boldsymbol{\theta})}{\exp[-\beta E(\boldsymbol{x};\boldsymbol{J})]} \tag{33.5}$$

该表达式可变换为两种有趣的形式：首先

$$\beta\tilde{F}(\boldsymbol{\theta}) = \beta \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{\theta}) E(\boldsymbol{x};\boldsymbol{J}) - \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{\theta}) \ln \frac{1}{Q(\boldsymbol{x};\boldsymbol{\theta})} \tag{33.6}$$

$$\equiv \beta \langle E(\boldsymbol{x};\boldsymbol{J}) \rangle_Q - S_Q \tag{33.7}$$

其中$\langle E(\boldsymbol{x};\boldsymbol{J}) \rangle_Q$是分布 $Q(\boldsymbol{x};\boldsymbol{\theta})$下的能量函数平均值，而 S_Q是分布 $Q(\boldsymbol{x};\boldsymbol{\theta})$ 的熵（在 S 的定义中，假设 k_B为 1，以便与第一部分中熵 H 的定义一致）。

其次，可以用 $P(\boldsymbol{x}|\beta,\boldsymbol{J})$的定义写出

$$\beta\tilde{F}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{\theta}) \ln \frac{Q(\boldsymbol{x};\boldsymbol{\theta})}{P(\boldsymbol{x}|\beta,\boldsymbol{J})} - \ln Z(\beta,\boldsymbol{J}) \tag{33.8}$$

$$= D_{\mathrm{KL}}(Q \parallel P) + \beta F \tag{33.9}$$

其中 F 是真实的自由能，其定义为

$$\beta F \equiv -\ln Z(\beta,\boldsymbol{J}) \tag{33.10}$$

而 $D_{\mathrm{KL}}(Q \parallel P)$是近似分布 $Q(\boldsymbol{x};\boldsymbol{\theta})$和真实分布 $P(\boldsymbol{x}|\beta,\boldsymbol{J})$之间的相对熵。因此，由 Gibbs 不等式，变参自由能 $\tilde{F}(\boldsymbol{\theta})$以 F 为下界，且只有在 $Q(\boldsymbol{x};\boldsymbol{\theta}) = P(\boldsymbol{x}|\beta,\boldsymbol{J})$时才能取得这个值。

因此，我们的策略是以适当的方式改变 $\boldsymbol{\theta}$，使得 $\beta\tilde{F}(\boldsymbol{\theta})$最小化。于是，这个近似的分布给出了真实分布的一个简化近似，它可能是有用的，而且 $\beta\tilde{F}(\boldsymbol{\theta})$是 βF 的上界。同样地，$\tilde{Z} \equiv \mathrm{e}^{-\beta\tilde{F}(\boldsymbol{\theta})}$是 Z 的下界。

可以计算出目标函数 $\beta\tilde{F}$ 吗？

我们已经达成共识，难以在 $\boldsymbol{x}$ 上求几种有趣的和值。例如，假定划分函数

$$Z = \sum_{\boldsymbol{x}} \exp(-\beta E(\boldsymbol{x};\boldsymbol{J})) \tag{33.11}$$

能量

$$\langle E \rangle_P = \frac{1}{Z} \sum_{\boldsymbol{x}} E(\boldsymbol{x};\boldsymbol{J}) \exp(-\beta E(\boldsymbol{x};\boldsymbol{J})) \tag{33.12}$$

和熵

$$S \equiv \sum_{\boldsymbol{x}} P(\boldsymbol{x}|\beta,\boldsymbol{J}) \ln \frac{1}{P(\boldsymbol{x}|\beta,\boldsymbol{J})} \tag{33.13}$$

都是不可计算的。那么,既然目标函数 $\beta\tilde{F}(\boldsymbol{\theta})$ 也是以所有 $\boldsymbol{x}$ 之和式(33.5)定义的,为什么要假设它是一个可以方便处理的量呢?这么说吧,对于一系列感兴趣的能量函数,及对于足够简单的近似分布,可以有效地计算出变参自由能。

33.2 自旋系统变参自由能的最小化

一个易于处理变参自由能的实例是能量函数由式(33.3)所定义的一个自旋系统,可以用下面这个可分离的近似分布对它进行近似。

$$Q(\boldsymbol{x};\boldsymbol{a}) = \frac{1}{Z_Q} \exp\Big(\sum_n a_n x_n \Big) \tag{33.14}$$

变参自由能(33.5)的变参 $\boldsymbol{\theta}$ 是向量 $\boldsymbol{a}$ 的元素。为了计算变参自由能,需要该分布的熵

$$S_Q = \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{a}) \ln \frac{1}{Q(\boldsymbol{x};\boldsymbol{a})} \tag{33.15}$$

和能量均值

$$\langle E(\boldsymbol{x};\boldsymbol{J}) \rangle_Q = \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{a}) E(\boldsymbol{x};\boldsymbol{J}) \tag{33.16}$$

该可分离的近似分布的熵就是单个自旋的熵之和(习题4.2)

$$S_Q = \sum_n H_2^{(e)}(q_n) \tag{33.17}$$

其中 q_n 是自旋 n 为 $+1$ 的概率

$$q_n = \frac{e^{a_n}}{e^{a_n} + e^{-a_n}} = \frac{1}{1 + \exp(-2a_n)} \tag{33.18}$$

且有

$$H_2^{(e)}(q) = q \ln \frac{1}{q} + (1-q) \ln \frac{1}{(1-q)} \tag{33.19}$$

在 Q 下的平均能量容易得到,因为 $\sum_{m,n} J_{mn} x_m x_n$ 是包括两个独立随机变量之积的各项之和。(由于没有自耦合,因此,当 $m = n$ 时,$J_{mn} = 0$。)如果定义 x_n 的均值为 $\bar{x}_n$,表示为

$$\bar{x}_n = \frac{e^{a_n} - e^{-a_n}}{e^{a_n} + e^{-a_n}} = \tanh(a_n) = 2q_n - 1 \tag{33.20}$$

可以得到

$$\langle E(\boldsymbol{x};\boldsymbol{J}) \rangle_Q = \sum_{\boldsymbol{x}} Q(\boldsymbol{x};\boldsymbol{a}) \Big[-\frac{1}{2} \sum_{m,n} J_{mn} x_m x_n - \sum_n h_n x_n \Big] \tag{33.21}$$

$$= -\frac{1}{2}\sum_{m,n} J_{mn}\bar{x}_m\bar{x}_n - \sum_n h_n\bar{x}_n \tag{33.22}$$

所以变参自由能表示为

$$\beta\tilde{F}(\boldsymbol{a}) = \beta\langle E(\boldsymbol{x};\boldsymbol{J})\rangle_Q - S_Q = \beta\left(-\frac{1}{2}\sum_{m,n} J_{mn}\bar{x}_m\bar{x}_n - \sum_n h_n\bar{x}_n\right) - \sum_n H_2^{(e)}(q_n) \tag{33.23}$$

现在考虑以变参 $\boldsymbol{a}$ 来最小化该函数。如果 $q = 1/(1+\mathrm{e}^{-2a})$，可得熵的导数为

$$\frac{\partial}{\partial q}H_2^{(e)}(q) = \ln\frac{1-q}{q} = -2a \tag{33.24}$$

因此，可得

$$\begin{aligned}\frac{\partial}{\partial a_m}\beta\tilde{F}(\boldsymbol{a}) &= \beta\left[-\sum_n J_{mn}\bar{x}_n - h_m\right]\left(2\frac{\partial q_m}{\partial a_m}\right) - \ln\left(\frac{1-q_m}{q_m}\right)\left(\frac{\partial q_m}{\partial a_m}\right)\\ &= 2\left(\frac{\partial q_m}{\partial a_m}\right)\left[-\beta\left(\sum_n J_{mn}\bar{x}_n + h_m\right) + a_m\right]\end{aligned} \tag{33.25}$$

当

$$a_m = \beta\left(\sum_n J_{mn}\bar{x}_n + h_m\right) \tag{33.26}$$

时，该导数等于零。因此，在任何满足式(33.26)的点处，$\tilde{F}(\boldsymbol{a})$都取得极值，且

$$\bar{x}_n = \tanh(a_n) \tag{33.27}$$

能量 $E(\boldsymbol{x}) = -x_1x_2$的双自旋系统的变参自由能如图 33.1 所示。

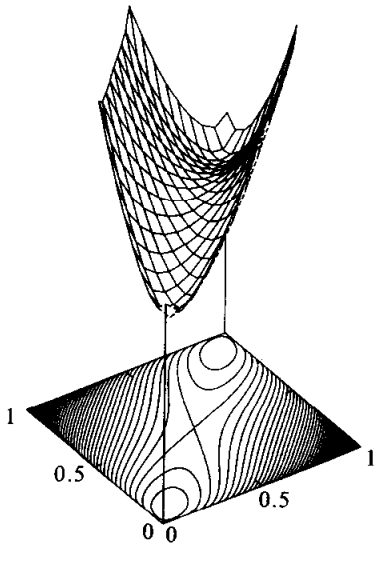


图 33.1 能量 $E(\boldsymbol{x}) = -x_1x_2$的双自旋系统的变参自由能，以两个变参 q_1 和 q_2 的函数表示。温度倒数为 $\beta = 1.44$。图中所绘函数为 $\beta\tilde{F} = -\beta\bar{x}_1\bar{x}_2 - H_2^{(e)}(q_1) - H_2^{(e)}(q_2)$，其中 $\bar{x}_n = 2q_n - 1$。注意：对于固定的 q_2，函数关于 q_1 是凸⌣的；而对于固定的 q_1，它关于 q_2 是凸⌣的。

变参自由能 $\tilde{F}(\boldsymbol{a})$可能是一个多峰函数，其中每一个驻点(最大点、最小点或鞍

点)满足式(33.26)和式(33.27)。对于具有任一耦合矩阵 $\boldsymbol{J}$ 的系统,应用这些公式的一种方式是,校正每一个参数 a_m 并用式(33.26)得到的相应 $\bar{x}_m$ 值,一次一个。这种参数异步校正肯定能够降低 $\beta\tilde{F}(\boldsymbol{a})$。

式(33.26)和式(33.27)可被看成是一个自旋系统的平均场公式。可变参数 a_n 可被认为是作用于孤立自旋 n 的一个假想场的强度。式(33.27)描述了自旋 n 的平均响应,而式(33.26)描述了如何根据对所有其他自旋的平均状态来设置场 a_m。

变参自由能的导数是一种有益的平均场理论观点,两个原因如下。

1. 该方法将一个目标函数 $\beta\tilde{F}$ 和平均场方程关联起来;这样的目标函数是有益的,因为它可以帮助确定最小化相同函数的其他动态系统。

2. 该理论易于推广到其他的近似分布。可以假设引入一个更复杂的近似 $Q(\boldsymbol{x};\boldsymbol{\theta})$,例如它可以捕获自旋间的相关性而不必将自旋建模为独立的。那么就可以计算变参自由能,并优化这个更复杂的近似的参数 $\boldsymbol{\theta}$。这种近似分布的自由度越高,其自由能的界限就变得越紧。然而,如果一个近似的复杂度增加,那么平均能量或熵的计算通常就变得更具挑战性了。

33.3 示例:铁磁体伊辛模型的平均场理论

在第31章讨论的简单伊辛模型中,当 m 和 n 为邻居时,每个耦合 J_{mn} 等于 J,否则为零。对于所有自旋,存在一个相同的应用场 $h_n = h$。一个非常简单的近似分布是具有单一可变参数 a 的分布,它定义了一个可分离的分布

$$Q(\boldsymbol{x};a) = \frac{1}{Z_Q}\exp\Big(\sum_n ax_n\Big) \tag{33.28}$$

其中所有自旋是独立的且具有相同的向上概率

$$q_n = \frac{1}{1+\exp(-2a)} \tag{33.29}$$

平均磁化为

$$\bar{x} = \tanh(a) \tag{33.30}$$

而定义了变参自由能最小值的式(33.26)变成

$$a = \beta(CJ\bar{x} + h) \tag{33.31}$$

其中 C 是一个自旋所涉及的耦合数;对于矩形二维伊辛模型,$C=4$。可以数值地求解式(33.30)和式(33.31),得出 $\bar{x}$——事实上,改变 $\bar{x}$ 并求出 β 是最容易的——并得到自由能最小值和最大值的图,以温度的函数表示,如图33.2所示。实线表示当 $C=4$、$J=1$ 时,$\bar{x}$ 和 $T=1/\beta$ 的关系曲线。

当 $h=0$ 时,在临界温度 T_C^{mft} 处存在一个分叉。[分叉是如图33.2中实线所示的一个转换:从以 a 的函数为最小值的系统(右边)到具有两个最小值和一个最大值的

系统(左边)的转换;最大值是3条线的中间那条。实线看起来就像一个叉子。]在该温度之上,变参自由能中只有一个最小值,位于 $a=0$ 和 $\bar{x}=0$ 处;该最小值对应于在所有状态上均匀的近似分布。在临界温度之下,对应于近似分布有两个打破平衡的最小值,此时所有自旋或者一起向上,或者一起向下。状态 $\bar{x}=0$ 仍然是变参自由能的驻点,但现在它是变参自由能的一个局部最大点。

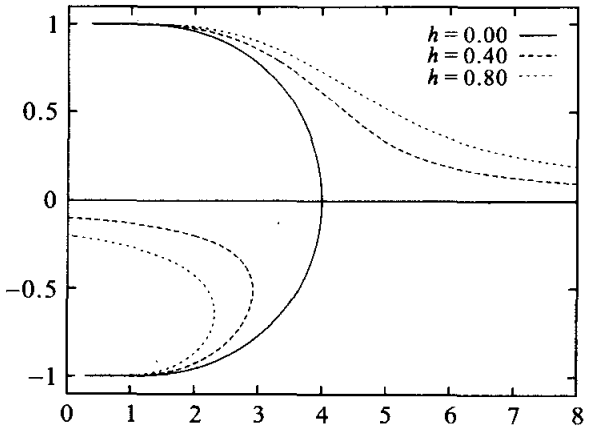


图 33.2 对于3个不同的应用场 h,对伊辛模型变参自由能极值问题的解。横轴:温度 $T=1/\beta$。纵轴:磁化 $\bar{x}$。根据平均场理论得到的临界温度为 $T_C^{\mathrm{mft}}=4$。

当 $h>0$ 时,在 $\bar{x}$ 值为正的任一温度处,都存在变参自由能的全局最小点,如图33.2中上面的虚线所示。只要 $h<JC$,如果该温度足够小的话,在自由能中还会有另一个局部最小点。这另外的最小点对应于应用场反方向中磁化的自保状态。这另一个最小点出现时的温度低于 T_C^{mft},并且当该值出现时,将伴随出现位于两个最小点之间的一个鞍点。这种类型的分叉称为鞍结点分叉。

每个自旋的变参自由能为

$$\beta\tilde{F}=\beta\left(-\frac{C}{2}J\bar{x}^2-h\bar{x}\right)-H_2^{(e)}\left(\frac{\bar{x}+1}{2}\right) \tag{33.32}$$



习题 33.1[2] 对于不同的温度值 T 和应用场 h,画出变参自由能的简图,以一个参数 $\bar{x}$ 的函数表示。

图33.2再现了真实伊辛系统的关键性质,即对于 $h=0$,存在一个临界温度,在该温度以下,系统存在大范围的次序,并且可以将它作为两个宏状态之一。然而,更进一步的研究表明,变参近似存在某些不足之处。首先,临界温度 T_C^{mft} 为4,差不多是真实临界温度 $T_C=2.27$ 的两倍。而且,变参模型在任意维数中具有等价的性质,包括 $d=1$,此时真实系统没有相变。所以不应将 T_C^{mft} 处的分叉描述为一个相变。

对于 $h=0$ 的情况,可以跟踪以 β 的函数所表示的全局最小值的轨迹,并找到近似分布的熵、热容和波动,通过第31章的矩阵方法将它们和实际 8×8 片断所得结果进行比较。如图33.3所示,最大的区别在于能量的波动。实际系统在临界温度附近

波动很大,而近似分布在其自旋之间不存在相关性,因而能量方差只是简单地依自旋数线性地增长。

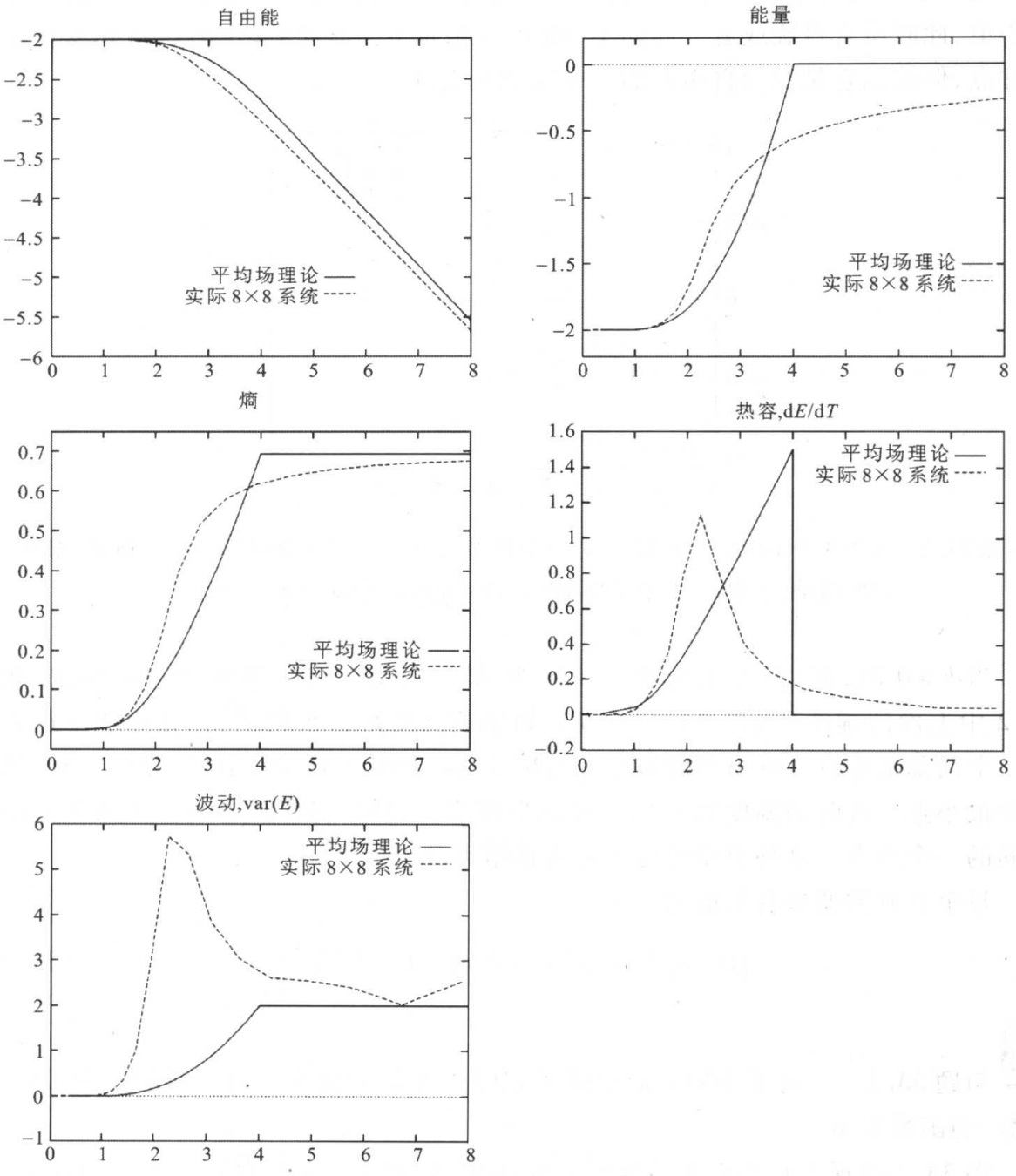


图 33.3 近似分布的性质与实际系统 8×8 片断的比较。注意:近似分布的变参自由能确实是真实系统自由能的一个上界。所有量都以"每自旋"的形式表示。

33.4 推理和数据建模中的变参法

在统计数据建模中,我们感兴趣的是对于已知数据 D 和模型假设 $\mathcal{H}$,参数向量

$\boldsymbol{w}$ 的后验概率分布 $P(\boldsymbol{w}|D,\mathcal{H})$。

$$P(\boldsymbol{w}|D,\mathcal{H}) = \frac{P(D|\boldsymbol{w},\mathcal{H})P(\boldsymbol{w}|\mathcal{H})}{P(D|\mathcal{H})} \tag{33.33}$$

在传统的模型拟合法中,优化单一参数向量 $\boldsymbol{w}$ 以找到该分布的峰值。我们真正感兴趣的是整个分布。如果希望进行模性比较的话,也可能会关注它的归一化常数 $P(D|\mathcal{H})$。概率分布 $P(\boldsymbol{w}|D,\mathcal{H})$ 通常是一个复杂的分布。在推理的变参法中,引入参数的一个近似概率分布 $Q(\boldsymbol{w};\boldsymbol{\theta})$,并优化该分布(通过改变它本身的参数 $\boldsymbol{\theta}$),以便它可以很好地近似参数的后验分布 $P(\boldsymbol{w}|D,\mathcal{H})$。

可以选取用于衡量近似品质的一个目标函数是变参自由能

$$\tilde{F}(\boldsymbol{\theta}) = \int \mathrm{d}^k \boldsymbol{w} Q(\boldsymbol{w};\boldsymbol{\theta}) \ln \frac{Q(\boldsymbol{w};\boldsymbol{\theta})}{P(D|\boldsymbol{w},\mathcal{H})P(\boldsymbol{w}|\mathcal{H})} \tag{33.34}$$

分母 $P(D|\boldsymbol{w},\mathcal{H})P(\boldsymbol{w}|\mathcal{H})$ 和后验概率 $P(\boldsymbol{w}|D,\mathcal{H}) = P(D|\boldsymbol{w},\mathcal{H})P(\boldsymbol{w}|\mathcal{H})/P(D|\mathcal{H})$ 仅相差一个乘积常数。所以,可以把变参自由能 $\tilde{F}(\boldsymbol{\theta})$ 看成 $-\ln P(D|\mathcal{H})$ 与 $Q(\boldsymbol{w};\boldsymbol{\theta})$ 和 $P(\boldsymbol{w}|D,\mathcal{H})$ 二者相对熵之和。$\tilde{F}(\boldsymbol{\theta})$ 以 $-\ln P(D|\mathcal{H})$ 为下界,仅当 $Q(\boldsymbol{w};\boldsymbol{\theta}) = P(\boldsymbol{w}|D,\mathcal{H})$ 时可以取得该值。对于某些模型和某些近似分布,可以计算出该自由能及其关于近似分布参数的导数。

采用变参自由能最小化所得的后验概率分布近似提供了一种有益的方法,对贝叶斯推理进行近似,从神经网络到纠错码的译码,它在许多领域都得到应用(Hinton 和 van Camp,1993 年;Hinton 和 Zemel,1994 年;Dayan 等,1995 年;Neal 和 Hinton,1998 年;MacKay,1995a)。这种方法有时也称作*总体学习*,以便和仅优化单个参数向量的传统学习过程进行对照。它的另外一个名字叫*变参贝叶斯法*。可以看一下总体学习在简单的高斯分布情况中是如何工作的。

33.5 未知高斯的情形:近似 μ 和 σ 的后验分布

将用一个近似总体 $Q(\mu,\sigma)$ 来拟合第 24 章研究过的后验分布。

$$P(\mu,\sigma|\{x_n\}_{n=1}^N) = \frac{P(\{x_n\}_{n=1}^N|\mu,\sigma)P(\mu,\sigma)}{P(\{x_n\}_{n=1}^N)} \tag{33.35}$$

$$= \frac{\frac{1}{(2\pi\sigma^2)^{N/2}}\exp\left(-\frac{N(\mu-\bar{x})^2+S}{2\sigma^2}\right)\frac{1}{\sigma_\mu}\frac{1}{\sigma}}{P(\{x_n\}_{n=1}^N)} \tag{33.36}$$

做单一假设:近似总体以 $Q(\mu,\sigma) = Q_\mu(\mu)Q_\sigma(\sigma)$ 形式分离。对 $Q_\mu(\mu)$ 和 $Q_\sigma(\sigma)$ 的函数形式未做任何限制。

写出变参自由能

$$\tilde{F}(Q) = \int \mathrm{d}\mu \mathrm{d}\sigma Q_\mu(\mu)Q_\sigma(\sigma) \ln \frac{Q_\mu(\mu)Q_\sigma(\sigma)}{P(D|\mu,\sigma)P(\mu,\sigma)} \tag{33.37}$$

对于固定的 $Q_\sigma(\sigma)$，单独地考虑以 $Q_\mu(\mu)$ 来优化 $\tilde{F}$，然后在固定的 $Q_\mu(\mu)$ 情况下，用 $Q_\sigma(\sigma)$ 来进行优化，就可以找到最优可分的分布 Q。

$Q_\mu(\mu)$ 的优化

作为 $Q_\mu(\mu)$ 的一个函数，$\tilde{F}$ 为

$$\tilde{F} = -\int \mathrm{d}\mu Q_\mu(\mu)\left[\int \mathrm{d}\sigma Q_\sigma(\sigma)\ln P(D|\mu,\sigma) + \ln[P(\mu)/Q_\mu(\mu)]\right] + \kappa \tag{33.38}$$

$$= \int \mathrm{d}\mu Q_\mu(\mu)\left[\int \mathrm{d}\sigma Q_\sigma(\sigma) N\beta \frac{1}{2}(\mu-\bar{x})^2 + \ln Q_\mu(\mu)\right] + \kappa' \tag{33.39}$$

其中 $\beta \equiv 1/\sigma^2$ 且 κ 表示不依赖于 $Q_\mu(\mu)$ 的常数。因此，对 Q_σ 的依赖性退化为一个简单的对以下均值的依赖性。

$$\bar{\beta} \equiv \int \mathrm{d}\sigma Q_\sigma(\sigma) 1/\sigma^2 \tag{33.40}$$

对于 $\beta = \bar{\beta}$ 的一个特定值，现在可以看出 $-N\bar{\beta}\frac{1}{2}(\mu-\bar{x})^2$ 是与后验分布相同的高斯函数的对数。由于通过设置 $Q=P$，可最小化相对熵 $\int Q\ln(Q/P)$，因此，可以马上写出对于固定的 Q_σ，最小化 $\tilde{F}$ 的分布 $Q_\mu^{\mathrm{opt}}(\mu)$ 为

$$Q_\mu^{\mathrm{opt}}(\mu) = P(\mu|D,\bar{\beta},\mathscr{H}) = \mathrm{Normal}(\mu;\bar{x},\sigma_{\mu|D}^2) \tag{33.41}$$

其中 $\sigma_{\mu|D}^2 = 1/(N\bar{\beta})$。

$Q_\sigma(\sigma)$ 的优化

用关于 β 的密度来表示 $Q_\sigma(\sigma)$，$Q_\sigma(\beta) \equiv Q_\sigma(\sigma)|\mathrm{d}\sigma/\mathrm{d}\beta| \propto 1/\beta$。[①] 作为 $Q_\sigma(\beta)$ 的函数，$\tilde{F}$ 为（忽略加性常量）

$$\tilde{F} = -\int \mathrm{d}\beta Q_\sigma(\beta)\left[\int \mathrm{d}\mu Q_\mu(\mu)\ln P(D|\mu,\sigma) + \ln[P(\beta)/Q_\sigma(\beta)]\right] \tag{33.42}$$

$$= \int \mathrm{d}\beta Q_\sigma(\beta)\left[(N\sigma_{\mu|D}^2 + S)\beta/2 - \left(\frac{N}{2}-1\right)\ln\beta + \ln Q_\sigma(\beta)\right] \tag{33.43}$$

其中对 μ 进行积分时，假设 $Q_\mu(\mu) = Q_\mu^{\mathrm{opt}}(\mu)$。此处，方括号中与 β 相关的表达式可看成是 β 上的伽马分布的对数——参见式(23.15)——给出对于固定的 Q_μ，最小化 $\tilde{F}$ 的分布。

$$Q_\sigma^{\mathrm{opt}}(\beta) = \Gamma(\beta;b',c') \tag{33.44}$$

---

① 先验 $P(\sigma) \propto 1/\sigma$ 变换成 $P(\beta) \propto 1/\beta$。

其中

$$\frac{1}{b'}=\frac{1}{2}(N\sigma_{\mu|D}^2+S)\text{ 和 }c'=\frac{N}{2} \tag{33.45}$$

在图 33.4 中,从任一初始条件开始,交替使用这两种更新规则(33.41,33.44)。算法在若干次迭代之后收敛于最优的近似总体。

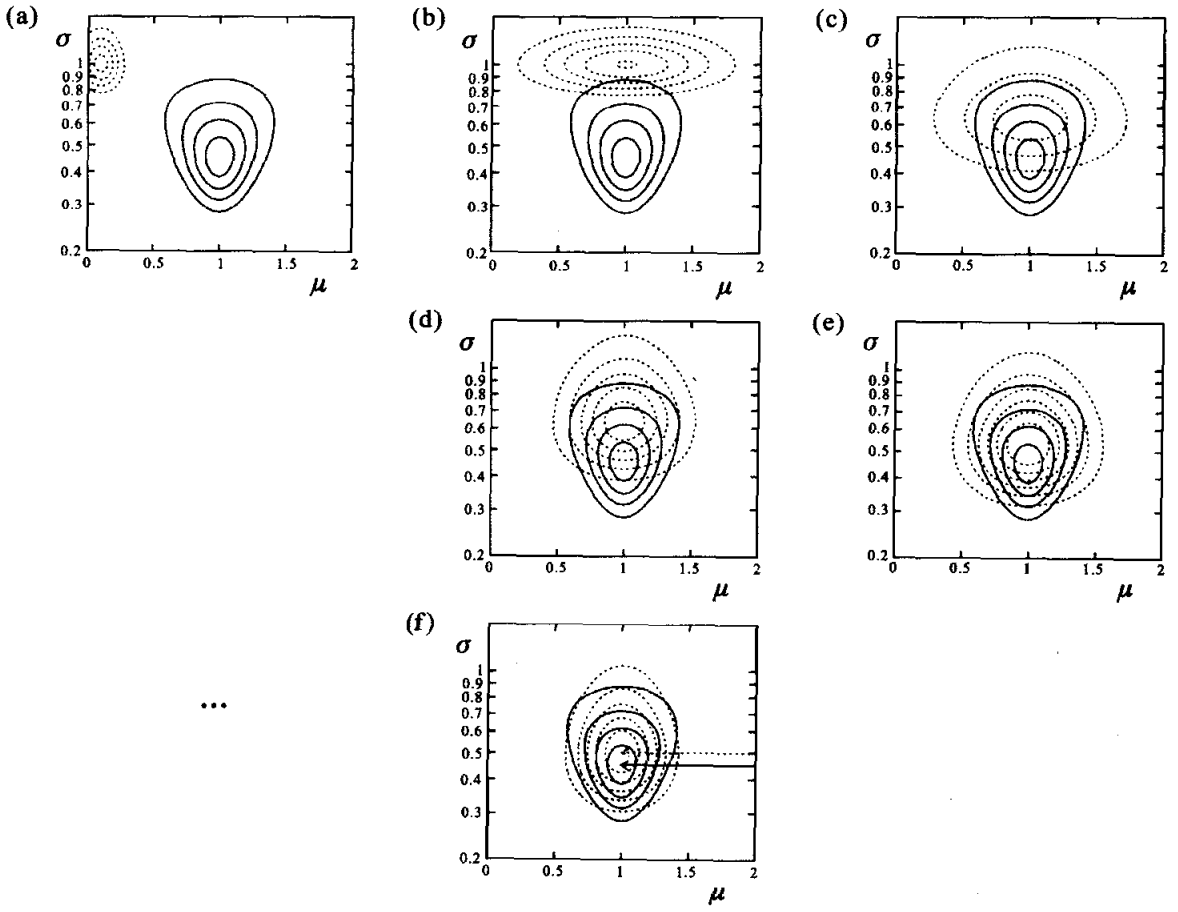


图 33.4 近似分布的优化。用实等势线描绘的后验分布 $P(\mu,\sigma|\{x_n\})$,它和图 24.1 中所示的相同。(a) 初始条件。近似分布 $Q(\mu,\sigma)$(虚等势线)是一个任意可分的分布。(b) Q_μ 已经用式(33.41)进行更新。(c) Q_σ 已经用方程(33.44)进行更新。(d) Q_μ 被再次更新。(e) Q_σ 被再次更新。(f) 收敛的近似(15 次迭代之后)。箭头指向两个分布的峰值,在 $\sigma_N=0.45$ 处(对于 P)和 $\sigma_{N-1}=0.5$ 处(对于 Q)。

联合优化 $Q_\mu(\mu)Q_\sigma(\sigma)$ 的直接解

在这个问题中,不必借助迭代计算来求最优的近似总体。式(33.41)和式(33.44)间接地定义了优化。必须同时满足 $\sigma_{\mu|D}^2=1/(N\bar{\beta})$ 和 $\bar{\beta}=b'c'$。问题的解为

$$1/\bar{\beta}=S/(N-1) \tag{33.46}$$

该式类似于 σ 的真实后验分布,它是满足 $c'=\frac{N-1}{2}$ 和 $1/b'=S/2$ 的伽马分布(参见式(24.13))。对于这个真实后验分布,β 的均值也满足 $1/\bar{\beta}=S/(N-1)$;唯一的区

别在于,近似分布的参数 c' 大了 1/2。

变分自由能最小化所给出的近似总是倾向于比真实分布更为紧凑。

总之,总体学习所给出的对后验的近似同传统的估计器非常一致。β 的近似后验分布是具有均值 $\bar{\beta}$ 的伽马分布,$\bar{\beta}$ 与方差 $\sigma^2 = S/(N-1) = \sigma_{N-1}^2$ 相对应。而且 μ 的近似后验分布是具有均值 $\bar{x}$、标准差 $\sigma_{N-1}/\sqrt{N}$ 的一个高斯分布。

变参自由能最小化法具有很好的性质:它是独立于参数化的;它避免了 MAP 法和拉普拉斯法中所经历的基相关性问题。

VIBES 是用图模型自动实现变参推理的一种易用软件包(Bishop 和 Winn,2000 年;Bishop 等,2002 年;Bishop 和 Winn,2003 年)。它在变参推理中所起的作用和 BUGS 在蒙特卡罗推理中的作用一样。

33.6 插 曲

我的一个学生问:

假设真实分布非常复杂,无法直接计算出来,那么你如何给出一个有用的近似分布呢?

可以从贝叶斯数据建模的场景中回答这个问题。设要研究的"真实"分布为一系列参数 $\boldsymbol{x}$ 的后验概率分布 $P(\boldsymbol{x}|D)$。数据建模的一种标准做法是通过求出似然函数 $P(D|\boldsymbol{x})$ 或后验分布的最大值,来找到单个"最佳匹配"的参数设置(如 $\boldsymbol{x}^*$)。这种标准做法的一种解释是,有关 $\boldsymbol{x}$ 的完整描述 $P(\boldsymbol{x}|D)$ 由一个 δ - 函数来近似,它是一个集中于 $\boldsymbol{x}^*$ 的概率分布。从这个角度来看,任意近似分布 $Q(\boldsymbol{x};\boldsymbol{\theta})$,无论它多么复杂,都是对标准方法所产生的尖峰函数的改进!所以即便只使用一个简单的高斯近似,效果也相当好。

现在讨论变参法的一个应用实例——数据聚类。

33.7 作为变参法的 K - 均值聚类和期望最大化算法

在第 20 章中,介绍了软 K - 均值聚类算法第 1 版。在第 22 章中,介绍了该算法的第 2 版和第 3 版,并启发我们将该算法当做一个最大似然算法。

K - 均值聚类是"期望最大化"(EM)算法的一个例子,可分为两步,称之为"赋值"和"更新",也可分别称为"E - 步骤"和"M - 步骤"。

现在给出 K - 均值聚类的更一般观点,这归功于 Neal 和 Hinton(1998 年),他们表明该算法优化了一个变参目标函数。Neal 和 Hinton 的推导适用于任意 EM 算法。

所有事件的概率

用 $\boldsymbol{\theta}$ 表示混合模型的参数——均值、标准差和权重。对于每一数据点，存在一个丢失变量（也可称为隐藏变量），该点的类标记为 k_n。已知所假设的模型为 $\mathcal{H}$，则所有事件的概率为

$$P(\{\boldsymbol{x}^{(n)},k_n\}_{n=1}^N,\boldsymbol{\theta}|\mathcal{H})=P(\boldsymbol{\theta}|\mathcal{H})\prod_{n=1}^{N}[P(\boldsymbol{x}^{(n)}|k_n,\boldsymbol{\theta})P(k_n|\boldsymbol{\theta})] \quad (33.47)$$

已知数据，所有事件的后验概率与所有事件的概率成正比。

$$P(\{k_n\}_{n=1}^N,\boldsymbol{\theta}|\{\boldsymbol{x}^{(n)}\}_{n=1}^N,\mathcal{H})=\frac{P(\{\boldsymbol{x}^{(n)},k_n\}_{n=1}^N,\boldsymbol{\theta}|\mathcal{H})}{P(\{\boldsymbol{x}^{(n)}\}_{n=1}^N|\mathcal{H})} \quad (33.48)$$

现在通过一个可分离的分布来近似该后验分布。

$$Q_k(\{k_n\}_{n=1}^N)Q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \quad (33.49)$$

并用通常的方式定义一个变参自由能。

$$\tilde{F}(Q_k,Q_{\boldsymbol{\theta}})=\sum_{\{k_n\}}\int \mathrm{d}^D\boldsymbol{\theta}Q_k(\{k_n\}_{n=1}^N)Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})\ln\frac{Q_k(\{k_n\}_{n=1}^N)Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{P(\{\boldsymbol{x}^{(n)},k_n\}_{n=1}^N,\boldsymbol{\theta}|\mathcal{H})} \quad (33.50)$$

$\tilde{F}$ 以负的显著性 $\ln P(\{\boldsymbol{x}^{(n)}\}_{n=1}^N|\mathcal{H})$ 为下界。现在可以通过一个“赋值”步骤和一个“更新”步骤构造一个迭代算法。在赋值步骤中，对于固定的 $Q_{\boldsymbol{\theta}}$，调整 $Q_k(\{k_n\}_{n=1}^N)$ 以降低 $\tilde{F}$；在更新步骤中，对于固定的 Q_k，调整 $Q_{\boldsymbol{\theta}}$ 以降低 $\tilde{F}$。

如果希望精确地得到软 K－均值算法，可对近似分布进一步施加一个约束：将 $Q_{\boldsymbol{\theta}}$ 限定为中心在 $\boldsymbol{\theta}$ 点估计值的 δ－函数，$\boldsymbol{\theta}=\boldsymbol{\theta}^*$。

$$Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})=\delta(\boldsymbol{\theta}-\boldsymbol{\theta}^*) \quad (33.51)$$

遗憾的是，这个分布为变参自由能贡献了一个无穷大积分 $\int \mathrm{d}^D\boldsymbol{\theta}Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})\ln Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$，所以最好把那个项从 $\tilde{F}$ 中省略，把它看成是一个加性常量。[如果我们的目的是最小化 $\tilde{F}$，那么采用 δ－函数 $Q_{\boldsymbol{\theta}}$ 并不是一个好主意！]接下来，我们的目的是推导软 K－均值算法。

▷ **习题 33.2**[2] 已知 $Q_{\boldsymbol{\theta}}(\boldsymbol{\theta})=\delta(\boldsymbol{\theta}-\boldsymbol{\theta}^*)$，证明最小化 $\tilde{F}$ 的最优 Q_k 是一个可分离的分布，其概率 $k_n=k$ 由责任 $r_k^{(n)}$ 给出。

▷ **习题 33.3**[2] 已知一个如上所述的可分离的 Q_k，证明通过软 K－均值算法的更新步骤，可得出最小化 $\tilde{F}$ 的最优 $\boldsymbol{\theta}^*$。（假设 $\boldsymbol{\theta}$ 的先验是均匀分布。）

习题 33.4[2] 通过允许 $Q_{\boldsymbol{\theta}}$ 为一个更通用的分布而非 δ－函数，可以立即改进软 K－均值聚类所得的无穷大的 $\tilde{F}$ 值。允许 $Q_{\boldsymbol{\theta}}$ 是一个可分离分布：$Q_{\mu}(\{\mu\})$、$Q_{\sigma}(\{\sigma\})$ 和 $Q_{\pi}(\pi)$ 的乘积，在此情况下推导更新步骤。讨论这个推广算法是否仍然

存在软 K - 均值的“发散”问题,此时算法把渐缩高斯局限于一个数据点上。

遗憾的是,虽然令 Q_θ 为一个非 δ - 函数看起来像是该算法的一个有前途的推广,而且排除了“发散”问题,在该近似推理法中还可能出现其他问题,涉及 $\tilde{F}$ 的局部最小值。进一步的读物可参见(MacKay,1997a;MacKay,2001 年)。

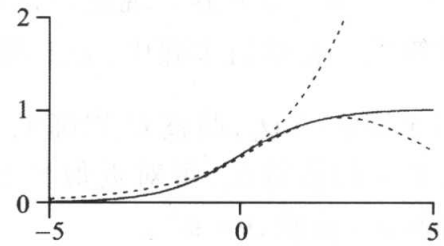
33.8 非自由能最小化的变参法

除了那些基于最小化近似分布 Q 和 P 之间的相对熵,还有其他的策略来近似复杂的分布 $P(\boldsymbol{x})$。一种由 Jaakkola 和 Jordan 最早提出的方法是为 P 创建可调整的上界 Q^U 和下界 Q^L,如图 33.5 所示。这些界限(未归一化的密度)由可变参数来参数化,调整这些参数以获得最紧的可能拟合。下界可被调整来最大化

$$\sum_{\boldsymbol{x}} Q^L(\boldsymbol{x}) \tag{33.52}$$

而上界可被调整来最小化

$$\sum_{\boldsymbol{x}} Q^U(\boldsymbol{x}) \tag{33.53}$$



上界

$$\frac{1}{1+\mathrm{e}^{-a}} \leqslant \exp(\mu a - H_2^e(\mu)), \quad \mu \in [0,1]$$

下界

$$\frac{1}{1+\mathrm{e}^{-a}} \geqslant g(\nu)\exp[(a-\nu)/2 - \lambda(\nu)(a^2-\nu^2)]$$

其中

$$\lambda(\nu) = [g(\nu) - 1/2]/2\nu。$$

图 33.5 Jaakkola - Jordan 变参法图解。Logistic 函数 $g(a) \equiv \dfrac{1}{1+\mathrm{e}^{-a}}$(实线)的上界和下界。这些上/下界都是 a 的指数函数或高斯函数,因此易于求积。该图显示了 S 形函数以及 $\mu = 0.505$ 和 $\nu = -2.015$ 的上界和下界。

使用归一化形式的优化界,就可计算对于预测分布的近似。关于这些方法的进阶读物可在参考文献(Jaakkola 和 Jordan,2000a;Jaakkola 和 Jordan,2000b;Jaakkola 和 Jordan,1996 年;Gibbs 和 MacKay,2000 年)中找到。

进一步的阅读:Bethe 和 Kikuchi 自由能

在第 26 章讨论过具有因式图形式的函数的和 - 积算法(第 26.1 节)。如果因式图是树形的,则和 - 积算法收敛,并可准确算出任一变量 x_n 的边缘函数,也可得到出现在公共因式(如 $\boldsymbol{x}_m$)中变量的子集的联合边缘函数。

和 - 积算法也可以应用于非树形的因式图。如果算法收敛到一个不动点,已证明该不动点是一个被称为 Kikuchi 自由能的消息函数的驻点(通常是一个最小值)。在因式图中,所有因式都是一个或两个变量的函数,在此特殊情况下,Kikuchi 自由能被称为 Bethe 自由能。

关于该思路的论文及由此导出的新近似推理算法,可以参见 Yedidia(2000 年);Yedidia 等(2000c);Welling 和 Teh(2001 年);Yuille(2001 年);Yedidia 等(2000b);Yedidia 等(2000a)。

33.9 进一步的习题



习题 33.5[2] 本习题探讨上面的论断,即通过变参自由能最小化所给出的近似分布总是趋向于比真实分布更为紧凑。考虑一个二维高斯分布 $P(\boldsymbol{x})$,其轴的方向为 $\boldsymbol{e}^{(1)}=(1,1)$ 和 $\boldsymbol{e}^{(2)}=(1,-1)$。设这两个方向的方差为 σ_1^2 和 σ_2^2。若该分布通过方差为 σ_Q^2 的球形高斯分布来近似,由变参自由能最小化所得的最优方差是多少?如果代之优化目标函数

$$G=\int \mathrm{d}\boldsymbol{x} P(\boldsymbol{x}) \ln \frac{P(\boldsymbol{x})}{Q(\boldsymbol{x};\sigma^2)} \tag{33.54}$$

则 σ^2 的最优值是什么?在 $\sigma_1/\sigma_2=10$ 的情况下,画出真实分布 $P(\boldsymbol{x})$ 和两个近似分布的等势线。

[注意:一般不可能计算目标函数 G,因为真实分布 $P(\boldsymbol{x})$ 下的积分通常是很难处理的。]



习题 33.6[2] 采用变参法来优化近似分布 Q,再将该分布作为重要性采样的提议密度。你是如何看待这种做法的?



习题 33.7[2] 定义两个概率分布 P 和 Q 的相对熵或 Kullback - Leibler 分歧,并陈述 Gibbs 不等式。考虑用一个可分离的分布 $Q(x,y)=Q_X(x)Q_Y(y)$ 来近似一个联合分布 $P(x,y)$ 的问题。证明如果该近似的目标函数为

$$G(Q_X,Q_Y)=\sum_{x,y}P(x,y)\log_2\frac{P(x,y)}{Q_X(x)Q_Y(y)}$$

当 Q_X 和 Q_Y 等于 x 和 y 的边缘分布时，取得 G 的最小值。

现在考虑另一个目标函数

$$F(Q_X,Q_Y)=\sum_{x,y}Q_X(x)Q_Y(y)\log_2\frac{Q_X(x)Q_Y(y)}{P(x,y)}$$

用一个可分离的分布 $Q(x,y)=Q_X(x)Q_Y(y)$ 来近似如下概率分布 $P(x,y)$。若将 Q_X 和 Q_Y 设置为 x 和 y 上的边缘分布，给出 $F(Q_X,Q_Y)$ 的值。

$P(x,y)$		x			
		1	2	3	4
y	1	1/8	1/8	0	0
	2	1/8	1/8	0	0
	3	0	0	1/4	0
	4	0	0	0	1/4

证明 $F(Q_X,Q_Y)$ 有 3 个不同的最小值，确定这些最小值，并计算对应于每个最小值的 F 值。

33.10 习题解答

习题 33.5 的解答 需要知道两个一维高斯分布之间的相对熵。

$$\int \mathrm{d}x\,\mathrm{Normal}(x;0,\sigma_Q)\ln\frac{\mathrm{Normal}(x;0,\sigma_Q)}{\mathrm{Normal}(x;0,\sigma_P)}$$

$$=\int \mathrm{d}x\,\mathrm{Normal}(x;0,\sigma_Q)\left[\ln\frac{\sigma_P}{\sigma_Q}-\frac{1}{2}x^2\left(\frac{1}{\sigma_Q^2}-\frac{1}{\sigma_P^2}\right)\right] \tag{33.55}$$

$$=\frac{1}{2}\left(\ln\frac{\sigma_P^2}{\sigma_Q^2}-1+\frac{\sigma_Q^2}{\sigma_P^2}\right) \tag{33.56}$$

所以，如果用方差均为 σ_Q^2 的 Q 来近似方差为 σ_1^2 和 σ_2^2 的 P 时，可以发现

$$F(\sigma_Q^2)=\frac{1}{2}\left(\ln\frac{\sigma_1^2}{\sigma_Q^2}-1+\frac{\sigma_Q^2}{\sigma_1^2}+\ln\frac{\sigma_2^2}{\sigma_Q^2}-1+\frac{\sigma_Q^2}{\sigma_2^2}\right) \tag{33.57}$$

求导可得

$$\frac{\mathrm{d}}{\mathrm{d}\ln(\sigma_Q^2)}F=\frac{1}{2}\left[-2+\left(\frac{\sigma_Q^2}{\sigma_1^2}+\frac{\sigma_Q^2}{\sigma_2^2}\right)\right] \tag{33.58}$$

其值为零，如果

$$\frac{1}{\sigma_Q^2}=\frac{1}{2}\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}\right) \tag{33.59}$$

因此，设近似分布的方差倒数为目标分布 P 的平均方差倒数。

当 $\sigma_1 = 10$ 且 $\sigma_2 = 1$ 时，得到 $\sigma_Q \simeq \sqrt{2}$，它只是比 σ_2 大一个因子 $\sqrt{2}$，完全和大的标准差 σ_1 无关。变参自由能最小化通常导出近似分布满足：这些分布的长度刻度与目标分布的最短刻度相匹配。该近似分布可以被视做太紧凑了。

相反的，如果使用目标函数 G，那么可以发现

$$G(\sigma_Q^2) = \frac{1}{2}\left(\ln\sigma_Q^2 + \frac{\sigma_1^2}{\sigma_Q^2} + \ln \sigma_Q^2 + \frac{\sigma_2^2}{\sigma_Q^2} \right) + \text{常数} \tag{33.60}$$

其中常数仅取决于 σ_1 和 σ_2。微分可得

$$\frac{\mathrm{d}}{\mathrm{d} \ln \sigma_Q^2} G = \frac{1}{2}\left[2 - \left(\frac{\sigma_1^2}{\sigma_Q^2} + \frac{\sigma_2^2}{\sigma_Q^2} \right) \right] \tag{33.61}$$

当

$$\sigma_Q^2 = \frac{1}{2}(\sigma_1^2 + \sigma_2^2) \tag{33.62}$$

时，其值为零。因此，设近似分布的方差为目标分布 P 的均方差。

当 $\sigma_1 = 10$ 且 $\sigma_2 = 1$ 时，得到 $\sigma_Q \simeq 10/\sqrt{2}$，它只比 σ_1 小一个因子 $\sqrt{2}$，与 σ_2 值无关。

这两个近似按比例如图 33.6 所示。

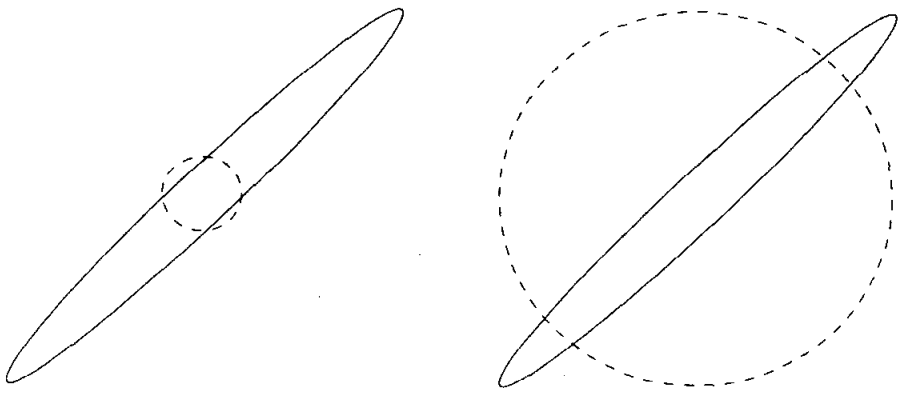


图 33.6 对双变量高斯分布（实线）的两个可分离高斯近似（虚线）。（a）最小化变参自由能的近似。（b）最小化目标函数 G 的近似。在每幅图中，曲线表示 $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} = 1$ 的等势线，其中 $\boldsymbol{A}$ 是高斯协方差矩阵的逆。

习题 33.6 的解答 最佳可能的变参近似当然是目标分布 P。假设这是不可能的，一个好的变参近似比真实分布更为紧凑。相反的，一个好的采样器比真实分布有更大的拖尾。一个过于紧凑的分布将是一个糟糕的采样器，其方差很大。

第 34 章

独立元素分析与隐含变量建模

34.1 隐含变量模型

很多统计模型是对可观测量运用隐含变量来描述一个概率分布的生成模型(即为该情况下的所有变量指定一个全概率密度的模型)。

隐含变量模型的例子包括第 22 章的混合模型。这个模型对来自简单概率分布的叠加混合体的可观测量进行建模(隐含变量是实例中的未知类标签);隐马尔科夫模型(Rabiner 和 Juang,1986 年;Durbin 等,1998 年)及因子分析。

纠错码的译码问题也可从隐含变量模型的角度来看(如图 34.1 所示)。此时编码矩阵 $\boldsymbol{G}$ 通常是预先知道的。在隐含变量建模时,与 $\boldsymbol{G}$ 等价的参数通常是未知的,而且必须由数据和隐含变量 $\boldsymbol{s}$ 一起推导。

通常,隐含变量服从一个简单的分布,而且往往是一个可分离的分布。因此当拟合隐含变量模型时,可以根据“独立元素”来寻找对数据的描述。“独立元素分析”算法对应于最简单的具有连续隐含变量的可能的隐含变量模型。

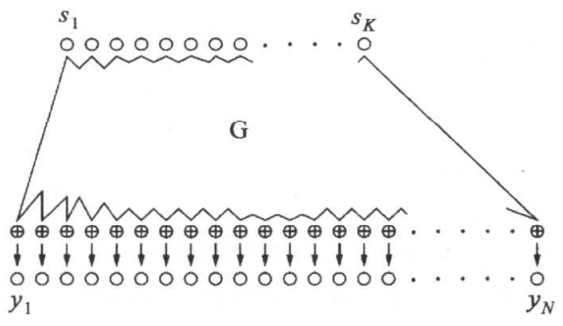


图 34.1 纠错码作为隐含变量模型。K 个隐含变量是独立信源比特 $s_1,\cdots,s_K$;它们通过生成矩阵 $\boldsymbol{G}$ 产生可观测量。

34.2 独立元素分析的生成模型

假设通过如下方式产生一组 N 个可观测量 $D=\{\boldsymbol{x}^{(n)}\}_{n=1}^{N}$。每个 J 维向量 $\boldsymbol{x}$ 是 I 个基本信源信号 $\boldsymbol{s}$ 的一个线性组合。

$$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{s} \tag{34.1}$$

其中混合系数矩阵 $\boldsymbol{G}$ 是未知的。

如果假设信源的数量等于可观测量的数量，即 $I = J$，就可以得到最简单的算法。我们的目的是恢复信源变量 $\boldsymbol{s}$（在某些乘法因子之内，可能交换过次序）。换言之，我们的目的是在仅已知一组例子 $\{\boldsymbol{x}\}$ 的情况下，产生 $\boldsymbol{G}$ 的逆（在一个后乘因子之内）。假设隐含变量是独立分布的，其边缘分布为 $P(s_i|\mathcal{H}) \equiv p_i(s_i)$。这里 $\mathcal{H}$ 表示这个模型所采用的形式以及隐含变量所采用的概率分布 p_i。

已知 $\boldsymbol{G}$ 和 $\mathcal{H}$，可观测量和隐含变量的概率是

$$P(\{\boldsymbol{x}^{(n)}, \boldsymbol{s}^{(n)}\}_{n=1}^{N}|\boldsymbol{G}, \mathcal{H}) = \prod_{n=1}^{N} [P(\boldsymbol{x}^{(n)}|\boldsymbol{s}^{(n)}, \boldsymbol{G}, \mathcal{H}) P(\boldsymbol{s}^{(n)}|\mathcal{H})] \tag{34.2}$$

$$= \prod_{n=1}^{N} \left[\left(\prod_j \delta(x_j^{(n)} - \sum_i G_{ji} s_i^{(n)}) \right) \left(\prod_i p_i(s_i^{(n)}) \right) \right] \tag{34.3}$$

假设向量 $\boldsymbol{x}$ 是在无噪声条件下产生的。在隐含变量建模中通常不做这一假设，因为无噪声的数据很少见；但它使得推理问题变得更易于解决。

似然函数

为了从数据 D 中获取关于 $\boldsymbol{G}$ 的信息，相关的量是似然函数

$$P(D|\boldsymbol{G}, \mathcal{H}) = \prod_n P(\boldsymbol{x}^{(n)}|\boldsymbol{G}, \mathcal{H}) \tag{34.4}$$

它是因式的一个乘积，其中每个因式可通过对隐含变量进行边缘化来获得。对 δ - 函数边缘化时，回忆 $\int \mathrm{d}s \delta(x - vs) f(s) = \frac{1}{v} f(x/v)$。在此处采用求和约定，例如使得 $G_{ji} s_i^{(n)} \equiv \sum_i G_{ji} s_i^{(n)}$。在似然中的一个因式由下式给出。

$$P(\boldsymbol{x}^{(n)}|\boldsymbol{G}, \mathcal{H}) = \int \mathrm{d}^I \boldsymbol{s}^{(n)} P(\boldsymbol{x}^{(n)}|\boldsymbol{s}^{(n)}, \boldsymbol{G}, \mathcal{H}) P(\boldsymbol{s}^{(n)}|\mathcal{H}) \tag{34.5}$$

$$= \int \mathrm{d}^I \boldsymbol{s}^{(n)} \prod_j \delta(x_j^{(n)} - G_{ji} s_i^{(n)}) \prod_i p_i(s_i^{(n)}) \tag{34.6}$$

$$= \frac{1}{|\det \boldsymbol{G}|} \prod_i p_i(G_{ij}^{-1} x_j) \tag{34.7}$$

$$\Rightarrow \ln P(\boldsymbol{x}^{(n)}|\boldsymbol{G}, \mathcal{H}) = -\ln|\det \boldsymbol{G}| + \sum_i \ln p_i(G_{ij}^{-1} x_j) \tag{34.8}$$

为了获得一个最大似然算法，寻找对数似然的梯度。如果引入 $\boldsymbol{W} \equiv \boldsymbol{G}^{-1}$，由单一例子所产生的对数似然可以写成

$$\ln P(\boldsymbol{x}^{(n)}|\boldsymbol{G}, \mathcal{H}) = \ln|\det \boldsymbol{W}| + \sum_i \ln p_i(W_{ij} x_j) \tag{34.9}$$

自此假设 $\det \boldsymbol{W}$ 是正的，所以可以忽略绝对值符号。求解过程中需要用到以下恒等式：

$$\frac{\partial}{\partial G_{ji}} \ln \det \boldsymbol{G} = G_{ij}^{-1} = W_{ij} \tag{34.10}$$

$$\frac{\partial}{\partial G_{ji}} G_{lm}^{-1} = -G_{lj}^{-1} G_{im}^{-1} = -W_{lj} W_{im} \tag{34.11}$$

$$\frac{\partial}{\partial W_{ij}} f = -G_{jm} \left(\frac{\partial}{\partial G_{lm}} f \right) G_{li} \tag{34.12}$$

定义 $a_i \equiv W_{ij} x_j$

$$\phi_i(a_i) \equiv \mathrm{d} \ln p_i(a_i) / \mathrm{d}a_i \tag{34.13}$$

以及 $z_i = \phi_i(a_i)$，这表明 a_i 需要向哪个方向改变从而使数据的概率变得更大。这样就可以通过式(34.10)和式(34.11)获得关于 G_{ji} 的梯度。

$$\frac{\partial}{\partial G_{ji}} \log P(\boldsymbol{x}^{(n)} | \boldsymbol{G}, \mathscr{H}) = -W_{ij} - a_i z_{i'} W_{i'j} \tag{34.14}$$

或者，关于 W_{ij} 的导数

$$\frac{\partial}{\partial W_{ij}} \ln P(\boldsymbol{x}^{(n)} | \boldsymbol{G}, \mathscr{H}) = G_{ji} + x_j z_i \tag{34.15}$$

如果选择改变 $\boldsymbol{W}$ 来提高这一梯度，可以得到学习规则

$$\Delta \boldsymbol{W} \propto [\boldsymbol{W}^{\mathrm{T}}]^{-1} + \boldsymbol{z}\boldsymbol{x}^{\mathrm{T}} \tag{34.16}$$

到目前为止的算法总结在算法 34.2 中。

对每个数据点 $\boldsymbol{x}$ 重复以下内容。

1. 将 $\boldsymbol{x}$ 输入一个线性映射器：$\boldsymbol{a} = \boldsymbol{W}\boldsymbol{x}$。

2. 将 $\boldsymbol{a}$ 输入一个非线性映射器：$z_i = \phi_i(a_i)$。

ϕ 的一个通常的选择是 $\phi = -\tanh(a_i)$。

3. 根据下式调整权重。

$$\Delta \boldsymbol{W} \propto [\boldsymbol{W}^{\mathrm{T}}]^{-1} + \boldsymbol{z}\boldsymbol{x}^{\mathrm{T}}$$

算法 34.2 独立元素分析——在线最速上升版。也可参考算法 34.4，它是首选的。

ϕ 的选择

函数 ϕ 的选择定义了隐式变量 s 假定的先验分布。

首先考虑线性选择 $\phi_i(a_i) = -\kappa a_i$，暗含地(通过式(34.13))假设隐式变量服从高斯分布。隐式变量的高斯分布在隐式变量旋转时是不变的，所以并无证据倾向于隐式变量空间的任何特定方位。因为线性算法将永远不能恢复矩阵 $\boldsymbol{G}$ 或原始信源，而变得没有吸引力。我们仅存的希望是信源是非高斯的。令人欣慰的是，大多数实际信源具有非高斯分布；通常它们具有比高斯更大的拖尾。

因此继续考虑常用的 tanh 非线性特性。如果

$$\phi_i(a_i) = -\tanh(a_i) \tag{34.17}$$

则暗含地假设

$$p_i(s_i) \propto 1/\cosh(s_i) \propto \frac{1}{\mathrm{e}^{s_i} + \mathrm{e}^{-s_i}} \tag{34.18}$$

这对于隐式变量而言，是一个比高斯分布有更大拖尾的分布。

还可以使用增益为 β 的 tanh 非线性，即 $\phi_i(a_i) = -\tanh(\beta a_i)$，它的隐式概率模型是 $p_i(s_i) \propto 1/[\cosh(\beta s_i)]^{1/\beta}$。在大 β 的极限情况下，非线性变成了一个阶梯函数，而概率分布 $p_i(s_i)$ 变成了一个双边的指数分布，$p_i(s_i) \propto \exp(-|s|)$。在极限 $\beta \to 0$ 时，$p_i(s_i)$ 趋向于一个均值为零、方差为 $1/\beta$ 的高斯分布。更大拖尾的分布也可以使用。学生分布和柯西分布会跳入我们的脑海。

实例分布

图 34.3(a ~ c)表示当元素服从 1/cosh 和柯西分布时，由独立元素模型所生成的典型分布。如图 34.3d 所示为来自柯西模型的某些样本。柯西分布，具有更大的拖尾，给出了预测分布如何依赖于假设生成参数 $\boldsymbol{G}$ 的最清晰的画面。

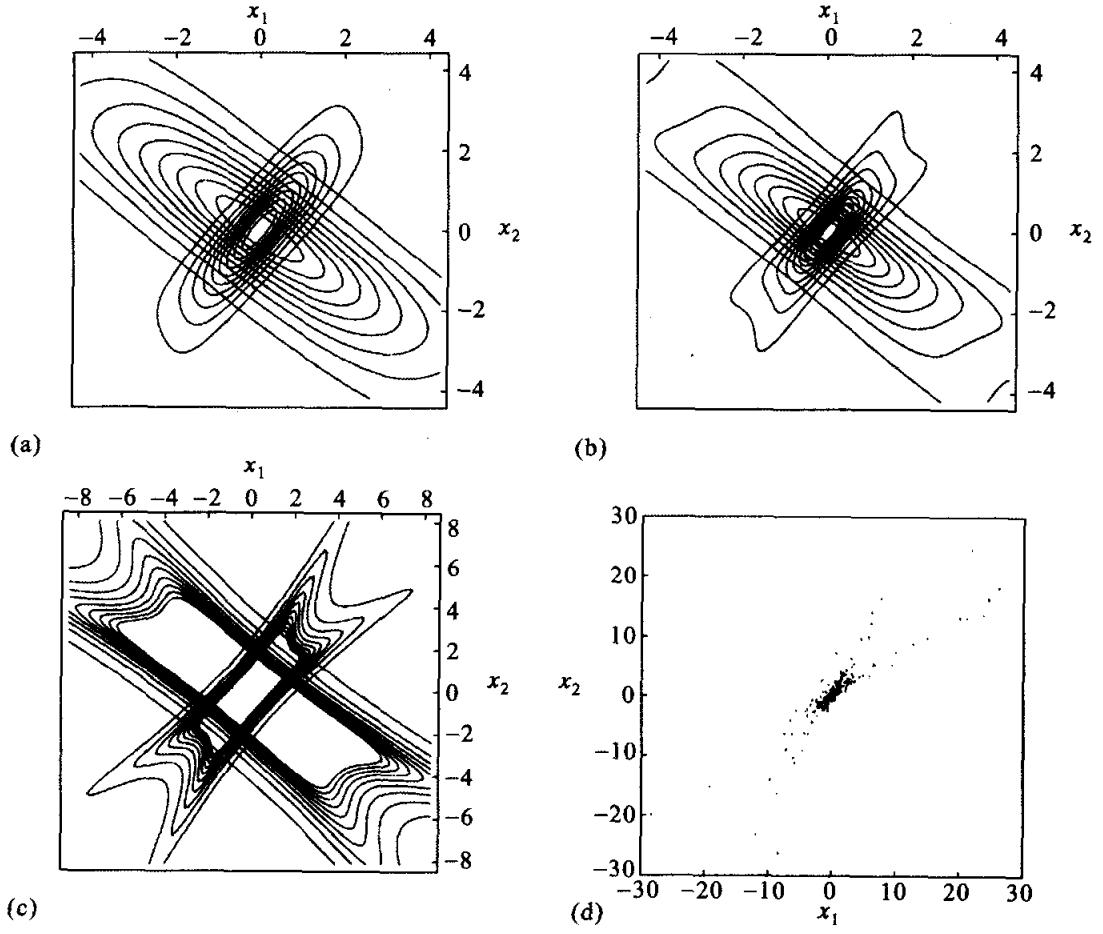


图 34.3 隐含于学习算法中的生成模型的图解。(a) 当隐式变量服从 1/cosh 分布时所生成的两个可观测量的分布，$\boldsymbol{G} = \begin{bmatrix} 3/4 & 1/2 \\ 1/2 & 1 \end{bmatrix}$(紧凑分布)和 $\boldsymbol{G} = \begin{bmatrix} 2 & -1 \\ -1 & 3/2 \end{bmatrix}$(宽分布)。(b) 当隐式变量服从柯西分布时，生成分布的等势线。学习算法采用适当的方法将这种不规则的目标拟合到经验数据，以便使似然最大化。在(b)中绘制的等势线并不能充分地代表这个大拖尾分布。(c) 柯西分布的一部分拖尾，假设等势线表示原点密度的 0.01…0.1 倍。(d) 如图(b)和(c)所示，一个生成式分布所产生的一些数据。你能够说出是哪一个吗？产生了 200 个样本，其中 196 个位于画出的区域中。

34.3 一个协变的、更简单而快速的学习算法

就这样推导出一个学习算法，它在似然函数上最疾速地下降。算法运行并非很快，即使对于简化数据也是如此；算法是病态的而且可以很好地表明一个一般的建议：尽管求出一个目标函数的梯度是一个极好的想法，然而直接沿梯度上升则可能不明智。算法是病态的这一事实可以通过这样的事实看到：它包括一个矩阵的逆，而这个逆可以是任意大甚至从未定义的。

通常的协变优化方法

协方差原理表明：一个一致的算法应该给出相同的结果而不依赖于变量的测量单位（Knuth，1968 年）。一个非协变算法的最好例子是普遍使用的最速下降规则。定义一个无量纲的目标函数 $L(\boldsymbol{w})$，并计算它关于某些参数 $\boldsymbol{w}$ 的导数，则 $\boldsymbol{w}$ 根据下述规则变化

$$\Delta w_i = \eta \frac{\partial L}{\partial w_i} \tag{34.19}$$

这个常用的等式在量纲上不一致：等式左边具有 $[w_i]$ 的量纲，等式右边具有 $1/[w_i]$ 的量纲。学习算法（34.19）的特性并不随向量 $\boldsymbol{w}$ 线性改变而协变。量纲的不一致性并没有太大影响，正如众多成功的梯度下降算法已经表明的，并且事实上，如果 η 随着 n 依 $1/n$ 下降（在线学习的过程中），则 Munro－Robbins 定理（Bishop，1992 年）证明参数将渐近地收敛到最大似然参数。但是非协变算法可能需要使用大量的迭代来获得这种收敛；的确，很多最速下降法的老用户喜欢使用诸如共轭梯度之类的算法，从而自适应地计算出目标函数的曲率。为式（34.19）所做的辩护指明：如果不是所有参数 w_i 具有相同的量纲，η 应该是一个有量纲的常数的论点就站不住脚。

如果算法具有形式

$$\Delta w_i = \eta \sum_{i'} M_{ii'} \frac{\partial L}{\partial w_i} \tag{34.20}$$

它应该是协变的，其中 $\boldsymbol{M}$ 是一个正定矩阵，它的 i、i' 元素具有量纲 $[w_i w_{i'}]$。可以从哪里得到这样的矩阵呢？这种矩阵的两个来源是度量和曲率。

度量与曲率

如果在参数空间 $\boldsymbol{w}$ 中存在一个自然的度量来定义距离，则矩阵 $\boldsymbol{M}$ 可以从度量中获得。经常会存在一种自然的选择。在特殊情况下，有一个众所周知的用来定义一个向量 $\boldsymbol{w}$ 长度的二次度量，矩阵可以从该二次型中获得。例如，如果长度是 $\boldsymbol{w}^2$，那么自然矩阵是 $\boldsymbol{M}=\boldsymbol{I}$，且最速下降法是合适的。

寻找度量的另一种方法是考虑目标函数的曲率，定义 $\boldsymbol{A} \equiv -\nabla\nabla L$（其中 $\nabla \equiv$

$\partial/\partial w$)。则矩阵 $M=A^{-1}$ 将给出一个协变算法；而且，这个算法是牛顿算法，所以我们认识到它会减轻最速下降法中的一个主要难题，即当目标函数是病态时，它到最小值的缓慢收敛。如果 L 是二次的，牛顿算法只需一步就收敛到最小值。

在某些问题中，它也许是曲率 A，它包括与数据相关的项和与数据无关的项；在这种情况下，也许可以选择使用与数据无关的项来定义度量(Gull，1989 年)。所得的算法将依旧是协变的，但是它不会执行一个精确的牛顿步骤。很明显，存在很多的协变算法，并非只有唯一的选择。而协变算法也只是所有算法集合中的一个小子集！

回到独立元素分析

对于目前的最大似然问题，已经计算了关于 G 的梯度和关于 $W=G^{-1}$ 的梯度。W 的最速下降法不是协变的。让我们构造另一个在对数似然曲率帮助下的协变算法。将对数似然关于 W 做二阶微分，得到两项，第一项是与数据无关的。

$$\frac{\partial G_{ji}}{\partial W_{kl}}=-G_{jk}G_{li} \tag{34.21}$$

第二项与数据有关。

$$\frac{\partial(z_i x_j)}{\partial W_{kl}}=x_j x_l \delta_{ik} z_i', \quad \text{在 } i \text{ 上不求和} \tag{34.22}$$

其中 z' 是 z 的导数。我们很想丢弃数据相关项并用 $[M^{-1}]_{(ij)(kl)}=[G_{jk}G_{li}]$ 来定义矩阵 M。但是，这个矩阵不是正定的(它至少具有一个非正特征值)，所以它不是对数似然曲率的一个好逼近，在一个最大似然解的邻域中，对数似然曲率必须是正定的。因此，必须从数据相关项中获得灵感。目的是为曲率寻找一个方便的逼近以获得一个协变算法，并不一定要执行一个准确的牛顿步骤。$x_j x_l \delta_{ik} z_i'$ 的平均值是多少？如果 G 的真值是 G^*，则

$$\langle x_j x_l \delta_{ik} z_i' \rangle = \langle G^*_{jm} s_m s_n G^*_{ln} \delta_{ik} z_i' \rangle \tag{34.23}$$

现在做几个严格的近似：用 G 的当前值来代替 G^*，并用 $\langle s_m s_n \rangle \langle z_i' \rangle \equiv \sum_{mn} D_i$ 来代替相关平均数 $\langle s_m s_n z_i' \rangle$。这里 $\boldsymbol{\Sigma}$ 是隐式变量的方差－协方差矩阵(假设是存在的)，并且 D_i 是曲率 $\mathrm{d}^2 \ln p_i(a)/\mathrm{d}a^2$ 的典型值。已知信源是独立的，$\boldsymbol{\Sigma}$ 和 $\boldsymbol{D}$ 是两个对角矩阵。这些近似促使通过下式给出矩阵 M。

$$[M^{-1}]_{(ij)(kl)}=G_{jm}\Sigma_{mn}G_{ln}\delta_{ik}D_i \tag{34.24}$$

即

$$M_{(ij)(kl)}=W_{mj}\Sigma^{-1}_{mn}W_{nl}\delta_{ik}D_i^{-1} \tag{34.25}$$

为了简单起见，进一步假设信源是彼此相似的，因此 $\boldsymbol{\Sigma}$ 和 $\boldsymbol{D}$ 都是齐次的，并且 $\boldsymbol{\Sigma D}=1$。这将导致一个按数据 $\boldsymbol{x}$ 的线性改变但非按隐式变量线性改变的协变算法。因此，通过

$$M_{(ij)(kl)}=W_{mj}W_{ml}\delta_{ik} \tag{34.26}$$

将这个矩阵与式(34.15)的梯度相乘，得到下列协变学习算法

$$\Delta W_{ij} = \eta(W_{ij} + W_{i'j}a_{i'}z_i) \tag{34.27}$$

注意到这个表达式并不要求矩阵 W 的任何逆矩阵。一旦求出了 z，唯一增加的计算是通过权重的后向传递求解量

$$x'_j = W_{i'j}a_{i'} \tag{34.28}$$

据此，得到协变算法

$$\Delta W_{ij} = \eta(W_{ij} + x'_j z_i) \tag{34.29}$$

公式右边的量 $(W_{ij} + x'_j z_i)$ 有时被称为自然梯度。协变的独立元素分析算法总结在算法 34.4 中。

对于每个数据点 $\boldsymbol{x}$ 重复以下内容。

1. 将 $\boldsymbol{x}$ 送入一个线性映射器

$$\boldsymbol{a} = \boldsymbol{W}\boldsymbol{x}$$

2. 将 $\boldsymbol{a}$ 送入一个非线性映射器

$$z_i = \phi_i(a_i)$$

其中 ϕ 的一个通常的选择是 $\phi = -\tanh(a_i)$。

3. 将 $\boldsymbol{a}$ 送回 $\boldsymbol{W}$

$$\boldsymbol{x}' = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{a}$$

4. 根据下式调整权重。

$$\Delta \boldsymbol{W} \propto \boldsymbol{W} + \boldsymbol{z}\boldsymbol{x}'^{\mathrm{T}}$$

算法 34.4 独立元素分析——协变版。

进一步的阅读

独立元素分析(ICA)最初是通过一种信息最大化方法推导出来的(Bell 和 Sejnowski,1995 年)。ICA 的另一个观点是，根据能量函数，它启发了更一般的模型，由 Hinton 等(2001 年)给出。对 ICA 的另一个推广可以在 Pearlmutter 和 Parra(1996 年，1997 年)中找到。现在关于 ICA 应用的文献非常多。(Miskin,2001 年；Miskin 和 MacKay,2000 年；Miskin 和 MacKay,2001 年)给出了关于类 ICA 模型的变参自由能最小化方法。关于盲分离的进阶阅读包括非 ICA 算法，可以在(Jutten 和 Herault，1991 年；Comon 等，1991 年；Hendin 等，1994 年；Amari 等，1996 年；Hojen - Sorensen 等，2002 年)中找到。

无限模型

尽管含有有限个隐式变量的隐式变量模型得到了广泛的使用，但是实际情况往往是需要使用非常多数量的隐式变量，才可以精确地捕获到相关情形的信念。

例如，考虑聚类。如果通过使用一个聚类模型来为单词建模，以进行语音识别，需要用到多少聚类？可能的单词数量是极大的(第 18.2 节)，所以真正想用的模型是在新的聚类出现时可用的。

此外，如果小心翼翼地为与每个英语单词对应的聚类建模，可能会发现，一个单

词的类本身应该建模为由类组成，实际上是类中含类的递阶。递阶中的第一级将持语者中的男性与女性分开，并将来自不同地区的人分开——印度、英国、欧洲等。在这些聚类中，将每个地区中持不同口音的人又划分为子聚类。这些子聚类也可以有子聚类——直到村庄、街道或家庭这一级。

因此，人们往往想要有无限数量的聚类；在某些情况下，聚类将具有一个递阶结构，而在其他情况下，递阶可能是平的。那么，这样的无限模型如何在有限的计算机中得以实现？又如何建立贝叶斯模型，以避免得到愚蠢的答案？

Neal(1991 年)提出了分类数据的无限混合模型，及用来仿真推理和预测的一种蒙特卡罗方法。具有平坦递阶结构的无限高斯混合模型由 Rasmussen(2000 年)给出。Neal(2001 年)证明如何利用 Dirichlet 扩散树来定义递阶聚类模型。这些观点中的大部分建立在 Dirichlet 过程之上(第 18.2 节)。这将继续是一个活跃的研究领域(Rasmussen 和 Ghahramani，2002 年；Beal 等，2002 年)。

34.4 习 题

习题 34.1[2] 重复算法的推导过程，但是假设在 $\boldsymbol{x}$ 中存在少量的噪声：$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{s} + \boldsymbol{n}$；于是用在 $x_j^{(n)}$ 上的一个均值为 $\sum_i G_{ji} s_i^{(n)}$ 的概率分布代替联合概率(34.3)中的项 $\delta\left(x_j^{(n)} - \sum_i G_{ji} s_i^{(n)}\right)$。证明：如果这个噪声分布具有足够小的标准差，则将得到同样的算法。

习题 34.2[2] 实现协变 ICA 算法并将其应用于简单数据上。

习题 34.3[2~3] 构建适合下列情况的算法：(a) $\boldsymbol{x}$ 包括大的高斯噪声；(b) 测量数多于隐式变量数($J > I$)；(c) 测量数少于隐式变量数($J < I$)。

因子分析假设观测量 $\boldsymbol{x}$ 可以用独立隐式变量 $\{s_k\}$ 和独立加性噪声来描述。因此，观测量 $\boldsymbol{x}$ 由下式给出。

$$\boldsymbol{x} = \boldsymbol{G}\boldsymbol{s} + \boldsymbol{n} \tag{34.30}$$

其中 $\boldsymbol{n}$ 是一个噪声向量，它的元素具有可分离的概率分布。在因子分析中，通常假设 $\{s_k\}$ 和 $\{n_i\}$ 的概率分布是零均值的高斯；噪声项可以具有不同的方差 σ_i^2。

习题 34.4[2] 为了从数据中推断 $\boldsymbol{G}$，构建一个最大似然算法，假设生成模型 $\boldsymbol{x} = \boldsymbol{G}\boldsymbol{s} + \boldsymbol{n}$ 是正确的，并且 $\boldsymbol{s}$ 和 $\boldsymbol{n}$ 具有独立高斯分布。将参数 σ_j^2 包括进来，描述每个 n_j 的方差，并最大化它们的似然。设每个 s_i 的方差为 1。

习题 34.5[2C] 实现 Rasmussen(2000 年)的无限高斯混合模型。

第 35 章

有关随机推理的专题

35.1 如果你不知情，你了解些什么？

例 35.1 在一个准确的实验里测量一个实变量 x。例如，x 可能是中子的半衰期，萤火虫发出的光的波长，Vostok 湖的深度，或木星的卫星 Io的质量。

x 的值以一个“1”开始的概率是多少？比如，该值可以是电子的电荷（以 S.I. 为单位）

$$e = 1.602\cdots \times 10^{-19}\text{C}$$

和玻尔兹曼常数

$$k = 1.380\ 66\cdots \times 10^{-23}\text{JK}^{-1}?$$

它以“9”开始的概率是多少，比如法拉第常数

$$\mathscr{F} = 9.648\cdots \times 10^{4}\text{C mol}^{-1}?$$

第二位数字的情况又如何？x 的尾数从“1.1…”开始的概率是多少，从“9.9…”开始的概率是多少？

解答 研究中子、萤火虫、南极洲或 Jove 的专家也许可以预测 x 的值，因此预测时对第一位数字是有一定把握的，但是如果预测者没有相关领域的知识，又该如何？与“一无所知”相对应的概率分布是什么？

解决这个问题的一种方法是注意到并未指定 x 的单位。如果中子的半衰期用两个星期而非秒来度量，要将数字 x 除以 1 209 600；如果用年来度量它，要将它除以 3×10^{7}。那么，关于 x（特别是它的第一位数字）的知识是否因为单位的改变而受影响？对于一位专家，答案是肯定的；但是如果某人的确对此一无所知，则其答案就是否定的；他们关于 x 的第一位数字的预测是与单位无关的。单位的任意性对应于当 x 与任意数相乘时概率分布的不变性。

如果你不知道一个量是用什么单位来度量的，则第一位数字的概率一定与对数刻度相应部分的长度成正比（如图 35.1 所示）。因此，一个数的第一位数字是 1 的概率为

$$p_1 = \frac{\log 2 - \log 1}{\log 10 - \log 1} = \frac{\log 2}{\log 10} \tag{35.1}$$

现在，$2^{10} = 1\ 024 \simeq 10^{3} = 1\ 000$，所以不需要计算器，就有 $10\log 2 \simeq 3\log 10$，并且

$$p_1 \simeq \frac{3}{10} \tag{35.2}$$

更一般地，第一位数字是 d 的概率为

$$(\log(d+1)-\log(d))/(\log 10-\log 1)=\log_{10}(1+1/d) \tag{35.3}$$

这个关于起始位数字的观测被称为 Benford 定律。一无所知并不对应于一个均匀概率分布。 □

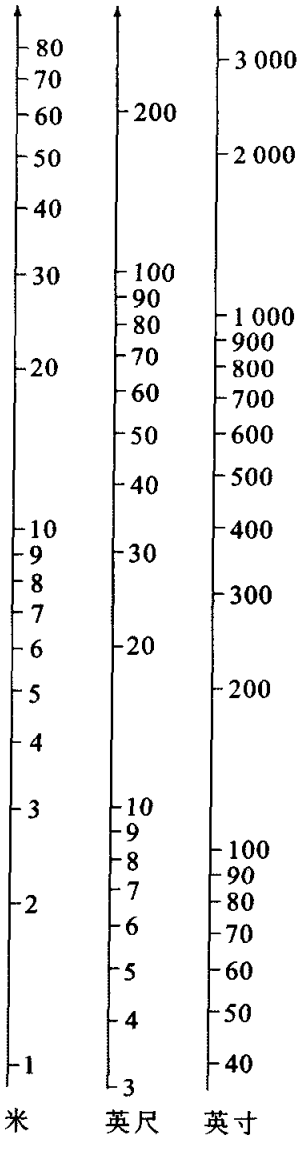
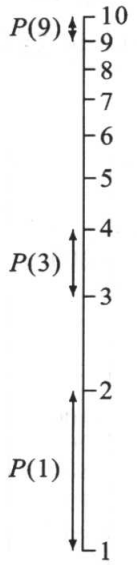


图 35.1 从对数刻度上看，使用不同单位的刻度之间的相互转换。

▷ **习题 35.2**[2] 一个大头针被翻转着抛入空中。大头针在空中瞬间与竖直线之间的夹角为 θ_1 的概率分布是什么？将翻转着的大头针拍摄下来。在照片中所记录的大头针与垂直线之间的夹角为 θ_3 的概率分布是什么？

▷ **习题 35.3**[2] **破记录**。考虑对某个量 x 的世界记录进行跟踪，例如地震量级，或世界锦标赛上的跳远距离。如果假设以稳定速率去冲击记录，并且假设输出 x 的内在概率分布 $P(x)$ 是不变的——尽管笔者认为在体育运动中这不太可能是正确的假设，但仍然是一个令人感兴趣的假设——再假设对 $P(x)$ 一无所知，怎样才能预测相

邻两次破记录日期之间的间隔呢?



35.2 Luria – Delbrück 分布

习题 35.4[ℱC] Luria 和 Delbrück(1943 年)在他们里程碑式的论文中表明,细菌可以发生从对病毒敏感到对病毒有抗性的变异。在实验结束时,他们试图通过指数增长种群中的变异总数,来估计变异率。这个问题是困难的,因为被测量的量(变异细菌数)具有一个大拖尾概率分布:在实验早期产生的变异可能引起数量巨大的变异。遗憾的是,Luria 和 Delbrück 不知道贝叶斯定理,他们用来处理大拖尾分布的方法涉及任意截断,并得出了对变异率的两种不同估计器。其中一个估计器(基于变异细菌的平均数,对几个实验求平均)具有大得惊人的方差,然而抽样理论学者继续使用它并在此基础上建立置信区间(Kepler 和 Oprea,2001 年)。在这个习题中,你可以给出正确的推理。

在每次培养中,没有抗体的单个细菌在 g 代繁殖之后,会产生 $N=2^g$ 个后代,除了由于变异带来的不同,所有都是克隆的。然后让最后的培养基沾染一个病毒,并测量具有抗体的细菌数量 n。根据现在已经接受的变异假设,这些有抗性的细菌是在群体成长过程中通过随机变异获得抗体的。变异率(每个细胞每代)a 约为 1 亿个中出现一个。变异机会总数是 N,因为 $\sum_{i=0}^{g-1} 2^i \simeq 2^g = N$。如果一个细菌在第 i 代变异,它的后代都得到变异的遗传,则由一个这种细菌产生具有抗体的细菌的最终总数为 2^{g-i}。

已知 M 次单独实验,每次都产生大小为 N 的群体,其中测得有抗体细菌的数量为 $\{n_m\}_{m=1}^{M}$,如何推断变异率 a?

对下列来自 Luria 和 Delbrück 的数据集进行推导，对于 $N = 2.4 \times 10^8$：$\{n_m\} = \{1,0,3,0,0,5,0,5,0,6,107,0,0,0,1,0,0,64,0,35\}$。[求解这个问题需要少量的计算。]

35.3 推导因果关系



习题 35.5[2] 在贝叶斯图形模型的研究团体中，很多论文是关于推导箭头指向何方的任务——即哪些结点是父结点，哪些结点是子结点。

因为“似然等价性”，推导因果关系是需要技巧的。如果对于两组参数中的任何一组，都存在另一组参数使得两个关于所有观测量的联合概率分布是相同的，则两个图形模型是似然等价的。一对似然等价模型的例子是 $A \to B$ 和 $B \to A$。模型 $A \to B$ 表明 A 是 B 的父结点，或者用不严格的术语，“A 导致 B”。“$B \to A$”为真的情况的一个例子是：B 是变量“房子里有盗贼”和 A 是变量“报警铃声响了”。这里 B 导致 A 是完全正确的。但是如果用于另一个例子，这种表述会引起混淆，$R \to D$，其中 R 表示“今天早晨下雨了”，D 表示“人行道是干的”。“R 导致 D”就令人迷惑不解了。因此，笔者将使用单词“B 是 A 的父结点”来表明因果关系。某些只使用似然的统计方法没法使用数据来区分似然等价的模型。而另一方面，在贝叶斯方法中，因为似然等价性并不强行要求在模型的两个参数空间赋予等价的先验密度，所以针对数据，两个似然等价模型还是能够加以区分的。

但是，许多贝叶斯图形模型的拥护者，也许是出于对非贝叶斯学派的同情，也许是出于不想表现得与他们存在差异的潜在愿望，故意抛弃了贝叶斯方法的潜在优势——从数据推导因果关系的能力——通过歪曲他们的模型使得这种能力消失；一个普遍的正统观念认为，应该选择使“先验等价”条件成立的先验，即选择先验使得似然等价的模型仍然具有等价的后验概率，然后应该使用这些先验中的某一个进行推理和预测。这个论点促使人们选择特殊构造的 Dirichlet 分布作为所有概率向量的先验分布。

在笔者看来，仅仅使用这些先验在观念上是错误的，它将导致不能推导出因果关系。先验应被用来描述一个人的假设；做到这一点后，就能够从数据中推导令人感兴趣的因果关系。

在这个习题中，你将构造这种推导的一个例子。

考虑如下简化问题，其中 A 和 B 是二进制变量。两个模型是 $\mathcal{H}_{A\to B}$ 和 $\mathcal{H}_{B\to A}$。$\mathcal{H}_{A\to B}$ 断言 A 的边缘概率来自一个参数为(1,1)的 Beta 分布，即均匀分布；并且两个条件分布 $P(b|a=0)$ 和 $P(b|a=1)$ 也同样独立地来自参数为(1,1)的 Beta 分布。另一个模型将类似的先验赋予 B 的边缘概率及已知 B 时 A 的条件分布。收集数据，假设 $F = 1\,000$ 个结果，计数为

	$a=0$	$a=1$	
$b=0$	760	5	765
$b=1$	190	45	235
	950	50	

(35.4)

两个假设的后验概率是多少?

提示:用符号解答这个习题是一个好主意,以便找出它所显现出来的所有简化。

$$\Psi(x)=\frac{\mathrm{d}}{\mathrm{d}x}\ln\Gamma(x)\simeq\ln(x)-\frac{1}{2x}+O(1/x^2) \tag{35.5}$$

因果关系的推导是一个复杂的专题。贝叶斯推理可以理性地用于推导图中箭头方向的事实似乎是一个被忽视的观点,但是它的确不够全面。关于因果关系的很多其他方面的讨论,请参见 Pearl(2000 年)。

35.4 进一步的习题

习题 35.6[ㄕ] 到达光子监测器的光子发射过程被认为是具有以下时变速率的一个泊松过程。

$$\lambda(t)=\exp(a+b\sin(\omega t+\phi)) \tag{35.6}$$

其中参数 a,b,ω 和 ϕ 是已知的。在时间 $t=0,\cdots,T$ 上采集数据。已知在时间 $\{t_n\}_{n=1}^{N}$ 上到达的光子数为 N,讨论 a,b,ω 和 ϕ 的推理。[进一步的阅读:Gregory 和 Loredo(1992 年)。]

▷**习题 35.7**[ㄖ] 一个包括两列数字的数据文件,在打印出来时,列之间的边界不清晰。所得的字符串如下。

891.10.0	912.20.0	874.10.0	870.20.0	836.10.0	861.20.0
903.10.0	937.10.0	850.20.0	916.20.0	899.10.0	907.10.0
924.20.0	861.10.0	899.20.0	849.10.0	887.20.0	840.10.0
849.20.0	891.10.0	916.20.0	891.10.0	912.20.0	875.10.0
898.20.0	924.10.0	950.20.0	958.10.0	971.20.0	933.10.0
966.20.0	908.10.0	924.20.0	983.10.0	924.20.0	908.10.0
950.20.0	911.10.0	913.20.0	921.25.0	912.20.0	917.30.0
923.50.0					

已知这些数据,讨论每种解析为正确的可能性有多大。

(a) 891.10.0→891. 10.0 等。

(b) 891.10.0→891.1 0.0 等。

[一个字符串的解析是该字符串在语法上的一个解释。例如,"Punch bores"可被解析为"Punch(名词)bores(动词)"或"Punch(祈使语气的动词)bores(复数名

词)"。]

▷**习题 35.8**[2] 在一个实验中,被测量的量$\{x_n\}$独立地来自一个均值为μ的双边指数分布

$$P(x|\mu)=\frac{1}{Z}\exp(-|x-\mu|)$$

其中Z是归一化常数,$Z=2$。均值μ是未知的。这个分布在$\mu=1$时的一个例子如图 35.2 所示。

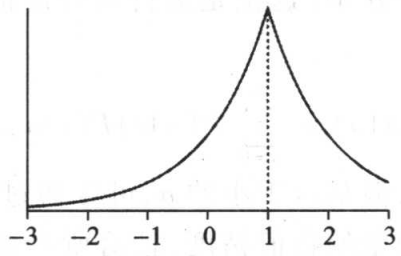
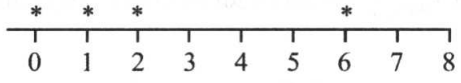


图 35.2 双边指数分布 $P(x|\mu=1)$

假设 4 个数据点为

$$\{x_n\}=\{0,0.9,2,6\},$$



这些数据传达了关于μ的什么信息?答案中要包括详细的示意图。给出μ的一系列可能值。

35.5 习题解答

习题 35.4 的解答 大小为N的种群有N个机会变异。变异次数的概率r大致服从泊松分布

$$P(r|a,N)=e^{-aN}\frac{(aN)^r}{r!} \tag{35.7}$$

(因为一个变异细菌的后代自身不能经历同样的变异,所以这稍微有些不准确。)每个变异细菌最后产生n_i个变异细菌,该数依赖于变异的生成时间。如果繁殖运作精准,则n_i为 1 的概率将是 1/2,为 2 的概率将是 1/4,为 4 的概率将是 1/8,且对于所有是 2 的幂的n_i,$P(n_i)=1/(2n)$。但是我们并不期望变异后裔会准确同步地分裂,而且与分裂次数相比,并不知道实验结束的准确定时。所有整数都可产生的这个分布的一个平滑版本是

$$P(n_i)=\frac{1}{Z}\frac{1}{n_i^2} \tag{35.8}$$

其中$Z=\pi^2/6=1.645$。[这个分布的矩都是错的,因为n_i永远不可能超过N,但是谁会关心矩呢?只是抽样理论统计学家搞错了对象,构建诸如$\hat{a}=\bar{n}/\ln(N)$之类的"无

偏倚估计器”。使用 $P(n_i)$ 的近似在似然函数中所引入的误差可被忽略。]

观测到的变异数 n 是和式

$$n = \sum_{i=1}^{r} n_i \tag{35.9}$$

在已知 r 的条件下，n 的概率分布是 r 个形如式(35.8)的相同分布的卷积。例如

$$P(n|r=2) = \sum_{n_1=1}^{n-1} \frac{1}{Z^2} \frac{1}{n_1^2} \frac{1}{(n-n_1)^2}, \quad n \geqslant 2 \tag{35.10}$$

在已知 a 的条件下，n 的概率分布(这正是进行贝叶斯推理所需要的)可用对 r 求和的方式给出。

$$P(n|a) = \sum_{r=0}^{N} P(n|r)P(r|a,N) \tag{35.11}$$

这个量不能以解析方式计算，但是对于小的 a，可以通过直接对 r 从 $r=0$ 到某个 r_{max} 值求和，容易计算至任何所期望的数值精度，且对所有要求的 n 值，还可通过显式卷积求出不超过 r_{max} 的每个 r 的 $P(n|r)$；如果 $r_{max} = n_{max}$(数据中所遇到的最大 n 值)，则可以精确地计算 $P(n|a)$；但是对于这个问题中的数据，$r_{max} = 9$ 就足以给出精确的结果；笔者使用 $r_{max} = 74$ 画出如图 35.3 所示的曲线。八进制信源码是可得到的。①

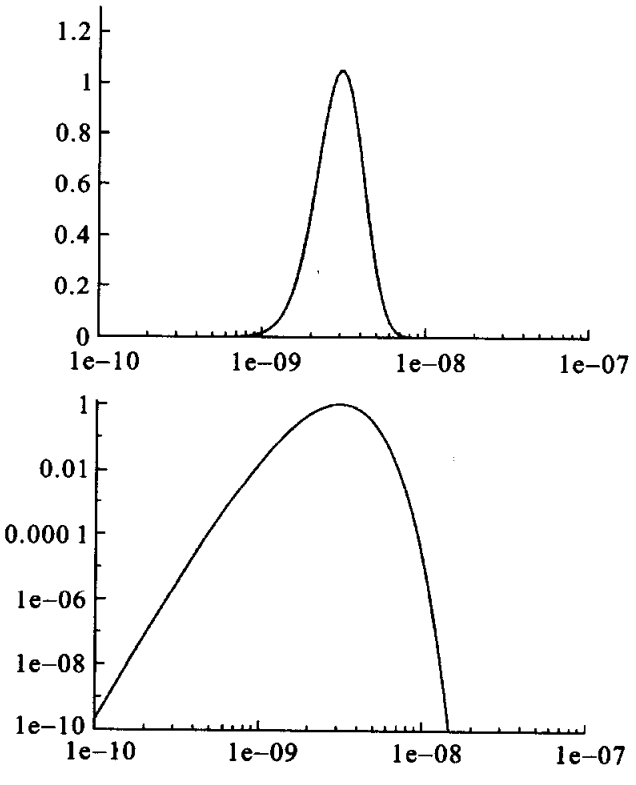


图 35.3 已知 Luria 和 Delbrück 的数据，变异速率 a 的似然，用线性坐标和对数坐标表示。纵轴：似然/10^{-23}；横轴：a。

① www.inference.phy.cam.ac.uk/itprnn/code/octave/luria0.m

顺便提及，像这个习题中的数据集，包含相当数量的零，在进行 Luria 和 Delbrück 的第二种近似时，其中只有很少几个被丢失，而该近似只保留关于多少 n 等于零和不为零的计数。使用这个减弱的数据集所建立起来的似然函数

$$L(a) = (e^{-aN})^{11}(1 - e^{-aN})^{9} \tag{35.12}$$

与使用全信息所计算的似然几乎没有差别。

习题 35.5 的解答 从下式中的 6 项

$$P(\boldsymbol{F} \mid \alpha \boldsymbol{m}) = \frac{\prod_i \Gamma(F_i + \alpha m_i)}{\Gamma(\sum_i F_i + \alpha)} \frac{\Gamma(\alpha)}{\prod_i \Gamma(\alpha m_i)} \tag{35.13}$$

约去很多因式，得到的是

$$\frac{P(\mathcal{H}_{A\to B} \mid \text{数据})}{P(\mathcal{H}_{B\to A} \mid \text{数据})} = \frac{(765+1)(235+1)}{(950+1)(50+1)} = \frac{3.8}{1} \tag{35.14}$$

存在适度的证据支持 $\mathcal{H}_{A\to B}$，因为从那个假设中推断出来的 3 个概率（约为 0.95、0.8 和 0.1）作为先验比从另一假设推断出来的 3 个概率（0.24、0.008 和 0.19）更为典型。如果认为先验在 3 个概率上是“均匀的”，那么这个说法听起来是荒谬的——在一个均匀先验的条件下，概率的任意设置都一定是等概率的？但是在自然基、对数单位基之下，先验与 $p(1-p)$ 成正比，而后验概率比可以通过下式来估计。

$$\frac{0.95 \times 0.05 \times 0.8 \times 0.2 \times 0.1 \times 0.9}{0.24 \times 0.76 \times 0.008 \times 0.992 \times 0.19 \times 0.81} \simeq \frac{3}{1} \tag{35.15}$$

它并不完全正确，但的确说明了对 $A \to B$ 的优先选择的依据所在。

第 36 章

决 策 论

决策论除了计算细节之外，就很平凡了（就像下棋一样）。

你可以选择各种动作 a。世界可能处于很多状态 $\boldsymbol{x}$ 中的一个；具体哪一个状态出现，也许会受你的动作的影响。世界的状态具有概率分布 $P(\boldsymbol{x}|a)$。最后，存在一个效用函数 $U(\boldsymbol{x},a)$，用来说明当世界处于状态 $\boldsymbol{x}$ 而且你选择了动作 a 时所得到的报酬。

决策论的任务是选择最大化如下期望效用的动作。

$$\mathcal{E}[U|\boldsymbol{a}] = \int \mathrm{d}^K \boldsymbol{x} U(\boldsymbol{x},a) P(\boldsymbol{x}|a) \tag{36.1}$$

就是这样的。计算问题是在 a 上最大化 $\mathcal{E}[U|a]$。[悲观主义者可能更喜欢定义一个丢失函数 L 而非一个效用函数 U，并最小化所期望的损失。]

那么关于决策论还有什么要说的吗？

在一个实际问题中，选择一个适当的效用函数也许非常困难。此外，当采取一系列行动时，且每个动作都提供关于 $\boldsymbol{x}$ 的信息，就必须考虑这个预期信息可能对后续动作产生影响。在一个决策问题中，所得的先验概率和后验概率计算的混合是很不一样的。对于一个实际问题，比如一个用棋盘玩的游戏，必做的思考及行为的树会变得非常巨大，而且要“做正确的事”也不容易，因为不能精确地计算出一个动作的期望效用（Russell 和 Wefald，1991 年；Baum 和 Smith，1993 年；Baum 和 Smith，1997 年）。

现在研究一个例子。

36.1 合理的勘探

假设你的任务是要确定一个 Tanzanite 矿点。你最后的动作将是从一序列 N 个点中选择一个点。第 n 个点具有净收益值 x_n，它在开始时是未知的，并且仅在选择点 n 之后才能得到其确切的值。[x_n 等于出售从那个点所得到的 Tanzanite 矿石的收入，减去购买这个矿点的费用，减去人工报酬，等等。]从一开始，基于已获得的信息，收益 x_n 具有概率分布 $P(x_n)$。

在你采取最后行动之前，你有机会去做一些勘探。在第 n 个点勘探的费用是 c_n，并产生数据 d_n，它会减小关于 x_n 的不确定性。[将假设 N 个点的收益是彼此无关的，在一个点处所进行的勘探仅产生关于那个点的信息，并不影响从那个点所获得的收

益。]

你的决策问题是：

已知初始的概率分布为 $P(x_1), P(x_2), \cdots, P(x_N)$，首先决定是否要进行勘探，在哪个点进行；然后根据你的勘探结果，选择开采哪个点。

为了简单起见，假设问题中的所有分布都是高斯的①，并集中考虑是要勘探一次还是不要勘探的问题。假设效用函数是 x_n 的线性函数；希望将期望收益最大化。如果不进行勘探，效用函数是

$$U = x_{n_a} \tag{36.2}$$

其中 n_a 是所选“动作”点；而如果做了勘探，则效用函数为

$$U = -c_{n_p} + x_{n_a} \tag{36.3}$$

其中 n_p 是进行勘探的点。

点 n 收益的先验分布是

$$P(x_n) = \text{Normal}(x_n; \mu_n, \sigma_n^2) \tag{36.4}$$

如果在点 n 处进行勘探，则数据 d_n 是 x_n 的一个噪声版本：

$$P(d_n | x_n) = \text{Normal}(d_n; x_n, \sigma^2) \tag{36.5}$$

▷ **习题 36.1**[2] 已知这些假设，证明 d_n 的先验概率分布是

$$P(d_n) = \text{Normal}(d_n; \mu_n, \sigma^2 + \sigma_n^2) \tag{36.6}$$

（提示：当独立变量相加时，方差相加），并且已知 d_n 条件下 x_n 的后验分布是

$$P(x_n | d_n) = \text{Normal}(x_n; \mu_n', \sigma_n^{2'}) \tag{36.7}$$

其中

$$\mu_n' = \frac{d_n/\sigma^2 + \mu_n/\sigma_n^2}{1/\sigma^2 + 1/\sigma_n^2} \text{ 和 } \frac{1}{\sigma_n^{2'}} = \frac{1}{\sigma^2} + \frac{1}{\sigma_n^2} \tag{36.8}$$

（提示：当高斯相乘时，精度相加）。

作为开始，计算不进行勘探时的期望效用值（即，立即选择点）；然后计算如下情形期望效用的值：先在一个点做一次勘探并做出选择。根据这两个结果，将决定是否勘探一次或零次，及如果勘探一次，应在哪个点。

所以，首先考虑不做任何勘探时的期望效用。



习题 36.2[2] 假设没有勘探，证明最优的动作是选择具有以下最大均值的点。

$$n_a = \underset{n}{\text{argmax}}\, \mu_n \tag{36.9}$$

而且这个动作的期望效用是

① 符号 $P(y) = \text{Normal}(y; \mu, \sigma^2)$ 表示，y 具有均值为 μ、方差为 σ^2 的高斯分布。

$$\mathscr{E}[U|\text{最优 } n] = \max_n \mu_n \tag{36.10}$$

[如果你的直觉是“最优的选择当然也要考虑不同不确定性的 σ_n?”,这个问题的答案是“合理的——如果这样,则效用函数是 x 的非线性函数。]

现在到了令人兴奋的时刻。是否要勘探? 一旦在一个点 n_p 完成勘探,将利用式(36.8)所给出的更新值 μ_n' 来代替均值 μ_{n_p},根据判决规则式(36.9)来选择点 n_p。使问题变得令人兴奋的是我们尚不知 d_n 的值,所以不知道动作 n_a 是什么;确实,做勘探的整个值都来源于这样一个事实:在缺少实验信息的情况下,结果 d_n 可能会改变所要采取的动作。

根据式(36.8)中由 d_n 所给出的新均值表达式,及 d_n 的已知方差(36.6),可以计算关键量 μ_n' 的概率分布,并通过对所有可能输出及其相关的动作求积分,计算出期望效用。



习题 36.3[2] 证明新均值 μ_n'(36.8)的概率分布是高斯的,其均值为 μ_n,方差为

$$s^2 \equiv \sigma_n^2 \frac{\sigma^2}{\sigma^2 + \sigma_n^2} \tag{36.11}$$

考虑在点 n 的勘探。设其他点的最大均值为 μ_1。当获得均值的新值 μ_n' 时,如果 $\mu_n' > \mu_1$,将选择点 n 并获得 μ_n' 的一个期望收益,如果 $\mu_n' < \mu_1$,将选择点 1 并获得 μ_1 的一个期望收益。

所以在点 n 勘探然后选择最佳点的期望效用是

$$\mathscr{E}[U|\text{在点 } n \text{ 勘探}] = -c_n + P(\mu_n' < \mu_1)\mu_1 + \int_{\mu_1}^{\infty} \mathrm{d}\mu_n' \mu_n' \mathrm{Normal}(\mu_n'; \mu_n, s^2) \tag{36.12}$$

勘探与不勘探的效用差别是人们所感兴趣的量,并且它取决于在没有勘探时所采取的动作;而那又取决于 μ_1 是否比 μ_n 大。

$$\mathscr{E}[U|\text{不进行勘探}] = \begin{cases} -\mu_1, & \mu_1 \geqslant \mu_n \\ -\mu_n, & \mu_1 \leqslant \mu_n \end{cases} \tag{36.13}$$

所以

$$\mathscr{E}[U|\text{在点 } n \text{ 勘探}] - \mathscr{E}[U|\text{不进行勘探}]$$

$$= \begin{cases} -c_n + \int_{\mu_1}^{\infty} \mathrm{d}\mu_n'(\mu_n' - \mu_1)\mathrm{Normal}(\mu_n'; \mu_n, s^2), & \mu_1 \geqslant \mu_n \\ -c_n + \int_{-\infty}^{\mu_1} \mathrm{d}\mu_n'(\mu_1 - \mu_n')\mathrm{Normal}(\mu_n'; \mu_n, s^2), & \mu_1 \leqslant \mu_n \end{cases} \tag{36.14}$$

可以画出随着勘探(省略 c_n)而改变的期望效用图,以差值 $(\mu_n - \mu_1)$(水平轴)和初始的标准差 σ_n(垂直轴)的函数表示。在图 36.1 中,噪声的方差为 $\sigma^2 = 1$。

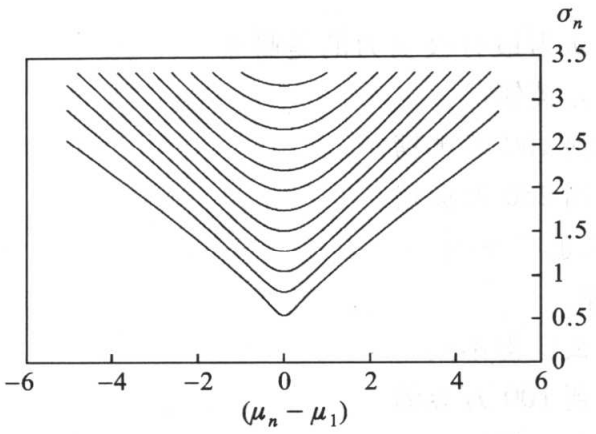


图 36.1 由于勘探而获得的期望效用增益的等势图。等势线从 0.1 到 1.2 以步长 0.1 等间隔分布。为了确定在点 n 是否值得勘探，找出等于 c_n 的等势线（勘探的费用）；等势线上的所有点 $[(\mu_n - \mu_1), \sigma_n]$ 都是值得勘探的。

36.2 进一步的阅读

如果我们采取动作的世界比勘探要略微复杂一些——例如，如果可以进行多重迭代的勘探，并且勘探费用是不确定的——那么寻找勘探和开采之间的最优平衡点变成一个困难得多的计算问题。*加强学习*讨论了这个问题的近似方法（Sutton 和 Barto，1998 年）。

36.3 进一步的习题

▷ **习题 36.4**[♂] **4 扇门问题**。

一个新的游戏秀使用与 3 扇门（习题 3.8）相似的规则，只是这里有 4 扇门，并且主持人解释说："首先，你指向其中的一扇门，然后我将打开另外的一扇，并保证它后面没有奖品。然后，你决定是否坚持最初的选择，还是切换到剩余的两扇门之一。接着我将打开另一扇没有奖品的门（但不是目前所选择的这扇）。然后，你将做出最后的选择，要么坚持你先前的决定，要么转向唯一剩余的一扇门。"

最优的策略是什么？你在第一次机会的时候应该切换吗？在第二次机会的时候应该切换吗？

▷ **习题 36.5**[♂] 决策论的一个难点是要准确地找出效用函数是什么。例如，对于大多数人来说，钱的效用是恶名昭著的非线性函数。

实际上，正如下述 Allias 悖论所说明的那样，很多人的行为不能用一个一致的效

用函数来刻画。

在这些选择中,你认为最有吸引力的是哪个?

A. 确保得到 100 万英镑。

B. 89% 的机会得到 100 万英镑。

10% 的机会得到 250 万英镑。

1% 的机会什么也得不到。

现在考虑这些选择:

C. 89% 的机会什么也得不到。

11% 的机会得到 100 万英镑。

D. 90% 的机会什么也得不到。

10% 的机会得到 250 万英镑。

很多人喜欢 A 胜过 B,同时,喜欢 D 胜过 C。证明这些偏爱与钱的任意效用函数 $U(x)$ 都是矛盾的。

习题 36.6[↗] **最优停止。**

N 名候选伴侣的队列在你的门口等候,都要向你求婚。他们以随机的顺序到达。当你与每位候选者见面时,你必须根据迄今为止的信息当即决定是与其结婚还是拒绝。每位候选伴侣都有一个理想程度 d_n,你可以在与他们见面时获知。你必须与他们之中的一位结婚,但是不允许你返回已经对其说"不"的人那里。

有几种方法来定义精确的问题。

(a) 假设你的目标是使理想程度 d_n 最大化,即你的效用函数是 $d_{\hat{n}}$,其中 $\hat{n}$ 是被选择的伴侣,你应采用何种策略?

(b) 假设你非常希望与一个最满意的人结婚(即,如果你做到这一点,你的效用函数是 1,否则是零);你应采用何种策略?

(c) 假设你非常希望与最理想的人结婚,并且你的策略是"策略 M":

策略 M——与前 M 个伴侣见面并对他们都说"不"。记住他们中的最大满意度 $d_{\max}$。然后依次与其他人见面,等待第一个满足 $d_n > d_{\max}$ 的伴侣的到来,并与之结婚。如果所到的人没有更满意的,则与最后第 N 位伴侣结婚(并感到痛苦)。

M 的最佳值是多少?

习题 36.7[↗] **将遗憾作为一个目标函数?**

上一道习题(b 和 c 部分)牵涉到一个基于遗憾的效用函数。如果一个人与第 10 满意候选人结婚,效用函数表明这个人会因为没机会选择最满意的而感到遗憾。

很多研究学习理论和决策论的人用"最小化最大可能的遗憾"作为一个目标函数,但是这样做合理吗?

设想弗雷德已经买了一张彩票,并在知道这张彩票是否中奖之前就卖给你。简单地说,彩票中奖的概率为 1/100,而且如果它中奖,它的价值为 10 英镑。弗雷德要以 1 英镑的价格卖给你。你会买吗?

可能的动作是"买"或"不买"。4 种可能动作的效用——在表 36.2 中给出了结

果对。笔者已经假设在小笔钱时，你的效用函数是线性的。如果你不购买彩票，不管彩票被证实是否中奖，其效用为零。如果你购买了彩票，要么以损失 1 英镑告终（概率为 99/100），要么赚到 9 英镑（概率为 1/100）。对于最小化最大可能遗憾的人们来说，所选择的动作是用来最小化最大可能的遗憾。4 个可能遗憾的结果如表 36.3 所示。如果你购买了彩票而且它没有中奖，你就有 1 英镑的一个遗憾，因为如果你没有购买它，你会赚 1 英镑。如果你没有购买彩票但是它中奖了，你会有 9 英镑的遗憾，因为如果你买了它，你将会赚 9 英镑。因此，最小化最大可能遗憾的行为是购买这张彩票。

表 36.2 彩票问题中的效用

	动作	
结果	购买	不买
未中奖	-1	0
中奖	+9	0

表 36.3 彩票问题中的遗憾

	动作	
结果	购买	不买
未中奖	1	0
中奖	0	9

讨论是否可在原理上证明这种使用遗憾来选择动作的做法是合理的。

上述问题可被转换为投资证券决策问题，设想你有 1 英镑在一天内投资两个可能的基金：弗雷德的彩票基金和现款基金。如果你将 f_1 英镑投入弗雷德的彩票基金，弗雷德承诺如果彩票中奖，他将返还你 $9f_1$ 英镑，否则什么都没有。剩余的 f_0 英镑（有 $f_0 = 1 - f_1$）作为现金。什么是最优的投资？证明最小化最大遗憾者会将他们 $f_1 = 9/10$ 的钱投资在高风险、高回报的彩票基金，仅将 $f_0 = 1/10$ 用于现金。可以证明这种投资方法合理吗？

习题 36.8[$\mathcal{F}$] **赌博怪僻** （来自 Cover 和 Thomas（1991 年））。一次赛马包括反复出现的 I 匹马，每次你都必须将所有钱用来下注。你在时刻 t 的赌注可以用一个归一化概率向量 $\boldsymbol{b}$ 与你的钱财 $m(t)$ 相乘来表示。赌注者所提供的几率是这样的：如果马 i 获胜，那么你获利 $m(t+1) = b_i o_i m(t)$。假设赌注者的几率是“公平的”，即

$$\sum_i \frac{1}{o_i} = 1 \tag{36.15}$$

并假设马 i 获胜的概率是 p_i，如果你的目标是 Cover 目标，设计出最优下注策略，即，最大化 $\log m(T)$ 的期望值。证明最优策略是设 $\boldsymbol{b}$ 等于 $\boldsymbol{p}$，与庄家的几率 $\boldsymbol{o}$ 无关。证明当使用这个策略时，期望钱以指数增长为

$$2^{nW(\boldsymbol{b},\boldsymbol{p})} \tag{36.16}$$

其中 $W = \sum_i p_i \log b_i o_i$。

如果你仅赌博一次，最优策略有什么不同吗？

你认为这个最优策略合理吗？按照通常的说法，你认为忽略庄家的几率是“最优的”吗？你对“Cover 目标”能得出什么结论？

习题 36.9[$\mathcal{F}$] 两个普通的骰子被反复地投掷；每次投掷的结果是两个数字之

和。Joe Shark 说 6 和 8 是他的幸运数字，并下同额赌注赌定 6 将会在第一个 7 被掷出之前被掷出。如果你是一个赌徒，你会下注吗？你获胜的概率是多少？Joe 然后下同额赌注赌定 8 将会在第一个 7 被掷出之前被掷出。你会下注吗？

赢得你的自信心之后，Joe 建议将两次赌博合并成一次赌博：他以一大笔钱，仍然下同额赌注，赌一个 8 和一个 6 会在掷出两个 7 之前出现。你会下注吗？你获胜的概率是多少？

第 37 章

贝叶斯推理与抽样理论

统计学上有两个学派。在已知最少假设的条件下，抽样理论者集中考虑确保在大多数时间有效的方法。而贝叶斯派学者则考虑在所有可获得的信息下进行推理，并在已知某个特定数据集的条件下回答感兴趣的问题。正如你可能已推测的，笔者强烈建议使用贝叶斯方法。

抽样理论是在统计学上广泛使用的方法，并且大多数刊物上的多数论文中都报道了他们的实验，其中使用了诸如置信区间、显著性水平和 p 值之类的量。一个 p 值（例如 $p=0.05$）是在已知数据概率分布的一个虚假设时，与观测到的结果同样极端或比它更极端的结果的概率。未经训练的读者（更令人担忧的是，可能还有很多论文的作者）经常将这样的一个 p 值当做一个贝叶斯概率来解释（例如，虚假设的后验概率）；对此，抽样理论派学者和贝叶斯派学者都认为是不正确的。

本章研究几个简单的推理问题，以比较统计学的这两种方法。

尽管在某些情况下，从贝叶斯方法和抽样理论所得的答案非常近似，还是可以找到存在很大差异的情况。已经在习题 3.15 中见过这样一个例子，其中一位抽样理论派学者得到一个小于 7% 的 p 值，并将其视为一个有力的证据来反驳虚假设，但是数据实际上更倾向于虚假设而不是另一个简单的备选方案。在第 3 章最后还给出了另一个例子，其中 p 值比神秘值 5% 还要小，然而数据又一次倾向于虚假设。因此，在某些情况下，抽样理论可能是乱开枪的：当这些结果实际上微弱地支持虚假设时，却宣称结果是“非常不可能的，应该拒绝虚假设”。正如大家将要看到的，还存在推理问题，当一个贝叶斯方法和通常的直觉都认为证据有力的时候，抽样理论却不能检测到“显著的”证据。最有说服力的是推理问题，其中抽样理论所分配的“显著性”会随着与试验设计有关的不相干因素而变化。

这一章只为那些对抽样理论/贝叶斯方法争论好奇的读者而作。如果你发现这一章中的任何部分难以理解，请跳过它。试图弄明白这个争论没什么意义。只要使用贝叶斯方法就行，它们比争论本身更容易理解！

37.1 一个医学的例子

我们正在努力减小一种称为*微软症*的令人不愉快的疾病的影响。在一群志愿者身上试验两种接种疫苗 A 和 B。疫苗 B 是一种控制治疗，它是一种没有有效成分的安慰剂对照治疗。在 40 个人中，随机地指定 30 个人接受 A 治疗，另外 10 个人进行

对照治疗 B。在他们接种疫苗后,对他们做一年的观察。在 A 组的 30 个人中,一人感染了微软症。在 B 组的 10 个人中,3 人感染了微软症。

治疗 A 比治疗 B 更好吗?

尝试抽样理论

对于问题"A 是否比 B 好?",标准的抽样理论方法是构建一个统计测试。测试通常将一个假设,如

$\mathcal{H}_1$:"A 和 B 具有不同的有效性"。

与一个虚假设,如

$\mathcal{H}_0$:"A 和 B 彼此之间具有完全一样的有效性"。

进行比较。一个新手也许会反对说"不,不,我要比较假设'A 比 B 好'与另一个假设'B 比 A 好'!"但是这种异议在抽样理论中并不受欢迎。

一旦定义好了两个假设,第一个假设就很少再次被提及——注意力仅集中在虚假设上。写到这里我不禁笑了,但这是真的! 纯粹基于数据是如何出乎 $\mathcal{H}_0$ 的意料,来决定接受还是拒绝虚假设,而不是基于 $\mathcal{H}_1$ 对数据的预测会有多好。人们选择一个统计量来度量一个数据集偏离虚假设有多远。在这里的例子中,所使用的标准统计量是被称为 χ^2(读为"chi 方")的量。为了计算 χ^2,求出每个数据测量值与假设虚假设为真时它的期望值的差值,并用假设虚假设为真时测量值的方差去除那个差值的平方。在目前的问题中,4 个数据测量值是整数 F_{A+}、F_{A-}、F_{B+} 和 F_{B-},即,进行治疗 A 且感染微软症的人的数量(F_{A+})、进行治疗 A 且没有感染微软症的人的数量(F_{A-}),依此类推。χ^2 的定义是

$$\chi^2 = \sum_i \frac{(F_i - \langle F_i \rangle)^2}{\langle F_i \rangle} \tag{37.1}$$

实际上,在我的基础统计学书籍(Spiegel,1988 年)中,我发现 Yates 的校正是值得推荐的①:

$$\chi^2 = \sum_i \frac{(|F_i - \langle F_i \rangle| - 0.5)^2}{\langle F_i \rangle} \tag{37.2}$$

在这种情况下,已知虚假设(治疗 A 和治疗 B 同样有效),并且两个结果的出现率分别为 f_+ 和 f_-,所期望的计数为

$$\begin{aligned} \langle F_{A+} \rangle &= f_+ N_A \quad \langle F_{A-} \rangle = f_- N_A \\ \langle F_{B+} \rangle &= f_+ N_B \quad \langle F_{B-} \rangle = f_- N_B \end{aligned} \tag{37.3}$$

根据 χ^2 有多大,测试接受或拒绝虚假设。为了使测试精确,并赋予它一个"显著性水平",必须计算出 χ^2 的抽样分布,考虑到 4 个数据点不独立的事实(它们满足两个约

① 如果你想了解关于 Yates 的校正,可以阅读一本抽样理论的教材。这一章的重点不是讲授抽样理论;我提及 Yates 的校正是因为这是一个专业的抽样理论派学者所可能使用的。

束 $F_{A+}+F_{A-}=N_A$ 和 $F_{B+}+F_{B-}=N_B$）以及参数 $f_\pm$ 未知的事实。这 3 个约束条件将数据的自由度从 4 降为 1。[如果你想学习关于“自由度”计算的更多知识，阅读一本抽样理论的书；在贝叶斯方法中，对此可以完全无需了解，当需要时，与自由度等效的量可以从一个贝叶斯分析中直接跳出来。]这些抽样分布被抽样理论矮人排列成表格形式，同时伴以它们在某些条件下正确的警告。例如，仅在期望数 F_i 为 5 或更大时，χ^2 的标准列表分布才是正确的。

一旦得到数据，抽样理论派学者就从以下数据中估计虚假设的未知参数 $f_\pm$。

$$\hat{f}_+=\frac{F_{A+}+F_{B+}}{N_A+N_B},\quad \hat{f}_-=\frac{F_{A-}+F_{B-}}{N_A+N_B} \tag{37.4}$$

并计算 χ^2。从这一点开始，抽样理论学派将其自身分为两个阵营。一个阵营使用下面的协议：首先，在查看数据之前，先选取测试的显著性水平（例如 5%），然后确定 χ^2 的临界值，大于这个值的虚假设就会遭拒。（如果虚假设是真的，这个显著性水平是统计量 χ^2 超过临界值的比例。）然后计算 χ^2，与临界值进行比较，并宣布测试结果及其显著性水平（它是事先固定的）。

第二个阵营先查询数据，寻找 χ^2，然后在 χ^2 分布表中寻找显著性水平 p，在该值上，χ^2 的观测值将是临界值。然后，通过给出这个 p 值来公布测试结果，这个 p 值是一个结果与观测值同样极端或比它更为极端的比例，如果虚假设为真，就能预期其发生。

现在应用这两种方法。第一阵营：选择 5% 作为显著性水平。具有一个自由度的 χ^2 的临界值是 $\chi^2_{0.05}=3.84$。$f_\pm$ 的估计值是

$$f_+=1/10,\quad f_-=9/10 \tag{37.5}$$

4 个测量值的期望值是

$$\langle F_{A+}\rangle=3 \tag{37.6}$$

$$\langle F_{A-}\rangle=27 \tag{37.7}$$

$$\langle F_{B+}\rangle=1 \tag{37.8}$$

$$\langle F_{B-}\rangle=9 \tag{37.9}$$

并且 χ^2（在式(37.1)中定义）是

$$\chi^2=5.93 \tag{37.10}$$

因为这个值超过 3.84，我们就在 0.05 显著性水平上拒绝“两种治疗等价”的虚假设。但是，如果使用 Yates 的校正，可以得到 $\chi^2=3.33$，因此接受这一虚假设。

在任何好的抽样理论书籍后面，都可以找到 χ^2 表。第二阵营的人用手指在 χ^2 表上移动，可找到 $\chi^2_{0.10}=2.71$。在 $\chi^2_{0.10}$ 和 $\chi^2_{0.05}$ 之间插值，第二阵营的人报告说“p 值是 $p=0.07$”。

注意：这个答案并不是说 A 比 B 更有效多少，它只是简单地说明 A 与 B 有“显著的”不同。并且这里，“显著的”仅意味着“统计上显著的”，而非实际上显著的。

普通人在读到“治疗与控制（$p=0.07$）有显著的不同”的语句时，可能会得出结论“这两种治疗的有效性存在 93% 的几率是不同的”。但是，“$p=0.07$”实际上意味

着“如果你做这个实验多次,并且两种治疗具有同等的有效性,则在7%的时间,你将发现χ^2的值比发生在这里的值更极端”。这几乎与我们想要知道的(即治疗A比B更好的可能性有多大)没有任何关系。

让我过去,我是一名贝叶斯派学者

好,现在来推导真正想知道的。丢弃“两种治疗方法的有效性完全相等”的假设,因为我们不相信它。存在两个未知参数p_{A+}和p_{B+},它们分别是对人给予两种治疗A和B并感染疾病的概率。

已知数据,可以推导出这两个概率,并且可以通过检查后验分布来回答令人感兴趣的问题。

后验分布为

$$P(p_{A+},p_{B+}|\{F_i\})=\frac{P(\{F_i\}|p_{A+},p_{B+})P(p_{A+},p_{B+})}{P(\{F_i\})} \tag{37.11}$$

似然函数为

$$P(\{F_i\}|p_{A+},p_{B+})=\binom{N_A}{F_{A+}}p_{A+}^{F_{A+}}p_{A-}^{F_{A-}}\binom{N_B}{F_{B+}}p_{B+}^{F_{B+}}p_{B-}^{F_{B-}} \tag{37.12}$$

$$=\binom{30}{1}p_{A+}^{1}p_{A-}^{29}\binom{10}{3}p_{B+}^{3}p_{B-}^{7} \tag{37.13}$$

应该使用什么先验分布?先验分布让我们有机会使用从其他实验得到的信息,或一个先验置信:尽管两个参数p_{A+}和p_{B+}彼此不同,但是预计它们会取类似的值。

这里将使用最简单的缺乏想象力的先验分布:在每个参数上的均匀分布。

$$P(p_{A+},p_{B+})=1 \tag{37.14}$$

现在可以画出后验分布图。假设p_{A+}和p_{B+}有一个可分离的先验,则后验分布也是可分离的:

$$P(p_{A+},p_{B+}|\{F_i\})=P(p_{A+}|F_{A+},F_{A-})P(p_{B+}|F_{B+},F_{B-}) \tag{37.15}$$

两个后验分布如图37.1所示(除了图未被归一化之外),联合后验概率如图37.2所示。

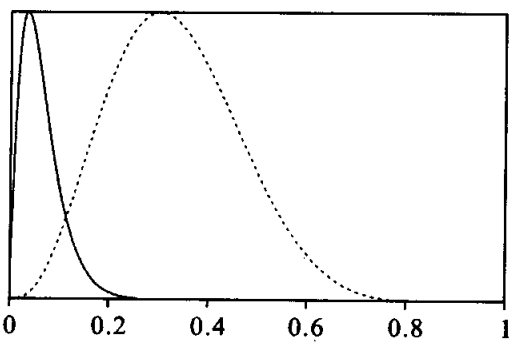


图37.1 两个有效性的后验概率。治疗A——实线;治疗B——虚线。

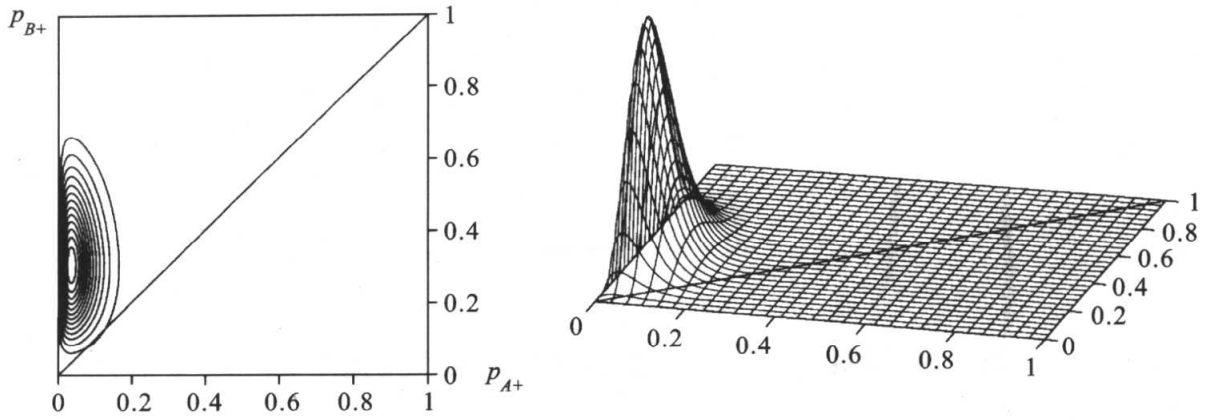


图 37.2 两个有效性的联合后验概率——等势图和曲面图

如果想知道问题“p_{A+} 小于 p_{B+} 的可能性有多大?”的答案,可以通过计算以下后验概率来精确回答。

$$P(p_{A+} < p_{B+} \mid \text{数据}) \tag{37.16}$$

它是联合后验概率 $P(p_{A+}, p_{B+} \mid \text{数据})$ 的积分,在图 37.2 中 $p_{A+} < p_{B+}$ 区域上,即图 37.3 中的阴影三角形。这个积分的值(通过对似然函数(37.13)在相关区域进行直接的数值积分可得)是 $P(p_{A+}, p_{B+} \mid \text{数据}) = 0.990$。

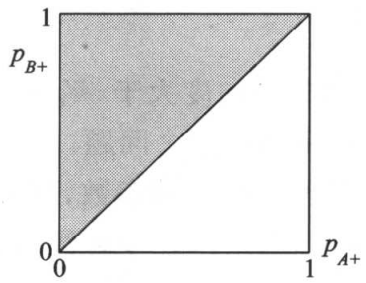


图 37.3 对阴影三角形中的所有点,命题 $p_{A+} < p_{B+}$ 都是真的。为了求出这个命题的概率,将在这个区域上对联合后验概率 $P(p_{A+}, p_{B+} \mid \text{数据})$(图 37.2)进行积分。

因此,已知数据和先验的假设,治疗 A 比治疗 B 要好的几率为 99%。总之,根据贝叶斯模型,数据(在疫苗 A 接种之后,30 个人中有 1 人感染疾病;在疫苗 B 接种之后,10 个人中有 3 人感染疾病)给出了强有力的证据——约为 99:1——治疗 A 要优于治疗 B。

在贝叶斯方法中,也容易回答其他相关的问题。例如,如果想知道“治疗 A 的有效性是治疗 B 的 10 倍的可能性有多大?”,可以将联合后验概率 $P(p_{A+}, p_{B+} \mid \text{数据})$ 在 $p_{A+} < 10p_{B+}$ 的区域上(如图 37.4 所示)进行积分得到。

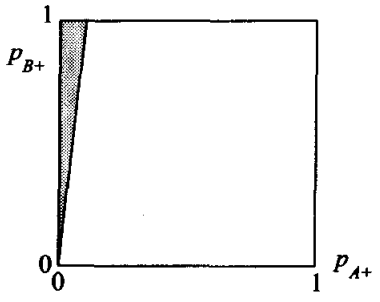


图 37.4 命题 $p_{A+} < 10p_{B+}$ 在阴影三角形中的所有点都是真的。

模型比较

如果存在这样一种情况:的确想要比较两个假设 $\mathcal{H}_0: p_{A+} = p_{B+}$ 和 $\mathcal{H}_1: p_{A+} \neq p_{B+}$,当然也可以用贝叶斯方法直接来做。

作为一个例子,考虑以下数据集。

D:一个受治疗者,给予治疗 A,然后感染微软症。

一个受治疗者,给予治疗 B,未被感染。

治疗	A	B
感染疾病	1	0
未感染疾病	0	1
治疗总数	1	1

在多大程度上,这个数据集对 $\mathcal{H}_1$ 的倾向度大于 $\mathcal{H}_0$?

通过计算每个假设的显著性来回答这个问题。假设模型的未知参数具有均匀先验。第一个假设 $\mathcal{H}_0: p_{A+} = p_{B+}$ 只有一个未知参数,称其为 p。

$$P(p|\mathcal{H}_0) = 1, \quad p \in (0,1) \tag{37.17}$$

将使用先前用过的模型 $\mathcal{H}_1$ 的两个参数的均匀先验

$$P(p_{A+}, p_{B+}|\mathcal{H}_1) = 1, \quad p_{A+} \in (0,1), p_{B+} \in (0,1) \tag{37.18}$$

现在,模型 $\mathcal{H}_0$ 下的数据 D 的概率等于已知 D 条件下,由 p 的推理所得到的归一化常数。

$$P(D|\mathcal{H}_0) = \int \mathrm{d}p P(D|p) P(p|\mathcal{H}_0) \tag{37.19}$$

$$= \int \mathrm{d}p p(1-p) \times 1 \tag{37.20}$$

$$= 1/6 \tag{37.21}$$

模型 $\mathcal{H}_1$ 下的数据 D 的概率由一个简单的二重积分给出。

$$P(D|\mathcal{H}_1) = \int\int \mathrm{d}p_{A+} \mathrm{d}p_{B+} P(D|p_{A+}, p_{B+}) P(p_{A+}, p_{B+}|\mathcal{H}_1) \tag{37.22}$$

$$= \int \mathrm{d}p_{A+} p_{A+} \int \mathrm{d}p_{B+} (1 - p_{B+}) \tag{37.23}$$

$$= 1/2 \times 1/2 \tag{37.24}$$

$$= 1/4 \tag{37.25}$$

因此，倾向于模型 $\mathcal{H}_1$ 的显著性之比（它表明了两个有效性不相等）是

$$\frac{P(D|\mathcal{H}_1)}{P(D|\mathcal{H}_0)} = \frac{1/4}{1/6} = \frac{0.6}{0.4} \tag{37.26}$$

所以如果两个假设的先验概率是 50:50，那么倾向于 $\mathcal{H}_1$ 的后验概率是 60:40。 □

使用贝叶斯方法不易得到适定问题的合理答案吗？

[抽样理论对这个问题的回答可能会涉及前面问题中所用的相同显著性测试；并且测试会产生一个“不显著”的结果。笔者认为这大大有利于对直觉认为显而易见的事物的认可，也就是数据 D 的确给出了*虚弱的*证据来支持 $\mathcal{H}_1$。贝叶斯方法量化了证据的虚弱程度。]

37.2 p 值对于无关信息的依赖性

在一个昂贵的实验室中，Bloggs 博士抛掷一枚标有 a 和 b 的硬币 12 次，所得结果为字符串

$$aaabaaaabaab$$

其中包括 3 个 b 和 9 个 a。

这些数据给出怎样的证据来支持“硬币存在倾向于 a 的偏倚”？

Bloggs 博士向他的抽样理论朋友请教。他的朋友说“设 r 为 b 的个数，$n = 12$ 是抛掷总数。我将 r 看做一个随机变量，并假定虚假设 $p_a = 0.5$ 为真，求当 r 取值为 $r = 3$ 或一个更极端值的概率”。因此他计算

$$P(r \leqslant 3 | n = 12, \mathcal{H}_0) = \sum_{r=0}^{3} \binom{n}{r} 1/2^n = \left(\binom{12}{0} + \binom{12}{1} + \binom{12}{2} + \binom{12}{3} \right) 1/2^{12}$$

$$= 0.07 \tag{37.27}$$

并报告说“在显著性水平为 5% 时，并不存在明显的偏倚证据倾向于 a”。或者，如果这位朋友更喜欢报告 p 值而非仅用 5% 与 p 进行比较，他会报告说“p 值是 7%，它并不像通常所认为的那样小”。如果一个双尾检测看起来更合适，他也许会计算双尾面积，它是上述概率的两倍，并报告说“p 值是 15%，它不是很小。”我们不会纠缠在单尾检测还是双尾检测的选择问题上，因为还有更重要的事情要做。

Bloggs 博士仔细地考察了计算式(37.27)，并回答说“不，不，实验中的随机变量不是 r；在做实验前我就决定将一直抛掷硬币直至看到 3 个 b；因此随机变量是 n”。

这样的实验设计并不少见。在关于纠错码的实验中，笔者通常模拟一个码的译码直到一个被选择的分组错误数 r（即这样多个 b）出现，因为 $\log p_b$ 的推理值的误差约为 $\sqrt{r}$，与 n 无关。



习题 37.1[2] 求出已知数据时关于硬币偏差 p_a 的贝叶斯推理，并确定一个贝叶斯推理是否依赖于所施行的停止规则。

根据抽样理论，为了估计结果 $n=12$ 的“显著性”，要求进行一个不同的计算。已知 $\mathcal{H}_0$ 条件下，n 的概率分布是前 $n-1$ 次抛掷刚好包含 $r-1$ 个 b 且第 n 次抛掷是一个 b 的概率。

$$P(n|\mathcal{H}_0,r)=\binom{n-1}{r-1}1/2^n \tag{37.28}$$

抽样理论派学者因此计算

$$P(n\geq 12|r=3,\mathcal{H}_0)=0.03 \tag{37.29}$$

他向 Bloggs 博士报告，“p 值是 3%——毕竟还是有明显的偏倚证据！”

你认为 Bloggs 博士该怎样做？他应该将这个具有绝妙 p 值的结果发表在某一个坚持认为所有实验结果都使用抽样理论来评估其“显著性”的刊物上吗？抑或将抽样理论派学者赶出门去，并寻找一个一致的、不依赖于停止规则的“显著性”评估方法？

在这一点上，听众分为两派。一半的听众从直觉上感到停止规则是无关的，并且可以心悦诚服地认为习题 37.1 的答案是“关于 p_a 的推理不依赖于停止规则”。另外一半，也许缘于抽样理论的系统训练，直观地认为 Bloggs 博士的停止规则（即出现第 3 个 b 时就停止的规则）也许以某种方式使实验产生了偏差。如果你在第二组，我鼓励你对整个事件再考虑一下，也希望你将最终改变主意并接受与似然原理相一致的观点，即停止规则与已获得的关于 p_a 的知识是毫不相干的。

作为一个思维实验，考虑某些偷窥者（为了省钱）在暗中监视 Bloggs 博士的实验：每当他抛掷硬币，窥探者就更新 r 和 n 的值。只要每次新的结果出现，窥探者就急切地根据数据做出推理。窥探者关于硬币偏差的置信度应取决于 Bloggs 博士关于是否继续实验的意图吗？

抽样理论的 p 值确实依赖于停止规则这一事实（事实上，当要求一个复杂的停止规则（例如，“系列概率比测试”）时，成册的抽样理论文献都在关注评估“显著性”的工作。）在笔者看来本身就是一个很有说服力的论据，证明它与 p 值根本就毫无关系。这个推理问题的一个贝叶斯解答在第 3.2 节和第 3.3 节以及习题 3.15 中已给出。

如果笔者为故事再增加一幕场景，会有助于澄清这个问题吗？一个看门人在偷听 Bloggs 博士的谈话内容，走进来并且说“我恰好注意到在你刚停止做硬币实验后，“荒诞部”官员就命令马上摧毁所有这样的硬币。因此，你的硬币被该部门的安全官员毁了。你无法继续实验，使之超过 $n=12$ 次抛掷了。在我看来，你也许该重新计算你的 p 值了吧？”

37.3 置信区间

在一个实验中，数据 D 来自一个具有未知参数 θ 的系统，抽样理论的一个标准概念是 θ 的置信区间。这样一个区间 $(\theta_{\min}(D),\theta_{\max}(D))$ 具有一个与之相关联的置信水平（比如95%），可以非正式地将它解释为“θ 位于置信区间内的概率”。

现在来精确地描述置信水平的真正含义，然后给出一个例子。一个置信区间是数据集 D 的一个函数 $(\theta_{\min}(D),\theta_{\max}(D))$。置信区间的置信水平是可以在数据到达之前就计算出来的性质。设想从 θ 的某个特定真值生成很多数据集，并计算区间 $(\theta_{\min}(D),\theta_{\max}(D))$，然后验证 θ 的真值是否位于该区间。如果对所有这些设想的实验进行重复并求平均值，则 θ 的真值位于置信区间内的比例是 f，而且这个性质对 θ 的所有真值都成立，则该置信区间的置信水平是 f。

例如，如果 θ 是一个标准差为1的高斯分布的均值，且 D 是来自该高斯的一个样本，那么对于 θ 而言，$(\theta_{\min}(D),\theta_{\max}(D)) = (D-2,D+2)$ 是一个95%的置信区间。

现在来看一个简单的例子，其中置信水平的意义变得更为清晰。设参数 θ 为一个整数，并设数据是一对点 x_1,x_2，从下面的分布中独立地抽样：

$$P(x|\theta)=\begin{cases}1/2, & x=\theta\\ 1/2, & x=\theta+1\\ 0, & \text{其他}\end{cases} \tag{37.30}$$

例如，如果 θ 是39，那么可以期望得到下列数据集

$$\begin{aligned}D=(x_1,x_2)&=(39,39)\text{，概率为 }1/4\\ (x_1,x_2)&=(39,40)\text{，概率为 }1/4\\ (x_1,x_2)&=(40,39)\text{，概率为 }1/4\\ (x_1,x_2)&=(40,40)\text{，概率为 }1/4\end{aligned} \tag{37.31}$$

现在考虑下列置信区间

$$[\theta_{\min}(D),\theta_{\max}(D)]=[\min(x_1,x_2),\min(x_1,x_2)] \tag{37.32}$$

例如，如果 $(x_1,x_2)=(40,39)$，则 θ 的置信区间应该是 $[\theta_{\min}(D),\theta_{\max}(D)]=[39,39]$。

考虑这个置信区间。它的置信水平是多少？通过考虑式(37.31)所示的4种可能性，可以发现置信区间包含真值的几率为75%。因此根据定义，置信区间具有75%的置信水平。

现在，如果所得到的数据是 $(x_1,x_2)=(29,29)$，又会怎样？好，可以计算置信区间，并且它是[29,29]。那么是否应该报告这个区间及其相关的置信水平75%？根据抽样理论的规则，这个结果会是正确的。但是这有意义吗？在这种情况下，我们真正了解到什么？直观地，或通过贝叶斯定理可以清楚地知道，θ 要么是29，要么是

28,并且两种概率值是等可能的(如果28和29的先验概率是相等的)。θ是29的后验概率是50%,是28的后验概率也是50%。

如果数据是$(x_1, x_2) = (29, 30)$,又会怎样?在这种情况下,置信区间仍然是[29,29],并且与之相关的置信水平还是75%。但是在这种情况下,根据贝叶斯定理或常识,可以100%地确信θ是29。

在这两种情况下,θ位于"75%的置信区间"的概率都不等于75%!

因此

1. 很多人关于抽样理论置信区间的解释方法是*不正确的*。

2. 已知某些数据,通常人们想要了解的(无论他们知道与否)是一个贝叶斯后验概率分布。

这些实验都是精心策划的吗?我是无事生非吗?如果你对我所表述的教条观点表示怀疑,我鼓励你去看一个案例研究:仔细研究习题35.4和参考文献(Kepler和Oprea,2001年),其中构建了抽样理论估计和一个突变率的置信区间。将两种方法都用在仿真数据上——仅基于似然函数计算的贝叶斯方法,及来自抽样理论的置信区间;如果你不觉得贝叶斯答案总是优于抽样理论的答案,并且做出通常要好很多这个结论,请让我知道。抽样理论花费很大的努力才能达到次优性,这就是为什么我热衷于贝叶斯方法的原因。贝叶斯方法是简单直白的,并且它们最优地使用数据中的所有信息。

37.4 某些折中的立场

可以用一个调解性的注解来结束讨论。很多抽样理论派学者是注重实效的——他们喜欢从统计方法集合中做出选择,选择具有"最佳"长效性质的某个东西。与之相对,我对一个适定问题仅存在一种答案的观点毫无疑问;但是并非一定要让抽样理论派学者也接受这个观点:作为替代,可以向他们提供贝叶斯估计器和贝叶斯置信区间,并要求评估这些方法的抽样理论性质。我们不需要提及方法是来自一个贝叶斯观点。如果抽样性质很好,那么注重实效的抽样理论派学者会选择使用贝叶斯方法。很多贝叶斯方法确实具有良好的抽样理论性质。也许为每个单独实例都提供最佳答案的一种方法从长远看也应该是好的,这并不令人吃惊!

也应承认另外一个共同点:尽管我相信大多数适定推理问题都只有唯一的正确答案,它可以通过贝叶斯方法得到,但并非所有问题都是适定的。在数据建模中经常出现的问题是"我正在使用的是一个合适的模型吗?"模型批评主义,即找出当前模型的缺陷,是抽样理论测试可以提供帮助的一项任务;在测试中,虚假设("当前模型是正确的")有很好的定义,但是没有指定其他可选模型。可以使用抽样理论来测量诸如p值之类,以指导人们查找模型中最需仔细审查的方面。

进一步的阅读

关于这个主题，笔者最喜欢的读物包括（Jaynes，1983 年；Gull，1988 年；Loredo，1990 年；Berger，1985 年；Jaynes，2003 年）。来自于统计学界的关于贝叶斯统计学的专著包括（Box 和 Tiao，1973 年；O'Hagan，1994 年）。

37.5 进一步的习题

▷ **习题 37.2**[JC] 一次交通调查记录了连续两天的交通情况。在星期五早晨，一个小时有 12 辆车。在星期六早晨，半个小时有 9 辆车。假设车辆服从泊松分布，到达率（每小时的车辆数）分别为 λ_F 和 λ_S。

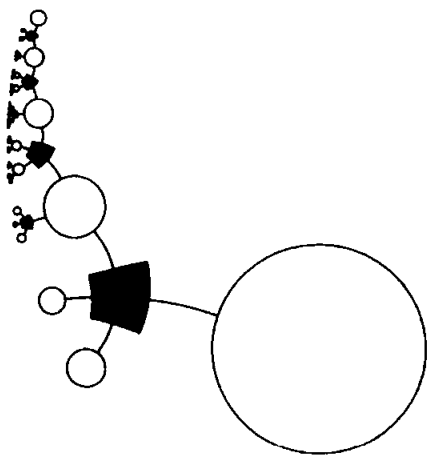
（a）λ_S 比 λ_F 大吗？

（b）是什么因素使得 λ_S 大于或小于 λ_F？

▷ **习题 37.3**[JC] 已知第 37.1 节所描述的数据 F_{A+}、F_{A-}、F_{B+} 和 F_{B-}，写一段程序对治疗 A 和 B 进行比较。程序的结果应该是（a）治疗 A 比治疗 B 更有效的概率；（b）$p_{A+} < 10p_{B+}$ 的概率；（c）$p_{B+} < 10p_{A+}$ 的概率。

第五部分

神经网络



第 38 章

神经网络引言

在神经网络领域中，我们研究理想化“神经元”网络的特性。

三个方面的因素推动了这个内涵丰富的交叉学科领域的研究工作。

生物学 理解大脑如何工作是科学界尚未解决的重要问题之一。一些神经网络模型试图揭示大脑是如何进行计算和记忆的。

工程 许多研究人员试图构造出能够“学习”、实现“模式识别”或“发现数据模式”的机器。

复杂系统 对神经网络感兴趣的第三个原因在于，它是一个复杂的自适应系统，本身就具有有趣的特性。

从一开始笔者就应该强调几点。

- 本书对该领域只是稍作涉猎。有许多有趣的神经网络模型我们都无暇论及。
- 我们并不想要求所讨论的模型是忠实于生物系统的模型。如果它们与生物系统存在任何关联，它们的相关性也只是在抽象级别上。
- 笔者将描述在非线性数据建模中广泛使用的一些神经网络方法，但是不能对该领域的发展现状给出完整的描述。如果读者希望用神经网络求解实际问题，请参阅相关文献。

38.1 记 忆

在下面几章中，将遇到几个使用简单学习算法的神经网络模型，这些算法使得神经网络具有记忆功能。也许应该先在数字计算中记忆的传统概念上稍事停留。一个记忆（如描述一个人的姓名和面部肖像的 5 000 比特所组成的一个字符串）被存储于数字计算机中的某一地址。为了恢复记忆，你需要知道地址。地址与记忆本身并没有关系。注意这个方案不具备如下特性。

1. 基于地址的记忆不具备联想功能。设想你已经知道一半的记忆，比如某个人的面部，并希望回忆起其余的记忆——他们的姓名。如果你的记忆是基于地址的，那么在不知道地址的情况下你是无法得到记忆的。[计算机科学家正竭尽全力把传统的基于地址的记忆包容在智能化软件中，从而得到内容可寻址的记忆，但是内容可寻址性无法自然地产生。它必须被额外添加。]

2. 基于地址的记忆没有鲁棒性或容错能力。如果描述地址的 1 比特出错，则将恢复一个完全不同的记忆。如果记忆中的 1 比特被翻转，则无论何时找回记忆，这个

错误都会出现。当然，在所有的现代计算机中，内存中都用到了纠错码，这样就可以检测并纠正少量的错误。但是这种容错特性并不是记忆系统所固有的特性。如果实现记忆恢复的某个硬件发生微小损坏，则可能导致所有功能的灾难性丧失。

3. 基于地址的记忆不是分布式的。对于某特定记忆进行存取的一系列计算机中，参与记忆恢复的只是极少数设备：CPU 和存储所需字节的电路。机器中数以百万计的其他设备都处于闲置状态。

是否存在多台设备参与所有计算的真正并行计算模型？[从这个角度来看，当前的并行计算机和串行计算机的区别极微。记忆的获取以同样方式进行，而计算过程的控制是在 CPU 中进行。只是多了几个 CPU 而已。大多数设备在大部分时间内都处于闲置状态。]

生物记忆系统是完全不同的。

1. 生物记忆是联想式的。回忆是内容可寻址的。已知一个人的姓名，往往可以回想起他的脸，反之亦然。显然，回忆是自发的，而不是出于某个 CPU 的请求。

2. 生物回忆具有容错性和鲁棒性。

- 回忆提示中的错误可以被纠正。例如要求你回想“一个非常有才智的美国政治家，他的政治家父亲不喜欢吃西兰花”。许多人会想起布什总统——尽管其中一个提示中含有一个错误。
- 可以容忍硬件故障。人脑是有噪声的肉团，处于连续的变化状态中，其中的细胞可能被自然过程、酒精和拳击等因素损坏。尽管人脑中的细胞和细胞中的蛋白质在不断变化，人们还是保留很多记忆未受影响。

3. 生物记忆是并行和分布式的——但在整个大脑中并非完全分布式的：看来确实存在一些功能性分工——但是在大脑的记忆存储部分，似乎有许多神经元参与了多种记忆的存储。

生物记忆系统的这些特点推动了“人工神经网络”的研究——这是由许多相互作用的简单元素所组成的并行分布式计算系统。人们希望这些模型系统也许能够就实际的生物神经网络如何实现神经计算给出一些提示。

38.2 术　语

每当描述一个神经网络算法时，我们通常给出三项内容。[如果对任何一个术语感到难以理解，最好的办法也许是直接进入下一章。]

体系结构　体系结构指明网络中包含哪些变量以及它们之间的拓扑关系——例如，一个神经网络中的变量可以是神经元之间连接的权重和神经元的活跃性。

活跃性规则　大部分神经网络模型具有小时间尺度的动态特性：局部规则定义神经元彼此响应时如何改变活跃性。活跃性规则通常取决于网络中的权重(参数)。

学习规则　学习规则规定了神经网络的权重随时间变化的方式。与活跃性规则

下的小时间尺度动态相比，这个学习通常被视为发生在一个较长的时间尺度上。通常学习规则取决于神经元的活跃性。它也可能由导师所提供的目标值以及当前的权重值决定。

这些规则从何而来？在很多情况下，活跃性规则和学习规则是由富有想象力的研究人员发明的。在另一些情况，活跃性规则和学习规则可能是从仔细选择的目标函数中推导出来的。

神经网络算法可大致分为两类。

有监督的神经网络 以输入和目标的形式向这种神经网络提供数据，其中的目标值是由导师指定神经网络对输入所应做出的响应。

无监督神经网络 以不可分的形式向这种神经网络提供数据——简单地为一个例子集合$\{\boldsymbol{x}\}$。某些学习算法就是力图简单地记住这些数据，以便在将来能够回忆这些例子。其他算法着重“推广”、发现数据“模式”或者提取数据中潜在的“特征”。

一些无监督算法能够进行预测（例如，一些算法可以在一个例子$\boldsymbol{x}$中“填加”缺失变量），所以也可视为有监督的网络。

第 39 章

单神经元分类器

39.1 单神经元

学习单神经元是出于两个原因。第一，许多神经网络模型是由单神经元组成的，所以详细了解它是有益的。第二，单神经元本身可以“学习”——各种标准的统计方法确实可以看成单神经元的形式——所以这个模型将作为第一个有监督神经网络的例子。

单神经元的定义

首先定义单神经元的体系结构和活跃性规则，然后再推导一个学习规则。

体系结构 单神经元有 I 个输入 x_i 和一个输出，这里称之为 y（如图 39.1 所示）。与每个输入相关的是一个权重 $w_i(i=1,\cdots,I)$。也可能存在另一个神经元参数，称为偏置 w_0，它可以被视为与一个数值始终为 1 的输入 x_0 相关的权重。单神经元是一个前馈设备，因为连接的方向是从神经元的输入到输出。

活跃性规则 活跃性规则分为两步。

1. 第一步，作为对所施加的输入 $\boldsymbol{x}$ 所做的响应，计算神经元的激活值

$$a = \sum_i w_i x_i \tag{39.1}$$

如果存在一个偏置，则其中的求和范围是 $i=0,\cdots,I$，否则是 $i=1,\cdots,I$。

2. 第二步，将输出 y 设为神经元激活值的函数 $f(a)$。输出也称为神经元的活跃性，不要和激活值 a 相混淆。[①] 有多种可用的激活函数；下面几种最为普遍。

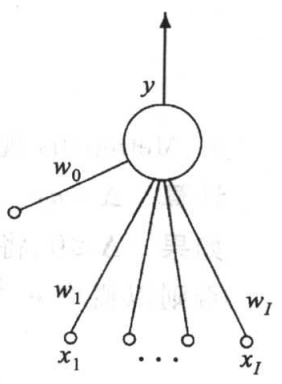


图 39.1 单神经元

（a）确定性激活函数：

i. 线性

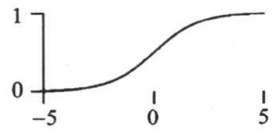
$$y(a) = a \tag{39.2}$$

① 激活值 活跃性

$a \rightarrow y(a)$

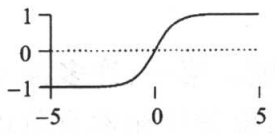
ii. S形(Logistic函数)

$$y(a)=\frac{1}{1+e^{-a}},\quad y\in(0,1) \tag{39.3}$$



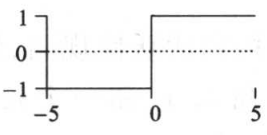
iii. S形(双曲正切tanh)

$$y(a)=\tanh(a),\quad y\in(-1,1) \tag{39.4}$$



iv. 阈值函数

$$y(a)=\Theta(a)\equiv\begin{cases}1, & a>0\\ -1, & a\leqslant 0\end{cases} \tag{39.5}$$



(b) 随机激活函数:y 随机地在 ±1 之间选取。

i. 热浴

$$y(a)=\begin{cases}1, & \text{概率为}\dfrac{1}{1+e^{-a}}\\ -1, & \text{其他}\end{cases} \tag{39.6}$$

ii. Metropolis规则所产生的输出依赖于前面的输出状态 y:

计算 $\Delta=ay$

如果 $\Delta\leqslant 0$,将 y 翻转到另一状态

否则以概率 $e^{-\Delta}$ 将 y 翻转到另一状态。

39.2 基本的神经网络概念

一个神经网络实现一个函数 $y(\boldsymbol{x};\boldsymbol{w})$。网络的"输出" y 是"输入" $\boldsymbol{x}$ 的非线性函数;该函数以"权重" $\boldsymbol{w}$ 为参数。

将研究单神经元,它用如下所示的关于 $\boldsymbol{x}$ 的函数产生在0和1之间的一个输出。

$$y(\boldsymbol{x};\boldsymbol{w})=\frac{1}{1+e^{-\boldsymbol{w}\cdot\boldsymbol{x}}} \tag{39.7}$$



习题 39.1[✓] 在何处已遇到过函数 $y(\boldsymbol{x};\boldsymbol{w})=1/(1+\mathrm{e}^{-\boldsymbol{w}\cdot\boldsymbol{x}})$？

采用线性 Logistic 函数的原因

在第 11.2 节中，假设在方差 - 协方差矩阵为 $\boldsymbol{A}^{-1}$ 的高斯信道上发送两个信号 $\boldsymbol{x}_0$ 和 $\boldsymbol{x}_1$ 之一，研究了“最佳脉冲检测问题”。假设接收信号是 $\boldsymbol{y}$ 时，源信号是 $s=1$ 而非 $s=0$ 的概率是

$$P(s=1\,|\,\boldsymbol{y})=\frac{1}{1+\exp(-a(\boldsymbol{y}))} \tag{39.8}$$

其中 $a(\boldsymbol{y})$ 是接收向量的线性函数

$$a(\boldsymbol{y})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{y}+\theta \tag{39.9}$$

这里 $\boldsymbol{w}\equiv\boldsymbol{A}(\boldsymbol{x}_1-\boldsymbol{x}_0)$。

还有其他一些原因促使我们采用线性 Logistic 函数——参见习题。

输入空间和权空间

为了方便起见，可以研究输入向量 $\boldsymbol{x}$ 和参数向量 $\boldsymbol{w}$ 都是二维的情况：$\boldsymbol{x}=(x_1,x_2)$，$\boldsymbol{w}=(w_1,w_2)$。然后可以将神经元所执行的函数写出来。

$$y(\boldsymbol{x};\boldsymbol{w})=\frac{1}{1+\mathrm{e}^{-(w_1x_1+w_2x_2)}} \tag{39.10}$$

如图 39.2 所示是在 $\boldsymbol{w}=(0,2)$ 的情况下，神经元的输出以输入向量的函数表示的形式。图中的两条横轴是输入 x_1 和 x_2，输出 y 是纵轴。注意到在任何垂直于 $\boldsymbol{w}$ 的直线上，输出是常数；在沿 $\boldsymbol{w}$ 方向的线上，输出是一个 S 形函数。

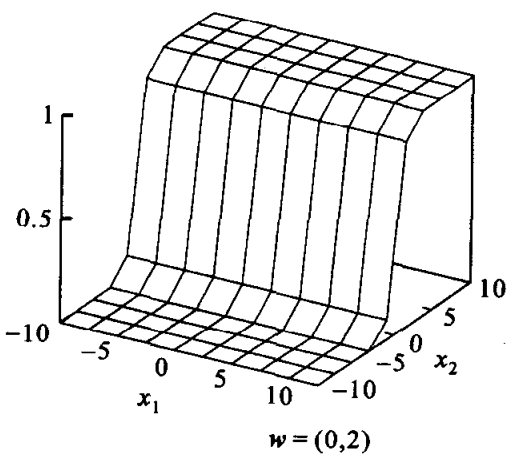


图 39.2 以输入的函数所表示的简单神经网络的输出

现在引入权空间的思想，即网络的参数空间。在这种情况下，有两个参数 w_1 和 w_2，所以权空间是二维的。这个权空间如图 39.3 所示。对于所选择的参数向量 $\boldsymbol{w}$

值,小插图所示为当 w 被设置为这些值时,网络所实现的 x 函数。每个小图都等价于图 39.2。所以 w 空间中的每个点都对应于 x 的一个函数。注意 S 形函数的增益(斜坡的梯度)随着 w 幅值的增加而增加。

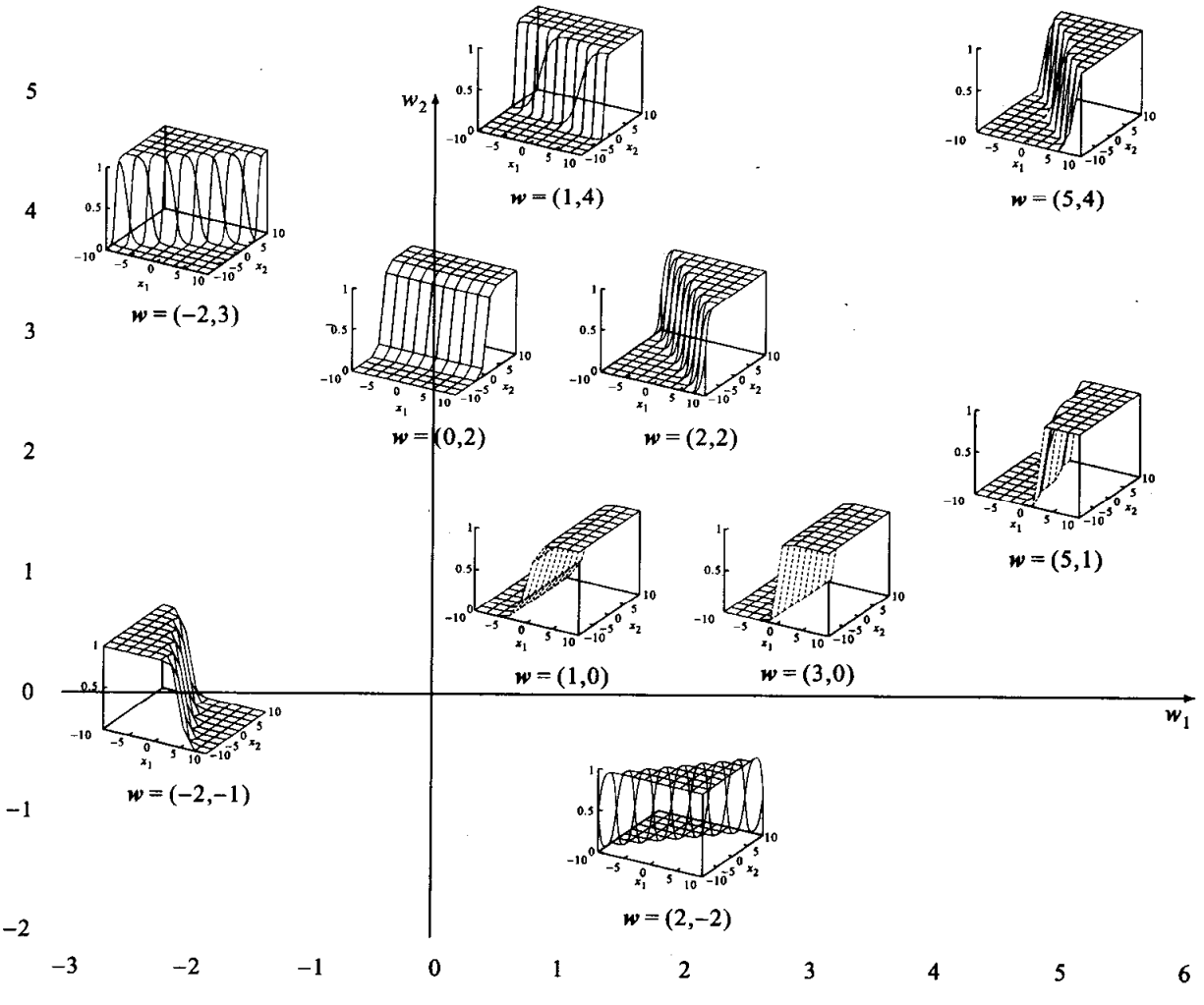


图 39.3 权空间

有监督神经网络的中心思想是这样的:已知一个输入向量 x 和一个目标 t 的关系的例子,希望神经网络能够"学习" x 和 t 之间的关系模型。对于任何已知 x,一个成功受训网络能够产生接近(在某种意义上)目标值 t 的一个输出 y。网络的训练涉及在网络的权空间中寻找 w 值,使之产生一个能够很好地拟合给定训练数据的函数。

通常定义一个*目标函数*或*误差函数*为 w 的函数,用来度量权重设置为 w 时网络完成任务的好坏程度。目标函数是这样一些项的和:每一项对应于一个输入/目标对 $\{x,t\}$,度量输出 $y(x;w)$ 与目标 t 的接近程度。训练过程是*函数极小化*的演练,即调整 w 以便找到使目标函数最小化的一个 w。许多函数极小化算法不仅要使用目标函数,而且要使用它关于参数 w 的*梯度*。对于一般的前馈神经网络,*反向传播*算法可以有效地估计输出 y 关于参数 w 的梯度,从而可以得到目标函数关于 w 的梯度。

39.3 将单神经元训练为二元分类器

假设有一个带二元标记$\{t^{(n)}\}_{n=1}^{N}$的输入数据集合$\{\boldsymbol{x}^{(n)}\}_{n=1}^{N}$和一个输出$y(\boldsymbol{x};\boldsymbol{w})$在0和1之间的单神经元。于是,可以写出以下*误差函数*

$$G(\boldsymbol{w}) = -\sum_{n}\left[t^{(n)}\ln y(\boldsymbol{x}^{(n)};\boldsymbol{w}) + (1-t^{(n)})\ln(1-y(\boldsymbol{x}^{(n)};\boldsymbol{w}))\right] \tag{39.11}$$

目标函数中的每一项都可认为是一个结果的信息量。也可以描述为经验概率分布$(t^{(n)},1-t^{(n)})$和神经元输出$(y,1-y)$所隐含的概率分布之间的相对熵。目标函数的下界为0,只在对所有n都有$y(\boldsymbol{x}^{(n)};\boldsymbol{w})=t^{(n)}$时才达到这个值。

现在来求目标函数关于$\boldsymbol{w}$的导数。



习题39.2[2] **反向传播算法**。证明导数$\boldsymbol{g}=\partial G/\partial \boldsymbol{w}$由下式给出。

$$g_j = \frac{\partial G}{\partial w_j} = \sum_{n=1}^{N} -(t^{(n)}-y^{(n)})x_j^{(n)} \tag{39.12}$$

注意量$e^{(n)}\equiv t^{(n)}-y^{(n)}$是关于例子$n$的*误差*——目标和输出之间的差异。对误差函数的梯度所进行的最简单处理就是使梯度下降(尽管这在量纲上往往是不正确的,因为梯度的量纲是[1/参数],而一个参数的变化量纲是[参数])。因为导数$\partial G/\partial \boldsymbol{w}$是定义如下的$\boldsymbol{g}^{(n)}$项之和。

$$g_j^{(n)} \equiv -(t^{(n)}-y^{(n)})x_j^{(n)} \tag{39.13}$$

其中$n=1,\cdots,N$,每次只将一个输入送入网络中,并从与$\boldsymbol{g}^{(n)}$相反的方向略微调整$\boldsymbol{w}$,就能得到一个简单的在线算法。

现在总结完整的学习算法。

在线梯度下降学习算法

体系结构 一个单神经元有I个输入x_i和一个输出y。有一个权重$w_i(i=1,\cdots,I)$与每个输入相关联。

活跃性规则 1. 第一步,作为对所接收到的输入$\boldsymbol{x}$(可以是任意实数)的响应,计算神经元的*激活值*

$$a = \sum_{i} w_i x_i \tag{39.14}$$

如果存在一个偏置,则其中的求和范围是$i=0,\cdots,I$,否则是$i=1,\cdots,I$。

2. 第二步,将输出y设为激活值的S形函数。

$$y(a) = \frac{1}{1+\mathrm{e}^{-a}} \tag{39.15}$$

可认为神经元的输出是,根据神经元,已知输入属于类1而非类0的概率。

学习规则 导师提供一个目标值 $t \in \{0,1\}$，表示对于已知输入的正确答案是什么。计算

误差信号

$$e = t - y \tag{39.16}$$

然后沿着可以减小这个误差幅度的方向来调整权重 w

$$\Delta w_i = \eta e x_i \tag{39.17}$$

其中 η 是“学习速率”。通常采用试探法将 η 设定为一个常数值，或者为仿真时间 τ 的一个递减函数，比如 η_0/τ。

对每一个出现的输入/目标对 (x,t) 重复活跃性规则和学习规则。如果数据集的大小 N 固定，可以多次循环数据。

批学习与在线学习

这里已经描述了在线学习算法：对于所出现的每个例子都修正权值。另外一种模式是对例子进行批处理：计算每次的输出和误差，并累积式(39.17)所指定的变化量，然后在批结束时进行修改。

单神经元分类器的批学习算法

对于每个输入/目标对 $(x^{(n)},t^{(n)})$ $(n=1,\cdots,N)$，计算 $y^{(n)}=y(x^{(n)};w)$，其中

$$y(x;w) = \frac{1}{1+\exp(-\sum_i w_i x_i)} \tag{39.18}$$

定义 $e^{(n)}=t^{(n)}-y^{(n)}$，对每个权重 w_i 计算

$$g_i^{(n)} = -e^{(n)}x_i^{(n)} \tag{39.19}$$

然后令

$$\Delta w_i = -\eta \sum_n g_i^{(n)} \tag{39.20}$$

这个批学习算法是一种梯度下降算法，而在线算法是一种随机梯度下降算法。实现批学习算法的源代码在算法 39.5 中给出。对于权重分别为 w_1 和 w_2 的两个输入的单神经元，及一个偏置 w_0，执行函数

$$y(x;w) = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2)}} \tag{39.21}$$

这个算法如图 39.4 所示。图 39.3 省略了偏置，与之不同，这里包含偏置 w_0。由 10 个有标记的例子所组成的数据集对神经元进行了训练。

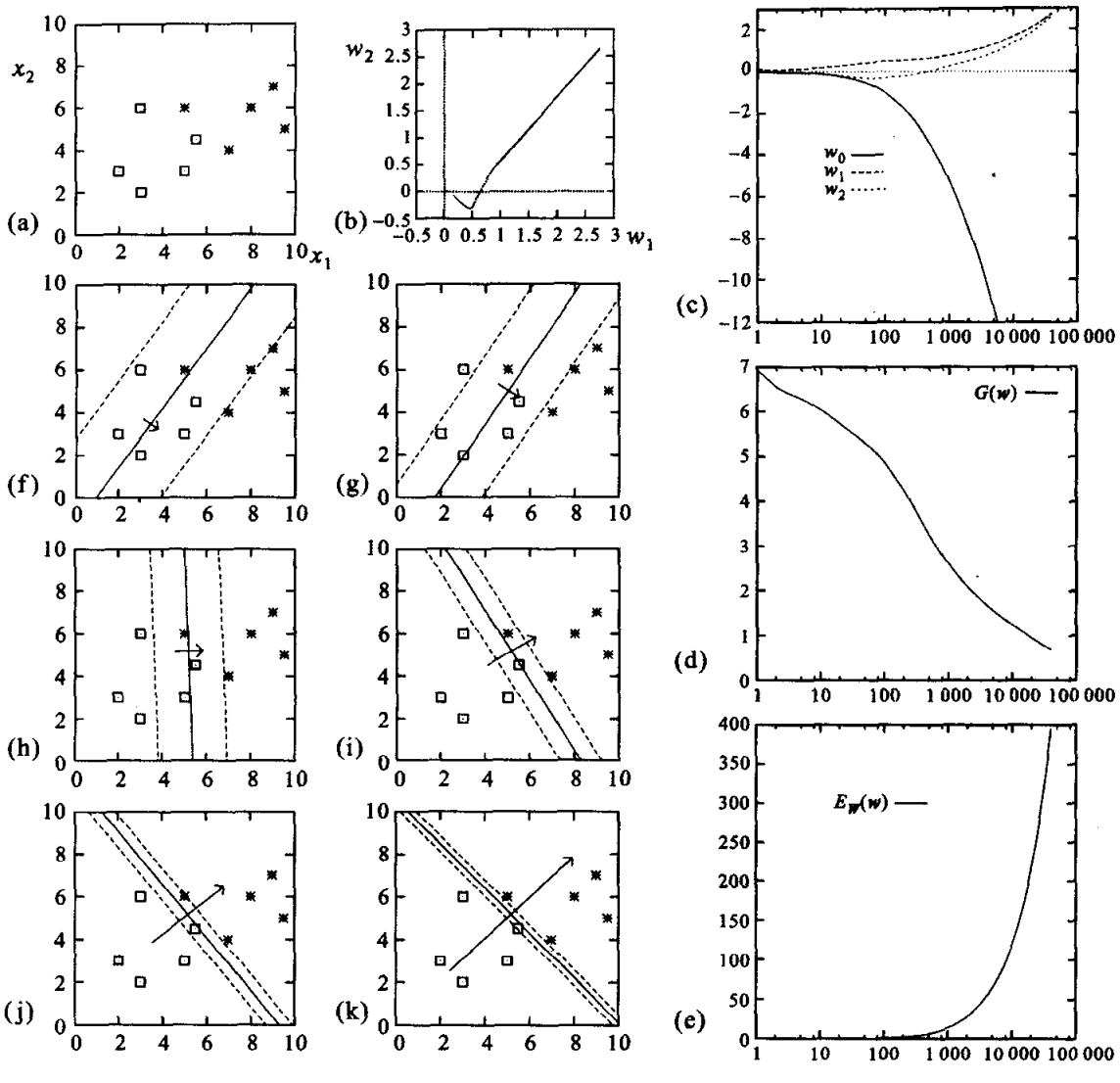


图 39.4 单神经元采用梯度下降法学习分类。神经元有两个权重 w_1 和 w_2 以及一个偏置 w_0。学习速率被设为 $\eta = 0.01$，使用算法 39.5 中的程序代码来执行批模式梯度下降算法。(a) 训练数据。(b) 权重 w_0、w_1 和 w_2 随迭代次数进行的演化(对数坐标)。(c) 权重 w_1 和 w_2 在权空间中的演化。(d) 作为迭代次数函数的目标函数 $G(w)$。(e) 作为时间函数的权重 $E_W(w)$ 的幅度。(f ~ k) 在 30、80、500、3 000、10 000 和 40 000 次迭代后的神经元运行函数(表示为它的 3 条等势线)。图中的等势线所对应的是 $a = 0, \pm 1$，即 $y = 0.5, 0.27$ 和 0.73。同时显示的是和 (w_1, w_2) 成正比的向量。权重越大，该向量就越大，而且等势线也就靠得越近。

```
global x ;                                  # x 是一个 N * I 维矩阵,包含了所有输入向量
global t ;                                  # t 是一个长度为 N 的向量,包含了所有目标
for l = 1:L                                 # 循环 L 次
    a = x * w ;                             # 计算所有激活值
    y = sigmoid(a) ;                        # 计算输出
    e = t - y ;                             # 计算误差
    g = - x' * e ;                          # 计算梯度向量
    w = w - eta * ( g + alpha * w );        # 使用学习速率 eta 以及权衰减 alpha,进行步进
endfor
function f = sigmoid ( v )
    f = 1.0 ./ ( 1.0 . + exp ( - v ) ) ;
endfunction
```

算法 39.5 带可选权衰减(速率为 alpha)的单神经元梯度下降优化器(批学习)的 Octave 源代码。Octave 中的记号:指令 a = x * w 使得所有输入向量所组成的($N \times I$)矩阵 x 与权向量 w 相乘,产生的向量 a 列出所有 N 个输入向量的激活值;x′表示 x 的转置;单指令 y = sigmoid(a)计算向量 a 中所有分量的 S 形函数。

39.4 在误差函数的下降之外:正则化

如果参数 η 被设为一个适当的值,这个算法就有用:算法找到对尽可能多的例子进行正确分类的那个权重 $\boldsymbol{w}$ 的设置。

如果例子实际上是线性可分的,则神经元会找出这个线性可分离性,而且随着仿真的继续进行,它的权重会发散为更大的值。在图 39.4(f ~ k)中可看到这种情况。这是过度拟合的一个例子,其中模型对数据拟合得很好,但对推广却可能起负面影响。

这种现象也许不是我们所期望的。如何才能纠正它?

对于过度拟合,一种特别的解决方案就是使用提前停止,即使用原计划的算法来使误差函数 $G(\boldsymbol{w})$ 最小化,然后通过在某点终止算法,来避免发生这种情况。

解决过度拟合的一种更有原则性的方案是采用正则化。正则化包括对目标函数做适当的修正,使之包含一个偏置来抵制我们所不喜欢的那些解 $\boldsymbol{w}$。在上面的例子中,所不喜欢的是图39.4(k)中形成的很突兀的判决边界;这个突兀的边界与大的权重值有关,所以使用一个正则化因子来惩罚大的权重值。修正目标函数为

$$M(\boldsymbol{w}) = G(\boldsymbol{w}) + \alpha E_{\boldsymbol{w}}(\boldsymbol{w}) \tag{39.22}$$

其中最简单的正则化因子选择是权衰减正则化因子

$$E_W(\boldsymbol{w}) = \frac{1}{2}\sum_i w_i^2 \tag{39.23}$$

正则化常数 α 被称为权衰减率。这个额外的项对于小的 $\boldsymbol{w}$ 值有利,并且可以阻止模型关于训练数据在一些细微之处过度拟合。量 α 也叫超参数。超参数在学习算法中起作用,但在网络的活跃性规则中不起作用。

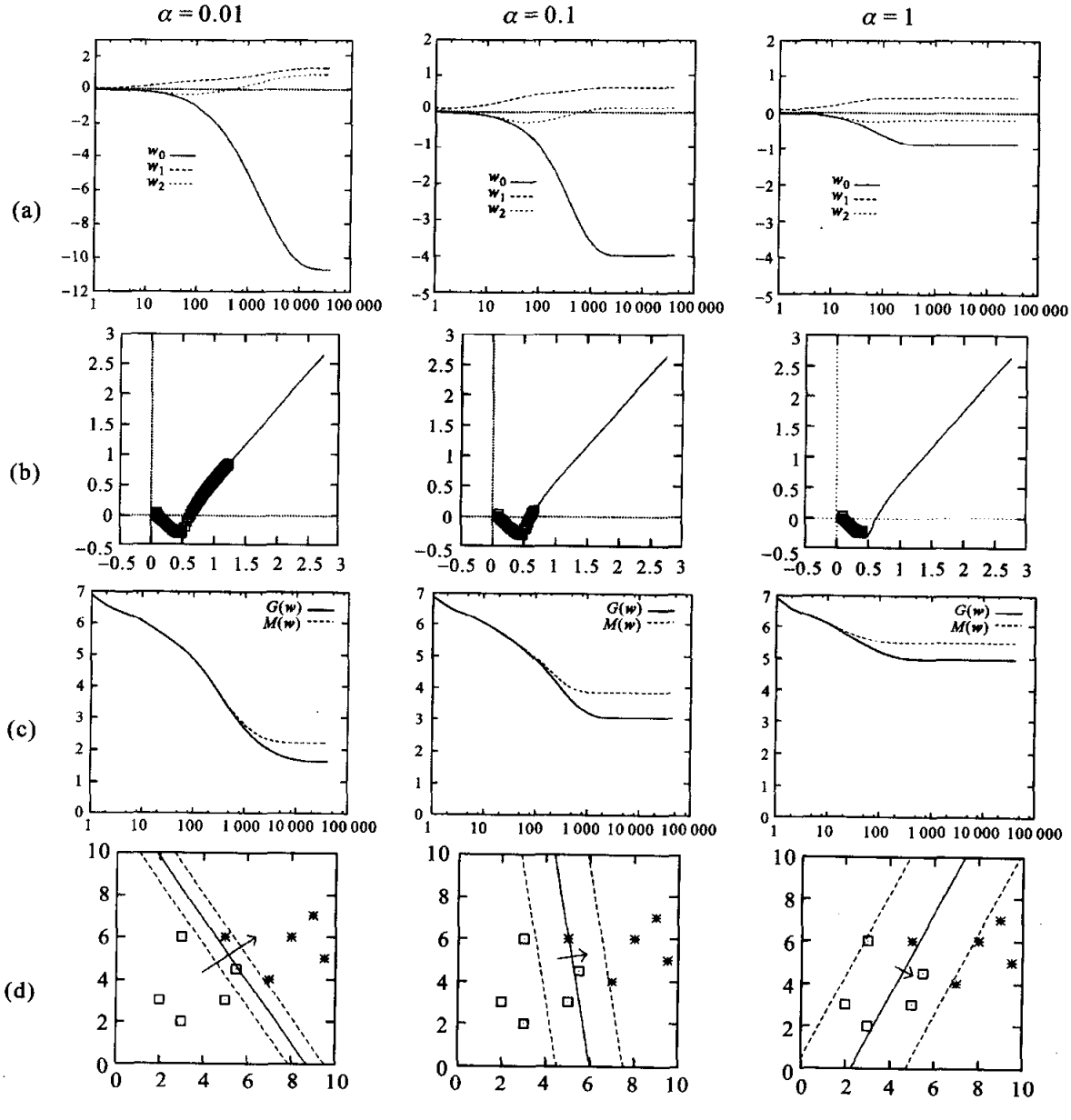


图 39.6 权衰减对单神经元学习的影响。目标函数是 $M(\boldsymbol{w}) = G(\boldsymbol{w}) + \alpha E_W(\boldsymbol{w})$。学习方法如图 39.4 所示。(a) 权 w_0, w_1 和 w_2 的演化。(b) 在权空间中权 w_1 和 w_2 的演化,用点表示,对比零权衰减时的轨迹,用细线表示(在图 39.4 中)。注意,对于这个问题,权衰减的效果非常类似于"提前停止"。(c) 作为迭代次数函数的目标函数 $M(\boldsymbol{w})$ 和误差函数 $G(\boldsymbol{w})$。(d) 经过 40 000 次迭代之后的神经元执行函数。

习题 39.3[1] 计算 $M(\boldsymbol{w})$ 关于 w_i 的导数。为什么上面的正则化因子被称为“权衰减”正则化因子？

算法 39.5 的梯度下降源代码实现了权衰减。使用权衰减率 $\alpha = 0.01, 0.1$ 和 1 的梯度下降算法如图 39.6 所示。当权衰减率增加时，所得解趋向于更宽的 S 形函数，其判决边界更靠近原点。

注意

通常，具有步长 η 的梯度下降法不是对函数进行最小化的最有效方法。对梯度下降算法的一个修正称为动量法，尽管它改善了收敛性能，但也不推荐使用。大部分神经网络专家使用更为先进的优化方法，诸如共轭梯度算法。[有时用符号 α 表示动量，请不要将动量和权衰减率相混淆。]

39.5 进一步的习题

采用线性神经元的更多原因

▷ **习题 39.4**[2] 考虑识别一个向量 $\boldsymbol{z}$ 来自两个高斯分布中的哪一个的任务。不同于第 11.2 节的情况，那里分布具有不同的均值，但有相同的方差 - 协方差矩阵，现在假设两个分布具有同样的均值但有不同的方差。在已知 $s(s \in \{0,1\})$ 的条件下，$\boldsymbol{z}$ 的概率为

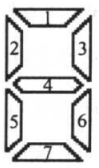
$$P(\boldsymbol{z} \mid s) = \prod_{i=1}^{I} \text{Normal}(z_i; 0, \sigma_{si}^2) \tag{39.24}$$

其中 σ_{si}^2 是源符号为 s 时 z_i 的方差。证明 $P(s=1|\boldsymbol{z})$ 可以写成形式

$$P(s=1|\boldsymbol{z}) = \frac{1}{1+\exp(-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+\theta)} \tag{39.25}$$

其中 x_i 是 z_i 的一个合适函数 $x_i = g(z_i)$。

习题 39.5[2] 有噪 LED




$\boldsymbol{c}(2) =$


$\boldsymbol{c}(3) =$


$\boldsymbol{c}(8) =$ 8

考虑一个 7 段 LED 显示器，编号如上图所示。所显示的状态是一个向量 $\boldsymbol{x}$。当控制器让显示器显示数字 s 时，例如 $s=2$，则每个元素 $x_j (j=1,\cdots,7)$ 要么以概率 $1-f$ 处

于想要的状态 $c_j(s)$，要么以概率 f 翻转。称 x 的两个状态为“+1”和“-1”。

(a) 假设想要的字符 s 实际上是 2 或 3，在已知状态 $\boldsymbol{x}$ 的条件下，s 的概率是多少？证明 $P(s=2|\boldsymbol{x})$ 可写成形式

$$P(s=2|\boldsymbol{x})=\frac{1}{1+\exp(-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+\theta)} \tag{39.26}$$

在 $f=0.1$ 的情况下，计算权重 $\boldsymbol{w}$ 的值。

(b) 假设 s 是 $\{0,1,2,\cdots,9\}$ 中的一个值，先验概率为 p_s，在已知状态 $\boldsymbol{x}$ 的条件下，s 的概率是多少？将你的答案写成形式

$$P(s\mid\boldsymbol{x})=\frac{\mathrm{e}^{a_s}}{\sum_{s'}\mathrm{e}^{a_{s'}}} \tag{39.27}$$

其中 $\{a_s\}$ 是 $\{c_j(s)\}$ 和 $\boldsymbol{x}$ 的函数。

对于有噪 LED，你能否给出更好的 10 个字符的符号表，即更不容易引起混乱的符号表？符号表的示例(15 个字符的符号表)如表 39.7 所示。

表 39.7 另一种 7 段 LED 显示器的 15 个字符的符号表

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	

▷ **习题 39.6**[2] 一个(3,1)纠错码由两个码字 $\boldsymbol{x}^{(1)}=(1,0,0)$ 和 $\boldsymbol{x}^{(2)}=(0,0,1)$ 组成。用一个概率分布为 $\{p_1,p_2\}$ 的源比特 $s\in\{1,2\}$ 来选择两个码字之一，在噪声水平为 f 的一个二进制对称信道上进行传输。接收向量是 $\boldsymbol{r}$。证明已知 $\boldsymbol{r}$ 时 s 的后验概率可以写成形式

$$P(s=1\mid\boldsymbol{r})=\frac{1}{1+\exp(-w_0-\sum_{n=1}^{3}w_n r_n)}$$

并给出系数 $\{w_n\}_{n=1}^{3}$ 和偏置 w_0 的表达式。

通过一个示意图来说明如何将这个最优译码器表示成一个“神经元”的形式。

在第 40 章之前要思考的问题

▷ **习题 40.1**[♂] $\sum_{K=0}^{N}\binom{N}{K}$等于多少?

[符号$\binom{N}{K}$表示组合数$\frac{N!}{K!\ (N-K)!}$。]

▷ **习题 40.2**[♂] 如果 Pascal 三角的首行(只包含单个数“1”)被称为第 0 行,那么在三角形第 N 行以上的数的总和是多少?

▷ **习题 40.3**[♂] 在一个球面上随机地选择 3 个点,则 3 个点在同一半球上的概率是多少?

这一章的素材最初源自 Polya(1954 年)和 Cover(1965 年),阐述方式参照 Yaser Abu - Mostafa。

第 40 章

单神经元的容量

40.1 用于通信的神经网络学习

许多神经网络模型都涉及权重 $\boldsymbol{w}$ 的集合对一个数据点集的反应，例如在已知的位置 $\{\boldsymbol{x}_n\}_{n=1}^N$ 上有 N 个目标值 $D_N=\{t_n\}_{n=1}^N$ 的一个集合。自适应权重被用来处理随后的输入数据。这一过程可以视为通信过程（如图 40.1 所示），其中发送方检查数据 D_N 并产生依赖于这些数据的消息 $\boldsymbol{w}$。然后接收方就可使用 $\boldsymbol{w}$；例如，接收方可以使用权重来试着重构数据 D_N。[用神经网络的话来说，这是用神经元来“记忆”而非“推广”；“推广”意味着从被观测数据外推出新位置 $\boldsymbol{x}_{N+1}$ 处的 t_{N+1} 值。]因此，正如一个磁盘驱动器是一个通信信道一样，自适应的网络权重 $\boldsymbol{w}$ 所起的作用相当于一个通信信道，把训练数据的信息传递到神经网络的一个未来用户。现在要着手解决的问题是，“这个信道的容量是多少？”——即“通过训练神经网络，可以存储多少信息？”

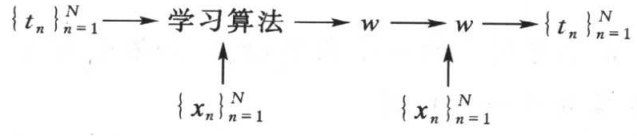


图 40.1　从通信的角度看待神经网络学习

如果存在一种学习算法来产生一个网络，它对所有输入的响应要么都是 +1，要么都是 0，具体取决于训练数据，则权重只允许我们识别两类数据集合。这种学习算法所能传递的关于数据的最大信息是 1 比特，而且这个信息量在两类数据是等可能时取得。如果充分利用神经网络的能力来表示其他函数，那么能多传递多少信息呢？

40.2 单神经元的容量

现在来看最简单的情况，即单个二进制阈值神经元。可以发现这样一个单神经元的容量是每个权重 2 比特。一个神经元若有 K 个输入，则能存储 $2K$ 比特的信息。

为了得到这个有趣的结果，设置一些规则以排除一些不太有意思的答案，比如，

“单神经元的容量是无限的,因为它的每个权重都是一个实数,所以能够传递无限个比特”。通过如下假设来排除这种答案:接收方不能直接查看权重,也不能通过观测神经元对任意选择输入的输出来探测权重。限定接收方只能在与训练集合$\{x_n\}$相同的N点固定集合上观测神经元的输出。关键问题是,已知只能在这N个点上观测函数,神经元可以产生多少个不同的可分辨函数。一个线性阈值函数可以产生多少个不同的N点二进制标记?这个数与最大可能的二进制标记数2^N相比又怎样?如果几乎所有2^N个标记都可以用神经元来实现,那么它就是能以小的差错概率传递所有N比特(目标值$\{t_n\}$)的一个通信信道。可将神经元的容量定义为保持差错概率非常小的最大N值。[稍微偏离了第9章关于容量的定义。]

因此检验下面的情况。发送方有一个神经元,该神经元具有K个输入和对N个点进行标记的一个数据集D_N。发送方通过一个自适应算法试图找出可准确重构该标记的$\boldsymbol{w}$。假设如果这样的$\boldsymbol{w}$存在,算法就可以找到它。于是,接收方计算N个输入值上的阈值函数。所有N个比特都能被正确复制的概率是多少?对于已知的K,不会使这个概率远小于1的N可达到多大?

一般位置

一个技术细节需要被确定下来:我们考虑的是什么样的输入集合$\{x_n\}$?答案也许和这个选择有关。假设点处于一般位置,这意味着$K=3$维,例如,没有3个点共线,没有4个点共面。

定义40.1 对于K维空间上的一个点集$\{x_n\}$,如果任何大小$\leqslant K$的子集都是线性无关的,则称这个点集处于一般位置。

例如,在$K=3$维空间中,如果没有3个点共线,也没有4个点共面,则该点的集合处于一般位置。直观的想法是,按照点之间的线性相关性,处于一般位置上的点就像空间中的随机点。你不会期望三维空间中的3个随机点位于一条直线上。

线性阈值函数

所考虑的神经元执行函数

$$y = f\left(\sum_{k=1}^{k} w_k x_k\right) \tag{40.1}$$

其中

$$f(a) = \begin{cases} 1, & a > 0 \\ 0, & a \leqslant 0 \end{cases} \tag{40.2}$$

这里没有偏置w_0;具有偏置的神经元的容量可以在最终结果中用$K+1$代替K得到,即,考虑其中的一个输入固定为1。(尽管这些输入点就不会处于一般位置了,但推导仍然有效。)

40.3 对阈值函数进行计数

用 $T(N,K)$ 表示 K 维空间中位于一般位置的 N 个点上不同阈值函数的个数。我们将推导 $T(N,K)$ 的一个公式。

从对一些情况进行手工计算开始。

对于任意 N,在 $K=1$ 维空间中

N 个点位于一条直线上。通过改变一个权重 w_1 的符号,可以把所有位于原点右边的点标记为 1,另一边标记为 0,或者反之亦然。因此,存在两个不同的阈值函数。$T(N,1)=2$。

对于任意 K,有 $N=1$ 个点

如果只有一个点 $\boldsymbol{x}^{(1)}$,那么可以通过设定 $\boldsymbol{w}=\pm\boldsymbol{x}^{(1)}$ 来实现两个可能的标记。所以,$T(1,K)=2$。

在 $K=2$ 维空间中

在有 N 个点的二维空间中,可以围绕原点随意旋转分隔线。每次直线通过一个点,就得到一个新函数。一旦把直线旋转 360°,就重新产生开始时的函数。因为点处于一般位置,所以分隔面(线)每次只能穿过一个点。在一次旋转中,每个点被穿过两次。所以有 $2N$ 个不同的阈值函数,$T(N,2)=2N$。

与二进制函数的总数 2^N 相比,可能注意到对于 $N\geqslant 3$,并非所有二进制函数都可以通过线性阈值函数来实现。$N=4$ 和 $K=2$ 时,一个著名的不可实现函数的例子是在点 $\boldsymbol{x}=(\pm 1,\pm 1)$ 上的异或函数。[这些点不在一般位置上,但是可以确认,即使这些点受扰动移到一般位置上,函数也还是不可实现的。]

在 $K=2$ 维空间,从权空间的角度考虑

可从另一角度使这个问题形象化。不是直观表示分隔二维输入空间上各点的一个平面,而是考虑二维权空间:如果权空间的区域将已知的数据点做上不同的标记,就为它们着以不同的颜色。这样,通过计算权空间可区分区域的数目,就可实现对阈值函数的计数。首先考虑在权空间中把一个特例 $\boldsymbol{x}^{(n)}$ 划分为第 1 类的权向量集。例如,图 40.2(a)表示二维 $\boldsymbol{x}$ - 空间中的一个点,图 40.2(b)表示 $\boldsymbol{w}$ 空间中两个对应的点集。一个权向量集合占据了半空间

$$\boldsymbol{x}^{(n)}\cdot\boldsymbol{w}>0 \tag{40.3}$$

另一个占据 $\boldsymbol{x}^{(n)} \cdot \boldsymbol{w} < 0$。在图 40.3(a)中,在输入空间中又增加了第二个点。有 4 种可能的标记:(1,1),(1,0),(0,1)和(0,0)。图 40.3(b)表示两个超平面 $\boldsymbol{x}^{(1)} \cdot \boldsymbol{w} = 0$ 和 $\boldsymbol{x}^{(2)} \cdot \boldsymbol{w} = 0$,它们分割了产生这些标记的权向量集。当 $N = 3$ 时(如图 40.4 所示),权空间被 3 个超平面分成 6 个区域。并非 8 个想象出的标记都可被实现。所以,$T(3,2) = 6$。

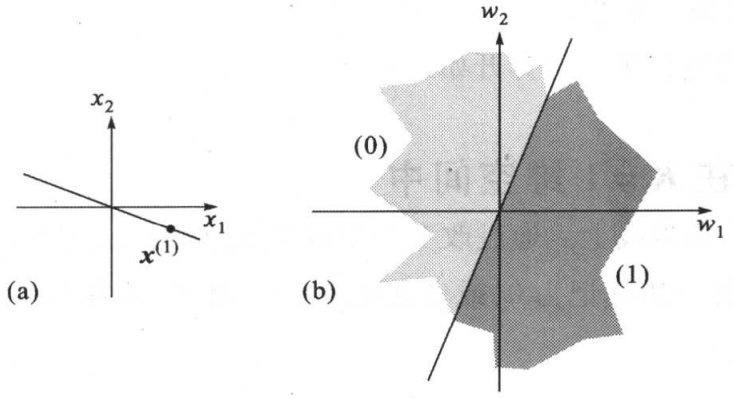


图 40.2 在一个二维输入空间上的一个数据点,及权空间中可给出该点的两个不同标记的两个区域。

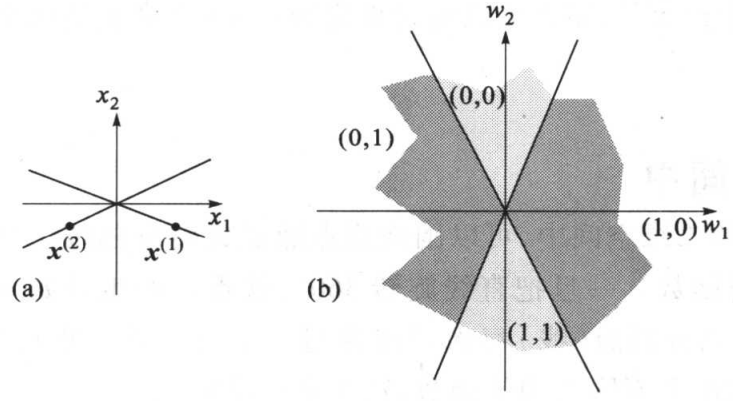


图 40.3 二维输入空间上的两个数据点,及权空间中给出 4 个不同标记的 4 个区域。

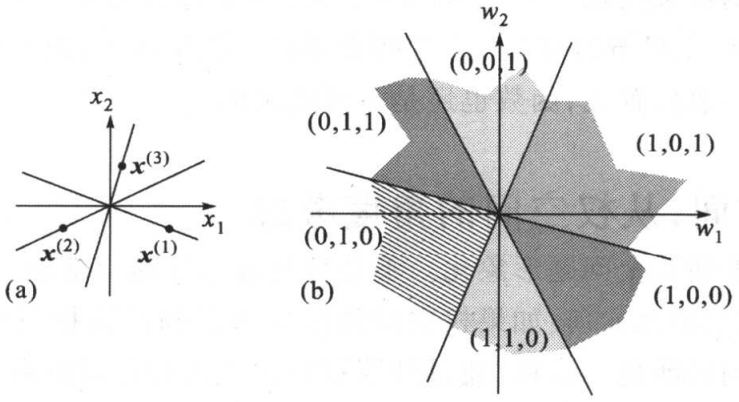


图 40.4 二维输入空间中的 3 个数据点,及为这些点做不同标记的权空间的 6 个区域。在这种情况下,标记(0,0,0)和(1,1,1)不能实现。对于一般位置上的任意 3 个点,总是有两个标记不能实现。

在 $K=3$ 维空间中

现在利用这种权空间可视化方法来研究三维情况。

设想一次增加一个点，同时对阈值函数进行计数。当 $N=2$ 时，权空间被两个超平面 $\boldsymbol{x}^{(1)} \cdot \boldsymbol{w}=0$ 和 $\boldsymbol{x}^{(2)} \cdot \boldsymbol{w}=0$ 分成4个区域；任何一个区域中的所有向量 $\boldsymbol{w}$ 在两个输入向量上产生同样的函数。所以 $T(2,3)=4$。

在一般位置上增加第三个点，从而在 $\boldsymbol{w}$ 空间上产生第三个平面，所以有8个可区分的区域，$T(3,3)=8$。3个对分的平面如图40.5(a)所示。

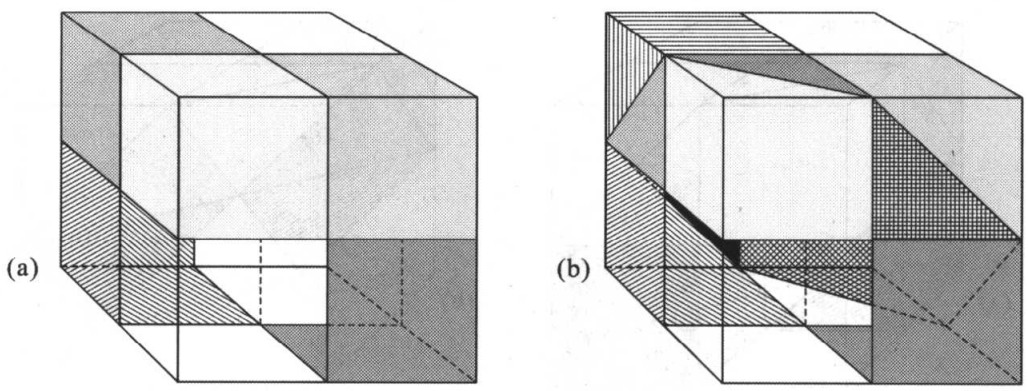


图40.5 关于 $T(3,3)$ 和 $T(4,3)$ 的权空间图解。(a) $T(3,3)=8$。3个超平面(对应于一般位置上的3个点)把3-空间分成8个区域，这里通过对一个虚空、半透明、以原点为中心的立方体表面的相关部分着色来表示。(b) $T(4,3)=14$，4个超平面把3-空间分成14个区域，图中只能看到13个(第14个区域在右面，看不到)。与图40.5(a)相比，所有非白色区域已被分成两部分。

此时，情况变得有些复杂。如图40.5(b)所示，三维 $\boldsymbol{w}$ 空间中的第4个平面不能和前面3个平面所产生的所有8个集合相交。已有区域中的6个被切成两部分，剩下的两个不受影响，所以 $T(4,3)=14$。三维空间中的4个点上的两个二进制函数不能通过线性阈值函数来实现。

现在已经把 $T(N,K)$ 的值填在表40.6中。能否从 $T(4,3)$ 的推导中获得任何深入认知，以便填充表中其他的 $T(N,K)$ 值？为什么 $T(4,3)$ 比 $T(3,3)$ 大6？

6是 $\boldsymbol{w}$ 空间(如图40.7(a)(b)所示)中新的超平面所对分的区域数。同样地，如果观察第 N 个超平面所组成的 $K-1$ 维子空间，那么该子空间被前面 $N-1$ 个超平面分成6个区域(如图40.7(c)所示)。这是前面已经遇到过的概念。比较图40.7(c)和图40.4(b)。在 $K-1$ 维空间中，$N-1$ 个超平面可以产生多少个区域？噢，当然是 $T(N-1,K-1)$！在当前 $N=4,K=3$ 的情况下，在前面章节可以发现 $T(3,2)=6$。所以

$$T(4,3)=T(3,3)+T(3,2) \tag{40.4}$$

表 40.6 手工计算的 $T(N,K)$ 值

N	K							
	1	2	3	4	5	6	7	8
1	2	2	2	2	2	2	2	2
2	2	4	4					
3	2	6	8					
4	2	8	14					
5	2	10						
6	2	12						

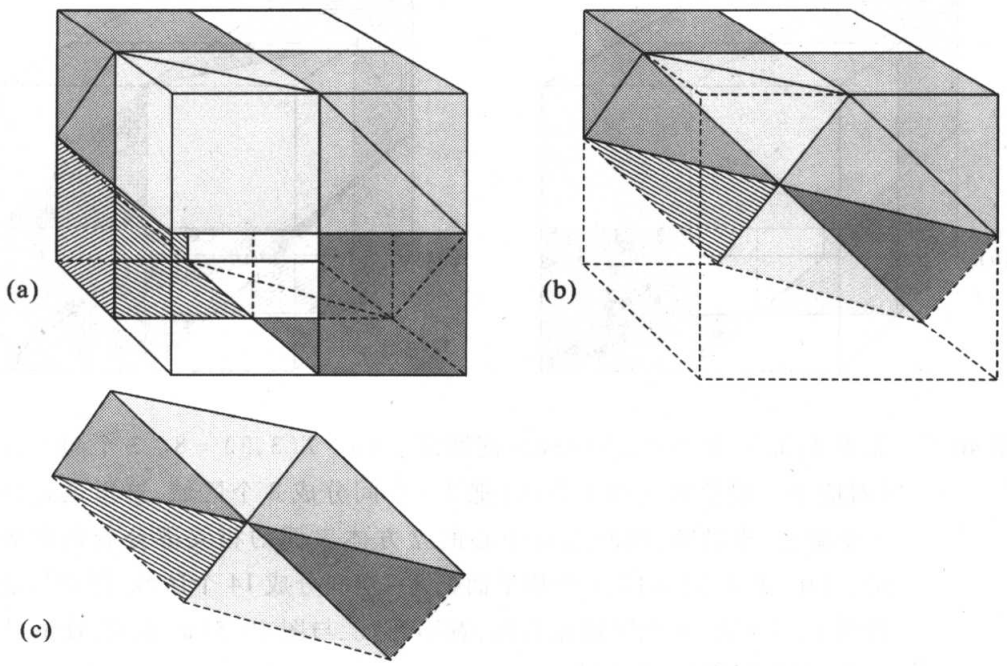


图 40.7 从 $T(3,3)$ 到 $T(4,3)$ 的切割过程图解。(a)增加一个超平面的图 40.5(a)的 8 个区域。所有未标为白色的区域都被分割成两部分。(b)这里,把虚空立方体变成实心的,所以能够看到哪个区域被第 4 个平面分割。立方体的前半部被切除了。(c)该图显示了新的二维超平面,该超平面被穿越它的 3 个一维超平面(直线)分成 6 个区域。这些区域中的每一个都对应于图 40.7(a)中的一个三维区域,该三维区域被这个新的超平面分成两部分。这表明 $T(4,3)-T(3,3)=6$。应该将图 40.7(c)和图 40.4(b)进行比较。

对于任意 N、K 的递推关系

对这张图进行推广,可以看到,在 K 维空间中增加第 N 个超平面时,它将对分前面 $N-1$ 个超平面所产生的 $T(N-1,K)$ 个区域中的 $T(N-1,K-1)$ 个区域。因此,增加第 N 个超平面后所得的总区域数是 $2T(N-1,K-1)$(因为 $T(N-1,K)$ 中有 $T(N-1,K-1)$ 个区域被分成两部分)加上剩余的未被第 N 个超平面所分割的区域

$T(N-1,K)-T(N-1,K-1)$,因而,关于 $T(N,K)$ 有下述方程。

$$T(N,K)=T(N-1,K)+T(N-1,K-1) \tag{40.5}$$

现在剩余工作是在已知边界条件 $T(N,1)=2$ 和 $T(1,K)=2$ 时,求解这个递推关系。

递推关系(40.5)看起来是否眼熟?也许你还记得通过把一行中相邻的两个数相加得到下面一行数字所构造的 Pascal 三角。Pascal 三角的 N,K 元素等于

$$C(N,K)\equiv\binom{N}{K}\equiv\frac{N!}{(N-K)!\ K!} \tag{40.6}$$

组合数 $\binom{N}{K}$ 满足方程

$$C(N,K)=C(N-1,K-1)+C(N-1,K),N>0 \tag{40.7}$$

[这里遵循约定:如果 $K>N$ 或 $K<0$,则 $\binom{N}{K}\equiv 0$。]所以 $\binom{N}{K}$ 满足所要求的递推关系(40.5)。这并不意味着 $T(N,K)=\binom{N}{K}$,因为许多函数都可以满足一个递推关系。

但是也许可以把 $T(N,K)$ 表示成形如 $C_{\alpha,\beta}(N,K)=\binom{N+\alpha}{K+\beta}$ 的组合函数的一个线性叠加。通过比较表 40.8 和表 40.6,可以看出如何才能满足边界条件:只需把 Pascal 三角向右平移 1,2,3,…;重合;相加;乘以 2,然后从整张表中去掉一行。因此有

$$T(N,K)=2\sum_{k=0}^{K-1}\binom{N-1}{k} \tag{40.8}$$

利用 Pascal 三角第 N 行之和为 2^N 这一事实,即 $\sum_{k=0}^{N-1}\binom{N-1}{k}=2^{N-1}$,可以把 $K-1\geqslant N-1$ 的情况简化为

$$T(N,K)=\begin{cases}2^N, & K\geqslant N\\ 2\sum_{k=0}^{K-1}\binom{N-1}{k}, & K<N\end{cases} \tag{40.9}$$

表 40.8 Pascal 三角

N	K							
	0	1	2	3	4	5	6	7
0	1							
1	1	1						
2	1	2	1					
3	1	3	3	1				
4	1	4	6	4	1			
5	1	5	10	10	5	1		

解释

很自然，要比较 $T(N,K)$ 和在 N 个点上的二进制函数的总数 2^N，比值 $T(N,K)/2^N$ 给出了可以用神经元来记忆一个任意标记 $\{t_n\}_{n=1}^N$ 的概率。对于所有 $N \leqslant K$，这两个函数都是相等的。所以 $N=K$ 是一条特别的直线，它定义了可以在其上实现任意标记的最大点数。这个点数就叫做这类函数的 Vapnik - Chervonenkis 维度（VC 维度）。所以在 K 维空间上，一个二进制阈值函数的 VC 维度就是 K。

有意思的是（对于大的 K），几乎能够实现任意标记的点数 N。对于 $N<2K$，比值 $T(N,K)/2^N$ 仍然大于 1/2，对于大的 K，比值非常接近 1。

对于我们而言，可以用以下误差函数很好地近似式(40.9)中的和。

$$\sum_{0}^{K}\binom{N}{k} \simeq 2^N \Phi\left(\frac{K-(N/2)}{\sqrt{N}/2}\right) \tag{40.10}$$

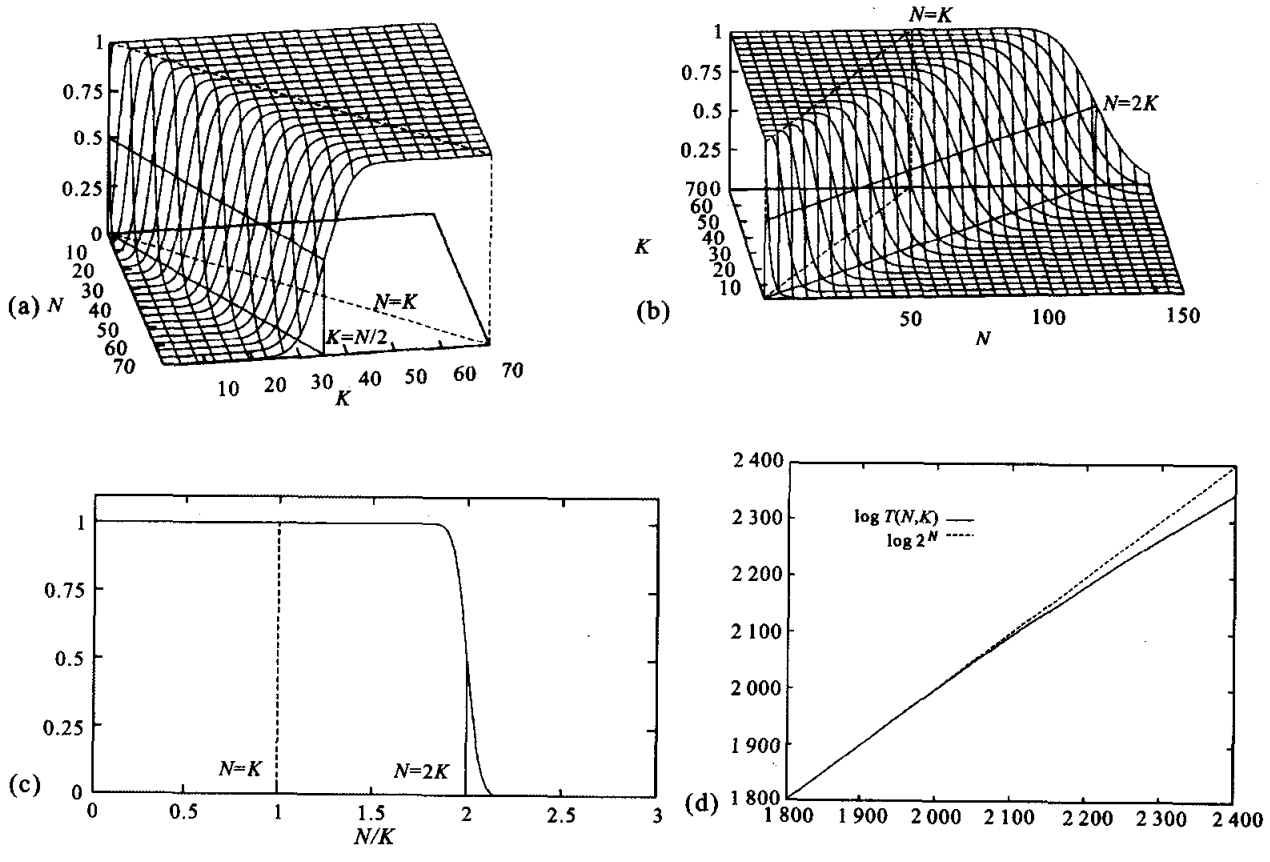


图 40.9 从各种角度表示，在 K 维空间中，N 个点上的函数是线性阈值函数的比例 $T(N,K)/2^N$。从(a)中可以看出，与 K 的依存关系可近似为一个误差函数，它在 $K=N/2$ 时通过 0.5；在 $K=N$ 时，比例达到 1。从(b)看出对 N 的依存关系，到 $N=K$ 为止一直为 1，在 $N=2K$ 时急剧下降。子图(c)显示在 $K=1\,000$ 时对 N/K 的依存关系。在 $N=2K$ 时可实现标记的比例急剧下降。子图(d)显示在 $K=1\,000$ 时，作为 N 的函数的 $\log_2 T(N,K)$ 值和 $\log_2 2^N$ 值。这些图是通过误差函数近似 $T/2^N$ 而画出的。

其中 $\Phi(z) \equiv \int_{-\infty}^{z} \exp(-z^2/2)/\sqrt{2\pi}\mathrm{d}z$。图 40.9 表示可实现比例 $T(N,K)/2^N$，以 N 和 K 的函数表示。要牢记的消息如图 40.9(c)所示：尽管比例 $T(N,K)/2^N$ 在 $N>K$ 时小于 1，直到 $N=2K$ 时小于 1 的幅度都可以忽略不计；此后，会急剧下降到零，于是在 $N>2K$ 时，只有一小部分二进制标记能够通过阈值函数实现。

结论

对于大的 K 值，一个线性阈值神经元的容量是每个权重 2 比特。

一个单神经元几乎肯定能准确地记忆不多于 $N=2K$ 个随机二进制标记，但是要再多记忆一些就几乎注定会失败。

40.4 进一步的习题

▷**习题 40.4**[♂] 二维空间上 $2N$ 个不同点所组成的有限集合是否能被一条直线分成两部分。

- 如果这些点在一般位置上？
- 如果这些点不在一般位置上？

在 K 维空间中的 $2N$ 个点能否被 $K-1$ 维的超平面分成两部分？



习题 40.5[♂] 在一个球的表面上随机地选取 4 个点。所有的点都位于同一半球上的概率是多少？这个问题与 $T(N,K)$ 有何关系？



习题 40.6[♂] 考虑 $K=3$ 维空间上的二进制阈值神经元和点集 $\{\boldsymbol{x}\}=\{(1,0,0),(0,1,0),(0,0,1),(1,1,1)\}$。求一个参数向量 $\boldsymbol{w}$ 使得神经元能够记住标记：(a) $\{t\}=\{1,1,1,1\}$；(b) $\{t\}=\{1,1,0,0\}$。

找出一个不可实现的标记 $\{t\}$。

▷**习题 40.7**[♂] 在这一章中，限定所有超平面都穿过原点。在这个习题中，去掉这一约束条件。

一般位置上的 N 条直线能在一个平面上产生多少个区域？3 条直线的示例情形如图 40.10 所示。

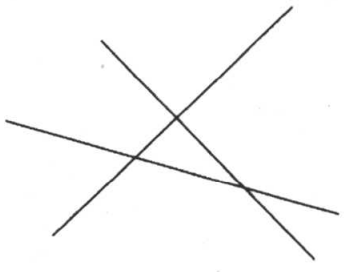


图 40.10 一个平面中的 3 条直线产生 7 个区域



习题 40.8[♂] 请用比特来估计你在实际生活中所得到的感官经验——视觉信息、听觉信息等。估计你已记住多少信息。估计莎士比亚作品中的信息含量。把这些信息量与你

大脑的容量进行比较:假设你有 10^{11} 个神经元,每个神经元有 1 000 个突触连接,并且一个神经元的容量结果(每个连接 2 比特)可以适用。你的大脑已装满了吗?

▷ **习题 40.9**[♪] 从通信信道的角度来看,一个尖峰神经元轴突的容量是每秒多少比特?[参见 Mackay 和 McCulloch(1952 年)关于这个问题的早期作品。]乘以视觉神经中的轴突数(约为 10^6)或耳蜗神经中的轴突数(每个耳朵大约是 50 000 个)后,重新估计获得感官经验的速度。

40.5 习题解答

习题 40.5 的解答 所有 4 个点位于同一半球面上的概率是

$$T(4,3)/2^4 = 14/16 = 7/8 \tag{40.11}$$

第 41 章

以学习作推理

41.1 用于推理的神经网络学习

在第 39 章中，通过最小化如下目标函数而将一个简单神经网络训练为分类器。

$$M(\boldsymbol{w}) = G(\boldsymbol{w}) + \alpha E_{\boldsymbol{w}}(\boldsymbol{w}) \tag{41.1}$$

该目标函数包括一个误差函数

$$G(\boldsymbol{w}) = -\sum_{n}[t^{(n)}\ln y(\boldsymbol{x}^{(n)};\boldsymbol{w}) + (1-t^{(n)})\ln(1-y(\boldsymbol{x}^{(n)};\boldsymbol{w}))] \tag{41.2}$$

和一个正则化因子

$$E_{\boldsymbol{w}}(\boldsymbol{w}) = \frac{1}{2}\sum_{i} w_i^2 \tag{41.3}$$

可以对这个神经网络的学习过程做出下述概率解释。

按其表面含义，将神经元的输出 $y(\boldsymbol{x};\boldsymbol{w})$ 解释为定义（当它的参数 $\boldsymbol{w}$ 被指定后）输入 $\boldsymbol{x}$ 属于类 $t=1$ 而非 $t=0$ 的概率。因此 $y(\boldsymbol{x};\boldsymbol{w}) \equiv P(t=1|\boldsymbol{x},\boldsymbol{w})$。于是，每一个 $\boldsymbol{w}$ 值都定义了关于类 1 相对于类 0 的概率的一个不同假设，以 $\boldsymbol{x}$ 的函数表示。

定义被观测数据 D 为目标值 $\{t\}$——假设输入 $\{\boldsymbol{x}\}$ 是已知的，且不会被建模。为了由已知数据推断 $\boldsymbol{w}$，需要一个似然函数和 $\boldsymbol{w}$ 的一个先验概率。似然函数度量参数 $\boldsymbol{w}$ 对被观测数据的预测有多好；它等于参数被设定为 $\boldsymbol{w}$ 的模型为被观测 t 值所赋予的概率。现将两个方程

$$\begin{aligned} P(t=1|\boldsymbol{w},\boldsymbol{x}) &= y \\ P(t=0|\boldsymbol{w},\boldsymbol{x}) &= 1-y \end{aligned} \tag{41.4}$$

重写成一个方程

$$P(t|\boldsymbol{w},\boldsymbol{x}) = y^t(1-y)^{1-t} = \exp[t\ln y + (1-t)\ln(1-y)] \tag{41.5}$$

所以可将误差函数 G 解释为负的对数似然

$$P(D|\boldsymbol{w}) = \exp[-G(\boldsymbol{w})] \tag{41.6}$$

类似地，正则化因子可通过参数上的一个对数先验概率分布来解释。

$$P(\boldsymbol{w}|\alpha)=\frac{1}{Z_W(\alpha)}\exp(-\alpha E_W) \tag{41.7}$$

如果 E_W 是上面所定义的二次型，则相应的先验分布是一个方差为 $\sigma_W^2=1/\alpha$ 的高斯分布，并且 $1/Z_W(\alpha)$ 等于 $(\alpha/2\pi)^{K/2}$，其中 K 是向量 $\boldsymbol{w}$ 中参数的个数。

于是，目标函数 $M(\boldsymbol{w})$ 对应于已知数据时对参数 $\boldsymbol{w}$ 的推理。

$$P(\boldsymbol{w}|D,\alpha)=\frac{P(D|\boldsymbol{w})P(\boldsymbol{w}|\alpha)}{P(D|\alpha)} \tag{41.8}$$

$$=\frac{e^{-G(\boldsymbol{w})}e^{-\alpha E_W(\boldsymbol{w})}/Z_W(\alpha)}{P(D|\alpha)} \tag{41.9}$$

$$=\frac{1}{Z_M}\exp(-M(\boldsymbol{w})) \tag{41.10}$$

所以通过（局部）最小化 $M(\boldsymbol{w})$ 而得的 $\boldsymbol{w}$ 可被理解为（局部）最可能的参数向量 $\boldsymbol{w}^*$。从现在起，把 $\boldsymbol{w}^*$ 当做 $\boldsymbol{w}_{\mathrm{MP}}$。

为什么可以自然而然地把误差函数解释为对数概率？误差函数通常是可加的。例如，G 是信息量的和，E_W 是权重的平方和。而另一方面，概率是可乘的：对于独立事件 X 和 Y，联合概率是 $P(x,y)=P(x)P(y)$。对数映射保持了这种对应性。

把 $M(\boldsymbol{w})$ 解释为一个对数概率有很多好处，稍后会对其中的几个进行讨论。

41.2 有两个权重的神经元的例子

对于一个神经元只有两个输入而没有偏置的情况

$$y(\boldsymbol{x},\boldsymbol{w})=\frac{1}{1+e^{-(w_1x_1+w_2x_2)}} \tag{41.11}$$

可以画出 $\boldsymbol{w}$ 的后验概率，$P(\boldsymbol{w}|D,\alpha)\propto\exp(-M(\boldsymbol{w}))$。假设接收到一些数据，如图41.1中左列所示。每个数据点由一个二维输入向量 $\boldsymbol{x}$ 和一个 t 值所组成，分别用×（$t=1$）或□（$t=0$）表示。以 $\boldsymbol{w}$ 函数表示的似然函数 $\exp(-G(\boldsymbol{w}))$ 如第二列所示。它是形如式(41.11)的函数的乘积。

传统学习算法的产物是 $\boldsymbol{w}$ 空间中的一个点，即最大化后验概率密度的估计值 $\boldsymbol{w}^*$。相反的，从贝叶斯观点来看，学习算法的产物是可能参数值的一个总体（图41.1的右下部）。我们不是选择一个特定的假设 $\boldsymbol{w}$，而是计算它们的后验概率。将似然函数同 $\boldsymbol{w}$ 空间上的一个先验分布（如图41.1右上方所示的一个宽的高斯分布）相乘，就得到后验分布。后验总体（在一个乘法常数之内）如图41.1第3列所示，而第4列是一幅等势图。随着数据量的增加（自顶向下），后验总体越来越聚集在最可能的值 $\boldsymbol{w}^*$ 周围。

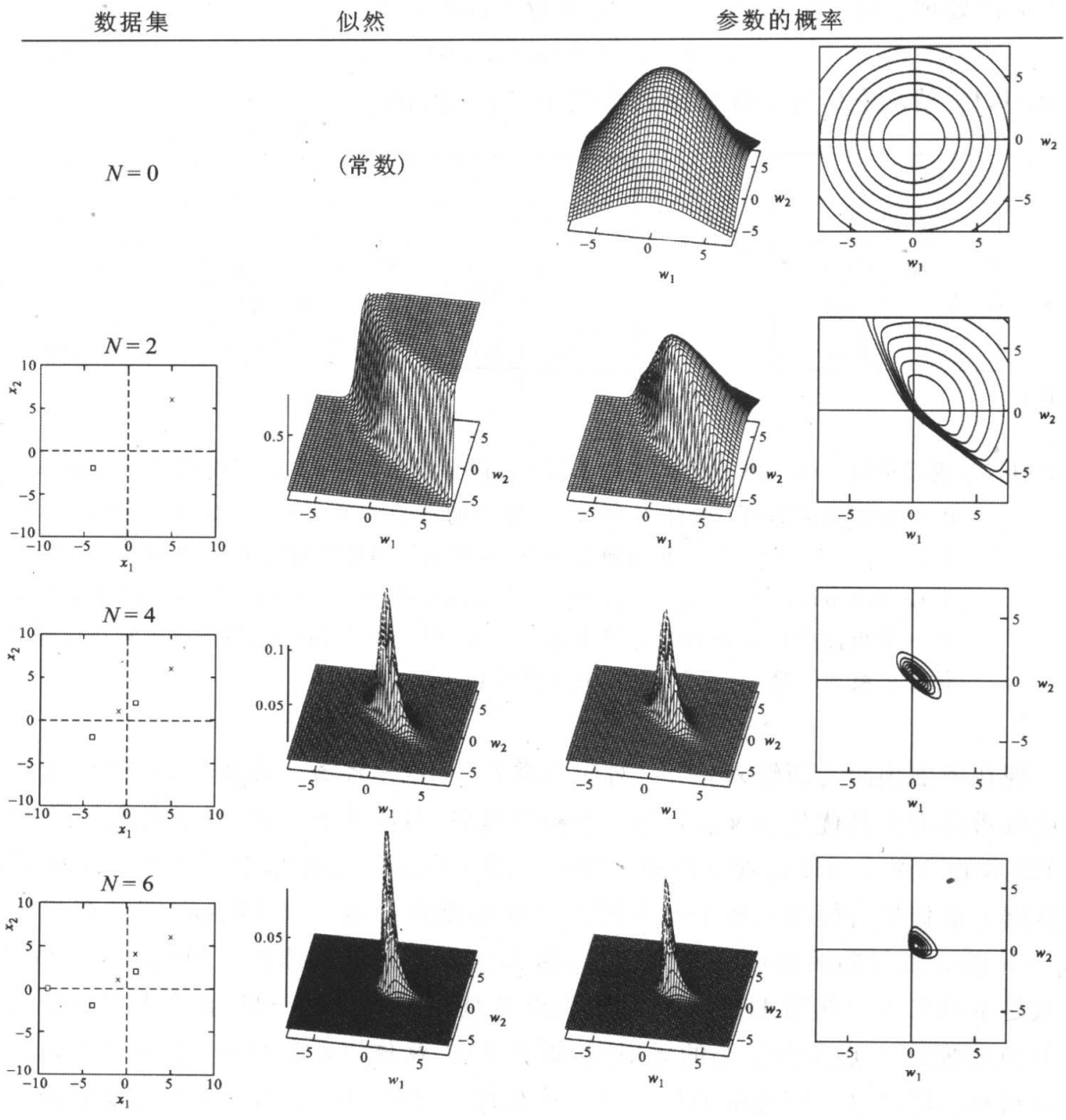


图 41.1 传统神经网络学习的贝叶斯解释和推广。在数据到达时,概率分布在参数空间上的演化。

41.3 优化之外:进行预测

现在考虑通过第 39.3 节中被训练成分类器的神经元来进行预测这一任务。这是具有两个输入、一个偏置的神经元。

$$y(\boldsymbol{x},\boldsymbol{w}) = \frac{1}{1+\mathrm{e}^{-(w_0+w_1x_1+w_2x_2)}} \tag{41.12}$$

在上一次处理它时，是通过最小化目标函数来训练它的。

$$M(w) = G(w) + \alpha E(w) \tag{41.13}$$

在 $\alpha = 0.01$ 情况下所得的优化函数如图 41.2(a)所示。

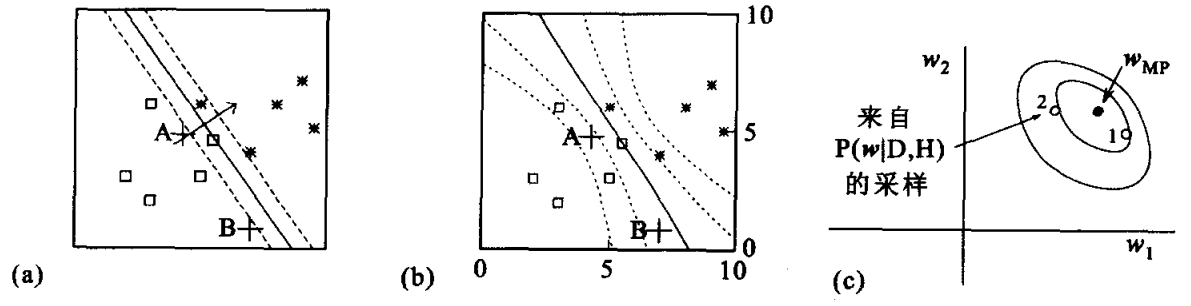


图 41.2 进行预测。(a) 由一个优化神经元 w_{MP} 执行的函数(用 3 条等势线表示)，用 $\alpha = 0.01$ 的权衰减训练(来自图 39.6)。所示的等势线对应于 $a = 0, \pm 1$，即 $y = 0.5$, 0.27 和 0.73。(b) 这些预测更合理吗？(所示的等势线对应于 $y = 0.5, 0.27, 0.73$, 0.12 和 0.88。)(c) w 的后验概率(示意性)；子图(b) 所示的贝叶斯预测是通过对每一个可能的权值 w 所得的预测取平均值，每个 w 值所占份额正比于它在后验总体上的概率。第 41.4 节描述了用来产生(b)的方法。

现在考虑预测与新输入 $x^{(N+1)}$ 对应的类 $t^{(N+1)}$ 这一任务。在进行预测时，通常都使用权重固定在其优化值 w_{MP} 上的一个神经网络，但这并非最优，通过考虑如图 41.2(a)所示的预测就可以直观地得知。这些预测是合理预测吗？考虑到达点 A 和 B 的新数据。最佳拟合模型以概率 0.2 把两个样本都划分为类 1，因为它们具有相同的 $w_{MP} \cdot x$ 值。如果确实知道 w 等于 w_{MP}，那么这些预测是正确的。但我们不知道 w。参数是不确定的。在图 41.2(b)中，因为点 B 离训练数据远一些，直观上也许倾向于给 B 点分配比 A 点少些信心的概率(接近 0.5)。最佳拟合参数 w_{MP} 往往给出过于自信的预测。解决这个问题的非贝叶斯方法是用一个经验确定的因子均匀地减弱所有预测(Copas，1983 年)。这样做并不理想，因为直觉告诉我们 B 点的预测强度应该比 A 点弱得多。一个贝叶斯观点可以有助于理解问题的症结，并提供了一个简单直接的解答。简而言之，通过考虑整个后验总体得到贝叶斯预测，如示意图 41.2(c)所示。

一个新的数据 $t^{(N+1)}$ 的贝叶斯预测涉及对参数(以及其他不能确定的量)进行边缘化。为了简单起见，假设权重 w 是唯一不确定量——假设权衰减率 α 和模型 $\mathcal{H}$ 本身都是固定的。那么根据求和法则，在位置 $x^{(N+1)}$ 处的一个新目标 $t^{(N+1)}$ 的预测概率是

$$P(t^{(N+1)} \mid x^{(N+1)}, D, \alpha) = \int \mathrm{d}^K w P(t^{(N+1)} \mid x^{(N+1)}, w, \alpha) P(w \mid D, \alpha) \tag{41.14}$$

其中 K 是 w 的维数，在这个简化问题中等于 3。于是，通过对每个可能的 w 的预测进行加权，就得到预测

$$P(t^{(N+1)}=1 \mid \boldsymbol{x}^{(N+1)}, \boldsymbol{w}, \alpha)=y(\boldsymbol{x}^{(N+1)}; \boldsymbol{w}) \tag{41.15}$$
$$P(t^{(N+1)}=0 \mid \boldsymbol{x}^{(N+1)}, \boldsymbol{w}, \alpha)=1-y(\boldsymbol{x}^{(N+1)}; \boldsymbol{w})$$

其中权重由 $\boldsymbol{w}$ 的后验概率 $P(\boldsymbol{w} \mid D, \alpha)$ 给出。最近在方程(41.10)中给出了这个后验概率,它等于

$$P(\boldsymbol{w} \mid D, \alpha)=\frac{1}{Z_M}\exp(-M(\boldsymbol{w})) \tag{41.16}$$

其中

$$Z_M=\int \mathrm{d}^K \boldsymbol{w} \exp(-M(\boldsymbol{w})) \tag{41.17}$$

总之,如果能够找到方法计算如下积分,就可以得到贝叶斯预测。

$$P(t^{(N+1)}=1 \mid \boldsymbol{x}^{(N+1)}, D, \alpha)=\int \mathrm{d}^K \boldsymbol{w} y(\boldsymbol{x}^{(N+1)}; \boldsymbol{w}) \frac{1}{Z_M}\exp(-M(\boldsymbol{w})) \tag{41.18}$$

这是在 $\boldsymbol{w}$ 的后验分布下,神经元在点 $\boldsymbol{x}^{(N+1)}$ 上的平均输出。

实现

应该如何计算积分(41.18)?对于简化问题,权空间是三维的;而对于实际的神经网络而言,维数 K 可能高达数千。

要实现对一般数据建模问题的贝叶斯推理,可以采用精确法(第 25 章)、蒙特卡罗采样法(第 29 章)或者是确定性的近似方法,例如采用拉普拉斯方法(第 27 章)或变参法(第 33 章)对 $P(\boldsymbol{w} \mid D, \alpha)$ 进行高斯近似。对于神经网络几乎没有精确的方法。对神经网络实现贝叶斯推断的两种主要方法是 Neal(1996 年)的蒙特卡罗方法和 MacKay(1991 年)的高斯近似方法。

41.4 单神经元的蒙特卡罗实现

首先,使用蒙特卡罗方法,在这种方法中,积分(41.18)可以这样计算:把 $y(\boldsymbol{x}^{(N+1)}; \boldsymbol{w})$ 作为 $\boldsymbol{w}$ 的函数 f,用下式计算其均值。

$$\langle f(\boldsymbol{w}) \rangle \simeq \frac{1}{R}\sum_r f(\boldsymbol{w}^{(r)}) \tag{41.19}$$

其中 $\{\boldsymbol{w}^{(r)}\}$ 是从后验分布 $\frac{1}{Z_M}\exp(-M(\boldsymbol{w}))$ 所得的样本(比较式(29.6))。用 Metropolis 方法得到样本(第 29.4 节)。附加说一句,这个蒙特卡罗方法的一个可能的缺点是:如果这种不可能事件和不可能参数值的联合很有可能发生,它不能有效地估计一个不太可能事件的概率,即非常接近于零的 $P(t \mid D, \mathcal{H})$。

如何生成样本 $\{\boldsymbol{w}^{(r)}\}$?Radford Neal 把哈密尔顿蒙特卡罗方法引入神经网络。在第 30 章遇到了这一复杂 Metropolis 方法,它利用了梯度信息。现在所采用的是简化版哈密尔顿蒙特卡罗方法,称为 Langevin 蒙特卡罗方法。

Langevin 蒙特卡罗方法

Langevin 方法(算法 41.4)可以归纳为"带有叠加噪声的梯度下降法",如图41.3所示。一个噪声向量 $\boldsymbol{p}$ 由单位方差的一个高斯分布生成。计算梯度 $\boldsymbol{g}$ 就得到 $\boldsymbol{w}$ 的步长

$$\Delta \boldsymbol{w} = -\frac{1}{2}\varepsilon^2 \boldsymbol{g} + \varepsilon \boldsymbol{p} \tag{41.20}$$

注意:如果忽略 εp 项,这就成了学习速率为 $\eta = \frac{1}{2}\varepsilon^2$ 的梯度下降法。根据目标函数 $M(\boldsymbol{w})$ 的值和梯度的变化,以一定的概率接受或拒绝 $\boldsymbol{w}$ 的步长,接受概率以保持详细平衡为准。

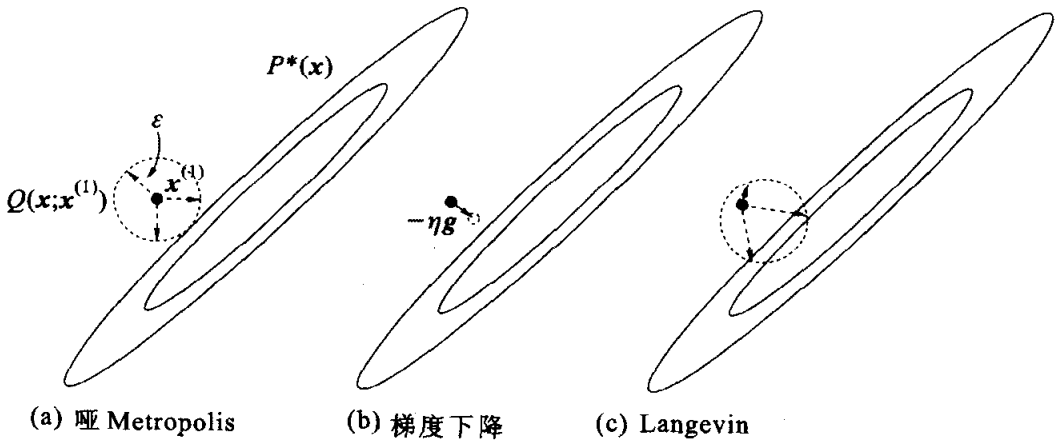


图 41.3 在二维空间中 Langevin 方法中的一步(c),与传统"哑"Metropolis 方法(a)和梯度下降(b)进行对比。Langevin 方法中的提议密度由"带噪声的梯度下降"给出。

Langevin 方法有一个自由参数 ε,它控制典型的步长大小。如果 ε 值被设置得太大,可能会拒绝动作。如果设置成过小的值,则在状态空间中的进程将十分缓慢。

Langevin 法示例

Langevin 方法如图 41.5、图 41.6 和图 41.7 所示。这里的目标函数是 $M(\boldsymbol{w}) = G(\boldsymbol{w}) + \alpha E_{\boldsymbol{w}}(\boldsymbol{w})$,其中 $\alpha = 0.01$。为了进行比较,这些图中包括了前面对同样目标函数所用梯度下降法优化的结果(如图 39.6 所示)。可以看出,$\boldsymbol{w}$ 的平均演化进程与梯度下降法中参数的演化进程相似。经过大约 10 000 次迭代之后,可以看出蒙特卡罗方法收敛到了后验分布。

```
g = gradM ( w ) ;                                  # 初始化 w,并设置梯度
M = findM ( w ) ;                                  # 设置目标函数
for l = 1:L                                        # 循环 L 次
    p = randn ( size(w) ) ;                        # 初始动量为 Normal(0,1)
    H = p'* p / 2 + M ;                            # 计算 H(w,p)

* p = p - epsilon * g / 2 ;                        # 产生 p 的半步步长
* wnew = w + epsilon * p ;                        # 产生步长 w
* gnew = gradM ( wnew ) ;                         # 求出新梯度
* p = p - epsilon * gnew / 2 ;                    # 产生 p 的半步步长

    Mnew = findM ( wnew ) ;                        # 搜索新的目标函数
    Hnew = p'* p / 2 + Mnew ;                      # 计算新的 H 值
    dH = Hnew - H ;                                # 决定是否接受
    if ( dH < 0 ) accept = 1 ;
    elseif ( rand( ) < exp( -dH) ) accept = 1 ;    # 与均匀变量进行比较
    else accept = 0 ;
    endif
    if( accept )g = gnew;w = wnew; M = Mnew;endif
endfor

function gM = gradM ( w )                          # 目标函数的梯度
    a = x * w ;                                    # 计算激活值
    y = sigmoid(a) ;                               # 计算输出
    e = t - y ;                                    #计算误差
    g = -x'* e ;                                   # 计算 G(w)的梯度
    gM = alpha * w + g ;
endfunction

function M = findM ( w )                           #目标函数
    G = -(t'* log(y) + (1-t') * log( 1-y )) ;
    EW = w'* w / 2 ;
    M = G + alpha * EW ;
endfunction
```

算法 41.4 Langevin 蒙特卡罗方法的 Octave 源代码。为了得到哈密尔顿蒙特卡罗方法,多次重复 4 行带 * 号标记的程序(算法 41.8)。

这次仿真中的平均接受率是 93%,所提议的移动只有 7% 被拒绝。如果采用大一些的步长 ε,状态空间中的进程也许会快一些,但是所选择的步长要使“下降率” $\eta = \frac{1}{2}\varepsilon^2$ 与先前的仿真步长相匹配。

进行贝叶斯预测

从迭代 10 000 到 40 000,每 1 000 次迭代对权重采样一次,相应的 x 函数绘制在图 41.6 中。可能函数的变化范围很大。通过将这 30 个 x 的函数做平均,得到贝叶

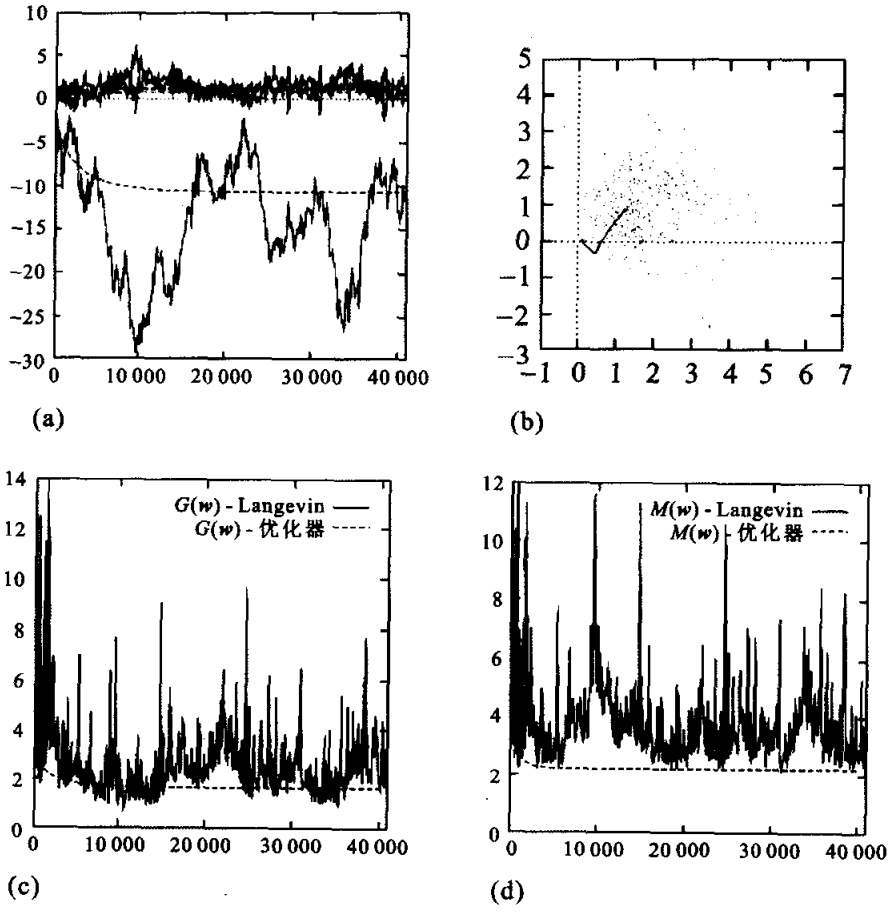


图41.5 采用Langevin蒙特卡罗方法的单神经元学习。(a) 权重 w_0、w_1 和 w_2 以迭代次数函数所表示的演化。(b) 权重 w_1 和 w_2 在权空间的演化。同时显示的还有一条直线,它是使用图39.6的优化器时权重的演化。(c) 误差函数 $G(w)$ 以迭代次数的函数表示。同时显示的还有图39.6优化过程中的误差函数。(d) 目标函数 $M(w)$ 以迭代次数的函数表示。也可参见图41.6和图41.7。

斯预测的蒙特卡罗近似。结果显示在图41.7中,并与优化参数所对应的预测进行比较。可见当离开高数据密度区域时,贝叶斯预测还是比较合适的。

贝叶斯分类器能够更好地辨识类别不确定的点。这个令人高兴的特性仅得益于对概率规则的机械应用。

优化与典型性

最后的观察是对蒙特卡罗采样过程函数 $G(w)$ 和 $M(w)$ 的特性与最优权 w_{MP} 时 G 和 M 的值进行比较(如图41.5所示)。函数 $G(w)$ 在 $G(w_{MP})$ 的值周围波动,尽管不是以对称方式。函数 $M(w)$ 也在波动,但是没有围绕 $M(w_{MP})$——显然也是不可能的,因为 M 在 w_{MP} 处达到最小,所以 M 不可能变得更小——而且,M 在接近

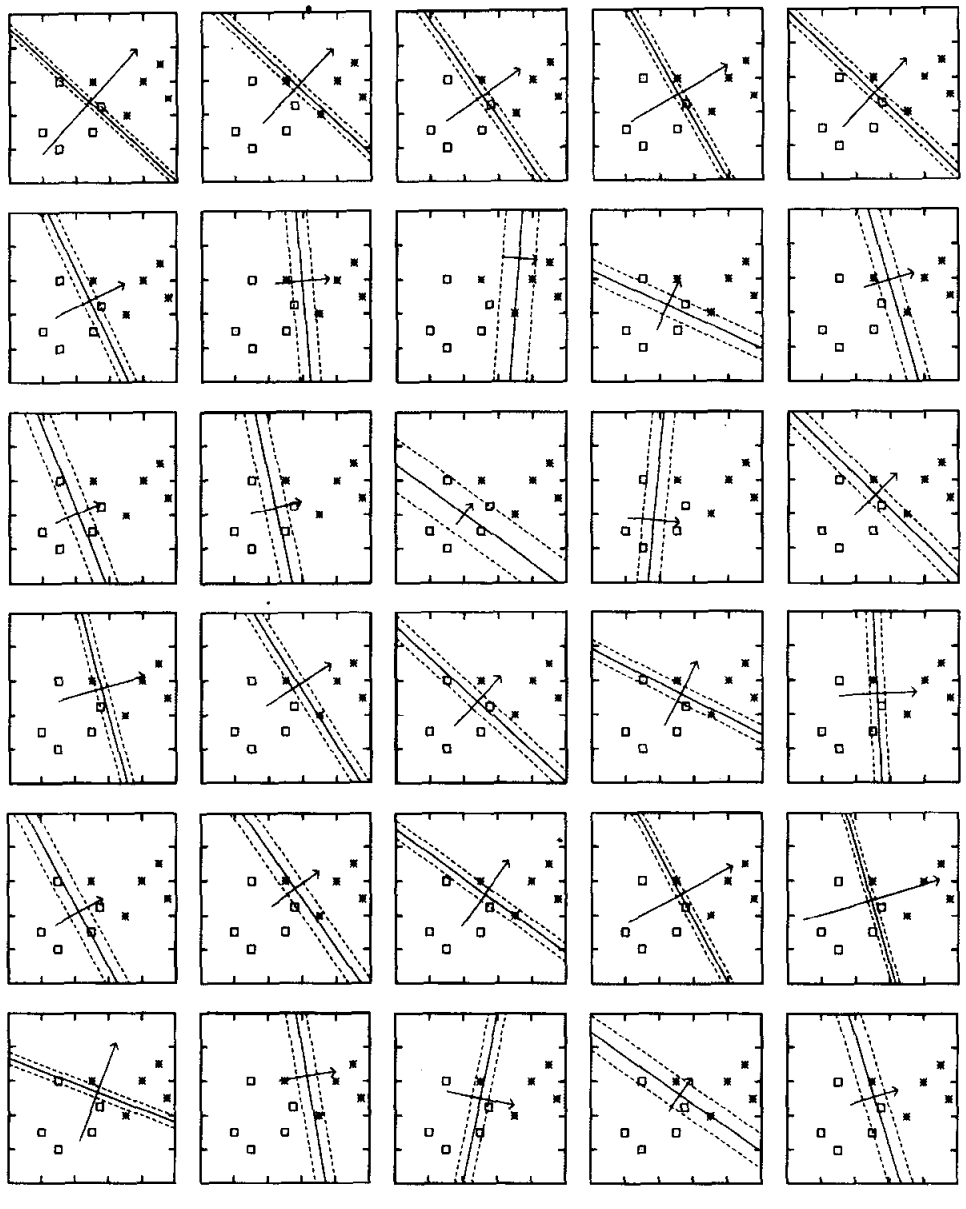


图 41.6 采用 Langevin 蒙特卡罗方法得到的样本。学习速率设定为 $\eta = 0.01$，权衰减速率设定为 $\alpha = 0.01$。步长为 $\varepsilon = \sqrt{2\eta}$。从 10 000 到 40 000 次迭代，用每1 000次迭代的 3 条等势线来表示神经元实现函数。等势线对应于 $a = 0, \pm 1$，即 $y = 0.5, 0.27$ 和 0.73。同时显示的是正比于 (w_1, w_2) 的向量。

$M(\boldsymbol{w}_{\mathrm{MP}})$时几乎不下降。用信息论的语言来说就是，$\boldsymbol{w}$ 的典型集与最可能状态 $\boldsymbol{w}_{\mathrm{MP}}$性质不同。

因此，得出一般性的消息——适用于所有数据模型，而不仅仅是神经网络：使用优化参数时必须谨慎，因为优化参数的性质也许并不代表典型的可能参数的性质；采用优化参数所得的预测往往是过度自信到不合理的程度。

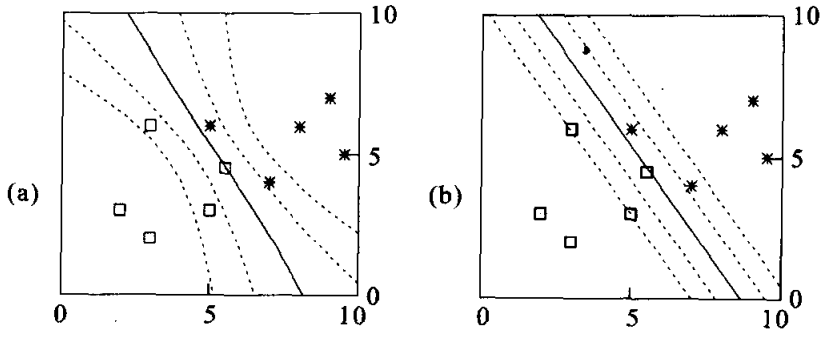


图 41.7 采用 Langevin 蒙特卡罗方法的贝叶斯预测，与优化参数法的预测进行比较。(a) 在迭代 10 000 和 40 000 之间平均分布的 30 个样本预测进行平均后所得的预测函数，如图 41.6 所示。等势线对应于 $a=0,\pm1,\pm2$，即 $y=0.5,0.27,0.73,0.12$ 和 0.88。(b) 为了进行比较，通过优化 $M(\boldsymbol{w})$，神经元参数设定为最可能值时的预测。

使用哈密尔顿蒙特卡罗方法减少随机游动行为

作为蒙特卡罗方法的最后探讨，现在把 Langevin 蒙特卡罗方法和其“大哥”哈密尔顿蒙特卡罗方法进行比较。如算法 41.8 所示，容易实现到哈密尔顿蒙特卡罗方法的改动。每个提议利用沿 $\boldsymbol{w},\boldsymbol{p}$ 空间动态轨迹的多次梯度估值，其中 $\boldsymbol{p}$ 是 Langevin 和哈密尔顿蒙特卡罗方法中的额外“动量”变量。每条轨迹的步数“Tau”都被随机地设置在 100 和 200 之间。步长 ε 保持固定，以便与以前的仿真结果有可比性；然而在实际应用中还是建议采用随机的步长。

```
wnew = w ;
gnew = g ;
for tau = 1:Tau

    p = p - epsilon * gnew / 2 ;          # 产生 p 的半步步长
    wnew = wnew + epsilon * p ;           # 产生步长 w

    gnew = gradM ( wnew ) ;               # 求出新梯度
    p = p - epsilon * gnew / 2 ;          # 产生 p 的半步步长

endfor
```

算法 41.8 哈密尔顿蒙特卡罗方法的 Octave 源代码。除了用这里给出的程序段替代带有 * 的 4 行之外，该算法与算法 41.4 中的 Langevin 法相同。

图 41.9 比较了 Langevin 和哈密尔顿蒙特卡罗方法的采样特性。随着仿真的进行，哈密尔顿蒙特卡罗仿真中状态的自相关性的衰减比 Langevin 仿真中要快得多。

对于这个简化问题,哈密尔顿蒙特卡罗方法在计算机机时利用效率方面至少高 10 倍。

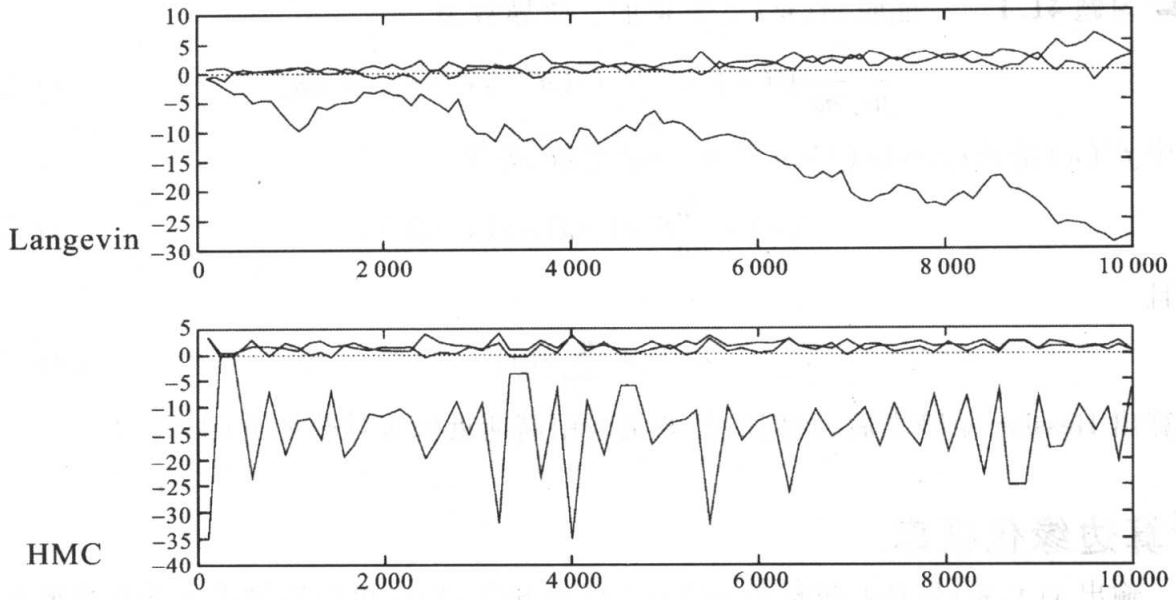


图 41.9 Langevin 蒙特卡罗法和哈密尔顿蒙特卡罗法(HMC)的采样特性比较。水平轴为计算梯度值的次数。每幅图表示前 10 000 次迭代时的权重。哈密尔顿蒙特卡罗仿真中的拒绝率是 8%。

41.5 通过高斯近似实现推理

物理学家喜欢选择非线性并对其进行局部线性化,他们喜欢用高斯分布来近似概率分布。这样的近似给出了求解如下积分问题的一种替代策略。

$$P(t^{(N+1)} = 1 \mid x^{(N+1)}, D, \alpha) = \int \mathrm{d}^K \boldsymbol{w} y(\boldsymbol{x}^{(N+1)}; \boldsymbol{w}) \frac{1}{Z_M} \exp(-M(\boldsymbol{w})) \quad (41.21)$$

刚才用蒙特卡罗方法对它估值。

从高斯近似后验概率开始。求出 $M(\boldsymbol{w})$ 的最小值(采用基于梯度的优化器),并在该处对 M 进行泰勒级数展开

$$M(\boldsymbol{w}) \simeq M(\boldsymbol{w}_{\mathrm{MP}}) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{\mathrm{MP}})^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{w} - \boldsymbol{w}_{\mathrm{MP}}) + \cdots \quad (41.22)$$

其中 $\boldsymbol{A}$ 是二阶导数矩阵,也称为 Hessian 矩阵,定义为

$$A_{ij} \equiv \frac{\partial^2}{\partial w_i \partial w_j} M(\boldsymbol{w}) \Big|_{\boldsymbol{w} = \boldsymbol{w}_{\mathrm{MP}}} \quad (41.23)$$

因此,定义高斯近似为

$$Q(\boldsymbol{w}; \boldsymbol{w}_{\mathrm{MP}}, \boldsymbol{A}) = [\det(\boldsymbol{A}/2\pi)]^{1/2} \exp\left[-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{\mathrm{MP}})^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{w} - \boldsymbol{w}_{\mathrm{MP}}) \right] \quad (41.24)$$

可以将矩阵 $\boldsymbol{A}$ 看做定义 $\boldsymbol{w}$ 的误差限。更准确地说,Q 是一个正态分布,其方差 - 协

方差矩阵是 $\boldsymbol{A}^{-1}$。



习题 41.1[⌀] 证明 $M(\boldsymbol{w})$ 关于 $\boldsymbol{w}$ 的二阶导数为

$$\frac{\partial^2}{\partial w_i \partial w_j} M(\boldsymbol{w}) = \sum_{n=1}^{N} f'(a^{(n)}) x_i^{(n)} x_j^{(n)} + \alpha \delta_{ij} \tag{41.25}$$

其中 $f'(a)$ 是 $f(a) \equiv 1/(1+\mathrm{e}^{-a})$ 的一阶导数，等于

$$f'(a) = \frac{\mathrm{d}}{\mathrm{d}a} f(a) = f(a)(1-f(a)) \tag{41.26}$$

而且

$$a^{(n)} = \sum_j w_j x_j^{(n)} \tag{41.27}$$

计算出 Hessian 矩阵之后，我们的任务是利用高斯近似来计算积分(41.21)。

计算边缘化概率

输出 $y(\boldsymbol{x};\boldsymbol{w})$ 只是通过标量 $a(\boldsymbol{x};\boldsymbol{w})$ 与 $\boldsymbol{w}$ 相关，所以可以通过求 a 的概率密度来减少积分的重数。假设在 $\boldsymbol{w} = \boldsymbol{w}_{\mathrm{MP}} + \Delta \boldsymbol{w}$ 上的局部高斯后验概率分布为 $P(\boldsymbol{w}|D,\alpha) \simeq (1/Z_Q)\exp\left(-\frac{1}{2}\Delta \boldsymbol{w}^{\mathrm{T}} \boldsymbol{A} \Delta \boldsymbol{w}\right)$。对于单神经元，激活值 $a(\boldsymbol{x};\boldsymbol{w})$ 是 $\boldsymbol{w}$ 的线性函数，并且 $\partial a/\partial \boldsymbol{w} = \boldsymbol{x}$，所以对于任何 $\boldsymbol{x}$，激活值 a 是高斯分布。

▷ **习题 41.2**[⌀] 假设 $\boldsymbol{w}$ 是均值为 $\boldsymbol{w}_{\mathrm{MP}}$、方差 - 协方差矩阵为 $\boldsymbol{A}^{-1}$ 的高斯分布，证明 $a(\boldsymbol{x})$ 的概率分布是

$$P(a|\boldsymbol{x},D,\alpha) = \mathrm{Normal}(a_{\mathrm{MP}}, s^2) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{(a-a_{\mathrm{MP}})^2}{2s^2}\right) \tag{41.28}$$

其中 $a_{\mathrm{MP}} = a(\boldsymbol{x}, \boldsymbol{w}_{\mathrm{MP}})$ 和 $s^2 = \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A}^{-1} \boldsymbol{x}$。

这意味着边缘化输出为

$$P(t=1 \mid \boldsymbol{x}, D, \alpha) = \psi(a_{\mathrm{MP}}, s^2) \equiv \int \mathrm{d}a f(a) \mathrm{Normal}(a_{\mathrm{MP}}, s^2) \tag{41.29}$$

应当将它与最可能的网络输出 $y(\boldsymbol{x};\boldsymbol{w}_{\mathrm{MP}}) = f(a_{\mathrm{MP}})$ 进行比较。一个 S 形函数与一个高斯函数之积的积分可以近似为

$$\psi(a_{\mathrm{MP}}, s^2) \simeq \phi(a_{\mathrm{MP}}, s^2) \equiv f(\kappa(s) a_{\mathrm{MP}}) \tag{41.30}$$

其中 $\kappa = 1/\sqrt{1+\pi s^2/8}$（如图 41.10 所示）。

示例

图 41.11 显示了在最优点 $\boldsymbol{w}_{\mathrm{MP}}$ 处拟合高斯近似的结果，及使用高斯近似和式(41.30)进行预测的结果。将这些预测与 Langevin 蒙特卡罗方法所得预测（如图 41.7 所示）相比，可以发现，尽管在定性上两者相同，但在数值上存在明显的差异。所以这两种方法中至少有一种是不完全准确的。

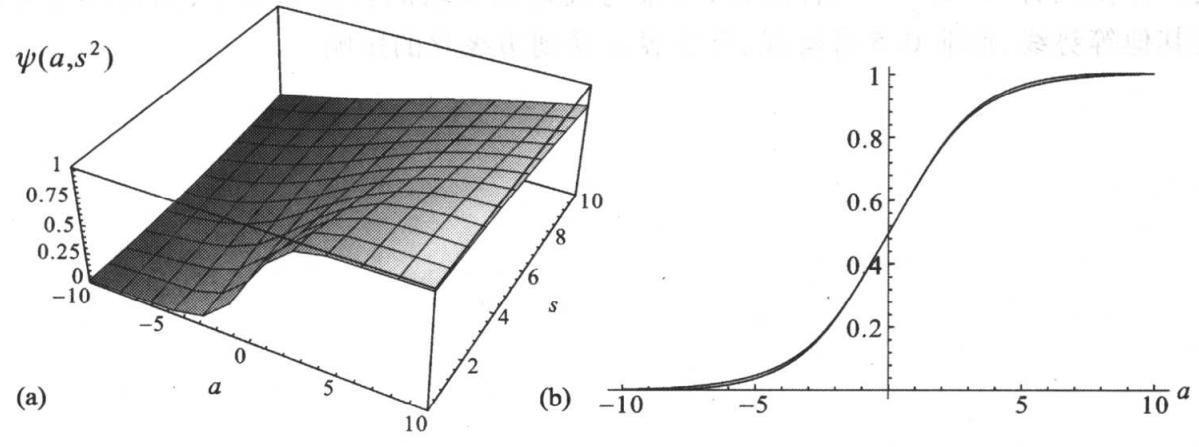


图 41.10 边缘化概率及其近似。(a) 函数 $\psi(a,s^2)$ 的数值计算。在(b)中,表示正文所定义 $s^2=4$ 时的函数 $\psi(a,s^2)$ 和 $\phi(a,s^2)$,以 a 的函数表示。来自 MacKay(1992b)。

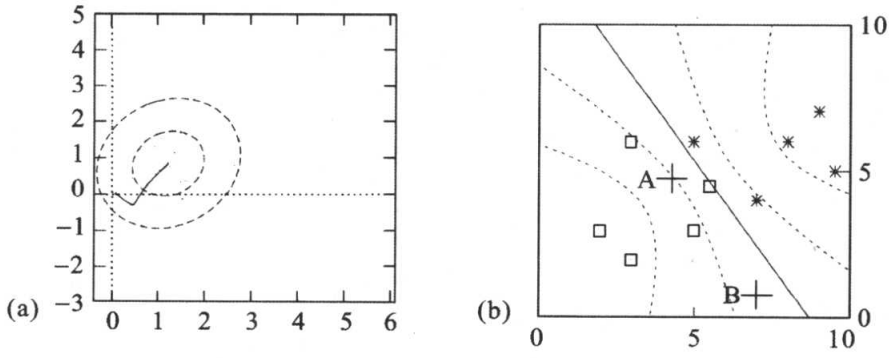


图 41.11 在权空间中的高斯近似以及输入空间中的近似预测。(a)高斯近似在权空间中(w_1,w_2)平面上的投影。同时也表示了 1 - 和 2 - 标准差的等势线。同时还有优化器的轨迹以及蒙特卡罗法的样例。(b) 从高斯近似和式(41.30)得到的预测函数(比较图 41.2)。

▷ **习题 41.3**[⌕] $P(\boldsymbol{w}|D,\alpha)$ 的高斯近似的拖尾是太大,还是太小,还是两者兼而有之?把 $P(\boldsymbol{w}|D,\alpha)$ 看做一个参数 w_i 的函数,并在对数尺度上考虑两个分布,这样做也许是有益的。讨论在什么条件下高斯近似最为准确。

为什么要边缘化?

如果将输出直接用于一个(0/1)决策,并且与误差相关联的代价是对称的,那么与使用最可能参数所给的输出相比,在这个高斯近似下使用边缘化输出,对分类器性能没有区别,因为两个函数在 $a_{\mathrm{MP}}=0$ 时都通过 0.5。但是,比如,如果除了可以说

“我猜测0”和“我猜测1”之外，还可以说“我不知道”，那么这些贝叶斯输出还是会不同。即使只有“0”和“1”两种选择，如果与误差相关联的代价不相等，则判决边界将是其他等势线，而非0.5等势线，且边界会受到边缘化的影响。

关于有监督神经网络的后记

我的一个学生罗伯特问我：

也许我还没有把握基础性的内容，但是有监督的神经网络似乎等价于用事先定义好的函数对已知的数据进行拟合，然后进行外推——有什么差别吗？

我赞成罗伯特的观点。至今已学习的有监督神经网络只是用于拟合数据的参数化非线性函数。你可能会赞同罗伯特的另一观点：

无监督网络似乎比有监督网络有意思得多。我很惊讶它会起作用！

第 42 章

Hopfield 网络

已经用 3 章篇幅来研究单神经元。现在可以把多个神经元连在一起，使一个神经元的输出是另一神经元的输入，从而形成神经网络。

根据连接特性，神经元可以分为两类。如图 42.1 所示。

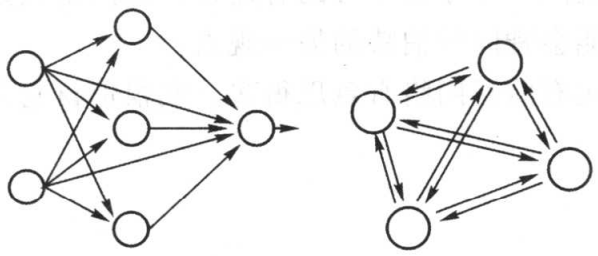


图 42.1 （a）前馈网络。（b）反馈网络。

前馈网络 在一个前馈网络中，所有连接使得网络形成一个有向无环图。

反馈网络 任何非前馈网络的网络称为反馈网络。

在这一章，将讨论一个叫做 Hopfield 网络的全连接反馈网络。在 Hopfield 网络中，权重被限定为对称的，即从神经元 i 到神经元 j 的权重等于从神经元 j 到神经元 i 的权重。

Hopfield 网络有两种应用。首先，可以用于联想记忆。其次，可以用于求解最优化问题。本章将首先讨论联想记忆的思想，也称为内容可寻址的记忆。

42.1 Hebbian 学习

在第 38 章中，讨论了传统数字记忆与生物记忆之间的对比。最显著的差别也许是生物记忆中的联想特性。

Donald Hebb（1949 年）提出的一个简单模型抓住了联想记忆的基本思想。假设在神经元之间的活跃性是正相关时，其权重就增加

$$\frac{\mathrm{d}w_{ij}}{\mathrm{d}t} \sim \text{相关性}(x_i, x_j) \tag{42.1}$$

假设出现激励 m（例如香蕉的气味），神经元 m 的活跃性就会增加；神经元 n 和另一

激励 n 相关联(例如看见一个黄色物体)。如果这两个激励——黄色视觉和香蕉气味——同时在环境中出现,则 Hebbian 学习规则(42.1)将增大权重 w_{nm} 和 w_{mn}。这意味在以后若单独出现激励 n,使得活跃性 x_n 较大,则从 n 到 m 的正权值也会使神经元 m 被激活。所以对黄色物体的视觉响应是对香蕉气味的自动联想,称之为“模式补足”。联想记忆的工作过程不要求教师出现。无需信号来表明已检测到相关性或者需要有联想。无监督的、局部的学习算法和无监督的、局部活跃性规则自发地产生联想记忆。

这些思想看起来如此简单和有效,所以肯定和大脑中记忆的工作原理有关。

42.2 二进制 Hopfield 网络定义

权重的约定 通常的约定还是用 w_{ij} 表示从神经元 j 到神经元 i 的连接。

体系结构 Hopfield 网络包含 I 个神经元。它们通过权重满足 $w_{ij}=w_{ji}$ 的双向、对称连接实现全连接。不存在自连接,因此对所有 i 都有 $w_{ii}=0$。有可能包括偏置 w_{i0}(可以看成连到神经元“0”的权重,该神经元的活跃性永远是 $x_0=1$)。用 x_i 表示神经元 i 的活跃性(它的输出)。

活跃性规则 大致地说,Hopfield 网络的活跃性规则就是每个神经元像具有阈值激活函数的单神经元一样更新自己的状态。

$$x(a)=\Theta(a)\equiv\begin{cases}1, & a\geqslant 0\\ -1, & a<0\end{cases} \tag{42.2}$$

因为 Hopfield 网络中存在反馈(每个神经元的输出是所有其他神经元的输入),必须指定状态更新的次序。更新可以是同步的或异步的。

同步更新——所有神经元都计算其激活值

$$a_i=\sum_j w_{ij}x_j \tag{42.3}$$

然后同时将状态更新为

$$x_i=\Theta(a_i) \tag{42.4}$$

异步更新——每次只有一个神经元计算激活值并更新状态。被选神经元次序可以是固定次序或随机次序。

Hopfield 网络的特性对上面的选择可能很敏感。

学习规则 学习规则是为了使一组理想的记忆 $\{\boldsymbol{x}^{(n)}\}$ 成为 Hopfield 网络活跃性规则的稳定状态。每个记忆都处于二进制模式,$x_i\in\{-1,1\}$。

权重通过外积之和或 Hebb 规则来设置。

$$w_{ij}=\eta\sum_n x_i^{(n)}x_j^{(n)} \tag{42.5}$$

其中 η 是一个无足轻重的常数。为了防止最大可能权重随 N 而增长,可能选择设置 $\eta=1/N$。



习题 42.1[↗] 解释为什么 η 值对上面定义的 Hopfield 网络不是一个重要参数。

42.3 连续 Hopfield 网络的定义

利用同样的体系结构和学习规则，可以定义活跃性为 -1 和 1 之间实数的 Hopfield 网络。

活跃性规则 Hopfield 网络的活跃性规则就是每个神经元像具有 S 形激活函数的单神经元一样更新自己的状态。更新可以是同步或异步的，并涉及方程

$$a_i = \sum_j w_{ij} x_j \tag{42.6}$$

和

$$x_i = \tanh(a_i) \tag{42.7}$$

学习规则和二进制 Hopfield 网络一样，但是与 η 值有关。或者可以固定 η，并在激活函数中引入一个增益 $\beta \in (0, \infty)$：

$$x_i = \tanh(\beta a_i) \tag{42.8}$$



习题 42.2[↗] 在前面什么地方遇见过式(42.6)、式(42.7)和式(42.8)？

42.4 Hopfield 网络的收敛性

希望所定义的 Hopfield 网络能够进行如图 42.2 所示意的联想回忆。希望 Hopfield 网络的活跃性规则会进行部分记忆或受损记忆，并通过模式补足或纠错来恢复其原始记忆。

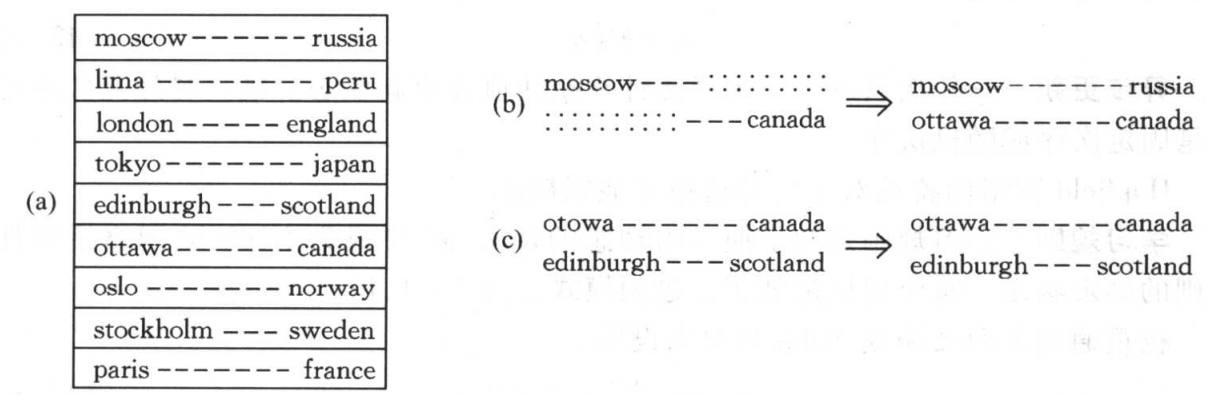


图 42.2 联想记忆(示意)。(a) 一列期望的记忆。(b) 联想记忆的第一目的是对已知的部分模式进行模式补足。(c) 记忆的第二目的是纠错。

但是为什么能期望有任何模式在活跃性规则下是稳定的，更不用说理想记忆？

本章只讨论连续 Hopfield 网络，因为二进制网络只是它的一个特例。在讨论变参法（第 33.2 节）时，就已经遇到过活跃性规则（42.6，42.8）：当时在近似一个自旋系统，其能量函数为

$$E(\boldsymbol{x};\boldsymbol{J}) = -\frac{1}{2}\sum_{m,n} J_{mn}x_m x_n - \sum_n h_n x_n \tag{42.9}$$

具有可分离的分布

$$Q(\boldsymbol{x};\boldsymbol{a}) = \frac{1}{Z_Q}\exp\Big(\sum_n a_n x_n\Big) \tag{42.10}$$

并对后者进行优化，以最小化变参自由能

$$\beta\tilde{F}(\boldsymbol{a}) = \beta\sum_x Q(\boldsymbol{x};\boldsymbol{a})E(\boldsymbol{x};\boldsymbol{J}) - \sum_x Q(\boldsymbol{x};\boldsymbol{a})\ln\frac{1}{Q(\boldsymbol{x};\boldsymbol{a})} \tag{42.11}$$

可以发现一对迭代方程

$$a_m = \beta\Big(\sum_n J_{mn}\bar{x}_n + h_m\Big) \tag{42.12}$$

和

$$\bar{x}_n = \tanh(a_n) \tag{42.13}$$

可以保证降低变参自由能

$$\beta\tilde{F}(\boldsymbol{a}) = \beta\Big(-\frac{1}{2}\sum_{m,n} J_{mn}\bar{x}_m\bar{x}_n - \sum_n h_n\bar{x}_n\Big) - \sum_n H_2^{(e)}(q_n) \tag{42.14}$$

如果简单地用 w 替代 J，x 替代 $\bar{x}$，w_{i0} 替代 h_n，可见 Hopfield 网络的方程与一组最小化式（42.15）的平均场方程相同。

$$\beta\tilde{F}(\boldsymbol{x}) = -\beta\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{x} - \sum_i H_2^{(e)}[(1+x_i)/2] \tag{42.15}$$

对于在系统动态演化过程中减小且有下界的函数，存在一个通用的名称：系统的 Lyapunov 函数。能够证明系统存在 Lyapunov 函数是很有用的：如果一个系统具有 Lyapunov 函数，则系统的动态过程必定会收敛到一个不动点（Lyapunov 函数的局部最小值）或者极限环（Lyapunov 函数在环上为常数）。具有 Lyapunov 函数的系统不会出现混沌行为。如果系统具有 Lyapunov 函数，那么状态空间可分成多个吸引域，每个域有一个吸引子。

所以，连续 Hopfield 网络的活跃性规则（如果异步地实现）具有一个 Lyapunov 函数。该 Lyapunov 函数是每个参数 a_i 的凸函数，所以 Hopfield 网络的动态过程总会收敛到一个稳定的不动点。

这个收敛性证明主要取决于 Hopfield 网络具有对称连接这一事实。它还取决于异步进行的更新。



习题 42.3[2] 构造一个例子证明：如果一个反馈网络不具有对称连接，则其动态有可能不收敛到一个不动点。

习题 42.4[21] 构造一个例子证明：如果一个 Hopfield 网络被同步更新，则对于某些初始条件，动态有可能不收敛到一个不动点。

42.5 执行中的联想记忆

如图 42.3 所示为 25 单元二进制 Hopfield 网络的动态特性，通过 Hebbian 学习规

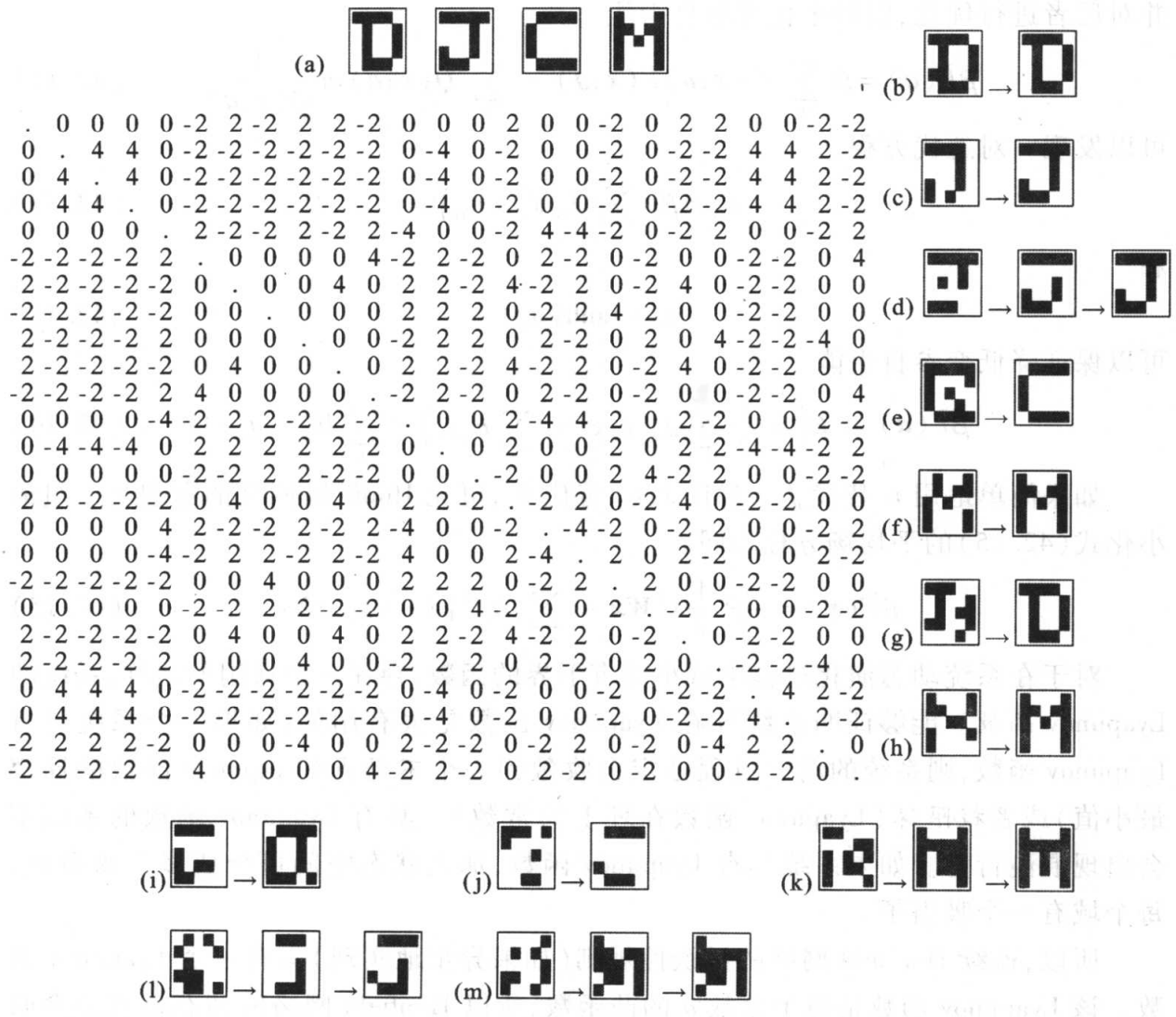


图 42.3 存储 4 个记忆的二进制 Hopfield 网络。(a) 4 个记忆和权矩阵。(b ~ h) 初始状态和期望记忆状态之间有 1 位、2 位、3 位、4 位甚至 5 位不同，经过 1 次或 2 次迭代就可以恢复到原来的记忆。(i ~ m) 和这些记忆差别很大的一些初始状态会演化为 4 个记忆之外的其他稳定状态；在(i)中，稳定状态看起来像是“D”和“J”这两个记忆的混合；稳定状态(j)似乎是“J”和“C”的混合；在(k)中，可以发现“M”记忆的受损版(相差 2 位)；(l) 是“J”的受损版(相差 4 位)，(m)中的状态初看似乎是虚假的，其实是稳定状态(l)的反转。

则学习 4 个模式。4 个模式在图 42.3(a)中用 5×5 的二值图像显示。对于 12 个初始条件,子图(b~m)显示了逐次迭代下的网络状态,所有 25 个单元在每次迭代都是异步更新的。对于随机地围绕记忆摄动的初始条件,往往只要一次迭代就可以纠正所有错误。除了 4 个期望的记忆以外,网络还有更多的稳定状态:任何稳定状态的逆状态也是一个稳定状态;有几个稳定状态可解释为记忆的混合。

脑损伤

网络在严重损坏的情况下,仍然能够较好地进行联想记忆。对于图 42.3 所示网络中的 300 个权重,如果随机地将其中 50 或 100 个权值设为零,发现所期望的记忆仍然是吸引人的稳定状态。想象一下,是否会有数字计算机在 20% 元件已经损坏的情况下,仍能正常地工作!

▷ **习题 42.5**[⌀] 实现一个 Hopfield 网络,并验证它神奇的鲁棒纠错能力。

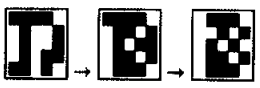
更多记忆

还可以在网络中填充更多的记忆。图 42.4(a)是一组 5 个记忆。当通过 Hebbian 学习规则训练网络时,即使在随机地删除 26 个权重的情况下(如权矩阵中的"x"

(a)



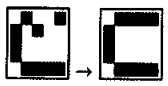
(b)



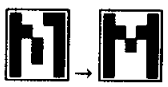
(c)



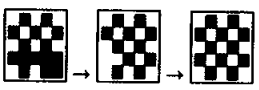
(d)



(e)



(f)



```
 .  -1   1  -1   1   x   x  -3   3   x   x  -1   1  -1   x  -1   1  -3   x   1   3  -3   1   x  -1
-1   .   3   5  -1  -1  -3  -1  -3  -1  -3   1   x   1  -3   1  -1  -1  -1  -1  -3   5   3   3  -3
 1   3   .   3   1  -3  -1   x  -1  -3  -1  -1   x  -1  -1  -1   1  -3   1  -3  -1   3   5   1  -1
-1   5   3   .  -1  -1  -3  -1  -3  -1  -3   1  -5   1  -3   1  -1  -1  -1  -1  -3   5   x   3  -3
 1  -1   1  -1   .   1  -1  -3   x   x   3  -5   1  -1  -1   3   x  -3   1  -3   3  -1   1  -3   3
 x  -1  -3  -1   1   .  -1   1  -1   1   3  -1   1  -1  -1   3  -3   1   x   1   x  -1  -3   1   3
 x  -3  -1  -3  -1  -1   .  -1   1   3   1   1   3  -3   5  -3   1  -1  -1   x   1  -3  -1  -1   1
-3  -1   x  -1  -3   1  -1   .  -1   1  -1   3   1   x  -1  -1   1   5   1   1  -1   x  -3   1  -1
 3  -3  -1  -3   x  -1   1  -1   .  -1   1  -3   3   1   1   1  -1  -1   3  -1   5  -3  -1   x   1
 x  -1  -3  -1   x   1   3   1  -1   .  -1   3   1  -1   3  -1   x   1  -3   5  -1  -1  -3   1  -1
 x  -3  -1  -3   3   3   1  -1   1  -1   .  -3   3  -3   1   1  -1  -1  -1  -1   1  -3  -1  -1   5
-1   1  -1   1  -5  -1   1   3  -3   3  -3   .  -1   1   1  -3   3   x  -1   3  -3   1  -1   3  -3
 1   x   x  -5   1   1   3   1   3   1   3  -1   .  -1   3  -1   1   1   1   1   3  -5  -3  -3   3
-1   1  -1   1  -1  -1  -3   x   1  -1  -3   1  -1   .   x   1  -1   3   3  -1   1   1  -1  -1  -3
 x  -3  -1  -3  -1  -1   5  -1   1   3   1   1   3   x   .   x   3  -1  -1   3   1  -3  -1  -1   1
-1   1  -1   1   3   3  -3  -1   1  -1   1  -3  -1   1   x   .  -5  -1  -1  -1   1   1  -1  -1   1
 1  -1   1  -1   x  -3   3   1  -1   x  -1   3   1  -1   3  -5   .   1   1   1  -1  -1   1   1  -1
-3  -1  -3  -1  -3   1  -1   5  -1   1  -1   x   1   3  -1  -1   1   .   1   1  -1  -1  -3   1  -1
 x  -1   1  -1   1   x  -1   1   3  -3  -1  -1   1   3  -1  -1   1   1   .  -3   3  -1   1  -3  -1
 1  -1  -3  -1  -3   1   x   1  -1   5  -1   3   1  -1   3  -1   1   1  -3   .   x  -1  -3   1  -1
 3  -3  -1  -3   3   x   1  -1   5  -1   1  -3   3   1   1   1  -1  -1   3   x   .  -3  -1  -5   1
-1   5   3   5  -1  -1  -3   x  -3  -1  -3   1  -5   1  -3   1  -1  -1  -1  -1  -3   .   3   x  -3
 1   3   5   x   1  -3  -1  -3  -1  -3  -1  -1  -3  -1  -1  -1   1  -3   1  -3  -1   3   .   1  -1
 x   3   1   3  -3   1  -1   1   x   1  -1   3  -3  -1  -1  -1   1   1  -3   1  -5   x   1   .  -1
-1  -3  -1  -3   3   3   1  -1   1  -1   5  -3   3  -3   1   1  -1  -1  -1  -1   1  -3  -1  -1   .
```

图 42.4 存储了 5 个记忆的 Hopfield 网络,它的 300 个权重中删除了 26 个。(a) 5 个记忆和网络的权重,删除的权重用"x"表示。(b~f) 初始状态和一个记忆相差 3 个随机位:有些得到恢复,但有些却收敛到其他状态。

所示),所有 5 个记忆还是稳定状态。然而,吸引域比以前要小:图 42.4(b ~ f)所显示的是在每个记忆附近的随机选择起始状态(翻转了 3 位)后所得的动态特性。只正确地恢复了 3 个记忆。

如果试图存储过多模式,联想记忆会灾难性地失效。如图 42.5 所示,增加第 6 个模式时,其中只有一个模式是稳定的;其他都演变为两个虚假稳定状态中的一个。

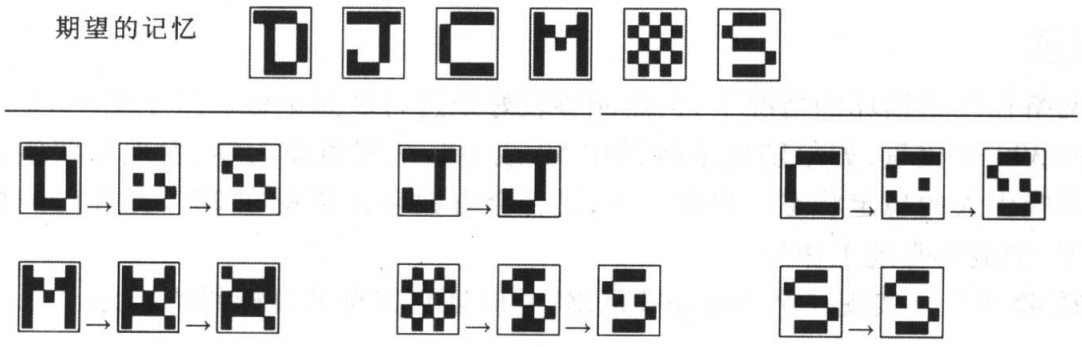


图 42.5 过载 Hopfield 网络,用 6 个记忆进行训练,其中大部分记忆是不稳定的。

42.6 连续时间的连续 Hopfield 网络

对于从异步到同步更新的微小变化,Hopfield 网络的特性不具备鲁棒性。这一事实可能令人担忧;这个模型是生物网络的一个有用模型吗?结果证明当我们转向连续时间版 Hopfield 网络时,这一问题就不复存在了。

假设每个神经元的活跃性 x_i 是时间的连续函数 $x_i(t)$,并假设激活值 $a_i(t)$ 可根据如下公式瞬时地计算出来。

$$a_i(t) = \sum_j w_{ij} x_j(t) \tag{42.16}$$

假设神经元对其激活值的响应由以下微分方程决定。

$$\frac{\mathrm{d}}{\mathrm{d}t} x_i(t) = -\frac{1}{\tau}(x_i(t) - f(a_i)) \tag{42.17}$$

其中 $f(a)$ 是激活函数,例如 $f(a) = \tanh(a)$。对于一个稳定的激活值 a_i,活跃性 $x_i(t)$ 在时间常数 τ 下以指数方式松弛为 $f(a_i)$。

现在,有一个很好的结果:只要权矩阵是对称的,该系统就可以用变参自由能(42.15)作为它的 Lyapunov 函数。

▷ **习题 42.6**[ㄏ] 通过计算 $\frac{\mathrm{d}}{\mathrm{d}t}\tilde{F}$,证明变参自由能函数 $\tilde{F}(\boldsymbol{x})$ 是连续时间 Hopfield 网络的一个 Lyapunov 函数。

如果系统的动态对满足 $\frac{\mathrm{d}}{\mathrm{d}t} x_i(t) \propto \frac{\partial}{\partial x_i} L$ 的 L 实施最速下降,那么特别容易证明函

数 L 是 Lyapunov 函数。在连续时间的连续 Hopfield 网络情况下，事情并非那么简单，但是 $\frac{\mathrm{d}}{\mathrm{d}t}x_i(t)$ 的每个分量与 $\frac{\partial}{\partial x_i}\tilde{F}$ 具有相同的符号，这意味着通过定义一个适当的度量，Hopfield 网络动态确实对 $\tilde{F}(\boldsymbol{x})$ 实施最速下降。

42.7 Hopfield 网络的容量

可以将单神经元中的学习看做一种通信——训练数据集合的标记在时间上的某一点和后来的一点之间的通信。可以发现线性阈值神经元的容量是每个权重 2 比特。

类似地，也可以把 Hopfield 网络的联想记忆看成是通信信道（如图 42.6 所示）。利用 Hebb 规则方程(42.5)或者其他一些学习规则，把一系列期望的记忆编码成一组权重 $\boldsymbol{W}$。接收方只接收权重 $\boldsymbol{W}$，找出 Hopfield 网络的稳定状态，并将其解释成原先的记忆。如图所示，通信系统失效的方式可能是以下几种。

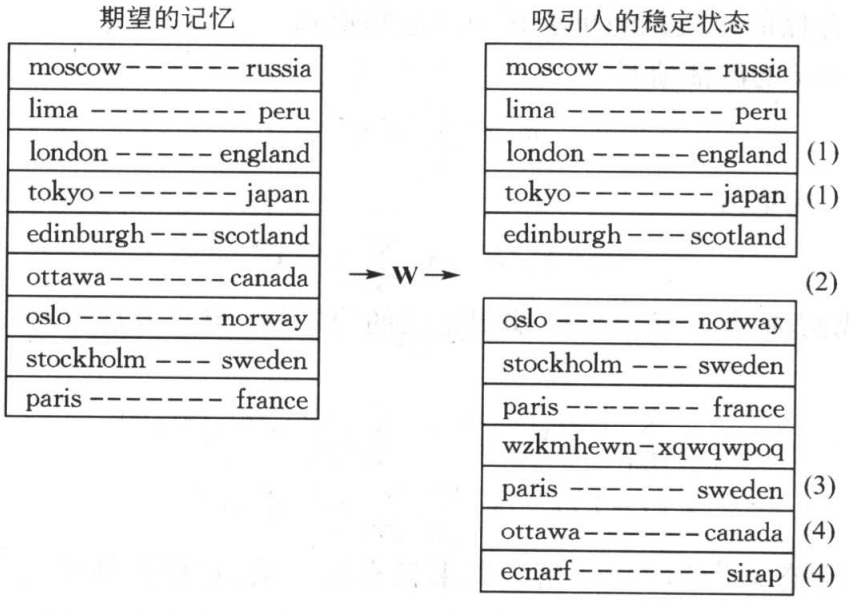


图 42.6 Hopfield 网络的失效模式（仅仅是示意）。一列期望的记忆和所得的一列稳定的吸引状态。注意(1) 保持了记忆，但包含少量错误；(2) 完全丢失的期望记忆（在期望记忆及其附近没有吸引人的稳态）；(3) 同原列表无关的虚假稳定状态；(4) 从期望记忆中滋生出的虚假稳定状态。

1. 一些记忆中的单个比特可能受损，即 Hopfield 网络中的稳定状态和期望的记忆之间存在一定偏差。

2. 整个记忆可能不在网络的吸引子之列；或者是尽管存在稳定状态，但是吸引

域太小,无法用于模式补足和纠错。

3. 可能出现同期望的记忆无关的额外虚假记忆。

4. 也可能通过混合和反转操作从期望的记忆中产生其他虚假记忆。

在这些失效模式中,模式 1 和模式 2 显然是不理想的,尤其是模式 2。模式 3 也许没什么太大影响,只要每个期望的记忆有大的吸引域。第 4 个失效模式在某些情况下甚至可以视为有益的。例如,要求网络能够记住一些有效句子实例,如"约翰爱玛丽"和"约翰得到蛋糕",也许会欣喜地发现"约翰喜欢蛋糕"也是网络的一个稳定状态。可以把这种行为称作"推广"。

具有 I 个神经元的 Hopfield 网络的容量可定义为:在失效模式 2 的概率不大时,可存储的随机模式的个数 N。如果要求失效模式 1 也有很小的概率,那么所得网络容量就会小很多。现在来研究容量的这些定义。

Hopfield 网络容量——严格定义

首先假设二进制 Hopfield 网络的状态设置为期望模式 $\boldsymbol{x}^{(n)}$,使用 Hebb 学习规则进行学习,通过考虑一个期望模式中 1 比特的稳定状态,可以研究该网络的信息存储容量。假设要存储的模式是随机选择的二进制模式。

特定神经元 i 的激活值是

$$a_i = \sum_j w_{ij} x_j^{(n)} \tag{42.18}$$

其中,对于 $i \neq j$,权重为

$$w_{ij} = x_i^{(n)} x_j^{(n)} + \sum_{m \neq n} x_i^{(m)} x_j^{(m)} \tag{42.19}$$

这里把 $\boldsymbol{W}$ 拆成两项,第一项是增强期望记忆的"信号",第二项是"噪声"。代入 w_{ij} 后的激活值为

$$a_i = \sum_{j \neq i} x_i^{(n)} x_j^{(n)} x_j^{(n)} + \sum_{j \neq i} \sum_{m \neq n} x_i^{(m)} x_j^{(m)} x_j^{(n)} \tag{42.20}$$

$$= (I-1) x_i^{(n)} + \sum_{j \neq i} \sum_{m \neq n} x_i^{(m)} x_j^{(m)} x_j^{(n)} \tag{42.21}$$

第一项是期望状态 $x_i^{(n)}$ 的 $(I-1)$ 倍。如果只有这一项,它将把神经元牢牢地限制在期望状态。第二项为 $(I-1)(N-1)$ 个随机量 $x_i^{(m)} x_j^{(m)} x_j^{(n)}$ 之和。稍微想一下就可以证实这些量是均值为 0、方差为 1 的独立随机二进制变量。

因此考虑 a_i 在随机模式总体下的统计特性,可以得出 a_i 的均值为 $(I-1)x_i^{(n)}$,方差为 $(I-1)(N-1)$。

为了简洁起见,假设 I 和 N 足够大,可以忽略 I 和 $I-1$ 以及 N 和 $N-1$ 之间的差异。于是,可以表述我们的结论:a_i 是均值为 $Ix_i^{(n)}$、方差为 IN 的高斯分布。

如果让网络处于状态 $\boldsymbol{x}^{(n)}$,则所选比特稳定的概率是多少?在 Hopfield 网络动态过程的第一次迭代中,比特 i 翻转的概率(如图 42.7 所示)为

$$P(i \text{ 不稳定}) = \Phi\left(-\frac{I}{\sqrt{IN}}\right) = \Phi\left(-\frac{1}{\sqrt{N/I}}\right) \tag{42.22}$$

其中

$$\Phi(z)\equiv\int_{-\infty}^{z}\mathrm{d}z\,\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-z^2/2} \tag{42.23}$$

重要的量 N/I 是存储模式数和神经元数之比。例如，如果要在 Hopfield 网络中存储 $N\simeq 0.18I$ 个模式，那么在第一次迭代中，特定模式中的特定比特就有 1% 的可能性不稳定。

现在可以对不容许期望记忆受损的情况，推导关于容量的第一个结果。

▷**习题 42.7**[2] 假定希望所有期望的模式都是完全稳定的（当网络处于任何期望的模式状态中时，不希望有任何比特发生翻转），而且要求任何错误的总概率小于一个很小的数 ε。对于大的 z，采用误差函数的近似

$$\Phi(-z)\simeq\frac{1}{\sqrt{2\pi}}\frac{\mathrm{e}^{-z^2/2}}{z} \tag{42.24}$$

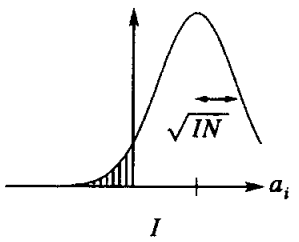


图 42.7 在 $x_i^{(n)}=1$ 时激活值 a_i 的概率密度；比特 i 被翻转的概率是尾部区域的面积。

证明可存储模式的最大数量 $N_{\max}$ 为

$$N_{\max}\simeq\frac{I}{4\ln I+2\ln(1/\varepsilon)} \tag{42.25}$$

然而，如果允许有少量的记忆被损坏，则可存储模式的个数会增加。

统计物理学者的容量

导出公式(42.22)的分析告诉我们，如果在 Hopfield 网络中存储 $N\simeq 0.18I$ 个模式，则从期望的记忆开始，在第一次迭代中约有 1% 的比特是不稳定的。分析并未表明在后续迭代中会发生什么情况。这些比特的翻转可能会导致其他比特也不稳定，造成翻转比特数的增加。这个过程可能会造成雪崩现象，使得网络的状态与所期望的记忆相距甚远。

事实上，当 N/I 很大时，的确会发生雪崩。当 N/I 较小时则不会发生——在每个期望的记忆附近都有一个稳定状态。至于大 I 的极限，Amit 等(1985 年)利用统计物理方法得出了两种特性之间的数值转变。一个急剧的不连续点出现在

$$N_{\mathrm{crit}}=0.138I \tag{42.26}$$

在这个临界值以下，在每个期望的记忆附近都可能有一个稳定状态，此时只有少量比特被翻转。当 N/I 超过 0.138，系统只处于虚假稳定状态，就是自旋玻璃状态，其中没有状态和任何期望记忆相关。就在临界值之下，当一个期望记忆演化为与之相关的稳定状态时，翻转比特的比例是 1.6%。图 42.8 显示了期望记忆和最近稳定状态之间的重叠，它是 N/I 的一个函数。

在其他一些 N/I 值处，模型特性发生了其他转移，总结如下。

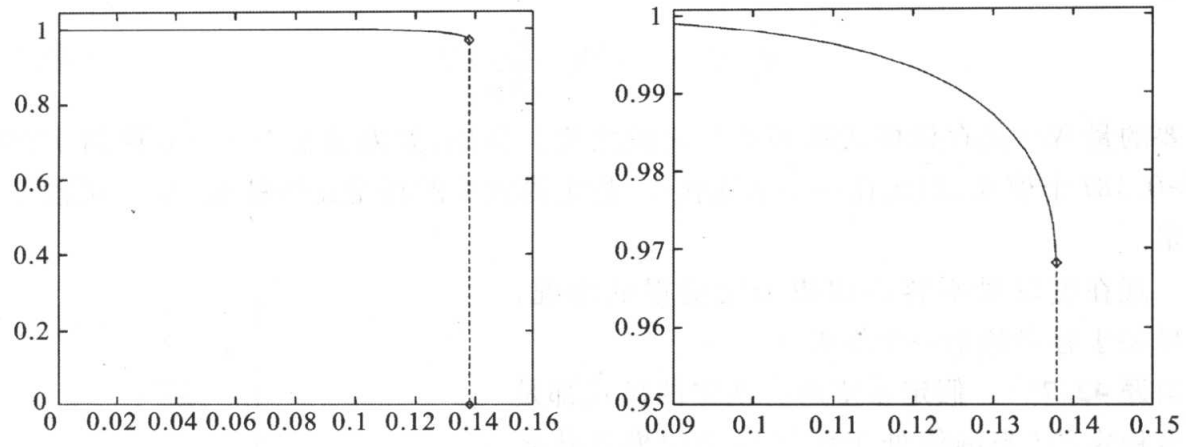


图 42.8 期望记忆和最靠近它的稳定状态的重叠，以负载比 N/I 的函数表示。重叠定义为经比例缩放的内积 $\sum_i x_i x_i^{(n)}/I$。如果回忆完全正确，重叠为 1；如果稳定状态有 50% 的比特翻转，则重叠为 0。在 $N/I=0.138$ 时，发生急剧的转变，重叠从 0.97 降到 0。

对于所有 N/I，存在和期望记忆不相关的稳定自旋玻璃状态。

对于 $N/I>0.138$，自旋玻璃状态是仅有的稳定状态。

对于 $N/I\in(0,0.138)$，存在靠近期望记忆的稳定状态。

对于 $N/I\in(0,0.05)$，和期望记忆相关的稳定状态的能量低于虚假自旋玻璃状态。

对于 $N/I\in(0.05,0.138)$，自旋玻璃状态占主导地位——存在比与期望记忆相关的稳定状态能量更低的自旋玻璃状态。

对于 $N/I\in(0,0.03)$，存在另外的混合状态，这是几个期望记忆的组合。这些稳定状态不像与期望记忆相关的稳定状态那样有较低能量。

总之，如果根据上面讨论的急剧不连续点来定义容量，具有 I 个神经元的 Hopfield 网络的容量是 $0.138I$ 个随机二进制模式，每个长度是 I，每个在接收时有 1.6% 的比特被翻转。以比特表达的容量是

$$0.138I^2\times(1-H_2(0.016))=0.122I^2\text{比特} \tag{42.27}$$

因为网络中有 $I^2/2$ 个权重，也可以把容量表示为每个权重 0.24 比特。

42.8 改善 Hebb 规则的容量

前面一节所讨论的是网络的权重使用 Hebbian 学习规则的 Hopfield 网络的容量。通过定义目标函数为度量网络存储所有记忆的好坏程度，并对其最小化，可以得到比 Hebb 规则更好的结果。

为了使联想记忆有用，它必须至少能纠正 1 个翻转比特。构造一个目标函数来度量翻转比特是否倾向于被正确地恢复。具体想法是，对于网络中的每个神经元 i，该神经元的权重应该满足如下规则：

对于每个模式 $\boldsymbol{x}^{(n)}$，如果除 i 之外的神经元被正确设置为 $x_j = x_j^{(n)}$，则神经元 i 的激活值应使其理想输出为 $x_i = x_i^{(n)}$。

熟悉这个规则的思想吗？是的，这正是我们希望第 39 章中的单神经元所能做到的。每个模式 $\boldsymbol{x}^{(n)}$ 对单神经元 i 定义了一个输入 - 目标对。也对所有其他神经元定义了一个输入 - 目标对。

所以，正如为训练单神经元为分类器而定义的目标函数(39.11)，可以定义

$$G(\boldsymbol{W}) = -\sum_i \sum_n t_i^{(n)} \ln y_i^{(n)} + (1 - t_i^{(n)}) \ln(1 - y_i^{(n)}) \tag{42.28}$$

其中

$$t_i^{(n)} = \begin{cases} 1, & x_i^{(n)} = 1 \\ 0, & x_i^{(n)} = -1 \end{cases} \tag{42.29}$$

以及

$$y_i^{(n)} = \frac{1}{1 + \exp(-a_i^{(n)})}, \quad \text{其中 } a_i^{(n)} = \sum w_{ij} x_j^{(n)} \tag{42.30}$$

可以借用为单神经元所设计的算法(算法 39.5)来设计优化 Hopfield 网络的算法，算法 42.9。使用 Octave 方便的语法，算法几乎不需改动；额外的行是为了满足以下约束条件：自身权重 w_{ii} 都为零，权矩阵应该是对称的($w_{ij} = w_{ji}$)。

```
w = x' * x ;                                  #用 Hebb 规则初始化权重
for l = 1:L                                   # 循环 L 次
    for i = 1:I
      w(i,i) = 0 ;                            # 保证自身权重为零
    end
    a = x * w ;                               # 计算所有激活值
    y = sigmoid(a) ;                          # 计算所有输出
    e = t - y ;                               # 计算所有误差
    gw = x' * e ;                             # 计算梯度
    gw = gw + gw';                            #使梯度对称
    w = w + eta * ( gw - alpha * w ) ;        # 进行步进
endfor
```

算法 42.9 对 Hopfield 网络的权重进行优化，使它可以实现联想记忆的 Octave 源代码。对比算法 39.5。数据矩阵 x 有 I 列和 N 行。矩阵 t 除了用 0 代替 -1 之外，和 x 相同。

不出所料，这个学习算法的性能优于只进行一次的 Hebbian 学习规则。图 42.5

的 6 个模式用 Hebb 规则是不能被记忆的，而用算法 42.9 进行学习时，所有 6 个模式都达到稳定状态。

习题 42.8[✍] 实现这个学习规则，并经验式地探讨它记忆随机模式的容量；再就雪崩特性将它与 Hebb 规则进行比较。

42.9 用于优化问题的 Hopfield 网络

因为 Hopfield 网络的动态过程会使能量函数最小化，很自然的一个问题是能否把有趣的优化问题映射到 Hopfield 网络。生物数据处理问题经常涉及满足约束的成分：例如在场景解释中，人们可能希望推断每个可见元素的空间位置、方位、亮度和纹理，及物体中哪些可见元素连接在一起。这些推理受限于已知数据以及关于目标连续性的先验知识。

Hopfield 和 Tank(1985 年)建议，对于一个有趣的满足约束问题，可以设计二进制或连续 Hopfield 网络的权重，使得网络的设置过程可以使问题的目标函数最小化。

旅行推销员问题

Hopfield 网络得以应用的一个经典满足约束问题是旅行推销员问题。

已知 K 个城市的集合以及城市之间 $K(K-1)/2$ 个距离所组成的矩阵。任务是找到穿越城市的闭合路径，使得每个城市被访问一次，且总路径最短。旅行推销员问题在难度上等同于 NP - 完全问题。

Hopfield 和 Tank 所提出的方法代表以正方形分布的 $I = K^2$ 个神经元的网络状态问题的临时解，每个神经元表示特定城市出现在路径中一个特定点上的假设。考虑神经元的状态在 0 和 1 之间要比在 -1 和 1 之间方便一些。如图 42.10(a)所示是 4 - 城市旅行推销员问题的两个解状态。

Hopfield 网络中的权重起到两个作用。首先，它们必须定义一个能量函数，只有当网络的状态表示合法旅途时才最小化该能量。一个合法状态看起来像是一个置换矩阵，每行只有一个“1”，每列只有一个“1”。通过把处于同行或同列的任意一对神经元之间的权重设置成较大的负值，对所有神经元都设置一个正的偏置，以确保有 K 个神经元开启，就可实现这一规则。图 42.10(b)显示了连接到一个神经元“$B2$”的负权重，它表示命题“城市 B 在旅途中第 2 个到达”。

其次，权重必须对将要最小化的目标函数(总的距离)进行编码。这可以通过让负权重和相邻列结点之间的适当距离成正比来完成。例如，在相邻列的 B 和 D 结点之间，权重应该是 $-d_{BD}$。连接到神经元 $B2$ 的负权重如图 42.10(c)所示。当网络处于合法状态，结果是总的能量为相应路径的总距离，加上和偏置相关联的能量所给出的一个常数。

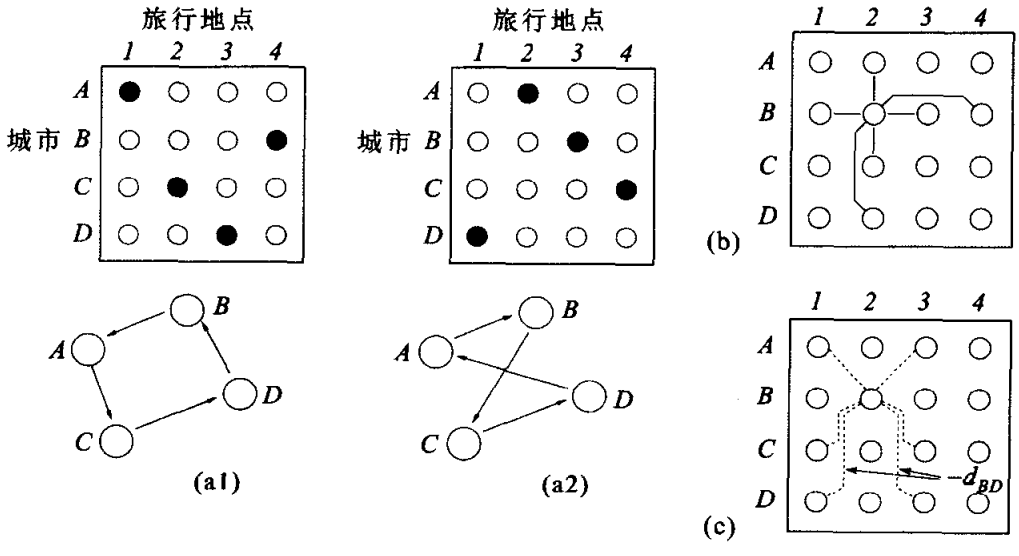


图 42.10 用于解决旅行推销员问题的 Hopfield 网络，其中 $K=4$ 个城市。(a1，a2) 16 个神经元的两个解状态，用黑 =1，白 =0 表示激活值；对应于这些网络状态的路径。(b) 结点 $B2$ 和其他结点之间的负权重，这些权重使得路径合法。(c) 负权重构成了距离目标函数。

因为 Hopfield 网络最小化它的能量，可以希望二进制或连续 Hopfield 网络的动态过程使状态达到最小值，它是合法的并可能是最优的路径。然而对于较大的旅行推销员问题，如果不进行仔细修改，是不可能实现这个愿望的。我们还没有指定使路径合法化的权重大小(相对于距离权重的大小)，设定这个比例因子是很困难的。如果使用“大”的权值来强制合法性，网络动态将进入与距离没有什么关系的合法状态。如果使用“小”的权值来强制合法性，则距离权重可能使网络进入比任何合法状态具有更低能量的非法状态。原先构造的能量函数使得目标函数和解的合法性之间存在潜在的冲突。这个难题被 Sree Aiyer(1991 年)解决，他说明如何修正距离权重使之不与解的合法性相抵触，及如何定义一个连续 Hopfield 网络使得它的动态始终限于“合法的子空间”中。Aiyer 采用分段非凸或确定性退火方法通过 Hopfield 网络求得好的解。确定性退火方法从 0 到 ∞ 逐渐增加网络神经元的增益 β，在这个点上，网络状态对应于一条合法路径。把这一方法用于 30 个城市的旅行推销员问题时所生成的轨迹序列如图 42.11(a)所示。

如图 42.11(b)所示为由 Aiyer 求得的应用连续 Hopfield 网络的“旅行学者问题”的解。

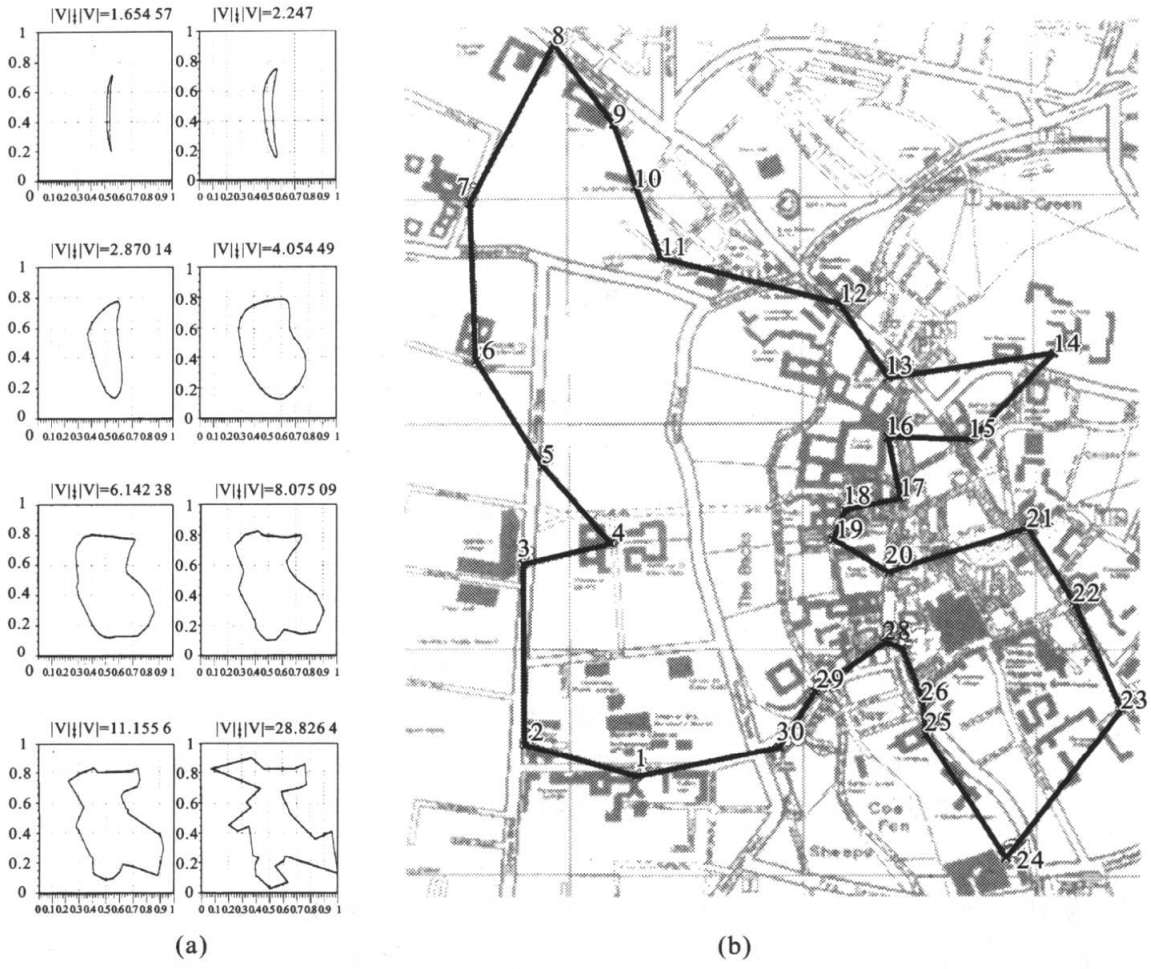


图 42.11 (a) 解决旅行推销员问题的连续 Hopfield 网络的状态演化，这是应用 Aiyer(1991 年)分段非凸方法得到的；网络的状态被投影到二维空间中；在该空间中，利用神经元活跃性作为质量函数，对于旅途中的每个点，通过寻找质量中心来确定城市位置。(b) 旅行学者问题。连接 27 个剑桥学院、工程系、大学图书馆和 Sree Aiyer屋的最短路径。来自参考文献 Aiyer(1991 年)。

42.10 进一步的习题

▷习题 42.9[$\mathscr{P}$] **存储两个记忆**

两个二进制记忆 $\boldsymbol{m}$ 和 $\boldsymbol{n}$($m_i, n_i \in \{-1, +1\}$)采用 Hebbian 学习规则存储在一个 Hopfield 网络中，权重的设置为

$$w_{ij} = \begin{cases} m_i m_j + n_i n_j, & i \neq j \\ 0, & i = j \end{cases} \tag{42.31}$$

偏置 b_i 设为 0。

将网络的状态置为 $\boldsymbol{x} = \boldsymbol{m}$。计算神经元 i 的激活值 a_i，并证明它可以写成形式

$$a_i = \mu m_i + \nu n_i \tag{42.32}$$

通过比较信号强度 μ 和噪声幅值 $|\nu|$，证明 $\boldsymbol{x} = \boldsymbol{m}$ 是网络动态的稳定状态。

将网络的状态置为与 $\boldsymbol{m}$ 有 D 处不同的 $\boldsymbol{x}$

$$\boldsymbol{x} = \boldsymbol{m} + 2\boldsymbol{d} \tag{42.33}$$

其中摄动 $\boldsymbol{d}$ 满足 $d_i \in \{-1,0,1\}$。D 是 $\boldsymbol{d}$ 中非零元的个数，对于每个非零的 d_i 有 $d_i = -m_i$。定义 $\boldsymbol{m}$ 和 $\boldsymbol{n}$ 之间的重叠为

$$o_{mn} = \sum_{i=1}^{I} m_i n_i \tag{42.34}$$

重新计算神经元 i 的激活值 a_i。证明：如果翻转比特数满足

$$D < \frac{1}{4}(I - |o_{mn}| - 2) \tag{42.35}$$

网络的动态将把 $\boldsymbol{x}$ 恢复成 $\boldsymbol{m}$。假设向量 $\boldsymbol{x}$ 是带噪声的 $\boldsymbol{m}$ 或 $\boldsymbol{n}$，则该翻转的比特数和能被最优译码器纠正的最大翻转比特数相比会怎样？

习题 42.10[$\mathcal{P}$] Hopfield **网络作为一组二进制分类器**。这个习题探讨了无监督网络和有监督网络之间的联系。如果一个 Hopfield 网络的所有期望记忆都是吸引人的稳定状态，那么该网络中的每个神经元的权值趋向于解决该神经元自己的分类问题。将这组记忆写成 $\boldsymbol{x}'^{(n)}, x_i^{(n)}$ 的形式，其中 $\boldsymbol{x}'$ 表示 $i' \neq i$ 的所有分量 $x_{i'}$，并令 $\boldsymbol{w}'$ 表示 $i' \neq i$ 的权重 $w_{ii'}$ 组成的向量。

利用已知单神经元容量，证明在有 I 个神经元的 Hopfield 网络中存储多于 $2I$ 个随机记忆几乎是不可能的。

Lyapunov 函数

习题 42.11[$\mathcal{P}$] Erik **谜题**。在 Conway 生命游戏的简化版中，细胞布置于方的网格中（如图 42.12 所示）。每个细胞可以是活的，也可以是死的。活的细胞不会死。如果死细胞的近邻中有两个或更多的细胞是活的，则死细胞可以变活。（邻居是北、南、东和西。）按照这一规则，为了使整个 $N \times N$ 方格中的细胞都成活，所需的最少活细胞数是多少？

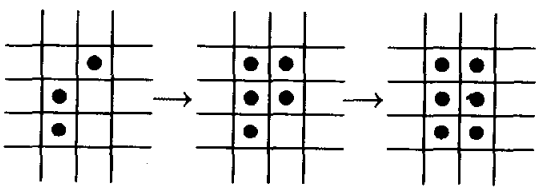


图 42.12 Erik 动态

对 d 维空间中的同一游戏版本，规则是：如果有 d 个邻居是活的，那么你可以成活。为了使整个 $N \times N \times \cdots \times N$ 超立方体中的细胞成活，所需的最少活细胞数是多少？（这些活细胞如何布置？）

东南谜题

东南谜题是在半无限的国际象棋棋盘上进行的，从西北（左上）角开始。存在以下 3 条规则。

1. 在起始位置，在最西北方块处布置一个棋子（图 42.13(a)）。

2. 在任何已知方块中不容许多于 1 个棋子。

3. 在每一步，从棋盘上移去一个棋子，用两个棋子来取代，其中一个在东面方块中，另一个在南面方块中，如图 42.13(b)所示。每一步都在棋盘上增加了一个棋子。

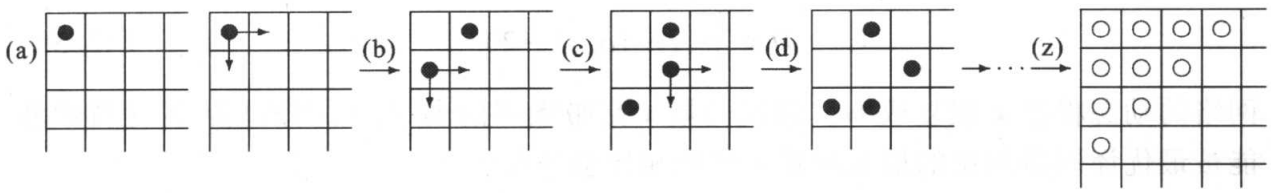


图 42.13 东南谜题

在步骤(b)之后，下一步可以选择两个棋子中的任意一个。图 42.13(c)显示了移动下面一个棋子的结果。在下一步移动中，可以选择最下面的一个棋子，或者选择 3 个棋子中的中间那个；不能选择最上面的棋子，因为这将违背规则 2。在移动(d)中，选择了中间棋子。现在除了最左边的棋子之外，可以移动任何一个棋子。

现在难题来了。

▷ **习题 42.12**[✓] 是否有可能得出一个布局，使如图 42.13(z)标出的最靠近西北角的所有 10 个方块为空？

[提示：这个难题与数据压缩有关。]

42.11 习题解答

习题 42.3 的解答 采用两个神经元的二进制反馈网络，并设 $w_{12}=1$ 和 $w_{21}=-1$。那么，每当神经元 1 被更新，它将和神经元 2 相匹配，而每当神经元 2 被更新，它将翻转到和神经元 1 相反的状态。所以，不存在稳定状态。

习题 42.4 的解答 采用两个神经元的二进制 Hopfield 网络，设 $w_{12}=w_{21}=1$，并设初始条件为 $x_1=1, x_2=-1$。那么，如果动态是同步的，两个神经元在每次迭代时都会翻转它们的状态。动态不会收敛到不动点。

习题 42.12 的解答 这个问题的关键在于，注意到它和构造二进制符号码的相似性。从空字符串开始，可以不断地把一个码字分裂成两个以建造一棵二进制树。每个码字的隐含概率是 2^{-l}，其中 l 是码字在二进制树上的深度。只要把一个码字分

裂成两个，就得到了两个长度增加 1 位的新码字，每个新码字的隐含概率都等于老码字的一半。对于一个完全二进制码，Kraft 等式断定隐含概率之和为 1。

类似地，在东南游戏中，可以把“权重”和棋盘上的棋子联系在一起。如果把权重 1 赋予左上方块中的棋子；权重 1/2 赋予任何到左上方块的距离为 1 的棋子；权重 1/4 赋予任何到左上方块距离为 2 的棋子，依此类推，所谓“距离”就是城市街区之间的距离；那么任何合法的东南移动不会改变棋盘上所有棋子的总权重。有两种方式得到 Lyapunov 函数：可以是状态值保持恒定的状态函数；或者是有下界且状态值总在下降或保持恒定的状态函数。总权重就是第二种类型的 Lyapunov 函数。

起始权重为 1，因此现在有了一个强大的工具：状态的守衡函数。有可能发现 10 个最高权重的方块是空的，而总权重为 1 的布局吗？如果棋盘上所有其他方块被占据（图 42.14），总权重是多少？总权重是 $\sum_{l=4}^{\infty}(l+1)2^{-l}$，等于 3/4。所以不可能使所有 10 个方块为空。

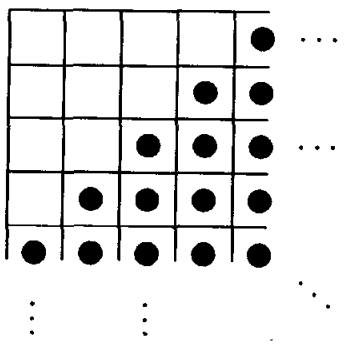


图 42.14 这是东南谜题的一个可能布局吗？

第 43 章

玻耳兹曼机

43.1 从 Hopfield 网络到玻耳兹曼机

注意到二进制 Hopfield 网络最小化一个能量函数

$$E(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{x} \tag{43.1}$$

而且具有激活函数 $x_n = \tanh(a_n)$ 的连续 Hopfield 网络可以看做对与这个能量函数相关的概率分布的近似

$$P(\boldsymbol{x}\mid\boldsymbol{W}) = \frac{1}{Z(\boldsymbol{W})}\exp[-E(\boldsymbol{x})] = \frac{1}{Z(\boldsymbol{W})}\exp\left[\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{x}\right] \tag{43.2}$$

这些观察启发人们用一种神经网络模型实现以上概率分布的想法。

随机 Hopfield 网络或玻耳兹曼机(Hinton 和 Sejnowski,1986 年)具有以下活跃性规则。

玻耳兹曼机的活跃性规则:计算激活值 a_n 之后

$$\text{以概率}\frac{1}{1+\mathrm{e}^{-2a}}\text{设置 } x_n = +1 \tag{43.3}$$

否则设置 $x_n = -1$。

这个规则实现了对概率分布(43.2)的 Gibbs 采样。

玻耳兹曼机的学习

已知一组来自客观世界的例子 $\{\boldsymbol{x}^{(n)}\}_1^N$,人们感兴趣的也许是调整权重 $\boldsymbol{W}$,使得生成模型

$$P(x\mid\boldsymbol{W}) = \frac{1}{Z(\boldsymbol{W})}\exp\left[\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{x}\right] \tag{43.4}$$

能够与这些例子很好地匹配。写出贝叶斯定理,得到对于已知数据的权重的后验概率,可以由此推导出一个学习算法

$$P(\boldsymbol{W}\mid\{\boldsymbol{x}^{(n)}\}_1^N\}) = \frac{[\prod_{n=1}^{N}P(\boldsymbol{x}^{(n)}\mid\boldsymbol{W})]P(\boldsymbol{W})}{P(\{\boldsymbol{x}^{(n)}\}_1^N\})} \tag{43.5}$$

把注意力集中在分子的第一项(似然函数)上,并导出最大似然算法(尽管得到全贝

叶斯方法可能有利，就像对单神经元情形所做的）。对以下对数似然函数求 w_{ij} 的导数。

$$\ln\Big[\prod_{n=1}^{N} P(\boldsymbol{x}^{(n)} \mid \boldsymbol{W})\Big] = \sum_{n=1}^{N}\Big[\frac{1}{2}\boldsymbol{x}^{(n)\mathrm{T}}\boldsymbol{W}\boldsymbol{x}^{(n)} - \ln Z(\boldsymbol{W})\Big] \tag{43.6}$$

并记住 $\boldsymbol{W}$ 被定义为对称的：$w_{ji} = w_{ij}$。



习题 43.1[2] 证明 $\ln Z(\boldsymbol{W})$ 关于 w_{ij} 的导数为

$$\frac{\partial}{\partial w_{ij}}\ln Z(\boldsymbol{W}) = \sum_{\boldsymbol{x}} x_i x_j P(\boldsymbol{x}\mid\boldsymbol{W}) = \langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})} \tag{43.7}$$

[这个习题和习题 22.12 相似。]

因此，对数似然函数的导数为

$$\frac{\partial}{\partial w_{ij}}\ln P(\{\boldsymbol{x}^{(n)}\}_1^N \mid \boldsymbol{W}) = \sum_{n=1}^{N}\left[x_i^{(n)}x_j^{(n)} - \langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})}\right] \tag{43.8}$$

$$= N\left[\langle x_i x_j\rangle_{\text{数据}} - \langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})}\right] \tag{43.9}$$

这个梯度与两项之差成正比。第一项是 x_i 和 x_j 之间的经验式相关函数

$$\langle x_i x_j\rangle_{\text{数据}} \equiv \frac{1}{N}\sum_{n=1}^{N}\left[x_i^{(n)}x_j^{(n)}\right] \tag{43.10}$$

第二项是 x_i 和 x_j 在当前模型下的相关函数

$$\langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})} \equiv \sum_{\boldsymbol{x}} x_i x_j P(\boldsymbol{x}\mid\boldsymbol{W}) \tag{43.11}$$

第一个相关函数 $\langle x_i x_j\rangle_{\text{数据}}$ 很容易计算——它就是客观世界活跃性之间的经验式相关函数。第二个相关函数 $\langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})}$ 不容易计算，但是可以用蒙特卡罗方法进行估计，即在玻耳兹曼机的活跃性规则方程(43.3)进行迭代时，观测 $x_i x_j$ 的平均值。

在 $\boldsymbol{W}=0$ 的特殊情况下，可以精确地计算梯度，因为根据对称性，相关函数 $\langle x_i x_j\rangle_{P(\boldsymbol{x}\mid\boldsymbol{W})}$ 必须为零。如果使用学习速率为 η 的梯度下降算法调整权重，经过一次迭代，权重变为

$$w_{ij} = \eta\sum_{n=1}^{N}\left[x_i^{(n)}x_j^{(n)}\right] \tag{43.12}$$

这正是由训练 Hopfield 网络的 Hebb 规则方程(16.5)所给出的权重值。

对玻耳兹曼机学习的解释

看待梯度(43.9)中两项的一种方法是“清醒”和“睡眠”规则。当网络“清醒”时，它测量客观世界中 x_i 和 x_j 之间的相关程度，权重是按比例增加的。当网络“睡着”时，它利用生成模型(43.4)“梦想”客观世界，并测量模型世界中 x_i 和 x_j 之间的相关程度；这些相关性决定了权重是按比例减小的。如果梦境中的二阶相关函数与客观世界中的相关函数相匹配，那么这两项达到平衡，而权重不会变化。

对 Hopfield 网络和简单玻耳兹曼机的批评

目前为止，在已经讨论的 Hopfield 网络和玻耳兹曼机中，所有神经元都对应于可视变量 x_i。结果是一个统计模型，经过优化后，它能够刻画环境的二阶统计特性。[一个总体 $P(x)$ 的二阶统计特性就是所有乘积对 $x_i x_j$ 的期望值 $\langle x_i x_j \rangle$。]但是现实世界中的系统往往具有高阶相关性；为了有效地对它进行描述，必须考虑高阶相关性。二阶相关性本身往往携带很少甚至不包含任何有用信息。

例如，考虑椅子二进制图像的总体。可以想象各式各样椅子的图像——4 腿椅子、躺椅、带轮子的五腿椅、木椅、软垫椅和没有腿的摇椅。一个儿童能够轻易地把这些图像和胡萝卜、鹦鹉的图像区分开。但是笔者估计这些原始数据的二阶统计特性对于描述总体没有什么用处。二阶统计特性只能反映两个像素是否处于彼此相同的状态。为了很好地得到椅子图像的生成模型，需要高阶的概念。

一个能反映高阶统计重要性的更简单的图像总体是“移位总体”，它有两种风格。图 43.1(a)所示为来自“普通移位总体”的几个样本。在每个图像中，底部的 8 个像素是顶部 8 个像素经过移位后的一个副本：或者是向左移动 1 个像素，或者是没有移动，或者是向右移动 1 个像素。(顶部 8 个像素是随机设定的。)这个总体是从两只眼睛到达大脑早期视觉信号的一个简单模型。从两只眼睛来的信号基本相似，但是由于视觉世界深度的变化，会有一些小的平移。这个总体易于描述，但是它的二阶统计特性没有包含有用信息。一个像素和其上 3 个像素中任何一个之间的相关系数是 1/3。任何其他两个像素之间的相关系数是零。

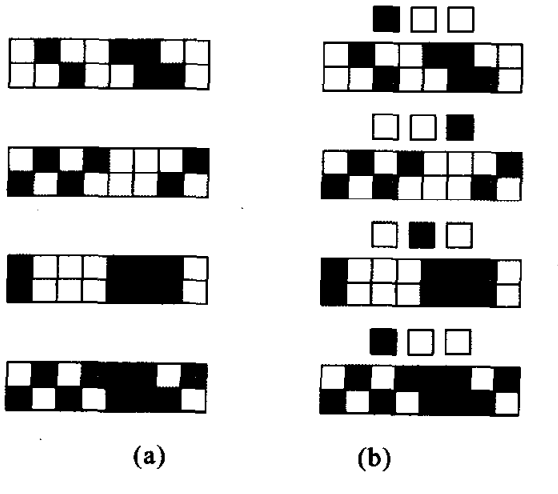


图 43.1 “移位”总体。(a) 来自普通移位总体的 4 个样本。(b) 来自带标记移位总体的 4 个相应样本。

图 43.1(b)所示为来自“带标记的移位总体”的几个样本。通过增加 3 个神经元使问题得以简化，3 个神经元将视觉图像标记为“左移”、“不移”或“右移”子总体的

实例。即使有了这些附加信息，如果只利用二阶统计特性，这个总体仍然是不可学习的。任何标记神经元和任何图像神经元之间的二阶相关性是零。需要能够刻画环境中更高阶统计特性的模型。

那么，如何才能建立这样的模型？一个想法可以是直接创建刻画高阶相关性的模型，比如

$$P'(\boldsymbol{x} \mid \boldsymbol{W}, \boldsymbol{V}, \cdots) = \frac{1}{Z'} \exp\left(\frac{1}{2} \sum_{ij} w_{ij} x_i x_j + \frac{1}{6} \sum_{ij} v_{ijk} x_i x_j x_k + \cdots \right) \tag{43.13}$$

利用随机更新，高阶玻耳兹曼机同样是容易模拟的，高阶参数 v_{ijk} 的学习规则和 w_{ij} 的学习规则是等价的。

▷ **习题 43.2**[⇗] 推导对数似然函数关于 v_{ijk} 的梯度。

在生物神经元中发现的脊细胞有可能负责检测少量输入信号之间的相关性。然而，要能够刻画足够高的统计特性以便很好地描述椅子图像总体，所需项数多得令人难以想象。为了捕获 128×128 像素图像中的四阶统计特性，需要超过 10^7 个参数。

所以测量图像矩的方法不是描述图像内在结构的好方法。也许作为替代或补充，需要隐式变量，统计学家也称为本征变量。这是 Hinton 和 Sejnowski（1986 年）所引入的重要发明。它的基本思想是可视变量之间的高阶相关性可以通过包含附加隐式变量来描述，并且坚持使用只包含变量间二阶相关性的模型；隐式变量导致了可视变量之间的高阶相关性。

43.2 具有隐式单元的玻耳兹曼机

现在在随机模型中增加隐神经元。这些神经元与被观测变量不相对应；它们可以在式（43.4）所定义的概率模型中起到任何作用。也许它们实际上只起到释义的作用，即有效地进行“特征提取”。

具有隐单元的玻耳兹曼机的学习

具有隐单元的玻耳兹曼机的活跃性规则与原来的玻耳兹曼机相同。通过最大化似然函数也可以得到学习规则，但是现在要考虑隐单元状态未知这一事实。用 $\boldsymbol{x}$ 表示可视单元的状态，用 $\boldsymbol{h}$ 表示隐单元的状态，用 y_i 表示一个神经元（可视或隐含）的一般状态，这里 $\boldsymbol{y} \equiv (\boldsymbol{x}, \boldsymbol{h})$。当可视神经元被限定于状态 $\boldsymbol{x}^{(n)}$ 时，网络的状态是 $\boldsymbol{y}^{(n)} \equiv (\boldsymbol{x}^{(n)}, \boldsymbol{h})$。已知单一数据样例 $\boldsymbol{x}^{(n)}$ 时 $\boldsymbol{W}$ 的似然函数为

$$P(\boldsymbol{x}^{(n)} \mid \boldsymbol{W}) = \sum_{\boldsymbol{h}} P(\boldsymbol{x}^{(n)}, \boldsymbol{h} \mid \boldsymbol{W}) = \sum_{\boldsymbol{h}} \frac{1}{Z(\boldsymbol{W})} \exp\left[\frac{1}{2} [\boldsymbol{y}^{(n)}]^{\mathrm{T}} \boldsymbol{W} \boldsymbol{y}^{(n)} \right] \tag{43.14}$$

其中

$$Z(\boldsymbol{W}) = \sum_{\boldsymbol{x},\boldsymbol{h}} \exp\left[\frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right] \tag{43.15}$$

方程(43.14)也可以写成

$$P(\boldsymbol{x}^{(n)} \mid \boldsymbol{W}) = \frac{Z_{\boldsymbol{x}^{(n)}}(\boldsymbol{W})}{Z(\boldsymbol{W})} \tag{43.16}$$

其中

$$Z_{\boldsymbol{x}^{(n)}}(\boldsymbol{W}) = \sum_{\boldsymbol{h}} \exp\left[\frac{1}{2}[\boldsymbol{y}^{(n)}]^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}^{(n)}\right] \tag{43.17}$$

与前面一样对似然函数求导,可以发现关于任意权重 w_{ij} 的导数又是“清醒”项和“睡眠”项之差

$$\frac{\partial}{\partial w_{ij}} \ln P(\{\boldsymbol{x}^{(n)}\}_1^N \mid \boldsymbol{W}) = \sum_n \{\langle y_i y_j \rangle_{P(\boldsymbol{h}\mid\boldsymbol{x}^{(n)},\boldsymbol{W})} - \langle y_i y_j \rangle_{P(\boldsymbol{x},\boldsymbol{h}\mid\boldsymbol{W})}\} \tag{43.18}$$

如果将可视变量限定在状态 $\boldsymbol{x}^{(n)}$,而隐式变量在其条件分布下任意采样,并在这种情况下进行玻耳兹曼机的仿真,那么第一项 $\langle y_i y_j \rangle_{P(\boldsymbol{h}\mid\boldsymbol{x}^{(n)},\boldsymbol{W})}$ 就是 y_i 和 y_j 之间的相关函数。

第二项 $\langle y_i y_j \rangle_{P(\boldsymbol{x},\boldsymbol{h}\mid\boldsymbol{W})}$ 是当玻耳兹曼机从它的模型分布中生成样本时,y_i 和 y_j 之间的相关函数。

Hinton 和 Sejnowski 表明像带标记的移位总体这种非平凡总体,可以用带有隐式单元的玻耳兹曼机进行学习。隐式单元的作用就是特征检测器,即发现可能和 3 个移位之一相关联的模式。

因为对数似然函数梯度的计算依赖于采用蒙特卡罗方法得到的两个梯度之差,所以玻耳兹曼机的仿真非常耗时。因此,玻耳兹曼机没有得到广泛应用。如何利用更有效的计算创建具有相同能力的模型,是非常活跃的研究领域(Hinton 等,1995 年;Dayan 等,1995 年;Hinton 和 Ghahramani,1997 年;Hinton,2001 年;Hinton 和 Teh,2001 年)。

43.3 习 题

▷ **习题 43.3**[扩] 能否用一个没有隐式单元的玻耳兹曼机学习“竖条和横条”总体(图 43.2)?[你可能会感到惊讶!]

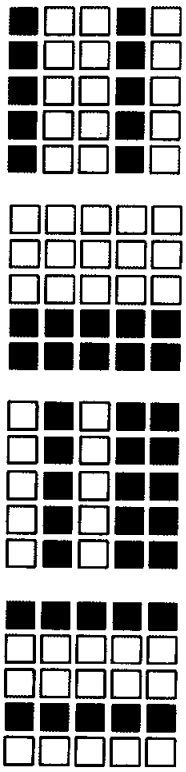


图 43.2 来自“竖条和横条”总体的 4 个样本。每个样本是这样生成的:首先选择一个方向,水平或垂直;然后,对于那个方向的每一行转子(分别是每个竖条或横条),以概率1/2对所有自旋进行切换。

第 44 章

多层网络的有监督学习

44.1 多层感知器

在神经网络课程中，如果不讨论有监督的多层网络，也称为反向传播网络，就不算完整。

*多层感知器*是一个前馈网络。它有输入神经元、隐神经元和输出神经元。隐神经元可以排列成一系列的层。大部分普通多层感知器有单一隐层，也就是“二层”网络，这里数字“二”所计的神经元层数中不包括输入。

这样一个前馈网络定义了从输入 $\boldsymbol{x}$ 到输出 $\boldsymbol{y}=\boldsymbol{y}(\boldsymbol{x};\boldsymbol{w},\mathscr{A})$ 的非线性参数化映射。输出是输入和参数 $\boldsymbol{w}$ 的连续函数；网络的体系结构，即映射的函数形式，用 $\mathscr{A}$ 表示。可以“训练”前馈网络来完成回归和分类任务。

回归网络

在回归问题的情形，具有一个隐层的网络的映射具有形式

$$\text{隐层：}\quad a_j^{(1)}=\sum_l w_{jl}^{(1)}x_l+\theta_j^{(1)};\quad h_j=f^{(1)}(a_j^{(1)}) \tag{44.1}$$

$$\text{输出层：}\quad a_i^{(2)}=\sum_j w_{ij}^{(2)}h_j+\theta_i^{(2)};\quad y_i=f^{(2)}(a_i^{(2)}) \tag{44.2}$$

其中，比如 $f^{(1)}(a)=\tanh(a)$，$f^{(2)}(a)=a$。l 标识输入 $x_1,\cdots,x_L$，j 标识隐式单元，而 i 标识输出。“权重”$\boldsymbol{w}$ 和“偏置”θ 一起构成了参数向量 $\boldsymbol{w}$。处于隐层的非线性 S 形函数 $f^{(1)}$ 会使神经网络具有比标准线性回归模型更大的计算灵活性。可以用图形方式将神经网络表示成一组互相连接的神经元所形成的层（图 44.1）。

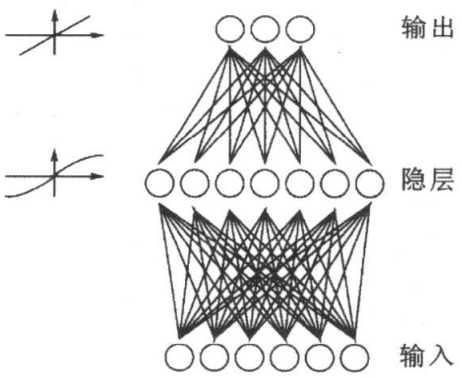


图 44.1 一个典型的二层网络，有 6 个输入、7 个隐式单元和 3 个输出。每条线代表一个权重。

这些网络能够实现什么功能？

在第 39 章探讨了单神经元的权空间，查看它能产生什么功能，类似地，现在探讨多层

网络的权空间。在图 44.2 和图 44.3 中，笔者选择含有大量（H 个）隐式单元的单输入－单输出网络，将偏置和权重 $\theta_j^{(1)}$、$w_{jl}^{(1)}$、$\theta_i^{(2)}$ 以及 $w_{ij}^{(2)}$ 都设定为随机值，并画出所得的函数 $y(x)$。笔者将隐式单元的偏置 $\theta_j^{(1)}$ 设定为来自均值为零、标准差为 σ_{bias} 的高斯分布随机值；输入隐层的权重 $w_{jl}^{(1)}$ 是标准差为 σ_{in} 的随机值；偏置 $\theta_i^{(2)}$ 和输出权重 $w_{ij}^{(2)}$ 是标准差为 σ_{out} 的随机值。

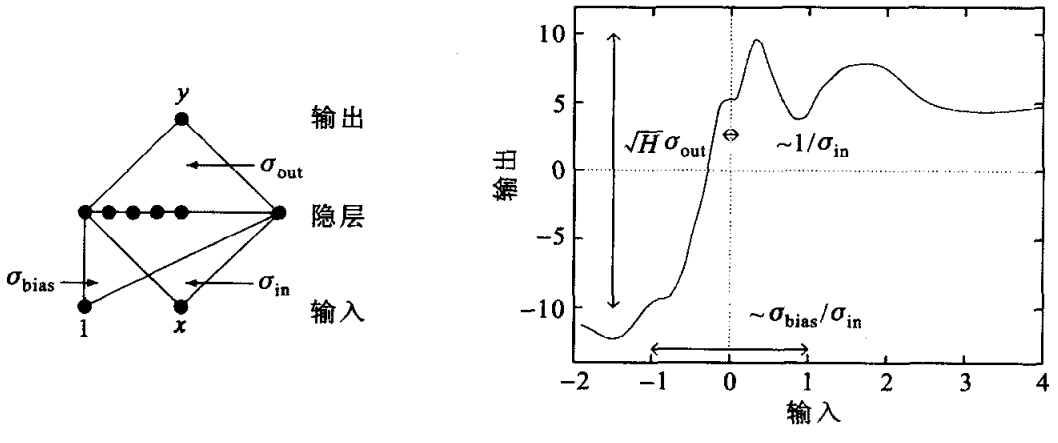


图 44.2 随机网络所产生函数的性质。具有随机权重的网络所产生的典型函数的垂直尺度的量级是 $\sqrt{H}\sigma_{\text{out}}$；函数显著变化的水平范围量级是 $\sigma_{\text{bias}}/\sigma_{\text{in}}$；最短的水平长度量级是 $1/\sigma_{\text{in}}$。由包含 $H=400$ 个隐式单元且具有参数 $\sigma_{\text{bias}}=4$，$\sigma_{\text{in}}=8$ 和 $\sigma_{\text{out}}=0.5$ 的高斯权重的随机网络所产生的函数。

所得函数的种类取决于 σ_{bias}、σ_{in} 和 σ_{out} 的值。权重和偏置越大，所得函数越复杂，所具有的功能越多，对输入变量也就越敏感。对于具有随机权重的网络所产生的典型函数，其垂直尺度的量级是 $\sqrt{H}\sigma_{\text{out}}$；函数显著变化的水平范围量级是 $\sigma_{\text{bias}}/\sigma_{\text{in}}$；最短的水平长度量级是 $1/\sigma_{\text{in}}$。

Radford Neal（1996 年）也已经证明在 $H\to\infty$ 的极限情况，通过随机化权值所产生的函数在统计特性上与隐式单元数无关；所以，非常有趣的是，函数的复杂性与模型中参数的个数无关。决定典型函数复杂性的是权重的特征幅度。因此，可以预计，当用模型来拟合实际数据时，控制权重的特征幅度是控制拟合函数复杂性的一条重要途径。

如图 44.4 所示为具有两个输入、一个输出的网络所产生的一个典型函数。应该和传统线性回归模型所产生的函数进行比较，那是一个平面。神经网络能够产生比线性回归更复杂的函数。

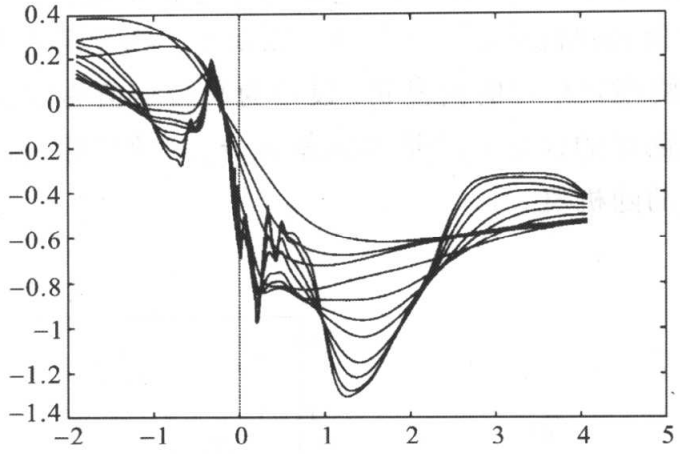


图 44.3 来自单输入网络函数的先验分布样本。对于 $\sigma_{\text{bias}}=8$, 6,4,3,2,1.6,1.2,0.8,0.4,0.3,0.2 这一列值中的每一个和 $\sigma_{\text{in}}=5\sigma_{\text{bias}}^{w}$,显示了一个随机函数。网络的另一些超参数是 $H=400$,$\sigma_{\text{out}}^{w}=0.05$。

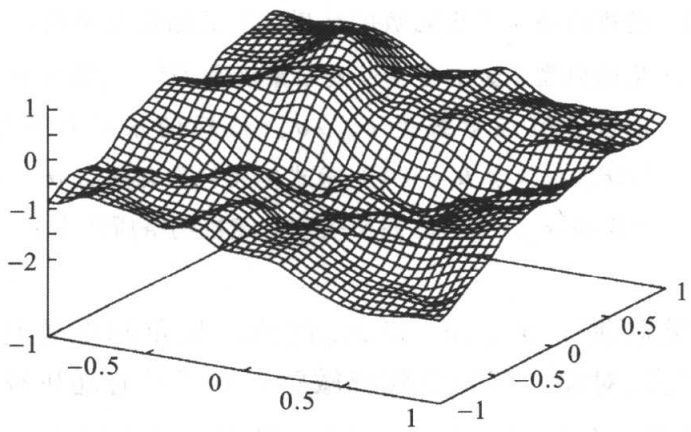


图 44.4 来自 2 - 输入网络先验的样本。典型函数由 $\{H,\sigma_{\text{in}}^{w},\sigma_{\text{bias}}^{w},\sigma_{\text{out}}^{w}\}=\{400,8.0,8.0,0.05\}$ 的 2 - 输入网络产生。

44.2 如何以传统方式训练回归网络

网络利用数据集 $D=\{\boldsymbol{x}^{(n)},\boldsymbol{t}^{(n)}\}$ 进行训练,它通过调整 $\boldsymbol{w}$ 来最小化误差函数,比如

$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_n \sum_i (t_i^{(n)} - y_i(\boldsymbol{x}^{(n)};\boldsymbol{w}))^2 \tag{44.3}$$

这个目标函数是一些项的和，每个输入/目标对$\{\boldsymbol{x},\boldsymbol{t}\}$一项，它度量了输出$\boldsymbol{y}(\boldsymbol{x},\boldsymbol{w})$和目标$\boldsymbol{t}$的接近程度。

通过重复计算E_D的梯度对目标函数进行最小化。梯度可以采用反向传播算法有效地计算(Rumelhart 等，1986 年)，它利用链式法则求导数。

通常要包括正则化因子(也称为权衰减)，于是目标函数修正为

$$M(\boldsymbol{w}) = \beta E_D + \alpha E_W \tag{44.4}$$

其中，如$E_W = \frac{1}{2}\sum_i w_i^2$。这个附加项有利于取小$\boldsymbol{w}$值，这样可以减轻模型对数据训练中噪声过度拟合的趋势。

Rumelhart 等(1986 年)证明通过$M(\boldsymbol{w})$的梯度下降，训练多层感知器以得到一些非平凡问题的解，如判断一幅图像是否对称。这些网络已经成功地应用于各种现实任务中，如英文文本的发音(Sejnowski 和 Rosenberg，1987 年)、多镜望远镜的聚焦(Angel 等，1990 年)。

44.3 用于推理的神经网络学习

上面的神经网络学习过程可以有以下概率解释。[这里重复并推广第 41 章的讨论。]

将误差函数解释为定义一个噪声模型，βE_D是负的对数似然

$$P(D|\boldsymbol{w},\beta,\mathcal{H}) = \frac{1}{Z_D(\beta)}\exp(-\beta E_D) \tag{44.5}$$

所以，式(44.3)中误差平方和E_D就是假设目标变量上的噪声为高斯分布，参数β定义了噪声水平$\sigma_\nu^2 = 1/\beta$。

同样，正则化因子可用参数的对数先验概率分布进行解释。

$$P(\boldsymbol{w}|\alpha,\mathcal{H}) = \frac{1}{Z_W(\alpha)}\exp(-\alpha E_W) \tag{44.6}$$

如果E_W具有上面定义的二次形式，则相应的先验分布是方差为$\sigma_w^2 = 1/\alpha$的高斯分布。概率模型$\mathcal{H}$指定了网络的体系结构$\mathcal{A}$、似然(44.5)和先验(44.6)。

于是，目标函数$M(\boldsymbol{w})$对应于已知数据时关于参数$\boldsymbol{w}$的推理

$$P(\boldsymbol{w}|D,\alpha,\beta,\mathcal{H}) = \frac{P(D|\boldsymbol{w},\beta,\mathcal{H})P(\boldsymbol{w}|\alpha,\mathcal{H})}{P(D|\alpha,\beta,\mathcal{H})} \tag{44.7}$$

$$= \frac{1}{Z_M}\exp(-M(\boldsymbol{w})) \tag{44.8}$$

(局部)最小化$M(\boldsymbol{w})$所得$\boldsymbol{w}$可解释成(局部)最可能的参数向量$\boldsymbol{w}_{\mathrm{MP}}$。

在现阶段，将$M(\boldsymbol{w})$解释成对数概率并未增加新的内容。但是在继续考虑其他

推理时，就会出现新的工具。首先，确定分类网络的概率解释，这里用到了同样的工具。

二进制分类网络

如果数据集合中的目标 $\boldsymbol{t}$ 是二进制分类标记(0,1)，很自然要使用输出 $y(\boldsymbol{x};\boldsymbol{w},\mathcal{A})$ 限定于0和1之间的神经网络，并将它解释成概率 $P(t=1|\boldsymbol{x},\boldsymbol{w},\mathcal{A})$。例如，具有一个隐层的网络可以用前馈方程(44.1)和(44.2)描述，其中 $f^{(2)}(a)=1/(1+e^{-a})$。用负对数似然来代替误差函数 βE_D。

$$G(\boldsymbol{w}) = -\left[\sum_n t^{(n)} \ln y(\boldsymbol{x}^{(n)};\boldsymbol{w}) + (1-t^{(n)})\ln(1-y(\boldsymbol{x}^{(n)};\boldsymbol{w}))\right] \quad (44.9)$$

于是，总的目标函数是 $M=G+\alpha E_W$。注意这里不包括参数 β(因为没有高斯噪声)。

多类分类网络

对于多类分类问题，可以把目标值表示成向量 $\boldsymbol{t}$，其中只有单一元素被设为1，表示正确的类，所有其他元素都设为0。在这种情况下，比较适合采用“软最大化”网络，它具有总和为1的耦合输出；这些输出可解释成类概率 $y_i=P(t_i=1|\boldsymbol{x},\boldsymbol{w},\mathcal{A})$。式(44.2)的最后部分被替换成

$$y_i = \frac{e^{a_i}}{\sum_{i'} e^{a_{i'}}} \quad (44.10)$$

此时负对数似然为

$$G(\boldsymbol{w}) = -\sum_n \sum_i t_i^{(n)} \ln y_i(\boldsymbol{x}^{(n)};\boldsymbol{w}) \quad (44.11)$$

如同回归网络的情况，目标函数 $M(\boldsymbol{w})=G+\alpha E_W$ 的最小化对应于形如式(44.8)的推理。可以从这个解释中得到许多有用的结果。

44.4 对有监督前馈神经网络使用贝叶斯方法的好处

以统计的观点，有监督的神经网络就是非线性曲线拟合设备。然而，曲线拟合并非一项轻而易举的工作。如图44.5所示，插值模型的有效复杂性是非常重要的。考虑影响模型复杂度的一个控制参数，例如正则化常数 α(权衰减参数)。当控制参数发生变化而增加模型复杂度(从图44.5(a~c)依次向下，从左到右穿过图44.5(d))，模型对训练数据的最佳拟合也越来越好。然而，作为模型的经验性能，检验误差先下降，再增加。一个过于复杂的模型会对数据过度拟合，不便推广。这个问题也可能使得多层感知器的体系结构、径向基函数网络的基函数半径以及在多维回归问题中输入变量本身等选择变得复杂。因此，寻找适合数据的模型控制参数值就是一个重要且非平凡的问题。

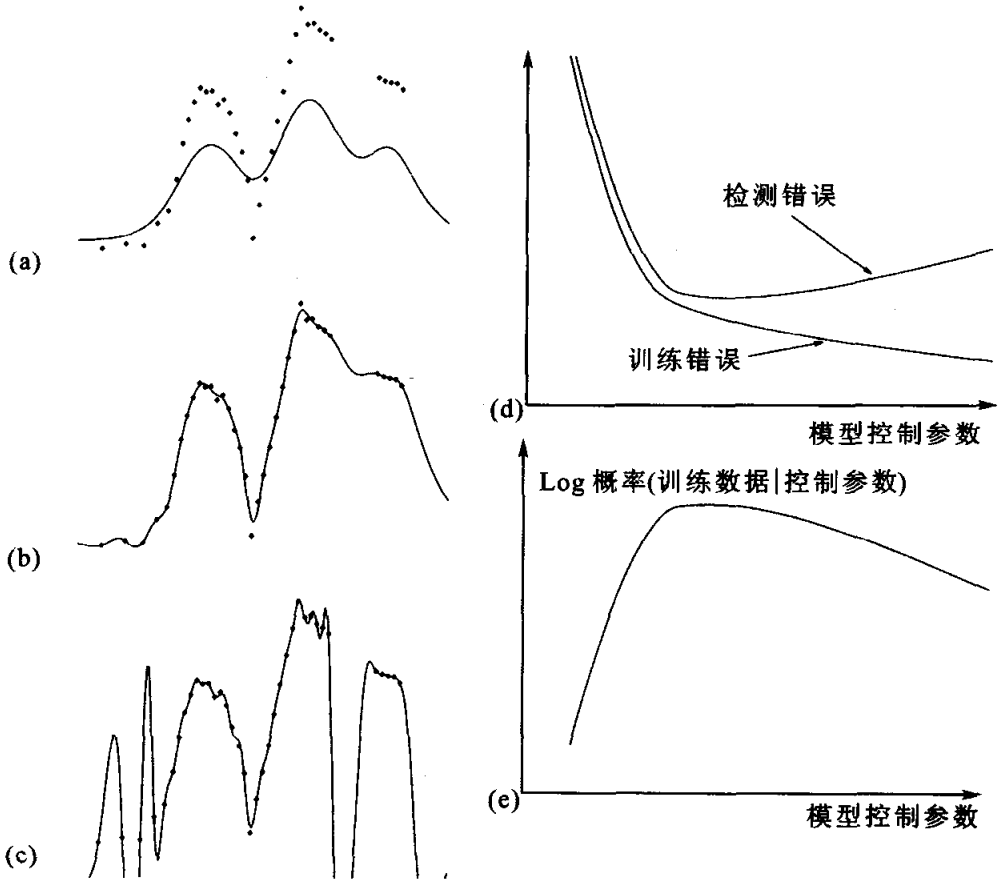


图 44.5 模型复杂性的优化。子图(a~c)显示用于一个简单数据集插值的径向基函数模型,有一个输入变量和一个输出变量。当正则化常数发生变化以增加模型复杂度时(从(a)到(c)),插值函数可以使训练数据拟合得越来越好,但是超过某一点时,模型的推广能力(检测误差的能力)减弱。概率论可使人们无需检验集合就优化控制参数。

可以通过贝叶斯方法控制模型复杂度,来解决过度拟合问题。

如果给出模型的概率解释,可以计算对应控制参数其他值的显著性。如第 28 章所述,结果证明过于复杂的模型往往是可能性较低的,显著性 P(数据|控制参数)可以用做优化模型控制参数的目标函数(图 44.5(e))。图 44.5(b)显示了最大化显著性的 α 的设置。

模型控制参数的贝叶斯优化有 4 个重要优点。(1) 无需“测试集”或“校验集”,因此所有可供训练数据都可用于模型拟合和模型比较。(2) 可以在线优化正则化常数,也就是和普通模型参数同时优化。(3) 与互验证方法相比,贝叶斯目标函数没有噪声。(4) 可以计算显著性关于控制参数的梯度,于是有可能同时优化大量控制参数。

概率建模也能自然地处理不确定性。它提供了一种独特的方法将参数的不确定性包含到预测中,就是边缘化;正如在第 41 章已经看到的,这个过程能做出更好的预

测。如图 44.6 所示为一个受训神经网络的预测误差限。



图 44.6 一个受训回归网络的预测误差限。实线是对以数据点训练的多层感知器最佳拟合参数的预测。误差限(虚线)由参数 $\boldsymbol{w}$ 的不确定性产生。注意当数据点稀少时,误差限变大。

贝叶斯推理的实现

如第 41 章所述,多层网络的贝叶斯推理可以通过蒙特卡罗采样或使用高斯近似的确定性方法来实现(Neal,1996 年;MacKay,1992c)。

在数据建模的贝叶斯框架内,很容易改进概率模型。例如,如果确信一个问题中的某些输入变量与预测量无关,但不知道是哪些量,就可以定义具有多个超参数的新模型,来反映有关输入变量的不确定性(MacKay, 1994b; Neal, 1996 年; MacKay, 1995b);这些模型可自动地从数据中推断哪些输入变量和问题有关。

44.5 习 题

习题 44.1[4] **如何度量分类器的品质**。你刚写完一个分类器的新算法,现在要度量它对一个测试集合的性能,并和其他分类器进行比较。你要采用什么性能度量?有几个标准答案。假设分类器的输出是 $y(\boldsymbol{x})$,其中 $\boldsymbol{x}$ 是输入,对此不进一步讨论,真实的目标值是 t。对于最简单的分类器,y 和 t 都是二进制变量,你也许想考虑 y 和 t 是更一般目标的情况。

在测试集上最常用的性能度量是*误差率*——分类器*误划分*的比例。这样的度量迫使分类器只给出 0/1 输出,并忽略分类器可能提供的任何其他信息——例如,关于预测确定性的指示。遗憾的是,误差率未必度量分类器的输出*提供了多少信息*。考虑频率表,即分类器 0/1 输出(横轴)和实际 0/1 变量(纵轴)的联合频率。所给数字都用百分比表示。误差率 e 是两个非对角数之和,可称为假正率 e_+ 和假负率 e_-。

在下面 3 个分类器中,A 和 B 具有相同的误差率 10%,C 有较大的误差率 12%。

分类器 A

t \ y	0	1
0	90	0
1	10	0

分类器 B

t \ y	0	1
0	80	10
1	0	10

分类器 C

t \ y	0	1
0	78	12
1	0	10

但是很显然分类器 A 仅对所有情况都简单猜测输出为 0，没有携带关于 t 的信息；但是分类器 B 具有包含信息的输出：如果 $y=0$，可以肯定实际的 t 就是零；如果 $y=1$，则有 50% 的可能性 $t=1$，而相应于先验概率 $P(t=1)=0.1$。分类器 C 所包含的信息比 B 略少些，但却比不含信息的分类器 A 有用得多。①

作为性能度量，误差率的一种改进方法是记录 (e_+,e_-) 对，即假正误差率和假负误差率（对分类器 A 和 B 分别是 $(0,0.1)$ 和 $(0.1,0)$）。在两种误差对应于不同代价的应用中，对这两种差错概率进行区分尤为重要。然而，“误差率对”存在两个问题。

- 首先，如果只简单告诉你分类器 A 有误差率 $(0,0.1)$ 和分类器 B 有误差率 $(0.1,0)$，就不能立刻看出分类器 A 实际上是完全没有价值的。的确，应该有一个性能度量，能够给 A 最差的得分！
- 其次，如果研究多类分类器问题，如数字识别，那么误差类型数会从 2 增加到 $10\times9=90$ 个——每种可能的类 t 与 t' 的混淆有一个。如果存在某种明智的方式把这 90 个数归并到一个可排序的数中，且比误差率更合理，那么会更好。

不喜欢误差率的另一个原因在于，对于准确地说明不确定性的分类器没有适当的奖励。考虑有 3 个输出值的分类器：“0”、“1”和一个拒绝类“?”，后者表示分类器不确定应该属于哪一类。考虑由以下百分比频率表所表示的分类器 D 和 E。

分类器 D

t \ y	0	?	1
0	74	10	6
1	0	1	9

分类器 E

t \ y	0	?	1
0	78	6	6
1	0	5	5

两个分类器都有 $(e_+,e_-,r)=(6\%,0\%,11\%)$。但它们是同样好的分类器吗？比较分类器 E 和 C，这两个分类器是等价的。E 只是伪装的 C——只要在 C 的输出为“1”时投掷一枚硬币，确定输出是“1”还是“?”，就可以得到 E。所以 E 等价于 C，因而比 B 差。现在来比较 D 和 B，你能证明 D 是比 B 包含更多信息量（从而比 E 优越）的分类器吗？然而 D 和 E 却有相同的 (e_+,e_-,r) 值。

人们经常画出*误差 - 拒绝曲线*（也就是 ROC 曲线；ROC 代表“接收机工作特性”），它表明当 r 从 0 变为 1 时，总的 $e=(e_++e_-)$ 随 r 的变化曲线，并用这些曲线对分类器进行比较（图 44.7）。[对于二进制分类问题的特例，可以转而画出 e_+ 相对

① 按常识为分类器排序：（最佳）$B>C>A$（最差）。按误差率为分类器排序：（最佳）$A=B>C$（最差）。

于 e_- 的曲线。]但是如已经看到的那样,误差率是有些模糊的性能度量。画出一个误差率与另一误差率的函数关系曲线,是否能够克服它的这一缺点呢?

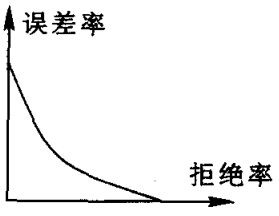


图 44.7 一条误差 - 拒绝曲线。有些人用这条曲线下面的面积来度量分类器的品质。

对于这个习题,或者构造一个明确的例子来表明误差 - 拒绝曲线以及曲线下的面积并非比较分类器的好方法;或者证明它们是好方法。

作为比较分类器的建议性替代方法,考虑输出和目标之间的*互信息*

$$I(T;Y) \equiv H(T) - H(T|Y) = \sum_{y,t} P(y)P(t|y)\log\frac{P(t)}{P(t|y)} \qquad (44.12)$$

它度量了分类器输出携带多少"比特"关于目标的信息。

计算上面 A ~ E 等几个分类器的互信息。

研究这个性能度量,并讨论它是否是一个有用的度量。实用时它存在不足吗?

关于第 45 章

多层感知器之类的前馈神经网络是非线性回归和分类问题的常用工具。从贝叶斯观点来看，神经网络模型的选择可以看成是定义非线性函数的先验概率分布，而神经网络的学习过程可以解释成未知函数的后验概率分布。（有些学习算法搜寻具有最大后验概率的函数，而其他蒙特卡罗方法从后验概率中抽样。）

在网络很大但其他方面都标准的极限情况下，Neal（1996 年）证明由贝叶斯神经网络确定的非线性函数的先验分布落在一类称为高斯过程的概率分布中。神经网络模型的超参数决定了高斯过程的特征长度。Neal 的观察启发了放弃参数化网络而直接采用高斯过程这一思想。于是，就可用对高斯过程的协方差矩阵的简单矩阵操作代替优化网络参数的计算。

这一章将回顾 Williams 和 Rasmussen（1996 年）、Neal（1997b）、Barber 和 Williams（1997 年）、Gibbs 和 MacKay（2000 年）关于这一思想的工作，还要评判前馈网络是否已经被其他技术所取代，以完成有监督的回归和分类任务。

▷ **习题 45.1**[2] 很抱歉这一章相当枯燥。没有简单的解释性例子，图也很少。这个习题要求你为自己创建有趣的图来解释这一章的思想。

在以下网址可以得到用自由语言 octave 编写的计算机演示源代码。

http://www.inference.phy.cam.ac.uk/mackay/itprnn/software.html

在以下网址可以得到 Radford Neal 关于高斯过程的软件：

http://www.cs.toronto.edu/~radford/

第 45 章

高斯过程

Rumelhart、Hinton 和 Williams(1986 年)关于神经网络有监督学习的文章发表之后,关于用非线性参数模型,如多层感知器和径向基函数,建立高维数据关系经验模型的兴趣十分高涨。在这些建模方法的贝叶斯解释中,假设数据$\{\boldsymbol{x}^{(n)}, t_n\}_{n=1}^N$背后的关系是以 $\boldsymbol{w}$ 为参数的非线性函数 $y(\boldsymbol{x})$,而模型对数据的自适应对应于已知数据时对函数的推理。以 $\boldsymbol{X}_N \equiv \{\boldsymbol{x}^{(n)}\}_{n=1}^N$表示一组输入向量,以 $\boldsymbol{t}_N \equiv \{t_n\}_{n=1}^N$表示相应的目标值集合。$y(\boldsymbol{x})$用如下后验概率分布描述。

$$P(y(\boldsymbol{x}) \mid \boldsymbol{t}_N, \boldsymbol{X}_N) = \frac{P(\boldsymbol{t}_N \mid y(\boldsymbol{x}), \boldsymbol{X}_N) P(y(\boldsymbol{x}))}{P(\boldsymbol{t}_N \mid \boldsymbol{X}_N)} \tag{45.1}$$

在右边的两项中,第一项 $P(\boldsymbol{t}_N \mid y(\boldsymbol{x}), \boldsymbol{X}_N)$是已知函数 $y(\boldsymbol{x})$时目标值的概率,对于回归问题,往往假定它是可分离的高斯分布;第二项 $P(y(\boldsymbol{x}))$是模型所假设函数的先验分布。在对参数模型的选择中,及对模型拟合过程中所用正则化因子的选择中,都隐含了这一先验分布。典型的先验分布规定函数 $y(\boldsymbol{x})$是连续和平滑的,高频功率小于低频功率,但是参数模型的使用使得先验的确切含义有些模糊。

现在,为了预测 t 的未来值,需要考虑的只有假设先验 $P(y(\boldsymbol{x}))$和假设噪声模型 $P(\boldsymbol{t}_N \mid y(\boldsymbol{x}), \boldsymbol{X}_N)$,与函数 $y(\boldsymbol{x}; \boldsymbol{w})$的参数化无关。

高斯过程建模的思想是把先验 $P(y(\boldsymbol{x}))$直接放在函数空间上,而不对 $y(\boldsymbol{x})$进行参数化。函数上最简单的先验类型被称为高斯过程。可以认为是将有限向量空间上的高斯分布推广到无限维函数空间。正如均值和协方差矩阵完全确定高斯分布那样,高斯过程由均值和协方差函数来确定。这里均值是 $\boldsymbol{x}$ 的函数(往往采用零函数),协方差是函数 $C(\boldsymbol{x}, \boldsymbol{x}')$,表示函数 y 在点 $\boldsymbol{x}$ 和 $\boldsymbol{x}'$的值之间的期望协方差。任何数据建模问题中的函数 $y(\boldsymbol{x})$被假设为高斯分布的单一样本。对于许多不同空间和时间问题,高斯过程都是建好的模型——例如,布朗运动、Langevin 过程和 Wiener 过程都是高斯过程的例子;广泛应用于建模语音波形的 Kalman 滤波器也是高斯过程模型;地理统计中的“kriging”方法就是高斯过程回归方法。

高斯过程的潜力

也许有人认为用高斯分布这么简单的东西不可能再现神经网络插值方法的有趣特性,但是大家将会看到,许多流行的非线性插值方法等价于特别的高斯过程。(笔者采用术语“插值”同时表示“回归”问题——用曲线拟合噪声数据,及拟合严格通过已知数据点的插值函数的任务。)

也许有人会认为用无限维函数空间上的先验进行推理,计算复杂度会变得无限

大。但是通过集中考虑观测数据和希望预测的量的联合概率分布，就能够用数据点数 N 的多项式函数量级的资源做出预测。

45.1 非线性回归的标准方法

问题

已知 N 个数据点 $\boldsymbol{X}_N, \boldsymbol{t}_N = \{\boldsymbol{x}^{(n)}, t_n\}_{n=1}^N$。输入 $\boldsymbol{x}$ 是有固定输入维度 I 的向量。目标 t 可以是实数，这时的任务是回归或插值，也可以是类变量，例如 $t \in \{0,1\}$，这时的任务是分类。暂时把注意力集中在回归问题上。

假设观测数据所隐含的函数为 $y(\boldsymbol{x})$，现在的任务是由已知数据推断出函数，并预测函数在新点 $\boldsymbol{x}^{(N+1)}$ 上的值——或者观测值 t_{N+1}。

问题的参数方法

在回归的参数方法中，通过由参数 $\boldsymbol{w}$ 参数化的非线性函数 $y(\boldsymbol{x};\boldsymbol{w})$ 表示未知函数 $y(\boldsymbol{x})$。

例 45.2 固定基函数。利用一组基函数 $\{\phi_h(\boldsymbol{x})\}_{h=1}^H$ 可得

$$y(\boldsymbol{x};\boldsymbol{w}) = \sum_{h=1}^{H} w_h \phi_h(\boldsymbol{x}) \tag{45.2}$$

如果基函数是 $\boldsymbol{x}$ 的非线性函数，如以固定点 $\{\boldsymbol{c}_h\}_{h=1}^H$ 为中心的径向基函数

$$\phi_h(\boldsymbol{x}) = \exp\left[-\frac{(\boldsymbol{x}-\boldsymbol{c}_h)^2}{2r^2}\right] \tag{45.3}$$

则 $y(\boldsymbol{x};\boldsymbol{w})$ 是 $\boldsymbol{x}$ 的非线性函数；但是，由于 y 与参数 $\boldsymbol{w}$ 的依存关系是线性的，有时也称它为“线性”模型。用神经网络术语，这个模型就像是一个多层网络，从输入层到非线性隐层的连接是固定的；只有输出权重 $\boldsymbol{w}$ 是自适应的。

其他可能的固定基函数组包括诸如 $\phi_h(\boldsymbol{x}) = x_i^p x_j^q$ 的多项式，其中 p 和 q 是和 h 有关的整数幂。

例 45.3 自适应基函数。另一方面，可以从基函数得到函数 $y(\boldsymbol{x})$，该基函数依赖于包括在向量 $\boldsymbol{w}$ 中的另外参数。在一个具有非线性隐式单元和一个线性输出的两层前馈神经网络中，函数可以写成

$$y(\boldsymbol{x};\boldsymbol{w}) = \sum_{h=1}^{H} w_h^{(2)} \tanh\left(\sum_{i=1}^{I} w_{hi}^{(1)} x_i + w_{h0}^{(1)}\right) + w_0^{(2)} \tag{45.4}$$

其中 I 是输入空间的维数，权向量 $\boldsymbol{w}$ 由输入权重 $\{w_{hi}^{(1)}\}$、隐单元偏置 $\{w_{h0}^{(1)}\}$、输出权 $\{w_h^{(2)}\}$ 和输出偏置 $w_0^{(2)}$ 组成。在这个模型中，y 和 $\boldsymbol{w}$ 之间的依存关系是非线性的。

选择完参数化之后，通过推断参数 $\boldsymbol{w}$ 来推理函数 $y(\boldsymbol{x};\boldsymbol{w})$。参数的后验概率是

$$P(\boldsymbol{w}|t_N,\boldsymbol{X}_N)=\frac{P(t_N|\boldsymbol{w},\boldsymbol{X}_N)P(\boldsymbol{w})}{P(t_N|\boldsymbol{X}_N)} \tag{45.5}$$

因子 $P(t_N|\boldsymbol{w},\boldsymbol{X}_N)$ 表示已知参数 $\boldsymbol{w}$（因此函数 y 也已知）时观测数据点的概率。这个概率分布往往取为可分高斯分布，每个数据点 t_n 和潜在值 $y(\boldsymbol{x}^{(n)};\boldsymbol{w})$ 相差一个加性噪声。因子 $P(\boldsymbol{w})$ 指定了参数的先验概率分布。它也可以取为可分高斯分布。如果 y 与 $\boldsymbol{w}$ 的依存关系是非线性的，后验分布 $P(\boldsymbol{w}|t_N,\boldsymbol{X}_N)$ 一般不是高斯分布。

可以用各种不同的方式来实现推理。在拉普拉斯方法中，用 $\boldsymbol{w}$ 来最小化目标函数

$$M(\boldsymbol{w})=-\ln[P(t_N|\boldsymbol{w},\boldsymbol{X}_N)P(\boldsymbol{w})] \tag{45.6}$$

找到局部最可能的参数，然后用 M 的曲率 $\partial^2 M(\boldsymbol{w})/\partial w_i\partial w_j$ 来定义 $\boldsymbol{w}$ 的误差限。或者采用更通用的马尔科夫链蒙特卡罗技术从后验分布 $P(\boldsymbol{w}|t_N,\boldsymbol{X}_N)$ 获得样本。

得到了已知数据时推理 $\boldsymbol{w}$ 的一种表示之后，通过在参数上进行边缘化得到预测。

$$P(t_{N+1}|t_N,\boldsymbol{X}_{N+1})=\int \mathrm{d}^H\boldsymbol{w}P(t_{N+1}|\boldsymbol{w},\boldsymbol{x}^{(N+1)})P(\boldsymbol{w}|t_N,\boldsymbol{X}_N) \tag{45.7}$$

如果得到后验分布 $P(\boldsymbol{w}|t_N,\boldsymbol{X}_N)$ 的高斯表示，那么可以直接计算这个积分。或者采用蒙特卡罗方法，由后验分布 $P(\boldsymbol{w}|t_N,\boldsymbol{X}_N)$ 生成 R 个样本 $\boldsymbol{w}^{(r)}$，将预测分布近似为

$$P(t_{N+1}|t_N,\boldsymbol{X}_{N+1})\simeq\frac{1}{R}\sum_{r=1}^{R}P(t_{N+1}|\boldsymbol{w}^{(r)},\boldsymbol{x}^{(N+1)}) \tag{45.8}$$

非参数方法

在非参数方法中进行预测，不用对未知函数 $y(\boldsymbol{x})$ 进行显式参数化；$y(\boldsymbol{x})$ 存在于 $\boldsymbol{x}$ 所有连续函数的无限维空间中。回归问题的一个众所周知的非参数方法是样条平滑方法（Kimeldorf 和 Wahba，1970 年）。一维回归问题的样条解可描述为：定义 $y(\boldsymbol{x})$ 的估计器为最小化如下泛函的函数 $\hat{y}(\boldsymbol{x})$。

$$M(y(x))=\frac{1}{2}\beta\sum_{n=1}^{N}(y(x^{(n)})-t_n)^2+\frac{1}{2}\alpha\int \mathrm{d}x[y^{(p)}(x)]^2 \tag{45.9}$$

其中 $y^{(p)}$ 是 y 的第 p 阶导数，p 是一个正整数。如果 p 设为 2，则所得函数 $\hat{y}(\boldsymbol{x})$ 是立方样条，即分段立方函数，在数据点 $\{x^{(n)}\}$ 上有“结”——二阶导数不连续。

这种估计方法可以解释成贝叶斯方法，函数 $y(\boldsymbol{x})$ 的先验为

$$\ln P(y(x)|\alpha)=-\frac{1}{2}\alpha\int \mathrm{d}x[y^{(p)}(x)]^2+\text{常数} \tag{45.10}$$

在独立高斯噪声的假设下，测量数据 $t_N=\{t_n\}_{n=1}^N$ 的概率为

$$\ln P(t_N|y(x),\beta)=-\frac{1}{2}\beta\sum_{n=1}^{N}(y(x^{(n)})-t_n)^2+\text{常数} \tag{45.11}$$

[式(45.10)和式(45.11)中的常数分别是 α 和 β 的函数。严格地说，因为加在 $y(\boldsymbol{x})$ 上的阶次为 $(p-1)$ 的任意多项式没有约束，先验(45.10)是非适定的。通过在式

(45.10)上添加合适的$(p-1)$项,这个非适定性很容易得到修正。]已知关于式(45.9)中函数的这个解释,$M(y(x))$在一个附加常数范围内等于后验概率$P(y(x)|t_N,\alpha,\beta)$的负对数,且样条估计过程可解释为产生一个贝叶斯 MAP 估计。贝叶斯视角容许得到样条估计的误差限和从后验分布中抽取典型样本,它给出推断超参数α和β的一种自动方法。

评论

样条先验是高斯过程

式(45.10)所定义的先验分布是高斯过程的第一个例子。若不考虑数学精确性,高斯过程可定义为在函数$y(x)$的空间上写成如下形式的概率分布。

$$P(y(x)|\mu(x),A)=\frac{1}{Z}\exp\left[-\frac{1}{2}(y(x)-\mu(x))^{\mathrm{T}}A(y(x)-\mu(x))\right] \tag{45.12}$$

其中$\mu(x)$是均值函数,A是线性算子,两个函数的内积$y(x)^{\mathrm{T}}z(x)$可定义为,例如$\int \mathrm{d}x y(x)z(x)$。

如果用D表示把$y(x)$映射到其导数的线性算子,可以将式(45.10)写成

$$\ln P(y(x)|\alpha)=-\frac{1}{2}\alpha\int \mathrm{d}x[D^p y(x)]^2+\text{常数}=-\frac{1}{2}y(x)^{\mathrm{T}}Ay(x)+\text{常数} \tag{45.13}$$

它和式(45.12)具有同样的形式,其中$\mu(x)=0$和$A\equiv[D^p]^{\mathrm{T}}D^p$。

为了使式(45.12)中的先验分布是适定先验,A必须是正定算子,即满足对于除$y(x)=0$以外的所有函数$y(x)$,都有$y(x)^{\mathrm{T}}Ay(x)>0$。

可以将样条写成参数模型

与式(45.2)一样,样条可以用一个固定基函数的无限集表示,过程如下。首先重新调节x轴尺度,使得区间$(0,2\pi)$比所感兴趣的x值范围要宽得多。设基函数是傅里叶集合$\{\cos hx,\sin hx,h=0,1,2,\cdots\}$,因此函数是

$$y(x)=\sum_{h=0}^{\infty}w_{h(\cos)}\cos(hx)+\sum_{h=1}^{\infty}w_{h(\sin)}\sin(hx) \tag{45.14}$$

利用正则化因子

$$E_w(w)=\sum_{h=0}^{\infty}\frac{1}{2}h^{\frac{p}{2}}w_{h(\cos)}^2+\sum_{h=1}^{\infty}\frac{1}{2}h^{\frac{p}{2}}w_{h(\sin)}^2 \tag{45.15}$$

来定义w的高斯先验

$$P(w|\alpha)=\frac{1}{Z_w(\alpha)}\exp(-\alpha E_w) \tag{45.16}$$

如果$p=2$,就有如同式(45.9)那样的立方样条正则化因子$E_w(w)=\int y^{(2)}(x)^2\mathrm{d}x$;

如果 $p=1$，正则化因子就是 $E_w(\boldsymbol{w})=\int y^{(1)}(x)^2\mathrm{d}x$，等等。（为了使先验合适，必须在项 $w_{0(\cos)}$ 上增加另一个正则化因子。）因此从先验 $P(y(x))$ 的角度来看，“非参数”样条法和其他参数法之间不存在本质差别。

表示与预测无关

至少从预测的观点来看，有两个感兴趣的对象。第一个是式(45.7)所定义的条件分布 $P(t_{N+1}|\boldsymbol{t}_N,\boldsymbol{X}_{N+1})$。如果希望和其他模型进行比较，另一个感兴趣对象就是已知模型下所有观测数据的联合概率，即显著性 $P(\boldsymbol{t}_N|\boldsymbol{X}_N)$，它是出现在式(45.5)中的归一化常数。两个量都不涉及未知函数 $y(x)$ 的表示。所以最终选择的表示是无关的。

现在要讨论的问题是，对于常用参数模型，这两个量具有什么形式？对于固定基函数的标准模型和未知参数为高斯分布的情况，在已知模型的条件下，所有被观测数据的联合概率 $P(\boldsymbol{t}_N|\boldsymbol{X}_N)$ 是均值为零、协方差矩阵由基函数确定的多变量高斯分布；这意味着条件分布 $P(t_{N+1}|\boldsymbol{t}_N,\boldsymbol{X}_{N+1})$ 也是一个高斯分布，其均值线性地依赖于目标值 $\boldsymbol{t}_N$。标准参数模型是高斯过程的简单例子。

45.2 从参数模型到高斯过程

线性模型

考虑利用 H 个固定基函数（例如式(45.3)所定义的一维径向基函数）的一个回归问题。

假设已经给定一列 N 个输入点 $\{\boldsymbol{x}^{(n)}\}$，并定义 $N\times H$ 矩阵 $\boldsymbol{R}$ 是基函数 $\{\phi_h(\boldsymbol{x})\}_{h=1}^{H}$ 在点 $\{\boldsymbol{x}_n\}$ 处的值的矩阵。

$$R_{nh}\equiv\phi_h(\boldsymbol{x}^{(n)}) \tag{45.17}$$

定义向量 $\boldsymbol{y}_N$ 为 $y(\boldsymbol{x})$ 在 N 个点上的值所组成的向量。

$$y_n\equiv\sum_h R_{nh}w_h \tag{45.18}$$

如果 $\boldsymbol{w}$ 的先验分布是零均值的高斯分布

$$P(\boldsymbol{w})=\mathrm{Normal}(\boldsymbol{w};\boldsymbol{0},\sigma_w^2\boldsymbol{I}) \tag{45.19}$$

由于 $\boldsymbol{y}$ 是 $\boldsymbol{w}$ 的线性函数，所以也是均值为零的高斯分布。$\boldsymbol{y}$ 的协方差矩阵是

$$\boldsymbol{Q}=\langle\boldsymbol{y}\boldsymbol{y}^{\mathrm{T}}\rangle=\langle\boldsymbol{R}\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{R}^{\mathrm{T}}\rangle=\boldsymbol{R}\langle\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\rangle\boldsymbol{R}^{\mathrm{T}} \tag{45.20}$$

$$=\sigma_w^2\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}} \tag{45.21}$$

所以 $\boldsymbol{y}$ 的先验分布是

$$P(\boldsymbol{y})=\mathrm{Normal}(\boldsymbol{y};\boldsymbol{0},\boldsymbol{Q})=\mathrm{Normal}(\boldsymbol{y};\boldsymbol{0},\sigma_w^2\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}) \tag{45.22}$$

对于任意选择的点 $\boldsymbol{X}_N$，函数 $\boldsymbol{y}$ 的 N 个值向量服从高斯分布这一结果都是正确的。这是定义一个高斯过程的特性。如果对任何有限的选择点 $\boldsymbol{x}^{(1)},\boldsymbol{x}^{(2)},\cdots,\boldsymbol{x}^{(N)}$，密度 $P(y(\boldsymbol{x}^{(1)}),y(\boldsymbol{x}^{(2)}),\cdots,y(\boldsymbol{x}^{(N)}))$ 都是高斯的，则函数 $y(\boldsymbol{x})$ 的概率分布是一个高斯过程。

现在，如果基函数的个数 H 小于数据点数 N，则矩阵 $\boldsymbol{Q}$ 不会满秩。此时 $\boldsymbol{y}$ 的概率分布可想像为 $\boldsymbol{y}$ 存在的 N 维空间中的 H 维子空间上的扁平椭圆薄饼。

目标值是什么情况？如果假设每个目标值 t_n 与相应的函数值 y_n 只相差一个方差为 σ_ν^2 的加性高斯噪声，则 $\boldsymbol{t}$ 也有一个高斯先验分布

$$P(\boldsymbol{t}) = \text{Normal}(\boldsymbol{t};\boldsymbol{0},\boldsymbol{Q}+\sigma_\nu^2\boldsymbol{I}) \tag{45.23}$$

用 $\boldsymbol{C}$ 表示 $\boldsymbol{t}$ 的协方差矩阵

$$\boldsymbol{C} = \boldsymbol{Q}+\sigma_\nu^2\boldsymbol{I} = \sigma_w^2\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}+\sigma_\nu^2\boldsymbol{I} \tag{45.24}$$

无论 $\boldsymbol{Q}$ 是否满秩，因为 $\sigma_\nu^2\boldsymbol{I}$ 是满秩的，所以协方差矩阵 $\boldsymbol{C}$ 总是满秩的。

协方差矩阵 $\boldsymbol{Q}$ 看起来像什么？一般来说，$\boldsymbol{Q}$ 的 (n,n') 元素是

$$Q_{nn'} = [\sigma_w^2\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}]_{nn'} = \sigma_w^2\sum_h \phi_h(\boldsymbol{x}^{(n)})\phi_h(\boldsymbol{x}^{(n')}) \tag{45.25}$$

而 $\boldsymbol{C}$ 的 (n,n') 元素是

$$C_{nn'} = \sigma_w^2\sum_h \phi_h(\boldsymbol{x}^{(n)})\phi_h(\boldsymbol{x}^{(n')}) + \delta_{nn'}\sigma_\nu^2 \tag{45.26}$$

其中若 $n=n'$，则 $\delta_{nn'}=1$，其余都是 0。

例 45.4 以一维情形的径向基函数为例。如果假设基函数是均匀分布的，其中标记为 h 的基函数以点 $x=h$ 为中心，则 $Q_{nn'}$ 的表示最简单；令 $H\to\infty$，使得关于 h 的求和变成积分；为了避免有一个随着 H 发散的协方差矩阵，最好使 σ_w^2 按 $S/(\Delta H)$ 缩放，其中 ΔH 是 x 轴上单位长度的基函数个数，S 是一个常数，则

$$Q_{nn'} = S\int_{h_{\min}}^{h_{\max}} \mathrm{d}h\,\phi_h(x^{(n)})\phi_h(x^{(n')}) \tag{45.27}$$

$$= S\int_{h_{\min}}^{h_{\max}} \mathrm{d}h\exp\left[-\frac{(x^{(n)}-h)^2}{2r^2}\right]\exp\left[-\frac{(x^{(n')}-h)^2}{2r^2}\right] \tag{45.28}$$

如果积分限为 $\pm\infty$，可求解积分

$$Q_{nn'} = \sqrt{\pi r^2}S\exp\left[-\frac{(x^{(n')}-x^{(n)})^2}{4r^2}\right] \tag{45.29}$$

现在得到关于插值问题的一个新视角。不由基函数和参数的先验来指定函数的先验分布，只要通过协方差函数就可以简单地对先验进行概括

$$C(x^{(n)},x^{(n')}) \equiv \theta_1\exp\left[-\frac{(x^{(n')}-x^{(n)})^2}{4r^2}\right] \tag{45.30}$$

这里采用一个新名称 θ_1，表示前面之外的常数项。

从这个特例加以推广，插值方法的前景就开阔了。已知任何有效协方差函数 $C(\boldsymbol{x},\boldsymbol{x}')$（随后就会讨论“有效”为何意），可以定义位置 $\boldsymbol{X}_N$ 上的 N 个函数值的协方差矩阵 $\boldsymbol{Q}$ 为

$$Q_{nn'} = C(\boldsymbol{x}^{(n)},\boldsymbol{x}^{(n')}) \tag{45.31}$$

假设高斯噪声时，相应的 N 个目标值的协方差矩阵 $\boldsymbol{C}$ 为

$$C_{nn'} = C(\boldsymbol{x}^{(n)}, \boldsymbol{x}^{(n')}) + \sigma_\nu^2 \delta_{nn'} \tag{45.32}$$

总之，数据集合中 N 个目标值 t 的先验概率为

$$P(t) = \text{Normal}(t; \boldsymbol{0}, \boldsymbol{C}) = \frac{1}{Z} e^{-\frac{1}{2} t^T C^{-1} t} \tag{45.33}$$

如图 45.1 所示为源自这个高斯过程和其他一些简单高斯过程的样本。

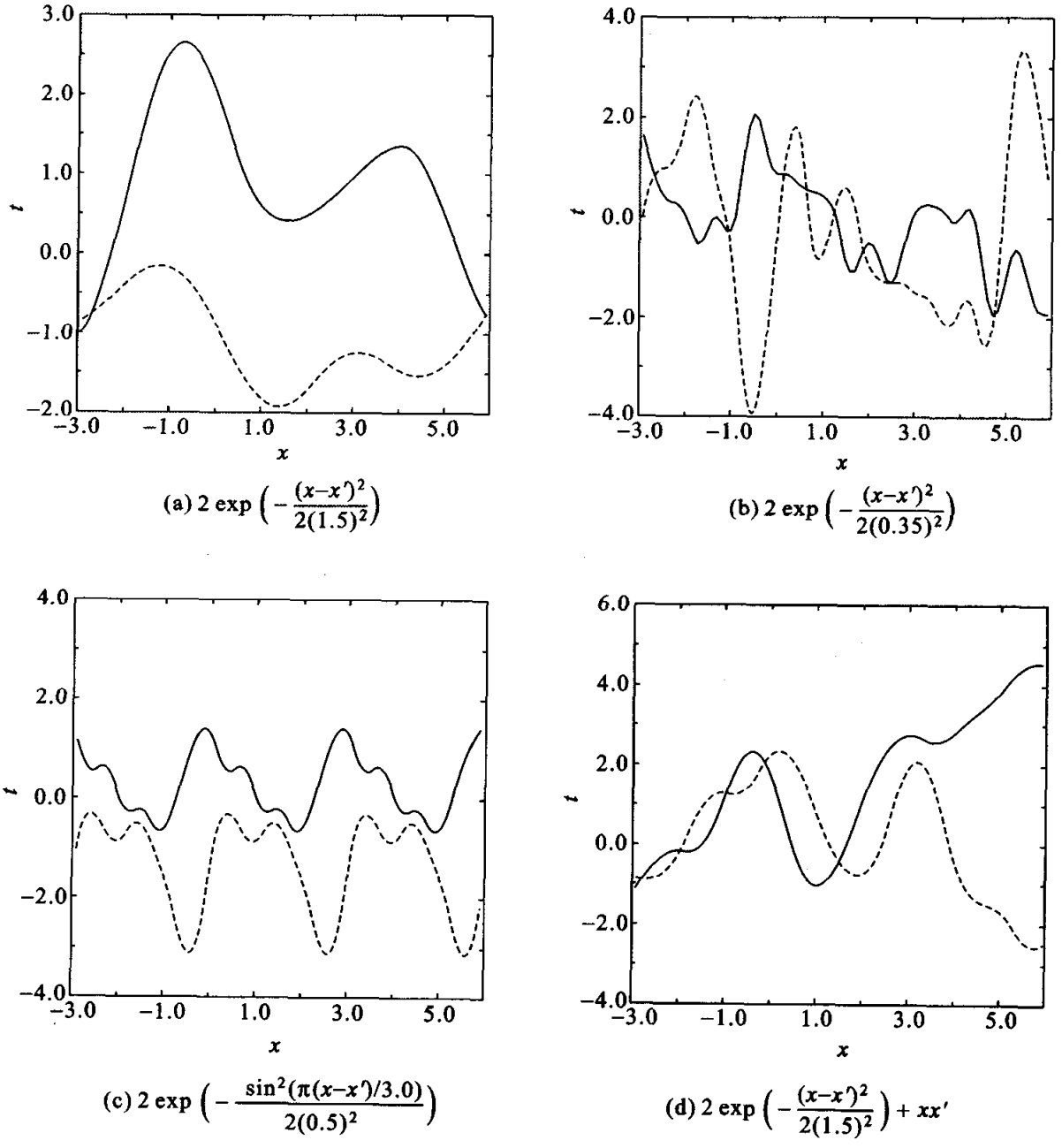


图 45.1 从高斯过程先验中抽取的样本。每个子图显示了从高斯过程先验中抽取的两个函数。4 个相应的协方差函数在每个图的下方给出。随着从(a)到(b)的长度尺度的减小，函数变化更快了。可以看出(c)中协方差函数的周期特性。(d)中协方差函数存在非平稳项 xx' 同一条直线的协方差相对应，所以典型函数包括线性趋势。来自 Gibbs(1997 年)。

多层神经网络与高斯过程

通过对含有大量隐式单元的标准多层感知器的一个选择所定义的函数先验分布的一些随机样本如图 44.2 和图 44.3 所示。这些样本与图 45.1 所示高斯过程样本并没有相差太远。实际上，Neal(1996 年)证明，如果假设标准“权衰减”先验，当隐神经元数趋于无穷时，具有单隐层的神经网络的特性(如式(45.4))趋向于高斯过程。高斯过程的协方差函数依赖于网络权重的具体先验假设，及隐式单元的激活函数。

45.3 在回归中使用一个已知高斯过程模型

已经花费一些时间谈论先验分布。现在回到数据和预测问题。如何利用高斯过程进行预测？

形成式(45.32)所定义的协方差矩阵 $\boldsymbol{C}$ 之后，现在的任务是在已知观测向量 $\boldsymbol{t}_N$ 的情况下推导 t_{N+1}。联合密度 $P(t_{N+1},\boldsymbol{t}_N)$ 是一个高斯分布；所以条件分布

$$P(t_{N+1}|\boldsymbol{t}_N)=\frac{P(t_{N+1},\boldsymbol{t}_N)}{P(\boldsymbol{t}_N)} \tag{45.34}$$

也是一个高斯分布。可用下标来区分不同大小的协方差矩阵 $\boldsymbol{C}$，如 $\boldsymbol{C}_{N+1}$ 是向量 $\boldsymbol{t}_{N+1}\equiv(t_1,\cdots,t_{N+1})^{\mathrm{T}}$ 的 $(N+1)\times(N+1)$ 协方差矩阵。定义 $\boldsymbol{C}_{N+1}$ 的子矩阵为

$$\boldsymbol{C}_{N+1}\equiv\begin{bmatrix}[\boldsymbol{C}_N] & [\boldsymbol{k}]\\ [\boldsymbol{k}^{\mathrm{T}}] & [\kappa]\end{bmatrix} \tag{45.35}$$

后验分布(45.34)为

$$P(t_{N+1}|\boldsymbol{t}_N)\propto\exp\left[-\frac{1}{2}[\boldsymbol{t}_N t_{N+1}]\boldsymbol{C}_{N+1}^{-1}\begin{bmatrix}\boldsymbol{t}_N\\ t_{N+1}\end{bmatrix}\right] \tag{45.36}$$

可以通过用蛮力求 $\boldsymbol{C}_{N+1}$ 的逆，来计算 t_{N+1} 后验分布的均值和标准差。然而，却有一个更精巧的预测分布表达式，只要在数据集合大小为 N 时对一些新点进行预测，该表达式就是有用的。利用划分求逆公式，可以把 $\boldsymbol{C}_{N+1}^{-1}$ 写成 $\boldsymbol{C}_N$ 和 $\boldsymbol{C}_N^{-1}$ 的形式(Barnett，1979 年)。

$$\boldsymbol{C}_{N+1}^{-1}=\begin{bmatrix}\boldsymbol{M} & \boldsymbol{m}\\ \boldsymbol{m}^{\mathrm{T}} & m\end{bmatrix} \tag{45.37}$$

其中

$$m=(\kappa-\boldsymbol{k}^{\mathrm{T}}\boldsymbol{C}_N^{-1}\boldsymbol{k})^{-1} \tag{45.38}$$

$$\boldsymbol{m}=-m\boldsymbol{C}_N^{-1}\boldsymbol{k} \tag{45.39}$$

$$\boldsymbol{M} = \boldsymbol{C}_N^{-1} + \frac{1}{m}\boldsymbol{m}\boldsymbol{m}^{\mathrm{T}} \tag{45.40}$$

把这个矩阵代入式(45.36),可得

$$P(t_{N+1}|\boldsymbol{t}_N) = \frac{1}{Z}\exp\left[-\frac{(t_{N+1}-\hat{t}_{N+1})^2}{2\sigma^2_{\hat{t}_{N+1}}}\right] \tag{45.41}$$

其中

$$\hat{t}_{N+1} = \boldsymbol{k}^{\mathrm{T}}\boldsymbol{C}_N^{-1}\boldsymbol{t}_N \tag{45.42}$$

$$\sigma^2_{\hat{t}_{N+1}} = \kappa - \boldsymbol{k}^{\mathrm{T}}\boldsymbol{C}_N^{-1}\boldsymbol{k} \tag{45.43}$$

新点上的预测均值是 $\hat{t}_{N+1}$,$\sigma_{\hat{t}_{N+1}}$ 定义了这个预测的误差限。注意:为了在 $\boldsymbol{x}^{(N+1)}$ 上进行预测,无需求 $\boldsymbol{C}_{N+1}$ 的逆,只需求 $\boldsymbol{C}_N$ 的逆。所以高斯过程容许实现一个包含 H 个基函数的模型,其中 H 可以比数据点数 N 大得多;计算复杂度的量级是 N^3,与 H 无关。[将在后面讨论降低这个开销的方法。]

由高斯过程产生的预测完全取决于协方差矩阵 $\boldsymbol{C}$。现在讨论可能选择来定义 $\boldsymbol{C}$ 的几种协方差函数,及如何针对数据自动地选择协方差函数。

45.4 协方差函数的例子

选择协方差函数的唯一约束是对任何点集 $\{\boldsymbol{x}_n\}_{n=1}^N$ 必须生成非负定的协方差矩阵。用 $\boldsymbol{\theta}$ 表示协方差函数的参数。$\boldsymbol{t}$ 的协方差矩阵的元素为

$$C_{mn} = C(\boldsymbol{x}^{(m)}, \boldsymbol{x}^{(n)}; \boldsymbol{\theta}) + \delta_{mn}\mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\theta}) \tag{45.44}$$

其中 C 是协方差函数,$\mathcal{N}$ 为噪声模型,可以是平稳的或在空间上变化的,如

$$\mathcal{N}(\boldsymbol{x};\boldsymbol{\theta}) = \begin{cases} \theta_3, & \text{对与输入无关的噪声} \\ \exp\left(\sum_{j=1}^{J}\beta_j\phi_j(\boldsymbol{x})\right), & \text{对与输入相关的噪声} \end{cases} \tag{45.45}$$

C 的连续性决定了来自高斯过程先验的典型样本的连续性。由 Abrahamsen(1997年)撰写的一篇关于高斯过程的综述性文章给出了许多有效的协方差函数。

平稳协方差函数

一个平稳协方差函数就是对于某些函数 D,满足平移不变性

$$C(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}) = D(\boldsymbol{x}-\boldsymbol{x}';\boldsymbol{\theta}) \tag{45.46}$$

即协方差只是距离的函数,也称为自协方差函数。如果 C 仅依赖于 $\boldsymbol{x}$ 和 $\boldsymbol{x}'$ 之间距离的幅值,则称协方差函数是齐次的。平稳协方差函数也可用函数 D 的傅里叶变换来描述,这就是高斯过程的功率谱。这个傅里叶变换必须是频率的正函数。一种构造

有效平稳协方差函数的方法是找到频率的一个正函数，并定义 D 是它的傅里叶逆变换。

例 45.5 令功率谱是频率的高斯函数。因为高斯的傅里叶变换还是高斯函数，这个功率谱的自协方差函数是距离的一个高斯函数。这个论证重新推导了在式(45.30)得到的协方差函数。

稍作推广，具有超参数 $\boldsymbol{\theta}=(\theta_1,\theta_2,\{r_i\})$ 的 C 的常见形式是

$$C(\boldsymbol{x},\boldsymbol{x}',\boldsymbol{\theta}) = \theta_1 \exp\left[-\frac{1}{2}\sum_{i=1}^{I}\frac{(x_i-x'_i)^2}{r_i^2}\right]+\theta_2 \tag{45.47}$$

$\boldsymbol{x}$ 是一个 I 维向量，r_i 是与输入 x_i 有关的长度尺度，y 在方向 i 上的长度尺度有很大变化。一个非常大的长度尺度意味着可以预计 y 基本上是那个输入的一个常值函数。这样的输入可以说是无关的，如同在神经网络的自动相干性确定法中那样(MacKay，1994a；Neal，1996 年)。超参数 θ_1 定义了一个典型函数在垂直尺度方向上的变化。超参数 θ_2 允许整个函数偏离零值一个未知常数——为了理解这一项，考察式(45.25)并考虑 $\phi(\boldsymbol{x})=1$ 的基函数。

另一个平稳协方差函数是

$$C(x,x') = \exp(-|x-x'|^{\nu}), \quad 0<\nu\leqslant 2 \tag{45.48}$$

当 $\nu=2$ 时，这是前面协方差函数的一个特例。当 $\nu\in(1,2)$ 时，来自这个先验的典型函数是平滑的，但不是解析函数。当 $\nu\leqslant 1$ 时，典型函数是连续的但不平滑。

在第 i 个输入方向上有已知周期 λ_i 的函数建模协方差函数是

$$C(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}) = \theta_1 \exp\left[-\frac{1}{2}\sum_i\left(\frac{\sin\left(\frac{\pi}{\lambda_i}(x_i-x'_i)\right)}{r_i}\right)^2\right] \tag{45.49}$$

图 45.1 显示了从具有不同协方差函数的高斯过程中所抽取的一些随机样本。

非平稳协方差函数

最简单的非平稳协方差函数对应于有线性趋势的函数。考虑平面 $y(\boldsymbol{x})=\sum_i w_i x_i + c$。如果 $\{w_i\}$ 和 c 是均值为零、方差分别为 σ_w^2 和 σ_c^2 的高斯分布，那么平面的协方差函数为

$$C_{\text{lin}}(\boldsymbol{x},\boldsymbol{x}';\{\sigma_w,\sigma_c\}) = \sum_{i=1}^{I}\sigma_w^2 x_i x'_i + \sigma_c^2 \tag{45.50}$$

包括线性项的随机样本函数的一个例子如图 45.1(d)所示。

45.5 高斯过程模型的适应性

假设协方差函数的形式已被选定，但是取决于未确定的超参数 $\boldsymbol{\theta}$。希望从数据

中“学习”这些超参数。这个学习过程等价于神经网络的超参数推理，例如，权衰减超参数。它是一个复杂度控制问题，可用贝叶斯奥卡姆剃刀很好地解决。

在理想情况下，希望定义超参数的先验分布，并对它们进行积分以获得预测，即想求出

$$P(t_{N+1}|\boldsymbol{x}_{N+1},\mathscr{D}) = \int P(t_{N+1}|\boldsymbol{x}_{N+1},\boldsymbol{\theta},\mathscr{D})P(\boldsymbol{\theta}|\mathscr{D})\,\mathrm{d}\boldsymbol{\theta} \tag{45.51}$$

但是这个积分通常是不易求解的。可以采取以下两种方法。

1. 用最可能的超参数值来近似积分。

$$P(t_{N+1}|\boldsymbol{x}_{N+1},\mathscr{D}) \simeq P(t_{N+1}|\boldsymbol{x}_{N+1},\mathscr{D},\boldsymbol{\theta}_{\mathrm{MP}}) \tag{45.52}$$

2. 或者用蒙特卡罗方法在 $\boldsymbol{\theta}$ 上进行数值积分（Williams 和 Rasmussen，1996 年；Neal，1997b）。

如果可以计算出 $\boldsymbol{\theta}$ 的后验概率的梯度，这两种方法都可以十分有效地实现。

梯度

$\boldsymbol{\theta}$ 的后验概率是

$$P(\boldsymbol{\theta}|\mathscr{D}) \propto P(\boldsymbol{t}_N|\boldsymbol{X}_N,\boldsymbol{\theta})P(\boldsymbol{\theta}) \tag{45.53}$$

第一项的对数（超参数的显著性）是

$$\ln P(\boldsymbol{t}_N|\boldsymbol{X}_N,\boldsymbol{\theta}) = -\frac{1}{2}\ln\det \boldsymbol{C}_N - \frac{1}{2}\boldsymbol{t}_N^{\mathrm{T}}\boldsymbol{C}_N^{-1}\boldsymbol{t}_N - \frac{N}{2}\ln 2\pi \tag{45.54}$$

它关于超参数 θ 的导数是

$$\frac{\partial}{\partial\theta}\ln P(\boldsymbol{t}_N|\boldsymbol{X}_N,\boldsymbol{\theta}) = -\frac{1}{2}\mathrm{Trace}\left(\boldsymbol{C}_N^{-1}\frac{\partial\boldsymbol{C}_N}{\partial\theta}\right) + \frac{1}{2}\boldsymbol{t}_N^{\mathrm{T}}\boldsymbol{C}_N^{-1}\frac{\partial\boldsymbol{C}_N}{\partial\theta}\boldsymbol{C}_N^{-1}\boldsymbol{t}_N \tag{45.55}$$

评论

假设先验的求导很直接，现在可以搜寻 $\boldsymbol{\theta}_{\mathrm{MP}}$。但是要认识到两个问题。首先，如图 45.2 所示，显著性可能是多模的。适当的先验和合理的优化策略往往能够去除不好的最优点。其次，也许最重要的是，计算对数似然的梯度需要计算 $\boldsymbol{C}_N^{-1}$。任何精确求逆方法（如 Cholesky 分解、LU 分解或 Gauss－Jordan 消去法）都有 N^3 量级的计算代价，所以对于大的训练数据集，计算梯度会非常耗时。实现预测（式（45.42）和（45.43））和梯度计算（式（45.55））的近似方法是很活跃的研究领域。基于 Skilling（1993 年）思想的一种方法是用迭代方法近似 $\boldsymbol{C}^{-1}\boldsymbol{t}$ 和 $\mathrm{Trace}\boldsymbol{C}^{-1}$，计算复杂度是 $O(N^2)$（Gibbs 和 MacKay，1996 年；Gibbs，1997 年）。关于这个主题的进一步参考文献列在本章的末尾。

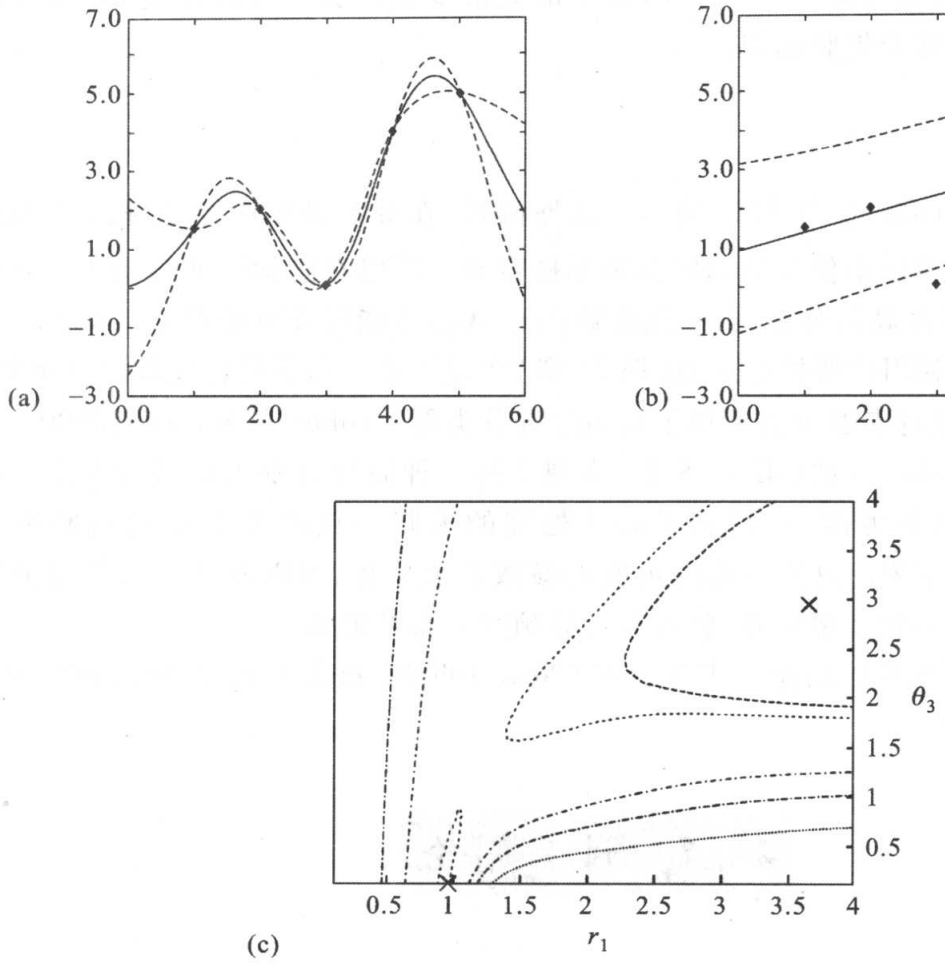


图 45.2　高斯过程的多模似然函数。用简单协方差函数(45.47)对 5 个数据点的集合建模，一个超参数 θ_3 控制噪声的方差。子图(a)和(b)所示为超参数 $\boldsymbol{\theta}$ 设为如下两个不同的值时，最可能的插值函数和它的 1σ 误差限：(a) $r_1 = 0.95, \theta_3 = 0.0$；(b) $r_1 = 3.5, \theta_3 = 3.0$；(局部地)最大化似然函数 $P(\boldsymbol{t}_N | \boldsymbol{X}_N, \boldsymbol{\theta})$。图(c)显示了似然等势线，它是 r_1 和 θ_3 的一个函数，用 × 表示两个最大值。参见 Gibbs(1997 年)。

45.6 分　类

一旦确定了一个变量可以有一个合适的高斯过程先验，就可将高斯过程归入分类建模中。

在二进制分类问题中，可以定义量 $a_n \equiv a(\boldsymbol{x}^{(n)})$ 使得是 1 类而非 0 类的概率为

$$P(t_n = 1 | a_n) = \frac{1}{1 + \mathrm{e}^{-a_n}} \tag{45.56}$$

a 为大的正值时，相应的概率接近于 1；a 为大的负值时，定义的概率接近于 0。在一

个分类问题中，一般想让概率 $P(t_n=1)$ 是 $\boldsymbol{x}$ 的光滑变化函数。可以定义 $a(\boldsymbol{x})$ 为高斯过程先验以实现这个先验期望。

实现

因为似然函数(45.56)不是 a_n 的一个高斯函数，在分类模型中完成推理及用高斯过程模型来适应数据不像在回归问题中那样容易。所以在已知一些观测 $\boldsymbol{t}$ 的条件下，$\boldsymbol{a}$ 的后验分布不是高斯分布，归一化常数 $P(\boldsymbol{t}_N|\boldsymbol{X}_N)$ 不能写成解析形式。Barber 和 Williams(1997 年)利用拉普拉斯近似(第 27 章)实现了基于高斯过程先验的分类器。Neal(1997b)利用蒙特卡罗方法实现了高斯过程分类器。Gibbs 和 MacKay(2000 年)基于 Jaakkola 和 Jordan 方法(第 33.8 节)实现了另一种简便且令人高兴的方法。在这个变形高斯过程分类器中，得到了易于处理的未归一化的关于 $\boldsymbol{a}$ 的后验密度 $P(\boldsymbol{t}_N|\boldsymbol{a})P(\boldsymbol{a})$ 的上下界。这些界都由可变的参数来参数化，可以调整它以获得最贴近的拟合。利用归一化的最优界，就可以计算预测分布的近似。

多类分类问题也可以用蒙特卡罗方法(Neal，1997b)和变参法(Gibbs，1997 年)来求解。

45.7 讨 论

高斯过程的实现和使用比较简单。因为模型中只有很少的参数需要手工确定(一般来说，只有超参数的先验)，在不可能对每个问题做精调的情况下，高斯过程是自动完成任务的有力工具。我们好像没有因为这种简单性而损失任何性能。

很容易构造具有特定理想特性的高斯过程；例如，可以构造一个简明的自动确定相干性模型。

使用高斯过程的一个明显问题是对 $N\times N$ 矩阵求逆的计算代价。当数据点数 N 超过1 000时，求逆的直接法代价就过于高昂了。

是否把婴儿和洗澡水一同倒掉?

根据 1987 年的流行说法，神经网络要成为可以发现数据特征和模式的智能机器。与之相对，高斯过程则是简单平滑设备。高斯过程怎样替代神经网络呢？是神经网络被过度夸大了还是我们低估了平滑方法的功能？

笔者认为这两种说法都是正确的。高斯过程的成功表明现实世界中的许多数据建模问题都可用合理的平滑方法很好地解决。最有意思的问题，例如特征提取，是不能用高斯过程来解决的。但是，多层感知器也无法解决。要解决机器学习问题，也许需要一个有别于有监督前馈映射方法的全新起点。

进一步的阅读

用于回归的高斯过程研究一点也不新鲜。天文学家 T. N. Thiele 在 1880 年用高斯过程分析时间序列(Lauritzen,1981 年)。在 20 世纪 40 年代,Wiener - Kolmogorov 预测理论被用于军事目标轨迹的预测(Wiener,1948 年)。在地理统计学领域,Matheron(1963 年)提出了利用最优线性估计器作为回归问题的框架,该方法被他命名为"Kriging",以纪念一位南非采矿工程师 D. G. Krige。这个框架等价于高斯过程方法的回归。Kriging 方法在最近 30 年得到了长足的发展(综述参见 Cressie(1993 年)),包括几种贝叶斯处理(Omre,1987 年;Kitanidis,1986 年)。然而,高斯过程模型的地理统计学方法主要集中在低维问题上,基本上忽略了模型的任何概率解释。Kalman 滤波器广泛应用于实现一维平稳高斯过程的推理,是流行的语音和音乐模型(Bar - Shalom 和 Fortmann,1988 年)。广义径向基函数(Poggio 和 Girosi,1989 年)、ARMA 模型(Wahba,1990)和可变度量核方法(Lowe,1995 年)都和高斯过程密切相关。也可参见 O'Hagan(1978 年)。

用高斯过程来代替有监督神经网络的思想首先由 Williams 和 Rasmussen(1996 年)以及 Neal(1997b)探讨。Rasmussen(1996 年)对高斯过程和其他的神经网络方法以及 MARS 进行了全面比较。用高斯过程来降低数据建模的复杂度这一方法仍然是活跃的研究领域(Poggio 和 Girosi,1990 年;Luo 和 Wahba,1997 年;Tresp,2000 年;Williams 和 Seeger,2001 年;Smola 和 Bartlett,2001 年;Rasmussen,2002 年;Seeger 等,2003 年;Opper 和 Winther,2000 年)。

更长一些的高斯过程综述见(MacKay,1998b)。用分层贝叶斯模型控制回归问题复杂度的综述文章见(MacKay,1992a)。

高斯过程和支持向量学习机(Scholkopf 等,1995 年;Vapnik,1995 年)有很多相同之处。二者都是基于内核的预测器,核是协方差函数的另一个名称。探索两者联系的支持向量的一种贝叶斯方法可在(Chu 等,2001 年;Chu 等,2002 年;Chu 等,2003b;Chu 等,2003a)中找到。

第 46 章

反 卷 积

46.1 传统图像重构方法

最优线性滤波器

在很多成像问题中，测量数据$\{d_n\}$和对应的图像$\boldsymbol{f}$有线性关系

$$d_n = \sum_k R_{nk} f_k + n_n \tag{46.1}$$

向量$\boldsymbol{n}$表示污染实际数据的不可避免的噪声。对于拍了一张模糊照片的照相机而言，向量$\boldsymbol{f}$表示实际图像，$\boldsymbol{d}$表示模糊的和有噪声的图像，线性算子$\boldsymbol{R}$是由照相机的点扩散函数所定义的卷积。在这一特例中，实际图像和数据向量处在同一空间中；但是对它们加以区别是十分重要的。用下标$n = 1, \cdots, N$表示测量数据，下标$k, k' = 1, \cdots, K$表示图像的像素。

人们也许会猜想：既然是线性运算造成了图像模糊，也许可以用另外一个线性运算去除模糊。通过两种方法可以得到最优线性滤波器。

贝叶斯方法

假设已知线性算子$\boldsymbol{R}$，噪声$\boldsymbol{n}$服从高斯独立分布，已知标准方差σ_ν。

$$P(\boldsymbol{d} \mid \boldsymbol{f}, \sigma_\nu, \mathscr{H}) = \frac{1}{(2\pi\sigma_\nu^2)^{N/2}} \exp\left(- \sum_n \left(d_n - \sum_k R_{nk} f_k \right)^2 \Big/ (2\sigma_\nu^2) \right) \tag{46.2}$$

假设图像的先验概率也是高斯的，具有缩放参数σ_f。

$$P(\boldsymbol{f} \mid \sigma_f, \mathscr{H}) = \frac{\det{}^{-\frac{1}{2}} \boldsymbol{C}}{(2\pi\sigma_f^2)^{K/2}} \exp\left(- \sum_{k,k'} f_k C_{kk'} f'_k / (2\sigma_f^2) \right) \tag{46.3}$$

如果假设像素之间不存在相关性，则对称的满秩矩阵$\boldsymbol{C}$等于单位矩阵$\boldsymbol{I}$。更为复杂的“固有相关函数”模型是$\boldsymbol{C} = [\boldsymbol{G}\boldsymbol{G}^{\mathrm{T}}]^{-1}$，其中$\boldsymbol{G}$是一个卷积，它将我们从一个想象的无关“隐藏”图像变为实际相关图像。不要把固有相关函数和点扩散函数$\boldsymbol{R}$相混淆，后者定义了图像到数据的映射。

如果已知图像$\boldsymbol{f}$的所有元素都是正的，那么零均值的高斯先验假设显然是不恰当的，但还是让我们继续。现在可以写出已知数据$\boldsymbol{d}$时图像$\boldsymbol{f}$的后验概率。

$$P(\boldsymbol{f}|\boldsymbol{d},\sigma_\nu,\sigma_f,\mathscr{H})=\frac{P(\boldsymbol{d}|\boldsymbol{f},\sigma_\nu,\mathscr{H})P(\boldsymbol{f}|\sigma_f,\mathscr{H})}{P(\boldsymbol{d}|\sigma_\nu,\sigma_f,\mathscr{H})} \tag{46.4}$$

用文字表示就是

$$\text{后验}=\frac{\text{似然}\times\text{先验}}{\text{显著性}} \tag{46.5}$$

“显著性”$P(\boldsymbol{d}|\sigma_\nu,\sigma_f,\mathscr{H})$是后验分布的归一化常数。它在这里并不重要，但是可用于更复杂的分析中，比如比较σ_ν和σ_f的不同值，或不同点的扩散函数$\boldsymbol{R}$。

因为后验分布是$\boldsymbol{f}$的两个高斯函数之积，也是一个高斯分布，所以可以用它的均值（也是最可能的图像$\boldsymbol{f}_{\mathrm{MP}}$）和如下协方差矩阵来表征。

$$\boldsymbol{\Sigma}_{f|d}\equiv[-\nabla\nabla\log P(\boldsymbol{f}|\boldsymbol{d},\sigma_\nu,\sigma_f,\mathscr{H})]^{-1} \tag{46.6}$$

它定义了$\boldsymbol{f}$的联合误差限。在这个公式中，符号∇表示关于图像参数$\boldsymbol{f}$的微分。可以通过对后验对数进行微分、令导数为0求得$\boldsymbol{f}_{\mathrm{MP}}$。得到

$$\boldsymbol{f}_{\mathrm{MP}}=\left[\boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}+\frac{\sigma_\nu^2}{\sigma_f^2}\boldsymbol{C}\right]^{-1}\boldsymbol{R}^{\mathrm{T}}\boldsymbol{d} \tag{46.7}$$

算子$\left[\boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}+\frac{\sigma_\nu^2}{\sigma_f^2}\boldsymbol{C}\right]^{-1}\boldsymbol{R}^{\mathrm{T}}$称为最优线性滤波器。当$\frac{\sigma_\nu^2}{\sigma_f^2}\boldsymbol{C}$项可以忽略时，最优线性滤波器是伪逆$[\boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}]^{-1}\boldsymbol{R}^{\mathrm{T}}$。$\frac{\sigma_\nu^2}{\sigma_f^2}\boldsymbol{C}$项正则化这个病态的逆。

最优线性滤波器也能变成形式

$$\text{最优线性滤波器}=\boldsymbol{C}^{-1}\boldsymbol{R}^{\mathrm{T}}\left[\boldsymbol{R}\boldsymbol{C}^{-1}\boldsymbol{R}^{\mathrm{T}}+\frac{\sigma_\nu^2}{\sigma_f^2}\boldsymbol{I}\right]^{-1} \tag{46.8}$$

最小均方误差的推导

最优线性滤波器的非贝叶斯推导首先假设用数据的线性函数来“估计”实际图像$\boldsymbol{f}$

$$\hat{\boldsymbol{f}}=\boldsymbol{W}\boldsymbol{d} \tag{46.9}$$

然后通过最小化$\hat{\boldsymbol{f}}$和未知实际图像之间的期望误差平方和，来“优化”线性算子$\boldsymbol{W}$。在下面的公式中，重复下标k,k',n上的和是隐含的。$\langle\cdot\rangle$是在随机变量$\{n_n\}$的统计以及我们预计会遇到的图像全集$\boldsymbol{f}$上取期望。假设噪声是零均值，并且与自身和其他所有噪声二阶不相关，且$\langle n_n n_{n'}\rangle=\sigma_\nu^2\delta_{nn'}$。

$$\langle E\rangle=\frac{1}{2}\langle(W_{kn}d_n-f_k)^2\rangle \tag{46.10}$$

$$=\frac{1}{2}\langle(W_{kn}R_{nj}f_j-f_k)^2\rangle+\frac{1}{2}W_{kn}W_{kn}\sigma_\nu^2 \tag{46.11}$$

关于$\boldsymbol{W}$求导，并引入$\boldsymbol{F}\equiv\langle f_j f_j\rangle$（比较上面贝叶斯推导中的$\sigma_f^2\boldsymbol{C}^{-1}$），求得最优线性滤波器为

$$\boldsymbol{W}_{\text{opt}} = \boldsymbol{F}\boldsymbol{R}^{\mathrm{T}}[\boldsymbol{R}\boldsymbol{F}\boldsymbol{R}^{\mathrm{T}} + \sigma_{\nu}^{2}\boldsymbol{I}]^{-1} \tag{46.12}$$

如果令 $\boldsymbol{F} = \sigma_f^2 \boldsymbol{C}^{-1}$，得到贝叶斯推导的最优线性滤波器(46.8)。在这个推导中所用的特别假设是选择二次误差度量，及决定采用线性估计器。有趣的是，没有明确假设高斯分布，这个推导得出了和贝叶斯后验模式相同的估计器，$\boldsymbol{f}_{\text{MP}}$。

贝叶斯方法的优点是，可以质疑这些假设并对它们做出修正，以便能更好地进行重构。

其他图像模型

图像模型 $P(\boldsymbol{f}|\mathscr{H})$ 和客观世界匹配得越好，图像重构就越好，回答给定问题所需的数据就越少。导致最优线性滤波器的高斯模型与客观世界匹配得很差。例如，高斯先验(46.3)未反映出图像中所有像素强度为正的事实。这样的忽略导致最明显的人为痕迹：被观测图像有高的对比度或大的黑块。将最优线性滤波器应用于天文数据上时，在重构中产生了负的区域，相应地对应于吸收了望远镜能量的天空区域！用于图像反卷积的最大熵模型取得了巨大成功(Gull 和 Daniell，1978 年)，主要是因为模型迫使重构的图像为正。消除虚假的负区域和补充的虚假正区域，重构质量大大提高。

“经典的最大熵”模型分配了一个熵的先验

$$P(\boldsymbol{f}|\alpha, \boldsymbol{m}, \mathscr{H}_{\text{经典}}) = \exp(\alpha S(\boldsymbol{f}, \boldsymbol{m}))/Z \tag{46.13}$$

其中

$$S(\boldsymbol{f}, \boldsymbol{m}) = \sum_i (f_i \ln(m_i/f_i) + f_i - m_i) \tag{46.14}$$

(Skilling，1989 年)。这个模型加强正性；参数 α 定义了像素值预期偏离默认图像 $\boldsymbol{m}$ 的一个特征动态范围。

“固有相关函数最大熵”模型(Gull，1989 年)通过 $\boldsymbol{f} = \boldsymbol{G}\boldsymbol{h}$，把空间相关的期望引入 $\boldsymbol{f}$ 的先验，其中 $\boldsymbol{G}$ 是固有相关函数的卷积，把经典的最大熵先验和相应隐式图像 $\boldsymbol{h}$ 联系在一起。

概率电影

不仅求出了最可能的图像 $\boldsymbol{f}_{\text{MP}}$，而且还得到了误差限 $\boldsymbol{\Sigma}_{f|d}$，接下来的一项任务是将这些误差限可视化。无论是否用蒙特卡罗方法推导 $\boldsymbol{f}$，围绕后验分布的相关随机游动可用于对不确定性和相关性进行可视化。对于高斯后验分布，可以使用下式产生相关的单位正态随机向量序列 $\boldsymbol{n}$

$$\boldsymbol{n}^{(t+1)} = c\boldsymbol{n}^{(t)} + s\boldsymbol{z} \tag{46.15}$$

其中 $\boldsymbol{z}$ 是单位正态随机向量，且 $c^2 + s^2 = 1$(c 控制序列的记忆有多持续)。于是可得图像序列定义为

$$\boldsymbol{f}^{(t)} = \boldsymbol{f}_{\text{MP}} + \boldsymbol{\Sigma}_{f|d}^{1/2}\boldsymbol{n}^{(t)} \tag{46.16}$$

其中 $\boldsymbol{\Sigma}_{f|d}^{1/2}$ 是 $\boldsymbol{\Sigma}_{f|d}$ 的 Cholesky 分解。

46.2 用于图像反卷积的有监督神经网络

神经网络研究者往往采用下面的策略。已知一个当前用标准算法求解的问题：把算法所完成的计算解释成从一个输入到一个输出的参数化映射，并称这个映射为一个神经网络；然后使参数适应数据，以便映射能更好地完成任务。神经网络可以通过构造来重现标准算法，所以这个数据驱动的自适应过程只会使性能更好。

采用这种方法来改善标准算法有几个理由。

1. 算法往往不被设计成优化实际的目标函数。例如在语音识别中，设计一个隐式马尔科夫模型对语音信号建模，对它进行拟合，以便在已知训练数据字符串的条件下使生成概率最大化；但是真正的目标却是辨别不同的单词。如果使用一个不充分的模型，模型的神经网络式训练将使模型的有限资源集中在与辨别任务有关的方面。应用于语音识别的隐式马尔科夫模型的辨别训练确实改进了性能。

2. 与标准模型相比，神经网络具有更高的灵活性；一些自适应参数可能被最初的设计者看成固定特征。灵活的网络可以发现原先模型中未包含的数据特性。

46.3 人的反卷积

人脑的很大一部分都用于视觉。人类视觉系统中一个被忽视的特征是视网膜上的原始图像非常模糊：尽管在日光条件下，不论明暗，大多数人都能以 1 弧分的分辨率(1/60 弧度)看到东西；视网膜上的图像会通过一个宽度高达 5 弧分的点扩散函数而变模糊(Wald 和 Griffin，1947 年；Howarth 和 Bradley，1986 年)。非常奇妙的是，人可分辨的像素在面积上比点源在视网膜上所产生的斑点小 25 倍。

Isaac Newton 意识到了这个谜题。很难制造一个不使色彩发生失常的透镜；而人的角膜同普通玻璃制造的透镜一样，折射蓝光的能力比红光强。通常，眼睛在可见光谱的中部(绿色处)能正确地聚焦。所以，如果看由红、绿和蓝光所组成的单一白点，视网膜上的图像就是强烈聚焦的绿点，环绕着宽一些的红斑点，再叠加在更宽的蓝斑点上。红色和蓝色斑点的宽度正比于瞳孔的直径，它在光线暗淡的条件下最大。[斑点大致同心，尽管大多数人稍微有点偏置，以至于在一个眼睛里红斑点的中心偏左一点，蓝斑点则偏右一点，而在另一只眼睛里则相反。这个轻微的偏置解释了，当观看涂在一个黑色背景下的蓝色和红色时，为什么大多数人感觉到蓝色写在比红色稍微远一些的地方。对于少数人而言，偏置的方向相反，所以红/蓝深度的感知也是相反的。但是这一效应(许多人都知道，例如在电影院里已经注意到)和本文正在讨论的色彩失常相比是很小的。]

借助于一张卡片和一个计算机彩色屏幕，就可以亲自生动地演示眼睛里的色彩失常现象有多大了。

在一间除计算机屏幕之外没有其他光源的暗室里，笔者会给你一个印象最深的结果——保证会让你吃惊；即使房间透进一些日光，只要不是强烈的阳光照射，仍然可以看见相当强的效果。在一张卡片上切割一条 1.5 mm 宽的裂缝。在黑色背景的计算机屏幕上显示一些小的彩色目标。笔者特别推荐用纯红、纯蓝、洋红（即红加蓝）和白色（红加蓝加绿）的细竖条。屏幕上还有一些黑白文本。远离屏幕站或坐，直至只能看清文本（如果你的视力正常，也许是 4 m 左右）。现在，在你的一只眼睛前方竖直地举着卡片裂缝，闭上另一只眼睛。让裂缝接近你的眼睛（碰到你的眼睫毛），并从缝隙中看出去。慢慢地左右摇动裂缝，让它交替地在瞳孔前方的左边和右边。你看见什么？笔者看见红色目标来回地摆动，蓝色目标来回地摇动，距离很大，方向相反，而白色目标保持静止且几乎没有变形。可以看到细的洋红色目标被分解成红的和蓝的部分。测量在这个暗室中红目标和蓝目标的运动幅度有多大：对处于昏暗房间中的我来说，超过了 5 弧分。然后检验在这个条件下你能看得多清楚——比如，可以看看屏幕上的文本：你通过整个瞳孔所能看见的特征不是远小于红色和蓝色部分摇摆的距离吗？当你用整个瞳孔时，落在视网膜上的必定是模糊图像，模糊的直径等于摆动幅度。

早期视觉处理的主要功能之一肯定是消除色彩失常。神经科学家有时推测视网膜中心细胞和侧面膝状核（视网膜中心细胞映射至大脑的主要区域）中的细胞之所以有对色彩互斥的环绕中心的接收区域（例如，长波长位于中心，中波长位于周边），就是为了实现“特征提取”和“边缘检测”，但是我认为这一观点是错误的。我们在视觉处理的第一阶段有环绕中心的滤波器（至少在视网膜的中央凹），原因是为了完成消除色彩失常的繁重任务。

笔者推测 McCollough 效应，一种极持久的色彩与方位关联（McCollough，1965 年；MacKay 和 MacKay，1974 年）是通过自适应机制产生的，它对人的色彩失常消除（反卷积）“电路”进行调整。由于所研究的点扩散函数随瞳孔直径改变，改变时间在数秒之内，所以反卷积电路必须能迅速调整；McCollough 效应确实可以在 30 秒内发生。同时在一只眼睛被遮挡的情况下，效应的持续时间很长，因为反卷积电路在人体睡眠时保持调整好的状态，以便睡醒的瞬间就能清晰地看见东西，这使人真正感兴趣。

笔者也想知道色彩视觉进化的主要原因是否不因“为了更好地识别水果”，而是“为了可以更清楚地分辨黑白”——如果能够得到图像中的色彩信息，即使在完全黑和白的世界中，消除色彩失常也很容易。

最后的猜想：当看东西的时候，为什么眼睛能进行微 - 快速扫视？这些微小的眼睛运动角度大小大于视网膜中央凹的锥体间距（约为 1 弧分，这是眼睛的分辨率）。典型的微 - 快速扫视大小是 5 ~ 10 弧分（Ratliff 和 Riggs，1950 年）。它和色彩失常有同样的大小，这是巧合吗？微 - 快速扫视在具有高分辨率视觉的反卷积机制中必定

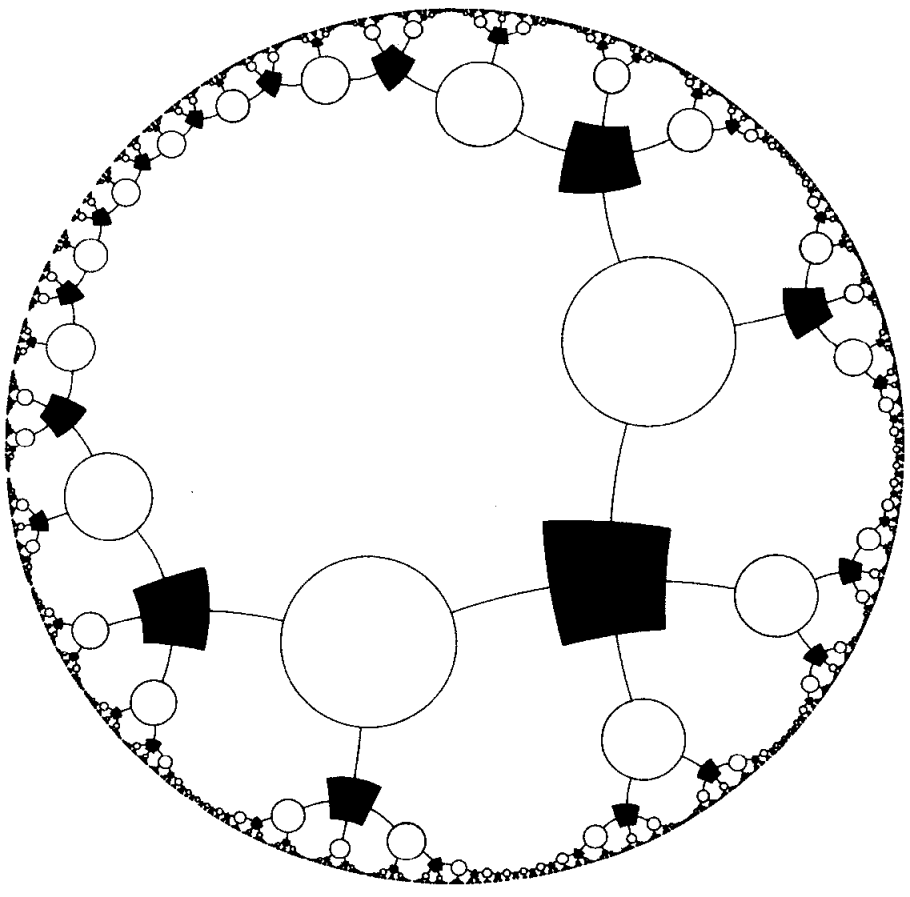
扮演着重要角色。

46.4 习 题

习题 46.1[♪] 用一个循环(大礼帽)点扩散函数加上噪声来使图像变模糊。然后用最优线性滤波器对模糊的有噪图像做反卷积。求误差限,并用概率电影使其可视化。

第六部分

稀疏图码



关于第六部分

通信原理的核心问题是构建一个编码系统和一个译码系统，使得在噪声信道上进行可靠通信成为可能。在20世纪90年代，在用稀疏随机图所定义的码逼近香农(Shannon)极限方面获得了显著的进展，并且它们使用一个简单的基于概率的消息传递算法来译码。

在一个*稀疏图码*中，图的结点代表传输比特及其所满足的约束条件。对于码率为 $R = K/N$、码字长度为 N 的一个线性码，约束的个数为 $M = N - K$。任意线性码都可用一个图来描述，但是稀疏图码的特殊之处在于每一约束条件仅涉及图中少量的变量：所以图的边数与 N 大致成线性关系，而非二次方关系。

在随后的4章中，将研究4类稀疏图码：其中在纠错方面性能优越的有3类，即*低密度奇偶校验码*、*turbo码*和*重复累计码*；一类*数字喷泉码*，它在纠正删除错误时性能突出。

所有这些码都可通过图中的一个局部消息传递算法“和-积算法”来译码，尽管这种算法并非完美的最大似然译码器，但是实践结果打破了记录。

第 47 章

低密度奇偶校验码

一个低密度奇偶校验码(或 Gallager 码)是一个具有奇偶校验矩阵 $\boldsymbol{H}$ 的分组码,矩阵的每行和每列都是“稀疏的”。

一个规则的 Gallager 码是一个低密度奇偶校验码,$\boldsymbol{H}$ 的每列具有相同权重 j,而每行具有相同权重 k;规则 Gallager 码是在满足这些约束的前提下随机构造出来的。$j=3$ 和 $k=4$ 的一个低密度奇偶校验码如图 47.1 所示。

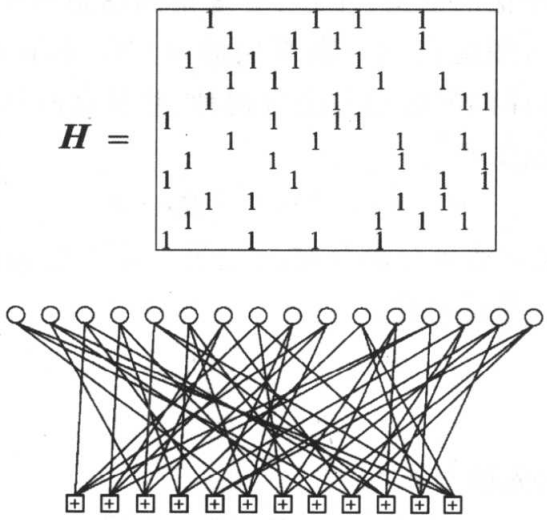


图 47.1 一个码率为 $\frac{1}{4}$、长度为 $N=16$、约束条件数 $M=12$ 的低密度奇偶校验码所对应的低密度奇偶校验矩阵与图。每个白圈代表一个发送比特。每个比特受到 $j=3$ 个约束,用⊞方块表示。每个约束条件要求与它相连的 $k=4$ 个比特之和为偶数。

47.1 理论特性

低密度奇偶校验码适合于理论研究。下面的结果由 Gallager(1963 年)和 MacKay(1999b)证明。

低密度奇偶校验码,尽管可以简单地构造出来,但在已知最优译码器时却是一个好码(在第 11.4 节所述意义下的好码)。并且,它们具有好的距离(在第 13.2 节所

述意义下)。这两个结果在任意列权重 $j \geqslant 3$ 时都成立。此外,存在一种低密度奇偶校验码序列,它的 j 随着 N 逐渐增加,与此同时比值 j/N 依旧趋于零,这种码是非常好的码,并且有非常好的距离。

但是,没有一个最优译码器,并且对低密度奇偶校验码进行译码是一个 NP - 完全问题。那么在实际中应该怎么办呢?

47.2 实用的译码

假定一个信道的输出为 $\boldsymbol{r}$,希望寻找码字 $\boldsymbol{t}$,使它的似然 $P(\boldsymbol{r}|\boldsymbol{t})$ 最大。所有低密度奇偶校验码的有效译码策略都是消息传递算法。已知的最佳算法是和 - 积算法,也被称为迭代概率译码或置信度传播。

假设信道是一个无记忆的信道(不过,更复杂的信道也不难处理,只要对表示错误之间期望相关性的复杂图运行和 - 积算法即可(Worthen 和 Stark,1998 年))。对于任意无记忆信道,解决译码问题的方法有两种,并且这两种方法都导致一个一般性问题"寻找使下式最大化的 $\boldsymbol{x}$"。

$$P^{*}(\boldsymbol{x}) = P(\boldsymbol{x})\mathbb{1}[\boldsymbol{H}\boldsymbol{x} = \boldsymbol{z}]' \tag{47.1}$$

其中 $P(\boldsymbol{x})$ 是二进制向量 $\boldsymbol{x}$ 的可分离分布,$\boldsymbol{z}$ 是另一个二进制向量。这两种方法都用因子图(第 26 章)来表示译码问题。

码字译码的观点

首先,注意到码字的先验分布

$$P(\boldsymbol{t}) \propto \mathbb{1}[\boldsymbol{H}\boldsymbol{t} = \boldsymbol{0} \bmod 2] \tag{47.2}$$

可以用一个因子图来表征(图 47.2(a)),其因式分解为

$$P(\boldsymbol{t}) \propto \prod_{m} \mathbb{1}\left[\sum_{n \in \mathcal{M}(m)} t_n = 0 \bmod 2\right] \tag{47.3}$$

(今后将忽略"mod 2")。码字的后验分布由这个先验乘以似然给出,后者引入另外 N 个因子,每个接收比特对应于 1 个。

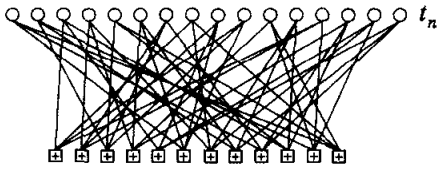
$$\begin{aligned} P(\boldsymbol{t}|\boldsymbol{r}) &\propto P(\boldsymbol{t})P(\boldsymbol{r}|\boldsymbol{t}) \\ &\propto \prod_{m} \mathbb{1}\left[\sum_{n \in \mathcal{M}(m)} t_n = 0\right] \prod_{n} P(r_n|t_n) \end{aligned} \tag{47.4}$$

与这个函数相对应的因子图如图 47.2(b)所示。除了在发送比特上增加了似然"挂件"之外,这个图与前一图完全一样。

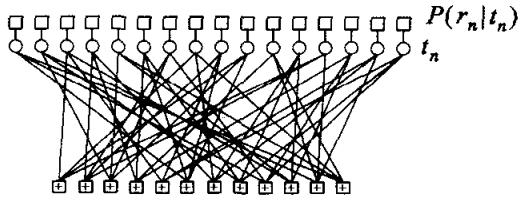
从这个角度来看,接收信号 $\boldsymbol{r}_n$ 可以是任意符号;相关的只是 $P(\boldsymbol{r}_n|\boldsymbol{t}_n)$ 的值。

校正子译码观点

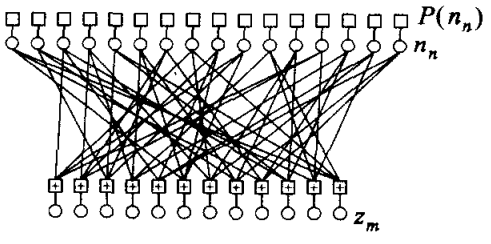
或者也可以根据接收二进制向量 $\boldsymbol{r}$ 和噪声向量 $\boldsymbol{n}$ 来观察信道输出,这里 $\boldsymbol{n}$ 的



(a) 码字的先验分布为 $P(\boldsymbol{t}) \propto 1[\boldsymbol{Ht}=0]$。变量结点是发送比特 $\{t_n\}$。每个⊞结点代表因子 $1\left[\sum_{n\in\mathcal{M}(m)} t_n = 0 \bmod 2\right]$。



(b) 码字的后验分布 $P(\boldsymbol{t}|\boldsymbol{r}) \propto P(\boldsymbol{t})P(\boldsymbol{r}|\boldsymbol{t})$。上面的每个函数结点代表一个似然因子 $P(r_n|t_n)$。



(c) 噪声 $\boldsymbol{n}$ 和校正子 z 的联合概率，$P(\boldsymbol{n},z) = P(\boldsymbol{n})1[z=\boldsymbol{Hn}]$。现在顶部的变量结点是噪声比特 $\{n_n\}$。底部被加变量结点是校正子的值 $\{z_m\}$。每个 $z_m = \sum_n H_{mn}n_n \bmod 2$ 都由一个⊞因子执行。

图 47.2 与一个低密度奇偶校验码有关的因子图

概率分布$P(\boldsymbol{n})$可以从信道的性质和从信道输出可获得的任何附加信息中推导出来。

例如，对于一个二进制对称信道，定义噪声为 $\boldsymbol{r}=\boldsymbol{t}+\boldsymbol{n}$，校正子为 $z=\boldsymbol{Hr}$，噪声模型为$P(n_n)=f$。对于其他信道，如输出为 $\boldsymbol{y}$ 的高斯信道，可以按照意愿定义一个接收二进制向量 $\boldsymbol{r}$，并从 $\boldsymbol{y}$ 得到一个有效的二进制噪声模型 $P(\boldsymbol{n})$（习题 9.18 和习题 25.1）。

可以将噪声 $\boldsymbol{n}$ 和校正子 $z=\boldsymbol{Hn}$ 的联合概率因式分解为

$$\begin{aligned} P(\boldsymbol{n},z) &= P(\boldsymbol{n})1[z=\boldsymbol{Hn}] \\ &= \prod_n P(n_n) \prod_m 1\left[z_m = \sum_{n\in\mathcal{M}(m)} n_n\right] \end{aligned} \tag{47.5}$$

这个函数的因子图如图47.2(c)所示。与图47.2(a)类似，变量 $\boldsymbol{n}$ 和 z 也可以绘制在一个“置信度网络”中（也叫“贝叶斯网络”、“因果网络”或“影响图”），但是从上面的

圆结点(代表变量 $\boldsymbol{n}$)边上有箭头到下面的方结点(现在代表变量 $\boldsymbol{z}$)。可以说每个比特 x_n 都是 j 个校验 z_m 的父结点,每个校验 z_m 是 k 个比特的子结点。

两个译码观点涉及基本相同的图。两种版本的译码问题都可表述为一般性的译码问题"寻找使得下式最大化的 $\boldsymbol{x}$"。

$$P^*(\boldsymbol{x}) = P(\boldsymbol{x})\mathbb{1}[\boldsymbol{Hx} = \boldsymbol{z}]' \tag{47.6}$$

在码字的译码观点中,$\boldsymbol{x}$ 是码字 $\boldsymbol{t}$,且 $\boldsymbol{z}=0$;在校正子译码观点中,$\boldsymbol{x}$ 是噪声 $\boldsymbol{n}$,$\boldsymbol{z}$ 是校正子。

当使用和-积算法时,采用哪种观点都可以。两个译码算法是同构的并将给出等价的结果(除非受数值误差影响)。

笔者偏向于使用校正子译码观点,因为它有一个优点:它无需执行编码器就能真实地仿真一个译码问题。

现在从一般性的译码问题角度来进行讨论。

47.3 用和-积算法来译码

已知观测的校验,目标是计算每个 n 的边缘后验概率 $P(x_n = 1 | \boldsymbol{z}, \boldsymbol{H})$。要精确计算它们是很困难的,这是因为图中包括了很多回路。但是,假设引入的错误相对较小,实现适用于没有回路的译码算法,是令人感兴趣的。这种忽略回路的方法在人工智能文献中已有应用,但是现在并不赞成使用该算法,因为它会导致不准确的概率。不过,如果正在对一个好的纠错码进行译码,无需关心准确的边缘概率,只需要正确的码字。而且,在一个好码以可达速率进行通信时,通常认为后验概率会高度集中在最可能的译码上;所以正在应对一个特殊的概率分布,在其他领域所获得的经验可能无法应用于该概率分布之上。

和-积算法已经在第26章介绍过。现在明确地写出它是如何解决译码问题的。

$$\boldsymbol{Hx} = \boldsymbol{z} \pmod 2$$

为了简便起见,重新吸收图47.2(c)中结点 x 和 z 上的"挂件",并对和-积算法做相应的修改。于是 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 所在的图是最初的图(图47.2(a)),它的边由 $\boldsymbol{H}$ 中的1定义。图中包括两类结点,称之为校验结点和比特结点。连接校验结点和比特结点的图是一个二分图:比特结点仅与校验结点相连,反之亦然。在每次迭代中,一个概率比值沿着图中的每条边传递,而每个比特结点 x_n 更新它要处于状态1的概率。

用 $\mathcal{N}(m) \equiv \{n : H_{mn} = 1\}$ 表示参与校验点 m 的比特 n 的集合。同样,定义比特 n 参与的校验结点集合 $\mathcal{M}(n) \equiv \{m : H_{mn} = 1\}$。用 $\mathcal{N}(m) \backslash n$ 表示不包含比特 n 在内的集合 $\mathcal{N}(m)$。算法有两个轮换的部分,其中与图中每条边有关的 q_{mn} 和 r_{mn} 两个量迭代地更新。假定通过校验结点 m 以外的校验结点获得信息,则量 q^x_{mn} 表示 $\boldsymbol{x}$ 的比特 n 的值为 x 的概率。如果 $\boldsymbol{x}$ 的比特 n 固定在 x,而其他比特具有一个由概率 $\{q_{mn'} : n' \in \mathcal{N}(m) \backslash n\}$ 给出的可分离的分布,则量 r^x_{mn} 表示满足校验 m 的概率。如果通过矩阵 $\boldsymbol{H}$

所定义的二分图不包含回路，则在一个固定次数的迭代之后，算法将产生所有比特的准确后验概率。

初始化。设 $p_n^0 = P(x_n = 0)$（比特 x_n 为 0 的先验概率），且设 $p_n^1 = P(x_n = 1) = 1 - p_n^0$。如果采用校正子译码观点，且信道是一个二进制对称信道，则 p_n^1 将等于 f。如果噪声水平以已知方式变化（例如，如果信道是一个有实输出的二进制输入高斯信道），则 p_n^1 被初始化为合适的归一化似然。对 $H_{mn} = 1$ 的每个 (n, m)，变量 q_{mn}^0 和 q_{mn}^1 将分别被初始化为 p_n^0 和 p_n^1。

水平步骤。在算法的水平步骤（从矩阵 $\boldsymbol{H}$ 的角度来看是水平的），遍历校验结点 m，并对每个 $n \in \mathcal{N}(m)$ 计算两个概率：第一个是 r_{mn}^0，即当 $x_n = 0$ 时，假定其他比特 $\{x_{n'} : n' \neq n\}$ 具有由概率 $\{q_{mn'}^0, q_{mn'}^1\}$ 给出的可分离的分布，观测值 z_m 发生的概率，它定义为

$$r_{mn}^0 = \sum_{\{x_{n'}: n' \in \mathcal{N}(m) \backslash n\}} P(z_m | x_n = 0, \{x_{n'} : n' \in \mathcal{N}(m) \backslash n\}) \prod_{n' \in \mathcal{N}(m) \backslash n} q_{mn'}^{x_{n'}} \tag{47.7}$$

第二是 r_{mn}^1，即当 $x_n = 1$ 时，观测值 z_m 发生的概率，它定义为

$$r_{mn}^1 = \sum_{\{x_{n'}: n' \in \mathcal{N}(m) \backslash n\}} P(z_m | x_n = 1, \{x_{n'} : n' \in \mathcal{N}(m) \backslash n\}) \prod_{n' \in \mathcal{N}(m) \backslash n} q_{mn'}^{x_{n'}} \tag{47.8}$$

这些和式中的条件概率是 0 或 1，这取决于被观测 z_m 是否与 x_n 和 $\{x_{n'}\}$ 的假设值相匹配。

这些概率可以使用基于式（47.7）和式（47.8）的各种显而易见的方法来计算。按如下方法进行计算也许会非常高效（如果 $|\mathcal{N}(m)|$ 很大）：将 $z_m + x_n$ 视为具有状态 0 和 1 的马尔科夫链的最终状态，这个链以状态 0 开始，并根据所加入的不同 $x_{n'}$ 进行转移，转移概率由相应的 $q_{mn'}^0$ 和 $q_{mn'}^1$ 给出。假定 $x_n = 0$ 或 $x_n = 1$ 时，z_m 取得其观测值的概率可以用前向－后向算法（第 25.3 节）有效地求出。

这种方法的一种特别方便的实现使用前向和后向传递，并在其中计算差值 $\delta q_{mn} = q_{mn}^0 - q_{mn}^1$ 的乘积。从下面的恒等式可以得到 $\delta r_{mn} = r_{mn}^0 - r_{mn}^1$。

$$\delta r_{mn} = (-1)^{z_m} \prod_{n' \in \mathcal{N}(m) \backslash n} \delta q_{mn'} \tag{47.9}$$

这个恒等式通过下面的观测迭代得到：如果 $\zeta = x_\mu + x_\nu \bmod 2$，且 x_μ 和 x_ν 为 0 和 1 的概率分别为 q_μ^0, q_ν^0 和 q_μ^1, q_ν^1，则 $P(\zeta = 1) = q_\mu^1 q_\nu^0 + q_\mu^0 q_\nu^1$ 和 $P(\zeta = 0) = q_\mu^0 q_\nu^0 + q_\mu^1 q_\nu^1$。因此 $P(\zeta = 0) - P(\zeta = 1) = (q_\mu^0 - q_\mu^1)(q_\nu^0 - q_\nu^1)$。

利用下式重新得到 r_{mn}^0 和 r_{mn}^1

$$r_{mn}^0 = \frac{1}{2}(1 + \delta r_{mn}), \quad r_{mn}^1 = \frac{1}{2}(1 - \delta r_{mn}) \tag{47.10}$$

变成差 δq 的变换和从 δr 变回 $\{r\}$ 的变换可被看做一个傅里叶变换和一个傅里叶逆变换。

垂直步骤。垂直步骤使用已计算的 r_{mn}^0 和 r_{mn}^1 的值并更新概率 q_{mn}^0 和 q_{mn}^1 的值。对于每个 n，计算

$$q_{mn}^0 = \alpha_{mn} p_n^0 \prod_{m' \in \mathcal{M}(n) \backslash m} r_{m'n}^0 \tag{47.11}$$

$$q_{mn}^{1} = \alpha_{mn} p_{n}^{1} \prod_{m' \in \mathcal{M}(n) \backslash m} r_{m'n}^{1} \tag{47.12}$$

其中 α_{mn} 要选择满足 $q_{mn}^{0} + q_{mn}^{1} = 1$ 的值。在向下传递和向上传递中，都可以有效地计算这些乘积。

也可以在迭代中计算 q_{n}^{0} 和 q_{n}^{1} 的“准后验概率”，由下式给出

$$q_{n}^{0} = \alpha_{n} p_{n}^{0} \prod_{m \in \mathcal{M}(n)} r_{mn}^{0} \tag{47.13}$$

$$q_{n}^{1} = \alpha_{n} p_{n}^{1} \prod_{m \in \mathcal{M}(n)} r_{mn}^{1} \tag{47.14}$$

这些量都用来产生一个尝试性的译码 $\hat{\boldsymbol{x}}$，它的一致性用来决定是否可以中止译码算法。（如果 $\boldsymbol{H}\hat{\boldsymbol{x}} = \boldsymbol{z}$，则中止。）

此时，算法是从水平步骤重复的。

完成即停止的译码方法。 所推荐的译码过程是，如果 $q_{n}^{1} > 0.5$，则设 $\hat{\boldsymbol{x}}_{n}$ 为 1，并看校验 $\boldsymbol{H}\hat{\boldsymbol{x}} = \boldsymbol{z} \bmod 2$ 是否全部满足，当它们满足时停止；如果经过某个较大次数的迭代后（例如 200 或1 000）都没有成功地译码，则宣告失败。如果译码失败，也许还会给出 $\hat{\boldsymbol{x}}$，但是要标记整个分组是一个误码。

注意到在传递中这个译码过程和 Turbo 码界广泛使用的做法之间的差异；在 Turbo 码中，译码算法运行固定次数的迭代（而不考虑译码器是否在某些更早时间就找到了一个一致的状态）。这种实现浪费了计算机机时，并且模糊了未检测到的差错和已检测到的差错之间的差别。在程序中，如果译码器寻找到一个 $\hat{\boldsymbol{x}}$ 满足 $\boldsymbol{H}\hat{\boldsymbol{x}} = \boldsymbol{z} \bmod 2$，但不等于真值 $\boldsymbol{x}$ 时，就发生了“未检测到”的错误。如果算法运行到最大迭代次数后还未找到一个有效的译码，则发生了“已检测到”的错误。因为未检测到的错误揭示了一个码的距离特性，所以它具有科学上的价值。而在一个工程实践中，如果实际可能的话，标记出一个已知包含已检测错误的分组码，看来好像更适合。

代价。 在一个蛮力方法中，用来产生生成矩阵所需时间的数量级为 N^{3}，其中 N 是分组大小。编码时间的数量级为 N^{2}，但是编码仅涉及二进制算术，所以对于这里研究的分组长度，所需时间远远小于仿真高斯信道所需时间。译码的每次迭代包括大约 $6Nj$ 次浮点乘法，所以译码 1 比特所需运算总数（假设 20 次迭代）约为 $120t/R$，与分组长度无关。对于在下一节出现的码，需要大约 800 次运算。

可以通过 Richardson 和 Urbanke（2001b）发明的巧妙编码技巧或通过特殊构造的奇偶校验矩阵（MacKay 等，1999 年）来降低编码的复杂度。

通过传递低精度的消息来代替实数，译码复杂度可以被降低，并只有很小的性能损失（Richardson 和 Urbanke，2001a）。

47.4 Gallager 码的图示

图 47.3 ~ 图 47.7 形象地展示了低密度奇偶校验码在二进制对称信道和高斯

信道上提供可靠通信的条件。可以在万维网上看到这些动画演示(MacKay, 1997b)。

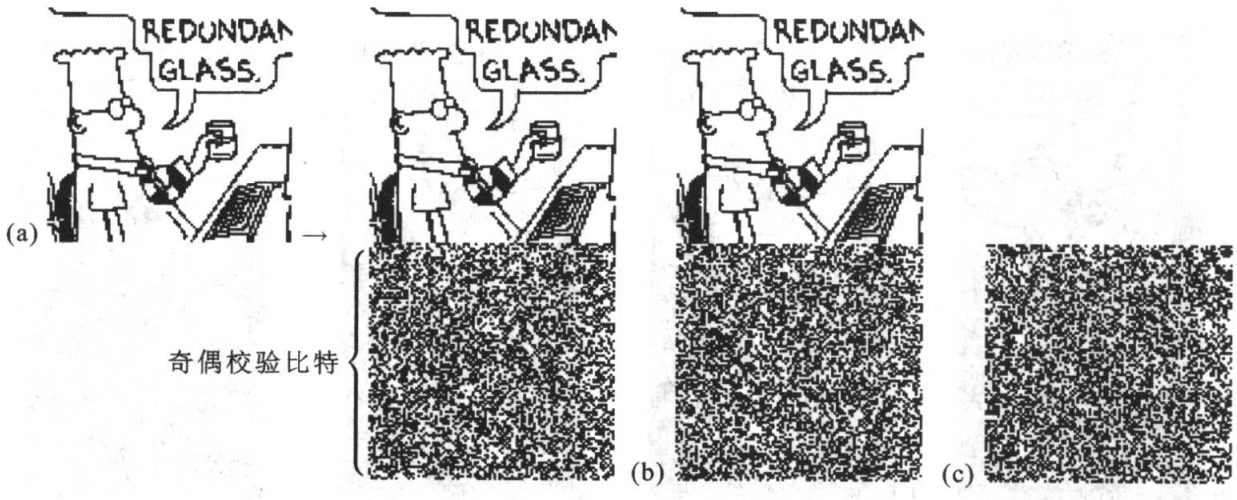


图 47.3 一个码率为 1/2 的 Gallager 码的编码示意图。编码器来自一个很稀疏的、每列有 3 个 1 的 10 000 × 20 000 奇偶校验矩阵。(a) 该码产生由 10 000 个信源比特和 10 000个奇偶校验比特所组成的传输向量。(b) 这里,信源序列通过改变第 1 比特而发生变化。注意到很多奇偶校验比特改变了。每个奇偶比特依赖于大约一半的信源比特。(c) 在 $\boldsymbol{s}=(1,0,0,\cdots,0)$ 情形下的传输。该向量是传输(a)和(b)之差(模 2)。[Dilbert 图像版权©,1997, United Feature Syndicate, Inc. 授权使用。]



图 47.4 一个低密度奇偶校验矩阵,$N=20\ 000$ 列,每列权重 $j=3$,$M=10\ 000$ 行,每行权重 $k=6$。

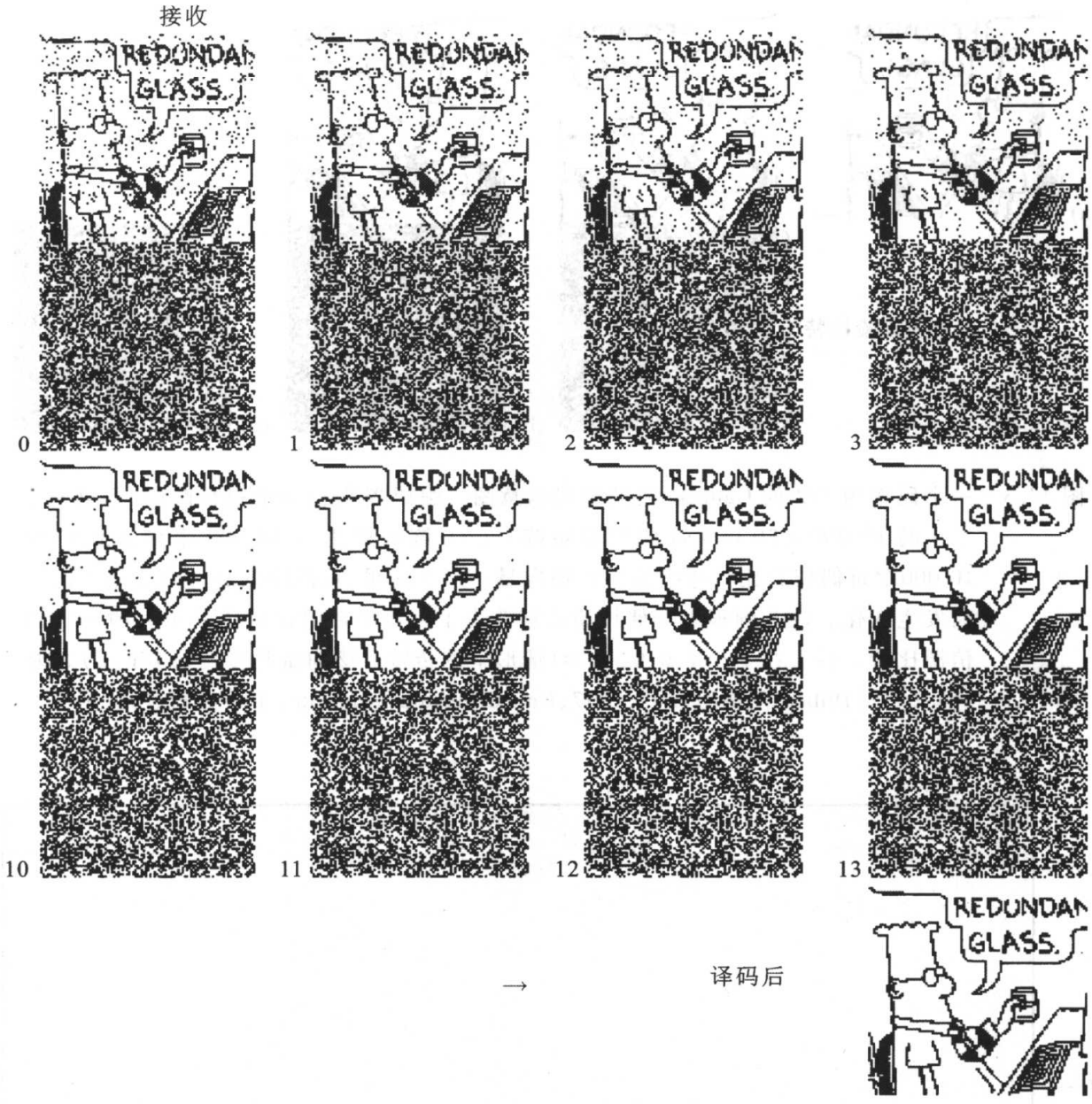


图 47.5 噪声水平 $f=7.5\%$ 信道所接收到的使用迭代概率译码的一个低密度奇偶校验码的一次传输。一系列图片表明最佳估计，在 0、1、2、3、10、11、12、13 次迭代后由迭代译码器逐比特给出。当最佳估计和奇偶校验一致时，译码器在第 13 次迭代之后停止。这个最终的译码是无误的。

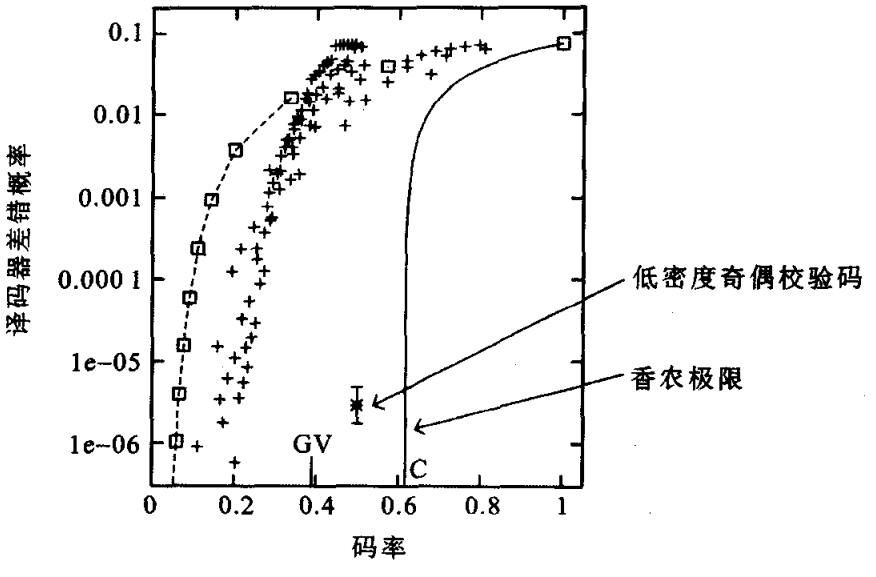


图 47.6 在 $f=7.5\%$ 的二进制对称信道上，使用低密度奇偶校验码（带误差限）的差错概率与代数码的比较。方块：循环码和(7,4)汉明码；其他点：Reed - Muller 码和 BCH 码。

编码

如图 47.3 所示是一个 Gallager 码的编码操作，该码的奇偶校验矩阵是一个每列有 3 个 1 的 10 000 × 20 000 矩阵（图 47.4）。图 47.3(b)和(c)显示了当 10 000 个信源比特之一被改变时，发送向量发生的变化的情况，由此表明生成矩阵的高密度。当然，这里显示的信源图是高度冗余的，并且这种图确实应在编码之前得到压缩。在这些图示中选择冗余图像是为了更容易地看出迭代译码中的纠错过程。译码算法并未利用信源向量的冗余信息，且不管信源向量如何选择，译码算法将以同样方式工作。

迭代译码

在噪声水平 $f=7.5\%$ 的信道上传输和接收到的向量如图 47.5 左上方所示。图 47.5 中随后的图片显示了迭代概率译码过程。一系列图片显示了 0、1、2、3、10、11、12、13 次迭代后，由迭代译码器逐个比特给出的最佳估计。译码器在第 13 次迭代后停止，此时最佳估计和奇偶校验不相悖。这个最终的译码是无误的。

在一个不寻常的噪声传输情形，译码算法不能找到一个有效的译码。对于这个码和一个 $f=7.5\%$ 的信道，这种失败在每 100 000 次传输中只发生一次。如图 47.6 所示为这个差错率与典型纠错码的分组差错率的比较。

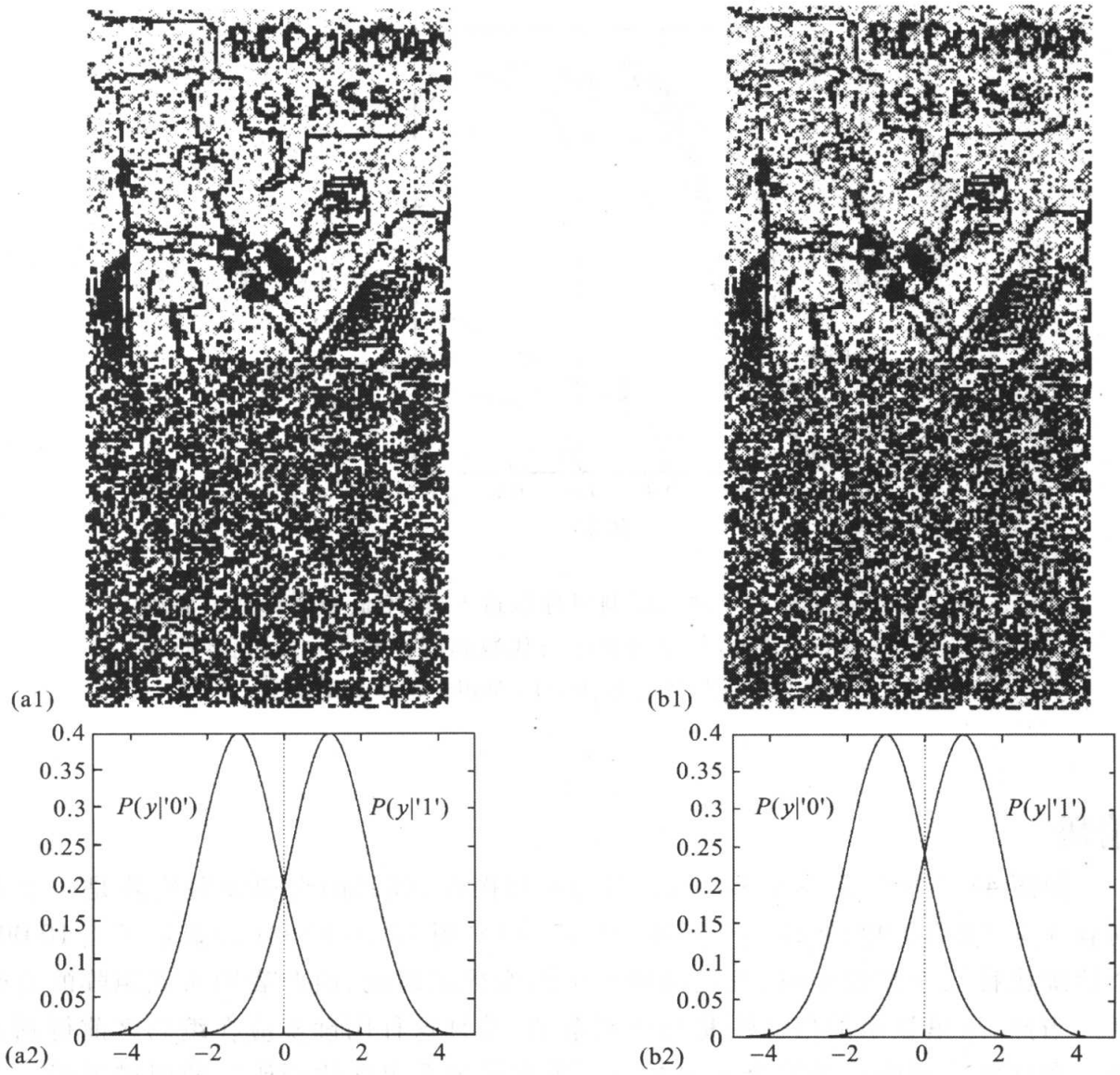


图 47.7 高斯信道中的 Gallager 码图示。(a1) 通过 $x/\sigma = 1.185$($E_b/N_0 = 1.47$ dB)的高斯信道传输后的接收向量。灰度代表归一化似然值。这个传输可以通过和 - 积译码器进行正确译码。译码失败的经验式概率约为 10^{-5}。(a2) 在两种可能的输入条件下,$x/\sigma = 1.185$的信道输出 y 的概率分布。(b1) $x/\sigma = 1.0$ 的高斯信道中所接收的传输,它对应于香农极限。(b2) 在两种可能的输入条件下,$x/\sigma = 1.0$ 的信道输出 y 的概率分布。

高斯信道

图 47.7 左侧显示了通过 $x/\sigma = 1.185$ 的高斯信道传输后的接收向量。灰度表示归一化似然的值,$\dfrac{P(y|t=1)}{P(y|t=1)+P(y|t=0)}$。这个信噪比 $x/\sigma = 1.185$ 是码率为 $\dfrac{1}{2}$ 的 Gallager 码能可靠通信的噪声水平(差错概率约为 10^{-5})。为了表明我们靠香农极限

有多近，右半图显示当信噪比下降到$\frac{x}{\sigma}=1.0$时的接收向量，它对应于码率为$\frac{1}{2}$的码的香农极限。

随着码的参数而变化的性能

图 47.8 显示参数 N 和 j 如何影响低密度奇偶校验码的性能。正如香农所预言，增加码长会导致性能的提升。对 j 的依赖遵循不同的模式。假定一个最优译码器，当码最接近随机码时将获得最佳性能，即码有最大的 j。但是，和 - 积译码器在稠密图上只有很小的改善，所以当 j 取值很小时，才可获得最佳性能。在图中所示 j 的取值中，对分组长度为 816 来说，$j=3$ 是最好的，其分组差错概率降至 10^{-5}。

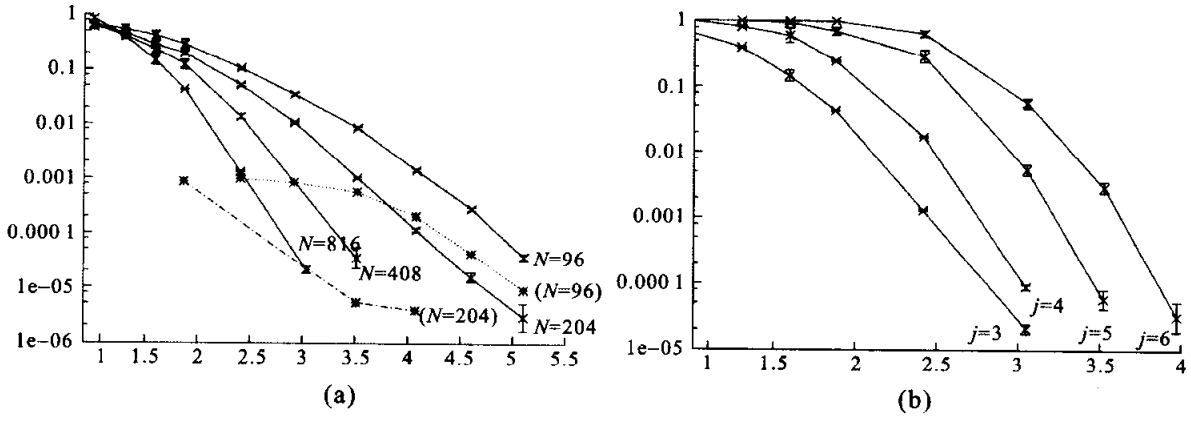


图 47.8 高斯信道中码率为 1/2 的 Gallager 码的性能。纵轴：分组差错概率。横轴：信噪比 E_b/N_0。(a) $(j,k)=(3,6)$码对分组长度 N 的依存关系。从左到右：$N=816$，$N=408$，$N=204$，$N=96$。虚线表示未检测差错的频率，它仅在分组长度小至 $N=96$ 或 $N=204$ 时可以测量出来。(b) 分组长度为 $N=816$ 的码对列权重 j 的依存关系。

这一观察促使人们用某些权重为 2 的列来构造 Gallager 码。如图 47.9(b)所示为 $M/2$ 列的权重为 2 的一个构造。权重为 2 的列太多了，码的性能会变得很差。

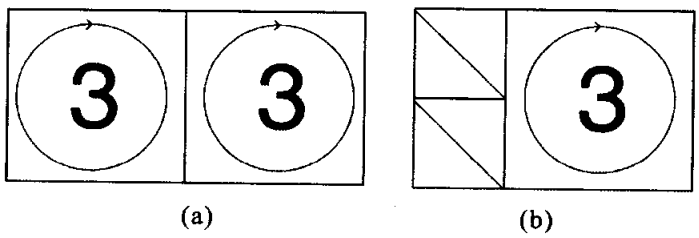


图 47.9 (a) 构造一个 $j=3,k=6,R=1/2$ 的完全规则 Gallager 码的示意图；(b) 构造一个码率为 1/3 的接近规则 Gallager 码的示意图。记号：图中的整数表示在相邻方块中叠加的置换矩阵个数。对角线表示单位矩阵。

正如后面要讨论的，可以将码设计得更不规则以获得更好的结果。

47.5 密度演化

一个用来研究译码算法的方法是假设它运行在一个与 Gallager 码图具有相同本地拓扑的无限树形图上。矩阵 $\boldsymbol{H}$ 越大，译码性质就越接近那些无限图。

假设一个没有回路的无限置信度网络，其中的每个比特 x_n 连接到 j 个校验点，且每个校验点 z_m 连接到 k 个比特（图 47.11）。考虑这个网络中的迭代信息流，并查看一个比特的平均熵，以迭代次数的函数表示。在每次迭代中，一个比特累积了来自本地网络到一个半径等于迭代次数的圆形区域内的信息。只有随着迭代次数的增加，一个比特的平均熵降为零时，才会得到成功的译码。

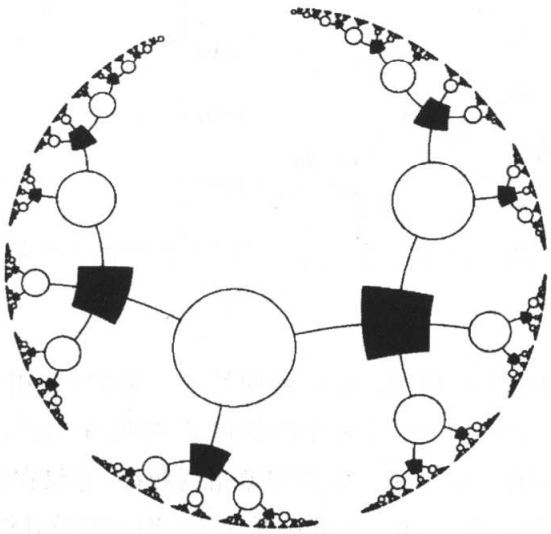


图 47.11 列权重为 $j=3$、行权重为 $k=4$ 的 Gallager 码图的本地拓扑结构。白色结点表示比特 x_l；黑色结点表示校验 z_m；每条边对应于 $\boldsymbol{H}$ 中的一个 1。

一个无限置信度网络的迭代可以通过蒙特卡罗方法来仿真，这是 Gallager（1963 年）首次使用的技术。假设以 1 比特为中心、以 I（迭代的总次数）为半径的网络。我们的目标是计算已知半径 I 之类的所有校验点的状态 $\boldsymbol{z}$ 时中心比特 x 的条件熵。要计算已知某个校正子 $\boldsymbol{z}$ 时中心比特为 1 的概率，会涉及从网络外到中心的 I 步传播。在第 i 次迭代时，半径 $I-i+1$ 处的概率 r 被变换为 qs，再在半径 $I-i$ 处变为 rs；变化方式与半径 $I-i$ 处未知比特的状态 x 有关。在蒙特卡罗方法中，与其对这个网络进行精确仿真（这将花费随着 I 呈指数增长的时间），不如在每次迭代中产生 $\{r,x\}$ 值的一个代表性样本（比如大小为 100）。对于具有参数 j、k 的一个规则网络，在第 i 次迭代中，队列中每对新的 $\{r,x\}$ 是这样产生的：从 x 的分布中抽取一个新的 x，并从第（i

-1)次迭代队列中随机抽取$(j-1)(k-1)$对$\{r,x\}$进行替换;将它们汇聚成一段树形图(图 47.12),并且自顶向下运行和-积算法来找出对应于新结点的新r值。

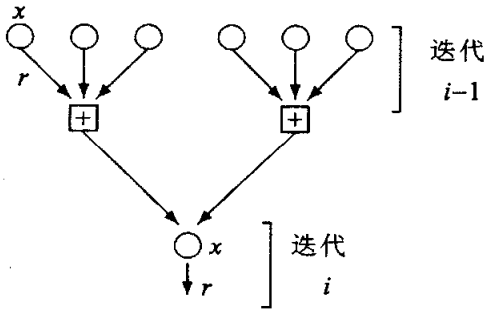


图 47.12 密度演化的蒙特卡罗仿真过程所构建的一段树形图。这个段适合于$j=3,k=4$的规则 Gallager 码。

作为一个实例,如图 47.10 所示为$j=4,k=8$且噪声密度f在 0.01 和 0.10 之间,每次迭代使用 10 000 个采样值时的运行结果。在足够低的噪声水平下的运行结果表明,经过次数较少的迭代,熵会降为零。那些在高噪声水平下的运行,熵会降为一个非零值,并对应于一次译码失败。

这两种特性之间的界线叫做二进制对称信道译码算法的阈值。图 47.10 通过蒙特卡罗仿真表明,规则$(j,k)=(4,8)$码的阈值约为 0.075。Richardson 和 Urbanke (2001a)用直接分析方法巧妙地推导出规则码的阈值。这些阈值的一部分列在表 47.13 中。

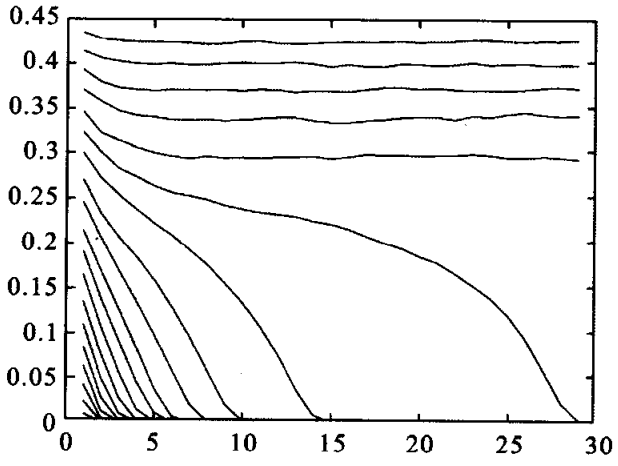


图 47.10 随着$j=4,k=8$的译码过程,密度演化的蒙特卡罗仿真。每条曲线表示一个比特的平均熵,以迭代次数的函数表示,它是用每次迭代 10 000 个样本的蒙特卡罗算法估计出来的。二进制对称信道的噪声水平f以步长 0.005 由图底($f=0.010$)到图顶($f=0.100$)增加。在$f=0.075$附近,显然存在一个阈值,在此阈值之上时算法不能确定x。来自 MacKay(1999b)。

表 47.13 假设采用和 - 积译码算法，规则低密度奇偶校验码的阈值 f_{max}，来自 Richardson 和 Urbanke(2001a)。码率为 1/2 的码的香农极限为 $f_{max}=0.11$。

(j,k)	f_{max}
(3,6)	0.084
(4,8)	0.076
(5,10)	0.068

近似的密度演化

从实用角度考虑，可以这样降低密度演化的计算代价：对密度演化中信息的概率分布做高斯近似，并且仅更新这些近似中的参数。这些方法产生了一个叫做 EXIT 图的示意图，关于它们的更进一步信息，请参见(ten Brink，1999 年；Chung 等，2001 年；ten Brink 等，2002 年)。

47.6 改进 Gallager 码

自从 Gallager 码被重发现以来，已经找到了提高其性能的两种方法。

将比特位和校验位组合在一起

首先，可以这样构造 Gallager 码：将变量结点组合成包含多个(比如 3 个)二进制变量的元变量，而校验结点也同样组合成元校验。与以前一样，一个稀疏图可以通过连接元变量点和元校验点来构成，而组内变量点和校验点具体如何连接则具有很大的自由度。一种用来设置连线的方法是，在有限域 $GF(q)$ 诸如 $GF(4)$ 或 $GF(8)$ 上进行，用 $GF(q)$ 的元素来定义低密度奇偶校验矩阵，并使用诸如表 47.14 所给 $GF(4)$ 之类的映射方法将二进制信息变换为 $GF(q)$。现在，当在译码中进行消息传递时，这些消息是二进制变量联合的概率和似然。例如，如果每个组包含 3 个二进制变量，那么似然会描述这些比特的 8 种可能状态。

表 47.14 $GF(4)$ 和二进制之间的消息符号转换

$GF(4)$	↔	二进制
0	↔	00
1	↔	01
A	↔	10
B	↔	11

$GF(4)$和二进制矩阵元素之间的转换如表 47.15 所示。

表 47.15 $GF(4)$和二进制矩阵元素之间的转换。通过这种方法，可以将 $GF(4)$上的一个 $M \times N$ 奇偶校验矩阵转换成一个 $2M \times 2N$ 的二进制奇偶校验矩阵。

$GF(4)$	→	二进制
0	→	00 00
1	→	10 01
A	→	11 10
B	→	01 11

使用经过仔细优化的结构，$GF(4)$、$GF(8)$和 $GF(16)$上所得码的性能要比与之相当的二进制 Gallager 码几乎好 1 个分贝。

$GF(q)$中译码计算代价的数量级为 $q\log q$，如果校验结点使用合适的傅里叶变换：校验到变量信息的更新规则

$$r_{mn}^{a} = \sum_{x:x_n=a} \mathbb{I}\left[\sum_{n' \in \mathcal{N}(m)} H_{mn'} x_{n'} = z_m\right] \prod_{j \in \mathcal{N}(m)\backslash n} q_{mj}^{x_j} \tag{47.15}$$

是量 q_{mj}^{a}的一个卷积，所以可以用 $q_{mj}^{a}(j \in \mathcal{N}(m)\backslash n)$的傅里叶变换之积取代求和，然后再进行傅里叶逆变换。$GF(4)$的傅里叶变换如算法 47.16 所示。

$$\begin{aligned}
F^0 &= [f^0 + f^1] + [f^A + f^B] \\
F^1 &= [f^0 - f^1] + [f^A - f^B] \\
F^A &= [f^0 + f^1] - [f^A + f^B] \\
F^B &= [f^0 - f^1] - [f^A - f^B]
\end{aligned}$$

算法 47.16 $GF(4)$的傅里叶变换。$GF(2)$上的一个函数 f 的傅里叶变换 F 由 $F^0 = f^0 + f^1$和 $F^1 = f^0 - f^1$给出。$GF(2^k)$上的变换可以视做在每个 k 维上的一系列二进制变换。除了还要除以 2^k之外，逆变换与傅里叶变换相同。

使图变得不规则

改进 Gallager 码的第二种方法是使它们的图变得不规则，这种方法由 Luby 等(2001b)提出。不给所有变量结点赋予相同的阶次 j，而是使某些变量结点用 2 阶，有些用 3 阶，有些用 4 阶，还有一些用 20 阶。也可以给校验结点赋予不等的阶次——这将有助于提高删除信道的性能，但是对于高斯信道而言，结果证明最好的图具有规则的校验阶次。

图 47.17 说明 Gallager 码的这两种改进方法所带来的好处，集中考虑码率为 1/4 的码。使二进制码变得不规则，可以获得约 0.4 dB 的性能提升；从 $GF(2)$ 切换到 $GF(16)$ 可以得到大约 0.6 dB 的提升；而 Matthew Davey 码综合了这些特点——它是 $GF(8)$ 上的不规则码，与规则二进制 Gallager 码相比约提升 0.9 dB。

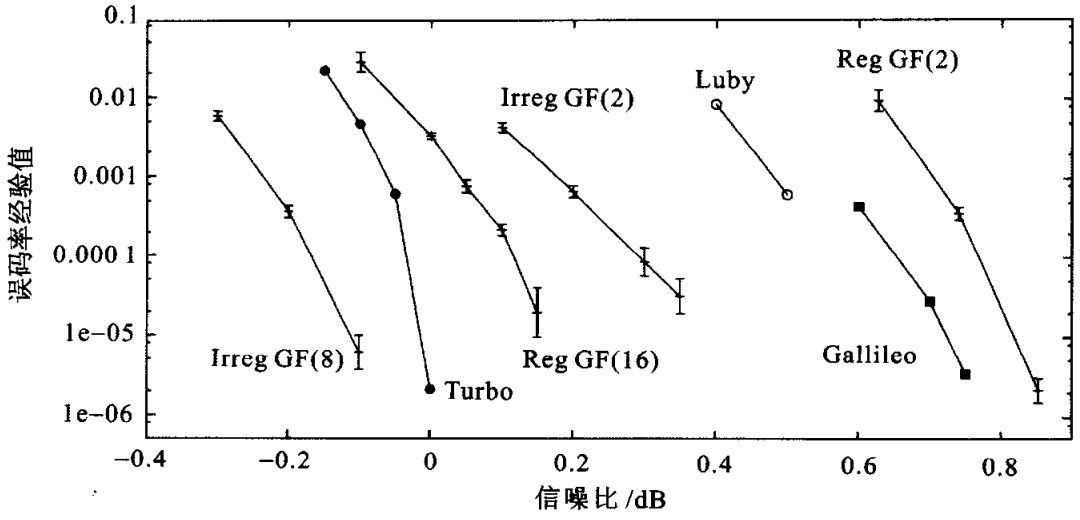


图 47.17 规则二进制 Gallager 码和不规则码、$GF(q)$ 上的码和其他码率为 1/4 的优秀编码的比较。从左（最好的性能）到右为：$GF(8)$ 上的不规则低密度奇偶校验码，分组长度为 48 000 比特（Davey，1999 年）；JPL turbo 码（JPL，1996 年），分组长度为 65 536；$GF(16)$ 上的规则低密度奇偶校验码，分组长度为 24 448 比特（Davey 和 Mackay，1998 年）；不规则二进制低密度奇偶校验码，分组长度为 16 000 比特（Davey，1999 年）；Luby 等（1998 年）不规则二进制低密度奇偶校验码，分组长度 64 000 比特；Galileo 的 JPL 码（1992 年，这是码率为 1/4 的最著名的码）；规则二进制低密度奇偶校验码：分组长度为 40 000比特（MacKay，1999b）。香农极限约为 −0.79 dB。到 2003 年，已经被构造出了更好的稀疏图码。

Richardson 等（2001 年）提出优化一个 Gallager 码分布（profile，即每个阶次上的行数和列数）的方法，并由此导出一个低密度奇偶校验码，它在和 - 积算法译码时的性能与香农极限只有微小差距。

Gallager 码的代数构造

还可以用第三种方法来提升规则 Gallager 码的性能：设计一个具有冗余稀疏约束的码。存在一种差集循环码，例如，$N = 273$ 和 $K = 191$，但是码所满足的不是 $M = 82$ 个而是 N 即 273 个低权重约束条件（图 47.18）。不可能使一个随机 Gallager 码在它们的校验点上到处有这么多冗余。差集循环码的性能比一个等价随机 Gallager 码要好 0.7 dB。

一个未解决的问题是如何发明一种码，使它既有差集循环码的出色性能又可以

差集循环码						
N	7	21	73	273	1057	4161
M	4	**10**	**28**	**82**	**244**	**730**
K	3	11	45	191	813	3431
d	4	**6**	**10**	**18**	34	66
k	3	**5**	**9**	17	33	65

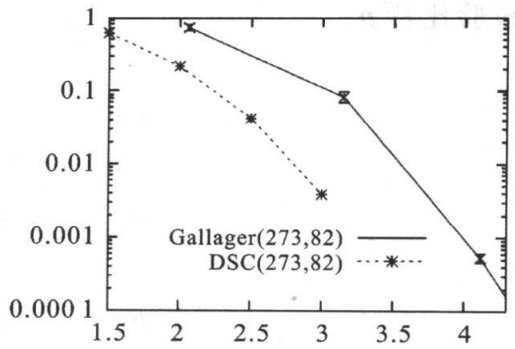


图 47.18 用代数方法构造的低密度奇偶校验码，它满足很多冗余约束，其性能超过一个同等的随机 Gallager 码。表中列出了某些差集循环码的 N、M、K、距离 d 和行权重 k，并重点突出了具有大 d/N、小 k 和大 N/M 的码。在比较中，Gallager 码有 $(j,k)=(4,13)$，其码率等于 $N=273$ 的差集循环码。

有不同的分组长度和码率。称这项任务为 Tanner 挑战。

47.7 低密度奇偶校验码的快速编码

现在讨论低密度奇偶校验码的快速编码方法，它比标准方法要快。在标准方法中，生成矩阵 $\boldsymbol{G}$ 是通过高斯消元法建立的（代价量级为 M^3），然后将每个分组与 $\boldsymbol{G}$ 相乘来编码（代价量级为 M^2）。

阶梯码

在线性时间里可以很容易地对有 M 列权重为 2 或更少的特定低密度奇偶校验矩阵进行编码。例如，如果矩阵有如下式右边所示的阶梯结构

$$\boldsymbol{H}=\left[\begin{array}{cc:c}
\cdot & \cdot & 1 \\
\cdot & \cdot & 1\ 1 \\
\cdot & \cdot & \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1 \\
\cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ 1
\end{array}\right] \tag{47.16}$$

并且如果数据 s 被读入第一个 K 比特中，那么可以在线性时间内从左到右计算出 M 个校验比特 p。

$$\begin{aligned} p_1 &= \sum_{n=1}^{K} H_{1n}s_n \\ p_2 &= p_1 + \sum_{n=1}^{K} H_{2n}s_n \\ p_3 &= p_2 + \sum_{n=1}^{K} H_{3n}s_n \\ &\vdots \\ p_M &= p_{M-1} + \sum_{n=1}^{K} H_{Mn}s_n \end{aligned} \tag{47.17}$$

如果将 H 矩阵分为两部分 $[H_s|H_p]$，就可以分两步来描述编码操作：首先计算一个中间的奇偶校验向量 $v = H_s s$；然后将 v 送入一个累加器以产生 p。

在计算(47.17)中的和式时，如果能利用 H 的稀疏特性，那这种编码方法的代价是线性的。

一般低密度奇偶校验码的快速编码

Richardson 和 Urbanke(2001b)展示了一种极好的方法，用这种方法对任意低密度奇偶校验码的编码代价可以从直接法的 M^2 降为 $N+g^2$，其中差距 g 很可能是一个小的常数，并且即使在最恶劣的情形，它也只是与 N 的一小部分成比例。

在第一步，通过行交换和列交换的方法将奇偶校验矩阵重新排列为一个近似下三角形式，如图 47.19 所示。原矩阵 H 非常稀疏，所以 A、B、T、C、D 和 E 这 6 个矩阵也是很稀疏的。矩阵 T 是下三角矩阵并且对角线元素全部为 1。

$$H = \begin{bmatrix} A & B & T \\ C & D & E \end{bmatrix} \tag{47.18}$$

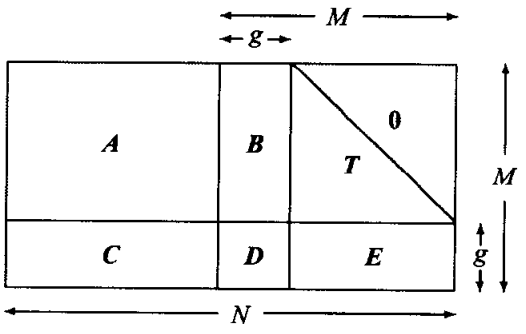


图 47.19 奇偶校验矩阵的近似下三角形式

长度为 $K = N - M$ 的信源向量 s 被编码成一个传输向量 $t = [s, p_1, p_2]$，步骤如

下。

1. 计算信源向量的上校正子

$$z_A = As \tag{47.19}$$

这可以在线性时间内完成。

2. 找出第二奇偶比特的设置 p_2^A，使得上校正子为零。

$$p_2^A = -T^{-1}z_A \tag{47.20}$$

通过回代可以在线性时间内得出这个向量，即，计算 p_2^A 的第一个比特，然后是第二个，然后是第三个，如此等等。

3. 计算向量 $[s, 0, p_2^A]$ 的下校正子

$$z_B = Cs - Ep_2^A \tag{47.21}$$

这也可以在线性时间内完成。

4. 现在准备求一个“聪明”比特。定义矩阵

$$F \equiv -ET^{-1}B + D \tag{47.22}$$

并求出它的逆矩阵 F^{-1}。这个计算只需要完成一次，而它的代价量级为 g^3。这个逆矩阵 F^{-1} 是一个稠密 $g \times g$ 矩阵。[如果 F 是不可逆的，那么 H 不是满秩的，或者对 H 做进一步的列交换可以产生一个可逆的 F。]

设第一个校验比特 p_1 为

$$p_1 = -F^{-1}z_B \tag{47.23}$$

这个操作的代价量级是 g^2。

声明：此时，找到了第一个奇偶比特 p_1 的正确设置。

5. 抛弃临时奇偶比特 p_2^A 并找出新的上校正子

$$z_C = z_A + Bp_1 \tag{47.24}$$

这可以在线性时间内完成。

6. 寻找第二奇偶比特 p_2 的设置，使得上校正子为零

$$p_2 = -T^{-1}z_C \tag{47.25}$$

这个向量可以用回代法在线性时间内求出。

47.8 进一步的阅读

Gallager 在 1962 年首次研究低密度奇偶校验码，然后该码就被编码理论界普遍遗忘了。Tanner(1981 年)通过引入更一般的约束结点，推广了 Gallager 的工作。现在被称为 turbo 乘积码的码实际上应该叫做 Tanner 乘积码，因为它们是 Tanner 提出的，并由他的同事(Karplus 和 Krit，1991 年)进行硬件实现。为 Gallager 码在 20 世纪 90 年代得到重生做出贡献的论文包括(Wiberg 等，1995 年；Mackay 和 Neal，1995 年；MacKay 和 Neal，1996 年；Wiberg，1996 年；Mackay，1999b；Spielman，1996 年；Sipser 和

Spielman,1996 年)。Gallager 码的低精度译码算法和快速编码算法在(Richardson 和 Urbanke,2001a;Richardson 和 Urbanke,2001b)中讨论。MacKay 和 Davey(2000 年)证明了低密度奇偶校验码性能超过 Reed - Solomon 码,甚至在 Reed - Solomon 的强项:高码率和短的分组长度方面也是如此。其他重要文献包括(Luby 等,2001a;Luby 等,2001b;Luby 等,1997 年;Davey 和 MacKay,1998 年;Richardson 等,2001 年;Chung 等,2001 年)。用来设计不规则低密度奇偶校验码的有用工具包括(Chung 等,1999 年;Urbanke,2001 年)。

关于和 - 积算法的深入讨论参见(Wiberg,1996 年;Frey,1998 年;McEliece 等,1998 年)。

从群论和编码理论的角度对低密度奇偶校验码译码所进行的讨论参见(Forney,2001 年;Offer 和 Soljanin,2000 年;Offer 和 Soljanin,2001 年);关于这一主题的背景知识参见(Hartmann 和 Rudolph,1976 年;Terras,1999 年)。关于低密度奇偶校验码实用设计的文献日益增多(Mao 和 Banihashemi,2000 年;Mao 和 Banihashemi,2001 年;ten Brink 等,2002 年);现在从硬盘驱动器到卫星通信等应用都采用了这种码。

低密度奇偶校验码可应用于量子纠错问题,参见 Mackay 等(2003 年)。

47.9 习 题

习题 47.1[2] **译码算法的"双曲正切"版**。在第 47.3 节,首先用 $q_{mn}^{0/1}$ 和 $r_{mn}^{0/1}$ 来讨论低密度奇偶校验码的和 - 积译码算法,然后用量 δ_q 和 δ_r 来讨论。还有第三种描述,用如下对数概率比来代替其中的 $\{q\}$

$$l_{mn} \equiv \ln \frac{q_{mn}^{0}}{q_{mn}^{1}} \tag{47.26}$$

证明

$$\delta q_{mn} \equiv q_{mn}^{0} - q_{mn}^{1} = \tanh(l_{mn}/2) \tag{47.27}$$

推导 $\{r\}$ 和 $\{l\}$ 的更新规则。

习题 47.2[2] 经常有人问我:"对于其他线性码,例如代数码,为什么不通过变换它们的奇偶校验矩阵使它们变为低密度的,并应用和 - 积算法来进行译码?"[回忆一下,只要矩阵 $\boldsymbol{P}$ 是可逆的,$\boldsymbol{H}$ 中行的任意线性组合 $\boldsymbol{H}' = \boldsymbol{PH}$ 对一个码而言是一个有效的奇偶校验矩阵;所以对任意一个码,都有很多奇偶校验矩阵。]

解释为什么一个随机线性码不具有一个低密度奇偶校验矩阵。[这里,低密度意味着"行权重至多为 k",其中 k 是某个 $<<N$ 的小常数。]

习题 47.3[3] 证明如果一个低密度奇偶校验码具有多于 M 个权重为 2 的列(比如 αM 列,其中 $\alpha > 1$),那么该码有权重量级为 $\log M$ 的码字。

习题 47.4[3] 在第 13.5 节中,求出了权重计数函数 $A(w)$ 在所有随机线性码的总体上取平均的期望值。也可以对低密度奇偶校验码的总体进行这种计算(Gallag-

er,1963 年;MacKay,1999b;Litsyn 和 Shevelev,2002 年)。但是,$A(w)$的平均值似乎并非总是很好地指示出 $A(w)$在总体中的典型值。例如,如果当 w 取某一特殊值时,99% 的码有 $A(w)=0$,1% 的码有 $A(w)=100\,000$,那虽然也许可以说 $A(w)$的典型值是 0,但其平均值却为 1 000。试寻找低密度奇偶校验码的典型权重计数函数。

47.10 习题解答

习题 47.2 的解答 考虑这样一些码:其码率为 R,分组长度为 N,有 $K=RN$ 个信源比特和 $M=(1-R)N$ 个奇偶校验比特。设所有码的比特排序,使得前 K 个比特是独立的,于是如愿意就可以将码表示成系统化形式

$$\boldsymbol{G}=[\mathbf{1}_K|\boldsymbol{P}^{\mathrm{T}}];\boldsymbol{H}=[\boldsymbol{P}|\mathbf{1}_M] \tag{47.28}$$

不同线性码的个数是矩阵 $\boldsymbol{P}$ 的个数,等于 $\mathcal{N}_1=2^{MK}=2^{N^2R(1-R)}$。是否可以将所有这些码表示成不同的低密度奇偶校验码呢?

行权重为 k 的低密度奇偶校验矩阵的个数为

$$\binom{N}{k}^M \tag{47.29}$$

而它们所能定义的不同码的最大数为

$$\mathcal{N}_2=\binom{N}{k}^M \Big/ M! \tag{47.30}$$

它远远小于 $\mathcal{N}_1$。所以,根据鸽巢原理,不可能将每个随机线性码都映射到一个低密度 $\boldsymbol{H}$。

第 48 章

卷积码与 Turbo 码

这一章与第 25 章紧密相关，是其后续内容。它使用了码、网格和前向 - 后向算法等思想。

48.1 卷积码概述

当学习线性分组码时，是从三个方面来描述它们的。

1. 生成矩阵描述了如何将一串 K 个任意信源比特转换成一串 N 个传输比特。

2. 奇偶校验矩阵指定了一个有效码字所要满足的 $M = N - K$ 个奇偶校验约束条件。

3. 码的网格使用边线带标记的网格中的路径来描述它的有效码字。

本书没有涵盖用来描述某些分组码的第 4 种方法，即代数方法。(a) 因为在很多其他关于编码理论的书中都已经很好地讨论过了；(b) 因为正如书中的这部分所讨论的，纠错码的最新技术发展很少用到代数编码理论；(c) 因为笔者不能胜任教授这方面的内容。

将从两个方面描述卷积码：第一，从信源比特 $\boldsymbol{s}$ 生成发送比特 $\boldsymbol{t}$ 的机制方面；第二，从用来描述有效传输所满足的约束条件的网格方面。

48.2 线性反馈移位寄存器

让一个信源流通过一个线性滤波器，来生成卷积码的一次传输。这个滤波器使用了一个移位寄存器和线性输出函数，可能还会用到线性反馈。

笔者绘制移位寄存器的方式是从右到左：比特随着时间从右向左滚动。

图 48.1 给出了 3 个可以用来定义卷积码的线性反馈移位寄存器。包围在比特 $z_1 \cdots z_7$ 周围的矩形盒子表示滤波器的记忆，也称为它的状态。所有这 3 个滤波器都有一个输入和两个输出。在每个时钟周期，信源输出 1 比特，而滤波器输出 2 比特 $t^{(a)}$ 和 $t^{(b)}$。将这些比特连成一串，可以从信源流 $s_1 s_2 s_3 \cdots$ 得到一个传输流 $t_1^{(a)} t_1^{(b)} t_2^{(a)} t_2^{(b)} t_3^{(a)} t_3^{(b)} \cdots$。因为对于每个信源比特，都有两个发送比特，所以如图 48.1 所示是码率为 1/2 的码。因为这些滤波器要求 $k = 7$ 比特的记忆，所以它们所定义的码叫做*约束长度为 7 的码*。

卷积码有三种风格，对应于图 48.1 中的 3 类滤波器。

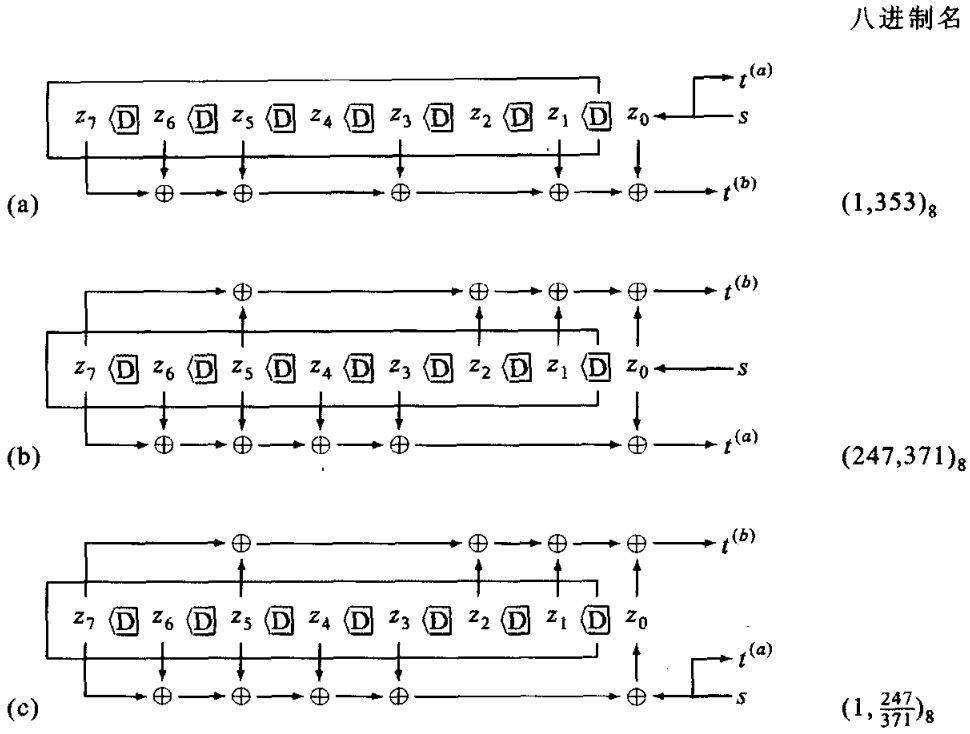


图 48.1 生成码率为 1/2 的卷积码的线性反馈移位寄存器。符号Ⓓ表明有一个时钟周期延时的副本。符号⊕表示无延时的线性模 2 加法。

系统化的非递归编码器

如图 48.1(a)所示滤波器没有反馈。同时它还具有这样一个性质：输出比特 $t^{(a)}$ 与信源比特 s 相同。因为在传输流中透明地复制了信源比特，所以称这个编码器为*系统化的*；又因为它没有反馈，所以它是*非递归的*。另一发送比特 $t^{(b)}$ 是滤波器状态的一个线性函数。描述该函数的一种方法是，将其视做如下两个长度为 $k+1$ 的二进制向量的点乘(模 2)：二进制向量 $\boldsymbol{g}^{(b)}=(1,1,1,0,1,0,1,1)$ 和状态向量 $\boldsymbol{z}=(z_k, z_{k-1},\cdots,z_1,z_0)$。将比特 z_0 包含在状态向量里，该比特是在下一个周期进入记忆的第一个比特。对于存在一个从状态比特 z_κ 到发送比特 $t^{(b)}$ 的抽头(一个向下的箭头)的每个 κ，向量 $\boldsymbol{g}^{(b)}$ 都有 $g_\kappa^{(b)}=1$。

描述这些二进制抽头向量的一种便捷方法是使用八进制。因此，这个滤波器使用抽头向量 353_8。笔者已经从右至左画出了延时线，以便较容易地将图表和这些八进制数字联系起来(如表 48.2 所示)。

表 48.2 延时线中的抽头如何转换成八进制

11	101	011
↓	↓	↓
3	5	3

非系统化的非递归编码器

如图 48.1(b)所示的滤波器也没有反馈,但是它不是系统化的。它使用两个抽头向量 $\boldsymbol{g}^{(a)}$ 和 $\boldsymbol{g}^{(b)}$ 来产生它的两个发送比特。这个编码器因此是非系统化的和非递归的。因为它们的复杂度增加了,所以对于相同的约束长度,非系统码可能会具有优于系统化非递归码的纠错能力。

系统化的递归编码器

如图 48.1(c)所示的滤波器与图 48.1(b)中的非系统化非递归滤波器类似,但是它使用了先前构成 $\boldsymbol{g}^{(a)}$ 的抽头来产生一个线性信号,并与信源比特一起反馈到移位寄存器中。与以前一样,输出 $t^{(b)}$ 是状态向量的一个线性函数。另一输出是 $t^{(a)}=s$,所以滤波器是系统化的。

一个递归码通常可以方便地看做一个八进制比值,例如,图 48.1(c)的码记为 $(247/371)_8$。

系统化递归码和非系统化非递归码的等价性

图 48.1(b)和(c)的两个滤波器是码等价的,因为它们定义了相同的码字集。对于非系统化的非递归码的每个码字,可以选择另一个编码器的信源流,使得它们的输出是相同的(反之亦然)。

为了证明这一点,如图 48.3(a)和(b)所示,用 p 表示量 $\sum_{k=1}^{k} g_k^{(a)} z_k$,它显示了一对更小的但却等价的滤波器。如果两个传输是等价的——即两幅图的 $t^{(a)}$ 和 $t^{(b)}$ 分别相等——则在每个周期,系统码中的信源比特一定是 $s=t^{(a)}$。所以现在只需简单地确证对于 s 的这种选择,假设状态在开始时是匹配的,系统码的移位寄存器与非系统码的移位寄存器具有相同的状态序列。在图 48.3(a)中有

$$t^{(a)}=p\oplus z_0^{\text{非递归}} \tag{48.1}$$

然而在图 48.3(b)中,有

$$z_0^{\text{递归}}=t^{(a)}\oplus p \tag{48.2}$$

替换 $t^{(a)}$ 并使用 $p\oplus p=0$,立即发现

$$z_0^{\text{递归}}=z_0^{\text{非递归}} \tag{48.3}$$

因此,一个非系统化非递归码的任意

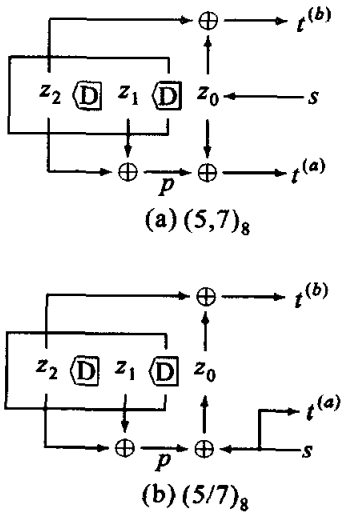


图 48.3 两个码率为 1/2、约束长度为 $k=2$ 的卷积码:(a) 非递归的;(b) 递归的。两个码等价。

码字是具有相同抽头的系统化递归码的一个码字——从某种意义上说，具有相同抽头是指在图 48.3(a)和(b)中有垂直箭头之处都相同，尽管在(b)中有一个箭头不是朝下而是朝上。

尽管这两个码是等价的，但两个编码器的特性却不同。非递归编码器具有一个有限冲激响应，即如果输入字符串中除了一个 1 之外全部为 0，则所得的输出流包括有限个 1。当这个 1 比特通过记忆中的所有状态时，延时线返回到全零状态。图48.4(a)显

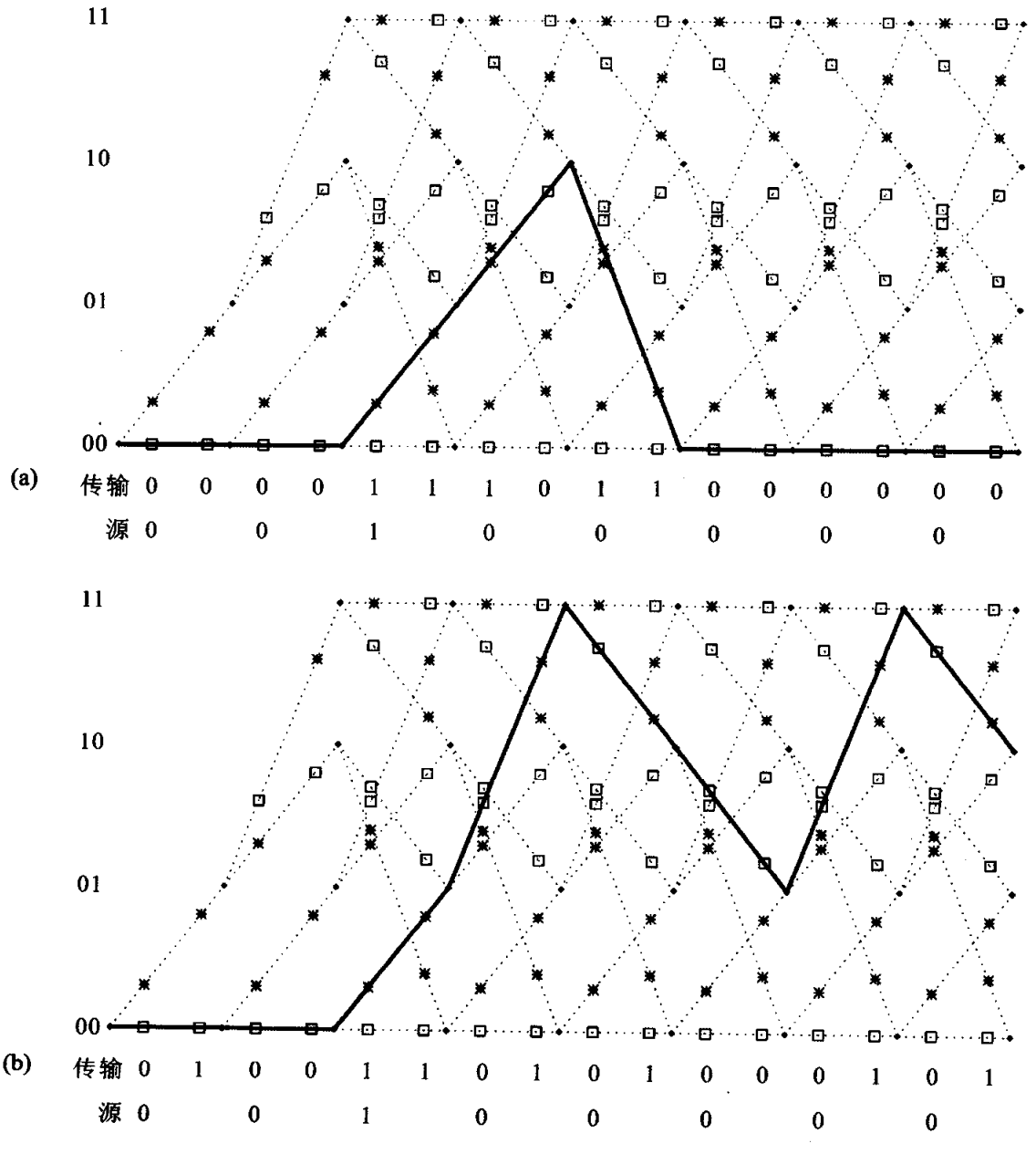


图 48.4 图 48.3 中码率为 1/2 的卷积码的网格图。假设滤波器的初始状态为$(z_2, z_1) = (0,0)$。横坐标是时间，纵坐标是滤波器在每个时间步的状态。每条线段上都标有要发送的符号 $t^{(a)}$ 和 $t^{(b)}$，星号表示状态“1”，方块表示状态“0”。当信源序列为 00100000 时，穿过网格图的路径用实线突出地标出来了。虚线表示其他信源序列的可能状态轨迹。

示了由信源字符串 $\boldsymbol{s}=(0,0,1,0,0,0,0,0)$ 产生的状态序列。

图 48.4(b)给出了图 48.3(b)的递归码的网格图和在同样信源字符串 $\boldsymbol{s}=(0,0,1,0,0,0,0,0)$ 下这个滤波器的响应。这个滤波器具有一个无限冲激响应。响应稳定于周期状态,其周期等于 3 个时钟周期。

▷ **习题 48.1**[1] 要使得递归滤波器的状态序列和传输与非递归滤波器的一样,递归滤波器的输入应该是什么?(提示:参见图 48.5。)

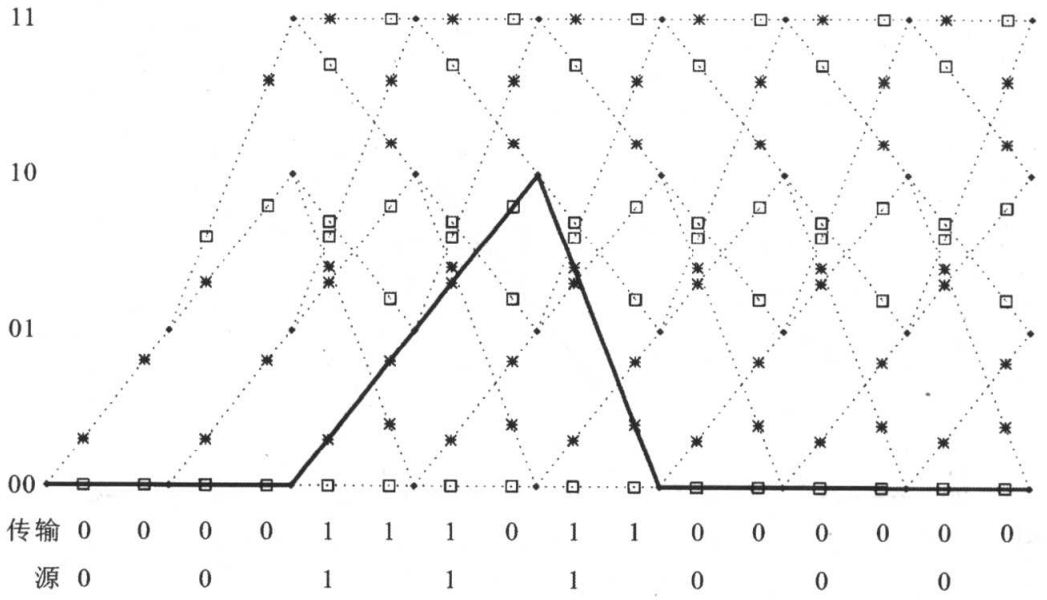


图 48.5 系统化递归码的信源序列 00111000 穿过网格图时,产生与非系统化非递归情形相同的路径。

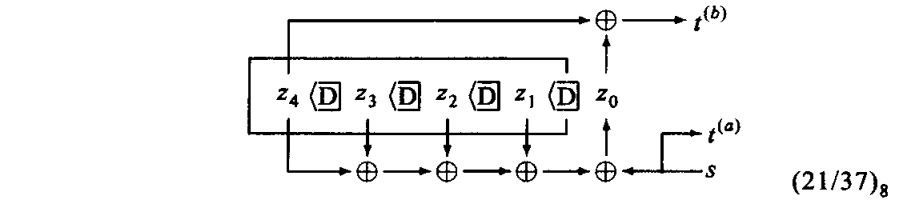
通常一个有 k 比特记忆的线性反馈移位寄存器具有周期最大为 2^k-1 的周期性冲激响应,对应于滤波器,访问它的状态空间中的每个非零状态。

顺便提及,廉价的伪随机数发生器和廉价的密码产品所使用的正是这些周期序列,不过 k 值会大于 7;随机数种子或密钥选择记忆的初始状态。因此,密码分析问题和卷积码的译码之间存在紧密的联系。

48.3 卷积码的译码

接收方接收到一个比特流,希望推导出状态序列,并由此得出信源流。每个比特的后验概率可以通过在第 25.3 节中介绍的和 - 积算法找到(也称为前向 - 后向或 BCJR 算法)。最可能的状态序列可以用第 25.3 节中的最小和算法来寻找(也称为 Viterbi 算法)。这个任务的核心如图 48.6 所示,它画出了一个有 16 状态码的网格图中每条边线所对应的代价;假设信道为二进制对称信道,并且接收向量除了 1 比特已经被翻转之外,都等于一个码字。根据似然值的不同,有 3 种类型的线:实线表示与

接收字符串的两个比特精确对应的网格图中的边线；粗虚线表示与 1 比特相匹配而另一比特失配的边线；细虚线表示边线与两个比特都不匹配。最小和算法尽可能多地使用实线在网格图中搜寻路径；更精确地，它最小化路径代价，其中一条实线的代价为零，一条粗虚线的代价为 1，一条细虚线的代价为 2。



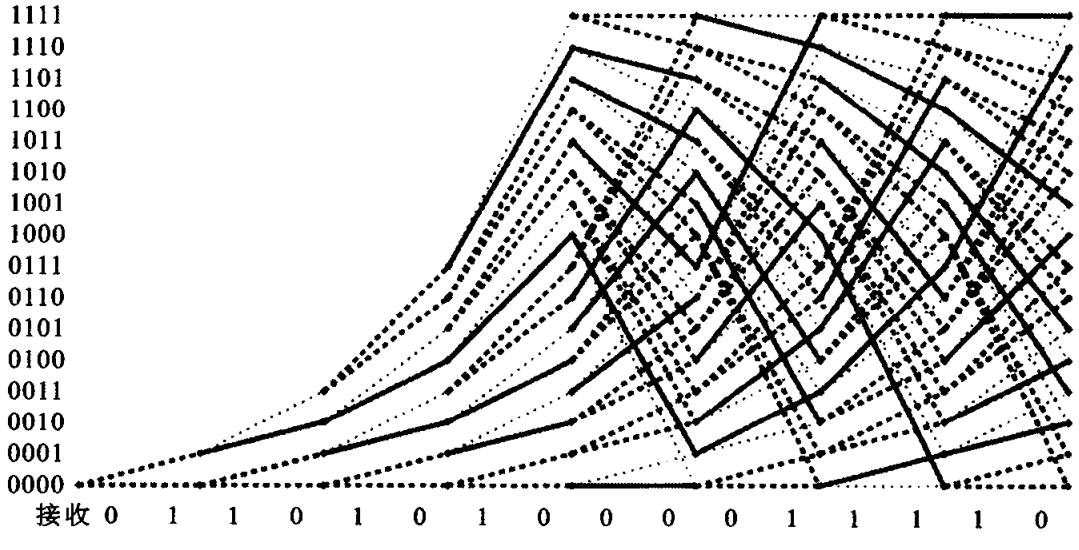


图 48.6 $k=4$ 的码的网格图，并在上面标出了当接收向量等于只有 1 比特被翻转的码字时的似然函数。根据似然的值，有 3 种线的类型：实线表示与接收字符串中相应的两个比特准确匹配时的网格线；粗虚线表示与 1 比特相匹配而另一比特失配的边线；细虚线表示边线与两个比特都失配。

▷ **习题 48.2**[1] 是否可以找出最可能的路径和被翻转的比特？

不平等的保护

到目前为止，所讨论的卷积码有一个缺点，就是它们对信源比特提供不平等的保护。图 48.7 显示了仅仅两个发送比特不同时网格图中的两条路径。最后的信源比特比其他信源比特受到更少的保护。这种对比特的不平等保护促使了网格图的终止。

一个被终止的网格图如图 48.8 所示。终止使得每个码字所用的信源比特数略微减少。这里，4 个信源比特被转换为奇偶比特，因为 $k=4$ 个记忆比特必须归零。

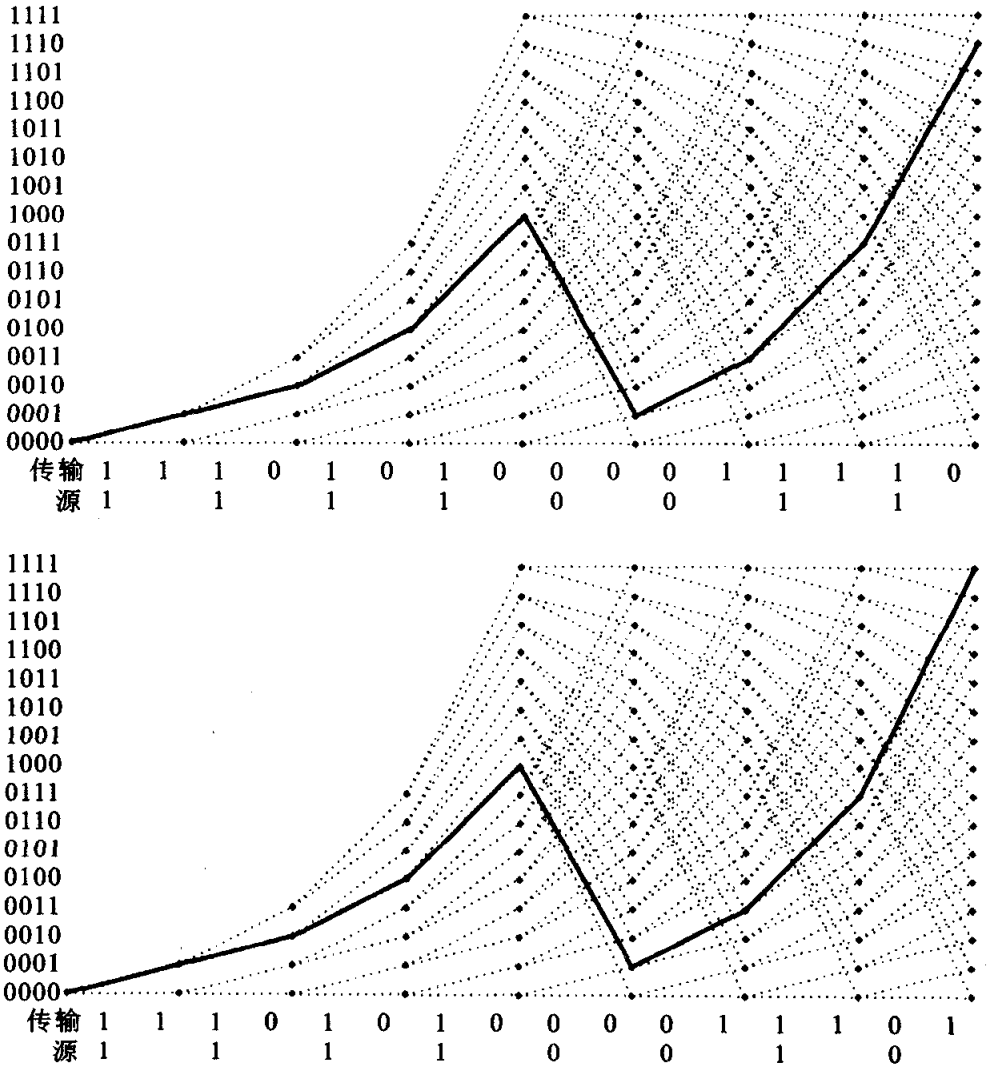


图 48.7 只有两个发送比特有异的两条路径

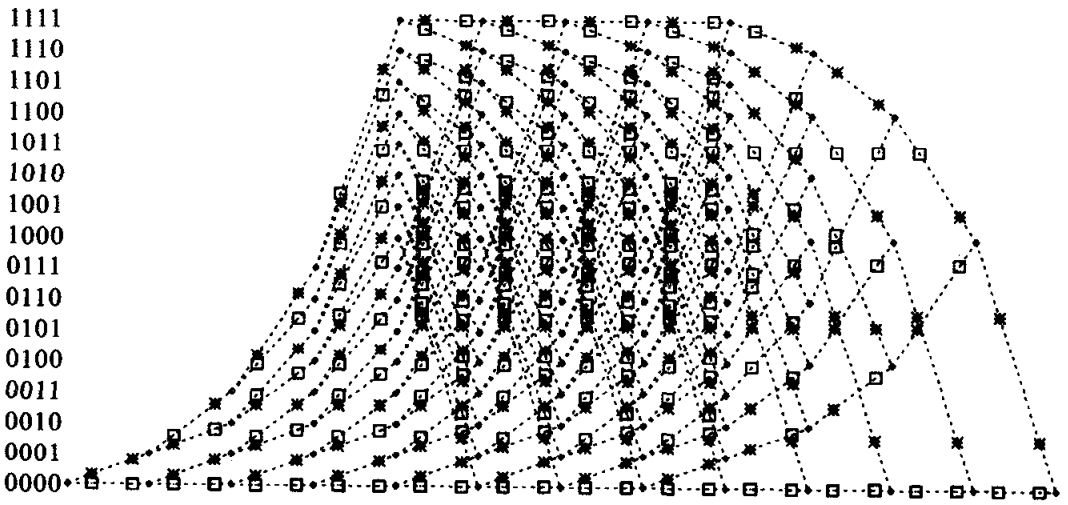


图 48.8 一个被终止的网格图

48.4 Turbo 码

一个(N,K)Turbo 码由若干(通常是两个)分量卷积码编码器和等数量的交织器来定义,其中交织器是$K\times K$置换矩阵。不失一般性,取第一个交织器为单位矩阵。一串K个信源比特按照与之相关的交织器所定义的顺序,送入各个分量编码器进行编码,并发送各分量编码器的输出比特。通常选择第一个分量编码器为系统化编码器,正如图 48.6 中的递归滤波器所示;第二个分量编码器是一个码率为 1 的非系统化编码器,它仅仅输出奇偶比特。于是,发送码字包括K个信源比特,后接由第一个卷积码所生成的M_1个奇偶比特和由第二个卷积码所生成的M_2个奇偶比特。所得 Turbo 码的码率为 1/3。

Turbo 码可以用因子图来表示,其中的两个大矩形结点表示两个网格图(图 48.9(a));K个信源比特和最前面的M_1个奇偶比特加入第一个网格,而K个信源比特和最后M_2个奇偶比特加入第二个网格。每个码字比特或者参与一个网格图,或者参与两个网格图,这取决于该比特是奇偶比特还是信源比特。每个网格图结点包含一个完全类似图 48.8 所示的被终止网格图,只是要长 1 000 倍。[还可以用其他因子图来表示 Turbo 码,这些网格图使用更多的基本结点,但是这里给出的因子图所产生的是用于 Turbo 码的和 - 积算法的标准版本。]

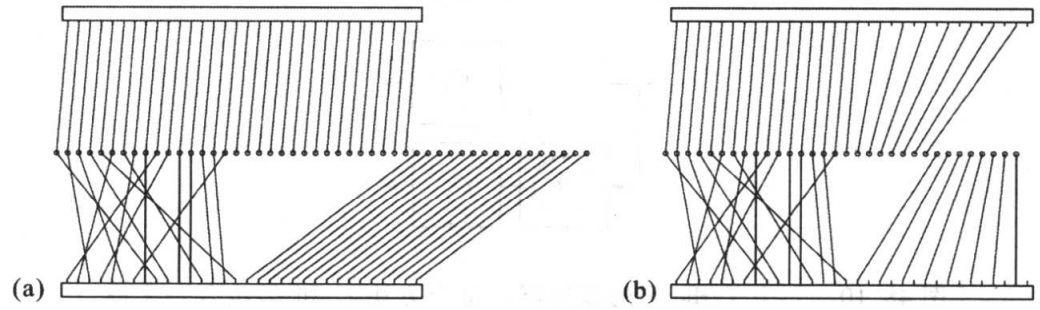


图 48.9 用因子图表示的(a) 码率为 1/3 的 Turbo 码和(b) 码率为 1/2 的 Turbo 码。圆圈代表码字比特。两个矩形表示码率为 1/2 的卷积码的网格图,系统比特占据左半矩形,奇偶比特占据右半矩形。在码率为 1/2 的 Turbo 码中,将这些分量码的截除表示为在每个网格图中有一半的奇偶比特没有连接。

如果要求一个码率更小的 Turbo 码,诸如 1/2,对码率为 1/3 的码所进行的一个标准修改是将某些奇偶比特删除(图 48.9(b))。

使用第 26 章所描述的和 - 积算法,来对 Turbo 码进行译码。在第一次迭代时,每个网格图接收到信道似然,并运行前向 - 后向算法,在已知其他比特信息的前提

下,计算每个比特为 1 或 0 的相对似然值。然后这些似然从一个网格图传给另外一个,并在传递过程中乘以信道的似然。这样就为第二次迭代做好准备:使用更新后的概率,在每个网格图中再次运行前向 - 后向算法。希望在大约 10 次或 20 次这样的迭代之后,能找到正确的译码。通常在某个固定次数的迭代之后停止,但是还可以做得更好。

随后的这个过程可以在每次迭代中作为停止判据使用。在每个网格图的每个时间步中,根据局部信息,确定最可能的边线。如果这些最可能的边线连接成两条有效路径,每个网格图中一条,并且如果这两条路径彼此相合,则在此时停止是合理的,因为随后的迭代不太可能使译码器偏离该码字。如果最大次数的迭代完成之后还没有满足停止标准,则可以报告一个译码错误。推荐使用这个停止程序有几方面原因:它可以大大节省译码时间而不招致差错概率的损失;它允许识别出译码器所检测到的译码失败——得知某个特定的分组被完全破坏对于接收方来说是非常有用的信息!此外,当区分已检测到的和未检测到的错误时,未检测出来的错误有助于深入了解码的低权重码字,这可能会改进编码设计过程。

这里所描述的 Turbo 码具有优异性能(如图 48.10 所示),其译码差错概率可以小至约 10^{-5} 的水平,但是随机构造的 Turbo 码从 10^{-5} 这一水平开始有出现错误平台的倾向。这种错误平台是由低权重码字产生的。为了降低错误平台的高度,可以试图通过改变随机结构来增加这些低权重码字的权重。对 Turbo 码进行调整就像是玩魔术,它从未成功地完全消除低权重码字;更精确地讲,低权重码字仅可以通过牺牲 Turbo 码的优异性能来消除。与之相对,低密度奇偶校验码很少有错误平台。

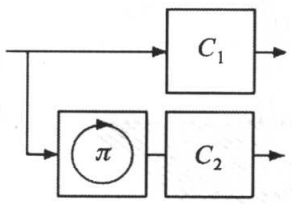


图 48.10 一个 Turbo 码的编码器。每个方块 C_1 和 C_2 包含一个卷积码。信源比特在送入 C_2 之前使用一个置换 π 进行重排序。通过串联或交织两个卷积码的输出,可得出发送码字。

48.5 卷积码和 Turbo 码的奇偶校验矩阵

最后将码率为 1/2 卷积码视为一个线性分组码,并讨论它的奇偶校验矩阵。采用如下约定:一个分组的 N 个比特由 $N/2$ 个比特的 $t^{(a)}$ 和 $N/2$ 个比特的 $t^{(b)}$ 组成。

▷ **习题 48.3**[2] 证明一个卷积码具有如图 48.11(a)所示低密度奇偶校验矩阵。

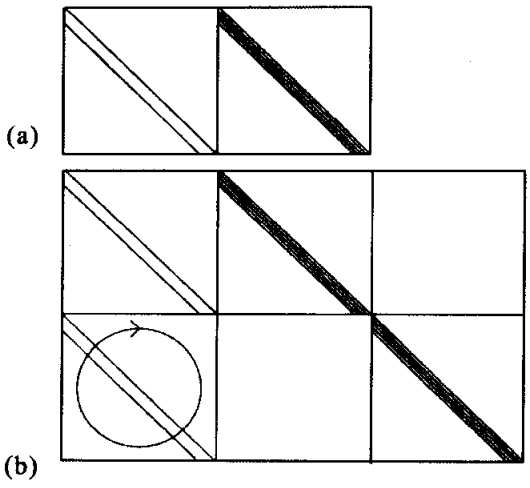


图 48.11 (a) 一个码率为 1/2 的卷积码的奇偶校验矩阵示意图;(b) 一个码率为 1/3 的 Turbo 码的奇偶校验矩阵示意图。注意:对角线代表一个单位矩阵。对角线上的一条带子代表对角线上带子内都是 1。方格内的一个圆圈代表在这个方阵中所有列的随机排列。一个方格内的数字代表叠加在其上的随机置换矩阵的数量。水平线和垂直线表示矩阵中分组的界线。

提示:通过考虑非系统化的非递归编码器(图 48.1(b)),很容易找出一个卷积码所满足的奇偶约束条件。考虑将已经通过卷积滤波器 b 的一串码流送入滤波器 a,同样地将已经通过卷积滤波器 a 的一串码流送入滤波器 b;比较所得的两个比特流。忽略网格图的终止。

通过列出两个分量网格图(图 48.11(b))所满足的约束条件,可以写出一个 Turbo 码的奇偶校验矩阵。所以 Turbo 码也是低密度奇偶校验码的一个特例。如果一个 Turbo 码被删除,那么它就不一定具有低密度奇偶校验矩阵,但是如下一章所述,它总是具有一个稀疏的广义奇偶校验矩阵。

进一步的阅读

对于想进一步学习卷积码的读者,笔者强烈推荐参考文献 Johannesson 和 Zigangirov(1999 年)。笔者很想涵盖的一个主题是序贯译码。序贯译码仅仅探究网格图中最有希望的路径,当积累的证据表明转错了方向时,则由原路退回。当网格图太大,不能应用最大似然算法即最小和算法时,使用序贯译码。可以在 Johannesson 和 Zigangirov(1999 年)中读到关于序贯译码的知识。

要进一步学习 Turbo 码中和 - 积算法的使用和很少使用但是强烈推荐的用于终止其译码算法的停止判据,Frey(1998 年)是最基本的读物。(在这本书中还有许多好内容!)

48.6 习题解答

习题 48.2 的解答 第一个比特被翻转了。最可能的路径是图 48.7 中上面的那一条。

第 49 章

重复累加码

第 1 章讨论了在一个噪声信道上通信的一种非常简单但并非很有效的方法：重复码。现在讨论几乎也是如此简单但性能却异常好的码。

Divsalar 等(1998 年)是出于理论研究的目的来研究重复累加码的，因为简单的类 Turbo 码比凌乱的 Turbo 码更易于分析。结果证明它们的实际性能与其他稀疏图码的性能一样好。

49.1 编 码 器

1. 取 K 个信源比特 $s_1 s_2 s_3 \cdots s_K$

2. 将每个比特重复 3 次，得 $N = 3K$ 比特

$$s_1 s_1 s_1 s_2 s_2 s_2 s_3 s_3 s_3 \cdots s_K s_K s_K$$

3. 用一个随机置换来改变这 N 个比特的顺序(一个固定的随机置换——对每个码字都相同)。令置换后的比特流为 $\boldsymbol{u}$。

$$u_1 u_2 u_3 u_4 u_5 u_6 u_7 u_8 u_9 \cdots u_N$$

4. 发送累加和。

$$\begin{aligned} t_1 &= u_1 \\ t_2 &= t_1 + u_2 \pmod 2 \\ &\cdots \\ t_n &= t_{n-1} + u_n \pmod 2 \\ &\cdots \\ t_N &= t_{N-1} + u_N \pmod 2 \end{aligned} \tag{49.1}$$

5. 这就是它！

49.2 图

重复累加码如图 49.1(a)所示。图中使用了 4 种类型的结点：等式约束条件⊟，中间二进制变量(黑色圆圈)，奇偶约束条件⊞和发送比特(白色圆圈)。

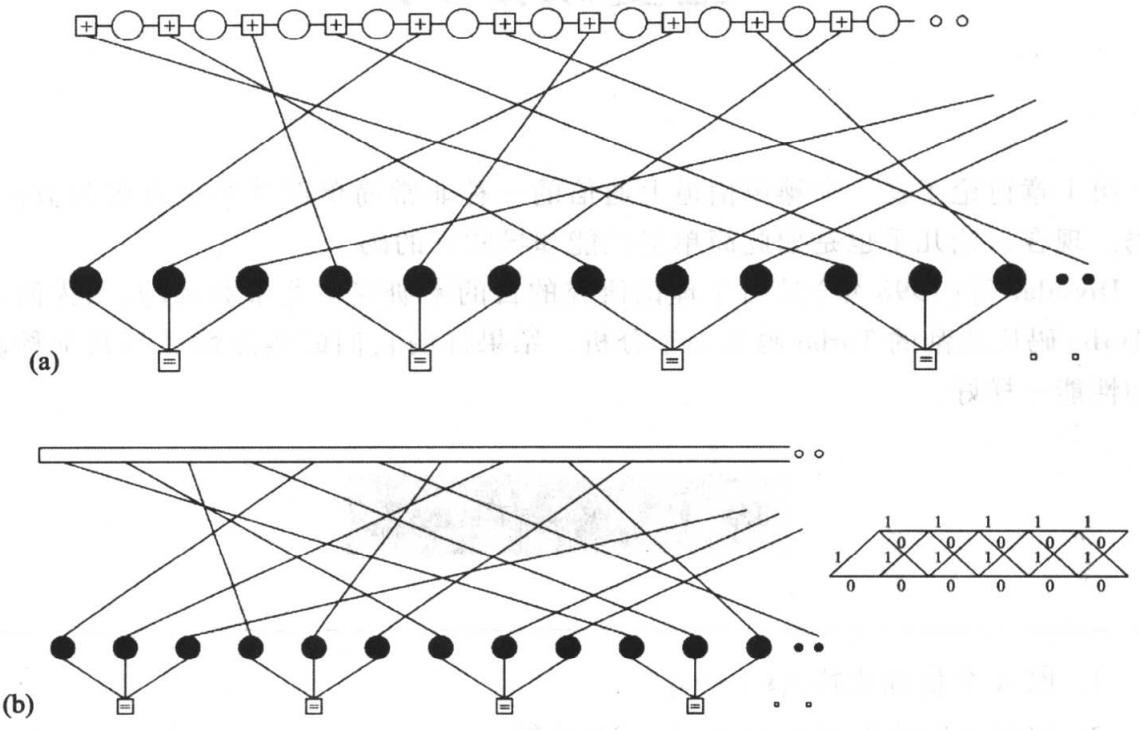


图 49.1 码率为 1/3 的一个重复累加码的因子图。(a) 使用基本结点。每个白色圆圈代表一个发送比特。每个⊡约束强制它所连接的 3 个比特之和为偶数。每个黑色圆圈代表一个中间二进制变量。每个⊡约束强制它所连接的 3 个变量是相等的。(b) 常用于译码的因子图。顶部矩形代表累加器的网格图,如插图所示。

信源在底部设置黑色比特的值,一次 3 个,累加器在顶部计算发送比特。

这个图是码字的先验概率的一个因子图,其中圆圈是二进制变量结点,方形代表两类因子结点。如通常一样,每个⊡给出一个形如$\mathbb{1}\left[\sum x = 0 \bmod 2\right]$的因子;每个⊡给出一个形如$\mathbb{1}\left[x_1 = x_2 = x_3\right]$的因子。

49.3 译 码

一般对图 49.1(b)所描述的因子图运用和 - 积算法,来对重复累加码进行译码。上面的盒子表示累加器的网格图,包括信道似然。在每次迭代的前半部分,上面的网格图接收网格图中的每次变化的似然,并运行前向 - 后向算法以便产生每个变量结点的似然。在迭代的后半部分,这些似然在⊡结点上相乘,产生新的似然信息,并送回网格图中。

和 Gallager 码和 Turbo 码一样，也可使用“完成了就停止”的译码方法，所以能区分未检测到的错误（由码中的低权重码字产生）和检测到的错误（当译码器被卡住并知道它未能成功地找到有效答案）。

如图 49.2 所示为高斯信道上 6 个随机构造的重复累加码的性能。如果并不在意分组差错概率为 10^{-4} 时所出现的差错平台，对于一个如此简单的码而言，性能也算好得惊人了（比较图 47.17）。

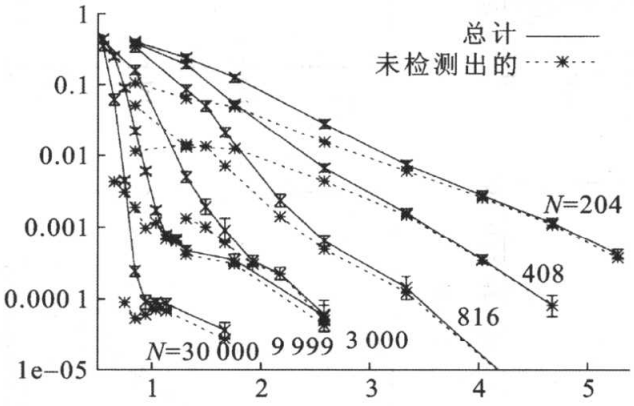


图 49.2 6 个码率为 1/3 的重复累加码在高斯信道上的性能。分组长度范围从 $N=204$ 到 $N=30\,000$。纵轴：分组差错概率；横轴：E_b/N_0。虚线表示未检测出差错的频率。

49.4 译码时间的经验分布

研究一个稀疏图码的译码所要求的和 - 积算法的迭代次数 τ 是很有意思的。已知一个码和一组信道条件，译码时间随每次试验而随机变化。可以发现译码时间的柱状图对于大的 τ，服从一个幂律，即 $P(\tau) \propto \tau^{-p}$。幂次 p 取决于信噪比并随着信噪比的下降而变小（所以分布会有更大的拖尾）。已经观察到重复累加码以及不规则和规则 Gallager 码的幂律。如图 49.3（ii）和（iii）所示为在两个不同的信噪比条件下重复累加码的译码时间分布。幂律在几个数量级上延续。

习题 49.1[$\mathcal{A}$] 研究这些幂律。密度演化有没有预测到它们？一个码的设计是否能以一种有用的方式来调整幂律？

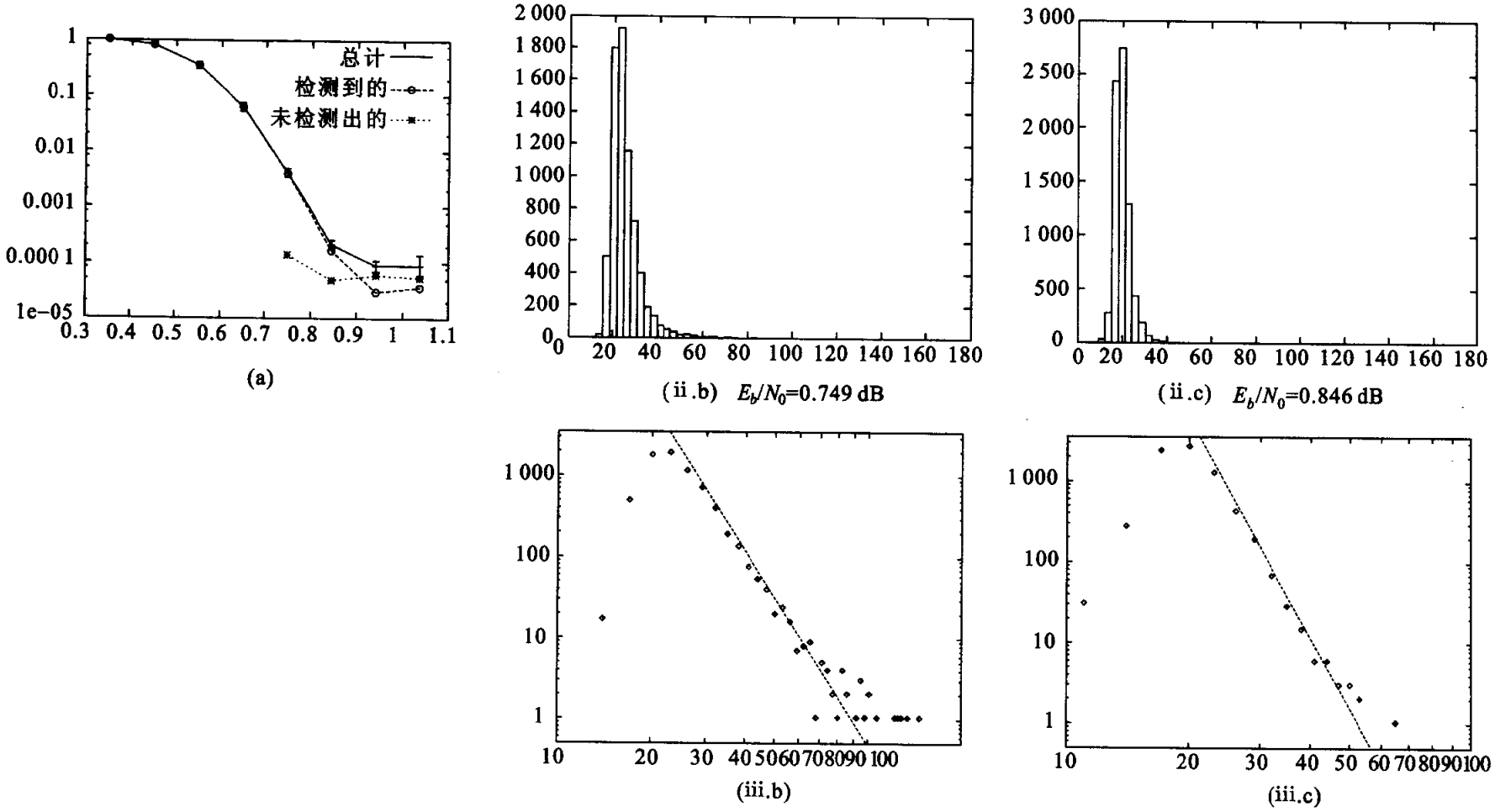


图 49.3 寻找信源分组长度为 $K=10\,000$ 和发送分组长度为 $N=30\,000$ 的重复累加码的一个有效译码所需迭代次数的柱状图。(a) RA 码的分组差错概率与信噪比的关系图。(ii.b) $x/\sigma=0.89$ 和 $E_b/N_0=0.749$ dB 时的柱状图。(ii.c) $x/\sigma=0.90$ 和 $E_b/N_0=0.846$ dB 时的柱状图。(iii.b,iii.c) 幂律的拟合:(ii.b) $(1/\tau^6)$ 和(ii.c) $(1/\tau^9)$。

49.5 广义奇偶校验矩阵

笔者发现将稀疏图码彼此关联在一起并通过一种通用的表示来描述它们是很有帮助的。Forney(2001 年)提出了一种标准图的思想,其中只有⊞和⊟结点,并且所有变量结点的阶数为 1 或 2;阶数为 2 的变量结点可以用将一个⊞结点连接到一个⊟结点的边线来表示。广义奇偶校验矩阵是标准图的一种图形表示法。在一个奇偶校验矩阵中,列是发送比特,行是线性约束条件。在一个广义奇偶校验矩阵中,可以包括附加的列,它代表了未被传输的状态变量。考虑这些状态变量的一种方法是在传输之前对码进行删除。

状态变量用相应列上的水平线标示。如在(MacKay,1999b;MacKay 等,1998 年)中,广义奇偶校验矩阵的其他示意性记号如下。

- 正方形中的对角线表示那部分矩阵包含一个单位矩阵。
- 两条或更多平行对角线表示一个带状对角矩阵,它在每行上都有相应数量的 1。
- 上面带一个箭头的水平椭圆表示一个分组中相应列进行了随机置换。
- 上面带一个箭头的垂直椭圆表示一个分组中相应行进行了随机置换。
- 用一个圆圈所围绕的一个整数表示叠加的随机置换矩阵的数量。

定义 一个广义奇偶校验矩阵是一对$\{\boldsymbol{A},\boldsymbol{p}\}$,其中$\boldsymbol{A}$是一个二进制矩阵,$\boldsymbol{p}$是一列被删除的比特。矩阵定义了一组有效向量$\boldsymbol{x}$,满足

$$\boldsymbol{A}\boldsymbol{x}=0 \tag{49.2}$$

对于每个有效向量,存在从$\boldsymbol{x}$中将$\boldsymbol{p}$所指示的比特删除而得到的一个码字$\boldsymbol{t}(\boldsymbol{x})$。对于任意一个码,有很多广义奇偶校验矩阵。

具有广义奇偶校验矩阵$\{\boldsymbol{A},\boldsymbol{p}\}$的一个码的码率可以通过如下方法估计。如果$\boldsymbol{A}$是$L\times M'$,且$\boldsymbol{p}$删除$S$比特并选择$N$比特发送($L=N+S$),则码字约束条件的有效数$M$为

$$M=M'-S \tag{49.3}$$

信源比特数为

$$K=N-M=L-M' \tag{49.4}$$

且码率大于或等于

$$R=1-\frac{M}{N}=1-\frac{M'-S}{L-S} \tag{49.5}$$

例子

重复码。一个简单的码率为 1/3 的重复码的生成矩阵、奇偶校验矩阵和广义奇偶校验矩阵如图 49.4 所示。

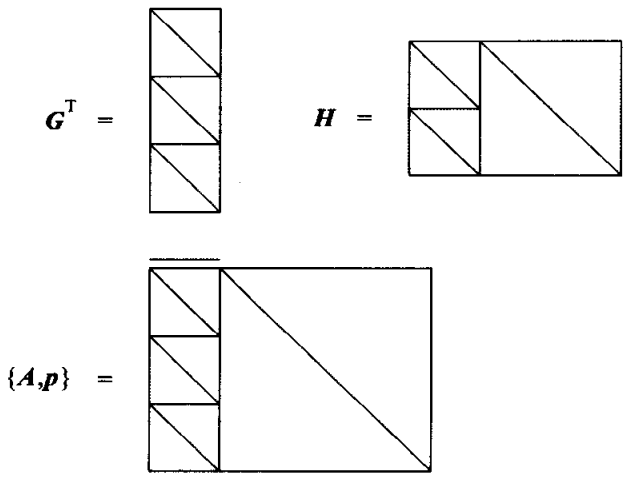


图 49.4 码率为 1/3 的重复码的生成矩阵、奇偶校验矩阵和广义奇偶校验矩阵。

系统化低密度生成矩阵码。在一个(N,K)系统化低密度生成矩阵码中，不存在状态变量。一个长度为 N 的发送码字 t 由下式给出

$$t = G^T s \tag{49.6}$$

其中

$$G^T = \begin{bmatrix} I_K \\ P \end{bmatrix} \tag{49.7}$$

I_K表示 $K \times K$ 单位矩阵，P 是一个非常稀疏的 $M \times K$ 矩阵，其中 $M = N - K$。这个码的奇偶校验矩阵为

$$H = [P \mid I_M] \tag{49.8}$$

对于码率为 1/3 的码，这个奇偶校验矩阵也许可以表示成如图 49.5 所示形式。

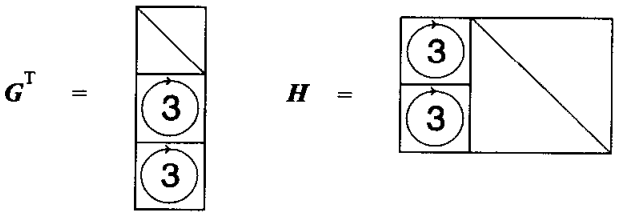


图 49.5 一个系统化低密度生成矩阵码的生成矩阵和奇偶校验矩阵。该码的码率为 1/3。

非系统化低密度生成矩阵码。在一个(N,K)非系统化低密度生成矩阵码中，一个长度为N的发送码字t由下式给出

$$t = G^{\mathrm{T}}s \tag{49.9}$$

其中G^{T}是一个非常稀疏的$N \times K$矩阵。这个码的广义奇偶校验矩阵为

$$A = [\overline{G^{\mathrm{T}}} \mid I_N] \tag{49.10}$$

且相应的广义奇偶校验方程为

$$Ax = 0, \quad \text{其中 } x = \begin{bmatrix} s \\ t \end{bmatrix} \tag{49.11}$$

尽管这个简单码的奇偶校验矩阵通常是一个复杂的稠密矩阵，其广义奇偶校验矩阵依旧保持着这个码本质上的简单性。

对于码率为1/2的情况，这个广义奇偶校验矩阵也许可以表示成如图49.6所示的形式。

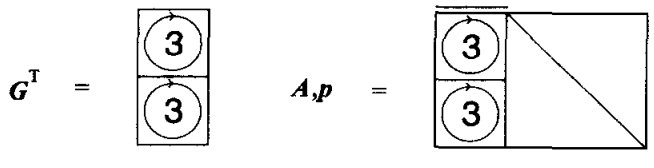


图49.6 一个非系统化的低密度生成矩阵码的生成矩阵和广义奇偶校验矩阵。该码的码率为1/2。

低密度奇偶校验码与线性MN码。一个码率为1/3的低密度奇偶校验码的奇偶校验矩阵如图49.7(a)所示。

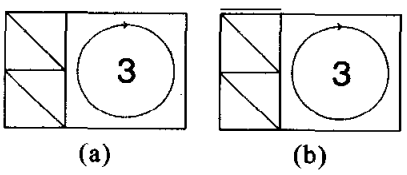
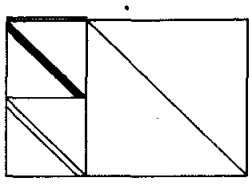


图49.7 (a) 一个码率为1/3的有$M/2$列权重为2的Gallager码的广义奇偶校验矩阵；(b) 一个码率为1/2的线性MN码的广义奇偶校验矩阵。

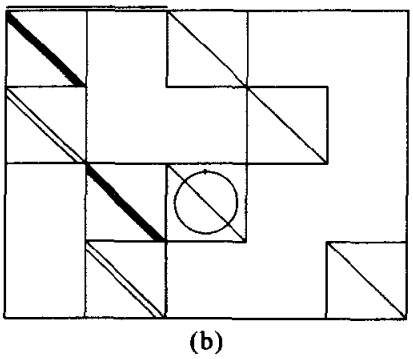
一个线性MN码是一个非系统化的低密度奇偶校验码。一个MN码中的K个状态比特是信源比特。如图49.7(b)所示是一个码率为1/2的线性MN码的广义奇偶校验矩阵。

卷积码。在一个非系统化的、非递归的卷积码中，作为状态比特的信源比特被送入一个延时线，并发送延时线的两个线性函数。在图49.8(a)中，这两个奇偶比特流被表示成两个长度为K的相邻向量。[通常将这两个奇偶比特流进行交织；因为这里与比特重排无关，所以就没有在图上画出。]

级联。两个码的“并行级联”用其中一个示意图表示，将两个码的矩阵对齐使得



(a)



(b)

图 49.8 (a) 一个码率为 1/2 的卷积码的广义奇偶校验矩阵。(b) 由两个卷积码并行级联而构成的一个码率为 1/3 的 Turbo 码的广义奇偶校验矩阵。

"信源比特"排成一排,并在矩阵中加入零元素块,使得两个码的状态比特和奇偶比特占据不同的列。下面的 Turbo 码给出了一个例子。在"串行级联"中,与第一个码的发送比特相对应的列和第二个码的信源比特所对应的列对齐。

Turbo 码。一个 Turbo 码是两个卷积码的并行级联。一个码率为 1/3 的 Turbo 码的广义奇偶校验矩阵如图 49.8(b)所示。

重复累加码。一个码率为 1/3 的重复累加码的广义奇偶校验矩阵如图 49.9 所示。重复累加码等价于阶梯码(第 47.7 节)。

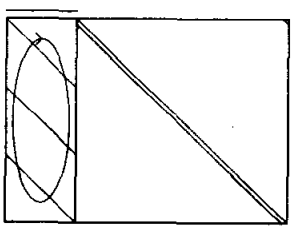


图 49.9 一个码率为 1/3 的重复累加码的广义奇偶校验矩阵。

交叉。两个码的交叉所对应的广义奇偶校验矩阵是这样构成的:将它们的广义奇偶校验矩阵放置在一起,使得发送比特的所有列正确地对齐,而与这两个分量码有关的任意删除比特占据不同的列。

关于第 50 章

以下习题为数字喷泉码提供一个很有帮助的背景。

▷ **习题 50.1**[𝒻] 一位作者通过检阅随机书页来校对他的一本厚达 $K = 700$ 页的书。他做了 N 次书页检阅，但未采取任何预防措施来避免重复检阅同一页。

（a）在进行 $N = K$ 次书页检阅之后，从未被检查过的书页占多大比重？

（b）在 $N > K$ 次书页检阅之后，一页或多页从未被检查过的概率是多少？

（c）证明为了保证所有 K 页已经被检阅的概率为 $1 - \delta$，要求 $N \simeq K\ln(K/\delta)$ 次书页检查。

[这个问题通常被表述为将 N 个球随机地扔进 K 个橱柜；每个橱柜至少得到一个球的概率是多少？]

第 50 章

数字喷泉码

数字喷泉码是在删除信道中打破编码记录的稀疏图码。

删除信道非常重要。例如，在因特网上所传送的文件被划分为分组，并且每个分组要么是接收无误要么就是没有收到。描述这种情况的一个简单信道模型是一个 q 进制删除信道：对于输入字符集 $\{0,1,2,\cdots,q-1\}$ 中的所有输入，它的无误传输概率为 $1-f$，传送输出“?”的概率为 f。字符集的大小 q 等于 2^l，其中 l 是一个分组中的比特数。

在这种信道上通信的常用方法是采用从接收端到发送端的一个反馈信道来控制被删除分组的重传。例如，接收端可能发送返回信息来指明被丢失的分组，于是对它们进行重传。或者，接收端也许返回信息以确认每个接收到的分组；发送端知道哪个分组被确认并重传其他分组，直至所有分组都被确认。

这些简单的重传协议具有这样的优点：无论删除概率 f 为何值，它们都管用，但是那些学习过香农理论的纯理论者会觉得这些重传协议浪费资源。如果删除概率 f 很大，那么第一个协议要发送大量反馈信息。在第二个协议下，接收端最终可能会接收某些分组的多重冗余副本，导致反馈信道的负荷过重。根据香农定理，不需要反馈信道：无论是否有反馈，前向信道的容量是 $(1-f)l$ 比特。

简单重传协议的浪费在一个带删除的广播信道中显得格外突出——在这种信道中，一个发送端对多个接收端广播，而每个接收端接收分组的随机 $(1-f)$ 部分。如果一个或多个接收端所丢失的每个分组都要被重传，那么这些重传会导致极高的信息冗余。每个接收端都会重复接收大多数的重传分组。

所以，人们将乐于使用删除纠错码，它们不需要反馈或几乎不需要反馈。删除纠错的经典分组码称为 Reed - Solomon 码。一个 (N,K) Reed - Solomon 码（在一个大小为 $q=2^l$ 的字符集上）具有理想的特性：如果 N 个发送符号中的任意 K 个被接收，那么可以恢复原来的 K 个信源符号。[进一步的信息请参见 Berlekamp（1968 年）或 Lin 和 Costello（1983 年）；在 $N<q$ 时存在 Reed - Solomon 码。]但是 Reed - Solomon 码也存在缺点：它们只是在 K、N 和 q 较小时才是实用的：编码和译码的标准实现的代价量级为 $K(N-K)\log_2 N$ 次分组操作。此外，对于一个 Reed - Solomon 码，就像对任意分组码一样，必须估计删除概率 f 并在发送之前选择码率 $R=K/N$。如果 f 不巧比预期的要大，则接收端接收到少于 K 个符号，应该怎么办？希望使用一种简单方法来实时地扩展码，以产生一个更低速率 (N',K) 的码。对于 Reed - Solomon 码，不存在这种实时方法。

有一种更好的方法，由 Michael Luby(2002 年)首创于他所在的数字喷泉公司，这是第一家基于稀疏图码业务的公司。

这里所描述的数字喷泉码，LT 码，[①]是 Luby 在 1998 年发明的。一个数字喷泉码的基本思想如下。编码器是一个产生无穷尽的水滴供给(编码后的分组)的喷泉；假设原信源文件的大小为 Kl 比特，每个“水滴”包含 l 个已编码比特。现在，任何希望接收编码文件的人端一个桶到喷泉下收集水滴，直至桶中的水滴数稍大于 K。于是他们就可以恢复原文件。

信源信息所能产生的编码分组数可以是无限的，从这个角度来说，数字喷泉码是*无码率*的；并且所产生的编码分组数可以实时地确定。不管信道上的删除事件的统计特性如何，可以根据需要发送足够多的编码分组，以便译码器能恢复信源数据。当 K'比 K 略微大一些时(实际中，约大 5%)，信源数据可以从任意 K'个编码分组的集合中被译码。

数字喷泉码还很奇妙地具有低的编码和译码复杂度。K 个分组可以以概率 $1-\delta$ 进行通信，其平均编码和译码代价的量级都是 $K\ln(K/\delta)$次分组操作。

Luby 称这些码为*通用的*，因为对于任意删除信道，它们同时接近最优，并且当文件长度 K 增加时，它们会变得非常高效。额外开销 $K'-K$ 在$\sqrt{K}(\ln(K/\sigma))^2$这一量级。

50.1 一个数字喷泉码的编码器

每个编码分组 t_n是从信源文件 $s_1s_2s_3\cdots s_K$按如下方式产生的。

1. 随机地从一个阶数分布 $\rho(d)$中选择分组的阶数 d_n；正如后面将要讨论的那样，ρ 的正确选择取决于信源文件的大小 K。

2. 均匀地随机选择 d_n个不同的输入分组，并将 t_n值设定为这 d_n个分组按位进行的模 2 和。可以通过连续地对分组进行异或得出这个和。

这个编码操作定义了将编码分组与信源分组联系在一起的图。如果平均阶数$\bar{d}$比 K 要小得多，那么该图是稀疏的。可以将所得码视为一个不规则低密度生成矩阵码。

译码器需要了解所接收到的各个分组的阶数，及它在图中与哪个信源分组相连。这个信息可以通过不同方法发送到译码器。例如，如果发送端和接收端具有同步化时钟，它们可以使用相同的伪随机数发生器，由时钟提供种子，来选择每个随机阶数

① LT 表示“Luby 变换”

和每个连接集合。或者,发送端可以使用一个随机密钥 κ_n,假定这个密钥、阶数和连接可由一个伪随机过程确定,并且在分组头中发送那个密钥。只要分组大小 l 比密钥的大小大得多(密钥仅仅需要大约 32 比特),这个密钥只引入了一个很小的额外开销。

50.2 译 码 器

在一个删除信道的情形,很容易对一个稀疏图码进行译码。译码器的任务是从 $\boldsymbol{t}=\boldsymbol{G}\boldsymbol{s}$ 中恢复 $\boldsymbol{s}$,其中 $\boldsymbol{G}$ 是与图相关联的矩阵。解决这个问题的一种简单方法是通过消息传递。如果愿意,可以将译码算法视为和 - 积算法,但是所有信息要么是完全不确定的消息要么是完全确定的消息。不确定消息表明一个消息分组 s_k 可能以等概率为任意值;确定消息表明 s_k 以概率 1 具有某一特殊值。

消息的简洁性允许人们对译码过程进行一个简单描述。称编码分组 $\{t_n\}$ 为校验结点。

1. 寻找一个仅与一个信源分组 s_k 相连的校验结点 t_n。(如果不存在这样的校验结点,这个译码算法将在这一点中止,并且不能恢复所有信源分组。)

(a) 设 $s_k = t_n$。

(b) 将 s_k 加到所有与 s_k 相连的校验 $t_{n'}$ 上:

$$\text{对于所有满足 } G_{n'k}=1 \text{ 的 } n', t_{n'} := t_{n'} + s_k \tag{50.1}$$

(c) 删除与信源分组 s_k 相连的所有边线。

2. 重复步骤(1),直至确定所有 $\{s_k\}$。

对于每个分组只有 1 比特这种极简单的情形,译码过程如图 50.1 所示。存在 3 个信源分组(由上面的 3 个圆圈表示)和 4 个接收分组(由下面的校验符号表示),在算法开始时,它们的值为 $t_1t_2t_3t_4 = 1011$。

在第一次迭代时,与一个信源比特相连的唯一校验结点是第一个校验结点(窗口 a)。相应地设置信源比特 s_1(窗口 b),丢弃校验结点,然后将 s_1 的值(1)加到与之相连的校验结点上(窗口 c),并将 s_1 与图断开。在第二次迭代开始时(窗口 c),第 4 个校验结点与单个信源比特 s_2 相连。将 t_4 设定为 s_2 的值(为 0,在窗口 d 中),并将 s_2 加到与之相连的两个校验上(窗口 e)。最后,可以发现两个校验结点都连接到 s_3,并且它们所给的 s_3 值是一致的(正如我们所希望!),在窗口 f 中恢复该值。

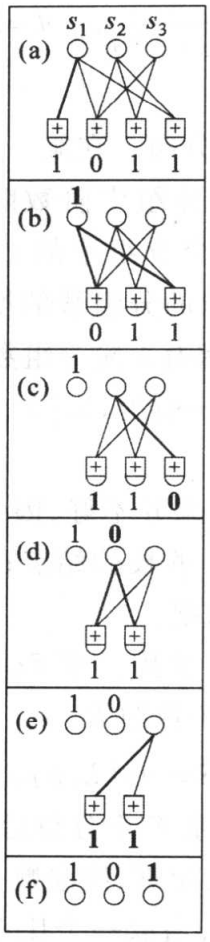


图 50.1 $K=3$ 个信源比特、$N=4$ 个编码比特的一个数字喷泉码的译码例子

50.3 设计阶数分布

阶数的概率分布 $\rho(d)$ 是设计的关键部分：偶尔使用的编码分组必须具有高的阶数（即 d 与 K 相当），以保证不存在信源分组所连接的不是 1 的情况。很多分组必须具有低的阶数，这样译码操作才可以开始，并继续运行，同时使得编码和译码中所包含的加法运算总数较少。对于一个已知阶数分布 $\rho(d)$，可以使用一个恰当形式的密度演化来预测译码过程的统计特性。

理想地，为了避免冗余，希望接收图具有这样的性质：在每次迭代时，只有一个校验结点具有阶数 1。在每次迭代时，在处理这个校验结点时，以特定方式减少图中的阶数，使之出现一个阶数为 1 的新校验结点。期望通过理想的孤立子分布实现理想的特性

$$\rho(1)=1/K$$
$$\rho(d)=\frac{1}{d(d-1)},\quad d=2,3,\cdots,K \tag{50.2}$$

在这个分布下，期望阶数约为 $\ln K$。

▷ **习题 50.2**[z] 推导理想孤立子分布。在第一次迭代($t=0$)时，设 d 阶的分组个数为$h_0(d)$；证明(对于 $d>1$) d 阶分组中阶数降为 $d-1$ 阶的分组数的期望值为 $h_0(d)d/K$；在第 t 次迭代时，当 K 个分组中的 t 个已被恢复且 d 阶分组的个数为 $h_t(d)$时，d 阶的分组数降为 $d-1$ 阶的分组数的期望值为 $h_t(d)d/K-t$。由此证明，为了使对于所有 $t\in\{0,\cdots,K-1\}$ 都有 1 阶分组数的期望值满足 $h_t(1)=1$，必须起始于$h_0(1)=1$和 $h_0(2)=K/2$；且更一般地，$h_t(2)=(K-t)/2$；然后对 $d=3$ 以上的 $h_0(d)$，采用递归法求解。

这个阶数分布在实际中应用效果并不好，因为围绕期望特性的起伏变动使得在译码过程中很可能不存在阶数为 1 的校验结点；此外，部分信源结点会完全没有连接。一个小的改动可以解决这些问题。

鲁棒孤立子分布具有两个额外参数，c 和 δ；对它进行设计以保证在整个译码过程中，阶数为 1 的校验结点的期望个数约为

$$S\equiv c\ln(K/\delta)\sqrt{K} \tag{50.3}$$

而不是 1。参数 δ 是接收到一定数量 K'个分组后译码失败的概率的一个界限。如果目的是证明 Luby 关于 LT 码的主要定理，则参数 c 是在 1 这个数量级的一个常数；而在实际中，可以将它视为自由参数，当取一个比 1 小一些的值时，会给出好的结果。定义一个正函数

$$\tau(d)=\begin{cases}\dfrac{S}{K}\dfrac{1}{d}, & d=1,2,\cdots,(K/S)-1\\ \dfrac{S}{K}\ln(S/\delta), & d=K/S\\ 0, & d>K/S\end{cases} \tag{50.4}$$

(参见图 50.2 和习题 50.4)，然后将理想的孤立子分布 ρ 加到 τ 上，并归一化以获得鲁棒的孤立子分布 μ

$$\mu(d)=\frac{\rho(d)+\tau(d)}{Z} \tag{50.5}$$

其中 $Z=\sum_d \rho(d)+\tau(d)$。接收端保证译码运行完成的概率不低于 $1-\delta$ 时所需的编码分组个数为 $K'=KZ$。

Luby(2002 年)的分析解释了 τ 后面的小 d 端可以保证译码过程开始，而 τ 在 $d=K/S$处所包含的峰值保证每个信源分组都可能与一个校验点有至少一次连接。Luby 的关键结果是当常数 c 取合适的值时，接收 $K'=K+2\ln(S/\delta)S$ 个校验就可以确保以不小于 $1-\delta$ 的概率恢复所有的分组。在图 50.3 中，笔者设置了相当大的译

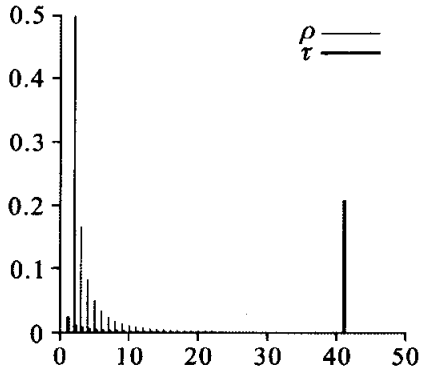


图 50.2 $K = 10\ 000, c = 0.2, \delta = 0.05$ 情形下的分布 $\rho(d)$ 和 $\tau(d)$，此时有 $S = 244, K/S = 41$ 且 $Z \simeq 1.3$。分布 τ 在 $d = 1$ 和 $d = K/S$ 处最大。

码失败允许概率 δ，因为实际的失败概率比 Luby 的保守分析所建议的概率要小得多。

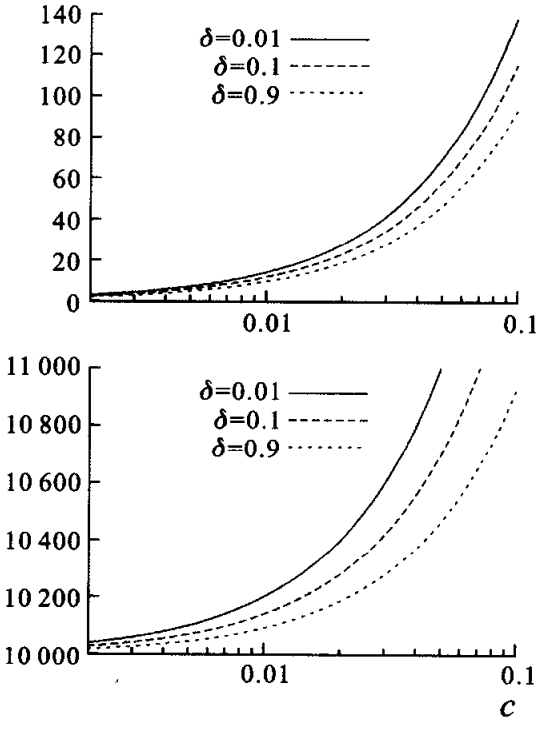


图 50.3 对于 $K = 10\ 000$，阶数为 1 的校验点的个数 S（上半图）和量 K'（下半图），以两个参数 c 和 δ 的函数表示。Luby 的主要定理证明存在一个 c 值，使得在已知 K' 个接收分组时，译码算法将以 $1 - \delta$ 的概率恢复 K 个信源分组。

在实际中，可以调解 LT 码，使得恢复一个原大小为 $K \simeq 10\ 000$ 个分组的文件所需的额外开销大致为 5%。图 50.4 表示两种参数设置分别达到平均小于 5% 和 10%

的开销时，所需要的实际分组个数的柱状图。

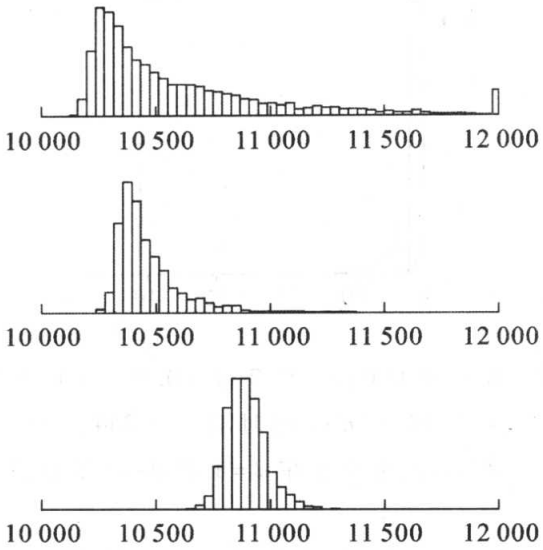


图 50.4 要恢复一个大小为 $K = 10\ 000$ 个分组的文件，所需的实际分组个数 N 的柱状图。参数如下：
上面的柱状图：$c = 0.01, \delta = 0.5$（$S = 10, K/S = 1010$ 和 $Z \simeq 1.01$）；
中间的柱状图：$c = 0.03, \delta = 0.5$（$S = 30, K/S = 337$ 和 $Z \simeq 1.03$）；
下面的柱状图：$c = 0.1, \delta = 0.5$（$S = 99, K/S = 101$ 和 $Z \simeq 1.1$）。

50.4 应 用

数字喷泉码在很多不同的情况下都是一个优秀的解决方案。现在来讨论两个。

存储

你希望能备份一个大文件，但是你意识到你的磁盘和硬盘驱动器可能都是不可靠的，因为每天都以大约 10^{-3} 的几率发生灾难性的故障，而此时某些被存储的分组会永久性地在一个设备中丢失。试问应该如何存储你的文件？

一个数字喷泉码可用于将编码后的分组发送到所有地方，在每个可用的存储设备上。然后要恢复大小为 K 个分组的备份文件，只需从随便什么地方找到 $K' \simeq K$ 个分组就可以了。碰到被破坏的分组也没关系；可以简单地跳过它们并在其他地方寻找更多的分组。

在文件的恢复速度方面，这种存储方法同样具有优势。在一个硬盘驱动器中，通常的做法是在一个硬盘驱动器的连续扇区中存储一个文件，这样允许快速的文件读取；但是，如果发生意外，一个分组丢失了（因为读取磁头离开磁轨一会儿，就会给出一串突发的错误，且不能用分组的纠错码来纠正），磁盘必须转一整圈到磁头进行第

二次读取时，才能恢复分组。旋转一周所花费的时间会给文件系统带来不理想的延迟。

如果文件改为用数字喷泉原理来存储，将数字“水滴”存储在一个或多个连续的硬盘扇区，则不必再忍受重读分组所带来的延迟；分组的丢失变得无关紧要，且硬盘驱动器也可以因此工作得更快，容许更高的噪声水平，用于噪声信道编码的资源也更少了。

▷ **习题 50.3**[ℯ] 比较多硬盘驱动器可靠存储的数字喷泉方法和 RAID（独立磁盘的冗余阵列）。

广播

假设在一个地区有一万个用户希望从一个广播设备中接收一部数字电影。广播设备可以通过广播网络将电影以分组的形式发送，例如使用宽带电话线或通过卫星。

假设并非所有家庭都接收到所有分组。比如每家丢失其中的 $f=0.1\%$。在标准方法中，文件以一个普通的未编码分组序列进行发送，每个家庭会通知广播设备丢失了 fK 个分组，并要求重传它们。当一万个用户都要求这样的重传，那么几乎每个分组都有一个重传请求。因此广播设备必须重复整个广播大约两次，以确保大多数用户接收到整部电影，并且很多用户需要等待大约两倍于理想的时间才能完成下载。

如果广播设备使用一个数字喷泉码对电影进行编码，每个用户可以从任意 $K' \simeq K$ 个分组中恢复电影。所以广播仅需持续较短的时间，比如 1.1 K 个分组的时间，并且每个家庭都非常有可能成功地恢复整个文件。

另一个应用是向汽车广播数据。假设要通过卫星来更新车载导航数据库。有成千上万辆汽车，并且它们只能在驶上开阔路段才可以接收数据；不存在反馈信道。发送数据的一种通常方法是将数据放入一个转盘式磁带存储器，以固定的周期序列广播分组。“是的，一辆汽车也许穿过一条隧道并错过几百个分组，但是它可以在一小时后，当转盘已经旋转一周（我们希望）时收集那些丢失的分组；或许是第二天……”

如果卫星改为使用一个数字喷泉码，那么每辆汽车仅需接收与原始文件大小相等的数据（再加 5%）即可。

进一步的阅读

数字喷泉公司出售的编码器和译码器与这里所描述的 LT 码相比，具有更高的效率，并且它们在各种分组长度下都能很好地工作，而不仅仅在诸如 $K \geq 10\,000$ 之类的大长度情况下才好用。Shokrollahi（2003 年）提出了 Raptor 码，这是具有线性时间编码和译码的 LT 码的一个扩展。

50.5 进一步的习题

▷**习题 50.4**[2] 理解鲁棒的孤立子分布。

重复习题 50.2 的分析,但现在的目标是让阶数为 1 的分组的期望个数对所有 t 都是 $h_t(1) = 1 + S$,而非 1。证明最初需要的分组个数为

$$h_0(d) = \frac{K}{d(d-1)} + \frac{S}{d}, \quad d > 1 \tag{50.6}$$

将第二项在 $d = K/S$ 后截短并将其替换为 $d = K/S$ 处的峰值信号(参见式(50.4))的原因是,确保译码复杂度不会增长到超过 $O(K\ln K)$。

如果分组的柱状图如式(50.6)所给,估计分组的期望个数 $\sum_d h_0(d)$ 和稀疏图中边线的期望条数 $\sum_d h_0(d)d$(它决定了译码复杂度)。当使用鲁棒的孤立子分布(50.4)时,比较分组和边线的期望数。

习题 50.5[1] 证明在 $d = K/S$ 处的峰值(式(50.4))是对式(50.6)中的高权重分组的拖尾的合适替换。

习题 50.6[3C] 用实验方法来研究在 $d = K/S$ 处的峰值信号(式(50.4))对于成功的译码有多么必要。同时研究 $\rho(d)$ 在 $d = K/S$ 之后的拖尾是否必需。如果去掉所有高权重的阶项,包括 $d = K/S$ 处的峰值信号和 $\rho(d)$ 在 $d = K/S$ 之后的拖尾,会出现什么情况?

习题 50.7[1] 对于 Luby 主要定理:即接收到 $K' = K + 2\ln(S/\delta)S$ 个校验,就可以保证所有信源分组被恢复的概率不小于 $1 - \delta$,补充详细的证明。

习题 50.8[2C] 对于包含 $K = 10\,000$ 个分组的文件,优化一个数字喷泉码的阶数分布。为你的优化选择一个合理的目标函数,比如最小化完全译码所需的分组数 N 的均值,或 N 的柱状图的第 95 个百分分布的值(图 50.4)。

▷**习题 50.9**[3] 为广播一个数据流到汽车的情形建模,量化数字喷泉与转盘磁带方法相比所具有的优势。

习题 50.10[2] 构建一个简单的例子来说明第 50.2 节中的数字喷泉译码器是次优的——有时即使可用的信息足够来译码整个文件时,它也会放弃。与之相比,最优译码器的代价如何?

▷**习题 50.11**[2] 如果每个发送分组都是通过将信源分组随机地加在一起而产生的,其中每个信源分组被包含的概率为 1/2,证明接收到 $K' = K$ 个分组就足以使最优译码器恢复 K 个信源分组的概率略小于 1/2。[换言之,一个大小为 $K \times K$ 的随机矩阵是满秩矩阵的概率是多少?]

证明:如果接收到 $K' = K + \Delta$ 个分组还不足以使最优译码器满足要求的概率约为 $2^{-\Delta}$。

▷**习题 50.12**[比C] Richardson 和 Urbanke(2001b)为稀疏图码(第 47.7 节)的快速编码推导出一种方法,来处理最优译码所要求的矩阵转换。试用这种方法来实现一个最优数字喷泉译码器。既然你已经改变了译码器,就可以使用具有更大权重的分组来重新优化阶数分布。你还可以降低多少额外开销?证明这种方法可以使数字喷泉码适合作为具有各种分组长度的删除纠错码,而不仅仅适于大分组长度(此时 LT 码也是优秀的)。

▷**习题 50.13**[5] 数字喷泉码是删除信道上优秀的无码率的码。为一个同时具有删除和噪声的信道构造一个无码率的码。

50.6 稀疏图码小结

为噪声信道设计纠错码的一种简单方法最早由 Gallager(1962 年)提出,最近又被重新发现和推广,并且已经改变了通信理论。Gallager 的低密度奇偶校验码及其同类码的实际性能已经大大优于此前一些年充斥于教材中的码。

在一个噪声信道中,哪种稀疏图码是“最佳”的,取决于所选择的码率和分组长度、所允许的编码和译码复杂度以及是否可以接受偶尔有未被检测出来的错误等因素。低密度奇偶校验码是最通用的;很容易构造一个有竞争力的几乎可以是任何速率和分组长度的低密度奇偶校验码,并且低密度奇偶校验码基本上不会产生检测不到的错误。

对于删除信道这一特例,最佳稀疏图码是数字喷泉码。

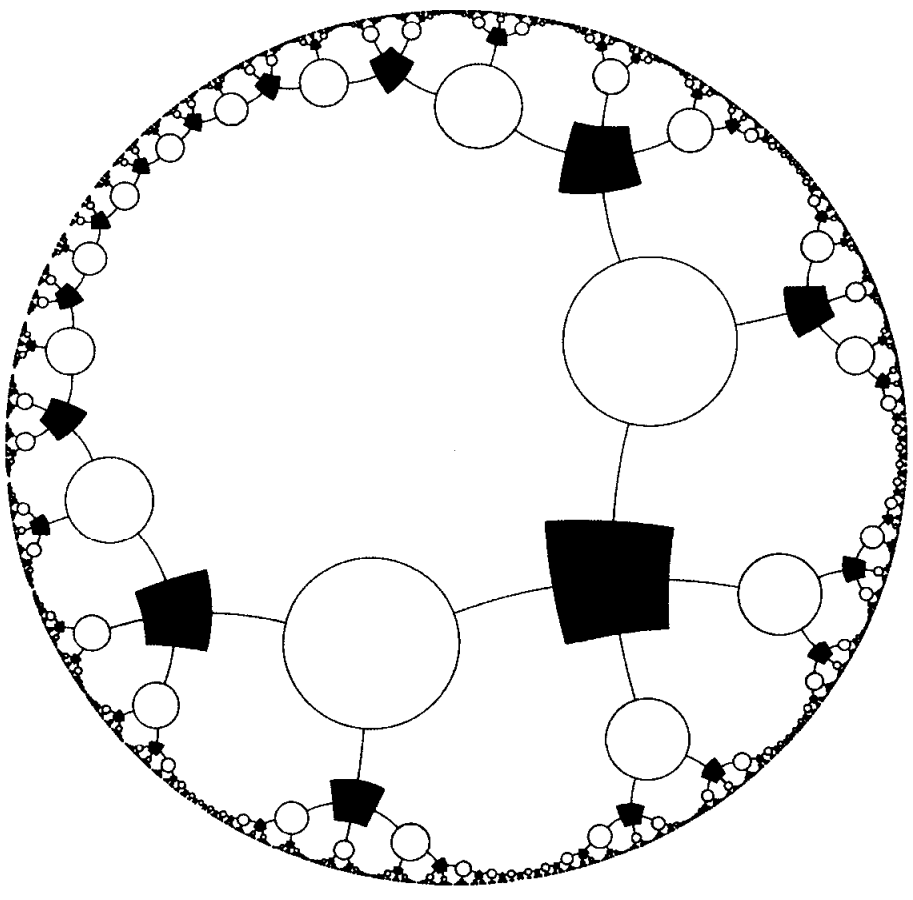
50.7 结　论

通信问题的最佳解决方案是:

> 将一个简单的伪随机码同一个消息传递译码器相结合。

第七部分

附　录



附录 A 记　　号

$P(A|B,C)$表示什么意思？ $P(A|B,C)$读做“已知 B 为真且 C 为真的前提下，A 为真的概率”。或简单地表述为“已知 B 和 C 的条件下，A 的概率”。

log 和 ln 表示什么意思？ 在本书中，$\log x$ 表示以 2 为底的对数 $\log_2 x$；$\ln x$ 表示自然对数 $\log_e x$。

$\hat{s}$ 表示什么意思？ 通常，变量上的“尖帽”表示一个猜测或一个估计值。因此，$\hat{s}$ 表示对 s 值的一个猜测。

积分。 $\int f(u)\,du$ 和 $\int du f(u)$ 二者没有区别。在两种情况下，被积函数都是 $f(u)$。

$\prod_{n=1}^{N}$ 表示什么意思？ 这与求和号 $\sum_{n=1}^{N}$ 类似，但是它表示一个乘积。它读做“n 从 1 到 N 的乘积”。因此，比如

$$\prod_{n=1}^{N} n = 1 \times 2 \times 3 \times \cdots \times N = N! = \exp\left[\sum_{n=1}^{N} \ln n\right] \tag{A.1}$$

笔者喜欢选择求和或求积中的自由变量，这里的 n 是求和范围的小写字母。所以，n 通常从 1 取到 N，m 通常从 1 取到 M。这是笔者从 Yaser Abu-Mostafa 那里学到的一个习惯，我认为这样可以使公式更容易理解。

$\binom{N}{n}$ 表示什么意思？ 读做“N 选 n”，是从一个大小为 N 的集合中无序地选择 n 个对象的方法的个数。

$$\binom{N}{n} = \frac{N!}{(N-n)!\ n!} \tag{A.2}$$

该函数叫做组合函数。

$\Gamma(x)$ 表示什么意思？ 伽马函数定义为 $\Gamma(x) \equiv \int_0^\infty du\, u^{x-1} e^{-u}, x > 0$。伽马函数是阶乘函数在实数自变量上的扩展。一般地，$\Gamma(x+1) = x\Gamma(x)$，而对整数自变量而言，$\Gamma(x+1) = x!$。Digamma 函数定义为 $\Psi(x) \equiv \frac{d}{dx}\log \Gamma(x)$。

对于大的 x 值（实际上，$0.1 \leqslant x \leqslant \infty$）有

$$\log \Gamma(x) \simeq \left(x - \frac{1}{2}\right)\log(x) - x + \frac{1}{2}\log 2\pi + O(1/x) \tag{A.3}$$

而对于小的 x 值（实际上，$0 \leqslant x \leqslant 0.5$）有

$$\log \Gamma(x) \simeq \log \frac{1}{x} - \gamma_e x + O(x^2) \tag{A.4}$$

其中 γ_e 为欧拉常数。

$H_2^{-1}(1-R/C)$ 表示什么意思？ 正如 $\sin^{-1}(s)$ 表示函数 $s = \sin(x)$ 的反函数，$H_2^{-1}(h)$ 表示函数 $h = H_2(x)$ 的反函数一样。

在人们使用 $\sin^2 x$ 表示 $(\sin x)^2$ 时，可能存在潜藏的混乱，因为人们可能会以为 $\sin^{-1}(s)$ 表示

$1/\sin(s)$;因此笔者倾向于避免使用 $\sin^2 x$。

$f'(x)$表示什么意思? 答案与上下文有关。通常,上标处的"一撇"用于表示微分

$$f'(x) \equiv \frac{\mathrm{d}}{\mathrm{d}x} f(x) \tag{A.5}$$

类似地,点号表示相对于时间 t 的微分

$$\dot{x} \equiv \frac{\mathrm{d}}{\mathrm{d}t} x \tag{A.6}$$

但是,"一撇"也是表示"另一个变量"的一个有用标号,比如"一个变量的新值"。因此,比如 x'可能表示"x 的一个新值"。而且,如果有两个从 1 到 N 的整数,笔者经常将这样的整数命名为 n 和 n'。

因此,笔者遵循的规则是:如果"一撇"出现在可能是函数的一个表达式中,比如 $f'(x)$ 或 $h'(y)$,那么它表示微分;否则,它表示"另一个变量"。

什么是误差函数? 该函数有不同的定义方法。笔者定义它为一个标准正态分布(方差 = 1)的累计概率

$$\Phi(z) \equiv \int_{-\infty}^{z} \exp(-z^2/2)/\sqrt{2\pi}\mathrm{d}z$$

$$\mathrm{d}z \tag{A.7}$$

$\mathscr{E}(r)$表示什么意思? $\mathscr{E}[r]$读做"r 的期望值"或"r 的数学期望",它是 r 的均值。"期望值"的另一种表示符号为一对尖括号$\langle r\rangle$。

$|x|$表示什么意思? 竖杠"|·|"有两种含义。如果 $\mathscr{A}$ 是一个集合,那么$|\mathscr{A}|$表示该集合中的元素个数;如果 x 是一个数,那么$|x|$表示 x 的绝对值。

$[A][P]$表示什么意思? 这里,A 和 P 是行数相同的矩阵。$[A][P]$表示将 A 放在 P 旁边所形成的双倍宽度的矩阵。其中的竖杠是为了避免和乘积 AP 混淆。

X^{T} 表示什么意思? 上标 T 读做"转置"。转置一个行向量,会将它变成列向量

$$(1,2,3)^{\mathrm{T}} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \tag{A.8}$$

反之亦然。[用黑体类型($\boldsymbol{x}$)表示向量,一般我的向量为列向量。]

类似地,也可对矩阵进行转置。如果 M_{ij}是矩阵 M 的第 i 行第 j 列元素,且 $N = M^{\mathrm{T}}$,那么 $N_{ji} = M_{ij}$。

Trace M 和 det M 表示什么意思? 一个矩阵的迹是它的对角元素之和

$$\mathrm{Trace}\, M = \sum_i M_{ii} \tag{A.9}$$

$\det M$ 表示矩阵 M 的行列式。

δ_{mn}表示什么意思? δ 矩阵是单位矩阵。

$$\delta_{mn} = \begin{cases} 1, & m = n \\ 0, & m \neq n \end{cases}$$

单位矩阵的另一个记号是 I 或 $\mathbf{1}$。有时,笔者会在这个记号中包含一个下标来表示矩阵的大小,如 I_K 表示$(K \times K)$的单位矩阵。

$\delta(x)$表示什么意思? δ 函数具有性质

$$\int \mathrm{d}x\, f(x)\delta(x) = f(0) \tag{A.10}$$

$\delta(S)$的另一个可能的含义是真值函数,在命题S为真时,它取1;但笔者用另一个记号来表示这层意思。毕竟,符号δ已经用得够频繁了,除了上面所提的两个角色,它还表示小实数δ和增量运算符(比如δx)!

$\mathbb{1}[S]$表示什么意思? $\mathbb{1}[S]$是真值函数,在命题S为真时,它取1,否则取0。比如,集合$T=\{-2,1,3\}$中的正数个数可写成

$$\sum_{x\in T}\mathbb{1}[x>0] \tag{A.11}$$

“:=”和“=”之间的区别是什么? 在算法中,$x:=y$表示将y值赋给变量x以更新其值。

与此不同,$x=y$是一个命题,表示x等于y。

有关概率分布的更多定义和记号,请参看第23章和第29章。

附录 B　一些物理知识

B.1　关于相变

与温度为 $T=1/\beta$ 的热容器接触的系统处于状态 x，其概率分布为

$$P(x|\beta)=\frac{1}{Z(\beta)}\exp(-\beta E(x)) \tag{B.1}$$

划分函数为

$$Z(\beta)=\sum_{x}\exp(-\beta E(x)) \tag{B.2}$$

逆温度 β 可以解释为定义了熵和能量之间的交换速率。$(1/\beta)$ 是热容器要增加 1 Nat 的熵所必须获得的能量。

通常，系统还会受其他一些参数的影响，比如它所处盒子的体积 V。在这种情况下，Z 也是 V 的一个函数，$Z(\beta,V)$。

对于任意具有有限个状态的系统，函数 $Z(\beta)$ 显然是 β 的连续函数，因为它只是指数函数之和。此外，$Z(\beta)$ 关于 β 的所有导数也都是连续的。

然而，相变实质上是这样的：相变所对应的值 β 和 V 叫做临界点，在这一点上，Z 的导数不连续或分散。

马上可以推断

只有具有无限个状态的系统才能表现出相变。

通常，包含一个描述系统大小的参数 N。相变可以出现在 $N\to\infty$ 的极限情况。实际系统可能具有类似于 10^{23} 的 N 值。

如果简单地将 N 个划分函数为 $Z_{(1)}(\beta)$ 的独立系统组合成一个大系统，不会发生什么有趣的事情。N 个独立的相同系统的划分函数简单地为

$$Z_{(N)}(\beta)=[Z_{(1)}(\beta)]^{N} \tag{B.3}$$

现在，虽然函数 $Z_{(N)}(\beta)$ 可能是随 β 快速变化的函数，但并不意味着它表现出了相变。看待划分函数的自然方法是对数形式

$$\ln Z_{(N)}(\beta)=N\ln Z_{(1)}(\beta) \tag{B.4}$$

将原系统复制 N 次，简单地将所有性质（如系统能量和热容）按比例放大一个 N 倍因子。因此，如果原系统没有表现出相变，那么放大后的系统也不会。

只有具有远距离相关性的系统才会表现出相变。

远距离相关性并不要求远程能量耦合；比如，磁铁只有近程耦合（在相邻自旋之间），但是这些耦合却足以产生远距离的次序。

为什么导数分散的点是有趣的?

$\ln Z$ 的导数描述了系统热容(二阶导数)或系统能量波动之类的性质。如果 $\ln Z$ 的二阶导数在温度 $1/\beta$ 处分散,那么系统热容也在此处分散,这意味着系统可以不改变温度而吸收或释放能量(想想冰在冰水中融化的过程);当系统在该温度下处于平衡状态,它的能量会有很大波动,这与正常的大数定律特性不同,后者能量只是改变$\sqrt{N}$的一部分。

表现出相变的一个简化系统

设想有一组 N 个耦合自旋,它们具有如下能量,是其状态 $x \in \{0,1\}^N$ 的函数。

$$E(x) = \begin{cases} -N\varepsilon, & x = (0,0,0,\cdots,0) \\ 0, & \text{其他} \end{cases} \tag{B.5}$$

该能量函数描述了所有自旋对齐于零方向的基状态;在此状态下,每个自旋的能量为 $-\varepsilon$;如果任何自旋改变状态,那么能量为 0。这个模型像是磁耦合的一种极端情况,磁耦合有促使一对自旋对齐的倾向。

可以将它和具有 N 个独立自旋、能量为

$$E^0(x) = \varepsilon \sum_n (2x_n - 1) \tag{B.6}$$

的普通系统进行比较。与第一个系统相仿,具有独立自旋的系统有能量为 $-N\varepsilon$ 的单一基态(0,0,0,…,0),且它大约有 2^N 个能量非常接近 0 的状态,因此独立自旋系统和耦合自旋系统的低温和高温性质几乎相同。

耦合自旋系统的划分函数为

$$Z(\beta) = e^{\beta N\varepsilon} + 2^N - 1 \tag{B.7}$$

函数

$$\ln Z(\beta) = \ln(e^{\beta N\varepsilon} + 2^N - 1) \tag{B.8}$$

的草图如图 B.1(a)所示,同图画出的还有其低温特性

$$\ln Z(\beta) \simeq N\beta\varepsilon, \beta \to \infty \tag{B.9}$$

及其高温特性

$$\ln Z(\beta) \simeq N\ln 2, \beta \to 0 \tag{B.10}$$

箭头标出的点为

$$\beta = \frac{\ln 2}{\varepsilon} \tag{B.11}$$

在此点处,两条渐近线相交。在极限 $N \to \infty$ 时,$\ln Z(\beta)$ 的曲线在这个点处弯曲得越来越厉害(图 B.1(b))。

$\ln Z$ 的二阶导数描述了系统能量方差,在 $\beta = \ln 2/\varepsilon$ 处有一个峰值,大致等于

$$\frac{N^2\varepsilon^2}{4} \tag{B.12}$$

其所对应系统的一半时间处于基态,一半时间处于其他状态。

在这个临界点上,系统热容因而和 N^2 成正比;每个自旋的热容则正比于 N,它对于无穷 N 也是无穷大,这与有相变时系统的特性不同,后者每个原子的热容为一个有限值。

为了便于比较,图 B.2 画出了普通独立自旋系统的划分函数和能量方差。

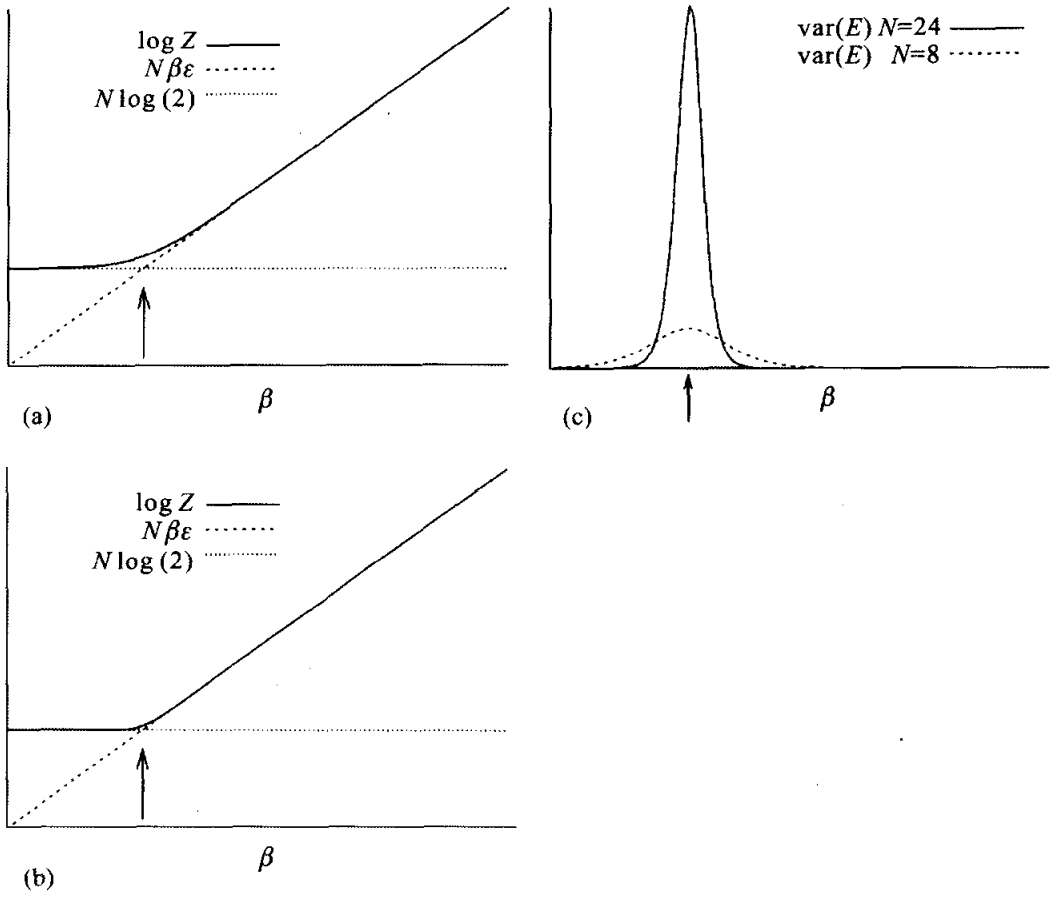


图 B.1　(a) 在大 N 时表现出相变的简化系统的划分函数。箭头所标出的点为 $\beta_c = \log 2/\varepsilon$。(b) 对更大的 N 时的同一内容。(c) 对于两种系统大小，以 β 函数所表示的系统能量方差。随着 N 的增加，该方差在临界点 β_c 处会有一个更尖的尖峰。与图 B.2 进行对比。

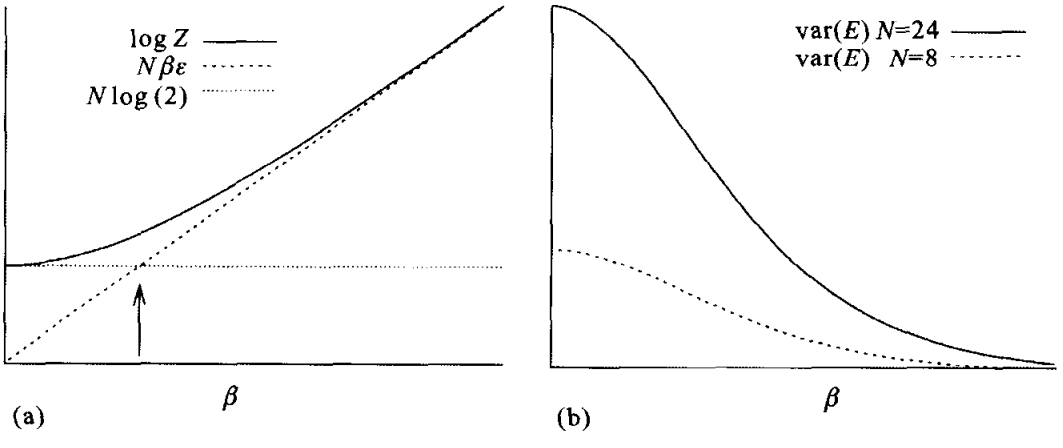


图 B.2　由 N 个独立自旋所组成系统的(a) 划分函数；(b) 能量方差。无论 N 有多大，划分函数从一条渐近线向另一条逐渐变化；能量方差不会出现峰值。在高温(即小 β)时，波动最大，并随系统大小 N 线性变化。

更一般的情况

相变可以划分为“一阶”和“连续”变化。在一阶相变中，有一个或多个阶次参数非连续变化；而在连续相变中，所有阶次参数都连续变化。[什么是阶次参数？它是系统状态的一个标量函数；或更精确一些，是这样一个函数的期望值。]

在临界点附近，第 4 章所定义的“典型性”概念不成立。比如，我们的简化系统，在其临界点处，有 50% 的可能性位于能量为 $-N\varepsilon$ 的状态，而位于其他能量为 0 的各个状态的可能性约为 $1/2^{N+1}$。因此，与由 N 个独立同等分布部分所组成的系统不同，$\ln 1/P(x)$ 不太可能近似于系统在这个点上的熵。

记住信息量（$\ln 1/P(x)$）和能量是紧密相关的。如果典型性成立，那么系统的能量具有可忽略的波动，反之亦然。

附录 C　一些数学知识

C.1　有限域论

多数线性码可以用伽罗瓦理论的语言来表达

为什么伽罗瓦域是表达线性码的一种合适语言？先来看一个定义和几个例子。

域 F 是满足如下性质的一个集合 $F=\{0,F'\}$。

1. F 在加法运算“+”下形成一个阿贝尔群，其中 0 为单位元。[阿贝尔群的意思是，所有元素都是可交换的，即满足 $a+b=b+a$。]

2. F' 在乘法运算“·”下形成一个阿贝尔群；任何元素与 0 相乘等于 0。

3. 这些运算满足分配律 $(a+b)\cdot c=a\cdot c+b\cdot c$。

比如，实数形成一个域，其中的“+”和“·”表示普通加法和乘法。

伽罗瓦域 $GF(q)$ 是具有有限个元素 q 的域。

对于任意 $q=p^m$，其中 p 为素数而 m 为正整数，存在一个唯一的伽罗瓦域；除此之外，不存在其他有限域。

$GF(2)$。$GF(2)$ 的加法和乘法如表 C.1 所示。这里的加法和乘法都是模 2 的。

表 C.1　$GF(2)$ 的加法表和乘法表

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

$GF(p)$。对于任意素数 p，加法和乘法法则都对应于普通模 p 加法和乘法。

$GF(4)$。对于 $m>1$ 时的 $GF(p^m)$，这些法则不是普通加法和乘法。比如，表 C.2 所列 $GF(4)$ 的加法和乘法法则都不是模 4 的。例如，注意 $1+1=0$。那么，如何才能描述 $GF(4)$？结果证明这些元素可以和多项式相关联。考虑 x 的一阶多项式函数，其系数为 $GF(2)$ 的元素。如果表 C.3 所示加法和乘法是以多项式 x^2+x+1 为模，而表中多项式的系数都来自 $GF(2)$，那么这些多项式遵循 $GF(4)$ 的加法和乘法法则。例如，$B\cdot B=x^2+(1+1)x+1=x=A$。每个元素也可表示为如表 C.3 所示的比特模式，其中加法为模 2 按位相加，而乘法用一种合适的进位运算定义。

表 C.2 $GF(4)$的加法表和乘法表

+	0	1	A	B
0	0	1	A	B
1	1	0	B	A
A	A	B	0	1
B	B	A	1	0

·	0	1	A	B
0	0	0	0	0
1	0	1	A	B
A	0	A	B	1
B	0	B	1	A

表 C.3 $GF(4)$中元素的表示

元素	多项式	比特模式
0	0	00
1	1	01
A	x	10
B	$x+1$	11

$GF(8)$。可以将 $GF(8)$的元素记为$\{0,1,A,B,C,D,E,F\}$。每个元素可以映射成 $GF(2)$上的一个多项式。这里的加法和乘法运算是模 x^3+x+1 的多项式加法和乘法。乘法表如下图所示。

元素	多项式	二进制表示
0	0	000
1	1	001
A	x	010
B	$x+1$	011
C	x^2	100
D	x^2+1	101
E	x^2+x	110
F	x^2+x+1	111

·	0	1	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0
1	0	1	A	B	C	D	E	F
A	0	A	C	E	B	1	F	D
B	0	B	E	D	F	C	1	A
C	0	C	B	F	E	A	D	1
D	0	D	1	C	A	F	B	E
E	0	E	F	1	D	B	A	C
F	0	F	D	A	1	E	C	B

为什么伽罗瓦域与线性码有关？假设将一个二进制生成矩阵 $\boldsymbol{G}$ 和二进制向量 $\boldsymbol{s}$ 推广到元素来自更大集合的矩阵和向量，并对定义乘积 $\boldsymbol{Gs}$ 的加法和乘法运算进行推广。为了产生对称信道的一个合适输入，如果对于随机 $\boldsymbol{s}$，乘积 $\boldsymbol{Gs}$ 以等概率产生放大集合中的所有元素，则会方便许多。如果这些元素在加法和乘法运算下都形成一个群，那么均匀分布最容易得到保证，因为这些运算不会破坏元素间的对称。当一个乘法群的两个随机元素相乘时，可以等概率地产生所有元素。对

其他集合(如整数集),这一点不成立,因为乘法运算产生某些元素(合数)的可能性更大些。由定义可知,伽罗瓦域避免了这种破坏对称效应。

C.2　特征向量与特征值

一个方形矩阵 $\boldsymbol{A}$ 的右特征向量是满足下式的非零向量 $\boldsymbol{e}_R$(如表 C.4 所示)

$$\boldsymbol{A}\boldsymbol{e}_R = \lambda \boldsymbol{e}_R \tag{C.1}$$

其中 λ 是对应于此特征向量的特征值。特征值可以是实数也可以是复数,且可以取 0 值。特征向量可以是实向量也可以是复向量。

表 C.4　一些矩阵及其特征向量

矩阵	特征值和特征向量 $\boldsymbol{e}_L$、$\boldsymbol{e}_R$
$\begin{bmatrix}1&2&0\\1&1&0\\0&0&1\end{bmatrix}$	2.41: $\begin{bmatrix}0.58\\0.82\\0\end{bmatrix}$ $\begin{bmatrix}0.82\\0.58\\0\end{bmatrix}$　1: $\begin{bmatrix}0\\0\\1\end{bmatrix}$ $\begin{bmatrix}0\\0\\1\end{bmatrix}$　−0.41: $\begin{bmatrix}-0.58\\0.82\\0\end{bmatrix}$ $\begin{bmatrix}-0.82\\0.58\\0\end{bmatrix}$
$\begin{bmatrix}0&1\\1&1\end{bmatrix}$	1.62: $\begin{bmatrix}0.53\\0.85\end{bmatrix}$ $\begin{bmatrix}0.53\\0.85\end{bmatrix}$　−0.62: $\begin{bmatrix}0.85\\-0.53\end{bmatrix}$ $\begin{bmatrix}0.85\\-0.53\end{bmatrix}$
$\begin{bmatrix}1&1&0&0\\0&0&0&1\\1&0&0&0\\0&0&1&1\end{bmatrix}$	1.62: $\begin{bmatrix}0.60\\0.37\\0.37\\0.60\end{bmatrix}$ $\begin{bmatrix}0.60\\0.37\\0.37\\0.60\end{bmatrix}$　0.5+0.9i: $\begin{bmatrix}0.1-0.5i\\-0.3-0.4i\\0.3+0.4i\\-0.1+0.5i\end{bmatrix}$ $\begin{bmatrix}0.1-0.5i\\0.3+0.4i\\-0.3-0.4i\\-0.1+0.5i\end{bmatrix}$ 0.5−0.9i: $\begin{bmatrix}0.1+0.5i\\-0.3+0.4i\\0.3-0.4i\\-0.1-0.5i\end{bmatrix}$ $\begin{bmatrix}0.1+0.5i\\0.3-0.4i\\-0.3+0.4i\\-0.1-0.5i\end{bmatrix}$　−0.62: $\begin{bmatrix}0.37\\-0.60\\-0.60\\0.37\end{bmatrix}$ $\begin{bmatrix}0.37\\-0.60\\-0.60\\0.37\end{bmatrix}$

矩阵 $\boldsymbol{A}$ 的左特征向量是满足下式的向量 $\boldsymbol{e}_L$

$$\boldsymbol{e}_L^T\boldsymbol{A} = \lambda \boldsymbol{e}_L^T \tag{C.2}$$

下列关于右特征向量的命题也适用于左特征向量。

- 如果一个矩阵有两个或多个线性无关的右特征向量对应于同一特征值,则该特征值称为矩阵的退化特征值或重复特征值。这些特征向量的任何线性组合对应于同一特征值的另一右特征

向量。

- 由定义可知,一个矩阵的主右特征向量对应于最大相关特征值的右特征向量。
- 如果一个实矩阵有一个对应于复特征值$\lambda = x + yi$的右特征向量,那么它也有对应于共轭特征值$\lambda^* = x - yi$的一个右特征向量。

对称矩阵

如果$\boldsymbol{A}$是一个大小为$N \times N$的实对称矩阵,那么

1. $\boldsymbol{A}$的所有特征值和特征向量都是实的。
2. $\boldsymbol{A}$的每一左特征向量都是$\boldsymbol{A}$的同一特征值的右特征向量,反之亦然。
3. 可以找到一组N个特征向量和特征值$\{\boldsymbol{e}^{(a)}, \lambda_a\}_{a=1}^{N}$是正交的,也就是说

$$\boldsymbol{e}^{(a)} \cdot \boldsymbol{e}^{(b)} = \delta_{ab} \tag{C.3}$$

该矩阵可以表示为这些特征向量的外积的加权和

$$\boldsymbol{A} = \sum_{a=1}^{N} \lambda_a [\boldsymbol{e}^{(a)}][\boldsymbol{e}^{(a)}]^{\mathrm{T}} \tag{C.4}$$

(尽管笔者经常使用i和n作为大小为I和N的集合的下标,将使用下标a和b来标识特征向量,即便它们的个数是N。这是为了避免和特征向量的元素相混淆,因为元素用n作下标,例如$e_n^{(a)}$。)

一般方形矩阵

一个$N \times N$矩阵最多有N个不同的特征值。一般来说,有N个不同的特征值,每一个都会有一个左特征向量和一个右特征向量。如果有两个或更多的特征值重合,每一个不同的非零特征值至少会有一个左特征向量和一个右特征向量。

对应于不同特征值的左特征向量和右特征向量是正交的,即

$$\text{如果 } \lambda_a \neq \lambda_b, \text{那么 } \boldsymbol{e}_{\mathrm{L}}^{(a)} \cdot \boldsymbol{e}_{\mathrm{R}}^{(b)} = 0。 \tag{C.5}$$

非负矩阵

定义:如果一个非零矩阵$\boldsymbol{C}$的所有元素满足$C_{mn} \geqslant 0$,那么$\boldsymbol{C}$就是一个非负矩阵。类似地,如果一个非零向量$\boldsymbol{c}$的所有元素都满足$c_i \geqslant 0$,那么$\boldsymbol{c}$就是一个非负向量。

性质:一个非负矩阵具有一个非负的主特征向量。可以有其他不是非负的特征向量对应于同一特征值。但是,如果一个非负矩阵的主特征值不是退化的,那么该矩阵只有一个特征向量$\boldsymbol{e}^{(1)}$,而且它是非负的。

一般来说,所有其他特征值在绝对幅度上要小一些。[在特殊情况下,可以有几个特征值具有相同幅度。]

转移概率矩阵

非负矩阵的一个重要例子是转移概率矩阵$\boldsymbol{Q}$(如表 C.5 所示)。

表 C.5　用于生成通过网格的随机路径的转移概率矩阵

矩阵	特征值和特征向量 e_L、e_R
$\begin{bmatrix} 0 & 0.38 \\ 1 & 0.62 \end{bmatrix}$	1　　-0.38 $\begin{bmatrix} 0.71 \\ 0.71 \end{bmatrix}$　$\begin{bmatrix} 0.36 \\ 0.93 \end{bmatrix}$　　$\begin{bmatrix} -0.93 \\ 0.36 \end{bmatrix}$　$\begin{bmatrix} -0.71 \\ 0.71 \end{bmatrix}$
$\begin{bmatrix} 0 & 0.35 & 0 \\ 0 & 0 & 0.46 \\ 1 & 0.65 & 0.54 \end{bmatrix}$	1　　$-0.2-0.3\mathrm{i}$ $\begin{bmatrix} 0.58 \\ 0.58 \\ 0.58 \end{bmatrix}$　$\begin{bmatrix} 0.14 \\ 0.41 \\ 0.90 \end{bmatrix}$　　$\begin{bmatrix} -0.8+0.1\mathrm{i} \\ -0.2-0.5\mathrm{i} \\ 0.2+0.2\mathrm{i} \end{bmatrix}$　$\begin{bmatrix} 0.2-0.5\mathrm{i} \\ -0.6+0.2\mathrm{i} \\ 0.4+0.3\mathrm{i} \end{bmatrix}$ $-0.2+0.3\mathrm{i}$ $\begin{bmatrix} -0.8-0.1\mathrm{i} \\ -0.2+0.5\mathrm{i} \\ 0.2-0.2\mathrm{i} \end{bmatrix}$　$\begin{bmatrix} 0.2+0.5\mathrm{i} \\ -0.6-0.2\mathrm{i} \\ 0.4-0.3\mathrm{i} \end{bmatrix}$

定义：转移概率矩阵 $\boldsymbol{Q}$ 的每一列都是一个概率向量，即它满足 $\boldsymbol{Q} \geqslant 0$ 且

$$\sum_i Q_{ij} = 1, \quad \text{对所有 } j \text{ 成立} \tag{C.6}$$

这一性质可以使用全 1 向量 $\boldsymbol{n} = (1,1,\cdots,1)^{\mathrm{T}}$ 重写为

$$\boldsymbol{n}^{\mathrm{T}}\boldsymbol{Q} = \boldsymbol{n}^{\mathrm{T}} \tag{C.7}$$

因此，$\boldsymbol{n}$ 是 $\boldsymbol{Q}$ 对应于特征值 $\lambda_1 = 1$ 的主左特征向量

$$\boldsymbol{e}_{\mathrm{L}}^{(1)} = \boldsymbol{n} \tag{C.8}$$

因为 $\boldsymbol{Q}$ 是一个非负矩阵，所以它有一个非负的主右特征向量 $\boldsymbol{e}_{\mathrm{R}}^{(1)}$。一般来说，对于一个各态历经的马尔科夫过程，这个特征向量是对应于幅度为 1 的特征值的唯一右特征向量（说明性的例外请参见表 C.6）。如果将该向量归一化，使得 $\boldsymbol{e}_{\mathrm{R}}^{(1)} \cdot \boldsymbol{n} = 1$，它叫做转移概率矩阵的不变分布。它是在 $\boldsymbol{Q}$ 下保持不变的概率密度。与前面直接辨认出的主左特征向量不同，通常不能未经计算直接得出主右特征向量。

矩阵最多可以有 $N-1$ 个其他右特征向量，并且全部和左特征向量 $\boldsymbol{n}$ 正交，也就是说，它们是零和向量。

表 C.6 转移概率矩阵及其特征向量说明非各态历经的两种方式

矩阵	特征值和特征向量 e_L、e_R
(a) $\begin{bmatrix} 0.90 & 0.20 & 0 & 0 \\ 0.10 & 0.80 & 0 & 0 \\ 0 & 0 & 0.90 & 0.20 \\ 0 & 0 & 0.10 & 0.80 \end{bmatrix}$	1: $\begin{bmatrix} 0 \\ 0 \\ 0.71 \\ 0.71 \end{bmatrix}$ $\begin{bmatrix} 0 \\ 0 \\ 0.89 \\ 0.45 \end{bmatrix}$; 1: $\begin{bmatrix} 0.71 \\ 0.71 \\ 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} 0.89 \\ 0.45 \\ 0 \\ 0 \end{bmatrix}$ 0.70: $\begin{bmatrix} 0.45 \\ -0.89 \\ 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} 0.71 \\ -0.71 \\ 0 \\ 0 \end{bmatrix}$; 0.70: $\begin{bmatrix} 0 \\ 0 \\ -0.45 \\ 0.89 \end{bmatrix}$ $\begin{bmatrix} 0 \\ 0 \\ -0.71 \\ 0.71 \end{bmatrix}$
(a′) $\begin{bmatrix} 0.90 & 0.20 & 0 & 0 \\ 0.10 & 0.79 & 0.02 & 0 \\ 0 & 0.01 & 0.88 & 0.20 \\ 0 & 0 & 0.10 & 0.80 \end{bmatrix}$	1: $\begin{bmatrix} 0.50 \\ 0.50 \\ 0.50 \\ 0.50 \end{bmatrix}$ $\begin{bmatrix} 0.87 \\ 0.43 \\ 0.22 \\ 0.11 \end{bmatrix}$; 0.98: $\begin{bmatrix} -0.18 \\ -0.15 \\ 0.66 \\ 0.72 \end{bmatrix}$ $\begin{bmatrix} -0.66 \\ -0.28 \\ 0.61 \\ 0.33 \end{bmatrix}$ 0.70: $\begin{bmatrix} 0.20 \\ -0.40 \\ -0.40 \\ 0.80 \end{bmatrix}$ $\begin{bmatrix} 0.63 \\ -0.63 \\ -0.32 \\ 0.32 \end{bmatrix}$; 0.69: $\begin{bmatrix} -0.19 \\ 0.41 \\ -0.44 \\ 0.77 \end{bmatrix}$ $\begin{bmatrix} -0.61 \\ 0.65 \\ -0.35 \\ 0.30 \end{bmatrix}$
(b) $\begin{bmatrix} 0 & 0 & 0.90 & 0.20 \\ 0 & 0 & 0.10 & 0.80 \\ 0.90 & 0.20 & 0 & 0 \\ 0.10 & 0.80 & 0 & 0 \end{bmatrix}$	1: $\begin{bmatrix} 0.50 \\ 0.50 \\ 0.50 \\ 0.50 \end{bmatrix}$ $\begin{bmatrix} 0.63 \\ 0.32 \\ 0.63 \\ 0.32 \end{bmatrix}$; 0.70: $\begin{bmatrix} -0.32 \\ 0.63 \\ -0.32 \\ 0.63 \end{bmatrix}$ $\begin{bmatrix} 0.50 \\ -0.50 \\ 0.50 \\ -0.50 \end{bmatrix}$ −0.70: $\begin{bmatrix} 0.32 \\ -0.63 \\ -0.32 \\ 0.63 \end{bmatrix}$ $\begin{bmatrix} -0.50 \\ 0.50 \\ 0.50 \\ -0.50 \end{bmatrix}$; −1: $\begin{bmatrix} 0.50 \\ 0.50 \\ -0.50 \\ -0.50 \end{bmatrix}$ $\begin{bmatrix} 0.63 \\ 0.32 \\ -0.63 \\ -0.32 \end{bmatrix}$

(a) 多于一个特征值为 1 的主特征向量,因为状态空间分为两个非连通部分。(a′) 一个小的扰动破坏了主特征向量的变化。(b) 在这一链下,密度可能在状态空间的两个部分之间摆动不定。除了不变分布之外,还有另一特征值为 −1 的右特征向量。通常这种周边密度对应于幅 1 复特征值。

C.3　摄动理论

摄动理论在本书中并未讨论，但它在本书所涉及的领域中很有用。在本节，推导方形矩阵（不一定是对称矩阵）的特征向量和特征值的一阶摄动理论。关于摄动理论的多数材料都集中于对称矩阵，但是非对称矩阵（比如转移矩阵）也值得涉猎。

假设一个 $N\times N$ 矩阵 $\boldsymbol{H}$ 是一个实参数 ε 的函数 $\boldsymbol{H}(\varepsilon)$，以 $\varepsilon=0$ 为起点。假设可以对 $\boldsymbol{H}(\varepsilon)$ 进行合适的泰勒级数展开

$$\boldsymbol{H}(\varepsilon)=\boldsymbol{H}(0)+\varepsilon\boldsymbol{V}+\cdots \tag{C.9}$$

其中

$$\boldsymbol{V}\equiv\frac{\partial\boldsymbol{H}}{\partial\varepsilon} \tag{C.10}$$

假设对于所有感兴趣的 ε，$\boldsymbol{H}(\varepsilon)$ 都有一个完整的 N 个右特征向量和左特征向量集合，且这些特征向量及其特征值是 ε 的连续函数。最后这个假设未必是一个好假设：如果 $\boldsymbol{H}(0)$ 具有退化的特征值，那么特征向量有可能不是 ε 的连续函数；此时，需要退化摄动理论。退化摄动理论是一个很有意思的主题，但是这里只考虑非退化的情况。

将特征向量和特征值写成如下形式

$$\boldsymbol{H}(\varepsilon)\boldsymbol{e}_{\mathrm{R}}^{(a)}(\varepsilon)=\lambda^{(a)}(\varepsilon)\boldsymbol{e}_{\mathrm{R}}^{(a)}(\varepsilon) \tag{C.11}$$

并进行泰勒级数展开得

$$\lambda^{(a)}(\varepsilon)=\lambda^{(a)}(0)+\varepsilon\mu^{(a)}+\cdots \tag{C.12}$$

其中

$$\mu^{(a)}\equiv\frac{\partial\lambda^{(a)}(\varepsilon)}{\partial\varepsilon} \tag{C.13}$$

和

$$\boldsymbol{e}_{\mathrm{R}}^{(a)}(\varepsilon)=\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\cdots \tag{C.14}$$

其中

$$\boldsymbol{f}_{\mathrm{R}}^{(a)}\equiv\frac{\partial\boldsymbol{e}_{\mathrm{R}}^{(a)}}{\partial\varepsilon} \tag{C.15}$$

可以类似地定义 $\boldsymbol{e}_{\mathrm{L}}^{(a)}$ 和 $\boldsymbol{f}_{\mathrm{L}}^{(a)}$。将这些左向量定义为行向量，因此不需要“转置”运算符并可将它们省去。

可以自由地对特征向量的幅度加上我们想要的任何约束。每一个左特征向量和右特征向量都有任意的幅度。可以使用如下自然约束。首先，对内积加上约束

$$\boldsymbol{e}_{\mathrm{L}}^{(a)}(\varepsilon)\boldsymbol{e}_{\mathrm{R}}^{(a)}(\varepsilon)=1,\quad \text{对所有 } a \text{ 成立} \tag{C.16}$$

将特征向量对 ε 展开，式（C.16）意味着

$$(\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{L}}^{(a)}+\cdots)(\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\cdots)=1 \tag{C.17}$$

由此，可以抽出与 ε 有关的项，得

$$\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\boldsymbol{f}_{\mathrm{L}}^{(a)}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=0 \tag{C.18}$$

现在可以自由地选择两个约束

$$\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\,\boldsymbol{f}_{\mathrm{R}}^{(a)}=0,\ \boldsymbol{f}_{\mathrm{L}}^{(a)}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=0 \tag{C.19}$$

对于对称矩阵这种特殊情况，由欧几里得范数定义的特征向量的长度恒定。

好，现在已经完成所有角色的设定，定义式（C.11）和式（C.9）可以为我们提供关于泰勒级数

展开(C.13)和(C.15)的什么信息呢？将式(C.11)用 ε 进行展开

$$(\boldsymbol{H}(0)+\varepsilon\boldsymbol{V}+\cdots)(\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\cdots)=(\boldsymbol{\lambda}^{(a)}(0)+\varepsilon\mu^{(a)}+\cdots)(\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\cdots) \tag{C.20}$$

注意 ε 项的阶次，可得

$$\boldsymbol{H}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=\boldsymbol{\lambda}^{(a)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\mu^{(a)}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0) \tag{C.21}$$

将上式左乘 $\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)$，可以得出一些有意思的结果

$$\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{H}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{\lambda}^{(a)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\mu^{(a)}\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)$$

$$\Rightarrow\boldsymbol{\lambda}^{(b)}\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=\boldsymbol{\lambda}^{(a)}(0)\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}+\mu^{(a)}\delta_{ab} \tag{C.22}$$

设定 $b=a$，可得

$$\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=\mu^{(a)} \tag{C.23}$$

另一方面，若选择 $b\neq a$，可得

$$\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)=[\boldsymbol{\lambda}^{(a)}(0)-\boldsymbol{\lambda}^{(b)}(0)]\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)} \tag{C.24}$$

$$\Rightarrow\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)}=\frac{1}{\boldsymbol{\lambda}^{(a)}(0)-\boldsymbol{\lambda}^{(b)}(0)}\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0) \tag{C.25}$$

现在假设右特征向量 $\{\boldsymbol{e}_{\mathrm{R}}^{b}(0)\}_{b=1}^{N}$ 形成一个完整的基，肯定可以写出

$$\boldsymbol{f}_{\mathrm{R}}^{(a)}=\sum_{b}w_b\boldsymbol{e}_{\mathrm{R}}^{(b)}(0) \tag{C.26}$$

其中

$$w_b=\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)} \tag{C.27}$$

因此，比较式(C.25)和式(C.27)，有

$$\boldsymbol{f}_{\mathrm{R}}^{(a)}=\sum_{b\neq a}\frac{\boldsymbol{e}_{\mathrm{L}}^{(b)}(0)\boldsymbol{V}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)}{\boldsymbol{\lambda}^{(a)}(0)-\boldsymbol{\lambda}^{(b)}(0)}\boldsymbol{e}_{\mathrm{R}}^{(b)}(0) \tag{C.28}$$

式(C.23)和式(C.28)分别给出了特征值和特征向量的一阶导数，也解决了一阶摄动理论问题。

二阶摄动理论

如果将特征向量公式(C.11)展开到 ε 的二次项，并假设公式

$$\boldsymbol{H}(\varepsilon)=\boldsymbol{H}(0)+\varepsilon\boldsymbol{V} \tag{C.29}$$

是精确的，也就是说，$\boldsymbol{H}$ 纯粹是 ε 的一个线形函数，那么就有

$$(\boldsymbol{H}(0)+\varepsilon\boldsymbol{V})\left(\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\frac{1}{2}\varepsilon^2\boldsymbol{g}_{\mathrm{R}}^{(a)}+\cdots\right)$$

$$=\left(\boldsymbol{\lambda}^{(a)}(0)+\varepsilon\mu^{(a)}+\frac{1}{2}\varepsilon^2\nu^{(a)}+\cdots\right)\left(\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\varepsilon\boldsymbol{f}_{\mathrm{R}}^{(a)}+\frac{1}{2}\varepsilon^2\boldsymbol{g}_{\mathrm{R}}^{(a)}+\cdots\right) \tag{C.30}$$

其中 $\boldsymbol{g}_{\mathrm{R}}^{(a)}$ 和 $\nu^{(a)}$ 是特征向量和特征值的二阶导数。令式(C.30)中两边 ε 的二次项相等，得

$$\boldsymbol{V}\boldsymbol{f}_{\mathrm{R}}^{(a)}+\frac{1}{2}\boldsymbol{H}(0)\boldsymbol{g}_{\mathrm{R}}^{(a)}=\frac{1}{2}\boldsymbol{\lambda}^{(a)}(0)\boldsymbol{g}_{\mathrm{R}}^{(a)}+\frac{1}{2}\nu^{(a)}\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\mu^{(a)}\boldsymbol{f}_{\mathrm{R}}^{(a)} \tag{C.31}$$

将上式左乘 $\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)$，得

$$\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{V}\boldsymbol{f}_{\mathrm{R}}^{(a)}+\frac{1}{2}\boldsymbol{\lambda}^{(a)}\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{g}_{\mathrm{R}}^{(a)}$$

$$=\frac{1}{2}\boldsymbol{\lambda}^{(a)}(0)\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{g}_{\mathrm{R}}^{(a)}+\frac{1}{2}\nu^{(a)}\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{e}_{\mathrm{R}}^{(a)}(0)+\mu^{(a)}\boldsymbol{e}_{\mathrm{L}}^{(a)}(0)\boldsymbol{f}_{\mathrm{R}}^{(a)} \tag{C.32}$$

由约束(C.19),得 $e_L^{(a)}(0)f_R^{(a)}$ 这一项为 0,于是有

$$e_L^{(a)}(0)Vf_R^{(a)}=\frac{1}{2}\nu^{(a)} \tag{C.33}$$

因此,特征值对 ε 的二阶导数为

$$\frac{1}{2}\nu^{(a)}=e_L^{(a)}(0)V\sum_{b\neq a}\frac{e_L^{(b)}(0)Ve_R^{(a)}(0)}{\lambda^{(a)}(0)-\lambda^{(b)}(0)}e_R^{(b)}(0) \tag{C.34}$$

$$=\sum_{b\neq a}\frac{\left[e_L^{(b)}(0)Ve_R^{(a)}(0)\right]\left[e_L^{(a)}(0)Ve_R^{(b)}(0)\right]}{\lambda^{(a)}(0)-\lambda^{(b)}(0)} \tag{C.35}$$

这是所能得到的最好的摄动展开式。

小结

如果引入 V_{ba} 作为 $e_L^{(b)}(0)Ve_R^{(a)}(0)$ 的简单记号,可以将 $H(\varepsilon)=H(0)+\varepsilon V$ 的特征向量展开到一阶为

$$e_R^{(a)}(\varepsilon)=e_R^{(a)}(0)+\varepsilon\sum_{b\neq a}\frac{V_{ba}}{\lambda^{(a)}(0)-\lambda^{(b)}(0)}e_R^{(b)}(0)+\cdots \tag{C.36}$$

特征值展开到二阶

$$\lambda^{(a)}(\varepsilon)=\lambda^{(a)}(0)+\varepsilon V_{aa}+\varepsilon^2\sum_{b\neq a}\frac{V_{ba}V_{ab}}{\lambda^{(a)}(0)-\lambda^{(b)}(0)}+\cdots \tag{C.37}$$

C.4　一些数字

	$2^{8\,192}$	$10^{2\,466}$	大小为 1 K 的不同文件的个数
	$2^{1\,024}$	10^{308}	带 32×32 个自旋的二维伊辛模型的状态数
$2^{1\,000}$		10^{301}	长度为 1 000 的二进制串的个数
2^{500}		3×10^{150}	
	2^{469}	10^{141}	包含 100 个 1 和 900 个 0 的长度为 1 000 的二进制串的个数
	2^{266}	10^{80}	宇宙中电子的个数
2^{200}		1.6×10^{60}	
	2^{190}	10^{57}	太阳系中电子的个数
	2^{171}	3×10^{51}	地球上电子的个数
2^{100}		10^{30}	
	2^{98}	3×10^{29}	宇宙的年龄/皮秒
	2^{58}	3×10^{17}	宇宙的年龄/秒
2^{50}		10^{15}	
2^{40}		10^{12}	

续表

		10^{11}	人脑中神经元的个数
		10^{11}	DVD 所存储的比特数
		3×10^{10}	小麦基因组中的比特数
		6×10^{9}	人类基因组中的比特数
	2^{32}	6×10^{9}	地球上的人口
2^{30}		10^{9}	
		2.5×10^{8}	胼胝体中的纤维数
		2×10^{8}	C. Elegans(一种蠕虫)的基因组所包含的比特数
		2×10^{8}	Arabidopsis thaliana(一种和西兰花相近的开花植物)基因组所包含的比特数
	2^{25}	3×10^{7}	一年/秒
		2×10^{7}	本书的压缩 PostScript 文件所包含的比特数
		2×10^{7}	UNIX 内核的比特数
		10^{7}	E. Coli 基因组或一张软盘所包含的比特数
		4×10^{6}	人类和黑猩猩区分以来的年数
2^{20}		10^{6}	1 048 576
		2×10^{5}	人类和黑猩猩区分以来的代数
		3×10^{4}	人类基因组中的基因数
		3×10^{4}	Arabidopsis thaliana 基因组中的基因数
		1.5×10^{3}	基因中碱基的对数
2^{10}	e^{7}	10^{3}	$2^{10} = 1\ 024$; $e^{7} = 1\ 096$
2^{0}		10^{0}	1
	2^{-2}	2.5×10^{-1}	因每天吸一包烟而死亡的人寿概率
		10^{-2}	因车祸而死亡的人寿概率
2^{-10}		10^{-3}	
		10^{-5}	因为每天饮用 2 升含 12 p. p. b. 苯的水而致癌的人寿概率
2^{-20}		10^{-6}	
		3×10^{-8}	每一代每一个核苷在传递 DNA 编码时的误码率
2^{-30}		10^{-9}	
2^{-60}		10^{-18}	纠错后硬盘出现未检测出的错误的概率

英汉词汇表

acceptance rate 接收率
acceptance ratio method 接收几率方法
accumulator 累加器
activation 激活
activation function 激活函数
activity 活动性
activity rule 活动性规则
adaptive direction sampling 自适应方向采样
adaptive models 自适应模型
adaptive rejection sampling 自适应拒绝采样
address 地址
alchemists 炼金术士
Algorithm 算法
  Covariant 协变算法
  EM EM 算法
  exact sampling 精确采样算法
  expectation-maximization 期望最大化算法
  function minimization 函数最小化算法
  genetic 基因算法
  Hamiltonian Monte Carlo 哈密尔顿 - 蒙特卡罗算法
  independent component analysis 独立元素分析算法
  Langevin Monte Carlo Langevin - 蒙特卡罗算法
  leapfrog 蛙跳算法
  max-product 最大积算法
  perfect simulation 完美仿真算法
  sum-product 和 - 积算法
  Viterbi Viterbi 算法
Allias paradox Allias 悖论
alphabetical ordering 按字母次序排序
America 美国
American 美国的
amino acid 氨基酸
anagram 变位字
annealed importance sampling 退火重要性采样
annealing 退火
  deterministic 确定性退火
  importance sampling 退火重要性采样
antiferromagnetic 反铁磁性的
ape 猿
approximation 近似
  by Gaussian 高斯近似
  Laplace 拉普拉斯近似
  of complex distribution 复杂分布近似
  of density evolution 概率密度进化的近似
  saddle-point 鞍点近似
  Stirling Stirling 近似
  variational 变参近似
arabic 阿拉伯的
architecture 体系结构
arithmetic coding 算术编码
  decoder 解码器
  software 软件
  uses beyond compression 压缩之外的用途
arithmetic progression 算术级数
arms race 军备竞争
associative memory 联想记忆
assumptions 假设
astronomy 天文学
asymptotic equipartition 渐近式均分
  why it is a misleading term 为什么它是一个易于令人误解的术语
Atlantic 大西洋的
automatic relevance determination 自动相关性确定
automobile data reception 汽车数据接收

average, *see* expectation 平均,参见期望
AWGN 加性高斯白噪声

background rate 背景速率
backpropagation 反向传播
backward pass 反向传递
bad, *see* error-correcting code, bad 差的,参见差的纠错码
balance 平衡
Baldwin effect Baldwin 效应
ban (unit) 班(单位)
bandwidth 带宽
bar-code 条形码
base transitions 基转换
base-pairing 基配对
basis dependence 基依赖性
bat 蝙蝠
battleships 战舰
Bayes' theorem 贝叶斯定理
Bayesian 贝叶斯的
Bayesian belief networks 贝叶斯置信网络
Bayesian inference 贝叶斯推理
BCH codes BCH 码
BCJR algorithm BCJR 算法
Belief 置信度
belief propagation, *see* sum-product algorithm 置信度传播,参见和 - 积算法
Benford's law Benford 法则
bent coin 变形硬币
Bernoulli distribution Bernoulli 分布
bet 打赌
beta distribution Beta 分布
Bethe free energy Bethe 自由能
Bhattacharyya parameter Bhattacharyya 参数
bias 偏差
  in neural net 神经网络中的
  in statistics 统计学上的
biased 有偏的
biexponential distribution 双指数分布
bifurcation 分叉
binary entropy function 二进制熵函数
binary erasure channel 二进制删除信道
binary images 二进制图像
binary representations 二进制表示
binary symmetric channel 二进制对称信道
binding DNA DNA 结合
binomial distribution 二项式分布
bipartite graph 二分图
birthday 生日
bit (unit) 比特(单位)
bits back 比特返回
bivariate Gaussian 双变量高斯
black 黑色的
Bletchley Park Bletchley 公园
Blind Watchmaker 盲眼钟表匠
block code, *see* source code or error-correcting code 分组码,参见信源编码或纠错码
block-sorting 分组排序
blow up 发散
blur 弄模糊
Boltzmann entropy 玻耳兹曼熵
bombes 炸弹
book ISBN 国际标准书号
bound 界
bounded-distance decoder 有限距离译码器
box 方块
boyish matters 男孩式事物
brain 大脑
Bridge 桥牌
British 英国的
broadcast channel 广播信道
Brownian motion 布朗运动
BSC, *see* channel, binary symmetric BSC,参见二进制对称信道
budget 预算
Buffon's needle Buffon 针
burglar alarm and earthquake 防盗警报与地震
Burrows-Wheeler transform Burrows-Wheeler 变换
burst errors 突发错误
bus-stop paradox 公交车站悖论

cable labeling 电缆标示
calculator 计算器
camera 照相机
canonical 正则的
capacity 容量
  channel with synchronization errors 具有同步误差的信道容量
  constrained channel 受限信道容量
  Gaussian channel 高斯信道容量
  Hopfield network Hopfield 网络容量
  neural network 神经网络容量
  neuron 神经元容量
  symmetry argument 容量的对称性论证
car data reception 汽车数据接收
card 纸牌
casting out nines 舍 9 法
Cauchy distribution 柯西分布
caution, *see* sermon 告诫,参见训诫
  equipartition 均分
  Gaussian distribution 高斯分布
  importance sampling 重要性采样
  sampling theory 抽样理论
cave 洞穴
caveat, *see* caution 警告,参见告诫
cellphone, *see* mobile phone 蜂窝电话,参见移动电话
cellular automaton 蜂窝式自动机
central-limit theorem 中心极限定理
  see law of large numbers 参见大数定理
centre of gravity 重心
chain rule 链式法则
challenges 挑战
channel 信道
  AWGN 加性高斯白噪声信道
  binary erasure 二进制删除信道
  binary symmetric 二进制对称信道
  broadcast 广播信道
  bursty 突发信道
  capacity 信道容量
    connection with physics 与物理学相关联的信道
  coding theorem, *see* noisy-channel coding theorem 信道编码定理,参见噪声信道编码定理
  complex 复信道
  constrained 受限信道
  continuous 连续信道
  discrete memoryless 离散无记忆信道
  erasure 删除信道
  extended 扩展信道
  fading 衰落信道
  Gaussian 高斯信道
  input ensemble 输入总体信道
  multiple access 多址信道
  multiterminal 多终端信道
  noiseless 无噪信道
  noisy 有噪信道
  noisy typewriter 有噪打字机
  symmetric 对称信道
  two-dimensional 二维信道
  unknown noise level 具有未知噪声水平的信道
  variable symbol durations 具有可变符号周期的信道
  with dependent sources 具有相关信源的信道
  with memory 有记忆信道
  Z channel Z 信道
cheat 作弊
Chebyshev inequality 契比雪夫不等式
checkerboard 西洋跳棋棋盘
Chernoff bound Chernoff 界
chess board 棋盘
chi-squared χ^2
Cholesky decomposition Cholesky 分解
chromatic aberration 色差, 像差
cinema 电影
circle 圈
classical statistics 经典统计学
  criticisms 批判主义
classifier 分类器
Claude Shannon 克劳德·香农
clustering 聚类

coalescence 汇合
cocked hat 三角帽
code, *see* error-correcting code, source code (for data compression), symbol code, arithmetic coding, linear code, random code or hash code 码，参见纠错码、信源码(用于数据压缩)、符号码、算术编码、线性码、随机码或散列码
dual, *see* error-correcting code 对偶,参见纠错码
dual for constrained channel 受限信道的对偶
variable-length 变长(的)
code-equivalent 码等价
codebreakers 密码破译者
codeword, *see* source code, symbol code, or error-correcting code 码字,参见信源码,符号码或纠错码
coding theory 编码理论
coin 硬币
coincidence 巧合
collective 集合(式)
collision 冲突
coloured noise 有色噪声
combination 组合
commander 指挥官
communication 通信
broadcast 广播通信
of dependent information 具有相关信息的通信
over noiseless channels 无噪信道上的通信
perspective on learning 从通信角度来看的学习
competitive learning 竞争学习
complexity 复杂度
complexity control 复杂度控制
compress 压缩
compression, *see* source code 压缩,参见信源码
future methods 未来的压缩方法
lossless 无损压缩
lossy 有损压缩
of already-compressed files 已压缩文件的压缩
of *any* file 任意文件的压缩
universal 通用压缩
computer 计算机
concatenation 级联
error-correcting codes 纠错码的级联
in compression 压缩中的级联
in Markov chains 马尔科夫链的级联
concave⌒ 凹⌒
conditional entropy 条件熵
cones 圆锥细胞
confidence interval 置信区间
confidence level 置信水平
confused gameshow host 困惑的游戏秀主持人
conjugate gradient 共轭梯度
conjugate prior 共轭先验(分布)
conjuror 魔术师
connection between 相互之间的联系
channel capacity and physics 信道容量与物理之间的联系
error correcting code and latent variable model 纠错码与隐式变量模型之间的联系
pattern recognition and error-correction 模式识别与纠错之间的联系
supervised and unsupervised learning 监督式学习与非监督式学习之间的联系
vector quantization and error-correction 向量量化与纠错之间的联系
connection matrix 连接矩阵
constrained channel 受限信道
constraint satisfaction 满足约束
content-addressable memory 内容可寻址的内存
continuous channel 连续信道
control treatment 控制治疗
conventions, *see notation* 约定,参见记号
error function 误差函数
logarithms 对数
matrices 矩阵
vectors 向量
convex hull 凸包

convex⌣ 凸⌣
convexity 凸性
convolution 卷积
convolutional code 卷积码
Copernicus 哥白尼
correlated sources 相关信源
correlations 相关性
  among errors 误差之间的相关性
  and phase transitions 相关性与相变
  high-order 高阶相关性
  in images 图像中的相关性
cost function 代价函数
cost of males 雄性的代价
counting 计数
counting argument 计数论据
coupling from the past 与过去的耦合
covariance 协方差
covariance function 协方差函数
covariance matrix 协方差矩阵
covariant algorithm 协变算法
Cox axioms Cox 公理
crib 栅栏
critical fluctuations 临界波动
critical path 关键路径
cross-validation 交叉确认
crossover 交叉
crossword 纵横字谜
cryptanalysis 密码分析学
cryptography 密码学
  digital signatures 数字签名
  tamper detection 篡改检测
cumulative probability function 累计概率函数
cycles in graphs 图中的回路
cyclic 循环的

data compression, *see* source code 数据压缩,参见信源码
data entry 数据项
data modelling, *see* modelling 数据建模,参见建模
data set 数据集
death penalty 死刑
deciban (unit) 分班(单位)
decibel 分贝
decibels 分贝(复制)
decision theory 决策论
decoder 译码器
  bitwise 逐比特译码器
  bounded-distance 距离受限译码器
  codeword 译码器码字
  probability of error 译码器差错概率
degree 度
degree sequence, *see* profile 度序列,参见文档
degrees of belief 置信度
degrees of freedom 自由度
déjà vu 似曾相识的感觉;幻觉记忆
delay line 延迟线
deletions 删除
delta function δ 函数
density evolution 密度进化
density modelling 密度建模
dependent sources 相关信源
depth of lake 湖深
design theory 设计理论
detailed balance 详细平衡
detection of forgery 伪造检测
deterministic annealing 确定性退火
dictionary 字典
difference-set cyclic code 差集循环码
differentiator 微分器
diffusion 扩散
digital cinema 数字影院
digital fountain 数字喷泉
digital signature 数字签名
digital video broadcast 数字视频广播
dimensions 维数
dimmer 二聚物
directory 目录
Dirichlet distribution Dirichlet 分布
Dirichlet model Dirichlet 模型
discriminant function 判别式函数
discriminative training 判别训练

disease 疾病
disk drive 磁盘驱动器
distance 距离
  D_{KL} D_{KL}距离
  bad 坏距离
  distance distribution 距离分布
  entropy distance 熵距离
  Gilbert-Varshamov Gilbert-Varshamov 距离
  good 良性距离
  Hamming 汉明距离
  isn't everything 距离并不代表一切
  of code 码距
    good/bad 良性/恶性距离
  of code, and error probability 码距与差错概率
  of concatenated code 级联码距离
  of product code 乘积码距离
  relative entropy 相对熵距离
  very bad 恶性距离
distribution 分布
  beta Beta 分布
  biexponential 双指数分布
  binomial 二项式分布
  Cauchy 柯西分布
  Dirichlet Dirichlet 分布
  exponential 指数分布
  gamma 伽马分布
  Gaussian 高斯分布
    sample from 从高斯分布中采样
  inverse-cosh 逆 cosh 分布
  log-normal 对数正态分布
  Luria-Delbrück Luria-Delbrück 分布
  normal 正态分布
  over periodic variables 在周期变量上的分布
  Poisson 泊松分布
  Student-t "学生"t 分布
  useful 有用的分布
  Von Mises Von Mises 分布
divergence 发散
DNA DNA
  replication DNA 复制
do the right thing 做正确的事
dodecahedron code 12 面体码
dongle 挂件
doors, on game show (游戏秀中的)门
draw straws 抽签
dream 梦
DSC, *see* difference-set cyclic code DSC,参见差集循环码
dual 对偶
dumb Metropolis 哑 Metropolis

earthquake and burglar alarm 地震与防盗警报
earthquake, during game show 地震,在游戏秀中
edge 边
eigenvalue 特征值
EM algorithm EM 算法
E-mail 电子邮件
empty string 空字符串
encoder 编码器
energy 能量
English 英语
Enigma Enigma 机
ensemble 总体
  extended 扩展的
ensemble learning 总体学习
entropic distribution 熵分布
entropy 熵
  Boltzmann 玻耳兹曼熵
  conditional 条件熵
  Gibbs Gibbs 熵
  joint 联合熵
  marginal 边缘熵
  mutual information 互信息熵
  of continuous variable 连续变量的熵
  relative 相对熵
entropy distance 熵距离
epicycles 本轮
equipartition 均分
erasure channel 删除信道
erasure-correction 删除纠正
erf 误差函数 erf

ergodic 各态历经的
error bars 误差限
error correction, *see* error -correcting code 纠错,参见纠错码
in DNA replication 在 DNA 复制中的纠错
in protein synthesis 在蛋白质合成中的纠错
error detection 检错
error floor 误差下限
error function 误差函数
error probability 差错概率
block 分组差错概率
in compression 压缩中的差错概率
error-correcting code 纠错码
bad 差的纠错码
block code 分组纠错码
concatenated 级联纠错码
convolutional 卷积纠错码
cyclic 循环纠错码
decoding 纠错码译码
density evolution 纠错码密度进化
difference-set cyclic 差集循环纠错码
distance, *see* distance of code 纠错码距离,参见码距
dodecahedron 12 面体纠错码
dual 对偶纠错码
erasure channel 删除信道的纠错码
Gallager, *see* error-correcting codes, low-density parity-check Gallager 纠错码,参见低密度奇偶校验纠错码
Golay Golay 纠错码
good 好纠错码
Hamming 汉明纠错码
in DNA replication DNA 复制中的纠错码
in protein synthesis 蛋白质合成中的纠错码
interleaving 纠错码交织
linear 线性纠错码
noisy-channel coding theorem 线性纠错码的噪声信道编码定理
low-density generator-matrix 低密度纠错码的生成矩阵
low-density parity-check 低密度奇偶校验码
fast encoding 低密度奇偶校验码的快速编码
profile 低密度奇偶校验码的文档
LT code LT 纠错码
maximum distance separable 最大可分隔距离纠错码
nonlinear 非线性纠错码
P_3 P_3 纠错码
parity-check code 奇偶校验码
pentagonful 五边形纠错码
perfect 完美纠错码
practical 实际纠错码
product code 乘积码
quantum 量子纠错码
random 随机纠错码
random linear 随机线性纠错码
rate 纠错码率
rateless 无码率纠错码
rectangular 矩形纠错码
Reed-Solomon code Reed-Solomon 码
repeat-accumulate 重复-累计码
repetition 重复纠错码
simple parity 简单奇偶校验纠错码
sparse graph 稀疏图纠错码
density evolution 稀疏图纠错码的密度进化
syndrome decoding 校正子译码
variable rate 变码率纠错码
very bad 极差纠错码
very good 极好纠错码
weight enumerator 权重枚举器
with varying level of protection 具有可变保护级的纠错码
error-reject curves 误差拒绝曲线
errors, *see* channel 差错,参见信道
estimate 估计
estimator 估计器
eugenics 优生学
euro 欧元
evidence 显著性,证据
typical behaviour of 显著性的典型特性

evolution 进化
  as learning 随学习的进化
  Baldwin effect Baldwin 效应的进化
  colour vision 色彩视觉的进化
  of the genetic code 基因代码的进化
evolutionary computing 进化型计算
exact sampling, *see* Monte Carlo methods 精确采样,参见蒙特卡罗方法
exchange rate 交换率
exchangeability 可交换性
exclusive or 异或
EXIT chart EXIT 图
expectation 期望
expectation propagation 期望传播
expectation-maximization algorithm, *see* EM algorithm 期望最大化算法,参见 EM 算法
experimental design 试验设计
experimental skill 试验技巧
explaining away 辩解
exploit 开采
explore 勘探
exponential distribution 指数分布
  on integers 在整数上服从指数分布
exponential-family 指数族
expurgation 删除
extended channel 扩展信道
extended code 扩展码
extended ensemble 扩展总体
extra bit 额外比特
extreme value 极值
eye movements 眼部运动

factor analysis 因式分析
factor graph 因式图
factorial 阶乘
fading channel 衰落信道
feedback 反馈
female 雌性
ferromagnetic 铁磁的
Fibonacci 斐波纳契
field 域
file storage 文件存储
finger 手指
finite field theory, *see* Galois field 有限域理论,参见伽罗瓦域
fitness 适应性
fixed point 不动点
Florida 佛罗里达
fluctuation analysis 波动分析
fluctuations 波动
focus 聚焦
football pools 足球场
forensic 适于法庭的
forgery 伪造
forward pass 前向传递
forward probability 前向概率
forward-backward algorithm 前向 - 反向算法
Fourier transform 傅里叶变换
fovea (视网膜中央)小凹
free energy 自由能
  minimization 自由能最小化
  variational 变参自由能
frequency 频度
frequentist, *see* sampling theory 频率论者,参见抽样理论
Frobenius-Perron theorem Frobenius-Perron 定理
frustration 受挫
full probabilistic model 完全随机模型
function minimization 函数最小化
functions 函数

gain 增益
Galileo code 伽利略码
Gallager code, *see* error-correcting codes low-density parity-check Gallager 码,参见 Gallager 纠错码,Gallager 低密度校验码
Galois field 伽罗瓦域
game, *see* puzzle 游戏,参见谜
  Bridge 桥牌游戏
  guess that tune 猜曲调游戏
  guessing 猜测游戏

life 生命游戏
sixty-three "63"游戏
submarine 潜水艇
three doors 三门游戏
twenty questions 20 问题游戏
game show 游戏秀
game-playing 玩游戏
gamma distribution 伽马分布
gamma function 伽马函数
ganglion cells 神经节细胞
Gaussian channel 高斯信道
Gaussian distribution 高斯分布
N-dimensional *N* 维高斯分布
approximation 近似高斯分布
parameters 高斯分布参数
sample from 从高斯分布中采样
Gaussian processes 高斯过程
variational Gaussian process classifier 变参高斯过程分类器
general position 一般位置
generalization 推广
generalized parity-check matrix 广义奇偶校验矩阵
generating function 生成函数
generative model 生成模型
generator matrix 生成矩阵
genes 基因
genetic algorithm 基因算法
genetic code 基因代码
genome 基因组
geometric progression 几何级数
geostatistics 地理统计学的
GF(q), *see* Galois field GF(q),参见伽罗瓦域
Gibbs entropy Gibbs 熵
Gibbs sampling, *see* Monte Carlo methods
Gibbs 采样，参见蒙特卡罗方法
Gibbs' inequality Gibbs 不等式
Gilbert-Varshamov conjecture Gilbert-Varshamov 猜想
Gilbert-Varshamov distance Gilbert-Varshamov 距离
Gilbert-Varshamov rate Gilbert-Varshamov 率
girlie stuff 女孩式事物，
Glauber dynamics Glauber 动力学
Golay code Golay 码
golden ratio 黄金比例
good, *see* error-correcting code, good 良性,参见良性纠错码
gradient descent 梯度下降
natural 自然的
graduated non-convexity 逐级非凸性
grain size 粒度大小
graph 图
factor graph 因式图
of code 码图
graphs and cycles 图与回路
guerilla 游击队
guessing game 猜测游戏

Hamiltonian Monte Carlo 哈密尔顿-蒙特卡罗
Hamming code 汉明码
graph 汉明码图
Hamming distance 汉明距离
handwritten digits 手写数字
hard drive 硬盘驱动器
hash code 散列码
hash function 散列函数
linear 线性
one-way 单向
hat puzzle 帽子谜题
heat bath 热浴
heat capacity 热容
Hebbian learning Hebbian 学习
Hertz 赫兹
Hessian Hessian(矩阵)
hidden Markov models 隐马尔科夫模型
hidden neurons 隐神经元
hierarchical clustering 递阶聚类
hierarchical model 递阶模型
high dimensions, life in 高维，其中的生活
hint for computing mutual information 计算互信息的暗示
hitchhiker 搭便车者

homogeneous 齐次的
Hopfield network Hopfield 网络
  capacity Hopfield 网络容量
hot-spot 热点
Huffman code 霍夫曼码
  'optimality' 霍夫曼码的最优性
  disadvantages 霍夫曼码的缺点
  general alphabet 通用字母的霍夫曼码
human 人类
human-machine interfaces 人－机界面
hybrid Monte Carlo, *see* Hamiltonian Monte Carlo 混合蒙特卡罗,参见哈密尔顿－蒙特卡罗
hydrogen bond 氢键
hyperparameter 超参数
hypersphere 超球面
hypothesis testing, *see* model comparison, sampling theory 假设检验,参见模型比较,抽样理论

i. i. d. 独立同等分布
ICA, *see* independent component analysis ICA,参见独立元素分析
ICF (intrinsic correlation function) ICF(内在相关函数)
identical twin 相同的孪生子
identity matrix 单位矩阵
ignorance 未知
ill-posed problem 病态问题
image 图像
  integral 整体图像
image analysis 图像分析
image compression 图像压缩
image models 图像模型
image processing 图像处理
image reconstruction 图像重构
implicit assumptions 隐含假设
implicit probabilities 隐含概率
importance sampling, *see* Monte Carlo methods 重要性采样,参见蒙特卡罗方法
improper 不当的
improper prior 不当先验分布
in-car navigation 车内导航
independence 独立
independent component analysis 独立元素分析
indicator function 指示函数
inequality 不等式
inference 推理
  and learning 推理与学习
inference problems 推理问题
  bent coin 变形硬币的推理问题
information 信息
information content 信息量
  how to measure 如何度量信息含量
  Shannon 香农信息量
information maximization 信息最大化
information retrieval 信息恢复
information theory 信息论
inner code 内码
Inquisition 宗教裁判所
Insertions 插值
instantaneous 瞬时的
integral image 整体图像
interleaving 交织
internet 互联网
intersection 交集
intrinsic correlation function 内在相关函数
invariance 不变性
invariant distribution 不变分布
inverse probability 逆概率
inverse-arithmetic-coder 逆算术编码器
inverse-cosh distribution 逆 cosh 分布
inverse-gamma distribution 逆伽马分布
inversion of hash function 散列函数的逆
investment portfolio 投资证券
irregular 不规则的
ISBN 国际标准书号
Ising model 伊辛模型
iterative probabilistic decoding 迭代式概率译码

Jacobian 雅可比的
Jeffreys prior Jeffreys 先验概率

Jensen's inequality Jensen 不等式
Jet Propulsion Laboratory 喷气推进实验室
Johnson noise Johnson 噪声
joint ensemble 联合总体
joint entropy 联合熵
joint typicality 联合典型性
joint typicality theorem 联合典型性定理
journal publication policy 期刊出版政策
judge 法官
juggling 摆动
junction tree algorithm 联合树算法
jury 陪审团

K - means clustering K - 均值聚类
  derivation 推导
  soft 软 K - 均值聚类
    kaboom 软 K - 均值聚类的发散
Kalman filter 卡尔曼滤波器
kernel 核
key points
  communication 要点通信
  how much data needed 要点通信需要多少数据
  how to solve probability problems 如何求解要点通信的概率问题
  likelihood principle 要点通信的似然原理
  model comparison 要点通信的模型比较
  Monte Carlo 蒙特卡罗要点通信
keyboard 键盘
Kikuchi free energy Kikuchi 自由能
KL distance KL 距离
Knowlton-Graham partitions Knowlton-Graham 划分
Kraft inequality Kraft 不等式
Kriging Kriging 方法
Kullback - Leibler divergence, *see* relative entropy Kullback - Leibler 发散,参见相对熵

Lagrange multiplier 拉格朗日因子
lake 湖
Langevin method Langevin 方法
Langevin process Langevin 过程
language model 语言模型
Laplace approximation, *see* Laplace's method 拉普拉斯近似,参见拉普拉斯方法
Laplace model 拉普拉斯模型
Laplace prior 拉普拉斯先验概率
Laplace's method 拉普拉斯方法
Laplace's rule 拉普拉斯规则
latent variable 隐式变量
latent variable model 隐式变量模型
  compression 隐式变量模型的压缩
law of large numbers 大数定理
lawyer 律师
leaf 叶
leapfrog algorithm 蛙跳算法
learning 学习
  as communication 作为通信的学习
  as inference 作为推理的学习
  Hebbian Hebbian 学习
  in evolution 进化中的学习
learning algorithms 学习算法
  backpropagation 反向传播学习算法
  Boltzmann machine Boltzmann 机学习算法
  classification 分类学习算法
  competitive learning 竞争式学习算法
  Hopfield network Hopfield 网络学习算法
  K - means clustering K - 均值聚类学习算法
  multilayer perceptron 多层感知器学习算法
  single neuron 单神经元学习算法
learning rule 学习规则
Lempel - Ziv coding Lempel - Ziv 编码
  criticisms Lempel - Ziv 编码批判
life 生命
life in high dimensions 高维空间中的生活
likelihood 似然函数
  contrasted with probability 似然函数和概率的对比
  subjectivity 似然函数的主观性
likelihood equivalence 似然等价性
likelihood principle 似然原理
limit cycle 极限环

linear block code 线性分组码
  decoding 线性分组码的译码
  noisy-channel coding theorem 线性分组码的噪声信道编码定理
linear feedback shift register 线性反馈移位寄存器
linear regression 线性回归
little 'n' large data set 小数据集与大数据集
log-normal 对数正态
logarithms 对数
logit 对数单位
long thin strip 细长条
loopy 循环式
loopy belief propagation 循环式置信度传播
loopy message-passing 循环式信息传递
lossy compression 有损压缩
low-density generator-matrix code 低密度生成矩阵码
low-density parity-check code 低密度奇偶校验码
  staircase 阶梯式低密度奇偶校验码
LT code LT 码
Lyapunov function Lyapunov 函数

machine learning 机器学习
macho 有男子汉气概的
magician 魔术师
magnetic recording 磁性记录
majority vote 多数表决
male 雄性
MAP, *see* maximum a *posteriori* MAP,参看最大后验
MAP decoding MAP 译码
mapping 映射
marginal entropy 边缘熵
marginal likelihood, *see* evidence 边缘似然函数,参见显著性
marginalization 边缘化
Markov chain 马尔科夫链
Markov chain Monte Carlo, *see* Monte Carlo methods 马尔科夫链蒙特卡罗,参见蒙特卡罗方法
Markov model, *see* Markov chain and hidden Markov model 马尔科夫模型,参见马尔科夫链与隐马尔科夫模型
marriage 婚姻
matrix 矩阵
matrix identities 矩阵恒等式
max-product 最大积
maxent, *see* maximum entropy maxent,参见最大熵
maximum distance separable 最大可分隔距离
maximum entropy 最大熵
maximum likelihood 最大似然
maximum likelihood decoder 最大似然译码器
maximum a *posteriori* decoder 最大后验译码器
maximum a *posteriori* 最大后验
MCMC (Markov chain Monte Carlo), *see* Monte Carlo methods MCMC (马尔科夫链蒙特卡罗),参见蒙特卡罗方法
MDL, *see* minimum description length MDL,参见最小描述长度
mean 均值
mean field theory 平均场论
melody 旋律
memory 记忆
  address-based 基于地址的记忆
  associative 联想式记忆
  content-addressable 内容可寻址内存
message passing 消息传递
  BCJR BCJR 消息传递
  belief propagation 置信度传播式消息传递
  forward-backward 前向 - 反向消息传递
  in graphs with cycles 有回路图中的消息传递
  loopy 往复式消息传递
  sum-product algorithm 消息传递的和 - 积算法
  Viterbi Viterbi 消息传递
metacode 元码
metric 度量
Metropolis method, *see* Monte Carlo methods Metropolis 方法,参见蒙特卡罗方法
micro-saccades 微 - 快速扫视

microsoftus 微软症
microwave oven 微波炉
min-sum algorithm 最小和算法
mine (hole in ground) 矿井(地洞)
minimax 极小 - 极大
minimization 最小化
minimum description length 最小描述长度
minimum distance 最小距离
  distance of code 最小码距
mirror 镜
mixing coefficients 混合系数
mixture 混合
  in Markov chains 马尔科夫链中的混合
mixture distribution 混合分布
mixture modelling 混合建模
mixture of Gaussians 高斯变量的混合
MLP, *see* multilayer perceptron MLP,参见多层感知器
MML, *see* minimum description length MML,参见最小描述长度
mobile phone 移动电话
model 模型
  latent variable 隐式变量模型
model comparison 模型比较
  typical behaviour of evidence 显著性的典型特性模型比较
  typical evidence 典型显著性的模型比较
modelling 建模
  density modelling 密度建模
models of images 图像模型
moderation, *see* marginalization 适度,参见边缘化
molecules 分子
momentum 动量
Monte Carlo methods 蒙特卡罗方法
  acceptance rate 接收率
  acceptance ratio method 接收比方法
  annealed importance sampling 退火重要性采样
  coalescence 蒙特卡罗方法的汇合
  dependence on dimensionality 蒙特卡罗方法对维度的依赖性
  exact sampling 精确采样蒙特卡罗方法
  for visualization 蒙特卡罗方法的可视化
  Gibbs sampling Gibbs 采样蒙特卡罗方法
  Hamiltonian Monte Carlo 哈密尔顿 - 蒙特卡罗法
  hybrid Monte Carlo, *see* Hamiltonian Monte Carlo 混合蒙特卡罗法,参见哈密尔顿 - 蒙特卡罗法
  importance sampling 重要性采样蒙特卡罗法
  weakness of 蒙特卡罗法的缺点
  information communication in 蒙特卡罗法中的信息通信
  Langevin method Langevin 法
  Markov chain Monte Carlo 马尔科夫链蒙特卡罗法
  Metropolis method Metropolis 法
  dumb Metropolis 哑 Metropolis 法
  multi-state 多态
  overrelaxation 超松弛法
  ordered 有序超松弛法
perfect simulation 完美仿真
random walk suppression 随机游动抑制
random-walk Metropolis 随机游动 Metropolis 法
rejection sampling 拒绝采样
  adaptive 自适应拒绝采样
  reversible jump 可逆跳转
  simulated annealing 模拟退火
  thermodynamic integration 热力学整合
  umbrella sampling 伞状采样
Monty Hall problem Monty Hall 问题
Morse Morse 式电码
motorcycle 摩托车
movie 电影
multilayer perceptron 多层感知器
multiple access channel 多址信道
multiterminal networks 多终端网络
multivariate Gaussian 多变量高斯(分布)
Munro-Robbins theorem Munro-Robbins 定理
murder 谋杀
music 音乐

mutation rate 突变率
mutual information 互信息
  how to compute 如何计算互信息
myth 神话
  compression 关于压缩的神话

nat (unit) 奈特(单位)
natural gradient 自然梯度
natural selection 自然选择
navigation 导航
network 网络
neural network 神经网络
  capacity 神经网络容量
  learning as communication 作为通信的神经网络学习
  learning as inference 作为推理的神经网络学习
neuron 神经元
  capacity 神经元容量
Newton algorithm 牛顿算法
nines '9
noise, *see* channel 噪声, 参见信道
  coloured 有色噪声
  spectral density 噪声谱密度
  white 白噪声
noisy channel, *see* channel 噪声信道, 参见信道
noisy typewriter 有噪打字机
noisy-channel coding theorem 噪声信道编码定理
  Gaussian channel 高斯噪声信道编码定理
  linear codes 线性码的噪声信道编码定理
  poor man's version 噪声信道编码定理的穷人版
noisy-or 有噪或
non-confusable inputs 不易混淆的输入
noninformative 不含信息的
nonlinear 非线性的
nonlinear code 非线性码
nonparametric data modelling 非参数数据建模
nonrecursive 非递归的
noodle, Buffon's Buffon 针
normal, *see* Gaussian 正态, 参见高斯
normal graph 正态图
normalizing constant, *see* partition function 归一化常数, 参见划分函数
not-sum 非和
notation, *see* conventions 记号, 参见约定
  absolute value 绝对值记号
  conventions of this book 本书记号约定
  convex/concave 凸/凹记号
  entropy 熵记号
  expectation 期望记号
  intervals 区间记号
  logarithms 对数记号
  matrices 矩阵记号
  probability 概率记号
  set size 集合大小记号
  transition probability 转移概率记号
  vectors 向量记号
NP-complete NP-完全
nucleotide 核苷
nuisance parameters 麻烦参数
numerology 数字命理学
Nyquist sampling theorem 奈奎斯特采样定理

objective function 目标函数
Occam factor 奥卡姆因式
Occam's razor 奥卡姆剃刀
octal 八进制的
octave octave 软件
Ode to Joy 欢乐颂
one-way hash function 单向散列函数
optic nerve 视觉神经
optimal decoder 最优译码器
optimal input distribution 最优输入分布
optimal linear filter 最优线性滤波器
optimal stopping 最优停止
optimization 最优化
  gradient descent 梯度下降最优化
  Newton algorithm 牛顿算法最优化
  of model complexity 模型复杂度的最优化
ordered overrelaxation 有序超松弛

orthodox statistics, *see* sampling theory 正统统计学，参见抽样理论
outer code 外部码
overfitting 超拟合
overrelaxation 超松弛
  ordered 有序超松弛

p-value *p* 值
packet 分组
paradox 悖论
  Allias Allias 悖论
  bus-stop 公交车站悖论
  heat capacity and fluctuations 热容及其波动悖论
paramagnetic, *see* Ising model 全磁性的，参见伊辛模型
paranormal 超出正常范围的
parasite 寄生虫
parent 父结点
parity 奇偶
parity-check bits 奇偶校验比特
parity-check constraints 奇偶校验约束
parity-check matrix 奇偶校验矩阵
  generalized 广义奇偶校验矩阵
parity-check nodes 奇偶校验结点
parse 解析
Parsons code Parsons 码
parthenogenesis 单性生殖
partial order 偏序
partial partition functions 部分划分函数
particle filter 粒子过滤器
partition 划分
partition function 划分函数
  analogy with lake 与湖作类比
partitioned inverse 分块逆
path-counting 路径计数
pattern recognition 模式识别
pentagonful code 五边形码
perfect code 完美码
perfect simulation 完美仿真
periodic variable 周期变量
permutation 置换
phase transition 相变
philosophy 哲学
phone 电话
  cellular, *see* mobile phone 蜂窝电话，参见移动电话
phone directory 电话簿
phone number 电话号码
photon counter 光子计数器
physics 物理学
pigeon-hole 鸽巢
pigeon-hole principle 鸽巢原理
pitchfork bifurcation 二分叉
plaintext 纯文本
plankton 浮游生物
point estimate 点估计
point spread function 点扩展函数
pointer 指针
Poisson distribution 泊松分布
Poisson process 泊松过程
polymer 聚合体
poor man's coding theorem 穷人编码定理
porridge 浓汁
positive definite 正定
positivity 阳性
posterior probability 后验概率
power cost 功率代价
power law 幂律
practical, *see* error-correcting code, practical 实际的，参见实际纠错码
precision 精度
precisions add 精度叠加
prediction 预测
predictive distribution 预测分布
prefix code 前缀码
prior 先验概率
  assigning 先验概率赋值
  improper 不当先验概率
  subjectivity 先验概率的主观性
prior equivalence 先验等价性
priority of bits in a message 消息中的比特优先级
prize, on game show 游戏秀中的奖品

probabilistic model 随机模型
probabilistic movie 随机动画
probability 概率
  Bayesian 贝叶斯概率
  contrasted with likelihood 概率与似然函数的对比
  density 概率密度
probability distributions, *see* distribution 概率分布，参见分布
probability of block error 分组差错概率
probability propagation, *see* sum-product algorithm 概率传播，参见和－积算法
product code 乘积码
profile, of random graph 随机图的文档
pronunciation 发音
proper 合适
proposal density 提议密度
prosecutor's fallacy 检举人的谬误
prospecting 勘探
protein 蛋白质
  regulatory 调节型蛋白质
protein synthesis 蛋白质合成
protocol 协议
pseudoinverse 伪逆
puncturing 删余
puzzle, *see* game 谜题，参见游戏
  cable labelling 电缆标示谜题
  chessboard 棋盘谜题
  fidelity of DNA replication DNA 复制保真谜题
  hat 帽子谜题
  life 生命谜题
  magic trick 魔术谜题
  poisoned glass 有毒玻璃杯谜题
  southeast 东南谜题
  transatlantic cable 跨大西洋电缆谜题
  weighing 12 balls 12 个球的称量谜题

quantum error-correction 量子纠错
queue 队列
QWERTY 标准的传统键盘

R_3, *see* repetition code R_3，参见重复码
race 竞赛
radial basis function 径向基函数
radio 比例
radix 基
RAID 独立磁盘冗余阵列
random 随机
random cluster model 随机聚类模型
random code 随机码
  for compression 用于压缩的随机码
random variable 随机变量
random walk 随机游动
  suppression 随机游动抑制
random-coding exponent 随机编码指数
random-walk Metropolis method 随机游动 Metropolis 法
rant, *see* sermon 激昂的谴责，参见训诫
  confidence level 置信水平
  p-value *p* 值
Raptor codes Raptor 码
rate 率
rate-distortion theory 信息率失真定理
reading aloud 大声朗读
receiver operating characteristic 接收机工作特性
recognition 识别
record breaking 打破纪录
rectangular code 矩形码
reducible 可约的
redundancy 冗余
  in channel code 信道码中的冗余
redundant array of independent disks 独立磁盘冗余阵列
redundant constraints in code 码的冗余约束
Reed－Solomon code Reed－Solomon 码
regression 回归
regret 遗憾
regular 正则的
regularization 正则化
regularization constant 正则化常数

reinforcement learning 加强学习
rejection 拒绝
rejection sampling, *see* Monte Carlo methods 拒绝采样,参见蒙特卡罗方法
relative entropy 相对熵
reliability function 可靠性函数
repeat-accumulate code 重复 - 累计码
  connection to low-density parity-check code 与低密度奇偶校验码的联系
repetition code 重复码
responsibility 责任
retransmission 重传
reverse 逆
reversible jump 可逆跳转
ROC 接收机工作特性
roman 罗马
rule of thumb 经验规则
runlength 游程
runlength-limited channel 有限游程信道

saccades 快速扫视
saddle-point approximation 鞍点近似
sample 样本
  from Gaussian 从高斯分布中采样
sampler density 样本密度
sampling distribution 采样分布
sampling theory 抽样理论
  criticisms 对抽样理论的批判
sandwiching method 三明治法
satellite 卫星
satellite communications 卫星通信
scaling 比例缩放
Schottky anomaly Schottky 异常
secret 秘密
secretary problem 秘书问题
security 安全
seek time 找寻时间
self-delimiting 自划界的
self-dual 自对偶的
self-orthogonal 自正交的
self-punctuating 自标断的
separation 分隔
sequence 序列
sequential decoding 顺序译码
sequential probability ratio test 依次概率比检验
sermon, *see* caution 训诫,参见告诫
  classical statistics 经典统计学
  confidence level 置信水平
  dimensions 维数
  gradient descent 梯度下降
  illegal integral 非法积分
  importance sampling 重要性采样
  interleaving 交织
  MAP method MAP 法
  maximum entropy 最大熵
  maximum likelihood 最大似然
  maximum a *posteriori* method 最大后验法
  most probable is atypical 最可能的是非典型的
  p-value *p* 值
  sampling theory 抽样理论
  sphere-packing 填球
  stopping rule 停止规则
  turbo codes Turbo 码
  unbiased estimator 无偏估计器
  worst-case-ism 从最坏处着想主义
set 集合
Shannon, *see* noisy-channel coding theorem, source coding theorem, informatio n content 香农,参见噪声信道编码定理,信源编码定理,信息量
shannon (unit) 香农(单位)
Shannon information content 香农信息量
shattering 脱离
shifter ensemble 移位总体
Shortening 缩短
Sigmoid S 形的
signal-to-noise ratio 信噪比
significance 重要性
significance level 重要性水平
simplex 单纯型
Simpson's paradox Simpson 悖论
Simpson, O. J., *see* wife-beaters O. J. Simpson,

参见殴妻者
simulated annealing, *see* annealing
模拟退火，参见退火
sleep 睡眠
Slepian – Wolf, *see* dependent sources
Slepian-Wolf，参见非独立信源
slice sampling, *see* Monte Carlo methods
切片采样，参见蒙特卡罗法
multi-dimensional 多维切片采样
soft K-means clustering 软 K – 均值聚类
software 软件
arithmetic coding 算术编码软件
BUGS · BUGS 软件
Dasher Dasher 软件
free 免费软件
Gaussian processes 高斯过程软件
hash function 散列函数软件
VIBES VIBES 软件
solar system 太阳系
soldier 士兵
soliton distribution 孤立子分布
sound 音响
source code 信源码
algorithms 信源码算法
block code 分组信源码
block-sorting compression 分组排序压缩信源码
Burrows-Wheeler transform Burrows – Wheeler 变换信源码
for complex sources 复杂信源的信源码
for constrained channel 受限信道的信源码
for integers 整数的信源码
Huffman, *see* Huffman code 霍夫曼信源码，参见霍夫曼码
implicit probabilities 隐概率信源码
optimal lengths 信源码的最优长度
prefix code 前缀信源码
software, *see* software 信源码软件，参见软件
stream codes 符号流码
supermarket 超市
symbol code 符号码
optimal 最优符号码
uniquely decodeable 单一可译码的符号码
variable symbol durations 可变符号周期的符号码
source coding theorem 信源编码定理
southeast puzzle 东南谜题
span 跨度
sparse graph 稀疏图
profile 文档
sparse-graph code 稀疏图码
density evolution 稀疏图码的密度进化
sparsifiers 稀疏器
species 物种
speculation about vision 关于视觉的思索
spell 拼写
sphere packing 填球
sphere-packing exponent 填球指数
spin system 自旋系统
spines 脊
spline 样条
spread spectrum 扩频
spring 弹簧
spy 刺探
square 正方形
staircase 阶梯
stalactite 钟乳石
standard deviation 标准差
stars 星
state diagram 状态图
statistic 统计量
sufficient 充分
statistical physics 统计物理
statistical test 统计检验
steepest descents 最速下降
stereoscopic vision 立体视觉
stiffness 硬度
Stirling's approximation Stirling 近似
stochastic 随机的
stochastic dynamics, *see* Hamiltonian Monte Carlo
随机动力学，参见哈密尔顿 – 蒙特卡罗
stochastic gradient 随机梯度

stop-when-it's-done 完成就停止
stopping rule 停止规则
straws, drawing 抽签
stream codes 符号流码
student 学生
Student-*t* distribution "学生"*t* 分布
subjective probability 主观概率
submarine 潜水艇
subscriber 用户
subset 子集
substring 子串
sufficient statistics 充分统计量
sum rule 求和规则
sum-product algorithm 和-积算法
summary 求和式
summary state 概括状态
summation convention 求和约定
super-channel 超信道
supermarket for codewords 码字超市
support vector 支撑向量
surprise value 惊奇值
survey propagation 概观传播
suspicious coincidences 可疑的巧合
symbol code 符号码
  budget 符号码预算
  codeword 符号码码字
  disadvantages 符号码的缺点
  optimal 最优符号码
  self-delimiting 自划界符号码
  supermarket 符号码超市
symmetric channel 对称信道
symmetry argument 对称性论据
synchronization 同步
synchronization errors 同步误差
syndrome 校正子
syndrome decoding 校正子译码
system 系统
systematic 系统化的

t-distribution, *see* Student-*t* *t* 分布,参见学生 *t* 分布
tail 拖尾
tamper detection 篡改检测
Tanner product code Tanner 乘积码
tap 抽头
telephone 电话
telephone directory 电话簿
telephone number 电话号码
telescope 望远镜
temperature 温度
termination 终止
terminology 术语
  Monte Carlo methods 蒙特卡罗方法
test 检验
  fluctuation 波动检验
  statistical 统计检验
text entry 文本项
thermal distribution 热分布
thermodynamic integration 热力学整合
thermodynamics 热力学
  third law 热力学第三定律
thin shell 薄层
third law of thermodynamics 热力学第三定理
three cards 3 张牌
three doors 3 扇门
threshold 阈值
tiling 镶嵌
time-division 时分
timing 定时
training data 训练数据
transatlantic 横跨大西洋
transfer matrix method 转移矩阵方法
transition 转移
transition probability matrix 转移概率矩阵
translation-invariant 平移不变的
travelling salesman problem 旅行推销员问题
tree 树
trellis 网格
termination 终止
trellis section 网格段
triangle 三角
truth function 真值函数

tube 细管
turbo code Turbo 码
turbo product code Turbo 乘积码
twenty questions 20 个问题
twin 孪生
twos 2
typical set 典型集合
  for compression 用于压缩的典型集合
  for noisy channel 噪声信道的典型集合
typical-set decoder 典型集合译码器
typicality 典型性
  behaviour of evidence 显著性的特性

umbrella sampling 伞状采样
unbiased estimator 无偏估计器
uncompression 解压缩
union 并
union bound 联合上界
uniquely decodeable 单一可译码的
units 单位
universal 通用的
universality 普适性
urn 缸
user interfaces 用户界面
utility 效用

vaccination 接种疫苗
Vapnik-Chervonenkis dimension Vapnik – Chervonenkis 维数
variable-length code 变长码
variable-rate error-correcting codes 变速率纠错码
variance 方差
variance-covariance matrix 方差 – 协方差矩阵
variances add 方差相加
variational Bayes 变参贝叶斯
variational free energy 变参自由能
  minimization 变参自由能的最小化
variational methods 变参法
  typical properties 变参法的典型特性
variational Gaussian process 变形高斯过程
  classifier 变参高斯过程分类器
VC dimension VC 维数
vector quantization 向量量化
very good, *see* error-correcting code, very good 很好的,参见很好的纠错码
VGC (variational Gaussian process classifier)
VGC(变形高斯过程分类器)
Viterbi algorithm Viterbi 算法
volume 体积
Von Mises distribution Von Mises 分布

waiting for a bus 等候公交车
warning, *see* caution 警告,参见告诫
Watson-Crick base pairing Watson-Crick 基配对
weather collator 天气核对员
weighing babies 称婴儿体重
weighing problem 称重问题
weight 权重
  importance sampling 重要性采样的权重
  in neural net 神经网络中的权重
  of binary vector 二进制向量的权重
weight decay 权重衰减
weight enumerator 权重枚举器
  typical 典型的权重枚举器
weight space 权空间
what number comes next? 接下来是哪个数?
white 白色的
white noise 白噪声
Wiener process Wiener 过程
wife-beater 殴妻者
window 窗户
word-English 英语单词
world record 世界纪录
worst-case-ism 从最坏处着想主义
writing 书写

Z channel Z 信道
Zipf plot Zipf 图
Zipf's law Zipf 定律

参考文献

ABRAHAMSEN, P. (1997) A review of Gaussian random fields and correlation functions. Technical Report 917, Norwegian Computing Center, Blindern, N - 0314 Oslo, Norway. 2nd edition.

ABRAMSON, N. (1963) *Information Theory and Coding*. McGraw-Hill.

ADLER, S. L. (1981) Over-relaxation method for the Monte-Carlo evaluation of the partition function for multiquadratic actions. *Physical Review D - Particles and Fields* **23** (12): 2901 - 2904.

AIYER, S. V. B. (1991) *Solving Combinatorial Optimization Problems Using Neural Networks*. Cambridge Univ. Engineering Dept. PhD dissertation. CUED/F-INFENG/TR 89.

AJI, S., JIN, H., KHANDEKAR, A., MCELIECE, R. J., and MACKAY, D. J. C. (2000) BSC thresholds for code ensembles based on 'typical pairs' decoding. In *Codes, Systems and Graphical Models*, ed. by B. Marcus and J. Rosenthal, volume 123 of *IMA Volumes in Mathematics and its Applications*, pp. 195 - 210. Springer.

AMARI, S., CICHOCKI, A., and YANG, H. H. (1996) A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems*, ed. by D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, volume 8, pp. 757 - 763. MIT Press.

AMIT, D. J., GUTFREUND, H., and SOMPOLINSKY, H. (1985) Storing infinite numbers of patterns in a spin glass model of neural networks. *Phys. Rev. Lett.* **55**: 1530 - 1533.

ANGEL, J. R. P., WIZINOWICH, P., LLOYD-HART, M., and SANDLER, D. (1990) Adaptive optics for array telescopes using neural-network techniques. *Nature* **348**: 221 - 224.

BAHL, L. R., COCKE, J., JELINEK, F., and RAVIV, J. (1974) Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Trans. Info. Theory* **IT - 20**: 284 - 287.

BALDWIN, J. (1896) A new factor in evolution. *American Naturalist* **30**: 441 - 451.

BAR-SHALOM, Y., and FORTMANN, T. (1988) *Tracking and Data Association*. Academic Press.

BARBER, D., and WILLIAMS, C. K. I. (1997) Gaussian processes for Bayesian classification via hybrid Monte Carlo. In *Neural Information Processing Systems 9*, ed. by M. C. Mozer, M. I. Jordan, and T. Petsche, pp. 340 - 346. MIT Press.

BARNETT, S. (1979) *Matrix Methods for Engineers and Scientists*. McGraw-Hill.

BATTAIL, G. (1993) We can think of good codes, and even decode them. In *Eurocode'92. Udine, Italy, 26 - 30 October*, ed. by P. Camion, P. Charpin, and S. Harari, number 339 in CISM Courses and Lectures, pp. 353 - 368. Springer.

BAUM, E., BONEH, D., and GARRETT, C. (1995) On genetic algorithms. In *Proc. Eighth Annual Conf. on Computational Learning Theory*, pp. 230 - 239. ACM.

BAUM, E. B., and SMITH, W. D. (1993) Best play for imperfect players and game tree search. Technical report, NEC, Princeton, NJ.

BAUM, E. B., and SMITH, W. D. (1997) A Bayesian approach to relevance in game

playing. *Artificial Intelligence* **97** (1 - 2): 195 - 242.

BAUM, L. E., and PETRIE, T. (1966) Statistical inference for probabilistic functions of finite-state Markov chains. *Ann. Math. Stat.* **37**: 1559 - 1563.

BEAL, M. J., GHAHRAMANI, Z., and RASMUSSEN, C. E. (2002) The infinite hidden Markov model. In *Advances in Neural Information Processing Systems 14*. MIT Press.

BELL, A. J., and SEJNOWSKI, T. J. (1995) An information maximization approach to blind separation and blind deconvolution. *Neural Computation* **7** (6): 1129 - 1159.

BENTLEY, J. (2000) *Programming Pearls*. Addison-Wesley, second edition.

BERGER, J. (1985) *Statistical Decision theory and Bayesian Analysis*. Springer.

BERLEKAMP, E. R. (1968) *Algebraic Coding Theory*. McGraw-Hill.

BERLEKAMP, E. R. (1980) The technology of error-correcting codes. *IEEE Trans. Info. Theory* **68**: 564 - 593.

BERLEKAMP, E. R., MCELIECE, R. J., and VAN TILBORG, H. C. A. (1978) On the intractability of certain coding problems. *IEEE Trans. Info. Theory* **24** (3): 384 - 386.

BERROU, C., and GLAVIEUX, A. (1996) Near optimum error correcting coding and decoding: Turbo-codes. *IEEE Trans. on Communications* **44**: 1261 - 1271.

BERROU, C., GLAVIEUX, A., and THITIMAJSHIMA. P. (1993) Near Shannon limit error-correcting coding and decoding: Turbo-codes. In *Proc. 1993 IEEE International Conf. on Communications, Geneva, Switzerland*, pp. 1064 - 1070.

BERZUINI, C., BEST, N. G., GILKS, W. R., and LARIZZA, C. (1997) Dynamic conditional independence models and Markov chain Monte Carlo methods. *J. American Statistical Assoc.* **92** (440): 1403 - 1412.

BERZUINI, C., and GILKS, W. R. (2001) Following a moving target - Monte Carlo inference for dynamic Bayesian models. *J. Royal Statistical Society Series B - Statistical Methodology* **63** (1). 127 - 146.

BHATTACHARYYA, A. (1943) On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc.* **35**: 99 - 110.

BISHOP, C. M. (1992) Exact calculation of the Hessian matrix for the multilayer perceptron. *Neural Computation* **4** (4): 494 - 501.

BISHOP, C. M. (1995) *Neural Networks for Pattern Recognition*. Oxford Univ. Press.

BISHOP, C. M., WINN, J. M., and SPIEGELHALTER, D. (2002) VIBES: A variational inference engine for Bayesian networks. In *Advances in Neural Information Processing Systems XV*, ed. by S. Becker, S. Thrun, and K. Obermayer.

BLAHUT, R. E. (1987) *Principles and Practice of Information Theory*. Addison-Wesley.

BOTTOU, L., HOWARD, P. G., and BENGIO, Y. (1998) The Z-coder adaptive binary coder. In *Proc. Data Compression Conf., Snowbird, Utah, March 1998*, pp. 13 - 22.

BOX, G. E. P., and TIAO, G. C. (1973) *Bayesian Inference in Statistical Analysis*. Addison-Wesley.

BRAUNSTEIN, A., MÉZARD, M., and ZECCHINA, R., (2003) Survey propagation: an algorithm for satisfiability. cs . CC/0212002.

BRETTHORST, G. (1988) *Bayesian Spectrum Analysis and Parameter Estimation*. Springer. Also available at bayes . wustl. edu.

BRIDLE, J. S. (1989) Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neuro-computing. Algorithms, Architectures and Applications*, ed. by F. Fougelman-

Soulie and J. Hérault. Springer – Verlag.

BULMER, M. (1985) *The Mathematical Theory of Quantitative Genetics.* Oxford Univ. Press.

BURROWS, M., and WHEELER, D. J. (1994) A block-sorting lossless data compression algorithm. Technical Report 124, Digital SRC.

BYERS, J., LUBY, M., MITZENMACHER, M., and REGE, A. (1998) A digital fountain approach to reliable distribution of bulk data. In *Proc. ACM SIGCOMM' 98, September 2 – 4, 1998.*

CAIRNS-SMITH, A. G. (1985) *Seven Clues to the Origin of Life.* Cambridge Univ. Press.

CALDERBANK, A. R., and SHOR, P. W. (1996) Good quantum error-correcting codes exist. *Phys. Rev. A* **54**: 1098. quant-ph/9512032.

CARROLL, L. (1998) *Alice's Adventures in Wonderland; and, Through the Looking-glass: and what Alice Found There.* Macmillan Children's Books.

CHILDS, A. M., PATTERSON, R. B., and MACKAY, D. J. C. (2001) Exact sampling from non-attractive distributions using summary states. *Physical Review E* **63**: 036113.

CHU, W., KEERTHI, S. S., and ONG, C. J. (2001) A unified loss function in Bayesian framework for support vector regression. In *Proc. 18th International Conf. on Machine Learning*, pp. 51 – 58.

CHU, W., KEERTHI, S. S., and ONG, C. J. (2002) A new Bayesian design method for support vector classification. In *Special Section on Support Vector Machines of the 9th International Conf. on Neural Information Processing.*

CHU, W., KEERTHI, S. S., and ONG, C. J. (2003a) Bayesian support vector regression using a unified loss function. *IEEE Trans. on Neural Networks.* Submitted.

CHU, W., KEERTHI, S. S., and ONG, C. J. (2003b) Bayesian trigonometric support vector classifier. *Neural Computation.*

CHUNG, S.-Y., RICHARDSON, T. J., and URBANKE, R. L. (2001) Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation. *IEEE Trans. Info. Theory* **47** (2): 657 – 670.

CHUNG, S.-Y., URBANKE, R. L., and RICHARDSON, T. J., (1999) LDPC code design applet. lids. mit. edu/ ~ sychung/ gaopt. html.

COMON, P., JUTTEN, C., and HERAULT, J. (1991) Blind separation of sources. 2. Problems statement. *Signal Processing* **24** (1): 11 – 20.

COPAS, J. B. (1983) Regression, prediction and shrinkage (with discussion). *J. R. Statist. Soc. B* **45** (3): 311 – 354.

COVER, T. M. (1965) Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Trans. on Electronic Computers* **14**: 326 – 334.

COVER, T. M., and THOMAS, J. A. (1991) *Elements of Information Theory.* Wiley.

COWLES, M. K., and CARLIN, B. P. (1996) Markov-chain Monte-Carlo convergence diagnostics – a comparative review. *J. American Statistical Assoc.* **91** (434): 883 – 904.

COX, R. (1946) Probability, frequency, and reasonable expectation. *Am. J. Physics* **14**: 1 – 13.

CRESSIE, N. (1993) *Statistics for Spatial Data.* Wiley.

DAVEY, M. C. (1999) *Error-correction using Low-Density Parity-Check Codes.* Univ. of Cambridge PhD dissertation.

DAVEY, M. C., and MACKAY, D. J. C. (1998) Low density parity check codes over GF(q). *IEEE Communications Letters* **2** (6): 165 – 167.

DAVEY, M. C., and MACKAY, D. J. C. (2000) Watermark codes: Reliable communi-

cation over insertion/deletion channels. In *Proc. 2000 IEEE International Symposium on Info. Theory*, p. 477.

DAVEY, M. C., and MACKAY, D. J. C. (2001) Reliable communication over channels with insertions, deletions and substitutions. *IEEE Trans. Info. Theory* **47** (2): 687 – 698.

DAWID, A., STONE, M., and ZIDEK, J. (1996) Critique of E. T Jaynes's 'paradoxes of probability theory'. Technical Report 172, Dept. of Statistical Science, Univ. College London.

DAYAN, P., HINTON, G. E., NEAL, R. M., and ZEMEL, R. S. (1995) The Helmholtz machine. *Neural Computation* **7** (5): 889 – 904.

DIVSALAR, D., JIN, H., and MCELIECE, R. J. (1998) Coding theorems for 'turbo-like' codes. In *Proc. 36th Allerton Conf. on Communication, Control, and Computing, Sept. 1998*, pp. 201 – 210. Allerton House.

DOUCET, A., DE FREITAS, J., and GORDON, N. eds. (2001) *Sequential Monte Carlo Methods in Practice.* Springer.

DUANE, S., KENNEDY, A. D., PENDLETON, B. J., and ROWETH, D. (1987) Hybrid Monte Carlo. *Physics Letters B* **195**: 216 – 222.

DURBIN, R., EDDY, S. R., KROGH, A., and MITCHISON, G. (1998) *Biological Sequence Analysis. Probabilistic Models of Proteins and Nucleic Acids.* Cambridge Univ. Press.

DYSON, F. J. (1985) *Origins of Life.* Cambridge Univ. Press.

ELIAS, P. (1975) Universal codeword sets and representations of the integers. *IEEE Trans. Info. Theory* **21** (2): 194 – 203.

EYRE-WALKER, A., and KEIGHTLEY, P. (1999) High genomic deleterious mutation rates in hominids. *Nature* **397**: 344 – 347.

FELSENSTEIN, J. (1985) Recombination and sex: is Maynard Smith necessary? In *Evolution. Essays in Honour of John Maynard Smith*, ed. by P. J. Greenwood, P. H. Harvey, and M. Slatkin, pp. 209 – 220. Cambridge Univ. Press.

FERREIRA, H., CLARKE, W., HELBERG, A., ABDEL-GHAFFAR, K. S., and VINCK, A. H. (1997) Insertion/deletion correction with spectral nulls. *IEEE Trans. Info. Theory* **43** (2): 722 – 732.

FEYNMAN, R. P. (1972) *Statistical Mechanics.* Addison – Wesley.

FORNEY, JR., G. D. (1966) *Concatenated Codes.* MIT Press.

FORNEY, JR., G. D. (2001) Codes on graphs: Normal realizations. *IEEE Trans. Info. Theory* **47** (2): 520 – 548.

FREY, B. J. (1998) *Graphical Models for Machine Learning and Digital Communication.* MIT Press.

GALLAGER, R. G. (1962) Low density parity check codes. *IRE Trans. Info. Theory* **IT-8**: 21 – 28.

GALLAGER, R. G. (1963) *Low Density Parity Check Codes.* Number 21 in MIT Research monograph series. MIT Press. Available from www. inference. phy. cam. ac. uk /mackay/ gallager / papers/.

GALLAGER, R. G. (1968) *Information Theory and Reliable Communication.* Wiley.

GALLAGER, R. G. (1978) Variations on a theme by Huffman. *IEEE Trans. Info. Theory* **IT – 24** (6): 668 – 674.

GIBBS, M. N. (1997) *Bayesian Gaussian Processes for Regression and Classification.* Cambridge Univ. PhD dissertation. www. inference. phy. cam. ac. uk/mng10/.

GIBBS, M. N., and MACKAY, D. J. C., (1996) Efficient implementation of Gaussian processes for interpolation. www. inference. phy. cam. ac. uk/mackay/abstracts /gpros. html.

GIBBS, M. N., and MACKAY, D. J. C. (2000) Variational Gaussian process classifiers. *IEEE Trans. on Neural Networks* 11 (6): 1458 – 1464.

GILKS, W., ROBERTS, G., and GEORGE, E. (1994) Adaptive direction sampling. *Statistician* **43**: 179 – 189.

GILKS, W., and WILD, P. (1992) Adaptive rejection sampling for Gibbs sampling. *Applied Statistics* **41**: 337 – 348.

GILKS, W. R., RICHARDSON, S., and SPIEGELHALTER, D. J. (1996) *Markov Chain Monte Carlo in Practice.* Chapman and Hall.

GOLDIE, C. M., and PINCH, R. G. E. (1991) *Communication theory.* Cambridge Univ. Press.

GOLOMB, S. W., PEILE, R. E., and SCHOLTZ, R. A. (1994) *Basic Concepts in Information Theory and Coding: The Adventures of Secret Agent 00111.* Plenum Press.

GOOD, I. J. (1979) Studies in the history of probability and statistics. XXXVII. A. M. Turing's statistical work in World War II. *Biometrika* **66** (2): 393 – 396.

GRAHAM, R. L. (1966) On partitions of a finite set. *Journal of Combinatorial Theory* **1**: 215 – 223.

GRAHAM, R. L., and KNOWLTON, K. C., (1968) Method of identifying conductors in a cable by establishing conductor connection groupings at both ends of the cable. U. S. Patent 3,369,177.

GREEN, P. J. (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82**: 711 – 732.

GREGORY, P. C., and LOREDO, T. J. (1992) A new method for the detection of a periodic signal of unknown shape and period. In *Maximum Entropy and Bayesian Methods*, ed. by G. Erickson and C. Smith. Kluwer. Also in *Astrophysical Journal*, **398**, pp. 146 – 168, Oct 10, 1992.

GULL, S. F. (1988) Bayesian inductive inference and maximum entropy. In *Maximum Entropy and Bayesian Methods in Science and Engineering, vol. 1: Foundations*, ed. by G. Erickson and C. Smith, pp. 53 – 74. Kluwer.

GULL, S. F. (1989) Developments in maximum entropy data analysis. In *Maximum Entropy and Bayesian Methods, Cambridge 1988*, ed. by J. Skilling, pp. 53 – 71. Kluwer.

GULL, S. F., and DANIELL, G. (1978) Image reconstruction from incomplete and noisy data. *Nature* **272**: 686 – 690.

HAMILTON, W. D. (2002) *Narrow Roads of Gene Land, Volume 2: Evolution of Sex.* Oxford Univ. Press.

HANSON, R., STUTZ, J., and CHEESEMAN, P. (1991a) Bayesian classification theory. Technical Report FIA – 90 – 12 – 7 – 01, NASA Ames.

HANSON, R., STUTZ, J., and CHEESEMAN, P. (1991b) Bayesian classification with correlation and inheritance. In *Proc. 12th Intern. Joint Conf. on Artificial Intelligence, Sydney, Australia*, volume 2, pp. 692 – 698. Morgan Kaufmann.

HARTMANN, C. R. P., and RUDOLPH, L. D. (1976) An optimum symbol by symbol decoding rule for linear codes. *IEEE Trans. Info. Theory* **IT** – 22: 514 – 517.

HARVEY, M., and NEAL, R. M. (2000) Inference for belief networks using coupling from the past. In *Uncertainty in Artificial Intelligence: Proc. Sixteenth Conf.*, pp. 256 – 263.

HEBB, D. O. (1949) *The Organization of Behavior.* Wiley.

HENDIN, O., HORN, D., and HOPFIELD, J. J. (1994) Decomposition of a mixture of signals in a model of the olfactory bulb. *Proc. Natl. Acad. Sci. USA* **91** (13): 5942 – 5946.

HERTZ, J., KROGH, A., and PALMER, R. G. (1991) *Introduction to the Theory of Neural Computation.* Addison-Wesley.

HINTON, G. (2001) Training products of experts by *minimizing contrastive divergence.* Technical Report 2000 - 004, Gatsby Computational Neuroscience Unit, Univ. College London.

HINTON, G., and NOWLAN, S. (1987) How learning can guide evolution. *Complex Systems* **1**: 495 - 502.

HINTON, G. E., DAYAN, P., FREY, B. J., and NEAL, R. M. (1995) The wake-sleep algorithm for unsupervised neural networks. *Science* **268** (5214): 1158 - 1161.

HINTON, G. E., and GHAHRAMANI, Z. (1997) Generative models for discovering sparse distributed representations. *Philosophical Trans. Royal Society B.*

HINTON, G. E., and SEJNOWSKI, T. J. (1986) Learning and relearning in Boltzmann machints. In *Parallel Distributed Processing*, ed. by D. E. Rumelhart and J. E. McClelland, pp. 282 - 317. MIT Press.

HINTON, G. E., and TEH, Y. W. (2001) Discovering multiple constraints that are frequently approximately satisfied. In *Uncertainty in Artificial Intelligence: Proc. Seventeenth Conf. (UAI - 2001)*, pp. 227 - 234. Morgan Kaufmann.

HINTON, G. E., and VAN CAMP, D. (1993) Keeping neural networks simple by minimizing the description length of the weights. In *Proc. 6th Annual Workshop on Comput. Learning Theory*, pp. 5 - 13. ACM Press, New York, NY.

HINTON, G. E., WELLING, M., TEH, Y. W., and OSINDERO, S. (2001) A new view of ICA. In *Proc. International Conf. on Independent Component Analysis and Blind Signal Separation*, volume 3.

HINTON, G, E., and ZEMEL, R. S. (1994) Autoencoders, minimum description length and Helmholtz free energy. In *Advances in Neural Information Processing Systems 6*, ed. by J. D. Cowan, G. Tesauro, and J. Alspector. Morgan Kaufmann.

HODGES, A. (1983) *Alan Turing: The Enigma.* Simon and Schuster.

HOJEN-SORENSEN, P. A., WINTHER, O., and HANSEN, L. K. (2002) Mean field approaches to independent component analysis. *Neural Computation* **14**: 889 - 918.

HOLMES, C., and DENISON, D. (2002) Perfect sampling for wavelet reconstruction of signals. *IEEE Trans. Signal Processing* **50**: 237 - 244.

HOLMES, C., and MALLICK, B. (1998) Perfect simulation for orthogonal model mixing. Technical report, Imperial College, London.

HOPFIELD, J. J. (1974) Kinetic proofreading: A new mechanism for reducing errors in biosynthetic processes requiring high specificity. *Proc. Natl. Acad. Sci. USA* **71** (10): 4135 - 4139.

HOPFIELD, J. J. (1978) Origin of the genetic code: A testable hypothesis based on tRNA structure, sequence, and kinetic proofreading. *Proc. Natl. Acad. Sci. USA* **75** (9): 4334 - 4338.

HOPFIELD, J. J. (1980) The energy relay: A proofreading scheme based on dynamic cooperativity and lacking all characteristic symptoms of kinetic proofreading in DNA replication and protein synthesis. *Proc. Natl. Acad. Sci. USA* **77** (9): 5248 - 5252.

HOPFIELD, J. J. (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79**: 2554 - 8.

HOPFIELD, J. J. (1984) Neurons with graded response properties have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA* **81**: 3088 - 92.

HOPFIELD, J. J. (1987) Learning algorithms

and probability distributions in feed-forward and feed-back networks. *Proc. Natl. Acad. Sci. USA* **84**: 8429 - 33.

HOPFIELD, J. J., and BRODY, C. D. (2000) What is a moment? "Cortical" sensory integration over a brief interval. *Proc. Natl. Acad. Sci* **97**: 13919 - 13924.

HOPFIELD, J. J., and BRODY, C. D. (2001) What is a moment? Transient synchrony as a collective mechanism for spatiotemporal integration. *Proc. Natl. Acad. Sci* **98**: 1282 - 1287.

HOPFIELD, J. J., and TANK, D. W. (1985) Neural computation of decisions in optimization problems. *Biol. Cybernetics* **52**: 1 - 25.

HOWARTH, P., and BRADLEY, A. (1986) The longitudinal aberration of the human eye and its correction. *Vision Res.* **26**: 361 - 366.

HUBER, M. (1998) Exact sampling and approximate counting techniques. In *Proc. 30th ACM Symposium on the Theory of Computing*, pp. 31 - 40.

ICHIKAWA, K., BHADESHIA, H. K. D. H., and MACKAY, D. J. C. (1996) Model for hot cracking in low-alloy steel weld metals. *Science and Technology of Welding and Joining* **1**: 43 - 50.

ISARD, M., and BLAKE, A. (1996) Visual tracking by stochastic propagation of conditional density. In *Proc. Fourth European Conf. Computer Vision*, pp. 343 - 356.

ISARD, M., and BLAKE, A. (1998) Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision* **29** (1): 5 - 28.

JAAKKOLA, T. S., and JORDAN, M. I. (1996) Computing upper and lower bounds on likelihoods in intractable networks. In *Proc. Twelfth Conf. on Uncertainty in AI*. Morgan Kaufmann.

JAAKKOLA, T. S., and JORDAN, M. I. (2000a) Bayesian logistic regression: a variational approach. *Statistics and Computing* **10**: 25 - 37.

JAAKKOLA, T. S., and JORDAN, M. I. (2000b) Bayesian parameter estimation via variational methods. *Statistics and Computing* **10** (1): 25 - 37.

JAYNES, E. T. (1983) Bayesian intervals versus confidence intervals. In *E. T. Jaynes. Papers on Probability, Statistics and Statistical Physics*, ed. by R. D. Rosenkrantz, p. 151. Kluwer.

JAYNES, E. T. (2003) *Probability Theory: The Logic of Science*. Cambridge Univ. Press. Edited by G. Larry Bretthorst.

JENSEN, F. V. (1996) *An Introduction to Bayesian Networks*. UCL press.

JOHANNESSON, R., and ZIGANGIROV, K. S. (1999) *Fundamentals of Convolutional Coding*. IEEE Press.

JORDAN, M. I. ed. (1998) *Learning in Graphical Models*. NATO Science Series. Kluwer Academic Publishers.

JPL, (1996) Turbo codes performance. Available from www331. jpl. nasa. gov/public / TurboPerf. html.

JUTTEN, C., and HERAULT, J. (1991) Blind separation of sources. 1. An adaptive algorithm based on neuromimetic architecture. *Signal Processing* **24** (1): 1 - 10.

KARPLUS, K., and KRIT, H. (1991) A semi-systolic decoder for the PDSC - 73 error-correcting code. *Discrete Applied Mathematics* **33**: 109 - 128.

KEPLER, T., and OPREA, M. (2001) Improved inference of mutation rates: I. An integral representation of the Luria-Delbrück distribution. *Theoretical Population Biology* **59**: 41 - 48.

KIMELDORF, G. S., and WAHBA, G. (1970) A correspondence between Bayesian estimation of stochastic processes and smoothing by splines. *Annals of Math. Statistics* **41** (2): 495 - 502.

KITANIDIS, P. K. (1986) Parameter uncertainty in estimation of spatial functions: Bayesian analysis. *Water Resources Research* **22**. 499 - 507.

KNUTH, D. E. (1968) *The Art of Computer Programming*. Addison Wesley.

KONDRASHOV, A. S. (1988) Deleterious mutations and the evolution of sexual reproduction. *Nature* **336** (6198): 435 - 440.

KSCHISCHANG, F. R., FREY, B. J., and LOELIGER, H.-A. (2001) Factor graphs and the sum-product algorithm. *IEEE Trans. Info. Theory* **47** (2): 498 - 519.

KSCHISCHANG, F. R., and SOROKINE, V. (1995) On the trellis structure of block codes. *IEEE Trans. Info. Theory* **41** (6): 1924 - 1937.

LAURITZEN, S. L. (1981) Time series analysis in 1880, a discussion of contributions made by T. N. Thiele. *ISI Review* **49**: 319 - 333.

LAURITZEN, S. L. (1996) *Graphical Models*. Number 17 in Oxford Statistical Science Series. Clarendon Press.

LAURITZEN, S. L., and SPIEGELHALTER, D. J. (1988) Local computations with probabilities on graphical structures and their application to expert systems. *J. Royal Statistical Society B* **50**: 157 - 224.

LEVENSHTEIN, V. I. (1966) Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics - Doklady* **10** (8): 707 - 710.

LIN, S., and COSTELLO, JR., D. J. (1983) *Error Control Coding: Fundamentals and Applications*. Prentice-Hall.

LITSYN, S., and SHEVELEV, V. (2002) On ensembles of low-density parity-check codes: asymptotic distance distributions. *IEEE Trans. Info. Theory* **48** (4): 887 - 908.

LOREDO, T. J. (1990) From Laplace to supernova SN 1987A: Bayesian inference in astrophysics. In *Maximum Entropy and Bayesian Methods, Dartmouth, U. S. A., 1989*, ed. by P. Fougere, pp. 81 - 142. Kluwer.

LOWE, D. G. (1995) Similarity metric learning for a variable kernel classifier. *Neural Computation* **7**: 72 - 85.

LUBY, M. (2002) LT codes. In *Proc. The 43rd Annual IEEE Symposium on Foundations of Computer Science, November 16 - 19 2002*, pp. 271 - 282.

LUBY, M. G., MITZENMACHER, M., SHOKROLLAHI, M. A., and SPIELMAN, D. A. (1998) Improved low-density parity-check codes using irregular graphs and belief propagation. In *Proc. IEEE International Symposium on Info. Theory*, p. 117.

LUBY, M. G., MITZENMACHER, M., SHOKROLLAHI, M. A., and SPIELMAN, D. A. (2001a) Efficient erasure correcting codes. *IEEE Trans. Info. Theory* **47** (2): 569 - 584.

LUBY, M. G., MITZENMACHER, M., SHOKROLLAHI, M. A., and SPIELMAN, D. A. (2001b) Improved low-density parity-check codes using irregular graphs and belief propagation. *IEEE Trans. Info. Theory* **47** (2): 585 - 598.

LUBY, M. G., MITZENMACHER, M., SHOKROLLAHI, M. A., SPIELMAN, D. A., and STEMANN, V. (1997) Practical loss-resilient codes. In *Proc. Twenty-Ninth Annual ACM Symposium on Theory of Computing (STOC)*.

LUO, Z., and WAHBA, G. (1997) Hybrid adaptive splines. *J. Amer. Statist. Assoc.* **92**: 107 - 116.

LURIA, S. E., and DELBRÜCK, M. (1943) Mutations of bacteria from virus sensitivity to virus resistance. *Genetics* **28**: 491 - 511. Reprinted in *Microbiology: A Centenary Perspective*, Wolfgang K. Joklik, ed., 1999, ASM Press, and available from www. esp. org/.

LUTTRELL, S. P. (1989) Hierarchical vector quantisation. *Proc. IEE Part I* **136**: 405 - 413.

LUTTRELL, S. P. (1990) Derivation of a class of training algorithms. *IEEE Trans. on Neural Networks* **1** (2): 229 - 232.

MACKAY, D. J. C. (1991) *Bayesian Methods for Adaptive Models.* California institute of Technology PhD dissertation.

MACKAY. D. J. C. (1992a) Bayesian interpolation. *Neural Computation* **4** (3): 415 - 447.

MACKAY, D. J. C. (1992b) The evidence framework applied to classification networks. *Neural Computation* **4** (5): 698 - 714.

MACKAY, D. J. C. (1992c) A practical Bayesian framework for backpropagation networks. *Neural Computation* **4** (3): 448 - 472.

MACKAY, D. J. C. (1994a) Bayesian methods for backpropagation networks. In *Models of Neural Networks III*, ed. by E. Domany, J. L. van Hemmen, and K. Schulten, chapter 6, pp. 211 - 254. Springer.

MACKAY, D. J. C. (1994b) Bayesian non-linear modelling for the prediction competition. In *ASHRAE Trans., V. 100, Pt. 2*, pp. 1053 - 1062. American Society of Heating, Refrigeration, and Air-conditioning Engineers.

MACKAY, D. J. C. (1995a) Free energy minimization algorithm for decoding and cryptanalysis. *Electronics Letters* **31** (6): 446 - 447.

MACKAY, D. J. C. (1995b) Probable networks and plausible predictions - a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems* **6**: 469 - 505.

MACKAY, D. J. C., (1997a) Ensemble learning for hidden Markov models. www. inference. phy. cam. ac. uk/mackay/abstracts/ensemblePaper. html.

MACKAY, D. J. C., (1997b) Iterative probabilistic decoding of low density parity check codes. Animations available on world wide web. www. inference. phy. cam. ac. uk/mackay/codes/gifs/.

MACKAY, D. J. C. (1998a) Choice of basis for Laplace approximation. *Machine Learning* **33** (1): 77 - 86.

MACKAY, D. J. C. (1998b) Introduction to Gaussian processes. In *Neural Networks and Machine Learning*, ed. by C. M. Bishop, NATO ASI Series, pp. 133 - 166. Kluwer.

MACKAY, D. J. C. (1999a) Comparison of approximate methods for handling hyperparameters. *Neural Computation* **11** (5): 1035 - 1068.

MACKAY, D. J. C. (1999b) Good error correcting codes based on very sparse matrices. *IEEE Trans. Info. Theory* 45 (2): 399 - 431.

MACKAY, D. J. C., (2000) An alternative to runlength-limiting codes: Turn timing errors into substitution errors. Available from www. inference. phy. cam. ac. uk/mackay/.

MACKAY, D. J. C., (2001) A problem with variational free energy minimization. www. inference. phy. cam. ac. uk/mackay/abstracts/minima. html.

MACKAY, D. J. C., and DAVEY, M. C. (2000) Evaluation of Gallager codes for short block length and high rate applications. In *Codes, Systems and Graphical Models*, ed. by B. Marcus and J. Rosenthal, volume 123 of *IMA Volumes in Mathematics and its Applications*, pp. 113 - 130. Springer.

MACKAY, D. J. C., MITCHISON, G. J., and MCFADDEN, P. L., (2003) Sparse-graph codes for quantum error-correction. quant-ph/0304161. Submitted to *IEEE Trans. Info. Theory* May 8, 2003.

MACKAY, D. J. C., and NEAL, R. M. (1995) Good codes based on very sparse matrices. In *Cryptography and Coding. 5th IMA Conf, LNCS* 1025, ed. by C. Boyd, pp. 100 - 111. Springer.

MACKAY, D. J. C., and NEAL, R. M. (1996) Near Shannon limit performance of low density

parity check codes, *Electronics Letters* **32** (18): 1645 – 1646. Reprinted *Electronics Letters*, **33**(6). 457 – 458, March 1997.

MACKAY, D. J. C., and PETO, L. (1995) A hierarchical Dirichlet language model. *Natural Language Engineering* **1** (3): 1 – 19.

MACKAY, D. J. C., WILSON, S. T., and DAVEY, M. C. (1998) Comparison of constructions of irregular Gallager codes. In *Proc. 36th Allerton Conf. on Communication, Control, and Computing, Sept. 1998*, pp. 220 – 229. Allerton House.

MACKAY, D. J. C., WILSON, S. T., and DAVEY, M. C. (1999) Comparison of constructions of irregular Gallager codes. *IEEE Trans. on Communications* **47** (10): 1449 – 1454.

MACKAY, D. M., and MACKAY, V. (1974) The time course of the McCollough effect and its physiological implications. *J. Physiol.* **237**: 38 – 39.

MACKAY, D. M., and MCCULLOCH, W. S. (1952) The limiting information capacity of a neuronal link. *Bull. Math. Biophys.* **14**: 127 – 135.

MACWILLIAMS, F. J., and SLOANE, N. J. A. (1977) *The Theory of Error-correcting Codes.* North-Holland.

MANDELBROT, B. (1982) *The Fractal Geometry of Nature.* W. H. Freeman.

MAO, Y., and BANIHASHEMI, A. (2000) Design of good LDPC codes using girth distribution. In *IEEE International Symposium on Info. Theory, Italy, June, 2000.*

MAO, Y., and BANIHASHEMI, A. (2001) A heuristic search for good LDPC codes at short block lengths. In *IEEE International Conf. on Communications.*

MARINARI, E., and PARISI, G. (1992) Simulated tempering – a new Monte-Carlo scheme. *Europhysics Letters* **19** (6): 451 – 458.

MATHERON, G. (1963) Principles of geostatistics. *Economic Geology* **58**: 1246 – 1266.

MAYNARD SMITH, J. (1968) 'Haldane's dilemma' and the rate of evolution. *Nature* **219** (5159): 1114 – 1116.

MAYNARD SMITH, J. (1978) *The Evolution of Sex.* Cambridge Univ. Press.

MAYNARD SMITH, J. (1988) *Games, Sex and Evolution.* Harvester – Wheatsheaf.

MAYNARD SMITH, J., and SZÁTHMARY, E. (1995) *The Major Transitions in Evolution.* Freeman.

MAYNARD SMITH, J., and SZÁTHMARY, E. (1999) *The Origins of Life.* Oxford Univ. Press.

MCCOLLOUGH, C. (1965) Color adaptation of edge-detectors in the human visual system. *Science* **149**: 1115 – 1116.

MCELIECE, R. J. (2002) *The Theory of Information and Coding.* Cambridge Univ. Press, second edition.

MCELIECE, R. J., MACKAY, D. J. C., and CHENG, J.-F. (1998) Turbo decoding as an instance of Pearl's 'belief propagation' algorithm. *IEEE Journal on Selected Areas in Communications* **16** (2): 140 – 152.

MCMILLAN, B. (1956) Two inequalities implied by unique decipherability. *IRE Trans. Inform. Theory* **2**: 115 – 116.

MINKA, T. (2001) *A family of algorithms for approximate Bayesian inference.* MIT PhD dissertation.

MISKIN, J. W. (2001) *Ensemble Learning for Independent Component Analysis.* Dept. of Physics, Univ. of Cambridge PhD dissertation.

MISKIN, J. W., and MACKAY, D. J. C. (2000) Ensemble learning for blind image separation and deconvolution. In *Advances in Independent Component Analysis*, ed. by M. Girolami. Springer.

MISKIN, J. W., and MACKAY, D. J. C.

(2001) Ensemble learning for blind source separation. In *ICA: Principles and Practice*, ed. by S. Roberts and R. Everson. Cambridge Univ. Press.

MOSTELLER, F., and WALLACE, D. L. (1984) *Applied Bayesian and Classical Inference. The case of* The Federalist *papers*. Springer.

NEAL, R. M. (1991) Bayesian mixture modelling by Monte Carlo simulation. Technical Report CRG - TR - 91 - 2, Computer Science, Univ. of Toronto.

NEAL, R. M. (1993a) Bayesian learning via stochastic dynamics. In *Advances in Neural Information Processing Systems* 5, ed. by C. L. Giles, S. J. Hanson, and J. D. Cowan, pp. 475 - 482. Morgan Kaufmann.

NEAL, R. M. (1993b) Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG - TR - 93 - 1, Dept. of Computer Science, Univ. of Toronto.

NEAL, R. M. (1995) Suppressing random walks in Markov chain Monte Carlo using ordered. overrelaxation. Technical Report 9508, Dept. of Statistics, Univ. of Toronto.

NEAL, R. M. (1996) *Bayesian Learning for Neural Networks*. Springer.

NEAL, R. M. (1997a) Markov chain Monte Carlo methods based on 'slicing' the density function. Technical Report 9722, Dept. of Statistics, Univ. of Toronto.

NEAL, R. M. (1997b) Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical Report CRG - TR - 97 - 2, Dept. of Computer Science, Univ. of Toronto.

NEAL, R. M. (1998) Annealed importance sampling. Technical Report 9805, Dept. of Statistics, Univ. of Toronto.

NEAL, R. M. (2001) Defining priors for distributions using Dirichlet diffusion trees. Technical Report 0104, Dept. of Statistics, Univ. of Toronto.

NEAL, R. M. (2003) Slice sampling. *Annals of Statistics* **31**(3): 705 - 767.

NEAL, R. M., and HINTON, G. E. (1998) A new view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, ed. by M. I. Jordan, NATO Science Series, pp. 355 - 368. Kluwer.

NIELSEN, M., and CHUANG, I. (2000) *Quantum Computation and Quantum Information*. Cambridge Univ. Press.

OFFER, E., and SOLJANIN, E. (2000) An algebraic description of iterative decoding schemes. In *Codes, Systems and Graphical Models*, ed. by B. Marcus and J. Rosenthal, volume 123 of *IMA Volumes in Mathematics and its Applications*, pp. 283 - 298. Springer.

OFFER, E., and SOLJANIN, E. (2001) LDPC codes: a group algebra formulation. In *Proc. Internat. Workshop on Coding and Cryptography WCC 2001, 8 - 12 Jan. 2001, Paris.*

O'HAGAN, A. (1978) On curve fitting and optimal design for regression. *J. Royal Statistical Society, B* **40**: 1 - 42.

O'HAGAN, A. (1987) Monte Carlo is fundamentally unsound. *The Statistician* **36**: 247 - 249.

O'HAGAN, A. (1994) *Bayesian Inference*, volume 2B of *Kendall's Advanced Theory of Statistics*. Edward Arnold.

OMRE, H. (1987) Bayesian kriging - merging observations and qualified guesses in kriging. *Mathematical Geology* **19**: 25 - 39.

OPPER, M., and WINTHER, O. (2000) Gaussian processes for classification: Mean-field algorithms. *Neural Computation* **12** (11): 2655 - 2684.

PATRICK, J. D., and WALLACE, C. S. (1982) Stone circle geometries: an information theory approach. In *Archaeoastronomy in the*

Old World, ed. by D. C. Heggie, pp. 231 – 264. Cambridge Univ. Press.

PEARL, J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

PEARL, J. (2000) *Causality*. Cambridge Univ. Press.

PEARLMUTTER, B. A., and PARRA, L. C. (1996) A context-sensitive generalization of ICA. In. *International Conf. on Neural Information Processing, Hong Kong*, pp. 151 – 157.

PEARLMUTTER, B. A., and PARRA, L. C. (1997) Maximum likelihood blind source separation: A context-sensitive generalization of ICA. In *Advances in Neural Information Processing Systems*, ed. by M. C. Mozer, M. I. Jordan, and T. Petsche, volume 9, p. 613. MIT Press.

PINTO, R. L., and NEAL, R. M. (2001) Improving Markov chain Monte Carlo estimators by coupling to an approximating chain. Technical Report 0101, Dept. of Statistics, Univ. of Toronto.

POGGIO, T., and GIROSI, F. (1989) A theory of networks for approximation and learning. Technical Report A. I. 1140, MIT.

POGGIO, T., and GIROSI, F. (1990) Networks for approximation and learning. *Proc. IEEE* **78**: 1481 – 1497.

POLYA, G. (1954) *Induction and Analogy in Mathematics*. Princeton Univ. Press.

PROPP, J. G., and WILSON, D. B. (1996) Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms* **9** (1 – 2): 223 – 252.

RABINER, L. R., and JUANG, B. H. (1986) An introduction to hidden Markov models. *IEEE ASSP Magazine* pp. 4 – 16.

RASMUSSEN, C. E. (1996) *Evaluation of Gaussian Processes and Other Methods for Non-Linear Regression*. Univ. of Toronto PhD dissertation.

RASMUSSEN, C. E. (2000) The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems 12*, ed. by S. Solla, T. Leen, and K.-R. Müller, pp. 554 – 560. MIT Press.

RASMUSSEN, C. E., (2002) Reduced rank Gaussian process learning. Unpublished manuscript.

RASMUSSEN, C. E., and GHAHRAMANI, Z. (2002) Infinite mixtures of Gaussian process experts. In *Advances in Neural Information Processing Systems 14*, ed. by T. G. Diettrich, S. Becker, and Z. Ghahramani. MIT Press.

RASMUSSEN, C. E., and GHAHRAMANI, Z. (2003) Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems XV*, ed. by S. Becker, S. Thrun, and K. Obermayer.

RATLIFF, F., and RIGGS, L. A. (1950) Involuntary motions of the eye during monocular fixation. *J. Exptl. Psychol.* **40**:687 – 701.

RATZER, E. A., and MACKAY, D. J. C. (2003) Sparse low-density parity-check codes for channels with cross-talk. In *Proc.* 2003 *IEEE Info. Theory Workshop, Paris.*

REIF, F. (1965) *Fundamentals of Statistical and Thermal Physics*. McGraw-Hill.

RICHARDSON, T., SHOKROLLAHI, M. A., and URBANKE, R. (2001) Design of capacity-approaching irregular low-density parity check codes. *IEEE Trans. Info. Theory* **47** (2): 619 – 637.

RICHARDSON, T., and URBANKE, R. (2001a) The capacity of low-density parity check codes under message-passing decoding. *IEEE Trans. Info. Theory* **47** (2): 599 – 618.

RICHARDSON, T., and URBANKE, R. (2001b) Efficient encoding of low-density parity-check codes. *IEEE Trans. Info. Theory* **47** (2): 638 – 656.

RIDLEY, M. (2000) *Mendel's Demon: gene justice and the complexity of life.* Phoenix.

RIPLEY, B. D. (1991) *Statistical Inference for Spatial Processes.* Cambridge Univ. Press.

RIPLEY, B. D. (1996) *Pattern Recognition and Neural Networks.* Cambridge Univ. Press.

RUMELHART, D. E., HINTON, G. E., and WILLIAMS, R. J. (1986) Learning representations by back-propagating errors. *Nature* **323**: 533 – 536.

RUSSELL, S., and WEFALD, E. (1991) *Do the Right Thing: Studies in Limited Rationality.* MIT Press.

SCHNEIER, B. (1996) *Applied Cryptography.* Wiley.

SCHOLKOPF, B., BURGES, C., and VAPNIK, V. (1995) Extracting support data for a given task. In *Proc. First International Conf. on Knowledge Discovery and Data Mining*, ed. by U. M. Fayyad and R. Uthurusamy. AAAI Press.

SCHOLTZ, R. A. (1982) The origins of spread-spectrum communications. *IEEE Trans. on Communications* **30** (5): 822 – 854.

SEEGER, M., WILLIAMS, C. K. I., and LAWRENCE, N. (2003) Fast forward selection to speed up sparse Gaussian process regression. In *Proc. Ninth International Workshop on Artificial Intelligence and Statistics*, ed. by C. Bishop and B. J. Frey. Society for Artificial Intelligence and Statistics.

SEJNOWSKI, T. J. (1986) Higher order Boltzmann machines. In *Neural networks for computing*, ed. by J. Denker, pp. 398 – 403. American Institute of Physics.

SEJNOWSKI, T. J., and ROSENBERG, C. R. (1987) Parallel networks that learn to pronounce English text. *Journal of Complex Systems* **1** (1): 145 – 168.

SHANNON, C. E. (1948) A mathematical theory of communication. *Bell Sys. Tech. J.* **27**: 379 – 423, 623 – 656.

SHANNON, C. E. (1993) The best detection of pulses. In *Collected Papers of Claude Shannon*, ed. by N. J. A. Sloane and A. D. Wyner, pp. 148 – 150. IEEE Press.

SHANNON, C. E., and WEAVER, W. (1949) *The Mathematical Theory of Communication.* Univ. of Illinois Press.

SHOKROLLAHI, A. (2003) Raptor codes. Technical report, Laboratoire d'algorithmique, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. Available from algo. epfl. ch/.

SIPSER, M., and SPIELMAN, D. A. (1996) Expander codes. *IEEE Trans. Info. Theory* **42** (6.1): 1710 – 1722.

SKILLING, J. (1989) Classic maximum entropy. In *Maximum Entropy and Bayesian Methods, Cambridge* 1988, ed. by J. Skilling. Kluwer.

SKILLING, J. (1993) Bayesian numerical analysis. In *Physics and Probability*, ed. by W. T. Grandy, Jr. and P. Milonni. Cambridge Univ. Press.

SKILLING, J., and MACKAY, D. J. C. (2003) Slice sampling – a binary implementation. *Annals of Statistics* **31** (3): 753 – 755. Discussion of *Slice Sampling* by Radford M. Neal.

SLEPIAN, D., and WOLF, J. (1973) Noiseless coding of correlated information sources. *IEEE Trans. Info. Theory* **19**: 471 – 480.

SMOLA, A. J., and BARTLETT, P. (2001) Sparse Greedy Gaussian Process Regression. In *Advances in Neural Information Processing Systems 13*, ed. by T. K. Leen, T. G. Diettrich, and V. Tresp, pp. 619 – 625. MIT Press.

SPIEGEL, M. R. (1988) *Statistics.* Schaum's outline series. McGraw-Hill, 2nd edition.

SPIELMAN, D. A. (1996) Linear-time encodable and decodable error-correcting codes. *IEEE Trans. Info. Theory* **42** (6.1): 1723 – 1731.

SUTTON, R. S., and BARTO, A. G. (1998)

Reinforcement Learning: An Introduction. MIT Press.

SWANSON, L. (1988) A new code for Galileo. In *Proc.* 1988 *IEEE International Symposium Info. Theory*, pp. 94 – 95.

TANNER, M. A. (1996) *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions.* Springer Series in Statistics. Springer, 3rd edition.

TANNER, R. M. (1981) A recursive approach to low complexity codes. *IEEE Trans. Info. Theory* **27** (5): 533 – 547.

TEAHAN, W. J. (1995) Probability estimation for PPM. In *Proc. NZCSRSC'95*. Available from citeseer. nj. nec. com/teahan95probability. html.

TEN BRINK, S. (1999) Convergence of iterative decoding. *Electronics Letters* **35** (10): 806 – 808.

TEN BRINK, S., KRAMER, G., and ASHIKHMIN, A., (2002) Design of low-density parity-check codes for multi-antenna modulation and detection. Submitted to *IEEE Trans. on Communications.*

TERRAS, A. (1999) *Fourier Analysis on Finite Groups and Applications.* Cambridge Univ. Press.

THOMAS, A., SPIEGELHALTER, D. J., and GILKS, W. R. (1992) BUGS: A program to perform Bayesian inference using Gibbs sampling. In *Bayesian Statistics 4*, ed. by J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, pp. 837 – 842. Clarendon Press.

TRESP, V. (2000) A Bayesian committee machine. *Neural Computation* **12** (11): 2719 – 2741.

URBANKE, R., (2001) LdpcOpt – a fast and accurate degree distribution optimizer for LDPC code ensembles. lthcwww. epfl. ch/research/ldpcopt/.

VAPNIK, V. (1995) *The Nature of Statistical Learning Theory.* Springer.

VITERBI, A. J. (1967) Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Info. Theory* **IT** – *13*: 260 – 269.

WAHBA, G. (1990) *Spline Models for Observational Data.* Society for Industrial and Applied Mathematics. CBMS-NSF Regional Conf. series in applied mathematics.

WAINWRIGHT, M. J., JAAKKOLA, T., and WILLSKY, A. S. (2003) Tree-based reparameterization framework for analysis of sum-product and related algorithms. *IEEE Trans. Info. Theory* **45**(9): 1120 – 1146.

WALD, G., and GRIFFIN, D. (1947) The change in refractive power of the eye in bright and dim light. *J. Opt. Soc. Am.* **37**: 321 – 336.

WALLACE, C., and BOULTON, D. (1968) An information measure for classification. *Comput. J.* **11** (2): 185 – 194.

WALLACE, C. S., and FREEMAN, P. R. (1987) Estimation and inference by compact coding. *J. R. Statist. Soc. B* **49** (3): 240 – 265.

WARD, D. J., BLACKWELL, A. F., and MACKAY, D. J. C. (2000) Dasher – A data entry interface using continuous gestures and language models. In *Proc. User Interface Software and Technology 2000*, pp. 129 – 137.

WARD, D. J., and MACKAY, D. J. C. (2002) Fast hands-free writing by gaze direction. *Nature* **418** (6900): 838.

WELCH, T. A. (1984) A technique for high-performance data compression. *IEEE Computer* **17** (6): 8 – 19.

WELLING, M., and TEH, Y. W. (2001) Belief optimization for binary networks: A stable alternative to loopy belief propagation. In *Uncertainty in Artificial Intelligence: Proc. Seventeenth*

Conf. (*UAI-2001*), pp. 554 – 561. Morgan Kaufmann.

WIBERG, N. (1996) *Codes and Decoding on General Graphs.* Dept. of Elec. Eng., Linköping, Sweden PhD dissertation. LinkÖping Studies in Science and Technology No. 440.

WIBERG, N., LOELIGER, H.-A., and KÖTTER, R. (1995) Codes and iterative decoding on general graphs. *European Trans. on Telecommunications* **6**: 513 – 525.

WIENER, N. (1948) *Cybernetics.* Wiley.

WILLIAMS, C. K. I., and RASMUSSEN, C. E. (1996) Gaussian processes for regression. In *Advances in Neural Information Processing Systems 8*, ed. by D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo. MIT Press.

WILLIAMS, C. K. I., and SEEGER, M. (2001) Using the Nyström Method to Speed Up Kernel Machines. In *Advances in Neural Information Processing Systems 13*, ed. by T. K. Leen, T. G. Diettrich, and V. Tresp, pp. 682 – 688. MIT Press.

WITTEN, I. H., NEAL, R. M., and CLEARY, J. G. (1987) Arithmetic coding for data compression. *Communications of the ACM* **30** (6): 520 – 540.

WOLF, J. K., and SIEGEL, P. (1998) On two-dimensional arrays and crossword puzzles. In *Proc. 36th Allerton Conf. on Communication, Control, and Computing, Sept. 1998*, pp. 366 – 371. Allerton House.

WORTHEN, A. P., and STARK, W. E. (1998) Low-density parity check codes for fading channels with memory. In *Proc. 36th Allerton Conf. on Communication, Control, and Computing, Sept. 1998*, pp. 117 – 125.

YEDIDIA, J. S. (2000) An idiosyncratic journey beyond mean field theory. Technical report, Mitsubishi Electric Res. Labs. TR-2000-27.

YEDIDIA, J. S., FREEMAN, W. T., and WEISS, Y. (2000) Generalized belief propagation. Technical report, Mitsubishi Electric Res. Labs. TR-2000-26.

YEDIDIA, J. S., FREEMAN, W. T., and WEISS, Y. (2001a) Bethe free energy, Kikuchi approximations and belief propagation algorithms. Technical report, Mitsubishi Electric Res. Labs. TR-2001-16.

YEDIDIA, J. S., FREEMAN, W. T., and WEISS, Y. (2001b) Characterization of belief propagation and its generalizations. Technical report, Mitsubishi Electric Res. Labs. TR-2001-15.

YEDIDIA, J. S., FREEMAN, W. T., and WEISS, Y. (2002) Constructing free energy approximations and generalized belief propagation algorithms. Technical report, Mitsubishi Electric Res. Labs. TR – 2002 – 35.

YEUNG, R. W. (1991) A new outlook on Shannon-information measures. *IEEE Trans. Info. Theory* **37** (3.1): 466 – 474.

YUILLE, A. L. (2001) A double-loop algorithm to minimize the Bethe and Kikuchi free energies. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, ed. by M. Figueiredo, J. Zerubia, and A. Jain, number 2134 in LNCS, pp. 3 – 18. Springer.

ZIPF, G. K. (1949) *Human Behavior and the Principle of Least Effort.* Addison-Wesley.

ZIV, J., and LEMPEL, A. (1977) A universal algorithm for sequential data compression. *IEEE Trans. Info. Theory* **23** (3): 337 – 343.

ZIV, J., and LEMPEL, A. (1978) Compression of individual sequences via variable-rate coding. *IEEE Trans. Info. Theory* **24**(5): 530 – 536.

Document generated by Anna’s Archive around 2023-2024 as part of the DuXiu collection (https://annas-blog.org/duxiu-exclusive.html).

Images have been losslessly embedded. Information about the original file can be found in PDF attachments. Some stats (more in the PDF attachments):

{
  "filename": "MTE3MDYwOTIuemlw",
  "filename_decoded": "11706092.zip",
  "filesize": 107980289,
  "md5": "925a6d83b54745d2524a2604e51e2c7a",
  "header_md5": "9878ebb7ad7730829df5afaf70bc66d4",
  "sha1": "399eb302d81eccaf925ae3a3fc8d52622b30c2a1",
  "sha256": "efe177735199690553be3bc4dca86eca706ddf56b6e5ba9bcd2781bfcc85499e",
  "crc32": 266594377,
  "zip_password": "",
  "uncompressed_size": 122422572,
  "pdg_dir_name": "",
  "pdg_main_pages_found": 743,
  "pdg_main_pages_max": 743,
  "total_pages": 763,
  "total_pixels": 4311279616,
  "pdf_generation_missing_pages": false
}