



克里斯托弗·M·
毕晓普 与 休·
毕晓普

深度学习

基础
与概念

RapidAI Research
公益翻译



Springer

深度学习

克里斯托弗·M·毕晓普 • 休·毕晓普

深度学习

基础与概念



克里斯托弗·M·毕晓普
微软研究院
英国剑桥

休·毕晓普
Wayve科技有限公司
英国伦敦

国际标准书号978 - 3 - 031 - 45468 - 4 (电子书)

<https://doi.org/10.1007/978-3-031-45468-4>

C 编辑（如适用）和作者，2024年由施普林格自然瑞士股份公司独家授权
本作品受版权保护。出版商拥有独家授权，涵盖全部或部分材料，具体包括翻译、重印、插图再利用、朗诵、广播、缩微胶片复制或以任何其他物理方式复制，以及传输或信息存储与检索、电子改编、计算机软件，或采用现在已知或今后开发的类似或不同方法。

本出版物中使用的通用描述性名称、注册名称、商标、服务标记等，即使没有具体声明，也并不意味着这些名称不受相关保护法律法规的约束，因此不能随意使用。

可以合理地认为，出版商、作者和编辑都相信本书中的建议和信息在出版之日是真实准确的。对于本书所含内容或可能出现的任何错误或遗漏，无论是出版商还是作者或编辑都不提供明示或暗示的保证。对于已出版地图中的管辖权主张和机构隶属关系，出版商保持中立。

封面图片：maksimee / 阿拉米图片库
本施普林格印记由注册公司施普林格自然瑞士股份公司出版
注册公司地址：瑞士楚格市 Gewerbestrasse 11 号，邮编 6330

本产品中的纸张可回收利用。

前言

深度学习使用经过大数据集训练的多层神经网络来解决复杂的信息处理任务，并已成为机器学习领域最成功的范例。在过去十年中，深度学习彻底改变了许多领域，包括计算机视觉、语音识别和自然语言处理，并且它正在被越来越多地应用于医疗保健、制造业、商业、金融、科学发现以及许多其他领域。最近，被称为大语言模型的大规模神经网络，包含大约一万亿个可学习参数，已被发现展现出通用人工智能的初步迹象，并且正在引发技术史上最大的变革之一。

本书的目标

这种不断扩大的影响伴随着机器学习研究出版物数量和广度的激增，而且创新步伐仍在加速。对于该领域的新手来说，掌握关键思想的挑战，更不用说赶上研究前沿，似乎令人生畏。在此背景下，《深度学习：基础与概念》旨在让机器学习新手以及该领域的有经验者全面理解深度学习的基础思想以及现代深度学习架构和技术的关键概念。这些内容将为读者提供未来专业化的坚实基础。由于该领域变化的广度和速度，我们有意避免试图对最新研究进行全面综述。相反，本书的许多价值源于对关键思想的提炼，尽管该领域本身预计将继续快速发展，但这些基础和概念可能经得起时间的考验。例如，在撰写本文时，大语言模型发展非常迅速，但在过去五年中，底层的Transformer架构和注意力机制基本保持不变，而机器学习的许多核心原理已经为人所知数十年了。

技术的负责任使用

深度学习是一项具有广泛适用性的强大技术，有潜力为世界创造巨大价值并应对社会一些最紧迫的挑战。然而，这些相同的特性意味着深度学习也有可能被蓄意滥用并造成意外危害。我们选择不讨论深度学习使用的伦理或社会方面，因为这些主题非常重要且复杂，需要比这样一本技术教科书更全面的探讨。然而，这些考虑应该基于对基础技术及其工作原理的扎实理解，所以我们希望这本书能为这些重要讨论做出有价值的贡献。尽管如此，还是强烈鼓励读者关注其工作的更广泛影响，并在学习技术本身的同时了解深度学习和人工智能的负责任使用。

本书结构

这本书被结构化为相对大量的较小的、易于理解的章节，每个章节探讨一个特定主题。这本书具有线性结构，即每个章节仅依赖于早期章节中涵盖的材料。它非常适合教授一门为期两个学期的本科或研究生机器学习课程，但对那些从事积极研究或自学的人也同样适用。

只有通过一定程度的数学运用，才能对机器学习有清晰的理解。具体而言，机器学习的核心有三个数学领域：概率论、线性代数和多元微积分。本书对概率论中所需的概念进行了自成体系的介绍，并包含一个附录，总结了线性代数中的一些有用结果。假定读者已经对多元微积分的基本概念有所了解，不过也有附录介绍变分法和拉格朗日乘数。然而，本书的重点是传达对概念的清晰理解，强调具有现实世界实用价值的技术而非抽象理论。在可能的情况下，我们尝试从多个互补的角度呈现更复杂的概念，包括文本描述、图表和数学公式。此外，文中讨论的许多关键算法都在单独的框中进行了总结。这些内容不涉及计算效率问题，而是作为对文中数学解释的补充。因此，我们希望本书的内容能为来自各种背景的读者所理解。

从概念上讲，这本书或许最自然地可被视为《Pattern Recognition (Bishop, 1995b)》的后续之作，该书首次从统计学角度全面论述了神经网络。它也可被视为《Pattern Recognition and Machine Learning (Bishop, 2006)》的姊妹篇，后者涵盖了机器学习中更广泛的主题，尽管它早于深度学习革命。

不过，为确保这本新书内容完整，已从Bishop(2006)中选取了合适的材料并进行重构，以聚焦深度学习所需的那些基础概念。这意味着Bishop(2006)中讨论的机器学习里许多有趣的主题如今仍具价值，但在这本新书中被省略了。例如，Bishop(2006)对贝叶斯方法有一定深度的讨论，而本书几乎完全是非贝叶斯的。

本书配有一个网站，提供辅助材料，包括本书的免费数字版、习题答案以及PDF和JPEG格式的可下载图形版本：

<https://www.bishopbook.com>

本书可按以下BibTex条目引用：

```
@book{Bishop:DeepLearning24,  
    author = {Christopher M. Bishop and Hugh Bishop},  
    title = {Deep Learning: Foundations and Concepts},  
    year = {2024},  
    publisher = {Springer}  
}
```

如果您对本书有任何反馈或想报告任何错误，请发送至
feedback@bishopbook.com

参考文献

本着专注于核心思想的精神，我们不会试图提供全面的文献综述，鉴于该领域变化的规模和速度，无论如何这都是不可能的。然而，我们确实提供了一些关键研究论文、综述文章以及其他进一步阅读来源的参考文献。在许多情况下，这些文献还提供了重要的实现细节，我们在正文中略过这些细节，以免分散读者对正在讨论的核心概念的注意力。

关于机器学习这一主题，特别是深度学习，已经有很多书籍出版。在水平和风格上与本书最接近的包括毕晓普（2006年）、古德费洛、本吉奥和库尔维尔（2016年）、墨菲（2022年）、墨菲（2023年）以及普林斯（2023年）所著的书籍。

在过去十年中，机器学习学术研究的性质发生了显著变化，许多论文在提交给同行评审会议和期刊之前，甚至取而代之的是，先在存档网站上发布。这些网站中最受欢迎的是arXiv，发音为“archive”，网址为

<https://arXiv.org>

该网站允许论文更新，这常常导致与不同历年相关的多个版本，这可能会在应引用哪个版本以及针对哪一年的版本方面造成一些模糊性。它还提供每篇论文的PDF免费访问。因此，我们采用了一种简单的方法，即根据首次上传年份引用论文，不过我们建议阅读最新版本。

arXiv上的论文使用arXiv: YYMM. XXXXX的编号方式进行索引，其中YY和MM分别表示首次上传的年份和月份。后续版本通过在编号后附加版本号N来表示，格式为arXiv:YYMM.XXXXXvN。

练习

每章结尾都有一组练习，旨在强化文中解释的关键概念，或以重要方式对其进行拓展和推广。这些练习是文本的重要组成部分，每个练习都根据难度进行分级，从 (\star) （表示只需几分钟就能完成的简单练习）到 $(\star\star\star)$ （表示复杂得多的练习）。强烈鼓励读者尝试这些练习，因为积极参与材料能大大提高学习效果。所有练习的详细解答可从本书网站下载PDF文件获取。

数学符号

我们采用与Bishop (2006)相同的符号表示法。有关机器学习背景下的数学概述，请参阅Deisenroth、Faisal和Ong (2020)。

向量用小写粗体罗马字母表示，如 \mathbf{x} ，而矩阵用大写粗体罗马字母表示，如 \mathbf{M} 。除非另有说明，所有向量都假定为列向量。上标T表示矩阵或向量的转置，因此 \mathbf{x}^T 将是一个行向量。符号 (w_1, \dots, w_M) 表示一个具有 M 个元素的行向量，相应的列向量写作

$\mathbf{w} = (w_1, \dots, w_M)^T$ 。 $M \times M$ 单位矩阵（也称为幺矩阵）表示为 \mathbf{I}_M ，如果其维度没有歧义，将简记为 \mathbf{I} 。它的元素 I_{ij} 在 $i = j$ 时等于1，在 $i \neq j$ 时等于0。单位矩阵的元素有时用 δ_{ij} 表示。符号1表示一个所有元素都为1的列向量。 $\mathbf{a} \oplus \mathbf{b}$ 表示向量 \mathbf{a} 和 \mathbf{b} 的拼接，因此如果 $\mathbf{a} = (a_1, \dots, a_N)$ 且 $\mathbf{b} = (b_1, \dots, b_M)$ ，那么 $\mathbf{a} \oplus \mathbf{b} = (a_1, \dots, a_N, b_1, \dots, b_M)$ 。 $|x|$ 表示标量 x 的模（正值部分），也称为绝对值。我们用 $\det \mathbf{A}$ 表示矩阵 \mathbf{A} 的行列式。

符号 $x \sim p(x)$ 表示 x 是从分布 $p(x)$ 中采样得到的。在存在歧义的情况下，我们将使用下标，如 $p_x(\cdot)$ ，来表示所指的是哪个密度。关于随机变量 x 的函数 $f(x, y)$ 的期望用 $\mathbb{E}_x[f(x, y)]$ 表示。在对于平均所依据的变量不存在歧义的情况下，这将通过省略后缀来简化，例如 $\mathbb{E}[x]$ 。如果 x 的分布以另一个变量 z 为条件，那么相应的条件期望将写为 $\mathbb{E}_x[f(x) | z]$ 。类似地， $f(x)$ 的方差用 $\text{var}[f(x)]$ 表示，对于向量变量，协方差写为 $\text{cov}[\mathbf{x}, \mathbf{y}]$ 。我们还将使用 $\text{cov}[\mathbf{x}]$ 作为 $\text{cov}[\mathbf{x}, \mathbf{x}]$ 的简写符号。

符号 \forall 表示“对于所有”，因此 $\forall m \in \mathcal{M}$ 表示集合 \mathcal{M} 内 m 的所有值。我们用 \mathbb{R} 表示实数。在图上，节点 i 的邻居集表示为 $\mathcal{N}(i)$ ，不应将其与高斯分布或正态分布 $\mathcal{N}(x | \mu, \sigma^2)$ 混淆。泛函表示为 $f[y]$ ，其中 $y(x)$ 是某个函数。附录B中讨论了泛函的概念。花括号 $\{\}$ 表示一个集合。

符号 $g(x) = \mathcal{O}(f(x))$ 表示当 $x \rightarrow \infty$ 时 $|f(x)/g(x)|$ 是有界的。例如，如果 $g(x) = 3x^2 + 2$ ，那么 $g(x) = \mathcal{O}(x^2)$ 。符号 $\lfloor x \rfloor$ 表示 x 的“向下取整”，即小于或等于 x 的最大整数。

如果我们有 N 个独立同分布 (i.i.d.) 的 D 维向量 $\mathbf{x} = (x_1, \dots, x_D)^T$ 的值 $\mathbf{x}_1, \dots, \mathbf{x}_N$ ，我们可以将这些观测值组合成一个维度为 $N \times D$ 的数据矩阵 \mathbf{X} ，其中 \mathbf{X} 的第 n 行对应于行向量 \mathbf{x}_n^T 。因此， \mathbf{X} 的第 n, i 个元素对应于第 n 个观测值 \mathbf{x}_n 的第 i 个元素，并记为 x_{ni} 。对于一维变量，我们用 \mathbf{X} 表示这样一个矩阵，它是一个列向量，其第 n 个元素是 x_n 。注意， \mathbf{X} (其维度为 N) 使用不同的字体以区别于 \mathbf{x} (其维度为 D)。

致谢

我们要向许多审阅章节草稿并提供宝贵反馈的人表示诚挚的感谢。特别要感谢塞缪尔·阿尔巴尼、克里斯蒂安·博德纳、约翰·布伦斯克尔、韦塞尔·布鲁因斯马、伊格纳斯·布德维蒂斯、陈驰、陈耀一、陈龙、费加尔·科特、山姆·德夫林、亚历山大·杜鲁梅里克、塞巴斯蒂安·埃勒特、卡塔琳娜·埃莱兹、安德鲁·冯、葛宏、保罗·格拉德科夫、保拉·戈里·乔吉、约翰·戈斯曼、韩腾达、许朱妍、卡特娅·霍夫曼、黄钦伟、黄永超、朱利奥·伊萨基尼、马修·约翰逊、普拉吉娅·凯尔、阿塔尔瓦·凯尔卡尔、利昂·克莱因、普什梅特·科利、邦妮·克鲁夫特、阿德里安·李、刘海光、卢子恒、朱利亚·路易丝、斯特拉蒂斯·马尔库、塞尔吉奥·瓦尔卡雷尔·马库阿、克日什托夫·马齐亚尔兹、马特伊·梅泽拉、劳伦斯·米德利、奥斯曼·穆尼尔、费利克斯·穆西尔、伊莉斯·范德波尔、秦涛、艾萨克·里德、大卫·罗森伯格、劳埃德·拉塞尔、马克西米利安·谢贝克、梅根·斯坦利、卡琳·施特劳斯、克拉克·邓普顿、马龙·托巴本、阿尔多·萨耶格·帕索斯 - 特雷霍、理查德·特纳、马克斯·韦林、魏福如、罗伯特·韦斯顿、克里斯·威廉姆斯、夏迎策、谢淑芳、伊琳娜·扎波罗热茨、克劳迪奥·泽尼、张谢源以及其他许多通过宝贵讨论做出贡献的同事。我们还要感谢我们的编辑保罗·德鲁加斯以及施普林格的其他许多人，还有文字编辑乔纳森·韦布利，感谢他们在本书出版过程中的支持。

我们要特别感谢马库斯·斯文森 (Markus Svensén)，他为毕晓普 (2006 年) 的图表和排版提供了巨大帮助，包括 LATEX 样式文件，本书也使用了这些文件。我们还要感谢许多科学家，他们允许我们复制其已发表作品中的图表。具体图表的致谢信息见相关图表说明。

克里斯要衷心感谢微软营造了一个极具启发性的研究环境，并提供了撰写本书的机会。然而，本书中表达的观点和意见仅代表作者本人，因此不一定与微软及其附属机构的观点相同。能与我的儿子休 (Hugh) 合作编写这本书是一种巨大的荣幸和乐趣，这本书始于新冠疫情首次封锁期间的一个联合项目。

休要感谢Wayve Technologies Ltd慷慨地允许他兼职工作，以便他能够参与本书的撰写，同时感谢公司为他提供了一个鼓舞人心且支持性的工作和学习环境。本书中表达的观点不一定与Wayve或其附属机构的观点相同。他要感谢他的未婚妻杰迈玛一直以来的支持，以及她在语法和文体方面的咨询。他还要感谢克里斯，在休的一生中，克里斯一直是一位出色的同事，也是他的灵感源泉。

最后，我们俩都要向我们的家庭成员珍娜和马克表示衷心的感谢，感谢他们为我们做了太多太多，在此难以一一列举。似乎很久以前，我们都聚集在安塔利亚的海滩上观看日全食，并为《模式识别与机器学习》的献词页拍摄了一张全家福！

克里斯·毕晓普和休·毕晓普
英国剑桥
2023年10月

内容

前言	v
内容	xi
1 深度学习革命	1
1.1 深度学习的影响	2
1.1.1 医学诊断	2
1.1.2 蛋白质结构	3
1.1.3 图像合成	4
1.1.4 大语言模型	5
1.2 一个教程示例	6
1.2.1 合成数据	6
1.2.2 线性模型	8
1.2.3 误差函数	8
1.2.4 模型复杂度	9
1.2.5 正则化	12
1.2.6 模型选择	14
1.3 机器学习简史	16
1.3.1 单层网络	17
1.3.2 反向传播	18
1.3.3 深度网络	20
2 概率	23
2.1 概率规则	25
2.1.1 一个医学筛查示例	25
2.1.2 和规则与积规则	26
2.1.3 贝叶斯定理	28
2.1.4 重新审视医学筛查	30
2.1.5 先验概率和后验概率	31

2.1.6 自变量	31
2.2 概率密度	32
2.2.1 示例分布	33
2.2.2 期望与协方差	34
2.3 高斯分布	36
2.3.1 均值与方差	37
2.3.2 似然函数	37
2.3.3 最大似然估计的偏差	39
2.3.4 线性回归	40
2.4 密度变换	42
2.4.1 多元分布	44
2.5 信息论	46
2.5.1 熵	46
2.5.2 物理学视角	47
2.5.3 微分熵	49
2.5.4 最大熵	50
2.5.5 库尔贝克-莱布勒散度	51
2.5.6 条件熵	53
2.5.7 互信息	54
2.6 贝叶斯概率	54
2.6.1 模型参数	55
2.6.2 正则化	56
2.6.3 贝叶斯机器学习	57
练习	58
 3 标准分布	 65
3.1 离散变量	66
3.1.1 伯努利分布	66
3.1.2 二项分布	67
3.1.3 多项分布	68
3.2 多元高斯分布	70
3.2.1 高斯分布的几何形状	71
3.2.2 矩	74
3.2.3 局限性	75
3.2.4 条件分布	76
3.2.5 边缘分布	79
3.2.6 贝叶斯定理	81
3.2.7 最大似然法	84
3.2.8 序贯估计	85
3.2.9 高斯混合	86
3.3 周期变量	89
3.3.1 冯·米塞斯分布	89
3.4 指数族	94
3.4.1 充分统计量	97
3.5 非参数方法	98

3.5.1 直方图	98
3.5.2 核密度	100
3.5.3 最近邻	103
练习	105
4 单层网络：回归		111
4.1 线性回归	112
4.1.1 基函数	112
4.1.2 似然函数	114
4.1.3 最大似然	115
4.1.4 最小二乘法的几何原理	117
4.1.5 序贯学习	117
4.1.6 正则化最小二乘法	118
4.1.7 多个输出	119
4.2 决策理论	120
4.3 偏差-方差权衡	123
练习	128
5 单层网络：分类		131
5.1 判别函数	132
5.1.1 两类	132
5.1.2 多类	134
5.1.3 K 编码中的1-of- K 编码	135
5.1.4 分类的最小二乘法	136
5.2 决策理论	138
5.2.1 误分类率	139
5.2.2 预期损失	140
5.2.3 拒绝选项	142
5.2.4 推理与决策	143
5.2.5 分类器准确性	147
5.2.6 ROC曲线	148
5.3 生成式分类器	150
5.3.1 连续输入	152
5.3.2 最大似然解	153
5.3.3 离散特征	156
5.3.4 指数族	156
5.4 判别式分类器	157
5.4.1 激活函数	158
5.4.2 固定基函数	158
5.4.3 逻辑回归	159
5.4.4 多类逻辑回归	161
5.4.5 概率单位回归	163
5.4.6 典范链接函数	164
练习	166

6 深度神经网络	171
6.1 固定基函数的局限性	172
6.1.1 维度诅咒	172
6.1.2 高维空间	175
6.1.3 数据流形	176
6.1.4 数据依赖基函数	178
6.2 多层网络	180
6.2.1 参数矩阵	181
6.2.2 通用逼近	181
6.2.3 隐藏单元激活函数	182
6.2.4 权重空间对称性	185
6.3 深度网络	186
6.3.1 分层表示	187
6.3.2 分布式表示	187
6.3.3 表示学习	188
6.3.4 迁移学习	189
6.3.5 对比学习	191
6.3.6 通用网络架构	193
6.3.7 张量	194
6.4 误差函数	194
6.4.1 回归	194
6.4.2 二元分类	196
6.4.3 多类分类	197
6.5 混合密度网络	198
6.5.1 机器人运动学示例	198
6.5.2 条件混合分布	199
6.5.3 梯度优化	201
6.5.4 预测分布	202
练习	204
7 梯度下降	209
7.1 误差曲面	210
7.1.1 局部二次近似	211
7.2 梯度下降优化	213
7.2.1 梯度信息的使用	214
7.2.2 批量梯度下降	214
7.2.3 随机梯度下降	214
7.2.4 小批量	216
7.2.5 参数初始化	216
7.3 收敛	218
7.3.1 动量	220
7.3.2 学习率调度	222
7.3.3 RMSProp 和 Adam	223
7.4 归一化	224
7.4.1 数据归一化	226

7.4.2 批量归一化	227
7.4.3 层归一化	229
练习	230
8 反向传播	233
8.1 梯度评估	234
8.1.1 单层网络	234
8.1.2 一般前馈网络	235
8.1.3 一个简单示例	238
8.1.4 数值微分	239
8.1.5 雅可比矩阵	240
8.1.6 海森矩阵	242
8.2 自动微分	244
8.2.1 前向模式自动微分	246
8.2.2 反向模式自动微分	249
练习	250
9 正则化	253
9.1 归纳偏置	254
9.1.1 逆问题	254
9.1.2 无免费午餐定理	255
9.1.3 对称性与不变性	256
9.1.4 等变性	259
9.2 权重衰减	260
9.2.1 一致正则化器	262
9.2.2 广义权重衰减	264
9.3 学习曲线	266
9.3.1 提前停止	266
9.3.2 双重下降	268
9.4 参数共享	270
9.4.1 软权重共享	271
9.5 残差连接	274
9.6 模型平均	277
9.6.1 随机失活	279
练习	281
10 卷积网络	287
10.1 计算机视觉	288
10.1.1 图像数据	289
10.2 卷积滤波器	290
10.2.1 特征检测器	290
10.2.2 平移不变性	291
10.2.3 填充	294
10.2.4 步长卷积	294
10.2.5 多维卷积	295
10.2.6 池化	296

10.2.7 多层卷积	298
10.2.8 示例网络架构	299
10.3 可视化训练后的卷积神经网络	302
10.3.1 视觉皮层	302
10.3.2 可视化训练后的滤波器	303
10.3.3 显著性图	305
10.3.4 对抗攻击	306
10.3.5 合成图像	308
10.4 对象检测	308
10.4.1 边界框	309
10.4.2 交并比	310
10.4.3 滑动窗口	311
10.4.4 跨尺度检测	313
10.4.5 非极大值抑制	314
10.4.6 快速区域卷积神经网络	314
10.5 图像分割	315
10.5.1 卷积分割	315
10.5.2 上采样	316
10.5.3 全卷积网络	318
10.5.4 U型网络架构	319
10.6 风格迁移	320
练习	322
 11 结构化分布	325
11.1 图形模型	326
11.1.1 有向图	326
11.1.2 因式分解	327
11.1.3 离散变量	329
11.1.4 高斯变量	332
11.1.5 二元分类器	334
11.1.6 参数与观测值	334
11.1.7 贝叶斯定理	336
11.2 条件独立性	337
11.2.1 三个示例图	338
11.2.2 解释消除	341
11.2.3 D-分离	343
11.2.4 朴素贝叶斯	344
11.2.5 生成模型	346
11.2.6 马尔可夫毯	347
11.2.7 作为滤波器的图	348
11.3 序列模型	349
11.3.1 隐藏变量	352
练习	353

12个变换器	357
12.1注意力机制	358
12.1.1变换器处理	360
12.1.2注意力系数	361
12.1.3自注意力机制	362
12.1.4网络参数	363
12.1.5 缩放点积注意力机制	366
12.1.6 多头注意力机制	366
12.1.7 变换器层	368
12.1.8 计算复杂度	370
12.1.9 位置编码	371
12.2 自然语言	374
12.2.1 词嵌入	375
12.2.2 分词	377
12.2.3 词袋模型	378
12.2.4 自回归模型	379
12.2.5 循环神经网络	380
12.2.6 时间反向传播	381
12.3 变压器语言模型	382
12.3.1 解码器变压器	383
12.3.2 采样策略	386
12.3.3 编码器变压器	388
12.3.4 序列到序列变压器	390
12.3.5 大型语言模型	390
12.4 多模态变换器	394
12.4.1 视觉变换器	395
12.4.2 生成式图像变换器	396
12.4.3 音频数据	399
12.4.4 文本转语音	400
12.4.5 视觉与语言变换器	402
练习	403
13 图神经网络	407
13.1 图上的机器学习	409
13.1.1 图属性	410
13.1.2 邻接矩阵	410
13.1.3 排列不变性	411
13.2 神经消息传递	412
13.2.1 卷积滤波器	413
13.2.2 图卷积网络	414
13.2.3 聚合算子	416
13.2.4 更新算子	418
13.2.5 节点分类	419
13.2.6 边缘分类	420
13.2.7 图分类	420

13.3 通用图网络	420
13.3.1 图注意力网络	421
13.3.2 边嵌入	421
13.3.3 图嵌入	422
13.3.4 过平滑	422
13.3.5 正则化	423
13.3.6 几何深度学习	424
练习	425
14 采样	429
14.1 基本采样算法	430
14.1.1 期望	430
14.1.2 标准分布	431
14.1.3 拒绝采样	433
14.1.4 自适应拒绝采样	435
14.1.5 重要性采样	437
14.1.6 采样重要性重采样	439
14.2 马尔可夫链蒙特卡罗	440
14.2.1 梅特罗波利斯算法	441
14.2.2 马尔可夫链	442
14.2.3 梅特罗波利斯-黑斯廷斯算法	445
14.2.4 吉布斯采样	446
14.2.5 祖先采样	450
14.3 朗之万采样	451
14.3.1 基于能量的模型	452
14.3.2 最大化似然度	453
14.3.3 朗之万动力学	454
练习	456
15 离散潜在变量	459
15.1 K 均值聚类	460
15.1.1 图像分割	464
15.2 高斯混合模型	466
15.2.1 似然函数	468
15.2.2 最大似然	470
15.3 期望最大化算法	474
15.3.1 高斯混合	478
15.3.2 与 K 均值的关系	480
15.3.3 伯努利分布的混合	481
15.4 证据下界	485
15.4.1 重新审视期望最大化算法	486
15.4.2 独立同分布数据	488
15.4.3 参数先验	489
15.4.4 广义期望最大化算法	489
15.4.5 顺序期望最大化算法	490
练习	490

16 连续潜变量	495
16.1 主成分分析	497
16.1.1 最大方差公式	497
16.1.2 最小误差公式	499
16.1.3 数据压缩	501
16.1.4 数据白化	502
16.1.5 高维数据	504
16.2 概率潜在变量	506
16.2.1 生成模型	506
16.2.2 似然函数	507
16.2.3 最大似然	509
16.2.4 因子分析	513
16.2.5 独立成分分析	514
16.2.6 卡尔曼滤波器	515
16.3 证据下界	516
16.3.1 期望最大化	518
16.3.2 主成分分析的期望最大化	519
16.3.3 因子分析的期望最大化	520
16.4 非线性潜变量模型	522
16.4.1 非线性流形	522
16.4.2 似然函数	524
16.4.3 离散数据	526
16.4.4 生成建模的四种方法	527
练习	527
17 生成对抗网络	533
17.1 对抗训练	534
17.1.1 损失函数	535
17.1.2 实际中的GAN训练	536
17.2 图像GAN	539
17.2.1 循环GAN	539
练习	544
18 归一化流	547
18.1 耦合流	549
18.2 自回归流	552
18.3 连续流	554
18.3.1 神经微分方程	554
18.3.2 神经常微分方程反向传播	555
18.3.3 神经常微分方程流	557
练习	559

19 自动编码器	563
19.1 确定性自动编码器	564
19.1.1 线性自动编码器	564
19.1.2 深度自动编码器	565
19.1.3 稀疏自动编码器	566
19.1.4 去噪自动编码器	567
19.1.5 掩码自动编码器	567
19.2 变分自动编码器	569
19.2.1 摊销推理	572
19.2.2 重参数化技巧	574
练习	578
20 扩散模型	581
20.1 前向编码器	582
20.1.1 扩散核	583
20.1.2 条件分布	584
20.2 反向解码器	585
20.2.1 训练解码器	587
20.2.2 证据下界	588
20.2.3 重写证据下界	589
20.2.4 预测噪声	591
20.2.5 生成新样本	592
20.3 得分匹配	594
20.3.1 得分损失函数	595
20.3.2 修正得分损失	596
20.3.3 噪声方差	597
20.3.4 随机微分方程	598
20.4 引导扩散	599
20.4.1 分类器引导	600
20.4.2 无分类器引导	600
练习	603
附录A 线性代数	609
A.1 矩阵恒等式	609
A.2 迹与行列式	610
A.3 矩阵导数	611
A.4 特征向量	612
附录B 变分法	617
附录C 拉格朗日乘数法	621
参考文献	625
索引	641



1

The Deep Learning Revolution



如今，机器学习是最重要且发展最快的技术领域之一。机器学习的应用正变得无处不在，从数据中学习得到的解决方案正日益取代传统的手工编写算法。这不仅提高了现有技术的性能，还为一系列新功能打开了大门，如果必须手动明确设计新算法，这些新功能将是不可想象的。

机器学习的一个特定分支，即深度学习，已成为一种极其强大且通用的数据学习框架。深度学习基于被称为神经网络的计算模型，这些模型最初受到人类大脑学习和信息处理机制的启发。人工智能领域，即AI，试图在机器中重现大脑的强大能力，如今机器学习和AI这两个术语经常互换使用。

当前使用的许多人工智能系统代表了机器学习的应用，这些应用旨在解决非常具体和集中的问题，虽然这些应用非常有用，但它们远远不及人类大脑巨大的能力广度。这导致了“通用人工智能”(AGI)一词的引入，以描述构建具有更大灵活性的机器的愿望。经过几十年的稳步发展，机器学习现在已进入一个非常快速发展的阶段。最近，称为大语言模型的大规模深度学习系统开始展现出非凡的能力，这些能力被描述为通用人工智能的初步迹象 (Bubeck等人，2023年)。

1.1. 深度学习的影响

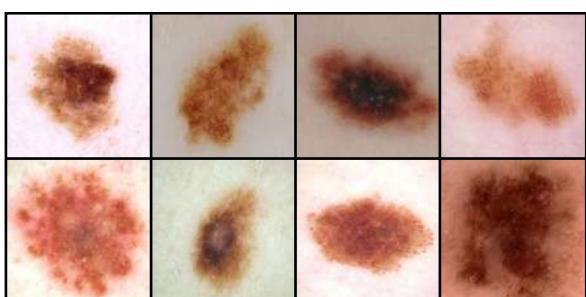
我们通过考虑四个来自不同领域的例子来开始机器学习的讨论，以说明这项技术的广泛适用性，并介绍一些基本概念和术语。这些例子以及许多其他例子特别值得注意的是，它们都使用了深度学习的相同基本框架的变体来解决。这与传统方法形成了鲜明对比，在传统方法中，不同的应用使用广泛不同的专门技术来处理。应该强调的是，我们选择的例子只是深度神经网络适用性广度的一小部分，并且几乎每个有计算作用的领域都适合深度学习的变革性影响。

1.1.1 医学诊断

首先考虑将机器学习应用于皮肤癌诊断问题。黑色素瘤是最危险的皮肤癌类型，但如果早期发现是可治愈的。图1.1展示了皮肤病变的示例图像，上排是恶性黑色素瘤，下排是良性痣。区分这两类图像显然非常具有挑战性，几乎不可能手动编写一个算法，能够以任何合理的准确率成功对这类图像进行分类。

这个问题已经通过深度学习成功解决 (Esteva等人，2017年)。

图1.1 上排为对应危险恶性黑色素瘤的皮肤病变示例，下排为良性痣的示例。未经训练的人眼很难区分这两类。



该解决方案是使用大量病变图像创建的，这些图像被称为训练集，其中每一幅图像都被标记为恶性或良性，这些标签是通过活检测试获得的，活检测试被认为能提供病变的真实类别。训练集用于确定深度神经网络中约2500万个可调参数（称为权重）的值。从数据中设置参数值的这个过程被称为学习或训练。目标是让训练好的网络仅从图像本身就能预测新病变的正确标签，而无需进行耗时的活检步骤。这是一个监督学习问题的示例，因为对于每个训练示例，网络都会被告知正确的标签。这也一个分类问题的示例，因为每个输入都必须被分配到一组离散的类别（在这种情况下为良性或恶性）。输出由一个或多个连续变量组成的应用被称为回归问题。回归问题的一个示例是预测化学制造过程中的产量，其中输入包括温度、压力和反应物浓度。

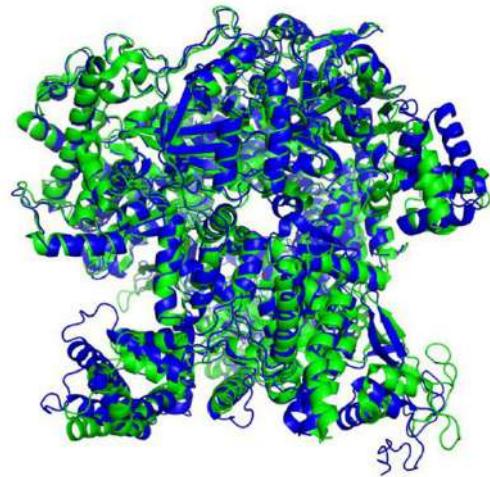
此应用的一个有趣方面是，可用的带标签训练图像数量相对较少，约为129,000张，因此深度神经网络首先在一个包含128万张日常物体（如狗、建筑物和蘑菇）图像的最大数据集上进行训练，然后在病变图像数据集上进行微调。这是迁移学习的一个例子，其中网络从日常物体的大数据集中学习自然图像的一般属性，然后专门用于病变分类的特定问题。通过使用深度学习，皮肤病变图像的分类已经达到了超过专业皮肤科医生的准确率水平（Brinker等人，2019年）。

1.1.2 蛋白质结构

蛋白质有时被称为生物体的基石。它们是生物分子，由一条或多条称为氨基酸的长链单元组成，共有22种不同类型，蛋白质由氨基酸序列决定。一旦蛋白质在活细胞内合成，它就会折叠成复杂的三维结构，其行为和相互作用很大程度上由其形状决定。在给定氨基酸序列的情况下计算这种三维结构，半个世纪以来一直是生物学中的一个基本开放性问题，直到深度学习出现之前进展相对较小。

3D结构可以通过诸如X射线晶体学、低温电子显微镜或核磁共振光谱等技术进行实验测量。然而，这可能极其耗时，并且对于某些蛋白质来说可能具有挑战性，例如由于难以获得纯样品或因为结构取决于环境。相比之下，蛋白质的氨基酸序列可以以较低的成本和更高的通量通过实验确定。因此，人们非常有兴趣能够直接从氨基酸序列预测蛋白质的3D结构，以便更好地理解生物过程或用于药物发现等实际应用。可以训练深度学习模型以氨基酸序列作为输入并生成3D结构作为输出，其中训练数据由一组氨基酸序列和3D结构都已知的蛋白质组成。

图1.2名为T1044/6VR4的蛋白质三维形状示意图。绿色结构显示了通过X射线晶体学确定的真实结构，而叠加的蓝色结构显示了通过名为AlphaFold的深度学习模型获得的预测结果。[经许可改编自Jumper等人（2021年）的研究。]



因此，蛋白质结构预测是监督学习的另一个例子。一旦系统经过训练，它就可以将新的氨基酸序列作为输入，并预测相关的三维结构（Jumper等人，2021年）。图1.2比较了蛋白质的预测三维结构和通过X射线晶体学获得的真实结构。

1.1.3 图像合成

在到目前为止讨论的两个应用中，神经网络学习将输入（皮肤图像或氨基酸序列）转换为输出（分别为病变分类或3D蛋白质结构）。现在我们来看一个例子，其中训练数据仅由一组样本图像组成，训练网络的目标是创建同类的新图像。这是一个无监督学习的例子，因为与病变分类和蛋白质结构的例子不同，这些图像没有标签。图1.3展示了一个深度神经网络生成的合成图像示例，该网络是在一组在摄影棚纯色背景下拍摄的人脸图像上训练的。这种合成图像质量极高，很难将它们与真人照片区分开来。

这是一个生成模型的示例，因为它可以生成与用于训练模型的示例不同但具有相同统计属性的新输出示例。这种方法的一个变体允许根据作为提示已知的输入文本字符串生成图像，以便图像内容反映文本输入的语义。生成式人工智能一词用于描述以图像、视频、音频、文本、候选药物分子或其他模态形式生成输出的深度学习模型。



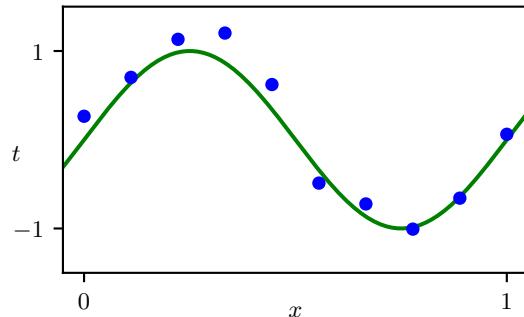
图1.3 使用无监督学习训练的深度神经网络生成的合成面部图像。[来自<https://generated.photos。>]

1.1.4 大语言模型

近年来，机器学习领域最重要的进展之一是开发了用于处理自然语言和其他形式的序列数据（如源代码）的强大模型。大型语言模型（LLM）使用深度学习来构建丰富的内部表示，以捕捉语言的语义属性。一类重要的大型语言模型，称为自回归语言模型，可以生成语言作为输出，因此，它们是生成式人工智能的一种形式。此类模型将一系列单词作为输入，并在输出时生成一个表示序列中下一个单词的单个单词。然后，可以将附加了新单词的扩充序列再次输入模型，以生成后续单词，并且可以重复此过程以生成一长串单词。此类模型还可以输出一个特殊的“停止”单词，以表示文本生成的结束，从而使它们能够输出有限长度的文本并停止。此时，用户可以在将完整序列反馈回模型以触发进一步的单词生成之前，将自己的一系列单词附加到序列中。通过这种方式，人类有可能与神经网络进行对话。

此类模型可以通过提取训练对来在大型文本数据集上进行训练，每个训练对由随机选择的单词序列作为输入，并以已知的下一个单词作为目标输出。这是自监督学习的一个示例，其中学习从输入到输出的函数，但标记的输出是从输入训练数据中自动获得的，无需单独的人工衍生标签。

图1.4 $N = 10$ 个点的训练数据集的绘图，以蓝色圆圈表示，每个点都包含输入变量 x 的一个观测值以及相应的目标变量 t 。绿色曲线显示了用于生成数据的函数 $\sin(2\pi x)$ 。我们的目标是在不知道绿色曲线的情况下，预测 x 某个新值对应的 t 的值。



由于可以从多个来源获得大量文本，这种方法允许扩展到非常大的训练集和相关的非常大的神经网络。

第12章

大语言模型可以展现出非凡的能力，这些能力被描述为新兴通用人工智能的初步迹象（Bubeck等人，2023年），我们将在本书后面详细讨论此类模型。在下一页，我们给出了一个基于名为GPT-4

（OpenAI，2023年）的模型的语言生成示例，以回应输入提示“写出存在无穷多个质数这一事实的证明；通过两方就证明进行争论的对话，以莎士比亚戏剧的风格来做这件事。”。

1.2. 一个教程示例

对于机器学习领域的新手来说，许多基本概念和大部分术语都可以在一个简单示例的背景下引入，该示例涉及将多项式拟合到一个小型合成数据集（Bishop，2006年）。这是一种监督学习问题的形式，在这种问题中，给定输入变量的值，我们希望对目标变量进行预测。

1.2.1 合成数据

我们用 x 表示输入变量，用 t 表示目标变量，并且我们假设这两个变量在实轴上都取连续值。假设我们有一个训练集，它由 N 个 x 的观测值组成，记为 x_1, \dots, x_N ，以及 t 值的相应观测值，记为 t_1, \dots, t_N 。我们的目标是针对 x 的某个新值预测 t 的值。能够对以前未见过的输入进行准确预测是机器学习中的一个关键目标，这被称为泛化。

我们可以使用从正弦函数采样生成的合成数据集来说明这一点。图1.4展示了包含 $N = 10$ 个数据点的训练集的绘图，其中输入值是通过在 $[0, 1]$ 范围内均匀间隔地选择 x_n 的值生成的， $n = 1, \dots, N$ 。相关的目标数据值是通过首先计算函数 $\sin(2\pi x)$ 对于 x 的每个值而获得的。

GPT-4: 素数有无穷多个的证明

第一方：

你不明白，我的朋友，素数根本没有上限，这些不可分割的数，无论看起来多么不可能。

第二方：

但你怎么能如此确定地知道呢？
为了证明这个事实，你能展示什么？

第一方：

想想看，我心存疑虑的同伴，一个能阐明的巧妙证明：假设存在一个最后的质数，我们终将见到的最大质数。为简便起见，我们称这个数为“P”，然后，我的朋友，我们打算构造一个新数“Q”，比“P”大且同样是质数。

第二方：

但是，我的朋友，该如何做到呢？
这证明还未真正成功。

第一方：

很简单，朋友，只需等着瞧，我们将把质数相乘，这是关键。从二到“P”的所有质数相乘，会得出我们要找的新“Q”。但我们还必须再加上一样东西，以使证明更无懈可击，我们在末尾加个一，现在证明就在眼前。

第二方：

我现在明白了，一清二楚，因为“Q”必定是个质数，亲爱的。它不能被任何质数整除，所有比它小的质数都不行。而且既然我们选了“P”作为最后一个质数，这就证明我们的假设不成立了。我们看到不存在最后的质数，数字是无穷无尽的。

第2.3节

然后给每个这样的点添加一小部分随机噪声（由高斯分布控制）以获得相应的目标值 t_n 。通过以这种方式生成数据，我们捕捉到了许多现实世界数据集的一个重要属性，即它们具有一种潜在的规律性，我们希望学习这种规律性，但单个观测值会受到随机噪声的干扰。这种噪声可能源于本质上随机的（即随机的）过程，如放射性衰变，但更典型的是由于存在本身未被观测到的变异性来源。

在本教程示例中，我们知道生成数据的真实过程，即正弦函数。在机器学习的实际应用中，我们的目标是根据有限的训练集发现数据中的潜在趋势。然而，知道生成数据的过程使我们能够阐明机器学习中的重要概念。

1.2.2 线性模型

我们的目标是利用这个训练集来预测目标变量对于输入变量的某个新值 \hat{x} 的 \hat{t} 值。正如我们稍后将看到的，这涉及到隐式地尝试发现潜在函数 $\sin(2\pi x)$ 。这本质上是一个难题，因为我们必须从有限的数据集推广到整个函数。此外，观测数据被噪声干扰，所以对于给定的 \hat{x} ， \hat{t} 的合适值存在不确定性。概率论提供了一个框架，用于以精确和定量的方式表达这种不确定性，而决策论允许我们利用这种概率表示，根据适当的标准做出最优预测。从数据中学概率是机器学习的核心，将在本书中详细探讨。

然而，首先我们将以相当非正式的方式进行，并考虑一种基于曲线拟合的简单方法。具体来说，我们将使用以下形式的多项式函数来拟合数据

$$y(x, \mathbf{w}) = w_0 + w_1x + w_2x^2 + \dots + w_Mx^M = \sum_{j=0}^M w_jx^j \quad (1.1)$$

其中 M 为多项式的阶数， x^j 表示 x 的 j 次幂。多项式系数 w_0, \dots, w_M 由向量 \mathbf{w} 共同表示。请注意，尽管多项式函数 $y(x, \mathbf{w})$ 是 x 的非线性函数，但它是系数 \mathbf{w} 的线性函数。像这个多项式这样在未知参数上是线性的函数具有重要性质，也有显著局限性，被称为线性模型。

第4章

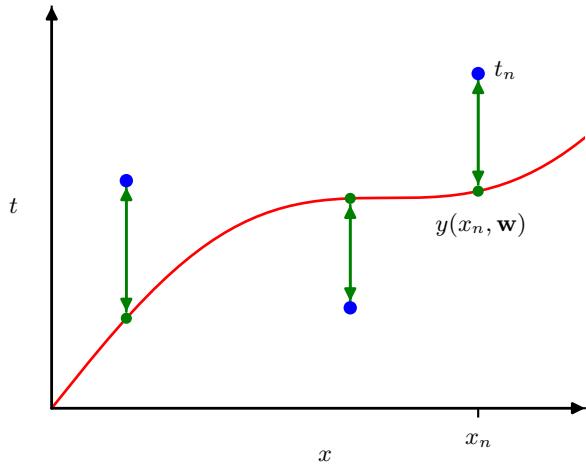
1.2.3 误差函数

系数的值将通过使多项式拟合训练数据来确定。这可以通过最小化一个误差函数来实现，该误差函数衡量对于任何给定的 \mathbf{w} 值，函数 $y(x, \mathbf{w})$ 与训练集数据点之间的不匹配程度。

第2章

第5章

图1.5误差函数 (1.2) 对应于每个数据点相对于函数 $y(x, \mathbf{w})$ 的位移 (由垂直绿色箭头表示) 的平方和的一半。



一种广泛使用的简单误差函数选择是每个数据点 x_n 的预测值 $y(x_n, \mathbf{w})$ 与相应目标值 t_n 之间差值的平方和，其定义为

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 \quad (1.2)$$

第2.3.4节

其中包含 $1/2$ 这个因子是为了后续方便。我们稍后将从概率论出发推导这个误差函数。这里我们只需注意到它是一个非负量，当且仅当函数 $y(x, \mathbf{w})$ 恰好通过每个训练数据点时它才为零。平方和误差函数的几何解释如图1.5所示。

练习4.1

我们可以通过选择 \mathbf{w} 的值来解决曲线拟合问题，使得 $E(\mathbf{w})$ 尽可能小。由于误差函数是系数 \mathbf{w} 的二次函数，其关于系数的导数在 \mathbf{w} 的元素中是线性的，因此误差函数的最小化有一个唯一解，用 \mathbf{w}^* 表示，可以以封闭形式找到。得到的多项式由函数 $y(x, \mathbf{w}^*)$ 给出。

1.2.4 模型复杂度

仍然存在选择多项式阶数 M 的问题，正如我们将看到的，这将成为一个称为模型比较或模型选择的重要概念的示例。在图1.6 中，我们展示了将阶数为 $M = 0, 1, 3$ 和 9 的多项式拟合到图1.4所示数据集的结果的四个示例。

注意，常数 ($M = 0$) 和一阶 ($M = 1$) 多项式对数据的拟合效果很差，因此对函数 $\sin(2\pi x)$ 的表示也很差。三阶 ($M = 3$) 多项式似乎对图1.6所示示例中的函数 $\sin(2\pi x)$ 拟合得最好。当我们使用更高阶的多项式 ($M = 9$) 时，我们对训练数据得到了很好的拟合。

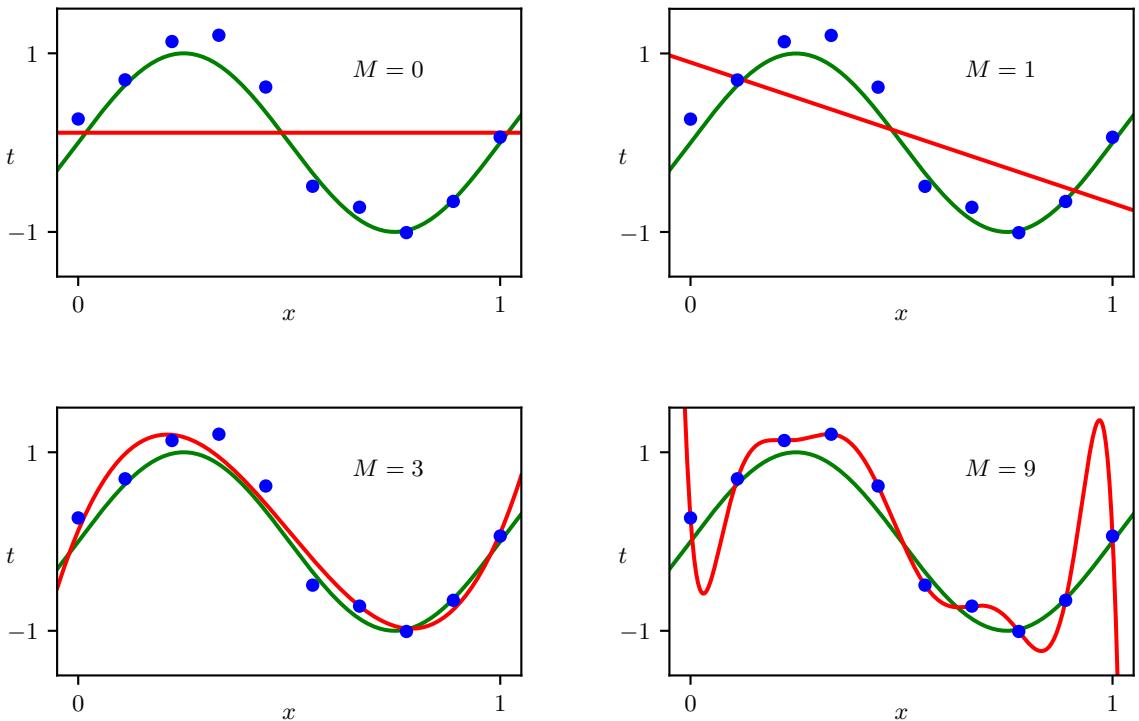


图1.6 不同阶数 M 的多项式曲线（以红色曲线显示），通过最小化误差函数(1.2)拟合到图1.4所示的数据集中。

实际上，该多项式恰好通过每个数据点以及 $E(\mathbf{w}^*) = 0$ 。然而，拟合曲线剧烈振荡，对函数 $\sin(2\pi x)$ 的表示非常差。后一种行为被称为过拟合。

我们的目标是通过对新数据做出准确预测来实现良好的泛化。通过考虑一组单独的数据（称为测试集），我们可以对泛化性能对 M 的依赖性获得一些定量的见解，该测试集包含100个使用与生成训练集点相同的过程生成的数据点。对于 M 的每个值，我们可以评估训练数据中由(1.2)给出的 $E(\mathbf{w}^*)$ 的残差值，并且我们还可以评估测试数据集的 $E(\mathbf{w}^*)$ 。与其评估误差函数 $E(\mathbf{w})$ ，有时使用由以下定义的均方根（RMS）误差会更方便

$$E_{\text{RMS}} = \sqrt{\frac{1}{N} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2} \quad (1.3)$$

其中除以 N 使我们能够在平等的基础上比较不同大小的数据集，并且平方根确保了 E_{RMS} 与目标变量 t 在相同的尺度（和相同的单位）上进行测量。训练集和测试集的图表

图1.7 由(1.3)定义的均方根误差在训练集和独立测试集上针对 M 的不同值进行评估的图形。

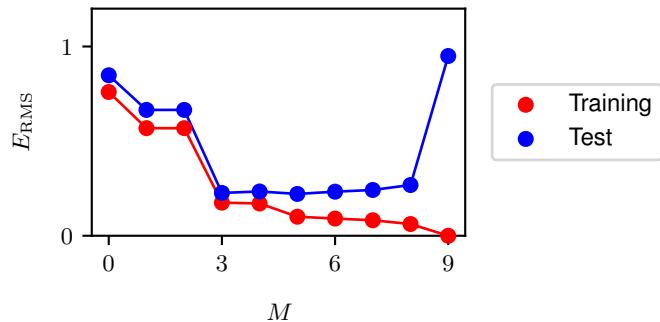


图1.7展示了针对 M 的不同值的均方根误差。测试集误差衡量的是我们在预测 x 的新数据观测值的 t 值时的表现。从图1.7中可以注意到， M 的较小值会导致测试集误差相对较大，这可以归因于相应的多项式相当不灵活，无法捕捉函数 $\sin(2\pi x)$ 中的振荡。 $3 \leq M \leq 8$ 范围内的 M 值会使测试集误差较小，并且这些值也能合理地表示生成函数 $\sin(2\pi x)$ ，如图1.6中 $M = 3$ 所示。

对于 $M = 9$ ，训练集误差趋于零，正如我们所预期的，因为这个多项式包含与10个系数 w_0, \dots, w_9 相对应的10个自由度，所以可以精确地调整到训练集中的10个数据点。然而，测试集误差变得非常大，并且，正如我们在图1.6中看到的，相应的函数 $y(x, \mathbf{w}^*)$ 表现出剧烈的振荡。

这可能看似自相矛盾，因为给定阶数的多项式包含所有低阶多项式作为特殊情况。因此， $M = 9$ 多项式能够生成至少与 $M = 3$ 多项式一样好的结果。此外，我们可能会认为新数据的最佳预测器是生成数据的函数 $\sin(2\pi x)$ （我们稍后会看到确实如此）。我们知道函数 $\sin(2\pi x)$ 的幂级数展开包含所有阶数的项，所以我们可能期望随着我们增加 M ，结果会单调改善。

通过检查从不同阶数的多项式中获得的系数 \mathbf{w}^* 的值，我们可以对该问题有一些深入了解，如表1.1所示。我们看到，随着 M 的增加，系数的大小通常会变大。特别是对于 $M = 9$ 多项式，系数已经针对数据进行了精细调整。它们具有很大的正值和负值，以便相应的多项式函数精确匹配每个数据点，但在数据点之间（特别是在范围的末端附近），该函数表现出图1.6中观察到的大幅振荡。直观地说，正在发生的事情是，具有较大 M 值的更灵活的多项式越来越多地针对目标值上的随机噪声进行调整。

通过检查学习模型在数据集大小变化时的行为，可以进一步深入了解这种现象，如图1.8所示。我们看到，对于给定的模型复杂度，过拟合问题变得不那么严重

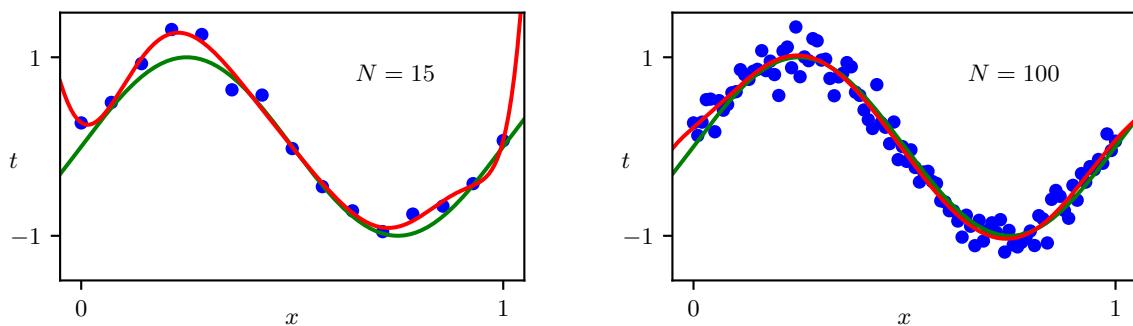


图1.8 使用 $M = 9$ 多项式对 $N = 15$ 个数据点（左图）和 $N = 100$ 个数据点（右图）最小化平方和误差函数(1.2)得到的解的绘图。我们看到，增加数据集的大小会减少过拟合问题。

随着数据集大小的增加。另一种说法是，对于更大的数据集，我们可以用更复杂（换句话说更灵活）的模型来拟合数据。经典统计学中有时提倡的一个粗略启发式方法是，数据点的数量应不少于模型中可学习参数数量的某个倍数（比如5或10）。然而，当我们在本书后面讨论深度学习时，我们会看到，使用参数数量明显多于训练数据点数量的模型也能获得出色的结果。

第9.3.2节

1.2.5 正则化

必须根据可用训练集的大小来限制模型中的参数数量，这有点不尽人意。根据要解决的问题的复杂性来选择模型的复杂度似乎更合理。作为限制参数数量的替代方法，一种经常用于控制过拟合现象的技术是正则化，它涉及在误差函数(1.2)中添加一个惩罚项，以抑制系数具有较大的量级。

表1.1各种阶数多项式的系数 w^* 表。观察随着多项式阶数的增加，系数的典型量级如何急剧增加。

	$M = 0$	$M = 1$	$M = 3$	$M = 9$
w_0^*	0.11	0.90	0.12	0.26
w_1^*		-1.58	11.20	-66.13
w_2^*			-33.67	1,665.69
w_3^*			22.43	-15,566.61
w_4^*				76,321.23
w_5^*				-217,389.15
w_6^*				370,626.48
w_7^*				-372,051.47
w_8^*				202,540.70
w_9^*				-46,080.94

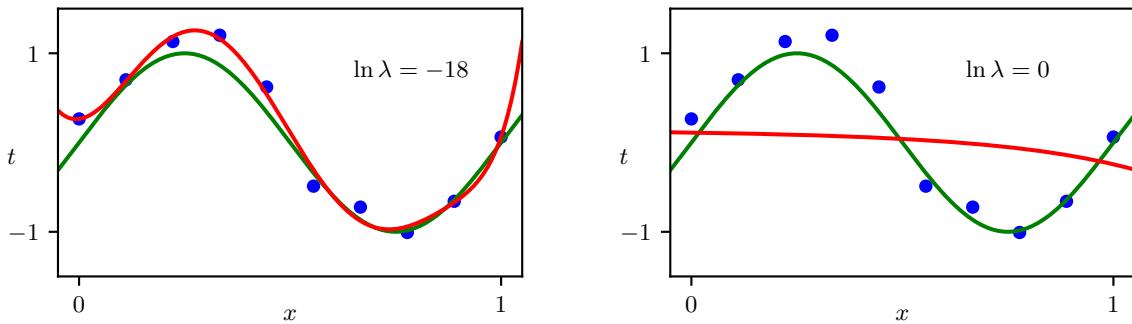


图1.9 使用正则化误差函数 (1.4) 对图1.4所示数据集拟合的 $M = 9$ 多项式的曲线，其中正则化参数 λ 取对应于 $\ln \lambda = -18$ 和 $\ln \lambda = 0$ 的两个值。无正则化器的情况，即 $\lambda = 0$ ，对应于 $\ln \lambda = -\infty$ ，显示在图1.6的右下角。

最简单的这种惩罚项采用所有系数平方和的形式，从而得到如下形式的修正误差函数

$$\tilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2 \quad (1.4)$$

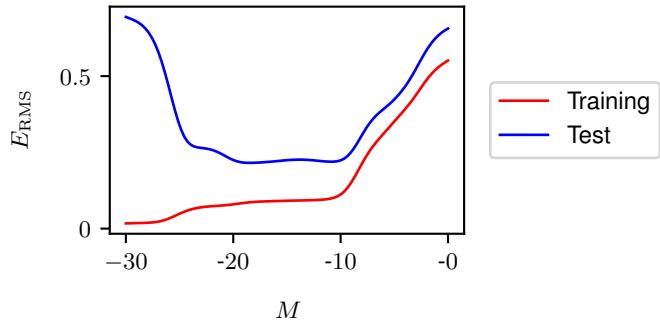
其中 $\|\mathbf{w}\|^2 \equiv \mathbf{w}^T \mathbf{w} = w_0^2 + w_1^2 + \dots + w_M^2$ ，并且系数 λ 控制正则化项与平方和误差项相比的相对重要性。注意，正则化器中通常会省略系数 w_0 ，因为它的包含会导致结果依赖于目标变量的原点选择 (Hastie、Tibshirani和Friedman, 2009)，或者它可能会被包含但有自己的正则化系数。同样，(1.4) 中的误差函数可以用封闭形式精确地最小化。在统计文献中，这样的技术被称为收缩方法，因为它们会减小系数的值。在神经网络的背景下，这种方法被称为权重衰减，因为神经网络中的参数被称为权重，并且这个正则化器鼓励它们衰减到零。

第9.2.1节 练习4.2

图1.9展示了将 $M = 9$ 阶多项式拟合到与之前相同数据集的结果，但现在使用由(1.4)给出的正则化误差函数。我们看到，对于 $\ln \lambda = -18$ 的值，过拟合已得到抑制，并且我们现在获得了对基础函数 $\sin(2\pi x)$ 更接近的表示。然而，如果我们对 λ 使用太大的值，那么我们再次得到较差的拟合，如图1.9中 $\ln \lambda = 0$ 所示。拟合多项式的相应系数在表1.2中给出，表明正则化具有减小系数大小的预期效果。

通过绘制训练集和测试集的均方根误差值 (1.3) 相对于 $\ln \lambda$ 的关系，可以看出正则化项对泛化误差的影响，如图1.10所示。我们看到 λ 现在控制着模型的有效复杂度，从而决定了过拟合的程度。

图1.10 均方根误差(1.3)与 $\ln \lambda$ 关于 $M = 9$ 多项式的关系图。



1.2.6 模型选择

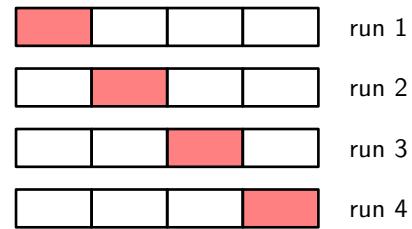
λ 数量是超参数的一个示例，其值在最小化误差函数以确定模型参数 w 的过程中是固定的。我们不能简单地通过联合最小化关于 w 和 λ 的误差函数来确定 λ 的值，因为这将导致 $\lambda \rightarrow 0$ 以及一个训练误差小或为零的过拟合模型。类似地，多项式的阶数 M 是模型的一个超参数，简单地关于 M 优化训练集误差将导致 M 的值很大以及相关的过拟合。因此，我们需要找到一种方法来确定超参数的合适值。上述结果提出了一种实现此目的的简单方法，即通过获取可用数据并将其划分为一个用于确定系数 w 的训练集和一个单独的验证集，也称为留出集或开发集。然后，我们选择在验证集上误差最低的模型。如果使用有限大小的数据集多次迭代模型设计，那么可能会出现对验证数据的一些过拟合，因此可能有必要留出第三个测试集，最终在该测试集上评估所选模型的性能。

对于某些应用，用于训练和测试的数据供应将受到限制。为了构建一个好的模型，我们应该尽可能多地使用可用数据进行训练。然而，如果验证集太小，它将给出相对有噪声的预测性能估计。解决这个困境的一种方法是使用交叉验证，如图1.11所示。

表1.2 具有不同正则化参数 λ 值的 $M = 9$ 多项式的系数 w^* 表。请注意， $\ln \lambda = -\infty$ 对应于无正则化的模型，即图1.6右下角的图形。我们看到，随着 λ 值的增加，典型系数的大小会变小。

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^*	0.26	0.26	0.11
w_1^*	-66.13	0.64	-0.07
w_2^*	1,665.69	43.68	-0.09
w_3^*	-15,566.61	-144.00	-0.07
w_4^*	76,321.23	57.90	-0.05
w_5^*	-217,389.15	117.36	-0.04
w_6^*	370,626.48	9.87	-0.02
w_7^*	-372,051.47	-90.02	-0.01
w_8^*	202,540.70	-70.90	-0.01
w_9^*	-46,080.94	75.26	0.00

图1.11 S 折交叉验证技术（此处以 $S = 4$ 为例进行说明），包括获取可用数据并将其划分为大小相等的 S 组。然后，使用 $S - 1$ 组来训练一组模型，然后在其余组上对这些模型进行评估。然后，对留出组的所有 S 种可能选择（此处由红色块表示）重复此过程，并对 S 次运行的性能分数进行平均。

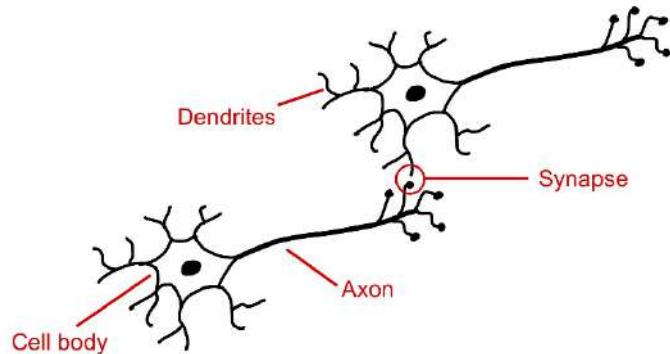


这使得可用数据的比例 $(S - 1)/S$ 可用于训练，同时利用所有数据来评估性能。当数据特别稀缺时，可能适合考虑 $S = N$ 的情况，其中 N 是数据点的总数，这就产生了留一法技术。

交叉验证的主要缺点是，必须执行的训练运行次数会增加 S 倍，这对于训练本身计算成本很高的模型来说可能会出现问题。使用单独数据评估性能的交叉验证等技术的另一个问题是，对于单个模型我们可能有多个复杂度超参数（例如，可能有几个正则化超参数）。在最坏的情况下，探索此类超参数的设置组合可能需要进行与超参数数量呈指数关系的训练运行次数。现代机器学习的最新技术涉及在相应的大数据集上训练的极大模型。因此，超参数设置的探索空间有限，并且严重依赖于从小模型获得的经验和启发式方法。

这个将多项式拟合到由正弦函数生成的合成数据集的简单示例说明了机器学习中的许多关键思想，我们将在后续章节中进一步使用这个示例。然而，机器学习的实际应用在几个重要方面有所不同。用于训练的数据集大小可能会大几个数量级，并且通常会有更多的输入变量，例如在图像分析中可能有数百万个，以及多个输出变量。将输出与输入相关联的可学习函数由一类称为神经网络的模型控制，这些模型可能有大量参数，也许有数千亿个，并且误差函数将是这些参数的高度非线性函数。误差函数不再能通过闭式解最小化，而是必须通过基于误差函数关于参数的导数评估的迭代优化技术来最小化，所有这些都可能需要专门的计算硬件并产生大量的计算成本。

图1.12 示意图展示了来自人类大脑的两个神经元。这些电活性细胞通过称为突触的连接进行通信，其强度会随着网络学习而改变。



1.3. 机器学习简史

机器学习有着悠久而丰富的历史，包括对多种替代方法的探索。在这里，我们关注基于神经网络的机器学习方法的发展，因为这些方法代表了深度学习的基础，并且已被证明是解决实际应用中机器学习问题的最有效方法。

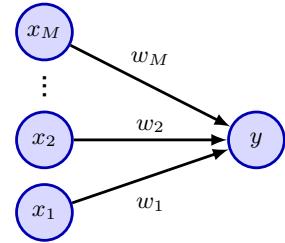
神经网络模型最初受到对人类和其他哺乳动物大脑中信息处理研究的启发。大脑中的基本处理单元是称为神经元的电活性细胞，如图1.12所示。当一个神经元“激发”时，它会沿着轴突发送电脉冲，到达称为突触的连接点，这些连接点与其他神经元形成连接。在突触处会释放称为神经递质的化学信号，这些信号可以刺激或抑制后续神经元的激发。

人类大脑总共包含约900亿个神经元，每个神经元平均与其他神经元有数千个突触，从而形成一个总共约有100万亿个(10^{14})突触的复杂网络。如果某个特定神经元从其他神经元的放电中接收到足够的刺激，那么它也会被诱导放电。然而，一些突触具有负向或抑制性作用，即输入神经元的放电会降低输出神经元放电的可能性。一个神经元能够促使另一个神经元放电的程度取决于突触的强度，而这些强度的变化代表了大脑存储信息和从经验中学习的关键机制。

神经元的这些特性已在非常简单的数学模型中得到体现，这些模型被称为人工神经网络，它们构成了学习计算方法的基础

(McCulloch和Pitts, 1943年)。许多此类模型通过对其他神经元的输出进行线性组合来描述单个神经元的特性，然后使用非线性函数对其进行变换。

图1.13 一个简单的神经网络图，代表描述单个神经元的变换(1.5)和(1.6)。
多项式函数(1.1)可视为该模型的一个特例。



这可以用数学形式表示为

$$a = \sum_{i=1}^M w_i x_i \quad (1.5)$$

$$y = f(a) \quad (1.6)$$

其中 x_1, \dots, x_M 表示与向该神经元发送连接的其他神经元活动相对应的 M 输入，并且 w_1, \dots, w_M 是连续变量，称为权重，其表示相关突触的强度。量 a 称为预激活，非线性函数 $f(\cdot)$ 称为激活函数，输出 y 称为激活。我们可以看到多项式(1.1)可以被视为这种表示的一个特定实例，其中输入 x_i 由单个变量 x 的幂给出，并且函数 $f(\cdot)$ 恰好是恒等函数 $f(a) = a$ 。由(1.5)和(1.6)给出的简单数学公式从20世纪60年代至今一直是神经网络模型的基础，并且可以如图1.13所示以图表形式表示。

1.3.1 单层网络

人工神经网络的历史大致可根据网络的复杂程度分为三个不同阶段，网络的复杂程度通过处理“层”的数量来衡量。由(1.5)和(1.6)描述的简单神经模型可被视为具有单层处理，这对应于图1.13中的单层连接。神经计算历史上最重要的此类模型之一是感知机 (Rosenblatt, 1962年)，其中激活函数 $f(\cdot)$ 是形式为的阶跃函数

$$f(a) = \begin{cases} 0, & \text{if } a \leq 0, \\ 1, & \text{if } a > 0. \end{cases} \quad (1.7)$$

这可以被视为神经放电的简化模型，其中神经元当且仅当总加权输入超过0的阈值时才会放电。感知机由罗森布拉特 (1962年) 首创，他开发了一种特定的训练算法，该算法具有一个有趣的特性，即如果存在一组权重值，使得感知机能够对其训练数据实现完美分类，那么该算法保证能在有限步骤内找到解决方案 (毕晓普, 2006年)。除了学习算法外，感知机还有专门的模拟硬件实现，如图1.14所示。



图1.14 马克1型感知机硬件示意图。左边的照片展示了如何使用一个简单的相机系统获取输入，在这个系统中，一个输入场景（在这种情况下是一个印刷字符）被强光照射，图像聚焦到一个由硫化镉光电池组成的 20×20 阵列上，得到一个原始的400像素图像。感知机还有一个接线板，如中间照片所示，它允许尝试不同的输入特征配置。通常这些配置是随机连接的，以展示感知机无需精确布线就能学习的能力，这与现代数字计算机不同。右边的照片展示了其中一个可学习权重的机架。每个权重都使用一个旋转可变电阻（也称为电位器）来实现，由一个电动机驱动，从而允许学习算法自动调整权重的值。

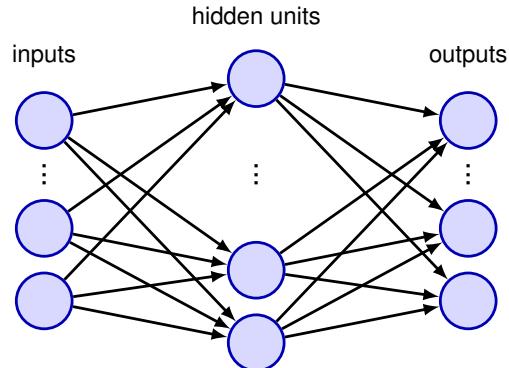
典型的感知机配置有多层处理，但其中只有一层可以从数据中学
习，因此感知机被认为是一个“单层”神经网络。

起初，感知机以类似大脑的方式从数据中学习的能力被认为是
非同寻常的。然而，很明显该模型也有重大局限性。明斯基和佩珀特
(1969年)分析了感知机的特性，他们给出了单层网络能力有限的
形式证明。不幸的是，他们还推测类似的局限性会扩展到具有多
层可学习参数的网络。尽管后一种推测被证明是大错特错的，但结
果是打击了对神经网络模型的热情，这导致了在20世纪70年代和80
年代初对神经网络缺乏兴趣和资金投入。此外，由于缺乏训练多层
网络的有效算法，研究人员无法探索多层网络的特性，因为诸如感
知机算法等技术只适用于单层模型。请注意，尽管感知机早已从实
际的机器学习中消失，但这个名称仍然存在，因为现代神经网络有
时也被称为多层次感知机或MLP。

1.3.2 反向传播

训练具有多层可学习参数的神经网络问题的解决方案来自于微
分学的使用以及基于梯度的优化方法的应用。一个重要的变化是用
具有非零梯度的连续可微激活函数取代阶跃函数(1.7)。另一个关键
修改是引入可微误差函数，该函数定义了给定参数值选择对训练集
中目标变量的预测效果。

图1.15 一个具有两层参数的神经网络，其中箭头表示信息在网络中的流动方向。每个隐藏单元和每个输出单元都计算形式如(1.5)和(1.6)所示的函数，其中激活函数 $f(\cdot)$ 是可微的。



第1.2.3节

当我们使用平方和误差函数(1.2)来拟合多项式时，我们看到了这样一个误差函数的示例。

有了这些变化，我们现在有了一个误差函数，其关于网络中每个参数的导数都可以进行评估。我们现在可以考虑具有多层参数的网络。图1.15展示了一个具有两个处理层的简单网络。中间层的节点称为隐藏单元，因为它们的值不出现在训练集中，训练集仅提供输入和输出的值。图1.15中的每个隐藏单元和每个输出单元都计算由(1.5)和(1.6)给出形式的函数。对于给定的一组输入值，所有隐藏单元和输出单元的状态可以通过重复应用(1.5)和(1.6)来评估，其中信息沿着箭头方向在网络中向前流动。因此，这种模型有时也称为前馈神经网络。

第8章

为了训练这样一个网络，首先使用随机数生成器初始化参数，然后使用基于梯度的优化技术进行迭代更新。这涉及评估误差函数的导数，这可以在一个称为误差反向传播的过程中高效完成。在反向传播中，信息从输出通过网络反向流向输入（Rumelhart、Hinton和Williams，1986）。存在许多不同的优化算法，它们利用要优化的函数的梯度，但在机器学习中最流行的算法也是最简单的，称为随机梯度下降。

第7章

能够训练具有多层权重的神经网络是一项突破，这引发了自20世纪80年代中期左右起该领域兴趣的再度兴起。这也是该领域超越对神经生物学启发的关注并发展出更严谨、更有原则的基础的时期（毕晓普，1995b）。特别是，人们认识到概率论以及统计学领域的思想在神经网络和机器学习中起着核心作用。一个关键的见解是，从数据中学习涉及背景假设，有时也称为先验知识或归纳偏差。这些假设可以明确纳入，例如通过设计神经网络的结构，使得皮肤病变的分类不依赖于病变在图像中的位置，或者它们可能采取由模型的数学形式或训练方式产生的隐含假设的形式。

反向传播和基于梯度的优化方法的发展极大地提高了神经网络解决实际问题的能力。然而，人们也观察到，在多层网络中，只有最后两层的权重会学习到有用的值。除了少数例外，特别是用于图像分析的称为卷积神经网络的模型（LeCun等人，1998），具有两层以上的网络很少有成功的应用。同样，这限制了可以用这类网络有效解决的问题的复杂性。为了在许多应用中获得合理的性能，有必要使用手工制作的预处理将输入变量转换到一个新的空间，希望在这个空间中机器学习问题更容易解决。这个预处理阶段有时也称为特征提取。虽然这种方法有时是有效的，但如果可以从数据中学习特征而不是手工制作，显然会好得多。

到新千年开始时，现有的神经网络方法再次达到了其能力极限。研究人员开始探索大量神经网络的替代方法，如核方法、支持向量机、高斯过程等等。神经网络再次失宠，尽管有一群热情的研究人员继续追求用一种真正有效的方法来训练多层网络的目标。

1.3.3 深度网络

神经网络发展的第三个也是当前阶段始于21世纪的第二个十年。一系列的发展使得具有多层权重的神经网络能够得到有效训练，从而消除了此前这些技术在能力上的限制。具有多层权重的网络被称为深度神经网络，专注于此类网络的机器学习子领域被称为深度学习（LeCun、Bengio和Hinton，2015）。

深度学习起源中的一个重要主题是神经网络规模的显著增加，这是根据参数数量来衡量的。虽然在20世纪80年代，具有几百或几千个参数的网络很常见，但这个数字稳步上升到数百万，然后是数十亿，而目前最先进的模型可能有大约一万亿（ 10^{12} ）个参数。具有许多参数的网络需要相应的大数据集，以便训练信号能够为这些参数产生良好的值。大规模模型和大数据集的结合反过来又需要在训练模型时进行大规模的计算。专门用于为视频游戏等应用快速渲染图形数据而开发的称为图形处理单元（GPU）的处理器，被证明非常适合神经网络的训练，因为网络一层中的单元计算的函数可以并行评估，这与GPU的大规模并行性非常匹配（Krizhevsky、Sutskever和Hinton，2012年）。如今，最大模型的训练是在由专门的高速互连连接的数千个GPU的大型阵列上进行的。

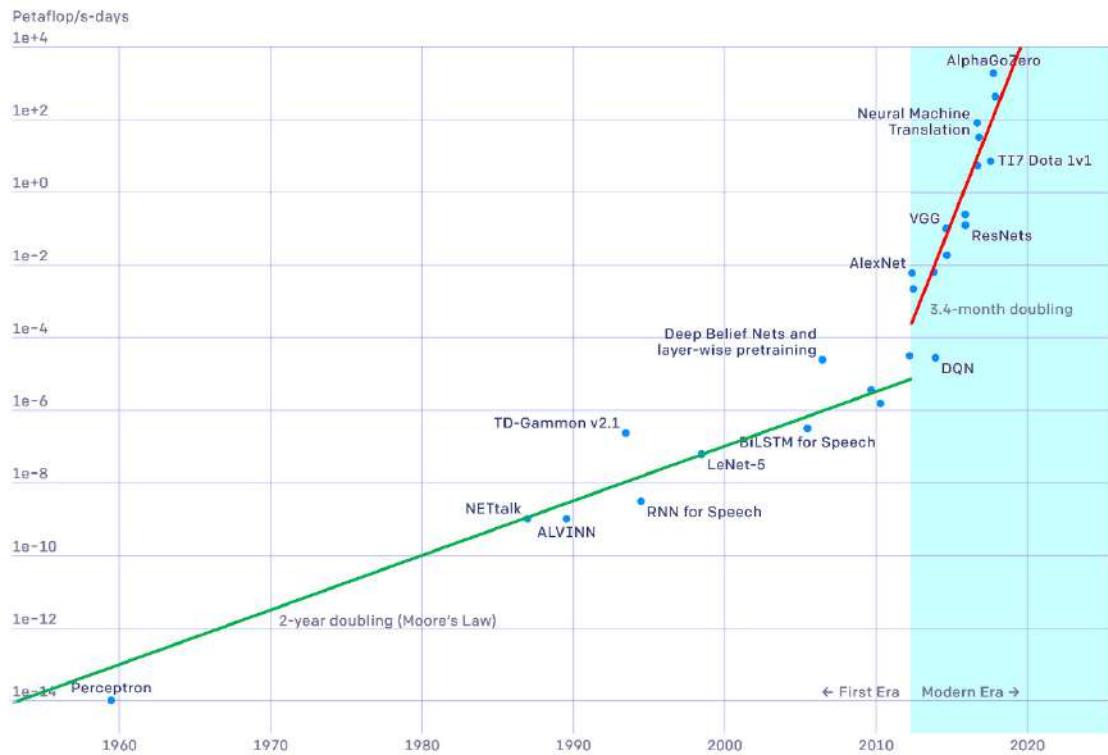


图1.16 以每秒千万亿次浮点运算-天数为单位衡量的训练一个先进神经网络所需的计算周期数随日期变化的曲线图，展示了指数增长的两个不同阶段。[经OpenAI许可使用。]

图1.16展示了多年来训练最先进神经网络所需的计算周期数是如何增长的，呈现出两个不同的增长阶段。纵轴采用指数刻度，单位是千万亿次浮点运算每秒 - 天，其中一亿亿次浮点运算代表 10^{15} (一千万亿) 次浮点运算，而一亿亿次浮点运算每秒就是每秒进行一亿亿次浮点运算。一亿亿次浮点运算每秒 - 天表示以一亿亿次浮点运算每秒的速率进行24小时的计算，大约是 10^{20} 次浮点运算，因此，图表的顶线代表了令人印象深刻的 10^{24} 次浮点运算。图表上的一条直线表示指数增长，我们看到从感知机时代到2012年左右，翻倍时间约为2年，这与由于摩尔定律导致的计算能力的总体增长是一致的。从2012年起，这标志着深度学习时代，我们再次看到指数增长，但现在翻倍时间是3.4个月，相当于计算能力每年增长10倍！

人们经常发现，由于架构创新或纳入更复杂形式的归纳偏差而带来的性能提升，很快就会被单纯扩大训练数据量以及相应扩大模型规模和用于训练的相关计算能力所取代（萨顿，2019年）。不仅大型模型在特定任务上可以具有卓越的性能，而且它们可能能够用同一个经过训练的神经网络解决更广泛的不同问题。大型语言模型就是一个显著的例子，因为单个网络不仅具有非凡的能力广度，甚至能够超越为解决特定问题而设计的专业网络。

第12.3.5节

我们已经看到，深度在使神经网络实现高性能方面起着重要作用。看待深度神经网络中隐藏层作用的一种方式是表征学习（Bengio、Courville和Vincent，2012），即网络学习将输入数据转换为语义上有意义的新表征，从而为最后一层或多层解决问题创造一个容易得多的问题。正如我们在皮肤病变分类中看到的那样，这种内部表征可以通过迁移学习重新用于解决相关问题。值得注意的是，用于处理图像的神经网络可能会学习到与哺乳动物视觉皮层中观察到的非常相似的内部表征。可以适应或微调以用于一系列下游任务的大型神经网络称为基础模型，并且可以利用大型、异构数据集来创建具有广泛适用性的模型（Bommasani等人，2021）。

第10.3节

除了缩放之外，还有其他一些发展有助于深度学习的成功。例如，在简单的神经网络中，训练信号在通过深度网络的连续层进行反向传播时会变弱。解决这个问题的一种技术是引入残差连接（He等人，2015a），这有助于训练具有数百层的网络。另一个关键发展是引入了自动微分方法，其中执行反向传播以评估误差函数梯度的代码是从用于指定前向传播的代码中自动生成的。这使得研究人员能够快速试验神经网络的不同架构，并非常容易地以多种方式组合不同的架构元素，因为只需要显式编码相对简单的前向传播函数。此外，机器学习的许多研究都是通过开源进行的，这使得研究人员能够在他人的工作基础上进行构建，从而进一步加快了该领域的进展速度。

第9.5节



2

Probabilities

在几乎每一个机器学习应用中，我们都必须应对不确定性。例如，一个将皮肤病变图像分类为良性或恶性的系统，在实际中永远无法达到完美的准确率。我们可以区分两种不确定性。第一种是认知不确定性（源自希腊语episteme，意为知识），有时也称为系统不确定性。它的出现是因为我们只能看到有限大小的数据集。随着我们观察更多的数据，例如更多良性和恶性皮肤病变图像的示例，我们就能更好地预测新示例的类别。然而，即使有一个无限大的数据集，由于第二种不确定性，即偶然不确定性（也称为固有或随机不确定性，有时简称为噪声），我们仍然无法达到完美的准确率。一般来说，噪声的出现是因为我们只能观察到关于世界的部分信息，因此，减少这种不确定性来源的一种方法是收集不同类型的数据。这一点将在下文说明。

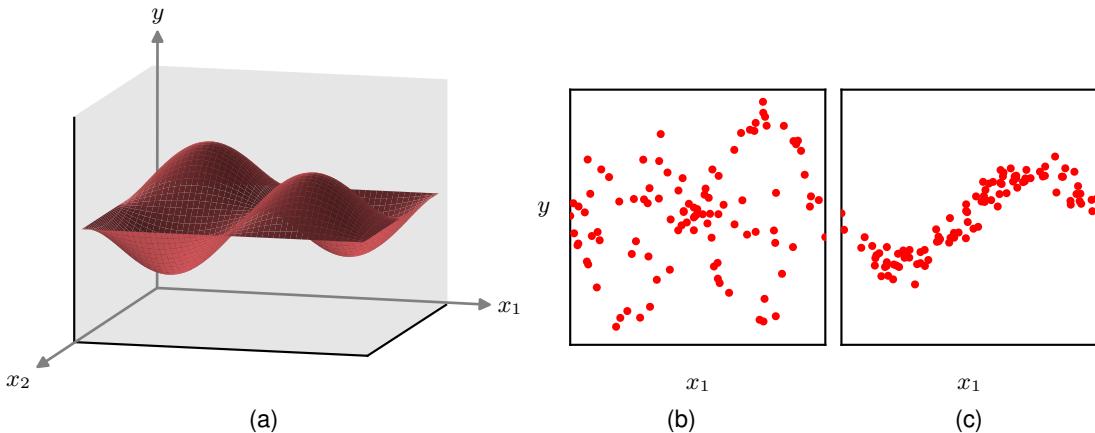


图2.1将简单正弦曲线回归问题扩展到二维。(a)函数 $y(x_1, x_2) = \sin(2\pi x_1) \sin(2\pi x_2)$ 的绘图。通过为 x_1 和 x_2 选择值、计算 $y(x_1, x_2)$ 的相应值，然后添加高斯噪声来生成数据。(b) 100个数据点的绘图，其中 x_2 未被观测到，显示出高噪声水平。(c) 100个数据点的绘图，其中 x_2 被固定为值 $x_2 = \frac{\pi}{2}$ ，模拟能够测量 x_2 以及 x_1 的效果，显示出低得多的噪声水平。

第1.2节使用图2.1中正弦曲线示例到二维的扩展。

作为一个实际例子，皮肤病变的活检样本比单独的图像信息量大得多，可能会大大提高我们判断新病变是否为恶性的准确性。给定图像和活检数据，内在的不确定性可能非常小，通过收集大量训练数据集，我们或许能够将系统不确定性降低到较低水平，从而高精度地预测病变的类别。

这两种不确定性都可以使用概率论框架来处理，概率论为不确定性的量化和处理提供了一个一致的范式，因此构成了机器学习的核心基础之一。我们将看到，概率由两个简单的公式——求和规则和乘积规则——所支配。当与决策理论相结合时，这些规则至少在原则上允许我们在掌握所有可用信息的情况下做出最优预测，即使这些信息可能不完整或模糊不清。

概率的概念通常是根据可重复事件的频率来引入的。例如，考虑图2.2中所示的弯曲硬币，假设硬币的形状使得如果将其大量抛掷，它凹面朝上的次数为 60%，因此凸面朝上的次数为 40%。我们说凹面朝上的概率是 60% 或 0.6。严格来说，在这种情况下，概率是在无限多次“试验”或抛硬币的极限情况下定义的。因为硬币必定要么凹面朝上要么凸面朝上，所以这些概率相加为 100% 或 1.0。这种根据可重复事件的频率来定义概率的方式是频率主义统计学观点的基础。

现在假设，尽管我们知道硬币凹面朝上的概率是 0.6，但我们不被允许查看硬币本身，并且我们不知道哪一面是正面，哪一面是反面。

图2.2 概率既可以被视为与可重复事件相关的频率，也可以被视为不确定性的量化。一枚弯曲的硬币可用于说明这种差异，如文中所讨论的。



第2.6节

练习2.40

如果被要求就硬币抛掷后正面朝上还是反面朝上进行打赌，那么对称性表明我们的打赌应该基于正面朝上的概率为0.5这一假设，而且实际上更仔细的分析表明，在没有任何额外信息的情况下，这确实是理性的选择。在这里，我们使用概率的意义比仅仅作为事件的频率更宽泛。硬币的凸面是正面还是反面本身并不是一个可重复事件，它只是未知的。将概率用作不确定性的量化是贝叶斯观点，并且更具一般性，因为它将频率主义概率作为一种特殊情况包含在内。如果我们得到一系列硬币抛掷的结果，通过利用贝叶斯推理，我们就可以了解硬币的哪一面是正面。我们观察到的结果越多，对于硬币哪一面是哪一面的不确定性就越低。

在非正式地介绍了概率的概念之后，我们现在转向对概率进行更详细的探讨，并讨论如何定量地使用它们。本章其余部分所阐述的概念将为贯穿全书所讨论的许多主题奠定核心基础。

2.1. 概率规则

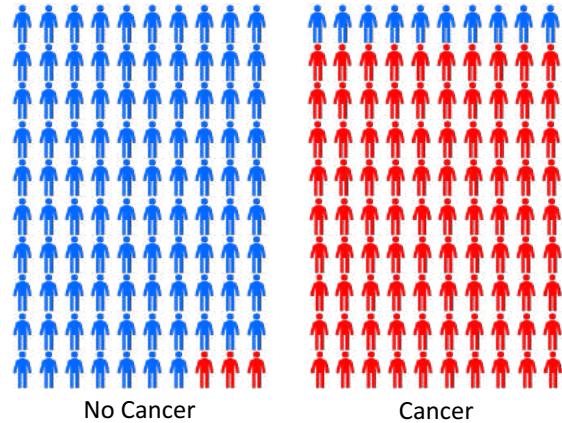
在本节中，我们将推导出两条简单的规则，用于规范概率的行为。然而，尽管它们看似简单，但这些规则将证明是非常强大且广泛适用的。我们将通过首先引入一个简单的例子来引出概率规则。

2.1.1 一个医学筛查示例

考虑对人群进行筛查以实现癌症早期检测的问题，假设人群中有1%的人实际患有癌症。理想情况下，我们的癌症检测对于任何患有癌症的人都给出阳性结果，而对于任何没有癌症的人都给出阴性结果。然而，检测并非完美无缺，所以我们假设当对没有癌症的人进行检测时，其中3%的人会检测呈阳性。这些被称为假阳性。同样，当对确实患有癌症的人进行检测时，其中10%的人会检测呈阴性。这些被称为假阴性。各种错误率如图2.3所示。

基于这些信息，我们可能会问以下问题：(1) “如果我们对人群进行筛查，某人检测呈阳性的概率是多少？”，(2) “如果某人检测结果为阳性，他们实际患有癌症的概率是多少？”。

图2.3癌症检测准确性的说明。在左边显示的每100名接受检测但没有癌症的人中，平均有3人会检测呈阳性。对于右边显示的患有癌症的人，每100名接受检测的人中，平均有90人会检测呈阳性。



我们可以通过详细研究癌症筛查案例来回答这类问题。然而，相反地，我们将暂停对这个具体例子的讨论，首先推导概率的一般规则，即概率的和规则与积规则。然后我们将通过回答我们的两个问题来说明这些规则的用法。

2.1.2 和规则与积规则

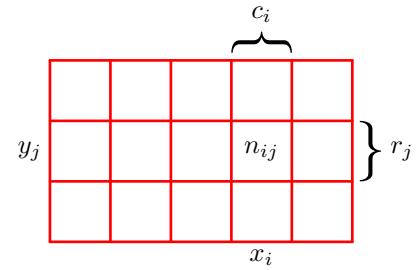
为了推导概率规则，考虑图2.4中所示的稍微更一般的示例，该示例涉及两个变量 X 和 Y 。在我们的癌症示例中， X 可以表示癌症的存在或不存在，而 Y 可以是一个表示测试结果的变量。由于这些变量的值在人与人之间可能以通常未知的方式变化，它们被称为随机变量或随机变数。我们假设 X 可以取 x_i 中的任何值，其中 $i = 1, \dots, L$ ，并且 Y 可以取 y_j 中的值，其中 $j = 1, \dots, M$ 。考虑总共 N 次试验，在这些试验中我们对变量 X 和 Y 都进行采样，并且让 $X = x_i$ 和 $Y = y_j$ 同时发生的试验次数为 n_{ij} 。此外，让 X 取 x_i 值（无论 Y 取何值）的试验次数用 c_i 表示，类似地，让 Y 取 y_j 值的试验次数用 r_j 表示。

X 取值为 x_i 且 Y 取值为 y_j 的概率记为 $p(X = x_i, Y = y_j)$ ，称为 $X = x_i$ 和 $Y = y_j$ 的联合概率。它由落入单元格 i, j 的点数占总点数的比例给出，因此

$$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N}. \quad (2.1)$$

这里我们隐含地考虑了极限 $N \rightarrow \infty$ 。类似地， X 取值为 x_i 而与 Y 的值无关的概率记为 $p(X = x_i)$ ，由落入列 i 的总点数的比例给出，所以

图2.4我们可以通过考虑一个随机变量 X 来推导概率的加法和乘法规则，该随机变量取值为 $\{x_i\}$ ，其中 $i = 1, \dots, L$ ，以及另一个随机变量 Y ，其取值为 $\{y_j\}$ ，其中 $j = 1, \dots, M$ 。在这个示例中，我们有 $L = 5$ 和 $M = 3$ 。如果我们考虑这些变量的实例总数 N ，那么我们将 $X = x_i$ 和 $Y = y_j$ 同时出现的实例数记为 n_{ij} ，它是数组中相应单元格中的实例数。对应于 $X = x_i$ 的第 i 列中的实例数记为 c_i ，对应于 $Y = y_j$ 的第 j 行中的实例数记为 r_j 。



$$p(X = x_i) = \frac{c_i}{N} \quad (2.2)$$

自 $\sum_i c_i = N$ 以来，我们看到

$$\sum_{i=1}^L p(X = x_i) = 1 \quad (2.3)$$

因此，概率之和如要求的那样为一。因为图2.4中 i 列中的实例数就是该列每个单元格中实例数的总和，所以我们有 $c_i = \sum_j n_{ij}$ ，因此，根据(2.1)和(2.2)，我们有

$$p(X = x_i) = \sum_{j=1}^M p(X = x_i, Y = y_j), \quad (2.4)$$

这就是概率的求和规则。注意， $p(X = x_i)$ 有时被称为边缘概率，它是通过对其他变量（在这种情况下是 Y ）进行边缘化或求和得到的。

如果我们只考虑那些满足 $X = x_i$ 的情况，那么满足 $Y = y_j$ 的此类情况的比例记为 $p(Y = y_j | X = x_i)$ ，并称为在 $X = x_i$ 条件下 $Y = y_j$ 的条件概率。它是通过找出列 i 中落在单元格 i, j 中的那些点的比例得到的，因此由下式给出

$$p(Y = y_j | X = x_i) = \frac{n_{ij}}{c_i}. \quad (2.5)$$

对 j 两边求和并使用 $\sum_j n_{ij} = c_i$ ，我们得到

$$\sum_{j=1}^M p(Y = y_j | X = x_i) = 1 \quad (2.6)$$

表明条件概率已正确归一化。根据(2.1)、(2.2)和(2.5)，我们可以得出以下关系：

$$\begin{aligned} p(X = x_i, Y = y_j) &= \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N} \\ &= p(Y = y_j | X = x_i)p(X = x_i), \end{aligned} \quad (2.7)$$

这就是概率的乘积法则。

到目前为止，我们一直非常小心地区分随机变量，例如 X ，以及随机变量可以取的值，例如 x_i 。因此， X 取 x_i 值的概率表示为 $p(X = x_i)$ 。虽然这有助于避免歧义，但它会导致一种相当繁琐的表示法，而且在许多情况下不需要如此拘泥。相反，我们可以简单地写 $p(X)$ 来表示随机变量 X 上的分布，或者写 $p(x_i)$ 来表示针对特定值 x_i 评估的分布，前提是上下文可以清楚地理解其含义。

使用这种更紧凑的表示法，我们可以将概率论的两个基本规则写成以下形式：

$$\text{sum rule} \quad p(X) = \sum_Y p(X, Y) \quad (2.8)$$

$$\text{product rule} \quad p(X, Y) = p(Y | X)p(X). \quad (2.9)$$

这里 $p(X, Y)$ 是一个联合概率，表述为“ X 和 Y 的概率”。类似地，量 $p(Y | X)$ 是一个条件概率，表述为“在 X 给定的情况下 Y 的概率”。最后，量 $p(X)$ 是一个边缘概率，简单地说是“ X 的概率”。这两条简单规则构成了我们将在整本书中使用的所有概率机制的基础。

2.1.3 贝叶斯定理

根据乘积规则，结合对称性属性 $p(X, Y) = p(Y, X)$ ，我们立即得到以下条件概率之间的关系：

$$p(Y | X) = \frac{p(X | Y)p(Y)}{p(X)}, \quad (2.10)$$

这被称为贝叶斯定理，它在机器学习中起着重要作用。注意贝叶斯定理是如何将方程左侧的条件分布 $p(Y | X)$ 与右侧“反转”的条件分布 $p(X | Y)$ 联系起来的。利用求和规则，贝叶斯定理中的分母可以用分子中出现的量来表示：

$$p(X) = \sum_Y p(X | Y)p(Y). \quad (2.11)$$

因此，我们可以将贝叶斯定理中的分母视为归一化常数，它确保(2.10)左侧的条件概率分布在 Y 的所有值上的总和等于1。

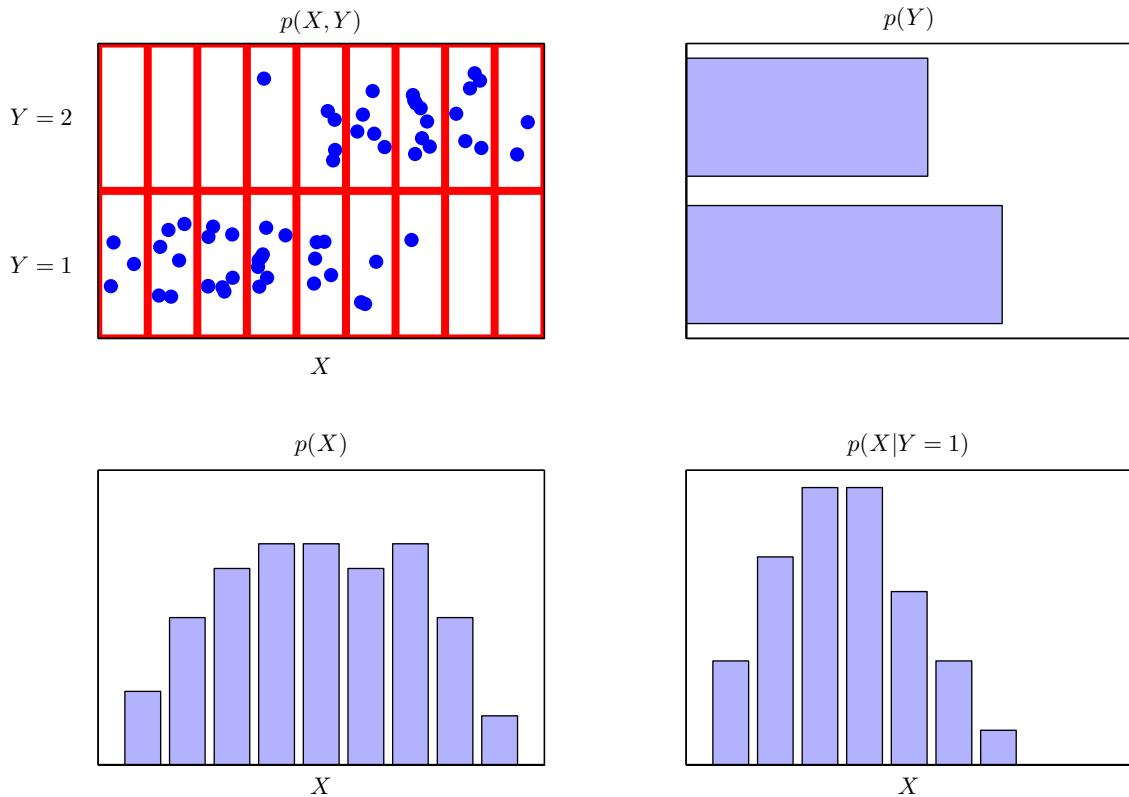


图2.5两个变量的分布示意图， X 有九个可能值， Y 有两个可能值。左上角的图展示了从这些变量的联合概率分布中抽取的60个点的样本。其余的图展示了边际分布 $p(X)$ 和 $p(Y)$ 的直方图估计，以及与左上角图底部行对应的条件分布 $p(X | Y = 1)$ 。

在图2.5中，我们展示了一个简单的示例，涉及两个变量的联合分布，以说明边际分布和条件分布的概念。这里，从联合分布中抽取了 $N = 60$ 个数据点的有限样本，并显示在左上角。右上角是具有 Y 两个值中每个值的数据点比例的直方图。根据概率的定义，当样本量 $N \rightarrow \infty$ 时，这些比例在极限情况下将等于相应的概率 $p(Y)$ 。我们可以将直方图视为一种简单的方法，用于在仅从该分布中抽取有限数量的点的情况下对概率分布进行建模。图2.5中的其余两个图显示了 $p(X)$ 和 $p(X | Y = 1)$ 的相应直方图估计。

第3.5.1节

2.1.4 重新审视医学筛查

现在让我们回到癌症筛查的例子，应用概率的求和与乘积规则来回答我们的两个问题。为了清晰起见，在处理这个例子时，我们将再次明确区分随机变量及其实例化。我们将用变量 C 表示是否患有癌症，它可以取两个值： $C = 0$ 对应“无癌症”， $C = 1$ 对应“有癌症”。我们假设人群中每一百人中有一人患有癌症，所以我们有

$$p(C = 1) = 1/100 \quad (2.12)$$

$$p(C = 0) = 99/100, \quad (2.13)$$

分别。请注意，这些满足 $p(C = 0) + p(C = 1) = 1$ 。

现在让我们引入第二个随机变量 T ，它代表一项筛查测试的结果，其中 $T = 1$ 表示阳性结果，意味着患有癌症，而 $T = 0$ 表示阴性结果，意味着没有癌症。如图2.3所示，我们知道对于那些患有癌症的人，检测结果为阳性的概率是 90%，而对于那些没有癌症的人，检测结果为阳性的概率是 3%。因此，我们可以写出所有四个条件概率：

$$p(T = 1|C = 1) = 90/100 \quad (2.14)$$

$$p(T = 0|C = 1) = 10/100 \quad (2.15)$$

$$p(T = 1|C = 0) = 3/100 \quad (2.16)$$

$$p(T = 0|C = 0) = 97/100. \quad (2.17)$$

再次注意，这些概率是归一化的，以便

$$p(T = 1|C = 1) + p(T = 0|C = 1) = 1 \quad (2.18)$$

类似地

$$p(T = 1|C = 0) + p(T = 0|C = 0) = 1. \quad (2.19)$$

我们现在可以使用概率的求和与乘积规则来回答我们的第一个问题，并评估随机接受检测的人检测结果为阳性的总体概率：

$$\begin{aligned} p(T = 1) &= p(T = 1|C = 0)p(C = 0) + p(T = 1|C = 1)p(C = 1) \\ &= \frac{3}{100} \times \frac{99}{100} + \frac{90}{100} \times \frac{1}{100} = \frac{387}{10,000} = 0.0387. \end{aligned} \quad (2.20)$$

我们发现，如果随机对一个人进行检测，即使此人实际患癌症的概率为 1%，检测结果呈阳性的概率仍大致为 4%。由此，根据求和规则可得 $p(T = 0) = 1 - 387/10,000 = 9613/10,000 = 0.9613$ ，因此，此人未患癌症的概率大致为 96%。

现在考虑我们的第二个问题，这也是接受筛查的人特别感兴趣的问题：如果检测呈阳性，那么此人患癌症的概率是多少？

这要求我们根据测试结果评估患癌的概率，而 (2.14) 至 (2.17) 中的概率给出的是根据某人是否患癌对测试结果的概率分布。我们可以通过使用贝叶斯定理 (2.10) 来解决反转条件概率的问题，从而得到

$$p(C = 1|T = 1) = \frac{p(T = 1|C = 1)p(C = 1)}{p(T = 1)} \quad (2.21)$$

$$= \frac{90}{100} \times \frac{1}{100} \times \frac{10,000}{387} = \frac{90}{387} \simeq 0.23 \quad (2.22)$$

因此，如果随机对一个人进行测试且测试结果为阳性，那么他们实际患癌的概率为23%。根据求和规则，由此可得 $p(C = 0|T = 1) = 1 - 90/387 = 297/387 \simeq 0.77$ ，即他们没有患癌的概率为77%。

2.1.5 先验概率和后验概率

我们可以用癌症筛查的例子来对贝叶斯定理进行如下重要诠释。如果在某人接受检测之前，有人问我们这个人患癌症的可能性有多大，那么我们所掌握的最全面信息由概率 $p(C)$ 提供。我们称其为先验概率，因为它是在我们观察检测结果之前就已知的概率。一旦我们得知这个人检测结果呈阳性，那么我们就可以使用贝叶斯定理来计算概率 $p(C|T)$ ，我们将其称为后验概率，因为它是在我们观察到检测结果 T 之后得到的概率。

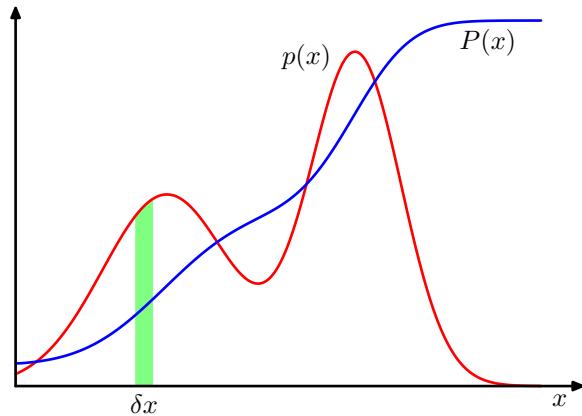
在这个例子中，患癌症的先验概率是1%。然而，一旦我们观察到检测结果为阳性，我们发现患癌症的后验概率现在是 23%，正如我们直观预期的那样，这是一个显著更高的患癌概率。然而，我们注意到，即使从图2.3来看该检测似乎相当“准确”，检测呈阳性的人实际患癌的几率仍只有23%。这个结论对许多人来说似乎有违直觉。原因与患癌的低先验概率有关。尽管该检测提供了患癌的有力证据，但必须使用贝叶斯定理将其与先验概率相结合，才能得出正确的后验概率。

2.1.6 自变量

最后，如果两个变量的联合分布可分解为它们各自边缘分布的乘积，即 $p(X, Y) = p(X)p(Y)$ ，那么 X 和 Y 被称为是独立的。独立事件的一个例子是抛硬币的连续抛掷。根据乘积规则，我们可以看到 $p(Y|X) = p(Y)$ ，所以给定 X 时 Y 的条件分布确实与 X 的值无关。在我们的癌症筛查示例中，如果检测呈阳性的概率与该人是否患有癌症无关，那么 $p(T|C) = p(T)$ ，这意味着根据贝叶斯定理 (2.10) 我们有 $p(C|T) = p(C)$ ，因此观察检测结果并不会改变患癌的概率。当然，这样的检测是无用的，因为检测结果并不能告诉我们该人是否患有癌症。

练习2.1

图2.6离散变量的概率概念可以扩展到连续变量 x 上的概率密度 $p(x)$ ，并且使得 x 落在区间 $(x, x + \delta x)$ 内的概率由 $p(x)\delta x$ 给出，其中 $\delta x \rightarrow 0$ 。概率密度可以表示为累积分布函数 $P(x)$ 的导数。



2.2. 概率密度

除了考虑定义在离散值集上的概率外，我们还希望考虑关于连续变量的概率。例如，我们可能希望预测给患者使用何种剂量的药物。由于这种预测存在不确定性，我们希望对这种不确定性进行量化，并且同样可以利用概率。然而，我们不能简单地直接应用到目前为止所讨论的概率概念，因为以无限精度观察连续变量的特定值的概率实际上将为零。相反，我们需要引入概率密度的概念。在这里，我们将把讨论限制在相对非正式的层面。

我们将连续变量 x 上的概率密度 $p(x)$ 定义为：对于 $\delta x \rightarrow 0$ ， x 落在区间 $(x, x + \delta x)$ 的概率由 $p(x)\delta x$ 给出。图2.6对此进行了说明。那么 x 落在区间 (a, b) 的概率由下式给出

$$p(x \in (a, b)) = \int_a^b p(x) dx. \quad (2.23)$$

由于概率是非负的，并且由于 x 的值必须位于实轴上的某个位置，概率密度 $p(x)$ 必须满足两个条件

$$p(x) \geq 0 \quad (2.24)$$

$$\int_{-\infty}^{\infty} p(x) dx = 1. \quad (2.25)$$

x 落在区间 $(-\infty, z)$ 的概率由定义的累积分布函数给出

$$P(z) = \int_{-\infty}^z p(x) dx, \quad (2.26)$$

如图2.6所示，其满足 $P'(x) = p(x)$ 。

如果我们有几个连续变量 x_1, \dots, x_D ，用向量 \mathbf{x} 共同表示，那么我们可以定义一个联合概率密度 $p(\mathbf{x}) = p(x_1, \dots, x_D)$ ，使得 \mathbf{x} 落在包含点 \mathbf{x} 的无穷小体积 $\delta\mathbf{x}$ 中的概率由 $p(\mathbf{x})\delta\mathbf{x}$ 给出。这个多元概率密度必须满足

$$p(\mathbf{x}) \geq 0 \quad (2.27)$$

$$\int p(\mathbf{x}) d\mathbf{x} = 1 \quad (2.28)$$

其中积分是在整个 \mathbf{x} 空间上进行的。更一般地，我们也可以考虑离散和连续变量组合的联合概率分布。

概率的加法和乘法规则以及贝叶斯定理，同样适用于概率密度以及离散变量和连续变量的组合。如果 \mathbf{x} 和 \mathbf{y} 是两个实变量，那么加法和乘法规则具有以下形式

$$\text{sum rule} \quad p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{y}) d\mathbf{y} \quad (2.29)$$

$$\text{product rule} \quad p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\mathbf{x})p(\mathbf{x}). \quad (2.30)$$

类似地，贝叶斯定理可以写成以下形式

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{y})p(\mathbf{y})}{p(\mathbf{x})} \quad (2.31)$$

其中分母由下式给出

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{y})p(\mathbf{y}) d\mathbf{y}. \quad (2.32)$$

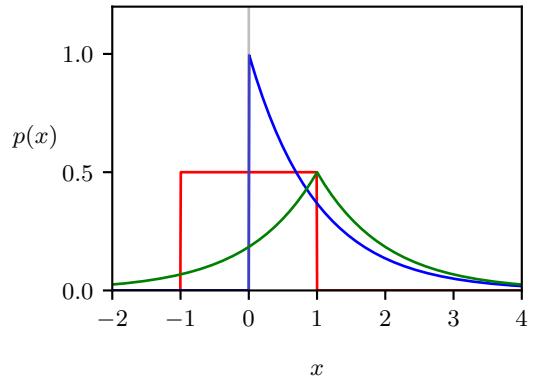
连续变量的加法和乘法规则的形式证明需要一门叫做测度论的数学分支（费勒，1966年），超出了本书的范围。然而，通过将每个实变量划分为宽度为 Δ 的区间，并考虑这些区间上的离散概率分布，可以直观地看到其有效性。取极限 $\Delta \rightarrow 0$ ，然后将求和转换为积分，即可得到所需结果。

2.2.1示例分布

有许多形式的概率密度被广泛使用，它们本身就很 important，同时也是构建更复杂概率模型的基石。最简单的形式是 $p(x)$ 为常数且与 x 无关的那种，但这无法归一化，因为(2.28)中的积分会发散。无法归一化的分布称为非恰当分布。然而，我们可以有在有限区域（比如 (c, d) ）上为常数而在其他地方为零的均匀分布，在这种情况下(2.28)意味着

$$p(x) = 1/(d - c), \quad x \in (c, d). \quad (2.33)$$

图2.7 范围为 $(-1, 1)$ 的均匀分布（红色）、参数为 $\lambda = 1$ 的指数分布（蓝色）以及参数为 $\mu = 1$ 和 $\gamma = 1$ 的拉普拉斯分布（绿色）。绘图。



另一种简单的密度形式是指数分布，由下式给出

$$p(x|\lambda) = \lambda \exp(-\lambda x), \quad x \geq 0. \quad (2.34)$$

指数分布的一种变体，称为拉普拉斯分布，它允许峰值移动到位置 μ ，其表达式为

$$p(x|\mu, \gamma) = \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right). \quad (2.35)$$

常数分布、指数分布和拉普拉斯分布如图2.7所示。

另一个重要的分布是狄拉克 δ 函数，其表达式为

$$p(x|\mu) = \delta(x - \mu). \quad (2.36)$$

除了在 $x = \mu$ 处，此函数在其他各处均定义为零，并具有根据(2.28)积分分为1的性质。通俗来讲，我们可以将其视为位于 $x = \mu$ 处的一个无限窄且无限高的尖峰，其面积为1。最后，如果我们有由 $\mathcal{D} = \{x_1, \dots, x_N\}$ 给出的 x 的有限观测集，那么我们可以使用狄拉克 δ 函数来构建经验分布，该经验分布由

$$p(x|\mathcal{D}) = \frac{1}{N} \sum_{n=1}^N \delta(x - x_n), \quad (2.37)$$

它由以每个数据点为中心的狄拉克 δ 函数组成。由(2.37)定义的概率密度按要求积分为1。

练习2.6

2.2.2 期望和协方差

涉及概率的最重要运算之一是求函数的加权平均值。概率分布 $p(x)$ 下某个函数 $f(x)$ 的加权平均值称为 $f(x)$ 的期望，将用 $\mathbb{E}[f]$ 表示。对于离散分布，它通过对 x 的所有可能值求和得到，形式如下

$$\mathbb{E}[f] = \sum_x p(x)f(x) \quad (2.38)$$

其中均值由 x 不同值的相对概率加权。对于连续变量，期望通过对相应概率密度进行积分来表示：

$$\mathbb{E}[f] = \int p(x)f(x) dx. \quad (2.39)$$

在任何一种情况下，如果我们给定从概率分布或概率密度中抽取的有限数量的 N 个点，那么期望可以近似为这些点上的有限和：

练习2.7

$$\mathbb{E}[f] \simeq \frac{1}{N} \sum_{n=1}^N f(x_n). \quad (2.40)$$

(2.40)中的近似在 $N \rightarrow \infty$ 的极限情况下变得精确。

有时我们会考虑几个变量的函数的期望，在这种情况下，我们可以使用下标来表示对哪个变量进行平均，例如

$$\mathbb{E}_x[f(x, y)] \quad (2.41)$$

表示函数 $f(x, y)$ 关于 x 分布的平均值。注意 $\mathbb{E}_x[f(x, y)]$ 将是 y 的函数。

我们也可以考虑关于条件分布的条件期望，所以

$$\mathbb{E}_x[f|y] = \sum_x p(x|y)f(x), \quad (2.42)$$

这也是 y 的一个函数。对于连续变量，条件期望具有以下形式

$$\mathbb{E}_x[f|y] = \int p(x|y)f(x) dx. \quad (2.43)$$

$f(x)$ 的方差定义为

$$\text{var}[f] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2] \quad (2.44)$$

并提供了 $f(x)$ 围绕其均值 $\mathbb{E}[f(x)]$ 变化程度的一种度量。展开平方项，我们可以看到方差也可以用 $f(x)$ 和 $f(x)^2$ 的期望来表示：

$$\text{var}[f] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2. \quad (2.45)$$

特别地，我们可以考虑变量 x 本身的方差，其定义为

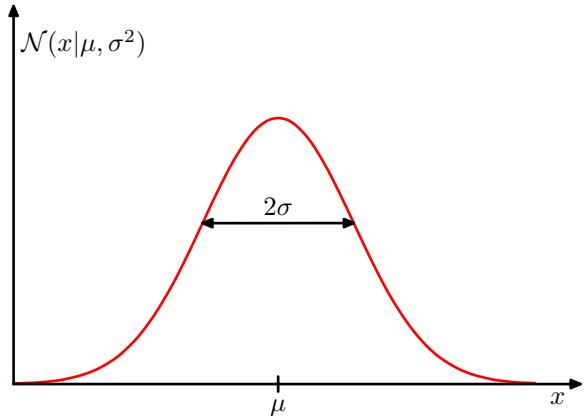
$$\text{var}[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2. \quad (2.46)$$

对于两个随机变量 x 和 y ，协方差衡量了这两个变量一起变化的程度，其定义为

$$\begin{aligned} \text{cov}[x, y] &= \mathbb{E}_{x,y}[\{x - \mathbb{E}[x]\}\{y - \mathbb{E}[y]\}] \\ &= \mathbb{E}_{x,y}[xy] - \mathbb{E}[x]\mathbb{E}[y]. \end{aligned} \quad (2.47)$$

练习2.8

图2.8 单个连续变量 x 的高斯分布曲线，显示了均值 μ 和标准差 σ 。



练习2.9

如果 x 和 y 相互独立，那么它们的协方差等于零。

对于两个向量 \mathbf{x} 和 \mathbf{y} ，它们的协方差是一个由下式给出的矩阵

$$\begin{aligned}\text{cov}[\mathbf{x}, \mathbf{y}] &= \mathbb{E}_{\mathbf{x}, \mathbf{y}} [\{\mathbf{x} - \mathbb{E}[\mathbf{x}]\}\{\mathbf{y}^T - \mathbb{E}[\mathbf{y}^T]\}] \\ &= \mathbb{E}_{\mathbf{x}, \mathbf{y}} [\mathbf{x}\mathbf{y}^T] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{y}^T].\end{aligned}\quad (2.48)$$

如果我们考虑向量 \mathbf{x} 各分量之间的协方差，那么我们使用一个稍微简单一点的符号表示 $\text{cov}[\mathbf{x}] \equiv \text{cov}[\mathbf{x}, \mathbf{x}]$ 。

2.3. 高斯分布

连续变量最重要的概率分布之一称为正态分布或高斯分布，我们将在本书其余部分广泛使用这种分布。对于单个实值变量 x ，高斯分布定义为

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}, \quad (2.49)$$

它表示由两个参数控制的 x 上的概率密度： μ ，称为均值，以及 σ^2 ，称为方差。方差的平方根，由 σ 给出，称为标准差，方差的倒数，写作 $\beta = 1/\sigma^2$ ，称为精度。我们很快就会看到这种术语的动机。图2.8展示了高斯分布的曲线图。虽然高斯分布的形式可能看起来很随意，但我们稍后会看到它自然地源于最大熵的概念以及中心极限定理的观点。

第2.5.4节第 3.2节

从(2.49)我们可以看出高斯分布满足

$$\mathcal{N}(x|\mu, \sigma^2) > 0. \quad (2.50)$$

此外，很容易证明高斯分布是归一化的，所以

$$\int_{-\infty}^{\infty} \mathcal{N}(x|\mu, \sigma^2) dx = 1. \quad (2.51)$$

因此，(2.49)满足有效概率密度的两个要求。

2.3.1 均值和方差

在高斯分布下，我们可以很容易地求出 x 函数的期望。特别地， x 的平均值由下式给出

$$\mathbb{E}[x] = \int_{-\infty}^{\infty} \mathcal{N}(x|\mu, \sigma^2) x dx = \mu. \quad (2.52)$$

因为参数 μ 代表分布下 x 的平均值，所以它被称为均值。(2.52)中的积分被称为分布的一阶矩，因为它是 x 的一次幂的期望。我们可以类似地计算由下式给出的二阶矩

$$\mathbb{E}[x^2] = \int_{-\infty}^{\infty} \mathcal{N}(x|\mu, \sigma^2) x^2 dx = \mu^2 + \sigma^2. \quad (2.53)$$

由(2.52)和(2.53)可知， x 的方差由下式给出

$$\text{var}[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2 = \sigma^2 \quad (2.54)$$

因此， σ^2 被称为方差参数。分布的最大值称为其众数。对于高斯分布，众数与均值重合。

练习2.14

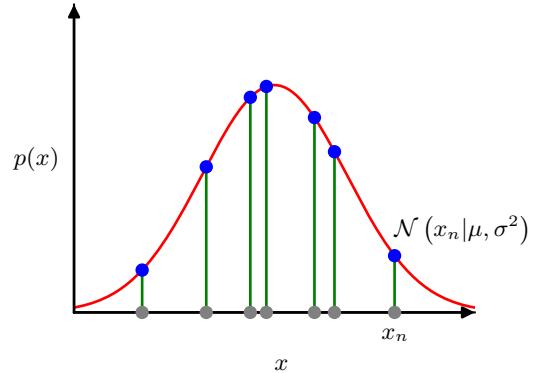
2.3.2 似然函数

假设我们有一个观测数据集，表示为行向量 $\mathbf{x} = (x_1, \dots, x_N)$ ，代表标量变量 x 的 N 次观测。请注意，我们使用字体 \mathbf{x} 来将其与 D 维向量值变量的单个观测区分开来，我们用列向量 $\mathbf{x} = (x_1, \dots, x_D)^T$ 来表示该变量。我们假设观测值是从均值 μ 和方差 σ^2 未知的高斯分布中独立抽取的，并且我们希望从数据集中确定这些参数。给定一组有限观测值来估计分布的问题称为密度估计。应该强调的是，密度估计问题从根本上说是不稳定的，因为有无限多个概率分布可能产生观测到的有限数据集。实际上，在每个数据点 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 处非零的任何分布 $p(\mathbf{x})$ 都是潜在的候选者。在这里，我们将分布空间限制为高斯分布，这导致了一个明确的解决方案。

从同一分布中独立抽取的数据点被称为独立同分布，通常缩写为 i.i.d. 或 IID。我们已经看到，两个独立事件的联合概率等于每个事件的边际概率之积。

图2.9 高斯分布的似然函数图示，红色曲线所示。

这里灰色点表示值为 $\{x_n\}$ 的数据集，似然函数(2.55)由蓝色点表示的 $p(x)$ 的相应值的乘积给出。最大化似然涉及调整高斯的均值和方差，以便最大化这个乘积。



因为我们的数据集 \mathbf{x} 是独立同分布的，所以我们可以将给定 μ 和 σ^2 时数据集的概率写成如下形式

$$p(\mathbf{x}|\mu, \sigma^2) = \prod_{n=1}^N \mathcal{N}(x_n|\mu, \sigma^2). \quad (2.55)$$

当被视为 μ 和 σ^2 的函数时，这称为高斯的似然函数，并在图2.9中以图表形式进行解释。

一种使用观测数据集来确定概率分布中参数的常见方法，即最大似然法，是找到使似然函数最大化的参数值。这可能看起来是一个奇怪的标准，因为从我们前面关于概率论的讨论来看，最大化给定数据下参数的概率，而不是给定参数下数据的概率，似乎更自然。实际上，这两个标准是相关的。

然而，首先，我们将通过最大化似然函数 (2.55) 来确定高斯分布中未知参数 μ 和 σ^2 的值。在实践中，最大化似然函数的对数更方便。因为对数是其自变量的单调递增函数，最大化函数的对数等同于最大化函数本身。取对数不仅简化了后续的数学分析，而且在数值计算上也有帮助，因为大量小概率的乘积很容易导致计算机的数值精度下溢，而通过计算对数概率的和可以解决这个问题。根据 (2.49) 和 (2.55)，对数似然函数可以写成以下形式

$$\ln p(\mathbf{x}|\mu, \sigma^2) = -\frac{1}{2\sigma^2} \sum_{n=1}^N (x_n - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi). \quad (2.56)$$

关于 μ 最大化(2.56)，我们得到如下给出的最大似然解

练习2.15

$$\mu_{\text{ML}} = \frac{1}{N} \sum_{n=1}^N x_n, \quad (2.57)$$

它是样本均值，即观测值 $\{x_n\}$ 的均值。类似地，对 σ^2 最大化(2.56)，我们得到方差的最大似然解，形式为

$$\sigma_{\text{ML}}^2 = \frac{1}{N} \sum_{n=1}^N (x_n - \mu_{\text{ML}})^2, \quad (2.58)$$

它是相对于样本均值 μ_{ML} 测量的样本方差。注意，我们正在对(2.56)关于 μ 和 σ^2 进行联合最大化，但对于高斯分布， μ 的解与 σ^2 的解解耦，因此我们可以首先评估(2.57)，然后随后使用此结果评估(2.58)。

2.3.3 最大似然的偏差

最大似然技术在深度学习中被广泛使用，并且是大多数机器学习算法的基础。然而，它有一些局限性，我们可以用单变量高斯来举例说明。

我们首先注意到，最大似然解 μ_{ML} 和 σ_{ML}^2 是数据集值 x_1, \dots, x_N 的函数。假设这些值中的每一个都是从真实参数为 μ 和 σ^2 的高斯分布中独立生成的。现在考虑 μ_{ML} 和 σ_{ML}^2 关于这些数据集值的期望。很容易证明

练习2.16

$$\mathbb{E}[\mu_{\text{ML}}] = \mu \quad (2.59)$$

$$\mathbb{E}[\sigma_{\text{ML}}^2] = \left(\frac{N-1}{N}\right)\sigma^2. \quad (2.60)$$

我们看到，当在给定大小的数据集上进行平均时，均值的最大似然解将等于真实均值。然而，方差的最大似然估计将把真实方差低估一个因子 $(N-1)/N$ 。这是一种称为偏差的现象的一个例子，其中随机量的估计器与真实值系统地不同。图2.10给出了这个结果背后的直觉。

请注意，偏差的出现是因为方差是相对于均值的最大似然估计来衡量的，而均值本身是根据数据进行调整的。相反，假设我们可以获取真实均值 μ ，并使用它通过估计器来确定方差

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{n=1}^N (x_n - \mu)^2. \quad (2.61)$$

练习2.17

然后我们发现

$$\mathbb{E}[\hat{\sigma}^2] = \sigma^2, \quad (2.62)$$

它是无偏的。当然，我们无法获取真实均值，只能获取观测到的数据值。根据结果(2.60)，对于高斯分布，方差参数的以下估计是无偏的：

$$\tilde{\sigma}^2 = \frac{N}{N-1} \sigma_{\text{ML}}^2 = \frac{1}{N-1} \sum_{n=1}^N (x_n - \mu_{\text{ML}})^2. \quad (2.63)$$

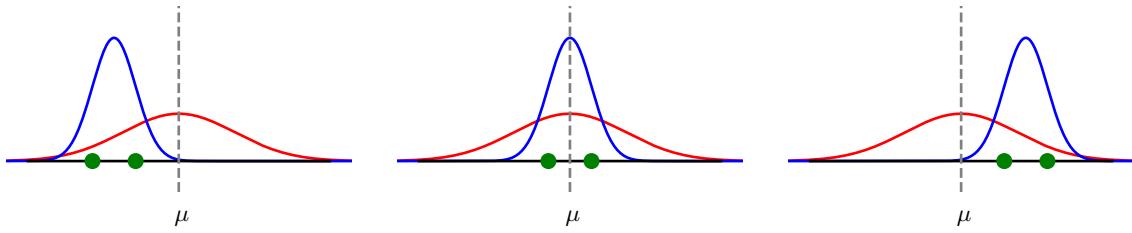


图2.10展示了在使用最大似然法确定高斯分布的均值和方差时偏差是如何产生的。红色曲线表示生成数据的真实高斯分布，三条蓝色曲线表示通过使用最大似然结果(2.57)和(2.58)拟合三个数据集（每个数据集由两个绿色显示的数据点组成）所获得的高斯分布。在这三个数据集上进行平均，均值是正确的，但方差被系统地低估了，因为它是相对于样本均值而不是相对于真实均值来测量的。

然而，在诸如神经网络这样的复杂模型中校正最大似然法的偏差并非易事。

请注意，随着数据点数量 N 的增加，最大似然解的偏差变得不那么显著。在极限情况 $N \rightarrow \infty$ 下，方差的最大似然解等于生成数据的分布的真实方差。对于高斯分布，除了 N 较小时，这种偏差不会被证明是一个严重问题。然而，在本书中，我们将关注具有许多参数的复杂模型，对于这些模型，与最大似然相关的偏差问题将更加严重。事实上，最大似然中的偏差问题与过拟合问题密切相关。

第2.6.3节

2.3.4 线性回归

第1.2节 我们已经看到线性回归问题如何用误差最小化来表示。在这里，我们回到这个例子，并从概率角度来看待它，从而对误差函数和正则化有一些深入了解。

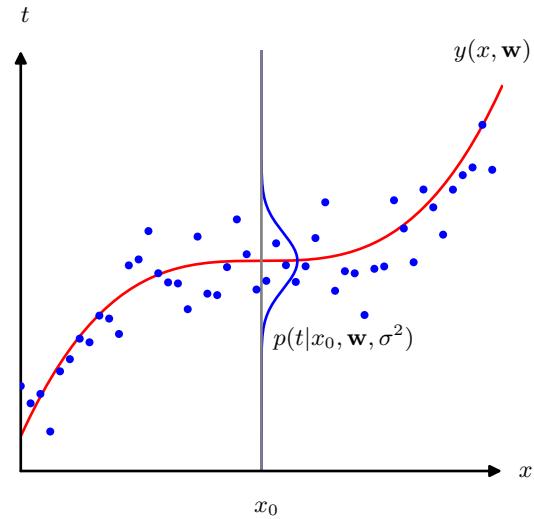
回归问题的目标是，通过使用一组包含 N 个输入值 $\mathbf{x} = (x_1, \dots, x_N)$ 及其相应目标值 $\mathbf{t} = (t_1, \dots, t_N)$ 的训练数据，能够针对输入变量 x 的某个新值对目标变量 t 进行预测。我们可以使用概率分布来表示我们对目标变量值的不确定性。为此，我们将假设，给定 x 的值， t 的相应值具有高斯分布，其均值等于由(1.1)给出的多项式曲线的值 $y(x, \mathbf{w})$ ，其中 \mathbf{w} 是多项式系数，方差为 σ^2 。因此，我们有

$$p(t|x, \mathbf{w}, \sigma^2) = \mathcal{N}(t|y(x, \mathbf{w}), \sigma^2). \quad (2.64)$$

图2.11以示意图的形式说明了这一点。

我们现在使用训练数据 $\{\mathbf{x}, \mathbf{t}\}$ ，通过最大似然法来确定未知参数 \mathbf{w} 和 σ^2 的值。

图2.11 给定 x 时 t 的高斯条件分布的示意图，由(2.64)定义，其中均值由多项式函数 $y(x, \mathbf{w})$ 给出，方差由参数 σ^2 给出。



如果假设数据是从分布(2.64)中独立抽取的，那么似然函数由下式给出

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \sigma^2) = \prod_{n=1}^N \mathcal{N}(t_n | y(x_n, \mathbf{w}), \sigma^2). \quad (2.65)$$

正如我们之前对简单高斯分布所做的那样，最大化似然函数的对数会很方便。代入由(2.49)给出的高斯分布，我们得到形式为的对数似然函数

$$\ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \sigma^2) = -\frac{1}{2\sigma^2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi). \quad (2.66)$$

首先考虑多项式系数的最大似然解的评估，它将由 \mathbf{w}_{ML} 表示。这些是通过相对于 \mathbf{w} 最大化(2.66)来确定的。为此，我们可以省略(2.66)右侧的最后两项，因为它们不依赖于 \mathbf{w} 。此外，请注意，用正的常数系数缩放对数似然不会改变相对于 \mathbf{w} 的最大值的位置，因此我们可以用 $1/2$ 替换系数 $1/2\sigma^2$ 。最后，与其最大化对数似然，我们可以等效地最小化负对数似然。因此我们看到，就确定 \mathbf{w} 而言，最大化似然等效于最小化由下式定义的平方和误差函数

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2. \quad (2.67)$$

因此，平方和误差函数是在高斯噪声分布假设下最大化似然的结果。

练习2.18 我们也可以使用最大似然法来确定方差参数 σ^2 。关于 σ^2 最大化 (2.66) 可得

$$\sigma_{\text{ML}}^2 = \frac{1}{N} \sum_{n=1}^N \{y(x_n, \mathbf{w}_{\text{ML}}) - t_n\}^2. \quad (2.68)$$

注意，我们可以首先确定控制均值的参数向量 \mathbf{w}_{ML} ，然后像简单高斯分布那样，用它来求方差 σ_{ML}^2 。

确定了参数 \mathbf{w} 和 σ^2 之后，我们现在可以对 x 的新值进行预测。由于我们现在有了一个概率模型，这些预测是根据预测分布来表达的，该分布给出了 t 上的概率分布，而不是简单的点估计，它是通过将最大似然参数代入(2.64)得到的，即

$$p(t|x, \mathbf{w}_{\text{ML}}, \sigma_{\text{ML}}^2) = \mathcal{N}(t|y(x, \mathbf{w}_{\text{ML}}), \sigma_{\text{ML}}^2). \quad (2.69)$$

2.4. 密度变换

我们现在转向讨论概率密度在变量的非线性变换下如何变换。当我们讨论一类称为归一化流的生成模型时，这个性质将发挥关键作用。它还突出表明，在这种变换下，概率密度的行为与简单函数不同。

考虑单个变量 x ，假设我们进行变量变换 $x = g(y)$ ，那么函数 $f(x)$ 就变成了一个由下式定义的新函数 $\tilde{f}(y)$

$$\tilde{f}(y) = f(g(y)). \quad (2.70)$$

现在考虑一个概率密度 $p_x(x)$ ，并再次使用 $x = g(y)$ 进行变量变换，从而得到一个关于新变量 y 的密度 $p_y(y)$ ，其中下标表示 $p_x(x)$ 和 $p_y(y)$ 是不同的密度。对于 δx 的小值，落在范围 $(x, x + \delta x)$ 内的观测值将被变换到范围 $(y, y + \delta y)$ ，其中 $x = g(y)$ ，且 $p_x(x)\delta x \simeq p_y(y)\delta y$ 。因此，如果我们取极限 $\delta x \rightarrow 0$ ，我们得到

$$\begin{aligned} p_y(y) &= p_x(x) \left| \frac{dx}{dy} \right| \\ &= p_x(g(y)) \left| \frac{dg}{dy} \right|. \end{aligned} \quad (2.71)$$

这里出现模量 $|\cdot|$ 是因为导数 dy/dx 可能为负，而密度是按长度比缩放的，长度比始终为正。

这种变换密度的方法可能非常强大。通过进行变量 $y = f(x)$ 的非线性变换，其中 $f(x)$ 是单调函数，使得 $0 \leq f'(x) < \infty$ ，任何密度 $p(y)$ 都可以从处处非零的固定密度 $q(x)$ 获得。

练习2.19

变换性质 (2.71) 的一个结果是，概率密度最大值的概念取决于变量的选择。假设 $f(x)$ 在 \hat{x} 处有一个众数（即最大值），使得 $f'(\hat{x}) = 0$ 。 $\tilde{f}(y)$ 的相应众数将出现在通过对 (2.70) 两边关于 y 求导得到的值 \hat{y} 处：

$$\tilde{f}'(\hat{y}) = f'(g(\hat{y}))g'(\hat{y}) = 0. \quad (2.72)$$

假设在众数处 $g'(\hat{y}) \neq 0$ ，那么 $f'(g(\hat{y})) = 0$ 。然而，我们知道 $f'(\hat{x}) = 0$ ，所以我们看到用变量 x 和 y 表示的众数位置通过 $\hat{x} = g(\hat{y})$ 相关联，正如人们所期望的那样。因此，相对于变量 x 找到众数等同于首先转换到变量 y ，然后相对于 y 找到众数，然后再转换回 x 。

现在考虑概率密度 $p_x(x)$ 在变量 $x = g(y)$ 变化下的行为，其中关于新变量的密度为 $p_y(y)$ ，并由(2.71)给出。为了处理(2.71)中的模，我们可以写成 $g'(y) = s|g'(y)|$ ，其中 $s \in \{-1, +1\}$ 。那么(2.71)可以写成

$$p_y(y) = p_x(g(y))sg'(y)$$

其中我们使用了 $1/s = s$ 。然后对两边关于 y 求导得到

$$p'_y(y) = sp'_x(g(y))\{g'(y)\}^2 + sp_x(g(y))g''(y). \quad (2.73)$$

由于(2.73)式右侧存在第二项，关系 $\hat{x} = g(\hat{y})$ 不再成立。因此，通过最大化 $p_x(x)$ 得到的 x 值，将不同于先转换为 $p_y(y)$ ，再对 y 进行最大化，然后再转换回 x 所得到的值。这使得密度模式依赖于变量的选择。然而，对于线性变换，(2.73)式右侧的第二项消失，所以在这种情况下，最大值的位置根据 $\hat{x} = g(\hat{y})$ 进行变换。

这个效果可以用一个简单的例子来说明，如图2.12所示。我们从考虑图2.12中红色曲线所示的关于 x 的高斯分布 $p_x(x)$ 开始。接下来，我们从这个分布中抽取 $N = 50,000$ 个点的样本，并绘制它们值的直方图，正如预期的那样，该直方图与分布 $p_x(x)$ 一致。现在考虑从 x 到 y 的变量的非线性变换，由

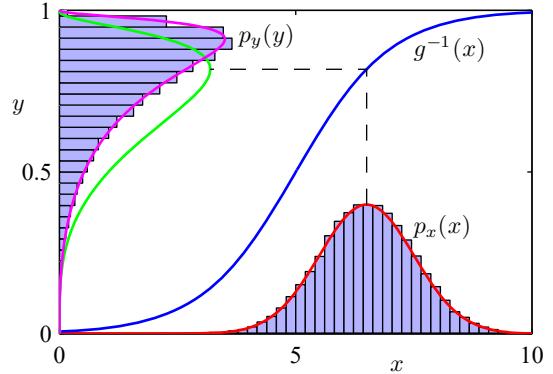
$$x = g(y) = \ln(y) - \ln(1-y) + 5. \quad (2.74)$$

这个函数的逆由

$$y = g^{-1}(x) = \frac{1}{1 + \exp(-x+5)}, \quad (2.75)$$

给出，它是一个逻辑 sigmoid 函数，在图2.12中由蓝色曲线表示。

图2.12变量非线性变化下密度模式转换的示例，说明了与简单函数相比的不同行为。



如果我们简单地将 $p_x(x)$ 作为 x 的函数进行变换，就会得到图2.12中所示的绿色曲线 $p_x(g(y))$ ，并且我们看到密度 $p_x(x)$ 的众数通过sigmoid函数变换为此曲线的众数。然而， y 上的密度反而根据(2.71)进行变换，并由图左侧的品红色曲线表示。请注意，其众数相对于绿色曲线的众数发生了偏移。

为了证实这一结果，我们取50000个 x 值的样本，使用(2.75)计算 y 的相应值，然后绘制它们值的直方图。我们看到这个直方图与图2.12中的品红色曲线匹配，而不是绿色曲线。

2.4.1 多元分布

我们可以将结果(2.71)扩展到定义在多个变量上的密度函数。考虑一个关于 D 维变量 $\mathbf{x} = (x_1, \dots, x_D)^T$ 的密度函数 $p(\mathbf{x})$ ，并假设我们变换到一个新变量 $\mathbf{y} = (y_1, \dots, y_D)^T$ ，其中 $\mathbf{x} = \mathbf{g}(\mathbf{y})$ 。这里我们将自己限制在 \mathbf{x} 和 \mathbf{y} 具有相同维度的情况下。那么变换后的密度函数由(2.71)的推广形式给出

$$p_{\mathbf{y}}(\mathbf{y}) = p_{\mathbf{x}}(\mathbf{x}) |\det \mathbf{J}| \quad (2.76)$$

其中 \mathbf{J} 是雅可比矩阵，其元素由偏导数 $J_{ij} = \partial g_i / \partial y_j$ 给出，因此

$$\mathbf{J} = \begin{bmatrix} \frac{\partial g_1}{\partial y_1} & \cdots & \frac{\partial g_1}{\partial y_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_D}{\partial y_1} & \cdots & \frac{\partial g_D}{\partial y_D} \end{bmatrix}. \quad (2.77)$$

直观地说，我们可以将变量的变化视为扩展空间的某些区域并收缩其他区域，围绕点 \mathbf{x} 的无穷小区域 $\Delta \mathbf{x}$ 被变换为围绕点 $\mathbf{y} = \mathbf{g}(\mathbf{x})$ 的区域 $\Delta \mathbf{y}$ 。雅可比行列式的绝对值表示这些体积的比率，并且是在积分内进行变量变换时出现的相同因子。

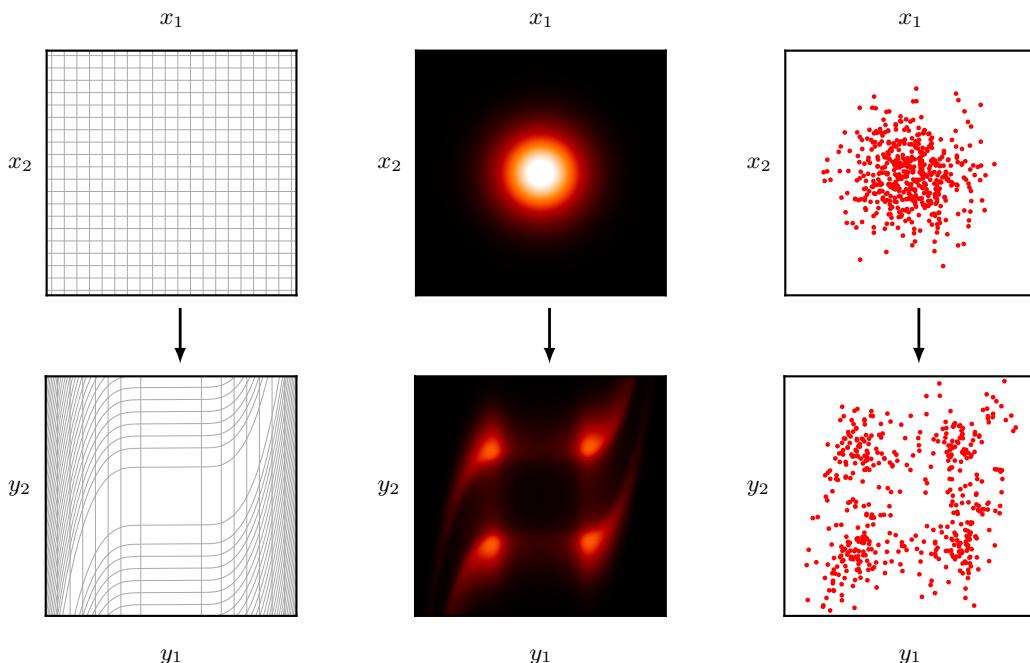


图2.13二维变量变化对概率分布影响的示意图。左列展示变量的变换，而中间和右列分别展示对高斯分布及其样本的相应影响。

公式(2.77)源于区域 $\Delta\mathbf{x}$ 中的概率质量与 $\Delta\mathbf{y}$ 中的概率质量相同这一事实。同样，我们取模以确保密度非负。

我们可以通过对二维高斯分布应用变量变换来说明这一点，如图2.13的上一行所示。这里从 \mathbf{x} 到 \mathbf{y} 的变换由下式给出
练习2.20

$$y_1 = x_1 + \tanh(5x_1) \quad (2.78)$$

$$y_2 = x_2 + \tanh(5x_2) + \frac{x_1^3}{3}. \quad (2.79)$$

底行还展示了 \mathbf{x} 空间中高斯分布的样本以及 \mathbf{y} 空间中相应的变换样本。

2.5. 信息论

概率论构成了另一个重要框架——信息论的基础，信息论对数据集中存在的信息进行量化，并且在机器学习中发挥着重要作用。在这里，我们简要介绍信息论的一些关键要素，这些要素在本书后面我们会用到，包括各种形式的熵这一重要概念。有关信息论与机器学习联系的更全面介绍，请参阅麦凯（2003）。

2.5.1 熵

我们首先考虑一个离散随机变量 x ，并询问当我们观察到该变量的特定值时会接收到多少信息。信息量可以被视为得知 x 的值时的“惊讶程度”。如果我们被告知一个极不可能发生的事件刚刚发生，那么我们接收到的信息将比被告知某个非常可能发生的事件刚刚发生时更多；如果我们知道该事件肯定会发生，那么我们将不会接收到任何信息。因此，我们对信息内容的度量将取决于概率分布 $p(x)$ ，所以我们寻找一个量 $h(x)$ ，它是概率 $p(x)$ 的单调函数，并表示信息内容。 $h(\cdot)$ 的形式可以通过注意到如果我们有两个不相关的事件 x 和 y ，那么从观察这两个事件中获得的信息应该是分别从每个事件中获得的信息之和，即 $h(x, y) = h(x) + h(y)$ 。两个不相关的事件在统计上是独立的，所以 $p(x, y) = p(x)p(y)$ 。从这两个关系中，可以很容易地证明 $h(x)$ 必须由 $p(x)$ 的对数给出，所以我们有

练习2.21

$$h(x) = -\log_2 p(x) \quad (2.80)$$

负号确保信息为正或零。注意，低概率事件 x 对应高信息内容。对数的底数选择是任意的，目前我们将采用信息论中常用的以2为底的对数约定。在这种情况下，正如我们很快将看到的， $h(x)$ 的单位是比特（“二进制数字”）。

现在假设发送方希望将一个随机变量的值传输给接收方。他们在这个过程中传输的平均信息量是通过对分布 $p(x)$ 求(2.80)的期望得到的，如下所示

$$H[x] = -\sum_x p(x) \log_2 p(x). \quad (2.81)$$

这个重要的量被称为随机变量 x 的熵。注意 $\lim_{\epsilon \rightarrow 0} (\epsilon \ln \epsilon) = 0$ ，所以每当遇到 x 的一个值使得 $p(x) = 0$ 时，我们将取 $p(x) \ln p(x) = 0$ 。

到目前为止，我们对信息 (2.80) 和相应的熵 (2.81) 的定义给出了一个相当启发式的动机。现在我们表明这些定义确实具有有用性质。

考虑一个具有八个可能状态的随机变量 x ，每个状态的可能性相同。为了将 x 的值传达给接收者，我们需要传输一条长度为3位的消息。注意，这个变量的熵由下式给出

$$H[x] = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3 \text{ bits.}$$

现在考虑一个例子 (Cover和Thomas, 1991)，一个具有八个可能状态 $\{a, b, c, d, e, f, g, h\}$ 的变量，其各自的概率由 $(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64})$ 给出。在这种情况下，熵由下式给出

$$H[x] = -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{4} \log_2 \frac{1}{4} - \frac{1}{8} \log_2 \frac{1}{8} - \frac{1}{16} \log_2 \frac{1}{16} - \frac{4}{64} \log_2 \frac{1}{64} = 2 \text{ bits.}$$

我们发现非均匀分布的熵比均匀分布的熵小，当我们稍后讨论熵与无序性的关系时，将对此有更深入的理解。目前，让我们考虑如何将变量状态的标识传送给接收者。和之前一样，我们可以用一个3位数字来做到这一点。然而，我们可以利用非均匀分布，对于更可能出现的事件使用较短的编码，代价是对于不太可能出现的事件使用较长的编码，以期获得更短的平均编码长度。例如，可以通过使用以下一组编码字符串来表示状态 $\{a, b, c, d, e, f, g, h\}$ ：0、10、110、1110、111100、111101、111110和111111。那么需要传输的编码的平均长度为

$$\text{平均码长} = \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{16} \times 4 + 4 \times \frac{1}{64} \times 6 = 2 \text{ 比特,}$$

这再次与随机变量的熵相同。请注意，不能使用更短的编码字符串，因为必须能够将此类字符串的串联明确地解析为其组成部分。例如，11001110 唯一地解码为状态序列 c, a, d 。熵与最短编码长度之间的这种关系是普遍存在的。无噪声编码定理 (香农, 1948 年) 指出，熵是传输随机变量状态所需比特数的下限。

从现在起，我们将转而使用自然对数来定义熵，因为这将为本书其他地方的观点提供更便利的联系。在这种情况下，熵的度量单位是奈特（源于“自然对数”）而非比特，二者仅相差 $\ln 2$ 倍。

2.5.2 物理学视角

我们从指定随机变量状态所需的平均信息量的角度引入了熵的概念。事实上，熵的概念在物理学中有更早的起源，它最初是在平衡态热力学的背景下引入的，后来通过统计力学的发展被赋予了作为无序度量的更深刻解释。我们可以通过考虑一组 N 个相同的物体来理解熵的这种不同观点，这些物体要被分配到一组箱子中，使得第 i 个箱子中有 n_i 个物体。

考虑将对象分配到箱子的不同方式的数量。有 N 种方法选择第一个对象， $(N - 1)$ 种方法选择第二个对象，依此类推，导致将所有 N 个对象分配到箱子的总方法数为 $N!$ ，其中 $N!$ (读作“ N 阶乘”) 表示乘积 $N \times (N - 1) \times \cdots \times 2 \times 1$ 。然而，我们不希望区分每个箱子内对象的重新排列。在第 i 个箱子中有 $n_i!$ 种重新排列对象的方法，因此将 N 个对象分配到箱子的总方法数由下式给出

$$W = \frac{N!}{\prod_i n_i!}, \quad (2.82)$$

这被称为多重性。然后，熵被定义为多重性的对数，乘以一个常数因子 $1/N$ ，使得

$$H = \frac{1}{N} \ln W = \frac{1}{N} \ln N! - \frac{1}{N} \sum_i \ln n_i!. \quad (2.83)$$

我们现在考虑极限 $N \rightarrow \infty$ ，其中分数 n_i/N 保持固定，并应用斯特林近似：

$$\ln N! \simeq N \ln N - N, \quad (2.84)$$

其结果为

$$H = - \lim_{N \rightarrow \infty} \sum_i \left(\frac{n_i}{N} \right) \ln \left(\frac{n_i}{N} \right) = - \sum_i p_i \ln p_i \quad (2.85)$$

这里我们使用了 $\sum_i n_i = N$ 。这里 $p_i = \lim_{N \rightarrow \infty} (n_i/N)$ 是一个物体被分配到第 i 个箱子的概率。在物理术语中，物体在箱子中的具体分配称为微观状态，而通过比率 n_i/N 表示的占据数的总体分布称为宏观状态。多重性 W ，它表示给定宏观状态下的微观状态数，也称为宏观状态的权重。

我们可以将这些箱视为离散随机变量 X 的状态 x_i ，其中 $p(X = x_i) = p_i$ 。那么随机变量 X 的熵为

$$H[p] = - \sum_i p(x_i) \ln p(x_i). \quad (2.86)$$

在少数几个值附近尖锐峰值的分布 $p(x_i)$ 将具有相对较低的熵，而那些在许多值上更均匀分布的分布将具有较高的熵，如图2.14所示。

因为 $0 \leq p_i \leq 1$ ，所以熵是非负的，并且当其中一个 $p_i = 1$ 以及所有其他 $p_{j \neq i} = 0$ 时，它将等于其最小值0。最大熵配置可以通过使用拉格朗日乘数最大化 H 来找到，以强制对概率进行归一化约束。因此，我们最大化

$$\tilde{H} = - \sum_i p(x_i) \ln p(x_i) + \lambda \left(\sum_i p(x_i) - 1 \right) \quad (2.87)$$

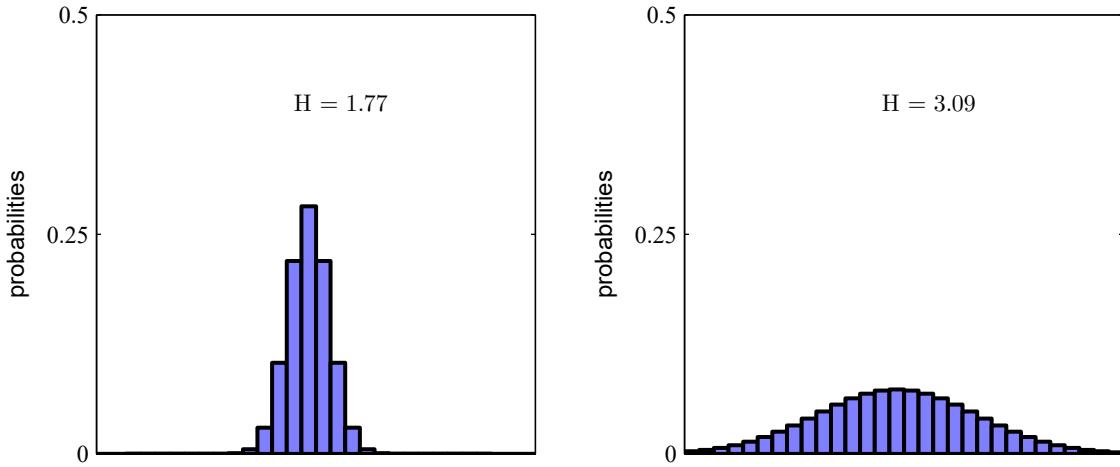


图2.14 30个区间上两个概率分布的直方图，说明了更宽泛分布的熵 H 值更高。最大熵将来自均匀分布，其结果为 $H = -\ln(1/30) = 3.40$.

练习2.22
练习2.23

由此我们发现所有的 $p(x_i)$ 都相等，且由 $p(x_i) = 1/M$ 给出，其中 M 是状态 x_i 的总数。那么熵的相应值为 $H = \ln M$ 。这个结果也可以从詹森不等式推导得出（稍后讨论）。为了验证驻点确实是最大值，我们可以计算熵的二阶导数，结果为

$$\frac{\partial \tilde{H}}{\partial p(x_i) \partial p(x_j)} = -I_{ij} \frac{1}{p_i} \quad (2.88)$$

其中 I_{ij} 是单位矩阵的元素。我们看到这些值都是负的，因此，驻点确实是最大值。

2.5.3 微分熵

我们可以将熵的定义扩展到连续变量 x 上的分布 $p(x)$ ，如下所示。首先将 x 划分为宽度为 Δ 的区间。然后，假设 $p(x)$ 是连续的，均值定理 (Weisstein, 1999) 告诉我们，对于每个这样的区间，在范围 $i\Delta \leq x_i \leq (i+1)\Delta$ 内必定存在一个值 x_i ，使得

$$\int_{i\Delta}^{(i+1)\Delta} p(x) dx = p(x_i) \Delta. \quad (2.89)$$

现在我们可以通过每当 x 落在第 i 个区间时将任何值 x 赋给值 x_i 来对连续变量 x 进行量化。那么观察到值 x_i 的概率就是 $p(x_i)\Delta$ 。

这给出了一个离散分布，其熵具有以下形式

$$H_{\Delta} = - \sum_i p(x_i) \Delta \ln(p(x_i) \Delta) = - \sum_i p(x_i) \Delta \ln p(x_i) - \ln \Delta \quad (2.90)$$

其中我们使用了 $\sum_i p(x_i) \Delta = 1$ ，它由(2.89)和(2.25)得出。我们现在省略

(2.90)式右边的第二项 $-\ln \Delta$ ，因为它与 $p(x)$ 无关，然后考虑极限 $\Delta \rightarrow 0$ 。在这个极限下，(2.90)式右边的第一项将趋近于 $p(x) \ln p(x)$ 的积分，于是

$$\lim_{\Delta \rightarrow 0} \left\{ - \sum_i p(x_i) \Delta \ln p(x_i) \right\} = - \int p(x) \ln p(x) dx \quad (2.91)$$

等式右边的量被称为微分熵。我们看到，熵的离散形式和连续形式相差一个量 $\ln \Delta$ ，该量在极限 $\Delta \rightarrow 0$ 中发散。这反映出非常精确地指定一个连续变量需要大量的比特。对于定义在多个连续变量上的密度，用向量 \mathbf{x} 共同表示，微分熵由下式给出

$$H[\mathbf{x}] = - \int p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x}. \quad (2.92)$$

2.5.4 最大熵

我们看到，对于离散分布，最大熵配置对应于变量所有可能状态上概率的均匀分布。现在让我们考虑连续变量的相应结果。如果要明确地定义这个最大值，就需要约束 $p(x)$ 的一阶矩和二阶矩，并保持归一化约束。因此，我们在以下三个约束条件下最大化微分熵：

$$\int_{-\infty}^{\infty} p(x) dx = 1 \quad (2.93)$$

$$\int_{-\infty}^{\infty} xp(x) dx = \mu \quad (2.94)$$

$$\int_{-\infty}^{\infty} (x - \mu)^2 p(x) dx = \sigma^2. \quad (2.95)$$

附录C

可以使用拉格朗日乘数法进行约束最大化，这样我们就相对于 $p(x)$ 最大化以下泛函：

$$\begin{aligned} & - \int_{-\infty}^{\infty} p(x) \ln p(x) dx + \lambda_1 \left(\int_{-\infty}^{\infty} p(x) dx - 1 \right) \\ & + \lambda_2 \left(\int_{-\infty}^{\infty} xp(x) dx - \mu \right) + \lambda_3 \left(\int_{-\infty}^{\infty} (x - \mu)^2 p(x) dx - \sigma^2 \right). \end{aligned} \quad (2.96)$$

附录B

使用变分法，我们将这个泛函的导数设为零，得到

$$p(x) = \exp \left\{ -1 + \lambda_1 + \lambda_2 x + \lambda_3 (x - \mu)^2 \right\}. \quad (2.97)$$

练习2.24

通过将这个结果回代到三个约束方程中可以找到拉格朗日乘数，最终得到结果：

$$p(x) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{(x - \mu)^2}{2\sigma^2} \right\}, \quad (2.98)$$

因此，使微分熵最大化的分布是高斯分布。注意，在最大化熵时我们没有将分布约束为非负。然而，由于得到的分布确实是非负的，我们事后发现这样的约束是不必要的。

练习2.25

如果我们评估高斯分布的微分熵，我们会得到

$$H[x] = \frac{1}{2} \{ 1 + \ln(2\pi\sigma^2) \}. \quad (2.99)$$

因此，我们再次看到，随着分布变得更宽，即随着 σ^2 增加，熵会增加。这个结果还表明，与离散熵不同，微分熵可以为负，因为对于 $\sigma^2 < 1/(2\pi e)$ ，(2.99) 中的 $H(x) < 0$ 。

2.5.5 库尔贝克 - 莱布勒散度

在本节中，到目前为止，我们已经介绍了信息论中的一些概念，包括熵的关键概念。现在我们开始将这些概念与机器学习联系起来。考虑某个未知分布 $p(\mathbf{x})$ ，假设我们使用近似分布 $q(\mathbf{x})$ 对其进行建模。如果我们使用 $q(\mathbf{x})$ 来构建一种编码方案，以便将 \mathbf{x} 的值传输给接收者，那么由于使用 $q(\mathbf{x})$ 而不是真实分布 $p(\mathbf{x})$ ，(假设我们选择了一种高效的编码方案) 指定 \mathbf{x} 的值所需的平均额外信息量（以奈特为单位）由下式给出

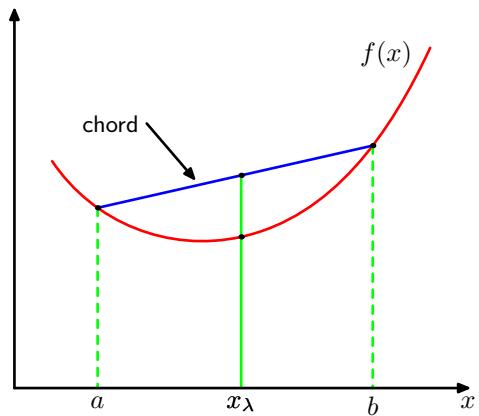
$$\begin{aligned} KL(p\|q) &= - \int p(\mathbf{x}) \ln q(\mathbf{x}) d\mathbf{x} - \left(- \int p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x} \right) \\ &= - \int p(\mathbf{x}) \ln \left\{ \frac{q(\mathbf{x})}{p(\mathbf{x})} \right\} d\mathbf{x}. \end{aligned} \quad (2.100)$$

这被称为分布 $p(\mathbf{x})$ 和 $q(\mathbf{x})$ 之间的相对熵或库尔贝克-莱布勒散度，或KL散度（库尔贝克和莱布勒，1951年）。注意，它不是一个对称量，也就是说 $KL(p\|q) \neq KL(q\|p)$ 。

我们现在证明，当且仅当 $p(\mathbf{x}) = q(\mathbf{x})$ 时，库尔贝克-莱布勒散度满足 $KL(p\|q) \geq 0$ 且取等号。为此，我们首先引入凸函数的概念。如果一个函数 $f(x)$ 具有每条弦都位于函数之上或与函数重合的性质，如图2.15所示，那么该函数就被称为凸函数。

在从 $x = a$ 到 $x = b$ 的区间内， x 的任何值都可以写成 $\lambda a + (1 - \lambda)b$ 的形式，其中 $0 \leq \lambda \leq 1$ 。弦上的对应点由 $\lambda f(a) + (1 - \lambda)f(b)$ 给出，函数的对应值为 $f(\lambda a + (1 - \lambda)b)$ 。

图2.15 一个凸函数 $f(x)$ 是指其每一条弦（蓝色所示）都位于函数（红色所示）之上或与函数重合。



那么凸性意味着

$$f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b). \quad (2.101)$$

练习2.32

这等同于要求函数的二阶导数处处为正。凸函数的例子有 $x \ln x$ （对于 $x > 0$ ）和 x^2 。如果仅当 $\lambda = 0$ 和 $\lambda = 1$ 时等式成立，则该函数称为严格凸函数。如果一个函数具有相反的性质，即每一条弦都位于函数之下或与函数重合，那么它被称为凹函数，对于严格凹函数有相应的定义。如果函数 $f(x)$ 是凸函数，那么 $-f(x)$ 将是凹函数。

练习2.33

使用归纳证明技术，我们可以从(2.101)证明一个凸函数 $f(x)$ 满足

$$f\left(\sum_{i=1}^M \lambda_i x_i\right) \leq \sum_{i=1}^M \lambda_i f(x_i) \quad (2.102)$$

其中 $\lambda_i \geq 0$ 和 $\sum_i \lambda_i = 1$ ，对于任意一组点 $\{x_i\}$ 。结果 (2.102) 被称为詹森不等式。如果我们将 λ_i 解释为离散变量 x 取值为 $\{x_i\}$ 时的概率分布，那么 (2.102) 可以写成

$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)] \quad (2.103)$$

其中 $\mathbb{E}[\cdot]$ 表示期望。对于连续变量，詹森不等式具有以下形式

$$f\left(\int \mathbf{x} p(\mathbf{x}) d\mathbf{x}\right) \leq \int f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x}. \quad (2.104)$$

我们可以将形式为 (2.104) 的詹森不等式应用于库尔贝克 - 莱布勒散度 (2.100)，得到

$$\text{KL}(p\|q) = - \int p(\mathbf{x}) \ln \left\{ \frac{q(\mathbf{x})}{p(\mathbf{x})} \right\} d\mathbf{x} \geq - \ln \int q(\mathbf{x}) d\mathbf{x} = 0 \quad (2.105)$$

我们使用 $-\ln x$ 是一个凸函数，以及归一化条件 $\int q(\mathbf{x})d\mathbf{x} = 1$ 。事实上， $-\ln x$ 是一个严格凸函数，所以当且仅当对于所有 \mathbf{x} ， $q(\mathbf{x}) = p(\mathbf{x})$ 时等式才成立。因此，我们可以将库尔贝克-莱布勒散度解释为两个分布 $p(\mathbf{x})$ 和 $q(\mathbf{x})$ 之间差异的一种度量。

我们发现数据压缩与密度估计（即对未知概率分布进行建模的问题）之间存在密切关系，因为当我们知道真实分布时，就能实现最有效的压缩。如果我们使用的分布与真实分布不同，那么编码效率必然较低，并且平均而言，必须传输的额外信息（至少）等于这两个分布之间的库尔贝克 - 莱布勒散度。

假设数据是从我们希望建模的未知分布 $p(\mathbf{x})$ 中生成的。我们可以尝试使用由一组可调参数 θ 控制的某个参数分布 $q(\mathbf{x} | \theta)$ 来近似这个分布。确定 θ 的一种方法是相对于 θ 最小化 $p(\mathbf{x})$ 和 $q(\mathbf{x} | \theta)$ 之间的库尔贝克-莱布勒散度。我们不能直接这样做，因为我们不知道 $p(\mathbf{x})$ 。然而，假设我们已经观察到从 $p(\mathbf{x})$ 中抽取的有限训练点集 \mathbf{x}_n ，用于 $n = 1, \dots, N$ 。然后，使用(2.40)，相对于 $p(\mathbf{x})$ 的期望可以通过对这些点的有限和来近似，以便

$$\text{KL}(p \| q) \simeq \frac{1}{N} \sum_{n=1}^N \left\{ -\ln q(\mathbf{x}_n | \theta) + \ln p(\mathbf{x}_n) \right\}. \quad (2.106)$$

(2.106)右侧的第二项与 θ 无关，而第一项是在分布 $q(\mathbf{x} | \theta)$ 下使用训练集评估得到的关于 θ 的负对数似然函数。因此，我们看到最小化这个库尔贝克-莱布勒散度等同于最大化对数似然函数。

练习2.34

2.5.6 条件熵

现在考虑两组变量 \mathbf{x} 和 \mathbf{y} 之间的联合分布，由 $p(\mathbf{x}, \mathbf{y})$ 给出，我们从中抽取 \mathbf{x} 和 \mathbf{y} 的值对。如果 \mathbf{x} 的值已经已知，那么指定 \mathbf{y} 相应值所需的额外信息由 $-\ln p(\mathbf{y} | \mathbf{x})$ 给出。因此，指定 \mathbf{y} 所需的平均额外信息可以写成

$$H[\mathbf{y} | \mathbf{x}] = - \iint p(\mathbf{y}, \mathbf{x}) \ln p(\mathbf{y} | \mathbf{x}) d\mathbf{y} d\mathbf{x}, \quad (2.107)$$

练习2.35

这被称为给定 \mathbf{x} 时 \mathbf{y} 的条件熵。利用乘积法则很容易看出，条件熵满足以下关系：

$$H[\mathbf{x}, \mathbf{y}] = H[\mathbf{y} | \mathbf{x}] + H[\mathbf{x}] \quad (2.108)$$

其中 $H[\mathbf{x}, \mathbf{y}]$ 是 $p(\mathbf{x}, \mathbf{y})$ 的微分熵， $H[\mathbf{x}]$ 是边际分布 $p(\mathbf{x})$ 的微分熵。因此，描述 \mathbf{x} 和 \mathbf{y} 所需的信息由单独描述 \mathbf{x} 所需的信息加上在已知 \mathbf{x} 的情况下指定 \mathbf{y} 所需的额外信息之和给出。

2.5.7 互信息

当两个变量 x 和 y 相互独立时，它们的联合分布将分解为各自边缘分布 $p(x, y) = p(x)p(y)$ 的乘积。如果变量不独立，我们可以通过考虑联合分布与边缘分布乘积之间的库尔贝克-莱布勒散度，来了解它们是否“接近”独立，具体如下

$$\begin{aligned} I[x, y] &\equiv \text{KL}(p(x, y) \| p(x)p(y)) \\ &= - \iint p(x, y) \ln \left(\frac{p(x)p(y)}{p(x, y)} \right) dx dy, \end{aligned} \quad (2.109)$$

这被称为变量 x 和 y 之间的互信息。根据库尔贝克-莱布勒散度的性质，我们可以看到，当且仅当 x 和 y 相互独立时， $I[x, y] \geq 0$ 取等号。利用概率的和规则与积规则，我们可以看到互信息通过条件熵与之相关

练习2.38

$$I[x, y] = H[x] - H[x|y] = H[y] - H[y|x]. \quad (2.110)$$

因此，互信息表示由于被告知 y 的值（反之亦然）而导致的关于 x 的不确定性的减少。从贝叶斯的角度来看，我们可以将 $p(x)$ 视为 x 的先验分布，而将 $p(x | y)$ 视为在我们观察到新数据 y 之后的后验分布。因此，互信息表示由于新观察 y 而导致的关于 x 的不确定性的减少。

2.6. 贝叶斯概率

当我们考虑图2.2中的弯曲硬币时，我们根据随机、可重复事件的频率引入了概率的概念，例如硬币凹面朝上的概率。我们将其称为概率的经典或频率主义解释。我们还介绍了更一般的贝叶斯观点，其中概率提供了不确定性的量化。在这种情况下，我们的不确定性在于硬币的凹面是正面还是反面。

使用概率来表示不确定性并非临时选择，而是如果我们要在进行合理且连贯的推理时尊重常识就不可避免的。例如，考克斯（1946年）表明，如果使用数值来表示信念程度，那么一组简单的编码此类信念常识属性的公理将唯一地导致一组用于操纵信念程度的规则，这些规则等同于概率的加法和乘法规则。因此，自然地将这些量称为（贝叶斯）概率。

对于我们所假设的那枚弯曲硬币，在没有更多信息的情况下，硬币凹面为正面的概率是0.5。现在假设我们被告知了这枚硬币几次抛掷的结果。直观地看，这似乎应该能为我们提供一些关于凹面是否为正面的信息。例如，假设我们看到出现反面的抛掷次数比出现正面的次数多得多。

练习2.40

鉴于硬币更有可能凹面朝上落地，这提供了证据表明凹面更有可能是反面。事实上，这种直觉是正确的，而且，我们可以使用概率规则对此进行量化。贝叶斯定理现在有了新的意义，因为它允许我们通过纳入抛硬币提供的数据，将凹面为正面的先验概率转换为后验概率。此外，这个过程是迭代的，这意味着后验概率成为纳入更多抛硬币数据的先验概率。

第3.1.2节

贝叶斯观点的一个方面是，先验知识的纳入是自然产生的。例如，假设一枚外观正常的硬币被抛了三次，每次都正面朝上。正面朝上概率的最大似然估计会给出1，这意味着所有未来的抛掷都将正面朝上！相比之下，采用任何合理先验的贝叶斯方法会得出不那么极端的结论。

第1.2节

2.6.1 模型参数

贝叶斯视角为机器学习的多个方面提供了有价值的见解，我们可以用正弦曲线回归示例来说明这些见解。在这里，我们用 \mathcal{D} 表示训练数据集。我们已经在线性回归的背景下看到，可以使用最大似然法来选择参数，其中 \mathbf{w} 被设置为使似然函数 $p(\mathcal{D} | \mathbf{w})$ 最大化的值。这相当于选择使观测数据集的概率最大化的 \mathbf{w} 值。在机器学习文献中，似然函数的负对数被称为误差函数。由于负对数是单调递减函数，最大化似然性等同于最小化误差。这导致了参数值的特定选择，记为 \mathbf{w}_{ML} ，然后用于对新数据进行预测。

我们已经看到，训练数据集的不同选择，例如包含不同数量的数据点，会为 \mathbf{w}_{ML} 带来不同的解决方案。从贝叶斯的角度来看，我们也可以使用概率论的机制来描述模型参数中的这种不确定性。在观察数据之前，我们可以以先验概率分布 $p(\mathbf{w})$ 的形式捕捉我们对 \mathbf{w} 的假设。观察到的数据 \mathcal{D} 的影响通过似然函数 $p(\mathcal{D} | \mathbf{w})$ 来表达，现在贝叶斯定理采用以下形式

$$p(\mathbf{w} | \mathcal{D}) = \frac{p(\mathcal{D} | \mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}, \quad (2.111)$$

这使我们能够在以后验概率 $p(\mathbf{w} | \mathcal{D})$ 的形式观察到 \mathcal{D} 之后评估 \mathbf{w} 中的不确定性。

需要强调的是，当将数量 $p(\mathcal{D} | \mathbf{w})$ 视为参数向量 \mathbf{w} 的函数时，它被称为似然函数，它表示对于 \mathbf{w} 的不同值，观测数据集出现的可能性有多大。请注意，似然 $p(\mathcal{D} | \mathbf{w})$ 不是 \mathbf{w} 上的概率分布，并且它关于 \mathbf{w} 的积分（不一定）等于1。

根据似然的这个定义，我们可以用文字表述贝叶斯定理：

$$\text{posterior} \propto \text{likelihood} \times \text{prior} \quad (2.112)$$

其中所有这些量都被视为 \mathbf{w} 的函数。 (2.111) 中的分母是归一化常数，它确保左侧的后验分布是有效的概率密度且积分值为1。实际上，通过对 (2.111) 两边关于 \mathbf{w} 进行积分，我们可以根据先验分布和似然函数来表示贝叶斯定理中的分母：

$$p(\mathcal{D}) = \int p(\mathcal{D}|\mathbf{w})p(\mathbf{w}) d\mathbf{w}. \quad (2.113)$$

在贝叶斯和频率主义范式中，似然函数 $p(\mathcal{D} | \mathbf{w})$ 都起着核心作用。然而，在这两种方法中使用它的方式从根本上说是不同的。在频率主义框架下， \mathbf{w} 被视为一个固定参数，其值由某种形式的“估计器”确定，并且（至少在概念上）通过考虑可能数据集 \mathcal{D} 的分布来确定该估计的误差范围。相比之下，从贝叶斯观点来看，只有一个数据集 \mathcal{D} （即实际观察到的那个数据集），并且参数的不确定性通过 \mathbf{w} 上的概率分布来表示。

2.6.2 正则化

1.2.5节

我们可以利用这种贝叶斯观点来深入了解在正弦曲线回归示例中用于减少过拟合的正则化技术。我们不是通过相对于 \mathbf{w} 最大化似然函数来选择模型参数，而是可以最大化后验概率 (2.111) 。这种技术称为最大后验估计，或简称为MAP估计。等效地，我们可以最小化后验概率的负对数。对 (2.111) 两边取负对数，我们得到

$$-\ln p(\mathbf{w}|\mathcal{D}) = -\ln p(\mathcal{D}|\mathbf{w}) - \ln p(\mathbf{w}) + \ln p(\mathcal{D}). \quad (2.114)$$

(2.114) 式右侧的第一项是通常的对数似然。第三项可以省略，因为它不依赖于 \mathbf{w} 。第二项是 \mathbf{w} 的函数形式，它被加到对数似然上，我们可以将其视为一种正则化形式。为了更明确这一点，假设我们选择先验分布 $p(\mathbf{w})$ 为 \mathbf{w} 每个元素的独立零均值高斯分布的乘积，使得每个分布具有相同的方差 s^2 ，于是

$$p(\mathbf{w}|s) = \prod_{i=0}^M \mathcal{N}(w_i|0, s^2) = \prod_{i=0}^M \left(\frac{1}{2\pi s^2} \right)^{1/2} \exp \left\{ -\frac{w_i^2}{2s^2} \right\}. \quad (2.115)$$

代入 (2.114) 式，我们得到

$$-\ln p(\mathbf{w}|\mathcal{D}) = -\ln p(\mathcal{D}|\mathbf{w}) + \frac{1}{2s^2} \sum_{i=0}^M w_i^2 + \text{const.} \quad (2.116)$$

如果我们考虑对数似然由 (2.66) 给出的线性回归模型的特殊情况，那么我们发现最大化后验分布等同于最小化函数

练习2.41

$$E(\mathbf{w}) = \frac{1}{2\sigma^2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{1}{2s^2} \mathbf{w}^\top \mathbf{w}. \quad (2.117)$$

我们看到，这采用了前面以(1.4)形式遇到的正则化平方和误差函数的形式。

2.6.3 贝叶斯机器学习

贝叶斯观点使我们能够推动正则化的使用，并推导出正则化项的特定形式。然而，仅使用贝叶斯定理并不构成对机器学习的真正贝叶斯处理，因为它仍然在为 \mathbf{w} 寻找单个解，因此没有考虑 \mathbf{w} 值中的不确定性。假设我们有一个训练数据集 \mathcal{D} ，我们的目标是在给定新输入值 x 的情况下预测某个目标变量 t 。因此，我们感兴趣的是给定 x 和 \mathcal{D} 时 t 的分布。根据概率的和规则与积规则，我们有

$$p(t|x, \mathcal{D}) = \int p(t|x, \mathbf{w})p(\mathbf{w}|\mathcal{D}) d\mathbf{w}. \quad (2.118)$$

我们看到，预测是通过对 \mathbf{w} 的所有可能值进行加权平均 $p(t|x, \mathbf{w})$ 得到的，其中加权函数由后验概率分布 $p(\mathbf{w}|\mathcal{D})$ 给出。区分贝叶斯方法的关键区别在于对参数空间的这种积分。相比之下，传统的频率主义方法使用通过优化损失函数（如正则化平方和）获得的参数点估计。

这种对机器学习的全贝叶斯处理提供了一些强大的见解。例如，在多项式回归的背景下早些时候遇到的过拟合问题，是使用最大似然法产生的一种病态情况的例子，而当我们使用贝叶斯方法对参数进行边缘化时，这种情况就不会出现。同样，我们可能有多个潜在模型可用于解决给定问题，比如回归示例中不同阶数的多项式。最大似然法只是简单地选择给出数据最高概率的模型，但这有利于越来越复杂的模型，从而导致过拟合。全贝叶斯处理涉及对所有可能模型进行平均，每个模型的贡献由其后验概率加权。此外，这种概率通常对于中等复杂度的模型最高。非常简单的模型（如低阶多项式）概率较低，因为它们无法很好地拟合数据，而非常复杂的模型（如高阶多项式）概率也较低，因为对参数的贝叶斯积分会自动且巧妙地惩罚复杂度。有关应用于机器学习（包括神经网络）的贝叶斯方法的全面概述，请参阅毕晓普（2006年）。

不幸的是，贝叶斯框架存在一个主要缺点，这在 (2.118) 中很明显，它涉及在参数空间上进行积分。现代深度学习模型可能有数百万或数十亿个参数，即使是对这种积分的简单近似通常也是不可行的。

第1.2节

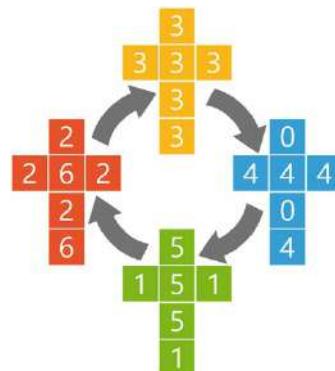
第9.6节

事实上，在计算资源有限且有大量训练数据的情况下，对大型神经网络应用最大似然技术（通常辅以一种或多种正则化形式）往往比将贝叶斯方法应用于小得多的模型要好。

练习

- 2.1 (*) 在癌症筛查示例中，我们使用的癌症先验概率为 $p(C = 1) = 0.01$ 。实际上，癌症的发病率通常要低得多。考虑 $p(C = 1) = 0.001$ 的情况，重新计算检测呈阳性时患癌症的概率 $p(C = 1 | T = 1)$ 。直观地说，这个结果可能会让很多人感到惊讶，因为检测似乎具有很高的准确性，但检测呈阳性时患癌症的概率仍然很低。
- 2.2 (**) 确定性数字满足传递性，所以如果 $x > y$ 且 $y > z$ ，那么就有 $x > z$ 。然而，当我们讨论随机数时，这个性质就不再适用了。图2.16 展示了一组四个立方体骰子，它们按循环顺序排列。证明这四个骰子中的每一个掷出比循环中前一个骰子更高数字的概率都是 $2/3$ 。这样的骰子被称为非传递性骰子，这里展示的具体例子被称为埃夫隆骰子。

图2.16 非传递性立方体骰子的一个例子，其中每个骰子都被“展开”以显示每个面上的数字。骰子按循环顺序排列，使得每个骰子掷出比循环中前一个骰子更高数字的概率都是 $2/3$ 。



- 2.3 (*) 考虑一个由两个独立随机变量之和给出的变量 \mathbf{y} ，其中 $\mathbf{y} = \mathbf{u} + \mathbf{v}$ 且 $\mathbf{u} \sim p_{\mathbf{u}}(\mathbf{u})$ 和 $\mathbf{v} \sim p_{\mathbf{v}}(\mathbf{v})$ 。证明分布 $p_{\mathbf{y}}(\mathbf{y})$ 由下式给出

$$p(\mathbf{y}) = \int p_{\mathbf{u}}(\mathbf{u})p_{\mathbf{v}}(\mathbf{y} - \mathbf{u}) d\mathbf{u}. \quad (2.119)$$

这被称为 $p_{\mathbf{u}}(\mathbf{u})$ 和 $p_{\mathbf{v}}(\mathbf{v})$ 的卷积。

- 2.4 (**) 验证均匀分布(2.33)是否正确归一化，并求出其均值和方差的表达式。

- 2.5 (**) 验证指数分布(2.34)和拉普拉斯分布(2.35)是否正确归一化。

2.6 (*) 利用狄拉克 δ 函数的性质，证明经验密度(2.37)被正确归一化。

2.7 (*) 通过利用经验密度(2.37)，证明(2.39)给出的期望可以由从形式为(2.40)的密度中抽取的有限样本集的和来近似。

2.8 (*) 利用定义(2.44)，证明 $\text{var}[f(x)]$ 满足(2.45)。

2.9 (*) 证明如果两个变量 x 和 y 是独立的，那么它们的协方差为零。

2.10 (*) 假设两个变量 x 和 z 在统计上是独立的。证明它们和的均值和方差满足

$$\mathbb{E}[x+z] = \mathbb{E}[x] + \mathbb{E}[z] \quad (2.120)$$

$$\text{var}[x+z] = \text{var}[x] + \text{var}[z]. \quad (2.121)$$

2.11 (*) 考虑两个具有联合分布 $p(x, y)$ 的变量 x 和 y 。证明以下两个结果：

$$\mathbb{E}[x] = \mathbb{E}_y [\mathbb{E}_x[x|y]] \quad (2.122)$$

$$\text{var}[x] = \mathbb{E}_y [\text{var}_x[x|y]] + \text{var}_y [\mathbb{E}_x[x|y]]. \quad (2.123)$$

这里 $\mathbb{E}_x[x|y]$ 表示在条件分布 $p(x|y)$ 下 x 的期望，条件方差采用类似的表示法。

2.12 (★★★) 在本练习中，我们证明单变量高斯分布的归一化条件(2.51)。为此考虑积分

$$I = \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2\sigma^2}x^2\right) dx \quad (2.124)$$

我们可以通过首先将其平方写成以下形式来计算该积分

$$I^2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2\sigma^2}x^2 - \frac{1}{2\sigma^2}y^2\right) dx dy. \quad (2.125)$$

现在进行从笛卡尔坐标 (x, y) 到极坐标 (r, θ) 的变换，然后代入 $u = r^2$ 。证明通过对 θ 和 u 进行积分，然后对两边取平方根，我们得到

$$I = (2\pi\sigma^2)^{1/2}. \quad (2.126)$$

最后，利用这个结果证明高斯分布 $\mathcal{N}(x | \mu, \sigma^2)$ 是归一化的。

2.13 (**) 通过变量变换，验证由(2.49)给出的单变量高斯分布满足(2.52)。接下来，通过对归一化条件两边关于 $\star\star$ 求导，验证高斯分布满足(2.53)。最后，证明(2.54)成立。

$$\int_{-\infty}^{\infty} \mathcal{N}(x|\mu, \sigma^2) dx = 1 \quad (2.127)$$

关于 σ^2 ，验证高斯分布满足(2.53)。最后，证明(2.54)成立。

2.14 (*) 证明高斯分布(2.49)的众数（即最大值）由 μ 给出。

2.15 (*) 通过将对数似然函数(2.56)关于 μ 和 σ^2 的导数设为零，验证结果(2.57)和(2.58)。

2.16 (**) 使用结果(2.52)和(2.53)，证明

$$\mathbb{E}[x_n x_m] = \mu^2 + I_{nm}\sigma^2 \quad (2.128)$$

其中 x_n 和 x_m 表示从均值为 μ 、方差为 σ^2 的高斯分布中采样的数据点，并且如果 $n = m$ ，则 I_{nm} 满足 $I_{nm} = 1$ ，否则满足 $I_{nm} = 0$ 。因此证明结果(2.59)和(2.60)。

2.17 (**) 使用定义(2.61)，证明结果(2.62)，该结果表明基于真实均值的高斯分布方差估计器的期望由真实方差 σ^2 给出。

2.18 (*) 证明关于 σ^2 最大化(2.66)会得到结果(2.68)。

2.19 (**) 利用变量变换下概率密度的变换性质(2.71)证明，通过进行非线性变量变换 $y = f(x)$ ，其中 $f(x)$ 是单调函数，使得 $0 \leq f'(x) < \infty$ ，任何密度 $p(y)$ 都可以从处处非零的固定密度 $q(x)$ 得到。写出 $f(x)$ 满足的微分方程，并画出说明密度变换的图。

2.20 (*) 计算由(2.78)和(2.79)定义的变换的雅可比矩阵的元素。

2.21 (*) 在2.5节中，我们引入了熵 $h(x)$ 的概念，即观察具有分布 $p(x)$ 的随机变量 x 的值时获得的信息。我们看到，对于独立变量 x 和 y ，当 $p(x, y) = p(x)p(y)$ 时，熵函数是可加的，因此 $h(x, y) = h(x) + h(y)$ 。在本练习中，我们以函数 $h(p)$ 的形式推导 h 和 p 之间的关系。首先证明 $h(p^2) = 2h(p)$ ，然后通过归纳法证明 $h(p^n) = nh(p)$ ，其中 n 是正整数。因此，证明 $h(p^{n/m}) = (n/m)h(p)$ ，其中 m 也是正整数。这意味着 $h(p^x) = xh(p)$ ，其中 x 是正有理数，因此，当它是正实数时，通过连续性也成立。最后，证明这意味着 $h(p)$ 必须采取 $h(p) \propto \ln p$ 的形式。

- 2.22 (*) 使用拉格朗日乘数法证明，对于离散变量，最大化熵 (2.86) 会得到一种分布，其中所有概率 $p(x_i)$ 都相等，并且此时熵的相应值为 $\ln M$ 。
- 2.23 (*) 考虑一个 M 状态的离散随机变量 x ，并使用形式为 (2.102) 的詹森不等式来证明其分布 $p(x)$ 的熵满足 $H[x] \leq \ln M$ 。
- 2.24 (**) 使用变分法证明泛函(2.96)的驻点由(2.97)给出。然后使用约束条件 (2.93)、(2.94)和(2.95)消去拉格朗日乘数，从而证明最大熵解由高斯分布(2.98)给出。
- 2.25 (*) 使用结果(2.94)和(2.95)证明单变量高斯分布(2.98)的熵由(2.99)给出。
- 2.26 (**) 假设 $p(\mathbf{x})$ 是某种固定分布，并且我们希望用高斯分布 $q(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \mu, \Sigma)$ 对其进行近似。通过写出高斯分布 $q(\mathbf{x})$ 的库尔贝克-莱布勒散度 $KL(p \parallel q)$ 的形式，然后求导，证明关于 μ 和 Σ 对 $KL(p \parallel q)$ 进行最小化会得到结果： μ 由 $p(\mathbf{x})$ 下 \mathbf{x} 的期望给出，且 Σ 由协方差给出。
- 2.27 (**) 评估两个高斯分布 $p(x) = \mathcal{N}(x | \mu, \sigma^2)$ 和 $q(x) = \mathcal{N}(x | m, s^2)$ 之间的Kullback-Leibler散度(2.100)。
- 2.28 (**) 散度的 α 族由以下定义

$$D_\alpha(p \parallel q) = \frac{4}{1 - \alpha^2} \left(1 - \int p(x)^{(1+\alpha)/2} q(x)^{(1-\alpha)/2} dx \right) \quad (2.129)$$

其中 $-\infty < \alpha < \infty$ 是一个连续参数。证明库尔贝克 - 莱布勒散度 $KL(p \parallel q)$ 对应于 $\alpha \rightarrow 1$ 。这可以通过写出 $p^\epsilon = \exp(\epsilon \ln p) = 1 + \epsilon \ln p + O(\epsilon^2)$ 然后求 $\epsilon \rightarrow 0$ 来完成。类似地，证明 $KL(q \parallel p)$ 对应于 $\alpha \rightarrow -1$ 。

- 2.29 (**) 考虑具有联合分布 $p(\mathbf{x}, \mathbf{y})$ 的两个变量 \mathbf{x} 和 \mathbf{y} 。证明这对变量的微分熵满足

$$H[\mathbf{x}, \mathbf{y}] \leq H[\mathbf{x}] + H[\mathbf{y}] \quad (2.130)$$

当且仅当 \mathbf{x} 和 \mathbf{y} 统计独立时取等号。

- 2.30 (*) 考虑一个具有分布 $p(\mathbf{x})$ 和相应熵 $H[\mathbf{x}]$ 的连续变量向量 \mathbf{x} 。假设我们对 \mathbf{x} 进行非奇异线性变换以获得一个新变量 $\mathbf{y} = \mathbf{Ax}$ 。证明相应的熵由 $H[\mathbf{y}] = H[\mathbf{x}] + \ln \det \mathbf{A}$ 给出，其中 $\det \mathbf{A}$ 表示 \mathbf{A} 的行列式。

- 2.31 (**) 假设两个离散随机变量 x 和 y 之间的条件熵 $H[y | x]$ 为零。证明：对于 x 的所有满足 $p(x) > 0$ 的值，变量 y 必定是 x 的函数。换句话说，对于每个 x ，只有一个 y 值使得 $p(y | x) \neq 0$ 。

2.32 (*) 严格凸函数被定义为这样一种函数，即其每一条弦都位于函数上方。证明这等同于函数的二阶导数为正的条件。

2.33 (**) 用归纳法证明，凸函数的不等式(2.101)蕴含结果(2.102)。

2.34 (*) 证明，在相差一个加性常数的情况下，经验分布(2.37)与模型分布 $q(x | \theta)$ 之间的Kullback-Leibler散度(2.100)等于负对数似然函数。

2.35 (*) 利用定义(2.107)以及概率的乘积法则，证明结果(2.108)。

2.36 (***) 考虑两个二元变量 x 和 y ，其联合分布如下所示

		y
	0	0 1/3
x	1	0 1/3

计算以下量：

$$\begin{array}{ll} \text{(a)} H[x] \\ \text{(b)} H[y] \end{array}$$

$$\text{(c)} H[y | x] \quad \begin{array}{l} \text{(e)} H[x, y] \\ \text{(f)} I[x, y]. \end{array}$$

绘制一个维恩图来展示这些不同数量之间的关系。

2.37 (*) 通过对 $f(x) = \ln x$ 应用詹森不等式(2.102)，证明一组实数的算术平均值从不小于它们的几何平均值。

2.38 (*) 使用概率的和规则与积规则，证明互信息 $I(\mathbf{x}, \mathbf{y})$ 满足关系(2.110)。

2.39 (**) 假设两个变量 z_1 和 z_2 是独立的，使得 $p(z_1, z_2) = p(z_1)p(z_2)$ 。证明这些变量之间的协方差矩阵是对角矩阵。这表明独立性是两个变量不相关的充分条件。现在考虑两个变量 y_1 和 y_2 ，其中 y_1 关于0对称分布且 $y_2 = y_1^2$ 。写出条件分布 $p(y_2 | y_1)$ 并观察到它依赖于 y_1 ，从而表明这两个变量不是独立的。现在证明这两个变量之间的协方差矩阵再次是对角矩阵。为此，使用关系式 $p(y_1, y_2) = p(y_1)p(y_2 | y_1)$ 来证明非对角项为零。这个反例表明零相关性不是独立性的充分条件。

2.40 (*) 考虑图2.2中的弯曲硬币。假设凸面为正面的先验概率是0.1。现在假设硬币被抛掷10次，我们被告知其中8次正面朝上，2次反面朝上。使用贝叶斯定理来评估凹面为正面的后验概率。计算下一次抛掷正面朝上的概率。

2.41 (★) 通过将(2.115)代入(2.114)并利用线性回归模型对数似然的结果(2.66), 推导出正则化误差函数的结果(2.117)。

深度学习



3

Standard Distributions

在本章中，我们将讨论概率分布的一些具体示例及其性质。这些分布不仅本身具有研究价值，还可以构成更复杂模型的基础，并将在整本书中广泛使用。

本章所讨论的分布的一个作用是，在给定有限观测集 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 的情况下，对随机变量 \mathbf{x} 的概率分布 $p(\mathbf{x})$ 进行建模。这个问题被称为密度估计。需要强调的是，密度估计问题从根本上说是不适当的，因为有无限多个概率分布可能产生观测到的有限数据集。实际上，在每个数据点 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 处非零的任何分布 $p(\mathbf{x})$ 都是一个潜在的候选。选择合适分布的问题与模型选择问题相关，这在多项式曲线拟合的背景下已经遇到过，并且是机器学习中的一个核心问题。

第1.2节

我们先考虑离散变量的分布，然后再探讨连续变量的高斯分布。这些是参数分布的具体例子，之所以这样称呼是因为它们由相对较少的可调参数控制，比如高斯分布的均值和方差。要将此类模型应用于密度估计问题，我们需要一个程序，用于在给定观测数据集的情况下确定参数的合适值，而我们的主要关注点将是最大化似然函数。在本章中，我们将假设数据观测是独立同分布的（i.i.d.），而在后续章节中，我们将探讨涉及结构化数据的更复杂场景，在这种情况下这个假设不再成立。

参数方法的一个局限性在于它假设分布具有特定的函数形式，而这对于特定应用可能并不合适。另一种方法是非参数密度估计方法，其中分布的形式通常取决于数据集的大小。此类模型仍包含参数，但这些参数控制的是模型复杂度而非分布形式。本章最后简要介绍三种分别基于直方图、最近邻和核的非参数方法。像这样的非参数技术的一个主要局限性是它们需要存储所有训练数据。换句话说，参数数量会随着数据集大小的增加而增加，因此对于大数据集，该方法会变得效率极低。深度学习通过考虑基于具有大量但固定数量参数的神经网络的灵活分布，将参数模型的效率与非参数方法的通用性结合起来。

3.1. 离散变量

我们首先考虑离散变量的简单分布，从二元变量开始，然后再转向多状态变量。

3.1.1 伯努利分布

考虑一个单一的二元随机变量 $x \in \{0, 1\}$ 。例如， x 可能描述抛硬币的结果，其中 $x = 1$ 代表“正面”， $x = 0$ 代表“反面”。如果这是一枚有损坏的硬币，比如图2.2中所示的那枚，正面朝上的概率不一定与反面朝上的概率相同。 $x = 1$ 的概率将用参数 μ 表示，因此

$$p(x = 1 | \mu) = \mu \tag{3.1}$$

其中 $0 \leq \mu \leq 1$ ，由此可得 $p(x = 0 | \mu) = 1 - \mu$ 。因此， x 上的概率分布可以写成以下形式

$$\text{Bern}(x | \mu) = \mu^x (1 - \mu)^{1-x}, \tag{3.2}$$

练习3.1

这就是所谓的伯努利分布。

很容易验证该分布是归一化的，并且其均值和方差由下式给出

$$\mathbb{E}[x] = \mu \quad (3.3)$$

$$\text{var}[x] = \mu(1 - \mu). \quad (3.4)$$

现在假设我们有一个关于 x 观测值的数据集 $\mathcal{D} = \{x_1, \dots, x_N\}$ 。我们可以构建似然函数，它是 μ 的函数，假设观测值是独立地从 $p(x | \mu)$ 中抽取的，那么

$$p(\mathcal{D}|\mu) = \prod_{n=1}^N p(x_n|\mu) = \prod_{n=1}^N \mu^{x_n}(1-\mu)^{1-x_n}. \quad (3.5)$$

我们可以通过最大化似然函数，或者等效地通过最大化似然函数的对数来估计 μ 的值，因为对数是单调函数。伯努利分布的对数似然函数由下式给出

$$\ln p(\mathcal{D}|\mu) = \sum_{n=1}^N \ln p(x_n|\mu) = \sum_{n=1}^N \{x_n \ln \mu + (1-x_n) \ln(1-\mu)\}. \quad (3.6)$$

3.4节

此时，请注意对数似然函数仅通过 N 观测值 x_n 的总和 $\sum_n x_n$ 依赖于它们。这个总和为该分布下的数据提供了一个充分统计量的示例。如果我们将 $\ln p(\mathcal{D} | \mu)$ 关于 μ 的导数设为零，我们就得到了最大似然估计量：

$$\mu_{\text{ML}} = \frac{1}{N} \sum_{n=1}^N x_n, \quad (3.7)$$

它也被称为样本均值。用 m 表示此数据集中 $x = 1$ (正面) 的观测次数，我们可以将 (3.7) 写成如下形式

$$\mu_{\text{ML}} = \frac{m}{N} \quad (3.8)$$

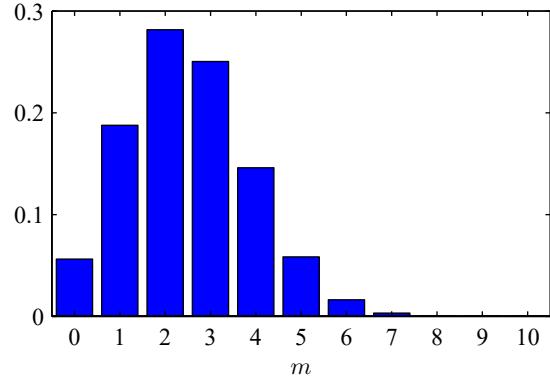
这样，在这个最大似然框架下，出现正面的概率由数据集中正面观测的比例给出。

3.1.2 二项分布

已知数据集大小为 N ，我们还可以计算出关于 $x = 1$ 的观测次数 m 的二元变量 x 的分布。这称为二项分布，从(3.5)式我们可以看出它与 $\mu^m(1-\mu)^{N-m}$ 成正比。为了得到归一化系数，注意在 N 次抛硬币中，我们必须累加得到 m 次正面的所有可能方式，这样二项分布可以写成

$$\text{Bin}(m|N, \mu) = \binom{N}{m} \mu^m (1-\mu)^{N-m} \quad (3.9)$$

图3.1 二项分布(3.9)作为 m 的函数，针对 $N = 10$ 和 $\mu = 0.25$ 的直方图。



其中

$$\binom{N}{m} \equiv \frac{N!}{(N-m)!m!} \quad (3.10)$$

练习3.3

是从总共 N 个相同对象中无放回地选择 m 个对象的方法数。图 3.1 显示了针对 $N = 10$ 和 $\mu = 0.25$ 的二项分布曲线。

练习2.10

二项分布的均值和方差可以通过使用以下结果来找到：对于独立事件，和的均值是均值之和，和的方差是方差之和。因为 $m = x_1 + \dots + x_N$ ，并且因为对于每个观测值，均值和方差分别由(3.3)和(3.4)给出，所以我们有

$$\mathbb{E}[m] \equiv \sum_{m=0}^N m \text{Bin}(m|N, \mu) = N\mu \quad (3.11)$$

$$\text{var}[m] \equiv \sum_{m=0}^N (m - \mathbb{E}[m])^2 \text{Bin}(m|N, \mu) = N\mu(1-\mu). \quad (3.12)$$

练习3.4

这些结果也可以直接用微积分证明。

3.1.3 多项分布

二元变量可用于描述能取两个可能值之一的量。然而，我们经常遇到能取 K 个可能的互斥状态之一的离散变量。虽然有多种表示此类变量的方法，但我们很快会看到，一种特别方便的表示是 K 选 1 方案，有时称为“独热编码”，其中变量由一个 K 维向量 \mathbf{x} 表示，该向量中的一个元素 x_k 等于 1，其余所有元素等于 0。例如，如果我们有一个能取 $K = 6$ 个状态的变量，并且该变量的一个特定观测恰好对应于 $x_3 = 1$ 的状态，那么 \mathbf{x} 将表示为

$$\mathbf{x} = (0, 0, 1, 0, 0, 0)^T. \quad (3.13)$$

注意，这样的向量满足 $\sum_{k=1}^K x_k = 1$ 。如果我们用参数 μ_k 表示 $x_k = 1$ 的概率，那么 \mathbf{x} 的分布由下式给出

$$p(\mathbf{x}|\boldsymbol{\mu}) = \prod_{k=1}^K \mu_k^{x_k} \quad (3.14)$$

其中 $\boldsymbol{\mu} = (\mu_1, \dots, \mu_K)^T$ ，并且参数 μ_k 被约束为满足 $\mu_k \geq 0$ 和 $\sum_k \mu_k = 1$ ，因为它们表示概率。分布(3.14)可以被视为伯努利分布到多于两个结果的推广。很容易看出该分布是归一化的：

$$\sum_{\mathbf{x}} p(\mathbf{x}|\boldsymbol{\mu}) = \sum_{k=1}^K \mu_k = 1 \quad (3.15)$$

并且

$$\mathbb{E}[\mathbf{x}|\boldsymbol{\mu}] = \sum_{\mathbf{x}} p(\mathbf{x}|\boldsymbol{\mu})\mathbf{x} = \boldsymbol{\mu}. \quad (3.16)$$

现在考虑一个包含 N 个独立观测值 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 的数据集 \mathcal{D} 。相应的似然函数具有以下形式

$$p(\mathcal{D}|\boldsymbol{\mu}) = \prod_{n=1}^N \prod_{k=1}^K \mu_k^{x_{nk}} = \prod_{k=1}^K \mu_k^{\left(\sum_n x_{nk}\right)} = \prod_{k=1}^K \mu_k^{m_k} \quad (3.17)$$

在其中我们可以看到，似然函数仅通过 K 个量依赖于 N 个数据点：

$$m_k = \sum_{n=1}^N x_{nk}, \quad (3.18)$$

它表示 $x_k = 1$ 的观测次数。这些被称为该分布的充分统计量。请注意，变量 m_k 受到约束

第3.4节

$$\sum_{k=1}^K m_k = N. \quad (3.19)$$

为了找到 $\boldsymbol{\mu}$ 的最大似然解，我们需要在考虑约束条件 (3.15) 的情况下，关于 μ_k 最大化 $\ln p(\mathcal{D} | \boldsymbol{\mu})$ ，即 μ_k 的和必须为1。这可以通过使用拉格朗日乘数 λ 并最大化

附录C

$$\sum_{k=1}^K m_k \ln \mu_k + \lambda \left(\sum_{k=1}^K \mu_k - 1 \right). \quad (3.20)$$

将(3.20)关于 μ_k 的导数设为零，我们得到

$$\mu_k = -m_k/\lambda. \quad (3.21)$$

我们可以通过将(3.21)代入约束条件 $\sum_k \mu_k = 1$ 以得到 $\lambda = -N$ 来求解拉格朗日乘数 λ 。因此，我们得到 μ_k 的最大似然解形式为

$$\mu_k^{\text{ML}} = \frac{m_k}{N}, \quad (3.22)$$

它是满足 $x_k = 1$ 的 N 观测值的比例。

我们还可以考虑在参数向量 μ 和观测总数 N 的条件下，数量 m_1, \dots, m_K 的联合分布。根据(3.17)，其形式为

$$\text{Mult}(m_1, m_2, \dots, m_K | \mu, N) = \binom{N}{m_1 m_2 \dots m_K} \prod_{k=1}^K \mu_k^{m_k}, \quad (3.23)$$

这被称为多项分布。归一化系数是将 N 个对象分成大小为 m_1, \dots, m_K 的 K 组的方式数，由下式给出

$$\binom{N}{m_1 m_2 \dots m_K} = \frac{N!}{m_1! m_2! \dots m_K!}. \quad (3.24)$$

请注意，二态量既可以表示为二元变量并使用二项分布 (3.9) 进行建模，也可以表示为二选一变量并使用带有 $K = 2$ 的分布 (3.14) 进行建模。

3.2. 多元高斯分布

第2.3节

高斯分布，也称为正态分布，是一种广泛用于连续变量分布的模型。我们已经看到，对于单个变量 x ，高斯分布可以写成如下形式

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\} \quad (3.25)$$

其中 μ 是均值， σ^2 是方差。对于一个 D 维向量 \mathbf{x} ，多元高斯分布具有以下形式

$$\mathcal{N}(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu)\right\} \quad (3.26)$$

其中 μ 是 D 维均值向量， Σ 是 $D \times D$ 协方差矩阵， $\det \Sigma$ 表示 Σ 的行列式。

高斯分布出现在许多不同的情境中，并且可以从各种不同的角度来理解。例如，我们已经看到对于

第2.5节

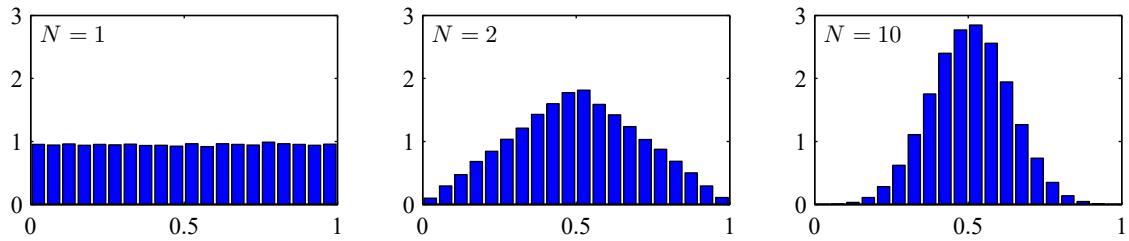


图3.2 针对 N 的不同取值， N 个均匀分布数字均值的直方图。我们观察到，随着 N 增加，分布趋于高斯分布。

练习3.8

对于单个实变量，使熵最大化的分布是高斯分布。此属性也适用于多元高斯分布。

高斯分布出现的另一种情况是当我们考虑多个随机变量的和时。中心极限定理告诉我们，在某些温和条件下，一组随机变量的和（其本身当然也是一个随机变量），随着和中项数的增加，其分布会越来越接近高斯分布（Walker, 1969）。我们可以通过考虑 N 个变量 x_1, \dots, x_N 来说明这一点，每个变量在区间 $[0, 1]$ 上具有均匀分布，然后考虑均值 $(x_1 + \dots + x_N)/N$ 的分布。对于较大的 N ，这种分布趋于高斯分布，如图3.2所示。在实际中，随着 N 的增加，向高斯分布的收敛可能非常迅速。这个结果的一个后果是，二项分布

(3.9)，它是由随机二元变量 x 的 N 次观测之和定义的关于 m 的分布，随着 $N \rightarrow \infty$ （对于 $N = 10$ ，见图3.1）将趋于高斯分布。

高斯分布有许多重要的分析性质，我们将详细考虑其中的几个。因此，本节在技术上比前面的一些章节更为复杂，需要熟悉各种矩阵恒等式。

附录A

3.2.1 高斯分布的几何性质

我们首先考虑高斯分布的几何形式。高斯分布对 \mathbf{x} 的函数依赖是通过二次型实现的

$$\Delta^2 = (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}), \quad (3.27)$$

它出现在指数中。量 Δ 称为从 $\boldsymbol{\mu}$ 到 \mathbf{x} 的马氏距离。当 $\boldsymbol{\Sigma}$ 为单位矩阵时，它简化为欧几里得距离。高斯分布在 \mathbf{x} 空间中该二次型为常数的曲面上是常数。

练习3.11

首先，注意到矩阵 $\boldsymbol{\Sigma}$ 可以不失一般性地取为对称的，因为任何反对称分量都会从指数中消失。现在考虑协方差矩阵的特征向量方程

$$\boldsymbol{\Sigma} \mathbf{u}_i = \lambda_i \mathbf{u}_i \quad (3.28)$$

练习3.12 其中 $i = 1, \dots, D$ 。由于 Σ 是一个实对称矩阵，其特征值将是实数，并且其特征向量可以选择形成一个正交归一集合，因此

$$\mathbf{u}_i^T \mathbf{u}_j = I_{ij} \quad (3.29)$$

其中 I_{ij} 是单位矩阵的 i, j 元素并满足

$$I_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases} \quad (3.30)$$

协方差矩阵 Σ 可以根据其特征向量展开表示为

练习3.13

$$\Sigma = \sum_{i=1}^D \lambda_i \mathbf{u}_i \mathbf{u}_i^T \quad (3.31)$$

类似地，逆协方差矩阵 Σ^{-1} 可以表示为

$$\Sigma^{-1} = \sum_{i=1}^D \frac{1}{\lambda_i} \mathbf{u}_i \mathbf{u}_i^T. \quad (3.32)$$

将(3.32)代入(3.27)，二次型变为

$$\Delta^2 = \sum_{i=1}^D \frac{y_i^2}{\lambda_i} \quad (3.33)$$

其中我们定义了

$$y_i = \mathbf{u}_i^T (\mathbf{x} - \boldsymbol{\mu}). \quad (3.34)$$

我们可以将 $\{y_i\}$ 解释为由正交向量 \mathbf{u}_i 定义的新坐标系，这些向量相对于原始 x_i 坐标进行了平移和旋转。形成向量 $\mathbf{y} = (y_1, \dots, y_D)^T$ ，我们有

$$\mathbf{y} = \mathbf{U}(\mathbf{x} - \boldsymbol{\mu}) \quad (3.35)$$

其中 \mathbf{U} 是一个矩阵，其行由 \mathbf{u}_i^T 给出。由(3.29)可知 \mathbf{U} 是一个正交矩阵，即它满足 $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}$ ，其中 \mathbf{I} 是单位矩阵。

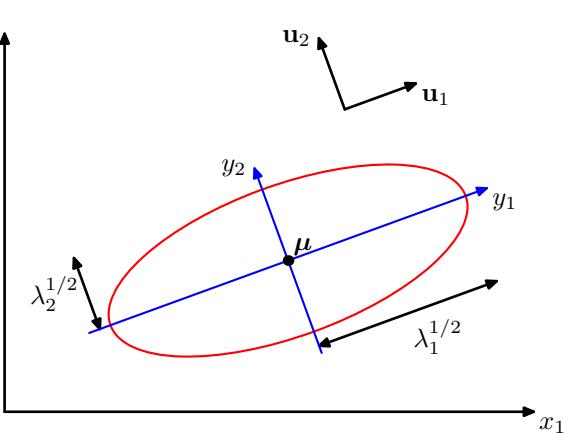
附录A

二次型，进而高斯密度，在使得(3.33)为常数的曲面上是常数。如果所有特征值 λ_i 都是正的，那么这些曲面表示椭球体，其中心在 $\boldsymbol{\mu}$ ，其轴沿 \mathbf{u}_i 方向定向，并且在轴方向上的缩放因子由 $\lambda_i^{1/2}$ 给出，如图3.3所示。

第16章

为了使高斯分布有良好的定义，协方差矩阵的所有特征值 λ_i 必须严格为正，否则该分布不能被正确归一化。其特征值严格为正的矩阵被称为正定矩阵。当我们讨论潜变量模型时，我们会遇到一个或多个特征值为零的高斯分布，在这种情况下，该分布是奇异的，并且局限于较低维度的子空间。

图3.3红色曲线显示了二维空间 $x = (x_1, x_2)$ 中高斯分布的恒定概率密度的椭圆表面，在该表面上密度是其在 $\mathbf{x} = \mu$ 处值的 $\exp(-1/2)$ 。椭圆的轴由协方差矩阵的特征向量 \mathbf{u}_i 定义，对应的特征值为 λ_i 。



如果所有特征值都是非负的，则称协方差矩阵为半正定矩阵。

现在考虑由 y_i 定义的新坐标系中高斯分布的形式。从 \mathbf{x} 坐标系转换到 \mathbf{y} 坐标系时，我们有一个雅可比矩阵 \mathbf{J} ，其元素由下式给出

$$J_{ij} = \frac{\partial x_i}{\partial y_j} = U_{ji} \quad (3.36)$$

U_{ji} 是矩阵 \mathbf{U}^T 的元素。利用矩阵 \mathbf{U} 的正交归一性，我们可以看到雅可比矩阵行列式的平方为

$$|\mathbf{J}|^2 = |\mathbf{U}^T|^2 = |\mathbf{U}^T| |\mathbf{U}| = |\mathbf{U}^T \mathbf{U}| = |\mathbf{I}| = 1 \quad (3.37)$$

因此， $|\mathbf{J}| = 1$ 。此外，协方差矩阵 $|\Sigma|$ 的行列式可以写成其特征值的乘积，因此

$$|\Sigma|^{1/2} = \prod_{j=1}^D \lambda_j^{1/2}. \quad (3.38)$$

因此，在 y_j 坐标系中，高斯分布具有以下形式

$$p(\mathbf{y}) = p(\mathbf{x}) |\mathbf{J}| = \prod_{j=1}^D \frac{1}{(2\pi\lambda_j)^{1/2}} \exp\left\{-\frac{y_j^2}{2\lambda_j}\right\}, \quad (3.39)$$

它是 D 个独立单变量高斯分布的乘积。因此，特征向量定义了一组新的平移和旋转坐标，相对于这些坐标，联合概率分布分解为独立分布的乘积。然后，在 \mathbf{y} 坐标系中分布的积分是

$$\int p(\mathbf{y}) d\mathbf{y} = \prod_{j=1}^D \int_{-\infty}^{\infty} \frac{1}{(2\pi\lambda_j)^{1/2}} \exp\left\{-\frac{y_j^2}{2\lambda_j}\right\} dy_j = 1 \quad (3.40)$$

我们在此处使用了单变量高斯分布归一化的结果(2.51)。这证实了多元高斯分布(3.26)确实是归一化的。

3.2.2 矩

我们现在研究高斯分布的矩，从而对参数 μ 和 Σ 给出一种解释。高斯分布下 \mathbf{x} 的期望由下式给出

$$\begin{aligned}\mathbb{E}[\mathbf{x}] &= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\} \mathbf{x} d\mathbf{x} \\ &= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2}\mathbf{z}^T \Sigma^{-1} \mathbf{z} \right\} (\mathbf{z} + \boldsymbol{\mu}) d\mathbf{z}\end{aligned}\quad (3.41)$$

我们在此处使用 $\mathbf{z} = \mathbf{x} - \boldsymbol{\mu}$ 进行了变量变换。注意指数是 \mathbf{z} 各分量的偶函数，并且由于对这些分量的积分是在 $(-\infty, \infty)$ 范围内进行的，因子 $(\mathbf{z} + \boldsymbol{\mu})$ 中 \mathbf{z} 的项将因对称性而消失。因此，

$$\mathbb{E}[\mathbf{x}] = \boldsymbol{\mu}, \quad (3.42)$$

所以我们将 $\boldsymbol{\mu}$ 称为高斯分布的均值。

我们现在考虑高斯分布的二阶矩。在单变量情况下，我们考虑由 $\mathbb{E}[x^2]$ 给出的二阶矩。对于多元高斯分布，有由 $\mathbb{E}[x_i x_j]$ 给出的 D^2 个二阶矩，我们可以将它们组合在一起形成矩阵 $\mathbb{E}[\mathbf{x} \mathbf{x}^T]$ 。这个矩阵可以写成

$$\begin{aligned}\mathbb{E}[\mathbf{x} \mathbf{x}^T] &= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\} \mathbf{x} \mathbf{x}^T d\mathbf{x} \\ &= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2}\mathbf{z}^T \Sigma^{-1} \mathbf{z} \right\} (\mathbf{z} + \boldsymbol{\mu})(\mathbf{z} + \boldsymbol{\mu})^T d\mathbf{z}\end{aligned}\quad (3.43)$$

在这里，我们再次使用 $\mathbf{z} = \mathbf{x} - \boldsymbol{\mu}$ 进行变量变换。注意，由于对称性，涉及 $\boldsymbol{\mu} \mathbf{z}^T$ 和 $\boldsymbol{\mu}^T \mathbf{z}$ 的交叉项将再次消失。项 $\boldsymbol{\mu} \boldsymbol{\mu}^T$ 是常数，可以提到积分外面，而积分本身等于1，因为高斯分布是归一化的。考虑涉及 $\mathbf{z} \mathbf{z}^T$ 的项。同样，我们可以利用(3.28)给出的协方差矩阵的特征向量展开式，以及特征向量集的完备性来进行书写

$$\mathbf{z} = \sum_{j=1}^D y_j \mathbf{u}_j \quad (3.44)$$

其中 $y_j = \mathbf{u}_j^T \mathbf{z}$ ，由此可得

$$\begin{aligned} & \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2} \mathbf{z}^T \Sigma^{-1} \mathbf{z} \right\} \mathbf{z} \mathbf{z}^T d\mathbf{z} \\ &= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \sum_{i=1}^D \sum_{j=1}^D \mathbf{u}_i \mathbf{u}_j^T \int \exp \left\{ -\sum_{k=1}^D \frac{y_k^2}{2\lambda_k} \right\} y_i y_j dy \\ &= \sum_{i=1}^D \mathbf{u}_i \mathbf{u}_i^T \lambda_i = \Sigma \end{aligned} \quad (3.45)$$

在此我们利用了特征向量方程(3.28)，以及除非 $i = j$ ，中间一行的积分由于对称性而消失这一事实。在最后一行中，我们利用了结果(2.53)和(3.38)，以及(3.31)。因此，我们有

$$\mathbb{E}[\mathbf{x}\mathbf{x}^T] = \mu\mu^T + \Sigma. \quad (3.46)$$

在定义单个随机变量的方差时，我们在求二阶矩之前减去了均值。类似地，在多变量情况下，再次减去均值很方便，从而得到由 \mathbf{x} 定义的随机向量的协方差

$$\text{cov}[\mathbf{x}] = \mathbb{E} [(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^T]. \quad (3.47)$$

对于高斯分布的特定情况，我们可以利用 $\mathbb{E}[\mathbf{x}] = \mu$ ，以及结果(3.46)，得到

$$\text{cov}[\mathbf{x}] = \Sigma. \quad (3.48)$$

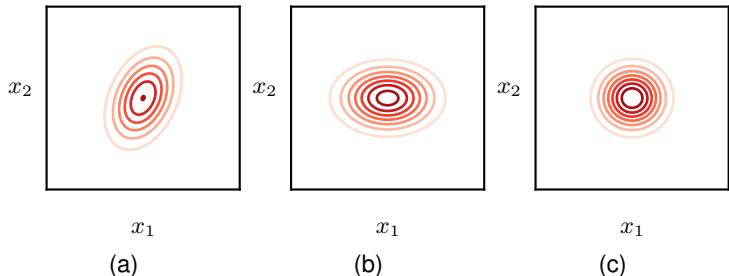
因为参数矩阵 Σ 在高斯分布下控制着 \mathbf{x} 的协方差，所以它被称为协方差矩阵。

3.2.3 局限性

尽管高斯分布 (3.26) 常被用作简单的密度模型，但它存在一些显著的局限性。考虑分布中的自由参数数量。一个一般的对称协方差矩阵 Σ 将有 $D(D+1)/2$ 个独立参数，并且在 μ 中还有另外 D 个独立参数，总共给出 $D(D+3)/2$ 个参数。对于较大的 D ，参数总数因此随 D 呈二次增长，并且处理和求逆大矩阵的计算任务可能变得难以承受。解决这个问题的一种方法是使用协方差矩阵的受限形式。如果我们考虑对角协方差矩阵，使得 $\Sigma = \text{diag}(\sigma_i^2)$ ，那么我们在密度模型中总共具有 $2D$ 个独立参数。相应的恒定密度轮廓由与坐标轴对齐的椭球体给出。我们可以进一步将协方差矩阵限制为与单位矩阵成比例， $\Sigma = \sigma^2 \mathbf{I}$ ，称为各向异性协方差，在模型中给出 $D+1$ 个独立参数以及恒定密度的球面。一般、对角和各向异性协方差矩阵的三种可能性如图3.4所示。

练习3.15

图3.4二维高斯分布的恒定概率密度轮廓，其中协方差矩阵为(a)一般形式，(b)对角形式，在这种情况下椭圆轮廓与坐标轴对齐，以及(c)与单位矩阵成比例，在这种情况下轮廓为同心圆。



不幸的是，尽管这些方法限制了分布中的自由度数量，并使协方差矩阵的求逆操作快得多，但它们也极大地限制了概率密度的形式，并限制了其捕捉数据中有趣相关性的能力。

高斯分布的另一个局限性在于它本质上是单峰的（即有一个单一的最大值），因此无法很好地逼近多峰分布。因此，高斯分布可能在灵活性方面表现得过高，因为它有过多的参数，而在能够充分表示的分布范围方面又过于有限。我们稍后会看到，引入潜在变量（也称为隐藏变量或未观测变量）可以解决这两个问题。特别地，通过引入离散潜在变量得到高斯混合，从而获得了丰富的多峰分布族。类似地，引入连续潜在变量会导致模型中自由参数的数量可以独立于数据空间的维度 D 进行控制，同时仍允许模型捕捉数据集中的主要相关性。

3.2.9节 第16章

3.2.4 条件分布

多元高斯分布的一个重要性质是，如果两组变量联合高斯，那么一组变量在另一组变量条件下的条件分布再次是高斯分布。类似地，任何一组变量的边际分布也是高斯分布。

首先，考虑条件分布的情况。假设 \mathbf{x} 是一个具有高斯分布 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$ 的 D 维向量，并且我们将 \mathbf{x} 划分为两个不相交的子集 \mathbf{x}_a 和 \mathbf{x}_b 。不失一般性，我们可以取 \mathbf{x}_a 构成 \mathbf{x} 的前 M 个分量， \mathbf{x}_b 包含其余的 $D - M$ 个分量，这样

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix}. \quad (3.49)$$

我们还定义了均值向量 $\boldsymbol{\mu}$ 的相应划分，如下所示

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{pmatrix} \quad (3.50)$$

以及由 Σ 给出的协方差矩阵

$$\Sigma = \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{pmatrix}. \quad (3.51)$$

注意，协方差矩阵的对称性 $\Sigma^T = \Sigma$ 意味着 Σ_{aa} 和 Σ_{bb} 是对称的，并且 $\Sigma_{ba} = \Sigma_{ab}^T$ 。

在许多情况下，使用协方差矩阵的逆会很方便：

$$\Lambda \equiv \Sigma^{-1}, \quad (3.52)$$

它被称为精度矩阵。事实上，我们将看到高斯分布的一些性质用协方差来表示最为自然，而其他一些性质从精度的角度来看则形式更简单。因此，我们还引入精度矩阵的分块形式：

$$\Lambda = \begin{pmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_{bb} \end{pmatrix} \quad (3.53)$$

练习3.16

对应于向量 \mathbf{x} 的划分(3.49)。由于不对称矩阵的逆也是对称的，我们看到 Λ_{aa} 和 Λ_{bb} 是对称的，并且 $\Lambda_{ba} = \Lambda_{ab}^T$ 。在这一点上应该强调的是，例如， Λ_{aa} 不是简单地由 Σ_{aa} 的逆给出的。事实上，我们很快就会研究分块矩阵的逆与其子块之间的关系。

我们首先找到条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的表达式。根据概率的乘积法则，我们发现这个条件分布可以通过将联合分布 $p(\mathbf{x}) = p(\mathbf{x}_a, \mathbf{x}_b)$ 中的 \mathbf{x}_b 固定为观测值，并对得到的表达式进行归一化，以获得 \mathbf{x}_a 上的有效概率分布来进行评估。我们不直接进行这种归一化，而是通过考虑由(3.27)给出的高斯分布指数中的二次形式，然后在计算结束时恢复归一化系数，从而更有效地获得解。如果我们利用划分(3.49)、(3.50)和(3.53)，我们得到

$$\begin{aligned} -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) = \\ -\frac{1}{2}(\mathbf{x}_a - \boldsymbol{\mu}_a)^T \Lambda_{aa} (\mathbf{x}_a - \boldsymbol{\mu}_a) - \frac{1}{2}(\mathbf{x}_a - \boldsymbol{\mu}_a)^T \Lambda_{ab} (\mathbf{x}_b - \boldsymbol{\mu}_b) \\ -\frac{1}{2}(\mathbf{x}_b - \boldsymbol{\mu}_b)^T \Lambda_{ba} (\mathbf{x}_a - \boldsymbol{\mu}_a) - \frac{1}{2}(\mathbf{x}_b - \boldsymbol{\mu}_b)^T \Lambda_{bb} (\mathbf{x}_b - \boldsymbol{\mu}_b). \quad (3.54) \end{aligned}$$

我们看到，作为 \mathbf{x}_a 的函数，这再次是一个二次型，因此，相应的条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 将是高斯分布。由于这个分布完全由其均值和协方差表征，我们的目标将是通过检查(3.54)来确定 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的均值和协方差的表达式。

这是一个与高斯分布相关的相当常见的操作的例子，有时称为“配方法”，在这个操作中，我们给定一个定义高斯分布中指数项的二次型，并且我们需要确定相应的均值和协方差。

通过注意到一般高斯分布 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \Sigma)$ 中的指数可以写成这样，这些问题可以直接解决。

$$-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) = -\frac{1}{2}\mathbf{x}^T \Sigma^{-1} \mathbf{x} + \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu} + \text{const} \quad (3.55)$$

其中“const”表示与 \mathbf{x} 无关的项，我们还利用了 Σ 的对称性。因此，如果我们采用一般二次形式并将其表示为(3.55)右侧给出的形式，那么我们可以立即将进入 \mathbf{x} 中二阶项的系数矩阵等同于逆协方差矩阵 Σ^{-1} ，并将 \mathbf{x} 中线性项的系数等同于 $\Sigma^{-1} \boldsymbol{\mu}$ ，由此我们可以得到 $\boldsymbol{\mu}$ 。

现在让我们将此过程应用于条件高斯分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ ，其指数中的二次型由(3.54)给出。我们将分别用 $\boldsymbol{\mu}_{a|b}$ 和 $\Sigma_{a|b}$ 表示此分布的均值和协方差。考虑(3.54)对 \mathbf{x}_a 的函数依赖关系，其中 \mathbf{x}_b 被视为常数。如果我们挑出所有在 \mathbf{x}_a 中二阶的项，我们有

$$-\frac{1}{2}\mathbf{x}_a^T \Lambda_{aa} \mathbf{x}_a \quad (3.56)$$

由此我们可以立即得出结论， $p(\mathbf{x}_a | \mathbf{x}_b)$ 的协方差（逆精度）由下式给出

$$\Sigma_{a|b} = \Lambda_{aa}^{-1}. \quad (3.57)$$

现在考虑(3.54)中所有在 \mathbf{x}_a 中线性的项：

$$\mathbf{x}_a^T \{\Lambda_{aa} \boldsymbol{\mu}_a - \Lambda_{ab} (\mathbf{x}_b - \boldsymbol{\mu}_b)\} \quad (3.58)$$

我们在其中使用了 $\Lambda_{ba}^T = \Lambda_{ab}$ 。从我们对一般形式(3.55)的讨论可知，此表达式中 \mathbf{x}_a 的系数必须等于 $\sum_{a|b}^{-1} \boldsymbol{\mu}_{a|b}$ ，因此，

$$\begin{aligned} \boldsymbol{\mu}_{a|b} &= \Sigma_{a|b} \{\Lambda_{aa} \boldsymbol{\mu}_a - \Lambda_{ab} (\mathbf{x}_b - \boldsymbol{\mu}_b)\} \\ &= \boldsymbol{\mu}_a - \Lambda_{aa}^{-1} \Lambda_{ab} (\mathbf{x}_b - \boldsymbol{\mu}_b) \end{aligned} \quad (3.59)$$

我们在其中使用了(3.57)。

结果(3.57)和(3.59)是根据原始联合分布 $p(\mathbf{x}_a, \mathbf{x}_b)$ 的分块精度矩阵表示的。我们也可以根据相应的分块协方差矩阵来表示这些结果。为此，我们使用以下关于分块矩阵求逆的恒等式：

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{M} & -\mathbf{MBD}^{-1} \\ -\mathbf{D}^{-1}\mathbf{CM} & \mathbf{D}^{-1} + \mathbf{D}^{-1}\mathbf{CM}\mathbf{BD}^{-1} \end{pmatrix} \quad (3.60)$$

我们在其中定义了

$$\mathbf{M} = (\mathbf{A} - \mathbf{BD}^{-1}\mathbf{C})^{-1}. \quad (3.61)$$

练习3.18

量 \mathbf{M}^{-1} 被称为矩阵 (3.60) 左侧关于子矩阵D的舒尔补。根据定义

$$\begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{pmatrix}^{-1} = \begin{pmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_{bb} \end{pmatrix} \quad (3.62)$$

并利用 (3.60)，我们有

$$\Lambda_{aa} = (\Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba})^{-1} \quad (3.63)$$

$$\Lambda_{ab} = -(\Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba})^{-1}\Sigma_{ab}\Sigma_{bb}^{-1}. \quad (3.64)$$

由此我们得到条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的均值和协方差的以下表达式：

$$\boldsymbol{\mu}_{a|b} = \boldsymbol{\mu}_a + \Sigma_{ab}\Sigma_{bb}^{-1}(\mathbf{x}_b - \boldsymbol{\mu}_b) \quad (3.65)$$

$$\Sigma_{a|b} = \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}. \quad (3.66)$$

比较式(3.57)和式(3.66)，我们发现条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 用分块精度矩阵表示时比用分块协方差矩阵表示时具有更简单的形式。注意，由式(3.65)给出的条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的均值是 \mathbf{x}_b 的线性函数，且由式(3.66)给出的协方差与 \mathbf{x}_b 无关。这是线性高斯模型的一个例子。

第11.1.4节

3.2.5 边际分布

我们已经看到，如果联合分布 $p(\mathbf{x}_a, \mathbf{x}_b)$ 是高斯分布，那么条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 也将是高斯分布。现在我们转向对由

$$p(\mathbf{x}_a) = \int p(\mathbf{x}_a, \mathbf{x}_b) d\mathbf{x}_b, \quad (3.67)$$

给出的边际分布的讨论，正如我们将看到的，它也是高斯分布。再一次，我们计算这个分布的策略将是关注联合分布指数中的二次型，从而确定边际分布 $p(\mathbf{x}_a)$ 的均值和协方差。

联合分布的二次型可以使用分块精度矩阵表示为形式(3.54)。我们的目标是对 \mathbf{x}_b 进行积分，最容易实现的方法是首先考虑涉及 \mathbf{x}_b 的项，然后通过配方法来简化积分。只挑选出那些涉及 \mathbf{x}_b 的项，我们有

$$-\frac{1}{2}\mathbf{x}_b^T \Lambda_{bb} \mathbf{x}_b + \mathbf{x}_b^T \mathbf{m} = -\frac{1}{2}(\mathbf{x}_b - \Lambda_{bb}^{-1} \mathbf{m})^T \Lambda_{bb} (\mathbf{x}_b - \Lambda_{bb}^{-1} \mathbf{m}) + \frac{1}{2} \mathbf{m}^T \Lambda_{bb}^{-1} \mathbf{m} \quad (3.68)$$

我们在此定义了

$$\mathbf{m} = \Lambda_{bb} \boldsymbol{\mu}_b - \Lambda_{ba}(\mathbf{x}_a - \boldsymbol{\mu}_a). \quad (3.69)$$

我们看到，对 \mathbf{x}_b 的依赖已被转化为与(3.68)式右侧第一项相对应的高斯分布的标准二次形式，再加上一项不依赖于 \mathbf{x}_b （但依赖于 \mathbf{x}_a ）的项。因此，当我们对这个二次形式取指数时，我们会发现(3.67)式要求的对 \mathbf{x}_b 的积分将采取以下形式

$$\int \exp \left\{ -\frac{1}{2}(\mathbf{x}_b - \boldsymbol{\Lambda}_{bb}^{-1}\mathbf{m})^T \boldsymbol{\Lambda}_{bb} (\mathbf{x}_b - \boldsymbol{\Lambda}_{bb}^{-1}\mathbf{m}) \right\} d\mathbf{x}_b. \quad (3.70)$$

这种积分很容易通过注意到它是对一个未归一化高斯函数的积分来进行，所以结果将是归一化系数的倒数。我们从由(3.26)给出的归一化高斯函数的形式知道，这个系数与均值无关，只取决于协方差矩阵的行列式。因此，通过对 \mathbf{x}_b 进行配方，我们可以对 \mathbf{x}_b 进行积分，使得(3.68)左边贡献中唯一剩下的依赖于 \mathbf{x}_a 的项是(3.68)右边的最后一项，其中 \mathbf{m} 由(3.69)给出。将该项与(3.54)中依赖于 \mathbf{x}_a 的其余项相结合，我们得到

$$\begin{aligned} & \frac{1}{2} [\boldsymbol{\Lambda}_{bb}\boldsymbol{\mu}_b - \boldsymbol{\Lambda}_{ba}(\mathbf{x}_a - \boldsymbol{\mu}_a)]^T \boldsymbol{\Lambda}_{bb}^{-1} [\boldsymbol{\Lambda}_{bb}\boldsymbol{\mu}_b - \boldsymbol{\Lambda}_{ba}(\mathbf{x}_a - \boldsymbol{\mu}_a)] \\ & \quad - \frac{1}{2} \mathbf{x}_a^T \boldsymbol{\Lambda}_{aa} \mathbf{x}_a + \mathbf{x}_a^T (\boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a + \boldsymbol{\Lambda}_{ab}\boldsymbol{\mu}_b) + \text{const} \\ = & \quad - \frac{1}{2} \mathbf{x}_a^T (\boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba}) \mathbf{x}_a \\ & \quad + \mathbf{x}_a^T (\boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba}) \boldsymbol{\mu}_a + \text{const} \end{aligned} \quad (3.71)$$

其中“const”表示与 \mathbf{x}_a 无关的量。再次，通过与(3.55)比较，我们看到边缘分布 $p(\mathbf{x}_a)$ 的协方差由下式给出

$$\boldsymbol{\Sigma}_a = (\boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba})^{-1}. \quad (3.72)$$

类似地，均值由下式给出

$$\boldsymbol{\Sigma}_a (\boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba}) \boldsymbol{\mu}_a = \boldsymbol{\mu}_a \quad (3.73)$$

其中我们使用了(3.72)。协方差(3.72)是根据由(3.53)给出的分块精度矩阵表示的。我们可以像处理条件分布那样，根据由(3.51)给出的协方差矩阵的相应分块来重写它。这些分块矩阵通过以下关系联系起来

$$\begin{pmatrix} \boldsymbol{\Lambda}_{aa} & \boldsymbol{\Lambda}_{ab} \\ \boldsymbol{\Lambda}_{ba} & \boldsymbol{\Lambda}_{bb} \end{pmatrix}^{-1} = \begin{pmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{pmatrix}. \quad (3.74)$$

利用(3.60)，我们于是有

$$(\boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba})^{-1} = \boldsymbol{\Sigma}_{aa}. \quad (3.75)$$

因此，我们得到了直观上令人满意的结果，即边际分布 $p(\mathbf{x}_a)$ 的均值和协方差由下式给出

$$\mathbb{E}[\mathbf{x}_a] = \boldsymbol{\mu}_a \quad (3.76)$$

$$\text{cov}[\mathbf{x}_a] = \boldsymbol{\Sigma}_{aa}. \quad (3.77)$$

我们看到，对于边际分布，均值和协方差用分块协方差矩阵表示最为简单，这与条件分布相反，对于条件分布，分块精度矩阵能给出更简单的表达式。

我们关于分块高斯分布的边际和条件分布的结果可总结如下。给定一个联合高斯分布 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$ ，其中包含 $\boldsymbol{\Lambda} \equiv \boldsymbol{\Sigma}^{-1}$ 以及以下划分

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix}, \quad \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{pmatrix} \quad (3.78)$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{pmatrix}, \quad \boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{\Lambda}_{aa} & \boldsymbol{\Lambda}_{ab} \\ \boldsymbol{\Lambda}_{ba} & \boldsymbol{\Lambda}_{bb} \end{pmatrix} \quad (3.79)$$

那么条件分布由下式给出

$$p(\mathbf{x}_a | \mathbf{x}_b) = \mathcal{N}(\mathbf{x}_a | \boldsymbol{\mu}_{a|b}, \boldsymbol{\Lambda}_{aa}^{-1}) \quad (3.80)$$

$$\boldsymbol{\mu}_{a|b} = \boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{aa}^{-1} \boldsymbol{\Lambda}_{ab} (\mathbf{x}_b - \boldsymbol{\mu}_b) \quad (3.81)$$

边际分布由下式给出

$$p(\mathbf{x}_a) = \mathcal{N}(\mathbf{x}_a | \boldsymbol{\mu}_a, \boldsymbol{\Sigma}_{aa}). \quad (3.82)$$

我们在图3.5中使用一个涉及两个变量的示例来说明与多元高斯相关的条件和边际分布的概念。

3.2.6 贝叶斯定理

在3.2.4节和3.2.5节中，我们考虑了一个高斯 $p(\mathbf{x})$ ，其中我们将向量 \mathbf{x} 划分为两个子向量 $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$ ，然后找到了条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 和边缘分布 $p(\mathbf{x}_a)$ 的表达式。我们注意到条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的均值是 \mathbf{x}_b 的线性函数。这里我们假设给定一个高斯边缘分布 $p(\mathbf{x})$ 和一个高斯条件分布 $p(\mathbf{y} | \mathbf{x})$ ，其中 $p(\mathbf{y} | \mathbf{x})$ 的均值是 \mathbf{x} 的线性函数，协方差与 \mathbf{x} 无关。这是线性高斯模型的一个例子 (Roweis和Ghahramani, 1999)。我们希望找到边缘分布 $p(\mathbf{y})$ 和条件分布 $p(\mathbf{x} | \mathbf{y})$ 。这是几种生成模型中出现的一种结构，在这里推导一般结果将证明是方便的。

我们将边际分布和条件分布设为

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \quad (3.83)$$

$$p(\mathbf{y} | \mathbf{x}) = \mathcal{N}(\mathbf{y} | \mathbf{Ax} + \mathbf{b}, \mathbf{L}^{-1}) \quad (3.84)$$

第11.1.4节

第16章

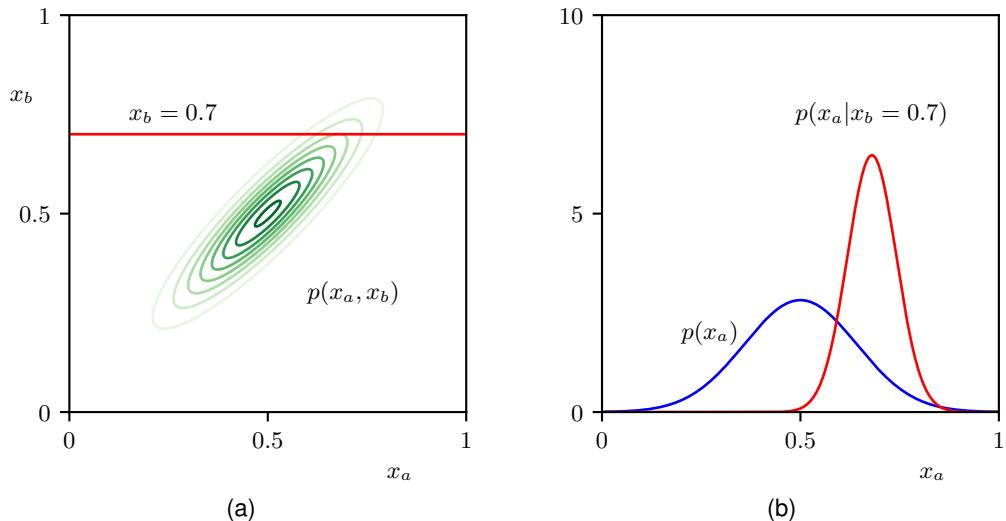


图3.5 (a) 两个变量上高斯分布 $p(x_a, x_b)$ 的等高线。(b) 边际分布 $p(x_a)$ (蓝色曲线) 和 $x_b = 0.7$ 的条件分布 $p(x_a | x_b)$ (红色曲线)。

其中 μ, \mathbf{A} 和 \mathbf{b} 是控制均值的参数, Λ 和 \mathbf{L} 是精度矩阵。如果 \mathbf{x} 的维度为 M 且 \mathbf{y} 的维度为 D , 那么矩阵 \mathbf{A} 的大小为 $D \times M$ 。

首先, 我们找到 \mathbf{x} 和 \mathbf{y} 联合分布的表达式。为此, 我们定义

$$\mathbf{z} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \quad (3.85)$$

然后考虑联合分布的对数:

$$\begin{aligned} \ln p(\mathbf{z}) &= \ln p(\mathbf{x}) + \ln p(\mathbf{y}|\mathbf{x}) \\ &= -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Lambda} (\mathbf{x} - \boldsymbol{\mu}) \\ &\quad -\frac{1}{2}(\mathbf{y} - \mathbf{Ax} - \mathbf{b})^T \mathbf{L} (\mathbf{y} - \mathbf{Ax} - \mathbf{b}) + \text{const} \end{aligned} \quad (3.86)$$

其中“const”表示与 \mathbf{x} 和 \mathbf{y} 无关的项。和之前一样, 我们看到这是 \mathbf{z} 分量的二次函数, 因此, $p(\mathbf{z})$ 是高斯分布。为了找到这个高斯分布的精度, 我们考虑(3.86)中的二阶项, 它可以写成

$$\begin{aligned} &-\frac{1}{2}\mathbf{x}^T (\boldsymbol{\Lambda} + \mathbf{A}^T \mathbf{L} \mathbf{A}) \mathbf{x} - \frac{1}{2}\mathbf{y}^T \mathbf{L} \mathbf{y} + \frac{1}{2}\mathbf{y}^T \mathbf{L} \mathbf{A} \mathbf{x} + \frac{1}{2}\mathbf{x}^T \mathbf{A}^T \mathbf{L} \mathbf{y} \\ &= -\frac{1}{2} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}^T \begin{pmatrix} \boldsymbol{\Lambda} + \mathbf{A}^T \mathbf{L} \mathbf{A} & -\mathbf{A}^T \mathbf{L} \\ -\mathbf{L} \mathbf{A} & \mathbf{L} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = -\frac{1}{2} \mathbf{z}^T \mathbf{R} \mathbf{z} \end{aligned} \quad (3.87)$$

因此, \mathbf{z} 上的高斯分布具有精度 (逆协方差) 矩阵

由.....给出

$$\mathbf{R} = \begin{pmatrix} \Lambda + \mathbf{A}^T \mathbf{L} \Lambda & -\mathbf{A}^T \mathbf{L} \\ -\mathbf{L} \Lambda & \mathbf{L} \end{pmatrix}. \quad (3.88)$$

练习3.23

协方差矩阵是通过取精度的逆得到的，这可以使用矩阵求逆公式(3.60)来完成，结果为

$$\text{cov}[\mathbf{z}] = \mathbf{R}^{-1} = \begin{pmatrix} \Lambda^{-1} & \Lambda^{-1} \mathbf{A}^T \\ \mathbf{A} \Lambda^{-1} & \mathbf{L}^{-1} + \mathbf{A} \Lambda^{-1} \mathbf{A}^T \end{pmatrix}. \quad (3.89)$$

类似地，通过识别(3.86)中的线性项，我们可以找到 \mathbf{z} 上高斯分布的均值，这些线性项由.....给出

$$\mathbf{x}^T \Lambda \mu - \mathbf{x}^T \mathbf{A}^T \mathbf{L} \mathbf{b} + \mathbf{y}^T \mathbf{L} \mathbf{b} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}^T \begin{pmatrix} \Lambda \mu - \mathbf{A}^T \mathbf{L} \mathbf{b} \\ \mathbf{L} \mathbf{b} \end{pmatrix}. \quad (3.90)$$

利用我们之前通过对多元高斯的二次型进行配方法得到的结果(3.55)，我们发现 \mathbf{z} 的均值由.....给出

$$\mathbb{E}[\mathbf{z}] = \mathbf{R}^{-1} \begin{pmatrix} \Lambda \mu - \mathbf{A}^T \mathbf{L} \mathbf{b} \\ \mathbf{L} \mathbf{b} \end{pmatrix}. \quad (3.91)$$

练习3.24

利用(3.89)，我们然后得到

$$\mathbb{E}[\mathbf{z}] = \begin{pmatrix} \mu \\ \mathbf{A} \mu + \mathbf{b} \end{pmatrix}. \quad (3.92)$$

3.2节

接下来，我们找到边际分布 $p(\mathbf{y})$ 的一个表达式，在该表达式中我们已对 \mathbf{x} 进行了边际化。回想一下，当用分块协方差矩阵表示时，高斯随机向量的一部分分量上的边际分布具有特别简单的形式。具体而言，其均值和协方差分别由(3.76)和(3.77)给出。利用(3.89)和(3.92)，我们看到边际分布 $p(\mathbf{y})$ 的均值和协方差由下式给出

$$\mathbb{E}[\mathbf{y}] = \mathbf{A} \mu + \mathbf{b} \quad (3.93)$$

$$\text{cov}[\mathbf{y}] = \mathbf{L}^{-1} + \mathbf{A} \Lambda^{-1} \mathbf{A}^T. \quad (3.94)$$

3.2节

此结果的一个特殊情况是当 $\mathbf{A} = \mathbf{I}$ 时，在这种情况下，边际分布简化为两个高斯分布的卷积，对于此我们看到卷积的均值是两个高斯分布均值之和，卷积的协方差是它们协方差之和。

最后，我们寻求条件 $p(\mathbf{x} | \mathbf{y})$ 的一个表达式。回想一下，条件分布的结果用分块精度矩阵来表示最为简便，利用式(3.57)和式(3.59)。将这些结果应用于式(3.89)和式(3.92)，我们看到条件分布 $p(\mathbf{x} | \mathbf{y})$ 的均值和协方差由下式给出

$$\mathbb{E}[\mathbf{x} | \mathbf{y}] = (\Lambda + \mathbf{A}^T \mathbf{L} \Lambda)^{-1} \{ \mathbf{A}^T \mathbf{L} (\mathbf{y} - \mathbf{b}) + \Lambda \mu \} \quad (3.95)$$

$$\text{cov}[\mathbf{x} | \mathbf{y}] = (\Lambda + \mathbf{A}^T \mathbf{L} \Lambda)^{-1}. \quad (3.96)$$

对这种条件分布的评估可以看作是贝叶斯定理的一个例子，在这个例子中，我们将 $p(\mathbf{x})$ 解释为 \mathbf{x} 上的先验分布。如果观察到变量 \mathbf{y} ，那么条件分布 $p(\mathbf{x} | \mathbf{y})$ 就代表了 \mathbf{x} 上相应的后验分布。在找到边际分布和条件分布后，我们实际上已经将联合分布 $p(\mathbf{z}) = p(\mathbf{x})p(\mathbf{y} | \mathbf{x})$ 表示为 $p(\mathbf{x} | \mathbf{y})p(\mathbf{y})$ 的形式。

这些结果可总结如下。给定 \mathbf{x} 的边际高斯分布以及给定 \mathbf{x} 时 \mathbf{y} 的条件高斯分布，形式为

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \quad (3.97)$$

$$p(\mathbf{y} | \mathbf{x}) = \mathcal{N}(\mathbf{y} | \mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1}), \quad (3.98)$$

那么 \mathbf{y} 的边际分布以及给定 \mathbf{y} 时 \mathbf{x} 的条件分布由下式给出

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y} | \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^T) \quad (3.99)$$

$$p(\mathbf{x} | \mathbf{y}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\Sigma}\{\mathbf{A}^T\mathbf{L}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Lambda}\boldsymbol{\mu}\}, \boldsymbol{\Sigma}) \quad (3.100)$$

其中

$$\boldsymbol{\Sigma} = (\boldsymbol{\Lambda} + \mathbf{A}^T\mathbf{L}\mathbf{A})^{-1}. \quad (3.101)$$

3.2.7 最大似然

给定一个数据集 $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T$ ，其中观测值 $\{\mathbf{x}_n\}$ 假定独立地从多元高斯分布中抽取，我们可以通过最大似然估计该分布的参数。对数似然函数由下式给出

$$\ln p(\mathbf{X} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{ND}{2} \ln(2\pi) - \frac{N}{2} \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}). \quad (3.102)$$

通过简单的重新排列，我们发现似然函数仅通过两个量依赖于数据集

$$\sum_{n=1}^N \mathbf{x}_n, \quad \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^T. \quad (3.103)$$

这些被称为高斯分布的充分统计量。利用(A.19)，对数似然函数关于 $\boldsymbol{\mu}$ 的导数由下式给出

$$\frac{\partial}{\partial \boldsymbol{\mu}} \ln p(\mathbf{X} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^N \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}), \quad (3.104)$$

将此导数设为零，我们得到均值的最大似然估计的解：

$$\boldsymbol{\mu}_{ML} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n, \quad (3.105)$$

练习3.28

它是观测到的数据点集的均值。关于 \sum 对(3.102)进行最大化要复杂得多。最简单的方法是忽略对称性约束，并证明所得解按要求是对称的。该结果的其他推导，其中明确施加了对称性和正定性约束，可在Magnus 和Neudecker(1999)中找到。结果如预期，并具有以下形式

$$\Sigma_{ML} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_{ML})(\mathbf{x}_n - \boldsymbol{\mu}_{ML})^T, \quad (3.106)$$

练习3.29

它涉及 $\boldsymbol{\mu}_{ML}$ ，因为这是关于 $\boldsymbol{\mu}$ 和 \sum 联合最大化的结果。注意， $\boldsymbol{\mu}_{ML}$ 的解(3.105)不依赖于 Σ_{ML} ，所以我们可以先评估 $\boldsymbol{\mu}_{ML}$ ，然后用它来评估 Σ_{ML} 。如果我们在真实分布下评估最大似然解的期望，我们会得到以下结果

$$\mathbb{E}[\boldsymbol{\mu}_{ML}] = \boldsymbol{\mu} \quad (3.107)$$

$$\mathbb{E}[\Sigma_{ML}] = \frac{N-1}{N} \Sigma. \quad (3.108)$$

我们看到均值的最大似然估计的期望等于真实均值。然而，协方差的最大似然估计的期望小于真实值，因此它是有偏的。我们可以通过定义一个不同的估计器 $\widetilde{\Sigma}$ 来纠正这种偏差，其定义如下

$$\widetilde{\Sigma} = \frac{1}{N-1} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_{ML})(\mathbf{x}_n - \boldsymbol{\mu}_{ML})^T. \quad (3.109)$$

显然，从(3.106)和(3.108)可知， $\widetilde{\Sigma}$ 的期望等于 \sum 。

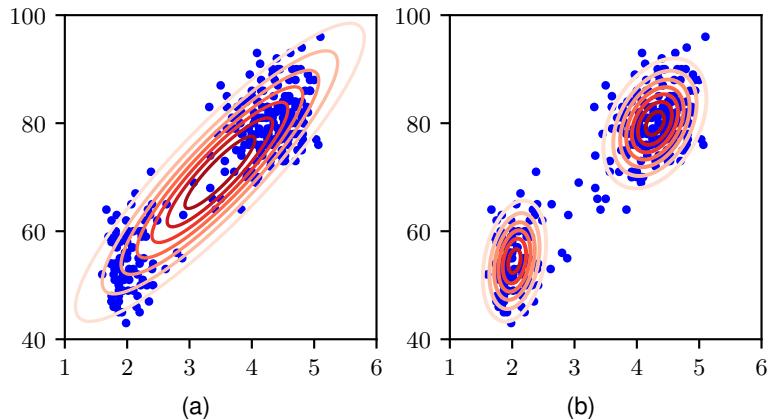
3.2.8 序贯估计

我们对最大似然解的讨论代表了一种批处理方法，其中一次性考虑整个训练数据集。另一种方法是使用序贯方法，这种方法允许逐个处理数据点然后丢弃。当一次性对所有数据点进行批处理不可行时，这些方法对于在线应用和大数据来说很重要。

考虑均值 $\boldsymbol{\mu}_{ML}$ 的最大似然估计量的结果(3.105)，当它基于 N 次观测时，我们将其记为 $\boldsymbol{\mu}_{ML}^{(N)}$ 。

图3.6老忠实间歇泉数据的绘图，其中红色曲线是恒定概率密度的等高线。

(a) 一个使用最大似然法拟合到数据的单高斯分布。注意，这个分布未能捕捉到数据中的两个聚类，实际上将其大部分概率质量置于聚类之间数据相对稀疏的中心区域。(b) 由两个高斯分布的线性组合给出的分布，同样通过最大似然法拟合，它能更好地表示数据。



如果我们剖析出最终数据点 \mathbf{x}_N 的贡献，我们得到

$$\begin{aligned}\mu_{\text{ML}}^{(N)} &= \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n \\ &= \frac{1}{N} \mathbf{x}_N + \frac{1}{N} \sum_{n=1}^{N-1} \mathbf{x}_n \\ &= \frac{1}{N} \mathbf{x}_N + \frac{N-1}{N} \mu_{\text{ML}}^{(N-1)} \\ &= \mu_{\text{ML}}^{(N-1)} + \frac{1}{N} (\mathbf{x}_N - \mu_{\text{ML}}^{(N-1)}).\end{aligned}\quad (3.110)$$

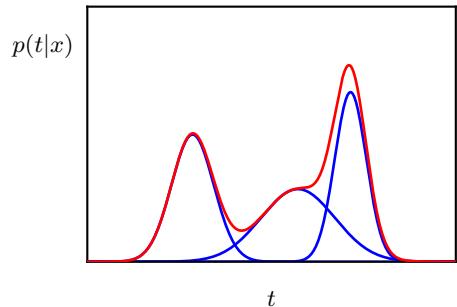
此结果有如下很好的解释。在观察到 $N-1$ 个数据点后，我们用 $\mu_{\text{ML}}^{(N-1)}$ 来估计 μ 。现在我们观察到数据点 \mathbf{x}_N ，然后通过将旧估计值沿“误差信号” $(\mathbf{x}_N - \mu_{\text{ML}}^{(N-1)})$ 的方向移动一个与 $1/N$ 成比例的小量，得到我们的修正估计值 $\mu_{\text{ML}}^{(N)}$ 。注意，随着 N 增加，后续数据点的贡献会变小。

3.2.9 高斯混合

虽然高斯分布有一些重要的分析特性，但在用于对真实数据集进行建模时，它存在显著的局限性。考虑图3.6(a)所示的示例。这就是所谓的“老忠实”数据集，它包含了对美国黄石国家公园老忠实间歇泉喷发的272次测量。每次测量给出喷发持续时间（以分钟为单位，水平轴）和到下一次喷发的时间（以分钟为单位，垂直轴）。我们看到数据集形成了两个主要的簇，并且简单的高斯分布无法捕捉到这种结构。

我们可能期望两个高斯分布的叠加能够更好地表示该数据集中的结构，事实证明确实如此，如图3.6(b)所示。

图3.7 一维高斯混合分布的示例，展示了三个高斯分布（每个都由一个系数缩放），以蓝色显示，它们的总和以红色显示。



第15章

通过对更基本的分布（如高斯分布）进行线性组合形成的这种叠加，可以被表述为称为混合分布的概率模型。在本节中，我们将以高斯分布为例来说混合模型的框架。更一般地，混合模型可以包括其他分布的线性组合，例如二元变量的伯努利分布的混合。在图3.7中，我们看到高斯分布的线性组合可以产生非常复杂的密度。通过使用足够数量的高斯分布，并调整它们的均值、协方差以及线性组合中的系数，几乎任何连续分布都可以以任意精度逼近。

因此，我们考虑形式为 K 的高斯密度的叠加

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad (3.111)$$

它被称为高斯混合模型。每个高斯密度 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ 被称为混合模型的一个分量，并且有其自己的均值 $\boldsymbol{\mu}_k$ 和协方差 $\boldsymbol{\Sigma}_k$ 。具有三个分量的二维高斯混合模型的等高线图和表面图如图3.8所示。

(3.111)中的参数 π_k 被称为混合系数。如果我们对(3.111)两边关于 \mathbf{x} 进行积分，并注意到 $p(\mathbf{x})$ 和各个高斯分量都是归一化的，我们得到

$$\sum_{k=1}^K \pi_k = 1. \quad (3.112)$$

此外，鉴于 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \geq 0$ ，要求 $p(\mathbf{x}) \geq 0$ 的一个充分条件是对于所有 k ， $\pi_k \geq 0$ 。将此与条件(3.112)相结合，我们得到

$$0 \leq \pi_k \leq 1. \quad (3.113)$$

因此，我们可以看到混合系数满足作为概率的要求，并且我们将证明这种对混合分布的概率解释非常强大。

第15章

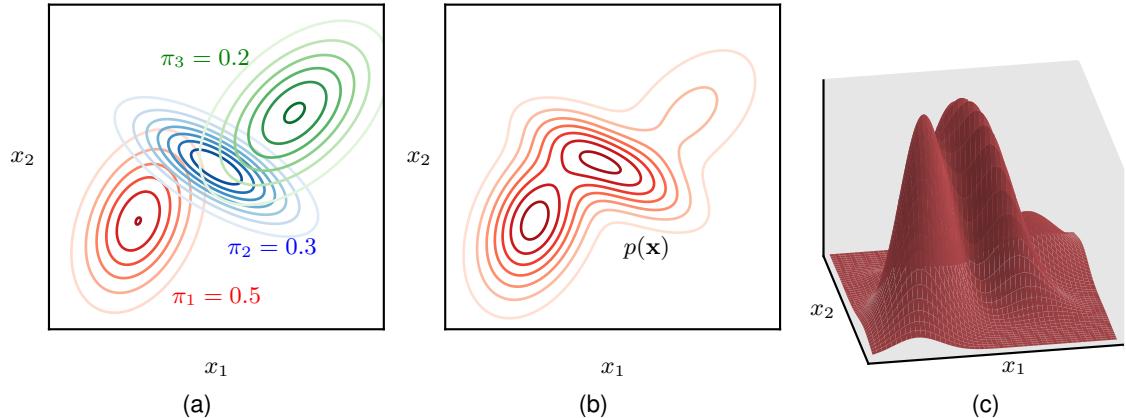


图3.8二维空间中三个高斯分布混合的示意图。(a) 混合各成分的恒定密度等高线，其中三个成分分别用红色、蓝色和绿色表示，混合系数的值显示在每个成分下方。(b) 混合分布的边缘概率密度 $p(\mathbf{x})$ 的等高线。(c) 分布 $p(\mathbf{x})$ 的表面图。

根据概率的和规则与积规则，边缘密度可以写成

$$p(\mathbf{x}) = \sum_{k=1}^K p(k)p(\mathbf{x}|k), \quad (3.114)$$

这等同于 (3.111)，在其中我们可以将 $\pi_k = p(k)$ 视为选择第 k 个组件的先验概率，而密度 $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = p(\mathbf{x} | k)$ 视为在 k 条件下 \mathbf{x} 的概率。正如我们将在后续章节中看到的，相应的后验概率 $p(k | \mathbf{x})$ (也称为责任) 起着重要作用。根据贝叶斯定理，这些由以下给出

$$\begin{aligned} \gamma_k(\mathbf{x}) &\equiv p(k|\mathbf{x}) \\ &= \frac{p(k)p(\mathbf{x}|k)}{\sum_l p(l)p(\mathbf{x}|l)} \\ &= \frac{\pi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_l \pi_l \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}. \end{aligned} \quad (3.115)$$

高斯混合分布的形式由参数 π 、 $\boldsymbol{\mu}$ 和 $\boldsymbol{\Sigma}$ 决定，其中我们使用了符号 $\pi \equiv \{\pi_1, \dots, \pi_K\}$, $\boldsymbol{\mu} \equiv \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K\}$ 和 $\boldsymbol{\Sigma} \equiv \{\boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K\}$ 。设置这些参数值的一种方法是使用最大似然法。根据(3.111)，似然函数的对数由下式给出

$$\ln p(\mathbf{X} | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\} \quad (3.116)$$

其中 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ 。我们立即发现，由于对数内部对 k 的求和，现在的情况比单个高斯分布要复杂得多。

因此，参数的最大似然解不再具有闭式解析解。最大化似然函数的一种方法是使用迭代数值优化技术。或者，我们可以采用一种称为期望最大化的强大框架，它在各种不同的深度生成模型中具有广泛的适用性。

3.3. 周期变量

尽管高斯分布本身以及作为更复杂概率模型的构建块都具有重要的实际意义，但在某些情况下，它们作为连续变量的密度模型是不合适的。在实际应用中出现的一个重要情况是周期变量的情况。

周期变量的一个例子是特定地理位置的风向。例如，我们可能在多个位置测量风向，并希望使用参数分布来总结这些数据。另一个例子是日历时间，我们可能对建模在24小时或年度周期内被认为是周期性的量感兴趣。这样的量可以方便地使用角（极）坐标 $0 \leq \theta < 2\pi$ 来表示。

我们可能会倾向于通过选择某个方向作为原点，然后应用诸如高斯分布之类的传统分布来处理周期性变量。然而，这样的方法会给出强烈依赖于原点的任意选择的结果。例如，假设我们在 $\theta_1 = 1^\circ$ 和 $\theta_2 = 359^\circ$ 处有两个观测值，并且我们使用标准单变量高斯分布对它们进行建模。如果我们将原点置于 0° ，那么这个数据集的样本均值将是 180° ，标准差为 179° ，而如果我们将原点置于 180° ，那么均值将是 0° ，标准差将是 1° 。我们显然需要为周期性变量开发一种特殊的方法。

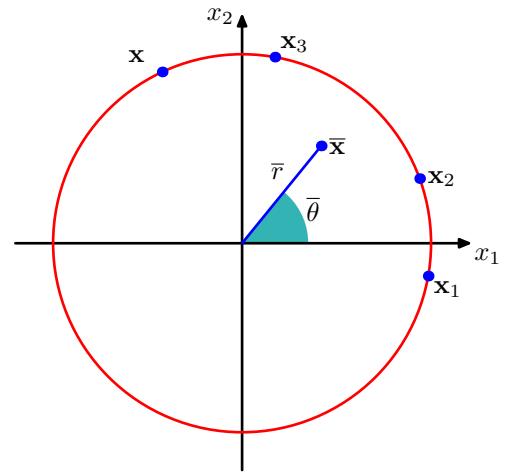
3.3.1 冯·米塞斯分布

让我们考虑评估一组关于周期变量 θ 的观测值 $\mathcal{D} = \{\theta_1, \dots, \theta_N\}$ 的均值的问题，其中 θ 以弧度为单位进行测量。我们已经看到简单平均值 $(\theta_1 + \dots + \theta_N)/N$ 将强烈依赖于坐标。为了找到均值的不变量度，请注意观测值可以被视为单位圆上的点，因此可以用二维单位向量 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 来描述，其中对于 $n = 1, \dots, N$ ， $\|\mathbf{x}_n\| = 1$ ，如图3.9所示。我们可以改为对向量 $\{\mathbf{x}_n\}$ 求平均值以得到

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n \quad (3.117)$$

然后找到这个平均值对应的角度 $\bar{\theta}$ 。显然，这个定义将确保均值的位置与角坐标的原点无关。注意， $\bar{\mathbf{x}}$ 通常会位于单位圆内。

图3.9 周期变量值 θ_n 作为二维向量 \mathbf{x}_n 生活在单位圆上的表示。还显示了这些向量的平均值 $\bar{\mathbf{x}}$ 。



观测值的笛卡尔坐标由 $\mathbf{x}_n = (\cos \theta_n, \sin \theta_n)$ 给出，我们可以将样本均值的笛卡尔坐标写成 $\bar{\mathbf{x}} = (\bar{r} \cos \bar{\theta}, \bar{r} \sin \bar{\theta})$ 的形式。代入(3.117)并使 x_1 和 x_2 分量相等，然后得到

$$\bar{x}_1 = \bar{r} \cos \bar{\theta} = \frac{1}{N} \sum_{n=1}^N \cos \theta_n, \quad \bar{x}_2 = \bar{r} \sin \bar{\theta} = \frac{1}{N} \sum_{n=1}^N \sin \theta_n. \quad (3.118)$$

取比值，并使用恒等式 $\tan \theta = \sin \theta / \cos \theta$ ，我们可以求解 $\bar{\theta}$ 得到

$$\bar{\theta} = \tan^{-1} \left\{ \frac{\sum_n \sin \theta_n}{\sum_n \cos \theta_n} \right\}. \quad (3.119)$$

很快，我们将看到这个结果如何自然地作为最大似然估计出现。

首先，我们需要定义一种高斯分布的周期推广，称为冯·米塞斯分布。这里我们将注意力限制在单变量分布上，尽管在任意维度的超球面上也可以找到类似的周期分布（马尔迪亚和贾普，2000）。

按照惯例，我们将考虑具有周期 2π 的分布 $p(\theta)$ 。在 θ 上定义的任何概率密度 $p(\theta)$ 不仅必须是非负的且积分等于1，而且还必须是周期性的。因此， $p(\theta)$ 必须满足三个条件：

$$p(\theta) \geq 0 \quad (3.120)$$

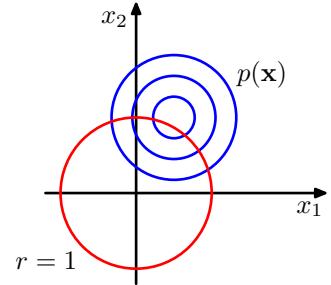
$$\int_0^{2\pi} p(\theta) d\theta = 1 \quad (3.121)$$

$$p(\theta + 2\pi) = p(\theta). \quad (3.122)$$

由 (3.122) 可知，对于任何整数 M ， $p(\theta + M2\pi) = p(\theta)$ 成立。

我们可以轻松获得一个满足以下这三个属性的类高斯分布。考虑两个变量 $\mathbf{x} = (x_1, x_2)$ 上的高斯分布

图3.10 冯·米塞斯分布可以通过考虑形如(3.123)的二维高斯分布推导得出，其密度等高线用蓝色表示，并以红色所示的单位圆为条件。



均值为 $\mu = (\mu_1, \mu_2)$ 且协方差矩阵为 $\Sigma = \sigma^2 \mathbf{I}$ ，其中 \mathbf{I} 是 2×2 单位矩阵，所以

$$p(x_1, x_2) = \frac{1}{2\pi\sigma^2} \exp \left\{ -\frac{(x_1 - \mu_1)^2 + (x_2 - \mu_2)^2}{2\sigma^2} \right\}. \quad (3.123)$$

$p(\mathbf{x})$ 的等高线是圆形，如图3.10所示。

现在假设我们考虑这个分布在固定半径圆上的值。那么根据构造，这个分布将是周期性的，尽管它不会被归一化。我们可以通过从笛卡尔坐标 (x_1, x_2) 转换到极坐标 (r, θ) 来确定这个分布的形式，所以

$$x_1 = r \cos \theta, \quad x_2 = r \sin \theta. \quad (3.124)$$

我们还通过以下方式将平均 μ 映射到极坐标中

$$\mu_1 = r_0 \cos \theta_0, \quad \mu_2 = r_0 \sin \theta_0. \quad (3.125)$$

接下来，我们将这些变换代入二维高斯分布 (3.123)，然后以单位圆 $r = 1$ 为条件，注意我们只对与 θ 的依赖关系感兴趣。关注高斯分布中的指数部分，我们有

$$\begin{aligned} & -\frac{1}{2\sigma^2} \{ (r \cos \theta - r_0 \cos \theta_0)^2 + (r \sin \theta - r_0 \sin \theta_0)^2 \} \\ &= -\frac{1}{2\sigma^2} \{ 1 + r_0^2 - 2r_0 \cos \theta \cos \theta_0 - 2r_0 \sin \theta \sin \theta_0 \} \\ &= \frac{r_0}{\sigma^2} \cos(\theta - \theta_0) + \text{const} \end{aligned} \quad (3.126)$$

其中“const”表示与 θ 无关的项。我们利用了以下三角恒等式：

$$\cos^2 A + \sin^2 A = 1 \quad (3.127)$$

$$\cos A \cos B + \sin A \sin B = \cos(A - B). \quad (3.128)$$

如果我们现在定义 $m = r_0/\sigma^2$ ，我们将得到 $p(\theta)$ 沿单位圆 $r = 1$ 分布的最终表达式，形式为

$$p(\theta|\theta_0, m) = \frac{1}{2\pi I_0(m)} \exp \{m \cos(\theta - \theta_0)\}, \quad (3.129)$$

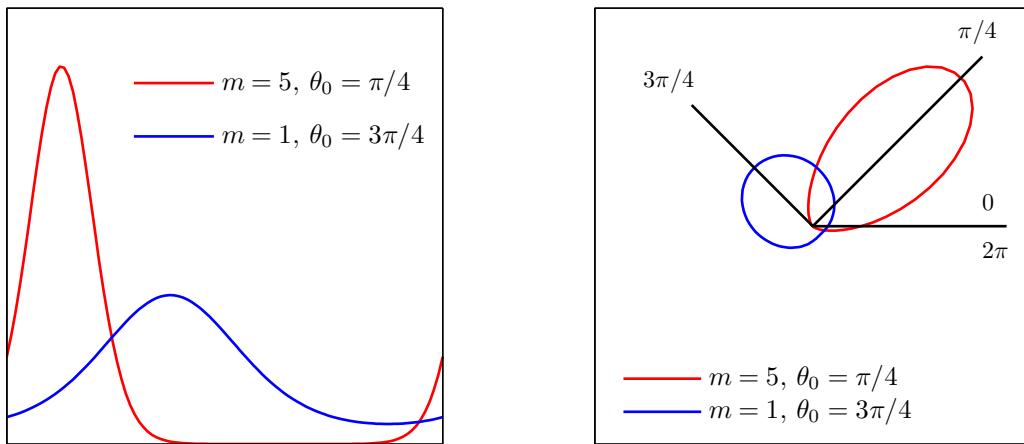


图3.11 针对两个不同参数值绘制的冯·米塞斯分布，左侧以笛卡尔坐标图展示，右侧为相应的极坐标图。

它被称为冯·米塞斯分布或循环正态分布。这里参数 θ_0 对应于分布的均值，而 m ，即所谓的浓度参数，类似于高斯分布的逆方差（即精度）。(3.129)中的归一化系数用 $I_0(m)$ 表示， $I_0(m)$ 是第一类零阶修正贝塞尔函数（Abramowitz和Stegun, 1965），其定义为

$$I_0(m) = \frac{1}{2\pi} \int_0^{2\pi} \exp \{m \cos \theta\} d\theta. \quad (3.130)$$

练习3.31

对于较大的 m ，分布近似为高斯分布。冯·米塞斯分布绘制于图3.11中，函数 $I_0(m)$ 绘制于图3.12中。

现在考虑冯·米塞斯分布中参数 θ_0 和 m 的最大似然估计量。对数似然函数由下式给出

$$\ln p(\mathcal{D}|\theta_0, m) = -N \ln(2\pi) - N \ln I_0(m) + m \sum_{n=1}^N \cos(\theta_n - \theta_0). \quad (3.131)$$

令关于 θ_0 的导数等于零，可得

$$\sum_{n=1}^N \sin(\theta_n - \theta_0) = 0. \quad (3.132)$$

为求解 θ_0 ，我们利用三角恒等式

$$\sin(A - B) = \cos B \sin A - \cos A \sin B \quad (3.133)$$

练习3.32

由此我们得到

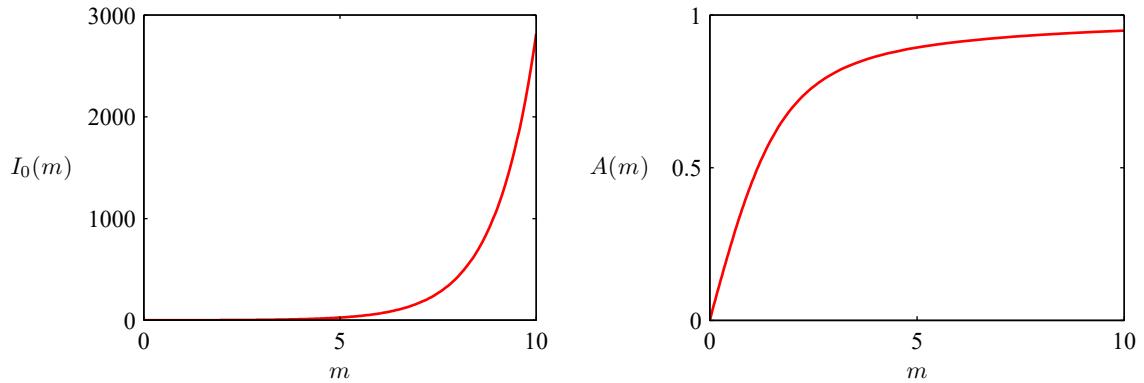


图3.12 由(3.130)定义的贝塞尔函数 $I_0(m)$ 的绘图, 以及由(3.136)定义的函数 $A(m)$ 。

$$\theta_0^{\text{ML}} = \tan^{-1} \left\{ \frac{\sum_n \sin \theta_n}{\sum_n \cos \theta_n} \right\}, \quad (3.134)$$

我们将其识别为之前在二维笛卡尔空间中观察值的均值所得到的结果(3.119)。

类似地, 关于 m 最大化(3.131)并利用 $I'_0(m) = I_1(m)$ (Abramowitz和Stegun, 1965), 我们有

$$A(m_{\text{ML}}) = \frac{1}{N} \sum_{n=1}^N \cos(\theta_n - \theta_0^{\text{ML}}) \quad (3.135)$$

其中我们代入了 θ_0^{ML} 的最大似然解(回想一下我们正在对 θ 和 m 进行联合优化), 并且我们定义了

$$A(m) = \frac{I_1(m)}{I_0(m)}. \quad (3.136)$$

函数 $A(m)$ 绘制在图3.12中。利用三角恒等式(3.128), 我们可以将(3.135)写成以下形式

$$A(m_{\text{ML}}) = \left(\frac{1}{N} \sum_{n=1}^N \cos \theta_n \right) \cos \theta_0^{\text{ML}} + \left(\frac{1}{N} \sum_{n=1}^N \sin \theta_n \right) \sin \theta_0^{\text{ML}}. \quad (3.137)$$

(3.137)式的右侧很容易求值, 并且函数 $A(m)$ 可以通过数值方法求逆。冯·米塞斯分布的一个局限性在于它是单峰的。通过形成冯·米塞斯分布的混合, 我们得到了一个灵活的框架来对周期性变量进行建模, 该框架可以处理多峰性。

为了完整起见, 我们简要提及一些构建周期性分布的替代技术。最简单的方法是使用观测值的直方图, 其中角坐标被划分为固定的区间。

第3.5节

这样做具有简单性和灵活性的优点，但也存在重大局限性，正如我们稍后更详细讨论直方图方法时将会看到的那样。另一种方法，与冯·米塞斯分布一样，从欧几里得空间上的高斯分布开始，但现在是在单位圆上进行边缘化而不是条件化（马尔迪亚和贾普，2000年）。然而，这会导致更复杂的分布形式，将不再进一步讨论。最后，实轴上的任何有效分布（如高斯分布）都可以通过将宽度为 2π 的连续区间映射到周期变量 $(0, 2\pi)$ 上，从而转化为周期分布，这相当于将实轴“缠绕”在单位圆上。同样，得到的分布比冯·米塞斯分布更难处理。

3.4. 指数族

在本章中，到目前为止我们所研究的概率分布（混合模型除外）是一类广泛分布的具体示例，这类分布称为指数族（Duda和Hart，1973年；Bernardo和Smith，1994年）。指数族的成员有许多重要的共同特性，一般性地讨论这些特性很有启发性。

给定参数 η ， \mathbf{x} 上的指数族分布被定义为具有以下形式的分布集合

$$p(\mathbf{x}|\boldsymbol{\eta}) = h(\mathbf{x})g(\boldsymbol{\eta}) \exp\{\boldsymbol{\eta}^T \mathbf{u}(\mathbf{x})\} \quad (3.138)$$

其中 \mathbf{x} 可以是标量或向量，可以是离散的或连续的。这里 $\boldsymbol{\eta}$ 被称为分布的自然参数， $\mathbf{u}(\mathbf{x})$ 是 \mathbf{x} 的某个函数。函数 $g(\boldsymbol{\eta})$ 可以解释为确保分布归一化的系数，因此，它满足

$$g(\boldsymbol{\eta}) \int h(\mathbf{x}) \exp\{\boldsymbol{\eta}^T \mathbf{u}(\mathbf{x})\} d\mathbf{x} = 1 \quad (3.139)$$

如果 \mathbf{x} 是离散变量，则积分用求和代替。

我们首先给出本章前面介绍的一些分布的例子，并证明它们确实属于指数族。首先考虑伯努利分布：

$$p(x|\mu) = \text{Bern}(x|\mu) = \mu^x(1-\mu)^{1-x}. \quad (3.140)$$

将右边表示为对数的指数形式，我们有

$$\begin{aligned} p(x|\mu) &= \exp\{x \ln \mu + (1-x) \ln(1-\mu)\} \\ &= (1-\mu) \exp\left\{\ln\left(\frac{\mu}{1-\mu}\right)x\right\}. \end{aligned} \quad (3.141)$$

与(3.138)进行比较，我们可以确定

$$\boldsymbol{\eta} = \ln\left(\frac{\mu}{1-\mu}\right) \quad (3.142)$$

我们可以求解 μ 得到 $\mu = \sigma(\eta)$ ，其中

$$\sigma(\eta) = \frac{1}{1 + \exp(-\eta)} \quad (3.143)$$

被称为逻辑 sigmoid 函数。因此，我们可以使用标准表示形式(3.138)将伯努利分布写成如下形式

$$p(x|\eta) = \sigma(-\eta) \exp(\eta x) \quad (3.144)$$

其中我们使用了 $1 - \sigma(\eta) = \sigma(-\eta)$ ，这很容易从(3.143)证明。与(3.138)比较可知

$$u(x) = x \quad (3.145)$$

$$h(x) = 1 \quad (3.146)$$

$$g(\eta) = \sigma(-\eta). \quad (3.147)$$

接下来考虑多项分布，对于单个观测值 \mathbf{x} ，其形式为

$$p(\mathbf{x}|\boldsymbol{\mu}) = \prod_{k=1}^M \mu_k^{x_k} = \exp \left\{ \sum_{k=1}^M x_k \ln \mu_k \right\} \quad (3.148)$$

其中 $\mathbf{x} = (x_1, \dots, x_M)^T$ 。同样，我们可以将其写成标准表示形式(3.138)，使得

$$p(\mathbf{x}|\boldsymbol{\eta}) = \exp(\boldsymbol{\eta}^T \mathbf{x}) \quad (3.149)$$

其中 $\eta_k = \ln \mu_k$ ，并且我们已经定义了 $\boldsymbol{\eta} = (\eta_1, \dots, \eta_M)^T$ 。再次，与(3.138)比较，我们有

$$\mathbf{u}(\mathbf{x}) = \mathbf{x} \quad (3.150)$$

$$h(\mathbf{x}) = 1 \quad (3.151)$$

$$g(\boldsymbol{\eta}) = 1. \quad (3.152)$$

注意，参数 η_k 不是独立的，因为参数 μ_k 受到约束

$$\sum_{k=1}^M \mu_k = 1 \quad (3.153)$$

因此，给定参数 μ_k 中的任何 $M - 1$ ，其余参数的值是固定的。在某些情况下，通过仅用 $M - 1$ 个参数来表示分布来消除此约束会很方便。这可以通过使用关系(3.153)来实现，即通过用其余 $\{\mu_k\}$ 来表示 μ_M 从而消除 μ_M ，其中 $k = 1, \dots, M - 1$ ，从而剩下 $M - 1$ 个参数。请注意，这些剩余参数仍然受这些约束

$$0 \leq \mu_k \leq 1, \quad \sum_{k=1}^{M-1} \mu_k \leq 1. \quad (3.154)$$

利用约束条件(3.153), 此表示形式中的多项分布变为

$$\begin{aligned} & \exp \left\{ \sum_{k=1}^M x_k \ln \mu_k \right\} \\ = & \exp \left\{ \sum_{k=1}^{M-1} x_k \ln \mu_k + \left(1 - \sum_{k=1}^{M-1} x_k \right) \ln \left(1 - \sum_{k=1}^{M-1} \mu_k \right) \right\} \\ = & \exp \left\{ \sum_{k=1}^{M-1} x_k \ln \left(\frac{\mu_k}{1 - \sum_{j=1}^{M-1} \mu_j} \right) + \ln \left(1 - \sum_{k=1}^{M-1} \mu_k \right) \right\}. \quad (3.155) \end{aligned}$$

我们现在确定

$$\ln \left(\frac{\mu_k}{1 - \sum_j \mu_j} \right) = \eta_k, \quad (3.156)$$

我们可以通过先对两边关于 k 求和, 然后重新排列并回代来求解 μ_k , 得到

$$\mu_k = \frac{\exp(\eta_k)}{1 + \sum_j \exp(\eta_j)}. \quad (3.157)$$

这称为softmax函数或归一化指数函数。因此, 在此表示形式中, 多项分布具有以下形式

$$p(\mathbf{x}|\boldsymbol{\eta}) = \left(1 + \sum_{k=1}^{M-1} \exp(\eta_k) \right)^{-1} \exp(\boldsymbol{\eta}^T \mathbf{x}). \quad (3.158)$$

这是指数族的标准形式, 参数向量为 $\boldsymbol{\eta} = (\eta_1, \dots, \eta_{M-1})^T$, 其中

$$\mathbf{u}(\mathbf{x}) = \mathbf{x} \quad (3.159)$$

$$h(\mathbf{x}) = 1 \quad (3.160)$$

$$g(\boldsymbol{\eta}) = \left(1 + \sum_{k=1}^{M-1} \exp(\eta_k) \right)^{-1}. \quad (3.161)$$

最后, 让我们考虑高斯分布。对于单变量高斯分布, 我们有

$$p(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2}(x - \mu)^2 \right\} \quad (3.162)$$

$$= \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2}x^2 + \frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}\mu^2 \right\}, \quad (3.163)$$

经过一些简单的重新排列后, 它可以写成标准指数族形式
(3.138), 其中

$$\boldsymbol{\eta} = \begin{pmatrix} \mu/\sigma^2 \\ -1/2\sigma^2 \end{pmatrix} \quad (3.164)$$

$$\mathbf{u}(x) = \begin{pmatrix} x \\ x^2 \end{pmatrix} \quad (3.165)$$

$$h(\mathbf{x}) = (2\pi)^{-1/2} \quad (3.166)$$

$$g(\boldsymbol{\eta}) = (-2\eta_2)^{1/2} \exp\left(\frac{\eta_1^2}{4\eta_2}\right). \quad (3.167)$$

最后，我们有时会使用 (3.138) 的一种受限形式，其中我们选择 $\mathbf{u}(\mathbf{x}) = \mathbf{x}$ 。然而，通过注意到如果 $f(\mathbf{x})$ 是一个归一化密度，那么这可以在某种程度上进行推广。

$$\frac{1}{s} f\left(\frac{1}{s}\mathbf{x}\right) \quad (3.168)$$

也是一个归一化密度，其中 $s > 0$ 是一个尺度参数。将这些结合起来，我们得到了一组受限的指数族类条件密度，形式为

$$p(\mathbf{x}|\boldsymbol{\lambda}_k, s) = \frac{1}{s} h\left(\frac{1}{s}\mathbf{x}\right) g(\boldsymbol{\lambda}_k) \exp\left\{\frac{1}{s}\boldsymbol{\lambda}_k^T \mathbf{x}\right\}. \quad (3.169)$$

请注意，我们允许每个类别有自己的参数向量 $\boldsymbol{\lambda}_k$ ，但我们假设这些类别共享相同的尺度参数 s 。

3.4.1 充分统计量

现在让我们考虑使用最大似然技术在一般指数族分布 (3.138) 中估计参数向量 $\boldsymbol{\eta}$ 的问题。对 (3.139) 两边关于 $\boldsymbol{\eta}$ 求梯度，我们有

$$\begin{aligned} \nabla g(\boldsymbol{\eta}) \int h(\mathbf{x}) \exp\{\boldsymbol{\eta}^T \mathbf{u}(\mathbf{x})\} d\mathbf{x} \\ + g(\boldsymbol{\eta}) \int h(\mathbf{x}) \exp\{\boldsymbol{\eta}^T \mathbf{u}(\mathbf{x})\} \mathbf{u}(\mathbf{x}) d\mathbf{x} = 0. \end{aligned} \quad (3.170)$$

重新整理并再次使用 (3.139)，然后得到

$$-\frac{1}{g(\boldsymbol{\eta})} \nabla g(\boldsymbol{\eta}) = g(\boldsymbol{\eta}) \int h(\mathbf{x}) \exp\{\boldsymbol{\eta}^T \mathbf{u}(\mathbf{x})\} \mathbf{u}(\mathbf{x}) d\mathbf{x} = \mathbb{E}[\mathbf{u}(\mathbf{x})]. \quad (3.171)$$

因此我们得到结果

$$-\nabla \ln g(\boldsymbol{\eta}) = \mathbb{E}[\mathbf{u}(\mathbf{x})]. \quad (3.172)$$

练习3.36

注意， $\mathbf{u}(\mathbf{x})$ 的协方差可以用 $g(\boldsymbol{\eta})$ 的二阶导数来表示，对于高阶矩也是类似的。因此，只要我们能对指数族的分布进行归一化，我们总能通过简单的求导找到它的矩。

现在考虑一组由 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ 表示的独立同分布数据，其似然函数由下式给出

$$p(\mathbf{X}|\boldsymbol{\eta}) = \left(\prod_{n=1}^N h(\mathbf{x}_n) \right) g(\boldsymbol{\eta})^N \exp \left\{ \boldsymbol{\eta}^T \sum_{n=1}^N \mathbf{u}(\mathbf{x}_n) \right\}. \quad (3.173)$$

将 $\ln p(\mathbf{X} | \boldsymbol{\eta})$ 关于 $\boldsymbol{\eta}$ 的梯度设为零，我们得到最大似然估计量 $\boldsymbol{\eta}_{ML}$ 需要满足的以下条件：

$$-\nabla \ln g(\boldsymbol{\eta}_{ML}) = \frac{1}{N} \sum_{n=1}^N \mathbf{u}(\mathbf{x}_n), \quad (3.174)$$

原则上可以求解得到 $\boldsymbol{\eta}_{ML}$ 。我们看到，最大似然估计器的解仅通过 $\sum_n \mathbf{u}(\mathbf{x}_n)$ 依赖于数据，因此 $\sum_n \mathbf{u}(\mathbf{x}_n)$ 被称为分布(3.138)的充分统计量。我们不需要存储整个数据集本身，只需要存储充分统计量的值。例如，对于伯努利分布，函数 $\mathbf{u}(x)$ 仅由 x 给出，因此我们只需要保留数据点的和 $\{x_n\}$ ，而对于高斯分布 $\mathbf{u}(x) = (x, x^2)^T$ ，所以我们应该同时保留 $\{x_n\}$ 的和以及 $\{x_n^2\}$ 的和。

如果我们考虑极限 $N \rightarrow \infty$ ，那么(3.174)的右侧变为 $\mathbb{E}[\mathbf{u}(\mathbf{x})]$ ，因此通过与(3.172)比较，我们发现在这个极限下， $\boldsymbol{\eta}_{ML}$ 将等于真值 $\boldsymbol{\eta}$ 。

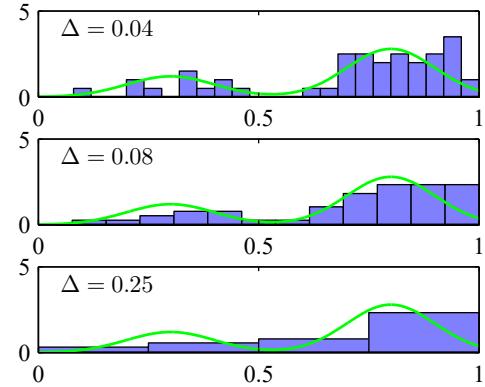
3.5. 非参数方法

在本章中，我们一直专注于使用具有特定函数形式的概率分布，这些分布由少量参数控制，其值要从数据集中确定。这被称为密度建模的参数方法。这种方法的一个重要局限性在于，所选的密度可能是生成数据的分布的一个糟糕模型，这可能导致预测性能不佳。例如，如果生成数据的过程是多峰的，那么高斯分布（必然是单峰的）永远无法捕捉到分布的这一方面。在本节的最后，我们考虑一些密度估计的非参数方法，这些方法对分布的形式几乎不做假设。

3.5.1 直方图

让我们从讨论用于密度估计的直方图方法开始，我们已经在图2.5中的边际和条件分布以及图3.2中的中心极限定理的背景下遇到过这些方法。在这里，我们更详细地探讨直方图密度模型的属性，重点关注具有单个连续变量 x 的情况。标准直方图只是将 x 划分为宽度为 Δ_i 的不同区间，然后计算落入 x 的观测值数量 n_i

图3.13 直方图法进行密度估计的示意图，其中从绿色曲线所示分布生成了一个包含50个数据点的数据集。基于(3.175)且具有共同箱宽 Δ 的直方图密度估计，针对 Δ 的不同值进行了展示。



在箱 i 中。为了将这个计数转换为归一化概率密度，我们只需除以观测总数 N 和箱宽 Δ_i ，以获得每个箱的概率值：

$$p_i = \frac{n_i}{N\Delta_i} \quad (3.175)$$

对此，很容易看出 $\int p(x)dx = 1$ 。这给出了一个在每个箱的宽度上恒定的密度 $p(x)$ 模型。通常选择箱具有相同的宽度 $\Delta_i = \Delta$ 。

在图3.13中，我们展示了一个直方图密度估计的示例。这里的数据是从对应于绿色曲线的分布中抽取的，该绿色曲线由两个高斯分布混合而成。还展示了对于箱宽 Δ 的三种不同选择的三个直方图密度估计示例。我们看到，当 Δ 非常小时（上图），得到的密度模型非常尖锐，有许多在生成数据集的基础分布中不存在的结构。相反，如果 Δ 太大（下图），那么结果是一个过于平滑的模型，因此无法捕捉绿色曲线的双峰特性。对于 Δ 的某个中间值（中图）可获得最佳结果。原则上，直方图密度模型也取决于箱的边缘位置的选择，尽管这通常比箱宽 Δ 的影响小得多。

请注意，直方图方法具有这样的特性（与即将讨论的方法不同）：一旦计算出直方图，数据集本身就可以丢弃，如果数据集很大，这可能是有利的。此外，如果数据点按顺序到达，直方图方法很容易应用。

在实践中，直方图技术对于快速可视化一两个维度的数据可能很有用，但不适用于大多数密度估计应用。一个明显的问题是，估计的密度存在不连续性，这是由于箱边缘造成的，而不是生成数据的基础分布的任何特性。直方图方法的一个主要限制是其随维度的缩放。如果我们将 D 维空间中的每个变量划分为 M 个箱，那么箱的总数将是 M^D 。

这种随 D 的指数缩放是维度诅咒的一个例子。在高维空间中，为提供局部概率密度的有意义估计所需的数据量将是令人望而却步的。

第1章

然而，密度估计的直方图方法确实给我们上了两个重要的课。首先，为了估计特定位置的概率密度，我们应该考虑位于该点某个局部邻域内的数据点。请注意，局部性的概念要求我们假设某种形式的距离度量，在这里我们一直假设为欧几里得距离。对于直方图，这种邻域属性由箱定义，并且有一个自然的“平滑”参数描述局部区域的空间范围，在这种情况下是箱宽。其次，为了获得好的结果，平滑参数的值既不能太大也不能太小。这让人想起多项式回归中模型复杂度的选择，其中多项式的次数 M ，或者正则化参数的值 λ ，对于某个中间值是最优的，既不太大也不太小。有了这些见解，我们现在转向讨论两种广泛使用的非参数密度估计技术，核估计器和最近邻，它们在维度上比简单的直方图模型具有更好的扩展性。

第3.1.2节

3.5.2 核密度

假设观测值是从某个未知概率密度 $p(\mathbf{x})$ 在某个 D 维空间中抽取的，我们将该空间视为欧几里得空间，并且我们希望估计 $p(\mathbf{x})$ 的值。根据我们之前关于局部性的讨论，让我们考虑某个包含 \mathbf{x} 的小区域 \mathcal{R} 。与该区域相关的概率质量由下式给出

$$P = \int_{\mathcal{R}} p(\mathbf{x}) d\mathbf{x}. \quad (3.176)$$

现在假设我们已经收集了一个数据集，该数据集由从 $p(\mathbf{x})$ 中抽取的 N 个观测值组成。因为每个数据点落在 \mathcal{R} 内的概率为 P ，所以位于 \mathcal{R} 内的点的总数 K 将根据二项分布进行分布：

$$\text{Bin}(K|N, P) = \frac{N!}{K!(N-K)!} P^K (1-P)^{N-K}. \quad (3.177)$$

使用(3.11)，我们看到落在该区域内的点的平均比例是 $\mathbb{E}[K/N] = P$ ，类似地，使用(3.12)，我们看到围绕该平均值的方差是 $\text{var}[K/N] = P(1-P)/N$ 。对于大的 N ，这种分布将在平均值附近急剧峰值，所以

$$K \simeq NP. \quad (3.178)$$

然而，如果我们还假设区域 \mathcal{R} 足够小，以至于概率密度 $p(\mathbf{x})$ 在该区域上大致恒定，那么我们有

$$P \simeq p(\mathbf{x})V \quad (3.179)$$

其中 V 是 \mathcal{R} 的体积。将(3.178)和(3.179)相结合，我们得到形式为的密度估计

$$p(\mathbf{x}) = \frac{K}{NV}. \quad (3.180)$$

注意，(3.180)的有效性取决于两个相互矛盾的假设，即区域 \mathcal{R} 足够小，以至于在该区域上密度近似恒定，并且（相对于该密度的值）足够大，以至于落在该区域内的点的数量 K 足以使二项分布具有尖锐的峰值。

我们可以通过两种不同的方式利用结果(3.180)。要么我们固定 K 并根据数据确定 V 的值，这就产生了稍后将讨论的 K -最近邻技术；要么我们固定 V 并根据数据确定 K ，这就产生了核方法。可以证明，只要 V 随着 N 收缩且 K 随着 N 以适当的速率增长，在极限 $N \rightarrow \infty$ 下， K -最近邻密度估计器和核密度估计器都收敛到真实概率密度(杜达和哈特，1973年)。

我们首先详细讨论核方法。首先，我们将区域 \mathcal{R} 设为以我们希望确定概率密度的点 \mathbf{x} 为中心的一个小超立方体。为了计算落在该区域内的点的数量 K ，方便起见定义以下函数：

$$k(\mathbf{u}) = \begin{cases} 1, & |u_i| \leq 1/2, \quad i = 1, \dots, D, \\ 0, & \text{otherwise,} \end{cases} \quad (3.181)$$

它表示一个以原点为中心的单位立方体。函数 $k(\mathbf{u})$ 是核函数的一个示例，在这种情况下，它也被称为Parzen窗。根据(3.181)，如果数据点 \mathbf{x}_n 位于以 \mathbf{x} 为中心、边长为 h 的立方体内，则量 $k((\mathbf{x} - \mathbf{x}_n)/h)$ 将为1，否则为零。因此，位于这个立方体内的数据点总数将是

$$K = \sum_{n=1}^N k\left(\frac{\mathbf{x} - \mathbf{x}_n}{h}\right). \quad (3.182)$$

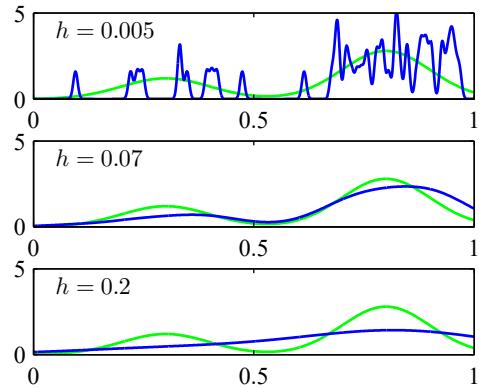
将此表达式代入(3.180)，然后得到 \mathbf{x} 处估计密度的以下结果：

$$p(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N \frac{1}{h^D} k\left(\frac{\mathbf{x} - \mathbf{x}_n}{h}\right) \quad (3.183)$$

我们用 $V = h^D$ 表示边长为 h 的超立方体在 D 维空间中的体积。利用函数 $k(\mathbf{u})$ 的对称性，我们现在可以重新解释这个方程，不是将其看作以 \mathbf{x} 为中心的单个立方体，而是看作以 N 个数据点 \mathbf{x}_n 为中心的 N 个立方体的总和。

就目前而言，核密度估计器 (3.183) 将遭遇直方图方法所面临的相同问题之一，即存在人为的不连续性，在这种情况下是在立方体的边界处。

图3.14将核密度模型(3.184)应用于用于演示图3.13中直方图方法的同一数据集的图示。我们看到 h 作为一个平滑参数，如果设置得太小(上图)，结果是一个噪声很大的密度模型，而如果设置得太大(下图)，那么生成数据的基础分布的双峰性质(由绿色曲线显示)就会被消除。对于 h 的某个中间值(中图)，可以得到最佳密度模型。



如果我们选择一个更平滑的核函数，就可以得到一个更平滑的密度模型，一个常见的选择是高斯函数，它产生了以下核密度模型：

$$p(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N \frac{1}{(2\pi h^2)^{D/2}} \exp \left\{ -\frac{\|\mathbf{x} - \mathbf{x}_n\|^2}{2h^2} \right\} \quad (3.184)$$

其中 h 表示高斯分量的标准差。因此，我们的密度模型是通过在每个数据点上放置一个高斯分布，将整个数据集上的贡献相加，然后除以 N ，以便正确地对密度进行归一化得到的。在图3.14中，我们将模型(3.184)应用于前面用于演示直方图技术的数据集。我们看到，正如预期的那样，参数 h 起到了平滑参数的作用，并且在 h 较小时对噪声的敏感性与在 h 较大时的过度平滑之间存在权衡。同样， h 的优化是一个模型复杂度问题，类似于直方图密度估计中的箱宽选择或曲线拟合中使用的多项式的次数选择。

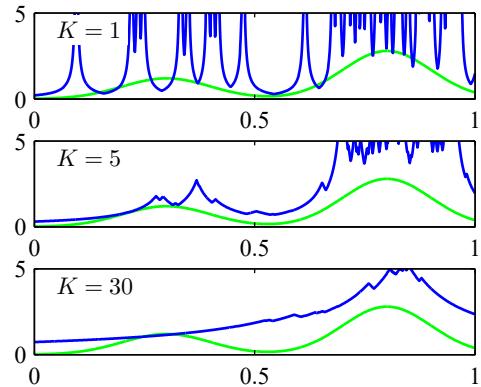
我们可以在(3.183)中选择任何其他满足条件的核函数 $k(\mathbf{u})$

$$k(\mathbf{u}) \geq 0, \quad (3.185)$$

$$\int k(\mathbf{u}) d\mathbf{u} = 1, \quad (3.186)$$

这确保了所得概率分布在任何地方都是非负的，并且积分值为1。由(3.183)给出的密度模型类被称为核密度估计器或Parzen估计器。它有一个很大的优点，即在“训练”阶段不涉及计算，因为这只需要存储训练集。然而，这也是它的一个很大的弱点，因为评估密度的计算成本随着数据集的大小线性增长。

图3.15 使用与图3.14和3.13相同的数据集进行 K -最近邻密度估计的示意图。我们看到参数 K 控制着平滑程度，因此 K 的小值会导致非常嘈杂的密度模型（上图），而大值（下图）会平滑掉生成数据集的真实分布（由绿色曲线显示）的双峰性质。



3.5.3 最近邻

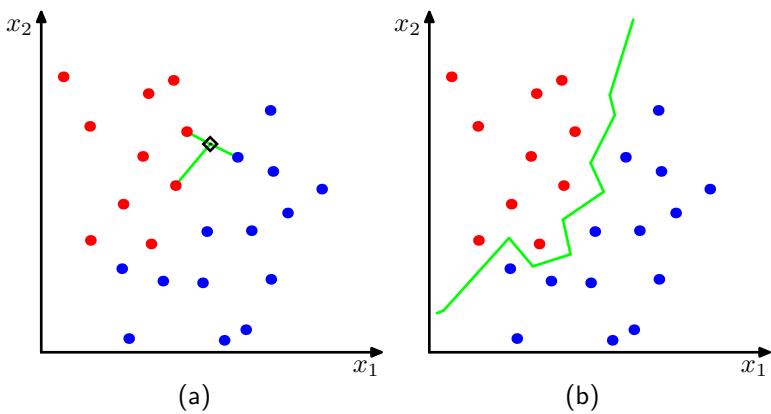
核密度估计方法的一个困难在于，控制核宽度的参数 h 对所有核都是固定的。在高数据密度区域， h 的大值可能导致过度平滑，并消除原本可以从数据中提取的结构。然而，减小 h 可能会导致数据空间中密度较小的其他地方的估计有噪声。因此， h 的最佳选择可能取决于数据空间内的位置。最近邻密度估计方法解决了这个问题。

因此，我们回到局部密度估计的一般结果(3.180)，不是固定 V 并从数据中确定 K 的值，而是考虑 K 的固定值，并使用数据找到 V 的合适值。为此，我们考虑一个以点 \mathbf{x} 为中心的小球体，在该点我们希望估计密度 $p(\mathbf{x})$ ，并且我们允许球体的半径增长，直到它恰好包含 K 个数据点。然后，密度 $p(\mathbf{x})$ 的估计值由(3.180)给出，其中 V 设置为所得球体的体积。这种技术被称为 K 最近邻，并且在图3.15中针对参数 K 的各种选择进行了说明，使用与图3.13和3.14中相同的数据集。我们看到， K 的值现在控制平滑程度，并且再次存在 K 的最佳选择，既不过大也不过小。请注意，由 K 最近邻产生的模型不是一个真正的密度模型，因为在所有空间上的积分发散。

练习3.38

我们通过展示如何将用于密度估计的 K 近邻技术扩展到分类问题来结束本章。为此，我们分别将 K 近邻密度估计技术应用于每个类别，然后利用贝叶斯定理。假设我们有一个数据集，其中包含类别 \mathcal{C}_k 中的 N_k 个点，总共有 N 个点，即 $\sum_k N_k = N$ 。如果我们希望对一个新点 \mathbf{x} 进行分类，我们以 \mathbf{x} 为中心绘制一个恰好包含 K 个点的球体，而不考虑它们的类别。假设这个球体的体积为 V ，并且包含来自类别 \mathcal{C}_k 的 K_k 个点。那么(3.180)提供了与每个类别相关的密度估计：

图3.16 (a) 在 K 最近邻分类器中，一个新点（用黑色菱形表示）根据 K 个最近的训练数据点的多数类成员关系进行分类，在这种情况下为 $K = 3$ 。 (b) 在最近邻 ($K = 1$) 分类方法中，得到的决策边界由形成不同类点对的垂直平分线的超平面组成。



$$p(\mathbf{x}|\mathcal{C}_k) = \frac{K_k}{N_k V}. \quad (3.187)$$

类似地，无条件密度由下式给出

$$p(\mathbf{x}) = \frac{K}{NV} \quad (3.188)$$

类先验由下式给出

$$p(\mathcal{C}_k) = \frac{N_k}{N}. \quad (3.189)$$

我们现在可以使用贝叶斯定理将(3.187)、(3.188)和(3.189)结合起来，以获得类成员的后验概率：

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})} = \frac{K_k}{K}. \quad (3.190)$$

通过将测试点 \mathbf{x} 分配给具有最大后验概率（对应于 K_k/K 的最大值）的类别，我们可以将错误分类的概率降至最低。因此，为了对新点进行分类，我们从训练数据集中识别出 K 个最近的点，然后将新点分配给该集合中具有最多代表的类别。平局情况可以随机打破。 $K = 1$ 的特殊情况称为最近邻规则，因为测试点被简单地分配到与训练集中最近点相同的类别。这些概念在图3.16中进行了说明。

最近邻 ($K = 1$) 分类器的一个有趣特性是，在极限 $N \rightarrow \infty$ 情况下，错误率永远不会超过最优分类器（即使使用真实类别分布的分类器，见Cover和Hart，1967年）可达到的最小错误率的两倍。

如前所述， K -最近邻方法和核密度估计器都需要存储整个训练数据集，如果数据集很大，这将导致计算成本高昂。

通过构建基于树的搜索结构，可以抵消这种影响，代价是进行一些额外的一次性计算，从而在不详尽搜索数据集的情况下高效地找到（近似）近邻。然而，这些非参数方法仍然受到严重限制。另一方面，我们已经看到，简单的参数模型在它们能够表示的分布形式方面非常受限。因此，我们需要找到非常灵活的密度模型，并且这些模型的复杂度可以独立于训练集的大小进行控制，这可以通过深度神经网络来实现。

练习

3.1 (*) 验证伯努利分布 (3.2) 满足以下性质：

$$\sum_{x=0}^1 p(x|\mu) = 1 \quad (3.191)$$

$$\mathbb{E}[x] = \mu \quad (3.192)$$

$$\text{var}[x] = \mu(1 - \mu). \quad (3.193)$$

证明伯努利分布的随机二元变量 x 的熵 $H[x]$ 由下式给出

$$H[x] = -\mu \ln \mu - (1 - \mu) \ln(1 - \mu). \quad (3.194)$$

3.2 (**) 由(3.2)给出的伯努利分布形式在 x 的两个值之间不对称。在某些情况下，使用等效公式 $x \in \{-1, 1\}$ 会更方便，在这种情况下，分布可以写成

$$p(x|\mu) = \left(\frac{1-\mu}{2}\right)^{(1-x)/2} \left(\frac{1+\mu}{2}\right)^{(1+x)/2} \quad (3.195)$$

其中 $\mu \in [-1, 1]$ 。证明分布(3.195)是归一化的，并计算其均值、方差和熵。

3.3 (**) 在本练习中，我们证明二项分布(3.9)是归一化的。首先，使用从总共 N 个中选择 m 个相同对象的组合数的定义(3.10)来证明

$$\binom{N}{m} + \binom{N}{m-1} = \binom{N+1}{m}. \quad (3.196)$$

使用这个结果通过归纳法证明以下结果：

$$(1+x)^N = \sum_{m=0}^N \binom{N}{m} x^m, \quad (3.197)$$

这就是所谓的二项式定理，它对 x 的所有实数值都成立。最后，证明二项分布是归一化的，以便

$$\sum_{m=0}^N \binom{N}{m} \mu^m (1-\mu)^{N-m} = 1, \quad (3.198)$$

这可以通过首先从求和中提出一个因子 $(1-\mu)^N$ ，然后利用二项式定理来完成。

3.4 (**) 证明二项分布的均值由(3.11)给出。为此，对归一化条件(3.198)两边关于 μ 求导，然后重新整理以得到 n 均值的表达式。类似地，通过对(3.198)关于 μ 求两次导，并利用二项分布均值的结果(3.11)，证明二项分布方差的结果(3.12)。

3.5 (*) 证明多元高斯分布(3.26)的众数由 μ 给出。

3.6 (**) 假设 \mathbf{x} 具有均值为 μ 且协方差为 Σ 的高斯分布。证明线性变换后的变量 $\mathbf{Ax} + \mathbf{b}$ 也是高斯分布，并求出其均值和协方差。

3.7 (***) 证明两个高斯分布 $q(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$ 和 $p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ 之间的库尔贝克-莱布勒散度由下式给出

$$\begin{aligned} & \text{KL}(q(\mathbf{x}) \| p(\mathbf{x})) \\ &= \frac{1}{2} \left\{ \ln \frac{|\boldsymbol{\Sigma}_p|}{|\boldsymbol{\Sigma}_q|} - D + \text{Tr}(\boldsymbol{\Sigma}_p^{-1} \boldsymbol{\Sigma}_q) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) \right\} \end{aligned} \quad (3.199)$$

其中 $\text{Tr}(\cdot)$ 表示矩阵的迹，且 D 是 \mathbf{x} 的维度。

3.8 (**) 本练习表明，对于给定协方差，具有最大熵的多元分布是高斯分布。分布 $p(\mathbf{x})$ 的熵由下式给出

$$H[\mathbf{x}] = - \int p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x}. \quad (3.200)$$

我们希望在所有分布 $p(\mathbf{x})$ 上最大化 $H[\mathbf{x}]$ ，条件是 $p(\mathbf{x})$ 归一化且具有特定均值和协方差，使得

$$\int p(\mathbf{x}) d\mathbf{x} = 1 \quad (3.201)$$

$$\int p(\mathbf{x}) \mathbf{x} d\mathbf{x} = \boldsymbol{\mu} \quad (3.202)$$

$$\int p(\mathbf{x}) (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T d\mathbf{x} = \boldsymbol{\Sigma}. \quad (3.203)$$

通过对(3.200)进行变分最大化，并使用拉格朗日乘数来强制执行约束(3.201)、(3.202)和(3.203)，证明最大似然分布由高斯分布(3.26)给出。

3.9 (★★★) 证明多元高斯分布 $\mathcal{N}(\mathbf{x} | \mu, \Sigma)$ 的熵由下式给出

$$H[\mathbf{x}] = \frac{1}{2} \ln |\Sigma| + \frac{D}{2} (1 + \ln(2\pi)) \quad (3.204)$$

其中 D 是 \mathbf{x} 的维度。

3.10 (★★★) 考虑两个分别具有均值 μ_1 和 μ_2 以及精度 τ_1 和 τ_2 的高斯分布随机变量 x_1 和 x_2 。推导变量 $x = x_1 + x_2$ 的微分熵表达式。为此，首先利用关系式求出 x 的分布

$$p(x) = \int_{-\infty}^{\infty} p(x|x_2)p(x_2) dx_2 \quad (3.205)$$

然后在指数中完成平方。接着观察到这表示两个高斯分布的卷积，其本身将是高斯分布，最后利用单变量高斯分布熵的结果 (2.99)。

3.11 (*) 考虑由(3.26)给出的多元高斯分布。通过将精度矩阵（逆协方差矩阵）写成一个对称矩阵和一个反对称矩阵的和，证明反对称项不出现在高斯分布的指数中，因此，不失一般性，可以认为精度矩阵是对称的。由于对称矩阵的逆也是对称的（见练习3.16），所以不失一般性，也可以选择协方差矩阵为对称的。

3.12 (★★★) 考虑一个实对称矩阵 Σ ，其特征值方程由(3.28)给出。通过对该方程取复共轭，减去原方程，然后与特征向量 \mathbf{u}_i 形成内积，证明特征值 λ_i 是实数。类似地，利用 Σ 的对称性证明，只要 $\lambda_j \neq \lambda_i$ ，两个特征向量 \mathbf{u}_i 和 \mathbf{u}_j 将是正交的。最后，证明在不失一般性的情况下，可以选择特征向量集为正交归一的，使得它们满足(3.29)，即使某些特征值为零。

3.13 (★★) 证明一个具有特征向量方程(3.28)的实对称矩阵 Σ 可以表示为特征向量的展开式，其系数由特征值给出，形式为(3.31)。类似地，证明逆矩阵 Σ^{-1} 具有形式(3.32)的表示。

3.14 (★★) 正定矩阵 Σ 可以定义为对于向量 \mathbf{a} 的任何实数值，二次型

$$\mathbf{a}^T \Sigma \mathbf{a} \quad (3.206)$$

为正的矩阵。证明 Σ 为正定的充要条件是由(3.28)定义的 Σ 的所有特征值 λ_i 都是正的。

3.15 (*) 证明一个大小为 $D \times D$ 的实对称矩阵有 $D(D+1)/2$ 个独立参数。

3.16 (*) 证明对称矩阵的逆矩阵本身也是对称的。

3.17 (**) 通过使用特征向量展开式(3.31)对坐标系进行对角化，证明对于常数马氏距离 Δ 的超椭球体内包含的体积由下式给出

$$V_D |\Sigma|^{1/2} \Delta^D \quad (3.207)$$

其中 V_D 是 D 维单位球体的体积，马氏距离由(3.27)定义。

3.18 (**) 通过将等式两边乘以矩阵来证明恒等式(3.60)

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} \quad (3.208)$$

并利用定义(3.61)。

3.19 (***) 在3.2.4节和3.2.5节中，我们考虑了多元高斯分布的条件分布和边缘分布。更一般地，我们可以将 \mathbf{x} 的分量划分为三组 $\mathbf{x}_a, \mathbf{x}_b$ 、 \mathbf{x}_c ，同时将均值向量 μ 和协方差矩阵 Σ 进行相应划分，形式如下

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \\ \boldsymbol{\mu}_c \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} & \Sigma_{ac} \\ \Sigma_{ba} & \Sigma_{bb} & \Sigma_{bc} \\ \Sigma_{ca} & \Sigma_{cb} & \Sigma_{cc} \end{pmatrix}. \quad (3.209)$$

利用3.2节的结果，求出已将 \mathbf{x}_c 边缘化后的条件分布 $p(\mathbf{x}_a | \mathbf{x}_b)$ 的表达式。

3.20 (**) 线性代数中一个非常有用的结果是伍德伯里矩阵求逆公式，如下所示

$$(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1} + \mathbf{D}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{D}\mathbf{A}^{-1}. \quad (3.210)$$

通过两边同时乘以 $(\mathbf{A} + \mathbf{BCD})$ ，证明此结果的正确性。

3.21 (*) 设 \mathbf{x} 和 \mathbf{z} 为两个独立随机向量，使得 $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x})p(\mathbf{z})$ 。证明它们的和 $\mathbf{y} = \mathbf{x} + \mathbf{z}$ 的均值等于各个变量均值之和。类似地，证明 \mathbf{y} 的协方差矩阵等于 \mathbf{x} 和 \mathbf{z} 的协方差矩阵之和。

3.22 (**) 考虑变量上的联合分布

$$\mathbf{z} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \quad (3.211)$$

其均值和协方差分别由(3.92)和(3.89)给出。利用结果(3.76)和(3.77)，证明边际分布 $p(\mathbf{x})$ 由(3.83)给出。类似地，利用结果(3.65)和(3.66)，证明条件分布 $p(\mathbf{y} | \mathbf{x})$ 由(3.84)给出。

3.23 (**) 使用分块矩阵求逆公式(3.60)，证明精度矩阵(3.88)的逆由协方差矩阵(3.89)给出。

3.24 (**) 从(3.91)出发并利用结果(3.89)，验证结果(3.92)。

3.25 (**) 考虑两个分别具有高斯分布 $p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ 和 $p(\mathbf{z}) = \mathcal{N}(\mathbf{z} | \boldsymbol{\mu}_{\mathbf{z}}, \boldsymbol{\Sigma}_{\mathbf{z}})$ 的多维随机向量 \mathbf{x} 和 \mathbf{z} ，以及它们的和 $\mathbf{y} = \mathbf{x} + \mathbf{z}$ 。通过考虑由边际分布 $p(\mathbf{x})$ 和条件分布 $p(\mathbf{y} | \mathbf{x})$ 的乘积组成的线性高斯模型，并利用结果(3.93)和(3.94)，证明 $p(\mathbf{y})$ 的边际分布由下式给出

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y} | \boldsymbol{\mu}_{\mathbf{x}} + \boldsymbol{\mu}_{\mathbf{z}}, \boldsymbol{\Sigma}_{\mathbf{x}} + \boldsymbol{\Sigma}_{\mathbf{z}}). \quad (3.212)$$

3.26 (***) 本练习及下一个练习提供了处理线性高斯模型中出现的二次型的实践机会，并且它们还可作为对正文所推导结果的独立检验。考虑由(3.83)和(3.84)给出的边际分布和条件分布所定义的联合分布 $p(\mathbf{x}, \mathbf{y})$ 。通过检查联合分布指数中的二次型，并使用3.2节中讨论的“配方法”技术，求出已对变量 \mathbf{x} 进行积分的边际分布 $p(\mathbf{y})$ 的均值和协方差的表达式。为此，利用伍德伯里矩阵求逆公式(3.210)。验证这些结果与(3.93)和(3.94)一致。

3.27 (****) 考虑与练习3.26中相同的联合分布，但现在使用配方法来找到条件分布 $p(\mathbf{x} | \mathbf{y})$ 的均值和协方差的表达式。同样，验证这些表达式与相应的表达式(3.95)和(3.96)是否一致。

3.28 (**) 为了找到多元高斯协方差矩阵的最大似然解，我们需要相对于 $\boldsymbol{\Sigma}$ 最大化对数似然函数(3.102)，注意协方差矩阵必须是对称且正定的。这里我们通过忽略这些约束并直接进行最大化来继续。使用附录A中的结果(A.21)、(A.26)和(A.28)，证明最大化对数似然函数(3.102)的协方差矩阵 $\boldsymbol{\Sigma}$ 由样本协方差(3.106)给出。我们注意到最终结果必然是对称且正定的(前提是样本协方差是非奇异的)。

3.29 (**) 使用结果(3.42)证明(3.46)。现在，使用结果(3.42)和(3.46)，证明

$$\mathbb{E}[\mathbf{x}_n \mathbf{x}_m^T] = \boldsymbol{\mu} \boldsymbol{\mu}^T + I_{nm} \boldsymbol{\Sigma} \quad (3.213)$$

其中 \mathbf{x}_n 表示从均值为 $\boldsymbol{\mu}$ 且协方差为 $\boldsymbol{\Sigma}$ 的高斯分布中采样得到的数据点，并且 I_{nm} 表示单位矩阵的 (n, m) 元素。因此，证明结果(3.108)。

3.30 (*) 本章中讨论周期变量时使用的各种三角恒等式可以很容易地从该关系中证明

$$\exp(iA) = \cos A + i \sin A \quad (3.214)$$

其中 i 是负一的平方根。通过考虑恒等式

$$\exp(iA) \exp(-iA) = 1 \quad (3.215)$$

证明结果(3.127)。类似地，使用恒等式

$$\cos(A - B) = \Re \exp\{i(A - B)\} \quad (3.216)$$

其中 \Re 表示实部，证明(3.128)。最后，通过使用 $\sin(A - B) = \Im \exp\{i(A - B)\}$ ，其中 \Im 表示虚部，证明结果(3.133)。

3.31 (**) 对于较大的 m ，冯·米塞斯分布(3.129)在众数 θ_0 附近变得尖锐地 peaked。通过定义 $\xi = m^{1/2}(\theta - \theta_0)$ 并对由

$$\cos \alpha = 1 - \frac{\alpha^2}{2} + O(\alpha^4) \quad (3.217)$$

给出的余弦函数进行泰勒展开，表明当 $m \rightarrow \infty$ 时，冯·米塞斯分布趋于高斯分布。

3.32 (*) 利用三角恒等式(3.133)，证明对于 θ_0 ，(3.132)的解由(3.134)给出。

3.33 (*) 通过计算冯·米塞斯分布(3.129)的一阶和二阶导数，并对 $m > 0$ 使用 $I_0(m) > 0$ ，证明当 $\theta = \theta_0$ 时分布取得最大值，当 $\theta = \theta_0 + \pi \pmod{2\pi}$ 时取得最小值。

3.34 (*) 通过将结果(3.118)与(3.134)以及三角恒等式(3.128)结合使用，证明冯·米塞斯分布浓度的最大似然解 m_{ML} 满足 $A(m_{ML}) = \bar{r}$ ，其中 \bar{r} 是在二维欧几里得平面中被视为单位向量的观测值均值的半径，如图3.9所示。

3.35 (*) 验证多元高斯分布可以转换为指数族形式(3.138)，并推导与(3.164)至(3.167)类似的 $\eta, \mathbf{u}(\mathbf{x}), h(\mathbf{x})$ 、 $g(\eta)$ 的表达式。

3.36 (*) 结果 (3.172) 表明，指数族中 $\ln g(\eta)$ 的负梯度由 $\mathbf{u}(\mathbf{x})$ 的期望给出。通过对 (3.139) 求二阶导数，证明

$$-\nabla \nabla \ln g(\eta) = \mathbb{E}[\mathbf{u}(\mathbf{x})\mathbf{u}(\mathbf{x})^T] - \mathbb{E}[\mathbf{u}(\mathbf{x})]\mathbb{E}[\mathbf{u}(\mathbf{x})^T] = \text{cov}[\mathbf{u}(\mathbf{x})]. \quad (3.218)$$

3.37 (**) 考虑一种类似直方图的密度模型，其中空间 \mathbf{x} 被划分为固定区域，在第 i 个区域上，密度 $p(\mathbf{x})$ 取常数值 h_i 。区域 i 的体积记为 Δ_i 。假设我们有一组关于 \mathbf{x} 的 N 个观测值，使得这些观测值中有 n_i 个落在区域 i 中。使用拉格朗日乘数来强制密度的归一化约束，推导 $\{h_i\}$ 的最大似然估计量的表达式。

3.38 (*) 证明 K 最近邻密度模型定义了一个不合适的分布，其在所有空间上的积分是发散的。



4

Single-layer Networks: Regression

在本章中，我们使用线性回归框架来讨论神经网络背后的一些基本概念，我们在多项式曲线拟合的背景下简要遇到过这个框架。我们将看到，线性回归模型对应于具有单层可学习参数的简单形式的神经网络。虽然单层网络的实际适用性非常有限，但它们具有简单的分析特性，并为引入许多核心概念提供了一个极好的框架，这些概念将为我们在后续章节中对深度神经网络的讨论奠定基础。

4.1. 线性回归

回归的目标是在给定输入变量的 D 维向量 \mathbf{x} 的值的情况下，预测一个或多个连续目标变量 t 的值。通常，我们会得到一个包含 N 个观测值 $\{\mathbf{x}_n\}$ 的训练数据集，其中 $n = 1, \dots, N$ ，以及相应的目标值 $\{t_n\}$ ，目标是针对 \mathbf{x} 的新值预测 t 的值。为此，我们制定一个函数 $y(\mathbf{x}, \mathbf{w})$ ，其对于新输入 \mathbf{x} 的值构成对 t 相应值的预测，其中 \mathbf{w} 表示可以从训练数据中学习的参数向量。

回归的最简单模型是涉及输入变量线性组合的模型：

$$y(\mathbf{x}, \mathbf{w}) = w_0 + w_1 x_1 + \dots + w_D x_D \quad (4.1)$$

其中 $\mathbf{x} = (x_1, \dots, x_D)^T$ 。术语线性回归有时专门指这种模型形式。该模型的关键特性是它是参数 w_0, \dots, w_D 的线性函数。然而，它也是输入变量 x_i 的线性函数，这对模型施加了重大限制。

4.1.1 基函数

我们可以通过考虑输入变量的固定非线性函数的线性组合来扩展由(4.1)定义的模型类别，形式如下

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(\mathbf{x}) \quad (4.2)$$

其中 $\phi_j(\mathbf{x})$ 被称为基函数。通过用 $M - 1$ 表示索引 j 的最大值，此模型中的参数总数将为 M 。

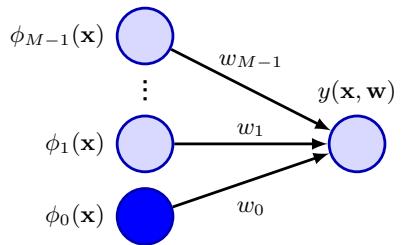
4.3节 参数 w_0 允许数据中有任何固定偏移，有时也称为偏差参数（不要与统计意义上的偏差混淆）。通常方便定义一个额外的虚拟基函数 $\phi_0(\mathbf{x})$ ，其值固定为 $\phi_0(\mathbf{x}) = 1$ ，这样 (4.2) 就变为

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) \quad (4.3)$$

其中 $\mathbf{w} = (w_0, \dots, w_{M-1})^T$ 和 $\phi = (\phi_0, \dots, \phi_{M-1})^T$ 。我们可以使用神经网络图来表示模型 (4.3)，如图4.1所示。

第6.1节 通过使用非线性基函数，我们允许函数 $y(\mathbf{x}, \mathbf{w})$ 成为输入向量 \mathbf{x} 的非线性函数。然而，形式为(4.2)的函数被称为线性模型，因为它们在 \mathbf{w} 中是线性的。正是参数中的这种线性将极大地简化这类模型的分析。然而，这也导致了一些重大限制。

图4.1 线性回归模型 (4.3) 可以表示为一个简单的神经网络图, 该图涉及单层参数。这里, 每个基函数 $\phi_j(\mathbf{x})$ 由一个输入节点表示, 实心节点表示“偏差”基函数 ϕ_0 , 函数 $y(\mathbf{x}, \mathbf{w})$ 由一个输出节点表示。每个参数 w_j 由一条将相应基函数连接到输出的线表示。



在深度学习出现之前, 机器学习中常用的做法是对输入变量 \mathbf{x} 进行某种形式的固定预处理, 也称为特征提取, 用一组基函数 $\{\phi_j(\mathbf{x})\}$ 来表示。目标是选择一组足够强大的基函数, 以便使用简单的网络模型解决由此产生的学习任务。不幸的是, 除了最简单的应用外, 很难手工设计出合适的基函数。深度学习通过从数据集中本身学习数据所需的非线性变换来避免这个问题。

第1章

当我们讨论使用多项式进行曲线拟合时, 我们已经遇到了一个回归问题的例子。如果我们考虑单个输入变量 x , 并且选择由 $\phi_j(x) = x^j$ 定义的基函数, 那么多项式函数(1.1)可以表示为(4.3)的形式。基函数还有许多其他可能的选择, 例如

$$\phi_j(x) = \exp\left\{-\frac{(x - \mu_j)^2}{2s^2}\right\} \quad (4.4)$$

其中 μ_j 控制输入空间中基函数的位置, 参数 s 控制其空间尺度。这些通常被称为“高斯”基函数, 不过需要注意的是, 它们并不一定需要有概率解释。特别地, 归一化系数并不重要, 因为这些基函数将与可学习参数 w_j 相乘。

另一种可能性是具有如下形式的Sigmoid基函数

$$\phi_j(x) = \sigma\left(\frac{x - \mu_j}{s}\right) \quad (4.5)$$

其中 $\sigma(a)$ 是由下式定义的逻辑Sigmoid函数

$$\sigma(a) = \frac{1}{1 + \exp(-a)}. \quad (4.6)$$

练习4.3

等效地, 我们可以使用tanh函数, 因为它通过 $\tanh(a) = 2\sigma(2a) - 1$ 与逻辑Sigmoid函数相关, 所以逻辑Sigmoid函数的一般线性组合在能够表示相同类别的输入 - 输出函数的意义上等同于tanh函数的一般线性组合。图4.2展示了这些基函数的不同选择。

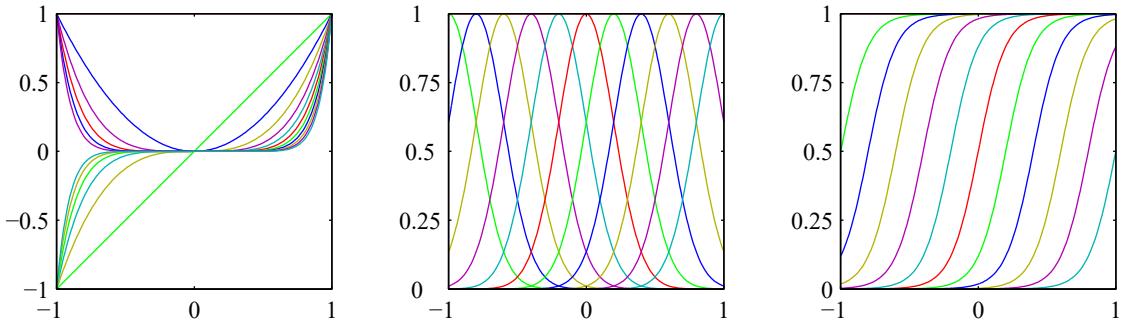


图4.2基函数示例，左侧为多项式，中间为形如(4.4)的高斯函数，右侧为形如(4.5)的Sigmoid基函数。

基函数的另一种可能选择是傅里叶基，它会导致正弦函数展开。每个基函数代表一个特定频率且具有无限空间范围。相比之下，局限于输入空间有限区域的基函数必然包含不同空间频率的频谱。在信号处理应用中，通常有兴趣考虑在空间和频率上都局域化的基函数，从而产生一类称为小波的函数(奥格登，1997；马拉特，1999；维达科维奇，1999)。为了简化应用，它们也被定义为相互正交。当输入值位于规则格点上时，小波最为适用，例如时间序列中的连续时间点或图像中的像素。

不过，本章中的大部分讨论与基函数集的选择无关，因此，除了数值示例外，我们不会指定基函数的具体形式。此外，为了使符号简单，我们将重点关注单个目标变量 t 的情况，尽管我们将简要概述处理多个目标变量所需的修改。

第4.1.7节

第1.2节

我们通过最小化平方和误差函数解决了将多项式函数拟合到数据的问题，并且我们还表明，在假设的高斯噪声模型下，这个误差函数可以作为最大似然解。现在我们回到这个讨论，更详细地考虑最小二乘法及其与最大似然的关系。

和以前一样，我们假设目标变量 t 由具有加性高斯噪声的确定性函数 $y(\mathbf{x}, \mathbf{w})$ 给出，因此

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \quad (4.7)$$

其中 ϵ 是一个均值为零、方差为 σ^2 的高斯随机变量。因此，我们可以写成

$$p(t|\mathbf{x}, \mathbf{w}, \sigma^2) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \sigma^2). \quad (4.8)$$

现在考虑一个输入数据集 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ ，其具有相应的目标值 t_1, \dots, t_N 。我们将目标变量 $\{t_n\}$ 分组为一个列向量，我们用 \mathbf{t} 表示它，这里选择这种字体是为了将它与多变量目标的单个观测值区分开来，多变量目标的单个观测值将表示为 \mathbf{t} 。假设这些数据点是从分布(4.8)中独立抽取的，我们得到了似然函数的一个表达式，它是可调参数 \mathbf{w} 和 σ^2 的函数：

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma^2) = \prod_{n=1}^N \mathcal{N}(t_n | \mathbf{w}^T \phi(\mathbf{x}_n), \sigma^2) \quad (4.9)$$

这里我们使用了(4.3)。对似然函数取对数，并利用单变量高斯的标准形式(2.49)，我们有

$$\begin{aligned} \ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma^2) &= \sum_{n=1}^N \ln \mathcal{N}(t_n | \mathbf{w}^T \phi(\mathbf{x}_n), \sigma^2) \\ &= -\frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi) - \frac{1}{\sigma^2} E_D(\mathbf{w}) \end{aligned} \quad (4.10)$$

其中平方和误差函数由下式定义

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{t_n - \mathbf{w}^T \phi(\mathbf{x}_n)\}^2. \quad (4.11)$$

第2.3.4节

在确定 \mathbf{w} 时，(4.10)中的前两项可视为常数，因为它们与 \mathbf{w} 无关。因此，正如我们之前所见，在高斯噪声分布下最大化似然函数等同于最小化平方和误差函数(4.11)。

4.1.3 最大似然

写下似然函数后，我们可以使用最大似然法来确定 \mathbf{w} 和 σ^2 。首先考虑关于 \mathbf{w} 的最大化。对数似然函数(4.10)关于 \mathbf{w} 的梯度具有以下形式

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma^2) = \frac{1}{\sigma^2} \sum_{n=1}^N \{t_n - \mathbf{w}^T \phi(\mathbf{x}_n)\} \phi(\mathbf{x}_n)^T. \quad (4.12)$$

将此梯度设为零可得

$$0 = \sum_{n=1}^N t_n \phi(\mathbf{x}_n)^T - \mathbf{w}^T \left(\sum_{n=1}^N \phi(\mathbf{x}_n) \phi(\mathbf{x}_n)^T \right). \quad (4.13)$$

求解 \mathbf{w} ，我们得到

$$\mathbf{w}_{ML} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t}, \quad (4.14)$$

它们被称为最小二乘问题的正规方程。这里 Φ 是一个 $N \times M$ 矩阵，称为设计矩阵，其元素由 $\Phi_{nj} = \phi_j(\mathbf{x}_n)$ 给出，因此

$$\Phi = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}. \quad (4.15)$$

量

$$\Phi^\dagger \equiv (\Phi^T \Phi)^{-1} \Phi^T \quad (4.16)$$

被称为矩阵 Φ 的摩尔 - 彭罗斯伪逆 (Rao 和 Mitra, 1971; Golub 和 Van Loan, 1996)。它可以被视为矩阵逆的概念对非方阵的推广。实际上，如果 Φ 是方阵且可逆，那么利用性质 $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ 我们可以看到 $\Phi^\dagger \equiv \Phi^{-1}$ 。

此时，我们可以深入了解偏差参数 w_0 的作用。如果我们明确偏差参数，那么误差函数 (4.11) 就会变成

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{t_n - w_0 - \sum_{j=1}^{M-1} w_j \phi_j(\mathbf{x}_n)\}^2. \quad (4.17)$$

将关于 w_0 的导数设为零并求解 w_0 ，我们得到

$$w_0 = \bar{t} - \sum_{j=1}^{M-1} w_j \overline{\phi_j} \quad (4.18)$$

这里我们定义了

$$\bar{t} = \frac{1}{N} \sum_{n=1}^N t_n, \quad \overline{\phi_j} = \frac{1}{N} \sum_{n=1}^N \phi_j(\mathbf{x}_n). \quad (4.19)$$

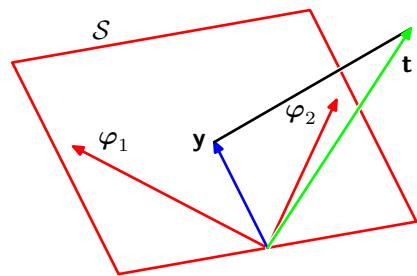
因此，偏差 w_0 补偿了目标值的平均值（在训练集上）与基函数值的平均值的加权和之间的差异。

我们还可以关于方差 σ^2 最大化对数似然函数 (4.10)，得到

$$\sigma_{ML}^2 = \frac{1}{N} \sum_{n=1}^N \{t_n - \mathbf{w}_{ML}^T \phi(\mathbf{x}_n)\}^2, \quad (4.20)$$

这样我们就看到方差参数的最大似然值由目标值围绕回归函数的残差方差给出。

图4.3 在一个以 t_1, \dots, t_N 的值为轴的 N 维空间中最小二乘解的几何解释。通过找到数据向量 t 在由基函数 $\phi_j(\mathbf{x})$ 所张成的子空间上的正交投影来获得最小二乘回归函数，其中每个基函数被视为一个长度为 N 且元素为 $\phi_j(\mathbf{x}_n)$ 的向量 φ_j 。



4.1.4 最小二乘法的几何原理

此时，考虑最小二乘解的几何解释是很有启发性的。为此，我们考虑一个 N 维空间，其坐标轴由 t_n 给出，使得 $\mathbf{t} = (t_1, \dots, t_N)^T$ 是这个空间中的一个向量。每个基函数 $\phi_j(\mathbf{x}_n)$ ，在 N 个数据点处求值，也可以表示为同一空间中的一个向量，用 φ_j 表示，如图4.3所示。注意， φ_j 对应于 Φ 的第 j 列，而 $\phi(\mathbf{x}_n)$ 对应于 Φ 的第 n 行的转置。如果基函数的数量 M 小于数据点的数量 N ，那么 M 个向量 $\phi_j(\mathbf{x}_n)$ 将跨越一个维度为 M 的线性子空间 S 。我们定义 \mathbf{y} 为一个 N 维向量，其第 n 个元素由 $y(\mathbf{x}_n, \mathbf{w})$ 给出，其中 $n = 1, \dots, N$ 。因为 \mathbf{y} 是向量 φ_j 的任意线性组合，它可以位于 M 维子空间的任何位置。平方和误差(4.11)（直到一个 $1/2$ 因子）然后等于 \mathbf{y} 和 \mathbf{t} 之间的欧几里得距离的平方。因此， \mathbf{w} 的最小二乘解对应于位于子空间 S 中且最接近 \mathbf{t} 的 \mathbf{y} 的选择。直观地，从图4.3中，我们预计这个解对应于 \mathbf{t} 在子空间 S 上的正交投影。确实如此，通过注意到 \mathbf{y} 的解由 $\Phi \mathbf{w}_{ML}$ 给出，然后确认这采取正交投影的形式，可以很容易地验证这一点。

练习4.4

在实际应用中，当 $\Phi^T \Phi$ 接近奇异时，直接求解正规方程可能会导致数值困难。特别是，当两个或更多基向量 φ_j 共线或几乎共线时，所得参数值可能会有很大的幅度。在处理实际数据集时，这种近似退化情况并不罕见。可以使用奇异值分解(SVD)技术(Deisenroth、Faisal和Ong, 2020)来解决由此产生的数值困难。请注意，添加正则化项可确保矩阵非奇异，即使存在退化情况。

4.1.5 顺序学习

最大似然解(4.14)需要一次性处理整个训练集，这被称为批处理方法。对于大数据集来说，这在计算上可能成本很高。如果数据集足够大，使用顺序算法(也称为在线算法)可能是值得的，在这种算法中，数据点一次处理一个，并且在每次呈现后更新模型参数。顺序学习也适用于实时应用，在这些应用中，数据观测以连续流的形式到达，并且必须在看到所有数据点之前进行预测。

第7章

我们可以通过应用随机梯度下降技术（也称为顺序梯度下降）来获得一种顺序学习算法，如下所示。如果误差函数包含对数据点 $E = \sum_n E_n$ 的求和，那么在呈现数据点 n 之后，随机梯度下降算法使用以下公式更新参数向量 $\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_n$ (4.21)

其中 τ 表示迭代次数， η 是一个适当选择的学习率参数。 \mathbf{w} 的值初始化为某个起始向量 $\mathbf{w}^{(0)}$ 。对于平方和误差函数 (4.11)，这给出了以下公式。

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \eta(t_n - \mathbf{w}^{(\tau)T} \phi_n) \phi_n \quad (4.22)$$

其中 $\phi_n = \phi(\mathbf{x}_n)$ 。这被称为最小均方或LMS算法。

4.1.6 正则化最小二乘法

第1.2节

我们之前已经介绍了在误差函数中添加正则化项以控制过拟合的想法，这样要最小化的总误差函数具有以下形式。

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w}) \quad (4.23)$$

其中 λ 是正则化系数，用于控制数据相关误差 $E_D(\mathbf{w})$ 和正则化项 $E_W(\mathbf{w})$ 的相对重要性。最简单的正则化形式之一由权重向量元素的平方和给出：

$$E_W(\mathbf{w}) = \frac{1}{2} \sum_j w_j^2 = \frac{1}{2} \mathbf{w}^T \mathbf{w}. \quad (4.24)$$

如果我们还考虑由下式给出的平方和误差函数

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{t_n - \mathbf{w}^T \phi(\mathbf{x}_n)\}^2, \quad (4.25)$$

那么总误差函数变为

$$\frac{1}{2} \sum_{n=1}^N \{t_n - \mathbf{w}^T \phi(\mathbf{x}_n)\}^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}. \quad (4.26)$$

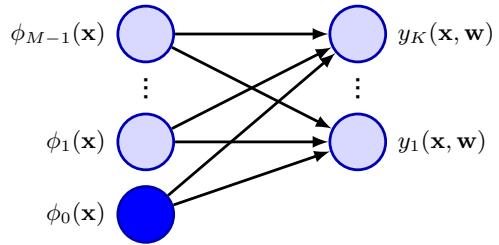
在统计学中，这种正则化器提供了一种参数收缩方法的示例，因为它将参数值向零收缩。它的优点是误差函数仍然是 \mathbf{w} 的二次函数，因此可以以封闭形式找到其精确的最小值。具体来说，将(4.26)关于 \mathbf{w} 的梯度设为零，并像之前一样求解 \mathbf{w} ，我们得到

$$\mathbf{w} = (\lambda \mathbf{I} + \Phi^T \Phi)^{-1} \Phi^T \mathbf{t}. \quad (4.27)$$

这表示最小二乘解(4.14)的简单扩展。

练习4.6

图4.4 作为具有单层连接的神经网络的线性回归模型的表示。每个基函数由一个节点表示，实心节点表示“偏差”基函数 ϕ_0 。同样，每个输出 y_1, \dots, y_K 由一个节点表示。节点之间的链接表示相应的权重和偏差参数。



4.1.7 多个输出

到目前为止，我们考虑的是只有一个目标变量 t 的情况。在某些应用中，我们可能希望预测 $K > 1$ 个目标变量，我们将它们统称为目标向量 $\mathbf{t} = (t_1, \dots, t_K)^T$ 。这可以通过为 \mathbf{t} 的每个分量引入一组不同的基函数来实现，从而导致多个独立的回归问题。然而，一种更常见的方法是使用同一组基函数来对目标向量的所有分量进行建模，以便

$$\mathbf{y}(\mathbf{x}, \mathbf{w}) = \mathbf{W}^T \phi(\mathbf{x}) \quad (4.28)$$

其中 \mathbf{y} 是一个 K 维列向量， \mathbf{W} 是一个参数 $M \times K$ 矩阵，并且 $\phi(\mathbf{x})$ 是一个 M 维列向量，其元素为 $\phi_j(\mathbf{x})$ ，和之前一样有 $\phi_0(\mathbf{x}) = 1$ 。同样，这可以表示为一个具有单层参数的神经网络，如图4.4所示。

假设我们将目标向量的条件分布设为如下形式的各向同性高斯分布

$$p(\mathbf{t}|\mathbf{x}, \mathbf{W}, \sigma^2) = \mathcal{N}(\mathbf{t}|\mathbf{W}^T \phi(\mathbf{x}), \sigma^2 \mathbf{I}). \quad (4.29)$$

如果我们有一组观测值 $\mathbf{t}_1, \dots, \mathbf{t}_N$ ，我们可以将它们组合成一个大小为 $N \times K$ 的矩阵 \mathbf{T} ，使得第 n 行由 \mathbf{t}_n^T 给出。类似地，我们可以将输入向量 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 组合成一个矩阵 \mathbf{X} 。然后对数似然函数由下式给出

$$\begin{aligned} \ln p(\mathbf{T}|\mathbf{X}, \mathbf{W}, \sigma^2) &= \sum_{n=1}^N \ln \mathcal{N}(\mathbf{t}_n | \mathbf{W}^T \phi(\mathbf{x}_n), \sigma^2 \mathbf{I}) \\ &= -\frac{NK}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{n=1}^N \|\mathbf{t}_n - \mathbf{W}^T \phi(\mathbf{x}_n)\|^2. \end{aligned} \quad (4.30)$$

和之前一样，我们可以关于 \mathbf{W} 最大化这个函数，得到

$$\mathbf{W}_{ML} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{T} \quad (4.31)$$

其中我们已将输入特征向量 $\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_N)$ 组合成一个矩阵 Φ 。如果我们针对每个目标变量 t_k 检查这个结果，我们有

$$\mathbf{w}_k = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t}_k = \Phi^\dagger \mathbf{t}_k \quad (4.32)$$

其中 \mathbf{t}_k 是一个 N 维列向量，其分量为 t_{nk} ， $n = 1, \dots, N$ 。因此，回归问题的解在不同目标变量之间解耦，我们只需要计算一个单一的伪逆矩阵 Φ^\dagger ，它由所有向量 \mathbf{w}_k 共享。

练习4.7

将其扩展到具有任意协方差矩阵的一般高斯噪声分布很简单。同样，这会导致解耦为 K 个独立的回归问题。这个结果并不奇怪，因为参数 \mathbf{W} 仅定义高斯噪声分布的均值，并且我们知道多元高斯均值的最大似然解与协方差无关。因此，从现在起，为了简单起见，我们将考虑单个目标变量 t 。

第3.2.7节

4.2. 决策理论

我们已将回归任务表述为对条件概率分布 $p(t | \mathbf{x})$ 进行建模的任务之一，并且我们为条件概率选择了一种特定形式，即具有由参数 \mathbf{w} 控制的、依赖于 \mathbf{x} 的均值 $y(\mathbf{x}, \mathbf{w})$ 的高斯分布(4.8)，其方差由参数 σ^2 给出。 \mathbf{w} 和 σ^2 都可以使用最大似然法从数据中学习得到。结果是由以下给出的预测分布

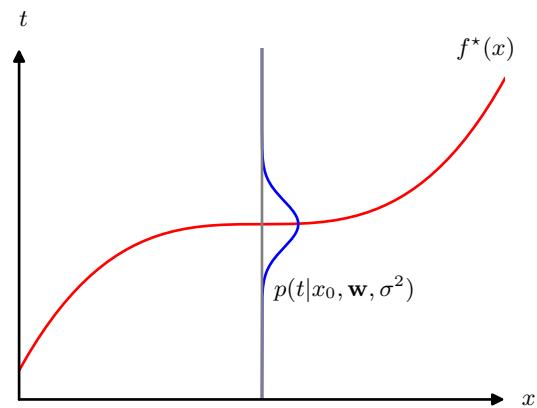
$$p(t | \mathbf{x}, \mathbf{w}_{\text{ML}}, \sigma_{\text{ML}}^2) = \mathcal{N}(t | y(\mathbf{x}, \mathbf{w}_{\text{ML}}), \sigma_{\text{ML}}^2). \quad (4.33)$$

预测分布表达了我们对于某个新输入 \mathbf{x} 的 t 值的不确定性。然而，对于许多实际应用，我们需要预测 t 的一个特定值，而不是返回整个分布，特别是在我们必须采取特定行动的情况下。例如，如果我们的目标是确定用于治疗肿瘤的最佳辐射水平，并且我们的模型预测了辐射剂量的概率分布，那么我们必须使用该分布来决定要施用的特定剂量。因此，我们的任务分为两个阶段。在第一阶段，称为推理阶段，我们使用训练数据来确定预测分布 $p(t | \mathbf{x})$ 。在第二阶段，称为决策阶段，我们使用这个预测分布来确定一个特定值 $f(\mathbf{x})$ ，它将取决于输入向量 \mathbf{x} ，并且根据某个标准是最优的。我们可以通过最小化一个依赖于预测分布 $p(t | \mathbf{x})$ 和 f 的损失函数来做到这一点。

直观地说，我们可能会选择条件分布的均值，这样我们就会使用 $f(\mathbf{x}) = y(\mathbf{x}, \mathbf{w}_{\text{ML}})$ 。在某些情况下，这种直觉是正确的，但在其他情况下，它可能会给出非常糟糕的结果。因此，将其形式化是很有用的，这样我们就可以理解它何时适用以及在什么假设下适用，而这样做的框架就叫做决策理论。

假设当真实值为 t 时，我们为预测选择一个值 $f(\mathbf{x})$ 。这样做时，我们会产生某种形式的惩罚或成本。这由一个损失来决定，我们将其表示为 $L(t, f(\mathbf{x}))$ 。当然，我们不知道 t 的真实值，所以我们不是最小化 L 本身，而是最小化平均损失或期望损失，即

图4.5 使期望平方损失最小化的回归函数 $f^*(x)$ ，
由条件分布 $p(t|x)$ 的均值给出。



由.....给出

$$\mathbb{E}[L] = \iint L(t, f(\mathbf{x})) p(\mathbf{x}, t) d\mathbf{x} dt \quad (4.34)$$

其中我们对输入变量和目标变量的分布进行平均，并由它们的联合分布 $p(\mathbf{x}, t)$ 加权。回归问题中损失函数的一个常见选择是由 $L(t, f(\mathbf{x})) = \{f(\mathbf{x}) - t\}^2$ 给出的平方损失。在这种情况下，期望损失可以写成

$$\mathbb{E}[L] = \iint \{f(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt. \quad (4.35)$$

重要的是不要将平方损失函数与前面介绍的平方误差和函数混淆。误差函数用于在训练期间设置参数，以确定条件概率分布 $p(t|\mathbf{x})$ ，而损失函数则控制如何使用条件分布来得出预测函数 $f(\mathbf{x})$ ，该预测函数为 \mathbf{x} 的每个值指定一个预测。

附录B

我们的目标是选择 $f(\mathbf{x})$ ，以便最小化 $\mathbb{E}[L]$ 。如果我们假设一个完全灵活的函数 $f(\mathbf{x})$ ，我们可以使用变分法正式地做到这一点，得到

$$\frac{\delta \mathbb{E}[L]}{\delta f(\mathbf{x})} = 2 \int \{f(\mathbf{x}) - t\} p(\mathbf{x}, t) dt = 0. \quad (4.36)$$

求解 $f(\mathbf{x})$ 并使用概率的和与积规则，我们得到

$$f^*(\mathbf{x}) = \frac{1}{p(\mathbf{x})} \int t p(\mathbf{x}, t) dt = \int t p(t|\mathbf{x}) dt = \mathbb{E}_t[t|\mathbf{x}], \quad (4.37)$$

它是在 \mathbf{x} 条件下 t 的条件均值，被称为回归函数。此结果如图4.5所示。它可以很容易地扩展到由向量 \mathbf{t} 表示的多个目标变量，在这种情况下，最优解是条件均值 $\mathbf{f}^*(\mathbf{x}) = \mathbb{E}_t[\mathbf{t}|\mathbf{x}]$ 。对于形式为(4.8)的高斯条件分布，条件均值将简单地为

练习4.8

$$\mathbb{E}[t|\mathbf{x}] = \int tp(t|\mathbf{x}) dt = y(\mathbf{x}, \mathbf{w}). \quad (4.38)$$

使用变分法来推导(4.37)意味着我们正在对所有可能的函数 $f(\mathbf{x})$ 进行优化。尽管我们在实践中能够实现的任何参数模型在其能够表示的函数范围上是有限的，但在后面章节中广泛讨论的深度神经网络框架提供了一类高度灵活的函数，对于许多实际目的而言，这些函数能够以高精度逼近任何期望的函数。

我们可以用一种稍有不同的方式来推导这个结果，这也将有助于阐明回归问题的本质。基于最优解是条件期望这一知识，我们可以如下展开平方项

$$\begin{aligned} \{f(\mathbf{x}) - t\}^2 &= \{f(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}] + \mathbb{E}[t|\mathbf{x}] - t\}^2 \\ &= \{f(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}^2 + 2\{f(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}\{\mathbb{E}[t|\mathbf{x}] - t\} + \{\mathbb{E}[t|\mathbf{x}] - t\}^2 \end{aligned}$$

为使符号简洁，我们用 $\mathbb{E}[t|\mathbf{x}]$ 表示 $\mathbb{E}_t[t|\mathbf{x}]$ 。将其代入损失函数(4.35)并对 t 进行积分，我们发现交叉项消失了，从而得到损失函数的如下形式的表达式

$$\mathbb{E}[L] = \int \{f(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\}^2 p(\mathbf{x}) d\mathbf{x} + \int \text{var}[t|\mathbf{x}] p(\mathbf{x}) d\mathbf{x}. \quad (4.39)$$

我们试图确定的函数 $f(\mathbf{x})$ 仅出现在第一项中，当 $f(\mathbf{x})$ 等于 $\mathbb{E}[t|\mathbf{x}]$ 时该项将被最小化，在这种情况下该项将消失。这正是我们之前推导的结果，表明最优最小二乘预测器由条件均值给出。第二项是 t 分布的方差，在 \mathbf{x} 上进行平均，代表目标数据的内在变异性，可视为噪声。由于它与 $f(\mathbf{x})$ 无关，它代表了损失函数的不可约最小值。

平方损失不是回归中损失函数的唯一可能选择。这里我们简要考虑平方损失的一种简单推广，称为闵可夫斯基损失，其期望由下式给出

$$\mathbb{E}[L_q] = \iint |f(\mathbf{x}) - t|^q p(\mathbf{x}, t) d\mathbf{x} dt, \quad (4.40)$$

这对于 $q = 2$ 可简化为预期的平方损失。在图4.6中，针对 q 的各种值绘制了函数 $|f - t|^q$ 与 $f - t$ 的关系曲线。 $\mathbb{E}[L_q]$ 的最小值由 $q = 2$ 的条件均值、 $q = 1$ 的条件中位数以及 $q \rightarrow 0$ 的条件众数给出。

练习4.12

请注意，高斯噪声假设意味着给定 \mathbf{x} 时 t 的条件分布是单峰的，这在某些应用中可能不合适。在这种情况下，平方损失可能会导致非常糟糕的结果，我们需要开发更复杂的方法。

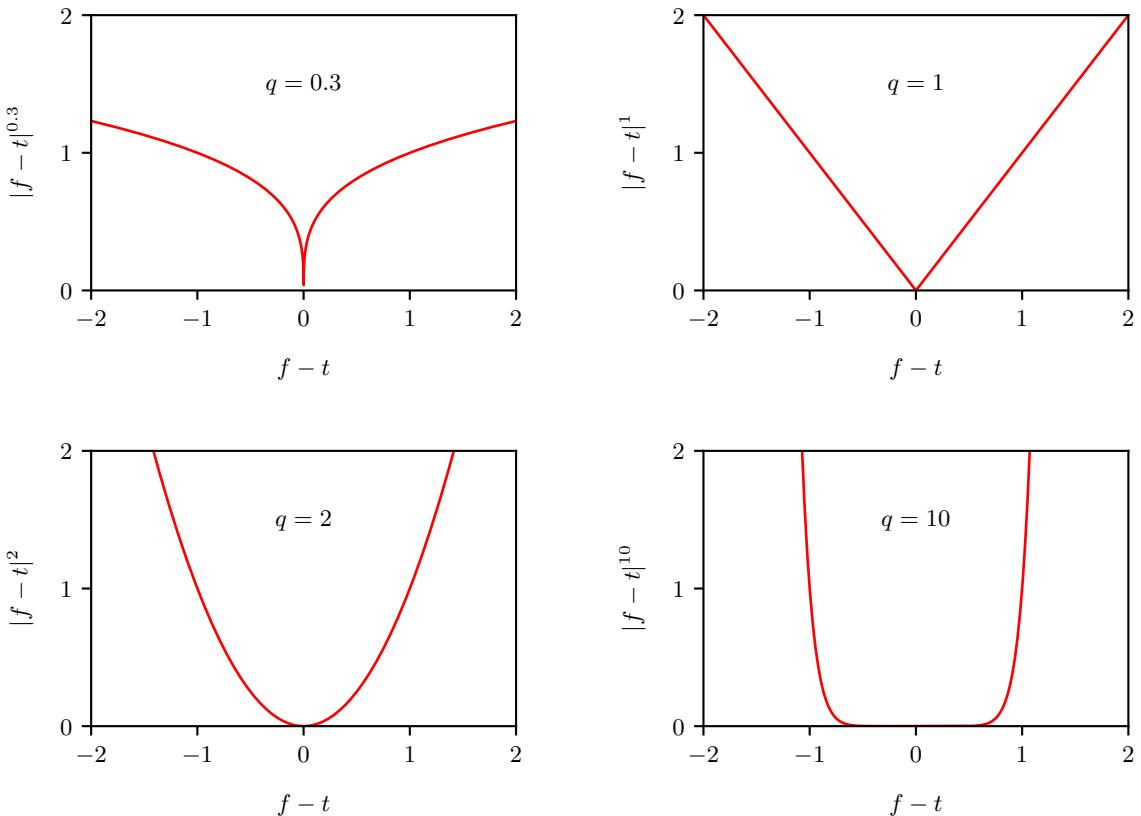


图4.6 针对 q 的不同值绘制的 $L_q = |f - t|^q$ 数量图。

第6.5节

例如，我们可以通过使用高斯混合来扩展此模型，以给出多峰条件分布，这在逆问题的解决方案中经常出现。我们在本节中的重点是回归问题的决策理论，在下一章中我们将为分类任务开发类似的概念。

第5.2节

4.3. 偏差-方差权衡

第1.2节

到目前为止，在我们对回归线性模型的讨论中，我们假设基函数的形式和数量都是给定的。我们还看到，如果使用有限大小的数据集训练复杂模型，使用最大似然法可能会导致严重的过拟合。然而，限制基函数的数量以避免过拟合会产生副作用，即限制模型捕捉数据中有趣和重要趋势的灵活性。虽然正则化项可以控制具有许多参数的模型的过拟合，但这就引出了如何确定正则化系数 λ 的合适值的问题。

寻求在权重向量 \mathbf{w} 和正则化系数 λ 两者上使正则化误差函数最小化的解决方案显然不是正确的方法，因为这会导致带有 $\lambda = 0$ 的无正则化解决方案。

从频率主义者的角度考虑模型复杂度问题（即偏差-方差权衡）是很有启发性的。尽管我们将在易于用简单示例说明这些想法的线性基函数模型的背景下引入这个概念，但该讨论具有非常广泛的适用性。然而请注意，过拟合实际上是最似然的一个不幸特性，当我们在贝叶斯设置中对参数进行边缘化时不会出现（Bishop, 2006）。

4.2节

当我们讨论回归问题的决策理论时，我们考虑了各种损失函数，一旦给定条件分布 $p(t|\mathbf{x})$ ，每个损失函数都会导致相应的最优预测。一个常见的选择是平方损失函数，其最优预测由条件期望给出，我们将其记为 $h(\mathbf{x})$ ，表达式为

$$h(\mathbf{x}) = \mathbb{E}[t|\mathbf{x}] = \int tp(t|\mathbf{x}) dt. \quad (4.41)$$

我们还看到，期望平方损失可以写成以下形式

$$\mathbb{E}[L] = \int \{f(\mathbf{x}) - h(\mathbf{x})\}^2 p(\mathbf{x}) d\mathbf{x} + \iint \{h(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt. \quad (4.42)$$

回想一下，第二项与 $f(\mathbf{x})$ 无关，它源于数据的内在噪声，代表了预期损失的最小可实现值。第一项取决于我们对函数 $f(\mathbf{x})$ 的选择，我们将寻求使该项最小的 $f(\mathbf{x})$ 的解。由于它是非负的，我们希望该项能达到的最小值是零。如果我们有无限的数据供应（以及无限的计算资源），原则上我们可以找到任意精度的回归函数 $h(\mathbf{x})$ ，这将代表 $f(\mathbf{x})$ 的最优选择。然而，在实践中，我们有一个数据集 \mathcal{D} ，它只包含有限数量 N 的数据点，因此，我们无法确切知道回归函数 $h(\mathbf{x})$ 。

如果我们要用一个由参数向量 \mathbf{w} 控制的函数来对 $h(\mathbf{x})$ 进行建模，那么从贝叶斯的角度来看，我们模型中的不确定性将通过关于 \mathbf{w} 的后验分布来表示。然而，频率主义的处理方法是基于数据集 \mathcal{D} 对 \mathbf{w} 进行点估计，而是试图通过以下思想实验来解释这个估计的不确定性。假设我们有大量大小为 N 的数据集，每个数据集都是从分布 $p(t, \mathbf{x})$ 中独立抽取的。对于任何给定的数据集 \mathcal{D} ，我们可以运行我们的学习算法并获得一个预测函数 $f(\mathbf{x}; \mathcal{D})$ 。来自该集合的不同数据集将给出不同的函数，从而导致平方损失的不同值。然后通过对这个数据集集合求平均值来评估特定学习算法的性能。

考虑(4.42)中第一项的被积函数，对于特定数据集 \mathcal{D} ，其形式为

$$\{f(\mathbf{x}; \mathcal{D}) - h(\mathbf{x})\}^2. \quad (4.43)$$

由于这个量将取决于特定的数据集 \mathcal{D} ，我们对数据集的总体求其平均值。如果我们在括号内加上和减去量 $\mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})]$ ，然后展开，我们得到

$$\begin{aligned} & \{f(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] + \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}^2 \\ &= \{f(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})]\}^2 + \{\mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}^2 \\ &+ 2\{f(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})]\}\{\mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}. \end{aligned} \quad (4.44)$$

我们现在对这个表达式关于 \mathcal{D} 求期望，并注意到最后一项将消失，得到

$$\begin{aligned} & \mathbb{E}_{\mathcal{D}} [\{f(\mathbf{x}; \mathcal{D}) - h(\mathbf{x})\}^2] \\ &= \underbrace{\{\mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}^2}_{\text{(bias)}^2} + \underbrace{\mathbb{E}_{\mathcal{D}} [\{f(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})]\}^2]}_{\text{variance}}. \end{aligned} \quad (4.45)$$

我们看到， $f(\mathbf{x}; \mathcal{D})$ 与回归函数 $h(\mathbf{x})$ 之间的期望平方差可以表示为两项之和。第一项称为平方偏差，它表示所有数据集上的平均预测与期望回归函数的差异程度。第二项称为方差，它衡量各个数据集的解围绕其平均值的变化程度，因此，这衡量了函数 $f(\mathbf{x}; \mathcal{D})$ 对特定数据集选择的敏感程度。稍后，当我们考虑一个简单示例时，我们将提供一些直观理解来支持这些定义。

到目前为止，我们考虑了单个输入值 \mathbf{x} 。如果我们将此展开式代回 (4.42)，我们得到期望平方损失的以下分解：

$$\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise} \quad (4.46)$$

其中

$$(\text{bias})^2 = \int \{\mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\}^2 p(\mathbf{x}) d\mathbf{x} \quad (4.47)$$

$$\text{variance} = \int \mathbb{E}_{\mathcal{D}} [\{f(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[f(\mathbf{x}; \mathcal{D})]\}^2] p(\mathbf{x}) d\mathbf{x} \quad (4.48)$$

$$\text{noise} = \iint \{h(\mathbf{x}) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt \quad (4.49)$$

并且偏差项和方差项现在指的是积分量。

我们的目标是将预期损失最小化，我们已将其分解为（平方）偏差、方差和一个常数噪声项的总和。正如我们将看到的，偏差和方差之间存在权衡，非常灵活的模型偏差低、方差高，而相对刚性的模型偏差高、方差低。具有最佳预测能力的模型是在偏差和方差之间实现最佳平衡的模型。通过考虑前面介绍的正弦数据集可以说明这一点。在这里，我们从正弦曲线 $h(x) = \sin(2\pi x)$ 独立生成 100 个数据集，每个数据集包含 $N = 25$ 个数据点。

数据集由 $l = 1, \dots, L$ 索引，其中 $L = 100$ 。对于每个数据集 $\mathcal{D}^{(l)}$ ，我们用 $M = 24$ 个高斯基函数以及一个常数“偏差”基函数拟合一个模型，总共得到 25 个参数。通过最小化正则化误差函数(4.26)，我们得到一个预测函数 $f^{(l)}(x)$ ，如图4.7所示。

最上面一行对应正则化系数 λ 的一个大值，该值导致低方差（因为左图中的红色曲线看起来相似）但高偏差（因为右图中的两条曲线非常不同）。相反，在最下面一行， λ 较小，存在大方差（由左图中红色曲线之间的高变异性显示）但低偏差（由平均模型拟合与原始正弦函数之间的良好拟合显示）。请注意，对具有 $M = 25$ 的复杂模型的许多解进行平均的结果是对回归函数的非常好的拟合，这表明平均可能是一个有益的过程。实际上，多个解的加权平均是贝叶斯方法的核心，尽管平均是相对于参数的后验分布，而不是相对于多个数据集。

我们还可以针对此示例定量地研究偏差 - 方差权衡。平均预测是根据

$$\bar{f}(x) = \frac{1}{L} \sum_{l=1}^L f^{(l)}(x), \quad (4.50)$$

进行估计的，然后积分平方偏差和积分方差由

$$(\text{bias})^2 = \frac{1}{N} \sum_{n=1}^N \{\bar{f}(x_n) - h(x_n)\}^2 \quad (4.51)$$

$$\text{variance} = \frac{1}{N} \sum_{n=1}^N \frac{1}{L} \sum_{l=1}^L \{f^{(l)}(x_n) - \bar{f}(x_n)\}^2 \quad (4.52)$$

其中，在由分布 $p(x)$ 加权的 x 上的积分，通过对从该分布中抽取的数据点进行有限求和来近似。这些量及其和在图4.8中作为 $\ln \lambda$ 的函数绘制。我们看到， λ 的小值允许模型针对每个单独数据集上的噪声进行精细调整，从而导致大的方差。相反， λ 的大值将权重参数拉向零，导致大的偏差。

请注意，偏差-方差分解的实际价值有限，因为它基于数据集集合的平均值，而在实际中我们只有单个观察到的数据集。如果我们有大量给定大小的独立训练集，我们最好将它们组合成一个更大的训练集，这当然会降低给定模型复杂度下的过拟合程度。尽管如此，偏差-方差分解通常能为模型复杂度问题提供有用的见解，并且虽然我们在本章从回归问题的角度引入了它，但基本直觉具有广泛的适用性。

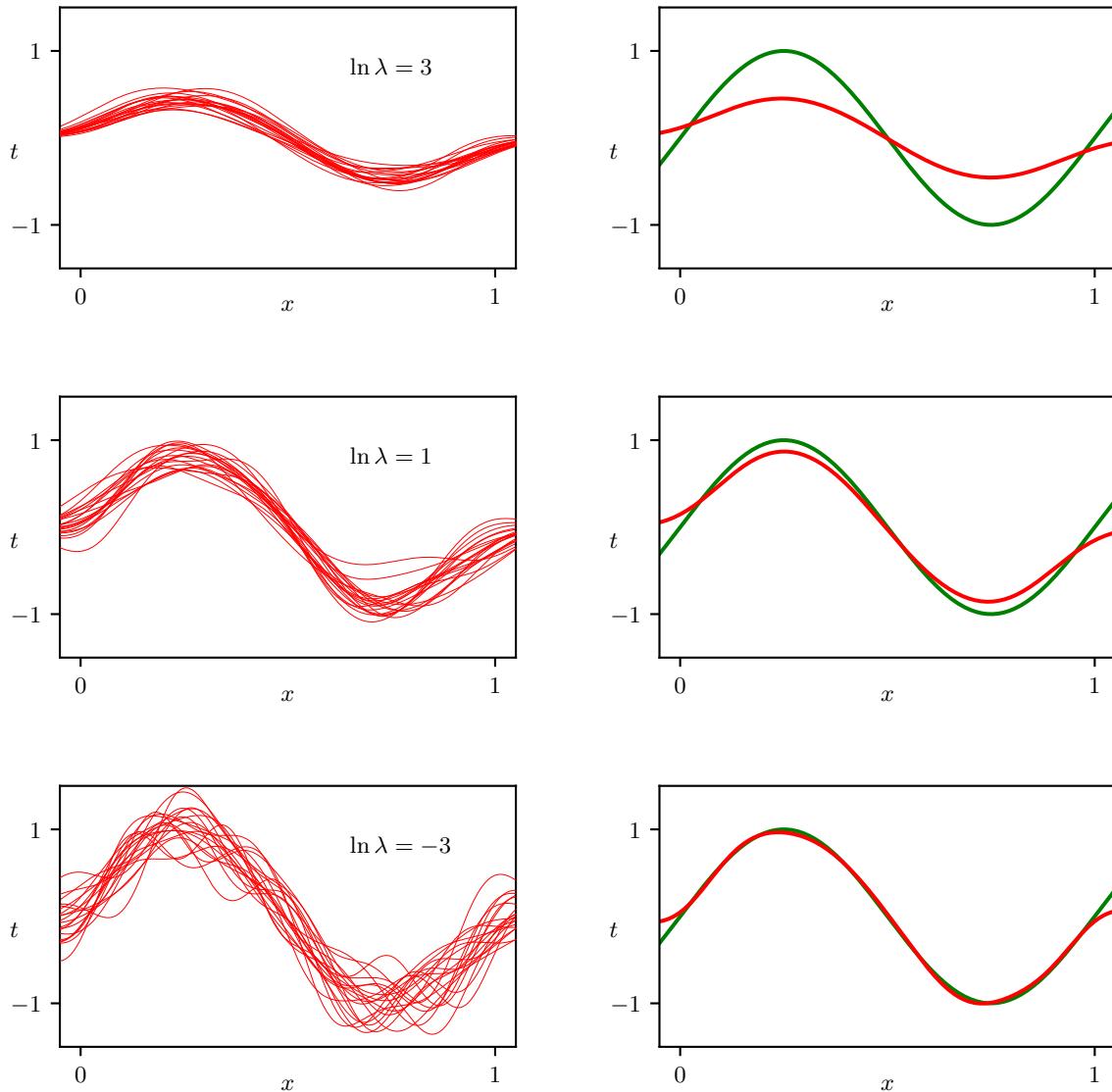
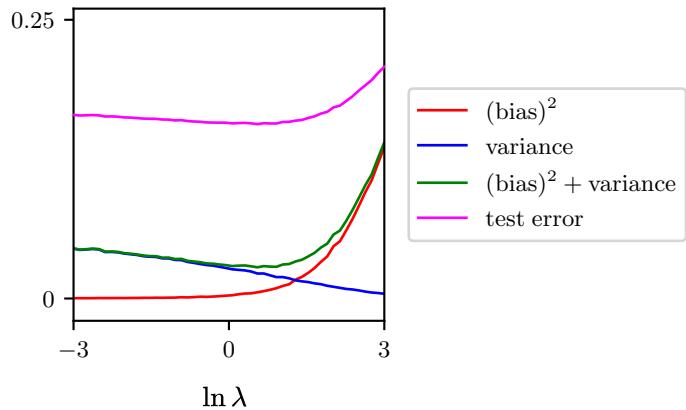


图4.7 使用第1章中的正弦数据，展示了偏差和方差对由正则化参数 λ 控制的模型复杂度的依赖性。有 $L = 100$ 个数据集，每个数据集有 $N = 25$ 个数据点，模型中有 24 个高斯基函数，因此包括偏差参数在内的参数总数为 $M = 25$ 。左列显示了针对 $\ln \lambda$ 的不同值将模型拟合到数据集的结果（为清晰起见，仅显示了 100 次拟合中的 20 次）。右列显示了 100 次拟合的相应平均值（红色）以及生成数据集的正弦函数（绿色）。

图4.8 平方偏差和方差的图，以及它们的和，与图4.7所示的结果相对应。还显示了测试数据集大小为1000个点时的平均测试集误差。 $(\text{偏差})^2 + \text{方差}$ 的最小值出现在 $\ln \lambda = 0.43$ 附近，这与在测试数据上产生最小误差的值接近。



练习

4.1 (*) 考虑由(1.2)给出的平方和误差函数，其中函数 $y(x, \mathbf{w})$ 由多项式(1.1)给出。证明使该误差函数最小化的系数 $\mathbf{w} = \{w_i\}$ 由以下线性方程组的解给出：

$$\sum_{j=0}^M A_{ij} w_j = T_i \quad (4.53)$$

其中

$$A_{ij} = \sum_{n=1}^N (x_n)^{i+j}, \quad T_i = \sum_{n=1}^N (x_n)^i t_n. \quad (4.54)$$

这里，后缀 i 或 j 表示一个分量的索引，而 $(x)^i$ 表示 x 的 i 次幂。

4.2 (*) 写出类似于(4.53)的耦合线性方程组，该方程组由使(1.4)给出的正则化平方和误差函数最小化的系数 w_i 满足。

4.3 (*) 证明由下式定义的双曲正切函数

$$\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}} \quad (4.55)$$

以及由(4.6)定义的逻辑 sigmoid 函数通过以下关系联系起来

$$\tanh(a) = 2\sigma(2a) - 1. \quad (4.56)$$

因此，证明形式为

$$y(x, \mathbf{w}) = w_0 + \sum_{j=1}^M w_j \sigma\left(\frac{x - \mu_j}{s}\right) \quad (4.57)$$

等同于以下形式的双曲正切函数的线性组合

$$y(x, \mathbf{u}) = u_0 + \sum_{j=1}^M u_j \tanh\left(\frac{x - \mu_j}{2s}\right) \quad (4.58)$$

并找到将新参数 $\{u_1, \dots, u_M\}$ 与原始参数 $\{w_1, \dots, w_M\}$ 相关联的表达式。

4.4 (★★★) 证明矩阵

$$\Phi(\Phi^T \Phi)^{-1} \Phi^T \quad (4.59)$$

取任意向量 \mathbf{v} 并将其投影到由 Φ 的列所张成的空间上。利用此结果证明最小二乘解(4.14)对应于向量 \mathbf{t} 到流形 \mathcal{S} 的正交投影，如图 4.3 所示。

4.5 (*) 考虑一个数据集，其中每个数据点 t_n 都与一个加权因子 $r_n > 0$ 相关联，使得平方和误差函数变为

$$E_D(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N r_n \{t_n - \mathbf{w}^T \phi(\mathbf{x}_n)\}^2. \quad (4.60)$$

找到一个使此误差函数最小化的解 \mathbf{w}^* 的表达式。从 (i) 数据相关的噪声方差和 (ii) 复制的数据点两个方面给出加权平方和误差函数的两种替代解释。

4.6 (*) 通过将 (4.26) 关于 \mathbf{w} 的梯度设为零，证明线性回归的正则化平方和误差函数的精确最小值由 (4.27) 给出。

4.7 (★★) 考虑一个多变量目标变量 \mathbf{t} 的线性基函数回归模型，其具有如下形式的高斯分布

$$p(\mathbf{t}|\mathbf{W}, \Sigma) = \mathcal{N}(\mathbf{t}|\mathbf{y}(\mathbf{x}, \mathbf{W}), \Sigma) \quad (4.61)$$

其中

$$\mathbf{y}(\mathbf{x}, \mathbf{W}) = \mathbf{W}^T \phi(\mathbf{x}) \quad (4.62)$$

连同一个训练数据集，该数据集由输入基向量 $\phi(\mathbf{x}_n)$ 和相应的目标向量 \mathbf{t}_n 组成，其中 $n = 1, \dots, N$ 。证明参数矩阵 \mathbf{W} 的最大似然解 \mathbf{W}_{ML} 具有这样的性质，即每一列都由形式为(4.14)的表达式给出，这是各向同性噪声分布的解。注意，这与协方差矩阵 Σ 无关。证明 Σ 的最大似然解由

$$\Sigma = \frac{1}{N} \sum_{n=1}^N (\mathbf{t}_n - \mathbf{W}_{ML}^T \phi(\mathbf{x}_n)) (\mathbf{t}_n - \mathbf{W}_{ML}^T \phi(\mathbf{x}_n))^T. \quad (4.63)$$

4.8 (*) 考虑将单个目标变量 t 的平方损失函数 (4.35) 推广到由向量 \mathbf{t} 描述的多个目标变量的情况, 如下所示

$$\mathbb{E}[L(\mathbf{t}, \mathbf{f}(\mathbf{x}))] = \iint \|\mathbf{f}(\mathbf{x}) - \mathbf{t}\|^2 p(\mathbf{x}, \mathbf{t}) d\mathbf{x} d\mathbf{t}. \quad (4.64)$$

使用变分法, 证明使该期望损失最小化的函数 $\mathbf{f}(\mathbf{x})$ 由下式给出

$$\mathbf{f}(\mathbf{x}) = \mathbb{E}_t[\mathbf{t}|\mathbf{x}]. \quad (4.65)$$

4.9 (*) 通过展开 (4.64) 中的平方, 推导与 (4.39) 类似的结果, 从而证明使目标变量向量 \mathbf{t} 的期望平方损失最小化的函数 $\mathbf{f}(\mathbf{x})$ 再次由 (4.65) 形式的 \mathbf{t} 的条件期望给出。

4.10 (**) 首先类似于 (4.39) 展开 (4.64), 重新推导结果 (4.65)。

4.11 (**) 以下分布

$$p(x|\sigma^2, q) = \frac{q}{2(2\sigma^2)^{1/q}\Gamma(1/q)} \exp\left(-\frac{|x|^q}{2\sigma^2}\right) \quad (4.66)$$

是单变量高斯分布的推广。这里 $\Gamma(x)$ 是由以下定义的伽马函数

$$\Gamma(x) = \int_{-\infty}^{\infty} u^{x-1} e^{-u} du. \quad (4.67)$$

证明此分布是归一化的, 使得

$$\int_{-\infty}^{\infty} p(x|\sigma^2, q) dx = 1 \quad (4.68)$$

并且当 $q = 2$ 时它简化为高斯分布。考虑一个回归模型, 其中目标变量由 $t = y(\mathbf{x}, \mathbf{w}) + \epsilon$ 给出, 并且 ϵ 是从分布(4.66)中抽取的随机噪声变量。证明对于输入向量 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ 和相应目标变量 $\mathbf{t} = (t_1, \dots, t_N)^T$ 的观测数据集, 关于 \mathbf{w} 和 σ^2 的对数似然函数由下式给出

$$\ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma^2) = -\frac{1}{2\sigma^2} \sum_{n=1}^N |y(\mathbf{x}_n, \mathbf{w}) - t_n|^q - \frac{N}{q} \ln(2\sigma^2) + \text{const} \quad (4.69)$$

其中, “const”表示与 \mathbf{w} 和 σ^2 均无关的项。请注意, 作为 \mathbf{w} 的函数, 这就是在4.2节中考虑的 L_q 误差函数。

4.12 (**) 考虑由(4.40)给出的 L_q 损失函数下回归问题的期望损失。写出 $y(\mathbf{x})$ 为使 $\mathbb{E}[L_q]$ 最小化必须满足的条件。证明, 对于 $q = 1$, 此解表示条件中位数, 即, 使得 $t < y(\mathbf{x})$ 的概率质量与 $t \geq y(\mathbf{x})$ 的概率质量相同的函数 $y(\mathbf{x})$ 。还证明 $q \rightarrow 0$ 的最小期望 L_q 损失由条件众数给出, 即, 对于每个 \mathbf{x} , 函数 $y(\mathbf{x})$ 等于使 $p(t|\mathbf{x})$ 最大化的 t 的值。



5

Single-layer Networks: Classification

在前一章中，我们探讨了一类回归模型，其中输出变量是模型参数的线性函数，因此可以表示为具有单层权重和偏差参数的简单神经网络。现在我们转向分类问题的讨论，在本章中，我们将关注一类类似的模型，这类模型同样可以表示为单层神经网络。在后面的章节中处理更一般的深度神经网络之前，这些模型将使我们能够引入许多分类的关键概念。

分类的目标是获取输入向量 $\mathbf{x} \in \mathbb{R}^D$ 并将其分配到 K 个离散类 \mathcal{C}_k 中的一个，其中 $k = 1, \dots, K$ 。在最常见的情况下，这些类被视为不相交的，因此每个输入被分配到一个且仅一个类。输入空间由此被划分为决策区域，其边界称为决策边界或决策表面。

在本章中，我们考虑用于分类的线性模型，这意味着决策面是输入向量 \mathbf{x} 的线性函数，因此，由 D 维输入空间内的 $(D - 1)$ 维超平面定义。其类别可以由线性决策面精确分离的数据集被称为线性可分的。线性分类模型可以应用于非线性可分的数据集，尽管并非所有输入都能被正确分类。

我们大致可以确定三种不同的解决分类问题的方法。最简单的方法是构建一个判别函数，直接将每个向量 \mathbf{x} 分配到特定的类别。然而，一种更强大的方法是在推理阶段对条件概率分布 $p(\mathcal{C}_k | \mathbf{x})$ 进行建模，然后使用这些分布做出最优决策。将推理和决策分开带来了许多好处。有两种不同的方法来确定条件概率 $p(\mathcal{C}_k | \mathbf{x})$ 。一种技术是直接对它们进行建模，例如将它们表示为参数模型，然后使用训练集优化参数。这将被称为判别概率模型。或者，我们可以对类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 以及类的先验概率 $p(\mathcal{C}_k)$ 进行建模，然后使用贝叶斯定理计算所需的后验概率：

$$p(\mathcal{C}_k | \mathbf{x}) = \frac{p(\mathbf{x} | \mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}. \quad (5.1)$$

这将被称为生成概率模型，因为它提供了从每个类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 生成样本的机会。在本章中，我们将讨论所有三种方法的示例：判别函数、生成概率模型和判别概率模型。

5.1. 判别函数

判别函数是一个将输入向量 \mathbf{x} 分配到 K 个类别之一的函数，记为 \mathcal{C}_k 。在本章中，我们将注意力限制在线性判别函数上，即那些决策面为超平面的函数。为了简化讨论，我们首先考虑两个类别，然后研究向 $K > 2$ 个类别的扩展。

5.1.1 两个类别

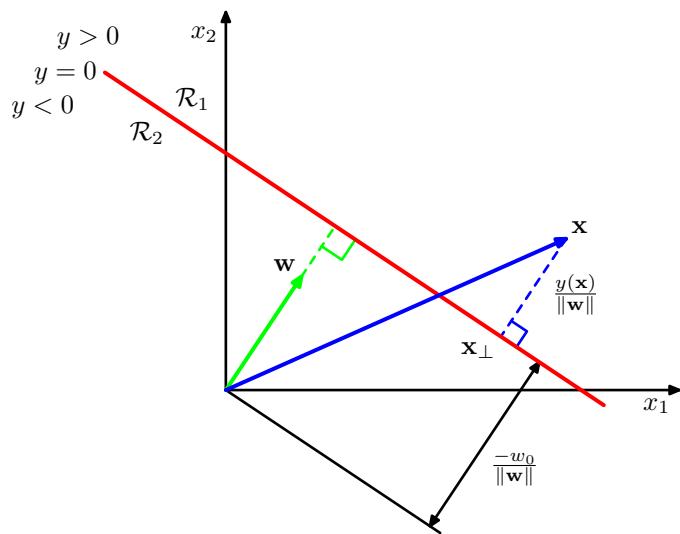
线性判别函数的最简单表示是通过对输入向量取线性函数得到的，使得

$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0 \quad (5.2)$$

其中 \mathbf{w} 被称为权重向量， w_0 是偏差（不要与统计意义上的偏差混淆）。如果 $y(\mathbf{x}) \geq 0$ ，则将输入向量 \mathbf{x} 分配给类别 \mathcal{C}_1 ，否则分配给类别 \mathcal{C}_2 。因此，相应的决策边界由关系 $y(\mathbf{x}) = 0$ 定义，它对应于 D 维输入空间内的一个 $(D - 1)$ 维超平面。

第5.2.4节

图5.1二维线性判别函数的几何示意图。所示的决策面为红色，垂直于 \mathbf{w} ，其与原点的位移由偏置参数 w_0 控制。此外，一般点 \mathbf{x} 到决策面的有符号正交距离由 $y(\mathbf{x})/\|\mathbf{w}\|$ 给出。



考虑两个点 \mathbf{x}_A 和 \mathbf{x}_B ，它们都位于决策面上。因为 $y(\mathbf{x}_A) = y(\mathbf{x}_B) = 0$ ，我们有 $\mathbf{w}^T(\mathbf{x}_A - \mathbf{x}_B) = 0$ ，因此向量 \mathbf{w} 与位于决策面内的每个向量正交，所以 \mathbf{w} 决定了决策面的方向。类似地，如果 \mathbf{x} 是决策面上的一个点，那么 $y(\mathbf{x}) = 0$ ，因此从原点到决策面的法向距离由

$$\frac{\mathbf{w}^T \mathbf{x}}{\|\mathbf{w}\|} = -\frac{w_0}{\|\mathbf{w}\|}. \quad (5.3)$$

因此我们可以看到，偏差参数 w_0 决定了决策面的位置。图5.1针对 $D = 2$ 的情况说明了这些特性。

此外，请注意， $y(\mathbf{x})$ 的值给出了点 \mathbf{x} 到决策面的垂直距离 r 的有符号度量。为了说明这一点，考虑任意一点 \mathbf{x} ，并设 \mathbf{x}_\perp 是它在决策面上的正交投影，那么

$$\mathbf{x} = \mathbf{x}_\perp + r \frac{\mathbf{w}}{\|\mathbf{w}\|}. \quad (5.4)$$

将此结果的两边都乘以 \mathbf{w}^T 并加上 w_0 ，并利用 $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$ 和 $y(\mathbf{x}_\perp) = \mathbf{w}^T \mathbf{x}_\perp + w_0 = 0$ ，我们得到

$$r = \frac{y(\mathbf{x})}{\|\mathbf{w}\|}. \quad (5.5)$$

此结果如图5.1所示。

第4.1.1节

与线性回归模型一样，有时使用更紧凑的表示法会很方便，在这种表示法中，我们引入一个额外的虚拟“输入”值 $x_0 = 1$ ，然后定义 $\tilde{\mathbf{w}} = (w_0, \mathbf{w})$ 和 $\tilde{\mathbf{x}} = (x_0, \mathbf{x})$ ，以便

$$y(\mathbf{x}) = \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}. \quad (5.6)$$

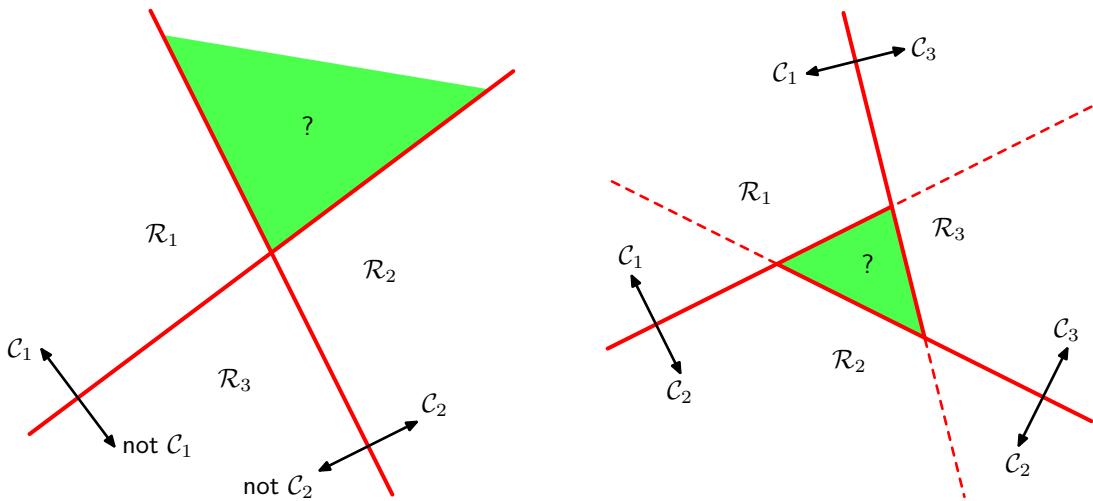


图5.2 试图从一组二类判别函数构建一个 K 类判别函数会导致出现模糊区域，如绿色部分所示。左边是一个示例，有两个判别函数用于区分 C_k 类中的点和不在 C_k 类中的点。右边是一个涉及三个判别函数的示例，每个判别函数用于分离一对类 C_k 和 C_j 。

在这种情况下，决策面是穿过 $(D + 1)$ 维扩展输入空间原点的 D 维超平面。

5.1.2 多类

现在考虑将线性判别函数扩展到 $K > 2$ 类。我们可能会试图通过组合多个二类判别函数来构建一个 K 类判别函数。然而，这会导致一些严重的困难 (Duda和Hart, 1973)，如下所示。

考虑一个具有 $K - 1$ 个分类器的模型，每个分类器解决一个二分类问题，即将特定类 C_k 中的点与不在该类中的点分开。这被称为一对其余分类器。图5.2左侧的示例展示了一个涉及三个类的例子，其中这种方法会导致输入空间中存在分类模糊的区域。

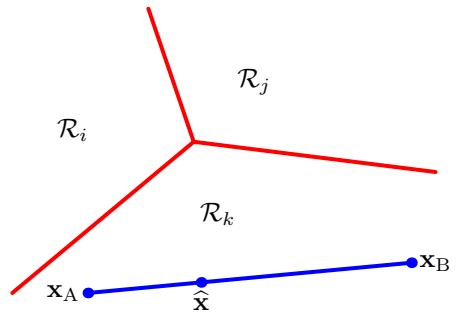
另一种方法是引入 $K(K - 1)/2$ 个二元判别函数，每个函数对应每一对可能的类。这被称为一对一分类器。然后根据判别函数之间的多数投票对每个点进行分类。然而，这也会遇到分类模糊区域的问题，如图5.2右侧的图所示。

我们可以通过考虑一个由 K 个形式为的线性函数组成的单一 K 类判别函数来避免这些困难

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0} \quad (5.7)$$

然后，如果对于所有的 $j \neq k$ 都有 $y_k(\mathbf{x}) > y_j(\mathbf{x})$ ，则将一个点 \mathbf{x} 分配给类别 C_k 。因此，类别 C_k 和类别 C_j 之间的决策边界由 $y_k(\mathbf{x}) = y_j(\mathbf{x})$ 给出，因此对应于一个由其定义的 $(D - 1)$ 维超平面

图5.3 多类线性判别器决策区域的图示，决策边界用红色表示。如果两个点 x_A 和 x_B 都位于同一个决策区域 \mathcal{R}_k 内，那么连接这两个点的直线上的任何点 \hat{x} 也必定位于 \mathcal{R}_k 内，因此，决策区域必定是单连通且凸的。



$$(\mathbf{w}_k - \mathbf{w}_j)^T \mathbf{x} + (w_{k0} - w_{j0}) = 0. \quad (5.8)$$

这与5.1.1节中讨论的两类情况的决策边界具有相同的形式，因此类似的几何性质也适用。

这样一个判别式的决策区域总是单连通且凸的。要明白这一点，考虑两个点 x_A 和 x_B ，它们都位于决策区域 \mathcal{R}_k 内，如图5.3所示。位于连接 x_A 和 x_B 的直线上的任何点 \hat{x} 都可以表示为如下形式

$$\hat{x} = \lambda x_A + (1 - \lambda) x_B \quad (5.9)$$

其中 $0 \leq \lambda \leq 1$ 。根据判别函数的线性，可得

$$y_k(\hat{x}) = \lambda y_k(x_A) + (1 - \lambda) y_k(x_B). \quad (5.10)$$

因为 x_A 和 x_B 都位于 \mathcal{R}_k 内部，所以对于所有 $j \neq k$ ，有 $y_k(x_A) > y_j(x_A)$ 且 $y_k(x_B) > y_j(x_B)$ ，因此 $y_k(\hat{x}) > y_j(\hat{x})$ ，进而 \hat{x} 也位于 \mathcal{R}_k 内部。所以， \mathcal{R}_k 是单连通且凸的。

请注意，对于两类情况，我们既可以采用这里讨论的形式主义，基于两个判别函数 $y_1(\mathbf{x})$ 和 $y_2(\mathbf{x})$ ，也可以使用基于单个判别函数 $y(\mathbf{x})$ 的更简单但本质上等效的公式。

第5.1.1节

5.1.3 K 选1编码

对于回归问题，目标变量 t 只是我们希望预测其值的实数向量。在分类中，有多种使用目标值来表示类别标签的方法。对于二分类问题，最方便的是二进制表示，其中有一个单一的目标变量 $t \in \{0, 1\}$ ，使得 $t = 1$ 表示类别 C_1 ，而 $t = 0$ 表示类别 C_2 。我们可以将 t 的值解释为类别为 C_1 的概率，概率值仅取0和1这两个极端值。对于 $K > 2$ 个类别，使用独热编码方案（也称为one-hot编码方案）很方便，其中 t 是长度为 K 的向量，使得如果类别是 C_j ，那么 t 的所有元素 t_k 都为零，除了元素 t_j ，其值为1。

例如，如果我们有 $K = 5$ 个类别，那么来自类别2的数据点将被赋予目标向量

$$\mathbf{t} = (0, 1, 0, 0, 0)^T. \quad (5.11)$$

同样，我们可以将 t_k 的值解释为该类别是 \mathcal{C}_k 的概率，其中概率仅取0和1两个值。

5.1.4 分类的最小二乘法

4.1.3节

对于线性回归模型，最小化平方和误差函数会得到参数值的简单闭式解。因此，人们很想看看是否可以将相同的最小二乘形式应用于分类问题。考虑一个具有 K 个类别的一般分类问题，以及目标向量 \mathbf{t} 的 K 选1二进制编码方案。在这种情况下使用最小二乘法的一个理由是，它近似于给定输入向量时目标值的条件期望 $\mathbb{E}[\mathbf{t} | \mathbf{x}]$ 。对于二进制编码方案，这个条件期望由后验类概率向量给出。不幸的是，这些概率通常近似得很差，实际上这些近似值可能超出 $(0, 1)$ 范围。然而，探索这些简单模型并理解这些局限性是如何产生的是很有启发性的。

练习5.1

每个类 \mathcal{C}_k 都由其自己的线性模型描述，以便

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0} \quad (5.12)$$

其中 $k = 1, \dots, K$ 。我们可以使用向量表示法方便地将这些组合在一起，以便

$$\mathbf{y}(\mathbf{x}) = \widetilde{\mathbf{W}}^T \tilde{\mathbf{x}} \quad (5.13)$$

其中 $\widetilde{\mathbf{W}}$ 是一个矩阵，其第 k 列由 $(D + 1)$ 维向量 $\tilde{\mathbf{w}}_k = (w_{k0}, \mathbf{w}_k^T)^T$ 组成，并且 $\tilde{\mathbf{x}}$ 是具有虚拟输入 $x_0 = 1$ 的相应增强输入向量 $(1, \mathbf{x}^T)^T$ 。然后将新输入 \mathbf{x} 分配给输出 $y_k = \tilde{\mathbf{w}}_k^T \tilde{\mathbf{x}}$ 最大的类。

我们现在通过最小化一个平方和误差函数来确定参数矩阵 $\widetilde{\mathbf{W}}$ 。考虑一个训练数据集 $\{\mathbf{x}_n, \mathbf{t}_n\}$ ，其中 $n = 1, \dots, N$ ，并定义一个矩阵 \mathbf{T} ，其第 n 行是向量 \mathbf{t}_n^T ，以及一个矩阵 $\tilde{\mathbf{X}}$ ，其第 n 行是 $\tilde{\mathbf{x}}_n^T$ 。然后平方和误差函数可以写成

$$E_D(\widetilde{\mathbf{W}}) = \frac{1}{2} \text{Tr} \left\{ (\tilde{\mathbf{X}} \widetilde{\mathbf{W}} - \mathbf{T})^T (\tilde{\mathbf{X}} \widetilde{\mathbf{W}} - \mathbf{T}) \right\}. \quad (5.14)$$

将关于 $\widetilde{\mathbf{W}}$ 的导数设为零并重新排列，我们得到 $\widetilde{\mathbf{W}}$ 的解的形式为

$$\widetilde{\mathbf{W}} = (\tilde{\mathbf{X}}^T \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^T \mathbf{T} = \tilde{\mathbf{X}}^{\dagger} \mathbf{T} \quad (5.15)$$

4.1.3节

其中 $\tilde{\mathbf{X}}^{\dagger}$ 是矩阵 $\tilde{\mathbf{X}}$ 的伪逆。然后我们得到如下形式的判别函数

$$\mathbf{y}(\mathbf{x}) = \widetilde{\mathbf{W}}^T \tilde{\mathbf{x}} = \mathbf{T}^T (\tilde{\mathbf{X}}^\dagger)^T \tilde{\mathbf{x}}. \quad (5.16)$$

具有多个目标变量的最小二乘解的一个有趣特性是，如果训练集中的每个目标向量都满足某个线性约束

$$\mathbf{a}^T \mathbf{t}_n + b = 0 \quad (5.17)$$

对于某些常数 \mathbf{a} 和 b ，那么对于 \mathbf{x} 的任何值，模型预测都将满足相同的约束，因此

$$\mathbf{a}^T \mathbf{y}(\mathbf{x}) + b = 0. \quad (5.18)$$

因此，如果我们对 K 个类别使用 K 选1编码方案，那么模型做出的预测将具有这样的特性，即对于 \mathbf{x} 的任何值， $\mathbf{y}(\mathbf{x})$ 的元素之和都为1。然而，仅这个求和约束不足以使模型输出被解释为概率，因为它们没有被约束在区间 $(0, 1)$ 内。

练习5.3

第2.3.4节

最小二乘法为判别函数参数提供了精确的闭式解。然而，即使作为判别函数（我们直接用它来做决策，而不考虑任何概率解释），它也存在一些严重问题。我们已经看到，在高斯噪声分布的假设下，平方和误差函数可以被视为负对数似然。如果数据的真实分布与高斯分布明显不同，那么最小二乘法可能会给出很差的结果。特别是，最小二乘法对异常值非常敏感，异常值是远离大部分数据的数据点。图5.4对此进行了说明。在这里我们看到，右图中的额外数据点导致决策边界的位置发生了显著变化，尽管这些点在左图中会被原始决策边界正确分类。平方和误差函数对远离决策边界的的数据点给予了过多的权重，即使它们被正确分类。异常值可能是由于罕见事件引起的，也可能仅仅是由于数据集中的错误。对极少数数据点敏感的技术被认为缺乏鲁棒性。为了进行比较，图5.4还展示了一种称为逻辑回归的技术的结果，它对异常值更具鲁棒性。

第5.4.3节

当我们回想起最小二乘法对应于高斯条件分布假设下的最大似然估计时，它的失败就不足为奇了，而二元目标向量显然具有与高斯分布相差甚远的分布。通过采用更合适的概率模型，我们可以获得比最小二乘法具有更好性质的分类技术，并且这些技术还可以推广到给出灵活的非线性神经网络模型，正如我们将在后面的章节中看到的那样。

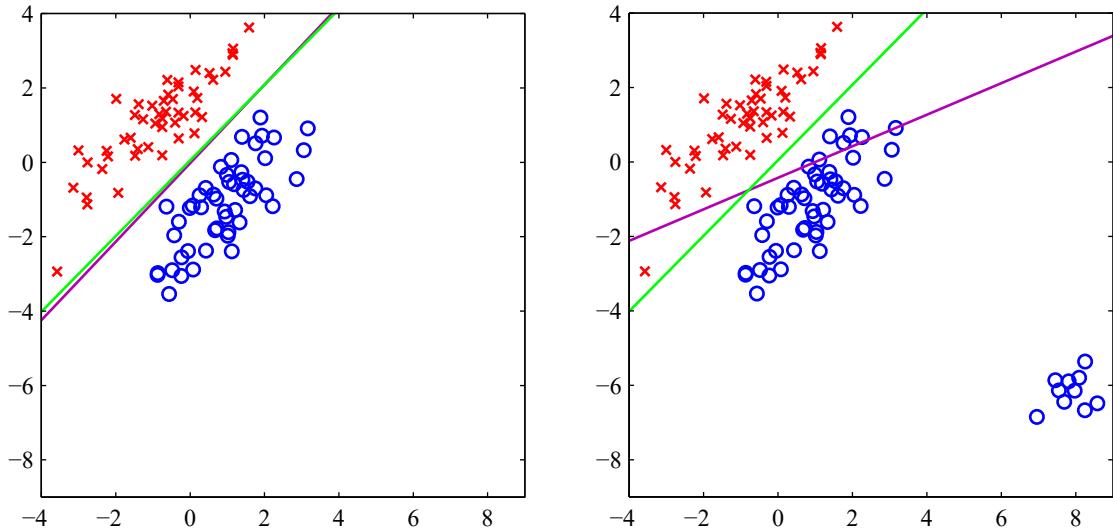


图5.4 左侧的图展示了来自两个类别的数据，分别用红色叉号和蓝色圆圈表示，同时还展示了通过最小二乘法（品红色曲线）和逻辑回归模型（绿色曲线）找到的决策边界。右侧的图展示了在图的右下角添加额外数据点时得到的相应结果，表明最小二乘法对异常值高度敏感，而逻辑回归则不然。

5.2. 决策理论

第4.2节

当我们讨论线性回归时，我们看到了机器学习中的预测过程如何可以分解为推理和决策这两个阶段。现在我们将更深入地探讨这个观点，特别是在分类器的背景下。

假设我们有一个输入向量 x 以及与之对应的目标变量向量 t ，我们的目标是在给定 x 的一个新值的情况下预测 t 。对于回归问题， t 将包含连续变量，并且通常会是一个向量，因为我们可能希望预测几个相关的量。对于分类问题， t 将表示类别标签。同样，如果我们有两个以上的类别， t 通常也会是一个向量。联合概率分布 $p(x, t)$ 提供了与这些变量相关的不确定性的完整总结。从一组训练数据中确定 $p(x, t)$ 是推理的一个例子，并且通常是一个非常困难的问题，本书的大部分内容都围绕这个问题的解决方案展开。然而，在实际应用中，我们通常必须对 t 的值做出具体的预测，或者更一般地说，根据我们对 t 可能取值的理解采取具体的行动，而这方面就是决策理论的主题。

例如，考虑我们之前的医疗诊断问题，在这个问题中，我们拍摄了一位患者皮肤病变的图像，并且我们希望确定该患者是否患有癌症。

在这种情况下，输入向量 \mathbf{x} 是图像中的像素强度集，输出变量 t 将表示无癌症，我们用类别 \mathcal{C}_1 表示，或者表示有癌症，我们用类别 \mathcal{C}_2 表示。例如，我们可以选择 t 为一个二元变量，使得 $t = 0$ 对应于类别 \mathcal{C}_1 ， $t = 1$ 对应于类别 \mathcal{C}_2 。我们稍后会看到，在处理概率时，这种标签值的选择特别方便。一般的推理问题则涉及确定联合分布 $p(\mathbf{x}, \mathcal{C}_k)$ ，或者等效地确定 $p(\mathbf{x}, t)$ ，它为我们提供了变量的最完整概率描述。虽然这可能是一个非常有用且信息丰富的量，但最终，我们必须决定是否对患者进行治疗，并且我们希望根据某种适当的标准（Duda 和 Hart, 1973）使这个选择是最优的。这就是决策步骤，决策理论的目标是它应该告诉我们如何在给定适当概率的情况下做出最优决策。我们会看到，一旦我们解决了推理问题，决策阶段通常非常简单，甚至是微不足道的。在这里，我们根据本书其余部分的需要，对决策理论的关键思想进行介绍。更多的背景知识以及更详细的阐述，可以在 Berger (1985) 和 Bather (2000) 中找到。

在进行更详细的分析之前，让我们先非正式地考虑一下我们可能期望概率在决策中如何发挥作用。当我们为一位新患者获取皮肤图像 \mathbf{x} 时，我们的目标是决定将该图像分配到两个类别中的哪一个。因此，我们感兴趣的是在给定图像的情况下两个类别的概率，这些概率由 $p(\mathcal{C}_k | \mathbf{x})$ 给出。使用贝叶斯定理，这些概率可以表示为以下形式

$$p(\mathcal{C}_k | \mathbf{x}) = \frac{p(\mathbf{x} | \mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}. \quad (5.19)$$

请注意，贝叶斯定理中出现的任何量都可以通过对适当变量进行边缘化或条件化，从联合分布 $p(\mathbf{x}, \mathcal{C}_k)$ 中获得。现在我们可以将 $p(\mathcal{C}_k)$ 解释为类别 \mathcal{C}_k 的先验概率，将 $p(\mathcal{C}_k | \mathbf{x})$ 解释为相应的后验概率。因此， $p(\mathcal{C}_1)$ 表示在拍摄图像之前一个人患癌症的概率。类似地， $p(\mathcal{C}_1 | \mathbf{x})$ 是后验概率，它是根据图像中包含的信息使用贝叶斯定理进行修正的。如果我们的目标是最小化将 \mathbf{x} 分配到错误类别的机会，那么直观地我们会选择后验概率较高的类别。现在我们证明这种直觉是正确的，并且我们还将讨论更一般的决策标准。

5.2.1 误分类率

假设我们的目标仅仅是尽可能减少误分类的数量。我们需要一个规则，将 \mathbf{x} 的每个值分配到可用类别之一。这样的规则会将输入空间划分为称为决策区域的区域 \mathcal{R}_k ，每个类别对应一个区域，使得 \mathcal{R}_k 中的所有点都被分配到类别 \mathcal{C}_k 。决策区域之间的边界称为决策边界或决策表面。请注意，每个决策区域不一定是连续的，可能由一些不相交的区域组成。为了找到最优决策规则，首先考虑两类的情况，例如在癌症问题中。当属于类别 \mathcal{C}_1 的输入向量被分配到类别 \mathcal{C}_2 ，或者反之亦然时，就会发生错误。

发生这种情况的概率由下式给出

$$\begin{aligned} p(\text{mistake}) &= p(\mathbf{x} \in \mathcal{R}_1, \mathcal{C}_2) + p(\mathbf{x} \in \mathcal{R}_2, \mathcal{C}_1) \\ &= \int_{\mathcal{R}_1} p(\mathbf{x}, \mathcal{C}_2) d\mathbf{x} + \int_{\mathcal{R}_2} p(\mathbf{x}, \mathcal{C}_1) d\mathbf{x}. \end{aligned} \quad (5.20)$$

我们可以自由选择将每个点 \mathbf{x} 分配到两个类别之一的决策规则。显然，为了最小化 p （错误率），我们应该安排将每个 \mathbf{x} 分配到在(5.20)中被积函数值较小的那个类别。因此，如果对于给定的 \mathbf{x} 值有 $p(\mathbf{x}, \mathcal{C}_1) > p(\mathbf{x}, \mathcal{C}_2)$ ，那么我们应该将那个 \mathbf{x} 分配到类别 \mathcal{C}_1 。根据概率的乘积法则，我们有 $p(\mathbf{x}, \mathcal{C}_k) = p(\mathcal{C}_k | \mathbf{x})p(\mathbf{x})$ 。因为因子 $p(\mathbf{x})$ 在两项中都相同，我们可以将这个结果重新表述为：如果将每个 \mathbf{x} 值分配到后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 最大的类别，那么犯错误的概率最小。图5.5展示了两个类别和单个输入变量 x 的这种情况。

对于 K 类的更一般情况，最大化正确概率会稍微容易一些，其由下式给出

$$\begin{aligned} p(\text{correct}) &= \sum_{k=1}^K p(\mathbf{x} \in \mathcal{R}_k, \mathcal{C}_k) \\ &= \sum_{k=1}^K \int_{\mathcal{R}_k} p(\mathbf{x}, \mathcal{C}_k) d\mathbf{x}, \end{aligned} \quad (5.21)$$

当选择区域 \mathcal{R}_k 使得每个 \mathbf{x} 被分配到 $p(\mathbf{x}, \mathcal{C}_k)$ 最大的类时，该式达到最大值。同样，使用乘积规则 $p(\mathbf{x}, \mathcal{C}_k) = p(\mathcal{C}_k | \mathbf{x})p(\mathbf{x})$ ，并注意到 $p(\mathbf{x})$ 这个因子在所有项中是共同的，我们看到每个 \mathbf{x} 应该被分配到后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 最大的类。

5.2.2 期望损失

对于许多应用而言，我们的目标将比仅仅最小化错误分类的数量更为复杂。让我们再次考虑医疗诊断问题。我们注意到，如果一个没有患癌症的患者被错误地诊断为患有癌症，其后果可能是他们会经历一些痛苦，并且还需要进一步的检查。相反，如果一个患有癌症的患者被诊断为健康，结果可能是由于缺乏治疗而过早死亡。因此，这两种错误的后果可能会有很大的不同。显然，即使以增加第一种错误的数量为代价，减少第二种错误的数量会更好。

我们可以通过引入一个损失函数（也称为成本函数）来将此类问题形式化，损失函数是对采取任何可用决策或行动所产生的损失的单一总体度量。然后我们的目标是最小化所产生的总损失。请注意，有些作者考虑的是效用函数，其值

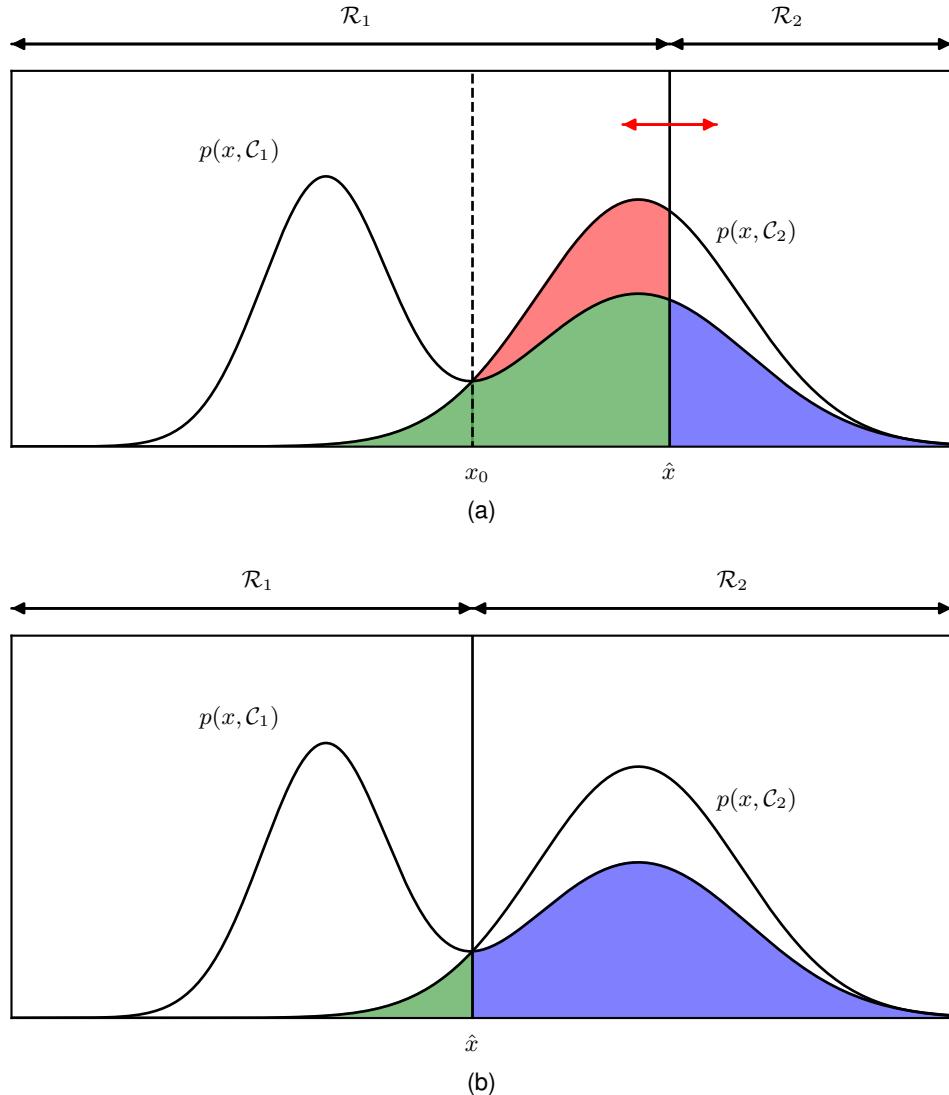


图5.5绘制了两个类别中每一个类别的联合概率 $p(x, \mathcal{C}_k)$ 与 x 的关系示意图，以及决策边界 $x = \hat{x}$ 。 $x \geq \hat{x}$ 的值被分类为类别 \mathcal{C}_2 ，因此属于决策区域 \mathcal{R}_2 ，而点 $x < \hat{x}$ 被分类为 \mathcal{C}_1 并属于 \mathcal{R}_1 。误差来自胆汁色、绿色和红色区域，因此对于 $x < \hat{x}$ ，误差是由于来自类别 \mathcal{C}_2 的点被误分类为 \mathcal{C}_1 （由红色和绿色区域的总和表示）。相反，对于区域 $x \geq \hat{x}$ 中的点，误差是由于来自类别 \mathcal{C}_1 的点被误分类为 \mathcal{C}_2 （由蓝色区域表示）。如 (a) 中红色双头箭头所示，通过改变决策边界 \hat{x} 的位置，蓝色和绿色区域的总面积保持不变，而红色区域的大小变化。 \hat{x} 的最佳选择是 $p(x, \mathcal{C}_1)$ 和 $p(x, \mathcal{C}_2)$ 的曲线相交的地方，如 (b) 所示并对应于 $\hat{x} = x_0$ ，因为在这种情况下红色区域消失。这等同于最小误分类率决策规则，该规则将 x 的每个值分配给具有更高后验概率 $p(\mathcal{C}_k | x)$ 的类别。

图5.6癌症治疗问题中一个损失矩阵的示例，其元素为 L_{kj} 。行对应真实类别，而列对应我们的决策标准所做出的类别分配。

	正常	癌症
正常癌 症		
癌 症		

它们旨在实现最大化。如果我们将效用简单地视为损失的负值，那么这些是等价的概念。在本文中，我们将采用损失函数约定。假设对于 \mathbf{x} 的一个新值，真实类别是 C_k ，并且我们将 \mathbf{x} 分配到类别 C_j （其中 j 可能等于也可能不等于 k ）。这样做时，我们会产生一定程度的损失，我们用 L_{kj} 表示，它可以被视为损失矩阵的 k, j 元素。例如，在我们的癌症示例中，我们可能有一个如图5.6所示形式的损失矩阵。这个特定的损失矩阵表明，如果做出了正确的决策，则不会产生损失；如果将健康患者诊断为患有癌症，则损失为1；而如果将患有癌症的患者诊断为健康，则损失为100。

最优解是使损失函数最小化的那个解。然而，损失函数取决于真实类别，而真实类别是未知的。对于给定的输入向量 \mathbf{x} ，我们对真实类别的不确定性通过联合概率分布 $p(\mathbf{x}, C_k)$ 来表示，因此我们转而寻求最小化平均损失，其中平均值是相对于此分布计算的，由下式给出

$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(\mathbf{x}, C_k) d\mathbf{x}. \quad (5.22)$$

每个 \mathbf{x} 可以独立地分配到决策区域 \mathcal{R}_j 中的一个。我们的目标是选择区域 \mathcal{R}_j 以最小化预期损失(5.22)，这意味着对于每个 \mathbf{x} ，我们应该最小化 $\sum_k L_{kj} p(\mathbf{x}, C_k)$ 。和以前一样，我们可以使用乘积规则

$p(\mathbf{x}, C_k) = p(C_k | \mathbf{x})p(\mathbf{x})$ 来消除 $p(\mathbf{x})$ 的公因子。因此，最小化预期损失的决策规则将每个新的 \mathbf{x} 分配到类 j ，对于该类，数量

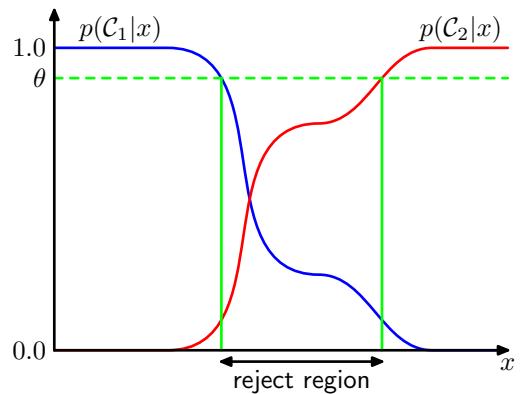
$$\sum_k L_{kj} p(C_k | \mathbf{x}) \quad (5.23)$$

是最小值。一旦我们为损失矩阵元素 L_{kj} 选择了值，这显然很容易做到。

5.2.3 拒绝选项

我们已经看到，分类错误源于输入空间中后验概率 $p(C_k | \mathbf{x})$ 最大的值明显小于1的区域，或者等效地，联合分布 $p(\mathbf{x}, C_k)$ 具有可比值的区域。这些都是我们对类别成员相对不确定的区域。在某些应用中，为了预期在做出分类决策的那些示例上获得较低的错误率，避免对困难情况做出决策是合适的。这被称为拒绝选项。

图5.7拒绝选项的说明。输入 x ，使得两个后验概率中较大的那个小于或等于某个阈值 θ ，将被拒绝。



例如，在我们假设的癌症筛查示例中，对于那些正确类别几乎没有疑问的图像，使用自动系统进行分类可能是合适的，而对于更模糊的情况，则要求进行活检以进行分类。我们可以通过引入一个阈值 θ 并拒绝那些后验概率 $p(C_k | \mathbf{x})$ 中最大的小于或等于 θ 的输入 \mathbf{x} 来实现这一点。图5.7展示了针对两个类别和单个连续输入变量 x 的这种情况。请注意，设置 $\theta = 1$ 将确保所有示例都被拒绝，而如果有 K 个类别，那么设置 $\theta < 1/K$ 将确保没有示例被拒绝。因此，被拒绝的示例比例由 θ 的值控制。

当给出损失矩阵时，我们可以通过考虑做出拒绝决策时产生的损失，轻松扩展拒绝标准以最小化预期损失。

练习5.10

5.2.4 推理与决策

我们已将分类问题分解为两个独立阶段，即推理阶段（在此阶段我们使用训练数据来学习 $p(C_k | \mathbf{x})$ 的模型）和随后的决策阶段（在此阶段我们使用这些后验概率进行最优类别分配）。另一种可能性是将两个问题一起解决，直接学习一个将输入 \mathbf{x} 映射到决策的函数。这样的函数称为判别函数。

事实上，我们可以确定三种不同的解决决策问题的方法，所有这些方法都已在实际应用中使用。按复杂度降序排列如下：

- (a) 首先，分别解决确定每个类别 C_k 的类别条件密度 $p(\mathbf{x} | C_k)$ 的推理问题。分别推断先验类别概率 $p(C_k)$ 。然后使用以下形式的贝叶斯定理

$$p(C_k | \mathbf{x}) = \frac{p(\mathbf{x} | C_k)p(C_k)}{p(\mathbf{x})} \quad (5.24)$$

来找到后验类别概率 $p(C_k | \mathbf{x})$ 。

与往常一样，贝叶斯定理中的分母可以根据分子中的量来确定，使用

$$p(\mathbf{x}) = \sum_k p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k). \quad (5.25)$$

等效地，我们可以直接对联合分布 $p(\mathbf{x}, \mathcal{C}_k)$ 进行建模，然后进行归一化以获得后验概率。在找到后验概率后，我们使用决策理论来确定每个新输入 \mathbf{x} 的类别归属。明确或隐含地对输入以及输出的分布进行建模的方法称为生成模型，因为通过从它们中采样，可以在输入空间中生成合成数据点。

(b) 首先，解决确定后验类别概率 $p(\mathcal{C}_k | \mathbf{x})$ 的推理问题，然后随后使用决策理论将每个新的 \mathbf{x} 分配到其中一个类别。直接对后验概率进行建模的方法称为判别模型。

(c) 找到一个称为判别函数的函数 $f(\mathbf{x})$ ，它将每个输入 \mathbf{x} 直接映射到一个类别标签上。例如，对于二分类问题， $f(\cdot)$ 可能是二值的，使得 $f = 0$ 表示类别 \mathcal{C}_1 ，而 $f = 1$ 表示类别 \mathcal{C}_2 。在这种情况下，概率不起作用。

让我们考虑这三种方法各自的优点。方法 (a) 要求最高，因为它涉及找到 \mathbf{x} 和 \mathcal{C}_k 上的联合分布。对于许多应用， \mathbf{x} 将具有高维度，因此，我们可能需要一个大的训练集才能以合理的精度确定类条件密度。请注意，类先验概率 $p(\mathcal{C}_k)$ 通常可以简单地从每个类中训练集数据点的比例来估计。然而，方法 (a) 的一个优点是它还允许从 (5.25) 确定数据 $p(\mathbf{x})$ 的边际密度。这对于检测在模型下概率较低且预测准确性可能较低的新数据点很有用，这被称为异常值检测或新奇性检测 (Bishop, 1994; Tarassenko, 1995)。

然而，如果我们只想做出分类决策，那么在实际上我们真正只需要后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 时，去寻找联合分布 $p(\mathbf{x}, \mathcal{C}_k)$ 可能会浪费计算资源且对数据要求过高，而后验概率可以通过方法 (b) 直接获得。实际上，如图5.8所示，类条件密度可能包含大量对后验概率影响不大的结构。人们对探索生成式和判别式机器学习方法的相对优点以及找到将它们结合的方法非常感兴趣 (Jebara, 2004; Lasserre, Bishop, and Minka, 2006)。

一种更简单的方法是 (c)，即我们使用训练数据来找到一个判别函数 $f(\mathbf{x})$ ，它将每个 \mathbf{x} 直接映射到一个类标签上，从而将推理和决策阶段合并为一个单一的学习问题。

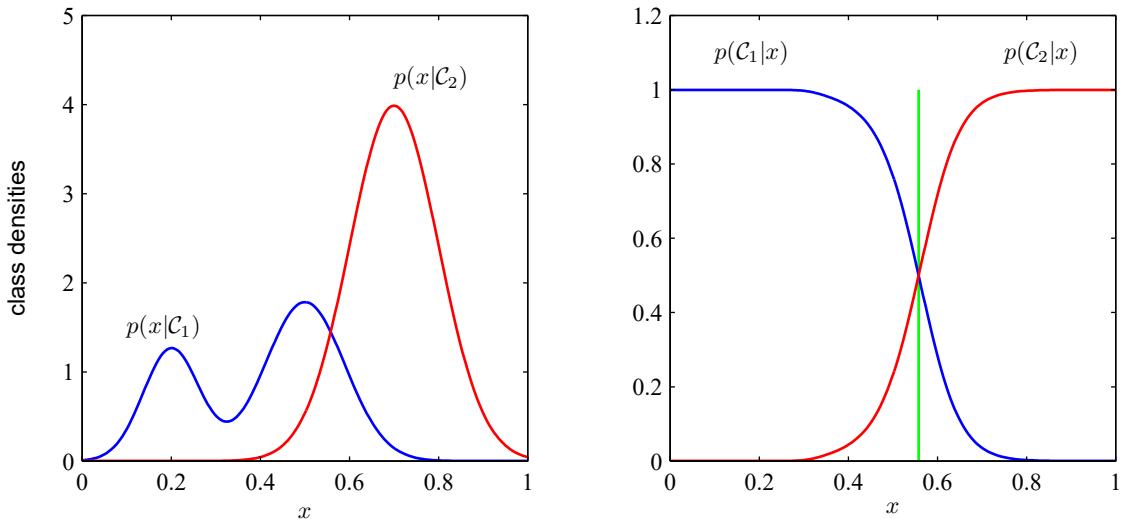


图5.8 具有单个输入变量 x 的两类的类条件密度示例（左图）以及相应的后验概率（右图）。请注意，左图中以蓝色显示的类条件密度 $p(\mathbf{x} | \mathcal{C}_1)$ 的左侧模态对后验概率没有影响。右图中的垂直绿线显示了在 x 中的决策边界，该边界给出了最小误分类率，假设先验类概率 $p(\mathcal{C}_1)$ 和 $p(\mathcal{C}_2)$ 相等。

在图5.8的示例中，这将对应于找到垂直绿线所示的 x 值，因为这是给出最小误分类概率的决策边界。

然而，对于选项(c)，我们无法再获得后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 。即使我们随后使用它们来做决策，计算后验概率也有许多强有力的理由。这些理由包括：

最小化风险。考虑一个损失矩阵的元素会不时修订的问题（比如在金融应用中可能出现的情况）。如果我们知道后验概率，我们可以通过适当地修改(5.23)轻松地修订最小风险决策标准。如果我们只有一个判别函数，那么损失矩阵的任何变化都将要求我们回到训练数据并重新解决推理问题。

拒绝选项。后验概率使我们能够确定一个拒绝标准，该标准将针对给定比例的被拒绝数据点最小化误分类率，或者更一般地说，最小化预期损失。

第2.1.1节

补偿类别先验概率。再次考虑我们的癌症筛查示例，假设我们从普通人群中收集了大量图像作为训练数据，并用这些数据构建一个自动筛查系统。由于癌症在普通人群中很少见，我们可能会发现，比如说，每1000个示例中只有1个对应于癌症的存在。

如果我们使用这样一个数据集来训练一个自适应模型，由于癌症类别中的样本比例较小，我们可能会遇到严重的困难。例如，一个将每个点都归类为正常类别的分类器将达到99.9%的准确率，并且可能很难避免这种简单的解决方案。此外，即使是一个大数据集，也将包含非常少的与癌症对应的皮肤图像示例，因此学习算法不会接触到广泛的此类图像示例，因此不太可能很好地泛化。一个来自每个类别的示例数量相等的平衡数据集将使我们能够找到一个更准确的模型。然而，我们 then have to compensate for the effects of our modifications to the training data. 假设我们使用了这样一个修改后的数据集并找到了后验概率的模型。从贝叶斯定理(5.24)中，我们看到后验概率与先验概率成正比，我们可以将其解释为每个类中的点的比例。因此，我们可以简单地取从我们人工平衡的数据集中获得的后验概率，除以该数据集中的类比例，然后乘以我们希望应用该模型的总体中的类比例。最后，我们需要进行归一化以确保新的后验概率之和为1。请注意，如果我们直接学习了一个判别函数而不是确定后验概率，则此过程不能应用。

组合模型。对于复杂的应用，我们可能希望将问题分解为若干较小的子问题，每个子问题都可以由一个单独的模块来处理。例如，在我们假设的医疗诊断问题中，我们可能有来自血液检测以及皮肤图像等方面的信息。与其将所有这些异构信息组合到一个巨大的输入空间中，构建一个系统来解释图像和另一个系统来解释血液数据可能会更有效。如果这两个模型中的每一个都给出了类别的后验概率，那么我们可以使用概率规则系统地组合输出。一种简单的方法是分别对每个类别假设，由 \mathbf{x}_I 表示的图像输入分布和由 \mathbf{x}_B 表示的血液数据输入分布是独立的，这样

$$p(\mathbf{x}_I, \mathbf{x}_B | \mathcal{C}_k) = p(\mathbf{x}_I | \mathcal{C}_k)p(\mathbf{x}_B | \mathcal{C}_k). \quad (5.26)$$

第11.2节

这是一个条件独立性属性的示例，因为当分布以类别 \mathcal{C}_k 为条件时独立性成立。给定图像和血液数据时的后验概率由下式给出

$$\begin{aligned} p(\mathcal{C}_k | \mathbf{x}_I, \mathbf{x}_B) &\propto p(\mathbf{x}_I, \mathbf{x}_B | \mathcal{C}_k)p(\mathcal{C}_k) \\ &\propto p(\mathbf{x}_I | \mathcal{C}_k)p(\mathbf{x}_B | \mathcal{C}_k)p(\mathcal{C}_k) \\ &\propto \frac{p(\mathcal{C}_k | \mathbf{x}_I)p(\mathcal{C}_k | \mathbf{x}_B)}{p(\mathcal{C}_k)}. \end{aligned} \quad (5.27)$$

因此，我们需要类别先验概率 $p(\mathcal{C}_k)$ ，我们可以很容易地从每个类别的数据点比例中估计出来，然后我们需要进行归一化

图5.9癌症治疗问题的混淆矩阵，其中行对应真实类别，列对应我们的决策标准所做出的类别分配。矩阵元素显示了真阴性、假阳性、假阴性和真阳性的数量。

$$\text{正常} \begin{pmatrix} N_{TN} & N_{FP} \\ N_{FN} & N_{TP} \end{pmatrix}$$

第11.2.3节 得到的后验概率之和为一。特定的条件独立假设 (5.26) 是朴素贝叶斯模型的一个例子。请注意，在这个模型下，联合边际分布 $p(\mathbf{x}_I, \mathbf{x}_B)$ 通常不会分解。我们将在后面的章节中看到如何构建不需要条件独立假设 (5.26) 的数据组合模型。使用输出概率而不是决策的模型的另一个优点是，它们可以很容易地相对于任何可调参数（例如多项式回归示例中的权重系数）进行微分，这使得它们可以使用基于梯度的优化方法进行联合组合和训练。

第7章

5.2.5 分类器准确性

分类器性能的最简单度量是正确分类的测试集点的比例。然而，我们已经看到，不同类型的错误可能会有不同的后果，如通过损失矩阵所表达的那样，因此我们通常并不只是希望最小化错误分类的数量。通过改变决策边界的位置，我们可以在不同类型的错误之间进行权衡，例如以最小化预期损失为目标。因为这是一个非常重要的概念，我们将引入一些定义和术语，以便更好地描述分类器的性能。

第2.1.1节

我们将再次考虑癌症筛查的例子。对于每个接受检测的人，都有一个关于他们是否患有癌症的“真实标签”，同时也有分类器做出的预测。如果对于某个特定的人，分类器预测其患有癌症，而这实际上就是真实标签，那么这个预测就被称为真阳性。然而，如果这个人实际上并没有患癌症，那么这就是假阳性。同样地，如果分类器预测一个人没有患癌症，且这是正确的，那么这个预测就被称为真阴性，否则就是假阴性。假阳性也被称为一类错误，而假阴性则被称为二类错误。如果 N 是参加测试的总人数，那么 N_{TP} 是真阳性的数量， N_{FP} 是假阳性的数量， N_{TN} 是真阴性的数量， N_{FN} 是假阴性的数量，其中

$$N = N_{TP} + N_{FP} + N_{TN} + N_{FN}. \quad (5.28)$$

这可以表示为一个混淆矩阵，如图5.9所示。然后，通过正确分类的比例来衡量的准确率由下式给出

$$\text{Accuracy} = \frac{N_{TP} + N_{TN}}{N_{TP} + N_{FP} + N_{TN} + N_{FN}}. \quad (5.29)$$

我们可以看到，如果存在严重的类别不平衡，准确率可能会产生误导。例如，在我们的癌症筛查示例中，每1000人中只有1人患有癌症，一个简单地判定没有人患癌症的朴素分类器将达到99.9%的准确率，但却完全没有用处。

可以根据这些数字定义其他几个量，其中最常遇到的是

$$\text{Precision} = \frac{N_{\text{TP}}}{N_{\text{TP}} + N_{\text{FP}}} \quad (5.30)$$

$$\text{Recall} = \frac{N_{\text{TP}}}{N_{\text{TP}} + N_{\text{FN}}} \quad (5.31)$$

$$\text{False positive rate} = \frac{N_{\text{FP}}}{N_{\text{FP}} + N_{\text{TN}}} \quad (5.32)$$

$$\text{False discovery rate} = \frac{N_{\text{FP}}}{N_{\text{FP}} + N_{\text{TP}}} \quad (5.33)$$

在我们的癌症筛查示例中，精确率表示检测结果为阳性的人确实患有癌症的概率估计，而召回率是指患有癌症的人被检测正确的概率估计。误报率是指正常的人被分类为患有癌症的概率估计，而错误发现率表示检测呈阳性但实际上没有癌症的人的比例。

通过改变决策边界的位置，我们可以改变两种错误之间的权衡。为了理解这种权衡，我们重新审视图5.5，但现在我们按照图5.10所示标记各个区域。我们可以将标记的区域与各种真率和假率联系如下：

$$N_{\text{FP}}/N = E \quad (5.34)$$

$$N_{\text{TP}}/N = D + E \quad (5.35)$$

$$N_{\text{FN}}/N = B + C \quad (5.36)$$

$$N_{\text{TN}}/N = A + C \quad (5.37)$$

我们在其中隐含地考虑了极限 $N \rightarrow \infty$ ，以便能够将观测次数与概率联系起来。

5.2.6 ROC曲线

概率分类器会输出后验概率，通过设置阈值可将其转换为决策。随着阈值的变化，我们可以以增加2类错误为代价来减少1类错误，反之亦然。为了更好地理解这种权衡，绘制接收器操作特性曲线或ROC曲线

(Fawcett, 2006) 是很有用的，这个名字源于测量雷达接收器性能的程序。这是真阳性率与假阳性率的关系图，如图5.11所示。

随着图5.10中的决策边界从 $-\infty$ 移动到 ∞ ，ROC曲线被描绘出来，然后可以通过在 y 轴上绘制正确检测癌症的累积比例与在 x 轴上错误检测的累积比例来生成。

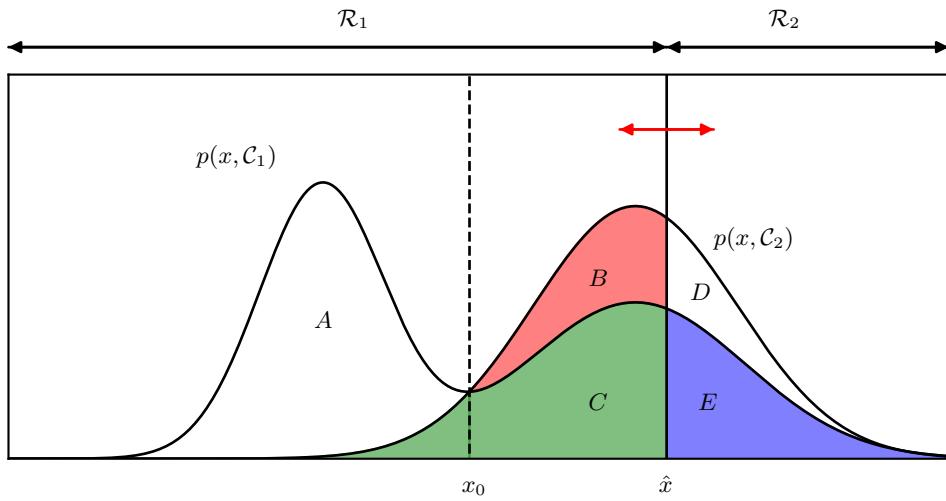


图5.10与图5.5一样，各个区域都有标注。在癌症分类问题中，区域 \mathcal{R}_1 被分配到正常类别，而区域 \mathcal{R}_2 被分配到癌症类别。

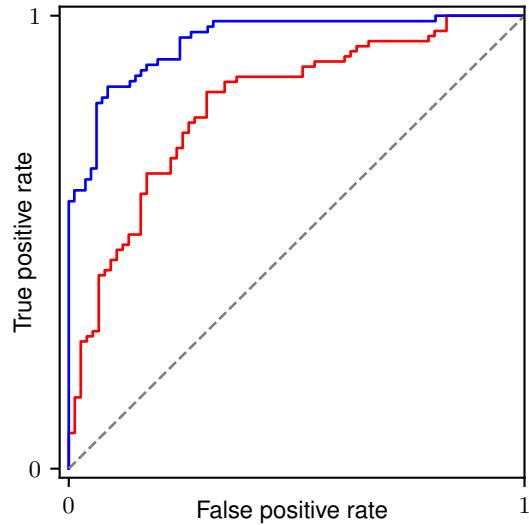
请注意，一个特定的混淆矩阵代表ROC曲线上的一点。最佳的分类器将由ROC图左上角的一个点表示。左下角代表一个简单的分类器，它将每个点都分配到正常类别，因此没有真正的阳性，但也没有假阳性。同样，右上角代表一个将所有内容都分配到癌症类别的分类器，因此没有假阴性，但也没有真正的阴性。在图5.11中，对于任何假阳性率的选择，蓝色曲线代表的分类器都比红色曲线的更好。然而，这样的曲线也有可能交叉，在这种情况下，哪一個更好的选择将取决于操作点的选择。

作为基线，我们可以考虑一个随机分类器，它只是以概率 ρ 将每个数据点分配为癌症，以概率 $1 - \rho$ 分配为正常。当我们改变 ρ 的值时，它将描绘出一条由对角线直线给出的ROC曲线，如图5.11所示。任何低于对角线的分类器的性能都比随机猜测差。

有时，用一个单一的数字来表征整个ROC曲线是很有用的。一种方法是测量曲线下的面积 (AUC)。AUC值为0.5表示随机猜测，而值为1.0表示完美分类器。

另一种度量是 F 分数，它是精确率和召回率的几何平均值，因此定义为

图5.11 接收者操作特征（ROC）曲线是真阳性率与假阳性率的关系图，它刻画了分类问题中第一类错误和第二类错误之间的权衡。上面的蓝色曲线表示比下面的红色曲线更好的分类器。这里的虚线曲线表示简单随机分类器的性能。



$$F = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (5.38)$$

$$= \frac{2N_{\text{TP}}}{2N_{\text{TP}} + N_{\text{FP}} + N_{\text{FN}}}. \quad (5.39)$$

当然，我们也可以将图5.9中的混淆矩阵与图5.6中的损失矩阵相结合，通过逐点相乘元素并对所得乘积求和来计算预期损失。

虽然ROC曲线可以扩展到两个以上的类别，但随着类别的数量增加，它很快就变得繁琐起来。

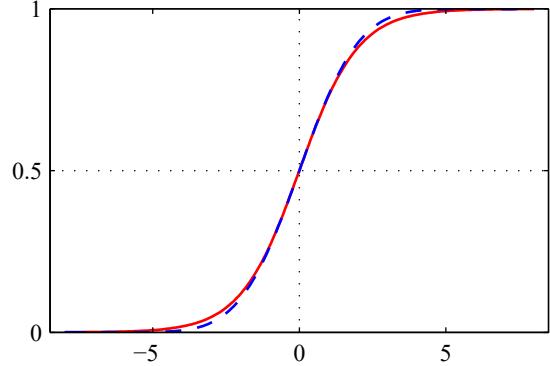
5.3. 生成式分类器

5.2.4节

接下来我们转向分类的概率视角，并展示具有线性决策边界的模型是如何从关于数据分布的简单假设中产生的。我们已经讨论了分类的判别式方法和生成式方法之间的区别。在这里，我们将采用一种生成式方法，在这种方法中，我们对类条件密度 $p(\mathbf{x} | C_k)$ 以及类先验概率 $p(C_k)$ 进行建模，然后通过贝叶斯定理使用这些来计算后验概率 $p(C_k | \mathbf{x})$ 。

首先，考虑具有两个类别的问题。类的后验概率

图5.12 由(5.42)定义的逻辑Sigmoid函数 $\sigma(a)$ 的绘图，以红色显示，以及缩放后的概率单位函数 $\Phi(\lambda a)$ ，对于 $\lambda^2 = \pi/8$ ，以蓝色虚线显示，其中 $\Phi(a)$ 由(5.86)定义。选择缩放因子 $\pi/8$ ，使得两条曲线的导数在 $a = 0$ 处相等。



\mathcal{C}_1 可以写成

$$\begin{aligned} p(\mathcal{C}_1|\mathbf{x}) &= \frac{p(\mathbf{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_1)p(\mathcal{C}_1) + p(\mathbf{x}|\mathcal{C}_2)p(\mathcal{C}_2)} \\ &= \frac{1}{1 + \exp(-a)} = \sigma(a) \end{aligned} \quad (5.40)$$

其中我们定义了

$$a = \ln \frac{p(\mathbf{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_2)p(\mathcal{C}_2)} \quad (5.41)$$

并且 $\sigma(a)$ 是由

$$\sigma(a) = \frac{1}{1 + \exp(-a)}, \quad (5.42)$$

定义的逻辑Sigmoid函数，它绘制在图5.12中。术语“Sigmoid”表示S形。这种类型的函数有时也称为“挤压函数”，因为它将整个实轴映射到一个有限区间。逻辑Sigmoid在前面的章节中已经出现过，并且在许多分类算法中起着重要作用。它满足以下对称性质：

$$\sigma(-a) = 1 - \sigma(a) \quad (5.43)$$

这很容易验证。逻辑 sigmoid 函数的反函数由下式给出

$$a = \ln \left(\frac{\sigma}{1 - \sigma} \right) \quad (5.44)$$

并且被称为 logit 函数。它表示两个类别的概率 $\ln [p(\mathcal{C}_1 | \mathbf{x})/p(\mathcal{C}_2 | \mathbf{x})]$ 之比的对数，也称为对数几率。注意，在(5.40)中，我们只是将后验概率改写为等效形式，因此逻辑 sigmoid 函数的出现可能看起来有些人为。

然而，如果 $a(\mathbf{x})$ 具有受限的函数形式，它将具有重要意义。我们很快会考虑 $a(\mathbf{x})$ 是 \mathbf{x} 的线性函数的情况，在这种情况下，后验概率由广义线性模型控制。

如果有 $K > 2$ 类，我们有

$$\begin{aligned} p(\mathcal{C}_k | \mathbf{x}) &= \frac{p(\mathbf{x} | \mathcal{C}_k) p(\mathcal{C}_k)}{\sum_j p(\mathbf{x} | \mathcal{C}_j) p(\mathcal{C}_j)} \\ &= \frac{\exp(a_k)}{\sum_j \exp(a_j)}, \end{aligned} \quad (5.45)$$

这被称为归一化指数，并且可以被视为逻辑Sigmoid函数的多类推广。这里，量 a_k 由下式定义

$$a_k = \ln(p(\mathbf{x} | \mathcal{C}_k) p(\mathcal{C}_k)). \quad (5.46)$$

归一化指数也被称为softmax函数，因为它表示“max”函数的平滑版本，因为，如果对于所有 $j \neq k$ ， $a_k \gg a_j$ ，那么 $p(\mathcal{C}_k | \mathbf{x}) \approx 1$ ，并且 $p(\mathcal{C}_j | \mathbf{x}) \approx 0$ 。

我们现在研究为类条件密度选择特定形式的后果，首先看连续输入变量 \mathbf{x} ，然后简要讨论离散输入。

5.3.1 连续输入

我们假设类条件密度是高斯分布的。然后我们将探索后验概率的结果形式。首先，我们假设所有类共享相同的协方差矩阵 Σ 。因此，类 \mathcal{C}_k 的密度由下式给出

$$p(\mathbf{x} | \mathcal{C}_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right\}. \quad (5.47)$$

首先，假设我们有两个类。根据(5.40)和(5.41)，我们有

$$p(\mathcal{C}_1 | \mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x} + w_0) \quad (5.48)$$

其中我们定义了

$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \quad (5.49)$$

$$w_0 = -\frac{1}{2} \boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1 + \frac{1}{2} \boldsymbol{\mu}_2^T \Sigma^{-1} \boldsymbol{\mu}_2 + \ln \frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}. \quad (5.50)$$

我们看到高斯密度指数中 \mathbf{x} 的二次项已经抵消（由于共同协方差矩阵的假设），导致逻辑 sigmoid 函数的自变量中 \mathbf{x} 的线性函数。图5.13给出了二维输入空间 \mathbf{x} 的这一结果。

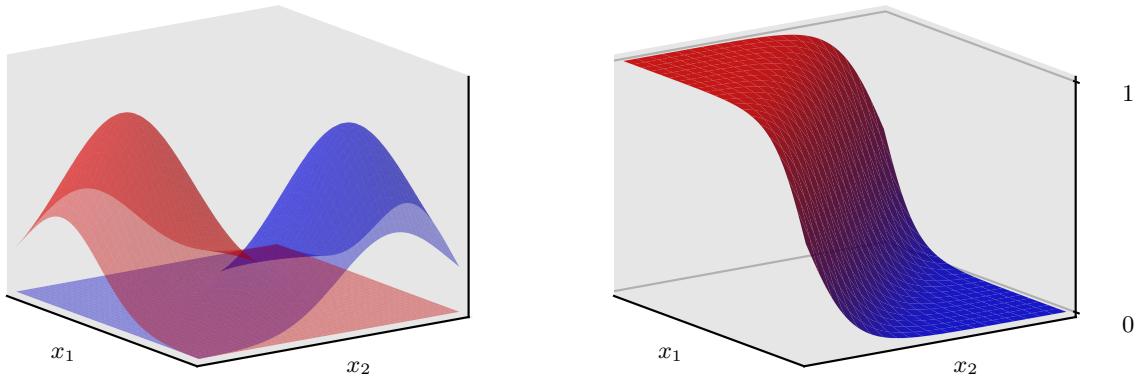


图5.13 左侧的图展示了两个类别（分别用红色和蓝色表示）的类条件密度。右侧是相应的后验概率 $p(\mathcal{C}_1 | \mathbf{x})$ ，它由 \mathbf{x} 的线性函数的逻辑 sigmoid 给出。右侧图中的表面使用由 $p(\mathcal{C}_1 | \mathbf{x})$ 给出的红色墨水比例和由 $p(\mathcal{C}_2 | \mathbf{x}) = 1 - p(\mathcal{C}_1 | \mathbf{x})$ 给出的蓝色墨水比例进行着色。

得到的决策边界对应于后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 为常数的曲面，因此将由 \mathbf{x} 的线性函数给出，所以决策边界在输入空间中是线性的。先验概率 $p(\mathcal{C}_k)$ 仅通过偏差参数 w_0 进入，因此先验概率的变化会使决策边界以及更一般地使后验概率恒定的平行轮廓产生平行移动。

对于 K 类的一般情况，后验概率由(5.45)给出，其中，根据(5.46)和(5.47)，我们有

$$a_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0} \quad (5.51)$$

其中我们定义了

$$\mathbf{w}_k = \Sigma^{-1} \boldsymbol{\mu}_k \quad (5.52)$$

$$w_{k0} = -\frac{1}{2} \boldsymbol{\mu}_k^T \Sigma^{-1} \boldsymbol{\mu}_k + \ln p(\mathcal{C}_k). \quad (5.53)$$

我们发现，由于共享协方差导致二次项的抵消， $a_k(\mathbf{x})$ 再次成为 \mathbf{x} 的线性函数。对应最小误分类率的最终决策边界将在两个后验概率（两个最大的）相等时出现，因此将由 \mathbf{x} 的线性函数定义。因此，我们再次得到一个广义线性模型。

如果我们放宽共享协方差矩阵的假设，允许每个类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 有自己的协方差矩阵 \sum_k ，那么早期的抵消将不再发生，我们将得到 \mathbf{x} 的二次函数，从而产生二次判别式。线性和二次决策边界如图5.14所示。

5.3.2 最大似然解

一旦我们为类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 指定了参数化函数形式，那么我们就可以使用最大似然法来确定参数的值，以及先验类概率 $p(\mathcal{C}_k)$ 。

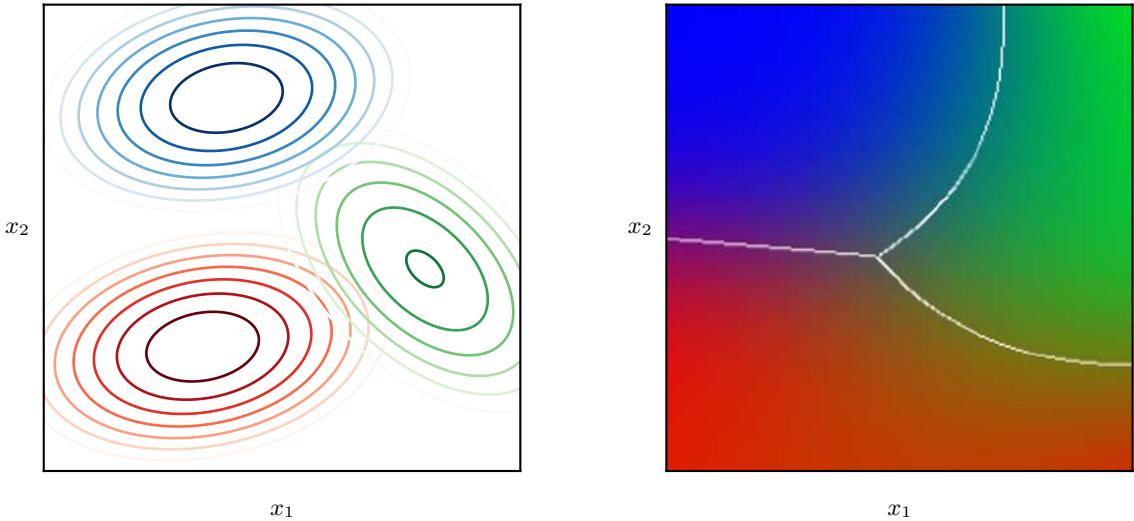


图5.14 左侧的图展示了三个类别的类条件密度，每个类别都具有高斯分布，分别用红色、绿色和蓝色表示，其中红色和蓝色类别具有相同的协方差矩阵。右侧的图展示了相应的后验概率，图像上的每个点都使用与三个相应类别的后验概率对应的红色、蓝色和绿色墨水比例进行着色。还展示了决策边界。请注意，具有相同协方差矩阵的红色和蓝色类别之间的边界是线性的，而其他类别对之间的边界是二次的。

这需要一个数据集，该数据集包含 \mathbf{x} 的观测值及其相应的类别标签。

首先，假设我们有两个类别，每个类别都有一个具有共享协方差矩阵的高斯类条件密度，并且假设我们有一个数据集 $\{\mathbf{x}_n, t_n\}$ ，其中 $n = 1, \dots, N$ 。这里 $t_n = 1$ 表示类别 \mathcal{C}_1 ， $t_n = 0$ 表示类别 \mathcal{C}_2 。我们将先验类概率记为 $p(\mathcal{C}_1) = \pi$ ，使得 $p(\mathcal{C}_2) = 1 - \pi$ 。对于来自类别 \mathcal{C}_1 的数据点 \mathbf{x}_n ，我们有 $t_n = 1$ ，因此

$$p(\mathbf{x}_n, \mathcal{C}_1) = p(\mathcal{C}_1)p(\mathbf{x}_n | \mathcal{C}_1) = \pi \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}).$$

同样对于类别 \mathcal{C}_2 ，我们有 $t_n = 0$ ，因此

$$p(\mathbf{x}_n, \mathcal{C}_2) = p(\mathcal{C}_2)p(\mathbf{x}_n | \mathcal{C}_2) = (1 - \pi) \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_2, \boldsymbol{\Sigma}).$$

因此，似然函数由下式给出

$$p(\mathbf{t}, \mathbf{X} | \pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \prod_{n=1}^N [\pi \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_1, \boldsymbol{\Sigma})]^{t_n} [(1 - \pi) \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_2, \boldsymbol{\Sigma})]^{1-t_n} \quad (5.54)$$

其中 $\mathbf{t} = (t_1, \dots, t_N)^T$ 。与通常情况一样，最大化似然函数的对数会很方便。首先考虑关于 π 的最大化。

对数似然函数中依赖于 π 的项为

$$\sum_{n=1}^N \{t_n \ln \pi + (1 - t_n) \ln(1 - \pi)\}. \quad (5.55)$$

将关于 π 的导数设为零并重新整理，我们得到

$$\pi = \frac{1}{N} \sum_{n=1}^N t_n = \frac{N_1}{N} = \frac{N_1}{N_1 + N_2} \quad (5.56)$$

其中 N_1 表示类别 \mathcal{C}_1 中的数据点总数， N_2 表示类别 \mathcal{C}_2 中的数据点总数。因此， π 的最大似然估计就是类别 \mathcal{C}_1 中数据点的比例，正如预期的那样。这个结果很容易推广到多类别情况，在多类别情况下，与类别 \mathcal{C}_k 相关的先验概率的最大似然估计同样由分配给该类别的训练集数据点的比例给出。

练习5.13

现在考虑关于 μ_1 的最大化。同样，我们可以从对数似然函数中挑出那些依赖于 μ_1 的项：

$$\sum_{n=1}^N t_n \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) = -\frac{1}{2} \sum_{n=1}^N t_n (\mathbf{x}_n - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_1) + \text{const.} \quad (5.57)$$

将关于 μ_1 的导数设为零并重新整理，我们得到

$$\boldsymbol{\mu}_1 = \frac{1}{N_1} \sum_{n=1}^N t_n \mathbf{x}_n, \quad (5.58)$$

这仅仅是分配给类别 \mathcal{C}_1 的所有输入向量 \mathbf{x}_n 的均值。通过类似的论证， μ_2 的相应结果由下式给出

$$\boldsymbol{\mu}_2 = \frac{1}{N_2} \sum_{n=1}^N (1 - t_n) \mathbf{x}_n, \quad (5.59)$$

这同样是分配给类别 \mathcal{C}_2 的所有输入向量 \mathbf{x}_n 的均值。

最后，考虑共享协方差矩阵 $\boldsymbol{\Sigma}$ 的最大似然解。挑出对数似然函数中依赖于 $\boldsymbol{\Sigma}$ 的项，我们有

$$\begin{aligned} & -\frac{1}{2} \sum_{n=1}^N t_n \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^N t_n (\mathbf{x}_n - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_1) \\ & - \frac{1}{2} \sum_{n=1}^N (1 - t_n) \ln |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{n=1}^N (1 - t_n) (\mathbf{x}_n - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_2) \\ & = -\frac{N}{2} \ln |\boldsymbol{\Sigma}| - \frac{N}{2} \text{Tr} \{ \boldsymbol{\Sigma}^{-1} \mathbf{S} \} \end{aligned} \quad (5.60)$$

我们已定义的地方

$$\mathbf{S} = \frac{N_1}{N} \mathbf{S}_1 + \frac{N_2}{N} \mathbf{S}_2 \quad (5.61)$$

$$\mathbf{S}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \boldsymbol{\mu}_1)(\mathbf{x}_n - \boldsymbol{\mu}_1)^T \quad (5.62)$$

$$\mathbf{S}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \boldsymbol{\mu}_2)(\mathbf{x}_n - \boldsymbol{\mu}_2)^T. \quad (5.63)$$

使用高斯分布最大似然解的标准结果，我们看到 $\sum = \mathbf{S}$ ，它分别表示与两个类别中每一个相关联的协方差矩阵的加权平均值。

练习 5.14
5.1.4 节

11.2.3 节

练习 5.16

此结果很容易扩展到 K 类问题，以获得相应的最大似然解，其中每个类条件密度是具有共享协方差矩阵的高斯分布。请注意，将高斯分布拟合到类别的方法对异常值不鲁棒，因为高斯分布的最大似然估计不鲁棒。

5.3.3 离散特征

现在让我们考虑离散特征值 x_i 。为了简单起见，我们首先来看二元特征值 $x_i \in \{0, 1\}$ ，并在稍后讨论对更一般离散特征的扩展。如果有 D 个输入，那么一个一般的分布将对应于每个类别的 2^D 个数字的表格，并且有 $2^D - 1$ 个独立变量（由于求和约束）。因为这会随着特征数量呈指数增长，所以我们可以寻求一种更受限的表示形式。在这里，我们将做出朴素贝叶斯假设，即特征值被视为独立的，并以类别 \mathcal{C}_k 为条件。因此，我们有如下形式的类条件分布

$$p(\mathbf{x}|\mathcal{C}_k) = \prod_{i=1}^D \mu_{ki}^{x_i} (1 - \mu_{ki})^{1-x_i}, \quad (5.64)$$

其中每个类包含 D 个独立参数。将其代入(5.46)式，得到

$$a_k(\mathbf{x}) = \sum_{i=1}^D \{x_i \ln \mu_{ki} + (1 - x_i) \ln (1 - \mu_{ki})\} + \ln p(\mathcal{C}_k), \quad (5.65)$$

这些又是输入值 x_i 的线性函数。对于 $K = 2$ 类，我们也可以考虑由(5.40)给出的逻辑 sigmoid 公式。对于取 $L > 2$ 状态的离散变量，也能得到类似的结果。

5.3.4 指数族

如我们所见，对于高斯分布和离散输入，后验类概率由具有逻辑 sigmoid ($K = 2$) 的广义线性模型给出

第3.4节

2类) 或softmax ($K \geq 2$ classes) 激活函数。这些是通过假设类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 是由以下给出的指数分布族子集中的成员而获得的更一般结果的特殊情况

$$p(\mathbf{x} | \boldsymbol{\lambda}_k, s) = \frac{1}{s} h\left(\frac{1}{s}\mathbf{x}\right) g(\boldsymbol{\lambda}_k) \exp\left\{\frac{1}{s}\boldsymbol{\lambda}_k^T \mathbf{x}\right\}. \quad (5.66)$$

这里的缩放参数 s 在所有类中共享。

对于两类问题，我们将这个类条件密度的表达式代入(5.41)，我们看到后验类概率再次由作用于线性函数 $a(\mathbf{x})$ 的逻辑 sigmoid 给出，其由下式给出

$$a(\mathbf{x}) = (\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)^T \mathbf{x} + \ln g(\boldsymbol{\lambda}_1) - \ln g(\boldsymbol{\lambda}_2) + \ln p(\mathcal{C}_1) - \ln p(\mathcal{C}_2). \quad (5.67)$$

类似地，对于 K 类问题，我们将类条件密度表达式代入(5.46) 得到

$$a_k(\mathbf{x}) = \boldsymbol{\lambda}_k^T \mathbf{x} + \ln g(\boldsymbol{\lambda}_k) + \ln p(\mathcal{C}_k) \quad (5.68)$$

所以它同样是 \mathbf{x} 的线性函数。

5.4. 判别式分类器

对于二分类问题，我们已经看到，对于指数族中广泛选择的类条件分布 $p(\mathbf{x} | \mathcal{C}_k)$ ，类 \mathcal{C}_1 的后验概率可以写成作用于 \mathbf{x} 的线性函数的逻辑 sigmoid 函数。类似地，对于多分类情况，类 \mathcal{C}_k 的后验概率由 \mathbf{x} 的线性函数的 softmax 变换给出。对于类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 的特定选择，我们使用最大似然法来确定密度的参数以及类先验 $p(\mathcal{C}_k)$ ，然后使用贝叶斯定理来找到后验类概率。这代表了生成建模的一个例子，因为我们可以采用这样一个模型，并通过从边际分布 $p(\mathbf{x})$ 或任何类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 中抽取 \mathbf{x} 的值来生成合成数据。

然而，另一种方法是明确使用广义线性模型的函数形式，并通过最大似然法直接确定其参数。在这种直接方法中，我们最大化通过条件分布 $p(\mathcal{C}_k | \mathbf{x})$ 定义的似然函数，这代表了一种判别式概率建模形式。判别式方法的一个优点是，通常需要确定的可学习参数会更少，正如我们即将看到的。它还可能带来更好的预测性能，特别是当类条件密度的假设形式对真实分布的近似很差时。

第4章

5.4.1 激活函数

在线性回归中，模型预测 $y(\mathbf{x}, \mathbf{w})$ 由参数的线性函数给出

$$y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{x} + w_0, \quad (5.69)$$

其在 $(-\infty, \infty)$ 范围内给出连续值输出。然而，对于分类问题，我们希望预测离散的类别标签，或者更一般地，预测位于 $(0, 1)$ 范围内的后验概率。为了实现这一点，我们考虑对该模型进行推广，其中我们使用非线性函数 $f(\cdot)$ 对 \mathbf{w} 和 w_0 的线性函数进行变换，以便

$$y(\mathbf{x}, \mathbf{w}) = f(\mathbf{w}^T \mathbf{x} + w_0). \quad (5.70)$$

在机器学习文献中， $f(\cdot)$ 被称为激活函数，而在统计学文献中其反函数被称为链接函数。决策面对应于 $y(\mathbf{x}) = 1$ 为常数，从而 $\mathbf{w}^T \mathbf{x} = 1$ 为常数，因此决策面是 \mathbf{x} 的线性函数，即使函数 $f(\cdot)$ 是非线性的。出于这个原因，由(5.70)描述的模型类别被称为广义线性模型 (McCullagh和Nelder, 1989)。然而，与用于回归的模型不同，由于非线性函数 $f(\cdot)$ ，它们在参数上不再是线性的。这将导致比线性回归模型更复杂的分析和计算特性。尽管如此，与后续章节将研究的更加灵活的非线性模型相比，这些模型仍然相对简单。

5.4.2 固定基函数

在本章到目前为止，我们已经考虑了直接处理原始输入向量 \mathbf{x} 的分类模型。然而，如果我们首先使用基函数向量 $\phi(\mathbf{x})$ 对输入进行固定的非线性变换，那么所有这些算法同样适用。在特征空间 ϕ 中得到的决策边界将是线性的，并且这些对应于原始 \mathbf{x} 空间中的非线性决策边界，如图5.15所示。在特征空间 $\phi(\mathbf{x})$ 中线性可分的类在原始观测空间 \mathbf{x} 中不一定是线性可分的。

请注意，正如我们在回归线性模型的讨论中一样，通常将其中一个基函数设置为常数，比如 $\phi_0(\mathbf{x}) = 1$ ，这样相应的参数 w_0 就起到偏差的作用。

对于许多具有实际意义的问题，在 \mathbf{x} 空间中，类条件密度 $p(\mathbf{x} | \mathcal{C}_k)$ 之间存在显著重叠。这对应于后验概率 $p(\mathcal{C}_k | \mathbf{x})$ ，对于至少某些 \mathbf{x} 值，这些后验概率不为0或1。在这种情况下，通过准确地对后验概率进行建模，然后应用标准决策理论来获得最优解。请注意，非线性变换 $\phi(\mathbf{x})$ 不能消除这种类重叠，尽管它们可以增加重叠程度或在原始观测空间中不存在重叠的地方产生重叠。然而，合适的非线性选择可以使对后验概率进行建模的过程更容易。然而，这种固定基函数模型有重要的局限性，这些将在后面的章节中通过允许基函数本身适应数据来解决。

第5.2节

第6.1节

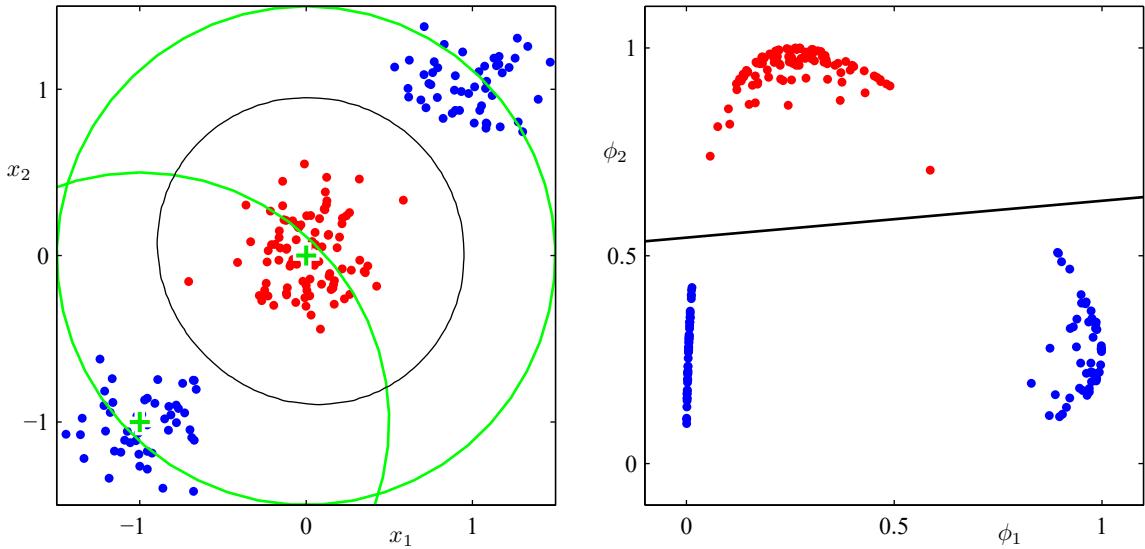


图5.15线性分类模型中非线性基函数的作用说明。左图展示了原始输入空间 (x_1, x_2) 以及来自两个类别（标记为红色和蓝色）的数据点。在这个空间中定义了两个“高斯”基函数 $\phi_1(\mathbf{x})$ 和 $\phi_2(\mathbf{x})$ ，其中心由绿色十字表示，轮廓由绿色圆圈表示。右图展示了相应的特征空间 (ϕ_1, ϕ_2) 以及由5.4.3节讨论的形式的逻辑回归模型给出的线性决策边界。这对应于原始输入空间中的非线性决策边界，如左图中的黑色曲线所示。

5.4.3 逻辑回归

我们首先考虑二分类问题。在5.3节对生成式方法的讨论中，我们看到在相当一般的假设下，类别 \mathcal{C}_1 的后验概率可以写成作用于特征向量 ϕ 的线性函数的逻辑 sigmoid 函数，使得

$$p(\mathcal{C}_1|\phi) = y(\phi) = \sigma(\mathbf{w}^T \phi) \quad (5.71)$$

其中 $p(\mathcal{C}_2|\phi) = 1 - p(\mathcal{C}_1|\phi)$ 。这里 $\sigma(\cdot)$ 是由(5.42)定义的逻辑 sigmoid 函数。在统计学的术语中，这个模型被称为逻辑回归，尽管应该强调的是，这是一个用于分类的模型，而不是用于连续变量的模型。

对于一个 M 维特征空间 ϕ ，该模型有 M 个可调整参数。相比之下，如果我们使用最大似然法拟合高斯类条件密度，我们将使用 $2M$ 个参数表示均值， $M(M+1)/2$ 个参数表示（共享的）协方差矩阵。再加上类先验概率 $p(\mathcal{C}_1)$ ，这总共给出了 $M(M+5)/2 + 1$ 个参数，该参数数量随 M 呈二次增长，这与逻辑回归中参数数量对 M 的线性依赖形成对比。对于较大的 M 值，直接使用逻辑回归模型具有明显优势。

我们现在使用最大似然法来确定逻辑回归模型的参数。为此，我们将利用逻辑Sigmoid函数的导数，它可以方便地用Sigmoid函数本身来表示：

练习5.18

$$\frac{d\sigma}{da} = \sigma(1 - \sigma). \quad (5.72)$$

对于数据集 $\{\phi_n, t_n\}$ ，其中 $\phi_n = \phi(\mathbf{x}_n)$ 且 $t_n \in \{0, 1\}$ ，以及 $n = 1, \dots, N$ ，似然函数可以写成

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^N y_n^{t_n} \{1 - y_n\}^{1-t_n} \quad (5.73)$$

其中 $\mathbf{t} = (t_1, \dots, t_N)^T$ 和 $y_n = p(\mathcal{C}_1 | \phi_n)$ 。像往常一样，我们可以通过取似然的负对数来定义一个误差函数，这就得到了交叉熵误差函数：

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^N \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\} \quad (5.74)$$

练习5.19

其中 $y_n = \sigma(a_n)$ 和 $a_n = \mathbf{w}^T \phi_n$ 。对误差函数关于 \mathbf{w} 求梯度，我们得到

$$\nabla E(\mathbf{w}) = \sum_{n=1}^N (y_n - t_n) \phi_n \quad (5.75)$$

我们在此处使用了(5.72)。我们看到，涉及逻辑斯蒂Sigmoid函数导数的因子已被消去，从而得到对数似然梯度的简化形式。特别地，数据点 n 对梯度的贡献由目标值与模型预测之间的“误差” $y_n - t_n$ 乘以基函数向量 ϕ_n 给出。此外，与(4.12)比较表明，这与线性回归模型的平方和误差函数的梯度形式完全相同。

第4.1.3节

第7章

最大似然解对应于 $\nabla E(\mathbf{w}) = 0$ 。然而，从(5.75)我们可以看出，由于 $y(\cdot)$ 中的非线性，这不再对应于一组线性方程，因此该方程没有闭式解。找到最大似然解的一种方法是使用随机梯度下降，其中 ∇E_n 是(5.75)右侧的第 n 项。随机梯度下降将是训练后续章节中讨论的高度非线性神经网络的主要方法。然而，最大似然方程只是“稍微”非线性，实际上由(5.71)定义模型的误差函数(5.74)是参数的凸函数，这使得可以使用一种称为迭代加权最小二乘法或IRLS的简单算法来最小化误差函数(Bishop, 2006)。然而，这不容易推广到更复杂的模型，如深度神经网络。

练习5.20

请注意，对于线性可分的数据集，最大似然法可能会出现严重的过拟合。出现这种情况的原因是，当对应于 $\sigma = 0.5$ （等同于 $\mathbf{w}^T \phi = 0$ ）的超平面将两类分开且 \mathbf{w} 的幅度趋于无穷大时，就会出现最大似然解。在这种情况下，逻辑Sigmoid函数在特征空间中变得无限陡峭，对应于一个Heaviside阶跃函数，以至于来自每个类 k 的每个训练点都被赋予一个后验概率 $p(C_k | \mathbf{x}) = 1$ 。此外，通常存在这样的解的连续统，因为任何分离超平面在训练数据点处都会产生相同的后验概率。最大似然法无法区分一个这样的解优于另一个解，并且在实践中找到的解将取决于优化算法的选择和参数初始化。请注意，即使与模型中的参数数量相比数据点数量很大，只要训练数据集是线性可分的，问题就会出现。可以通过向误差函数添加正则化项来避免奇异性。

第9章

5.4.4 多类别逻辑回归

第5.3节

在我们对多类别分类的生成模型的讨论中，我们已经看到，对于指数族中的一大类分布，后验概率由特征变量的线性函数的softmax变换给出，因此

$$p(C_k | \phi) = y_k(\phi) = \frac{\exp(a_k)}{\sum_j \exp(a_j)} \quad (5.76)$$

其中预激活值 a_k 由下式给出

$$a_k = \mathbf{w}_k^T \phi. \quad (5.77)$$

在那里，我们使用最大似然法分别确定类别条件密度和类别先验，然后使用贝叶斯定理找到相应的后验概率，从而隐式地确定参数 $\{\mathbf{w}_k\}$ 。在这里，我们考虑使用最大似然法直接确定该模型的参数 $\{\mathbf{w}_k\}$ 。为此，我们需要 y_k 关于所有预激活值 a_j 的导数。这些导数由下式给出

$$\frac{\partial y_k}{\partial a_j} = y_k(I_{kj} - y_j) \quad (5.78)$$

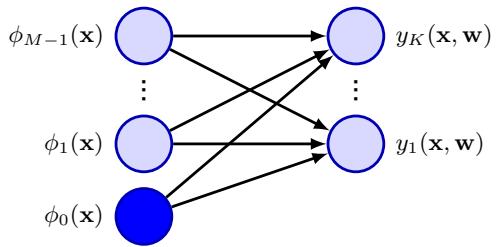
其中 I_{kj} 是单位矩阵的元素。

接下来我们写下似然函数。使用 K 中1-of编码方案来做这件事最为简便，在该方案中，属于类别 C_k 的特征向量 ϕ_n 的目标向量 \mathbf{t}_n 是一个二进制向量，除了等于1的元素 k 外，所有元素均为零。似然函数由下式给出

$$p(\mathbf{T} | \mathbf{w}_1, \dots, \mathbf{w}_K) = \prod_{n=1}^N \prod_{k=1}^K p(C_k | \phi_n)^{t_{nk}} = \prod_{n=1}^N \prod_{k=1}^K y_{nk}^{t_{nk}} \quad (5.79)$$

练习5.21

图5.16 多类线性分类模型表示为具有单层连接的神经网络。每个基函数由一个节点表示，实心节点表示“偏差”基函数 ϕ_0 ，而每个输出 y_1, \dots, y_N 也由一个节点表示。节点之间的链接表示相应的权重和偏差参数。



其中 $y_{nk} = y_k(\phi_n)$ ，并且 \mathbf{T} 是具有元素 t_{nk} 的目标变量的 $N \times K$ 矩阵。取负对数则得到

$$E(\mathbf{w}_1, \dots, \mathbf{w}_K) = -\ln p(\mathbf{T}|\mathbf{w}_1, \dots, \mathbf{w}_K) = -\sum_{n=1}^N \sum_{k=1}^K t_{nk} \ln y_{nk}, \quad (5.80)$$

这就是多类分类问题的交叉熵误差函数。

我们现在求误差函数关于参数向量 \mathbf{w}_j 之一的梯度。利用softmax函数导数的结果(5.78)，我们得到

练习5.22

第7章

我们在此处使用了 $\sum_k t_{nk} = 1$ 。同样，我们可以通过随机梯度下降来优化参数。

再一次，我们看到梯度呈现出与线性模型的平方和误差函数以及逻辑回归模型的交叉熵误差相同的形式，即误差 $(y_{nj} - t_{nj})$ 与基函数激活值 ϕ_n 的乘积。这些是我们稍后将探讨的更一般结果的示例。

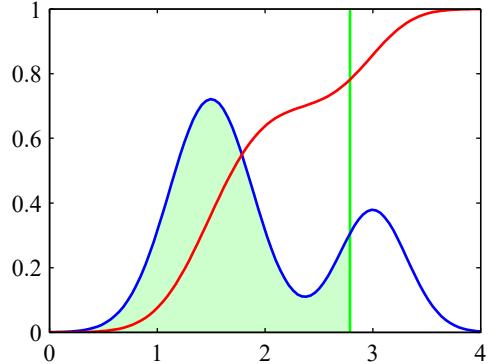
5.4.6节

线性分类模型可以表示为如图5.16所示的单层神经网络。如果我们考虑误差函数相对于权重 w_{ik} 的导数，该权重将基函数 $\phi_i(\mathbf{x})$ 与输出单元 t_k 相连，我们从(5.81)可得

$$\frac{\partial E(\mathbf{w}_1, \dots, \mathbf{w}_K)}{\partial w_{ij}} = \sum_{n=1}^N (y_{nk} - t_{nk}) \phi_i(\mathbf{x}_n). \quad (5.82)$$

将此与图5.16进行比较，我们可以看到，对于每个数据点 n ，该梯度呈现为权重链接输入端基函数的输出形式，而在输出端则带有“误差” $(y_{nk} - t_{nk})$ 。

图5.17 概率密度 $p(\theta)$ 的示意图示例，在此示例中由蓝色曲线表示，由两个高斯分布的混合给出，以及其累积分布函数 $f(a)$ ，由红色曲线表示。请注意，蓝色曲线在任何点的值，如垂直绿线所示的点，对应于红色曲线在同一点的斜率。相反，此时红色曲线的值对应于阴影绿色区域所示的蓝色曲线下方的面积。在随机阈值模型中，如果 $a = \mathbf{w}^T \phi$ 的值超过阈值，则类别标签取值为 $t = 1$ ，否则取值为 $t = 0$ 。这等同于由累积分布函数 $f(a)$ 给出的激活函数。



5.4.5 概率单位回归

我们已经看到，对于由指数族描述的广泛的类条件分布，得到的后验类概率由对特征变量的线性函数进行逻辑（或softmax）变换给出。然而，并非所有类条件密度的选择都会导致后验概率有如此简单的形式，这表明探索其他类型的判别概率模型可能是值得的。再次考虑两类情况，仍在广义线性模型的框架内，那么

$$p(t=1|a) = f(a) \quad (5.83)$$

其中 $a = \mathbf{w}^T \phi$ ，且 $f(\cdot)$ 是激活函数。

激发对链接函数进行替代选择的一种方法是考虑一个噪声阈值模型，如下所述。对于每个输入 ϕ_n ，我们评估 $a_n = \mathbf{w}^T \phi_n$ ，然后根据以下方式设置目标值

$$\begin{cases} t_n = 1, & \text{if } a_n \geq \theta, \\ t_n = 0, & \text{otherwise.} \end{cases} \quad (5.84)$$

如果 θ 的值从概率密度 $p(\theta)$ 中抽取，那么相应的激活函数将由累积分布函数给出

$$f(a) = \int_{-\infty}^a p(\theta) d\theta \quad (5.85)$$

如图5.17所示。

作为一个具体例子，假设密度 $p(\theta)$ 由均值为零、方差为单位1的高斯分布给出。相应的累积分布函数由下式给出

$$\Phi(a) = \int_{-\infty}^a \mathcal{N}(\theta|0, 1) d\theta, \quad (5.86)$$

这被称为概率单位函数。它呈 S 形，并在图 5.12 中与逻辑 S 形函数进行了比较。请注意，使用具有一般均值和方差的高斯分布不会改变模型，因为这相当于对线性系数 w 进行重新缩放。许多数值计算软件包可以计算由下式定义的密切相关的函数

$$\text{erf}(a) = \frac{2}{\sqrt{\pi}} \int_0^a \exp(-\theta^2/2) d\theta \quad (5.87)$$

练习5.23 被称为 erf 函数或误差函数（不要与机器学习模型的误差函数混淆）。它与概率单位函数的关系为

$$\Phi(a) = \frac{1}{2} \left\{ 1 + \frac{1}{\sqrt{2}} \text{erf}(a) \right\}. \quad (5.88)$$

基于概率单位激活函数的广义线性模型被称为概率单位回归。我们可以通过直接扩展前面讨论的思想，使用最大似然法来确定该模型的参数。在实践中，使用概率单位回归得到的结果往往与逻辑回归的结果相似。

在实际应用中可能出现的一个问题是异常值问题，例如，异常值可能通过测量输入向量 x 时的误差或目标值 t 的错误标注而产生。由于这些点可能位于理想决策边界的错误一侧很远的地方，它们会严重扭曲分类器。在这方面，逻辑回归和概率单位回归模型的表现有所不同，因为逻辑 Sigmoid 函数的尾部对于 $|x| \rightarrow \infty$ 渐近地像 $\exp(-x)$ 一样衰减，而对于概率单位激活函数，它们像 $\exp(-x^2)$ 一样衰减，因此概率单位模型对异常值可能更敏感得多。

5.4.6 典范链接函数

对于具有高斯噪声分布的线性回归模型，对应于负对数似然的误差函数由式(4.11)给出。如果我们对来自数据点 n 的误差函数贡献关于参数向量 w 求导，其形式为“误差” $y_n - t_n$ 乘以特征向量 ϕ_n ，其中 $y_n = w^T \phi_n$ 。类似地，对于逻辑 sigmoid 激活函数和交叉熵误差函数(5.74)的组合，以及对于具有多类交叉熵误差函数(5.80)的 softmax 激活函数，我们再次得到这种相同的简单形式。我们现在表明，这是假设目标变量的条件分布来自指数族以及对称为典范链接函数的激活函数进行相应选择的一般结果。

我们再次利用指数族分布的受限形式 (3.169)。请注意，这里我们将指数族分布的假设应用于目标变量 t ，这与第5.3.4节中我们将其应用于输入向量 x 的情况相反。因此，我们考虑形式为目标变量的条件分布

$$p(t|\eta, s) = \frac{1}{s} h\left(\frac{t}{s}\right) g(\eta) \exp\left\{\frac{\eta t}{s}\right\}. \quad (5.89)$$

使用与得出结果(3.172)相同的论证思路，我们看到 t 的条件均值（我们用 y 表示）由下式给出

$$y \equiv \mathbb{E}[t|\eta] = -s \frac{d}{d\eta} \ln g(\eta). \quad (5.90)$$

因此， y 和 η 必定相关，我们通过 $\eta = \psi(y)$ 来表示这种关系。

遵循Nelder和Wedderburn(1972)的方法，我们将广义线性模型定义为这样一种模型，其中 y 是输入（或特征）变量线性组合的非线性函数，使得

$$y = f(\mathbf{w}^T \phi) \quad (5.91)$$

其中 $f(\cdot)$ 在机器学习文献中被称为激活函数，而 $f^{-1}(\cdot)$ 在统计学中被称为链接函数。

现在考虑该模型的对数似然函数，作为 η 的函数，它由下式给出

$$\ln p(\mathbf{t}|\eta, s) = \sum_{n=1}^N \ln p(t_n|\eta, s) = \sum_{n=1}^N \left\{ \ln g(\eta_n) + \frac{\eta_n t_n}{s} \right\} + \text{const} \quad (5.92)$$

我们假设所有观测值共享一个共同的尺度参数（例如，对于高斯分布，它对应于噪声方差），因此 s 与 n 无关。关于模型参数 \mathbf{w} 的对数似然函数的导数由下式给出

$$\begin{aligned} \nabla_{\mathbf{w}} \ln p(\mathbf{t}|\eta, s) &= \sum_{n=1}^N \left\{ \frac{d}{d\eta_n} \ln g(\eta_n) + \frac{t_n}{s} \right\} \frac{d\eta_n}{dy_n} \frac{dy_n}{da_n} \nabla_{\mathbf{w}} a_n \\ &= \sum_{n=1}^N \frac{1}{s} \{t_n - y_n\} \psi'(y_n) f'(a_n) \phi_n \end{aligned} \quad (5.93)$$

其中 $a_n = \mathbf{w}^T \phi_n$ ，并且我们将 $y_n = f(a_n)$ 与 $\mathbb{E}[t|\eta]$ 的结果(5.90)一起使用。现在我们看到，如果我们为链接函数 $f^{-1}(y)$ 选择由下式给出的特定形式，将会有相当大的简化

$$f^{-1}(y) = \psi(y), \quad (5.94)$$

这给出了 $f(\psi(y)) = y$ ，因此得到 $f'(\psi)\psi'(y) = 1$ 。此外，由于 $a = f^{-1}(y)$ ，我们有 $a = \psi$ ，因此得到 $f'(a)\psi'(y) = 1$ 。在这种情况下，误差函数的梯度简化为

$$\nabla \ln E(\mathbf{w}) = \frac{1}{s} \sum_{n=1}^N \{y_n - t_n\} \phi_n. \quad (5.95)$$

我们已经看到，误差函数的选择与输出单元激活函数的选择之间存在自然的配对关系。尽管我们是在单层网络模型的背景下得出这个结果的，但同样的考虑也适用于后面章节中讨论的深度神经网络。

练习

5.1 (*) 考虑一个具有 K 个类别的分类问题以及一个使用 1-of- K 二进制编码方案的目标向量 \mathbf{t} 。证明条件期望 $\mathbb{E}[\mathbf{t} | \mathbf{x}]$ 由后验概率 $p(\mathcal{C}_k | \mathbf{x})$ 给出。

5.2 (**) 给定一组数据点 $\{\mathbf{x}_n\}$ ，我们可以将凸包定义为所有由以下给出的点 \mathbf{x} 的集合

$$\mathbf{x} = \sum_n \alpha_n \mathbf{x}_n \quad (5.96)$$

其中 $\alpha_n \geq 0$ 和 $\sum_n \alpha_n = 1$ 。考虑第二组点 $\{\mathbf{y}_n\}$ 及其对应的凸包。根据定义，如果存在一个向量 $\hat{\mathbf{w}}$ 和一个标量 w_0 ，使得对于所有 \mathbf{x}_n 都有 $\hat{\mathbf{w}}^T \mathbf{x}_n + w_0 > 0$ ，并且对于所有 \mathbf{y}_n 都有 $\hat{\mathbf{w}}^T \mathbf{y}_n + w_0 < 0$ ，那么这两组点将是线性可分的。证明如果它们的凸包相交，那么这两组点不可能是线性可分的，反之，如果它们是线性可分的，那么它们的凸包不相交。

5.3 (**) 考虑最小化平方和误差函数 (5.14)，并假设训练集中的所有目标向量都满足线性约束

$$\mathbf{a}^T \mathbf{t}_n + b = 0 \quad (5.97)$$

其中 \mathbf{t}_n 对应于 (5.14) 中矩阵 \mathbf{T} 的第 n 行。证明由于此约束，最小二乘解 (5.16) 给出的模型预测 $\mathbf{y}(\mathbf{x})$ 的元素也满足此约束，因此

$$\mathbf{a}^T \mathbf{y}(\mathbf{x}) + b = 0. \quad (5.98)$$

为此，假设基函数之一为 $\phi_0(\mathbf{x}) = 1$ ，使得相应的参数 w_0 起到偏差的作用。

5.4 (**) 将练习 5.3 的结果进行扩展，以证明如果目标向量同时满足多个线性约束，那么线性模型的最小二乘预测也将满足相同的约束。

5.5 (*) 使用定义(5.38)，以及(5.30)和(5.31)来推导 F 分数的结果(5.39)。

5.6 (**) 考虑两个非负数字 a 和 b ，并证明，如果 $a \leq b$ ，那么 $a \leq (ab)^{1/2}$ 。利用这个结果证明，如果二类分类问题的决策区域被选择为最小化错误分类的概率，那么这个概率将满足

$$p(\text{mistake}) \leq \int \{p(\mathbf{x}, \mathcal{C}_1)p(\mathbf{x}, \mathcal{C}_2)\}^{1/2} d\mathbf{x}. \quad (5.99)$$

5.7 (*) 给定一个具有元素 L_{kj} 的损失矩阵，如果对于每个 \mathbf{x} ，我们选择使 (5.23) 最小化的类别，那么期望风险将被最小化。

验证当损失矩阵由 $L_{kj} = 1 - I_{kj}$ 给出时，其中 I_{kj} 是单位矩阵的元素，这会简化为选择具有最大后验概率的类别的准则。这种形式的损失矩阵的解释是什么？

5.8 (*) 推导当存在一般损失矩阵和类别的一般先验概率时，使期望损失最小化的准则。

5.9 (*) 考虑以下形式的一组 N 数据点上后验概率的平均值

$$\frac{1}{N} \sum_{n=1}^N p(\mathcal{C}_k | \mathbf{x}_n). \quad (5.100)$$

通过取极限 $N \rightarrow \infty$ ，证明这个量趋近于先验类概率 $p(\mathcal{C}_k)$ 。

5.10 (**) 考虑一个分类问题，其中当来自类 \mathcal{C}_k 的输入向量被分类为属于类 \mathcal{C}_j 时产生的损失由损失矩阵 L_{kj} 给出，并且选择拒绝选项时产生的损失为 λ 。找到能给出最小期望损失的决策准则。验证当损失矩阵由 $L_{kj} = 1 - I_{kj}$ 给出时，这简化为5.2.3节中讨论的拒绝准则。 λ 与拒绝阈值 θ 之间的关系是什么？

5.11 (*) 证明逻辑斯谛Sigmoid函数 (5.42) 满足性质 $\sigma(-a) = 1 - \sigma(a)$ ，并且其反函数由 $\sigma^{-1}(y) = \ln\{y/(1-y)\}$ 给出。

5.12 (*) 使用 (5.40) 和 (5.41)，推导具有高斯密度的两类生成模型中后验类概率的结果 (5.48)，并验证参数 \mathbf{w} 和 w_0 的结果 (5.49) 和 (5.50)。

5.13 (*) 考虑一个用于 K 类的生成式分类模型，该模型由先验类概率 $p(\mathcal{C}_k) = \pi_k$ 和一般类条件密度 $p(\phi | \mathcal{C}_k)$ 定义，其中 ϕ 是输入特征向量。假设我们有一个训练数据集 $\{\phi_n, \mathbf{t}_n\}$ ，其中 $n = 1, \dots, N$ ，并且 \mathbf{t}_n 是长度为 K 的二元目标向量，它使用 K 选1编码方案，使得如果数据点 n 来自类 \mathcal{C}_k ，则其分量为 $t_{nj} = I_{jk}$ 。假设数据点是从该模型中独立抽取的，证明先验概率的最大似然解由下式给出

$$\pi_k = \frac{N_k}{N} \quad (5.101)$$

其中 N_k 是分配给类别 \mathcal{C}_k 的数据点数量。

5.14 (**) 考虑练习5.13中的分类模型，现在假设类条件密度由具有共享协方差矩阵的高斯分布给出，因此

$$p(\phi | \mathcal{C}_k) = \mathcal{N}(\phi | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}). \quad (5.102)$$

证明对于类别 \mathcal{C}_k 的高斯分布均值的最大似然解由下式给出

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N t_{nk} \boldsymbol{\phi}_n, \quad (5.103)$$

它表示分配给类别 \mathcal{C}_k 的那些特征向量的均值。类似地，证明共享协方差矩阵的最大似然解由下式给出

$$\boldsymbol{\Sigma} = \sum_{k=1}^K \frac{N_k}{N} \mathbf{S}_k \quad (5.104)$$

其中

$$\mathbf{S}_k = \frac{1}{N_k} \sum_{n=1}^N t_{nk} (\boldsymbol{\phi}_n - \boldsymbol{\mu}_k) (\boldsymbol{\phi}_n - \boldsymbol{\mu}_k)^T. \quad (5.105)$$

因此， Σ 由与每个类别相关的数据协方差的加权平均值给出，其中加权系数由类别的先验概率给出。

5.15 (**) 推导第5.3.3节中描述的具有离散二元特征的概率朴素贝叶斯分类器的参数 $\{\mu_{ki}\}$ 的最大似然解。

5.16 (**) 考虑一个具有 K 类别的分类问题，对于该问题，特征向量 $\boldsymbol{\phi}$ 有 M 个分量，每个分量可以取 L 个离散状态。设这些分量的值由一种 L 选 1 的二进制编码方案表示。进一步假设，在类别 \mathcal{C}_k 的条件下， $\boldsymbol{\phi}$ 的 M 个分量是独立的，从而使得类别条件密度相对于特征向量分量进行分解。证明由(5.46)给出的量 a_k ，它出现在描述后验类别概率的 softmax 函数的自变量中，是 $\boldsymbol{\phi}$ 的分量的线性函数。注意，这代表了一个朴素贝叶斯模型的例子。

第11.2.3节

5.17 (**) 推导练习5.16中描述的概率朴素贝叶斯分类器的参数的最大似然解。

5.18 (*) 验证由(5.42)定义的逻辑斯谛 Sigmoid 函数的导数的关系式(5.72)。

5.19 (*) 利用逻辑斯谛 Sigmoid 函数导数的结果(5.72)，证明逻辑回归模型的误差函数(5.74)的导数由(5.75)给出。

5.20 (*) 证明对于线性可分数据集，逻辑回归模型的最大似然解是通过找到一个向量 \mathbf{w} 得到的，其决策边界 $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) = 0$ 将类别分开，然后将 \mathbf{w} 的模设为无穷大。

5.21 (*) 证明 softmax 激活函数(5.76)的导数（其中 a_k 由(5.77)定义）由(5.78)给出。

5.22 (*) 使用softmax激活函数导数的结果(5.78), 证明交叉熵误差(5.80)的梯度由(5.81)给出。

5.23 (*) 证明概率单位函数(5.86)和误差函数(5.87)通过(5.88)相关联。

5.24 (**) 假设我们希望用一个缩放后的概率单位函数 $\Phi(\lambda a)$ 来近似由 (5.42) 式定义的逻辑斯谛 S 形函数 $\sigma(a)$, 其中 $\Phi(a)$ 由 (5.86) 式定义。证明如果选择 λ 使得这两个函数在 $a = 0$ 处的导数相等, 那么 $\lambda^2 = \pi/8$ 。

深度学习



6

Deep Neural Networks

第4章
第5章

近年来，神经网络已成为目前实际应用中最重要的机器学习技术，因此我们在本书中花费了大量篇幅来研究它们。前面的章节已经奠定了许多我们所需的基础。特别是，我们已经看到，由固定非线性基函数的线性组合构成的线性回归模型可以表示为具有单层权重和偏差参数的神经网络。同样，基于基函数线性组合的分类模型也可以看作是单层神经网络。在本章开始讨论更复杂的多层网络之前，这些内容使我们能够引入几个重要概念。

给定足够数量的适当选择的基函数，这样的线性模型可以将从输入到输出的任何给定非线性变换近似到任何所需的精度，因此似乎足以处理任何实际应用。

第6.3.6节

然而，这些模型存在一些严重的局限性，因此我们将通过探讨这些局限性并理解为何有必要使用从数据中学习得到的基函数来开始我们对神经网络的讨论。这自然地引出了对具有多层可学习参数的神经网络的讨论。这些被称为前馈网络或多层感知器。我们还将讨论拥有许多这样的处理层的好处，从而引出目前主导机器学习领域的深度神经网络的关键概念。

6.1. 固定基函数的局限性

第5章

用于分类的线性基函数模型基于基函数 $\phi_j(\mathbf{x})$ 的线性组合，并采用以下形式

$$y(\mathbf{x}, \mathbf{w}) = f \left(\sum_{j=1}^M w_j \phi_j(\mathbf{x}) + w_0 \right) \quad (6.1)$$

第4章

其中 $f(\cdot)$ 是一个非线性输出激活函数。用于回归的线性模型采用相同的形式，但 $f(\cdot)$ 被恒等函数取代。这些模型允许使用任意一组非线性基函数 $\{\phi_i(\mathbf{x})\}$ ，并且由于这些基函数的通用性，此类模型原则上可以为任何回归或分类问题提供解决方案。从某种微不足道的意义上来说这是正确的，因为如果其中一个基函数对应于所需的输入到输出变换，那么可学习的线性层只需将此基函数的值复制到模型的输出即可。

更一般地，我们期望一个足够大且丰富的基函数集能够使任何期望的函数被逼近到任意精度。因此，这样的线性模型似乎构成了机器学习中解决问题的通用框架。不幸的是，线性模型存在一些重大缺点，这些缺点源于基函数 $\phi_j(\mathbf{x})$ 是固定的且与训练数据无关的假设。为了理解这些局限性，我们首先来看随着输入变量数量增加时线性模型的行为。

6.1.1 维度诅咒

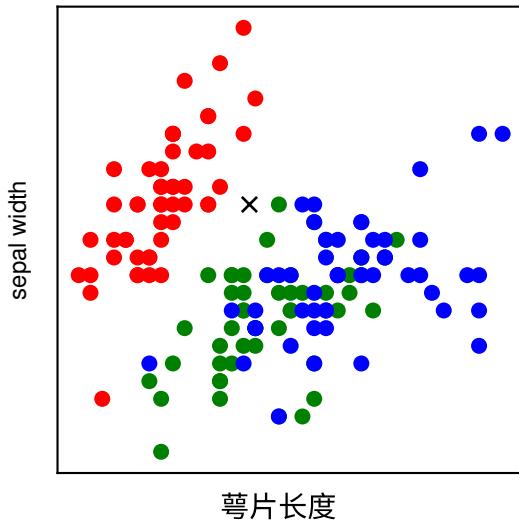
第1.2节

考虑一个针对单个输入变量的简单回归模型，其形式为阶数为 M 的多项式

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \dots + w_M x^M \quad (6.2)$$

让我们看看如果增加输入数量会发生什么。如果我们有 D 个输入变量 $\{x_1, \dots, x_D\}$ ，那么一个系数最高为 3 阶的一般多项式将具有以下形式

图6.1鸢尾花数据图，其中红色、绿色和蓝色点分别表示三种鸢尾花，坐标轴分别表示萼片长度和宽度的测量值。我们的目标是对一个新的测试点进行分类，例如由 \times 表示的那个点。



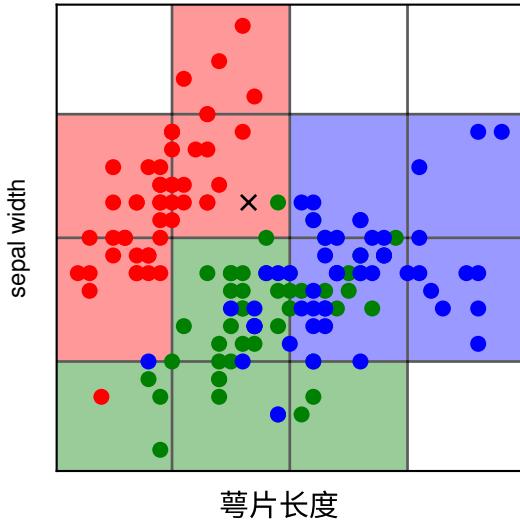
$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^D w_i x_i + \sum_{i=1}^D \sum_{j=1}^D w_{ij} x_i x_j + \sum_{i=1}^D \sum_{j=1}^D \sum_{k=1}^D w_{ijk} x_i x_j x_k. \quad (6.3)$$

随着 D 增加，独立系数数量的增长为 $\mathcal{O}(D^3)$ ，而对于阶数为 M 的多项式，系数数量的增长为 $\mathcal{O}(D^M)$ (Bishop, 2006)。我们看到，在更高维度的空间中，多项式会迅速变得难以处理且实用性不大。

在多维度空间中可能出现的严重困难有时被称为维度诅咒(Bellman, 1961)。它不仅限于多项式回归，实际上相当普遍。考虑使用线性模型解决分类问题。图6.1展示了鸢尾花数据集的数据图，该数据集包含从三种鸢尾花中每种选取的50个观测值。每个观测值有四个变量，分别代表萼片长度、萼片宽度、花瓣长度和花瓣宽度的测量值。对于此示例，我们仅考虑萼片长度和萼片宽度变量。将这150个观测值作为训练数据，我们的目标是通过将一个新的测试点(如图6.1中用十字标记的点)分配到三个类别之一来对其进行分类。我们观察到十字靠近几个红色点，因此我们可能会认为它属于红色类别。然而，附近也有一些绿色点，所以我们可能会认为它反而可能属于绿色类别。它似乎不太可能属于蓝色类别。这里的直觉是，十字的类别应由训练集中的附近点更强烈地确定，而由更远的点较弱地确定，事实证明这种直觉是合理的。

将这种直觉转化为学习算法的一种非常简单的方法是，如图6.2所示，将输入空间划分为规则的单元格。当我们得到一个测试点并希望预测其类别时，我们首先确定它属于哪个单元格，然后找到所有落在同一单元格中的训练数据点。

图6.2 一种解决分类问题的简单方法的示意
图，其中输入空间被划分为单元
格，任何新的测试点被分配到与
测试点位于同一单元格中具有最
多代表的类。正如我们即将看到
的，这种简单的方法有一些严重
的缺点。



预测测试点的类别与在与测试点相同单元格中具有最多训练点的类别相同（平局时随机打破）。我们可以将其视为一个基函数模型，其中每个网格单元格都有一个基函数 $\phi_i(\mathbf{x})$ ，如果 \mathbf{x} 位于网格单元格之外，则简单地返回零，否则返回落在单元格内的训练数据点的多数类。然后，模型的输出由所有基函数的输出之和给出。

这种简单方法存在许多问题，但当我们考虑将其扩展到具有大量输入变量的问题时，其中一个最严重的问题就会变得明显，这些问题对应于更高维度的输入空间。问题的根源如图6.3所示，该图表明，如果我们将空间区域划分为规则单元格，那么这些单元格的数量会随着空间维度呈指数增长。单元格数量呈指数增长带来的挑战是，我们需要指数级大量的训练数据来确保单元格不为空。

图6.3维度诅咒的说明，展示了规
则网格的区域数量如何随着空间
维度 D 呈指数增长。为了清晰起
见，对于 $D = 3$ 仅显示了立方体
区域的一个子集。

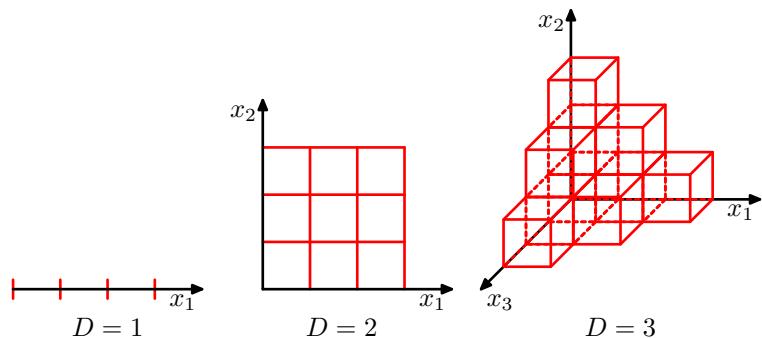
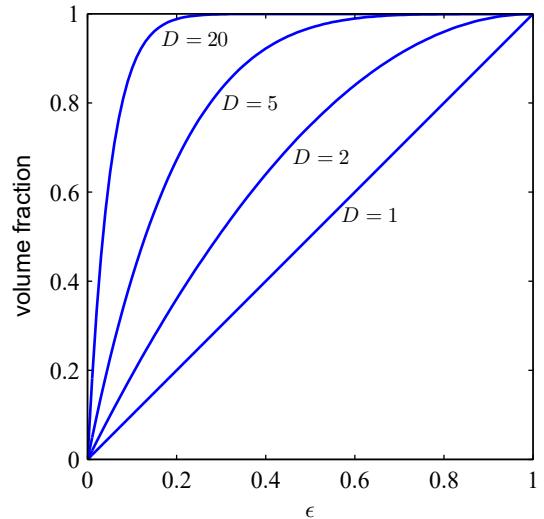


图6.4 对于不同维度 D 值，半径为 $r = 1$ 的超球体位于 $r = 1 - \epsilon$ 到 $r = 1$ 范围内的体积分数图。



我们已经在图6.2中看到，一些单元格不包含训练点。因此，此类单元格中的测试点无法分类。显然，在多于几个变量的空间中应用这种技术是没有希望的。多项式回归示例和鸢尾花数据分分类示例中的困难都源于基函数的选择与要解决的问题无关。如果我们要规避维度灾难，就需要在基函数的选择上更加精细。

6.1.4节

6.1.2 高维空间

然而，首先，我们将更仔细地研究具有更高维度的空间的性质，在这些空间中，我们通过在三维空间中度过的一生所形成的几何直觉可能会严重失效。作为一个简单的例子，考虑在 D 维空间中半径为 $r = 1$ 的超球体，并问半径在 $r = 1 - \epsilon$ 和 $r = 1$ 之间的超球体体积占总体积的比例是多少。我们可以通过注意到在 D 维空间中半径为 r 的超球体的体积 $V_D(r)$ 必须按 r^D 缩放来评估这个比例，所以我们写成

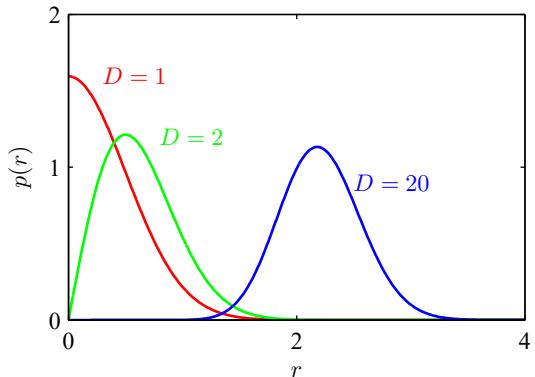
$$V_D(r) = K_D r^D \quad (6.4)$$

练习6.1 其中常数 K_D 仅取决于 D 。因此，所需的比例由下式给出

$$\frac{V_D(1) - V_D(1 - \epsilon)}{V_D(1)} = 1 - (1 - \epsilon)^D, \quad (6.5)$$

在图6.4中，它被绘制为 ϵ 的函数，其中 D 取不同的值。我们看到，对于较大的 D ，即使 ϵ 的值很小，这个分数也趋于1。因此，我们得到了一个显著的结果，即在高维空间中，超球体的大部分体积集中在靠近表面的一个薄壳中！

图6.5 高斯分布的概率密度相对于半径 r 随维度 D 的不同值的变化曲线。在高维空间中，高斯分布的大部分概率质量位于特定半径处的薄壳内。



练习6.3

作为与机器学习直接相关的另一个例子，考虑高维空间中高斯分布的行为。如果我们从笛卡尔坐标转换为极坐标，然后对方向变量进行积分，我们得到密度 $p(r)$ 作为到原点的半径 r 的函数的表达式。因此， $p(r)\delta r$ 是位于半径 r 处厚度为 δr 的薄壳内的概率质量。对于 D 的不同值，该分布绘制在图6.5中，我们看到对于大的 D ，高斯分布的概率质量集中在特定半径处的薄壳内。

在本书中，我们大量使用了涉及一个或两个变量的示例，因为这样能特别容易地以图形方式直观呈现这些空间。然而，读者应注意，并非所有在低维空间中形成的直觉都能推广到涉及多个维度的情况。

最后，尽管我们讨论了维度诅咒，但在高维空间中工作也可能有优势。考虑图6.6所示的情况。我们看到这个数据集，其中每个数据点由一对值 (x_1, x_2) 组成，是线性可分的，但当只观察到 x_1 的值时，类别有很强的重叠。因此，在高维空间中的分类问题要容易得多。

6.1.3 数据流形

对于图6.2中的多项式回归模型和基于网格的分类器，我们发现基函数的数量随着维度迅速增长，这使得这些方法对于涉及甚至几十个变量的应用都不切实际，更不用说在图像处理等应用中经常出现的数百万个输入了。问题在于基函数是提前固定的，不依赖于数据，甚至不依赖于正在解决的特定问题。我们需要找到一种方法来创建针对特定应用进行调整的基函数。

尽管维度诅咒确实给机器学习应用带来了重要问题，但它并不能阻止我们找到适用于高维空间的有效技术。原因之一是实际数据通常会局限于数据空间中具有较低有效维度的区域。

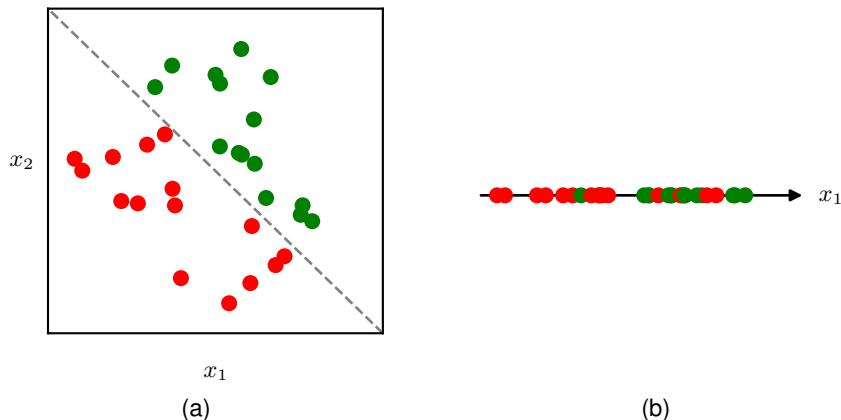
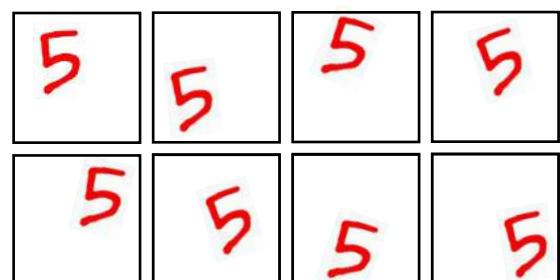


图6.6二维数据集 (x_1, x_2) 的图示，其中用绿色和红色圆圈表示的两类数据点可以被一个线性决策面分开，如(a)所示。然而，如果只测量变量 x_1 ，那么这些类别就不再可分，如(b)所示。

考虑图6.7所示的图像。每个图像都是高维空间中的一个点，其维度由像素数量决定。由于物体可以在图像内的不同垂直和水平位置以及不同方向出现，图像之间存在三个可变自由度，并且一组图像在一阶近似下将位于嵌入高维空间的三维流形上。由于物体位置或方向与像素强度之间的复杂关系，这个流形将是高度非线性的。

事实上，像素数量实际上是图像生成过程的一种人为产物，因为它们代表的是对连续世界的测量。以更高分辨率捕捉同一图像会增加数据空间的维度 D ，但不会改变图像仍存在于三维流形上这一事实。如果我们能将局部基函数与数据流形相关联，而不是与整个高维数据空间相关联，我们可能会预期所需基函数的数量将随流形的维度呈指数增长，而不是随数据空间的维度增长。由于流形的维度通常比数据空间低得多，这代表了一个巨大的改进。

图6.7手写数字图像的示例，这些图像在
数字在图像中的位置以及方
向上有所不同。此数据存在
于高维图像空间内的非线性
三维流形上。



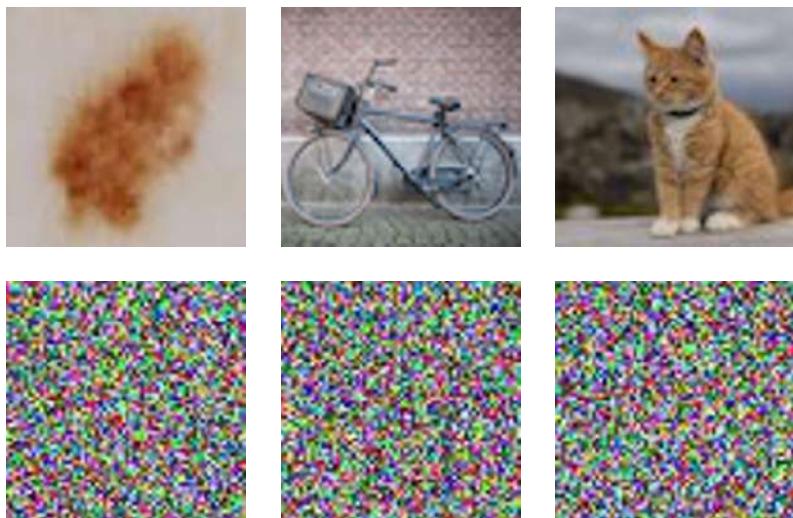


图6.8的上排展示了大小为 64×64 像素的自然图像示例，而下排展示了通过从可能的像素颜色上的均匀概率分布中绘制像素值而获得的相同大小的随机生成图像。

实际上，神经网络学习一组适应于数据流形的基函数。此外，对于特定应用，流形内并非所有方向都可能是重要的。例如，如果我们只想确定图6.7中物体的方向而不是位置，那么流形上只有一个相关自由度而不是三个。神经网络还能够学习流形上哪些方向与预测期望输出相关。

另一种理解真实数据局限于低维流形的方法是考虑生成随机图像的任务。在图6.8中，我们看到了自然图像的示例以及通过从均匀分布中随机独立地对每个像素的红色、绿色和蓝色强度进行采样而生成的具有相同分辨率的合成图像的示例。我们看到，没有一个合成图像看起来像自然图像。原因是这些随机图像缺乏自然图像所表现出的像素之间非常强的相关性。例如，自然图像中两个相邻像素具有相同或非常相似颜色的概率比随机示例中的两个相邻像素要高得多。图6.8中的每个图像都对应于高维空间中的一个点，然而自然图像只覆盖了这个空间的一小部分。

6.1.4 数据依赖基函数

我们已经看到，独立于所解决问题而选择的简单基函数可能会遇到重大限制，特别是在高维空间中。如果我们想在这种情况下使用基函数，那么一种方法是利用专家知识，以针对每个应用程序的特定方式手工构建基函数。

第9.1节

多年来，这一直是机器学习中的主流方法。基函数，通常称为特征，将由领域知识和反复试验的组合来确定。然而，这种方法取得的成功有限，并且被数据驱动的方法所取代，在数据驱动的方法中，基函数是从训练数据中学习的。领域知识在现代机器学习中仍然发挥着作用，但在设计网络架构时，它在更定性的层面上发挥作用，在那里它可以捕获适当的归纳偏差，正如我们将在后面的章节中看到的那样。

由于高维空间中的数据可能局限于低维流形，我们不需要密集填充整个输入空间的基函数，而是可以使用与数据流形本身相关联的基函数。一种方法是为训练集中的每个数据点关联一个基函数，这确保了基函数能自动适应基础数据流形。这种模型的一个例子是径向基函数 (Broomhead和Lowe, 1988)，其特性是每个基函数仅取决于到中心向量的径向距离（通常是欧几里得距离）。如果将基中心选择为输入数据值 $\{\mathbf{x}_n\}$ ，那么每个数据点就有一个基函数 $\phi_n(\mathbf{x})$ ，因此它将捕获整个数据流形。径向基函数的一个典型选择是

$$\phi_n(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_n\|^2}{s^2}\right) \quad (6.6)$$

其中 s 是控制基函数宽度的参数。虽然建立这样一个模型可能很快，但这种技术的一个主要问题是，对于大数据集来说，它在计算上变得难以处理。此外，该模型需要仔细的正则化以避免严重的过拟合。

一种相关的方法称为支持向量机 (SVM)（瓦普尼克, 1995年；朔尔科普夫和斯莫拉, 2002年；毕晓普, 2006年），解决此问题的方式是再次定义以每个训练数据点为中心的基函数，然后在训练过程中自动选择这些基函数的一个子集。结果，所得模型中基函数的有效数量通常远小于训练点的数量，尽管它通常仍然相对较大，并且通常会随着训练集的大小而增加。支持向量机也不会产生概率输出，并且它们自然不能推广到两个以上的类别。诸如径向基函数和支持向量机之类的方法已被深度神经网络所取代，深度神经网络在有效利用非常大的数据集方面要好得多。此外，正如我们稍后将看到的，神经网络能够学习深度层次表示，这对于在更复杂的应用中实现高预测准确性至关重要。

第7章

6.3节

第4章

6.2. 多层网络

在上一节中，我们看到，要将形式为(6.1)的线性模型应用于涉及大规模数据集和高维空间的问题，我们需要找到一组针对要解决的问题进行调整的基函数。神经网络背后的关键思想是选择本身具有可学习参数的基函数 $\phi_j(\mathbf{x})$ ，然后在训练期间允许这些参数与系数 $\{w_j\}$ 一起进行调整。然后，我们使用基于梯度的优化方法（如随机梯度下降）通过最小化误差函数来优化整个模型，其中误差函数是在模型中的所有参数上联合定义的。

当然，构建参数化非线性基函数有很多方法。一个关键要求是，它们必须是其可学习参数的可微函数，以便我们能够应用基于梯度的优化方法。最成功的选择是使用与(6.1)形式相同的基函数，这样每个基函数本身就是输入的线性组合的非线性函数，其中线性组合中的系数是可学习参数。请注意，这种构造显然可以递归扩展，以给出具有多层的层次模型，这构成了深度神经网络的基础。

考虑一个具有两层可学习参数的基本神经网络模型。首先，我们以如下形式构造输入变量 x_1, \dots, x_D 的 M 个线性组合

$$a_j^{(1)} = \sum_{i=1}^D w_{ji}^{(1)} x_i + w_{j0}^{(1)} \quad (6.7)$$

其中 $j = 1, \dots, M$ ，上标(1)表示相应参数处于网络的第一个“层”。我们将参数 $w_{ji}^{(1)}$ 称为权重，参数 $w_{j0}^{(1)}$ 称为偏差，而量 $a_j^{(1)}$ 称为预激活。然后，每个量 a_j 使用可微的非线性激活函数 $h(\cdot)$ 进行变换，以得到

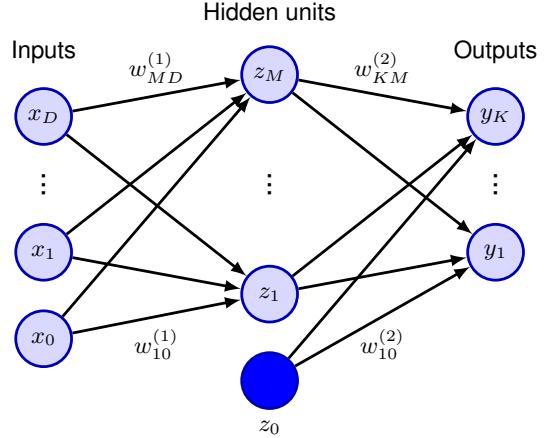
$$z_j^{(1)} = h(a_j^{(1)}), \quad (6.8)$$

它们表示(6.1)中基函数的输出。在神经网络的背景下，这些基函数被称为隐藏单元。我们很快将探讨非线性函数 $h(\cdot)$ 的各种选择，但在此我们注意到，只要导数 $h'(\cdot)$ 可以求值，那么整个网络函数将是可微的。根据(6.1)，这些值再次进行线性组合以得到

$$a_k^{(2)} = \sum_{j=1}^M w_{kj}^{(2)} z_j^{(1)} + w_{k0}^{(2)} \quad (6.9)$$

其中 $k = 1, \dots, K$ ，并且 K 是输出的总数。此变换对应于网络的第二层，并且 $w_{k0}^{(2)}$ 同样是偏置参数。最后， $\{a_k^{(2)}\}$ 使用适当的输出单元激活函数 $f(\cdot)$ 进行变换，以给出一组网络输出 y_k 。

图6.9 两层神经网络的网络图。输入、隐藏和输出变量由节点表示，权重参数由节点之间的链接表示。偏差参数由来自额外输入和隐藏变量 x_0 和 z_0 的链接表示，这些变量本身由实心节点表示。箭头表示前向传播期间信息在网络中的流动方向。



两层神经网络可以用图6.9所示的图表形式表示。

6.2.1 参数矩阵

第4.1.1节

正如我们在讨论线性回归模型时提到的，(6.7) 中的偏差参数可以通过定义一个额外的输入变量 x_0 来吸收到权重参数集中，该变量的值固定为 $x_0 = 1$ ，这样 (6.7) 就变成了

$$a_j = \sum_{i=0}^D w_{ji}^{(1)} x_i. \quad (6.10)$$

我们同样可以将第二层偏差吸收到第二层权重中，这样整个网络函数就变成了

$$y_k(\mathbf{x}, \mathbf{w}) = f \left(\sum_{j=0}^M w_{kj}^{(2)} h \left(\sum_{i=0}^D w_{ji}^{(1)} x_i \right) \right). \quad (6.11)$$

在本书的各个地方，另一种将被证明很方便的表示法是将输入表示为列向量 $\mathbf{x} = (x_1, \dots, x_N)^T$ ，然后将(6.11)中的权重和偏差参数收集到矩阵中，得到

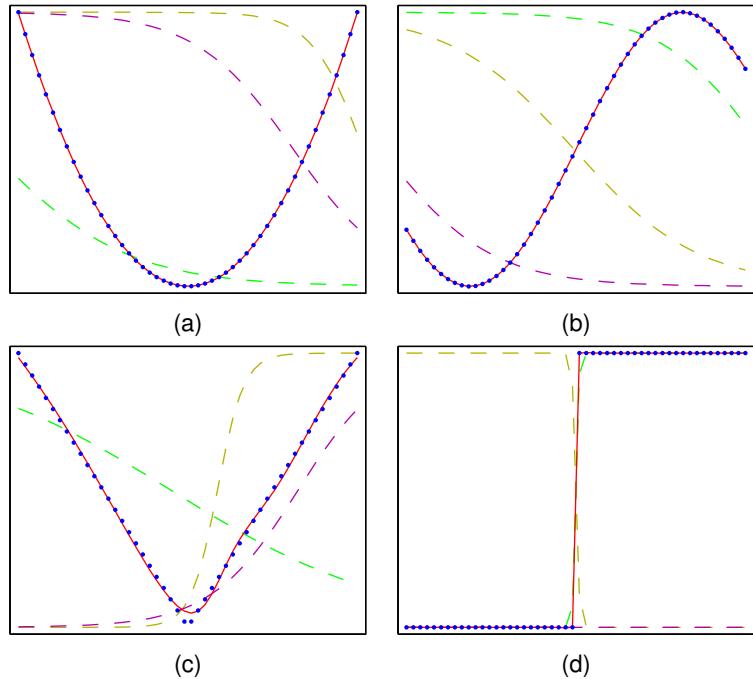
$$\mathbf{y}(\mathbf{x}, \mathbf{w}) = f (\mathbf{W}^{(2)} h (\mathbf{W}^{(1)} \mathbf{x})) \quad (6.12)$$

其中 $f(\cdot)$ 和 $h(\cdot)$ 分别在每个向量元素上进行求值。

6.2.2 通用逼近

图6.10展示了一个两层网络对广泛函数进行建模的能力。该图还展示了各个隐藏单元如何协同工作以逼近最终函数。图6.11展示了隐藏单元在一个简单分类问题中的作用。

图6.10 两层神经网络逼近四种不同函数的能力说明：(a) $f(x) = x^2$ ，(b) $f(x) = \sin(x)$ ，(c)， $f(x) = |x|$ ，以及(d) $f(x) = H(x)$ ，其中 $H(x)$ 是海维赛德阶跃函数。在每种情况下， $N = 50$ 个数据点（显示为蓝色点）已在区间 $(-1, 1)$ 上的 x 中均匀采样，并评估了 $f(x)$ 的相应值。然后使用这些数据点来训练一个具有三个隐藏单元的两层网络，这些隐藏单元具有双曲正切激活函数和线性输出单元。所得网络函数由红色曲线显示，三个隐藏单元的输出由三条虚线曲线显示。



两层前馈网络的逼近特性在20世纪80年代得到了广泛研究，各种定理表明，对于广泛的激活函数，这样的网络可以将定义在 \mathbb{R}^D 连续子集上的任何函数逼近到任意精度（船桥，1989；齐本科，1989；霍尼克、斯汀科姆和怀特，1989；莱什诺等人，1993）。从任何有限维离散空间到任何其他空间的函数也有类似结果。因此，神经网络被称为通用逼近器。

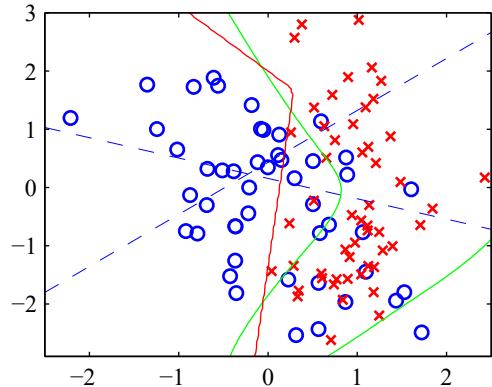
尽管这些定理令人安心，但它们只告诉我们存在一个能表示所需函数的网络。在某些情况下，它们可能需要具有指数级大量隐藏单元的网络。此外，它们对于这样的网络是否能通过学习算法找到没有任何说明。而且，我们稍后会看到，无免费午餐定理表明我们永远无法找到真正通用的机器学习算法。最后，虽然具有两层权重的网络是通用逼近器，但在实际应用中，考虑具有远多于两层且能学习分层内部表示的网络可能会带来巨大的好处。所有这些观点都支持了深度学习的发展趋势。

第9.1.2节

6.2.3 隐藏单元激活函数

我们已经看到，输出单元的激活函数由所建模的分布类型决定。然而，对于隐藏单元，唯一的要求是它们需要是可微的，这就留下了广泛的可能性。

图6.11 一个简单的两类分类问题的解决方案示例，该问题涉及使用具有两个输入、两个带有双曲正切激活函数的隐藏单元以及一个具有逻辑 sigmoid 激活函数的单个输出的神经网络处理合成数据。虚线蓝色线显示了每个隐藏单元的 $z = 0.5$ 轮廓，红线显示了网络的 $y = 0.5$ 决策面。为了进行比较，绿线表示根据用于生成数据的分布计算出的最优决策边界。



在大多数情况下，网络中的所有隐藏单元将被赋予相同的激活函数，尽管原则上没有理由不能在网络的不同部分应用不同的选择。

隐藏单元激活函数最简单的选择是恒等函数，这意味着所有隐藏单元都变为线性的。然而，对于任何这样的网络，我们总能找到一个没有隐藏单元的等效网络。这是因为连续线性变换的组合本身就是一个线性变换，所以它的表示能力不大于单个线性层。但是，如果隐藏单元的数量小于输入或输出单元的数量，那么这样一个网络能够生成的变换就不是从输入到输出最一般的线性变换，因为在隐藏单元处的降维过程中信息会丢失。考虑一个具有 N 个输入、 M 个隐藏单元和 K 个输出的网络，并且所有激活函数都是线性的。这样一个网络有 $M(N+K)$ 个参数，而直接从输入到输出的线性变换会有 NK 个参数。如果 M 相对于 N 或 K （或两者）较小，这会导致一个两层线性网络的参数比直接线性映射的参数少，对应于一个秩亏变换。这种线性单元的“瓶颈”网络对应于一种称为主成分分析的标准数据分析技术。然而，一般来说，使用线性单元的多层网络的兴趣有限，因为这样一个网络计算的整体函数仍然是线性的。

一个简单的非线性可微函数是由逻辑 sigmoid 给出的

$$\sigma(a) = \frac{1}{1 + \exp(-a)}, \quad (6.13)$$

其绘制在图5.12中。这在多层神经网络研究的早期被广泛使用，并且部分受到对生物神经元特性研究的启发。

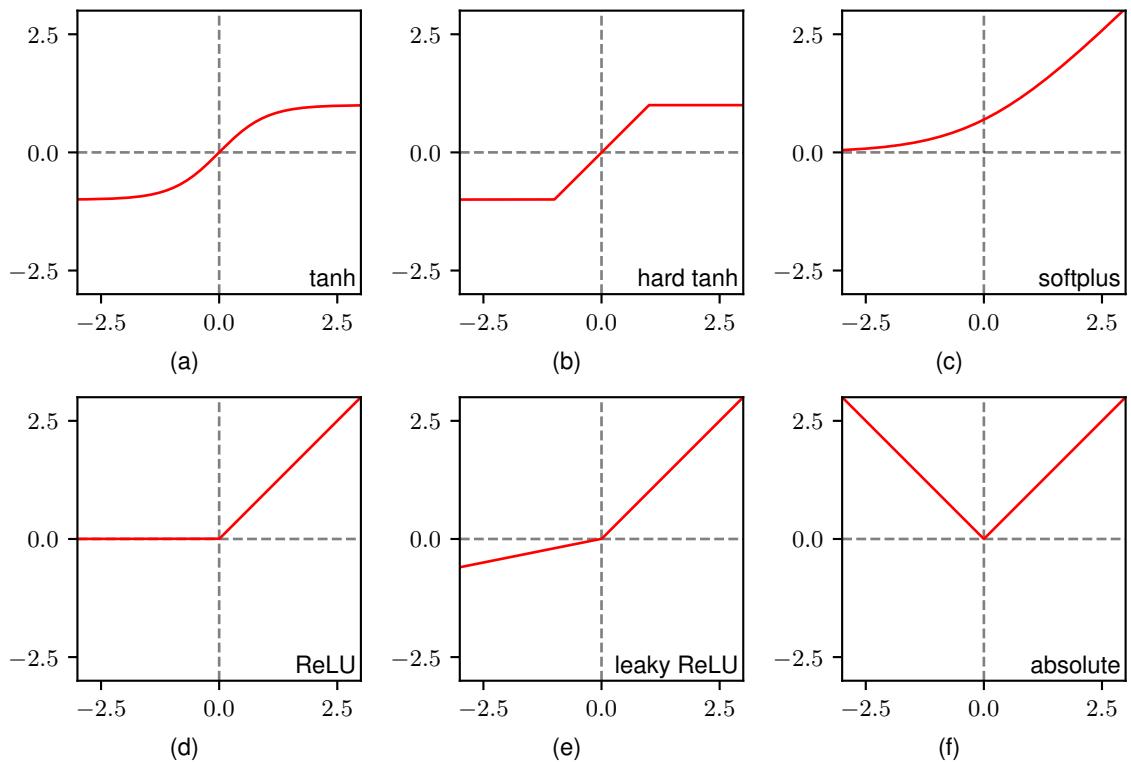


图6.12 各种非线性激活函数。

一个密切相关的函数是tanh，其定义为

$$\tanh(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}, \quad (6.14)$$

其绘制在图6.12(a)中。该函数与逻辑 sigmoid 函数的区别在于其输入和输出值的线性变换，因此对于任何具有逻辑 sigmoid 隐藏单元激活函数的网络，都存在一个具有 tanh 激活函数的等效网络。然而，在训练网络时，它们不一定等效，因为对于基于梯度的优化，需要初始化网络权重和偏差，因此如果改变激活函数，则必须相应地调整初始化方案。tanh 函数的一个“硬”版本（Collobert, 2004）由

$$h(a) = \max(-1, \min(1, a)) \quad (6.15)$$

给出，并绘制在图6.12(b)中。

逻辑 sigmoid 函数和 tanh 激活函数的一个主要缺点是，当输入具有大的正值或大的负值时，梯度会指数级地趋近于零。

练习6.7

我们稍后会讨论这个“梯度消失”问题，但目前我们注意到，通常最好使用具有非零梯度的激活函数，至少当输入取较大正值时是这样。一个这样的选择是由下式给出的softplus激活函数

$$h(a) = \ln(1 + \exp(a)), \quad (6.16)$$

其绘制在图6.12(c)中。对于 $a \gg 1$ ，我们有 $h(a) \simeq a$ ，所以即使激活函数的输入很大且为正时，梯度仍然非零，从而有助于缓解梯度消失问题。

一个更简单的激活函数选择是整流线性单元或ReLU，其定义为

$$h(a) = \max(0, a) \quad (6.17)$$

其绘制在图6.12(d)中。根据经验，这是表现最佳的激活函数之一，并且被广泛使用。注意，严格来说，当 $a = 0$ 时ReLU函数的导数未定义，但在实践中这可以安全地忽略。softplus函数(6.16)可以看作是ReLU的平滑版本，因此有时也称为软ReLU。

练习6.5

尽管ReLU对于正输入值具有非零梯度，但对于负输入则并非如此，这可能意味着在训练期间一些隐藏单元没有接收到“误差信号”。一种旨在避免此问题的ReLU修改版本称为泄漏ReLU，其定义为

$$h(a) = \max(0, a) + \alpha \min(0, a), \quad (6.18)$$

其中 $0 < \alpha < 1$ 。此函数如图6.12(e)所示。与ReLU不同，对于输入值 $a < 0$ ，它具有非零梯度，这确保了有信号驱动训练。此激活函数的一个变体使用 $\alpha = -1$ ，在这种情况下 $h(a) = |a|$ ，如图6.12(f)所示。另一个变体允许每个隐藏单元具有自己的值 α_j ，这可以在网络训练期间通过评估相对于 $\{\alpha_j\}$ 的梯度以及相对于权重和偏差的梯度来学习。

ReLU的引入相比之前的Sigmoid激活函数在训练效率上有了很大提升 (Krizhevsky、Sutskever和Hinton, 2012)。它不仅能让更深的网络得到高效训练，对权重的随机初始化也不那么敏感。它还非常适合低精度实现，比如8位定点数与64位浮点数相比，并且计算评估成本较低。许多实际应用通常将ReLU单元作为默认选择，除非目标是明确探索不同激活函数选择的效果。

6.2.4 权重空间对称性

前馈网络的一个特性是，对于权重向量 w 存多种不同选择，它们都能产生从输入到输出的相同映射函数 (Chen、Lu和Hecht-Nielsen, 1993)。考虑一个如图6.9所示形式的两层网络，其中 M 个隐藏单元具有tanh激活函数且两层全连接。

如果我们改变输入到特定隐藏单元的所有权重和偏差的符号，那么，对于给定的输入数据点，隐藏单元预激活的符号将被反转，因此激活值也将会被反转，因为双曲正切函数是奇函数，所以 $\tanh(-a) = -\tanh(a)$ 。通过改变从该隐藏单元输出的所有权重的符号，可以精确地补偿这种变换。因此，通过改变特定一组权重（和一个偏差）的符号，网络所代表的输入-输出映射函数不变，所以我们找到了两个不同的权重向量，它们产生相同的映射函数。对于 M 个隐藏单元，将有 M 个这样的“符号翻转”对称性，因此，任何给定的权重向量都将是一组 2^M 个等效权重向量中的一个。

同样地，想象一下我们将进入和离开某个特定隐藏单元的所有权重（以及偏差）的值与另一个不同隐藏单元相关联的权重（和偏差）的对应值进行互换。同样，这显然不会改变网络的输入 - 输出映射函数，但它对应于权重向量的不同选择。对于 M 个隐藏单元，任何给定的权重向量将属于与这种互换对称性相关联的 $M \times (M-1) \times \dots \times 2 \times 1 = M!$ 个等效权重向量的集合，对应于隐藏单元的 $M!$ 种不同排序。因此，网络将具有 $M!2^M$ 的整体权重空间对称因子。对于具有两层以上权重的网络，总对称级别将由这些因子的乘积给出，每个隐藏单元层对应一个因子。

事实证明，这些因素解释了权重空间中的所有对称性（可能因权重值的特定选择而产生的偶然对称性除外）。此外，这些对称性的存在并非 \tanh 函数的特殊属性，而是适用于广泛的激活函数 (Kurková 和 Kainen, 1994)。一般来说，权重空间中的这些对称性在实际中影响不大，因为网络训练旨在找到特定的参数设置，而其他等效设置的存在影响不大。然而，当使用贝叶斯方法来评估不同大小网络上的概率分布时，权重空间对称性确实会起作用 (Bishop, 2006)。

6.3. 深度网络

我们通过使线性回归或分类模型的基函数本身由可学习参数控制，推动了神经网络的发展，从而产生了图6.9所示的两层网络模型。多年来，这是使用最广泛的架构，主要是因为事实证明，有效训练超过两层的网络很困难。然而，将神经网络扩展到两层以上，即所谓的深度神经网络，会带来许多优势，我们将很快讨论，并且最近神经网络训练技术的进展对于多层网络是有效的。

第7章

我们可以轻松地将两层网络架构 (6.12) 扩展到任意有限数量 L 的层，其中第 $l = 1, \dots, L$ 层计算以下函数：

$$\mathbf{z}^{(l)} = h^{(l)}(\mathbf{W}^{(l)} \mathbf{z}^{(l-1)}) \quad (6.19)$$

其中 $h^{(l)}$ 表示与层 l 相关联的激活函数，并且 $\mathbf{W}^{(l)}$ 表示权重和偏差参数的相应矩阵。此外， $\mathbf{z}^{(0)} = \mathbf{x}$ 表示输入向量， $\mathbf{z}^{(L)} = \mathbf{y}$ 表示输出向量。

请注意，在文献中，关于此类网络中层数的计数术语存在一些混淆。因此，图6.9中的网络有时被描述为三层网络（计算单元层数并将输入视为单元），有时被描述为单隐藏层网络（计算隐藏单元层数）。我们建议采用一种术语，其中图6.9被称为两层网络，因为对于确定网络属性而言，可学习权重的层数很重要。

我们已经看到，如图6.9所示形式的网络，具有两层可学习参数，具有通用逼近能力。然而，具有两层以上的网络有时可以用比两层网络少得多的参数来表示给定函数。蒙图法尔等人（2014年）表明，网络函数将输入空间划分为数量与网络深度成指数关系，但与隐藏层宽度仅成多项式关系的区域。使用两层网络来表示相同的函数将需要指数数量的隐藏单元。

第10章

6.3.1 分层表示

虽然这是一个有趣的结果，但探索深度神经网络更有说服力的原因是，网络架构编码了一种特定形式的归纳偏差，即输出通过层次表示与输入空间相关。一个很好的例子是图像中的目标识别任务。图像像素与诸如“猫”这样的高级概念之间的关系非常复杂且非线性，对于一个两层网络来说将是一个极具挑战性的问题。然而，深度神经网络可以在早期层学习检测诸如边缘等低级特征，然后在后续层中将这些特征组合起来以形成诸如眼睛或胡须等高级特征，这些高级特征又可以在更后面的层中组合起来以检测猫的存在。这可以被视为一种组合归纳偏差，其中高级对象，如猫，由低级对象，如眼睛组成，而眼睛又由诸如边缘等更高级的元素组成。我们也可以反过来思考这个过程，即从诸如边缘等低级特征开始生成图像，然后将这些特征组合成诸如圆形等简单形状，再将这些形状依次组合成诸如猫等高级对象。在每个阶段，都有许多种组合不同组件的方式，随着深度的增加，可能性的数量呈指数级增长。

6.3.2 分布式表示

神经网络可以利用另一种组合形式，即分布式表示。从概念上讲，隐藏层中的每个单元都可以被视为在网络的该层次上表示一个“特征”，激活值高表示相应特征存在，激活值低表示其不存在。

在给定层中有 M 个单元，这样的网络可以表示 M 种不同的特征。然而，网络有可能学习到一种不同的表示，其中隐藏单元的组合表示特征，从而有可能允许具有 M 个单元的隐藏层表示 2^M 种不同的特征，随着单元数量呈指数增长。例如，考虑一个设计用于处理面部图像的网络。每个特定的面部图像可能有眼镜，也可能没有；可能有帽子，也可能没有；可能有胡须，也可能没有，这导致了八种不同的组合。虽然这可以由八个单元来表示，每个单元在检测到相应组合时“开启”，但也可以用三个单元更紧凑地表示，每个属性一个单元。这些属性可以相互独立存在（尽管从统计角度来看，它们的存在可能在某种程度上相关）。稍后，我们将详细探讨深度学习网络在训练过程中自行发现的各种内部表示。

第10章

第1.1.1节

第19.1节

6.3.3 表示学习

我们可以将深度神经网络的连续层视为对数据进行变换，从而使解决所需任务变得更容易。例如，一个成功学会将皮肤病变分类为良性或恶性的神经网络，必须学会将原始图像数据变换到一个新空间，该空间由最后一层隐藏单元的输出表示，以便网络的最后一层能够区分这两类。最后一层可以看作是一个简单的线性分类器，因此在最后一个隐藏层的表示中，这两类必须由一个线性表面很好地分开。发现一种非线性数据变换，使后续任务更容易解决的这种能力称为表示学习（Bengio、Courville 和 Vincent，2012）。学习到的表示，有时称为嵌入空间，由网络的一个隐藏层的输出给出，这样，任何输入向量，无论是来自训练集还是来自某个新数据集，都可以通过网络的前向传播变换到这个表示中。

表示学习特别强大，因为它使我们能够利用未标记的数据。通常，收集大量未标记的数据很容易，但获取相关标签可能更困难。例如，车辆上的摄像机在车辆在城市中行驶时可以收集大量城市场景的图像，但是对这些图像进行拍摄并识别相关物体，如行人和路标，将需要昂贵且耗时的人工标注。

从未标记的数据中学习称为无监督学习，并且已经开发了许多不同的算法来进行此操作。例如，可以训练神经网络以图像作为输入并创建相同的图像作为输出。为了使这成为一项不平凡的任务，网络可能会使用比图像像素数量少的单元的隐藏层，从而迫使网络学习某种图像压缩。只需要未标记的数据，因为训练集中的每个图像都充当输入向量和目标向量。这样的网络称为自动编码器。

第12章

第1.1.1节

第10章

目标是这种类型的训练将迫使网络为数据发现一些内部表示，这些表示对于解决其他任务（如图像分类）是有用的。

从历史上看，无监督学习在使第一个深度网络（除卷积网络外）成功训练方面发挥了重要作用。网络的每一层首先使用无监督学习进行预训练，然后使用基于梯度的监督训练对整个网络进行进一步训练。后来发现，在适当条件下，可以省略预训练阶段，并且可以纯粹使用监督学习从头开始训练深度网络。

然而，在其他情况下，预训练和表示学习仍然是深度学习的核心。预训练最显著的例子是在自然语言处理中，其中变压器模型在大量文本上进行训练，并且能够学习高度复杂的语言内部表示，这有助于实现令人印象深刻的人类水平及更高水平的各种能力。

6.3.4 迁移学习

为一个特定任务学习到的内部表示可能对相关任务也有用。例如，在一个由日常物体的大型标注数据集上训练的网络可以学习如何将图像表示转换为更适合对物体进行分类的表示。然后，可以使用一个较小的皮肤病变图像标注数据集对网络的最终分类层进行重新训练，以创建一个病变分类器。这是迁移学习的一个例子（Hospedales等人，2021），与仅使用病变图像数据进行训练相比，它可以实现更高的准确率，因为网络可以利用一般自然图像共有的共性。迁移学习如图6.13所示。

一般来说，迁移学习可用于通过使用来自相关任务B的数据来提高在某些训练数据短缺的任务A上的性能，任务B的数据更为丰富。这两个任务应具有相同类型的输入，并且任务之间应存在一些共性，以便从任务B中学到的低级特征或内部表示对任务A有用。当我们研究卷积网络时，我们会发现许多图像处理任务需要与深度神经网络早期层相对应的类似低级特征，而后期层则更专门针对特定任务，这使得此类网络非常适合迁移学习应用。

当任务A的数据非常稀缺时，我们可能只需重新训练网络的最后一层。相比之下，如果有更多的数据点，重新训练多个层是可行的。使用一个任务学习参数，然后将其应用于一个或多个其他任务的过程称为预训练。请注意，对于新任务，与其对整个网络应用随机梯度下降，将新的训练数据一次性通过固定的预训练网络以在新表示中评估训练输入要高效得多。然后可以仅对由最后几层组成的较小网络应用基于迭代梯度的优化。除了将预训练网络用作不同任务的固定预处理器之外，还可以应用微调，即让整个网络适应任务A的数据。

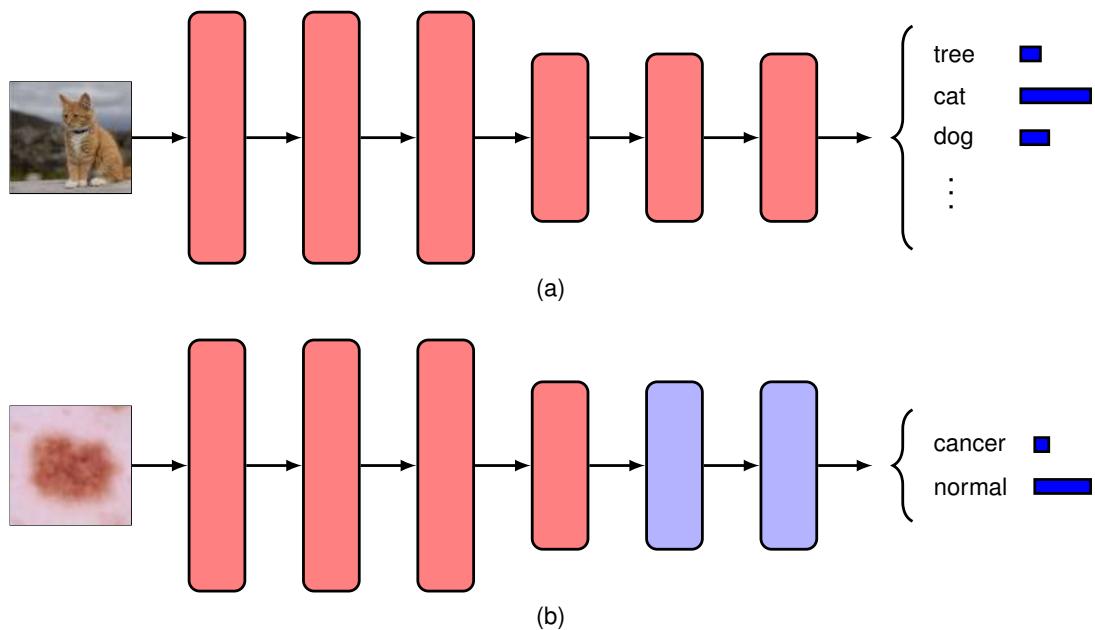


图6.13迁移学习的示意图。(a) 首先在具有丰富数据的任务上训练一个网络，例如自然图像的目标分类。(b) 网络的早期层（红色显示）从第一个任务复制而来，然后网络的最后几层（蓝色显示）在一个新任务上进行重新训练，例如皮肤病变分类，其训练数据较为稀缺。

这通常是通过以非常小的学习率进行有限次数的迭代来完成的，以确保网络不会过度拟合新任务可用的相对较小的数据集。

一种相关的方法是多任务学习 (Caruana, 1997)，其中一个网络同时联合学习多个相关任务。例如，我们可能希望构建一个垃圾邮件过滤器，允许不同用户拥有根据其特定偏好调整的不同分类器。训练数据可能包括许多不同用户的垃圾邮件和非垃圾邮件示例，但任何一个用户的示例数量可能相当有限，因此为每个用户训练一个单独的分类器会得到很差的结果。相反，我们可以合并数据集来训练一个更大的单一网络，例如，该网络可能共享早期层，但在后期层为不同用户具有单独的可学习参数。跨任务共享数据允许网络利用任务之间的共性，从而提高所有用户的准确性。有了大量的训练示例，可以使用具有更多参数的更深层次网络，这同样会导致性能提升。

跨多个任务的学习可以扩展到元学习，元学习也被称为学会学习。多任务学习旨在对一组固定的任务进行预测，而元学习的目标是对训练期间未见过的未来任务进行预测。

这不仅可以通过学习跨任务的共享内部表示来实现，还可以通过学习学习算法本身来实现（霍斯佩达莱斯等人，2021年）。元学习可用于促进例如分类模型在新类别的标记示例非常少的情况下对新类别的泛化。这被称为少样本学习。当仅使用单个标记示例时，它被称为单样本学习。

6.3.5 对比学习

最常见且强大的表示学习方法之一是对比学习（古特曼和海瓦林，2010年；奥尔德、李和维尼尔斯，2018年；陈、科恩布利思等人，2020年）。其理念是学习一种表示，使得某些输入对（称为正例对）在嵌入空间中接近，而其他输入对（称为负例对）则相距较远。直观的理解是，如果我们以语义相似的方式选择正例对，并选择语义不同的负例对，那么我们将学习到一个表示空间，其中相似的输入接近，这使得诸如分类等下游任务变得容易得多。与其他形式的表示学习一样，训练网络的输出通常不直接使用，而是使用较早层的激活来形成嵌入空间。对比学习与大多数其他机器学习任务不同，因为给定输入的误差函数仅相对于其他输入定义，而不是有每个输入的标签或目标输出。

假设我们有一个给定的数据点 \mathbf{x} ，称为锚点，对于该锚点，我们指定了另一个数据点 \mathbf{x}^+ ，它与 \mathbf{x} 一起构成一个正样本对。我们还必须指定一组数据点 $\{\mathbf{x}_1^-, \dots, \mathbf{x}_N^-\}$ ，其中每个数据点都与 \mathbf{x} 构成一个负样本对。现在，我们需要一个损失函数，该函数将奖励 \mathbf{x} 和 \mathbf{x}^+ 的表示之间的接近程度，同时鼓励每对 $\{\mathbf{x}, \mathbf{x}_n^-\}$ 之间有较大的距离。这种函数的一个例子，也是对比学习中最常用的损失函数，称为InfoNCE损失

(Gutmann和Hyvärinen, 2010; Oord、Li和Vinyals, 2018)，其中NCE表示“噪声对比估计”。假设我们有一个神经网络函数 $\mathbf{f}_w(\mathbf{x})$ ，它将输入空间 \mathbf{x} 中的点映射到一个由可学习参数 w 控制的表示空间。这个表示被归一化，使得 $\|\mathbf{f}_w(\mathbf{x})\| = 1$ 。然后，对于一个数据点 \mathbf{x} ，InfoNCE损失定义为

$$E(w) = -\ln \frac{\exp\{\mathbf{f}_w(\mathbf{x})^T \mathbf{f}_w(\mathbf{x}^+)\}}{\exp\{\mathbf{f}_w(\mathbf{x})^T \mathbf{f}_w(\mathbf{x}^+)\} + \sum_{n=1}^N \exp\{\mathbf{f}_w(\mathbf{x})^T \mathbf{f}_w(\mathbf{x}_n^-)\}}. \quad (6.20)$$

我们可以看到，在这个函数中，锚点的表示 $\mathbf{f}_w(\mathbf{x})$ 与正例的表示 $\mathbf{f}_w(\mathbf{x}^+)$ 之间的余弦相似度 $\mathbf{f}_w(\mathbf{x})^T \mathbf{f}_w(\mathbf{x}^+)$ 提供了我们对正例对在学习空间中接近程度的度量，并且使用相同的度量来评估锚点与负例的接近程度。请注意，该函数类似于分类交叉熵误差函数，其中正例对的余弦相似度给出标签类别的对数几率，负例对的余弦相似度给出错误类别的对数几率。

第9.1.3节

另请注意，负样本对至关重要，因为没有它们，嵌入模型只会学习到将每个点映射到相同表示的退化解决方案。

一种特定的对比学习算法主要由正负样本对的选择方式定义，这就是我们如何利用先验知识来指定什么是好的表示。例如，考虑学习图像表示的问题。在这里，一种常见的选择是通过以应保留图像语义信息但在像素空间中极大地改变图像的方式损坏输入图像来创建正样本对（Wu等人，2018年；He等人，2019年；Chen、Kornblith等人，2020年）。损坏与数据增强密切相关，示例包括旋转、平移和颜色偏移。然后可以使用数据集中的其他图像来创建负样本对。这种对比学习方法称为实例判别。

然而，如果我们能够获取类别标签，那么我们可以将同一类别的图像用作正样本对，将不同类别的图像用作负样本对。这放宽了对指定表示应不变的增强操作的依赖，并且还避免将两个语义相似的图像视为负样本对。由于依赖类别标签，这被称为监督对比学习（Khosla等人，2020年），并且它通常可以产生比简单地使用交叉熵分类学习表示更好的结果。

正样本对和负样本对的成员不一定必须来自相同的数据模态。在对比语言图像预训练（CLIP，Radford等人，2021年）中，一个正样本对由一幅图像及其相应的文本描述组成，并且使用两个单独的函数（每个模态一个）将输入映射到相同的表示空间。然后，负样本对是不匹配的图像和描述。这通常被称为弱监督，因为它依赖于带注释的图像，通过从互联网上抓取数据通常比手动用类别标记图像更容易获得这些图像。在这种情况下，损失函数由

$$\begin{aligned} E(\mathbf{w}) = & -\frac{1}{2} \ln \frac{\exp\{\mathbf{f}_w(\mathbf{x}^+)^T \mathbf{g}_\theta(\mathbf{y}^+)\}}{\exp\{\mathbf{f}_w(\mathbf{x}^+)^T \mathbf{g}_\theta(\mathbf{y}^+)\} + \sum_{n=1}^N \exp\{\mathbf{f}_w(\mathbf{x}_n^-)^T \mathbf{g}_\theta(\mathbf{y}^+)\}} \\ & -\frac{1}{2} \ln \frac{\exp\{\mathbf{f}_w(\mathbf{x}^+)^T \mathbf{g}_\theta(\mathbf{y}^+)\}}{\exp\{\mathbf{f}_w(\mathbf{x}^+)^T \mathbf{g}_\theta(\mathbf{y}^+)\} + \sum_{m=1}^M \exp\{\mathbf{f}_w(\mathbf{x}^+)^T \mathbf{g}_\theta(\mathbf{y}_m^-)\}} \end{aligned} \quad (6.21)$$

其中 \mathbf{x}^+ 和 \mathbf{y}^+ 表示正例对，其中 \mathbf{x} 是一个图像， \mathbf{y} 是其对应的文本描述， \mathbf{f}_w 表示从图像到表示空间的映射， \mathbf{g}_θ 是从文本输入到表示空间的映射。我们还需要数据集中的一组 $\{\mathbf{x}_1^-, \dots, \mathbf{x}_N^-\}$ 其他图像，对于这些图像，我们可以假设文本描述 \mathbf{y}^+ 不合适，以及一组 $\{\mathbf{y}_1^-, \dots, \mathbf{y}_M^-\}$ 与输入图像 \mathbf{x} 同样不匹配的文本描述。损失函数中的两项确保：(a)相对于其他图像表示，图像的表示与其文本描述表示接近；(b)相对于文本描述的其他表示，文本描述表示与其所描述图像的表示接近。

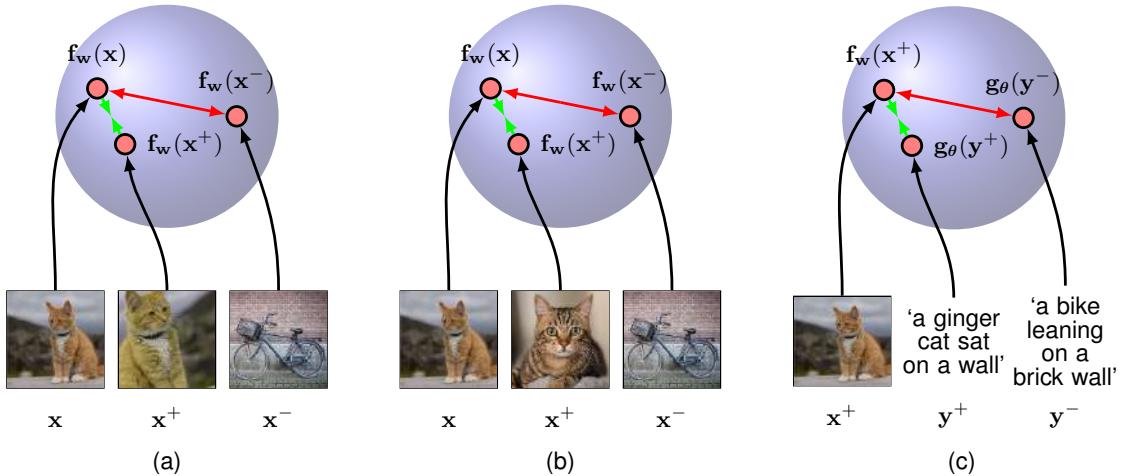


图6.14三种不同对比学习范式的示意图。(a)实例判别方法，其中正例对由锚点和同一图像的增强版本组成。这些被映射到归一化空间中的点，可以看作是一个单位超球体。彩色箭头表明损失会促使正例对的表示更接近，但会将负例对进一步分开。(b)监督对比学习，其中正例对由来自同一类别的两个不同图像组成。(c)CLIP模型，其中正例对由一幅图像和一个相关的文本片段组成。

虽然CLIP使用文本和图像对，但任何具有配对模态的数据集都可用于学习表示。我们讨论过的不同对比学习方法的比较如图6.14所示。

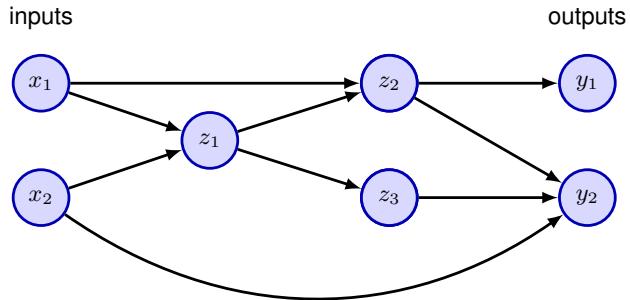
6.3.6 通用网络架构

到目前为止，我们已经探讨了由一系列全连接层组成的神经网络架构。然而，由于网络图与其数学函数之间存在直接对应关系，我们可以通过考虑更复杂的网络图来开发更通用的网络映射。这些必须限制为前馈架构，换句话说，是一个没有封闭有向环的架构，以确保输出是输入的确定性函数。图6.15中的一个简单示例说明了这一点。这种网络中的每个（隐藏或输出）单元计算一个由下式给出的函数

$$z_k = h \left(\sum_{j \in \mathcal{A}(k)} w_{kj} z_j + b_k \right) \quad (6.22)$$

其中 $\mathcal{A}(k)$ 表示节点 k 的祖先集，换句话说，是向单元 k 发送连接的单元集， b_k 表示相关的偏置参数。对于应用于网络输入的给定值集，连续应用(6.22)可以评估网络中所有单元的激活，包括输出单元的激活。

图6.15具有一般前馈拓扑结构的神经网络示例。请注意，每个隐藏单元和输出单元都有一个相关的偏差参数（为清晰起见省略）。



6.3.7 张量

我们看到线性代数在神经网络中起着核心作用，诸如数据集、激活值和网络参数等数量都表示为标量、向量和矩阵。然而，我们也会遇到更高维度的变量。例如，考虑一个由 N 幅彩色图像组成的数据集，每幅图像高 I 像素、宽 J 像素。每个像素在图像中由其行和列索引，并具有红色、绿色和蓝色值。数据集中的每幅图像都有这样一组值，因此我们可以用一个四维数组 \mathbf{X} 来表示特定的强度值，其元素为 x_{ijkn} ，其中 $i \in \{1, \dots, I\}$ 和 $j \in \{1, \dots, J\}$ 索引图像内的行和列， $k \in \{1, 2, 3\}$ 索引红色、绿色和蓝色强度， $n \in \{1, \dots, N\}$ 索引数据集中的特定图像。这些高维数组称为张量，标量、向量和矩阵都是张量的特殊情况。在本书后面讨论更复杂的神经网络架构时，我们会看到许多这样的张量示例。诸如GPU之类的大规模并行处理器特别适合处理张量。

6.4. 误差函数

第4章 第
5章

2.3.4节

在前面的章节中，我们探讨了用于回归和分类的线性模型，在此过程中，我们推导出了误差函数的合适形式以及输出单元激活函数的相应选择。选择误差函数的相同考虑因素适用于多层神经网络，因此为了方便起见，我们将在此总结要点。

6.4.1 回归

我们首先讨论回归问题，目前我们考虑一个可以取任何实值的单个目标变量 t 。在讨论了单层网络中的回归之后，我们假设 t 具有高斯分布，其均值依赖于 \mathbf{x} ，该均值由神经网络的输出给出，因此

$$p(t|\mathbf{x}, \mathbf{w}) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \sigma^2) \quad (6.23)$$

第6.5节

其中 σ^2 是高斯噪声的方差。当然，这是一个有点严格的假设，在某些应用中，我们需要扩展这种方法以允许更一般的分布。对于由(6.23)给出的条件分布，将输出单元激活函数设为恒等函数就足够了，因为这样的网络可以逼近从 \mathbf{x} 到 y 的任何连续函数。给定一个由 N 个独立同分布观测值 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ 组成的数据集，以及相应的目标值 $\mathbf{t} = \{t_1, \dots, t_N\}$ ，我们可以构造相应的似然函数：

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \sigma^2) = \prod_{n=1}^N p(t_n|y(\mathbf{x}_n, \mathbf{w}), \sigma^2). \quad (6.24)$$

请注意，在机器学习文献中，通常考虑的是最小化误差函数而不是最大化似然函数，因此在这里我们将遵循这个惯例。对似然函数(6.24)取负对数，我们得到误差函数

$$\frac{1}{2\sigma^2} \sum_{n=1}^N \{y(\mathbf{x}_n, \mathbf{w}) - t_n\}^2 + \frac{N}{2} \ln \sigma^2 + \frac{N}{2} \ln(2\pi), \quad (6.25)$$

它可用于学习参数 \mathbf{w} 和 σ^2 。首先考虑 \mathbf{w} 的确定。最大化似然函数等同于最小化由下式给出的平方和误差函数

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(\mathbf{x}_n, \mathbf{w}) - t_n\}^2 \quad (6.26)$$

我们已经舍弃了加性和乘性常数。通过最小化 $E(\mathbf{w})$ 找到的 \mathbf{w} 值将记为 \mathbf{w}^* 。请注意，这通常不会对应于似然函数的全局最大值，因为网络函数 $y(\mathbf{x}_n, \mathbf{w})$ 的非线性会导致误差 $E(\mathbf{w})$ 是非凸的，因此找到全局最优解通常是不可行的。此外，可以将正则化项添加到误差函数中，并对训练过程进行其他修改，这样得到的网络参数解可能与最大似然解有很大差异。

第9章

练习6.8

找到 \mathbf{w}^* 后，可以通过最小化误差函数(6.25)来找到 σ^2 的值，得到

$$\sigma^{2*} = \frac{1}{N} \sum_{n=1}^N \{y(\mathbf{x}_n, \mathbf{w}^*) - t_n\}^2. \quad (6.27)$$

请注意，一旦找到 \mathbf{w}^* 所需的迭代优化完成，就可以对此进行评估。

如果我们有多个目标变量，并且假设它们是独立的，在给定 \mathbf{x} 和 \mathbf{w} 的条件下，具有共享噪声方差 σ^2 ，那么目标值的条件分布由下式给出

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}) = \mathcal{N}(\mathbf{t}|\mathbf{y}(\mathbf{x}, \mathbf{w}), \sigma^2 \mathbf{I}). \quad (6.28)$$

遵循与单个目标变量相同的论证，我们发现关于权重最大化似然函数等同于最小化平方和误差函数：

练习6.9

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \|\mathbf{y}(\mathbf{x}_n, \mathbf{w}) - \mathbf{t}_n\|^2. \quad (6.29)$$

噪声方差由下式给出

$$\sigma^{2*} = \frac{1}{NK} \sum_{n=1}^N \|\mathbf{y}(\mathbf{x}_n, \mathbf{w}^*) - \mathbf{t}_n\|^2 \quad (6.30)$$

其中 K 是目标变量的维度。目标变量的条件独立性假设可以舍弃，代价是优化问题会稍微复杂一些。

练习6.10

第5.4.6节

回想一下，误差函数（由负对数似然给出）和输出单元激活函数之间存在自然的配对关系。在回归中，我们可以将网络视为具有恒等输出激活函数，使得 $y_k = a_k$ 。相应的平方和误差函数则具有性质

$$\frac{\partial E}{\partial a_k} = y_k - t_k. \quad (6.31)$$

6.4.2 二元分类

现在考虑二分类问题，其中我们有一个单一的目标变量 t ，使得 $t = 1$ 表示类别 \mathcal{C}_1 ， $t = 0$ 表示类别 \mathcal{C}_2 。根据对典范链接函数的讨论，我们考虑一个具有单个输出的网络，其激活函数是逻辑 sigmoid 函数 (6.13)，使得 $0 \leq y(\mathbf{x}, \mathbf{w}) \leq 1$ 。我们可以将 $y(\mathbf{x}, \mathbf{w})$ 解释为条件概率 $p(\mathcal{C}_1 | \mathbf{x})$ ，其中 $p(\mathcal{C}_2 | \mathbf{x})$ 由 $1 - y(\mathbf{x}, \mathbf{w})$ 给出。给定输入时目标的条件分布则是如下形式的伯努利分布

$$p(t | \mathbf{x}, \mathbf{w}) = y(\mathbf{x}, \mathbf{w})^t \{1 - y(\mathbf{x}, \mathbf{w})\}^{1-t}. \quad (6.32)$$

如果我们考虑一组独立观测的训练集，那么由负对数似然给出的误差函数就是如下形式的交叉熵误差

$$E(\mathbf{w}) = - \sum_{n=1}^N \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\} \quad (6.33)$$

其中 y_n 表示 $y(\mathbf{x}_n, \mathbf{w})$ 。西马尔、斯坦克劳斯和普拉特（2003年）发现，对于分类问题，使用交叉熵误差函数而非平方和误差函数会使训练速度更快，泛化能力也更强。

注意，(6.32) 中没有噪声方差 σ^2 的类似物，因为假设目标值的标签是正确的。然而，通过引入目标值 t 被翻转到错误值的概率 ϵ ，该模型可以很容易地扩展以考虑标签错误（奥珀和温特，2000年）。

练习6.11

这里 ϵ 可以预先设定，或者它可以被视为一个超参数，其值从数据中推断得出。

如果我们要进行 K 个独立的二元分类，那么我们可以使用一个具有 K 个输出的网络，每个输出都有一个逻辑 sigmoid 激活函数。与每个输出相关联的是一个二元类标签 $t_k \in \{0, 1\}$ ，其中 $k = 1, \dots, K$ 。如果我们假设在给定输入向量的情况下，类标签是独立的，那么目标的条件分布是

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}) = \prod_{k=1}^K y_k(\mathbf{x}, \mathbf{w})^{t_k} [1 - y_k(\mathbf{x}, \mathbf{w})]^{1-t_k}. \quad (6.34)$$

对相应的似然函数取负对数，然后得到以下误差函数：

练习6.13

$$E(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K \{t_{nk} \ln y_{nk} + (1 - t_{nk}) \ln(1 - y_{nk})\} \quad (6.35)$$

其中 y_{nk} 表示 $y_k(\mathbf{x}_n, \mathbf{w})$ 。同样，误差函数关于特定输出单元的预激活的导数采用形式(6.31)，就像在回归情况下一样。

练习6.14

第5.1.3节

6.4.3 多类分类

最后，我们考虑标准的多类分类问题，其中每个输入被分配到 K 个互斥类别中的一个。二元目标变量 $t_k \in \{0, 1\}$ 采用 K 选1编码方案来指示类别，并且网络输出被解释为 $y_k(\mathbf{x}, \mathbf{w}) = p(t_k = 1 | \mathbf{x})$ ，从而得到误差函数(5.80)，我们在此重现：

$$E(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^K t_{kn} \ln y_k(\mathbf{x}_n, \mathbf{w}). \quad (6.36)$$

对应于典范链接的输出单元激活函数由softmax函数给出：

第5.4.4节

$$y_k(\mathbf{x}, \mathbf{w}) = \frac{\exp(a_k(\mathbf{x}, \mathbf{w}))}{\sum_j \exp(a_j(\mathbf{x}, \mathbf{w}))}, \quad (6.37)$$

满足 $0 \leq y_k \leq 1$ 和 $\sum_k y_k = 1$ 。请注意，如果给所有 $a_k(\mathbf{x}, \mathbf{w})$ 都加上一个常数， $y_k(\mathbf{x}, \mathbf{w})$ 不会改变，这会导致误差函数在权重空间的某些方向上为常数。如果在误差函数中添加一个适当的正则化项，这种简并性就会消除。同样，误差函数关于特定输出单元的预激活的导数具有熟悉的形式(6.31)。

第9章

练习6.15

总之，根据要解决的问题类型，对输出单元激活函数和匹配误差函数都有自然的选择。对于回归问题，我们使用线性输出和平方和误差；对于多个独立的二元分类问题，我们使用逻辑 sigmoid 输出和交叉熵误差函数；对于多类分类问题，我们使用具有相应多类交叉熵误差函数的 softmax 输出。对于涉及两类的分类问题，我们可以使用单个逻辑 sigmoid 输出，或者也可以使用具有 softmax 输出激活函数的双输出网络。

这个过程非常通用，通过考虑其他形式的条件分布，我们可以将相关的误差函数推导为相应的负对数似然。在下一节考虑多模态网络输出时，我们将看到一个这样的例子。

6.5. 混合密度网络

在本章中，我们到目前为止讨论了神经网络，其输出表示简单的概率分布，包括连续变量的高斯分布或离散变量的二元分布。我们通过展示神经网络如何通过将网络输出视为更复杂分布（在这种情况下是高斯混合模型）的参数来表示更一般的条件概率来结束本章。这被称为混合密度网络，我们将看到如何定义相关的误差函数和相应的输出单元激活函数。

6.5.1 机器人运动学示例

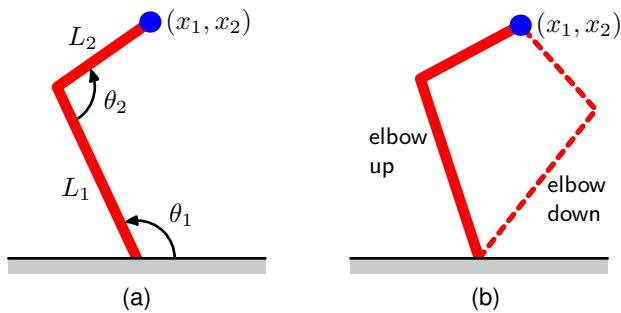
监督学习的目标是对条件分布 $p(t | x)$ 进行建模，对于许多简单的回归问题，该分布被选择为高斯分布。然而，实际的机器学习问题通常可能具有明显的非高斯分布。例如，在逆问题中可能会出现这种情况，其中分布可能是多模态的，在这种情况下，高斯假设可能导致非常差的预测。

作为一个逆问题的简单示例，考虑如图6.16所示的机器人手臂的运动学。正向问题涉及在给定关节角度的情况下找到末端执行器的位置，并且有唯一解。然而，在实际中，我们希望将机器人的末端执行器移动到特定位置，为此我们必须设置适当的关节角度。因此，我们需要解决逆问题，如图6.16所示，该问题有两个解。

正向问题通常对应于物理系统中的因果关系，并且通常有唯一解。例如，人体中特定的症状模式可能是由特定疾病的存引起。然而，在机器学习中，我们通常必须解决一个逆问题，例如尝试根据一组症状预测疾病的存。如果正向问题涉及多对一映射，那么逆问题将有多个解。例如，几种不同的疾病可能导致相同的症状。

练习6.16

图6.16 (a) 一个双连杆机器人手臂，其中末端执行器的笛卡尔坐标 (x_1, x_2) 由两个关节角度 θ_1 和 θ_2 以及手臂的（固定）长度 L_1 和 L_2 唯一确定。这被称为手臂的正向运动学。(b) 在实际中，我们必须找到能产生所需末端执行器位置的关节角度。这种逆运动学有对应于“肘部向上”和“肘部向下”的两种解决方案。



在机器人技术的示例中，运动学由几何方程定义，并且多模态性很容易显现出来。然而，在许多机器学习问题中，多模态性的存在，特别是在涉及高维空间的问题中，可能不太明显。不过，出于教程目的，我们将考虑一个简单的玩具问题，对于这个问题我们可以很容易地可视化多模态性。这个问题的数据是通过在区间 $(0, 1)$ 上均匀采样变量 x 生成的，得到一组值 $\{x_n\}$ ，并且相应的目标值 t_n 是通过计算函数 $x_n + 0.3 \sin(2\pi x_n)$ 然后在区间 $(-0.1, 0.1)$ 上添加均匀噪声获得的。然后通过保持相同的数据点但交换 x 和 t 的角色来获得逆问题。图6.17展示了正向问题和逆问题的数据集，以及通过最小化平方和误差函数来拟合具有六个隐藏单元和一个线性输出单元的两层神经网络的结果。在高斯假设下，最小二乘法对应于最大似然估计。我们看到，这为正向问题带来了一个很好的模型，但对于高度非高斯的逆问题则是一个非常差的模型。

6.5.2 条件混合分布

因此，我们寻求一个用于对条件概率分布进行建模的通用框架。

图6.17 左边是一个简单正向问题的数据集，其中红色曲线显示了通过最小化平方和误差函数拟合两层神经网络的结果。右边显示的相应逆问题是通过交换 x 和 t 的角色得到的。这里，同样通过最小化平方和误差函数进行训练的网络，由于数据集的多峰性，对数据的拟合效果很差。

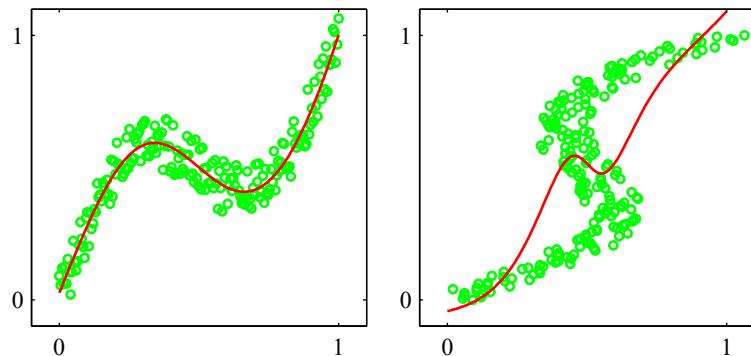
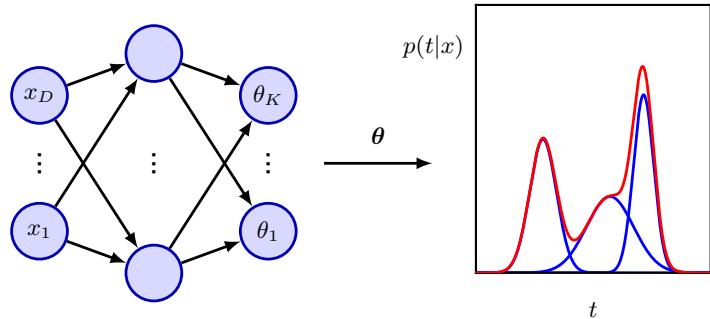


图6.18混合密度网络可以通过考虑 t 分布的参数混合模型来表示一般条件概率密度 $p(t|x)$ ，其参数由 x 作为输入向量的神经网络的输出确定。



这可以通过对 $p(t|x)$ 使用混合模型来实现，其中混合系数以及分量密度都是输入向量 x 的灵活函数，从而产生混合密度网络。对于 x 的任何给定值，混合模型为建模任意条件密度函数 $p(t|x)$ 提供了一种通用形式。如果我们考虑一个足够灵活的网络，那么我们就有了一个近似任意条件分布的框架。

这里我们将明确针对高斯分量开发模型，以便

$$p(t|x) = \sum_{k=1}^K \pi_k(x) \mathcal{N}(t|\mu_k(x), \sigma_k^2(x)). \quad (6.38)$$

这是一个异方差模型的示例，其中数据上的噪声方差是输入向量 x 的函数。除了高斯分布，我们可以对各分量使用其他分布，例如如果目标变量是二元而非连续的，则使用伯努利分布。我们还专门研究了各分量的各向同性协方差情况，尽管混合密度网络可以通过使用乔列斯基分解来表示协方差

(Williams, 1996)，从而很容易扩展以允许使用一般协方差矩阵。即使各分量是各向同性的，由于混合分布的原因，条件分布 $p(t|x)$ 相对于 t 的分量也不假设可分解（与标准平方和回归模型形成对比）。

我们现在将混合模型的各种参数，即混合系数 $\pi_k(x)$ 、均值 $\mu_k(x)$ 和方差 $\sigma_k^2(x)$ ，设定为由一个以 x 作为输入的神经网络的输出所控制。这种混合密度网络的结构如图6.18所示。混合密度网络与专家混合模型 (Jacobs等人，1991年) 密切相关。主要区别在于，专家混合模型中混合的每个组件模型都有独立的参数，而在混合密度网络中，相同的函数用于预测所有组件密度的参数以及混合系数，因此非线性隐藏单元在依赖于输入的函数之间共享。

例如，图6.18中的神经网络可以是一个具有Sigmoid (\tanh) 隐藏单元的两层网络。

如果混合模型 (6.38) 中有 K 个组件，并且如果 \mathbf{t} 有 L 个组件，那么网络将有 K 个由 a_k^π 表示的输出单元预激活，这些预激活确定混合系数 $\pi_k(\mathbf{x})$ ；有 a_k^σ 表示的输出，这些输出确定高斯标准差 $\sigma_k(\mathbf{x})$ ；以及 $K \times L$ 有 a_{kj}^μ 表示的输出，这些输出确定高斯均值 $\mu_k(\mathbf{x})$ 的组件 $\mu_{kj}(\mathbf{x})$ 。网络输出的总数由 $(L + 2)K$ 给出，这与简单预测目标变量条件均值的网络通常的 L 个输出不同。

混合系数必须满足约束条件

$$\sum_{k=1}^K \pi_k(\mathbf{x}) = 1, \quad 0 \leq \pi_k(\mathbf{x}) \leq 1, \quad (6.39)$$

这可以通过一组softmax输出实现：

$$\pi_k(\mathbf{x}) = \frac{\exp(a_k^\pi)}{\sum_{l=1}^K \exp(a_l^\pi)}. \quad (6.40)$$

类似地，方差必须满足 $\sigma_k^2(\mathbf{x}) \geq 0$ ，因此可以使用相应网络预激活的指数来表示

$$\sigma_k(\mathbf{x}) = \exp(a_k^\sigma). \quad (6.41)$$

最后，由于均值 $\mu_k(\mathbf{x})$ 具有实部，它们可以直接由网络输出表示：

$$\mu_{kj}(\mathbf{x}) = a_{kj}^\mu \quad (6.42)$$

其中输出单元激活函数由恒等式 $f(a) = a$ 给出。

混合密度网络的可学习参数包括神经网络中权重和偏差的向量 \mathbf{w} ，可以通过最大似然法或等效地通过最小化定义为似然负对数的误差函数来设置。对于独立数据，此误差函数采用以下形式

$$E(\mathbf{w}) = - \sum_{n=1}^N \ln \left\{ \sum_{k=1}^K \pi_k(\mathbf{x}_n, \mathbf{w}) \mathcal{N}(\mathbf{t}_n | \boldsymbol{\mu}_k(\mathbf{x}_n, \mathbf{w}), \sigma_k^2(\mathbf{x}_n, \mathbf{w})) \right\} \quad (6.43)$$

我们已经明确了对 \mathbf{w} 的依赖关系。

6.5.3 梯度优化

为了最小化误差函数，我们需要计算误差 $E(\mathbf{w})$ 关于 \mathbf{w} 各分量的导数。我们将在后面看到如何自动计算这些导数。然而，明确推导误差关于输出单元预激活的导数的合适表达式是很有启发性的，因为这突出了这些量的概率解释。由于误差函数 (6.43) 由一系列项组成，每个训练数据点对应一项，我们可以考虑针对具有相关目标向量 \mathbf{t}_n 的特定输入向量 \mathbf{x}_n 的导数。

第7章

总误差 E 的导数通过对所有数据点求和得到，或者每个数据点的单个梯度可直接用于基于梯度的优化算法中。

引入以下变量很方便：

$$\gamma_{nk} = \gamma_k(\mathbf{t}_n | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}_{nk}}{\sum_{l=1}^K \pi_l \mathcal{N}_{nl}} \quad (6.44)$$

其中 \mathcal{N}_{nk} 表示 $\mathcal{N}(\mathbf{t}_n | \mu_k(\mathbf{x}_n), \sigma_k^2(\mathbf{x}_n))$ 。这些量自然地解释为混合物各成分的后验概率，其中混合系数 $\pi_k(\mathbf{x})$ 被视为依赖于 \mathbf{x} 的先验概率。

练习6.17

误差函数关于控制混合系数的网络输出预激活的导数由下式给出

练习6.18

$$\frac{\partial E_n}{\partial a_k^\pi} = \pi_k - \gamma_{nk}. \quad (6.45)$$

类似地，关于控制成分均值的输出预激活的导数由下式给出

练习6.19

$$\frac{\partial E_n}{\partial a_{kl}^\mu} = \gamma_{nk} \left\{ \frac{\mu_{kl} - t_{nl}}{\sigma_k^2} \right\}. \quad (6.46)$$

最后，关于控制成分方差的输出预激活的导数由下式给出

练习6.20

$$\frac{\partial E_n}{\partial a_k^\sigma} = \gamma_{nk} \left\{ L - \frac{\|\mathbf{t}_n - \boldsymbol{\mu}_k\|^2}{\sigma_k^2} \right\}. \quad (6.47)$$

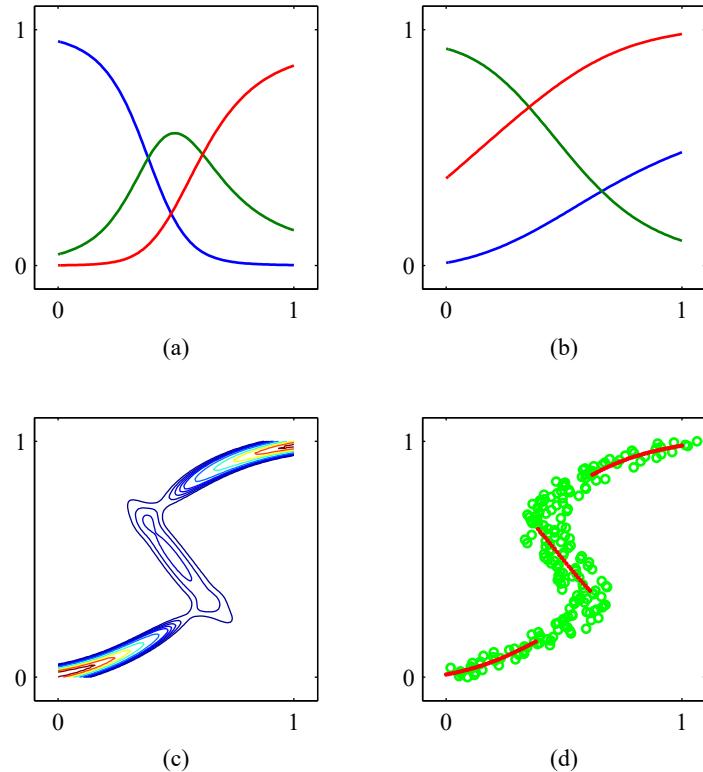
6.5.4 预测分布

我们通过回到图6.17所示的逆问题的玩具示例来说明混合密度网络的使用。图6.19展示了混合系数 $\pi_k(x)$ 、均值 $\mu_k(x)$ 以及对应于 $p(t|x)$ 的条件密度等高线的图。神经网络的输出以及混合模型中的参数必然是输入变量的连续单值函数。然而，我们从图6.19(c)中可以看出，该模型能够通过调制混合分量 $\pi_k(\mathbf{x})$ 的幅度，为 x 的某些值生成单峰条件密度，为其他值生成三峰条件密度。

一旦混合密度网络训练完成，它就能针对输入向量的任何给定值预测目标数据的条件密度函数。就预测输出向量值的问题而言，这种条件密度代表了对数据生成器的完整描述。根据这个密度函数，我们可以计算出在不同应用中可能感兴趣的更具体的量。其中最简单的一个就是均值，它对应于目标数据的条件平均值，其计算公式为

$$\mathbb{E}[\mathbf{t}|\mathbf{x}] = \int \mathbf{t} p(\mathbf{t}|\mathbf{x}) d\mathbf{t} = \sum_{k=1}^K \pi_k(\mathbf{x}) \boldsymbol{\mu}_k(\mathbf{x}) \quad (6.48)$$

图6.19 (a) 对于在图6.17所示数据上训练的混合密度网络中的三种混合成分，混合系数 $\pi_k(x)$ 作为 x 的函数的绘图。该模型有三个高斯成分，并使用一个两层神经网络，隐藏层有五个双曲正切S型单元和九个输出（对应于高斯成分的三个均值、三个方差和三个混合系数）。在 x 的小值和大值处，目标数据的条件概率密度是单峰的，只有一个高斯成分的先验概率有高值，而在 x 的中间值处，条件密度是三峰的，三个混合系数有可比的值。(b) 使用与混合系数相同的颜色编码绘制的均值 $\mu_k(x)$ 。(c) 对于相同混合密度网络的目标数据的相应条件概率密度的等高线图。(d) 条件密度的近似条件众数的绘图，由红色点表示。



我们在此处使用了(6.38)。由于通过最小二乘法训练的标准网络逼近条件均值，我们可以看到混合密度网络在特殊情况下可以重现传统的最小二乘结果。当然，正如我们已经指出的，对于多峰分布，条件均值的价值有限。

我们可以类似地评估密度函数关于条件均值的方差，得到

练习6.21

$$s^2(\mathbf{x}) = \mathbb{E} \left[\| \mathbf{t} - \mathbb{E}[\mathbf{t}|\mathbf{x}] \|^2 | \mathbf{x} \right] \quad (6.49)$$

$$= \sum_{k=1}^K \pi_k(\mathbf{x}) \left\{ \sigma_k^2(\mathbf{x}) + \left\| \boldsymbol{\mu}_k(\mathbf{x}) - \sum_{l=1}^K \pi_l(\mathbf{x}) \boldsymbol{\mu}_l(\mathbf{x}) \right\|^2 \right\} \quad (6.50)$$

我们在此处使用了(6.38)和(6.48)。这比相应的最小二乘结果更具一般性，因为方差是 \mathbf{x} 的函数。

我们已经看到，对于多峰分布，条件均值可能无法很好地表示数据。例如，在控制图6.16所示的简单机器人手臂时，我们需要从两个可能的关节角度设置中选择一个来实现所需的末端执行器位置，但这两个解决方案的平均值本身并不是一个解决方案。

在这种情况下，条件模式可能更有价值。由于混合密度网络的条件模式没有简单的解析解，因此需要进行数值迭代。一个简单的替代方法是在 x 的每个值处取最可能分量（即混合系数最大的分量）的均值。图6.19(d)中的玩具数据集展示了这一点。

练习

- 6.1 (★★★) 使用结果(2.126)推导 D 维中单位半径超球体的表面积 S_D 和体积 V_D 的表达式。为此，考虑以下通过从笛卡尔坐标转换为极坐标得到的结果：

$$\prod_{i=1}^D \int_{-\infty}^{\infty} e^{-x_i^2} dx_i = S_D \int_0^{\infty} e^{-r^2} r^{D-1} dr. \quad (6.51)$$

使用由下式定义的伽马函数

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt \quad (6.52)$$

结合(2.126)，对该方程两边进行求值，从而证明

$$S_D = \frac{2\pi^{D/2}}{\Gamma(D/2)}. \quad (6.53)$$

接下来，通过对半径从0到1进行积分，证明 D 维中单位超球体的体积由下式给出

$$V_D = \frac{S_D}{D}. \quad (6.54)$$

最后，使用结果 $\Gamma(1) = 1$ 和 $\Gamma(3/2) = \sqrt{\pi}/2$ 来证明(6.53)和(6.54)简化为 $D = 2$ 和 $D = 3$ 的常用表达式。

- 6.2 (★★★) 考虑在 D 维中半径为 a 的超球面以及边长为 $2a$ 的同心超立方体，使得超球面在其每条边的中心处与超立方体相切。通过使用练习6.1的结果，证明超球面体积与立方体体积之比由下式给出

$$\frac{\text{volume of hypersphere}}{\text{volume of cube}} = \frac{\pi^{D/2}}{D 2^{D-1} \Gamma(D/2)}. \quad (6.55)$$

现在使用以下形式的斯特林公式

$$\Gamma(x+1) \simeq (2\pi)^{1/2} e^{-x} x^{x+1/2}, \quad (6.56)$$

该公式对 $x \gg 1$ 有效，以证明当 $D \rightarrow \infty$ 时，比率(6.55)趋于零。

还需证明，从超立方体中心到其中一个角的距离除以到其中一条边的垂直距离为 \sqrt{D} ，因此当 $D \rightarrow \infty$ 时，该值趋向于 ∞ 。从这些结果中，我们可以看到，在高维空间中，立方体的大部分体积集中在大量的角上，而这些角本身变成了非常长的“尖刺”！

6.3 (★★) 在本练习中，我们探讨高斯分布在高维空间中的行为。考虑在 D 维中的高斯分布，其形式为

$$p(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{D/2}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2\sigma^2}\right). \quad (6.57)$$

我们希望找到在极坐标下作为半径函数的密度，其中方向变量已被积分掉。为此，证明在半径为 r 、厚度为 ϵ （其中 $\epsilon \ll 1$ ）的薄壳上概率密度的积分由 $p(r)\epsilon$ 给出，其中

$$p(r) = \frac{S_D r^{D-1}}{(2\pi\sigma^2)^{D/2}} \exp\left(-\frac{r^2}{2\sigma^2}\right) \quad (6.58)$$

其中 S_D 是 D 维空间中单位超球面的表面积。证明函数 $p(r)$ 有一个唯一的驻点，对于大的 D ，该驻点位于 $\hat{r} \simeq \sqrt{D}\sigma$ 。通过考虑 $p(\hat{r} + \epsilon)$ ，其中 $\epsilon \ll \hat{r}$ ，证明对于大的 D ，

$$p(\hat{r} + \epsilon) = p(\hat{r}) \exp\left(-\frac{3\epsilon^2}{2\sigma^2}\right), \quad (6.59)$$

这表明 \hat{r} 是径向概率密度的最大值，并且 $p(r)$ 在远离其在 \hat{r} 处的最大值时以长度尺度 σ 呈指数衰减。我们已经看到，对于大的 D ， $\sigma \ll \hat{r}$ ，因此我们看到大部分概率质量集中在大半径处的一个薄壳中。最后，证明概率密度 $p(\mathbf{x})$ 在原点处比在半径 \hat{r} 处大 $\exp(D/2)$ 倍。因此，我们看到高维高斯分布中的大部分概率质量位于与高概率密度区域不同的半径处。

6.4 (★★) 考虑形式为(6.11)的两层网络函数，其中隐藏单元非线性激活函数 $h(\cdot)$ 由如下形式的逻辑 sigmoid 函数给出

$$\sigma(a) = \{1 + \exp(-a)\}^{-1}. \quad (6.60)$$

证明存在一个等效网络，它计算的函数完全相同，但隐藏单元激活函数由 $\tanh(a)$ 给出，其中 \tanh 函数由(6.14)定义。提示：首先找到 $\sigma(a)$ 和 $\tanh(a)$ 之间的关系，然后证明两个网络的参数通过线性变换而不同。

6.5 (★★) swish 激活函数 (Ramachandran、Zoph 和 Le, 2017) 定义如下

$$h(x) = x\sigma(\beta x) \quad (6.61)$$

其中 $\sigma(x)$ 是由(6.13)定义的逻辑 sigmoid 激活函数。当在神经网络中使用时， β 可被视为一个可学习参数。使用软件绘制或手绘 swish 激活函数及其关于 $\beta = 0.1, \beta = 1.0$ 和 $\beta = 10$ 的一阶导数的图形。证明当 $\beta \rightarrow \infty$ 时，swish 函数变为 ReLU 函数。

6.6 (*) 我们在(5.72)中看到逻辑 sigmoid 激活函数的导数可以用函数值本身来表示。推导由(6.14)定义的 tanh 激活函数的相应结果。

6.7 (**) 证明由(6.16)给出的 softplus 激活函数 $\zeta(a)$ 满足以下性质：

$$\zeta(a) - \zeta(-a) = a \quad (6.62)$$

$$\ln \sigma(a) = -\zeta(-a) \quad (6.63)$$

$$\frac{d\zeta(a)}{da} = \sigma(a) \quad (6.64)$$

$$\zeta^{-1}(a) = \ln(\exp(a) - 1) \quad (6.65)$$

其中 $\sigma(a)$ 是由(6.13)给出的逻辑 sigmoid 激活函数。

6.8 (*) 证明关于方差 σ^2 最小化误差函数(6.25)会得到结果(6.27)。

6.9 (*) 证明对于多输出神经网络，在条件分布(6.28)下最大化似然函数等同于最小化平方和误差函数(6.29)。此外，证明使该误差函数最小化的噪声方差由(6.30)给出。

6.10 (**) 考虑一个涉及多个目标变量的回归问题，其中假设目标的分布在输入向量 \mathbf{x} 的条件下是如下形式的高斯分布

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}) = \mathcal{N}(\mathbf{t}|\mathbf{y}(\mathbf{x}, \mathbf{w}), \Sigma) \quad (6.66)$$

其中 $\mathbf{y}(\mathbf{x}, \mathbf{w})$ 是具有输入向量 \mathbf{x} 和权重向量 \mathbf{w} 的神经网络的输出，并且 Σ 是目标上假设的高斯噪声的协方差。给定一组 \mathbf{x} 和 \mathbf{t} 的独立观测值，如果我们假设 Σ 是固定且已知的，写出为了找到 \mathbf{w} 的最大似然解而必须最小化的误差函数。现在假设 Σ 也将从数据中确定，并写出 Σ 的最大似然解的表达式。请注意，与6.4.1节中讨论的独立目标变量的情况相比， \mathbf{w} 和 Σ 的优化现在是耦合的。

6.11 (**) 考虑一个二分类问题，其中目标值为 $t \in \{0, 1\}$ ，网络输出为 $y(\mathbf{x}, \mathbf{w})$ ，它表示 $p(t=1 | \mathbf{x})$ ，并且假设有概率 ϵ 训练数据点上的类别标签被错误设置。假设数据是独立同分布的，写出对应负对数似然的误差函数。验证当 $\epsilon = 0$ 时得到误差函数(6.33)。

请注意，与通常的交叉熵误差函数相比，此误差函数使模型对错误标记的数据具有鲁棒性。

- 6.12 (**) 用于二分类问题的误差函数 (6.33) 是针对具有逻辑 sigmoid 输出激活函数的网络推导出来的，因此 $0 \leq y(\mathbf{x}, \mathbf{w}) \leq 1$ ，并且数据具有目标值 $t \in \{0, 1\}$ 。如果我们考虑一个具有输出 $-1 \leq y(\mathbf{x}, \mathbf{w}) \leq 1$ 和目标值的网络，为类别 \mathcal{C}_1 的 $t = 1$ 和类别 \mathcal{C}_2 的 $t = -1$ 推导相应的误差函数。输出单元激活函数的合适选择是什么？
- 6.13 (*) 证明对于一个多类神经网络模型，最大化其似然性（其中网络输出具有解释 $y_k(\mathbf{x}, \mathbf{w}) = p(t_k = 1 | \mathbf{x})$ ）等同于最小化交叉熵误差函数 (6.36)。
- 6.14 (*) 证明对于具有逻辑 sigmoid 激活函数 $y_k = \sigma(a_k)$ 的输出单元，误差函数 (6.33) 关于预激活 a_k 的导数，其中 $\sigma(a)$ 由 (6.13) 给出，满足 (6.31)。
- 6.15 (*) 证明对于具有softmax激活函数 (6.37) 的输出单元，误差函数 (6.36) 关于预激活 a_k 的导数满足 (6.31)。
- 6.16 (**) 写出一对方程，用关节角度 θ_1 和 θ_2 以及连杆长度 L_1 和 L_2 来表示图6.16所示机器人手臂的笛卡尔坐标 (x_1, x_2) 。假设坐标系的原点由下臂的连接点给出。这些方程定义了机器人手臂的正向运动学。
- 6.17 (**) 证明由 (6.44) 定义的变量 γ_{nk} 可被视为混合分布 (6.38) 各分量的后验概率 $p(k | \mathbf{t})$ ，其中混合系数 $\pi_k(\mathbf{x})$ 被视为依赖于 \mathbf{x} 的先验概率 $p(k)$ 。
- 6.18 (**) 推导误差函数关于混合密度网络中控制混合系数的网络输出预激活值的导数的结果 (6.45)。
- 6.19 (**) 推导误差函数关于混合密度网络中控制分量均值的网络输出预激活值的导数的结果 (6.46)。
- 6.20 (**) 推导误差函数关于混合密度网络中控制分量方差的网络输出预激活值的导数的结果 (6.47)。
- 6.21 (***) 验证混合密度网络模型的条件均值和方差的结果 (6.48) 和 (6.50)。

深度学习



第13章 输出流形

在本章中，我们将通过把神经网络的权重和偏差视为流形上的坐标系，为每个神经网络关联一个流形。这个流形可以赋予一个黎曼度量，它描述了网络的内在几何结构。从以下几个角度来看，在这个几何框架中看待神经网络是很有用的。(i) 网络的最优权重和偏差对应于目标在流形上的正交投影的坐标。

(ii) 每个学习算法都涉及参数值随时间的变化，并且对应于这个流形上的一条曲线。最有效的学习过程对应于初始点和目标投影点之间的最短曲线，即测地线。

(iii) 正则化问题可以根据流形在周围目标空间中的平均曲率和第二基本形式来处理。也就是说，由于最平坦的流形对训练数据的过拟合最小，正则化可以被视为找到曲率最小的输出流形。

在下一节中，我们将概述流形的概念，并给出一些对深度学习有用的结果。对微分几何更多入门细节感兴趣的读者可参考[85]。

13.1 流形简介

流形是一种几何空间，至少在局部上类似于数值空间 \mathbb{R}^n 。流形中的每个点由一组 n 参数描述，这些参数被视为局部坐标。

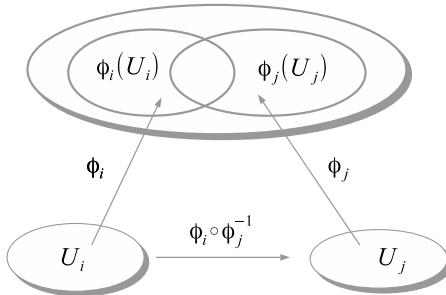


图13.1：流形的转移函数 $\phi_i \circ \phi_j^{-1}$ 是可微的。

参数的数量 n 就是流形的维度。无论参数化如何，流形都保持其自身的特性。有时，一个流形不能仅由一种参数化来定义，需要几种参数化才能覆盖整个空间。

如果参数化是可微的，即参数以可微的方式分配到流形中的点，则该流形称为可微流形。在有多个参数化 $\phi_i : U_i \rightarrow \mathcal{M}$ 的情况下，它们的过渡函数 $\phi_i \circ \phi_j^{-1}$ 必须是可微的，见图13.1。除此之外，还要求可微流形对所有参数化满足正则化条件：过渡函数 $\phi_i \circ \phi_j^{-1}$ 的雅可比矩阵具有最大秩。这个条件消除了流形上的任何尖点、角点或顶点。

我们注意到，在本书需要对神经网络进行建模的范围内，我们不会使用具有多个参数化的流形。因此，我们不会对具有多个参数化的流形进行过于详细的讨论。我们将在下面给出一些流形的例子。

示例 13.1.1（圆的流形） 平面中的圆的集合， \mathcal{C} ，可以使用三个参数组织成一个流形：圆心坐标， (a, b) ，以及圆半径， r 。参数空间是 $\mathcal{U} = \mathbb{R}^2 \times (0, \infty)$ ，流形参数化是 $\phi : \mathcal{U} \rightarrow \mathcal{C}$ ，其中 $\phi(a, b, r)$ 是以 (a, b) 为圆心且半径为 r 的圆。在这种情况下，流形 \mathcal{C} 仅由一个映射 ϕ 参数化。流形的每个元素都是一个圆，并且该流形的维度为 3。

示例13.1.2（直线流形） 设 P 为平面上的一个给定的点，其坐标为 (x_0, y_0) ，满足 $x_0 \neq 0$ 和 $y_0 \neq 0$ 。平面上经过点 P 的直线族 \mathcal{L} ，可以按以下方式视为一个流形。设 α 和 β 分别表示前面直线族中直线的 x 轴截距和 y 轴截距，见图13.2a。

那么 $\phi : \mathbb{R} \rightarrow \mathcal{L}$ ，其中 $\phi(\alpha)$ 是过 P 和 $(\alpha, 0)$ 的直线，是 \mathcal{L} 中所有非水平直线的一个参数化。类似地， $\psi : \mathbb{R} \rightarrow \mathcal{L}$ ，其中 $\psi(\beta)$ 是过 P 和 $(0, \beta)$ 的直线，是 \mathcal{L} 中所有非垂直直线的一个参数化。在这种情况下，流形 \mathcal{L} 是一维的，但它由两个参数化 ϕ 和 ψ 定义，因为它们都不能完全覆盖整个流形。转移函数 $\phi \circ \psi^{-1} : \mathbb{R} \setminus \{x_0\} \rightarrow \mathbb{R} \setminus \{y_0\}$ 由下式给出

$$\phi \circ \psi^{-1}(\alpha) = \beta = \frac{\alpha y_0}{\alpha - x_0}$$

并且是双射且可微的。因此，流形 \mathcal{L} 成为一个可微流形。

例13.1.3（曲面流形）单位上半球面

$$\mathcal{S}^+ = \{(u_1, u_2, z); (u_1, u_2) \in B(0, 1), z = (1 - u_1^2 - u_2^2)^{1/2}\}$$

是一个二维流形。它的参数化由 $\phi : B(0, 1) \rightarrow \mathbb{R}^3$ 给出，其中 $\phi(u_1, u_2) = (u_1, u_2, (1 - u_1^2 - u_2^2)^{1/2})$ ，见图13.2b。参数空间是开圆盘 $B(0, 1)$ 。值得注意的是，整个单位球面也是一个二维流形，但在这种情况下（至少）需要两种参数化（例如，从极点到在相对极点处与球面相切的平面的球极投影）。

如果坐标 z 可以写成 $z = f(u_1, u_2)$ ，其中 $f : \mathcal{U} \rightarrow \mathbb{R}^3$ 可微， \mathcal{U} 是 \mathbb{R}^2 中的开集，那么该流形是一个曲面，称为蒙日贴片。该曲面由两个实数 u_1 和 u_2 参数化。

例13.1.4（矩阵流形）具有实元素的 2×2 矩阵的集合 $\mathcal{M}_{2,2}(\mathbb{R})$ ，形成一个4维流形。参数化由 $\phi : \mathbb{R}^4 \rightarrow \mathcal{M}_{2,2}(\mathbb{R})$ 给出，

$$\phi(a, b, c, d) = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

具有实元素的 2×2 对角矩阵集 $\mathcal{D}_{2,2}(\mathbb{R})$ 构成一个二维流形。参数化是 $\phi : \mathbb{R}^2 \rightarrow \mathcal{M}_{2,2}(\mathbb{R})$ 。

$$\phi(a, d) = \begin{pmatrix} a & 0 \\ 0 & d \end{pmatrix}.$$

该矩阵仅由两个实数 a 和 d 参数化。实际上， $\mathcal{D}_{2,2}(\mathbb{R})$ 是 $\mathcal{M}_{2,2}(\mathbb{R})$ 的子流形，作为继承周围流形结构（坐标）的子集。

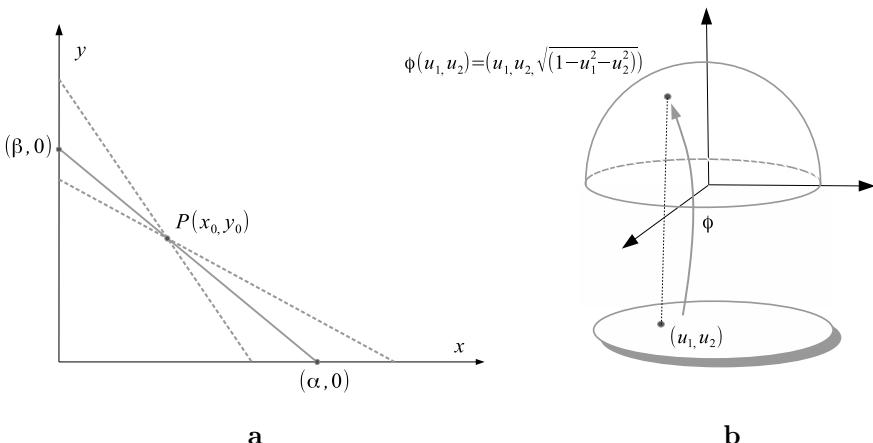


图13.2: a. 过点 $P(x_0, y_0)$ 的直线流形及其轴截距。b. 上半球面。

例13.1.5 (密度流形) 考慮所有一維高斯概率密度的集合

$$\mathcal{G} = \{p_{\mu,\sigma}; \mu \in \mathbb{R}, \sigma > 0\},$$

其中 $p_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$, $x \in \mathbb{R}$ 。集合 \mathcal{G} 成为由 μ 和 σ 参数化的二维流形。

下一个例子对本书主题具有特殊意义。

示例13.1.6 (Sigmoid神经元流形) 考虑一个具有 n 维输入 $\mathbf{x} \in \mathbb{R}^n$ 和一维输出 $y = \sigma(w^T \mathbf{x} + b)$ 的Sigmoid神经元, 其中 $w \in \mathbb{R}^n$ 和 $b \in \mathbb{R}$ 是神经元的权重和偏差。我们将 σ 取为逻辑函数。那么输出集

$$\mathcal{S} = \{\sigma(w^T \mathbf{x} + b); w \in \mathbb{R}^n, b \in \mathbb{R}\}$$

可以被视为一个由 w 和 b 参数化的 $(n + 1)$ 维流形。在下面，我们将通过证明 $y = y(w, b)$ 的雅可比矩阵的列是线性无关的来验证正则化条件。利用逻辑函数的性质，我们有

$$\begin{aligned}\frac{\partial y}{\partial b} &= \sigma'(w^T \mathbf{x} + b) = y(1 - y) \\ \frac{\partial y}{\partial w_i} &= \sigma'(w^T \mathbf{x} + b)x_j = y(1 - y)x_j,\end{aligned}$$

其中 $\mathbf{x}^T = (x_1, \dots, x_n)$ 和 $w^T = (w_1, \dots, w_n)$ 。考虑输出函数的零线性组合

$$\alpha_0 \frac{\partial y}{\partial b} + \sum_{j=1}^n \alpha_j \frac{\partial y}{\partial w_j} = 0, \quad \alpha_k \in \mathbb{R}.$$

由于 $y(1 - y) \neq 0$ ，前面的关系变为

$$\alpha_0 + \sum_{j=1}^n \alpha_j x_j = 0.$$

由于这个关系对任何 $x_j \in \mathbb{R}$ 都成立，所以可得

$\alpha_0 = \alpha_1 = \dots = \alpha_n = 0$ 。（为了证明这一点，我们选择所有 $x_j = 0$ 得到 $\alpha_0 = 0$ ；然后取 $x_j = \delta_{jk}$ 得到 $\alpha_k = 0$ ）。因此

$$\left\{ \frac{\partial y}{\partial b}, \frac{\partial y}{\partial w_1}, \dots, \frac{\partial y}{\partial w_n} \right\}$$

是线性无关的。这意味着雅可比矩阵， J_y ，具有秩 $n + 1$ 。我们可以将可微流形 \mathcal{S} 看作是在关于 \mathbb{R}^n 的函数的无限维空间中的一个 $(n + 1)$ 维光滑曲面（没有角或尖点）。

现在假设神经元经过训练以逼近连续目标函数 $z = z(\mathbf{x})$ 。如果 z 是流形 $z \in \mathcal{S}$ 上的一个点，那么存在一些参数值 $w^* \in \mathbb{R}^n$ 和 $b^* \in \mathbb{R}$ ，使得我们有精确表示 $y(w^*, b^*) = z$ 。然而，一般来说，大多数目标函数满足条件 $z \notin \mathcal{S}$ 。在这种情况下，我们需要通过训练来找到这些值

$$(w^*, b^*) = \arg \min_{w, b} \text{dist}(z, \mathcal{S}),$$

它们对应于 z 在曲面 \mathcal{S} 上的正交投影的坐标。例如，距离是在均方意义上测量的。从初始化 (w_0, b_0) 开始，学习算法应该产生一系列近似值 $(w_n, b_n)_n$ ，这些近似值收敛到投影坐标

$\lim_{n \rightarrow \infty} (w_n, b_n) = (w^*, b^*)$ 。如果参数更新是连续进行的（由无穷小学习率隐含），那么我们得到一条连接 (w_0, b_0) 和 (w^*, b^*) 的曲线 $c(t) = (w(t), b(t))$ 。这可以提升到流形 \mathcal{S} 上的曲线 $\gamma(t) = y \circ c(t)$ 。最快的学习算法对应于 $y(w_0, b_0)$ 和 $y(w^*, b^*)$ 之间的“最短”曲线。“最短”属性取决于流形 \mathcal{S} 的内在几何，这个主题将在下一节讨论。

本章将把此示例中的多种思想扩展到神经网络的一般情况。

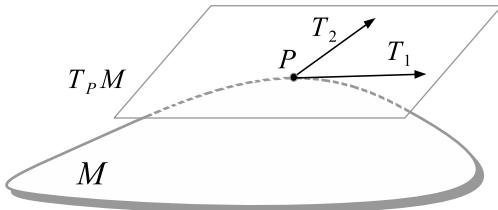


图13.3：切空间 $T_p \mathcal{M}$ 。

13.1.1 内在与外在

流形可以从两个不同的角度来看待：内在的和外在的。

内在观点是生活在流形上的局部观察者的视角。观察者的知识局限于局部坐标系。因此，在流形上测量的距离、切向量、它们的大小以及它们之间的角度都属于流形的内在几何。可以将这种观点想象成一只蚂蚁的视角，它生活在流形上，但无法获取流形之外的任何信息。

外在视角代表了从外部观察流形的观察者所获得的关于流形的知识。诸如法向量和流形形状等几何概念是外在的。这可以想象成一颗卫星的视角，它可以从外部观察流形。

在下面我们将介绍一些关于可微流形上微分几何的内在和外在概念。

13.1.2 切空间

如果流形 \mathcal{M} 有一个局部参数化 $\phi : \mathcal{U} \rightarrow \mathcal{M}$ ，其中 $\mathcal{U} \subset \mathbb{R}^n$ 是开集，那么与第 k 条坐标曲线相切的向量

$$h \rightarrow \phi(x_1, \dots, x_k + h, \dots, x_n)$$

由下式给出

$$T_k(p) = \frac{\partial \phi}{\partial x_k}(p), \quad p = \phi(x).$$

如果流形满足正则性条件，即 ϕ 在 p 处有一个满秩的雅可比矩阵，那么切向量 $\{T_1(p), \dots, T_n(p)\}$ 是线性无关的。

它们构成了一个维度为 n 的线性空间的基础，称为在 p 处 \mathcal{M} 的切空间，记为 $T_p\mathcal{M}$ ，见图13.3。

如果 $c(t)$ 是流形 \mathcal{M} 上的一条曲线，那么在局部参数化中，沿着 $c(t)$ 的速度是流形的一个切向量，其由下式给出

$$\dot{c}(t) = \sum_{k=1}^n \dot{c}^k(t) T_k(c(t)).$$

一个切向量场 U 是在每个点 $p \in \mathcal{M}$ 处切向量 $U_p \in T_p\mathcal{M}$ 的光滑赋值。在局部坐标中，它写为 $U_p = \sum_{k=1}^n U^k(p) T_k(p)$.

13.1.3 黎曼度量

流形上的主要内在概念是黎曼度量。这在局部坐标系中由一个对称、正定、非退化的矩阵 $(g_{ij})_{i,j}$ 给出，该矩阵依赖于点。其元素表示通过类似于毕达哥拉斯定理的过程测量流形上相邻点之间距离时使用的权重。因此，如果 P 和 P' 是流形上的两个相邻点，其坐标为 (x_i) 和 (x'_i) ，则它们之间的距离根据黎曼度量测量为

$$d(P, P') = \left(\sum_{i,j=1}^n g_{ij} (\Delta x)_i (\Delta x)_j \right)^{1/2},$$

其中 $(\Delta x)_i = x_i - x'_i$ ，并且 $g_{ij} = g_{ij}(P)$ 。在欧几里得空间的特殊情况下， \mathbb{R}^n ，当坐标是正交归一的时候，度量矩阵变成单位矩阵， $g_{ij} = \delta_{ij}$ 。现在距离由通常的勾股定理来度量。

$$d(P, P') = \left(\sum_{i=1}^n (\Delta x)_i^2 \right)^{1/2}.$$

通过类似的过程，我们可以将与流形相切的向量 v 的大小测量为

$$\|v\|_g = \left(\sum_{i,j=1}^n g_{ij} v^i v^j \right)^{1/2},$$

其中 v^j 表示在局部坐标系中向量的第 j 个分量。如果 g 是在 $T_p\mathcal{M}$ 中的基 $\{T_1, \dots, T_n\}$ 上由 $g(T_i, T_j) = g_{ij}$ 定义的双线性形式，那么前面的公式可以写成

$$\|v\|_g = g(v, v)^{1/2}.$$

系数 g_{ij} 在历史上被称为第一基本形式 g 的系数。

在欧几里得空间中，这变成了以下熟悉的向量长度公式：

$$\|v\| = \left(\sum_{i,j=1}^n (v^i)^2 \right)^{1/2} = \langle v, v \rangle^{1/2}.$$

如果现在 $c(t)$ 表示位于流形上并由变量 t 参数化的曲线，其速度向量 $\dot{c}(t)$ 是流形的切向量，其大小由下式给出

$$\|\dot{c}(t)\|_g = \left(\sum_{i,j=1}^n g_{ij}(c(t)) \dot{c}(t)^i \dot{c}(t)^j \right)^{1/2} = g(\dot{c}(t), \dot{c}(t))^{1/2}.$$

对速度 $\|\dot{c}(t)\|$ 关于时间 t 进行积分，我们得到相对于黎曼度量测量的曲线长度

$$L(c) = \int_a^b \|\dot{c}(t)\|_g dt,$$

其中 $a \leq t \leq b$ 。流形 \mathcal{M} 上两点 A 和 B 之间的黎曼距离定义为连接这两点的最短曲线的长度

$$d(A, B) = \inf_c \{ L(c); c : [a, b] \rightarrow \mathcal{M}, c(a) = A, c(b) = B \}.$$

对 (\mathcal{M}, g) 称为黎曼流形。本章将描述使用黎曼流形的神经网络。

13.1.4 测地线

另一个内在概念是测地线的概念。这是流形上两点之间的最短曲线。如果两点足够接近，它们之间总是存在一条测地线，并且这条测地线是唯一的。在局部坐标中，测地线 $c(t)$ 可以由一个非线性方程组描述为

$$\ddot{c}^k(t) + \sum_{i,j} \Gamma_{ij}^k \dot{c}^i(t) \dot{c}^j(t) = 0, \quad 1 \leq k \leq n, \quad (13.1.1)$$

其中 $\Gamma_{ij}^k = \Gamma_{ij}^k(c(t))$ 是第二类克里斯托费尔符号，例如参见[85]

$$\Gamma_{ij}^k = \frac{1}{2} g^{kr} \left(\frac{\partial g_{ir}}{\partial x_j} + \frac{\partial g_{jr}}{\partial x_i} - \frac{\partial g_{ij}}{\partial x_r} \right), \quad (13.1.2)$$

其中 (x_1, \dots, x_n) 表示流形上的局部坐标。由于它们依赖于度量系数 g_{ij} ，克里斯托费尔符号是内在的，因此测地线也是。接下来给出两个简单的例子。

1. 在欧几里得空间 $(\mathbb{R}^n, \delta_{ij})$ 中, 度量系数 δ_{ij} 的导数为零, 测地线方程变为 $\ddot{c}^k(t) = 0$ 。这意味着测地线是直线。

2. 在二维球面 S^2 的情况下, 测地线是大圆的弧。这里给出测地线方程会更复杂, 读者可参考一本微分几何的书。

测地线的显式计算只能在极少数特定类型的流形上进行, 并且这是一个相对复杂的过程。

13.1.5 列维 - 奇维塔联络

我们将从欧几里得空间的情况开始。设 f 是在 \mathbb{R}^n 上的一个可微函数, v 是一个向量。 f 关于 v 的方向导数由 $v(f) = \langle v, \nabla f \rangle$ 定义, 其中 $(\nabla f)^T = (\partial_{x_1} f, \dots, \partial_{x_n} f)$ 表示 f 的梯度。对象 $v(f)$ 表示 f 在 v 方向上的变化率。

设 $U = (U^1, \dots, U^n) = \sum_{i=1}^n U^i e_i$ 为 \mathbb{R}^n 上的一个向量场。 U 关于向量 v 的导数定义为

$$\nabla_v U = (v(U^1), \dots, v(U^n)) = \sum_{i=1}^n v(U^i) e_i. \quad (13.1.3)$$

现在, 设 $V = (V^1, \dots, V^n)$ 为 \mathbb{R}^n 上的另一个向量场。我们将 U 关于 V 的协变导数定义为

$$(\nabla_V U)_p = \nabla_v U,$$

其中 $v = V_p$, 右侧的项如(13.1.3)所定义。我们注意到 $\nabla_V U$ 是 \mathbb{R}^n 上的一个向量场, 它将向量 $(\nabla_V U)_p$ 与每个点 $p \in \mathbb{R}^n$ 相关联。

不难证明, 对于 \mathbb{R}^n 上的任何可微函数 f 以及 \mathbb{R}^n 上的任何向量场 U, V 和 W , 以下性质成立:

- (i) $\nabla_{fU} V = f \nabla_U V$,
- (ii) $\nabla_U (V + W) = \nabla_U V + \nabla_U W$,
- (iii) $\nabla_U (fV) = U(f)V + f \nabla_U V$ 。

下一个性质给出了求导与内积之间的相容性, 它是乘积法则的一个结果:

$$(iv) W \langle U, V \rangle = \langle \nabla_W U, V \rangle + \langle U, \nabla_W V \rangle,$$

协变微分的不可交换性由以下公式给出

(v) $\nabla_U V - \nabla_V U = [U, V]$ ，其中 $[U, V] = UV - VU$ 是向量场 U 和 V 的换位子。通过直接计算可以表明，换位子始终是一个向量场，其分量由下式给出

$$[U, V]^i = U(V^i) - V(U^i) = \sum_{j=1}^n \left(e_j(V^i)U^j - e_j(U^i)V^j \right).$$

现在考虑一个黎曼流形 (\mathcal{M}, g) 。流形 \mathcal{M} 上的线性联络 ∇ 是一个作用于 \mathcal{M} 的向量场的算子，满足上述性质 (i)-(iii)。流形上有许多线性联络。它们与黎曼结构 g 无关。然而，只有一种线性联络同时满足性质 (iv) 和 (v)。

定理13.1.7 在黎曼流形 (\mathcal{M}, g) 上存在唯一的线性联络，使得

$$\begin{aligned} Wg(U, V) &= g(\nabla_W U, V) + g(U, \nabla_W V), \\ \nabla_U V - \nabla_V U &= [U, V], \end{aligned}$$

对于 \mathcal{M} 上的所有向量场 U, V, W 。

这被称为 (\mathcal{M}, g) 上的列维 - 奇维塔联络。由于度量 g 唯一地确定了这个联络，所以列维 - 奇维塔联络是一个内蕴概念。关于这个联络的存在性和唯一性的证明，读者可参考例如书籍[22]的第7章。

值得注意的是，列维 - 奇维塔联络是一个内蕴概念，因为它通过克里斯托费尔符号(13.1.2)依赖于度量系数 g_{ij} ，如

$$\nabla_{T_i} T_j = \sum_k \Gamma_{ij}^k T_k,$$

其中 $\{T_1, \dots, T_n\}$ 是切空间的一个基。

13.1.6 子流形

设 \mathcal{M} 和 \mathcal{S} 为两个流形，使得 $\mathcal{S} \subset \mathcal{M}$ 且 \mathcal{S} 赋予了从 \mathcal{M} 诱导的拓扑和可微结构。那么 \mathcal{M} 上的任何黎曼度量 g 在 \mathcal{S} 上诱导出一个黎曼结构如下

$$h(U, V) = g|_{\mathcal{S}}(U, V),$$

其中 U, V 是 \mathcal{S} 上的向量场， $h = g|_{\mathcal{S}}$ 是 g 在 \mathcal{S} 的向量场上的限制。那么 (\mathcal{S}, h) 成为 (\mathcal{M}, h) 的一个黎曼子流形。

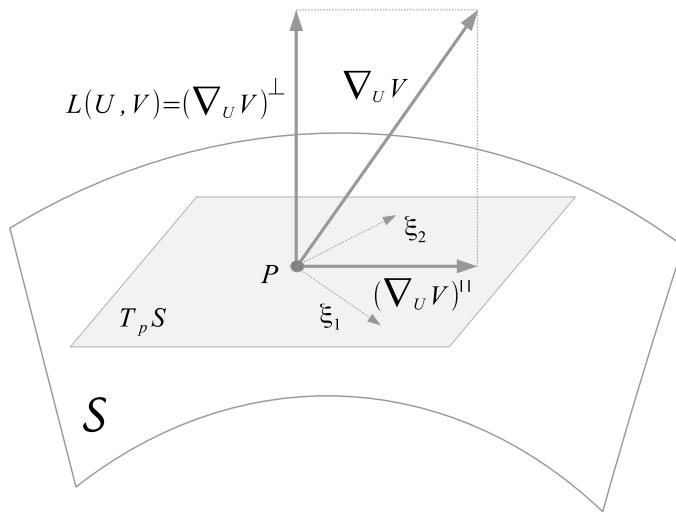


图13.4：高斯公式 $\nabla_U V = (\nabla_U V)^{\parallel\parallel} + L(U, V)$ 。

设 n 和 m 分别为 \mathcal{M} 和 S 的维度。那么对于任意点 $p \in S$ ，切空间 $T_p S$ 是线性空间 $T_p \mathcal{M}$ 中维度为 m 的线性子空间。我们可以考虑正交分解

$$T_p \mathcal{M} = T_p S \oplus \mathcal{V}_p, \quad (13.1.4)$$

其中 $\mathcal{V}_p = \{v; g(v, u) = 0, \forall u \in T_p \mathcal{M}\}$ 。这意味着对于任何向量 $w \in T_p \mathcal{M}$ ，存在两个正交向量 $u \in T_p S$ 和 $v \in \mathcal{V}_p$ ，使得 $w = u + v$ 。

13.1.7 第二基本形式

子流形相对于周围流形的形状由其第二基本形式描述。设 S 是黎曼流形 (\mathcal{M}, g) 的一个子流形，并用 ∇ 表示 (\mathcal{M}, g) 上的列维-奇维塔联络。那么对于子流形 S 上的任意两个向量场 U 和 V ，向量场 $\nabla_U V$ 不一定是 S 上的向量场。一般来说，根据 (13.1.4) 我们有以下正交分解：

$$(\nabla_U V)_p = (\nabla_U V)_p^{\parallel} + (\nabla_U V)_p^{\perp}, \quad (13.1.5)$$

带有 $(\nabla_U V)_p^{\parallel} \in T_p S$ 和 $(\nabla_U V)_p^{\perp} \in \mathcal{V}_p$ 。关系式(13.1.5)称为高斯公式，见图13.4。

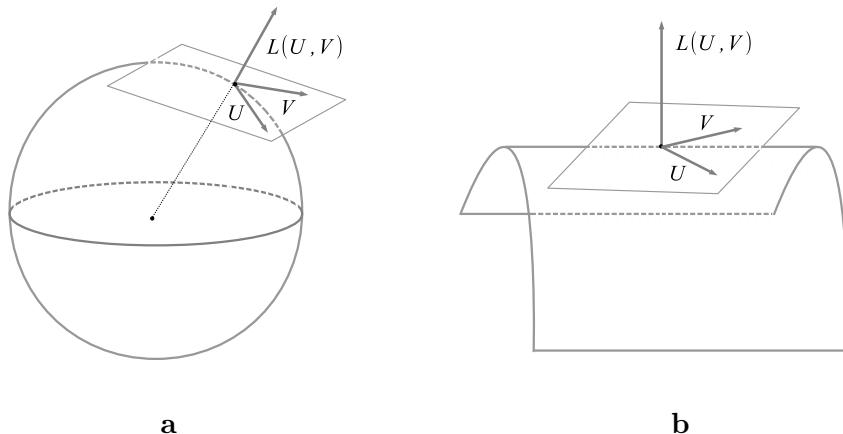


图13.5：第二基本形式垂直于曲面并描述其形状：a. 球体的情况。b. 圆柱面的情况。

可以证明算子 $\bar{\nabla} = \nabla^{\parallel}$ 是子流形 $(S, g|_S)$ 上的列维 - 奇维塔联络。

子流形 S 关于 M 的第二基本形式定义如下

$$L(U, V) = (\nabla_U V)^\perp \quad (13.1.6)$$

对于 S 上的任意 U, V 向量场, 见图13.5。

命题13.1.8 设 U, V, W 是 \mathcal{S} 上的任意向量场。第二基本形式满足以下性质：

- (i) L 是对称的, $L(U, V) = L(V, U)$;
 - (ii) L 是双线性的: $L(U + W, V) = L(U, V) + L(W, V)$;
 - (iii) 对于 \mathcal{S} 上的任意两个可微函数 f_1 和 f_2 ,
 $L(f_1U, f_2V) = f_1f_2L(U, V)$ 。

证明：(i) 利用列维 - 奇维塔联络 ∇ 和 $\bar{\nabla}$ 的性质，我们证明 L 是对称的

$$\begin{aligned}
L(U, V) - L(V, U) &= \left(\nabla_U V - \bar{\nabla}_U V \right) - \left(\nabla_V U - \bar{\nabla}_V U \right) \\
&= \left(\nabla_U V - \nabla_V U \right) - \left(\bar{\nabla}_U V - \bar{\nabla}_V U \right) \\
&= [U, V] - [U, V] \\
&\equiv 0.
\end{aligned}$$

其他性质很容易验证，留作读者练习。 ■

值得注意的是，性质(iii)表明，如果向量场 U 和 V 被函数 f_1 和 f_2 缩放，那么第二基本形式会被乘积 $f_1 f_2$ 缩放。

13.1.8 平均曲率向量场

设 (\mathcal{S}, h) 为黎曼流形 (\mathcal{M}, g) 的子流形，具有 $h = g|_{\mathcal{S}}$ ，并记第二基本形式为 L 。对于任意固定点 $p \in \mathcal{S}$ ，我们考虑切空间 $T_p \mathcal{S}$ 中的一个向量基 $\{T_1, \dots, T_m\}$ ，并定义 $L_{ij} = L(T_i, T_j)$ ，它是 \mathcal{S} 的法向量，对于任意 $1 \leq i, j \leq m$ ，因为 $L_{ij} \in \mathcal{V}_p$ 。设 $h_{ij} = h(T_i, T_j)$ 为 \mathcal{S} 上黎曼度量的系数， h^{ij} 表示逆矩阵 $(h_{ij})^{-1}$ 的系数。

在 p 处的平均曲率向量由以下“缩并”定义：

$$H_p = \sum_{i,j} h^{ij}(p) L_{ij}.$$

我们有 $H_p \in \mathcal{V}_p$ ，并且映射 $p \rightarrow H_p$ 定义了 \mathcal{S} 的平均曲率向量场，它在 \mathcal{S} 的每一点处都与 \mathcal{S} 垂直。

如果我们假设初始选择的基是正交归一的，那么定义会稍微简化一些。如果 $\{E_1, \dots, E_m\}$ 是 $T_p \mathcal{S}$ 中的一个正交归一基，那么 $h(E_i, E_j) = \delta_{ij}$ 以及在 p 处的平均曲率向量变为

$$H_p = \sum_{i,j} \delta_{ij} L(E_i, E_j) = \sum_{i=1}^m L(E_i, E_i).$$

平均曲率向量场是一个外在概念，因为它依赖于第二基本形式。

第二基本形式和平均曲率度量子流形的不同类型的外在曲率。例如，如果第二基本形式为零， $L_{ij} = 0$ ，那么子流形 \mathcal{S} 被称为完全测地的。这意味着 \mathcal{S} 的任何测地线也是 \mathcal{M} 的测地线。为了理解这个概念，读者应该想象三维空间中包含的平面的特殊情况：平面中的任何直线也是空间中的直线。这与平面在空间中不弯曲的事实是一致的。

如果平均曲率向量场为零， $H = 0$ ，那么 \mathcal{S} 被称为 \mathcal{M} 的极小曲面。几何解释是 \mathcal{S} 局部具有最小体积；也就是说，如果流形在局部受到扰动，那么它的体积会增加。第二基本形式和平均曲率的概念稍后将用于正则化目的。

13.2 与神经网络的关系

读者可能会问，到目前为止所介绍的微分几何概念与神经网络之间有什么关系。本节将简要讨论这种关系，而后面的章节将进行更详细的分析。

给定神经网络的作用是逼近某个目标函数 z 。我们假设 z 是目标流形 \mathcal{M} 的一个元素，例如 $[0, 1]$ 上连续函数的流形。神经网络的输出 y 由网络的权重和偏差 $\theta = (w, b)$ 参数化。这样，输出 y 属于一个输出流形 \mathcal{S} ，它被认为是目标流形 \mathcal{M} 的一个子流形。子流形 \mathcal{S} 的维度等于网络权重和偏差的数量，而在这种情况下 \mathcal{M} 的维度是无限的。¹ 值得注意的是，对于实际应用，目标是一个向量 $z^T = (z_1, \dots, z_n)$ ，这意味着流形 \mathcal{M} 的维度为 n 。

通过添加更多神经元进而增加更多参数，可以提高输出流形的维度。维度越大，逼近效果越好。然而，子流形 \mathcal{S} 的形状在避免过拟合方面也起着决定性作用。第二基本形式 L 描述了子流形 \mathcal{S} 在 \mathcal{M} 内部是如何弯曲的。从正则化的角度来看，我们更喜欢弯曲尽可能小的子流形，以便目标函数 z 在 \mathcal{S} 上的正交投影最终是唯一的，并且易于通过梯度下降法找到。

例如，我们将考虑示例13.1.6给出的流形的情况。在这种情况下，目标流形可以选择为空间 $\mathcal{M} = \mathcal{C}[0, 1]$ 。输出流形

$$\mathcal{S} = \{\sigma(w^T \mathbf{x} + b); w \in \mathbb{R}^n, b \in \mathbb{R}\}$$

是 \mathcal{M} 中的一个 $(n+1)$ 维子流形。在每一点 $y \in \mathcal{S}$ 处，切空间 $T_p \mathcal{S}$ 由这些函数的线性组合张成

$$\{y(1-y), y(1-y)x_1, \dots, y(1-y)x_n\}.$$

子流形 \mathcal{S} 可视为实值连续函数空间 $\mathcal{M} = \mathcal{C}([0, 1])$ 内的一个 $(n+1)$ 维超曲面。

¹ 定义在 $[0, 1]$ 上的连续函数的一个参数系是有理数集 $Q \cap [0, 1]$ 。

对于单个Sigmoid神经元，目标空间 \mathcal{M} 不能被曲面 \mathcal{S} 很好地逼近，因为存在连续函数 $f \in \mathcal{M}$ ，其到曲面 \mathcal{S} 的距离不能被任意缩小。等效地说，使用 \mathcal{S} 来逼近 \mathcal{M} 会导致欠拟合。

然而，增加网络中的神经元数量会导致更多的参数，从而得到更高维的逼近流形 \mathcal{S} 。希望是对于目标空间的任何元素 $f \in \mathcal{M}$ 以及任何固定的 $\epsilon > 0$ ，存在一个网络，它能产生足够高维的流形 \mathcal{S} ，使得 $\text{dist}(f, \mathcal{S}) < \epsilon$ ，其中距离是通过……来度量的。

$$\text{dist}(f, \mathcal{S}) = \inf_{s \in \mathcal{S}} \max_{x \in [0,1]} |f(x) - s(x)|.$$

通过添加更多神经元， \mathcal{S} 的维度可以明显增加，直到达到所需的近似效果。然而，在实际中，神经元的供应可能是有限的，这一事实导致了在保持神经元数量不变的情况下最大化 \mathcal{S} 维度的问题。我们将在下一节处理这个问题。

13.3 参数空间

假设我们有固定数量的计算单元 N ，并且需要使用这些单元作为隐藏神经元构建一个前馈神经网络，以使网络获得其最大容量，即它具有逼近目标函数的最大能力。具体来说，我们将寻找仅考虑 N 个隐藏神经元并为输出流形 \mathcal{S} 产生最大维度的网络架构。当网络参数数量最大化时即可实现这一点。

我们将从一个例子开始。假设我们有 $N = 10$ 个隐藏神经元，并考虑以下三种从前到深排序的前馈网络架构：

- (i) 仅一个隐藏层，该层中有 $N = 10$ 个隐藏神经元；
- (ii) 2个隐藏层，每层有5个隐藏神经元；
- (iii) 5个隐藏层，每层有2个隐藏神经元。

为简单起见，我们假设输入和输出都是一维的。

(i) 给出的网络有30个参数：10个权重 w_j （从输入到隐藏层）；10个偏差 b_j （每个隐藏神经元一个）；10个权重 α_j （从隐藏层到输出）。见图13.6。

(ii) 给出的网络有45个参数：5个权重 w_j （从输入到第一个隐藏层）； 5^2 个权重 $w_{ij}^{(1)}$ （从第一个隐藏层到第二个隐藏层）；5个权重 α_j （从第二个隐藏层到输出）；10个偏差 b_j （每个隐藏层中的每个神经元一个）。见图13.7a。

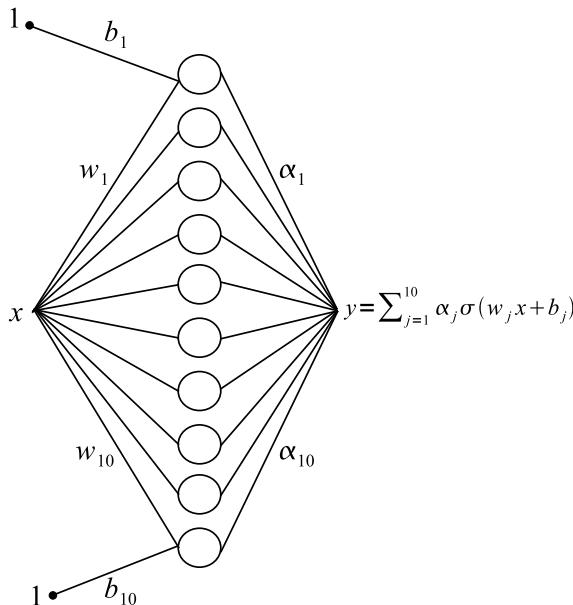


图13.6：一个具有 $N = 10$ 个隐藏神经元的单隐藏层神经网络，它依赖于30个参数。

(iii)给出的网络有30个参数：2个权重 w_j （从输入到第一个隐藏层）；16个中间权重（每两个隐藏层之间有4个）；2个权重 α_j （从最后一个隐藏层到输出）；10个偏差 b_j 。该网络如图13.7b所示。

我们得出结论，在情况(ii)中达到了参数的最大数量，而(ii)既不是所考虑的最浅的神经网络，也不是最深的神经网络。我们将证明这是一般前馈神经网络架构的典型行为。

我们现在考虑一个具有 $L - 1$ 个隐藏层的神经网络（第0层和 L 层分别保留用于输入层和输出层）。和往常一样，我们用 $d^{(\ell)}$ 表示第 ℓ 层中的神经元数量。出于简单起见，我们选择 $d^{(0)} = d^{(L)} = 1$ 。由于隐藏神经元的数量等于 N ，我们有

$$\sum_{\ell=1}^{L-1} d^{(\ell)} = N. \quad (13.3.7)$$

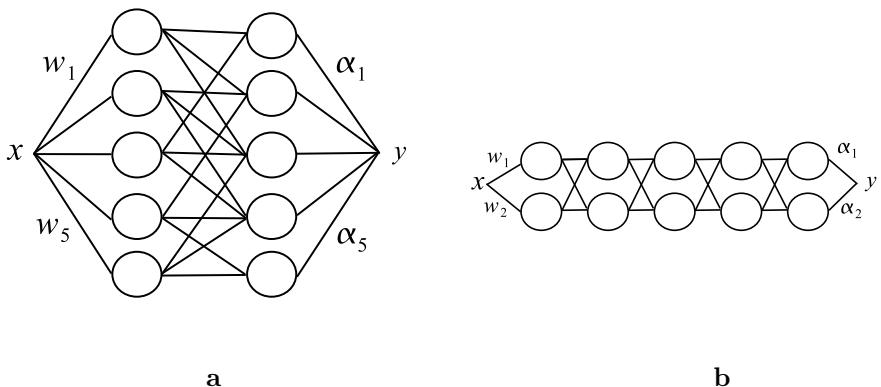


图13.7：具有 $N = 10$ 个神经元的两个神经网络：a. 一个依赖于45个参数的2隐藏层神经网络。b. 一个依赖于30个参数的5隐藏层神经网络。

层 $\ell - 1$ 和 ℓ 之间的权重 $w_{ij}^{(\ell)}$ 数量为 $d^{(\ell-1)}d^{(\ell)}$ 。偏差的总数等于 N ，每个隐藏神经元一个。那么，包括权重和偏差在内的参数总数由下式给出

$$d^{(0)}d^{(1)} + d^{(1)}d^{(2)} + \cdots + d^{(\ell-1)}d^{(\ell)} + \cdots + d^{(L-1)}d^{(L)} + N. \quad (13.3.8)$$

现在，找到最大容量前馈神经网络的问题可以等效地表述为：

在给定约束条件(13.3.7)的情况下, L 的层数、每个隐藏层 ℓ 中的神经元数量 $d^{(\ell)}$ 以及 $1 \leq \ell \leq L - 1$ 为何值时, 表达式(13.3.8)能达到最大值?

这个问题具有以下几何意义。每个乘积项 $d^{(\ell-1)}d^{(\ell)}$ 被解释为一个矩形的面积。然后，从一个尺寸为 $d^{(0)} \times d^{(1)}$ 的矩形开始，我们按照图13.8继续构建一个尺寸为 $d^{(1)} \times d^{(2)}$ 的矩形。偶数边垂直显示，奇数边水平显示。整个图形可以内接于一个矩形 \mathcal{R} ，其宽度等于 $d^{(1)} + d^{(3)} + \dots$ ，高度由 $d^{(0)} + d^{(2)} + \dots$ 给出。约束条件(13.3.7)在几何上等同于矩形 \mathcal{R} 的尺寸（即宽度和高度）之和是常数，等于 N 这一事实。目标是在上述约束条件下最大化矩形面积之和。

一种近似方法是首先最大化矩形 R 的面积。当宽度等于高度时，即当它变成正方形时，就会出现这种情况。然后，我们尝试使用之前的构造用矩形填充整个正方形 R ，以便尽可能多地填充它。

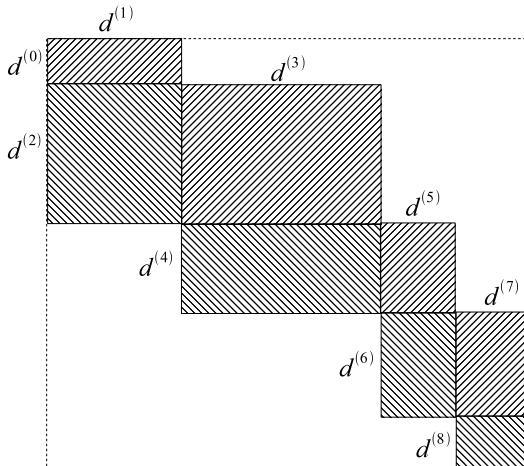


图13.8：和 $d^{(0)}d^{(1)} + d^{(1)}d^{(2)} + \dots + d^{(7)}d^{(8)}$ 的面积意义。

一个组合论证表明，只有当我们有两个隐藏层时才会出现最优构造。在这种情况下，情况如图13.9 a所示。如果我们有更多层，比如说3层，那么构造就不是最优的，因为会剩下更多未填充的空间，见图13.9 b。读者应该能够补充论证中缺失的细节。

如果我们从一开始就假设，这个问题有一个精确的数学解

$$d^{(1)} = d^{(2)} = \dots = d^{(L-1)},$$

也就是说，当每个隐藏层具有相同数量的神经元时。为了简单起见，令 $k = L - 1$ 表示网络中的隐藏层数，因此每个隐藏层有 N/k 个神经元。那么由(13.3.8)给出的权重数量变为

$$f_N(k) = \frac{N}{k} + (k-1)\left(\frac{N}{k}\right)^2 + \frac{N}{k} + N.$$

如下将针对 k 对其进行优化。我们首先根据 $1/k$ 重写该表达式，如下所示

$$\begin{aligned} f_N(k) &= \frac{2N}{k} + N^2 \frac{k-1}{k^2} + N \\ &= \frac{N}{k} \left(2 + N - N \frac{1}{k}\right) + N. \end{aligned}$$

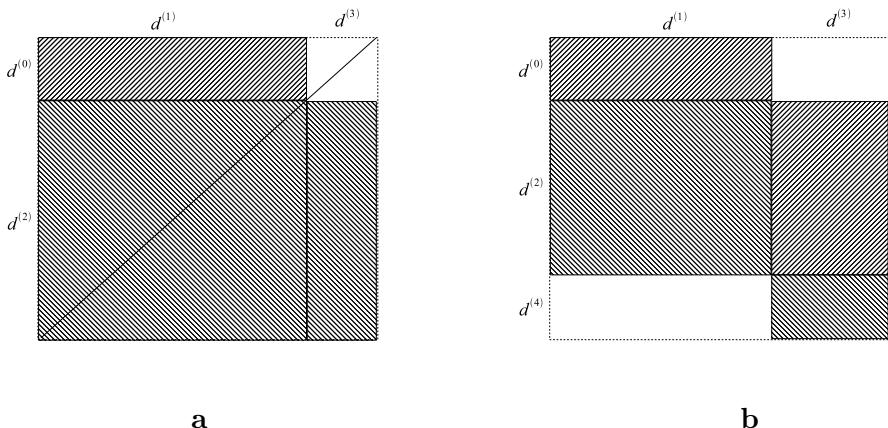


图13.9：a. 空白矩形是一个面积为 $d^{(0)}d^{(3)} = 1$ 的正方形。b. 空白矩形的面积大于1。

代入 $u = 1/k$ ，我们得到一个关于 u 的二次函数

$$\phi_N(u) = Nu \left(2 + N - Nu \right) + N = -N^2 u^2 + N(N+2)u + N.$$

$\phi_N(u)$ 的最大值在……时达到

$$u = \frac{N(N+2)}{2N^2} = \frac{N+2}{2N}.$$

这对应于由……给出的隐藏层数

$$k = \frac{2N}{N+2}.$$

即使这个数字不一定总是整数，但对于大量神经元 N ，隐藏层的最佳数量可以很好地用 $k = 2$ 近似。这就解释了为什么在 $N = 10$ 的情况下，拥有一个具有两个隐藏层、每个隐藏层有5个神经元的网络能达到最大容量。

参数的理论最大数量由 $f_N\left(\frac{2N}{N+2}\right)$ 给出。实际上，这等于值—

$$f_N(2) = \frac{N^2}{4} + 2N. \quad (13.3.9)$$

因此，输出流形 \mathcal{S} 的最大维度在隐藏神经元 N 的数量上呈二次增长。接下来我们将证明，这有时可能会导致数据过拟合。

前面的公式是在满足 $d^{(0)} = d^{(L)} = 1$ 的网络的特定情况下推导出来的。对于一般情况，请参见练习13.8.2。

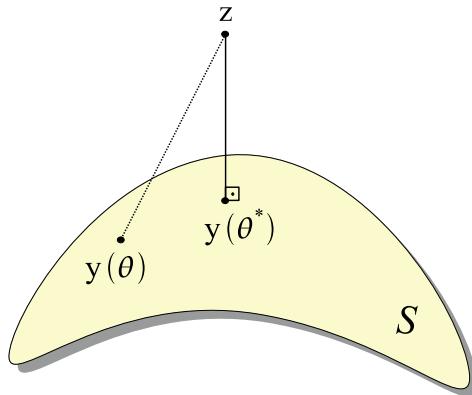


图13.10： \mathbf{z} 在流形 \mathcal{S} 上的正交投影是 $\mathbf{y}(\theta^*)$ 。

13.4 最优参数值

考虑训练集 $\{(x_1, z_1), (x_2, z_2), \dots, (x_n, z_n)\}$ ，并令 y_i 为网络对应于一维输入 x_i 的一维输出，其中 $1 \leq i \leq n$ 。那么 $y_i = y_i(\theta)$ ，其中 $\theta \in \mathbb{R}^r$ 是网络的参数向量。因此，向量 $\mathbf{y}^T = (y_1, \dots, y_n) \in \mathbb{R}^n$ 由 θ 参数化，从而在 \mathbb{R}^n 中描述了一个维度为 r 的流形 \mathcal{S} 。实际上，输出流形 \mathcal{S} 是 \mathbb{R}^n 中的一个 r 维曲面。训练网络等同于找到 θ^* 的精确值或近似值，对于该值，从 $\mathbf{z}^T = (z_1, \dots, z_n)$ 到 \mathcal{S} 的距离最小，即

$$\theta^* = \arg \min_{\theta} \text{dist}(\mathbf{z}, \mathcal{S}) = \arg \min_{\theta} \|\mathbf{z} - \mathbf{y}(\theta)\|,$$

其中距离是在欧几里得意义下测量的。这等同于 $\mathbf{y}(\theta^*)$ 是 \mathbf{z} 在 \mathcal{S} 上的正交投影这一事实，见图13.10。

如果网络有两个隐藏层，那么先前的分析给出 $r = \frac{N^2}{4} + 2N$ ，其中 N 是隐藏神经元的数量。如果 N 满足 $\frac{N^2}{4} + 2N \geq n$ ，那么网络表现出过拟合，因为它记住了整个训练集。我们将补充这一事实的一些数学细节。

假设我们仅有等式， $\frac{N^2}{4} + 2N = n$ 。那么子流形 \mathcal{S} 与目标空间 \mathbb{R}^n 具有相同的维度，因此它要么是整个空间，要么是其一个子集。所以，我们可以选择参数 θ ，使得点 \mathbf{z} 属于流形 \mathcal{S} ，这一事实意味着上述距离等于零。接下来给出这一推导的详细过程。

n 方程组

$$\begin{aligned}y_1(\theta) &= z_1 \\ \dots &= \dots \\ y_n(\theta) &= z_n\end{aligned}$$

有 n 个未知数， $\theta = (\theta_1, \dots, \theta_n)$ 。假设正则性假设成立，我们可以求该方程组的逆

$$\det\left(\frac{\partial y_j}{\partial \theta_i}\right) \neq 0,$$

从几何角度讲，这相当于表明输出流形 \mathcal{S} 在每个点都有一个切平面（流形上没有角点或尖点）。那么该方程组有唯一解， $\theta^* = \theta^*(\mathbf{y}, \mathbf{z})$ 。这表明 \mathbf{z} 是流形 \mathcal{S} 上的一个点，具有相应的参数 θ^* 。

我们注意到不等式 $\frac{N^2}{4} + 2N > n$ 不成立，因为这将意味着子流形的维度大于空间本身。等效地，这意味着上述系统的未知数 θ 的数量大于方程的数量 n 。这将意味着权重和偏差不能独立，这对于神经网络来说是一个矛盾的陈述。

示例13.4.1 我们将提供一个应用于MNIST数据集的示例，该数据集附带 $n = 55,000$ 训练数据， $\{(\mathbf{x}_i, \mathbf{z}_i)\}$ 。与先前的理论存在一些差异，因为每个输入数据 \mathbf{x}_i 是784维的（每个 28×28 图像被展平为一个784维向量），而每个输出 \mathbf{z}_i 是10维的（有10个数字类别）。在计算输出流形的维度时应考虑这些修改。在下面，我们将采用 $N = 500$ 个神经元并考虑以下前馈神经网络架构：

- (i) 784 - 500 - 10：具有 $N = 500$ 个神经元的单隐藏层网络；
- (ii) 784 - 250 - 250 - 10：每层具有250个神经元的双隐藏层网络；
- (iii) 784 - 50 - 50 - ... - 50 - 10：具有10个隐藏层、每层50个神经元的网络；

在这个例子中，我们有 $d^{(L)} = 10$ ，因为每个目标向量 \mathbf{z}_i 是10维的。由于有55,000个目标，这产生了一个维度为 $55,000 \times 10 = 550,000$ 的空间。因此，输出流形 \mathcal{S} 是数值空间 $\mathbb{R}^{550,000}$ 的一个子流形。接下来我们将讨论上述架构的输出流形的维度。

在 (i) 的情况下，输出流形 \mathcal{S} 的维度是

$$r = 784 \times 500 + 500 \times 10 + 500 = 397,500$$

由于从输入层到隐藏层有784个权重、500个偏置，以及从隐藏层到输出层的 500×10 个权重。在这种情况下，准确率约为97.2%，测试误差约为1.49。较小的测试误差表明网络处于最优状态（既没有过拟合也没有欠拟合）。

在情况 (ii) 中，输出流形的维度是

$$r = 784 \times 250 + 250 \times 250 + 250 \times 10 + 500 = 261,500,$$

大约是目标空间维度的一半。在这种情况下，网络的准确率为96.5%，低于前一个网络。并且测试误差为565，比前一种情况的测试误差大。在情况 (iii) 中，流形的维度是

$$r = 784 \times 50 + 9 \times 50 \times 50 + 500 + 500 = 62,700,$$

大约比目标空间的维度小17倍。这导致数据欠拟合，测试误差为1027就表明了这一事实。在这种情况下，准确率仅为94.5%。我们注意到随着网络加深，准确率会下降。

这些计算使用了40的批量大小，经过4000次迭代执行。我们采用了Adam优化算法，随着我们接近最优点，该算法会减小学习步长。所使用的成本函数是平方误差之和。

基于信息的学习还有另一种使用信息场的概念来表征最优参数的方法。考虑一个具有输入 X 和输出 Y 的两层神经网络，其激活函数 ϕ 不断增加。输出与输入的关系为

$$Y = \phi(W^T X + B).$$

由于 ϕ 是可逆的，那么 $W^T X + B = \phi^{-1}(Y)$ ，并且根据命题 D.5.1 我们有 $\mathcal{G}(Y) = \mathcal{G}(W^T X + B)$ 。由于添加常数不会改变信息场，见练习11.10.1，我们有 $\mathcal{G}(Y) = \mathcal{G}(W^T X)$ 。因此，输出信息 $\mathcal{E} = \mathcal{G}(Y)$ 与偏差 B 无关，但取决于权重系统 W 。由于 (W, B) 是相关输出流形上的坐标，我们可以将与该流形上每个点相关的参数依赖的西格玛场 $\mathcal{E} = \mathcal{E}_{W,B}$ 与之关联。信息在坐标 B 的方向上得以保留，即，它在切片 $\{W = \text{常数}\}$ 时是守恒的。

对于一个具有输入 X 和输出 \tilde{Y} 的三层神经网络，其输出与输入的关系为

$$\tilde{Y} = \phi\left(W^{(2)T} \phi(W^T X + B) + B^{(2)}\right).$$

与之前的计算类似，我们可以证明输出信息是

$$\tilde{\mathcal{E}} = \mathfrak{S}(\tilde{Y}) = \mathfrak{S}(W^{(2)}^T \phi(W^T X + B)).$$

这取决于权重 $W, W^{(2)}$ 以及第一层的偏差 B ，但与最后一层的偏差 $B^{(2)}$ 无关。

一般来说，任何前馈神经网络的输出流形都可以赋予一种信息结构。流形上的每个点都由一些权重和偏差参数化，我们将一个信息场与这些权重和偏差相关联。

如果 $\mathcal{I}, \mathcal{E}_{W,B}$ 和 \mathcal{Z} 分别表示输入、输出和目标信息字段，我们假设双重包含关系成立

$$\mathcal{E}_{W,B} \subset \mathcal{Z} \subset \mathcal{I}.$$

给定信息 $\mathcal{E}_{W,B}$ ，那么目标变量 Z 最好由条件期望 $\mathbb{E}[Z | \mathcal{E}_{W,B}]$ 来近似。最优权重由下式给出

$$(W^*, B^*) = \arg \min_{W, B} \|Z - \mathbb{E}[Z | \mathcal{E}_{W,B}]\|_{L^2}.$$

与权重 (W^*, B^*) 相对应的网络输出在信息意义上是目标 Z 的最佳近似。

接下来我们将指出精确学习的情况。现在假设，对于某些参数 (W, B) ，我们有

$$\mathcal{Z} \subset \mathcal{E}_{W,B} \subset \mathcal{I}.$$

那么任何目标变量 Z ，即一个关于 \mathcal{Z} 可测的随机变量，也是关于 $\mathcal{E}_{W,B}$ 可测的，在这种情况下 $Z = \mathbb{E}[Z | \mathcal{E}_{W,B}]$ 。这对应于精确学习，因为 Z 完全由网络的输出信息确定。

13.5 度量结构

我们已经看到，隐藏神经元的数量 N 决定了输出流形 \mathcal{S} 的维度。在有两个隐藏层的情况下，这个维度最大。然而，如果 r 小于最大值，可能会有几个相同维度 r 的流形，它们与不同的前馈网络架构相关联。

例如，图13.11 a、b中所示的编码器和解码器网络依赖于相同数量的参数，并且具有相同数量的隐藏神经元。然而，它们的任务却大不相同。这可以通过网络架构（这里指数字序列 $d^{(\ell)}$ ）对流形 \mathcal{S} 产生不同的度量效应来解释。

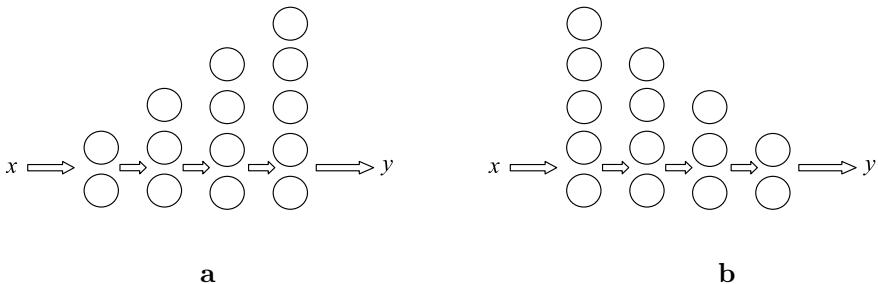


图13.11：两个对称网络，它们依赖相同数量的参数：a. 一个编码器。b. 一个解码器。参数数量为59；有14个偏差（每个神经元一个）和45个权重。

\mathcal{S} 的几何形状取决于网络输出 y 对权重和偏差的敏感度。因此，输出往往对位于靠近输入的第一层的权重不太敏感，而对位于靠近输出的最后一层的权重更敏感。

我们可以说神经网络与近似流形 S 相关联，并且训练网络意味着找到从目标点 \mathbf{z} 到流形 S 的最小距离。因此，流形 S 的几何性质，如形状、度量结构等，在神经网络的研究中将发挥重要作用。

度量结构 我们需要赋予 \mathcal{S} 一个黎曼度量，它将用于测量 \mathcal{S} 上点之间的距离以及 \mathcal{S} 的切向量之间的夹角。

由于 S 是维度为 r 的 \mathbb{R}^n 的子流形，自然地赋予 S 从 \mathbb{R}^n 的欧几里得结构诱导的自然度量。如果 θ_i 表示网络的参数（权重或偏差），那么 S 的基本切向量场由关于坐标 θ_i 的偏导数给出，如下所示

$$\xi_i = \frac{\partial \mathbf{y}}{\partial \theta_i} = \left(\frac{\partial y_1}{\partial \theta_i}, \dots, \frac{\partial y_n}{\partial \theta_i} \right), \quad 1 \leq i \leq r.$$

在 y 处 S 的切空间是由该点处所有基本切向量生成的线性空间。

$T_y\mathcal{S} = \text{span} \left\{ \frac{\partial y}{\partial \theta_i}; 1 \leq i \leq r \right\}$ 。如果 \mathcal{S} 是一个光滑流形，那么 $T_y\mathcal{S}$ 在每个点 $y \in \mathcal{S}$ 处具有恒定的维数 r 。这个条件可以等效地表述为最大秩条件

$$\text{rank}\left(\frac{\partial \mathbf{y}}{\partial \theta_i}\right)_i = r,$$

其表明基本切向量场是线性无关的。此正则性条件确保输出流形 \mathcal{S} 是光滑的（无角、尖点等）。在 $\mathbf{y}, v \in T_{\mathbf{y}}\mathcal{S}$ 处 \mathcal{S} 的一个切向量由线性组合定义

$$v = \sum_{i=1}^r v_i \frac{\partial \mathbf{y}}{\partial \theta_i},$$

其中 $v_i = v_i(\theta)$ 是 v 的分量。

\mathcal{S} 的自然度量结构由第一基本形式提供，其系数由

$$g_{ij}(\mathbf{y}) = \langle \xi_i, \xi_j \rangle = \left(\frac{\partial \mathbf{y}}{\partial \theta_i} \right)^T \frac{\partial \mathbf{y}}{\partial \theta_j} = \sum_{k=1}^n \frac{\partial y_k}{\partial \theta_i} \frac{\partial y_k}{\partial \theta_j}. \quad (13.5.10)$$

$r \times r$ 矩阵 $g = (g_{ij})$ 可用于计算切向量的长度、方向之间的夹角、 \mathcal{S} 上曲线的长度、点之间的距离、 \mathcal{S} 上区域的面积，并且一般来说，任何依赖于流形 \mathcal{S} 内在结构的数学概念。

我们回忆起，在微分几何中，内在和外在的概念通常分别用于指代源自 \mathcal{S} 局部和全局结构的几何信息。例如，测量 \mathcal{S} 上两条曲线之间的夹角可以使用局部信息来完成，也就是说，这可以由流形上的微观居民来执行，他不被允许离开流形。另一方面，网络的训练误差，即从外部目标点 \mathbf{z} 到 \mathcal{S} 的距离，是一个外在概念，因为它取决于流形居民在外部空间中飞越流形的能力，这使他能够进行测量。

向量的长度 考虑一个与流形 \mathcal{S} 相切的向量 $v = \sum_{i=1}^r v_i \frac{\partial \mathbf{y}}{\partial \theta_i}$ 。我们将用两种不同的方式来测量它的长度：外在地，作为 \mathbb{R}^n 中的一个向量；内地，作为 \mathcal{S} 的一个切向量。

如果 $\{e_k; 1 \leq k \leq n\}$ 表示 \mathbb{R}^n 中的自然正交基，那么 v 在 \mathbb{R}^n 中的 k 分量由下式给出

$$\langle v, e_k \rangle = v^T e_k = \sum_{i=1}^r v_i \frac{\partial y_k}{\partial \theta_i}, \quad 1 \leq k \leq n,$$

其中 $y_k = \langle \mathbf{y}, e_k \rangle$ 。那么 v 的欧几里得长度的平方由下式给出

$$\|v\|_{Eu}^2 = \sum_{k=1}^n \langle v, e_k \rangle^2.$$

相同的长度可以使用系数为 g_{ij} 的标量积以如下方式内在地计算：

$$\begin{aligned}\|v\|_g^2 &= \sum_{i,j} v_i v_j g_{ij} = \sum_{i,j} v_i v_j \sum_{k=1}^n \frac{\partial y_k}{\partial \theta_i} \frac{\partial y_k}{\partial \theta_j} \\ &= \sum_{k=1}^n \left(\sum_i v_i \frac{\partial y_k}{\partial \theta_i} \right) \left(\sum_j v_j \frac{\partial y_k}{\partial \theta_j} \right) = \sum_{k=1}^n \langle v, e_k \rangle^2 \\ &= \|v\|_{Eu}^2.\end{aligned}$$

我们在两种情况下都得到了相等的长度，这是预期的，因为 \mathcal{S} 的度量 g_{ij} 是从空间 \mathbb{R}^n 诱导而来的，并且长度与内在或外在方法无关。

曲线长度 假设神经网络的权重和偏差取决于一个额外的参数 s 。这可以是时间，或者是网络的某个超参数。那么

$\theta_i = \theta(s)$, $1 \leq i \leq r$ ，因此 $c(s) = \mathbf{y}(\theta(s))$ 在流形 \mathcal{S} 上表示一条曲线。因此，网络超参数的连续调整对应于流形上的一条曲线。如果 s 在 a 和 b 之间取值，那么曲线 $c(s)$ 的长度由积分内地定义

$$L(c) = \int_a^b \|\dot{c}(s)\| ds = \int_a^b \sum_{i,j} \dot{c}_i(s) \dot{c}_j(s) g_{ij}(c(s)) ds,$$

其中 $\dot{c}(s)$ 表示沿曲线的切向量。链式法则给出

$$\dot{c}(s) = \frac{d}{ds} c(s) = \frac{d}{ds} \mathbf{y}(\theta(s)) = \sum_i \frac{\partial \mathbf{y}}{\partial \theta_i} \frac{d\theta_i}{ds} = \langle \nabla_{\theta} \mathbf{y}, \dot{\theta}(s) \rangle. \quad (13.5.11)$$

测地线 有时，我们感兴趣的是在 \mathcal{S} 上找到两个给定的点之间最短长度的曲线。如果我们在 \mathbb{R}^n 上寻找两点之间的最短曲线，显然这是一条线段。然而，在流形 \mathcal{S} 的情况下，其特征更为复杂，最短距离的曲线是测地线。测地线的一个应用是找到给定初始点 $\mathbf{y}(\theta^0)$ 和最优点 $\mathbf{y}(\theta^*)$ 之间的最短曲线，最优点 $\mathbf{y}(\theta^*)$ 是目标点 \mathbf{z} 在 \mathcal{S} 上的正交投影。这条曲线对应于网络的最有效调优，因为参数调优对应于流形上的一条曲线。

值得注意的是，在 \mathcal{S} 的度量下测量的 \mathcal{S} 两点之间的距离至少与使用目标空间的度量测量的相同两点之间的距离一样大。

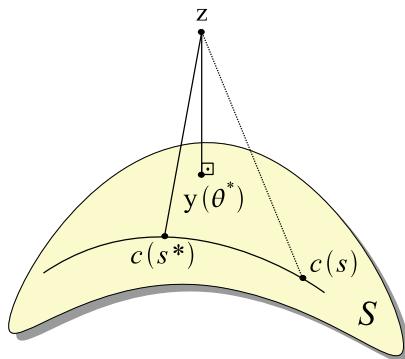


图13.12：最优参数 s^* 对应于曲线 $c(s)$ 上最接近目标 \mathbf{z} 的点。

在 S 是测地子流形的情况下，即其第二基本形式为零 $L = 0$ 时，这些距离相等。

最优参数值 假设在根据 s 修改参数 $\theta(s)$ 时，我们首先注意到精度有所提高，随后精度又下降。对于某个最优值 s^* ，误差达到最小。从几何角度来看，这对应于曲线 $c(s)$ 上最接近目标点 \mathbf{z} 的点，见图13.12。当向量 $\overrightarrow{\mathbf{z}c(s)}$ 与切线向量 $\dot{c}(s)$ 垂直时会出现这种情况，这一事实可以写成

$$(\mathbf{z} - c(s))^T \dot{c}(s) = 0.$$

若参数 s 被修改，使得速率 $\|\dot{c}(s)\|^2$ 为常数，² 然后求导并使用乘积法则，可得 $c(h)^T \dot{c}(h) = 0$ 。因此，上一个方程变为 $\mathbf{z}^T \dot{c}(h) = 0$ 。使用公式 (13.5.11) 意味着最优值 s^* 满足该方程

$$\langle \nabla_{\theta}(\mathbf{z}^T \mathbf{y}(s^*)), \dot{\theta}(s^*) \rangle = 0.$$

13.6 正则化

神经网络最理想的特性是泛化能力强。这意味着在使用训练集对网络进行优化之后，对于其他未见过的测试数据，网络仍应具有较高的准确率。

² 这也称为弧长参数，因为它与沿曲线 $c(s)$ 测量的弧长成比例。

为了实现这一目标，网络的构建应使其不会过度拟合训练数据。

如果我们考虑多项式回归的特殊情况，这种现象就更容易解释。考虑平面上的7个点， $(x_i, z_i), 1 \leq i \leq 7$ ，并使用三种类型的多项式模型进行回归。线性回归不是一个好模型，导致数据欠拟合，其特征是训练误差大，见图13.13 a。二次模型产生的训练误差相对较小，拟合效果良好。7次插值多项式产生过拟合，其特征是训练误差为零，测试误差大。我们选择合适回归模型的方法是在选择多项式次数时要简约。同时，多项式的次数应该足够大，以捕捉数据的主要趋势而不会过度拟合。

对于一般神经网络的情况，类似的观察结果也适用。在这种情况下，近似多项式被输出流形 \mathcal{S} 所取代，并且多项式的次数对应于 \mathcal{S} 的维度，即网络参数 θ_i 的数量。在这种情况下，简约原则转化为选择一个低维流形，即一个神经元数量较少的网络。

问题的设置如下：给定 N 个神经元和一个训练集 $\{(x_i, z_i); 1 \leq i \leq n\}$ ，构建一个从数据中学习且不过度拟合的数据神经网络。

接下来，我们将针对给定数量的神经元 N ，介绍一些正则化技术，即避免或减少数据过拟合的方法。

13.6.1 追求更小维度

由于 N 是给定的，我们需要确定隐藏层的数量以及每层中的神经元数量。我们已经看到，仅使用两个隐藏层会产生流形 \mathcal{S} 的最大维度，所以我们应该避免这种情况。我们要么采用浅层结构，只有一个隐藏层，要么采用深层结构，使用足够数量的隐藏层，使得 \mathcal{S} 的维度足够小，并且简约准则成立。

13.6.2 范数正则化

为了减少过拟合，应该使用较小的权重 w 。通过添加一个涉及权重范数的额外项来修改成本函数 $C(w, b)$ ，该项乘以一个正的拉格朗日乘数 λ ，它描述了对小权重的偏好（较大的 λ 对应较小的权重）。

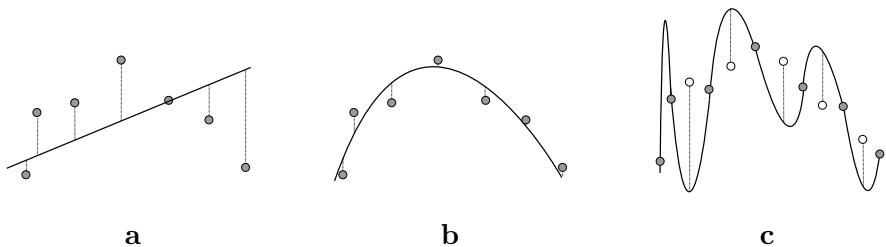


图13.13：通过7个点的多项式回归：(a) 使用直线会导致欠拟合；(b) 使用二次多项式会得到良好的拟合；(c) 使用七次多项式会使数据过拟合。

正则化成本函数变为

$$L(w) = C(w, b) + \lambda \|w\|^2,$$

其中 $\|\cdot\|$ 通常是 L^1 范数或 L^2 范数。这种正则化类型在3.11节中已更详细地讨论过。范数正则化的效果是在输出流形上寻找一个最优的点，该点位于某个邻域内。

13.6.3选择最平坦的流形

我们注意到，对于流形 S 的固定维度，有几种神经结构与之对应。例如，在图13.11给出的编码器和解码器结构中可以看到这一点。现在的问题是，从正则化的角度来看，哪种结构更好？

为了降低测试误差，我们将采用神经网络，其流形 S 尽可能平坦。这里的“平坦”是指在目标空间 \mathbb{R}^n 中弯曲尽可能小的流形。

平坦性是一个外在概念，可以通过第二基本形式用几何术语形式化。我们将从一些例子开始。

例13.6.1 考虑空间 \mathbb{R}^3 中的一个平面 \mathcal{P} 。它是平坦的，因为它不弯曲。这等同于观察到该平面的法向量是一个常向量场。曲面弯曲的方式由其法向量改变方向的速率来描述。这称为曲面的形状算子（或魏因加滕映射），它与不同类型的曲率有关，例如见[85]。

示例13.6.2 另一个示例涉及一条平面曲线， $c(s)$ ，其单位切向量为 $T(s)$ ，法向量为 $N(s)$ ，其中 s 表示弧长参数。法向量的变化率由 $N'(s) = -\kappa(s)T(s)$ 给出，其中 $\kappa(s)$ 是 $c(s)$ 的曲率，它描述了曲线的弯曲情况。

零曲率等同于法线变化率为零，这对应于一条直线曲线，即线段。

在所有相同维度的流形中，为了避免过拟合，我们需要选择尽可能平坦的那个。如第13.1.7节所述，第二基本形式 L 描述了流形 \mathcal{S} 在空间 \mathbb{R}^n 中是如何弯曲的。

我们在目标空间 $\mathcal{M} = \mathbb{R}^n$ 的特定情况下更明确地说明这一点。对于 \mathbb{R}^n 上的任意两个向量场 $U = \sum_k U^k e_k$ 和 $V = \sum_k V^k e_k$ ，定义 V 关于 U 的导数的求导算子 ∇ 为

$$\nabla_U V = (D_U V^1, \dots, D_U V^n),$$

其中 $D_U f$ 表示 f 关于 U 的方向导数。现在，如果考虑 U 和 V 是流形 \mathcal{S} 的切向量场，那么 $\nabla_U V$ 可以正交分解为

$$\nabla_U V = (\nabla_U V)^\parallel + (\nabla_U V)^\perp,$$

其中 $(\nabla_U V)^\parallel$ 是 $\nabla_U V$ 在 \mathcal{S} 的切空间上的投影，而 $(\nabla_U V)^\perp$ 表示 $\nabla_U V$ 的正交分量。法向分量

$$L(U, V) = (\nabla_U V)^\perp$$

表示 \mathcal{S} 关于 \mathbb{R}^n 的第二基本形式。映射 L 是对称且线性的，可以写成

$$L(U, V) = \sum_{\alpha, \beta=1}^r L_{\alpha\beta} U^\alpha V^\beta,$$

其中拉丁上标描述了相对于基本向量场 $\xi_\alpha = \frac{\partial \mathbf{y}(\theta)}{\partial \theta_\alpha}$ 的依赖性，其中 $U = \sum_{\alpha=1}^r U^\alpha \xi_\alpha$ 。第二基本形式的系数

$$L_{\alpha\beta} = L(\xi_\alpha, \xi_\beta)$$

是属于 \mathcal{S} 的法空间的向量值，其维数为 $n - k$ 。如果 $L_{\alpha\beta} = \sum_k L_{\alpha\beta}^k$ ，那么每个分量 $L_{\alpha\beta}^k$ 构成一个阶数为 r 的对称方阵。

消失的 L 形式一个消失的第二基本形式， $L = 0$ ，等同于其系数 $L_{\alpha\beta} = 0$ 的消失。在这种情况下， \mathcal{S} 被称为欧几里得空间 \mathbb{R}^n 的测地子流形。

等价的特征描述是， \mathcal{S} 中任何局部长度最小化曲线在 \mathbb{R}^n 中都是直线段。具有此属性的流形是 \mathbb{R}^n 的仿射子空间，见练习 13.8.6。在这种情况下，目标 \mathbf{z} 到 \mathcal{S} 的投影是唯一的。

L 形式的范数由于 L 是向量值的，为了正则化目的，我们将定义并使用 L 的范数。对于在 \mathbf{y} 处与 \mathcal{S} 相切的任何向量 U ，我们有 $L(U, U)$ 是 \mathbb{R}^n 中的向量，并且让 $\|L(U, U)\|_{Eu}$ 表示其欧几里得长度。我们通过以下方式定义 L 的范数

$$\|L\| = \max \left\{ \frac{\|L(U, U)\|_{Eu}}{\|U\|^2}; U \text{ tangent to } \mathcal{S} \right\}. \quad (13.6.12)$$

在此， $\|U\|$ 表示以 \mathbb{R}^n 为单位测量的 U 的长度，或者使用度量工具 \mathcal{S} 测量的长度。利用 L 的缩放特性（见命题 13.1.8 的第 (iii) 部分），这个范数可以等效地写为

$$\|L\| = \max_{\|U\|=1} \left\{ \|L(U, U)\|_{Eu}; U \text{ tangent to } \mathcal{S} \right\}.$$

这个范数与 L 的特征值的关系如下。由于对称性， $L_{\alpha\beta}^k = L_{\beta\alpha}^k$ ，所以矩阵 L^k 有实特征值。表达式

$$\max_{\|U\|=1} |L^k(U, U)| = |\lambda_k|,$$

给出了 L^k 的最大特征值的绝对值（见附录 G）。因此，

$$\|L\| = (\lambda_1^2 + \dots + \lambda_n^2)^{1/2}.$$

从几何角度来看，每个 λ_k 表示子流形 \mathcal{S} 在某个法向方向上的曲率，因此， $\|L\|$ 表示 \mathcal{S} 曲率的一种外在度量。通过保持 $\|L\|$ 较小，所有曲率都保持较小，由此我们可以控制 \mathcal{S} 在 \mathbb{R}^n 内部的弯曲程度。

新的成本函数是使用先前范数对距离平方进行正则化

$$C(w, b; \mu) = \frac{1}{2} \|\mathbf{y}(w, b) - \mathbf{z}\|^2 + \mu \|L\|, \quad (13.6.13)$$

其中超参数 μ 是一个拉格朗日乘数。因此，正则化过程是在两种效应之间进行权衡得到的：训练误差（从目标 \mathbf{z} 到 \mathcal{S} 的距离）的最小化和流形平坦度的最大化，见图 13.14。超参数 μ 捕捉这种权衡效应： μ 的值越大，流形越平坦，而 μ 的值越小，流形越靠近目标 \mathbf{z} 。

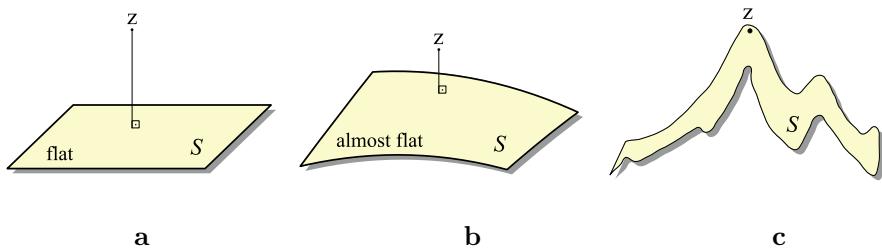


图13.14：使用具有相同维度但平坦度不同的流形 S 进行正则化：(a) 使用平面会导致与目标 z 的距离较大，这会使数据欠拟合；(b) 在曲率和到 z 的距离之间进行权衡会导致良好的拟合；(c) 使用高度弯曲的流形，我们总是可以迫使目标点 z 属于该流形，这种情况对应于过拟合。

示例13.6.1 (多项式回归) 本示例介绍了在输出流形概念的背景下多项式回归的情况。 r 次多项式

$$\psi(x; \theta) = x^r + \theta_1 x^{r-1} + \theta_2 x^{r-2} + \cdots + \theta_{r-1} x + \theta_r \quad (13.6.14)$$

用于在最小二乘意义上逼近坐标由下式给出的 n 个点

$$\mathcal{T} = \{(x_1, z_1), (x_2, z_2), \dots, (x_n, z_n)\}.$$

使用训练集 \mathcal{T} ，应调整多项式系数 θ_i ，以使平方误差之和最小化。我们假设值 x_i 是不同的，并且 $r < n$ 。我们将考虑 \mathbb{R}^n 中的向量 $\mathbf{x}^T = (x_1, x_2, \dots, x_n)$ 和 $\mathbf{z}^T = (z_1, z_2, \dots, z_n)$ 。输入向量 \mathbf{x} 和参数向量 $\theta = (\theta_1, \dots, \theta_r) \in \mathbb{R}^r$ 用于构建由 $\theta \rightarrow \mathbf{y}(\mathbf{x}; \theta) \in \mathbb{R}^n$ 参数化的流形 S 如下

$$\mathbf{y}(\mathbf{x}; \theta) = (\psi(x_1; \theta), \dots, \psi(x_n; \theta)).$$

给定多项式关系(13.6.14)，与 S 相切的向量场具有 $\xi_j = \sum_{k=1}^n x_k^{r-j} e_k$ 的形式。更具体地说，

$$\begin{aligned}\xi_1 &= \frac{\partial \mathbf{y}}{\partial \theta_1} = (x_1^{r-1}, x_2^{r-1}, \dots, x_n^{r-1}) \\ \xi_2 &= \frac{\partial \mathbf{y}}{\partial \theta_2} = (x_1^{r-2}, x_2^{r-2}, \dots, x_n^{r-2}) \\ &\dots \dots \dots \\ \xi_{r-1} &= \frac{\partial \mathbf{y}}{\partial \theta_{r-1}} = (x_1, x_2, \dots, x_n) \\ \xi_r &= \frac{\partial \mathbf{y}}{\partial \theta_r} = (1, 1, \dots, 1).\end{aligned}$$

由于 x_i 具有不同的值，以下范德蒙德行列式不为零

$$\det \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_r \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{r-1} & x_2^{r-1} & \cdots & x_n^{r-1} \end{pmatrix} = \prod_{i < j} (x_j - x_i) \neq 0,$$

因此，满足最大秩条件，即 $\text{rank} \left(\frac{\partial \mathbf{y}}{\partial \theta} \right) = r$ 。因此，向量场 $\{\xi_1, \dots, \xi_r\}$ 在 \mathcal{S} 的每一点 $\mathbf{y} \in \mathcal{S}$ 处张成 \mathcal{S} 的切空间 $T_{\mathbf{y}}\mathcal{S}$ 。 \mathcal{S} 的内蕴几何由具有分量的度量张量 g 描述

$$g_{ij} = \langle \xi_i, \xi_j \rangle = \sum_{k=1}^n x_k^{r-i} x_k^{r-j} = \sum_{k=1}^n x_k^{2r-(i+j)}, \quad 1 \leq i, j \leq r.$$

需要注意的是，矩阵系数 g_{ij} （进而 \mathcal{S} 的内蕴几何）不依赖于参数 θ_k 。

（对于具有自然度量 δ_{ij} 的欧几里得空间 \mathbb{R}^n ，也会出现类似情况）。这对应于一个内蕴平坦的子流形 \mathcal{S} 。由于 $\partial g_{ij}/\partial \theta_k = 0$ ，所以克里斯托费尔符号为零， $\Gamma_{ij}^k = 0$ 。这意味着子流形 \mathcal{S} 的黎曼曲率³为零。特别地，其列维-奇维塔联络在基 $\{\xi_i\}$ 上为零

$$\bar{\nabla}_{\xi_i} \xi_j = \sum_k \Gamma_{ij}^k \theta_k = 0.$$

因此，第二基本形式仅能用 \mathbb{R}^n 的列维-奇维塔联络表示为

$$L(\xi_i, \xi_j) = \nabla_{\xi_i} \xi_j - \underbrace{\bar{\nabla}_{\xi_i} \xi_j}_{=0} = \nabla_{\xi_i} \xi_j = \sum_{k=1}^n (D_{\xi_i} \xi_j^k) e_k = 0,$$

因为 $D_{\xi_i} \xi_j^k = \langle \xi_i, \text{grad } \xi_j^k \rangle = 0$ ，作为分量 $\xi_j^k = x_k^{r-j}$ 是常数（取决于固定的输入项 x_k ）。因此， $L = 0$ ，进而子流形 \mathcal{S} 也是外在平坦的。代价函数

$$C(\theta; \mu) = \frac{1}{2} \|\mathbf{y}(\mathbf{x}, \theta) - \mathbf{z}\|^2$$

³ 流形的黎曼曲率由张量描述

$$R_{ijk}^r = \partial_{\theta_i} \Gamma_{jk}^r - \partial_{\theta_j} \Gamma_{ik}^r + \Gamma_{ih}^r \Gamma_{jk}^h - \Gamma_{jh}^r \Gamma_{ik}^h,$$

对重复指标求和。

在这种情况下不需要正则化项 $\mu \| L \|$ 。

事实上，输出流形 S 是 \mathbb{R}^n 中维度为 r 的仿射超平面。通过将点 \mathbf{z} 投影到该超平面上可得到最优解。此投影可使用摩尔 - 彭罗斯伪逆显式计算，见附录中的G.2节。我们将条件约束 $\psi(x_j; \theta) = z_j, 1 \leq j \leq n$ 写成一个超定线性系统

$$x_j^r + \theta_1 x_j^{r-1} + \cdots + \theta_{r-1} x_j + \theta_r = z_j, \quad 1 \leq j \leq n.$$

这可以写成矩阵形式为

$$A\theta = \beta,$$

其中 $\theta^T = (\theta_1, \dots, \theta_r)$, $\beta^T = (z_1 - x_1^r, \dots, z_n - x_n^r)$ 和

$$A = \begin{pmatrix} x_1^{r-1} & \cdots & x_1 & 1 \\ \cdots & \cdots & \cdots & \cdots \\ x_n^{r-1} & \cdots & x_n & 1 \end{pmatrix}$$

最优参数 θ^* 可通过应用伪逆得到。

$$\theta^* = A^+ \beta,$$

其中 A^+ 由公式(G.2.6)给出。

\mathbf{z} 在超平面 S 上的投影由 $\mathbf{y}^* = \mathbf{y}(\mathbf{x}; \theta^*)$ 给出。

接下来我们将讨论仅与一个神经元相关联的输出流形。即使在这种简单情况下，计算也很复杂，而在一般神经网络的情况下，它并不总是能够显式地进行。

例13.6.2 我们考虑具有输入 $x \in \mathbb{R}$ 、输出 $y = \sigma(wx + b) \in \mathbb{R}$ 和实参数 w, b 的单个Sigmoid神经元的情况，见图13.15。我们取激活函数 σ 为逻辑函数。如果训练集由

$$(\mathbf{x}, \mathbf{z}) = \{(x_1, z_1), (x_2, z_2), \dots, (x_n, z_n)\},$$

那么与前一个神经元相关联的流形 S 由映射 $\psi: \mathbb{R}^2 \rightarrow \mathbb{R}^n$ 定义

$$\psi(w, b) = \mathbf{y}(w, b) = (y_1, \dots, y_n) = \sigma(wx_1 + b), \dots, \sigma(wx_n + b).$$

这表示包含在空间 \mathbb{R}^n 中的二维曲面，赋予了由 \mathbb{R}^n 的欧几里得结构诱导的黎曼度量。 S 的基本切向量场由偏导数给出如下

$$\begin{aligned} \xi_1 &= \frac{\partial \mathbf{y}(w, b)}{\partial w} = \sigma'(wx + b) \odot \mathbf{x} = \mathbf{y} \odot (1 - \mathbf{y}) \odot \mathbf{x} \\ \xi_2 &= \frac{\partial \mathbf{y}(w, b)}{\partial b} = \sigma'(wx + b) = \mathbf{y} \odot (1 - \mathbf{y}), \end{aligned}$$

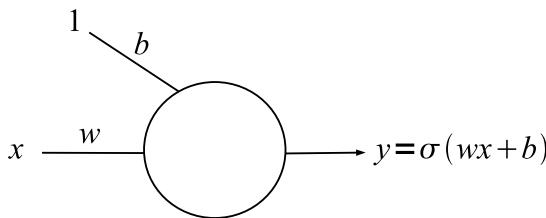


图13.15：与Sigmoid神经元相关联的流形 \mathcal{S} 是二维的。

其中我们使用了 $\sigma' = \sigma(1 - \sigma)$ 且 \odot 表示哈达玛积。 $\{\xi_1, \xi_2\}$ 的线性独立性由条件 $\text{rank}(\xi_1, \xi_2) = 2$ 保证，该条件显然满足。输出流形 \mathcal{S} 的内在几何由以下度量系数描述

$$\begin{aligned} g_{11} &= \langle \xi_1, \xi_1 \rangle = \langle \mathbf{y} \odot (1 - \mathbf{y}) \odot \mathbf{x}, \mathbf{y} \odot (1 - \mathbf{y}) \odot \mathbf{x} \rangle \\ &= \langle \mathbf{y} \odot \mathbf{y}, \mathbf{x} \odot \mathbf{x} \rangle - 2\langle \mathbf{y} \odot \mathbf{y} \odot \mathbf{y}, \mathbf{x} \odot \mathbf{x} \rangle + \langle \mathbf{y} \odot \mathbf{y} \odot \mathbf{x}, \mathbf{y} \odot \mathbf{y} \odot \mathbf{x} \rangle \\ g_{12} &= \langle \xi_1, \xi_2 \rangle = \langle \mathbf{y} \odot (1 - \mathbf{y}) \odot \mathbf{x}, \mathbf{y} \odot (1 - \mathbf{y}) \rangle \\ &= \langle \mathbf{y} \odot \mathbf{y}, \mathbf{x} \rangle - 2\langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \odot \mathbf{x} \rangle + \langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \odot \mathbf{y} \odot \mathbf{x} \rangle \\ g_{22} &= \langle \xi_2, \xi_2 \rangle = \langle \mathbf{y} \odot (1 - \mathbf{y}), \mathbf{y} \odot (1 - \mathbf{y}) \rangle = \langle \mathbf{y} - \mathbf{y} \odot \mathbf{y}, \mathbf{y} - \mathbf{y} \odot \mathbf{y} \rangle \\ &= \langle \mathbf{y}, \mathbf{y} \rangle - 2\langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \rangle + \langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \odot \mathbf{y} \rangle \\ g_{12} &= g_{21}. \end{aligned}$$

所有先前的公式都可以用幂和来表示，例如，

$$\langle \mathbf{y} \odot \mathbf{y}, \mathbf{x} \odot \mathbf{x} \rangle = \sum_k y_k^2 x_k^2, \quad \langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \rangle = \sum_k y_k^3, \quad \langle \mathbf{y} \odot \mathbf{y}, \mathbf{y} \odot \mathbf{y} \rangle = \sum_k y_k^4,$$

其中 $y_k = \sigma(wx_k + b)$ 。

对系数 g_{ij} 求导，可以潜在地计算第二类克里斯托费尔符号 (13.1.2)，然后使用切向量基 $\bar{\nabla}_{\xi_i} \xi_k = \sum_j \Gamma_{ij}^k(\mathbf{y}) \xi_j$ 在 \mathcal{S} 上得到列维-奇维塔联络。由于计算很繁琐，我们将采用不同的方法。

使用符号 $(\theta_1, \theta_2) = (w, b)$ ，目标空间 \mathbb{R}^n 上的列维-奇维塔联络变为 $\nabla_{\xi_i} \xi_j = \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}$ 。那么高斯分解(13.1.5)可写为

$$\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} = \left(\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} \right)^{\parallel} + \left(\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} \right)^{\perp}.$$

第二基本形式由法向部分给出

$$L_{ij} = \left(\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} \right)^\perp = \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} - \left(\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} \right)^\parallel. \quad (13.6.15)$$

右侧的两项都是可计算的。我们将从 $\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}$ 开始。利用关系 $\sigma''(x) = \sigma(x)(1 - \sigma(x))(2 - \sigma(x))$, 我们得到

$$\begin{aligned} \frac{\partial^2 \psi}{\partial \theta_1 \partial \theta_1} &= \frac{\partial^2 \psi}{\partial^2 w} = \sigma''(w\mathbf{x} + b) \odot x \odot x \\ &= \mathbf{y} \odot (1 - \mathbf{y}) \odot (2 - \mathbf{y}) \odot x \odot x; \\ \frac{\partial^2 \psi}{\partial \theta_2 \partial \theta_2} &= \frac{\partial^2 \psi}{\partial^2 b} = \sigma''(w\mathbf{x} + b) \\ &= \mathbf{y} \odot (1 - \mathbf{y}) \odot (2 - \mathbf{y}); \\ \frac{\partial^2 \psi}{\partial \theta_1 \partial \theta_2} &= \frac{\partial^2 \psi}{\partial w \partial b} = \sigma''(w\mathbf{x} + b) \odot x \\ &= \mathbf{y} \odot (1 - \mathbf{y}) \odot (2 - \mathbf{y}) \odot x; \\ \frac{\partial^2 \psi}{\partial \theta_2 \partial \theta_1} &= \frac{\partial^2 \psi}{\partial \theta_1 \partial \theta_2}. \end{aligned}$$

切向分量是基本向量场的线性组合

$$\left(\frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j} \right)^\parallel = \alpha_{ij}^1 \xi_1 + \alpha_{ij}^2 \xi_2,$$

其中系数 α_{ij}^k 可以明确地作为线性系统的解找到

$$\begin{aligned} g_{11}\alpha_{ij}^1 + g_{12}\alpha_{ij}^2 &= \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_1 \rangle \\ g_{12}\alpha_{ij}^1 + g_{22}\alpha_{ij}^2 &= \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_2 \rangle. \end{aligned}$$

这是

$$\alpha_{ij}^1 = g^{11} \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_1 \rangle + g^{12} \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_2 \rangle$$

$$\alpha_{ij}^2 = g^{21} \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_1 \rangle + g^{22} \langle \frac{\partial^2 \psi}{\partial \theta_i \partial \theta_j}, \xi_2 \rangle$$

其中 (g^{ij}) 是逆矩阵 $(g_{ij})^{-1}$ 。标量积项可以很容易地计算出来。

例如：

$$\begin{aligned}\left\langle \frac{\partial^2 \psi}{\partial \theta_1 \partial \theta_1}, \xi_1 \right\rangle &= \langle \mathbf{y} \odot (1 - \mathbf{y}) \odot (2 - \mathbf{y}) \odot \mathbf{x} \odot \mathbf{x}, \mathbf{y} \odot (1 - \mathbf{y}) \odot \mathbf{x} \rangle \\ &= \sum_{k=1}^n y_k^2 (1 - y_k)^2 (2 - y_k)^2 x_k^3.\end{aligned}$$

L_{ij} 系数现在可以使用(13.6.15)来计算。例如，

$$\begin{aligned}L_{11} &= \frac{\partial^2 \psi}{\partial w^2} - \left(\frac{\partial^2 \psi}{\partial w^2} \right)^{\parallel} \\ &= \mathbf{y} \odot (1 - \mathbf{y}) \odot (2 - \mathbf{y}) \odot \mathbf{x} \odot \mathbf{x} - \alpha_{11}^1 \xi_1 - \alpha_{11}^2 \xi_2 \\ &= \mathbf{y} \odot (1 - \mathbf{y}) \odot [(2 - \mathbf{y}) \odot \mathbf{x} \odot \mathbf{x} - \alpha_{11}^1 \mathbf{x} - \alpha_{11}^2].\end{aligned}$$

考虑范数

$$\|L\| = \sup_{\|U\|=1} \|L(U, U)\|,$$

它衡量 \mathcal{S} 的平坦度。在这种情况下，相关的变分问题是最小化正则化成本函数

$$C(w, b; \mu) = \frac{1}{2} \|\mathbf{y}(w, b) - \mathbf{z}\|^2 + \mu \|L\| = \frac{1}{2} \sum_{k=1}^n (\sigma(wx_k + b) - z_k)^2 + \mu \|L\|.$$

13.6.4 模型平均

一种降低测试误差的可靠技术是对具有相同输入的 N 个不同神经网络的输出进行平均，这些网络是分别训练的。每个特定网络都会产生误差 ϵ_i 。我们假设这些误差是均值为零、方差为 v 的独立随机变量，具有相同的分布。根据中心极限定理（附录中的定理D.6.4），对于足够大的 N ，平均误差 $\epsilon_{\text{ave}} = \frac{1}{N} \sum_{i=1}^N \epsilon_i$ 趋于正态分布，均值为零，方差为 v/N 。这意味着平均方法比其每个成员的性能都要好。

这个想法也可以在输出流形的背景下形式化。主要思想是将目标 \mathbf{z} 投影到与一些神经网络相关联的几个输出流形上，然后将投影的平均值视为目标 \mathbf{z} 的近似值。为了简单起见，我们仅针对两个网络的情况讨论此技术。

考虑两个深度相同的前馈神经网络，它们具有相同的输入 $\mathbf{x} \in \mathbb{R}^k$ ，并学习相同的目标 $\mathbf{z} \in \mathbb{R}^n$ 。设 $\mathbf{y} = \mathbf{y}(w, b)$ 和 $\tilde{\mathbf{y}} = \tilde{\mathbf{y}}(\tilde{w}, \tilde{b})$ 为这两个网络的输出，见图13.16。

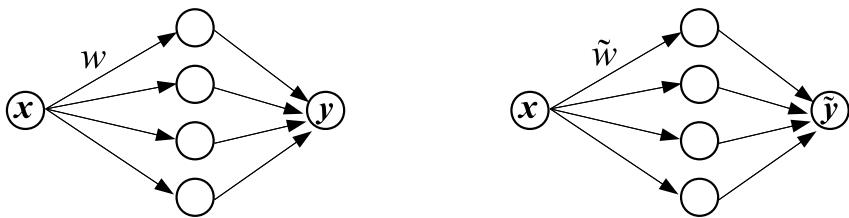


图13.16：两个具有相同输入 x 和输出 $y(w, b)$ 、 $\tilde{y}(\tilde{w}, \tilde{b})$ 的神经网络，学习相同的目标 z 。

它们被视为分别属于输出流形的点， $y \in \mathcal{S}$ 和 $\tilde{y} \in \tilde{\mathcal{S}}$ 。训练后，权重和偏差被设置为

$$(w, b) = \arg \min \|z - y\|^2, \quad (\tilde{w}, \tilde{b}) = \arg \min \|z - \tilde{y}\|^2.$$

这意味着 y 和 \tilde{y} 分别是目标 z 在流形 \mathcal{S} 和 $\tilde{\mathcal{S}}$ 上的正交投影。输出平均值 $\frac{1}{2}(y + \tilde{y})$ 是目标 z 的一个近似值，有望比每个 y 和 \tilde{y} 更好。

然而，我们可以通过采用凸组合来做得更好。在线段 $\{\lambda y + (1 - \lambda)\tilde{y}; \lambda \in [0, 1]\}$ 上存在一些点，它们比 y 和 \tilde{y} 都更靠近 z ，见图13.17。最近的点是 z 在这条线上的投影。这对应于目标的一个更好的近似器，它可以作为仅一个网络的输出获得。这是将前两个神经网络组合成具有以下属性的单个网络的模型组合，见图13.18：

- (i) 输入是 x ；
- (ii) 它的深度比给定网络多一个单元；
- (iii) 它的第 ℓ 层是给定网络第 ℓ 层的并集；
- (iv) 最后一层仅包含一个线性神经元；
- (v) 其参数由 $\{w, b, \tilde{w}, \tilde{b}, \lambda, 1 - \lambda\}$ 给出；
- (vi) 其结果为 $\lambda y + (1 - \lambda)\tilde{y}$ 。

模型组合可应用于任意数量的神经网络。所得网络的输出由各个网络结果的凸组合给出，系数的选择应使目标 z 与由结果确定的仿射空间之间的距离最小。

13.6.5 随机失活

随机失活是一种减少过拟合的有效方法，对一大类神经网络都很有效。

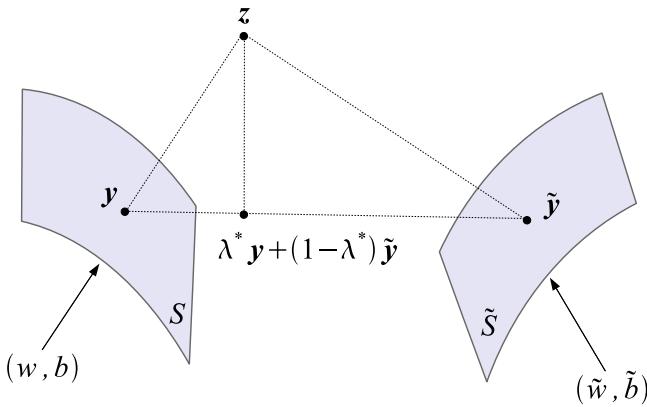


图13.17： \mathbf{z} 在线段 $\mathbf{y}\tilde{\mathbf{y}}$ 上的正交投影比 \mathbf{y} 和 $\tilde{\mathbf{y}}$ 都是更好的近似。这由 $\lambda^*\mathbf{y} + (1 - \lambda^*)\tilde{\mathbf{y}}$ 给出，其中 λ^* 是通过 $\lambda^* = \arg \min \| \mathbf{z} - \lambda\mathbf{y} - (1 - \lambda)\tilde{\mathbf{y}} \|$ 获得的。

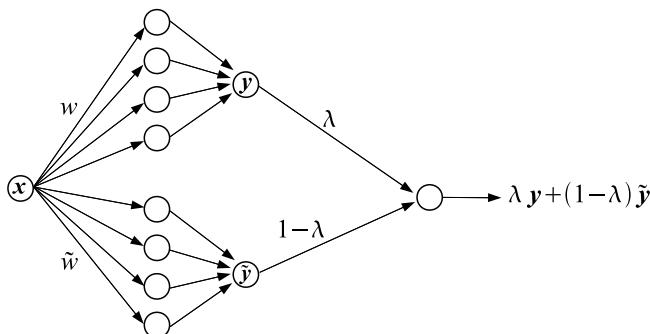


图13.18：两个网络的模型组合是一个比其两个部分都能产生更好学习效果的网络。

主要思想是从网络中临时丢弃或移除神经元（隐藏层、输入层，但不包括输出层）。移除哪些神经元是随机选择的。这种方法的成功之处在于打破了标准反向传播算法在神经元之间形成的共适应⁴。随机失活训练每个神经元能够在没有其他神经元帮助的情况下发挥作用。因此，得到的网络将能很好地泛化到新的未见数据上，从而产生较小的测试误差。

有时，通过与一家采用某种培训策略的公司进行比较，可以更形象地解释随机失活：该公司规定一定比例的员工放假一天，而其余员工则接受培训以完成那些休假员工的工作。被选中放假的员工是随机挑选的，即使这个比例保持不变。在这个培训期结束时，每个员工都了解其他员工的工作，因此，当公司面对一项新的、未见的任务时，员工们能够更高效地完成工作。

随机失活技术在以下意义上也类似于 L^2 正则化。由于随机失活的理念是训练神经元尽可能独立地发挥作用，这意味着网络对任何特定特征都不敏感。由于特征存储在权重中，因此权重系统必须充分收缩，这一事实与小范数的情况类似。

从网络中随机丢弃神经元，包括它们的输入和输出连接，等同于对一个子网络进行采样，该子网络与一个输出子流形相关联。训练这个子网络等同于找到目标 \mathbf{z} 在相关输出子流形上的投影。对几个子网络应用这个过程会通过在相关子流形上的投影产生目标的估计值。它们的平均值被用作目标的估计器。

删除一个隐藏神经元 考虑前馈神经网络第 ℓ 层中的一个神经元，其 $\ell \notin \{0, L\}$ ，即该神经元属于隐藏层。通过从网络中删除这个神经元，我们理解为将该神经元及其所有权重（与该神经元相连的权重），包括其偏差一起移除。这将导致一个与前一个神经网络具有相同输入的新神经网络。

新网络参数空间的维度比原网络参数空间的维度少 $d^{(\ell-1)} + d^{(\ell+1)} + 1$

◦

⁴ 这很容易理解，例如，如果你尝试倒着背诵字母表。大脑在从 A 到 Z 按时间顺序学习字母表时会建立协同适应。当尝试倒着背诵字母表时所面临的困难表明在学习过程中神经元之间形成了某些协同适应。

这是因为我们移除了 $d^{(\ell-1)}$ 个输入权重、 $d^{(\ell+1)}$ 个输出权重和 1 个偏差。与往常一样， $d^{(\ell)}$ 表示第 ℓ 层中的神经元数量。因此，通过舍弃一个神经元，网络输出所依赖的参数减少，这会降低网络容量并减少任何最终的过拟合。

训练后，网络输出成为目标 \mathbf{z} 在较小维度输出流形上的投影。目前尚不清楚这个投影是否比神经元随机失活之前应用的前一种近似更接近 \mathbf{z} 。同样不明显的是哪种神经元随机失活会产生最佳近似。

丢弃多个神经元 随机失活技术会从每一层中随机移除一定比例的神经元，然后对得到的输出取平均值，见图13.19。然而，丢弃过多的神经元会使参数空间的维度降低过多，因此会导致欠拟合。

当丢弃一定数量的神经元时，得到的相关输出流形是与初始网络相关的输出流形的子流形。这个子流形⁵ 的余维数由下式给出

$$k = \sum_{\ell=1}^{L-1} n^{(\ell)}(d^{(\ell-1)} + d^{(\ell+1)} + 1),$$

其中 $n^{(\ell)}$ 是从第 ℓ 层丢弃的神经元数量。如果从每一层丢弃相同的百分比 q ，那么 $n^{(\ell)} = qd^{(\ell)}/100$ 。

每次随机失活后，经过训练的网络输出表示目标 \mathbf{z} 在相关输出子流形上的投影。这些投影中的每一个都是目标 \mathbf{z} 的近似值。通过类似于蒙特卡罗方法的方式，所有这些投影的平均值表示 \mathbf{z} 的近似值，它减少了过拟合，并且比任何一个 \mathbf{y}_j 更不易产生偏差。

示例13.6.3 我们将使用一个具有一个隐藏层、一维输入和输出的神经网络来举例说明该方法，见图13.19。我们考虑隐藏层中的 N 个神经元，并每次均匀地随机失活一个神经元，得到以下输出

$$\mathbf{y}_j = \mathbf{y} - \lambda_j \sigma(w_j x + b_j), \quad j = 1, \dots, N,$$

⁵ 流形 \mathcal{M} 的子流形 \mathcal{S} 的余维数是它们维度的差，即 $k = \dim \mathcal{M} - \dim \mathcal{S}$ 。

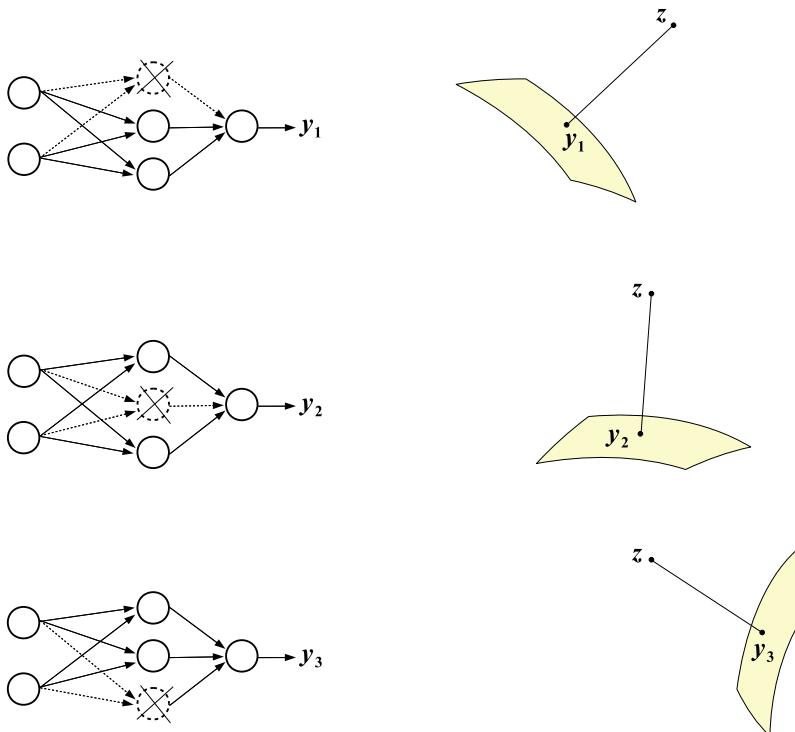


图13.19：当一次去掉一个神经元时，网络输出会产生目标 \mathbf{z} 在相关输出流形上的投影。投影的平均值 $\frac{1}{3}(\mathbf{y}_1 + \mathbf{y}_2 + \mathbf{y}_3)$ 被认为比任何一个 \mathbf{y}_j 都更能近似 \mathbf{z} 。

其中 $\mathbf{y} = \sum_{j=1}^N \lambda_j \sigma(w_j x + b_j)$ 表示初始网络输出。

由于每个 \mathbf{y}_j 是以概率 $q_i = q = \frac{1}{N}$ 选取的，那么预期网络输出由输出的平均值给出，如下所示

$$\sum_{j=1}^N q_j \mathbf{y}_j = \frac{1}{N} \sum_{j=1}^N \mathbf{y}_j = \left(1 - \frac{1}{N}\right) \mathbf{y} = (1-q) \mathbf{y},$$

其与初始网络的输出成正比， \mathbf{y} 。

乘性噪声 随机失活也可以看作是向网络中添加乘性噪声。由于每个神经元以概率 p 被保留，这意味着神经元的输出以概率 p 保持不变，并以概率 $q = 1 - p$ 消失（即，它以概率 q 被丢弃）。这等同于与一个伯努利随机变量相乘。如果在随机失活之前第 ℓ 层的第 i 个输出是 $x_i^{(\ell)}$ ，那么在随机失活之后它变为 $\tilde{x}_i^{(\ell)} = R_i^{(\ell)} x_i^{(\ell)}$ ，其中 $R_i^{(\ell)} \sim \text{Bernoulli}(p)$ ，伯努利随机变量的定义可以在附录 D.2 节中找到。因此，主方程(6.2.24)所描述的前馈操作

$$x_j^{(\ell)} = \phi\left(\sum_{i=1}^{d^{(\ell-1)}} w_{ij}^{(\ell)} x_i^{(\ell-1)} - b_j^{(\ell)}\right), \quad 1 \leq j \leq d^{(\ell)}$$

在辍学的情况下变为

$$x_j^{(\ell)} = \phi\left(\sum_{i=1}^{d^{(\ell-1)}} w_{ij}^{(\ell)} \tilde{x}_i^{(\ell-1)} - b_j^{(\ell)}\right), \quad 1 \leq j \leq d^{(\ell)},$$

其中 $\tilde{x}_i^{(\ell-1)} = R_i^{(\ell-1)} x_i^{(\ell-1)}$ 。以等效矩阵形式，方程(6.2.29)

$$X^{(\ell)} = \phi\left(W^{(\ell)^T} X^{(\ell-1)} - B^{(\ell)}\right)$$

变为

$$X^{(\ell)} = \phi\left(W^{(\ell)^T} R^{(\ell-1)} \odot X^{(\ell-1)} - B^{(\ell)}\right)$$

其中 $R^{(\ell-1)} = (R_i^{(\ell-1)})$ 是独立伯努利随机变量的向量， \odot 表示向量的哈达玛积。

注记13.6.4 经验证据表明，隐藏层的最优保留率通常为 $p = 0.5$ ，而输入层的约为 $p = 0.8$ 。

下一节建立了随机失活与 L_2 正则化之间的关系。

带随机失活的线性回归 本节讨论随机失活在经典线性回归问题中的应用。考虑输入向量 $X \in \mathbb{R}^n$ 和目标 $\mathbf{z} \in \mathbb{R}$ 。我们需要学习权重向量 $\mathbf{w} \in \mathbb{R}^n$ ，使得 $\|\mathbf{z} - X\mathbf{w}\|^2$ 最小化。应用随机失活后，新的目标函数变为

$$f(\mathbf{w}) = \mathbb{E}[\|\mathbf{z} - R \odot X\mathbf{w}\|^2],$$

其中 $R^T = (R_1, \dots, R_n)$ 是一个由独立伯努利随机变量组成的向量， $R_i \sim \text{Bernoulli}(p)$ 。利用 $\|a - b\|^2 = \|a\|^2 - 2a^T b + \|b\|^2$ 以及期望算子的线性性质，目标函数变为

$$\begin{aligned} f(\mathbf{w}) &= \|\mathbf{z}\|^2 - 2\mathbf{z}^T \mathbb{E}[R] \odot X\mathbf{w} + \mathbb{E}[\|R \odot X\mathbf{w}\|^2] \\ &= \|\mathbf{z}\|^2 - 2p\mathbf{z}^T X\mathbf{w} + p^2\|X\mathbf{w}\|^2 + \mathbb{E}[\|R \odot X\mathbf{w}\|^2] - p^2\|X\mathbf{w}\|^2 \\ &= \|\mathbf{z} - pX\mathbf{w}\|^2 + \mathbb{E}[\|R \odot X\mathbf{w}\|^2] - p^2\|X\mathbf{w}\|^2 \\ &= \|\mathbf{z} - pX\mathbf{w}\|^2 + \mathbb{E}[\mathbf{w}^T (R \odot X)^T (R \odot X)\mathbf{w}] - p^2\|X\mathbf{w}\|^2 \\ &= \|\mathbf{z} - pX\mathbf{w}\|^2 + \mathbb{E}[R^2]\mathbf{w}^T X^T X\mathbf{w} - p^2\|X\mathbf{w}\|^2 \\ &= \|\mathbf{z} - pX\mathbf{w}\|^2 + p\mathbf{w}^T X^T X\mathbf{w} - p^2\|X\mathbf{w}\|^2 \\ &= \|\mathbf{z} - pX\mathbf{w}\|^2 + p(1-p)\|X\mathbf{w}\|^2, \end{aligned}$$

我们在此处加上并减去了项 $p^2\|X\mathbf{w}\|^2$ 以形成范数的平方，并利用了伯努利变量的二阶矩为 p 这一性质。

将因子 p 吸收到权重 \mathbf{w} 中，目标函数变为

$$f(w) = \|\mathbf{z} - Xw\|^2 + \lambda\|Xw\|^2,$$

这是一个带有拉格朗日乘数 $\lambda = \frac{1-p}{p}$ 和 $w = p\mathbf{w}$ 的 L^2 正则化问题。当 p 趋于1时，所有神经元都被保留，而 λ 会变小。常数 λ 表示在随机失活过程中未保留和保留神经元之间的比例。因此，带有随机失活的线性回归等同于一个 L^2 正则化问题。

高斯噪声 为减少过拟合而向神经网络引入噪声的想法也适用于其他类型的噪声。Srivastava等人[116]描述了一种向每个神经元添加与其激活成正比的高斯噪声的方法。这意味着隐藏神经元的输出 $X^{(\ell)}$ 会受到与激活成正比的高斯噪声的干扰，即变为 $X^{(\ell)} + X^{(\ell)}G$ ，其中 $G \sim \mathcal{N}(0, 1)$ 。这种干扰也可以等效地以乘法方式写为 $X^{(\ell)}G'$ ，其中 $G' \sim \mathcal{N}(1, 1)$ 。值得注意的是，这种新型的随机失活至少与涉及伯努利随机变量的常规随机失活效果一样好。

总之，从神经网络的每个隐藏层和输入层中移除神经元会大幅降低相关输出流形的维度，导致网络容量下降，从而减少任何过拟合效应。读者可在论文[116]中找到更多细节。

通过插入噪声进行正则化 防止神经网络过拟合训练数据的一种方法是在训练期间向网络中插入噪声，然后在测试期间对噪声进行平均。例如，噪声可以是乘性的或加性的。在乘性噪声的情况下，我们将每一层的输出乘以一个随机变量 R （伯努利分布、均匀分布、高斯分布等）。现在，网络输出取决于输入 X 和噪声 R ，由 $Y = f_w(X, R)$ 给出，并成为一个随机变量。在训练时，我们为这种有噪声类型的输出找到 w 的最优值。最优值取决于 R ，如 $w^* = w^*(R)$ 所示。为了消除随机性，我们需要在测试时对随机变量 R 进行平均，如下所示

$$y = f(x) = \mathbb{E}_R[f_{w(R)}(x, R)] = \int f_{w(r)}(x, r)p(r) dr, \quad (13.6.16)$$

其中 $p(r)$ 是随机变量 R 的概率密度。

公式(13.6.16)的理论价值大于实际价值，因为右侧的积分很难精确计算。在实际中，我们针对由 r_1, \dots, r_N 给出的随机变量 R 的 N 个实例训练网络，并将权重的最优值获取为 w_i^*, \dots, w_N^* 。这意味着

$$w_i^* = \arg \min_w \|\mathbf{z} - f_w(x, r_i)\|,$$

其中 $\mathbf{z} = \mathbf{z}(x)$ 是网络需要学习的目标函数。在测试时，我们考虑所有 N 输出的平均值，通过以下蒙特卡洛公式评估期望 (13.6.16)：

$$f(x) = \frac{1}{N} \sum_{i=1}^N f_{w_i^*}(x, r_i). \quad (13.6.17)$$

值得注意的是，通过将 R 视为伯努利随机变量，经典的随机失活技术是作为(13.6.17)的一个特殊情况得到的。这意味着 R 以概率 p 取值1，以概率 $1 - p$ 取值0。乘以 $R = 0$ 的神经元激活相当于一个被随机失活的神经元，而乘以值 $R = 1$ 的激活是一个被保留的神经元。因此，每层中 $100(1 - p)$ % 的神经元被随机失活，而 $100p$ % 的神经元被保留。

由于将神经元激活乘以0与假设所有（进入和离开神经元的）权重都消失是一样的，那么(13.6.17)表示 N 个训练子网结果的平均值。

13.7 总结

本章从几何角度讨论神经网络。每个网络都与一个输出流形相关联。流形上的局部坐标是网络的权重和偏差。输出流形概念有助于理解诸如最优权重、学习过程、过拟合和欠拟合以及正则化技术等多个方面。

网络的最优权重和偏差对应于目标在输出流形上的正交投影的坐标。每种学习算法都会改变流形上的坐标，并对应于其上的一条曲线。赋予流形一个黎曼度量能够计算曲线长度，并定义测地线，即两点之间的最短曲线。目标在流形上的起始点和投影点之间的测地线对应于最有效的学习算法。

离输出流形太远的目标点表示欠拟合，而离输出流形太近或在流形上的目标点表示过拟合。

不同类型的正则化方法可以根据输出流形来处理。选择较小维度的输出流形意味着减少权重数量，进而意味着网络中的神经元数量减少，这会导致网络容量下降。

选择最平坦的输出流形会使模型对训练数据的过拟合程度最小。模型平均通过最小化距离来选择一个比其任何组件网络拟合度都更好的模型。随机失活技术属于这一类，也可以被视为一种乘法噪声正则化方法。文中讨论了随机失活与 L_2 正则化之间的关系。

13.8 练习

练习 13.8.1 使用一个 784 - 200 - 100 - 50 - 10 类型的前馈神经网络对 MNIST 数据进行分类。求相关输出流形的维度。(784 是输入大小，10 表示数字类别数量)。

练习 13.8.2 考虑一个神经网络，其输入和输出大小分别由 $d^{(0)}$ 和 $d^{(L)}$ 给出。隐藏神经元的数量用 N 表示。我们假设每个隐藏层中的神经元数量相等。证明输出流形具有最大维度的隐藏层数为

$$k = \frac{2N}{d^{(0)} + d^{(L)} + N}.$$

练习13.8.3 一个单隐藏层前馈神经网络，784 - N - 10，用于对MNIST数据进行分类。找出隐藏神经元数量 N 的范围，在此范围内网络会过拟合训练数据。

练习13.8.4 一个双隐藏层前馈神经网络，784 - h - h - 10，用于对MNIST数据进行分类。找出数量 h 的范围，在此范围内网络会过拟合训练数据。

练习13.8.5 设 $v, u \in T_y S$ 为两个切向量。证明在 \mathbb{R}^n 中 v 和 u 正交当且仅当 $g(u, v) = \sum_{i,j} u_i v_j g_{ij} = 0$ 。

练习13.8.6 \mathbb{R}^n 的子集 \mathcal{A} 若满足对于任意两点 $A, B \in \mathcal{A}$ 都有 $\lambda A + (1 - \lambda)B \in \mathcal{A}, \forall \lambda \in \mathbb{R}$ ，则称其为仿射子空间。设 L 是 S 关于 \mathbb{R}^n 的第二基本形式。证明以下条件等价：

- (a) $L = 0$ ；
- (b) S 中的任何测地线在 \mathbb{R}^n 中都是直线；
- (c) S 是 \mathbb{R}^n 的仿射子空间。

练习13.8.7 设 S 是流形 M 的子流形。证明以下条件等价：

- (a) S 关于 M 的第二基本形式为零， $L = 0$ ；
- (b) 任何在 S 中为测地线的曲线，在 M 中也是测地线。

练习13.8.8 设 $c(s)$ 为输出流形 $S, s \in [a, b]$ 上的一条曲线。它的长度和能量分别定义为

$$L(c) = \int_a^b \|\dot{c}(s)\| ds, \quad \mathcal{E}(c) = \frac{1}{2} \int_a^b \|\dot{c}(s)\|^2 ds,$$

其中 $\|\dot{c}\|$ 表示在 S 的度量结构中沿曲线的速度长度。

- (a) 证明曲线的长度和能量在曲线参数化下是不变的。也就是说，如果 $\phi : [c, d] \rightarrow [a, b]$ 是一个严格递增函数，那么曲线 $\gamma(t) = c(\phi(t))$ 和 $c(s)$ 具有相同的长度和能量。
- (b) 证明 $L(c)^2 \leq 2(b-a)\mathcal{E}(c)$ 。何时达到恒等式？
- (c) 设 $c_u(s), |u| \leq \epsilon$ 为 $c(s)$ 的一个光滑变分，且 $c_0(s) = c(s)$ 。可以证明，两个变分方程

$$\frac{d}{du} L(c_u)|_{u=0} = 0, \quad \frac{d}{du} \mathcal{E}(c_u)|_{u=0} = 0$$

可以写成

$$\ddot{c}^k(s) + \sum_{i,j} \Gamma_{ij}^k(c(s)) \dot{c}^i(s) \dot{c}^j(s) = 0, \quad 1 \leq k \leq n,$$

其中 $c(s) = (c^1(s), \dots, c^n(s))$ 。此外，前面的方程表示沿子流形 \mathcal{S} 的零加速度方程，也可以写成 $\nabla_{\dot{c}} \dot{c} = 0$ 。这些事实的意义是什么？

练习13.8.9 (a) 求二维单位球面 \mathbb{S}^2 在三维欧几里得空间 \mathbb{R}^3 中的嵌入曲率。

(b) 利用 (a) 部分求范数 $\| L \|$ 。试验不同的球参数化。你注意到了什么？

练习13.8.10 考虑两个Sigmoid神经元的模型组合。写出组合的输出并指定相关输出流形的维度。

练习13.8.11 列举从网络中删除神经元对相关输出流形的一些影响。

练习13.8.12 对于 \mathbb{R}^n 中的任意两个向量场

$$U = \sum_k U^k e_k, \quad V = \sum_k V^k e_k,$$

定义 $\nabla_U V = \sum_k U(V^k) e_k$ 。设 f 是 \mathbb{R}^n 上的一个光滑函数。证明以下关系：

- (a) $\nabla_{fU} V = f \nabla_U V$
- (b) $\nabla_U fV = U(f)V + f \nabla_U V$
- (c) $\nabla_U V = \nabla_V U$
- (d) $U \langle V, W \rangle = \langle \nabla_U V, W \rangle + \langle V, \nabla_U W \rangle$ ， 其中 W 是任何其他向量场。



第14章 神经流形

在本章中，我们将从信息几何的角度来研究神经网络。这将微分几何和概率论的技术应用于神经网络。

与第13章介绍的理论的不同之处在于，这里网络的输入和目标是随机变量的概率密度，并且神经网络的输出包含一些噪声扰动。这样，输入和输出的联合概率密度族 $p(x, y; \theta)$ 就成为一个统计流形，它由 θ 参数化；因此，权重和偏差在相关的统计流形中扮演坐标系的角色。在这个空间中，使用费希尔信息度量来测量两个神经网络之间的内在距离。粗略地说，费希尔度量表示训练分布中包含的关于网络自身权重和偏差的信息量。赋予费希尔度量的相关统计流形成为一个黎曼流形，称为神经流形。

在本章中，我们明确计算了几种简单类型网络的费希尔度量，并给出了自然梯度学习算法。对费希尔度量的理解有助于刻画参数空间中的最短曲线——测地线。这很重要，因为神经流形中的每一个运动都对应一个学习过程。自然梯度被定义为相对于由费希尔信息诱导的黎曼度量计算的梯度。

我们提出了自然梯度下降法，作为普通梯度下降的一种更好的替代方法，它能更快地收敛到代价函数的最小值。

还建立了参数空间中距离、曲线长度与能量之间的关系，以及神经流形上的库尔贝克-莱布勒散度。

14.1 统计流形

首先，我们将从神经网络的角度回顾信息几何中关于统计流形的一些术语。考虑由随机变量 X 给出的神经网络输入。输出变量是 $Y = f_\theta(X)$ ，其中 f_θ 是输入-输出映射， $\theta = (\mathbf{w}, \mathbf{b})$ 是网络参数。输入和输出分布分别用 $p_X(x)$ 和 $p_Y(y; \theta)$ 表示。联合输入-输出分布用 $p(x, y; \theta)$ 表示。这里，包含 θ 是为了表明密度对网络参数的依赖性。

目标由随机变量 Z 给出。 (X, Z) 的联合分布是训练分布 $p(x, z)$ 。由于网络输出不能完美复制目标，我们有 $Z = Y + \epsilon(\theta)$ ，其中 $\epsilon(\theta)$ 是一个依赖于网络参数的误差项。在使用均方成本函数作为损失函数的情况下，我们必须最小化误差的二阶矩

$$C(\theta) = \frac{1}{2}\mathbb{E}[(Z - Y)^2] = \frac{1}{2}\mathbb{E}[\epsilon(\theta)^2].$$

有噪声的神经元 信息几何在有噪声神经元的背景下应用于神经网络。这意味着为了尽可能抵消误差 $\epsilon(\theta)$ ，我们需要在网络中插入一些噪声。噪声的想法并不新鲜。我们在第12.8.2节和第13.6.5节中已经看到过，在网络中添加乘性噪声会导致随机失活技术。在这里，我们考虑一个加性噪声项 n ，所以现在网络的输出由

$$Y = f_\theta(X) + n. \quad (14.1.1)$$

值得注意插入噪声在正则化中所起的作用。

假设输出为一维，我们将考虑噪声项的两种情况：

1. 一种可能性是假设噪声为标准正态随机变量， $n \sim \mathcal{N}(0, 1)$ 。那么，在给定输入 X 的情况下，输出 Y 的条件概率在这种情况下由下式给出

$$p(y|x; \theta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y-f_\theta(x))^2}. \quad (14.1.2)$$

这是基于 $\mathbb{E}[Y | X = x] = f_\theta(x)$ 和 $\text{Var}(Y | X = x) = 1$ 这一事实。

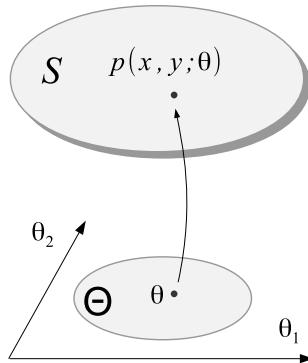


图14.1：一个统计流形 $\mathcal{S} = \{p(x, y; \theta); \theta \in \Theta\}$ 。

2. 另一种可能性是考虑在 -1 与 $1, n \sim \text{Unif}[-1, 1]$ 之间的均匀随机噪声。在这种情况下，条件概率由下式给出

$$p(y|x; \theta) = \begin{cases} \frac{1}{2}, & \text{if } f_\theta(x) - 1 \leq y \leq f_\theta(x) + 1 \\ 0, & \text{otherwise.} \end{cases}$$

现在可以使用条件概率公式来找到 (X, Y) 的联合分布

$$p(x, y; \theta) = p(x)p(y|x; \theta). \quad (14.1.3)$$

目标是调整参数 θ ，使得 $p(x, y; \theta)$ 尽可能多地匹配真实分布 $p(x, z)$ 。这可以通过以下几何方式实现。

统计流形 密度函数族 $\{\theta \rightarrow p(x, y; \theta); \theta \in \Theta\}$ ，在 θ 上参数化，可以被视为概率密度函数的无限维空间的一个子流形。这里， Θ 表示参数空间。还假设以下正则性条件：函数

$$\frac{\partial}{\partial \theta_1} p(x, y; \theta), \dots, \frac{\partial}{\partial \theta_N} p(x, y; \theta) \quad (14.1.4)$$

是线性独立的，其中 $\theta^T = (\theta_1, \dots, \theta_N) \in \Theta$ 。这个条件确保子流形是光滑的，并且在每个点 $p(x, y; \theta)$ 都有一个切空间。流形 $\mathcal{S} = \{p(x, y; \theta); \theta\}$ 被称为统计流形，见图14.1。

在这种设置中，训练密度 $p(x, z)$ 表示概率密度空间中的一个点，一般来说，它并不位于流形 \mathcal{S} 上，见图14.2。通过调整网络参数向量 θ ，我们试图最小化给定密度与相应流形之间的接近程度。如果存在最优参数，则由下式给出

$$\theta^* = \arg \min_{\theta} d(p(x, z), p(x, y; \theta)) = \arg \min_{\theta} D_{KL}(p(x, z) || p(x, y; \theta)),$$

其中 D_{KL} 表示库尔贝克-莱布勒散度。我们注意到可以考虑任何其他成本函数，但由于库尔贝克-莱布勒散度与最大似然估计的关系，它是首选的。

如果距离为零，那么训练分布 $p(x, z)$ 属于统计流形 \mathcal{S} ，并且学习变得精确，即存在参数 θ 的精确值使得 $p(x, z) = p(x, y; \theta)$ 。

统计流形 \mathcal{S} 的对数似然函数由下式定义

$$\ell(\theta) = \ell(x, y; \theta) = \ln p(x, y; \theta).$$

在实践中，上述距离是作为训练数据 $\{(x_1, z_1), \dots, (x_n, z_n)\}$ 上的训练误差来测量的。它由在 (x_k, z_k) 处评估的负对数似然的平均值给出。

$$C_{\text{train}}(\theta) = -\frac{1}{n} \sum_{k=1}^n \ell(x_k, z_k; \theta) = -\frac{1}{n} \prod_{k=1}^n \ln p(x_k, z_k; \theta).$$

最优参数 $\theta^* = \arg \min_{\theta} C_{\text{train}}(\theta)$ 是最大似然估计器 $\theta^* = \theta_{MSE}$ 。

$C_{\text{train}}(\theta)$ 与训练分布和模型分布之间的库尔贝克-莱布勒散度之间的关系在3.6节中已经指出，主要基于以下论点。

$$\begin{aligned} C_{\text{train}}(\theta) &= -\mathbb{E}^{P_{XZ}}[\ell(X, Y; \theta)] \\ &= -\mathbb{E}^{P_{XZ}}[\ln p(X, Y; \theta) - \ln p(X, Z) + \ln p(X, Z)] \\ &= \mathbb{E}^{P_{XZ}}\left[\ln \frac{p(X, Z)}{p(X, Y; \theta)}\right] - \mathbb{E}^{P_{XZ}}\ln[p(X, Z)] \\ &= D_{KL}(p(X, Z) || p(X, Y; \theta)) - H(p(X, Z)). \end{aligned}$$

由于香农熵 $H(p(X, Z))$ 与参数 θ 无关，因此最优参数

$$\theta^* = \arg \min_{\theta} C_{\text{train}}(\theta) = \arg \min_{\theta} D_{KL}(p(X, Z) || p(X, Y; \theta))$$

使训练分布 $p(X, Z)$ 与统计流形 $\mathcal{S} = \{p(X, y; \theta); \theta \in \Theta\}$ 之间的库尔贝克-莱布勒散度最小化。

上述统计流形 $\mathcal{S} = \{p(x, y; \theta); \theta \in \Theta\}$ 可以赋予一个度量结构，并将被视为一个黎曼流形。我们将在下一节介绍这个度量。

值得注意的是，包含目标值 z_k 的公式具有外在特征，因为它们访问关于统计流形外部点 $p(x, z)$ 的信息。涉及 x_k 和 $y_k = f_{\theta}(x_k)$ 的公式具有内在特征，因此它们属于统计流形 \mathcal{S} 的局部几何。从这个角度来看，上述训练误差 $C_{\text{train}}(\theta)$ 是一个外在对象。下一个目标是在统计流形上引入费希尔度量，它是一个内在对象。

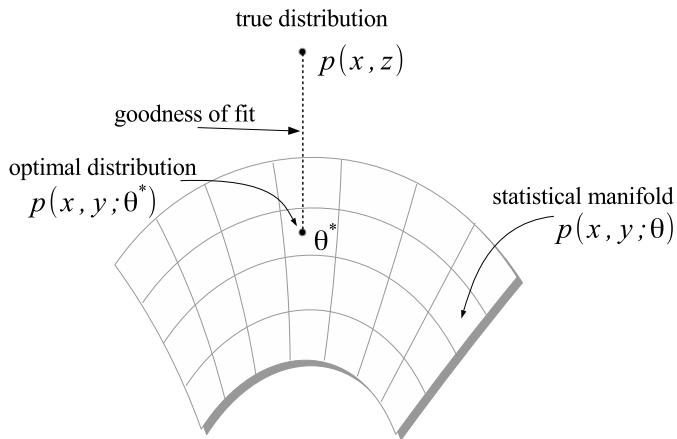


图14.2：拟合优度（损失函数）的几何图像。

14.2 费希尔信息

假设概率密度 $p(x; \theta)$ 依赖于实值参数 θ ，并且我们希望通过一个依赖于数据 x 的无偏估计量 $\hat{\theta} = \hat{\theta}(x)$ 来估计这个参数。这意味着在参数 θ 存在的情况下，所有数据集上差值 $\hat{\theta}(x) - \theta$ 的平均值为零。

$$\mathbb{E}[\hat{\theta}(x) - \theta] = \int (\hat{\theta}(x) - \theta)p(x; \theta) dx = 0.$$

为简单起见，记为 $p = p(x; \theta)$ ，并使用乘积法则对前一个关系式关于 θ 求导。

$$0 = \frac{\partial}{\partial \theta} \mathbb{E}[\hat{\theta}(x) - \theta] = \int (\hat{\theta}(x) - \theta) \frac{\partial p}{\partial \theta} dx - \int p dx.$$

由于 $\int p dx = 1$ 和 $\frac{\partial p}{\partial \theta} = \frac{\partial \ln p}{\partial \theta} p$ ，前一个关系式意味着

$$\int (\hat{\theta}(x) - \theta) \frac{\partial \ln p}{\partial \theta} p dx = 1.$$

一个关键特性是将密度 p 分解为两个平方根的乘积，然后将表达式重写为

$$\int ((\hat{\theta}(x) - \theta)\sqrt{p}) \left(\frac{\partial \ln p}{\partial \theta} \sqrt{p} \right) dx = 1.$$

取平方并使用柯西积分不等式，可得

$$1 \leq \int (\hat{\theta}(x) - \theta)^2 p dx \int \left(\frac{\partial \ln p}{\partial \theta} \right)^2 p dx.$$

右侧的每个积分都具有期望的意义。第一个，

$$\mathbb{E}[(\hat{\theta} - \theta)^2] = \int (\hat{\theta}(x) - \theta)^2 p dx,$$

是均方误差，而后者，

$$I(\theta) = \mathbb{E}\left[\left(\frac{\partial \ln p}{\partial \theta}\right)^2\right] = \int \left(\frac{\partial \ln p}{\partial \theta}\right)^2 p dx,$$

是关于 θ 的费希尔信息。前面的不等式现在可以写成

$$\mathbb{E}[(\hat{\theta} - \theta)^2] I(\theta) \geq 1,$$

或者等价地，写成

$$\mathbb{E}[(\hat{\theta} - \theta)^2] \geq \frac{1}{I(\theta)}. \quad (14.2.5)$$

这称为克拉美 - 罗不等式。它表明费希尔信息的倒数是均方误差的下界。¹ 因此，最小均方误差估计器 $\hat{\theta}_{MSE}$ 满足恒等式

$$\mathbb{E}[(\hat{\theta}_{MSE} - \theta)^2] = \frac{1}{I(\theta)}. \quad (14.2.6)$$

因此，当信息含量高时，误差就低，反之亦然。费希尔信息 $I(\theta)$ 表示对数似然函数 $\ell(\theta) = \ln p(x; \theta)$ 的导数内容。 $\ell(\theta)$ 相对于 θ 变化得越快，信息 $I(\theta)$ 就越大，均方误差就越小。我们稍后将看到，在克拉美 - 罗不等式中实现等式的估计量被称为费希尔有效。

费希尔信息是对包含在由密度族 $p(\theta)$ 建模的随机变量 X 中的关于未知参数 θ 的信息的一种评估。因此， θ 的一个估计量的方差大于先前信息的倒数，并且当它的方差尽可能低时它变得有效。如果在 X 中包含的关于 θ 的信息很少，那么克拉美 - 罗不等式表明 θ 的任何估计都是不精确的，从它有很大方差这个意义上来说。

¹ 有时，这等效地表述为 $\text{Var}(\hat{\theta}) \geq \frac{1}{I(\theta)}$ 。

我们还注意到，费希尔信息可以根据对数似然函数写成如下形式

$$I(\theta) = \mathbb{E}\left[\left(\frac{\partial\ell(\theta)}{\partial\theta}\right)^2\right].$$

这个表达式起到了密度函数 $p(x; \theta)$ 的作用，它仅依赖于一个参数 θ 。多维情况将在下面讨论。

多元情况 当参数 θ 是 n 维的 $\theta = (\theta_1, \dots, \theta_n)$ 时，可以引入类似的概念。在这种情况下，我们得到费希尔信息矩阵

$$g_{ij}(\theta) = \mathbb{E}\left[\frac{\partial\ell(\theta)}{\partial\theta_i}\frac{\partial\ell(\theta)}{\partial\theta_j}\right], \quad (14.2.7)$$

其中 $\ell(\theta) = \ln p(x; \theta)$ ，并且期望是关于 $p(x; \theta)$ 取的。下面给出其他等价表达式：

命题14.2.1 费希尔矩阵也可以表示为：

$$g_{ij}(\theta) = -\mathbb{E}\left[\frac{\partial^2\ell(\theta)}{\partial\theta_i\partial\theta_j}\right] \quad (14.2.8)$$

$$g_{ij}(\theta) = 4 \int \frac{\partial\sqrt{p(x; \theta)}}{\partial\theta_i} \frac{\partial\sqrt{p(x; \theta)}}{\partial\theta_j} dx. \quad (14.2.9)$$

证明：对 $\int p(x; \theta)dx = 1$ 关于 θ_i 求导可得

$$\int \partial_{\theta_i} p(x; \theta) dx = 0,$$

这等同于

$$\int \partial_{\theta_i} \ln p(x; \theta) p(x; \theta) dx = 0.$$

再对 θ_j 求一次导并应用乘积法则，我们得到

$$\begin{aligned} \int \partial_{\theta_j} \partial_{\theta_i} \ln p(x; \theta) p(x; \theta) dx + \int \partial_{\theta_i} \ln p(x; \theta) \partial_{\theta_j} p(x; \theta) dx &= 0 \\ \mathbb{E}\left[\frac{\partial^2\ell(\theta)}{\partial\theta_i\partial\theta_j}\right] + \int \partial_{\theta_i} \ln p(x; \theta) \partial_{\theta_j} \ln p(x; \theta) p(x; \theta) dx &= 0 \\ \mathbb{E}\left[\frac{\partial^2\ell(\theta)}{\partial\theta_i\partial\theta_j}\right] + g_{ij}(\theta) &= 0, \end{aligned}$$

这意味着关系式(14.2.8)。

为了证明关系式(14.2.9)，我们进行如下直接计算：

$$\begin{aligned} g_{ij}(\theta) &= \int \partial_{\theta_i} \ln p(x; \theta) \partial_{\theta_j} \ln p(x; \theta) p(x; \theta) dx \\ &= \int \frac{\partial_{\theta_i} p(x; \theta)}{p(x; \theta)} \frac{\partial_{\theta_j} p(x; \theta)}{p(x; \theta)} p(x; \theta) dx \\ &= 4 \int \frac{\partial_{\theta_i} p(x; \theta)}{2\sqrt{p(x; \theta)}} \frac{\partial_{\theta_j} p(x; \theta)}{2\sqrt{p(x; \theta)}} dx \\ &= 4 \int \frac{\partial \sqrt{p(x; \theta)}}{\partial \theta_i} \frac{\partial \sqrt{p(x; \theta)}}{\partial \theta_j} dx. \end{aligned}$$

■

关系式(14.2.8)将费舍尔矩阵与对数似然函数的负海森矩阵联系起来，并有以下几何解释。如果 $\ell(x; \theta) = \ln p(x; \theta)$ 是对应于观测值 x 和参数 θ 的对数似然函数，那么 θ 的最大似然估计量是

$$\hat{\theta}_{MLE} = \arg \max_{\theta} \ell(x; \theta).$$

由于在 $\theta = \hat{\theta}_{MLE}$ 处存在最大值，那么 $\frac{\partial}{\partial \theta_i} \ell(x; \hat{\theta}_{MLE}) = 0$ 。

在 $\hat{\theta}_{MLE}$ 附近展开，我们有

$$\begin{aligned} \ell(x; \theta) &= \ell(x; \hat{\theta}_{MLE}) + \frac{1}{2} \sum_{i,j} \frac{\partial^2}{\partial \theta_i \partial \theta_j} \ell(x; \hat{\theta}_{MLE}) (\hat{\theta}_{MLE,i} - \theta_i)(\hat{\theta}_{MLE,j} - \theta_j) \\ &\quad + O(\|\hat{\theta}_{MLE} - \theta\|^3) \end{aligned}$$

如 (14.2.8) 所示，由右侧求和的负系数期望给出的费希尔矩阵，衡量了 $\ell(x; \theta)$ 在 $\hat{\theta}_{MLE}$ 处的波峰曲率。

可以证明，在正则性假设 (14.1.4) 下，关系式 (14.2.9) 可用于证明 $g_{ij}(\theta)$ 是一个对称、正定且非退化的矩阵（例如，见 [22] 中的命题 1.6.2）。

因此，费希尔信息矩阵给出了统计流形 $\mathcal{S} = \{\theta \rightarrow p(x; \theta)\}$ 上一个黎曼度量（见 13.1.3 节）的系数，称为费希尔度量。这使得可以计算统计流形上向量的长度、角度、距离和面积。有时，在参数空间 Θ 上考虑费希尔度量 $g_{ij}(\theta)$ 。

一个自然的问题是，在所有可以定义在统计流形上的黎曼度量中，是什么使得费希尔度量与众不同？可以证明，费希尔度量具有以下两个性质，见[4]和[22]：

1. g_{ij} 在样本空间的重新参数化下是不变的。这意味着统计流形

$\mathcal{S} = \{p(x; \theta); \theta \in \Theta\}$ 和 $\tilde{\mathcal{S}} = \{p(h(x); \theta); \theta \in \Theta\}$ ，其中 h 是可逆且可微的函数，具有相等的度量， $g_{ij}(\theta) = \tilde{g}_{ij}(\theta)$ 。这种不变性性质可以在[22]的定理1.6.4中找到。

2. g_{ij} 在重新参数化下是协变的。这意味着，如果我们考虑一个依赖于 θ 的不同参数化 $\xi_j = \xi(\theta_1, \dots, \theta_N)$ ，那么这两种参数化下的费希尔矩阵通过以下关系相关联：

$$g_{ij}(\theta) = \sum_{k,r} g_{kr}(\xi) \Big|_{\xi=\xi(\theta)} \frac{\partial \xi^k}{\partial \theta^i} \frac{\partial \xi^r}{\partial \theta^j}.$$

关于这一事实的证明，请参阅 [22] 的定理1.6.5。

使费希尔度量如此特殊的是，它是唯一满足前两个不变性条件的度量。这一卓越结果的证明可在 [28] 中找到。

克拉美 - 罗不等式 该不等式 (14.2.5) 有一个多元版本。向量参数是 $\theta = (\theta_1, \dots, \theta_N)^T \in \mathbb{R}^N$ ，它代表参数空间 (Θ, g) 的一个坐标系。考虑一个估计量

$$\hat{\theta}(X) = (\hat{\theta}_1(X), \dots, \hat{\theta}_N(X))^T$$

它是无偏的， $\mathbb{E}[\hat{\theta}(X)] = \theta$ 。那么

$$\text{Cov}(\hat{\theta}(X)) \geq g(\theta)^{-1}. \quad (14.2.10)$$

这意味着对于所有 $\theta \in \Theta$ ，差分矩阵 $A_{ij} = \text{Cov}(\hat{\theta}_i(X), \hat{\theta}_j(X)) - g^{ij}(\theta)$ 是半正定的，即它具有非负特征值。在稍后讨论有效估计量时，这个不等式会很有用。

关于 Fisher 信息在科学中的进一步应用，读者可参考 [40]、[41] 和 [39]。关于在其他类型神经元中的应用，请参阅 [121]。

14.3 神经网络的神经流形

神经流形是一个与神经网络相关联的黎曼流形，如下所述。设 $y = f_\theta(x)$ 为给定神经网络的输入 - 输出映射，其输入和输出密度分别为 $p_X(x)$ 和 $p_Y(y; \theta)$ ，联合密度为 $p(x, y; \theta)$ ，其中 θ 是由网络的所有权重和偏差组成的向量参数。

与神经网络相关联的统计流形是 $\mathcal{S} = \{p(x, y; \theta); \theta \in \Theta\}$ 。费希尔度量可以通过公式(14.2.7)在 \mathcal{S} 上定义，其中我们考虑由 $\ell(x, y; \theta) = \ln p(x, y; \theta)$ 给出的对数似然函数。这也可以表示为以下二重积分：

$$g_{ij}(\theta) = \iint \frac{\partial \ell(x, y; \theta)}{\partial \theta_i} \frac{\partial \ell(x, y; \theta)}{\partial \theta_j} p(x, y; \theta) dx dy. \quad (14.3.11)$$

由于 $g = g_{ij}$ 是 \mathcal{S} 上的黎曼度量，那么 (\mathcal{S}, g) 成为一个黎曼流形。

定义14.3.1 与上述神经网络相关联的神经流形是黎曼流形 (\mathcal{S}, g) ，其中 $\mathcal{S} = \{p(x, y; \theta); \theta \in \Theta\}$ 是神经网络联合输入 - 输出密度的统计流形， θ 是网络权重和偏差，并且 g 是费希尔度量。

与神经网络 $p(x, y; \theta)$ 相关联的每个联合概率密度都可以被视为这个流形上的一个点。学习过程，即参数的调整，可以被可视化为神经流形上的一条曲线。

值得注意的是，度量 $g_{ij}(\theta)$ 独立于目标值 z_k ，即它是一个内在对象。所有从费希尔信息派生的概念将构成神经流形的内在几何。

接下来的计算将在公式(14.1.1)中的噪声为标准正态噪声，即 $n \sim \mathcal{N}(0, 1)$ 的假设下进行。有时，人们可能会考虑缩放噪声，即 $n \sim \mathcal{N}(0, s^2)$ ，并将标准差 s 视为超参数。使用(14.1.3)可得到以下对数似然函数

$$\begin{aligned} \ell(x, y; \theta) &= \ln p(x) + \ln p(y|x; \theta) \\ &= \ln p(x) - \ln(\sqrt{2\pi}) - \frac{1}{2}(y - f_\theta(x))^2, \end{aligned}$$

其偏导数为

$$\frac{\partial \ell(x, y; \theta)}{\partial \theta_k} = (y - f_\theta(x)) \frac{\partial f_\theta(x)}{\partial \theta_k}. \quad (14.3.12)$$

输入 - 输出映射相对于参数 $\frac{\partial f_\theta(x)}{\partial \theta_k}$ 的灵敏度，因神经网络的类型而异，是衡量每个网络复杂度的一种度量。我们将在一些特定情况下进行计算，然后在前馈神经网络的情况下给出一个通用的递归公式。

14.4 单个神经元的费希尔度量

在本节中，我们将给出单个神经元情况下费希尔度量的显式公式。我们考虑一个神经元，其输入由 n 维随机向量

$X = (X_1, \dots, X_n)^T$ 给出，输入 - 输出映射为

$f_\theta(x) = \phi(w^T x + b)$ ，参数为 $\theta = (w, b)$ ，激活函数为可微的 ϕ 。应用链式法则可得到偏导数

$$\frac{\partial f_\theta(x)}{\partial w_k} = x_k \phi'(w^T x + b), \quad \frac{\partial f_\theta(x)}{\partial b} = \phi'(w^T x + b).$$

然后公式(14.3.12)给出

$$\begin{aligned} \frac{\partial \ell(x, y; \theta)}{\partial w_k} &= x_k (y - \phi(w^T x + b)) \phi'(w^T x + b), \quad 1 \leq k \leq n \\ \frac{\partial \ell(x, y; \theta)}{\partial b} &= (y - \phi(w^T x + b)) \phi'(w^T x + b). \end{aligned}$$

由于 $\theta = (w_1, \dots, w_n, b)$ ，费希尔矩阵是 $(n+1)$ 维的。利用(14.3.11)并改变积分顺序，我们有

$$\begin{aligned} g_{00}(w, b) &= \iint \left(\frac{\partial \ell(x, y; \theta)}{\partial b} \right)^2 p(x, y) dx dy \\ &= \iint (y - \phi(w^T x + b))^2 \phi'(w^T x + b)^2 p(x) p(y|x; \theta) dx dy \\ &= \int \phi'(w^T x + b)^2 p(x) \int (y - \phi(w^T x + b))^2 p(y|x; \theta) dy dx. \end{aligned}$$

用(14.1.2)中的 $p(y|x; \theta)$ 进行替换并对变量 $u = \phi(w^T x + b)$ 进行变换后得到

$$\begin{aligned} g_{00}(w, b) &= \int \phi'(w^T x + b)^2 p(x) \underbrace{\left(\int u^2 \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2} du \right)}_{=1} dx \\ &= \int \phi'(w^T x + b)^2 p(x) dx \\ &= \mathbb{E}^{P_X} [\phi'(w^T X + b)^2], \end{aligned}$$

其中 \mathbb{E}^{P_X} 表示在输入 X 的概率分布下的期望算子。

同样地，

$$\begin{aligned}
 g_{0k}(w, b) &= \iint \frac{\partial \ell(x, y; \theta)}{\partial b} \frac{\partial \ell(x, y; \theta)}{\partial w_k} p(x, y) dx dy \\
 &= \iint (y - \phi(w^T x + b))^2 \phi'(w^T x + b)^2 x_k p(x) p(y|x; \theta) dx dy \\
 &= \int \phi'(w^T x + b)^2 x_k p(x) \underbrace{\int (y - \phi(w^T x + b))^2 p(y|x; \theta) dy}_{=1} dx \\
 &= \mathbb{E}^{P_X}[X_k \phi'(w^T X + b)^2].
 \end{aligned}$$

此外，

$$\begin{aligned}
 g_{jk}(w, b) &= \iint \frac{\partial \ell(x, y; \theta)}{\partial w_j} \frac{\partial \ell(x, y; \theta)}{\partial w_k} p(x, y) dx dy \\
 &= \iint (y - \phi(w^T x + b))^2 \phi'(w^T x + b)^2 x_j x_k p(x) p(y|x; \theta) dx dy \\
 &= \int \phi'(w^T x + b)^2 x_j x_k p(x) \underbrace{\int (y - \phi(w^T x + b))^2 p(y|x; \theta) dy}_{=1} dx \\
 &= \mathbb{E}^{P_X}[X_j X_k \phi'(w^T X + b)^2].
 \end{aligned}$$

总之，最后三个公式可以合并为一个公式，即

$$\tilde{g}_{ij}(\tilde{w}) = \mathbb{E}^{P_X}[X_i X_j \phi'(\tilde{w}^T \tilde{X})^2], \quad 0 \leq i, j \leq n, \quad (14.4.13)$$

其中 $\tilde{w}^T = (w^T, b)$ 和 $\tilde{X} = (X_0, X)$ ，以及 $X_0 = 1$ 。一般来说，公式(14.4.13)无法进一步简化。然而，如果 X_i 是相互独立的标准正态分布，且 $b = 0$ ，那么费舍尔矩阵及其逆矩阵存在闭式解，见[6]。在研究几种特定类型的神经元之后，我们将展开这个想法。

线性神经元 在这种情况下，激活函数是 $\phi(x) = x$ ，所以在前述公式中用1替换导数 $\phi'(x)$ ，可得

$$g_{00} = 1, \quad g_{0k}(w, b) = \mathbb{E}^{P_X}[X_k], \quad g_{jk}(w, b) = \mathbb{E}^{P_X}[X_j X_k].$$

这些公式表明，线性神经元的费舍尔矩阵描述了输入向量 $\tilde{X} = (1, X)$ 的自协方差。此外，由于 $g_{jk}(w, b)$ 不依赖于 $\theta = (w, b)$ ，那么 $\frac{\partial g_{ij}}{\partial \theta_k} = 0$ ，这意味着克里斯托费尔符号消失， $\Gamma_{ij}^k = 0$ 。因此，线性神经元的神经流形本质上是平坦的，其所有内在几何仅由输入协方差诱导。

ReLU神经元 考虑激活函数 $\phi(x) = \text{ReLU}(x)$ ，它是分段可微的。我们将按照附录F.2节中的广义意义对其求导。然后 $\phi'(x) = \text{ReLU}'(x) = H(x)$ ，见练习8.8.1。由于 $H^2(x) = H(x)$ ，我们得到

$$\begin{aligned} g_{00} &= \mathbb{E}^{P_X} [\text{ReLU}'(w^T X + b)^2] = \mathbb{E}^{P_X} [H(w^T X + b)^2] = \mathbb{E}^{P_X} [H(w^T X + b)] \\ &= \int_{\{w^T x + b \geq 0\}} p(x) dx = \mathbb{P}(w^T X + b \geq 0) = \mathbb{P}(X \in \mathcal{H}_{w,b}). \end{aligned}$$

因此，系数 g_{00} 表示输入向量 X 属于半空间 $\mathcal{H}_{w,b} = \{w^T x + b \geq 0\}$ 的概率。我们还注意到 $0 \leq g_{00} \leq 1$ 。它的值取决于超平面平移参数 b 的值，如下所示

$$\lim_{b \rightarrow \infty} g_{00} = 0, \quad \lim_{b \rightarrow -\infty} g_{00} = 1.$$

其他度量系数由下式给出

$$\begin{aligned} g_{0k} &= \mathbb{E}^{P_X} [X_k \text{ReLU}'(w^T X + b)^2] = \mathbb{E}^{P_X} [X_k H(w^T X + b)] \\ &= \int_{\{w^T x + b \geq 0\}} x_k p(x) dx, \\ g_{jk} &= \mathbb{E}^{P_X} [X_j X_k \text{ReLU}'(w^T X + b)^2] = \mathbb{E}^{P_X} [X_j X_k H(w^T X + b)] \\ &= \int_{\{w^T x + b \geq 0\}} x_j x_k p(x) dx. \end{aligned}$$

使用 $0 \leq H(w^T x + b) \leq 1$ ，可得

$g_{0k} \leq \mathbb{E}^{P_X} [X_k]$, $g_{jk} \leq \mathbb{E}^{P_X} [X_j X_k]$ 。事实上，我们有

$$\lim_{b \rightarrow \infty} g_{0k} = \lim_{b \rightarrow \infty} g_{jk} = 0, \quad \lim_{b \rightarrow -\infty} g_{0k} = \mathbb{E}[X_k], \quad \lim_{b \rightarrow -\infty} g_{jk} = \mathbb{E}[X_j X_k].$$

双曲正切神经元 激活函数 $\phi(x) = \tanh x$ 满足 $\phi'(x) = 1 - \tanh^2 x$ 。通常，为简便起见，我们记为 $\mathbf{t}(x) = \tanh x$ 。费希尔度量系数由下式给出

$$\begin{aligned} g_{00} &= \mathbb{E}^{P_X} [\mathbf{t}'(w^T X + b)^2] = \mathbb{E}^{P_X} [(1 - \mathbf{t}^2(w^T X + b))^2] \\ g_{0k} &= \mathbb{E}^{P_X} [X_k \mathbf{t}'(w^T X + b)^2] = \mathbb{E}^{P_X} [X_k (1 - \mathbf{t}^2(w^T X + b))^2] \\ g_{jk} &= \mathbb{E}^{P_X} [X_j X_k \mathbf{t}'(w^T X + b)^2] = \mathbb{E}^{P_X} [X_j X_k (1 - \mathbf{t}^2(w^T X + b))^2], \end{aligned}$$

对于 $1 \leq j, k \leq n$ 。值得注意这些不等式

$$0 \leq g_{00} \leq 1, \quad g_{0k} \leq \mathbb{E}^{P_X} [X_k], \quad g_{jk} \leq \mathbb{E}^{P_X} [X_j X_k].$$

14.5 费希尔矩阵及其逆矩阵

为了计算费希尔矩阵 (14.4.13) 及其逆矩阵, 必须假设一些额外条件。我们将考虑 $(X_1, \dots, X_n) \sim \mathcal{N}(0, \mathbb{I}_n)$, 即输入是一个多元标准正态随机变量。记为 $|w|^2 = w^T w = \sum w_i^2$ 并考虑这些函数

$$C_1(w, b) = \frac{1}{|w|^2 \sqrt{2\pi}} \int \phi'(|w|\epsilon + b)^2 e^{-\frac{1}{2}\epsilon^2} d\epsilon \quad (14.5.14)$$

$$C_2(w, b) = \frac{1}{|w|^2 \sqrt{2\pi}} \int \phi'(|w|\epsilon + b)^2 \epsilon^2 e^{-\frac{1}{2}\epsilon^2} d\epsilon \quad (14.5.15)$$

$$C_3(w, b) = \frac{1}{|w|^2 \sqrt{2\pi}} \int \phi'(|w|\epsilon + b)^2 \epsilon e^{-\frac{1}{2}\epsilon^2} d\epsilon. \quad (14.5.16)$$

利用 $w^T X = \sum w_i X_i \sim \mathcal{N}(0, |w|^2)$, 我们可以写成
 $w^T X = |w|\epsilon$, 其中 $\epsilon \sim \mathcal{N}(0, 1)$ 。然后我们有

$$\begin{aligned} g_{00}(w, b) &= \mathbb{E}[\phi'(w^T X + b)^2] = \mathbb{E}[\phi'(|w|\epsilon + b)^2] \\ &= \frac{1}{\sqrt{2\pi}} \int \phi'(|w|\epsilon + b)^2 e^{-\frac{1}{2}\epsilon^2} d\epsilon = |w|^2 C_1(w, b). \end{aligned} \quad (14.5.17)$$

我们接着计算 $g_{0k} = \mathbb{E}[X_k \phi'(w^T X + b)^2]$, $1 \leq k \leq n$ 。再次写成 $w^T X = |w|\epsilon$, 我们有

$$\begin{aligned} \sum_k g_{0k} w_k &= \mathbb{E}[w^T X \phi'(w^T X + b)^2] = \mathbb{E}[|w|\epsilon \phi'(|w|\epsilon + b)^2] \\ &= \frac{|w|}{\sqrt{2\pi}} \int \epsilon \phi'(|w|\epsilon + b)^2 e^{-\frac{1}{2}\epsilon^2} d\epsilon \\ &= |w|^3 C_3(w, b). \end{aligned} \quad (14.5.18)$$

设 v 为任意单位向量, 与 w 正交。那么

$$\begin{aligned} \sum_k g_{0k} v_k &= \mathbb{E}[v^T X \phi'(w^T X + b)^2] = \mathbb{E}[v^T X] \mathbb{E}[\phi'(w^T X + b)^2] \\ &= \mathbb{E}[\epsilon] \mathbb{E}[\phi'(w^T X + b)^2] = 0, \end{aligned}$$

在我们使用 $v^T X = |v|\epsilon = \epsilon \sim \mathcal{N}(0, 1)$ 的地方, $\mathbb{E}[\epsilon] = 0$ 以及 $w^T X$ 和 $v^T X$ 相互独立这一事实, 见练习14.13.3。由此可知向量 $(g_{01}, \dots, g_{0n})^T$ 与所有向量 v (它们与 w 垂直) 正交。因此, $(g_{01}, \dots, g_{0n})^T$ 必定与 w 成比例, 即存在 $\lambda \in \mathbb{R}$ 使得

$$g_{0k} = \lambda w_k, \quad 1 \leq k \leq n. \quad (14.5.19)$$

要确定 λ ，乘以 w_k 并求和

$$\sum g_{0k} w_k = \lambda \sum w_k^2 = \lambda |w|^2,$$

从哪里

$$\lambda = \frac{1}{|w|^2} \sum g_{0k} w_k = \lambda \sum w_k^2 = |w| C_3(w, b),$$

我们在此处使用了(14.5.18)。然后(14.5.19)得出

$$g_{0k} = w_k |w| C_3(w, b), \quad 1 \leq k \leq n.$$

注记14.5.1我们作如下注记：如果 $b = 0$ 且 $\phi(x) = \tanh(x)$ ，那么 $C_3(w, b) = 0$ ，因为 $\phi'(|w|\epsilon)^2$ 是关于 ϵ 的偶函数。因此，
 $g_{0k} = 0, 1 \leq k \leq n$.

我们将在下面证明矩阵

$$g_{jk} = \mathbb{E}[X_j X_k \phi'(w^T X + b)^2], \quad 1 \leq j, k \leq n$$

具有显式形式

$$g_{jk} = |w|^2 C_1(w, b) \delta_{jk} + (C_2(w, b) - C_1(w, b)) w_j w_k,$$

其中，对于 $j = k$ ， δ_{jk} 为1，否则为0。以等效矩阵形式，该矩阵

$$g = \mathbb{E}[XX^T \phi'(w^T X + b)^2]$$

可写成以下求和形式：

$$g = |w|^2 C_1(w, b) \mathbb{I}_n + (C_2(w, b) - C_1(w, b)) w w^T. \quad (14.5.20)$$

为简单起见，用 h 表示(14.5.20)式右边的矩阵。由于 g 和 h 都是对称矩阵，为证明 $g = h$ ，我们将使用练习14.13.1。因此，只需证明对于所有垂直于 w 的单位向量 v ，有 $w^T g w = w^T h w$ 和 $v^T g v = v^T h v$ 。我们将分两步进行：步骤1：证明 $w^T g w = w^T h w$ 。
 。左边可计算为

$$\begin{aligned} w^T g w &= w^T \mathbb{E}[XX^T \phi'(w^T X + b)^2] w = \mathbb{E}[(w^T X)^2 \phi'(w^T X + b)^2] \\ &= \mathbb{E}[|w|^2 \epsilon^2 \phi'(|w|\epsilon + b)^2] = \frac{|w|^2}{\sqrt{2\pi}} \int \epsilon^2 \phi'(|w|\epsilon + b)^2 e^{-\frac{\epsilon^2}{2}} d\epsilon \\ &= |w|^4 C_2(w, b). \end{aligned} \quad (14.5.21)$$

右边为

$$\begin{aligned} w^T h w &= |w|^2 C_1(w, b) w^T w + (C_2(w, b) - C_1(w, b)) w^T w w^T w \\ &= |w|^4 C_1(w, b) + (C_2(w, b) - C_1(w, b)) |w|^4 \\ &= |w|^4 C_2(w, b). \end{aligned} \quad (14.5.22)$$

由于(14.5.21)和(14.5.22)一致，我们得到了所需的恒等式。

步骤2：证明 $v^T gv = v^T hv$ 。利用关系式(14.5.17)以及 $w^T X$ 和 $v^T X$ 相互独立这一事实，见练习14.13.2，左边变为

$$\begin{aligned} v^T gv &= v^T \mathbb{E}[XX^T \phi'(w^T X + b)^2]v = \mathbb{E}[(v^T X)^2 \phi'(w^T X + b)^2] \\ &= \mathbb{E}[(v^T X)^2] \mathbb{E}[\phi'(w^T X + b)^2] = \underbrace{\mathbb{E}[\epsilon^2]}_{=1} \underbrace{\mathbb{E}[\phi'(|w|\epsilon + b)^2]}_{=g_{00}} \\ &= |w|^2 C_1(w, b). \end{aligned} \quad (14.5.23)$$

对于右边，我们有直接的计算

$$\begin{aligned} v^T hv &= v^T (|w|^2 C_1(w, b) \mathbb{I}_n + (C_2(w, b) - C_1(w, b))ww^T)v \\ &= |w|^2 C_1(w, b)v^T v + (C_2(w, b) - C_1(w, b))(v^T w)^2 \\ &= |w|^2 C_1(w, b), \end{aligned} \quad (14.5.24)$$

其中我们使用了正交归一条件 $v^T v = 1$ 和 $v^T w = 0$ 。由于(14.5.23)和(14.5.24)一致，我们证明了所需的恒等式。

总之，费希尔矩阵由以下 $(n+1) \times (n+1)$ 矩阵给出

$$\tilde{g} : \begin{cases} g_{00} = |w|^2 C_1(w, b) \\ g_{0k} = g_{k0} = w_k |w| C_3(w, b), \quad 1 \leq k \leq n \\ g_{jk} = |w|^2 C_1(w, b) \delta_{jk} + (C_2(w, b) - C_1(w, b)) w_j w_k, \quad 1 \leq j, k \leq n. \end{cases}$$

求 \tilde{g} 的逆 如果我们将费希尔矩阵写成如下形式会更容易

$$\tilde{g} = \begin{pmatrix} g_{00} & g_{01} & \cdots & g_{0n} \\ g_{10} & & & \\ \vdots & & g & \\ g_{n0} & & & \end{pmatrix}.$$

$(n \times n)$ 块 $g = g_{ij}, 1 \leq i, j \leq n$ 可以用封闭形式求逆。我们将寻找一个与(14.5.20)形式相似的逆。

$$g^{-1} = \rho_1 \mathbb{I}_n + \rho_2 ww^T,$$

并根据 w 和 b 确定函数 ρ_1 和 ρ_2 ，使得 $gg^{-1} = \mathbb{I}_n$ 。使用 $ww^T ww^T = w|w|^2 w^T = |w|^2 ww^T$ ，通过代数计算可得

$$\begin{aligned} gg^{-1} &= (|w|^2 C_1 \mathbb{I}_n + (C_2 - C_1)ww^T)(\rho_1 \mathbb{I}_n + \rho_2 ww^T) \\ &= |w|^2 C_1 \rho_1 \mathbb{I}_n + [\rho_1(C_2 - C_1) + |w|^2 \rho_2 C_2]ww^T. \end{aligned}$$

通过系数识别，我们提出

$$\begin{aligned}|w|^2 C_1 \rho_1 &= 1 \\ \rho_1(C_2 - C_1) + |w|^2 \rho_2 C_2 &= 1,\end{aligned}$$

其解为

$$\rho_1 = \frac{1}{|w|^2 C_1}, \quad \rho_2 = \frac{1}{|w|^4} \left(\frac{1}{C_2} - \frac{1}{C_1} \right).$$

因此， g 的逆由下式给出

$$g^{-1} = \frac{1}{|w|^2 C_1} \mathbb{I}_n + \frac{1}{|w|^4} \left(\frac{1}{C_2} - \frac{1}{C_1} \right) w w^T. \quad (14.5.25)$$

该结果在[6]中作为定理4出现。

对矩阵 \tilde{g} 求逆时，我们考虑两种情况：

情况1： $b = 0$ 和 $\phi(x) = \tanh(x)$ 。在此情况下

$$C_3(w, 0) = \frac{1}{|w|^2 \sqrt{2\pi}} \int \phi'(|w|\epsilon)^2 \epsilon e^{-\frac{1}{2}\epsilon^2} d\epsilon = 0,$$

由于 $\phi'(|w|\epsilon)^2$ 是偶函数。因此，对于 $1 \leq k \leq n$ ，有 $g_{0k} = g_{k0} = 0$ ，并且矩阵可以逐块求逆，如下所示

$$\tilde{g}^{-1} = \begin{pmatrix} \frac{1}{g_{00}} & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & g^{-1} & \\ 0 & & & \end{pmatrix},$$

g^{-1} 由(14.5.25)给出。

情况2：一般情况。我们将说明如何以迭代方式计算 \tilde{g}^{-1} 。

首先，我们将 \tilde{g} 分解为两个矩阵之和

$$\tilde{g} = \begin{pmatrix} 0 & g_{01} & \cdots & g_{0n} \\ g_{10} & & & \\ \vdots & & \mathbb{O}_n & \\ g_{n0} & & & \end{pmatrix} + \begin{pmatrix} g_{00} & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & g & \\ 0 & & & \end{pmatrix} = A_1 + A_2.$$

矩阵 A_2 是可逆的，其已知逆为

$$A_2^{-1} = \begin{pmatrix} \frac{1}{g_{00}} & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & g^{-1} & \\ 0 & & & \end{pmatrix}.$$

两个矩阵之和的求逆在附录G节中介绍。使用展开法，我们有

$$\tilde{g}^{-1} = (A_1 + A_2)^{-1} = A_2^{-1} \sum_{k \geq 0} (-1)^k (A_1 A_2^{-1})^k. \quad (14.5.26)$$

由于乘积

$$A_1 A_2^{-1} = \begin{pmatrix} 0 & \sum_{j=1}^n g_{0j} g^{j1} & \cdots & \sum_{j=1}^n g_{0j} g^{jn} \\ g_{10} & & & \\ \vdots & & \mathbb{O}_n & \\ g_{n0} & & & \end{pmatrix}$$

是一个稀疏矩阵，其幂的计算成本不高。

另一种求逆的迭代方法是构造由 $\tilde{g}_0^{-1} = \mathbb{O}_n$, $\tilde{g}_{n+1}^{-1} = f(\tilde{g}_n^{-1})$ 递归定义的序列 $(\tilde{g}_n^{-1})_{n \geq 0}$ ，其中 $f(M) = A_2^{-1} - M A_1 A_2^{-1}$ 是一个压缩映射。序列 \tilde{g}_n^{-1} 趋向于映射 f 的不动点，该不动点就是逆 \tilde{g}^{-1} 。

收敛条件 级数(14.5.26)和序列 \tilde{g}_n^{-1} 在满足某些条件时收敛。根据附录的G节，所需条件是 $\|A_1 A_2^{-1}\| < 1$ ，其中范数是最大特征值的值。我们将证明 $\|A_1 A_2^{-1}\| = g_{00}$ ，并且一些常见的神经元类别满足条件 $\|A_1 A_2^{-1}\| < 1$ 。

引理14.5.2 设 $a, b \in \mathbb{R}^n$ 是两个向量，使得 $a^T b > 0$ 。那么矩阵的特征值

$$M = \begin{pmatrix} 0 & a_1 & \cdots & a_n \\ b_1 & & & \\ \vdots & & \mathbb{O}_n & \\ b_n & & & \end{pmatrix}$$

对于所有 $j \geq 3$ ，都是 $\lambda_1 = (a^T b)^{1/2}$, $\lambda_2 = -(a^T b)^{1/2}$, $\lambda_j = 0$ 。

证明：见练习14.13.4。 ■

我们令 $M = A_1 A_2^{-1}$ 并证明 $a^T b < 1$ 。我们有

$$\begin{aligned} a^T b &= g_{10} \sum_{j_1} g_{0j_1} g^{j_1 1} + \cdots + g_{n0} \sum_{j_n} g_{0j_n} g^{j_n n} \\ &= \sum_p \sum_{j_r} g_{p0} g_{0j_r} g^{j_r p} = \sum_{j_r} g_{0j_r} \sum_p g_{p0} g^{j_r p} \\ &= g_{00}. \end{aligned}$$

如果激活函数几乎处处满足 $\|\phi'\|_\infty < 1$ (即其最陡斜率几乎在所有地方都小于1)，那么

$$g_{00} = \frac{1}{\sqrt{2\pi}} \int \phi'(|w|\epsilon + b)^2 e^{-\frac{1}{2}\epsilon^2} d\epsilon < \frac{1}{\sqrt{2\pi}} \int e^{-\frac{1}{2}\epsilon^2} d\epsilon = 1.$$

几个激活函数，如 $\tanh(\cdot)$ 或逻辑函数 $\sigma(\cdot)$ ，满足上述性质。因此，在这些情况下， $g_{00} < 1$ 且收敛条件得到满足。

14.6 神经网络的费希尔度量结构

即使在具有前馈神经网络的情况下，我们无法期望得到费希尔度量的显式公式，然而，我们可以通过一种类似于反向传播方法的迭代方法来获得度量结构。计算仍然是在网络中插入的噪声为标准正态分布的假设下进行的， $n \sim \mathcal{N}(0, 1)$ 。即使这个建模假设似乎有局限性，但为了简单起见我们还是考虑它。可以考虑其他类型的噪声，但计算不会那么顺利。

记为 $C(x, y; \theta) = \frac{1}{2}(f_\theta(x) - y)^2$ 。那么关系式(14.3.12)可写为

$$\frac{\partial \ell(x, y; \theta)}{\partial \theta_k} = -\frac{\partial C(x, y; \theta)}{\partial \theta_k}.$$

这等同于 $\nabla_{\theta} \ell(x, y; \theta) = -\nabla_{\theta} C(x, y; \theta)$ 。如果现在，我们将 $C(x, y; \theta)$ 视为二次成本函数（即使在此上下文中它具有不同的意义），我们可以通过第6章中介绍的反向传播方法计算梯度 $\nabla_{\theta} C(x, y; \theta)$ 。参数 θ_k 将分别被 $w_{ij}^{(\ell)}$ 和 $b_j^{(\ell)}$ 取代。按照第6章中的符号和计算方法，我们得到

$$\begin{aligned} \frac{\partial \ell(x, y; \theta)}{\partial w_{ij}^{(\ell)}} &= -\frac{\partial C(x, y; \theta)}{\partial w_{ij}^{(\ell)}} = -\frac{\partial C(x, y; \theta)}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial w_{ij}^{(\ell)}} \\ &= -\delta_j^{(\ell)} x_i^{(\ell-1)} \\ \frac{\partial \ell(x, y; \theta)}{\partial b_j^{(\ell)}} &= -\frac{\partial C(x, y; \theta)}{\partial b_j^{(\ell)}} = -\frac{\partial C(x, y; \theta)}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial b_j^{(\ell)}} \\ &= \delta_j^{(\ell)}, \end{aligned}$$

其中 $\delta_j^{(\ell)}$ 表示 $C(x, y; \theta)$ 相对于信号 $s_j^{(\ell)}$ 的灵敏度，可使用反向传播公式(6.2.22)进行计算

$$\delta_i^{(\ell-1)} = \phi'(s_i^{(\ell-1)}) \sum_{j=1}^{d^{(\ell)}} \delta_j^{(\ell)} w_{ij}^{(\ell)}. \quad (14.6.27)$$

最后一层的增量按以下方式计算

$$\begin{aligned} \delta_j^{(L)} &= \frac{\partial C}{\partial s_j^{(L)}} = \frac{\partial}{\partial s_j^{(L)}} \left(\frac{1}{2} (\phi(s_j^{(L)}) - y)^2 \right) \\ &= \phi'(s_j^{(L)}) (\phi(s_j^{(L)}) - y) \\ &= \phi'(s_j^{(L)}) (f_\theta(x) - y). \end{aligned} \quad (14.6.28)$$

为了克服费舍尔矩阵表达式的书写困难，我们使用符号 $\alpha = (i, j, \ell)$, $\alpha' = (i', j', \ell')$, $X_0 = 1$ 和 $w_{0j}^{(\ell)} = b_j^{(\ell)}$ 。然后 $\theta_\alpha = w_{ij}^{(\ell)}$ ，我们得到度量系数

$$\begin{aligned} g_{\alpha\alpha'} &= \mathbb{E}^{P_{XY}} \left[\frac{\partial \ell(x, y; \theta)}{\partial \theta_\alpha} \frac{\partial \ell(x, y; \theta)}{\partial \theta_{\alpha'}} \right] = \mathbb{E}^{P_{XY}} \left[\frac{\partial \ell(x, y; \theta)}{\partial w_{ij}^{(\ell)}} \frac{\partial \ell(x, y; \theta)}{\partial w_{i'j'}^{(\ell')}} \right] \\ &= \mathbb{E}^{P_{XY}} \left[X_i^{(\ell-1)} X_{i'}^{(\ell'-1)} \delta_j^{(\ell)} \delta_{j'}^{(\ell')} \right], \end{aligned}$$

其索引在以下范围内

$$0 \leq i, i' \leq d^{(\ell-1)}, \quad 1 \leq j, j' \leq d^{(\ell)}, \quad 1 \leq \ell \leq L,$$

其中 L 表示网络的深度。增量 $\delta_j^{(\ell)}$ 和 $\delta_{j'}^{(\ell')}$ 的表达式通过反向传播公式(14.6.27)获得。

如果 $\ell = \ell' = L$ ，使用(14.6.28)，度量系数的表达式变为

$$\begin{aligned} g_{\alpha\alpha'} \Big|_{\ell=\ell'=L} &= \mathbb{E}^{P_{XY}} \left[X_i^{(L-1)} X_{i'}^{(L-1)} \delta_j^{(L)} \delta_{j'}^{(L)} \right] \\ &= \mathbb{E}^{P_{XY}} \left[X_i^{(L-1)} X_{i'}^{(L-1)} \phi'(s_j^{(L)}) \phi'(s_{j'}^{(L)}) (f_\theta(X) - Y)^2 \right]. \end{aligned}$$

如果最后一层的激活函数是线性的， $\phi(x) = x$ ，则表达式更简单，可以按以下方式计算：

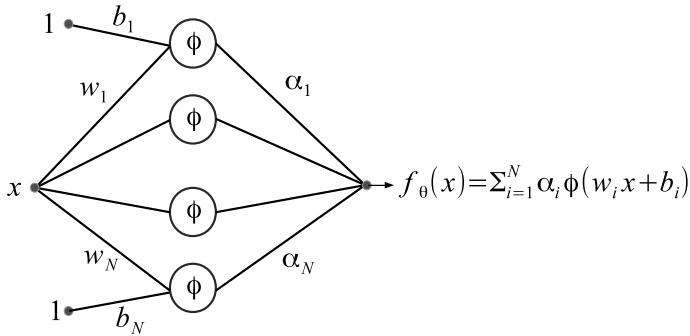


图14.3：具有激活函数 ϕ 和输入-输出映射 $f_\theta(x)$ 的单隐藏层神经网络。输出神经元中的激活是线性的。

$$\begin{aligned}
g_{\alpha\alpha'} \Big|_{\ell=\ell'=L} &= \mathbb{E}^{P_{XY}} \left[X_i^{(L-1)} X_{i'}^{(L-1)} (f_\theta(X) - Y)^2 \right] \\
&= \iint x_i^{(L-1)} x_{i'}^{(L-1)} (f_\theta(x) - y)^2 p(x, y) dx dy \\
&= \int x_i^{(L-1)} x_{i'}^{(L-1)} p(x) \int (f_\theta(x) - y)^2 p(y|x; \theta) dy dx \\
&= \int x_i^{(L-1)} x_{i'}^{(L-1)} p(x) dx \\
&= \mathbb{E}^{P_X} \left[X_i^{(L-1)} X_{i'}^{(L-1)} \right],
\end{aligned}$$

其中我们使用了

$$\int (f_\theta(x) - y)^2 p(y|x; \theta) dy = Var(n) = 1,$$

即，噪声 $n \sim \mathcal{N}(0, 1)$ 的方差为1。层激活 $X_i^{(L-1)}$ 通过前向传播公式迭代计算

$$X_i^{(\ell)} = \phi(W^{(\ell)^T} X^{(\ell-1)} - b_i^{(\ell)}).$$

我们将在以下具体情况下明确计算费希尔度量。

例14.6.1 (单隐藏层网络的费希尔度量) 我们将考虑具有一个隐藏层以及一维输入和输出的前馈神经网络的情况，见图14.3。激活函数

输出神经元中的是线性的，而隐藏层中的由 ϕ 表示。输入-输出映射由下式给出

$$f_{\theta}(x) = \sum_{i=1}^N \alpha_i \phi(w_i x + b_i).$$

直接计算可得到关于参数的偏导数

$$\begin{aligned}\frac{\partial f_{\theta}(x)}{\partial \alpha_j} &= \phi(w_j x + b_j) \\ \frac{\partial f_{\theta}(x)}{\partial w_i} &= \alpha_i x \phi'(w_i x + b_i) \\ \frac{\partial f_{\theta}(x)}{\partial b_i} &= \alpha_i \phi'(w_i x + b_i).\end{aligned}$$

对数似然函数的偏导数为

$$\begin{aligned}\frac{\partial \ell}{\partial \alpha_j} &= -\frac{\partial C}{\partial \alpha_j} = -\frac{1}{2} \frac{\partial}{\partial \alpha_j} (f_{\theta}(x) - y)^2 \\ &= -(f_{\theta}(x) - y) \frac{\partial f_{\theta}(x)}{\partial \alpha_j} = -(f_{\theta}(x) - y) \phi(w_j x + b_j) \\ \frac{\partial \ell}{\partial w_i} &= -\frac{\partial C}{\partial w_i} = -(f_{\theta}(x) - y) \frac{\partial f_{\theta}(x)}{\partial w_i} \\ &= -\alpha_i x (f_{\theta}(x) - y) \phi'(w_i x + b_i) \\ \frac{\partial \ell}{\partial b_i} &= -\frac{\partial C}{\partial b_i} = -\alpha_i (f_{\theta}(x) - y) \phi'(w_i x + b_i).\end{aligned}$$

然后可以按如下方式计算在方向 α_j, α_k 上的费舍尔矩阵的系数

$$\begin{aligned}g_{\alpha_j \alpha_k} &= \mathbb{E}^{P_{XY}} \left[\frac{\partial \ell}{\partial \alpha_j} \frac{\partial \ell}{\partial \alpha_k} \right] = \mathbb{E}^{P_{XY}} \left[\phi(w_j X + b_j) \phi(w_k X + b_k) (f_{\theta}(X) - Y)^2 \right] \\ &= \int \phi(w_j x + b_j) \phi(w_k x + b_k) p(x) \int (f_{\theta}(x) - y)^2 p(y|x; \theta) dy dx \\ &= \int \phi(w_j x + b_j) \phi(w_k x + b_k) p(x) dx \\ &= \mathbb{E}^{P_X} \left[\phi(w_j X + b_j) \phi(w_k X + b_k) \right].\end{aligned}$$

同样地，

$$\begin{aligned}
 g_{w_i w_j} &= \mathbb{E}^{P_{XY}} \left[\frac{\partial \ell}{\partial w_i} \frac{\partial \ell}{\partial w_j} \right] \\
 &= \mathbb{E}^{P_{XY}} \left[\alpha_i \alpha_j X^2 \phi'(w_i X + b_i) \phi'(w_j X + b_j) (f_\theta(X) - Y)^2 \right] \\
 &= \alpha_i \alpha_j \int x^2 \phi'(w_i x + b_i) \phi'(w_j x + b_j) p(x) \underbrace{\int (f_\theta(x) - y)^2 p(y|x; \theta) dy dx}_{=1} \\
 &= \alpha_i \alpha_j \int x^2 \phi'(w_i x + b_i) \phi'(w_j x + b_j) p(x) dx \\
 &= \alpha_i \alpha_j \mathbb{E}^{P_X} \left[X^2 \phi'(w_i X + b_i) \phi'(w_j X + b_j) \right].
 \end{aligned}$$

此外，通过类似的操作我们得到

$$\begin{aligned}
 g_{b_i b_j} &= \mathbb{E}^{P_{XY}} \left[\frac{\partial \ell}{\partial b_i} \frac{\partial \ell}{\partial b_j} \right] = \alpha_i \alpha_j \mathbb{E}^{P_X} \left[\phi'(w_i X + b_i) \phi'(w_j X + b_j) \right] \\
 g_{\alpha_j w_i} &= \alpha_i \mathbb{E}^{P_X} \left[X \phi(w_j X + b_j) \phi'(w_i X + b_i) \right] \\
 g_{\alpha_j b_i} &= \alpha_i \mathbb{E}^{P_X} \left[\phi(w_j X + b_j) \phi'(w_i X + b_i) \right] \\
 g_{w_i b_k} &= \alpha_i \alpha_k \mathbb{E}^{P_X} \left[X \phi'(w_i X + b_i) \phi'(w_k X + b_k) \right].
 \end{aligned}$$

我们注意到，上述所有系数取决于输入密度 $p(x)$ 、隐藏层中神经元的激活函数 $\phi(x)$ 以及网络的参数。

14.7 自然梯度

为了最小化依赖于权重和偏差的成本函数 $C(w, b)$ ，采用了梯度下降法。这涉及朝着负梯度 $-(\nabla_w C, \nabla_b C)$ 的方向迈出一步 $\eta > 0$ 。该梯度是使用由欧几里得度量 δ_{ij} 诱导的参数空间 Θ 的平坦几何来计算的。本节的想法是在坐标空间 Θ 上应用梯度下降法，但使用相对于费希尔度量计算的梯度。这种方法是可取的，因为它能更快地收敛到最优参数值 θ^* ，因为如在阿马里[6]中所指出的，最陡方向不是由欧几里得梯度捕获，而是由其自然梯度捕获。本节将介绍这个概念并展示其主要性质。

设 $(\mathcal{S} = \{p(x, y; \theta); \theta \in \Theta\}, g)$ 为与给定神经网络相关联的神经流形。

参数空间 Θ 可以如下赋予从 (\mathcal{S}, g) 诱导的度量 $g(\theta)$ 。如果 $\left\{ \frac{\partial}{\partial \theta_i} \right\}_{1 \leq i \leq N}$ 表示 Θ 上的坐标向量，那么在这个基础上定义度量就足够了，如下所示

$$g\left(\frac{\partial}{\partial \theta_i}, \frac{\partial}{\partial \theta_j}\right) = g_{ij}(\theta).$$

因此，参数空间连同 $g(\theta)$ 成为黎曼流形 $(\Theta, g(\theta))$ 。

考虑在参数空间 $f : \Theta \rightarrow \mathbb{R}$ 上定义的一个光滑函数（特别地，这可以是任何代价函数）。 f 的多方向变化由其梯度描述。欧几里得梯度是向量场

$$\nabla_{Eu} f = \sum_{k=1}^N \frac{\partial f}{\partial \theta_k} e_k = \left(\frac{\partial f}{\partial \theta_1}, \dots, \frac{\partial f}{\partial \theta_N} \right)^T,$$

其中 $\{e_k\}_k$ 是 \mathbb{R}^N 中的自然标准正交基。这种类型的梯度在第4章介绍的经典梯度下降法中非常有用。然而，在 Θ 赋予 Fisher 度量 $g(\theta)$ 的情况下， f 的梯度必须相应地进行调整。

f 的自然梯度是相对于 Fisher 度量 $g(\theta)$ 所取的梯度。这可以用坐标向量 $\left\{ \frac{\partial}{\partial \theta_k} \right\}_k$ 的基写成

$$\nabla_g f = \sum_{k=1}^N (\nabla_g f)^k \frac{\partial}{\partial \theta_k}, \quad (14.7.29)$$

其分量由 $(\nabla_g f)^k = \sum_{j=1}^N g^{kj}(\theta) \frac{\partial f}{\partial \theta_j}$ 给出，其中 $g^{kj}(\theta)$ 是逆矩阵的系数， $g^{-1}(\theta)$ 。关于欧几里得梯度的自然梯度 (14.7.29) 的一个等价公式是

$$\nabla_g f = g(\theta)^{-1} \nabla_{Eu} f. \quad (14.7.30)$$

作为一个应用，关于费希尔度量的欧几里得梯度和自然梯度的大小由以下关系给出

$$\begin{aligned} \|\nabla_g f\|_g^2 &= (\nabla_{Eu} f)^T g^{-1}(\theta) \nabla_{Eu} f \\ \|\nabla_{Eu} f\|_g^2 &= (\nabla_{Eu} f)^T g(\theta) \nabla_{Eu} f, \end{aligned}$$

见练习 14.13.12。公式 (14.7.30) 中与矩阵 $g(\theta)^{-1}$ 相乘会旋转并缩放欧几里得梯度 $\nabla_{Eu} f$ 以得到自然梯度 $\nabla_g f$ 。由于梯度 $\nabla_g f$ 和 $\nabla_{Eu} f$ 在相同的值处消失

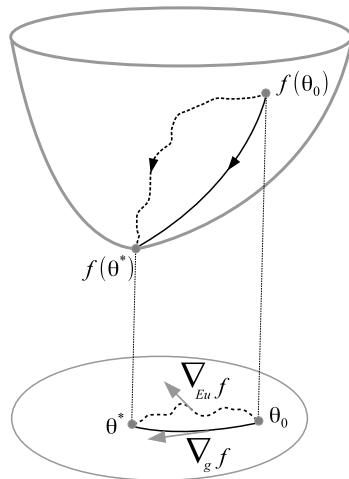


图14.4：自然梯度下降法比欧几里得梯度下降法更快地到达最小值。

θ , 见练习14.13.12, 由此可知梯度下降法的两种变体从相同的初始点 $f(\theta_0)$ 出发, 会到达相同的最小值 $f(\theta^*)$, 但路径不同, 见图14.4。在梯度下降法中用自然梯度代替欧几里得梯度可以提高该方法的效率。下一节将讨论这个概念的应用。

注记14.7.1值得注意的是, 对数似然函数的欧几里得梯度可用于将费希尔矩阵 $g(\theta) = (g_{ij}(\theta))_{ij}$ 表示为

$$g(\theta) = \mathbb{E}^{P_{XY}} [\nabla_{Eu} \ell(X, Y; \theta) (\nabla_{Eu} \ell(X, Y; \theta))^T] = \mathbb{E}^{P_{XY}} [\nabla_{Eu} \ell (\nabla_{Eu} \ell)^T].$$

14.8 自然梯度学习算法

我们在第4章中看到, 在欧几里得空间上定义的代价函数 $C(\theta)$ 的最陡方向由其欧几里得梯度 $\nabla_{Eu} C$ 给出。对于在诸如黎曼流形 (Θ, g) 这样的弯曲空间上定义的代价函数 $C(\theta)$, 这个结果是无效的。在这种情况下, 最陡方向是在自然梯度方向 $\nabla_g C$ 上实现的。本节讨论在神经学习中使用自然黎曼梯度的效果, 它基于Amari等人的工作, 见[5]、[6]、[99]、[130]。

最速下降方向 我们首先考虑一个单位向量场, $V = \sum V^i(\theta) \frac{\partial}{\partial \theta_i}$, 其 $\|V\|_g = 1$ 与参数空间 (Θ, g) 相切, 并研究代价函数 $C(\theta)$ 在 V 方向上的变化。 $C(\theta)$ 相对于 V 的变化率用 $V(C)$ 或 $\frac{\partial C}{\partial V}$ 表示, 且等于

$$\frac{\partial C}{\partial V} = V(C) = \sum_i V^i \frac{\partial C}{\partial \theta_i} = \langle V, \nabla_{Eu} C \rangle. \quad (14.8.31)$$

根据练习14.13.11的 (c) 部分, 我们有

$$\langle V, \nabla_{Eu} C \rangle = g(V, \nabla_g C).$$

结合柯西 - 施瓦茨不等式与方程(14.8.31)可得

$$\frac{\partial C}{\partial V} = g(V, \nabla_g C) \leq \|V\|_g \|\nabla_g C\|_g = \|\nabla_g C\|_g,$$

当 V 和 $\nabla_g C$ 成比例时等式成立。因此, 速率 $\frac{\partial C}{\partial V}$ 在 $V = \nabla_g C / \|\nabla_g C\|_g$ 方向上最高。

因此, 代价函数 $C(\theta)$ 的最速下降方向是负自然梯度, 其由下式给出

$$-\nabla_g C(\theta) = -g^{-1}(\theta) \nabla_{Eu} C(\theta).$$

有关涉及拉格朗日乘数的证明变体, 请参见练习14.13.9。

值得注意的是, 在特殊情况下, 当 (Θ, g) 是欧几里得空间 $(\mathbb{R}^n, \delta_{ij})$ 时, $g^{-1} = \mathbb{I}_n$, 因此我们得到了欧几里得梯度的方向。

[6]中引入的自然梯度学习算法通过以下规则更新参数 θ_n

$$\theta_{n+1} = \theta_n - \eta_n \nabla_g C(\theta_n), \quad (14.8.32)$$

学习率 $\eta_n \rightarrow 0$ 以某种方式如 $n \rightarrow \infty$ 那样。

在[130]中有人提出, 用 $\nabla_g C$ 替换 $\nabla_{Eu} C$ 有助于消除迭代陷入平稳状态的情况。自然梯度学习比普通梯度下降更有效的原因还有其他一些。在讨论这些原因之前, 我们先回顾一下两种类型的学习算法, 批量学习和在线学习。

批量学习 在这种情况下, 一批中的所有训练示例都用于获得最优权重向量。如果训练集是 $\{(x_1, z_1), \dots, (x_n, z_n)\}$, 那么代价函数取决于所有样本, 如下所示

$$C(\theta) = \frac{1}{2n} \sum_{j=1}^n |z_j - f_\theta(x_j)|^2.$$

如果数据是从相同的训练分布 $p_{XZ}(\theta)$ 中采样得到的，那么代价也可以写成期望的形式

$$C(\theta) = \frac{1}{2} E^{P_{XZ}(\theta)} [(Z - f_\theta(X))^2].$$

在这种情况下，常规梯度下降法通过在如下方向上采取步长来描述

$$\theta_{n+1} = \theta_n - \eta_n \nabla C(\theta_n), \quad n = 0, 1, 2, \dots$$

在线学习 这种方法在观察时仅对每个示例使用一次，假设示例是逐个给出的。代价函数采用简单形式

$$C(x_n, z_n, \theta_n) = \frac{1}{2} |z_n - f_{\theta_n}(x_n)|^2.$$

[3]和[107]中提出的梯度下降法采用如下规则

$$\theta_{n+1} = \theta_n - \eta_n \nabla C(x_n, y_n, \theta_n).$$

一般来说，在批量学习的情况下， θ_n 收敛到代价函数的真正最小值 θ^* 比在线学习的情况更准确。然而，如果学习率 η_n 以某种方式收敛到 0，并且梯度 $\nabla C(x_n, y_n, \theta_n)$ 被自然梯度所取代，那么在线学习就会渐近地与批量学习一样有效。为了进一步阐述这个想法，我们首先将引入一些关于估计器的概念。

估计量的类型 设 $\mathcal{S} = \{p_\theta; \theta \in \Theta\}$ 为一族密度函数， $\hat{\theta} = \hat{\theta}(x_1, \dots, x_n)$ 是基于从分布 p_θ 中抽取的数据 $\{x_1, \dots, x_n\}$ 对参数 θ 的一个估计量。那么：

- 如果 $\mathbb{E}^{\hat{P}_X} [\hat{\theta}] = \theta$ ，则称 $\hat{\theta}$ 是无偏的。
- 如果 $\hat{\theta}_n = \hat{\theta}(x_1, \dots, x_n)$ 依概率收敛于 $\hat{\theta}_n \rightarrow \theta$ ，即对于任意 $\epsilon > 0$ ，有 $\lim_{n \rightarrow \infty} P(|\hat{\theta}_n - \theta| < \epsilon) = 1$ ，则称 $\hat{\theta}_n = \hat{\theta}(x_1, \dots, x_n)$ 是相合的。关于依概率收敛的定义见附录 D.6.1 节。
- 如果 $\hat{\theta}$ 是无偏的且达到克拉美 - 罗不等式的下界，则称 $\hat{\theta}$ 是费希尔有效的。

$$Cov(\hat{\theta}) \geq g^{-1}(\theta), \quad \forall \theta \in \Theta,$$

即，它是一个最小方差无偏估计量。²

² 如果 A 和 B 是两个方阵，当 $A - B$ 是半正定的，即其所有特征值都是非负的时，我们记为 $A \geq B$ 。

- 如果 $\hat{\theta}_n$ 在渐近意义下达到克拉美 - 罗界中的等式，即渐近地达到等式，那么它是渐近费希尔有效的。

$$\lim_{n \rightarrow \infty} \text{Cov}(\hat{\theta}_n) = g^{-1}(\theta), \quad \forall \theta \in \Theta.$$

例如，在一个正确设定的模型中，一个著名的结果表明最大似然估计量 $\hat{\theta}_{MLE,N}$ ，依赖于 N 个独立样本 x_j 。

$$\hat{\theta}_{MLE,N} = \arg \min_{\theta} \frac{1}{N} \sum_{j=1}^N \ln p_{\theta}(x_j) = \arg \max_{\theta} \prod_{j=1}^N p_{\theta}(x_j),$$

它既是一致的（ $(\hat{\theta}_{MLE,N} \rightarrow \theta, N \rightarrow \infty)$ 依概率收敛），又是渐近有效的（当样本量 n 趋于无穷时达到克拉美 - 罗下界）。其他例子见练习14.13.13和14.13.14。

在这种情况下，克拉美 - 罗不等式可以写成

$$\mathbb{E}[(\hat{\theta}_{MLE,N} - \theta)(\hat{\theta}_{MLE,N} - \theta)^T] \geq \frac{1}{N} g^{-1}(\theta),$$

见练习14.13.16。最大似然估计量渐近费希尔有效性的事实可以写成

$$\lim_{N \rightarrow \infty} N \mathbb{E}[(\hat{\theta}_{MLE,N} - \theta)(\hat{\theta}_{MLE,N} - \theta)^T] = g^{-1}(\theta),$$

另见练习14.13.16。

在线学习中的费舍尔效率 由于在线学习中，训练示例仅在出现时使用一次，所以在线学习的渐近性能应该不如最优批量过程，因为在最优批量过程中，所有示例会被重复使用几个轮次。下一个结果表明，只要满足一些额外条件，实际上效率是成立的。下一个结果可在阿马里[6]中找到：

定理14.8.1 设代价函数为对数似然函数， $C(x, z; \theta) = \ln p(x, z; \theta)$ 。设 θ^* 表示数据采样所来自分布的参数真值，即 $(x_n, z_n) \sim p(x, z; \theta^*)$ 。那么在线学习的自然梯度学习规则

$$\hat{\theta}_{n+1} = \hat{\theta}_n - \frac{1}{n} \nabla_g C(x_n, z_n; \hat{\theta}_n)$$

产生一个估计器 $\hat{\theta}_n$ ，它是渐近费舍尔有效的，即

$$\lim_{n \rightarrow \infty} n \mathbb{E}[(\hat{\theta}_n - \theta^*)(\hat{\theta}_n - \theta^*)^T] = g(\theta^*)^{-1}.$$

证明思路是考虑协方差矩阵

$$V_n = \mathbb{E}[(\hat{\theta}_n - \theta^*)(\hat{\theta}_n - \theta^*)^T]$$

并证明它验证了渐近关系

$$V_n = \frac{1}{n}g(\theta^*)^{-1} + O\left(\frac{1}{n^2}\right).$$

这是通过从在线学习关系的两边减去 θ^* 得到的

$$\hat{\theta}_{n+1} = \hat{\theta}_n - \frac{1}{n}g^{-1}(\hat{\theta}_n)\nabla_{\theta}\ell(x_n, z_n; \hat{\theta}_n)$$

然后对两边的平方取期望。计算涉及对数似然函数导数的线性近似

$$\begin{aligned} \nabla_{\theta}\ell(x_n, z_n; \hat{\theta}_n) &= \nabla_{\theta}\ell(x_n, z_n; \theta^*) + (\hat{\theta}_n - \theta^*)^T \nabla_{\theta}\nabla_{\theta}\ell(x_n, z_n; \theta^*) \\ &\quad + O(\|\hat{\theta}_n - \theta^*\|^2), \end{aligned}$$

以及一些其他关系

$$\begin{aligned} \mathbb{E}[\nabla_{\theta}\ell(x, y; \theta^*)] &= 0 \\ \mathbb{E}[\nabla_{\theta}\nabla_{\theta}\ell(x, y; \theta^*)] &= -g(\theta^*) \\ g(\theta_n) &= g(\theta^*) + O\left(\frac{1}{n}\right). \end{aligned}$$

自适应实现自然梯度算法要求费希尔度量的逆矩阵 $g(\theta)^{-1}$ 是已知的，而这一事实很难以封闭形式出现。文献[8]中给出了一种直接估计逆矩阵 $g(\theta)^{-1}$ 并应用自然梯度在线学习的自适应方法

$$\begin{aligned} \hat{g}_{n+1}^{-1} &= (1 + \epsilon_n)\hat{g}_n^{-1} - \epsilon_n \hat{g}_n^{-1} \nabla_{Eu} f_n (\nabla_{Eu} f_n)^T \hat{g}_n^{-1} \\ \theta_{n+1} &= \theta_n - \eta_n \hat{g}_n^{-1} \nabla_{Eu} \ell(x_n, z_n; \theta_n), \end{aligned}$$

其中 $f_n = f_{\theta}(x_n)$ 是输入-输出映射， $g_n = g(\theta_n)$ ，并且 $\epsilon_n > 0$ 是一个小学习率。

14.9 对数似然与度量

本节阐述了对数似然函数变化量的大小与输入-输出映射的欧几里得梯度之间的关系。

如果神经网络的参数从 θ 到 $\theta' = \theta + d\theta$ 进行无穷小的扰动，那么输入 - 输出映射从 $f_\theta(x)$ 变为 $f_{\theta'}(x)$ ，其中

$$f_{\theta'}(x) = f_\theta(x) + \sum_{k=1}^N \frac{\partial f_\theta(x)}{\partial \theta_k} d\theta_k = f_\theta(x) + \langle \nabla_{E_u} f, d\theta \rangle, \quad (14.9.33)$$

具有无穷小扰动向量 $d\theta = (d\theta_1, \dots, d\theta_N)^T$ 。

在参数空间 Θ 中，相对于度量 g ，无穷小分离点 θ 和 θ' 之间距离的平方由二次型给出

$$\|d\theta\|_g^2 = \|\theta' - \theta\|_g^2 = (d\theta)^T g(\theta) d\theta = \sum_{i,j} g_{ij}(\theta) d\theta_i d\theta_j.$$

我们注意到它与欧几里得距离的相似性， $\|d\theta\|_{E_u}^2 = \sum_j (d\theta_j)^2$ 。参数的微小变化会对对数似然函数的变化产生影响。这由下一个结果给出。

命题14.9.1 (a) 对数似然函数的微小变化为

$$d\ell(x, y; \theta) = (y - f_\theta(x)) df_\theta(x);$$

(b) 其模的平方由下式给出

$$\|d\ell(x, y; \theta)\|_g^2 = (y - f_\theta(x))^2 (\nabla_{E_u} f_\theta(x))^T g(\theta) \nabla_{E_u} f_\theta(x) O(\|d\theta\|^2).$$

证明：(a) 对于 $\theta' = \theta + d\theta$ ，对数似然函数 $\ell(\theta) = \ell(x, y; \theta)$ 的变化为

$$\begin{aligned} \ell(\theta') - \ell(\theta) &= \sum_k \frac{\partial \ell(\theta)}{\partial \theta_k} d\theta_k = \sum_k (y - f_\theta(x)) \frac{\partial f_\theta(x)}{\partial \theta_k} d\theta_k \\ &= (y - f_\theta(x)) \sum_k \frac{\partial f_\theta(x)}{\partial \theta_k} d\theta_k = (y - f_\theta(x))(f_{\theta'}(x) - f_\theta(x)), \end{aligned}$$

我们使用了公式(14.3.12)和(14.9.33)。现在代入 $d\ell(x, y; \theta) = \ell(\theta') - \ell(\theta)$ 和 $df_\theta(x) = f_{\theta'}(x) - f_\theta(x)$ ，我们得到了所需的公式。

(b) 在来自第 (a) 部分的关系中，取 g -度量下的模的平方，我们有

$$\|d\ell(x, y; \theta)\|_g^2 = (y - f_\theta(x))^2 \|df_\theta(x)\|_g^2.$$

右侧的第二个因子可以评估为

$$\begin{aligned}\|df_\theta(x)\|_g^2 &= (df_\theta(x))^T g(\theta) df_\theta(x) = \sum_k \frac{\partial f_\theta}{\partial \theta_k} (d\theta_k)^T g(\theta) \sum_j \frac{\partial f_\theta}{\partial \theta_j} d\theta_j \\ &= \sum_{j,k} \frac{\partial f_\theta}{\partial \theta_k} \frac{\partial f_\theta}{\partial \theta_j} (d\theta_k)^T g(\theta) d\theta_j = \sum_{j,k} \frac{\partial f_\theta}{\partial \theta_k} \frac{\partial f_\theta}{\partial \theta_j} g_{jk}(\theta) \|d\theta_k\| \|d\theta_j\| \\ &= g(\nabla_{Eu} f_\theta, \nabla_{Eu} f_\theta) O(\|d\theta\|^2) = \|\nabla_{Eu} f_\theta\|_g^2 O(\|d\theta\|^2).\end{aligned}$$

我们使用了公式 $(d\theta_k)^T g(\theta) d\theta_j = g_{jk}(\theta) O(\|d\theta\|^2)$ ，它由线性代数关系 $e_k^T A e_j = A_{jk}$ 推导得出，其中 A 是一个矩阵， A_{jk} 是第 (j, k) 个元素， $\{e_k\}$ 是一个标准正交基；在我们的例子中， $A = g(\theta)$ 和 $e_k = d\theta_k / \|d\theta_k\|$ 。按照练习14.13.12的第 (b) 部分那样表示 $\|\nabla_{Eu} f_\theta\|_g^2$ ，我们得到了所需的关系。

14.10 与库尔贝克-莱布勒散度的关系

已知在与神经网络相关的神经流形 S 上，两个概率密度 $p(x, y; \theta)$ 和 $p(x, y; \theta')$ 之间的接近程度可以用库尔贝克-莱布勒散度来衡量。本节展示了这种接近程度与参数空间 $(\Theta, g(\theta))$ 中 θ 和 $\theta' = \theta + d\theta$ 之间的黎曼距离的关系，见图14.5。

以下结果很快就会有用。

引理14.10.1 如果 $\ell(x, y; \theta)$ 表示对数似然函数，我们有

$$\mathbb{E}^{P_{XY}(\theta)} \left[\frac{\partial}{\partial \theta_j} \ell(X, Y; \theta) \right] = 0.$$

证明：利用期望和对数似然函数的定义，我们有

$$\begin{aligned}\mathbb{E}^{P_{XY}(\theta)} \left[\frac{\partial}{\partial \theta_j} \ell(X, Y; \theta) \right] &= \iint \frac{\partial}{\partial \theta_j} \ell(x, y; \theta) p(x, y; \theta) dx dy \\ &= \iint \frac{\partial}{\partial \theta_j} p(x, y; \theta) dx dy = \frac{\partial}{\partial \theta_j} \underbrace{\iint p(x, y; \theta) dx dy}_{=1} \\ &= 0.\end{aligned}$$

前面的结果可用于将费希尔信息写成协方差形式：

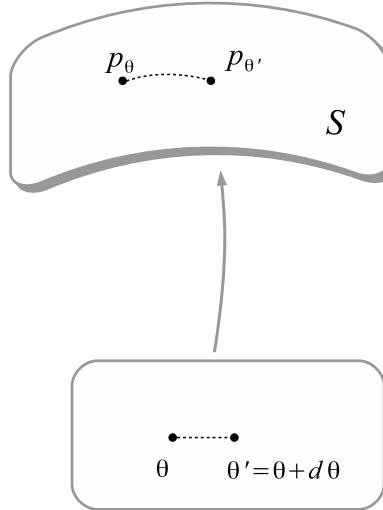


图14.5： θ 与 θ' 之间的黎曼距离与 p_θ 和 $p_{\theta'}$ 的库尔贝克-莱布勒散度有关。

推论14.10.2 费希尔矩阵由协方差矩阵给出

$$g_{ij}(\theta) = \text{Cov}(\partial_{\theta_i} \ell, \partial_{\theta_j} \ell), \quad (14.10.34)$$

其中 $\partial_{\theta_i} \ell = \frac{\partial}{\partial \theta_i} \ell(X, Y; \theta)$ 。

证明：利用引理14.10.1和协方差定义，我们有

$$\begin{aligned} \text{Cov}(\partial_{\theta_i} \ell, \partial_{\theta_j} \ell) &= \mathbb{E}^{P_{XY}(\theta)} \left[\frac{\partial}{\partial \theta_i} \ell(X, Y; \theta) \frac{\partial}{\partial \theta_j} \ell(X, Y; \theta) \right] \\ &\quad - \underbrace{\mathbb{E}^{P_{XY}(\theta)} \left[\frac{\partial}{\partial \theta_i} \ell(X, Y; \theta) \right]}_{=0} \underbrace{\mathbb{E}^{P_{XY}(\theta)} \left[\frac{\partial}{\partial \theta_j} \ell(X, Y; \theta) \right]}_{=0} \\ &= g_{ij}(\theta). \end{aligned}$$

■

为简单起见，记 $p_\theta = p(x, y; \theta)$ ，并考虑 $\theta' = \theta + d\theta$ 。下一个结果表明，用库尔贝克-莱布勒散度衡量的 p_θ 和 $p_{\theta'}$ 之间的接近程度是空间 (Θ, g) 中 θ 和 θ' 之间黎曼距离平方的一半。

命题14.10.3 库尔贝克-莱布勒散度的线性近似和二次近似由下式给出：

- (a) $D_{KL}(p_\theta \parallel p_{\theta'}) = O(\|d\theta\|^2)$;
- (b) $D_{KL}(p_\theta \parallel p_{\theta'}) = \frac{1}{2} \|d\theta\|_g^2 + O(\|d\theta\|^3)$.

证明：(a) 设由 $F_\theta(u) = D_{KL}(p_\theta \parallel p_{\theta+u})$ 给出的 $F_\theta : \mathbb{R}^N \rightarrow [0, \infty)$ 。根据散度性质，

$F_\theta(0) = D_{KL}(p_\theta \parallel p_\theta) = 0$ 。为了计算偏导数，我们考虑在 e_j 方向上的变化。利用导数作为极限的定义，我们有

$$\begin{aligned}\frac{\partial}{\partial u_j} F_\theta(0) &= \lim_{t \rightarrow 0} \frac{F_\theta(te_j) - F_\theta(0)}{t} = \lim_{t \rightarrow 0} \frac{D_{KL}(p_\theta \parallel p_{\theta+te_j})}{t} \\ &= \lim_{t \rightarrow 0} \frac{1}{t} \mathbb{E}^{p_\theta} [\ell(\theta) - \ell(\theta + te_j)] = -\lim_{t \rightarrow 0} \mathbb{E}^{p_\theta} \left[\frac{\ell(\theta + te_j) - \ell(\theta)}{t} \right] \\ &= -\mathbb{E}^{p_\theta} \left[\lim_{t \rightarrow 0} \frac{\ell(\theta + te_j) - \ell(\theta)}{t} \right] = -\mathbb{E}^{p_\theta} \left[\frac{\partial}{\partial \theta_j} \ell(\theta) \right] = 0,\end{aligned}$$

其中最后一个等式由引理14.10.1给出。

由于线性近似右边的前两项

$$F_\theta(u) = F_\theta(0) + \sum_j \frac{\partial}{\partial u_j} F_\theta(0) du_j + O(\|du\|^2)$$

为零，我们得到 $F_\theta(u) = O(\|du\|^2)$, which is equivalent to

$$D_{KL}(p_\theta \parallel p_{\theta'}) = O(\|d\theta\|^2).$$

(b) 类似地，取二阶偏导数我们得到

$$\begin{aligned}\frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} F_\theta(u) &= \frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} D_{KL}(p_\theta \parallel p_{\theta+u}) \\ &= \frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} \mathbb{E}^{p_\theta} [\ell(\theta) - \ell(\theta + u)] \\ &= -\mathbb{E}^{p_\theta} \left[\frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} \ell(\theta + u) \right].\end{aligned}$$

利用由(14.2.8)给出的费希尔度量系数的定义，我们有

$$\begin{aligned}\frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} F_\theta(u) \Big|_{u=0} &= -\mathbb{E}^{p_\theta} \left[\frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} \ell(\theta + u) \right] \Big|_{u=0} \\ &= -\mathbb{E}^{p_\theta} \left[\frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} \ell(\theta) \right] = g_{ij}(\theta).\end{aligned}$$

我们假设导数与期望算子可交换，这一事实对于高斯型密度总是成立的。

二次近似

$$\begin{aligned} F_\theta(u) &= F_\theta(0) + \sum_j \frac{\partial}{\partial u_j} F_\theta(0) du_j + \frac{1}{2} \sum_{j,k} \frac{\partial}{\partial u_i} \frac{\partial}{\partial u_j} F_\theta(0) du_i du_j + O(\|du\|^3) \\ &= \frac{1}{2} \sum_{j,k} \frac{\partial}{\partial u_i} g_{ij}(\theta) du_i du_j + O(\|du\|^3) \end{aligned}$$

可写为

$$\begin{aligned} D_{KL}(p_\theta || p_{\theta'}) &= \frac{1}{2} \sum_{j,k} \frac{\partial}{\partial u_i} g_{ij}(\theta) d\theta_i d\theta_j + O(\|d\theta\|^3) \\ &= \frac{1}{2} (d\theta)^T g(\theta) d\theta + O(\|d\theta\|^3) = \frac{1}{2} \|d\theta\|_g^2 + O(\|d\theta\|^3). \end{aligned}$$

■

在黎曼流形 (Θ, g) 上定义的一些微分几何概念可以用神经流形 \mathcal{S} 上的统计概念来表达。我们将针对曲线的能量和长度来进行此事。

设 $\theta : [a, b] \rightarrow \Theta$ 是参数空间 Θ 中的一条可微曲线，赋予费希尔度量 $g(\theta)$ 。曲线的能量是沿曲线的动能密度的积分

$$\mathcal{E}(\theta) = \frac{1}{2} \int_a^b \|\dot{\theta}(t)\|_g^2 dt.$$

我们将根据概率密度 $p_{\theta(t)}$ 的库尔贝克 - 莱布勒散度对能量进行定量表征。注意，赋值 $t \rightarrow p_{\theta(t)}$ 是神经流形 \mathcal{S} 上的一条曲线。

我们考虑一个等距划分 $a = t_0 < t_1 < \dots < t_n = b$ ，其中 $\Delta t = t_{k+1} - t_k = (b - a)/n$ ，并记为 $\theta_k = \theta(t_k)$ 。对于 n 很大时，点 θ_k 和 θ_{k+1} 之间的黎曼距离可以由命题 14.10.3 的第 (b) 部分表示为

$$\frac{1}{2} \|\theta_{k+1} - \theta_k\|_g^2 = D_{KL}(p_{\theta_k} || p_{\theta_{k+1}}).$$

利用这个，我们可以将能量评估为

$$\begin{aligned}\mathcal{E}(\theta) &= \frac{1}{2} \int_a^b \|\dot{\theta}(t)\|_g^2 dt = \lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{2} \frac{\|\theta_{k+1} - \theta_k\|_g^2}{(\Delta t)^2} \Delta t \\ &= \lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{\Delta t} D_{KL}(p_{\theta_k} || p_{\theta_{k+1}}) \\ &= \lim_{n \rightarrow \infty} \frac{n}{b-a} \sum_{k=1}^n D_{KL}(p_{\theta_k} || p_{\theta_{k+1}}).\end{aligned}$$

最近在[21]中提出了一个关于椭圆曲线变形的类似问题。

曲线 $\theta(t)$ 的长度是通过沿着曲线对速度 $\|\dot{\theta}(t)\|_g$ 关于时间参数 t 进行积分得到的，如下所示

$$L(\theta) = \int_a^b \|\dot{\theta}(t)\| dt.$$

长度可以用库尔贝克 - 莱布勒散度表示如下

$$\begin{aligned}L(\theta) &= \int_a^b \|\dot{\theta}(t)\| dt = \lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{\|\theta_{k+1} - \theta_k\|_g}{\Delta t} \Delta t \\ &= \lim_{n \rightarrow \infty} \sum_{k=1}^n \|\theta_{k+1} - \theta_k\|_g = \sqrt{2} \lim_{n \rightarrow \infty} \sum_{k=1}^n D_{KL}(p_{\theta_k} || p_{\theta_{k+1}})^{1/2},\end{aligned}$$

其中最后一个等式使用了命题14.10.3。

14.11 模拟退火方法

在前面的章节中，我们向神经网络的输出添加了高斯噪声 $n \sim \mathcal{N}(0, 1)$ ，见(14.1.1)，然后我们通过信息几何技术来处理这个问题。我们已经看到学习是通过自然梯度算法进行的，该算法涉及费希尔度量的逆。在本节中，我们将利用前面的结果与模拟退火方法建立联系。

应用于具有输出 $Y = f_\theta(X)$ 的深度神经网络的常规梯度下降方法，在大多数情况下，由于 f_θ 的高度非线性，只能得到代价函数的局部最小值。为了获得全局最小值，将使用模拟退火方法的一个变体。为此，我们将考虑一个可调噪声 $n_T \sim \mathcal{N}(0, T^2)$ ，其中 T 起到温度的作用。

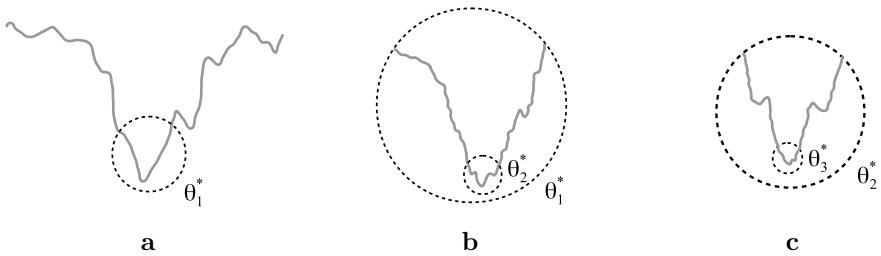


图14.6：退火方法：a. 对于高温 T_1 ，最优参数 θ_1^* 位于全局最小值附近。b. 将温度降至 T_2 ，我们通过新的最优点 θ_2^* 得到了对全局最小值更精确的近似。c. 继续降低温度，我们得到了对全局最小值越来越精确的近似。

启发式思想是从一个较大的温度 T 开始优化成本函数，然后根据特定的时间表将其降至零。如果时间表是 $T_1 > T_2 > \dots > T_N > 0$ ，我们用 θ_1^* 表示通过自然梯度学习方法获得的与温度 T_1 对应的最优参数。对于温度 T_2 的下一个最优参数值 θ_2^* 的搜索从 θ_1^* 开始，见图14.6。一般来说，与温度 T_k 对应的最优点 θ_k^* 是通过自然梯度下降获得的，该自然梯度下降从初始值 θ_{k-1}^* 开始搜索。对应于最低温度 T_N 的最后一个最优点 θ_N^* 最接近成本函数的真正全局最小值。

14.12 总结

本章介绍神经网络的信息几何。此类网络存在噪声，其输出由权重和偏差参数化的概率密度表征。因此，每个分布可视为空间中的一个点，当赋予费希尔度量时，该空间成为黎曼流形。这就是与给定网络相关联的神经流形。

此处涵盖的主题主要涉及神经流形的内蕴几何，它由费希尔信息度量定义。针对几种特定类型的网络明确计算了此度量，并将其应用于自然梯度学习算法，该算法是黎曼流形梯度下降算法的改进版本。在神经网络中插入噪声就如同提高热力学系统的温度。

模拟退火方法的一种变体与自然梯度下降方法相结合，以获得成本函数的全局最小值。

信息几何有几个重要的主题在本章中未涉及，例如神经流形的外在几何，它描述了网络相对于更大的概率密度流形的相对几何。感兴趣的读者可以在Amari [4]或Calin等人[22]中找到诸如嵌入曲率、对偶连接等主题。关于信息几何在机器学习中的更多应用，请读者参考[7]。

14.13 练习

练习14.13.1 设 $\{v_1, \dots, v_n\}$ 是 \mathbb{R}^n 中的一个标准正交基（即一组 n 向量，使得 $v_i^T v_j = \delta_{ij}$ ）。

- (a) 如果 G 是一个 $n \times n$ 对称矩阵，使得对于所有 $1 \leq j \leq n$ 都有 $v_j^T G v_j = 0$ ，证明 $G = \mathbb{O}_n$ (n 维零矩阵)。
- (b) 如果 A 和 B 是两个 $n \times n$ 对称矩阵，使得对于所有 $1 \leq j \leq n$ 都有 $v_j^T A v_j = v_j^T B v_j$ ，证明 $A = B$ 。

练习14.13.2 一个 2×2 矩阵如果具有以下形式，则称其为角度为 ϕ 的旋转矩阵

$$R = \begin{pmatrix} u_1 & u_2 \\ v_1 & v_2 \end{pmatrix} = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix}.$$

我们注意到 u 和 v 是正交向量，并且 $\det R = 1$ 。

- (a) 令 $X = (X_1, X_2) \sim \mathcal{N}(0, \mathbb{I}_2)$ ，并考虑如上的旋转矩阵 R 。

证明 $u^T X$ 和 $v^T X$ 是独立的，其中 $u^T = (u_1, u_2)$ 和

$$v^T = (v_1, v_2).$$

- (b) $u^T X$ 和 $v^T X$ 的分布是什么？

- (c) 证明：若 u 和 v 是平面中的两个正交向量，则存在 $\phi \in [0, 2\pi)$ 使得 $u^T = (\cos \phi, \sin \phi)$ 和 $v^T = (-\sin \phi, \cos \phi)$ 。

练习14.13.3 设 $X = (X_1, X_2)^T$ ，其中 X_1, X_2 为独立随机变量。

- (a) 考虑 \mathbb{R}^2 中的正交向量 u 和 w 。证明 $Y_1 = u^T X$ 和 $Y_2 = w^T X$ 也相互独立。

- (b) 证明仅当 u 和 w 仅正交时（向量的大小无关紧要），(a)部分成立。

Exercise 14.13.4 Prove Lemma 14.5.2

练习14.13.5 (a) 求具有激活函数 $\phi(x) = \sigma(x)$ 的Sigmoid神经元的Fisher度量，其中 $\sigma(x)$ 表示逻辑函数。

(b) 证明不等式

$$0 \leq g_{00} \leq \frac{1}{4^2}, \quad g_{0k}^2 \leq \frac{1}{4^4} \mathbb{E}[X_k^2], \quad g_{ij}^2 \leq \frac{1}{4^4} \mathbb{E}[X_j^2 X_k^2].$$

(c) 陈述并证明在激活函数 ϕ 的一般情况下，(b) 中给出的不等式的一个变体。

练习14.13.6 求具有输入 $X = (X_1, \dots, X_n)$ 的神经元的Fisher度量系数，其中 X_i 独立同分布， $X_i \sim$ 服从 $[0, 1]$ 上的均匀分布。

练习14.13.7 求具有激活函数 $\phi(x) = x$ 的单隐藏层神经网络的Fisher度量系数。用网络参数和输入变量 X 的前两个矩表示结果。

练习14.13.8 当输入为 $X \sim \mathcal{N}(0, 1)$ 时，求出具有激活函数 $\phi(x)$ 的单隐藏层神经网络的费希尔度量系数。

练习14.13.9 考虑损失函数 $L : \Theta \rightarrow \mathbb{R}$ ，切空间 $T_\theta \Theta$ 中的向量 v ，其中 $\|v\|_g^2 = 1$ ，以及学习步长 $\eta > 0$ 。参数在 v 方向上的微小变化，幅度为 η ，可以写成 $dw = \eta v$ ，因此线性近似变为

$$L(\theta + d\theta) = L(\theta) + \eta \nabla_{Eu} L(\theta)^T v.$$

我们需要找到方向 v ，使得 $L(\theta + d\theta)$ 最小化。为此，我们考虑拉格朗日泛函

$$F(v, \lambda) = \nabla L(\theta)^T v - \lambda \|v\|_g^2.$$

(a) 证明变分方程 $\frac{\partial F}{\partial v_i} = 0$ 蕴含 $\nabla_{Eu} L(w) = 2\lambda g(\theta)v$ 。

(b) 证明 $v = \nabla_g L(\theta) / \|\nabla_g L(\theta)\|_g$ 。

练习14.13.10 设 $p_{X_1}(x_1; \theta)$ 和 $p_{X_2}(x_2; \theta)$ 分别为随机变量 X_1 和 X_2 的概率密度。那么

$$g(X_1, X_2; \theta) = g(X_1; \theta) + g(X_2 | X_1; \theta) = g(X_2; \theta) + g(X_1 | X_2; \theta),$$

其中 $g(X_1 | X_2; \theta)$ 是由条件概率密度 $p_{X_1 | X_2}(x_1 | x_2; \theta)$ 定义的费希尔信息（即，给定 X_2 时，包含在 X_1 中关于 θ 的信息量）。

练习14.13.11 设 $X = \sum_{k=1}^N X^k \frac{\partial}{\partial \theta_k}$ 为 Θ 上的一个向量场。证明：

$$(a) \langle \nabla_{Eu} f, X \rangle_{Eu} = \sum_{k=1}^N X^k \frac{\partial f}{\partial \theta_k}. \quad —$$

$$(b) g(\nabla_g f, X) = \sum_{k=1}^N X^k \frac{\partial f}{\partial \theta_k}. \quad —$$

$$(c) \langle \nabla_{Eu} f, X \rangle_{Eu} = g(\nabla_g f, X).$$

练习14.13.12 证明：

$$(a) \|\nabla_g f\|_g^2 = (\nabla_{Eu} f)^T g^{-1}(\theta) \nabla_{Eu} f;$$

$$(b) \|\nabla_{Eu} f\|_g^2 = (\nabla_{Eu} f)^T g(\theta) \nabla_{Eu} f$$

(c) $\nabla_{Eu} f$ 和 $\nabla_g f$ 在相同的点处消失。

练习14.13.13 考虑一维随机变量 $X \sim \mathcal{N}(\mu, 1)$ ，且 $\hat{\mu}(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i$

是通过对变量 X 进行 n 次独立观测得到的均值 μ 的一个估计量。

(a) 证明 $\hat{\mu}(x_1, \dots, x_n)$ 是均值 μ 的无偏估计量；

(b) 求 X 的费希尔信息；

(c) 证明 $\hat{\mu}(x_1, \dots, x_n)$ 是费希尔有效的。

练习14.13.14 设 $X \sim \text{Pois}(\lambda)$ 是参数为 λ 的泊松分布离散随机变量，即

$$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}, \quad k = 0, 1, 2, \dots$$

构造参数 λ 的费希尔有效估计量。

练习14.13.15 考虑独立同分布的随机变量 $X_1, \dots, X_N \sim X$ ，其 $X \sim \mathcal{N}(\mu, 1)$ ，并考虑它们的平均值

$$\bar{X} = \frac{1}{N} (X_1 + \dots + X_N).$$

证明 \bar{X} 中包含的关于 μ 的信息是每个单个变量关于 μ 的信息之和，即

$$I(\bar{X}) = NI(X), \quad \text{其中 } I \text{ 表示一维随机变量的费希尔信息}.$$

练习14.13.16 (a) 设 X_1 和 X_2 为两个独立的随机变量，其概率密度为 $p_{X_1}(x_1; \theta), p_{X_2}(x_2; \theta)$ ，该概率密度依赖于参数 θ 。 X_i 中包含的关于 θ 的信息由费希尔信息 $g(X_i; \theta)$ 给出。证明 (X_1, X_2) 对中包含的费希尔信息是各个费希尔信息之和

$$g(X_1, X_2; \theta) = g(X_1; \theta) + g(X_2; \theta).$$

- (b) 陈述并证明对 n 个独立随机变量的推广。
- (c) 证明 N 中包含的关于独立同分布随机变量 $X_1, \dots, X_N \sim X$ 相对于 θ 的 Fisher 信息矩阵的逆是 $\frac{1}{N}g^{-1}(X; \theta)$ 。
- (d) 利用 (c) 部分解释为什么基于 N 个独立同分布随机变量 X_1, \dots, X_N 的 θ 的渐近有效估计量 $\hat{\theta}(N)$ 的定义如下

$$\lim_{N \rightarrow \infty} N\mathbb{E}[(\hat{\theta}(N) - \theta)(\hat{\theta}(N) - \theta)^T] = g^{-1}(\theta),$$

其中 $g(\theta)$ 是对应于其中一个随机变量的 Fisher 信息矩阵。

第五部分 其他架构



第15章池化

池化是一种机器学习技术，它提供输入的摘要，选择一些基本的局部特征，如最大值、最小值、平均值等。

它还充当信息压缩器；在离散情况下，它会将输入的维度按一定因子降低。因此它在分类问题中很有用。

池化的思想是考虑函数定义域的一个划分，并用该集合上函数的“最具代表性”值替换每个划分元素上的函数。这个过程会得到一个简单函数。池化的二维变体用于卷积神经网络的构建。

15.1 连续函数的逼近

本节讨论在紧集上连续函数的背景下应用的最大池化、最小池化和平均池化技术。为了简单起见，我们仅针对一维紧区间 $[a, b]$ 的情况证明结果，而读者可以轻松地将结果扩展到多维。

最大池化 设 $f : [a, b] \rightarrow \mathbb{R}$ 为连续函数，并考虑区间 $[a, b]$ 的等距划分

$$a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b.$$

划分大小 $\frac{b-a}{n}$ 称为步长。用 $M_i = \max_{[x_{i-1}, x_i]} f(x)$ 表示并考虑简

单函数 $S_n(x) = \sum_{i=1}^n M_i 1_{[x_{i-1}, x_i)}(x)$ 。该过程

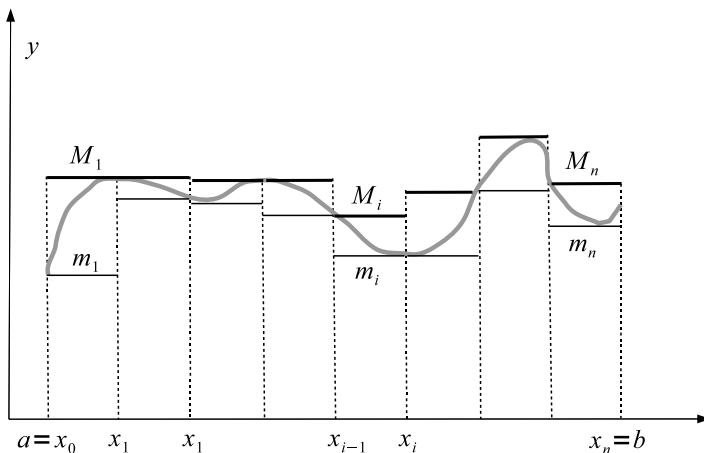


图15.1：当 n 增加时，差值 $M_i - m_i$ 趋近于零减小。

O. Calin, 《深度学习架构》，施普林格数据科学系列，用简单函数 $S_n(x)$ 逼近函数 $f(x)$ 被称为最大池化。更多细节可在Zhou和Chellappa [132]中找到。

作为一种变体，我们可以考虑 $m_i = \min_{[x_{i-1}, x_i]} f(x)$ 并定义简单函数 $s_n(x) = \sum_{i=1}^n m_i 1_{[x_{i-1}, x_i]}(x)$ 。最小池化是通过阶跃函数 $s_n(x)$ 逼近函数 $f(x)$ 的过程。我们注意到以下双重不等式成立：

$$s_n(x) \leq f(x) \leq S_n(x), \quad \forall n \geq 1.$$

平均池化 考虑函数 f 在区间 $[x_{i-1}, x_i]$ 上的平均值，由 $\mu_i = \frac{1}{x_i - x_{i-1}} \int_{x_{i-1}}^{x_i} f(u) du$ 给出。对每个区间上函数的平均值进行池化，我们得到函数 $A_n(x) = \sum_{i=1}^n \mu_i 1_{[x_{i-1}, x_i]}(x)$ 。

下一个结果表明，所有先前的池化函数都是 f 的“良好近似”。

定理15.1.1 设 $f : [a, b] \rightarrow \mathbb{R}$ 为连续函数。那么所有三个函数序列， $(S_n)_n$, $(s_n)_n$ 和 $(A_n)_n$ ，当 $n \rightarrow \infty$ 时在 $[a, b]$ 上一致收敛于 f 。这意味着， $\forall \epsilon > 0$ ，存在 $N \geq 1$ 使得

$$\begin{aligned} |S_n(x) - f(x)| < \epsilon, \quad |s_n(x) - f(x)| < \epsilon, \quad |A_n(x) - f(x)| < \epsilon, \\ \forall x \in [a, b], \forall n \geq N. \end{aligned}$$

证明：构造序列

$$u_n(x) = S_n(x) - s_n(x) = \sum_{i=1}^n (M_i - m_i) 1_{[x_{i-1}, x_i)}(x),$$

其满足以下性质，见图15.1：

- (i) $u_n(x) \geq 0$
- (ii) $u_{n+1}(x) \leq u_n(x)$ ，对于任意 $n \geq 1$ ；
- (iii) $u_n(x) \rightarrow 0$ ，当 $n \rightarrow \infty$ 时，对于任意固定的 x 。

步骤1. 我们证明 $(u_n)_n$ 在 $[a, b]$ 上一致收敛于0。

设 $\epsilon > 0$ 为任意固定值。由 f 在 $[a, b]$ 上的一致连续性，存在 $N \geq 1$ ，使得若 $|x - x'| < \frac{b-a}{N}$ ，则 $|f(x) - f(x')| < \epsilon$ 。现在，在划分的每个区间中存在值 $\xi_i, \xi'_i \in [x_{i-1}, x_i]$ ，使得 $M_i = f(\xi_i)$ 且 $m_i = f(\xi'_i)$ 。由于 $|\xi - \xi'| < \frac{b-a}{N}$ ，则 $M_i - m_i = |f(\xi_i) - f(\xi'_i)| < \epsilon$ 。这意味着

$$\sum_{i=1}^n (M_i - m_i) 1_{[x_{i-1}, x_i)}(x) < \epsilon, \quad \forall x \in [a, b].$$

这意味着 $|u_n(x)| < \epsilon, \forall x \in [a, b]$ ，以及 $\forall n > N$ ，即 $(u_n)_n$ 一致收敛于0。

步骤2。我们证明 $(S_n)_n$ 在 $[a, b]$ 上一致收敛于 f 。

由于 $s_n \leq f$ ，则以下不等式成立：

$$S_n - f \leq (S_n - s_n) + (s_n - f) \leq S_n - s_n = u_n,$$

对于任意的 $n \geq 1$ 。设 $\epsilon > 0$ 为任意固定值。结合步骤1与上述不等式，我们有

$$|S_n(x) - f(x)| \leq |u_n(x)| < \epsilon, \quad n \geq 1, \forall x \in [a, b].$$

这意味着当 $n \rightarrow \infty$ 时， $(S_n)_n$ 一致收敛于 f 。

步骤3。我们证明 $(s_n)_n$ 在 $[a, b]$ 上一致收敛于 f 。

这与步骤2类似。由于 $S_n \geq f$ ，那么

$$f - s_n = (f - S_n) + (S_n - s_n) \leq S_n - s_n = u_n,$$

对于任何 $n \geq 1$ 。设 $\epsilon > 0$ 为任意固定值。利用步骤1和前面的不等式，我们有

$$|f(x) - s_n(x)| \leq |u_n(x)| < \epsilon, \quad n \geq 1, \forall x \in [a, b].$$

这意味着当 $n \rightarrow \infty$ 时, $(s_n)_n$ 一致收敛于 f 。

步骤4。我们证明 $(A_n)_n$ 在 $[a, b]$ 上一致收敛于 f 。

设 $\epsilon > 0$ 为任意固定值。根据中值定理的积分形式, 存在一个 $x_i^* \in [x_{i-1}, x_i]$ 使得 $\mu_i = f(x_i^*)$ 。因此,

$$m_i \leq \mu_i \leq M_i.$$

乘以指示函数 $1_{[x_{i-1}, x_i]}(x)$ 并对 i 求和得到

$$s_n(x) \leq A_n(x) \leq S_n(x).$$

这意味着 $|A_n(x) - s_n(x)| \leq u_n(x)$ 和 $|S_n(x) - A_n(x)| \leq u_n$ 。使用步骤1可得, $|A_n(x) - s_n(x)| \rightarrow 0$ 和 $|S_n(x) - A_n(x)| \rightarrow 0$ 一致成立, 因为 $n \rightarrow \infty$ 。现在, 三角不等式给出

$$\begin{aligned} |A_n(x) - f(x)| &= |A_n(x) - f(x) + S_n(x) - S_n(x)| \\ &\leq |S_n(x) - A_n(x)| + |S_n(x) - f(x)| \\ &\leq \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon, \end{aligned}$$

这里我们使用了步骤2。 ■

注记15.1.2 池化可以扩展到多维情形, 其中 $f : K \rightarrow \mathbb{R}$ 是定义在紧集 $K \subset \mathbb{R}^n$ 上的连续函数。考虑紧集 K 的覆盖 $K = \bigcup_{i=1}^n \bar{A}_i$, 其中 $A_i \cap A_j = \emptyset$ 对于 $i \neq j$,

A_i 为不相交的开集且 \bar{A}_i 表示 A_i ¹ 的闭包。我们对最大值 $M_i = \max_{\bar{A}_i} f(x)$ 进行

池化并考虑逼近 $S_n(x) = \sum_{i=1}^n M_i 1_{A_i}(x)$ 。如果

$$\max_{1 \leq i \leq n} \sup_{x, y \in A_i} |x - y| \rightarrow 0$$

如 $n \rightarrow \infty$, 那么与前一个证明类似的证明表明 S_n 在 K 上一致收敛于 f 。

15.2 平移不变性

在本节中, 我们将证明最大池化和最小池化的局部平移不变性。对于由 $(T_a \circ f)(x) = f(x - a)$ 定义的平移算子, 考虑其表示法 T_a , 其中 f 是任意实变量函数, $a \in \mathbb{R}$ 也是。我们还用 $\mathcal{P}(f)$ 表示与给定划分相关联的 f 的最小或最大池化函数。

¹ 开集 A 的闭包是集合 A 及其边界。

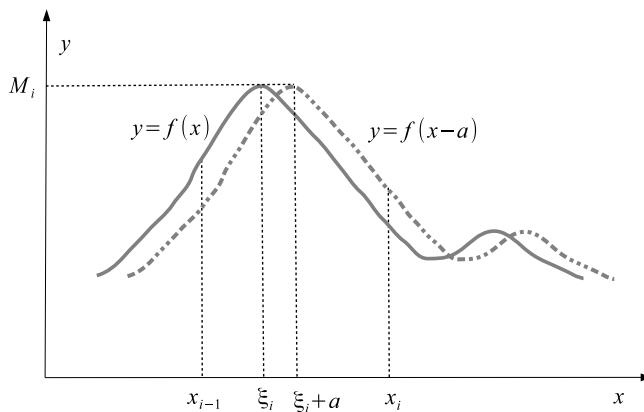


图15.2： f 和 $T_a \circ f$ 的最大池化。

命题15.2.1 设 $f : \mathbb{R} \rightarrow \mathbb{R}$ 为连续函数。存在 \mathbb{R} 的一个划分，使得

$$\mathcal{P}(T_a \circ f) = \mathcal{P}(f),$$

对于 a 的任何足够小的值。

证明：我们将在最大池化的情况下进行证明。最小池化的情况可以类似地处理。证明思路是，在小的平移下，最大值不会离开划分区间，见图15.2。

选择一个有限划分 $[x_i, x_{i+1}]$ $0 \leq i \leq N - 1$ ，使得限制 $f|_{[x_i, x_{i+1}]}$ 的最大值 ξ_i 位于开区间 (x_i, x_{i+1}) 内。存在一个 $\eta > 0$ 使得 $x_i + \eta < \xi_i < x_{i+1} - \eta$ 。然后选择 $a \in \mathbb{R}$ 使得 $|a| < \eta$ 。

由于 $T_a \circ f$ 的图像是通过将 f 的图像水平移动量 a 得到的，那么最大值不会离开这些区间，并且我们有

$$M_i(f) = \max_{[x_i, x_{i+1}]} f(x) = \max_{[x_i, x_{i+1}]} f(x - a) = \max_{[x_i, x_{i+1}]} (T_a \circ f)(x) = M_i(T_a \circ f).$$

因此，函数 f 和 $T_a \circ f$ 将具有相同的最大池化函数。 ■

注记15.2.1 (i) 不变性性质可扩展到多个维度，证明中只需做少许改动。

(ii) 上述性质在输入有小变化时为池化提供了稳定性。

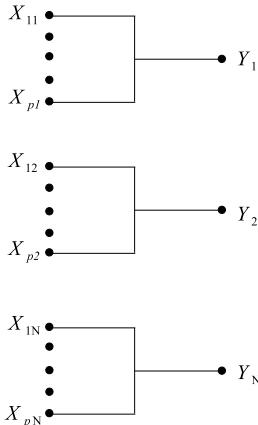


图15.3：具有 $Y_j = \max \{X_{1j}, \dots, X_{pj}\}$, $1 \leq j \leq N$ 的最大池化层。

15.3信息方法

看待池化的另一种方式是研究其对信息内容的影响。本节讨论最大池化的情况，但平均池化也有类似的结果。

考虑 n 个随机变量， X_1, X_2, \dots, X_n ，并令 $Y = \max \{X_1, \dots, X_n\}$ 。令 $\mathfrak{S}(X_i)$ 为由 X_i 生成的西格玛代数，并且

$$\mathfrak{S}(X) = \mathfrak{S}(X_1, \dots, X_n) = \mathfrak{S}(X_i) \vee \dots \vee \mathfrak{S}(X_n)$$

为由所有 X_i 生成的信息域。对于任何 $b \in \mathbb{R}$ ，我们有

$$\begin{aligned} Y^{-1}(-\infty, b] &= \{\omega; Y(\omega) \leq b\} = \{\omega; X_i(\omega) \leq b, \forall i = 1, \dots, n\} \\ &= \bigcap_{i=1}^n \{\omega; X_i(\omega) \leq b\} = \bigcap_{i=1}^n X_i^{-1}(-\infty, b] \in \bigcap_{i=1}^n \mathfrak{S}(X_i). \end{aligned}$$

因此， $\mathfrak{S}(Y) \subset \bigcap_{i=1}^n \mathfrak{S}(X_i)$ ，也就是说， n 个随机变量最大值的信息域包含在每个变量生成的信息域中。接下来，我们将把这个结果应用于神经网络。
定义15.3.1 如果前馈神经网络的 ℓ 层满足以下条件，我们就称其为池化层：

(i) $(\ell - 1)$ 层被划分为 N 类神经元的一个划分；

(ii) 属于同一类别的第 $(\ell - 1)$ 层的所有神经元被映射到第 ℓ 层的同一个神经元中，该神经元的激活值是它们相应的最大值；

(iii) 第 ℓ 层中的神经元数量为 $d^{(\ell)} = N$ ，其中 N 是划分的类别数量。

大致来说，池化层用该类中的最大神经元值替换一层的每个划分类别。在图15.3中，第 $(\ell - 1)$ 层包含值为 $X_{ij}, 1 \leq i \leq p, 1 \leq j \leq N$ 的神经元，分为 N 类

$$\{X_{11}, \dots, X_{p1}\}, \{X_{12}, \dots, X_{p2}\}, \dots, \{X_{1N}, \dots, X_{pN}\},$$

每类有 p 个神经元。每类被池化为其最大值

$$Y_j = \max\{X_{1j}, \dots, X_{pj}\}, \quad 1 \leq j \leq N.$$

根据先前的计算，池化层中每个神经元的信息满足包含关系

$$\mathfrak{S}(Y_j) \subset \bigcap_{i=1}^p \mathfrak{S}(X_{ij}). \quad (15.3.1)$$

池化层生成的信息由下式给出

$$\mathfrak{S}(Y) = \mathfrak{S}(Y_1, \dots, Y_N) = \bigvee_{j=1}^N \mathfrak{S}(Y_j) = \mathfrak{S}\left[\bigcup_{j=1}^N \mathfrak{S}(Y_j)\right].$$

包含关系 (15.3.1) 意味着

$$\mathfrak{S}(Y) \subset \mathfrak{S}\left[\bigcup_{j=1}^N \bigcap_{i=1}^p \mathfrak{S}(X_{ij})\right]. \quad (15.3.2)$$

使用附录A部分的公式 (b')，可得

$$\bigcup_{j=1}^N \bigcap_{i=1}^p \mathfrak{S}(X_{ij}) = \bigcup_{j=1}^N \bigcap_{i_r=1}^p \mathfrak{S}(X_{i_rj}) = \bigcap_{i_1, \dots, i_p} \left(\bigcup_{j=1}^N \mathfrak{S}(X_{i_rj}) \right).$$

然后，借助练习15.6.1的(a)部分，(15.3.2) 变为

$$\mathfrak{S}(Y) \subset \mathfrak{S}\left[\bigcap_{i_1, \dots, i_p} \left(\bigcup_{j=1}^N \mathfrak{S}(X_{i_rj}) \right) \right] \subset \bigcap_{i_1, \dots, i_p} \mathfrak{S}\left[\left(\bigcup_{j=1}^N \mathfrak{S}(X_{i_rj}) \right) \right],$$

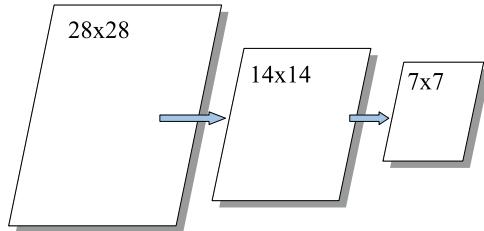


图15.4：应用于MNIST图像的两个最大池化层。

也可以写成

$$\mathfrak{S}(Y) \subset \bigcap_{i_1, \dots, i_p} \bigvee_{j=1}^N \mathfrak{S}(X_{i_r j}) = \bigcap_{i_1, \dots, i_p} \mathfrak{S}(X_{i_1 1}, \dots, X_{i_N N}).$$

这种关系有以下解释。从第一类中任选一个神经元，比如 $X_{i_1 1}$ 。这个神经元产生的信息是 $\mathfrak{S}(X_{i_1 1})$ 。如果对每一类都这样做，那么这些任意从各类中选取的神经元产生的信息就是 $\mathfrak{S}(X_{i_1 1}, \dots, X_{i_N N})$ 。前面的包含关系表明，池化层的信息 $\mathfrak{S}(Y)$ ，包含在任何一个信息集 $\mathfrak{S}(X_{i_1 1}, \dots, X_{i_N N})$ 中，而不管神经元的任意选择。

15.4 池化与分类

当需要降低输入的维度以匹配分类类别的数量时，通常会使用池化。例如，在MNIST数据的情况下，每个输入图像有 $28 \times 28 = 784$ 个像素，而有10个分类类别（数字 $0, 1, \dots, 9$ ）。例如，这可以被划分为 14×14 个正方形，每个正方形有 2×2 个像素。从每个 2×2 正方形中，我们只保留强度最大的像素。结果，我们得到一个 14×14 像素的图像，这是第一次池化的结果。第二次池化将 14×14 图像划分为 7×7 个正方形，每个正方形有 2×2 个像素；同样，我们从这些正方形中的每一个中只保留强度最大的像素，见图15.4。这是一个信息以不可逆的方式被丢弃的过程。

池化通常用作卷积的辅助操作。卷积层对输入信号进行滤波，去除噪声，而池化层则选择滤波后信号特征的大致摘要，见图15.5。卷积操作和卷积网络将在下一章中更详细地讨论。

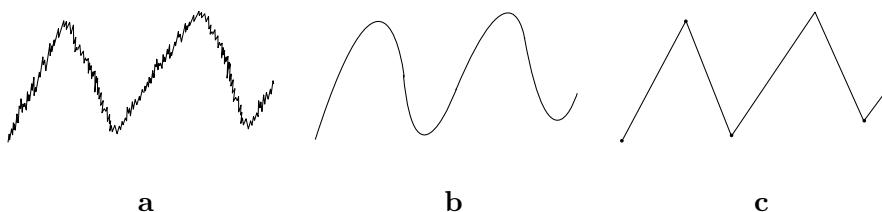


图15.5：卷积和池化的效果：a. 原始且有噪声的输入信号；b. 通过卷积获得的平滑信号；c. 通过选择最大值和最小值获得的池化信号。

15.5总结

池化是一种机器学习技术，通过该技术输入的维度会按一定因子降低。它也可以被视为一种信息压缩器，通过它信息以不可逆的方式被丢弃。池化与卷积一起用于分类问题。

15.6练习

练习15.6.1 设 (\mathcal{C}_i) 为可测集的集合。证明

$$(a) \mathfrak{S} \left(\bigcap_i \mathcal{C}_i \right) \subset \bigcap_i \mathfrak{S} (\mathcal{C}_i);$$

$$(b) \mathfrak{S} \left(\bigcup_i \mathcal{C}_i \right) \supset \bigcup_i \mathfrak{S} (\mathcal{C}_i).$$

练习15.6.2 有 $N = 2^n$ 名参与者参加一场国际象棋比赛。参与者两两比赛。在每一轮中，每对中的胜者与另一对中的胜者比赛。最终的胜者在第 n 轮后产生。根据最大池化过程解释这个程序。

练习15.6.3 (a) 在最小池化的情况下证明命题15.2.1。(b) 阐述并证明二维函数的命题15.2.1的一个版本。

在神经网络中，不建议连续放置池化层，因为它们的组合可以写成一个单一的池化层。下一个练习涉及更精确的陈述：

练习15.6.4 (a) 假设神经网络的所有层都是最大池化层。证明网络的最终输出是最大输入。

(b) 假设神经网络的所有层都是平均池化层。证明网络的最终输出是输入的平均值。

(c) 如果神经网络的所有层都是最小池化层，证明网络的最终输出是输入的最小值。

练习15.6.5 在神经网络中，一个最大池化层后面跟着一个最小池化层。

(a) 证明如果我们交换这些池化层的顺序，网络输出会发生变化。

(b) 如果将最小池化层替换为平均池化层，结果仍然成立吗？



第16章 卷积网络

卷积神经网络（CNN）是具有共享权重和稀疏交互的前馈神经网络，也就是说，大多数权重都等于零。由于其参数数量较少，卷积网络比任何大小相似的全连接层网络更易于训练，对其性能的负面影响也很小。

CNN在图像处理方面的出色性能归因于其对二维网格状拓扑结构的适应性。CNN是受生物学启发的人工智能，其设计基于神经科学原理，并成功应用于模式识别，见LeCun等人[74]。

与几乎任何其他前馈神经网络一样，CNN使用反向传播算法的一个版本进行训练，并使用ReLU作为激活函数。在本章中，我们将基于局部感受野、内核、卷积和特征图的概念来讨论CNN的架构。

16.1 离散一维信号

离散一维信号可以用实数的双无穷序列来描述

$$y = [\dots, y_{-2}, y_{-1}, y_0, y_1, y_2, \dots],$$

其中 y_k 表示在时间 t_k 测量的信号幅度（图16.1）。信号 y 被称为：

- 有限信号，如果 $\max_k |y_k| < \infty$ ，即 $\|y\|_\infty < \infty$ ；
- L^1 -有限信号，如果 $\sum_{k=-\infty}^{\infty} |y_k| < \infty$ ，即 $\|y\|_1 < \infty$ ；
- 有限能量信号，如果 $\sum_{k=-\infty}^{\infty} |y_k|^2 < \infty$ ，即 $\|y\|_2 < \infty$ 。

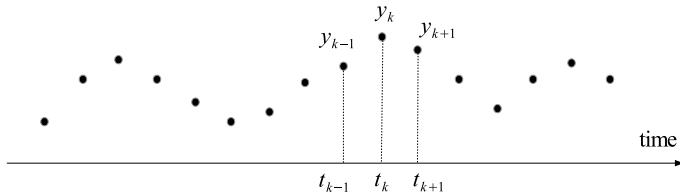


图16.1：离散一维信号。

- 若存在 $N \geq 1$ ，使得对于任意 $|k| > N$ 都有 $y_k = 0$ ，则为紧支集信号。

值得注意的是，具有紧支集的有限能量信号是 L^1 -有限的。这可由柯西不等式推出

$$\left(\sum_{k=-N}^N |y_k| \right)^2 \leq 2N \sum_{k=-N}^N |y_k|^2.$$

信号可以通过滤波进行处理。这涉及信号与核或滤波器之间的卷积运算，如下文我们将定义的那样。核（滤波器）是权重的紧支集序列

$$w = [\dots, w_{-2}, w_{-1}, w_0, w_1, w_2, \dots].$$

y 和 w 之间的卷积信号是信号

$$z = [\dots, z_{-2}, z_{-1}, z_0, z_1, z_2, \dots],$$

记为 $z = y * w$ ，并定义为

$$z_j = \sum_{k=-\infty}^{\infty} y_{j+k} w_k. \quad (16.1.1)$$

上述无穷和是有意义的，因为 w 只有有限个非零元素。卷积信号 z 的每个分量都是初始信号 y 各分量的加权和。卷积的作用是使用某个给定的权重系统对信号进行平均。等效地，滑动滤波器 w ，然后乘以 y 并求和，就得到了滤波后的信号 z 。

值得注意的是，公式(16.1.1)在信号处理中被称为互相关。

例16.1.1（移动平均信号）如果核由下式给出

$$w = [\dots, 0, 0, w_0 = \frac{1}{2}, w_1 = \frac{1}{2}, 0, 0, \dots]$$

那么 $z = y * w$ 是一个移动平均信号。在这种情况下，序列的每一项都被两个连续项的算术平均值所取代，即

$$z_j = \frac{1}{2}(y_{j-1} + y_j)$$

移位细节可从下表推断：

	y_{-2}	y_{-1}	y_0	y_1	y_2	y_3	y_4	z_j
$j = 0$	0	0	$1/2$	$1/2$	0	0	0	$z_0 = (y_0 + y_1)/2$
$j = 1$	0	0	0	$1/2$	$1/2$	0	0	$z_1 = (y_1 + y_2)/2$
$j = 2$	0	0	0	0	$1/2$	$1/2$	0	$z_2 = (y_2 + y_3)/2$

类似地，如果滤波器由 $w = [\dots, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0, \dots]$ 给出，我们得到一个移动平均信号，它是三个连续项的平均值

$$z_j = \frac{1}{3}(y_{j-1} + y_j + y_{j+1}).$$

16.2 连续一维信号

在这种情况下，信号幅度作为时间 $y = y(t)$ 的连续函数给出。先前的定义可以适用于连续情况如下：

- 如果 $\|y\|_\infty < \infty$ ，则为有限信号；
- L^1 - 有限信号，若 $\|y\|_1 = \int_{\mathbb{R}} |y(t)| dt < \infty$ ；
- 若 $\|y\|_2 = \left(\int_{\mathbb{R}} |y(t)|^2 dt \right)^{1/2} < \infty$ ，则为有限能量信号；
- 若存在 $u \geq 0$ 使得对于任意 $|t| > u$ 都有 $y(t) = 0$ ，则为紧支集信号。

在这种情况下，滤波器是具有紧支集的连续函数， $w = w(t)$ 。这意味着对于足够大的 $|t|$ ， $w(t) = 0$ 。卷积信号 $z = y * w$ 由下式定义

$$z(t) = (y * w)(t) = \int_{\mathbb{R}} y(u + t)w(u) du = \int_{\mathbb{R}} y(v)w(v - t) dv.$$

此公式是连续信号 y 和 w 之间互相关的连续版本。值得注意的是，数学文献中的卷积定义与前一个公式相比使用了相反的符号。然而，这就是该概念在神经网络中变得有用的方式。

与离散情况类似，任何具有紧支集的有限能量信号是 L^1 有限的这一说法在连续情况下也成立。此外，任何经过滤波的 L^1 有限信号也是 L^1 有限的。

这可由以下估计得出

$$\begin{aligned}\|z\|_1 &= |z(t)| dt = \int \left| \int_{\mathbb{R}} w(u) y(u+t) du \right| dt \\ &\leq \int \int |w(u) y(u+t)| du dt = \int |w(u)| \int |y(u+t)| dt du \\ &= \int |w(u)| \int |y(r)| dr du = \int |w(u)| \|y\|_1 du = \|y\|_1 \|w\|_1,\end{aligned}$$

在此我们使用富比尼定理颠倒了积分顺序。

16.3 离散二维信号

离散二维信号是一个无限矩阵 $y = [y_{ij}]_{i,j}$ ，其中 y_{ij} 是 (i, j) 像素的激活值。这样，任何黑白图像都可被视为二维信号。如果 (i, j) 像素是黑色的，那么激活值是 $y_{ij} = 1$ ；如果像素是白色的，激活值是 $y_{ij} = 0$ ；任何其他灰度值是介于0和1之间的一个数。

信号 y 被称为：

- L^1 - 有限的，如果 $\sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} |y_{jk}| < \infty$ ；
- 如果 $\sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} |y_{jk}|^2 < \infty$ ，则为有限能量信号；
- 如果存在 $N \geq 1$ ，使得对于任何 $|j| > N$ 和 $|k| > N$ ， $y_{jk} = 0$ ，则为紧支集信号。

在这种情况下，一个核是一个具有紧支集的信号 $w = [w_{ij}]$ 。信号 $y = [y_{jk}]$ 可以与核 $w = [w_{ij}]$ 卷积得到信号 $z = y * w$ ，如下所示

$$z_{ij} = \sum_{k=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} y_{i+k, j+r} w_{k,r}.$$

在二维情况下，输出 z_{ij} 也被称为特征图，因为它应该包含一些对于核 w 来说具有特征性的图像特征。

注记16.3.1 对于一个连续的二维信号 $y : \mathbb{R}^2 \rightarrow \mathbb{R}$ ，它与一个连续滤波器 $w : \mathbb{R}^2 \rightarrow \mathbb{R}$ 的卷积可通过积分公式得到

$$\begin{aligned}z(t_1, t_2) &= \iint_{\mathbb{R}^2} y(u_1 + t_1, u_2 + t_2) w(u_1, u_2) du_1 du_2 \\ &= \iint_{\mathbb{R}^2} y(v_1, v_2) w(v_1 - t_1, v_2 - t_2) dv_1 dv_2.\end{aligned}$$

读者应该不难将 L^1 -有限能量信号和紧支集信号的定义从离散情形扩展到连续情形。

示例16.3.2 (二维移动平均) 考虑具有 2×2 支持的特征图

$$w = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1/4 & 1/4 & 0 \\ 0 & 1/4 & 1/4 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

通过对四个相邻激活值求平均得到卷积信号

$$z_{ij} = \frac{1}{4}(y_{ij} + y_{i,j+1} + y_{i+1,j} + y_{i+1,j+1}).$$

在下面我们将用 \mathcal{C} 表示卷积算子, 所以 $\mathcal{C}(y)$ 是具有输入 y 的卷积层的输出。在前面的符号表示中我们有 $z = \mathcal{C}(y)$ 。我们也将用 $T_{a,b}$ 表示在向量 (a, b) 方向上按 $(T_{a,b} \circ y)_{ij} = y_{i-a,j-b}$ 进行的平移算子。我们注意到 L^1 有限性、有限能量和紧支集性质从一个信号传递到它的平移。

命题16.3.3 (平移等变性) 卷积运算保持平移, 即

$$\mathcal{C}(T_{a,b} \circ y) = T_{a,b} \circ \mathcal{C}(y).$$

证明: 首先我们注意到 $(T_{a,b} \circ y)_{ij} = y_{i-a,j-b}$ 。然后对于任意固定的指标 i 和 j , 我们有

$$\begin{aligned} \mathcal{C}(T_{a,b} \circ y)_{ij} &= ((T_{a,b} \circ y) * w)_{ij} = \sum_k \sum_r (T_{a,b} \circ y)_{i+k,j+r} w_{kr} \\ &= \sum_k \sum_r y_{i+k-a,j+r-b} w_{kr} = (y * w)_{i-a,j-b} \\ &= \left(T_{a,b}(y * w) \right)_{ij} = \left(T_{a,b} \circ \mathcal{C}(y) \right)_{ij}. \end{aligned}$$

■

先前的结果表明, 如果输入受到平移的影响, 那么在卷积之后, 输出也会受到相同平移的影响。因此, 由于许多输入图像特征, 如角点、边缘等, 在平移下是不变的, 它们仍将出现在卷积层的输出中。

注记16.3.4 池化对于数据的小平移是不变的, 见第15.2节, 即 $\mathcal{P}(T_{a,b} \circ y) = \mathcal{P}(y)$ 。此属性与卷积的平移等变性兼容, 这一事实使得池化和卷积可以一起应用。也就是说, 如果在卷积之后应用池化, 我们有

$$\mathcal{P} \circ \mathcal{C}(T_{a,b} \circ y) = \mathcal{P} \circ \mathcal{C}(y).$$

相反，如果在池化之后应用卷积，那么

$$\mathcal{C} \circ \mathcal{P}(T_{a,b} \circ y) = \mathcal{C} \circ \mathcal{P}(y).$$

16.4 一维输入的卷积层

卷积层类似于全连接层，不同之处在于它有许多零权重和重复的非零权重。考虑一个神经网络，其输入由紧支集信号 $x = [x_1, x_2, \dots, x_n]$ 给出，并设滑动核为 $w = [w_1, w_2]$ 。核相对于信号滑动的步长称为步幅。在图16.2a和b中，步幅分别为 $s = 1$ 和 $s = 2$ 。在这两种情况下，第二层中的神经元都具有 sigmoid 激活函数。

在图16.2 a的情况下，网络输出可以写成常见的形式， $Y = \sigma(WX + B)$ ，其中 $X = (x_1, \dots, x_6)^T$, $B = (b, b, b, b)^T$, $Y = (y_1, \dots, y_5)^T$ 。权重系统可以写成如下的 5×6 稀疏矩阵

$$W = \begin{pmatrix} w_1 & w_2 & 0 & 0 & 0 & 0 \\ 0 & w_1 & w_2 & 0 & 0 & 0 \\ 0 & 0 & w_1 & w_2 & 0 & 0 \\ 0 & 0 & 0 & w_1 & w_2 & 0 \\ 0 & 0 & 0 & 0 & w_1 & w_2 \end{pmatrix}.$$

我们注意到矩阵每行上重复的权重。

同样，在图16.2 b的情况下，网络输出可以写成 $Y = \sigma(WX + B)$ ，其中 $X = (x_1, \dots, x_6)^T$, $B = (b, b, b)^T$, $Y = (y_1, y_2, y_3)^T$ ，权重系统由 3×6 矩阵给出

$$W = \begin{pmatrix} w_1 & w_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_1 & w_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_1 & w_2 \end{pmatrix}.$$

与线性神经元的卷积 现在假设卷积神经网络的所有神经元都具有线性激活， $\phi(x) = x$ 。我们将证明，这个具有多个卷积层的神经网络等同于一个只有一个卷积层的网络。因此，在卷积网络中使用非线性激活函数对于深度学习至关重要。

只需证明两个卷积层等同于一个卷积层。考虑连续两层中的前向传播

$$X^{(1)} = W^{(1)}X^{(0)} + B^{(1)}$$

$$X^{(2)} = W^{(2)}X^{(1)} + B^{(2)},$$

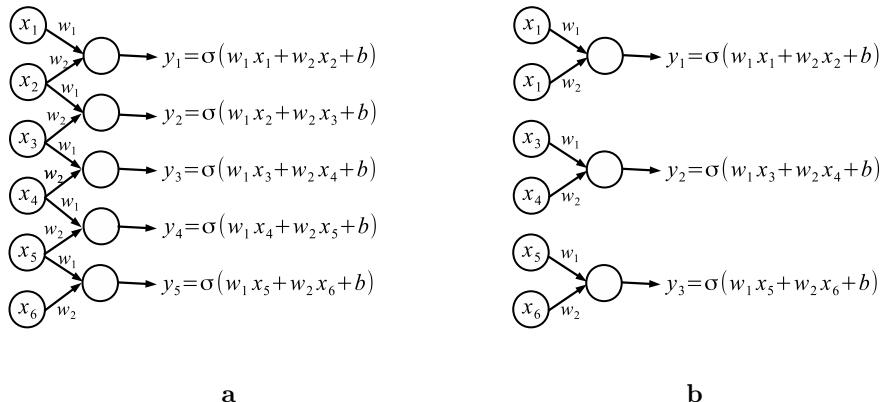


图16.2：两个卷积层：a步长等于1；b步长等于2。所有缺失的箭头都被赋予零权重。所有神经元具有相同的偏差 b 。

具有稀疏类型的权重矩阵。该组合提供

$$X^{(2)} = W^{(2)}(W^{(1)}X^{(0)} + B^{(1)}) + B^{(2)} = WX^{(0)} + B,$$

具有权重矩阵 $W = W^{(2)}W^{(1)}$ 和偏差向量 $B = W^{(2)}B^{(1)} + B^{(2)}$ 。我们需要证明矩阵 W 也是稀疏类型的。我们将在图16.2 a、b中表示的前两种情况下讨论这一点。

图16.2 a中表示的卷积层步长为 $s = 1$ ，仅使用两个共享权重， w_1, w_2 （支撑宽度为2）。我们考虑如图16.3 a所示的两层这种类型的层。然后 Y_1 取决于 X_1, X_2, X_3 ； Y_2 取决于 X_2, X_3, X_4 ，依此类推。卷积网络等同于图16.3 b中给出的两层网络。新网络取决于三个共享权重， ν_1, ν_2, ν_3 （支撑宽度为3），步长为1。图16.3 a、b中的两个网络都满足信息关系

$$\mathfrak{S}(Y_1) \subset \mathfrak{S}(X_1, X_2, X_3), \quad \mathfrak{S}(Y_2) \subset \mathfrak{S}(X_2, X_3, X_4)$$

$$\mathfrak{S}(Y_3) \subset \mathfrak{S}(X_3, X_4, X_5), \quad \mathfrak{S}(Y_4) \subset \mathfrak{S}(X_4, X_5, X_6).$$

可以这样说， Y_1 的感受野由单元 X_1, X_2, X_3 组成，这与全连接层的情况不同，在全连接层中，感受野将由所有先前的神经元组成。因此，卷积层传递的信息比全连接层少。

另一方面，图16.2 b中表示的卷积层步长为 $s = 2$ ，仅使用两个共享权重 w_1, w_2 （支撑宽度为2）。我们考虑如图16.4 a所示的两层这种类型的层。

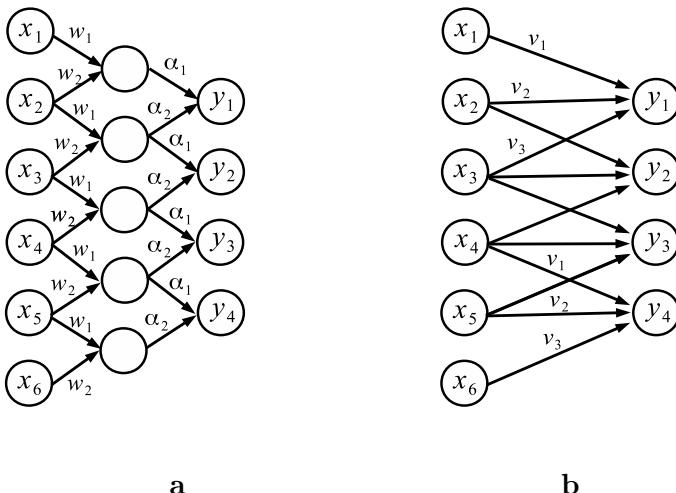


图16.3：两个等效卷积网络：a 步长为1且支撑宽度为2的单隐藏层卷积网络；b 步长为1且支撑宽度为3的两层卷积网络。

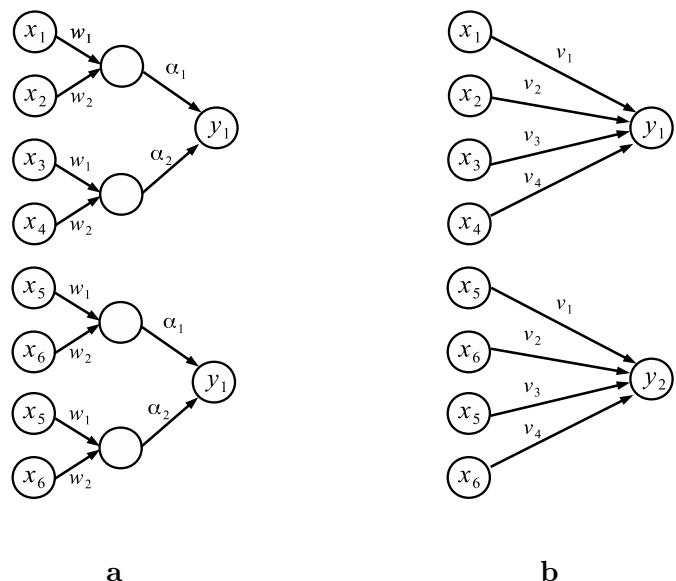


图16.4：两个等效卷积网络：a 步长等于2的单隐藏层卷积网络；b 步长等于4的两层卷积网络。

卷积网络等同于图16.4 b中给出的两层网络。新网络依赖于四个共享权重， $\nu_1, \nu_2, \nu_3, \nu_4$ （支撑宽度为4），步长为1。图16.4 a、b中的两个网络都满足信息关系

$$\mathfrak{S}(Y_1) \subset \mathfrak{S}(X_1, X_2, X_3, X_4), \quad \mathfrak{S}(Y_2) \subset \mathfrak{S}(X_2, X_3, X_4, X_5).$$

特别地， Y_1 的感受野由 X_1, X_2, X_3, X_4 组成。

注记16.4.1 卷积压缩信息。如果 s 表示步长，那么 $d^{(\ell)} = d^{(\ell-1)} + s$ ，即每一层神经元的数量按步长数减少。由于 s 是一个小数，所以压缩比池化的情况要小。

16.5 二维输入的卷积层

卷积网络在处理二维图像方面表现出了卓越的性能。在这种情况下，每个输入都是一个彩色图像（RGB格式），它可以被视为一个 $r \times c \times 3$ 类型的张量，其中 r 是图像的行数， c 是图像的像素列数。这相当于三个维度为 $r \times c$ 的通道，每个通道对应一种颜色。在进一步讨论之前，我们首先要讨论图像的空间切片与给定内核的卷积。

在接下来的内容中，我们将把图16.5 a中给出的 2×2 卷积核与图16.5 b中的 3×3 输入图像进行卷积，这由图16.5 b中的矩阵表示。卷积核与矩阵重叠，并在所有可能的位置水平和垂直移动。在每个位置，我们将核元素与矩阵元素的乘积相加，得到的数字即为输出。核重叠从图像的左上角开始，向右滑动一个像素。然后我们继续对底行进行如下操作：

$$1 \cdot 2 - 1 \cdot 1 + 2 \cdot 4 + 1 \cdot 3 = 12$$

$$1 \cdot 1 - 1 \cdot 1 + 2 \cdot 3 + 1 \cdot 5 = 11$$

$$1 \cdot 4 - 1 \cdot 3 + 2 \cdot 7 + 1 \cdot 6 = 21$$

$$1 \cdot 3 - 1 \cdot 5 + 2 \cdot 6 + 1 \cdot 0 = 10.$$

使用卷积算子 $*$ ，上述计算可写为

$$\begin{pmatrix} 2 & 1 & 1 \\ 4 & 3 & 5 \\ 7 & 6 & 0 \end{pmatrix} * \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 12 & 11 \\ 21 & 10 \end{pmatrix}.$$

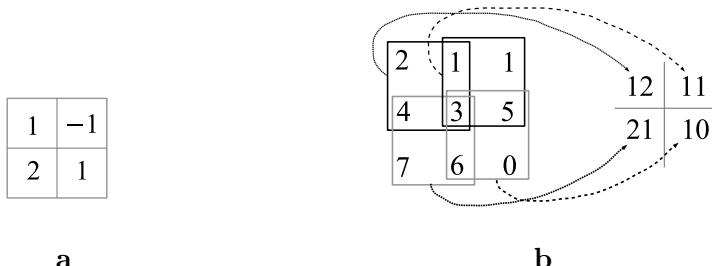


图16.5：卷积操作：a $A 2 \times 2$ 内核；b 卷积的输入和输出。

我们注意到，一个 2×2 内核与一个 3×3 图像之间的卷积是一个大小为 2×2 的输出。一般来说，如果内核大小为 $h \times k$ 且图像为 $H \times K$ ，则输出大小为 $(H - h + 1) \times (K - k + 1)$ 。关于任意步长的情况，请参见练习16.9.11。

在前面的示例中，图像的空间切片已与一个 2×2 内核进行卷积。此操作也称为特征图，因为根据所使用的内核，此过程可以检测不同类型的特征，例如水平或垂直边缘、角点等，读者可以在练习部分看到。

每个颜色通道都可以与相同或不同的内核进行单独卷积。然而，也可以使用三阶张量对所有三色通道图像进行卷积，即使用相同类型的3个内核序列。在这种情况下，卷积操作的定义方式类似。内核在图像上 $(0, 0, 0)$ 位置重叠，然后我们将它们对应元素的乘积求和以获得第一个输出。然后内核从顶部到底部移动一个像素，再从左到右移动以完成操作。

对于固定内核，张量 $X^{(\ell)}$ 的元素由三重索引项 $X_{ijk}^{(\ell)}$ 寻址，其中 $1 \leq i \leq r^{(\ell)}$, $1 \leq j \leq c^{(\ell)}$ 和 $1 \leq k \leq 3$ ，见图16.6。在 ℓ 层中的一个特征图，对于 k 通道并对应于给定内核 $w^{(\ell)}$ 和偏差 $b^{(\ell)}$ ，由下式给出

$$X_{ijk}^{(\ell)} = \phi \left(\sum_s \sum_p X_{i+p,j+r,k}^{(\ell-1)} w_{prk}^{(\ell)} + b^{(\ell)} \right),$$

其中激活函数 ϕ 通常取为ReLU以避免梯度消失。

由于 k 表示颜色通道，那么 $1 \leq k \leq 3$ 。CNN的第 ℓ 层， $X^{(\ell)}$ ，由与所有内核 $w^{(\ell)}$ 对应的形式为 $X_{ijk}^{(\ell)}$ 的所有三阶张量的集合给出。

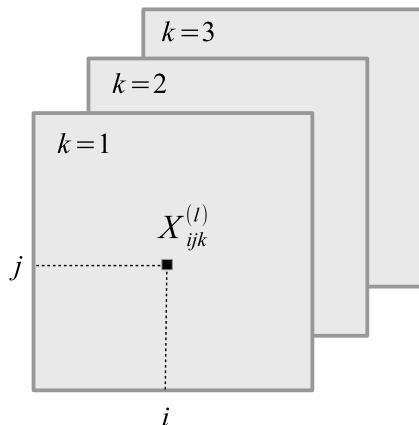


图16.6：张量 $X^{(\ell)} \in \mathbb{R}^{r^{(\ell)} \times c^{(\ell)} \times 3}$ 的元素。元素 $X^{(\ell)}_{ijk}$ 表示位于第 k 个通道中空间位置 (i, j) 处的激活。

如果 $f^{(\ell)}$ 表示第 ℓ 层中的特征图数量（该层使用的内核数量）， $r^{(\ell)} \times c^{(\ell)}$ 是第 ℓ 层中图像的尺寸（行数乘以列数），那么我们可以将第 ℓ 层的输出写为一个4阶张量 $X^{(\ell)} \in \mathbb{R}^{r^{(\ell)} \times c^{(\ell)} \times 3 \times f^{(\ell)}}$ 。

通常，序列 $r^{(\ell)}$ 和 $c^{(\ell)}$ 相对于 ℓ 是递减的。这是因为使用卷积层进行处理往往会使图像尺寸以步长的数量减少。¹（除此之外，在卷积层之间使用池化也会按一定因子压缩尺寸）。通道数保持不变，但特征数量 $f^{(\ell)}$ 会随着 ℓ 增加。

如果输入层由 $X^{(0)}$ 表示，那么第一个隐藏层 $X^{(1)}$ 包含 $X^{(0)}$ 层的特征。第二个隐藏层 $X^{(2)}$ 包含 $X^{(1)}$ 的特征，即输入特征的特征。一般来说，每一层都包含前一层的特征。最后，引入一个全连接层来整合来自最后一个卷积层的所有信息，该卷积层包含大量特征。如果网络用于分类目的，则可以使用softmax层，见图16.7。

¹In order to avoid this dimension reduction, one can use the trick of padding with zeros.

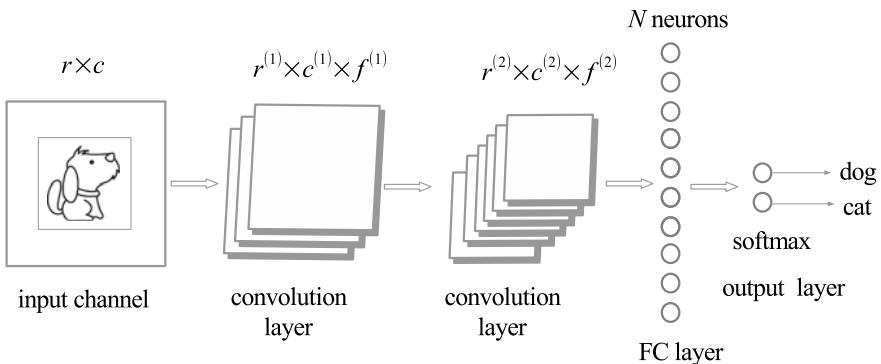


图16.7：具有两个卷积层、一个全连接层和一个softmax层（输出层）的卷积神经网络。

16.6 卷积神经网络的几何结构

在第13章中，我们将一个流形与每个神经网络相关联。网络权重和偏差是相关流形上的坐标系。其维度等于网络参数的数量，可通过公式 (13.3.8) 根据层大小表示。

$$d^{(0)}d^{(1)} + d^{(1)}d^{(2)} + \dots + d^{(\ell-1)}d^{(\ell)} + \dots + d^{(L-1)}d^{(L)} + N. \quad (16.6.2)$$

在卷积神经网络中，权重和偏差在神经元之间共享这一事实大大减少了网络的总参数数量，因此也减少了相关输出流形的维度。这一事实具有正则化效果，因此卷积神经网络通常不太容易过度拟合训练数据。

我们将对两个现有示例进行比较。在图16.3 a所示的卷积神经网络 (CNN) 的情况下，相关神经网络的维度由 $2 + 2 + 5 + 4 = 13$ 给出 (四个权重和九个偏差)。如果各层神经元数量保持不变，用全连接层网络替换该网络，那么公式 (13.3.8) 得出维度

$$6 \cdot 5 + 5 \cdot 4 + 5 + 4 = 59.$$

作为第二个例子，考虑图16.4 a所示的卷积神经网络。那么相关神经网络的维度由 $2 + 2 + 4 + 2 = 10$ 给出 (四个权重和六个偏差)。相关的全连接层网络有一个维度为 $6 \cdot 4 + 4 \cdot 2 + 4 + 2 = 38$ 的相关流形。我们注意到，在这两种情况下，全连接层网络的神经流形维度都比类似的卷积神经网络的维度大得多。对于所有卷积神经网络来说，一般都存在同样的现象。

16.7 等变性与不变性

卷积神经网络能够检测图像中的局部模式，而不管其位置如何。这是因为卷积神经网络确保了对平移的等变性，见命题16.3.3。这意味着如果输入图像被某个向量平移，那么网络各更高层中的激活模式也会被相同向量平移。因此，卷积神经网络在图像识别任务中取得成功的一个关键因素是它们的等变性特性。

下一个抽象层次是用输入的任何变换群来取代平移集，并通过网络相对于所考虑变换的等变性来解释网络参数的对称性。

群论在神经网络中的应用可见于拉万巴赫什等人[100]、孔多尔和特里维迪[66]、孔多尔[65]、科恩等人[26,27]以及巴托克等人[13]的研究。在接下来的内容中，我们将运用上述论文中的观点来简要讨论这个新出现的研究方向。

16.7.1 群

下一个定义定义了一种代数结构，它很快就会有用。

定义16.7.1 群是一个赋予了合成法则 $G \times G \rightarrow G$ （用乘法表示）并满足以下性质的集合 G ：

- (i) 对于任意 $g_1, g_2 \in G$ ， $g_1g_2 \in G$ ；
- (ii) 对于任意 $g_1, g_2, g_3 \in G$ ， $g_1(g_2g_3) = (g_1g_2)g_3$ ；
- (iii) 存在唯一元素 $e \in G$ ，使得对于任意 $x \in G$ ， $xe = ex = x$ ；
- (iv) 对于任意 $g \in G$ ，存在 $g^{-1} \in G$ 使得 $gg^{-1} = g^{-1}g = e$ 。

性质(i)表明 G 关于群运算封闭，而(ii)表明群运算是可结合的；(iii)表明群中存在中性元；逆元的存在性由(iv)给出。

如果在群运算合成中元素的顺序无关紧要，即对于任何 $g_1, g_2 \in G$ ， $g_1g_2 = g_2g_1$ ，那么群 G 称为可交换群。根据元素的数量，群 G 可以是有限的或无限的。

G 的任何子集 H ，如果它在与 G 相同的运算下构成一个群，就称为子群，并用 $H \leq G$ 表示。

示例16.7.2 整数集， $G = \mathbb{Z}$ ，赋予加法运算后构成一个交换群。 n 的逆元是 $-n$ ，中性元是 $e = 0$ 。类似地，整数格， $G = \mathbb{Z} \times \mathbb{Z}$ ，其分量上有加法运算

$$(n_1, n_2) + (m_1, m_2) = (n_1 + m_1, n_2 + m_2)$$

也构成一个交换群。它的中性元素是 $(0, 0)$ 。

$H = 3\mathbb{Z} = \{3m; m \in \mathbb{Z}\}$ 集构成 G 的一个子群，而 $K = \{(2i, 2j); i, j \in \mathbb{Z}\}$ 构成 $\mathbb{Z} \times \mathbb{Z}$ 的一个子群。

例16.7.3 设 $v \in \mathbb{R}^3$ 为一个向量，并通过 $\tau_v(x) = x + v$ 定义平移 $\tau_v: \mathbb{R}^3 \rightarrow \mathbb{R}^3$ 。集合 $G = \mathcal{T}(\mathbb{R}^3) = \{\tau_v; v \in \mathbb{R}^3\}$ 关于函数复合构成一个群，称为 \mathbb{R}^3 上的平移群。我们有 $\tau_v \circ \tau_u = \tau_{v+u}$ 和 $(\tau_v)^{-1} = \tau_{-v}$ 。中性元素是 $\tau_0 = Id$ ，即 \mathbb{R}^3 的恒等变换。

示例16.7.4 2×2 矩阵

$$R_\theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

表示平面 \mathbb{R}^2 绕原点逆时针旋转。这些旋转的集合

$SO(2) = \{R_\theta; \theta \in \mathbb{R}\}$ ，在矩阵乘法下构成一个群，称为 \mathbb{R}^2 的特殊正交群。我们可以很容易地验证 $R_\theta R_{\theta'} = R_{\theta'} R_\theta = R_{\theta+\theta'}$ 和 $R_\theta^{-1} = R_{-\theta}$ 。

由恒等变换和180度旋转（或绕原点翻转）构成的集合

$H = \{R_0, R_\pi\}$ 形成了 $SO(2)$ 的一个子群。

此外，集合 $K = \{R_0, R_{\pi/4}, R_{\pi/2}, R_{3\pi/4}\}$ 形成了 $SO(2)$ 的另一个子群，该子群由绕原点的90度旋转构成。

例16.7.5 集合 $G = \mathbb{R}^3$ 以及合成法则

$$(x_1, x_2, x_3) \circ (y_1, y_2, y_3) = (x_1 + y_1, x_2 + y_2, x_3 + y_3 + x_1 y_2)$$

形成一个群，称为三维海森堡群。这不是一个交换群。元素的逆由下式给出

$$(x_1, x_2, x_3)^{-1} = (-x_1, -x_2, -x_3 + x_1 x_2).$$

中性元素是 $e = (0, 0, 0)$ 。

16.7.2 群在集合上的作用

定义16.7.6 设 G 为一个群， M 为一个集合。 G 在 M 上的一个作用是一个映射 $\alpha: G \times M \rightarrow M$ ，使得：

(i) $\alpha(gg', x) = \alpha(g, \alpha(g', x))$, $\forall g, g' \in G$ 和 $x \in M$ ；

(ii) $\alpha(e, x) = x$, $\forall x \in M$ 。

我们称群 G 以作用 α 作用于集合 M ，并且元素 g 对 x 的作用是 $\alpha(g, x)$ 。因此，部分 (i) 指出乘积 gg' 对 x 的作用是 g 的作用与 g' 对 x 的作用的复合。部分 (ii) 指出中性元素的作用是恒等映射。

群 G 在集合 M 上的任何作用，由 $\alpha : G \times M \rightarrow M$ 给出，如下产生 M 的一族变换。对于任何固定的 $g \in G$ ，设 $T_g : M \rightarrow M$ 由 $T_gx = \alpha(g, x)$ 定义。这些变换的集合 $\{T_g; g \in G\}$ ，通过函数复合形成一个群。这由作用的性质得出，这些性质意味着 $T_gT_{g'} = T_{gg'}$ 和 $T_ex = x$ 。我们也有逆变换 $(T_g)^{-1} = T_{g^{-1}}$ 。

对于给定元素 $x \in M$ ，集合 $\mathcal{O}_x = \{T_gx; g \in G\}$ 称为 x 的轨道。如果 $y, z \in \mathcal{O}_x$ 是 x 轨道中的两个元素，那么存在 $g, g' \in G$ 使得 $y = T_gx$ 且 $z = T_{g'}x$ 。如果令 $u = g'g^{-1}$ ，那么 $z = T_uy$ ，即， $z \in \mathcal{O}_y$ 。事实上，可以证明 $\mathcal{O}_x = \mathcal{O}_y = \mathcal{O}_z$ 。

定义16.7.7 一个作用 $\alpha : G \times M \rightarrow M$ 被称为可迁的，如果对于任意两个元素 $x, y \in M$ ，存在 $g \in G$ 使得 $y = T_gx$ 。

等价地说，作用 α 是可迁的，如果对于任意 $x, y \in M$ 我们有 $\mathcal{O}_x = \mathcal{O}_y$ 。事实上，作用 α 是可迁的，当且仅当它只有一个轨道，即，对于所有 $x \in M$ ， $M = \mathcal{O}_x$ 。

我们说 M 是 G 的一个齐性空间，如果对于任意 $x, y \in M$ 存在一个 $g \in G$ 使得 $y = \alpha(g, x)$ 。这与该作用只有一个轨道这一事实等价。

示例16.7.8 设 $M = \mathbb{R}^3$ 和 $G = \mathcal{T}(\mathbb{R}^3)$ ，即在向量加法运算下空间 \mathbb{R}^3 中的平移群。那么 G 对 M 的作用如下：如果 $x \in \mathbb{R}^3$ 是一个向量且 $g = \tau_v$ 是向量 v 的平移，那么我们通过 $\alpha(g, x) = \tau_v(x)$ 定义 g 对 x 的作用， $\alpha(g, x) = \tau_v(x)$ 是通过将向量 v 加到 x 上得到的向量。更明确地说，我们有

$$\alpha(g, x) = \tau_v(x) = x + v.$$

读者可以轻松检查操作定义属性。该操作是可传递的，因为任何元素 $x \in \mathbb{R}^3$ 都可以转换为任何其他元素 $y \in \mathbb{R}^3$ 。

示例16.7.9 考虑 $M = \mathbb{R}^2$ 和 $G = SO(2)$ ，即在复合运算下二维欧几里得平面 \mathbb{R}^2 绕原点的旋转群。群 G 对集合 M 的作用如下：若 $x \in \mathbb{R}^2$ 是一个向量且 $g = R_\theta$ 是角度为 θ 的旋转，则我们定义 g 对 x 的作用为 $\alpha(g, x) = R_\theta x$ ，它是 x 在逆时针旋转角度 θ 后得到的向量。更明确地说，我们有

$$\alpha(g, x) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \cos \theta - x_2 \sin \theta \\ x_1 \sin \theta + x_2 \cos \theta \end{pmatrix}.$$

此操作不是可迁的。元素 $x \in \mathbb{R}^2$ 的轨道是以原点为中心、半径为 $\|x\|$ 的圆

$$\mathcal{O}_x = \{y \in \mathbb{R}^2; \|y\| = \|x\|\}.$$

例16.7.10 在本示例中， $M = \mathbb{R}^3$ 和 $G = (\mathbb{R}^3, \circ)$ 表示三维海森堡群。那么 G 作用于 M 如下

$$\alpha(g, x) = (g_1 + x_1, g_2 + x_2, g_3 + x_3 + g_1 x_2),$$

其中 $g = (g_1, g_2, g_3)$ 和 $x = (x_1, x_2, x_3)$ 。检验 α 是一个作用的最简单方法是注意到

$$\alpha(g, x) = L_g x, \quad (16.7.3)$$

其中 L_g 是群 (H, \circ) 上的左平移，即 $L_g x = g \circ x$ 。那么

$$\alpha(g', \alpha(g, x)) = \alpha(g', L_g x) = L_{g'} L_g x = L_{g'g} x = \alpha(g'g, x).$$

该作用的第二个性质显然满足，因为海森堡群中的中性元素是 $e = (0, 0, 0)$ 。值得注意的是，由关系(16.7.3)定义的群对自身的作用 $\alpha : G \times G \rightarrow G$ 称为左乘作用。海森堡群对空间 \mathbb{R}^3 的作用是可迁的，因为对于任何 $x, y \in \mathbb{R}^3$ ，我们有 $T_g x = y$ ，其中 $g = (y_1 - x_1, y_2 - x_2, y_3 - x_3 + x_1 x_2 - y_1 x_2)$ 。

最后一个例子引出了这样一种情况，即群 G 作用于集合 $M = G$ ，而这个集合就是群本身；在这种情况下，该作用的表现为 $\alpha : G \times G \rightarrow G$ 。如果我们定义 $\alpha(g, x) = xg^{-1}$ ，那么 α 就被称为左乘作用。如果考虑 $\alpha(g, x) = gxg^{-1}$ ，那么 α 就被称为共轭作用。

16.7.3 作用到函数的扩展

我们已经看到一个动作 $\alpha : G \times M \rightarrow M$ 如何为每个群元素 $g \in G$ 在 $M, T_g : M \rightarrow M$ 上诱导一个变换。现在我们将把这个变换扩展到 M 上的函数。我们用 $\mathcal{F}(M) = \{f : M \rightarrow \mathbb{R}\}$ 表示 M 上的实函数集。对于任何元素 $g \in G$ ，我们通过 $\mathbb{T}_g f = f'$ 定义变换

$\mathbb{T}_g : \mathcal{F}(M) \rightarrow \mathcal{F}(M)$ ，其中

$$f'(T_g(x)) = f(x), \quad x \in M.$$

这可以等效地写为

$$(\mathbb{T}_g f)(x') = f(T_{g^{-1}}(x')), \quad \forall x' \in M.$$

我们需要这个概念，因为神经网络每一层的激活都被视为函数。例如，每个MNIST数据都被视为在维度为 28×28 的集合 M 上定义的函数，其整数值介于0和255之间。

示例16.7.11 设 \mathbb{Z} 表示整数集，并考虑在分量上进行加法运算的群 $G = (\mathbb{Z} \times \mathbb{Z}, +)$ 。它通过作用 $\alpha : G \times M \rightarrow M$ 作用于格 $M = \mathbb{Z} \times \mathbb{Z}$ 。

$$\alpha((g_1, g_2), (x_1, x_2)) = (g_1 + x_1, g_2 + x_2), \quad \forall g_i, x_i \in \mathbb{Z}.$$

在 M 上诱导的变换是

$$T_{(g_1, g_2)}(x_1, x_2) = (g_1 + x_1, g_2 + x_2), \quad \forall (x_1, x_2) \in \mathbb{Z} \times \mathbb{Z}.$$

在 $\mathcal{F}(\mathbb{Z} \times \mathbb{Z})$ 上相关的扩展变换由下式给出

$$(\mathbb{T}_{(g_1, g_2)} f)(x_1, x_2) = f(T_{(g_1, g_2)^{-1}}(x_1, x_2)) = f(x_1 - g_1, x_2 - g_2),$$

它是函数 f 与向量 $-(g_1, g_2)$ 的平移之间的复合。

示例16.7.12 在示例16.7.9的情况下， \mathbb{R}^2 上的诱导变换由下式给出

$$T_{R_\theta} x = (x_1 \cos \theta - x_2 \sin \theta, x_1 \sin \theta + x_2 \cos \theta),$$

它是角度为 θ 的顺时针旋转。函数上的扩展变换为

$$(\mathbb{T}_{R_\theta} f)(x) = f(\mathbb{T}_{R_\theta^{-1}} x) = f(x_1 \cos \theta + x_2 \sin \theta, x_2 \cos \theta - x_1 \sin \theta).$$

16.7.4 等变性的定义

考虑一个作用于两个集合 M_1 和 M_2 的群 G ，其作用为

$$\alpha_1 : G \times M_1 \rightarrow M_1, \quad \alpha_2 : G \times M_2 \rightarrow M_2.$$

那么对于任何 $g \in G$ ，这些作用诱导出变换

$$T_g : M_1 \rightarrow M_1, \quad T'_g : M_2 \rightarrow M_2.$$

考虑对函数的扩展变换

$$\mathbb{T}_g : \mathcal{F}(M_1) \rightarrow \mathcal{F}(M_1), \quad \mathbb{T}'_g : \mathcal{F}(M_2) \rightarrow \mathcal{F}(M_2).$$

一个映射 $\Phi : \mathcal{F}(M_1) \rightarrow \mathcal{F}(M_2)$ 如果对于任何群元素 $g \in G$ 都满足我们有，则称其为 G -等变的

$$\Phi(\mathbb{T}_g f) = \mathbb{T}'_g(\Phi(f)), \quad \forall f \in \mathcal{F}(M_1).$$

这些概念可以按如下方式应用于神经网络。我们考虑 M_1 和 M_2 分别为前馈神经网络输入层和输出层中的神经元集合。这些层的激活，即输入 $x^{(0)}$ 和输出 $x^{(L)}$ ，分别是在 M_1 和 M_2 上定义的函数。将 $x^{(0)}$ 映射到 $x^{(L)}$ 的函数 $\Phi = f_{w,b}$ 是网络的输入 - 输出映射。网络关于群作用 G 的等变性质变为

$$f_{w,b}(\mathbb{T}_g x^{(0)}) = \mathbb{T}'_g(f_{w,b}(x^{(0)})) = \mathbb{T}'_g(x^{(L)}), \quad \forall g \in G. \quad (16.7.4)$$

这表明，在由群作用诱导的变换族中，当我们变换输入 $x^{(0)}$ 时，输出 $x^{(L)}$ 会以可预测的方式进行变换。

网络的等变性可以如下在每一层上定义，见康多 (Kondor) [66]。设 \mathcal{N} 是一个具有 $L + 1$ 层和层激活函数 $x^{(0)}, x^{(1)}, \dots, x^{(L)}$ 的前馈神经网络。如果 M_ℓ 表示第 ℓ 层中的神经元集合，那么 $x^{(\ell)} \in \mathcal{F}(M_\ell)$ 。假设存在一个群 G ，它作用于集合 M_0, \dots, M_L 。在 $\mathcal{F}(M_0), \dots, \mathcal{F}(M_L)$ 上的诱导变换相应地用 $\mathbb{T}^{(0)}, \dots, \mathbb{T}^{(L)}$ 表示。如果当输入通过 $x^{(0)} \rightarrow \mathbb{T}_g^{(0)}(x^{(0)})$ 进行变换时，对于任何 $g \in G$ ，层激活函数通过 $x^{(\ell)} \rightarrow \mathbb{T}_g^{(\ell)}(x^{(\ell)})$ 进行变换，那么神经网络 \mathcal{N} 被称为 G -等变前馈网络。

值得注意的是，当神经元集合 M_ℓ 为离散或连续时，这个定义在两种情况下都成立。

16.7.5 卷积与等变性

等变性的典型例子由卷积网络给出，卷积网络对于平移群的作用是等变的。用之前的符号表示，这意味着

$$(\mathbb{T}_g x^{(0)}) * w = \mathbb{T}'_g(x^{(0)} * w), \quad \forall g \in G,$$

对于任何滤波器 w ，见图16.8。这种关系在离散和连续情况下都可以证明。为了简单起见，我们只考虑一维情况。

离散情形可由以下推出

$$\begin{aligned} [(\mathbb{T}_g x^{(0)}) * w]_p &= \sum_i (\mathbb{T}_g x^{(0)})_{i+p} w_i = \sum_i x_{i+p-g}^{(0)} w_i \\ &= (x^{(0)} * w)_{p-g} = [\mathbb{T}'_g(x^{(0)} * w)]_p. \end{aligned}$$

连续情形下等变性的验证由以下给出

$$\begin{aligned} [(\mathbb{T}_g x^{(0)}) * w](t) &= \int_{\mathbb{R}} (\mathbb{T}_g x^{(0)})(u+t) w(u) du = \int_{\mathbb{R}} x^{(0)}(u+t-g) w(u) du \\ &= (x^{(0)} * w)(t-g) = [\mathbb{T}'_g(x^{(0)} * w)](t). \end{aligned}$$

先前的计算可以像下面这样推广到群上。首先，在离散群 G 上定义的任意两个函数 $f, \psi : G \rightarrow \mathbb{R}$ 的卷积定义为

$$(f * \psi)(t) = \sum_{y \in G} f(y) \psi(t^{-1}y).$$

G -等变性关系的验证与先前的计算类似

$$\begin{aligned} (\mathbb{T}_g x^{(0)} * w)(t) &= \sum_{y \in G} (\mathbb{T}_g x^{(0)})(y) w(t^{-1}y) = \sum_{y \in G} x^{(0)}(g^{-1}y) w(t^{-1}y) \\ &= \sum_{v \in G} x^{(0)}(v) w(t^{-1}gv) = \sum_{v \in G} x^{(0)}(v) w((g^{-1}t)^{-1}v) \\ &= (\mathbb{T}_g(x^{(0)} * w))(t), \end{aligned}$$

这里我们使用了变量替换 $v = g^{-1}y$ 以及变量 $v \in g^{-1}G = G$ 的事实。

等变性理论可以扩展到连续紧群，见[66]。在此情形下任意两个函数 $f, \psi : G \rightarrow \mathbb{R}$ 的卷积定义为

$$(f * \psi)(t) = \int_G f(y) \psi(t^{-1}y) d\mu(y),$$

其中 μ 是 G 上的左平移不变测度，对于 $\mu(G) = 1$ ，称为 G 上的哈尔测度。等变性的验证与离散情况类似，将求和替换为积分并使用哈尔测度的不变性性质

$$\begin{aligned} (\mathbb{T}_g x^{(0)} * w)(t) &= \int_G (\mathbb{T}_g x^{(0)})(y) w(t^{-1}y) d\mu(y) = \int_G x^{(0)}(g^{-1}y) w(t^{-1}y) d\mu(y) \\ &= \int_G x^{(0)}(v) w(t^{-1}gv) d\mu(v) = \int_G x^{(0)}(v) w((g^{-1}t)^{-1}v) d\mu(v) \\ &= (\mathbb{T}_g(x^{(0)} * w))(t). \end{aligned}$$

$$\begin{array}{ccc}
 x^{(0)} & \xrightarrow{\hspace{3cm}} & x^{(L)} = x^{(0)} * w \\
 \downarrow T_g & & \downarrow T_g \\
 T_g(x^{(0)}) & \xrightarrow{\hspace{3cm}} & T_g(x^{(0)}) * w
 \end{array}$$

图16.8：等变性可视为一个交换图。

值得注意的是，科恩等人[26]将该技术应用于分析球面图像。他们通过 $\mathbb{S}^2 = \{x \in \mathbb{R}^3; \|x\| = 1\}$ 将卷积的定义从平面域适配到球体。

$$(f * \psi)(R) = \int_{\mathbb{S}^2} f(x) \psi(R^{-1}x) dx,$$

其中 $\psi, f : \mathbb{S}^2 \rightarrow \mathbb{R}$ 是两个球面信号， $R \in SO(3)$ 是一个旋转。他们利用特殊正交群 $SO(3)$ （即保持距离且行列式为1的 3×3 矩阵组成的群）在球体 \mathbb{S}^2 上的作用来证明旋转等变性。

即使平面卷积对于平移群的作用总是等变的，但除非做出一些额外假设，否则它对于旋转群 $SO(2)$ 不是协变的。这将在以下内容中实现。

设 $R \in SO(2)$ 为平面的一个旋转，且 $f, w : \mathbb{R}^2 \rightarrow \mathbb{R}$ 。那么

$$\begin{aligned}
 [(\mathbb{T}_R f) * w](x) &= \sum_{y \in \mathbb{Z}^2} (\mathbb{T}_R f)(y) w(y - x) = \sum_{y \in \mathbb{Z}^2} f(R^{-1}y) w(y - x) \\
 &= \sum_{u \in \mathbb{Z}^2} f(u) w(Ru - x) = \sum_{u \in \mathbb{Z}^2} f(u) w(R(u - R^{-1}x)) \\
 &= \sum_{u \in \mathbb{Z}^2} f(u) (\mathbb{T}_{R^{-1}} w)(u - R^{-1}x) \\
 &= (f * \mathbb{T}_{R^{-1}} w)(R^{-1}x) = \mathbb{T}_R(f * \mathbb{T}_{R^{-1}} w).
 \end{aligned}$$

如果现在我们将 f 视为一个信号，将 w 视为一个滤波器，并且进一步假设 w 是一个旋转不变滤波器，即 $\mathbb{T}_{R^{-1}} w = w$ ，那么前面的公式就变成

$$[(\mathbb{T}_R f) * w](x) = \mathbb{T}_R(f * \mathbb{T}_{R^{-1}} w) = \mathbb{T}_R(f * w),$$

它表示平面卷积对于旋转不变滤波器的协变性。

对于 $G = \mathbb{Z}^2$ 组，我们认为旋转 R 是晶格 \mathbb{Z}^2 绕原点旋转90度、180度和270度。在这种情况下，滤波器 $w(Rx) = w(x)$ 的旋转不变性变为

$$w_{i,j} = w_{-j,i} = w_{i,-j} = w_{j,-i}, \quad \forall (i,j) \in \mathbb{Z}^2.$$

16.7.6 不变性的定义

不变性是等变性的一种特殊情况。在这种情况下，当输入通过变换族 $\{\mathbb{T}_g; g \in G\}$ 进行变换时，网络输出不会改变。在这种情况下，公式(16.7.4)变为

$$f_{w,b}(\mathbb{T}_g x^{(0)}) = f_{w,b}(x^{(0)}) = x^{(L)}, \quad \forall g \in G.$$

这意味着对于所有 $g \in G$ ， \mathbb{T}'_g 的恒等映射为 Id ，其中 Id 是 $\mathcal{F}(M_L)$ 的恒等映射。那么对于任何 $f \in \mathcal{F}(M_L)$

$$(\mathbb{T}'_g f)(x) = f(T'_{g^{-1}}(x)) = f(x), \quad \forall x \in M_L.$$

因此，对于所有 $x \in M_L$ 和 $g \in G$ ， $T'_g(x) = x$ 成立。这等同于 M_L 中每个元素的轨道就是该元素本身这一事实，即对于所有 $x \in M_L$ ， $\mathcal{O}_x = \{x\}$ 。

网络对局部平移不变性的一个典型例子出现在池化的情况下，见15.2节。接下来我们将在群的背景下讨论池化过程。

设 G 为一个群， $f : G \rightarrow \mathbb{R}$ 为一个特征映射。² 设 $U \subset G$ 为 G 的一个包含中性元素 $e \in U$ 的子集，称为池化域。最大池化算子 \mathcal{P} 由 $\mathcal{P}f : G \rightarrow \mathbb{R}$ 定义

$$(\mathcal{P}f)(x) = \max_{u \in xU} f(u).$$

一个特殊的情况是当合并域 U 是一个子群 $H \leq G$ 时。那么合并域 $\{xH; x \in G\}$ 构成群 G 的一个划分，即，如果 $x, y \in G$ ，那么 $xH = yH$ 或 $xH \cap yH = \emptyset$ 。此外，如果 G 是一个有限群， H 和 xH 具有相同数量的元素，见练习16.9.12。划分集 $\{xH; x \in G\}$ 被称为陪集，并且代表在 G 上以下等价关系的等价类

$$x \sim y \iff x^{-1}y \in H.$$

² 如果 $f : M \rightarrow \mathbb{R}$ ，那么 f 是一个激活映射，因为它描述了集合 M 中神经元的激活情况。然而，如果 $f : G \rightarrow \mathbb{R}$ ，那么 f 是一个特征映射，因为它描述了组 G 捕获的特征。

池化在由定义的陪集空间上诱导一个应用 ϕ

$$\phi(xH) = (\mathcal{P}f)(x).$$

示例16.7.13 这是一个使用 2×2 池化域进行最大池化的经典示例，该池化域在二维整数格上移动。为此，设 $G = \mathbb{Z} \times \mathbb{Z}$ 是一个在分量上进行加法运算的群，并考虑激活函数 $f: G \rightarrow \mathbb{R}$ 。考虑池化域

$$U = \{(i, j); -2 \leq i \leq 2, -2 \leq j \leq 2\}.$$

池化变为 $(\mathcal{P}f)(x) = \max_{u \in x+U} f(u)$ ，其中

$$x + U = \{(i, j); -2 + x_1 \leq i \leq 2 + x_1, -2 + x_2 \leq j \leq 2 + x_2\},$$

是域 U 沿向量 (x_1, x_2) , $x_i \in \mathbb{Z}$ 的平移。

如果池化域是 G 的一个子群

$$H = \{(3n_1, 3n_2); (n_1, n_2) \in \mathbb{Z} \times \mathbb{Z}\}$$

然后，在将 x 按3像素的倍数进行平移的情况下取最大值，这对应于步长为 $s = 2$ 像素。

组上下文下的最大池化不变性在下一个结果中展示，[27]。

命题16.7.14 池化与群作用可交换：

$$\mathcal{P}(\mathbb{T}_g f)(x) = \mathbb{T}_g(\mathcal{P}f)(x), \quad \forall x, g \in G, \quad f: G \rightarrow \mathbb{R}.$$

证明：进行替换 $u = g^{-1}h$ 并使用 \mathbb{T}_g 的定义，我们有

$$\begin{aligned} \mathcal{P}(\mathbb{T}_g f)(x) &= \max_{h \in xU} (\mathbb{T}_g f)(h) = \max_{h \in xU} f(g^{-1}h) \\ &= \max_{u \in g^{-1}xU} f(u) = \mathcal{P}f(g^{-1}x) \\ &= \mathbb{T}_g(\mathcal{P}f)(x). \end{aligned}$$

注记16.7.15 为了简单起见，我们仅考虑了只有一个通道的激活函数和滤波器。

通过考虑 $f, w : M \rightarrow \mathbb{R}^K$ ，结果可以扩展到多通道情况，其中 K 是通道数。卷积在离散情况下可以定义为

$$(f * w)(g) = \sum_{y \in G} \sum_{k=1}^K f_k(y) w_k(g^{-1}y) = \sum_{y \in G} \langle f(y), w(g^{-1}y) \rangle$$

在连续情况下定义为

$$(f * w)(g) = \int_G \sum_{k=1}^K f_k(y) w_k(g^{-1}y) d\mu(y) = \int_G \langle f(y), w(g^{-1}y) \rangle d\mu(y),$$

其中 μ 是紧致群 G 上的哈尔测度。

16.8 总结

卷积神经网络是专门用于处理图像数据的神经网络。它们在诸如手写数字识别等实际应用中极其成功。卷积神经网络包含卷积层，这些卷积层使用带核的卷积而不是常规的矩阵乘法。与输入图像尺寸相比，核通常是维度相对较小的矩阵。

在卷积神经网络（CNN）中，各层具有稀疏交互并共享权重和偏差。与内核维度匹配的局部感受野每次滑动一个（或多个）像素，首先水平滑动，然后垂直滑动，从顶部到底部，直到整个输入数据被完全扫描。内核项与感受野激活值之间的乘积之和产生一个数字，该数字作为特征图中的一个条目存储。输入数据与每个内核之间的卷积产生不同的特征图，这些特征图保留了数据的某些特征，例如角点、边缘或其他简单形状。

使用一定数量的内核来生成特征图。在多层次CNN的情况下，后续层由特征的特征组成。卷积层与池化层交替使用。最后，CNN包含一个（或多个）全连接层，然后是输出层的 softmax 激活。

卷积神经网络能够检测图像中的局部模式，而不管其位置如何，因为卷积神经网络对平移具有等变性。这意味着如果输入图像按某个向量进行平移，那么网络各更高层中的激活模式也会按相同向量进行平移。等变性属性可以针对作用于神经网络各层的任何变换群来定义。该理论适用于离散群和连续紧群。

16.9 练习

练习16.9.1 计算以下矩阵卷积：

$$\begin{pmatrix} 3 & 1 & 1 \\ 4 & 2 & 0 \\ 5 & -1 & 1 \end{pmatrix} * \begin{pmatrix} -1 & 0 \\ 1 & -2 \end{pmatrix}.$$

练习16.9.2 (a) 针对一维输入的情况，阐述并证明命题16.3.3的一个变体。

(b) 如果输入是张量，平移等变性是否仍然成立？

练习16.9.3 (索贝尔算子) (a) 证明图像与 3×3 核的卷积

$$K = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

强调水平边缘检测。

(b) 证明与转置核 K^T 的卷积可过滤垂直边缘。

练习16.9.4 证明图像与以下 3×3 核的卷积会使图像模糊：

(a) (盒式模糊)

$$K = \begin{pmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{pmatrix}$$

(b) (高斯模糊)

$$K = \begin{pmatrix} 1/16 & 1/8 & 1/16 \\ 1/8 & 1/4 & 1/8 \\ 1/16 & 1/8 & 1/16 \end{pmatrix}$$

练习16.9.5 给出支持图像与 3×3 核卷积

$$K = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

会锐化图像这一事实的论据。

练习16.9.6 考虑以下 3×3 卷积核

$$K_1 = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \quad K_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

$$K_3 = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}.$$

证明支持使用这些核进行卷积的效果是边缘检测的论据。请注意，每个核中元素的和为0。这意味着什么？

练习16.9.7 考虑以下卷积图像处理：从原始图像中每个像素值减去其左边相邻像素值。编写一个执行此操作的 3×3 核。

练习16.9.8 设 y 为二维离散信号。如果 \mathcal{C} 和 \mathcal{P} 分别表示卷积和池化算子，证明以下关系：

- (a) $\mathcal{P} \circ \mathcal{C}(T_{a,b} \circ y) = \mathcal{P} \circ \mathcal{C}(y)$.
- (b) $\mathcal{C} \circ \mathcal{P}(T_{a,b} \circ y) = \mathcal{C} \circ \mathcal{P}(y)$.

练习16.9.9 如果要对图16.7中给出的卷积神经网络进行正则化，应该在哪个层使用随机失活技术？

练习16.9.10 解释为什么在具有相同输入、相同深度和宽度的两个网络之间，卷积神经网络比全连接神经网络更不容易过拟合。

练习16.9.11 卷积层会压缩信息，即输出大小小于输入大小。

(i) 在一维卷积层的情况下，如果 N 表示输入的大小，使用大小为 F 且步幅为 S 移动的滤波器，我们会得到大小为 O 的输出，计算公式为

$$O = \frac{N - F}{S} + 1.$$

如果使用填充，用 P 表示，那么公式变为

$$O = \frac{N - F + 2P}{S} + 1.$$

(ii) 在二维卷积层的情况下，如果输入维度为 $W_1 \times H_1$ 且 F 是方形滤波器的大小，那么输出维度为 $W_2 \times H_2$ ，其中

$$W_2 = \frac{W_1 - F + 2P}{S} + 1, \quad H_2 = \frac{H_1 - F + 2P}{S} + 1.$$

练习16.9.12 设 H 是群 G 和 $x \in G$ 的子群。用 $xH = \{xh; h \in H\}$ 表示。证明子集 xH 具有以下性质：

- (i) $x \in xH$ ；
 - (ii) 如果 $x, y \in G$ ，那么要么 $xH = yH$ ，要么 $xH \cap yH = \emptyset$ ；
 - (iii) 如果 G 是有限群，那么 H 和 xH 具有相同数量的元素；
 - (iv) 考虑在 $G \times G$ 上定义的关系“ \sim ”，其定义为 $x \sim y \Leftrightarrow y^{-1}x \in H$ 。证明“ \sim ”满足：
- (a) $x \sim x$
 - (b) $x \sim y \Rightarrow y \sim x$
 - (c) $x \sim y$ 和 $y \sim z \Rightarrow x \sim z$ 。



第17章 循环神经网络

前馈全连接神经网络不能成功用于对数据序列进行建模。一些基本原因如下：它们无法处理变长输入序列，不共享参数，无法跟踪长期依赖关系，也无法保留有关输入数据顺序的信息。因此，需要一种能够成功满足上述所有要求的神经网络架构。这就是循环神经网络，即RNN（鲁梅尔哈特等人[108]），本章将围绕它展开讨论。

17.1 状态系统

状态系统比循环神经网络更容易理解，并且将作为后续讨论循环神经网络的基础。



图17.1：a. 一个与外部隔离的动态系统，由方程(17.1.1)给出。b. 一个由过程 X_t 驱动的动态系统，如方程(17.1.2)所示。

这就是为什么我们首先要考虑一个动力系统，其在时间 t 的状态 h_t 更新如下

$$h_t = f(h_{t-1}; \theta), \quad t \geq 1 \quad (17.1.1)$$

转移函数 $f : \mathbb{R}^k \times \mathbb{R}^m \rightarrow \mathbb{R}^k$ 是一个博雷尔可测函数， $\theta \in \mathbb{R}^m$ 表示一个向量参数，它独立于 t ，见图17.1a。我们假设状态 h_t 是随机变量（离散或连续），并用 $\mathcal{H}_t = \mathfrak{S}(h_t)$ 表示由 h_t 生成的 σ -代数（见附录D.5节中的定义）。每个状态 h_t 包含关于动力系统的一些信息 \mathcal{H}_t 。由于状态 h_t 由 h_{t-1} 确定，命题D.5.1意味着 $\mathcal{H}_t \subset \mathcal{H}_{t-1}$ ，即我们得到一个 σ -代数的递减序列。也就是说，每个状态包含的信息比前一个状态少或相同。

如果 $f(\cdot; \theta)$ 在第一个自变量中是可逆的，那么 $\mathcal{H}_t = \mathcal{H}_{t-1}$ ，也就是说，信息不会缩减。然而，这个条件过于严格，一般来说，随着 t 的增加，信息会严格减少。在下面，我们将考虑一个涉及收缩原理（定理7.7.1）应用的类似例子。

例17.1.1 假设 f 是可微的且满足不等式 $\|\partial f / \partial h\| < \lambda < 1$ 。那么根据中值定理，函数 f 在第一个自变量中成为一个 λ -收缩映射。

$$\|f(h; \theta) - f(h'; \theta)\| \leq \max_h \|\partial_h f\| \|h - h'\| < \lambda \|h - h'\|, \quad \forall h, h' \in \mathbb{R}^k.$$

对于样本空间中的任意两个状态 $\omega, \omega' \in \Omega$ ，我们将通过展开递推关系来评估它们在时间 t 时对应状态之间的距离，如下所示：

$$\begin{aligned} \|h_t(\omega) - h_t(\omega')\| &= \|f(h_{t-1}(\omega); \theta) - f(h_{t-1}(\omega'); \theta)\| < \lambda \|h_{t-1}(\omega) - h_{t-1}(\omega')\| \\ &< \lambda^2 \|h_{t-2}(\omega) - h_{t-2}(\omega')\| < \cdots < \lambda^t \|h_0(\omega) - h_0(\omega')\|. \end{aligned}$$

由于随机变量 h_0 几乎处处有限，那么对于几乎所有的 ω 和 ω' ， $\|h_0(\omega) - h_0(\omega')\|$ 是有限的。取 $t \rightarrow \infty$ 并使用夹逼定理，我们得到

$$\lim_{t \rightarrow \infty} \|h_t(\omega) - h_t(\omega')\| = 0,$$

对于几乎所有的 ω 和 ω' 。这意味着 $\lim_{t \rightarrow \infty} h_t = c$ 几乎处处为常数，并且这种收敛几乎必然成立，即 $\mathbb{P}(\omega; h_t(\omega) \rightarrow c) = 1$ 。事实上， c 是压缩函数 f 的不动点，即

$$f(c; \theta) = c.$$

显然， c 取决于参数 θ 。在这个例子中， σ -代数序列 \mathcal{H}_t 是严格递减的，其下极限是 $\mathcal{H}_\infty = \bigcap_{t>1} \mathcal{H}_t = \mathfrak{S}(c) = \{\emptyset, \Omega\}$ ，即平凡 σ -代数。因此，从长远来看，系统会丢失所有信息。忘记过去的信息是具有收缩转移函数的动态系统的一个特征。这将构成循环神经网络梯度消失问题的原因，本章稍后将对此进行探讨。

由于一般来说，由 \mathcal{H}_t 所投射的信息会随着 t 的增加而减少，所以我们将考虑在每个时间实例插入系统的额外信息。因此，我们考虑一个由随机过程 $X_t, t \geq 1$ 建模的外部信号驱动的动态系统。在这种情况下，递归可以写成

$$h_t = f(h_{t-1}, X_t; \theta), \quad t \geq 1, \quad (17.1.2)$$

见图17.1b。由于 h_t 现在由 h_{t-1} 和 X_t 共同决定，我们得到 $\mathcal{H}_t \subset \mathfrak{S}(h_{t-1}, X_t)$ 。利用两个随机变量的联合 σ -代数的定义

$$\mathfrak{S}(h_{t-1}, X_t) = \mathfrak{S}(\mathfrak{S}(h_{t-1}) \cup \mathfrak{S}(X_t)) = \mathfrak{S}(\mathcal{H}_{t-1} \cup \mathcal{I}_t),$$

之前的包含关系变为

$$\mathcal{H}_t \subset \mathfrak{S}(\mathcal{H}_{t-1} \cup \mathcal{I}_t), \quad (17.1.3)$$

其中 \mathcal{I}_t 表示由输入变量 X_t 生成的输入信息。

命题17.1.2 考虑一个具有给定初始状态 h_0 并由如(17.1.2)中所示的随机过程 X_t 驱动的动力系统。那么

$$\mathcal{H}_t \subset \mathfrak{S}(\mathcal{I}_1 \cup \dots \cup \mathcal{I}_t). \quad (17.1.4)$$

证明：证明过程源于对公式(17.1.3)和练习17.10.1的反复应用，这意味着

$$\begin{aligned} \mathcal{H}_t &\subset \mathfrak{S}(\mathcal{H}_{t-1} \cup \mathcal{I}_t) \\ &\subset \mathfrak{S}(\mathfrak{S}(\mathcal{H}_{t-2} \cup \mathcal{I}_{t-1}) \cup \mathcal{I}_t) = \mathfrak{S}(\mathcal{H}_{t-2} \cup \mathcal{I}_{t-1} \cup \mathcal{I}_t). \end{aligned}$$

通过归纳法，我们得到

$$\mathcal{H}_t \subset \mathfrak{S}(\mathcal{H}_0 \cup \mathcal{I}_1 \cup \dots \cup \mathcal{I}_t).$$

由于 h_0 是一个给定的常数，那么 $\mathcal{H}_0 = \{\emptyset, \Omega\}$ ，因此

$$\mathcal{H}_0 \cup \mathcal{I}_1 \cup \dots \cup \mathcal{I}_t = \mathcal{I}_1 \cup \dots \cup \mathcal{I}_t.$$



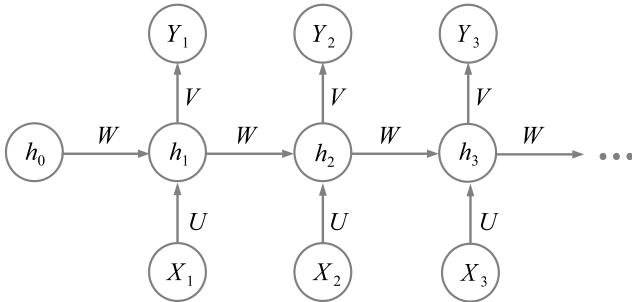


图17.2：循环神经网络的“多对多”配置。

注记17.1.3 值得注意因变量的使用。展开后我们有

$$h_t = f(h_{t-1}, X_t) = f(f(h_{t-2}, X_{t-1}), X_t) = g(h_{t-2}, X_{t-1}, X_t),$$

其中 \$g\$ 是可测的。由于 \$h_t\$ 由 \$h_{t-2}, X_{t-1}\$ 和 \$X_t\$ 确定，根据命题D.5.1 我们得到 \$\mathfrak{S}(h_t) \subset \mathfrak{S}(h_{t-2}, X_{t-1}, X_t)\$。

17.2 循环神经网络

循环神经网络可以通过几种方式引入。

1. 首先，它可被视为一个状态系统，其具有这样的属性：除初始状态外，每个状态都会产生一个结果，见图17.2。因此，状态 \$h_t\$ 从前一个状态 \$h_{t-1}\$ 以及当前输入 \$X_t\$ 获取信息，并产生一个结果 \$Y_t\$ 以及给下一个状态 \$h_{t+1}\$ 的一个输入。状态 \$h_t\$ 被称为循环神经网络的第 \$t\$ 个隐藏状态。\$X_t\$ 和 \$Y_t\$ 分别是第 \$t\$ 个输入和输出。

2. 我们也可以将循环神经网络（RNN）视为一系列普通的前馈神经网络，\$X_t \rightarrow h_t \rightarrow Y_t\$，（其输入层为 \$X_t\$，隐藏层为 \$h_t\$，输出层为 \$Y_t\$），它将信息从一个隐藏层传递到下一个隐藏层。

循环神经网络（RNN）中前向传播的标准方程如下所示

$$h_t = \tanh(Wh_{t-1} + UX_t + b) \quad (17.2.5)$$

$$Y_t = Vh_t + c. \quad (17.2.6)$$

因此，隐藏状态的值介于 -1 和 1 之间，并且输出是当前隐藏状态的仿射函数。

转移函数 f 是双曲正切函数与仿射函数的复合。参数 $\theta = (W, U, b, c)$ 由两个矩阵和两个向量组成。矩阵 W 和 U 分别表示隐藏状态到隐藏状态的转移以及输入到隐藏状态的转移。矩阵 V 表示从隐藏状态到输出的转移，而 b 和 c 是偏置向量。

17.3 循环神经网络中的信息

我们首先介绍一些术语。由 X_t 生成的输入信息用 \mathcal{I}_t 表示。隐藏信息由隐藏状态 $\mathcal{H}_t = \mathcal{G}(h_t)$ 生成，输出信息由输出 $\mathcal{E}_t = \mathcal{G}(Y_t)$ 生成。本节讨论这些 σ -代数之间的关系。

第一个结果展示了输出信息如何与输入和隐藏信息相关联。

命题17.3.1 考虑一个具有给定初始状态 h_0 和满足前向传播(17.2.5)-(17.2.6)的输入 X_t 的循环神经网络。那么

$$\mathcal{E}_t \subset \mathcal{H}_t \subset \mathcal{G}(\mathcal{I}_1 \cup \dots \cup \mathcal{I}_t). \quad (17.3.7)$$

证明：根据关系式(17.2.6)，变量 Y_t 由 h_t 确定，所以根据命题D.5.1 我们有 $\mathcal{G}(Y_t) \subset \mathcal{G}(h_t)$ ，这表明了(17.3.7)的第一个包含关系。第二个包含关系由命题17.1.2得出。 ■

考虑在每个时间步插入系统的信息 \mathcal{I}_t 是“新的”这种情况是很自然的，即输入变量 $X_t, t \in \{1, 2, \dots\}$ 是独立的情况。我们得到以下历史独立性属性。

命题17.3.2 考虑一个具有给定初始状态 h_0 和满足前向传播(17.2.5)-(17.2.6)的独立输入 X_t 的递归神经网络。那么两个 σ 代数 \mathcal{E}_{t-1} 和 \mathcal{H}_{t-1} 都与 \mathcal{I}_t 独立(即， Y_{t-1} 和 h_{t-1} 与 X_t 独立)。

证明：由于 X_t 与 X_1, \dots, X_{t-1} 相互独立，那么 \mathcal{I}_t 是一个与 $\{\mathcal{I}_1, \dots, \mathcal{I}_{t-1}\}$ 相互独立的信息体。然后它们生成的西格玛代数 $\mathcal{G}(\mathcal{I}_1 \cup \dots \cup \mathcal{I}_{t-1})$ 也与 \mathcal{I}_t 相互独立。由于根据命题17.3.1我们有 $\mathcal{H}_{t-1} \subset \mathcal{G}(\mathcal{I}_1 \cup \dots \cup \mathcal{I}_{t-1})$ ，所以可得 \mathcal{H}_{t-1} 与 \mathcal{I}_t 相互独立。 \mathcal{E}_{t-1} 与 \mathcal{I}_t 相互独立这一事实可由包含关系 $\mathcal{E}_t \subset \mathcal{H}_t$ 推出，见命题17.3.1。 ■

为了使用熵来评估信息，我们将假设变量 h_t, X_t 和 Y_t 具有相同的向量维度，因此 U, V 和 W 成为方阵。

我们进一步假设它们也是非奇异的。在这种情况下，我们可以从(17.2.6)中解出 h_t 为 $h_t = V^{-1}(Y_t - c)$ 。根据命题D.5.1，这意味着 $\mathcal{H}_t \subset \mathcal{E}_t$ 。由于在命题17.3.1中已经证明了反向包含关系，所以可以得出 $\mathcal{H}_t = \mathcal{E}_t$ 。

命题17.3.3 假设变量 h_t, X_t 和 Y_t 具有相同的向量维数，并且 $(X_t)_{t \geq 1}$ 是独立随机变量。

(a) 一个隐藏状态相对于前一个状态的条件熵满足

$$H(h_t|h_{t-1}) < H(X_t) + \ln |\det U|.$$

(b) 一个输出相对于前一个隐藏状态的条件熵满足

$$H(Y_t|h_{t-1}) < H(X_t) + \ln |\det(UV)|.$$

证明：(a) 设 $a_t = Wh_{t-1} + UX_t + b$ ，所以 $h_t = \tanh a_t$ 。我们使用命题12.1.2来计算条件熵

$$\begin{aligned} H(h_t|h_{t-1}) &= H(\tanh(a_t)|h_{t-1}) \\ &= H(a_t|h_{t-1}) - \mathbb{E}^{P_{a_t|h_{t-1}}} [\ln |\det J_{\tanh^{-1}}(h_t)|]. \end{aligned} \quad (17.3.8)$$

接下来我们将计算雅可比矩阵。由于 $a_t = \tanh^{-1}(h_t)$ ，并且 \tanh^{-1} 作用于分量，那么 $a_t^j = \tanh^{-1}(h_t^j)$ ，所以

$$J_{\tanh^{-1}}(h_t) = \left(\frac{\partial a_t^i}{\partial h_t^k} \right)_{j,k} = \left(\frac{\partial \tanh^{-1}(h_t^i)}{\partial h_t^k} \right)_{j,k} = \frac{\delta_{ik}}{1 - (h_t^i)^2}.$$

由于雅可比矩阵是对角矩阵，因此

$$\det J_{\tanh^{-1}}(h_t) = \prod_i \frac{1}{1 - (h_t^i)^2}.$$

将其代回(17.3.8)式可得

$$\begin{aligned} H(h_t|h_{t-1}) &= H(a_t|h_{t-1}) + \mathbb{E}^{P_{a_t|h_{t-1}}} [\ln \prod_i |1 - (h_t^i)^2|] \\ &< H(a_t|h_{t-1}), \end{aligned} \quad (17.3.9)$$

因为 $1 - (h_t^i)^2 \in (0, 1)$ 。接下来我们将条件熵 $H(a_t | h_{t-1})$ 计算为

$$\begin{aligned} H(a_t|h_{t-1}) &= H(Wh_{t-1} + UX_t + b|h_{t-1}) \\ &= H(UX_t|h_{t-1}) = H(UX_t) = H(X_t) + \ln |\det U|. \end{aligned}$$

我们考虑到 X_t 和 h_{t-1} 是独立的，然后使用命题12.1.2。将结果代回(17.3.9)得到了 (a) 部分所需的不等式。

(b) 使用 $Y_t = Vh_t + c$ ，类似的计算可得

$$\begin{aligned} H(Y_t|h_{t-1}) &= H(h_t|h_{t-1}) + \ln |\det V| \\ &< H(X_t) + \ln |\det U| + \ln |\det V|, \end{aligned}$$

这里我们使用了 (a) 部分的不等式。 ■

注记17.3.4 如果矩阵 U 和 V 的元素足够小，那么它们行列式的绝对值小于1。在这种正则化条件下，我们得到以下上界：

$$H(h_t|h_{t-1}) < H(X_t), \quad H(Y_t|h_{t-1}) < H(X_t).$$

这表明驱动信号 X_t 的熵大于由隐藏状态历史条件下的输出和隐藏状态的熵。

我们还注意到，根据推论3.5.2，条件熵 $H(h_t | h_{t-1})$ 由输入方差 $Var X_t$ 控制，即 X_t 的小方差意味着条件熵 $H(h_t | h_{t-1})$ 的值小。

17.4 损失函数

考虑一个具有 T 个单元的循环神经网络 (RNN)。我们将分别用 (X_1, \dots, X_T) 和 (Y_1, \dots, Y_T) 表示输入和输出。在图17.2 中所示的“多对多”RNN的情况下，目标由序列 (Z_1, \dots, Z_T) 给出。RNN的损失函数 L 表示序列 (Y_1, \dots, Y_T) 和 (Z_1, \dots, Z_T) 之间的“距离”。这可以被视为总和

$$L = \sum_{t=1}^T L_t,$$

其中个体损失函数 L_t 衡量 Y_t 相对于 Z_t 的接近程度。

对于个体损失函数 L_t ，我们有几种选择。在随机变量的情况下，个体损失可以是均方误差

$$L_t = \mathbb{E}[|Y_t - Z_t|^2],$$

要么是库尔贝克-莱布勒散度

$$L_t = D_{KL}(p_{(X_1, \dots, X_t), Z_t} || p_{\theta; (X_1, \dots, X_t), Z_t})$$

要么是交叉熵

$$L_t = S(p_{(X_1, \dots, X_t), Z_t}, p_{\theta; (X_1, \dots, X_t), Z_t}) = -\mathbb{E}^{p_{(X_1, \dots, X_t), Z_t}} [\log p_{\theta; (X_1, \dots, X_t), Z_t}],$$

其中 $\theta = (U, V, W, b, c)$ 是模型的参数。

在变量为连续变量的情况下，我们可以选择个体损失为欧几里得距离的平方

$$L_t = \frac{1}{2}(Y_t - Z_t)^2.$$

无论考虑哪种损失函数，计算关于参数 $\nabla_{\theta} L$ 的梯度都是一项昂贵的操作，并且在许多情况下会导致梯度问题，正如我们稍后将看到的。

17.5 时间反向传播

通过RNN的前向传播由公式(17.2.5)-(17.2.6)给出，这些公式提供了 h_t, Y_t 的值以及损失 L_t 。现在，为了最小化损失函数，应用梯度下降法就足够了，这需要计算梯度 $\nabla_{\theta} L$ 。计算梯度的方法称为时间反向传播，它是前几章研究的反向传播方法的一种变体。然而，现在这更复杂了，因为它是每个单独时间步的反向传播和跨时间的反向传播之间的一种组合。

由于一般情况的阐述包含复杂的符号，可能会造成潜在的混淆，我们将通过一个只有两步的循环神经网络（RNN）来举例说明时间反向传播的方法，见图17.3。在这种情况下，有两个输入 X_1, X_2 ，两个输出 Y_1, Y_2 ，以及两个隐藏状态 h_1, h_2 。为了简单起见，我们将它们视为一维连续变量。前向传播方程可以写成

$$\begin{aligned} a_1 &= Wh_0 + UX_1 + b & a_2 &= Wh_1 + UX_2 + b \\ h_1 &= \tanh a_1 & h_2 &= \tanh a_2 \\ Y_1 &= Vh_1 + c & Y_2 &= Vh_2 + c. \end{aligned}$$

这些公式用于计算损失函数。为了简单起见，我们考虑

$L_t = \frac{1}{2}(Y_t - Z_t)^2$ ，因此损失函数变为

$$L = L_1 + L_2 = \frac{1}{2}(Y_1 - Z_1)^2 + \frac{1}{2}(Y_2 - Z_2)^2 = \frac{1}{2}\|Y - Z\|_{Eu}^2.$$

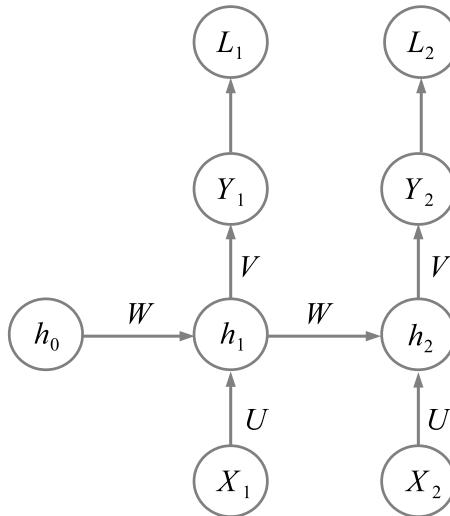


图17.3：具有两步的循环神经网络配置。

我们需要计算五个梯度

$$\nabla_{\theta} L = \left(\frac{\partial L}{\partial W}, \frac{\partial L}{\partial V}, \frac{\partial L}{\partial U}, \frac{\partial L}{\partial b}, \frac{\partial L}{\partial c} \right).$$

我们将预先计算一些很快就会有用的导数。使用链式法则，我们有

$$\begin{aligned} \frac{\partial h_1}{\partial W} &= \frac{\partial}{\partial W} \tanh(a_1) = \operatorname{sech}^2 a_1 \frac{\partial a_1}{\partial W} \\ &= (1 - \tanh^2 a_1) h_0 = (1 - h_1^2) h_0. \end{aligned}$$

类似的计算可用于得到

$$\begin{aligned} \frac{\partial h_2}{\partial W} &= (1 - h_2^2) h_1 \\ \frac{\partial h_1}{\partial b} &= 1 - h_1^2 \\ \frac{\partial h_2}{\partial b} &= 1 - h_2^2. \end{aligned}$$

由于 L_1 对 h_1 的依赖仅通过 Y_1 完成，那么链式法则可得

$$\frac{\partial L_1}{\partial h_1} = \frac{\partial L_1}{\partial Y_1} \frac{\partial Y_1}{\partial h_1} = (Y_1 - Z_1)V.$$

同样地，

$$\frac{\partial L_2}{\partial h_2} = (Y_2 - Z_2)V.$$

我们也有

$$\begin{aligned}\frac{\partial h_2}{\partial h_1} &= \frac{\partial}{\partial h_1} \tanh a_2 = \operatorname{sech}^2 a_2 \frac{\partial a_2}{\partial h_1} \\ &= (1 - \tanh^2 a_2)W = (1 - h_2^2)W.\end{aligned}$$

链式法则的另一个应用给出

$$\begin{aligned}\frac{\partial h_1}{\partial U} &= \frac{\partial}{\partial U} \tanh a_1 = (1 - h_1^2) \frac{\partial a_1}{\partial U} = (1 - h_1^2)X_1 \\ \frac{\partial h_2}{\partial U} &= \frac{\partial}{\partial U} \tanh a_2 = (1 - h_2^2) \frac{\partial a_2}{\partial U} = (1 - h_2^2)X_2.\end{aligned}$$

现在我们已经做好充分准备来计算损失函数的梯度。我们将从关于 V 的梯度开始。由于损失 L_t 通过 Y_t 依赖于 V ，我们有

$$\begin{aligned}\frac{\partial L}{\partial V} &= \frac{\partial L_1}{\partial V} + \frac{\partial L_2}{\partial V} \\ &= \frac{\partial L_1}{\partial Y_1} \frac{\partial Y_1}{\partial V} + \frac{\partial L_2}{\partial Y_2} \frac{\partial Y_2}{\partial V} \\ &= (Y_1 - Z_1) \frac{\partial}{\partial V} (Vh_1 + c) + (Y_2 - Z_2) \frac{\partial}{\partial V} (Vh_2 + c) \\ &= (Y_1 - Z_1)h_1 + (Y_2 - Z_2)h_2 = \sum_t (Y_t - Z_t)h_t.\end{aligned}$$

关于 c 的导数计算方式类似

$$\begin{aligned}\frac{\partial L}{\partial c} &= \frac{\partial L_1}{\partial c} + \frac{\partial L_2}{\partial c} \\ &= \frac{\partial L_1}{\partial Y_1} \frac{\partial Y_1}{\partial c} + \frac{\partial L_2}{\partial Y_2} \frac{\partial Y_2}{\partial c} \\ &= (Y_1 - Z_1) \frac{\partial}{\partial c} (Vh_1 + c) + (Y_2 - Z_2) \frac{\partial}{\partial c} (Vh_2 + c) \\ &= (Y_1 - Z_1) + (Y_2 - Z_2) = \sum_t (Y_t - Z_t).\end{aligned}$$

在计算关于 W 的梯度时，我们考虑到 L_1 仅通过 h_1 依赖于 W ，而 L_2 通过 h_1 和 h_2 两者依赖于 W （有两条边包含 W ）。链式法则的应用给出

$$\begin{aligned}
\frac{\partial L}{\partial W} &= \frac{\partial L_1}{\partial W} + \frac{\partial L_2}{\partial W} \\
&= \frac{\partial L_1}{\partial h_1} \frac{\partial h_1}{\partial W} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial W} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial h_1} \frac{\partial h_1}{\partial W} \\
&= (Y_1 - Z_1)V(1 - h_1^2)h_0 + (Y_2 - Z_2)V(1 - h_2^2)h_1 \\
&\quad + (Y_2 - Z_2)V(1 - h_2^2)W(1 - h_1^2)h_0.
\end{aligned}$$

由于存在两条涉及 U 的垂直边，因此可以应用类似的计算来得到

$$\begin{aligned}
\frac{\partial L}{\partial U} &= \frac{\partial L_1}{\partial U} + \frac{\partial L_2}{\partial U} \\
&= \frac{\partial L_1}{\partial h_1} \frac{\partial h_1}{\partial U} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial U} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial h_1} \frac{\partial h_1}{\partial U} \\
&= (Y_1 - Z_1)V(1 - h_1^2)X_1 + (Y_2 - Z_2)V(1 - h_2^2)X_2 \\
&\quad + (Y_2 - Z_2)V(1 - h_2^2)W(1 - h_1^2)X_1.
\end{aligned}$$

最后一个梯度是相对于偏差 b 计算的

$$\begin{aligned}
\frac{\partial L}{\partial b} &= \frac{\partial L_1}{\partial b} + \frac{\partial L_2}{\partial b} \\
&= \frac{\partial L_1}{\partial h_1} \frac{\partial h_1}{\partial b} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial b} + \frac{\partial L_2}{\partial h_2} \frac{\partial h_2}{\partial h_1} \frac{\partial h_1}{\partial b} \\
&= (Y_1 - Z_1)V(1 - h_1^2) + (Y_2 - Z_2)V(1 - h_2^2) \\
&\quad + (Y_2 - Z_2)V(1 - h_2^2)W(1 - h_1^2).
\end{aligned}$$

对于具有两个以上隐藏状态的循环神经网络，可以得到类似的公式。

17.6 梯度问题

循环神经网络中处理梯度时的以下困难首先由霍赫赖特[55]以及本吉奥等人[14,15]指出。

梯度消失问题 我们从上文的梯度公式中注意到，关于 W, U 和 b 的梯度涉及与矩阵 W, V 以及因子 $(1 - h_2^2)$ 和 $(1 - h_1^2)$ 相关的乘积。鉴于 $h_t = \tanh a_t \in (-1, 1)$ ，那么 $1 - h_2^2 \in (0, 1)$ 。因此，这类因子的乘积会使梯度减小。循环神经网络（RNN）越长，涉及 $(1 - h_t^2)$ 的乘积就越多，梯度也就越小。

梯度公式中涉及矩阵 W 的因子来自导数 $\frac{\partial h_2}{\partial h_1}$ 。在长度为 T 的循环神经网络（RNN）的情况下，存在更多涉及类型为 $\frac{\partial h_t}{\partial h_{t-1}}$ 的导数的乘积，这将产生幂 $\overline{W^{T-1}}$ 的效果。如果矩阵 W 具有特征值 $|\lambda_i| < 1$ ，那么幂 W^{T-1} 将具有特征值 $|\lambda_i|^{T-1} < 1$ 。这可从特征值分解 $W = MDM^t$ 得出，这意味着 $W^{T-1} = MD^{T-1}M^t$ ，其中 D 是沿对角线具有特征值的对角形式。由于对于较大的 T 我们有 $|\lambda_i|^{T-1} \rightarrow 0$ ，那么 $D^{T-1} \rightarrow O$ ，因此当 RNN 的长度增加时 W^{T-1} 趋于零矩阵，见附录中的命题 G.1.2。

先前的讨论可以归纳在梯度消失问题的概念之下。针对这个问题有几种补救措施，这些措施可以在不同程度上防止梯度缩小。

- (i) 该问题的一种解决方案是改变激活函数。由于因子 $(1 - h_t^2)$ 来自导数 $\tanh'(a_t)$ ，一种想法是用导数并非处处小于1的激活函数来替代，比如 $ReLU$ ，它在正激活值上恰好等于1。
- (ii) 另一种解决方案是将权重初始化为单位矩阵 $W = \mathbb{I}$ 。这一事实将防止权重过快缩小至零，因为对于更多次迭代，幂 W^p 的特征值将更接近1。
- (iii) 梯度消失问题最可靠的解决方法是采用一种涉及“门控单元”的新颖架构，比如长短期记忆（LSTM）或门控循环单元（GRU）。我们将在下一节讨论这种类型的架构。

梯度爆炸问题 我们现在假设矩阵 W 有一个特征值满足 $|\lambda_i| > 1$ 。由于 $W^p = M D^p M^t$ ，那么对于 p 很大时， D^p 的元素 λ_i^p 趋于无穷大。因此， W^p 的一些元素将趋于无穷大，这一事实导致了梯度爆炸问题。一种解决方法是将权重 W 初始化为单位矩阵，希望防止权重过早爆炸。另一种有用的技术是梯度裁剪，它基于当梯度变得太大时将其重新缩放回来，见米科洛夫[84]和帕斯卡努等人[96]。

17.7 长短期记忆单元

长短期记忆网络（LSTM）由霍赫雷特（Hochreiter）和施密德胡伯（Schmidhuber）于1997年提出[56]。这是一种递归神经网络（RNN），包含能够学习长期依赖关系的特殊单元。它们使用门控概念，门控选择性地允许信息通过一个Sigmoid层和逐点乘法。LSTM的功能涉及三种类型的门：遗忘门、更新门和输出门。

它基于引入一个内部单元状态， C_t ，并在每个单元中引入一个内循环。每个门的描述如下。

1. 遗忘门是一个用于遗忘不相关历史信息的 sigmoid 层。它由下式定义

$$f_t = \sigma(W_f h_{t-1} + U_f X_t + b_f),$$

其中 W_f, U_f 是矩阵， b_f 是偏置向量。由于 σ 是逻辑 sigmoid 函数，那么 $f_t \in (0, 1)$ 。这表示将被遗忘的过去状态的比例。它取决于上一状态 h_{t-1} 以及当前输入 X_t 。

2. 更新门选择性地更新内部单元状态值， C_t ，如下所示

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t,$$

其中 i_t 是由 sigmoid 层定义的缩放因子

$$i_t = \sigma(W_i h_{t-1} + U_i X_t + b_i),$$

其矩阵为 W_i, U_i ，偏置为 b_i ，并且

$$\tilde{C}_t = \tanh(W_c h_{t-1} + U_c X_t + b_c)$$

表示一个可以添加到内部状态的候选者，它属于区间 $(-1, 1)$ 。

乘积 $i_t \tilde{C}_t$ 表示候选者的一部分，用于更新内部状态 C_t 。项 $f_t C_{t-1}$ 表示从内部状态值中遗忘分数 f_t 后剩余的量。

3. 输出门提供隐藏状态的值

$$h_t = o_t \tanh(C_t),$$

作为一个决定输出量的因素与某个值的乘积，

$$o_t = \sigma(W_o h_{t-1} + U_o X_t + b_o),$$

以及通过对内部状态 C_t 应用双曲正切函数得到的介于 -1 和 1 之间的一个状态。

在RNN中使用LSTM单元有助于解决梯度消失问题，因为LSTM单元允许梯度无变化地流动。然而，它们仍然存在梯度爆炸问题。还有其他架构，如窥视孔LSTM[42]、窥视孔卷积LSTM[113]、GRU单元[24]等，也有类似的作用。

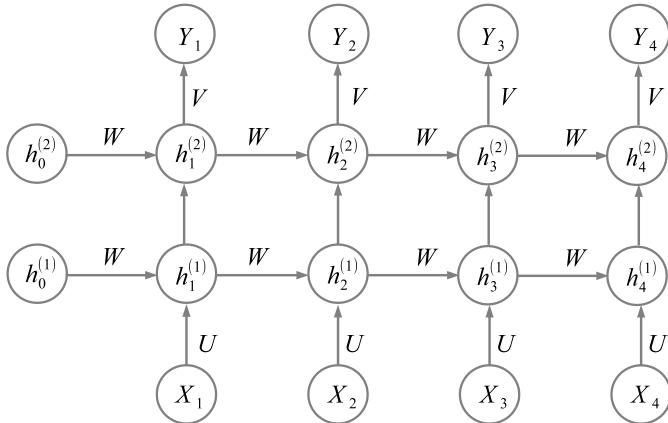


图17.4：具有 $T = 4$ 和 $L = 2$ 的深度循环神经网络。

17.8 深度循环神经网络

我们已经看到，循环神经网络可以看作是一系列单隐藏层参数共享的前馈神经网络，其隐藏层通过转移函数交换信息。在Graves [50]中首次出现了在循环神经网络中引入深度的想法，随后不久Pascanu等人[95]也提出了类似想法。

深度循环神经网络（RNN）是通过展开深度参数共享前馈神经网络得到的。在这种情况下，RNN的隐藏状态 $h_t^{(\ell)}$ 有两个索引。较低的索引 t 表示状态在时间上的位置，而较高的索引 ℓ 表示状态所属的层数，见图17.4。维度为 $T \times L$ 的深度RNN意味着一个具有 L 个水平层、 $1 \leq \ell \leq L$ 和 T 个垂直展开，且 $1 \leq t \leq T$ 的架构。

训练深度循环神经网络比训练简单循环神经网络更困难且成本更高。我们将为此提供一个支持性论据。考虑图17.4中给出的具有2层和4次循环的深度循环神经网络。在找到损失函数相对于 W 的梯度的过程中，除了其他导数之外，我们还需要计算导数 $\partial Y_4 / \partial W$ 。由于 Y_4 通过中间变量 $h_t^{(\ell)}$, $1 \leq t \leq 4, 1 \leq \ell \leq 2$ 依赖于 W ，应用链式法则会产生涉及两种类型因子的乘积： $\frac{\partial h_t^{(\ell)}}{\partial h_{t-1}^{(\ell)}}$ 和 $\frac{\partial h_t^{(2)}}{\partial h_t^{(1)}}$ 。正如我们在17.5节的计算中所看到的，第一个因子引入了与 W 的乘法。现在我们来看第二个因子。

如果我们假设这些层通过逻辑 sigmoid 函数相关联，如下所示

$$h_t^{(2)} = \sigma(W_h h_t^{(1)} + b_h),$$

那么利用 sigmoid 函数的性质，我们可以得到

$$\begin{aligned} \frac{\partial h_t^{(2)}}{\partial h_t^{(1)}} &= W_h \sigma'(W_h h_t^{(1)} + b_h) = W_h \sigma(W_h h_t^{(1)} + b_h)(1 - \sigma(W_h h_t^{(1)} + b_h)) \\ &= W_h h_t^{(2)}(1 - h_t^{(2)}). \end{aligned}$$

由于 $\sigma' \leq 1/4$ ，该项也会对导数产生收缩效应。RNN 越深，这种效应就越明显。上述所有效应都会导致更明显的梯度消失问题。

17.9 总结

循环神经网络（RNN）是专门用于处理诸如音频和视频等序列数据的神经网络。它们在语音识别、手写生成、文本生成、机器翻译、图像字幕、无约束手写识别等实际应用中已被证明非常成功。

RNN 是由具有相同架构的参数共享前馈神经网络的有限序列组成，其隐藏层通过转移函数交换信息。RNN 通过一种反向传播方法的变体进行训练，称为随时间反向传播，这比常规的前馈神经网络训练成本更高。

普通的香草循环神经网络（RNN）存在两个问题：梯度消失和梯度爆炸。针对这些问题有几种部分补救措施。梯度爆炸问题可以通过梯度裁剪和单位矩阵初始化来改善，而梯度消失问题可以通过采用包括门控单元的新型 RNN 架构来解决，如长短期记忆网络（LSTM）、门控循环单元（GRU）等。

包含多个水平层的循环神经网络被称为深度循环神经网络。它们在从原始序列数据中提取更复杂模式的问题中很有用。

17.10 练习

练习 17.10.1 (a) 设 $\mathcal{G}_1, \mathcal{G}_2$ 和 \mathcal{G}_3 为三个西格玛代数。证明

$$\mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3) = \mathfrak{S}(\mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2) \cup \mathcal{G}_3).$$

(b) 提出并证明一个推广结论。

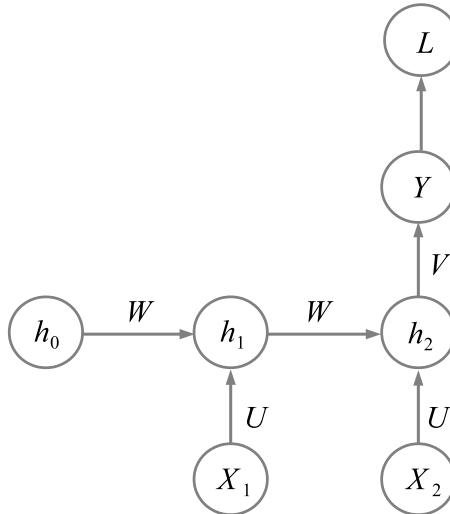


图17.5：用于练习17.10.3的“2对1”循环神经网络。

练习17.10.2 (a) 考虑矩阵

$$W = \begin{pmatrix} \frac{1}{10} & \frac{2}{10} \\ \frac{3}{10} & -\frac{4}{10} \end{pmatrix}$$

证明 $\lim_{n \rightarrow \infty} W^n = O_2$ ，其中 O_2 表示 2×2 零矩阵。

(b) 设 A 为 $k \times k$ 对称矩阵，并用 $\rho(A) = \max_{1 \leq i \leq k} |\lambda_i|$ 表示其谱半径，其中 λ_i 表示 A 的特征值。考虑矩阵 $W = \frac{1}{1+\rho(A)} A$ 。证明 $\lim_{n \rightarrow \infty} W^n = O_k$ 。

练习17.10.3 考虑一个具有两个一维输入 X_1 、 X_2 和一个输出 Y 的循环神经网络，见图17.5。损失函数为 $L = \frac{1}{2}(Y - Z)^2$ ，其中 Z 表示目标值。求在这种情况下通过时间反向传播算法的方程。

练习17.10.4 考虑图17.5中给出的循环神经网络，并假设输入 X_1, X_2 是随机变量，初始状态 h_0 是给定的，目标 Z 是随机变量。用 $\mathcal{E} = \mathfrak{S}(Y)$ 表示输出 σ -代数。如果 $\mathfrak{S}(Z) \subset \mathcal{E}$ ，我们就说 Z 是可学习的。

以下哪项总是正确的：

- (a) 如果 Z 比 $\mathcal{S}(Z) \subset \mathcal{S}(\mathcal{I}_1 \cup \mathcal{I}_2)$ 更易于学习。
- (b) 如果 $\mathcal{S}(Z) \subset \mathcal{S}(\mathcal{I}_1 \cup \mathcal{I}_2)$ ，那么 Z 是可学习的。

练习17.10.5 考虑一个一维动力系统，其状态更新如下

$$h_n = f(h_{n-1}; \theta), \quad n \geq 1,$$

其中转移函数是 $f(x; \theta) = \tanh(\theta x)$ ，且 $|\theta| < 1$ 。从长远来看，求系统的隐藏状态 $\lim_{n \rightarrow \infty} h_n$ 。

练习17.10.6 与练习17.10.5中的问题相同，但将转移函数替换为逻辑 sigmoid， $f(x; \theta) = \sigma(\theta x)$ 。

练习17.10.7 与练习17.10.5中的假设相同，但将转移函数替换为正弦函数， $f(x; \theta) = \sin(\theta x)$ 。证明动力系统的长期行为取决于参数 θ 和初始状态 h_0 。



第18章 分类

在分类问题中，神经网络必须能够对聚类进行分类，即给每个聚类分配一个标签。这些标签可以是自然数、空间中的点或向量，属于一个标签空间。分类过程等同于能够学习一个“聚类分裂函数”或决策图。训练集为每个聚类提供标签。这种分配定义了一个决策图。网络将能够通过学习这个决策图来对测试数据进行分类，即确定测试点属于哪个聚类。

18.1 等价关系

考虑超立方体， $I_n = [0, 1]^n$ 。任何子集 $\mathcal{S} \subset I_n \times I_n$ 都称为 I_n 上的一个关系。以下性质很快就会有用：

- (i) \mathcal{S} 若包含超立方体对角线 $\{(x, x); x \in I_n\}$ ，则称其为自反的。这表明若 $\forall x \in I_n$ ，那么 $(x, x) \in \mathcal{S}$ 。
- (ii) \mathcal{S} 若关于对角线 $\{(x, x); x \in I_n\}$ 对称，则称其为对称的。这意味着若 $(x, y) \in \mathcal{S}$ ，那么也有 $(y, x) \in \mathcal{S}$ 。
- (iii) 如果 $(x, y) \in \mathcal{S}$ 且 $(y, z) \in \mathcal{S}$ ，那么 $(x, z) \in \mathcal{S}$ ，则称 \mathcal{S} 是可传递的。此属性的几何解释是， \mathcal{S} 对于其顶点之一位于对角线上的矩形是封闭的。更确切地说，如果 $(x, y) \in \mathcal{S}$ ，那么存在一个唯一的矩形，其一个顶点位于此点，另一个顶点位于 (y, y) ，见图18.2 a。可传递性归结为这样一个事实，即如果矩形的三个顶点 $(x, y), (y, y), (y, z)$ 属于集合 \mathcal{S} ，那么第四个顶点 (x, z) 也属于 \mathcal{S} 。

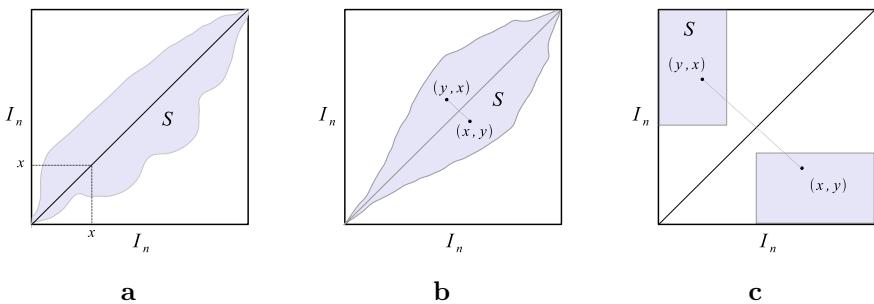


图18.1：a. 自反、非对称关系。b. 自反且对称关系。c. 对称非自反关系。

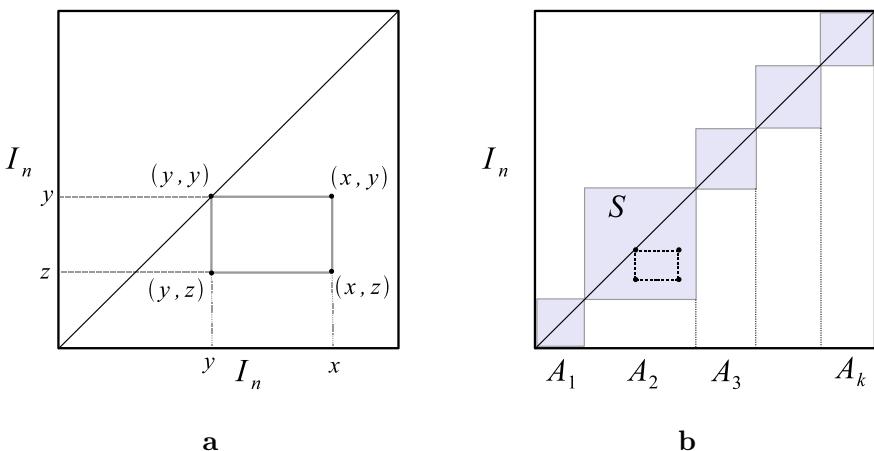


图18.2：a. 矩形规则：若 $(x, y), (y, y), (y, z) \in \mathcal{S}$ ，则 $(x, z) \in \mathcal{S}$ 。b. 等价关系 \mathcal{S} ，它对 I_n 进行有限划分。

传递关系的一个例子是 $[0, 1]$ 中有理数的格，由 $\mathcal{S} = (I_n \times I_n) \cap (\mathbb{Q} \times \mathbb{Q})$ 给出，读者可以很容易地验证前面的矩形性质。

先前的关系属性是相互排斥的，见图18.1 a、b和 c。

一个自反、对称且传递的关系 \mathcal{S} 被称为等价关系。等价关系的示例见图18.2 b。在这种情况下，集合 \mathcal{S} 是沿对角线的正方形内部的并集。

如果 $(x, y) \in \mathcal{S}$ ，则在关系 \mathcal{S} 下，两个点 $x, y \in I_n$ 被称为等价的。通常，我们写作 $x \sim y$ 。所有与给定的 x 等价的点用 $C_x = \{y \in I_n; x \sim y\}$ 表示，并且被称为 x 的等价类。所有等价类的集合，用 I_n / \sim 表示，被称为商集。

集合 I_n 的一个划分是具有以下性质的子集 $\{A_i\}_i$ 的一个集合：

- (i) 对于所有 i , $A_i \neq \emptyset$;
- (ii) 对于 $i \neq j$, $A_i \cap A_j = \emptyset$;
- (iii) $\bigcup A_i = I_n$ 。

如果指标集合 i 是有限的, 那么 $\{A_i\}_i$ 是一个有限划分。它们可用于将 I_n 中的点分类为 n 个不同的类。

划分与等价关系之间的关系由以下结果给出：

命题18.1.1 设 \sim 是 I_n 上的一个等价关系。那么存在 I_n 的一个划分 $\{A_i\}_i$, 使得:

- (i) 对于每个 $i, \forall x, y \in A_i$, 我们有 $x \sim y$;
- (ii) 对于 $\forall x, y \in I_n$ 和 $x \sim y$, 存在一个 i 使得 $x, y \in A_i$ 。

证明：该结果可以重新表述为，任何等价关系都会对 I_n 进行划分，且划分的元素就是该关系的等价类。设 C_x 是 x 的等价类。我们证明集合 $\{C_x\}_x$ 满足划分的性质。由于 $x \in C_x$, 那么显然 $C_x \neq \emptyset$ 且 $\bigcup_{x \in I_n} C_x = I_n$ 。很容易

看出 $x \sim y$ 与 $C_x = C_y$ 等价。假设我们有一个元素在两个不同类的交集中, 即 $z \in C_x \cap C_y$ 。那么 $z \in C_x$, 因此 $x \sim z$ 且 $z \in C_y$ 且 $z \sim y$ 。根据传递性 $x \sim y$, 这意味着 $C_x = C_y$, 矛盾。由此可知, 任意两个不同的类的交集为空。因此, 集合 $\{C_x\}_x$ 满足划分的性质。■

值得注意的是, 反过来也成立: 任何划分都定义了一个等价关系。如果 $\{A_i\}_i$ 是一个划分, 那么当且仅当存在一个 i 使得 $x, y \in A_i$ 时, 关系 $x \sim y$ 是 I_n 上的等价关系。

使用投影从等价关系 S 获得划分 $\{A_j\}_j$ 的方法如图18.2 b所示。这也提供了与给定划分相关联的等价关系 S 的可视化: 集合 S 由沿着由投影 A_j 构建的对角线排列的矩形的并集组成。这个集合 S 包含对角线 (是自反的), 是对称的, 并且满足矩形规则 (是传递的)。

在接下来的部分中, 我们将把给定的划分与不同的对象联系起来, 比如熵、决策函数、标签、决策图等。

18.2 划分的熵

在本节中, 我们将把熵的概念扩展到一个划分。为此, 我们将考虑一个概率空间 $(\Omega, \mathcal{F}, \mu)$ 和集合 Ω 的一个有限可测划分 $\mathcal{A} = (A_j)_{j \leq m}$, 即一个满足 $A_j \in \mathcal{F}$ 的划分。

度量 μ 可用于对集合 A_j 进行数值评估。关于概率测度 μ 的划分 \mathcal{A} 的熵定义如下

$$H(\mathcal{A}, \mu) = - \sum_{j=1}^m \mu(A_j) \ln \mu(A_j). \quad (18.2.1)$$

由于 $\mu(A_j) \in (0, 1)$ ，熵为正， $H(\mathcal{A}, \mu) > 0$ 。可以证明，当划分中的所有集合具有相同的测度 $\mu(A_1) = \dots = \mu(A_m) = \frac{1}{m}$ 时，划分 \mathcal{A} 的熵最大。

例18.2.1 我们假设 Ω 的每个元素 ω 都与一个非负数值标签相关联，例如权重或质量， $m(\omega)$ 。在这种情况下的概率测度是

$$\mu(A) = \frac{1}{M} \sum_{x \in A} \delta_x(A),$$

其中 $M = \mu(\Omega)$ 是 Ω 的总质量， δ_x 表示位于 x 的狄拉克测度。数字 $\mu(A)$ 给出了与集合 A 相对应的质量比例。熵(18.2.1)表示将集合 Ω 划分为质量不相等部分的不确定性。

示例18.2.2 设 $\Omega \subset \mathbb{R}^n$ 为有界博雷尔集。对于任何博雷尔集 $A \in \mathcal{B}(\Omega)$ ，我们定义概率测度 $\mu(A) = \frac{\lambda(A)}{\lambda(\Omega)}$ ，其中 λ 表示 Ω 上的勒贝格测度。在这种情况下，熵 (18.2.1) 表示将集合 Ω 划分为不等体积子集的不确定性。

示例18.2.3 设 μ 是关于可测空间 (Ω, \mathcal{F}) 上的测度 ν 绝对连续的测度。根据附录中的定理C.7 (见拉东 - 尼科迪姆定理)，存在一个非负可测函数 p 使得

$$\mu(A) = \int_A p(x) d\nu(x),$$

对于 Ω 中的任何可测集 A 。如果 p 是一个密度函数，即 $\int_{\Omega} p(x) d\nu(x) = 1$ ，那么 μ 就成为一个概率测度。与划分 \mathcal{A} 和测度 μ 相关的熵是

$$H(\mathcal{A}, \mu) = - \sum_{i=1}^m \int_{A_i} p(x) d\nu(x) \ln \left(\int_{A_i} p(x) d\nu(x) \right).$$

在测度成比例的特殊情况下，即 $\mu = c\nu$ ，其中 c 为常数，密度函数为 $p(x) = \frac{1}{\nu(\Omega)}$ ，并且之前的熵变为——

$$\begin{aligned}
 H(\mathcal{A}, \mu) &= -\sum_{i=1}^m \frac{1}{\nu(\Omega)} \nu(A_i) \ln \left(\frac{1}{\nu(\Omega)} \nu(A_i) \right) \\
 &= -\frac{1}{\nu(\Omega)} \sum_{j=1}^m \nu(A_j) (-\ln \nu(\Omega) + \ln \nu(A_j)) \\
 &= \ln \nu(\Omega) + \frac{1}{\nu(\Omega)} H(\mathcal{A}, \nu),
 \end{aligned}$$

这是关于两个成比例测度的划分 \mathcal{A} 的熵之间的一种关系。

18.3 决策函数

设 $\{A_1, \dots, A_k\}$ 为 I_n 的有限可测划分，即由 A_i 个博雷尔集构成的划分， $A_i \in \mathcal{B}(I_n)$ ，对所有 $i = 1, \dots, k$ 而言。决策函数是一个可测函数，它为划分中的每个类别赋予一个整数，即对于任何 $x \in A_j$ ， $f : I_n \rightarrow \mathbb{N}$, $f(x) = j$ 。我们可以将 j 视为与类别 A_j 相关联的标签。等价地， $f = \sum_{i=1}^k j \mathbf{1}_{A_j}$ ，其中 $\mathbf{1}_{A_j}$ 是集合 A_j 的指示函数。决策函数用于将数据分类到各个类别中。注意 $A_j = f^{-1}(j)$ ，见图18.3 b。

集合 $\{1, 2, \dots, k\}$ 被称为标签集，而包含标签集的空间，在这种情况下， \mathbb{R} 是标签空间。出于方便起见，标签被视为连续整数。

示例18.3.1 (情况 $k = 2$) 考虑在 \mathbb{R}^n 中由超平面 $\{w^T x + \theta = 0\}$ 分隔的两个可分离的点簇。然后，一个经典感知器可以使用决策函数 $f(x) = 1 + H(w^T x + \theta)$ 对每个点所属的簇进行二元决策。标签集是 $\{1, 2\}$ ，标签空间是 \mathbb{R} 。

示例18.3.2 考虑超立方体 I_n 中各点的某个属性，比如颜色。假设点可以具有 k 种可能的颜色。那么可以在 I_n 上定义一个等价关系：当且仅当两个点具有相同颜色时，它们是等价的。设 A_j 是 j 色的点的集合。那么集合 $\{A_j\}$ 构成 I_n 的一个划分，并且由 $f(A_j) = j$ 定义的从 I_n 到标签空间 $\{1, \dots, k\}$ 的映射 f 是一个分类规则。有时函数 f 被称为分类器。我们注意到在这个例子中，集合 A_j 不一定是博雷尔可测的。

下一个结果涉及决策函数的实现，见Cybenko [30]。由于在现实生活中，聚类并非完全可分离，该结果将考虑到这一点。

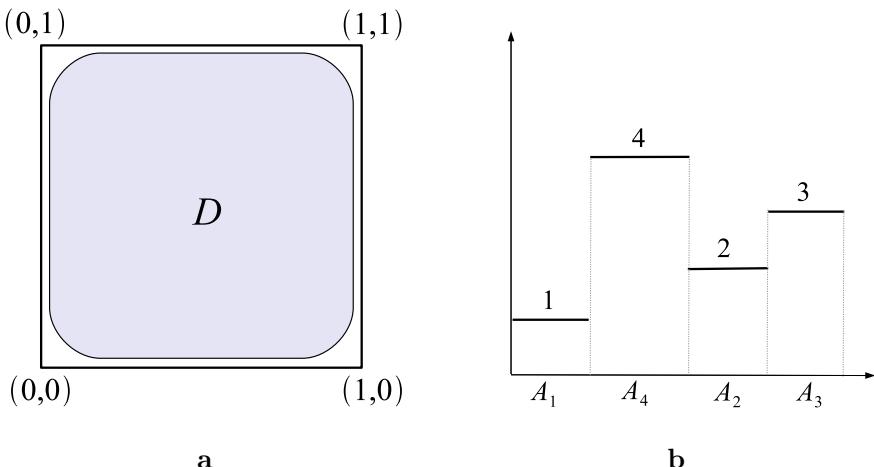


图18.3：a. I_2 中具有补集小勒贝格测度的子集 D 。b. 与一个划分相
关联的决策函数。

它表明具有单个内部层的神经网络可以实现任何决策函数，使得错误分类点的总勒贝格测度 λ 可以任意小，见图18.3 a。

命题18.3.3 设 f 为与 I_n 上的可测有限划分 $\{A_i\}_i$ 相关联的决策函数，且设 σ 为连续的S形函数。对于任意 $\epsilon > 0$ ，存在有限和

$G(x) = \sum_{j=1}^N \alpha_j \sigma(w_j^T x + \theta_j)$, $w_j^T \in \mathbb{R}^n$ 、 $\alpha_j, \theta_j \in \mathbb{R}$ 以及一个集合 $D \subset I_n$ ，
使得 $\lambda(D) \geq 1 - \epsilon$ 且

$$|G(x) - f(x)| < \epsilon, \quad \forall x \in D.$$

证明：根据卢津定理（见附录，C.8节），存在一个连续函数 $g : I_n \rightarrow \mathbb{R}$ 和一个集合 D ，使得对于所有 $x \in D$ ， $\lambda(D) > 1 - \epsilon$ 且 $g(x) = f(x)$ 。根据定理9.3.8（或定理9.3.6），前一种形式 $G(x)$ 的和在 $C(I_n)$ 中是稠密的，所以对于前面的 $g \in C(I_n)$ ，我们找到一个 $G(x)$ ，使得对于所有 $x \in I_n$ ， $|G(x) - g(x)| < \epsilon$ 。因此

$$|G(x) - f(x)| = |G(x) - g(x)| < \epsilon, \quad \forall x \in D.$$

■

请注意，这是一个存在性结果；函数 $G(x)$ 的实际构造（即找到权重 w_j, α_j 和阈值 θ_j ）是一个完全不同的问题。

注记18.3.4 对于每个决策函数，我们可以如下关联一个熵。给定一个有限划分 $\mathcal{A} = (A_i)_i$ 以及一个决策函数 f ，我们定义测度 μ 使得 $\mu(A_i) = \frac{f(A_i)}{\sum_i f(A_i)}$ ，并考虑熵 $H(\mathcal{A}, \mu)$ 为与划分 \mathcal{A} 和决策函数 f 相关联的熵。

18.4 独热向量决策映射

有时，用独热向量替换整数标签会更方便。例如，我们可以用独热向量 $e_1 = (1, 0, \dots, 0)^T, e_2 = (0, 1, 0, \dots, 0)^T$ 等，来代替标签1、2等。标签集由 $\{e_1, \dots, e_k\}$ 构成，而标签空间是 \mathbb{R}^n 。因此，我们得到了以下定义。

设 $\{A_1, \dots, A_k\}$ 是 I_n 的有限可测划分，即由 A_i 个博雷尔集构成的划分， $A_i \in \mathcal{B}(I_n)$ ，对所有 $i = 1, \dots, k$ 成立。独热向量决策映射是一个可测函数 $f : I_n \rightarrow \mathbb{R}^k$ ，它将一个独热向量与划分中的每个类别相关联，即对于任何 $x \in A_j$ ， $e_j = (0, \dots, 1, \dots, 0)^T$ ，有 $f(x) = e_j$ 。在这种情况下，与类别 A_j 相关联的标签是一个 k 维单位向量，并且所有这些标签向量在 \mathbb{R}^k 中构成一个基。

使用独热向量作为标签的优势是什么？在标签只是整数的情况下，集合 I_n 被映射到实线上，这为围绕某些给定整数的测试集提供了一些局部化。

在使用独热向量作为标签的情况下，集合 I_n 被映射到一个更高维的空间 \mathbb{R}^k 。这为测试集提供了更多空间，使其能够朝着线性独立的方向聚集，从而实现更好的类分离。

选择独热向量 e_j 作为标签只是为了方便。我们可以选择在 \mathbb{R}^k 中线性独立的任何其他 k 向量作为标签，最终将其组织成一个正交基。等效地，与其考虑 k 个独立向量，我们可以将标签视为 \mathbb{R}^k 中的 k 个点，由 P_1, P_2, \dots, P_k 表示，其位置向量是线性独立的。

标签空间的维度可以比 k 小， k 是 I_n 中的类别数量。例如，在图 18.4 中，我们考虑将 I_n 划分为 $k = 4$ 个类别，并分别为每个类别在 \mathbb{R} 和 \mathbb{R}^2 中关联一个标签点。

接下来的两个线性代数结果阐述了将标签用作点或独热向量之间的关系。

命题18.4.1 考虑 \mathbb{R}^k 中的 k 个不同的点 P_1, \dots, P_k 。那么存在一个线性函数 $f : \mathbb{R}^k \rightarrow \mathbb{R}^k$ ，使得对于 $j = 1, \dots, k$ ，有 $f(e_j) = P_j$ 。

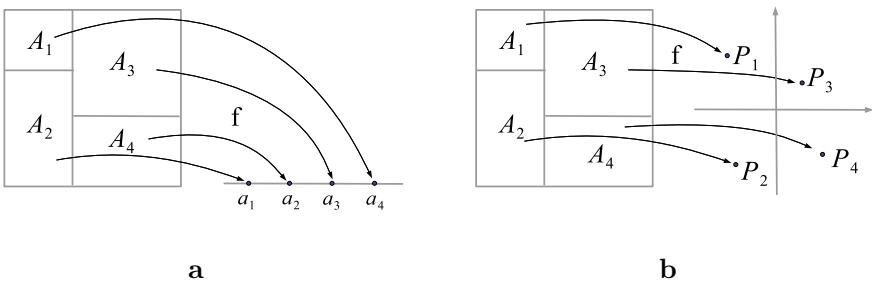


图18.4：与一个划分相关联的决策图：a. 标签空间是 \mathbb{R} 。b. 标签空间是 \mathbb{R}^2 。

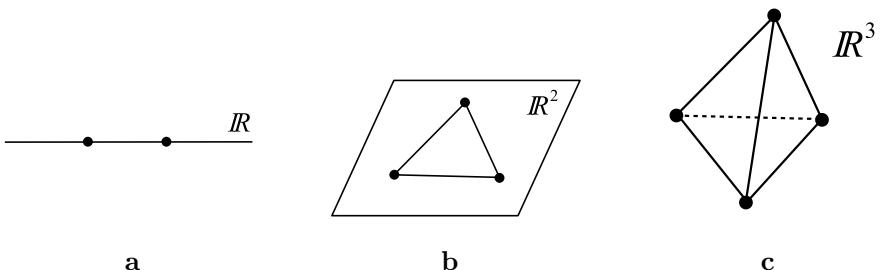


图18.5：处于一般位置的点：a. 直线上的两个不同点。b. 平面上的三个不共线点。c. 三维空间中的四个不共面点。

证明：设 $v_j = (v_j^1, \dots, v_j^k)^T$ 是 \mathbb{R}^k 中 P_j 点的坐标向量，所以我们可以写成

$$v_j = \sum_{i=1}^k v_j^i e_i。那么线性函数 f(x) = Wx，其矩阵为 W_{ij} = v_j^i，就是所需的函数。■$$

反之不一定成立。要使其成立，我们需要施加一个额外条件，该条件将在接下来介绍。

首先，我们注意到通过两个不同的点有唯一一条直线，而通过三个不共线的点有唯一一个平面。通过四个不共面的点有唯一一个三维超平面，依此类推，见图18.5。如果 k 个点处于足够一般的位置，由它们确定的超平面的维数是 $k - 1$ ；否则超平面的维数严格小于 $k - 1$ 。

定义18.4.2 如果在 \mathbb{R}^k 中的 P_1, \dots, P_k 个点不存在维数小于 $k - 1$ 的超平面包含它们，则称这些点处于一般位置。

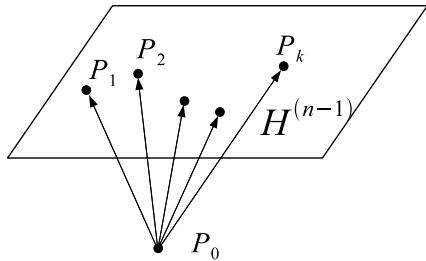


图18.6：如果向量 P_0P_j 线性无关，则点 P_1, \dots, P_k 处于一般位置。

等价地说，包含给定各点的超平面的最低维度是 $k - 1$ 。练习18.11.2中给出了另一个等价表述。下一个结果是使用向量对这个概念的一个有用的特征描述。它大致表明，改变空间的原点，各点的位置向量就会变得线性无关。

命题18.4.3 在 \mathbb{R}^k 中的点 P_1, \dots, P_k 处于一般位置，当且仅当在 \mathbb{R}^k 中存在一个点 P_0 ，使得向量 $\overrightarrow{P_0P_j}$ 在 $\mathbb{R}^k, j \in \{1, \dots, k\}$ 中线性无关。

证明：“ \Rightarrow ”假设点 P_1, \dots, P_k 处于一般位置。那么根据练习18.11.3，存在唯一的 $(k - 1)$ -超平面 \mathcal{H} ，它包含这些点。然后选择任意一点 $P_0 \notin \mathcal{H}$ 可得到线性无关向量 $\overrightarrow{P_0P_1}, \dots, \overrightarrow{P_0P_k}$ ，见图18.6。为了表明我们形成了一个消失的线性组合

$$\sum_{i=1}^k c_i \overrightarrow{P_0P_i} = 0,$$

并证明 $c_i = 0$ 。利用向量分解 $\overrightarrow{P_0P_i} = \overrightarrow{P_0P_1} + \overrightarrow{P_1P_i}$ ，我们可以写成

$$\sum_{i=1}^k c_i \overrightarrow{P_0P_i} = \left(\sum_{i=1}^k c_i \right) \overrightarrow{P_0P_1} + \sum_{i=2}^k c_i \overrightarrow{P_1P_i} = 0.$$

$\left\{ \overrightarrow{P_1P_2}, \dots, \overrightarrow{P_1P_k} \right\}$ 集在 \mathcal{H} 中构成一个独立向量系统，见练习18.11.3。由于 $P_0 \notin \mathcal{H}$ ，向量 $\overrightarrow{P_1P_0}$ 与先前的系统无关，因为它指向超平面 \mathcal{H} 之外。因此，先前的线性组合系数为零， $c_i = 0$ 。
 " \Leftarrow " 设 $P_0 \in \mathbb{R}^k$ 使得 $\left\{ \overrightarrow{P_0P_1}, \dots, \overrightarrow{P_0P_k} \right\}$ 线性独立。

如果点 $\{P_1, \dots, P_k\}$ 不在一般位置，那么它们必定包含在维度为 p 的超平面 \mathcal{P} 内，其中 $p < k - 1$ 。我们有

$$\mathcal{P} = \{Q \in \mathbb{R}^k; \overrightarrow{P_0Q} = \sum_{j=1}^k c_j \overrightarrow{P_0P_j}, \sum_{j=1}^k c_j = 1\}.$$

由于 $\left\{\overrightarrow{P_0P_1}, \dots, \overrightarrow{P_0P_k}\right\}$ 线性无关，超平面 \mathcal{P} 的维度为 $k - 1$ ，这导致了矛盾。■

下一个结果是命题18.4.1的逆命题。

命题18.4.4 考虑处于一般位置的 k 个不同点 $P_1, \dots, P_k \in \mathbb{R}^k$ 。那么存在一个线性函数 $f: \mathbb{R}^k \rightarrow \mathbb{R}^k$ ，使得对于 $j = 1, \dots, k$ ，有 $f(P_j) = e_j$ 。函数 f 是可逆的。

证明：根据命题18.4.3，我们可以选择一个点 P_0 ，使得 $\overrightarrow{P_0P_i}$ 线性无关。这些向量实际上在 \mathbb{R}^k 中构成一个基。设 $g: \mathbb{R}^k \rightarrow \mathbb{R}^k$ 是唯一的线性函数，使得 $g\left(\overrightarrow{P_0P_j}\right) = e_j, j = 1, \dots, k$ 。用 r 表示这样一个函数，它给 \mathbb{R}^k 中的每个点 P 赋予向量 $\overrightarrow{P_0P}$ ，即 $r(P) = \overrightarrow{P_0P}$ 。通过复合 $f = g \circ r$ 构造函数 f 。那么 f 是线性的，作为线性函数的复合，并且满足性质 $f(P_j) = e_j$ 。■

命题18.4.4确保了在 \mathbb{R}^k 中选择标签的两种方式是等价的，一种是作为独热向量 e_j ，另一种是作为一般形式的点 P_j 。我们将在后续章节中处理这两种情况。

18.5 线性可分性

在 \mathbb{R}^n 中的一组点 \mathcal{G} 是一组 n 元组 (x_1, \dots, x_n) ，假定具有某种个体标识。在 \mathbb{R}^n , \mathcal{G}_1 和 \mathcal{G}_2 中的两个聚类，如果存在一个维度为 $n - 1$ 的超平面 \mathcal{H} 将它们分开，则称它们是线性可分的。这意味着：

- (i) 超平面 \mathcal{H} 将空间 \mathbb{R}^n 划分为两个半空间， \mathcal{S}_1 和 \mathcal{S}_2 。
- (ii) 每个聚类都包含在其中一个半空间中： $\mathcal{G}_1 \subset \mathcal{S}_1$ 和 $\mathcal{G}_2 \subset \mathcal{S}_2$ 。如果超平面 \mathcal{H} 由方程定义

$$h(x) = a_1 x_1 + \dots + a_n x_n + d = 0,$$

那么对于任意点 $g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2$ ， \mathcal{G}_1 和 \mathcal{G}_2 的可分离性可以写成 $h(g_1)h(g_2) < 0$ 。这意味着 h 在每个聚类上保持恒定的相反符号，见图 18.7 a。

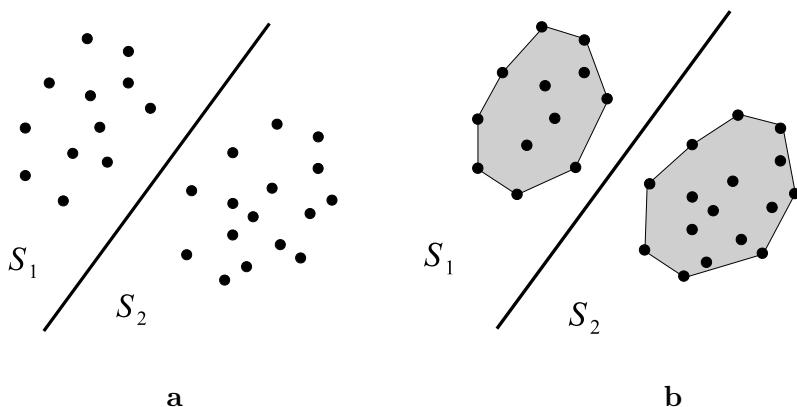


图18.7：a. 两个簇的线性可分性。b. 两个簇的凸包的线性可分性。

示例18.5.1 在 $\mathbb{R}, \mathcal{G}_1$ 和 \mathcal{G}_2 中的两个簇，如果存在一个数 α ，使得对于所有 $g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2$ 都有 $(g_1 - \alpha)(g_2 - \alpha) < 0$ ，那么这两个簇是可分的。这意味着对于所有 $g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2$ ，要么 $g_1 < \alpha < g_2$ ，要么 $g_2 < \alpha < g_1$ 。

如果对于任意两点 $A, B \in K$ ，线段 AB 都包含在集合 K 中，那么集合 $K \subset \mathbb{R}^n$ 就被称为凸集。例如，圆盘、三角形或四面体的内部都是凸集。

一般来说，一个聚类不是凸集。聚类 \mathcal{G} 的凸包是所有凸组合的集合

$$hull(\mathcal{G}) = \left\{ \sum_{g_i \in \mathcal{G}} \lambda_i g_i; \sum_{i=1}^n \lambda_i = 1, \lambda_i \geq 0 \right\}.$$

例如，如果一个聚类只有两个点，它的凸包就是由这两个点定义的闭线段。如果聚类包含三个点，它的凸包就是以给定的点为顶点的三角形（包括内部）。可以证明集合 $hull(\mathcal{G})$ 总是一个包含聚类 \mathcal{G} 的凸集，见练习 18.11.5

命题18.5.2 两个聚类 \mathcal{G}_1 和 \mathcal{G}_2 是线性可分的，当且仅当凸包 (\mathcal{G}_1) 和凸包 (\mathcal{G}_2) 是线性可分的。

证明：“ \Rightarrow ”如果 \mathcal{G}_1 和 \mathcal{G}_2 是线性可分的，那么设 \mathcal{H} 为将空间划分为两个半空间 \mathcal{S}_1 和 \mathcal{S}_2 的超平面，这两个半空间分隔这些聚类，即，

$$\mathcal{G}_1 \subset \mathcal{S}_1, \quad \mathcal{G}_2 \subset \mathcal{S}_2.$$

由于半空间 \mathcal{S}_1 和 \mathcal{S}_2 是凸集，利用凸包的凸极小性，见练习18.11.5，我们有

$$\mathcal{G}_1 \subset \text{hull}(\mathcal{G}_1) \subset \mathcal{S}_1, \quad \mathcal{G}_2 \subset \text{hull}(\mathcal{G}_2) \subset \mathcal{S}_2.$$

因此，凸包 $\text{hull}(\mathcal{G}_1)$ 和 $\text{hull}(\mathcal{G}_2)$ 被超平面 \mathcal{H} 分隔开。

下面给出一种直接从定义出发的证明变体：

假设 \mathcal{G}_1 和 \mathcal{G}_2 是线性可分的，并且设 \mathcal{H} 是具有方程的分隔超平面

$$h(x) = a_1x_1 + \cdots + a_nx_n + d = 0.$$

对于聚类凸包中的任意两点

$$g^1 = \sum_{g_i^1 \in \mathcal{G}_1} \lambda_i^1 g_i^1 \in \text{hull}(\mathcal{G}_1), \quad g^2 = \sum_{g_i^2 \in \mathcal{G}_2} \lambda_i^2 g_i^2 \in \text{hull}(\mathcal{G}_2),$$

利用函数 h 的线性性质，可得

$$\begin{aligned} h(g^1)h(g^2) &= h\left(\sum_{g_i^1 \in \mathcal{G}_1} \lambda_i^1 g_i^1\right)h\left(\sum_{g_j^2 \in \mathcal{G}_2} \lambda_j^2 g_j^2\right) \\ &= \left(\sum_{g_i^1 \in \mathcal{G}_1} \lambda_i^1 h(g_i^1)\right)\left(\sum_{g_j^2 \in \mathcal{G}_2} \lambda_j^2 h(g_j^2)\right) \\ &= \sum_{g_i^1 \in \mathcal{G}_1} \sum_{g_j^2 \in \mathcal{G}_2} \lambda_i^1 \lambda_j^2 h(g_i^1)h(g_j^2) < 0, \end{aligned}$$

因为 $\lambda_i^1 > 0, \lambda_j^2 > 0$ 并且我们使用了聚类可分性条件

$$h(g_i^1)h(g_i^1) < 0.$$

" \Leftarrow " 如果 $\text{hull}(\mathcal{G}_1)$ 和 $\text{hull}(\mathcal{G}_2)$ 是线性可分的，那么存在一个超平面 \mathcal{H} 将空间 \mathbb{R}^n 划分为两个半空间 \mathcal{S}_1 和 \mathcal{S}_2 ，使得 $\text{hull}(\mathcal{G}_1) \subset \mathcal{S}_1$ 和 $\text{hull}(\mathcal{G}_2) \subset \mathcal{S}_2$ 。利用明显的包含关系 $\mathcal{G}_1 \subset \text{hull}(\mathcal{G}_1)$ 和 $\mathcal{G}_2 \subset \text{hull}(\mathcal{G}_2)$ ，可以得出 $\mathcal{G}_1 \subset \mathcal{S}_1$ 和 $\mathcal{G}_2 \subset \mathcal{S}_2$ ，因此 \mathcal{G}_1 和 \mathcal{G}_2 是线性可分的。另见图18.7 b。 ■

即使两个簇有不同的点，但有时它们彼此足够接近，以至于它们的凸包相交。在这种情况下，最终只有通过非线性函数才能实现可分性。

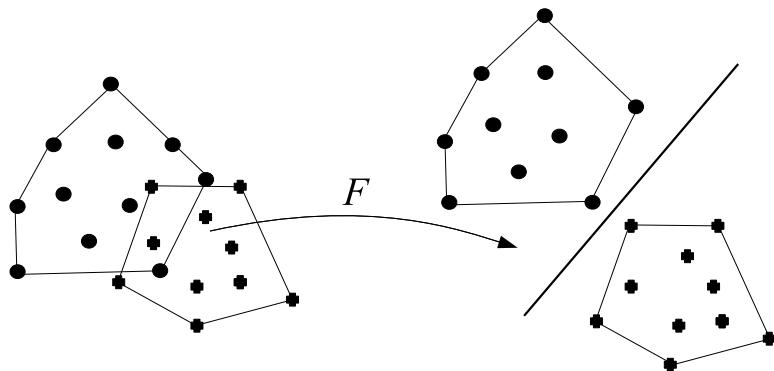


图18.8：两个相交簇的分离函数 F 是非线性的。

命题18.5.3 设 \mathcal{G}_1 和 \mathcal{G}_2 是 \mathbb{R}^n 中的两个簇，使得

$$\text{hull}(\mathcal{G}_1) \cap \text{hull}(\mathcal{G}_2) \neq \emptyset.$$

那么不存在线性函数 $F : \mathbb{R}^n \rightarrow \mathbb{R}^p$ 使得 $F(\mathcal{G}_1)$ 和 $F(\mathcal{G}_2)$ 是线性可分的。

证明：用反证法，假设存在线性函数 $F : \mathbb{R}^n \rightarrow \mathbb{R}^p$ ，使得 $F(\mathcal{G}_1)$ 和 $F(\mathcal{G}_2)$ 是线性可分的，即，在 \mathbb{R}^p 中存在一个由方程

$$h(x) = \sum_{i=1}^p a_i x_i + d = 0 \text{ 定义的超平面，使得}$$

$$\Phi(g_i^1)\Phi(g_i^2) < 0$$

对于所有的 $g_i^1 \in \mathcal{G}_1$ 和 $g_i^2 \in \mathcal{G}_2$ ，其中 $\Phi = h \circ F$ 。

考虑交集中的一个元素

$$g \in \text{hull}(\mathcal{G}_1) \cap \text{hull}(\mathcal{G}_2),$$

因此它有两种表示形式

$$g = \sum_i \lambda_i^1 g_i^1 = \sum_i \lambda_i^2 g_i^2, \quad g_i^1 \in \mathcal{G}_1, g_i^2 \in \mathcal{G}_2.$$

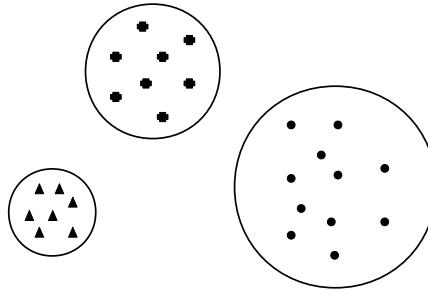


图18.9：三个聚类的凸分离。

利用 F 的线性性质，我们得到以下矛盾：

$$\begin{aligned} 0 \leq \Phi(g)\Phi(g) &= \Phi\left(\sum_i \lambda_i^1 g_i^1\right)\Phi\left(\sum_i \lambda_i^2 g_i^2\right) \\ &= \left(\sum_i \lambda_i^1 \Phi(g_i^1)\right)\left(\sum_j \lambda_j^2 \Phi(g_j^2)\right) \\ &= \sum_{i,j} \lambda_i^1 \lambda_j^2 \Phi(g_i^1)\Phi(g_j^2) < 0. \end{aligned}$$

因此，不存在线性的分离函数 F ，见图18.8。 ■

注记18.5.4 (i) 值得注意的是，也不存在仿射的分离函数，即形式为 $F(x) = Wx + b$ 的函数，其中 W 是一个 $n \times n$ 矩阵且 $b \in \mathbb{R}^n$ 。这是由于可分性是平移不变的。因此，线性神经元无法分离其凸包相交的两个聚类。为此，我们应该使用具有非线性激活函数的神经网络。

- (ii) 如果在 \mathbb{R}^n , \mathcal{G}_1 和 \mathcal{G}_2 中有两个聚类，并且存在一个可逆双连续映射 $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ ，使得聚类图像 $F(\mathcal{G}_1)$ 、 $F(\mathcal{G}_2)$ 在 \mathbb{R}^n 中是线性可分的，那么这两个聚类就被称为 F -可分的。这样的函数 F 被称为 \mathbb{R}^n 的同胚映射。神经网络理论的标准结果表明，一个前馈神经网络（具有足够多的隐藏层）可以学习非线性连续函数 F 。通过用感知器增强网络，我们可以进行最终的线性分类。因此，一个分类问题就简化为学习连续非线性函数 F 。
- (iii) 函数 F 的作用是将聚类分开，以便它们能够被线性分类。然而，存在聚类不能通过 \mathbb{R}^n 的同胚映射分开的情况。在这种情况下，我们需要一个额外的维度来分离聚类，并且连续函数应该是 $F : \mathbb{R}^n \rightarrow \mathbb{R}^p$ ，其中 $p > n$ 。

例如， $\mathcal{G}_1 = \{x \in \mathbb{R}^2; \|x\| < 1\}$ 和 $\mathcal{G}_2 = \{x \in \mathbb{R}^2; \|x\| > 2\}$ 在 \mathbb{R}^2 中不能通过持续拉开它们而分离，但如果我们垂直拉动其中一个，它们在 \mathbb{R}^3 中可以分离。

k 聚类的情况 考虑一族 k 聚类， $\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 。我们说，如果 \mathfrak{G} 的聚类是相互可分离的，即对于 $i \neq j$ ，任意两个聚类 \mathcal{G}_i 和 \mathcal{G}_j 是线性可分离的，那么 \mathfrak{G} 是一个线性可分族。

注记18.5.5 (i) 根据命题18.5.2，族 $\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 是线性可分的，当且仅当 $\{\text{hull}(\mathcal{G}_1), \dots, \text{hull}(\mathcal{G}_k)\}$ 是凸集的线性可分族。

(ii) 用 G_j 表示簇 \mathcal{G}_j 的质心。这意味着 G_j 是通过簇 \mathcal{G}_j 的元素的平均值得到的。应用练习18.11.7，我们得到

$$G_j \in \text{hull}(\mathcal{G}_j), j = 1, \dots, k.$$

18.6 凸可分性

可分性也可以用一种稍有不同但等价的方式来考虑。如果在 \mathbb{R}^n 中有 k 个闭球 B_1, \dots, B_k ，那么 \mathbb{R}^n 中的一族簇

$\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 就称为凸可分的，使得

(i) B_1, \dots, B_k 是不相交的；

(ii) 对于所有的 $j \in \{1, \dots, k\}$ ， $\mathcal{G}_j \subset B_j$ 。

特别地，如果 \mathbb{R}^2 中的两个聚类包含在两个不相交的圆盘中，那么它们是凸可分的。对于三个聚类的情况，见图18.9。以下结果表明了这两种凸性之间的等价性。

命题18.6.1 \mathbb{R}^n 中的两个聚类 $\mathcal{G}_1, \mathcal{G}_2$ 是凸可分的当且仅当它们是线性可分的。

证明：“ \Rightarrow ” 如果 \mathcal{G}_1 和 \mathcal{G}_2 在 \mathbb{R}^n 中是凸可分的，那么在 \mathbb{R}^n 中存在两个不相交的球， B_1, B_2 ，使得 $\mathcal{G}_1 \subset B_1$ 和 $\mathcal{G}_2 \subset B_2$ 。设 \mathcal{H} 是一个 $(n - 1)$ -超平面，它将不相交的球 B_1 和 B_2 分开（这个超平面可以垂直于球心线段构造，经过两个球外部的一个点）。那么 \mathcal{H} 将簇 \mathcal{G}_1 和 \mathcal{G}_2 分开

“ $__$ ” 为了简单起见，我们将考虑 $n = 2$ ；对于更高维度也可以进行类似的推理。假设有一条直线 ℓ 在平面中，它将 \mathcal{G}_1 和 \mathcal{G}_2 分开，并将平面分成两个半平面， \mathcal{S}_1 和 \mathcal{S}_2 ，其中 $\mathcal{G}_1 \subset \mathcal{S}_1$ 和 $\mathcal{G}_2 \subset \mathcal{S}_2$ 。设 M 是直线 ℓ 上离聚类 \mathcal{G}_1 最近的点。存在一个足够小的 $\epsilon > 0$ ，使得从 M 出发且与直线 ℓ 夹角等于 ϵ 的半直线不与 $\text{hull}(\mathcal{G}_1)$ 相交，见

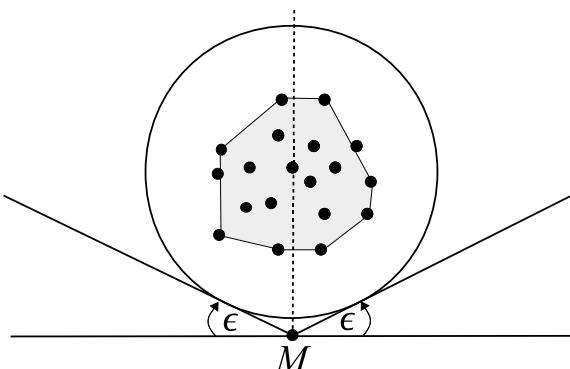


图18.10：构建一个包含聚类凸包且包含在给定半平面内的圆盘。

图18.10。设 β 为两条半直线所成角的平分线。那么对于任意点 $O \in \beta$ ，存在一个以 O 为圆心的唯一圆，它与两条半直线都相切。当距离 $\|OM\|$ 足够大时，聚类 \mathcal{G}_1 位于以 O 为圆心的球内。这就是球 B_1 的构建。类似地，我们可以在直线 ℓ 的另一侧构建另一个包含聚类 \mathcal{G}_2 的球 B_2 。

注记18.6.2 即使凸可分性看起来是一个更一般的概念且似乎更有意义，然而，能够使用神经网络处理的是线性可分性。

下一个结果涉及一族聚类的独热向量决策映射的存在性。

命题18.6.3 设 $\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 为 \mathbb{R}^k 中一族 k 聚类，它们是线性可分的。那么存在一个决策映射 $F: \mathbb{R}^k \rightarrow \mathbb{R}^k$ ，满足 $F(\mathcal{G}_j) = e_j, j = 1, \dots, k$.

证明：由于 $\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 是线性可分的，那么它是凸可分的。因此，在 \mathbb{R}^k 中存在 k 个互不相交的球 B_1, \dots, B_k ，使得对于所有 j ，都有 $\mathcal{G}_j \subset B_j$ 。考虑一个决策映射 $f: \mathbb{R}^k \rightarrow \mathbb{R}^k$ ，它将第 j 个球映射到 e_j ，即对于 $j = 1, \dots, k$ ，有 $f(B_j) = e_j$ 。证明完毕。■

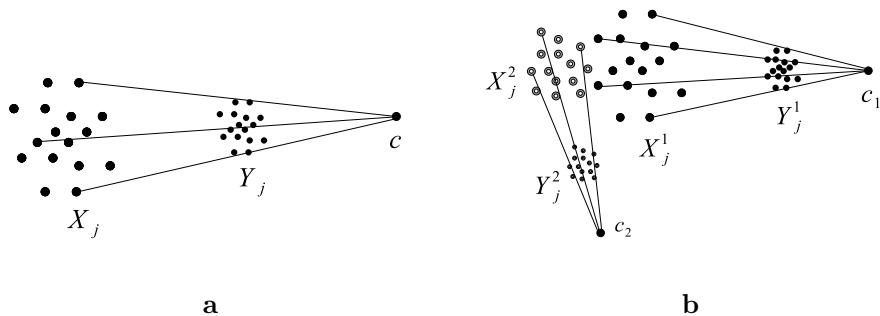


图18.11：a. 一个聚类向一个点收缩。b. 两个聚类向两个不同的点收缩。

18.7 向中心收缩

聚类的分离通常涉及将聚类朝着某些点拉开的变换，这些点随后被视为标签。

我们从单个聚类的最简单情况开始，该聚类被拉向一个称为中心的点。设 X_j 为聚类 \mathcal{G} 中各点的位置向量， C 为给定的中心及其位置向量 c 。由下式给出的变换

$$Y_j = \lambda X_j + (1 - \lambda)c,$$

对于 $\lambda \in (0, 1)$ ，将聚类 \mathcal{G} 收缩成位于点 C 附近的一个聚类，见图18.11 a。 λ 的值越小，图像聚类就越接近点 C 。

现在考虑两个聚类， \mathcal{G}_1 和 \mathcal{G}_2 ，以及两个不同的中心， C_1 和 C_2 。设 X_j^r 为聚类 r 中元素的位置向量，其中 $r \in \{1, 2\}$ ，并且 c_i 为聚类 C_i 的位置向量。这些变换

$$Y_j^1 = \lambda_1 X_j^1 + (1 - \lambda_1)c_1,$$

$$Y_j^2 = \lambda_2 X_j^2 + (1 - \lambda_2)c_2,$$

将第一个聚类朝着中心 C_1 的方向推动，而将第二个聚类朝着 C_2 的方向推动，见图18.11 b。新的聚类比之前的更加分离。

前面的方程仅针对聚类点编写。问题是是否可以将它们扩展为在聚类所在的整个空间上定义的全局函数。这将在下一节中讨论。

18.8 学习决策图

通过学习某些决策图可以实现聚类的分类。本节将讨论这个主题。

18.8.1 线性决策图

考虑 \mathbb{R}^2 中的两个聚类

$$\mathcal{G} = \{(x_1, y_1), \dots, (x_N, y_N)\}, \quad \tilde{\mathcal{G}} = \{(\tilde{x}_1, \tilde{y}_1), \dots, (\tilde{x}_M, \tilde{y}_M)\},$$

并假设 $\text{hull}(\mathcal{G}) \cap \text{hull}(\tilde{\mathcal{G}}) = \emptyset$ 。根据练习 18.11.9，聚类 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 是线性可分的。我们将研究一维和二维标签的情况。

一维标签 我们分别将两个标签 α 和 $\tilde{\alpha}$ 与簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 相关联。这些标签是两个不同的实数。由于簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 是线性可分的，所以寻找一个线性函数 $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ 是有意义的，该函数将簇 \mathcal{G} 映射到 α 的邻域内，将簇 $\tilde{\mathcal{G}}$ 映射到 $\tilde{\alpha}$ 的邻域内，并且还希望中点 $(\alpha + \tilde{\alpha})/2$ 能分隔集合 $f(\mathcal{G})$ 和 $f(\tilde{\mathcal{G}})$ 。

我们将寻找一个函数 f ，使其成为具有一维输出（输出层中有一个神经元）且无隐藏层的前馈神经网络的输入-输出函数，即我们假设它由下式给出

$$f_{w,b}(x, y) = w_1 x + w_2 y + b.$$

必须选择实参数 w_i, b ，使得图像集 $f(\mathcal{G})$ 和 $f(\tilde{\mathcal{G}})$ 分别围绕 α 和 $\tilde{\alpha}$ 紧密定位。这可以通过取

$$(w, b) = \arg \min F(w, b),$$

其中 F 是以下二次成本函数，用于衡量标签与图像之间的距离：

$$\begin{aligned} F(w, b) &= \frac{1}{2} \sum_{i=1}^N \left(f_{w,b}(x_i, y_i) - \alpha \right)^2 + \frac{1}{2} \sum_{j=1}^M \left(f_{w,b}(\tilde{x}_j, \tilde{y}_j) - \tilde{\alpha} \right)^2 \\ &= \frac{1}{2} \sum_{i=1}^N \left(w_1 x_i + w_2 y_i + b - \alpha \right)^2 + \frac{1}{2} \sum_{j=1}^M \left(w_1 \tilde{x}_j + w_2 \tilde{y}_j + b - \tilde{\alpha} \right)^2. \end{aligned}$$

对于方程 $\nabla F(w, b) = 0$ 的解实现最小值。在下面我们将计算梯度 $\nabla F = (\partial_{w_1} F, \partial_{w_2} F, \partial_b F)$ 。使用链式法则进行直接求导可得

$$\begin{aligned}
\partial_{w_1} F &= \sum_{i=1}^N x_i(w_1 x_i + w_2 y_i + b - \alpha) + \sum_{j=1}^M \tilde{x}_j(w_1 \tilde{x}_j + w_2 \tilde{y}_j + b - \tilde{\alpha}) \\
&= w_1 \left(\sum_{i=1}^N x_i^2 + \sum_{j=1}^M \tilde{x}_j^2 \right) + w_2 \left(\sum_{i=1}^N x_i y_i + \sum_{j=1}^M \tilde{x}_j \tilde{y}_j \right) \\
&\quad + b \left(\sum_{i=1}^N x_i + \sum_{j=1}^M \tilde{x}_j \right) - \alpha \sum_{i=1}^N x_i - \tilde{\alpha} \sum_{j=1}^M \tilde{x}_j
\end{aligned}$$

同样地，我们有

$$\begin{aligned}
\partial_{w_2} F &= w_1 \left(\sum_{i=1}^N x_i y_i + \sum_{j=1}^M \tilde{x}_j \tilde{y}_j \right) + w_2 \left(\sum_{i=1}^N y_i^2 + \sum_{j=1}^M \tilde{y}_j^2 \right) \\
&\quad + b \left(\sum_{i=1}^N y_i + \sum_{j=1}^M \tilde{y}_j \right) - \alpha \sum_{i=1}^N y_i - \tilde{\alpha} \sum_{j=1}^M \tilde{y}_j, \\
\partial_b F &= w_1 \left(\sum_{i=1}^N x_i + \sum_{j=1}^M \tilde{x}_j \right) + w_2 \left(\sum_{i=1}^N y_i + \sum_{j=1}^M \tilde{y}_j \right) \\
&\quad + (M+N)b - \alpha N - \tilde{\alpha} M.
\end{aligned}$$

考虑以下 3×3 矩阵：

$$A = \begin{pmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & N \end{pmatrix}, \quad \tilde{A} = \begin{pmatrix} \sum \tilde{x}_j^2 & \sum \tilde{x}_j \tilde{y}_j & \sum \tilde{x}_j \\ \sum \tilde{x}_j \tilde{y}_j & \sum \tilde{y}_j^2 & \sum \tilde{y}_j \\ \sum \tilde{x}_j & \sum \tilde{y}_j & M \end{pmatrix},$$

它们分别包含关于簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 的信息，例如 x 和 y 变量的大小、一阶矩和二阶矩，以及它们的相关性。再考虑另外两个向量

$$\beta = \alpha \begin{pmatrix} \sum x_i \\ \sum y_i \\ N \end{pmatrix}, \quad \tilde{\beta} = \tilde{\alpha} \begin{pmatrix} \sum \tilde{x}_j \\ \sum \tilde{y}_j \\ M \end{pmatrix},$$

它们取决于标签和一阶矩。那么向量方程 $\nabla F = 0$ 可以写成线性矩阵方程

$$(A + \tilde{A})X = \beta + \tilde{\beta},$$

其中 $X^T = (w_1, w_2, b)$ 。解由下式给出

$$X = (A + \tilde{A})^{-1}(\beta + \tilde{\beta}) = \tilde{A}^{-1}(\mathbb{I} + A\tilde{A}^{-1})^{-1}(\beta + \tilde{\beta}),$$

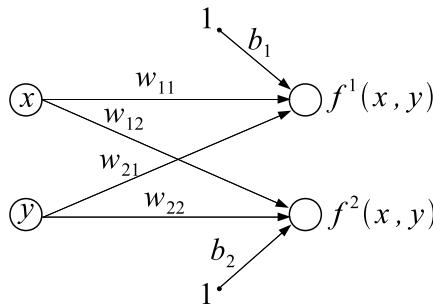


图18.12：没有隐藏层且有两个输出的神经网络。

其中我们使用了附录G中求和公式 (G.1.5) 的逆公式。解的存在性在条件 $\|A\tilde{A}^{-1}\| < 1$ 下成立，这意味着 A 的特征值分别小于 \tilde{A} 的特征值。上述条件的几何意义大致如下：较长的聚类也是较宽的聚类。在所有实际应用中，逆矩阵 $(A + \tilde{A})^{-1}$ 可以按照附录G 中的说明进行计算。

值得注意的是，只要它们的凸包相互不相交，这种方法就可以应用于两个以上的聚类。

上述考虑的标签 α 和 $\tilde{\alpha}$ 是实数；出于简单起见，可以考虑 $\alpha = 1$ 和 $\tilde{\alpha} = 0$ 。然而，这并不是唯一的好可能性。

二维标签 我们将在下面使用二维标签（例如平面中的向量或不同的点）对簇 $\mathcal{G}, \tilde{\mathcal{G}}$ 进行线性分类。假设簇 \mathcal{G} 的标签是 \mathbb{R}^2 中坐标为 (a_1, a_2) 的点 A 。类似地，簇 $\tilde{\mathcal{G}}$ 的标签是坐标为 $(\tilde{a}_1, \tilde{a}_2)$ 的点 \tilde{A} 。

如前所述，命题18.5.2表明聚类 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 是线性可分的。在这种情况下，我们将寻找一个线性函数 $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ ，它将点 A 邻域内的聚类 \mathcal{G} 和 A 邻域内的聚类 $\tilde{\mathcal{G}}$ 进行映射，见图18.13。

线性函数 $f = (f^1, f^2)$ 被构造为一个具有二维输出且无隐藏层的前馈神经网络的输入-输出函数，见图18.12。这意味着 f 在每个分量上都是线性的

$$f_{w,b}^1(x, y) = w_{11}x + w_{21}y + b_1$$

$$f_{w,b}^2(x, y) = w_{12}x + w_{22}y + b_2.$$

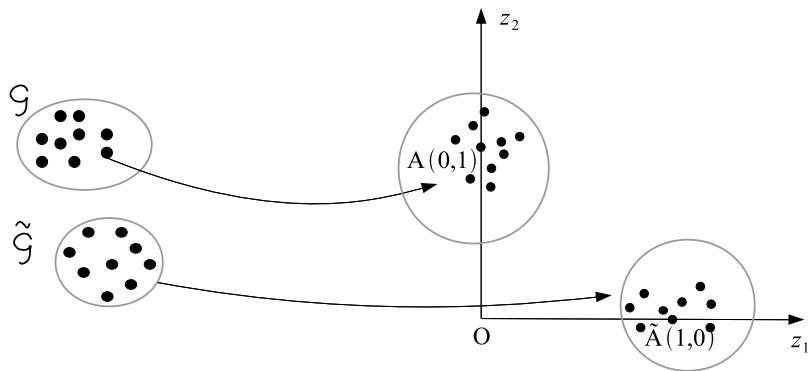


图18.13：线性可分簇 $\mathcal{G}, \tilde{\mathcal{G}}$ 被映射到点标签 A 和 \tilde{A} 的不相交邻域中。

权重 w_{ij} 和偏差 b_j 必须进行调整，以使图像集 $f(\mathcal{G})$ 和 $f(\tilde{\mathcal{G}})$ 分别尽可能接近点 A 和 \tilde{A} 。这可以通过取

$$(w, b) = \arg \min G(w, b),$$

其中 G 是到点标签的平方距离之和，由下式给出

$$\begin{aligned} G(w, b) &= \frac{1}{2} \sum_{i=1}^N d(f_{w,b}(x_i, y_i), A)^2 + \frac{1}{2} \sum_{j=1}^M d(f_{w,b}(\tilde{x}_j, \tilde{y}_j), \tilde{A})^2 \\ &= \frac{1}{2} \sum_{i=1}^N (f^1(x_i, y_i) - a_1)^2 + \frac{1}{2} \sum_{i=1}^N (f^2(x_i, y_i) - a_2)^2 \\ &\quad + \frac{1}{2} \sum_{j=1}^M (f^1(\tilde{x}_j, \tilde{y}_j) - \tilde{a}_1)^2 + \frac{1}{2} \sum_{j=1}^M (f^2(\tilde{x}_j, \tilde{y}_j) - \tilde{a}_2)^2 \\ &= F_1(w_{11}, w_{21}, b_1) + F_2(w_{12}, w_{22}, b_2), \end{aligned}$$

其中

$$\begin{aligned} F_1(w_{11}, w_{21}, b_1) &= \frac{1}{2} \sum_{i=1}^N (w_{11}x_i + w_{21}y_i + b_1 - a_1)^2 \\ &\quad + \frac{1}{2} \sum_{j=1}^M (w_{11}\tilde{x}_j + w_{21}\tilde{y}_j + b_1 - \tilde{a}_1)^2 \end{aligned}$$

$$\begin{aligned} F_2(w_{12}, w_{22}, b_2) &= \frac{1}{2} \sum_{i=1}^N (w_{12}x_i + w_{22}y_i + b_2 - a_2)^2 \\ &\quad + \frac{1}{2} \sum_{j=1}^M (w_{12}\tilde{x}_j + w_{22}\tilde{y}_j + b_2 - \tilde{a}_2)^2. \end{aligned}$$

对于方程 $\nabla G(w, b) = 0$ 的解，最小值得以实现，其中

$$\nabla G(w, b) = (\nabla_{w_{11}} F_1, \nabla_{b_1} F_1, \nabla_{w_{12}} F_2, \nabla_{b_2} F_2) \in \mathbb{R}^6.$$

该系统包含6个方程和6个未知数（4个权重和2个偏差）。给定 F_1 和 F_2 的公式，与一维标签的情况类似，可以证明该系统等效于以下两个线性系统：

$$(A + \tilde{A})X_1 = \beta_1 + \tilde{\beta}_1$$

$$(A + \tilde{A})X_2 = \beta_2 + \tilde{\beta}_2,$$

其中 A, \tilde{A} 的定义与前一种情况相同， $X_j^T = (w_{1j}, w_{2j}, b_j)$ ，并且

$$\beta_i = a_i \left(\frac{\sum x_i}{\sum y_i} \right), \quad \tilde{\beta}_i = \tilde{a}_i \left(\frac{\sum \tilde{x}_j}{\sum \tilde{y}_j} \right).$$

使用附录G中的公式(G.1.5)，解由下式给出

$$X_1 = \tilde{A}^{-1}(\mathbb{I} + A\tilde{A}^{-1})^{-1}(\beta_1 + \tilde{\beta}_1),$$

$$X_2 = \tilde{A}^{-1}(\mathbb{I} + A\tilde{A}^{-1})^{-1}(\beta_2 + \tilde{\beta}_2),$$

前提是条件 $\|A\tilde{A}^{-1}\| < 1$ 成立。

softmax函数的使用 大多数分类问题会使用一个额外的层来实现 softmax 函数，该函数在2.1节中已有介绍。对于前面的示例，我们将这样做，见图18.14。为了简单起见，我们可以选择标签为独热向量，即 $(a_1, a_2) = (0, 1) = e_1$ 和 $(\tilde{a}_1, \tilde{a}_2) = (1, 0) = e_2$ 。

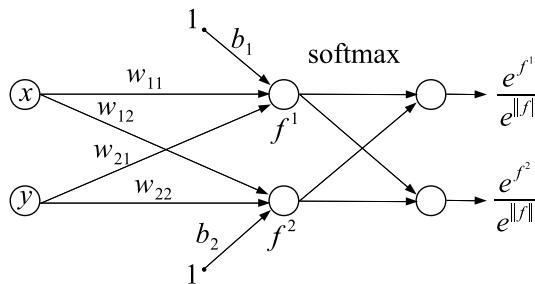


图18.14：具有实现softmax函数的额外层的神经网络。

新网络的结果是 $z = (z_1, z_2)$ ，其中

$$z_i = \frac{e^{f^i}}{\|e^f\|} = \frac{e^{f^i}}{e^{f^1} + e^{f^2}}, \quad i = 1, 2.$$

由于 $z_i > 0$ 和 $z_1 + z_2 = 1$ ，所以 z 属于连接点 $A = (0, 1)$ 和 $\tilde{A} = (1, 0)$ 的线段。那么聚类 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 被映射到线段 $A\tilde{A}$ ，其中 \mathcal{G} 的图像更靠近标签点 A ，而 $\tilde{\mathcal{G}}$ 的图像更靠近 \tilde{A} ，见图18.15。

假设聚类图像的分离点是线段的中点， $(1/2, 1/2)$ 。现在，设 (x, y) 是平面中的一个给定的点，并且我们必须确定点 (x, y) 属于哪个聚类。我们将 (x, y) 通过神经网络，如果结果属于线段 $A\tilde{A}$ 的上半部分，那么该点属于聚类 \mathcal{G} 。否则，它属于 $\tilde{\mathcal{G}}$ 。这个测试可以使用水平坐标或垂直坐标来完成。例如，使用坐标 z_1 ，我们有

如果 $z_1 < 1/2$ ，那么 (x, y) 属于 \mathcal{G} ；

如果 $z_1 > 1/2$ ，那么 (x, y) 属于 $\tilde{\mathcal{G}}$ 。

对于涉及超过2个聚类的分类问题，使用softmax也很成功。例如，在3个聚类的情况下， $\mathcal{G}_1, \mathcal{G}_2$ ，以及 \mathcal{G}_3 ，我们在 \mathbb{R}^3 中关联标签 $A_1 = (1, 0, 0), A_2 = (0, 1, 0)$ 和 $A_3 = (0, 0, 1)$ ，它们形成一个等边三角形，见图18.16。在这种情况下，分离点被一个由从三角形中心向其边作垂线得到的分离曲线系统所取代，见图18.17。为了确定一个点所属的决策区域，只需评估该点到顶点的距离。例如，如果 $\text{dist}(P, A_i) < 1/2$ ，那么点 P 属于聚类 \mathcal{G}_i 的图像。

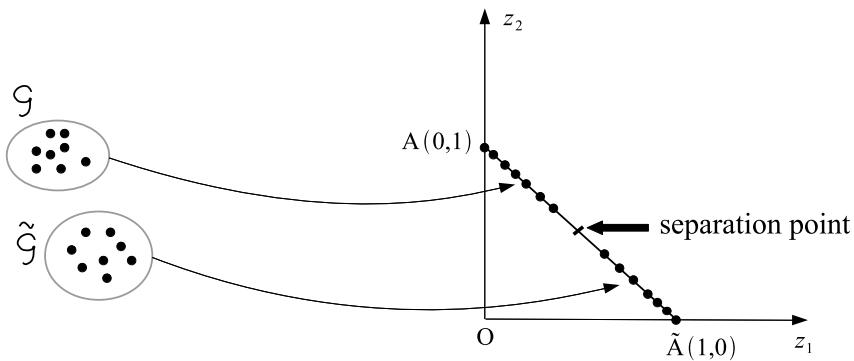


图18.15：线性可分的簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 被映射到一条线段的终端区域。在线段内部有一个分离点。

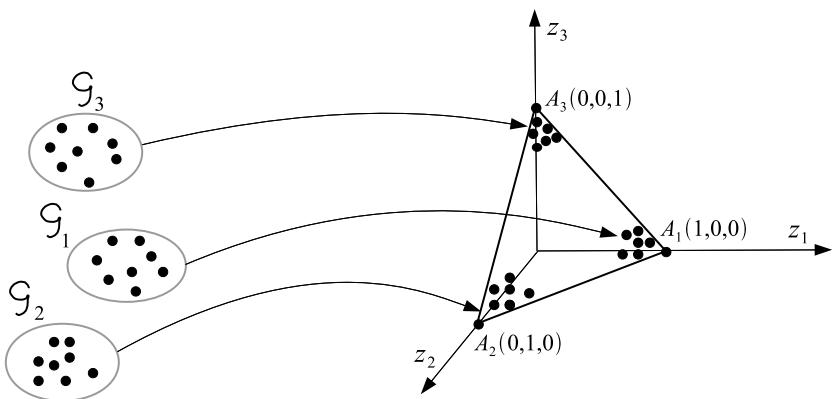


图18.16：线性可分的簇 \mathcal{G}_1 , \mathcal{G}_2 和 \mathcal{G}_3 被映射到一个三角形的顶点区域。

18.8.2 非线性决策图

考虑 \mathbb{R}^2 中的两个聚类

$$\mathcal{G} = \{(x_1, y_1), \dots, (x_N, y_N)\}, \quad \tilde{\mathcal{G}} = \{(\tilde{x}_1, \tilde{y}_1), \dots, (\tilde{x}_M, \tilde{y}_M)\},$$

并假设 $\text{hull}(\mathcal{G}) \cap \text{hull}(\tilde{\mathcal{G}}) \neq \emptyset$ 。

我们将两个实数 α 和 $\tilde{\alpha}$ 分别作为簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 的标签。根据命题18.5.2, 簇 \mathcal{G} 和 $\tilde{\mathcal{G}}$ 不是线性可分的。因此, 寻找一个非线性函数 $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ 是有意义的, 该函数将簇 \mathcal{G} 映射到 α 的邻域, 将簇 $\tilde{\mathcal{G}}$ 映射到 $\tilde{\alpha}$ 的邻域。

Sigmoid神经元 非线性函数最简单的情况由Sigmoid神经元产生, 其输入输出函数为

$$f_{w,b}(x, y) = \sigma(w_1 x + w_2 y + b),$$

其中 $\sigma(x)$ 是逻辑函数。在这种情况下, 我们假设标签由 $\alpha = 0$ 和 $\tilde{\alpha} = 1$ 给出。因此, 簇 \mathcal{G} 将被映射到0, 而簇 $\tilde{\mathcal{G}}$ 将被映射到1。为了使这个映射尽可能地围绕标签值局部化, 我们需要最小化成本函数

$$F(w, b) = \frac{1}{2} \sum_{i=1}^N \sigma(w_1 x_i + w_2 y_i + b)^2 + \frac{1}{2} \sum_{j=1}^M [\sigma(w_1 \tilde{x}_j + w_2 \tilde{y}_j + b) - 1]^2.$$

利用逻辑 sigmoid 函数 $\sigma' = \sigma(1 - \sigma)$ 的微分性质, 我们得到

$$\begin{aligned} \partial_{w_1} F &= \sum_{i=1}^N x_i \sigma^2(1 - \sigma) \Big|_{w_1 x_i + w_2 y_i + b} - \sum_{j=1}^M \tilde{x}_j \sigma(1 - \sigma)^2 \Big|_{w_1 \tilde{x}_j + w_2 \tilde{y}_j + b} \\ \partial_{w_2} F &= \sum_{i=1}^N y_i \sigma^2(1 - \sigma) \Big|_{w_1 x_i + w_2 y_i + b} - \sum_{j=1}^M \tilde{y}_j \sigma(1 - \sigma)^2 \Big|_{w_1 \tilde{x}_j + w_2 \tilde{y}_j + b} \\ \partial_b F &= \sum_{i=1}^N \sigma^2(1 - \sigma) \Big|_{w_1 x_i + w_2 y_i + b} - \sum_{j=1}^M \sigma(1 - \sigma)^2 \Big|_{w_1 \tilde{x}_j + w_2 \tilde{y}_j + b}. \end{aligned}$$

由于非线性, 不存在最小值的闭式解。找到最小值的方法是应用梯度下降法。近似序列通过迭代给出

$$\begin{aligned} w_1^{(n+1)} &= w_1^{(n)} - \eta \partial_{w_1} F(w_1^{(n)}, w_2^{(n)}, b^{(n)}) \\ w_2^{(n+1)} &= w_2^{(n)} - \eta \partial_{w_2} F(w_1^{(n)}, w_2^{(n)}, b^{(n)}) \\ b^{(n+1)} &= b^{(n)} - \eta \partial_b F(w_1^{(n)}, w_2^{(n)}, b^{(n)}), \end{aligned}$$

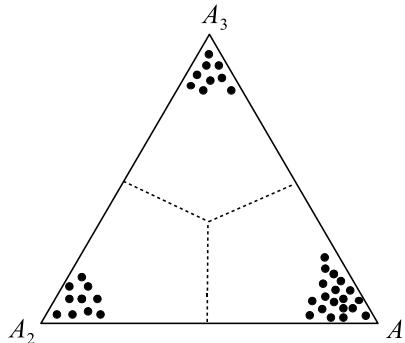


图18.17：虚线将三角形划分为三个决策区域。

学习率为 $\eta > 0$ ，初始值为 $w_1^{(0)} = w_2^{(0)} = b^{(0)} = 0$ 。值得注意的是，这是逻辑回归方法的一种变体，仅适用于两个聚类的分类。

单隐藏层网络 我们可以假设非线性函数 f 是一个具有三层且输出为一维的前馈神经网络的输入-输出函数。隐藏层中的神经元假定具有非线性激活函数。那么输入-输出映射为

$$f_{w,\lambda,b}(x,y) = \sum_{k=1}^K \lambda_k \sigma(w_{1k}x + w_{2k}y + b_k),$$

其中 K 是隐藏神经元的数量， w_{ij} 是从输入到隐藏层的权重， λ_k 是从隐藏层到输出的权重， b_k 是隐藏神经元的偏差，见图 18.18。函数 $f_{w,\lambda,b}$ 取决于 $4K$ 个参数 ($2K$ 权重 w_{ij} , K 偏差 b_k 以及 K 个权重 λ_k)。必须调整这些参数，以使以下成本函数最小化：

$$G(w, b, \lambda) = \frac{1}{2} \sum_{i=1}^N [f_{w,\lambda,b}(x_i, y_i) - \alpha]^2 + \frac{1}{2} \sum_{j=1}^M [f_{w,\lambda,b}(\tilde{x}_j, \tilde{y}_j) - \tilde{\alpha}]^2, \quad (18.8.2)$$

其中 $\alpha, \tilde{\alpha}$ 分别是与聚类 $\mathcal{G}, \tilde{\mathcal{G}}$ 相关联的标签。由于对于前一个成本函数的最小值没有封闭形式的解，因此需要一种梯度下降类型的算法，见练习 18.11.10。

在多个聚类的情况下，我们可以使用多个标签，并采用与之前类似的方法。然而，使用向量标签，比如独热向量，将为解决分类问题提供一个更简洁的变体。

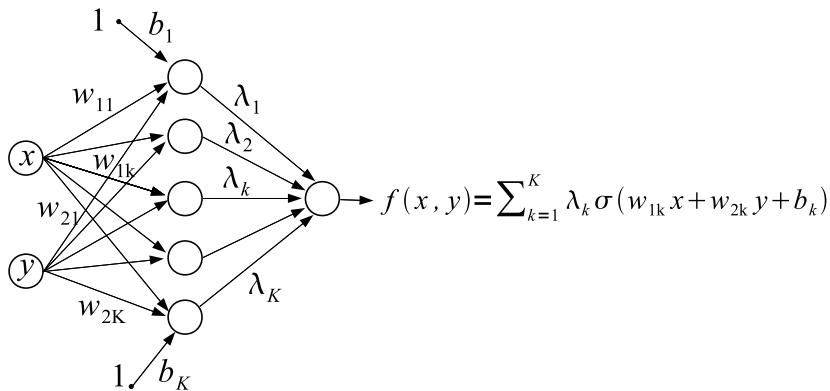


图18.18：具有一个隐藏层和一个输出的神经网络。

在这种情况下，标签空间是 \mathbb{R}^p ，其中 p 是需要分类的簇的数量，见练习18.11.11。

18.9 决策边界

考虑 \mathbb{R}^2 中的两个簇， \mathcal{G}_1 和 \mathcal{G}_2 。决策边界是 \mathbb{R}^2 中分隔这些簇的连续曲线，见图18.19。这意味着对于任何 $g_1 \in \mathcal{G}_1$ 和 $g_2 \in \mathcal{G}_2$ ，从 g_1 到 g_2 的任何连续曲线都会与决策边界曲线相交。

下一个结果表明，决策边界的存在在局部上等同于簇的线性分离。

定理18.9.1 (整流定理) 考虑 \mathbb{R}^2 中的两个聚类 \mathcal{G}_1 和 \mathcal{G}_2 以及它们之间的光滑决策边界。那么在决策曲线的邻域上存在一个取值于 \mathbb{R}^2 的光滑非线性函数 F ，使得 F 将决策曲线映射到 x 轴，将聚类 \mathcal{G}_1 映射到上半平面，将聚类 \mathcal{G}_2 映射到下半平面。

证明：用 $c(s), 0 \leq s \leq T$ 表示决策曲线，并考虑其通过法向流 $\Phi_t(c(s))$ 的演化，其中 $-\epsilon < t < \epsilon$ 。这是 $\Phi_t(c(s)) = c(s) + tN(s)$ ，其中 N 是 $c(s)$ 的单位法向量，见图18.20。对于足够小的 ϵ ，流没有任何奇点。簇 \mathcal{G}_1 对应于值 $t > 0$ ，簇 \mathcal{G}_2 对应于 $t < 0$ 。集合 $\mathcal{U} = \{\Phi_t(c(s)); |t| < \epsilon, 0 \leq s \leq T\}$ 定义了决策曲线的一个邻域。然后通过 $F(\Phi_t(c(s))) = (s, t)$ 定义映射 $F: \mathcal{U} \rightarrow \mathbb{R}^2$ 。

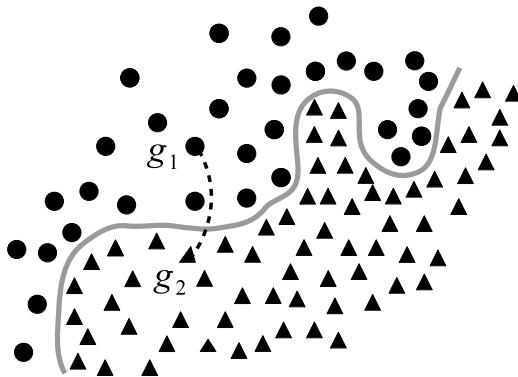


图18.19： \mathbb{R}^2 中两个聚类之间的决策边界曲线。

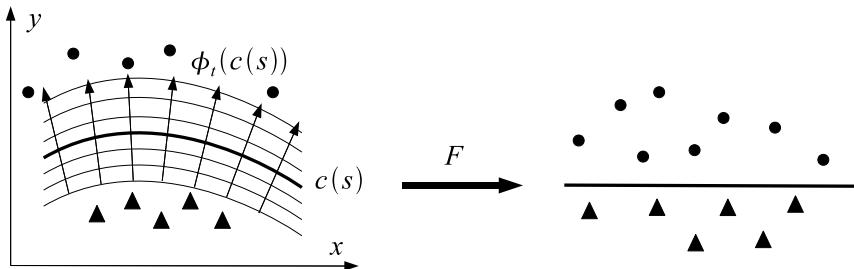


图18.20：函数 F 将决策曲线映射到 x 轴上。

由于 $\Phi_0(c(s)) = c(s)$ ，那么 F 将 $c(s)$ 映射到 $(s, 0)$ ，并将聚类 $\mathcal{G}_1 \cap \mathcal{U}$ 发送到上半平面，将聚类 $\mathcal{G}_2 \cap \mathcal{U}$ 发送到下半平面。

这个结果可以扩展到更高维度。如果聚类 \mathcal{G}_1 和 \mathcal{G}_2 在 \mathbb{R}^n 中，决策边界是 \mathbb{R}^n 中的一个 $(n - 1)$ -超曲面 \mathcal{H} ，它将聚类分开，即，对于任何 $g_1 \in \mathcal{G}_1$ 和 $g_2 \in \mathcal{G}_2$ ，从 g_1 到 g_2 的任何连续曲线都与超曲面 \mathcal{H} 相交。

与前一种情况类似，我们可以在 \mathcal{H} 的邻域中构建一个坐标系。该邻域中的任何点 P 都可以投影到 \mathcal{H} 上的点 P' 。与 P 相关联的坐标是 $(x_1, \dots, x_{n-1}, x_n)$ ，其中 (x_1, \dots, x_{n-1}) 是 P' 在 \mathcal{H} 上的坐标， x_n 是 P 投影到 \mathcal{H} 上的长度，并根据超曲面的方向赋予符号。因此，函数 $F(P) = (x_1, \dots, x_n)$ 将 \mathcal{H} 映射到超平面 $\{x_n = 0\}$ 。

涉及两个以上聚类的情况更为复杂，因为在这种情况下决策边界具有更丰富的结构。例如，如果聚类位于一个平面中，我们有以下定义。考虑在 \mathbb{R}^2 中 p 个点 $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_p$ 的聚类。在这种情况下，决策边界是 \mathbb{R}^2 中分隔聚类的连续曲线的连通系统。也就是说，对于任何 $g_i \in \mathcal{G}_i$ 和 $g_j \in \mathcal{G}_j$ ，其中 $i \neq j$ ，从 g_i 到 g_j 的任何连续曲线都会与决策边界相交。

18.10 总结

在聚类的分类过程中，必须为每个聚类准确分配标签。有几种类型的标签可以分配：数字、独热向量、点等。将标签分配给聚类的映射是一个决策映射，可以由神经网络学习。

如果两个聚类的凸包不相交，则它们是线性可分的。在这种情况下，单个感知机可以学习相关的决策函数。

如果两个聚类的凸包不相交，则聚类不是线性可分的，在这种情况下，使用具有非线性激活函数的神经网络来学习相关的决策函数。

在神经网络的最后一层使用softmax激活函数将 n 个聚类映射到一个 $(n - 1)$ 维多面体的角上。

整流定理指出，聚类之间存在决策边界在局部上等同于聚类的线性分离。

18.11 练习

练习18.11.1 设 $\mathcal{S} = (I_1 \times I_1) \cap (\mathbb{Q} \times \mathbb{Q})$ 为 $[0, 1] \times [0, 1]$ 中的有理数格。

- (a) 证明 \mathcal{S} 是一个传递关系；
- (b) \mathcal{S} 是等价关系吗？

练习18.11.2 证明当且仅当存在唯一的包含这些点的 $(k - 1)$ -超平面时， k 中的点处于一般位置。

练习18.11.3 设 P_1, \dots, P_k 为处于一般位置的 \mathbb{R}^k 中的 k 个点，且设 \mathcal{H} 为包含它们的唯一 $(k - 1)$ -超平面。证明集合 $\left\{ \overrightarrow{P_1P_2}, \dots, \overrightarrow{P_1P_k} \right\}$ 在 \mathcal{H} 中构成一个线性无关向量组。

练习18.11.4 考虑两个不同的实数 $x_1, x_2 \in \mathbb{R}$ 。证明存在唯一的仿射函数 $f(x) = ax + b$ 使得 $f(x_1) = 1$ 且 $f(x_2) = 2$ 。

练习18.11.5 设 \mathcal{G} 为 \mathbb{R}^n 中的一个聚类。

- (a) 证明凸包 (\mathcal{G}) 是一个凸集；
- (b) 证明 $\text{hull}(\mathcal{G})$ 是 \mathbb{R}^n 中包含 \mathcal{G} 的最小凸集；
- (c) 验证

$$\text{hull}(\mathcal{G}) = \bigcap_K \{K; \mathcal{G} \subset K \subset \mathbb{R}^n \text{ convex}\}.$$

练习18.11.6 设 $\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ 为一个非退化仿射函数，即

$\Phi(x) = Wx + b$ ，其中 $\det W \neq 0$ 。考虑 $\mathbb{R}^2, \mathcal{G}_1$ 和 \mathcal{G}_2 中的两个线性可分簇。证明 $\Phi(\mathcal{G}_1)$ 和 $\Phi(\mathcal{G}_2)$ 是线性可分的。换句话说，仿射函数保持线性可分性。

练习18.11.7 设 \mathcal{G} 为一个有限点簇，并用 G 表示其质心。证明 $G \in \text{hull}(\mathcal{G})$ 。

练习18.11.8 证明在 \mathbb{R}^n 中的一族聚类 $\mathfrak{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_k\}$ 是凸可分的，当且仅当该族是相互凸可分的，即对于任意 $i \neq j$ ，任意两个聚类 \mathcal{G}_i 和 \mathcal{G}_j 是凸可分的。

练习18.11.9 (a) 如果 A 和 B 是两个如 $A \cap B = \emptyset$ 这样的凸集，那么 A 和 B 是线性可分的；

(b) 如果 \mathcal{G}_1 和 \mathcal{G}_2 是两个满足 $\text{hull}(\mathcal{G}_1) \cap \text{hull}(\mathcal{G}_2) = \emptyset$ 的簇，那么 \mathcal{G}_1 和 \mathcal{G}_2 是线性可分的。

练习18.11.10 (a) 求梯度 $\nabla_{w,b,\lambda} G$ ，其中 G 由(18.8.2)给出。

(b) 写出最小值近似序列的梯度下降递归式。

练习18.11.11 考虑 p 个聚类， $\mathcal{G}_1, \dots, \mathcal{G}_p$ ，它们是 \mathbb{R}^2 中的点集。写出成本函数，该函数将向量 e_j 作为聚类 \mathcal{G}_j 的标签，适用于所有 $j = 1, \dots, p$ 。

练习18.11.12 证明由(18.2.1)给出的划分所关联的熵最大，当且仅当所有集合 A_j 具有相等的测度。



第19章 生成模型

到目前为止，神经网络在两种主要类型的重要问题中很有用：回归问题和分类问题。在进行回归时，使用了具有一维输出的神经网络，其输出层具有线性激活函数。在分类问题的情况下，使用具有多维输出的神经网络很有用，其输出层具有softmax激活函数。

在本节中，我们将处理另一个重要应用，即生成模型的构建。如果回归问题可以预测未来模式，分类问题可以识别字体，那么生成模型可以生成与训练数据非常相似的示例。

19.1 生成模型的需求

考虑生成模型有很多原因。它们的成功应用使其在许多工业领域非常受欢迎。生成模型生成与训练数据非常相似的示例的能力，使得它们在可用数据不足甚至缺失的情况下很有用。在许多实际情况中，数据增强可能成本高昂且耗时。生成模型可以以较低的成本完成这项工作。我们将提供一些示例。

1. 出于安全考虑，不能在现实生活中对用于驾驶汽车或操作机器人的神经网络进行训练。因此，使用生成模型来提供一个模拟环境，在该环境中对汽车进行训练，直到模型准备好在现实生活中部署。

2. 在制定商业计划的情况下，可以采用生成模型来模拟未来可能的商业环境。这些可能的环境是通过与过去事件的相似性生成的。

3. 另一个应用是对金融合约（如衍生品）进行定价。这些合约的价格取决于基础股票的价格。传统的计算技术是使用蒙特卡罗方法。这需要假设大量可能的股票价格模拟情况，在此基础上对金融合约进行定价。然后将所有模拟合约价格的平均值视为计算得出的合约价格。生成模型可用于模拟股票市场并对这些市场中的金融合约进行定价，从而提供有关实际合约价格的概念。

4. 其他例子包括生成模型在卡通和电影行业的应用，在这些行业中需要生成逼真的现实环境，如建筑物、树木、山脉等。生成模型还可用于生成音乐、绘画、人脸等。例如，iGAN软件就是为了从粗略的手绘草图开始生成图片而创建的。

接下来我们将讨论两种类型的生成模型：提供密度估计的模型和生成样本的模型。

19.2 密度估计

这里的任务是创建一台机器，它能从给定分布 $p_{\text{data}}(x)$ 中观测许多样本，并且之后能够从该分布中生成更多样本。我们区分以下两种情况：参数分布和非参数分布。

参数情况 如果有理由相信给定分布 $p_{\text{data}}(x)$ 可以由参数分布 $p_{\text{model}}(x; \theta)$ 估计，那么我们只需要调整 θ ，使得在数据上评估的模型分布的对数似然最大。

$$\theta^* = \arg \max_{\theta} \mathbb{E}_{x \sim p_{\text{data}}} [\ln p_{\text{model}}(x; \theta)].$$

我们注意到在实践中，期望是作为平均值来计算的。

$$\mathbb{E}_{x \sim p_{\text{data}}} [\ln p_{\text{model}}(x; \theta)] = \frac{1}{n} \sum_{i=1}^n \ln p_{\text{model}}(x_i; \theta),$$

其中 $x_i \sim p_{\text{data}}$ 是从给定分布中提取的一个样本。

选择参数 θ 的另一种等效方法是要求数据与模型分布之间的库尔贝克 - 莱布勒散度最小。

$$\theta^* = \arg \min_{\theta} D_{KL}[p_{data} || p_{model}].$$

这可从计算中得出

$$\begin{aligned}\theta^* &= \arg \max_{\theta} \mathbb{E}_{x \sim p_{data}} [\ln p_{model}(x; \theta)] = \arg \min_{\theta} \mathbb{E}_{x \sim p_{data}} [-\ln p_{model}(x; \theta)] \\ &= \arg \min_{\theta} \left\{ - \int p_{data}(x) \ln p_{model}(x) dx \right\} = \arg \min_{\theta} S(p_{data}, p_{model}) \\ &= \arg \min_{\theta} [S(p_{data}, p_{model}) - H(p_{data})] = \arg \min_{\theta} D_{KL}(p_{data} || p_{model}),\end{aligned}$$

在此处，我们利用了给定数据 $H(p_{data})$ 的熵与 θ 的独立性，以及交叉熵 S 和库尔贝克 - 莱布勒散度的定义。

非参数情形 在这种情况下，思路如下：我们选取一个简单分布（例如均匀分布或高斯分布），然后对该分布的样本应用非线性变换 G ，以获得来自期望分布 $p_{data}(x)$ 的样本。

我们首先将介绍一些术语和符号。 x 取值的空间由 \mathcal{X} 表示，代表我们所关注的空间。要由 G 变换为 $p_{data}(x)$ 的简单分布 $p(z)$ ，称为代码分布；从中采样 z 的空间 \mathcal{Z} 称为潜在空间。现在，对于每个随机选择 $z \in \mathcal{Z}$ ，我们希望从空间 \mathcal{X} 中获得 $x = G(z)$ 作为样本。

将其与随机变量术语建立联系可能会很有用。潜在变量 z 是一个随机变量的实例，该随机变量由 Z 表示，在潜在空间 \mathcal{Z} 中取值。来自 $p_{data}(x)$ 的样本 x 被视为空间 \mathcal{X} 上随机变量 X 的实例。在这些表示法下，我们正在寻找一个将变量 Z 映射到 X 的变换 G 。这可以通过几种方式实现。

1. 我们假设数据分布 p_{data} 是一维的，并且我们将考虑一个均匀编码分布， $p_{code} \sim \text{Unif}[0, 1]$ 。我们将证明满足 $G(Z) = X$ 的变换 $G : [0, 1] \rightarrow \mathcal{X}$ 由 $G = F_{data}^{-1}$ 给出，其中 $F_{data}(x) = \int_{-\infty}^x p_{data}(s) ds$ 是随机变量 X 的累积分布函数。

这里的关键点是，如果 $Z \sim \text{Unif}[0, 1]$ ，那么对于所有的 $z \in [0, 1]$ ， $P(Z \leq z) = z$ 。现在，为了证明随机变量 X 和 $Y = G(Z)$ 具有相同的分布，我们进行如下计算

$$F_Y(x) = P(Y \leq x) = P(G(Z) \leq x) = P(Z \leq F_{data}(x)) = F_{data}(x), \quad \forall x \in \mathcal{X}.$$

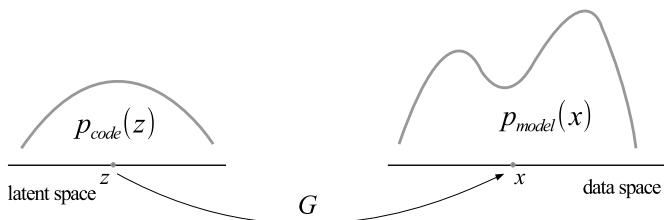


图19.1：潜在空间与数据空间之间的变换 G 。

综上所述，为了在遵循分布 p_{data} 的空间 \mathcal{X} 中生成 n 样本，我们均匀地选择 n 个随机实例 $z_i \in [0, 1]$ ，并考虑 $x_i = G(z_i)$ ，前提是 $G = F_{\text{data}}^{-1}$ 已知。那么所有构建的实例都满足 $x_i \sim p_{\text{data}}$ 。

由于在实际中分布很少是一维的，接下来我们将考虑多维情况。

2. 在这种情况下，编码分布 p_{code} 是多维的，且不一定是均匀的。潜在空间由 \mathcal{Z} 表示，并且我们考虑一个可逆的可微变换 $G : \mathcal{Z} \rightarrow \mathcal{X}$ ，其逆变换和雅可比行列式都可以计算，见图19.1。然后这种变量变换在 \mathcal{X} 上产生一个新的密度，其形式为

$$p_{\text{model}}(x) = p_X(x) = \frac{p_{\text{code}}(G^{-1}(x))}{|\det J_G(z)|}. \quad (19.2.1)$$

示例19.2.1 作为一个例子，我们将展示如何从正态多元分布 $\mathcal{N}(\mu, \Sigma)$ 构造样本，其中 μ 是 \mathbb{R}^n 中的给定向量， Σ 是一个对称、非退化且正定的 n 维矩阵。我们考虑仿射变换 $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ ，由 $G(z) = \mu + Az$ 给出，其中 $AA^T = \Sigma$ 。显然， $\det J_G(z) = \det A \neq 0$ ，因此 G 是可逆的，其逆为

$$G^{-1}(x) = A^{-1}(x - \mu).$$

我们选择编码分布为均值为零且协方差为单位矩阵的标准正态分布

$$p_{\text{code}}(z) = \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\|z\|^2},$$

即， $Z \sim \mathcal{N}(0, \mathbb{I}_n)$ 。使用公式(19.2.1)，我们得到以下密度：

$$\begin{aligned} p_X(x) &= \frac{1}{|\det A|} \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\|A^{-1}(x-\mu)\|^2} \\ &= \frac{1}{(\det \Sigma)^{1/2}} \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\langle A^{-1}(x-\mu), A^{-1}(x-\mu) \rangle^2} \\ &= \frac{1}{((2\pi)^n \det \Sigma)^{1/2}} e^{-\frac{1}{2}\langle (AA^T)^{-1}(x-\mu), (x-\mu) \rangle^2} \\ &= \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)}, \end{aligned}$$

其中我们使用了代数关系 $(A^{-1})^T A^{-1} = (A^T)^{-1} A^{-1} = (AA^T)^{-1}$ 。

值得注意的是，按照公式(19.2.1)使用此方法有一些缺点。首先，该方法假设雅可比行列式和变换 G 的逆是可计算的。此方法的另一个缺点是潜在空间的维度等于数据空间的维度，这一事实涉及太多参数。如果数据空间表示一幅图像，那么使用此方法将涉及与图像中的像素一样多的潜在变量！这就是构建一种潜在空间维度比数据空间小得多的方法的原因。

生成模型的第二种类型是样本生成，它生成更多样本而不是寻找密度函数。在这一组中，我们将讨论生成对抗网络 (GANs) 和生成矩匹配网络。我们将在下一节首先讨论对抗方面。

19.3 对抗博弈

我们考虑两个参与竞争博弈的玩家，他们争夺同一个收益函数：一个希望它高，而另一个希望它低。这样，玩家就成了对手，对相反的奖励感兴趣。他们每个人都控制收益函数的某些变量，并试图调整这些变量以获得最大利益。对于某些参数值，玩家可能会达到一种平衡。这可以通过以下例子来理解。

我们可以将一个玩家视为某种产品的卖家，另一个视为买家。双方必须就产品价格达成一致。卖家会试图抬高价格，而买家希望价格尽可能低。在谈判过程结束时，卖家同意充分降低产品价格，买家同意支付足够多的钱，这样双方就达成了买卖协议。

通过这种方式，他们得出均衡价格，该价格对应于博弈的纳什均衡点。

该问题可以如下表述为一个极小极大问题。考虑一个收益函数 $V(x, y)$ ，它取决于变量 x 和 y 。第一个参与者控制变量 x ，第二个参与者控制变量 y 。第一个参与者希望收益函数 $V(x, y)$ 最小化，而第二个参与者希望 $V(x, y)$ 最大化（这等同于最小化 $-V(x, y)$ ）。该问题可以表述为

$$(x^*, y^*) = \arg \max_y \min_x V(x, y).$$

均衡点可以通过如下连续时间变体的同步梯度下降法获得。

由于第一个参与者控制变量 x 并打算最小化 $V(x, y)$ ，它应该在 V 的负梯度方向上以步长 $\eta = \Delta t$ 调整 x

$$x(t + \Delta t) = x(t) - \eta \frac{\partial V}{\partial x}.$$

同样，由于第二个参与者控制变量 y 并希望最大化 $V(x, y)$ ，它应该朝着 V 的正梯度方向将 y 调整一个步长 $\eta = \Delta t$

$$y(t + \Delta t) = y(t) + \eta \frac{\partial V}{\partial y}.$$

取 $\Delta t \rightarrow 0$ ，我们得到一个学习率为无穷小的同步梯度下降，其中学习过程遵循一条平滑轨迹 $(x(t), y(t))$ ，该轨迹满足以下连续时间微分系统：

$$\frac{dx}{dt} = -\frac{\partial V}{\partial x} \quad (19.3.2)$$

$$\frac{dy}{dt} = \frac{\partial V}{\partial y}. \quad (19.3.3)$$

初始条件为 $(x(0), y(0)) = (x_0, y_0)$ ，平衡点（若存在）可通过极限 $(x^*, y^*) = \lim_{t \rightarrow \infty} (x(t), y(t))$ 得到。由于在平衡状态下我们有 $\frac{dx^*}{dt} = \frac{dy^*}{dt} = 0$ ，那么平衡点满足该系统

$$\begin{aligned} \frac{\partial V}{\partial x} &= 0 \\ \frac{\partial V}{\partial y} &= 0. \end{aligned}$$

值得注意的是，收益函数 $V(x, y)$ 沿学习曲线 $(x(t), y(t))$ 的变化由其微分给出，该微分由下式给出

$$dV = \frac{\partial V}{\partial x} dx + \frac{\partial V}{\partial y} dy = -\dot{x}dx + \dot{y}dy.$$

示例19.3.1 以每单位 p 的价格出售 q 单位某产品所获得的收入为 $V(p, q) = pq$ 。卖方可以控制价格 p ，并希望使收入 $V(p, q)$ 最大化。买方可以控制售出的单位数量 q ，并希望使支付的价格 $V(p, q)$ 最小化。这就变成了一个极小极大问题，将使用具有无穷小学习率的同步梯度下降法来解决。学习过程遵循轨迹 $(p(t), q(t))$ ，它满足连续时间系统(19.3.2)-(19.3.3)，在这种情况下变为

$$\begin{aligned}\frac{dp}{dt} &= q \\ \frac{dq}{dt} &= -p.\end{aligned}$$

再对 t 求一次导数，我们得到 $\ddot{p} = -p$ 和 $\ddot{q} = -q$ ，这意味着 $p(t)$ 和 $q(t)$ 都是 $\cos t$ 和 $\sin t$ 的线性组合

$$\begin{aligned}p(t) &= A_1 \cos t + B_1 \sin t \\ q(t) &= A_2 \cos t + B_2 \sin t,\end{aligned}$$

其中 A_i 和 B_i 为常数。将其代入微分方程组并利用初始条件，我们确定这些常数并将解写成

$$\begin{aligned}p(t) &= p_0 \cos t + q_0 \sin t \\ q(t) &= q_0 \cos t - p_0 \sin t.\end{aligned}$$

由于这可以用矩阵形式表示为 $(p, q)^T = \mathcal{R}(t)(p_0, q_0)^T$ ，其中

$$\mathcal{R}(t) = \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cot t \end{pmatrix}$$

表示在角度为 t 的平面内的旋转，由此可知解 $(p(t), q(t))$ 遵循以原点为中心、半径为 $r = \sqrt{p_0^2 + q_0^2}$ 的圆形轨迹。只要 $r \neq 0$ ，这个解就不会趋近于任何平衡点。唯一的平衡点在原点处得到， $(p^*, q^*) = (0, 0)$ ，并且这种情况发生在 $p_0 = q_0 = 0$ 的情形下。

19.4 生成对抗网络

生成对抗网络 (GANs) 由古德费洛等人于2014年提出[47]，如今被认为是最强大的生成模型之一。

GANs代表了由两个神经网络进行的非合作博弈：判别器和生成器。这些网络充当对手，拥有相反的奖励，一个网络的最坏情况输入由另一个网络产生。在这场竞争博弈中，每个网络都迫使另一个网络改进。

生成器网络 生成器是一个网络，其输入是从潜在空间中选择的一些随机噪声，输出是一幅图像 x ，该图像应该类似于数据空间中的图像。生成器输出为 $x = G(z; \theta^{(g)})$ ，其中 $\theta^{(g)}$ 是生成器网络参数， z 是潜在空间 \mathcal{Z} 中的一个潜在向量变量。我们应该用 Z 表示以 z 为实例在潜在空间 \mathcal{Z} 中取值的随机变量，并用 $p_{\text{code}}(z)$ 表示其概率密度。我们用 X 表示输出随机变量，即 $X = G(Z; \theta^{(g)})$ ，并用 $p_{\text{model}}(x; \theta^{(g)})$ 表示其密度。生成器函数 G 必须是可微的， Z 的维度小于 X 的维度。

判别器网络 判别器网络起到分类器的作用。这意味着判别器是一个接收输入 x （例如，一幅图像）并产生一个介于0和1之间的数字 $D(x; \theta^{(d)})$ 的网络，其中 $\theta^{(d)}$ 是判别器网络参数。这个数字可以被视为输入 x 被视作真实训练数据的概率，当判别器完全拒绝输入 x 属于训练数据时，该值被赋值为0。如果输入是一幅图像，判别器可以被视为一个卷积网络，并且可以使用梯度下降进行训练。判别器和生成器网络的功能可以在图19.2中看到。

生成器的任务是使判别器相信其输出是真实的训练数据；这意味着生成器希望 $D(G(z))$ 接近1。另一方面，判别器的目标是证明生成器是错误的，也就是说，对于所有 $z \in \mathcal{Z}$ ，它希望输出一个接近0的 $D(G(z))$ 值。

训练过程按周期进行。一开始，生成器并不那么智能，会产生一些随机噪声。随着时间的推移，生成器会改进，生成的图像越来越类似于训练数据集中的图像。与此同时，判别器使用生成器生成的一批图像和从训练数据集中选择的一批图像，对虚假图像和真实图像进行训练。

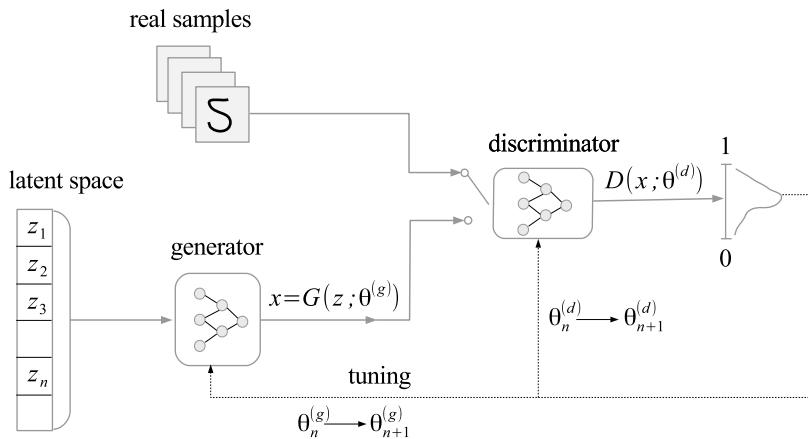


图19.2：生成对抗网络的剖析：将潜在变量输入生成器以生成“假”数据。然后反复对两个小批量数据进行采样，一个来自真实数据，另一个来自生成的数据，并让判别器判断其是假还是真。利用这个判断，我们可以更新网络参数以改进生成对抗网络的功能。

随着判别器在其工作中变得更加熟练，生成器将倾向于生成越来越类似于数据空间中的图像。如果训练成功，生成器最终将生成与真实训练图像完全无法区分的图像。此外，最终判别器将无法区分输出是假还是真，提供的概率输出为0.5。此时判别器变得无用，可以忽略，只保留生成器网络。

示例19.4.1 考虑一个旨在生成质数的生成对抗网络（GAN）。我们假设训练数据是前 n 个质数的集合， $\mathcal{X} = \{p_1, p_2, \dots, p_n\}$ 。如果生成器网络生成一个数字 x ，判别器网络将通过检查 x 与 p_i 之间的除法是否为偶数来测试这个数字是否为质数。如果存在一个质数 p_i 是 x 的除数，判别器产生 $D(x) = 0$ ，否则产生 $D(x) = 1$ 。一开始，生成器可能会生成，例如， $x = p_1 p_2$ ，判别器会很容易地将其分类为非质数。但随着时间的推移，生成器会了解到它必须生成一个大于所有提供的质数且不是它们倍数的数字，比如 $x = p_1 p_2 \dots p_n + 1$ 。在这种情况下，生成器会产生 $D(x) = 1$ ，即会将数字 x 分类为质数。生成器可以利用这些信息，下次产生输出 $x = p_1 \dots p_n (p_1 p_2 \dots p_n + 1) + 1$ ，依此类推。

我们注意到可以使用几个判别函数 $D(x)$ 。如果我们用 $\text{div}(x)$ 表示已知素数集合中 x 的除数数量，那么我们也可以定义 $D(x) = e^{2-\text{div}(x)}$ ，或者 $D(x) = \frac{2}{\text{div}(x)}$ ，并将它们视为判别函数。

损失函数 判别器试图最大化收益

$$V(G, D) = \mathbb{E}_{x \sim p_{\text{data}}} [\ln D(x)] + \mathbb{E}_{x \sim p_{\text{model}}} [\ln(1 - D(x))], \quad (19.4.4)$$

最大化第一项可确保对真实数据进行正确分类，而最大化第二项可确保对模型生成的数据进行正确分类。这是因为判别器的输出 $D(x)$ 对于真实数据 x 往往接近1，而对于生成数据 x ， $1 - D(x)$ 往往接近1。

示例19.4.2 判别器网络最简单的示例是一个Sigmoid神经元。如果输入是向量 $x^T = (x_1, \dots, x_n)$ ，那么判别器会产生概率 $D(x; w, b) = \sigma(w^T x + b) = \frac{1}{1+e^{-(w^T x + b)}}$ ，其中 w 和 b 分别表示权重和偏差。在接下来的内容中，我们将评估收益函数 (19.4.4)。利用对数函数和Softplus函数的性质，我们有

$$\ln D(x) = -\ln(1 + e^{-(w^T x + b)}) = -sp(-(w^T x + b)) = x - sp(w^T x + b),$$

然后

$$\mathbb{E}_{x \sim p_{\text{data}}} [\ln D(x)] = \mu_x - \mathbb{E}_{x \sim p_{\text{data}}} [sp(w^T x + b)],$$

其中 μ_x 表示生成器输出的均值。我们还有

$$1 - \ln D(x) = 1 - \sigma(w^T x + b) = \sigma(-w^T x - b) = \frac{1}{e^{w^T x + b}},$$

使得

$$\ln(1 - D(x)) = -\ln(1 + e^{w^T x + b}) = -sp(w^T x + b).$$

求期望，我们得到

$$\mathbb{E}_{x \sim p_{\text{model}}} [\ln(1 - D(x))] = -\mathbb{E}_{x \sim p_{\text{model}}} [sp(w^T x + b)].$$

因此，收益函数变为

$$\begin{aligned} V(G, D) &= \mu_x - \mathbb{E}_{x \sim p_{\text{data}}} [sp(w^T x + b)] - \mathbb{E}_{x \sim p_{\text{model}}} [sp(w^T x + b)] \\ &= \mu_x - \int (p_{\text{data}}(x) + p_{\text{model}}(x)) sp(w^T x + b) dx \\ &= \mu_x - \mathbb{E}_{x \sim p_m} [sp(w^T x + b)], \end{aligned}$$

其中 $p_m(x) = \frac{1}{2}(p_{\text{data}}(x) + p_{\text{model}}(x))$ 。

生成器的收益为 $-V(G, D)$ (因此有零和博弈之名), 即生成器旨在使 $-V(G, D)$ 最大化, 或者等效地, 使 $V(G, D)$ 最小化。

由于每个网络都试图最大化自身收益, 现在可以将其设定为如下极小极大问题:

$$G^* = \arg \min_G \max_D V(G, D),$$

其中 G^* 表示最优生成器。

目前我们将生成器视为固定的, 并处理内部的最大循环。对于给定的生成器 G , 最优鉴别器函数 D_G^* 由下式给出

$$D_G^* = \arg \max_D V(G, D). \quad (19.4.5)$$

下一个结果给出了这个鉴别器的值。

命题19.4.3 对于固定的生成器 G , 最优鉴别器(19.4.5)由下式给出

$$D_G^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)}. \quad (19.4.6)$$

证明: 考虑在空间 \mathcal{X} 上定义的分布 p_{data} 和 p_{model} , 收益函数可以写成如下积分形式:

$$V(G, D) = \int_{\mathcal{X}} [p_{\text{data}}(x) \ln D(x) + p_{\text{model}}(x) \ln(1 - D(x))] dx.$$

由于 G 是固定的, 我们将收益视为 $D(x)$ 的泛函, 形式为

$$F(D) = \int_{\mathcal{X}} L(D(x)) dx,$$

其拉格朗日函数为

$$L(D) = p_{\text{data}}(x) \ln D(x) + p_{\text{model}}(x) \ln(1 - D(x)).$$

临界点满足变分方程

$$\frac{dL(D)}{dD} = 0.$$

这就变成了

$$\frac{p_{\text{data}}(x)}{D(x)} - \frac{p_{\text{model}}(x)}{1 - D(x)} = 0.$$

求解 D ，我们得到解

$$D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_{model}(x)}.$$

为了表明这个临界点对应于一个最大值，我们考虑二阶变分并表明它具有负值

$$\frac{d^2 L(D)}{d D^2} = -\frac{p_{data}(x)}{D(x)^2} - \frac{p_{model}(x)}{(1 - D(x))^2}.$$

因此，对于给定的 G ， $D_G^*(x)$ 对应于 $V(G, D)$ 的最大值。■

注记19.4.4 (i) 我们注意到最优值 $D_G^*(x)$ 通过密度 $p_{model}(x)$ 依赖于 G ， $p_{model}(x)$ 是 $G(z)$ 的密度，其中 $z \sim p_{code}$ 。

(ii) 这是最优值存在唯一性的理论结果。然而，如果未给出 p_{data} ，该结果不实用。

现在，生成器最优值 G^* 由下式给出

$$G^* = \arg \min_G V(G, D_G^*).$$

为了找到它，我们首先需要在其最优点处评估收益 $V(G, D)$ 的最大值。

命题19.4.5 如果 D_{JS} 表示由(3.7.4)给出的詹森 - 香农散度，那么收益函数的最大值是

$$V(G, D_G^*) = 2D_{JS}(p_{model}(x) || p_{data}(x)) - \ln 4. \quad (19.4.7)$$

将公式(19.4.6)给出的值代入收益中，我们得到

$$\begin{aligned} V(G, D_G^*) &= \int_{\mathcal{X}} \left[p_{data}(x) \ln D_G^*(x) + p_{model}(x) \ln(1 - D_G^*(x)) \right] dx \\ &= \int_{\mathcal{X}} \left[p_{data}(x) \ln \frac{p_{data}(x)}{p_{data}(x) + p_{model}(x)} \right. \\ &\quad \left. + p_{model}(x) \ln \frac{p_{model}(x)}{p_{data}(x) + p_{model}(x)} \right] dx. \end{aligned}$$

在下面我们将加上然后减去 $\ln 2p_{data}(x)$ 和 $\ln 2p_{model}(x)$ ，并使用一些代数运算将其拆分为三个积分

$$\begin{aligned} V(G, D_G^*) &= \int_{\mathcal{X}} \left[(\ln 2 - \ln 2)p_{data}(x) + p_{data}(x) \ln \frac{p_{data}(x)}{p_{data}(x) + p_{model}(x)} \right. \\ &\quad \left. + (\ln 2 - \ln 2)p_{model}(x) + p_{model}(x) \ln \frac{p_{model}(x)}{p_{data}(x) + p_{model}(x)} \right] dx \\ &= I_1 + I_2 + I_3, \end{aligned}$$

哪里

$$\begin{aligned} I_1 &= - \int_{\mathcal{X}} \ln 2(p_{data}(x) + p_{model}(x)) dx \\ I_2 &= \int_{\mathcal{X}} p_{data}(x) \left[\ln 2 + \ln \frac{p_{model}(x)}{p_{data}(x) + p_{model}(x)} \right] dx \\ I_3 &= \int_{\mathcal{X}} p_{model}(x) \left[\ln 2 + \ln \frac{p_{model}(x)}{p_{data}(x) + p_{model}(x)} \right] dx. \end{aligned}$$

在下面我们计算每个积分。利用 p_{data} 和 p_{model} 是概率密度函数这一事实，我们有

$$I_1 = -\ln 2 \left[\underbrace{\int_{\mathcal{X}} p_{data}(x) dx}_{=1} + \underbrace{\int_{\mathcal{X}} p_{model}(x) dx}_{=1} \right] = -2\ln 2 = -\ln 4.$$

利用对数函数的性质，我们将第二个积分计算为库尔贝克 - 莱布勒散度

$$\begin{aligned} I_2 &= \int_{\mathcal{X}} p_{data}(x) \ln \frac{2p_{model}(x)}{p_{data}(x) + p_{model}(x)} dx \\ &= \int_{\mathcal{X}} p_{data}(x) \ln \frac{p_{model}(x)}{\frac{1}{2}(p_{data}(x) + p_{model}(x))} dx \\ &= D_{KL}\left(p_{data}(x) \parallel \frac{p_{data}(x) + p_{model}(x)}{2}\right). \end{aligned}$$

第三个积分可以用类似的方法计算

$$\begin{aligned} I_3 &= \int_{\mathcal{X}} p_{model}(x) \ln \frac{2p_{model}(x)}{p_{data}(x) + p_{model}(x)} dx \\ &= \int_{\mathcal{X}} p_{model}(x) \ln \frac{p_{model}(x)}{\frac{1}{2}(p_{data}(x) + p_{model}(x))} dx \\ &= D_{KL}\left(p_{model}(x) \parallel \frac{p_{data}(x) + p_{model}(x)}{2}\right). \end{aligned}$$

利用詹森 - 香农散度的定义(3.7.4)，我们得到

$$\begin{aligned} V(G, D_G^*) &= I_1 + I_2 + I_3 \\ &= -\ln 4 + D_{KL}\left(p_{data}(x) \parallel \frac{p_{data}(x) + p_{model}(x)}{2}\right) \\ &\quad + D_{KL}\left(p_{model}(x) \parallel \frac{p_{data}(x) + p_{model}(x)}{2}\right) \\ &= -\ln 4 + 2D_{JS}(p_{model}(x) \parallel p_{data}(x)). \end{aligned}$$

下一个结果展示了生成对抗网络 (GANs) 的一个预期事实，即，在平衡收敛时模型分布近似数据分布 (Goodfellow 等人[47])。

命题19.4.6全局最小值

$$G^* = \arg \min_G V(G, D_G^*)$$

当且仅当 $p_{\text{model}} = p_{\text{data}}$ 时达到。在这个最小值点处的值是 $-\ln 4$ 。

证明：利用关系式(19.4.7)，我们有

$$\begin{aligned} G^* &= \arg \min_G V(G, D_G^*) = \arg \min_G \left(2D_{JS}(p_{\text{model}}(x) || p_{\text{data}}(x)) - \ln 4 \right) \\ &= \arg \min_G \left(2D_{JS}(p_{\text{model}}(x) || p_{\text{data}}(x)) \right). \end{aligned}$$

根据命题3.7.1，右侧表达式当且仅当 $p_{\text{model}}(x) = p_{\text{data}}(x)$ 时达到全局最小值零。在这种情况下，我们显然有

$$\min_G V(G, D_G^*) = V(G^*, D_G^*) = -\ln 4.$$

通过将 $p_{\text{model}} = p_{\text{data}}$ 代入(19.4.6)得到平衡时的判别输出

$$D^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)} = \frac{1}{2}.$$

判别器的输出值为0.5这一事实意味着它无法区分生成的数据 x 是假的还是真的。在平衡状态下，收益函数的最大值是

$$\begin{aligned} V(G^*, D^*) &= \int_{\mathcal{X}} \left[p_{\text{data}}(x) \ln D^*(x) + p_{\text{model}}(x) \ln(1 - D^*(x)) \right] dx \\ &= \int_{\mathcal{X}} \left[p_{\text{data}}(x) \ln \frac{1}{2} + p_{\text{model}}(x) \ln \frac{1}{2} \right] dx \\ &= \ln \frac{1}{2} \int_{\mathcal{X}} \left[p_{\text{data}}(x) + p_{\text{model}}(x) \right] dx \\ &= 2 \ln \frac{1}{2} = -\ln 4, \end{aligned}$$

这恢复了前一个命题的结果。

一般来说，潜在空间 \mathcal{Z} 的维度小于数据空间 \mathcal{X} 的维度。在它们相等且生成器变换 $G : \mathcal{Z} \rightarrow \mathcal{X}$ 可微且可逆的情况下，模型和代码分布通过以下方式相关

$$p_{\text{model}}(x) = p_{\text{code}}(G^{-1}(x)) \det J_{G^{-1}(x)}.$$

此外，收益 $V(G, D)$ 的第二项可以写成关于编码密度的期望，如下所示

$$\mathbb{E}_{x \sim p_{model}} [\ln(1 - D(x))] = \mathbb{E}_{z \sim p_{code}} [\ln(1 - D(G(z)))].$$

计算涉及变量 $z = G^{-1}(x)$ 的变化，如下所示：

$$\begin{aligned}\mathbb{E}_{x \sim p_{model}} [\ln(1 - D(x))] &= \int_{\mathcal{X}} \ln(1 - D(x)) p_{model}(x) dx \\ &= \int_{\mathcal{X}} \ln(1 - D(x)) p_{code}(G^{-1}(x)) \det J_{G^{-1}(x)} dx \\ &= \int_{\mathcal{Z}} \ln(1 - D(G(z))) p_{code}(z) dz \\ &= \mathbb{E}_{z \sim p_{code}} [\ln(1 - D(G(z)))].\end{aligned}$$

注记19.4.7 极小极大问题 $\min_G \max_D V(G, D)$ 与 $\max_D \min_G V(G, D)$ 不同。前者是正确的版本，而后者效果不佳，因为生成器可能会将所有质量集中在最可能的点上，从而使判别器饱和。这会导致模式崩溃，即GAN产生相同输出的一种情况。

训练过程一个GAN通过应用同步随机梯度下降方法进行训练。我们反复对两个小批量数据进行采样，一个来自训练集，另一个来自生成的样本。然后我们同时对两个交叉的参与者运行梯度下降。这将导致每个参与者在梯度方向或相反方向上进行一次更新，如下所示：

$$\begin{aligned}\theta_{n+1}^{(d)} &= \theta_n^{(d)} - \eta \nabla_{\theta^{(d)}} V(\theta_n^{(d)}) \\ \theta_{n+1}^{(g)} &= \theta_n^{(g)} + \eta \nabla_{\theta^{(g)}} V(\theta_n^{(g)}),\end{aligned}$$

其中 $\eta > 0$ 是学习率， V 是由(19.4.4)定义的收益。梯度可以如下计算。利用 $D(x) = D(x, \theta^{(d)})$ 以及 p_{model} 和 p_{data} 与 $\theta^{(d)}$ 无关，我们有

$$\begin{aligned}\nabla_{\theta^{(d)}} V &= \nabla_{\theta^{(d)}} \mathbb{E}_{x \sim p_{data}} [\ln D(x, \theta^{(d)})] + \nabla_{\theta^{(d)}} \mathbb{E}_{x \sim p_{model}} [\ln(1 - D(x, \theta^{(d)}))] \\ &= \mathbb{E}_{x \sim p_{data}} \left[\frac{1}{D(x)} \frac{\partial D(x)}{\partial \theta^{(d)}} \right] - \mathbb{E}_{x \sim p_{model}} \left[\frac{1}{1 - D(x)} \frac{\partial D(x)}{\partial \theta^{(d)}} \right].\end{aligned}$$

如果 $D(x) = \sigma(a(x, \theta^{(d)}))$ ，其中 σ 是逻辑 sigmoid 函数，那么使用

$$\frac{\partial D(x)}{\partial \theta^{(d)}} = \sigma'(a(x, \theta^{(d)})) \frac{\partial a(x)}{\partial \theta^{(d)}} = D(x)(1 - D(x)) \frac{\partial a(x)}{\partial \theta^{(d)}},$$

梯度计算可以继续如下

$$\begin{aligned}\nabla_{\theta^{(d)}} V &= \mathbb{E}_{x \sim p_{\text{data}}} \left[(1 - D(x)) \frac{\partial a(x)}{\partial \theta^{(d)}} \right] - \mathbb{E}_{x \sim p_{\text{data}}} \left[D(x) \frac{\partial a(x)}{\partial \theta^{(d)}} \right] \\ &= \frac{\partial}{\partial \theta^{(d)}} \mathbb{E}_{x \sim p_{\text{data}}} [a(x)] - 2 \mathbb{E}_{x \sim \frac{1}{2}(p_{\text{data}} + p_{\text{model}})} \left[D(x) \frac{\partial a(x)}{\partial \theta^{(d)}} \right].\end{aligned}$$

值得注意的是，平衡点处的第一个期望等于库尔贝克-莱布勒散度 $D_{KL}(p_{\text{data}} \parallel p_{\text{model}})$ ，见练习19.7.5。我们假设概率密度 p_{data} 和 p_{model} 在无穷远处足够快地减小到零，使得求导算子与期望算子可交换。

梯度 $\nabla_{\theta^{(g)}} V$ 的计算方式类似，见练习19.7.7。

即使生成对抗网络被认为是最成功的生成模型，仍然存在一些未解决的问题，这些问题时当今研究的活跃领域。其中包括：梯度消失、模式崩溃、解决与计数、视角和全局结构相关的问题。

19.5 生成矩匹配网络

生成矩匹配网络由Li等人于2015年提出[76]。在这种情况下，生成器通过矩匹配进行训练，这是一种使生成器的输出矩尽可能接近训练数据相应矩的技术。

这个想法基于这样一个结果：如果两个随机变量具有相等的矩，那么它们的分布是相同的。如果随机变量属于特定类型，则所需的矩会更少。特别是，具有相同均值和方差的两个高斯随机变量是相同的。

这个生成模型是一个生成器网络，例如卷积网络，它被提供一个来自均匀（或高斯）分布的样本 z ，并产生一个输出 x ，例如一张人脸或数字的图像。如果网络参数用 θ 表示，那么结果变量 X 可以根据输入变量 Z 写成 $X = G(Z; \theta)$ ，例如 $Z \sim \text{Unif}[0, 1]$ 。 X 的密度将用 p_θ 表示。

我们假设描述训练数据（例如一组人脸或MNIST数据数字）的相应随机变量由 Y 表示，并且具有密度 q 。网络参数 θ 必须进行调整，以使 X 和 Y 的分布尽可能接近，这将通过匹配前 k 阶矩来完成。为此，我们选择函数 $\phi(x) = (x, x^2, \dots, x^k)^T$ 并考虑使由(3.8.5)定义的最大均值差异最小化的参数 θ

$$\theta^* = \arg \min_{\theta} d_{MMD}(p_{\theta}, q) = \arg \min_{\theta} \|\mu_{\phi}(G(Z; \theta)) - \mu_{\phi}(Y)\|_{Eu}.$$

在实践中，我们选取 n 个随机输入 $\{z_1, \dots, z_n\}$ ，这些输入会产生一个输出样本 $\{x_1, \dots, x_n\}$ ，然后从训练数据中选择另一个随机样本 $\{y_1, \dots, y_m\}$ 。为了匹配这两个样本所基于的分布的矩，我们需要考虑该参数

$$\theta^* = \arg \min_{\theta} \left\{ \frac{1}{n^2} \sum_{i,j} K(x_i, x_j) + \frac{1}{m^2} \sum_{i,j} K(y_i, y_j) - \frac{2}{mn} \sum_{i,j} K(x_i, y_j) \right\},$$

其中 $K(x, y) = \phi(x)^T \phi(y)$ ，并且我们使用了公式(3.8.6)。生成器网络使用反向传播方法进行训练。

19.6 总结

我们已经展示了生成网络如何用于从给定概率分布中生成样本，或者生成类似于训练集中数据的示例。生成模型的思想是为模型提供一个随机种子，并获得一个类似于训练数据或数据分布样本的结果。

GANs被赋予了核心角色，它是一种生成模型，其架构由两个网络组成，一个生成器和一个判别器网络，它们进行竞争游戏。这种非合作游戏通常被表示为画作伪造者（生成器）和专家调查员（判别器）之间的关系。伪造者试图欺骗专家，而专家则试图识别出伪制品是假的。专家对假画和真画进行训练。从长远来看，他们相互促使对方改进：伪造者做得非常好，以至于专家对一幅画是假是真感到困惑。在最常见的设置中，游戏是一个零和游戏，或者是一个极小极大问题。平衡点是收益的鞍点。

另一个讨论的模型是生成矩匹配网络。它由一个生成器网络组成，其参数经过调整，使得输出矩与训练分布的相应矩相匹配。

19.7 练习

练习 19.7.1 当学习率为 $\eta > 0$ 时，我们在离散情况下对具有收益 $V(x, y) = xy$ 的极小极大问题应用同步梯度下降法。

- (i) 编写系统更新内容。
- (ii) 根据 (x_0, y_0) 写出系统状态 (x_n, y_n) 。
- (iii) 序列 (x_n, y_n) 是否收敛？

练习19.7.2 考虑两个玩家之间的竞争博弈，他们争夺线性收益函数 $V(x, y) = ax + by$ ，其中 $a^2 + b^2 \neq 0$ 。第一个玩家控制变量 x 并希望使 $V(x, y)$ 最小化，而第二个玩家控制变量 y 并希望使收益函数最大化。

(i) 写出由具有无穷小学习率的同步梯度下降给出的学习动态的微分系统，并求解学习轨迹。

(ii) 证明不存在平衡解。

练习19.7.3 考虑两个玩家之间的竞争博弈，其收益函数为 $V(x, y) = \frac{1}{2}(x^2 - y^2)$ 。控制 x 变量的玩家旨在使收益最小化，而控制 y 变量的玩家则希望使其最大化。证明该博弈有唯一的均衡解。同时考虑连续学习和离散学习。

练习19.7.4 我们回忆一下，平面中向量场 $U = (U_1, U_2)$ 的扩张或收缩由其散度 $\operatorname{div} U = \partial_x U_1 + \partial_y U_2$ 给出。沿着学习流 $(x(t), y(t))$ 的速度向量场由 $U(x, y) = (\dot{x}(t), \dot{y}(t))$ 定义

- (i) 证明与收益函数 $V(x, t) = x^2 + y^2$ 相关的学习流的速度向量具有零散度。
- (ii) 证明学习流的速度向量散度为零，当且仅当相关的收益函数具有 $V(x, t) = F(x+y) + G(x-y)$ 的形式，其中 F 和 G 是两个任意的二次可微函数。
- (iii) 证明如果学习流的散度为零，那么沿着该流不存在平衡点。

练习19.7.5 (i) 考虑一个具有固定生成器 G 的GAN。证明最优判别器满足 $D_G^*(x) = \sigma(a(x))$ ，其中

$$e^{a(x)} = \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}$$

且 σ 是逻辑函数。

(ii) 当由 (i) 给出 $a(x)$ 时，证明 $\mathbb{E}_{x \sim p_{\text{data}}} [a(x)] = D_{KL}(p_{\text{data}} \parallel p_{\text{model}})$ 。

练习19.7.6 考虑一个具有模型分布 $p_{model}(x; \theta^{(g)})$ 的生成对抗网络。我们假设参数向量 $\theta^{(g)}$ 是通过最大似然估计得到的，即

$$\theta^{(g)} = \theta_{MLE} = \arg \max_{\theta} \mathbb{E}_{x \sim p_{data}} [\ln p_{model}(x; \theta)].$$

找到一个函数 $f(x)$ 使得 $\theta^* = \theta_{MLE}$ ，其中

$$\theta^* = \arg \max_{\theta} \mathbb{E}_{x \sim p_{model}} [f(x)].$$

练习19.7.7 考虑由(19.4.4)给出的生成对抗网络的收益。

(i) 证明

$$\nabla_{\theta^{(g)}} V = \int_{\mathcal{X}} \ln(1 - D(x)) \nabla_{\theta^{(g)}} p_{model}(x; \theta^{(g)}) dx.$$

(ii) 假设生成器函数 $G: \mathcal{Z} \rightarrow \mathcal{X}, x = G(z, \theta^{(g)})$ 是可逆且可微的。证明

$$\nabla_{\theta^{(g)}} V = -\mathbb{E}_{z \sim p_{code}} \left[\frac{1}{1 - D(G(z))} \frac{\partial D(x)}{\partial x} \Big|_{x=G(z)} \frac{\partial G(z)}{\partial \theta^{(g)}} \right].$$

(iii) 假设 $D(x) = \sigma(a(x))$ ，其中 σ 是逻辑函数。证明在 (ii) 部分的梯度公式变为

$$\nabla_{\theta^{(g)}} V = -\mathbb{E}_{z \sim p_{code}} \left[D(G(z)) \frac{da(x)}{dx} \Big|_{x=G(z)} \frac{\partial G(z)}{\partial \theta^{(g)}} \right].$$

练习19.7.8 考虑一个零和博弈，其中一个玩家控制变量 y 并希望最小化交叉熵收益

$$V(y, \hat{y}) = y \ln \hat{y} + (1 - y) \ln(1 - \hat{y}),$$

而另一个玩家控制变量 \hat{y} 并打算最小化收益 $-V(y, \hat{y})$ 。两个变量都在 $(0, 1)$ 中取值。求该博弈的平衡点。

练习19.7.9 为了避免梯度消失问题，使用了生成器收益的以下选择：

$$J^{(G)} = \mathbb{E}_{z \sim p_{code}} [\ln D(G(z, \theta^{(g)})�)].$$

在这种情况下，生成器会最大化判别器出错的对数概率。找到梯度 $\nabla_{\theta^{(g)}} J^{(G)}$ 。假设判别器是最优的。



第20章 随机网络

本章讨论随机网络，如霍普菲尔德网络和玻尔兹曼机。霍普菲尔德网络是一组相互作用的感知器，直到根据神经元之间的相互作用定义的某个代价函数（称为能量函数）最小化。它们在解决组合优化问题方面很有用。玻尔兹曼机类似于霍普菲尔德网络，其中感知器已被二元随机神经元取代。这样，玻尔兹曼机可以被视为霍普菲尔德网络的噪声版本。它们用于避免霍普菲尔德网络陷入能量函数的局部最小值。我们还介绍了玻尔兹曼机的平衡分布、其熵及其相关的费希尔信息度量。

20.1 随机神经元

我们已经看到，确定性感知器是一个计算单元，它接受 n 个输入， x_1, \dots, x_n ，并提供一维输出

$$y = H(w^T \mathbf{x} + b) = H\left(\sum_{i=1}^n w_i x_i + b\right) = \begin{cases} 1, & \text{if } w^T \mathbf{x} + b > 0 \\ 0, & \text{otherwise.} \end{cases}$$

一个二元随机神经元是一个接受 n 个输入 x_1, \dots, x_n 并给出输出 Y 的单元，该输出由二元随机变量 $Y \in \{0, 1\}$ 给出，其具有如下分布

$$\begin{aligned} P(Y = 1|\mathbf{x}) &= \sigma(w^T \mathbf{x} + b), \\ P(Y = 0|\mathbf{x}) &= 1 - \sigma(w^T \mathbf{x} + b) = \sigma(-w^T \mathbf{x} - b), \end{aligned}$$

其中 σ 为逻辑函数。如果输入是确定性的，则由 \mathbf{x} 给出的条件可以省略。

当输入满足 $w^T \mathbf{x} + b > 0$ 时，确定性感知器提供值1（概率为1），而随机神经元提供值1的概率为 $\sigma(w^T \mathbf{x} + b)$ 。

费希尔度量 随机神经元关于其权重的信息由其费希尔矩阵评估。这将在下面明确计算。对于任何 $1 \leq i, j \leq n$ ，我们有

$$\begin{aligned} g_{ij} &= \mathbb{E}[\partial_{w_i} \ell \partial_{w_j} \ell] \\ &= P(Y = 1|\mathbf{x}) \partial_{w_i} \ln P(Y = 1|\mathbf{x}) \partial_{w_j} \ln P(Y = 1|\mathbf{x}) \\ &\quad + P(Y = 0|\mathbf{x}) \partial_{w_i} \ln P(Y = 0|\mathbf{x}) \partial_{w_j} \ln P(Y = 0|\mathbf{x}) \\ &= \sigma(w^T \mathbf{x} + b) \partial_{w_i} \ln \sigma(w^T \mathbf{x} + b) \partial_{w_j} \ln \sigma(w^T \mathbf{x} + b) \\ &\quad + (1 - \sigma(w^T \mathbf{x} + b)) \partial_{w_i} \ln(1 - \sigma(w^T \mathbf{x} + b)) \partial_{w_j} \ln(1 - \sigma(w^T \mathbf{x} + b)). \end{aligned}$$

使用微分规则 $(\ln f(x))' = f'(x)/f(x)$ 和 Sigmoid 属性 $\sigma' = \sigma(1 - \sigma)$ ，我们得到

$$\begin{aligned} g_{ij} &= \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b))^2 x_i x_j + \sigma(w^T \mathbf{x} + b)^2 (1 - \sigma(w^T \mathbf{x} + b)) x_i x_j \\ &= \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) x_i x_j. \end{aligned}$$

如果将 $\tilde{\mathbf{x}} = (1, \mathbf{x})^T$ 和 $\tilde{w} = (b, w)$ 写成如下形式，该公式可以扩展。

$$\tilde{g}_{ij} = \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) \tilde{x}_i \tilde{x}_j,$$

或者，以矩阵形式表示为 $\tilde{g} = \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) \tilde{\mathbf{x}} \tilde{\mathbf{x}}^T$ 。由于矩阵 $\mathbf{x} \mathbf{x}^T$ 的任意两列成比例，所以我们有行列式 $\tilde{g} = 0$ 。同样，迹为 $\tilde{g} = (1 + \|x\|_2^2)\sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) > 0$ 。

方向 $\tilde{w} = (b, w)$ 上的信息密度由下式给出

$$\begin{aligned} \tilde{w}^T \tilde{g} \tilde{w} &= \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) \tilde{w}^T \tilde{\mathbf{x}} \tilde{\mathbf{x}}^T \tilde{w} \\ &= \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)) (\tilde{\mathbf{x}}^T \tilde{w})^2 \\ &= (w^T \mathbf{x} + b)^2 \sigma(w^T \mathbf{x} + b)(1 - \sigma(w^T \mathbf{x} + b)). \end{aligned}$$

这意味着沿超平面 $\{\mathbf{x} \in \mathbb{R}^n; w^T \mathbf{x} + b = 0\}$ 的密度为零。此外，当 sigmoid 函数饱和时，即对于 $w^T \mathbf{x} + b \rightarrow \pm\infty$ ，信息密度趋于零。

最大似然一个随机神经元通过最大化似然来学习。为此，我们考虑随机变量 $U = 2Y - 1$ 。然后对于 $Y = 0$ ， $U = -1$ ，对于 $Y = 1$ ， $U = 1$ 。此外，

$$P(U = 1) = P(Y = 1) = \sigma(w^T \mathbf{x} + b) = \sigma((w^T \mathbf{x} + b)U)$$

$$P(U = -1) = P(Y = 0) = 1 - \sigma(w^T \mathbf{x} + b) = \sigma(-w^T \mathbf{x} - b) = \sigma((w^T \mathbf{x} + b)U).$$

使用新变量 U 的优点是，在两种情况下概率具有相同的表达式。我们将训练集 $\{(\mathbf{x}_1, z_1), \dots, (\mathbf{x}_n, z_n)\}$ 转换为 $\{(\mathbf{x}_1, t_1), \dots, (\mathbf{x}_n, t_n)\}$ ，其中 $t_j = 2z_j - 1$ 。最优参数由下式给出

$$\begin{aligned}(w^*, b^*) &= \arg \max_{w,b} \prod_{j=1}^n \sigma((w^T \mathbf{x}_j + b)t_j) = \arg \max_{w,b} \ln \prod_{j=1}^n \sigma((w^T \mathbf{x}_j + b)t_j) \\ &= \arg \max_{w,b} \frac{1}{n} \sum_{j=1}^n \ln \sigma((w^T \mathbf{x}_j + b)t_j).\end{aligned}$$

成本函数

$$C(w, b) = \frac{1}{n} \sum_{j=1}^n \ln \sigma((w^T \mathbf{x}_j + b)t_j)$$

可以通过梯度上升法最大化。这意味着在每次参数更新时，在梯度方向上采取大小为 η 的步长

$$\begin{aligned}w^{(m+1)} &= w^{(m)} + \eta \nabla_w C(w, b) \\ b^{(m+1)} &= b^{(m)} + \eta \nabla_b C(w, b),\end{aligned}$$

使用链式法则和sigmoid的性质计算梯度：

$$\begin{aligned}\frac{\partial C}{\partial w_k} &= \frac{1}{n} \sum_{j=1}^n \frac{x_j^k t_j \sigma'((w^T \mathbf{x}_j + b)t_j)}{\sigma((w^T \mathbf{x}_j + b)t_j)} = \frac{1}{n} \sum_{j=1}^n x_j^k t_j \sigma(-(w^T \mathbf{x}_j + b)t_j) \\ \frac{\partial C}{\partial b} &= \frac{1}{n} \sum_{j=1}^n t_j \sigma(-(w^T \mathbf{x}_j + b)t_j).\end{aligned}$$

模拟退火方法 一个二元随机神经元在某种意义上收敛于一个确定性感知器。首先，我们对随机神经元的结果分布进行如下修改

$$\begin{aligned}P(Y = 1|\mathbf{x}) &= \sigma_c(w^T \mathbf{x} + b), \\ P(Y = 0|\mathbf{x}) &= 1 - \sigma_c(w^T \mathbf{x} + b) = \sigma_c(-w^T \mathbf{x} - b),\end{aligned}$$

其中 $\sigma_c(x) = \sigma(cx) = \frac{1}{1+e^{-cx}}$ ，且 $c \geq 0$ 。相关的成本函数为

$$C(w, b) = \frac{1}{n} \sum_{j=1}^n \ln \sigma_c((w^T \mathbf{x}_j + b)t_j).$$

对于 $c \rightarrow 0$ ，我们得到等概率状态，即

$$P(Y = 1|\mathbf{x}) = \frac{1}{2}, \quad P(Y = 0|\mathbf{x}) = \frac{1}{2},$$

无论输入值如何， \mathbf{x} 。这对应于最大噪声情况。

当 c 无界增加时，函数 $\sigma_c(x)$ 几乎在所有地方（除了原点）趋于海维赛德阶跃函数 $H(x)$ 。对于 $c \rightarrow \infty$ ，我们得到分布

$$\begin{aligned} P(Y = 1|\mathbf{x}) &= H(w^T \mathbf{x} + b) \\ P(Y = 0|\mathbf{x}) &= 1 - H(w^T \mathbf{x} + b). \end{aligned}$$

这等同于如果 $w^T \mathbf{x} + b > 0$ 和 $Y = 1$ ，则 $Y = 1$ ，否则的情况，这对应于一个确定性感知机。

我们注意到，从退火的角度来看，常数 c 被视为温度的倒数。增加 c 的值意味着降低温度，以便获得感知机成本函数的全局最小值。

20.2 玻尔兹曼分布

在给定温度下，粒子系统的热力学平衡由一种概率分布描述，称为玻尔兹曼分布。鉴于其在随机神经网络研究中的重要性，我们将在下文详细介绍这种分布。

考虑一个具有 N 个状态的热力学系统以及一个描述系统状态的随机变量 \mathbf{x} 。状态变量 \mathbf{x} 的可能值为 x_1, \dots, x_N ，其出现概率为 $p_j = P(\mathbf{x} = x_j)$ 。此外，我们假设每个状态 x_j 都与系统的一个能级 E_j 相关联， E_j 是一个正实数。等效地说，系统以概率 p_j 处于能量为 E_j 的状态。

系统的状态因粒子相互作用而改变。根据热力学第二定律¹，状态的变化应使系统的总熵增加。在没有任何其他约束的情况下，从长远来看，系统趋向于均匀分布，而均匀分布是已知熵最大的分布。对于均匀分布 $p_1 = \dots = p_N = \frac{1}{N}$ ，这一点得以实现。

¹ 这表明孤立热力学系统的熵会增加。

然而，假设粒子相互作用是在给定的规定温度 T 下发生的。温度与系统的平均能量成正比，即

$$T \sim \sum_{j=1}^N p_j E_j.$$

因此，寻找受约束 $\sum_{j=1}^N p_j E_j = k$ 且 k 为正的最大熵分布就足够了。这对应于系统在给定温度 k 下具有最大不确定性的状态。

为了找到在约束 $H(p) = -\sum_{j=1}^N p_j \ln p_j$ 下具有最大熵的分布

$$\sum_{j=1}^N p_j = 1 \quad (20.2.1)$$

$$\sum_{j=1}^N p_j E_j = k, \quad (20.2.2)$$

我们构造如下函数：

$$F(p_1, \dots, p_N, \lambda_1, \lambda_2) = -\sum_{j=1}^N p_j \ln p_j + \lambda_1 \left(\sum_{j=1}^N p_j E_j - k \right) + \lambda_2 \left(\sum_{j=1}^N p_j - 1 \right),$$

其中 $\lambda_1, \lambda_2 \in \mathbb{R}$ 为拉格朗日乘数。由于概率向量属于一个紧致集， $(p_1, \dots, p_n) \in [0, 1] \times \dots \times [0, 1]$ ，且 F 是一个连续函数，一个著名的结果表明 F 在上述超立方体内存在最大值。² 由于每个状态都以正概率出现，在定义域内部寻找最大值是有意义的。因此，我们可以使用变分方程

$$\frac{\partial F}{\partial p_j} = -(1 + \ln p_j) + \lambda_1 E_j + \lambda_2 = 0, \quad 1 \leq j \leq N,$$

这意味着 $p_j = ce^{\lambda_1 E_j}$ ，其中 $c = e^{\lambda_2 - 1}$ 。这是一个真正的最大值，因为海森矩阵

$$\frac{\partial^2 F}{\partial p_i \partial p_j} = \begin{pmatrix} -\frac{1}{p_1} & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & -\frac{1}{p_N} \end{pmatrix}$$

² 该定理指出，在紧致集上定义的连续函数是有界的，并且能达到其最小值和最大值。

是负定的。为了证明这种分布的唯一性，只需陈述拉格朗日乘数 λ_1, λ_2 的唯一性。我们将证明只有一对数 (λ_1, c) 满足约束系统 (20.2.1)-(20.2.2)。这可以等效地写为以下非线性系统：

$$\begin{aligned} G_1(c, \lambda_1) &= 1 \\ G_2(c, \lambda_1) &= k \end{aligned}$$

其中 $G_1(c, \lambda_1) = \sum_{j=1}^N ce^{\lambda_1 E_j}$ 和 $G_2(c, \lambda_1) = \sum_{j=1}^N ce^{\lambda_1 E_j} E_j$ 。根据反函数定理（见附录中的定理F.1），只要其雅可比行列式不为零，该系统就有唯一解。雅可比行列式可计算为

$$\begin{aligned} J_G &= \begin{vmatrix} \frac{\partial G_1}{\partial c} & \frac{\partial G_1}{\partial \lambda_1} \\ \frac{\partial G_2}{\partial c} & \frac{\partial G_2}{\partial \lambda_1} \end{vmatrix} = \begin{vmatrix} \sum e^{\lambda_1 E_j} & \sum ce^{\lambda_1 E_j} E_j \\ \sum e^{\lambda_1 E_j} E_j & \sum ce^{\lambda_1 E_j} E_j^2 \end{vmatrix} = \begin{vmatrix} \frac{1}{c} & k \\ \frac{k}{c} & \sum ce^{\lambda_1 E_j} E_j^2 \end{vmatrix} \\ &= \frac{1}{c} \left[\sum_{j=1}^N p_j E_j^2 - \left(\sum_{j=1}^N p_j E_j \right)^2 \right] > 0, \end{aligned}$$

这是柯西不等式的一个结果

$$\left(\sum_{j=1}^N p_j E_j \right)^2 = \left(\sum_{j=1}^N \sqrt{p_j} (\sqrt{p_j} E_j) \right)^2 < \sum_{j=1}^N p_j \sum_{j=1}^N p_j E_j^2 = \sum_{j=1}^N p_j E_j^2.$$

配分函数 如果我们定义配分函数 $Z = \sum_{j=1}^N e^{\lambda_1 E_j}$ ，解也可以写成

$$p_j = \frac{e^{\lambda_1 E_j}}{Z}, \quad 1 \leq j \leq N. \quad (20.2.3)$$

配分函数 Z 可按如下方式计算。对函数 $Z(\lambda_1)$ 关于 λ_1 求导，我们得到

$$\frac{d}{d\lambda_1} Z(\lambda_1) = \sum_{j=1}^N e^{\lambda_1 E_j} E_j = \sum_{j=1}^N Z p_j E_k = kZ,$$

在此我们用到了 $\sum_{j=1}^N p_j E_k = k$ 。因此， $Z(\lambda_1)$ 是微分方程的解

$$\begin{aligned} \frac{d}{d\lambda_1} Z(\lambda_1) &= kZ \\ Z(0) &= N, \end{aligned}$$

因此， $Z = Ne^{k\lambda_1}$ 。求解 $\lambda_1 = -\frac{\ln(N/Z)}{k}$ 并代入(20.2.3)可得

$$p_j = \frac{e^{-\frac{\ln(N/Z)}{k} E_j}}{Z} = e^{-\frac{E_j}{T}},$$

其中温度 T 是与系统平均能量 k 成正比的物理量度。在文献中，这被称为玻尔兹曼分布³ 或吉布斯分布[72]，它由玻尔兹曼[112] 引入。这是一种概率分布，用于表征处于热平衡状态且具有 N 种不同状态和相关能量 E_1, E_2, \dots, E_N 的系统。系统处于较低能量状态的概率较高，且概率随能量水平的增加呈指数下降。

20.3 玻尔兹曼机

玻尔兹曼机是一种由相互连接的二元随机神经元组成的神经网络（见第20.1节），其权重是对称的， $w_{ij} = w_{ji}$ ，以及 $w_{ii} = 0$ ，这些神经元对是否开启或关闭做出随机决策。具有 n 个神经元的玻尔兹曼机在状态空间 $\mathcal{X} = \{0, 1\}^n$ 上定义了一个概率分布，该分布由一个能量函数参数化，该能量函数描述了模型内部的相互作用。更准确地说，我们有：

定义20.3.1 玻尔兹曼机是一组具有状态 x_1, \dots, x_n 的 n 个噪声神经元，它们形成一个具有对称权重的网络。第 i 个神经元的状态根据以下规则随机更新

$$x_i = \begin{cases} 1, & \text{with probability } p_i; \\ 0, & \text{with probability } 1 - p_i, \end{cases}$$

其中

$$p_i = \sigma_{1/T}(\sum_j w_{ji}x_j + b_i) = \frac{1}{1 + e^{-(\sum_j w_{ji}x_j + b_i)/T}}. \quad (20.3.4)$$

常数 $T > 0$ 表示温度， $w_{ij} = w_{ji}$ 是第 i 个和第 j 个神经元之间的权重，具有自循环连接 $w_{ii} = 0$ 。常数 b_j 表示第 j 个神经元的偏差。

³ 物理学中使用的版本包括一个玻尔兹曼常数 $\kappa > 0$ ，我们认为它等于 1。该分布实际上写为 $p(\mathbf{x}) = \frac{1}{Z} e^{-E(\mathbf{x})/(\kappa T)}$ 。

此外，第 i 个神经元保持不活跃的概率为

$$1 - p_i = \sigma_{1/T}(-\sum_j w_{ji}x_j + b_i) = \frac{1}{1 + e^{(\sum_j w_{ji}x_j + b_i)/T}},$$

其中我们使用了 sigmoid 函数的互补性质， $\sigma(x) + \sigma(-x) = 1$ 。

玻尔兹曼机的状态由向量 $\mathbf{x} = (x_1, \dots, x_n)^T$ 描述。由于每个神经元以二进制方式激活， $x_i \in \{0, 1\}$ ，因此玻尔兹曼机可以具有 $N = 2^n$ 种状态。

作为热力学系统的玻尔兹曼机 我们首先引入一个依赖于网络状态的能量函数。 i 号神经元的信号电位由所有其他神经元对其自身的
作用定义，包括其自身的偏置，如 $u_i = -\left(\frac{1}{2} \sum_j w_{ji}x_j + b_i\right)$ 所示。包含负号是为了方便起见（以便最终得到类似于玻尔兹曼分布的分布），并且包含因子 $\frac{1}{2}$ 是因为权重 w_{ij} 的一半计入 i 号神经元，另一半计入 j 号神经元。所有其他神经元的激活（0 或 1）通过权重 w_{ij} 给出的突触强度对电位进行累加贡献。 i 号神经元的能量是其激活值 x_i 与信号电位 $\epsilon_i = u_i x_i$ 的乘积。网络状态
 $\mathbf{x} = (x_1, \dots, x_n)^T$ 的总能量是该状态下所有单个神经元能量的总和

$$\begin{aligned} E(\mathbf{x}) &= \sum_i \epsilon_i = \sum_i u_i x_i \\ &= -\frac{1}{2} \sum_{i,j} w_{ji} x_j x_i - \sum_i b_i x_i = -\sum_{i < j} w_{ji} x_j x_i - \sum_k b_i x_k. \end{aligned}$$

以矩阵形式我们可以写成 $E(\mathbf{x}) = -\frac{1}{2} \mathbf{x}^T w \mathbf{x} - \mathbf{x}^T b$ 。因此，相关的能量函数相对于神经元激活 x_i 是二次的。

现在考虑一个具有 $N = 2^n$ 个状态 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 的热力学系统。每个状态对应一个能量水平 $E_j = E(\mathbf{x}_j)$ ，其中 E 是先前定义的二次能量函数。

假设玻尔兹曼机在时间为零时处于某种状态。通过随机更新规则，系统的熵增加，直到达到热力学平衡。相应的平衡分布是玻尔兹曼分布 $p_j = \frac{e^{-E_j/T}}{Z}$, $1 \leq j \leq n$ ，并且无论机器的初始状态如何都能达到这一点。

有许多方法可以赋予网络一个能量函数。

然而,之前引入的二次函数似乎是自然的选择,因为它与来自热力学的跃迁概率兼容,热力学表明从能量为 E_i 的状态 \mathbf{x}_i 跃迁到能量为 E_j 的状态 \mathbf{x}_j 的概率由 $p_{ij} = \frac{1}{1+e^{(E_j-E_i)/T}}$ 给出。

假设网络从能量为 $E = E(\mathbf{x})$ 的状态 $\mathbf{x} = (x_1, \dots, x_k, \dots, x_n)$ 更新到能量为 $E' = E(\mathbf{x}')$ 的状态 $\mathbf{x}' = (x_1, \dots, x'_k, \dots, x_n)$ 。根据练习20.10.1,能级之间的差异为

$$E' - E = -\left(\sum_{i=1}^n w_{ki}x_i + b_k\right)(x'_k - x_k).$$

假设状态 $x_k = 0$ 更新为 $x'_k = 1$ 。那么差值变为

$$E' - E = -\left(\sum_{i=1}^n w_{ki}x_i + b_k\right), \text{ 更新公式(20.3.4)可以写成}$$

$$p = \frac{1}{1+e^{-(\sum_{i=1}^n w_{ki}x_i+b_k)/T}} = \frac{1}{1+e^{(E'-E)/T}},$$

这与上述转移概率公式一致。

如果状态 $x_k = 1$ 更新为 $x'_k = 0$, 那么差值变为 $E' - E = \sum_{i=1}^n w_{ki}x_i + b_k$, 我们再次得到转移概率

$$1-p = \frac{1}{1+e^{(\sum_{i=1}^n w_{ki}x_i+b_k)/T}} = \frac{1}{1+e^{(E'-E)/T}}.$$

为便于理解,我们考虑一个例子。

例20.3.2考虑一个具有三个神经元的网络,其连接是对称的且 $w_{ii} = 0$ 。它们的信号电位由下式给出

$$\begin{aligned} u_1 &= -(w_{12}x_2 + w_{13}x_3 + b_1) \\ u_2 &= -(w_{12}x_1 + w_{23}x_3 + b_2) \\ u_3 &= -(w_{13}x_1 + w_{23}x_2 + b_3). \end{aligned}$$

那么网络的能量由下式给出

$$E(\mathbf{x}) = -(w_{12}x_1x_2 + w_{13}x_1x_3 + w_{23}x_2x_3 + b_1x_1 + b_2x_2 + b_3x_3).$$

系统存在 $N = 2^3 = 8$ 种状态

$$\mathcal{X} = \{(0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1), (0, 1, 1), (1, 0, 1), (1, 1, 0), (1, 1, 1)\}$$

对应于以下能级:

$$\begin{array}{ll} E_1 = E(0, 0, 0) = 0 & E_5 = E(0, 1, 1) = -w_{23} - b_2 - b_3 \\ E_2 = E(1, 0, 0) = -b_1 & E_6 = E(1, 0, 1) = -w_{13} - b_1 - b_3 \\ E_3 = E(0, 1, 0) = -b_2 & E_7 = E(1, 1, 0) = -w_{12} - b_1 - b_2 \\ E_4 = E(0, 0, 1) = -b_3 & E_8 = E(1, 1, 1) = -w_{12} - w_{13} - w_{23} - b_1 - b_2 - b_3. \end{array}$$

玻尔兹曼分布， $p_j = \frac{e^{-E_j}}{Z}$ ，对于 $T = 1$ 的情况，由下式给出

$$p = \frac{1}{Z}(1, e^{b_1}, e^{b_2}, e^{b_3}, e^{w_{23}+b_2+b_3}, e^{w_{13}+b_1+b_3}, e^{w_{12}+b_1+b_2}, e^{w_{12}+w_{13}+w_{23}+b_1+b_2+b_3}),$$

其中 Z 是一个归一化因子。我们注意到所有能级（除了第一个）都依赖于网络权重和偏差。因此，网络参数的任何变化都会导致能级的变化，进而导致玻尔兹曼分布的变化。

人们可能有兴趣通过调整网络参数，使用玻尔兹曼分布来学习状态空间 \mathcal{X} 上的任意分布 q 。在当前情况下，这无法精确完成，因为我们需要使用仅6个参数 $b_1, b_2, b_3, w_{12}, w_{13}$ 和 w_{23} 来学习7个未知数 q_1, \dots, q_7 （因为 q_8 取决于其他未知数）。玻尔兹曼学习算法将通过最小化库尔贝克 - 莱布勒散度，提供一种以近似方式学习 q 的方法。

然而，存在学习是精确的情况。我们将在下一个示例中处理此问题。

示例20.3.3（精确学习） 考虑一个具有 $n = 2$ 个神经元的玻尔兹曼机。 $N = 2^2$ 状态由下式给出

$$\mathcal{X} = \{(0, 0), (1, 0), (0, 1), (1, 1)\}.$$

相关能量为

$$E(\mathbf{x}) = -w_{12}x_1x_2 - b_1x_1 - b_2x_2,$$

并且每个状态的能级由下式给出

$$E(0, 0) = 0, \quad E(1, 0) = -b_1, \quad E(0, 1) = -b_2, \quad E(1, 1) = -w_{12} - b_1 - b_2.$$

在 $T = 1$ 处，玻尔兹曼分布实现了热平衡

$$p = \frac{1}{Z}(1, e^{b_1}, e^{b_2}, e^{w_{12}+b_1+b_2}).$$

我们现在考虑 \mathcal{X} 上的任意分布 q

$$q_1 = q(0, 0), \quad q_1 = q(1, 0), \quad q_2 = q(0, 1), \quad q_3 = q(1, 1)$$

并且我们将找到参数 w, b 的精确值，使得 $p = q$ ，即玻尔兹曼机能够精确学习 q 。我们将参数识别为

$$\begin{aligned} p_1 &= \frac{1}{Z} = q_1 \Rightarrow Z = 1/q_1 \\ p_2 &= \frac{e^{b_1}}{Z} = q_2 \Rightarrow b_1 = \ln \frac{q_2}{q_1} \\ p_3 &= \frac{e^{b_2}}{Z} = q_3 \Rightarrow b_2 = \ln \frac{q_3}{q_1} \\ p_4 &= \frac{e^{w_{12} + b_1 + b_2}}{Z} = q_4 \Rightarrow w_{12} = \ln \frac{q_1 q_4}{q_2 q_3}. \end{aligned}$$

给定分布 q ，我们可以写出玻尔兹曼分布的精确表达式，它能精确学习 q ，如下所示

$$\begin{aligned} p(\mathbf{x}) &= \frac{e^{-E(\mathbf{x})}}{Z} = q_1 e^{w_{12}x_1x_2 + b_1x_1 + b_2x_2} = q_1 (e^{w_{12}})^{x_1x_2} (e^{b_1})^{x_1} (e^{b_2})^{x_2} \\ &= q_1 \left(\frac{q_1 q_4}{q_2 q_3} \right)^{x_1 x_2} \left(\frac{q_2}{q_1} \right)^{x_1} \left(\frac{q_3}{q_1} \right)^{x_2} \\ &= q_1^{(1-x_1)(1-x_2)} q_2^{x_1(1-x_2)} q_3^{x_2(1-x_1)} q_4^{x_1 x_2}. \end{aligned}$$

这种对分布 q 的精确学习仅适用于 $n = 2$ 神经元，因为在这种情况下方程的数量等于变量的数量。

在一般情况下，分布 p 中未知变量的数量为 $n(n+1)/2$ （我们使用了对称条件 $w_{ij} = w_{ji}, w_{ii} = 0$ ，并且还包括 b_k ），而方程的数量为 $2^n - 1$ （我们减去1以考虑线性关系 $\sum q_i = 1$ ）。对于 $n > 2$ ，我们总是有 $n(n+1)/2 < 2^n - 1$ ，仅在 $n = 2$ 的情况下等式成立。

20.4 玻尔兹曼学习

玻尔兹曼机可以作为定义在状态空间 \mathcal{X} 上的分布的近似器，见[2]。为了使玻尔兹曼机学习定义在状态空间 \mathcal{X} 上的给定分布 q ，我们需要在机器生成的所有分布 p 中选择分布 p^* ，它在库尔贝克-莱布勒散度的意义上最接近 q ，即

$$p^* = \arg \min_p D_{KL}(q||p) = \arg \min_p \sum_{\mathbf{x} \in X} q(\mathbf{x}) \ln \frac{q(\mathbf{x})}{p(\mathbf{x})}.$$

由于熵 $H(q) = \sum_{\mathbf{x} \in \mathcal{X}} q(\mathbf{x}) \ln q(\mathbf{x})$ 与 p 无关，之前的搜索等同于

$$p^* = \arg \max_p \sum_{\mathbf{x} \in X} q(\mathbf{x}) \ln p(\mathbf{x}).$$

其思想是，玻尔兹曼分布 $p(\mathbf{x}) = \frac{e^{-E(\mathbf{x})}}{Z(\mathbf{x})}$ 会在机器参数变化时发生改变。参数按照梯度上升法进行更新。利用链式法则，我们首先求出对数似然函数的导数

$$\begin{aligned}
\frac{\partial}{\partial w_{ij}} \ln p(\mathbf{x}) &= -\frac{\partial}{\partial w_{ij}} E(\mathbf{x}) - \frac{\partial}{\partial w_{ij}} \ln Z(\mathbf{x}) \\
&= \frac{\partial}{\partial w_{ij}} \left(\frac{1}{2} \mathbf{x}^T \mathbf{w} \mathbf{x} + \mathbf{x}^T \mathbf{b} \right) - \frac{1}{Z(\mathbf{x})} \frac{\partial}{\partial w_{ij}} Z(\mathbf{x}) \\
&= x_i x_j - \frac{1}{Z(\mathbf{x})} \sum_{\mathbf{x} \in \mathcal{X}} \frac{\partial}{\partial w_{ij}} e^{-E(\mathbf{x})} \\
&= x_i x_j - \sum_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}) x_i x_j \\
&= x_i x_j - \mathbb{E}^p[x_i x_j].
\end{aligned}$$

类似地，我们有

$$\begin{aligned}
\frac{\partial}{\partial b_j} \ln p(\mathbf{x}) &= -\frac{\partial}{\partial b_j} E(\mathbf{x}) - \frac{\partial}{\partial b_j} \ln Z(\mathbf{x}) \\
&= x_j - \sum_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}) x_j \\
&= x_j - \mathbb{E}^p[x_j].
\end{aligned}$$

代价函数 $C(w, b) = \sum_{\mathbf{x} \in X} q(\mathbf{x}) \ln p(\mathbf{x})$ 的梯度分量

现在计算如下

$$\begin{aligned}
\frac{\partial}{\partial w_{ij}} C(w, b) &= \sum_{\mathbf{x} \in X} q(\mathbf{x}) \frac{\partial}{\partial w_{ij}} \ln p(\mathbf{x}) \\
&= \sum_{\mathbf{x} \in X} q(\mathbf{x}) x_i x_j - \sum_{\mathbf{x} \in X} q(\mathbf{x}) \mathbb{E}^p[x_i x_j] \\
&= \mathbb{E}^q[x_i x_j] - \mathbb{E}^p[x_i x_j].
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial b_j} C(w, b) &= \sum_{\mathbf{x} \in X} q(\mathbf{x}) \frac{\partial}{\partial b_j} \ln p(\mathbf{x}) \\
&= \sum_{\mathbf{x} \in X} q(\mathbf{x}) x_j - \sum_{\mathbf{x} \in X} q(\mathbf{x}) \mathbb{E}^p[x_j] \\
&= \mathbb{E}^q[x_j] - \mathbb{E}^p[x_j].
\end{aligned}$$

学习遵循调整规则

$$\begin{aligned}\Delta w_{ij} &= \eta \frac{\partial}{\partial w_{ij}} C(w, b) \\ \Delta b_j &= \eta \frac{\partial}{\partial b_j} C(w, b),\end{aligned}$$

该规则恢复了在[2]中获得的学习规则

$$\begin{aligned}\Delta w_{ij} &= \eta (\mathbb{E}^q[x_i x_j] - \mathbb{E}^p[x_i x_j]) \\ \Delta b_j &= \eta (\mathbb{E}^q[x_j] - \mathbb{E}^p[x_j]),\end{aligned}$$

其中 $\eta > 0$ 是学习率。值得注意的是，学习规则有两个阶段：在第一阶段，连接权重 w_{ij} 在给定分布 q 下通过 x_i 和 x_j 的平均激活增加；在第二阶段，连接权重 w_{ij} 在玻尔兹曼分布 p 下通过 x_i 和 x_j 的平均激活减少。

注记20.4.1权重和偏差的变化会导致玻尔兹曼分布的扰动。其对参数的敏感性由下式给出

$$\begin{aligned}dp(\mathbf{x}) &= p(\mathbf{x}) d \ln p(\mathbf{x}) \\ &= p(\mathbf{x}) \sum_{i,j} \frac{\partial \ln p(\mathbf{x})}{\partial w_{ij}} dw_{ij} + p(\mathbf{x}) \sum_j \frac{\partial \ln p(\mathbf{x})}{\partial b_j} db_j \\ &= p(\mathbf{x}) \sum_{i,j} (x_i x_j - \mathbb{E}^p[x_i x_j]) dw_{ij} + p(\mathbf{x}) \sum_j (x_j - \mathbb{E}^p[x_j]) db_j.\end{aligned}$$

20.5 计算玻尔兹曼分布

玻尔兹曼分布是一种平衡分布，即无论初始状态如何选择，机器最终都会稳定到相同的分布。在本节中，我们将使用一种极限过程直接计算具有两个神经元的玻尔兹曼机的玻尔兹曼分布。从任意初始状态 $\mathbf{x}^0 = (x_1^0, x_2^0)$ 开始，根据玻尔兹曼机的更新规则（20.3.4）调整状态，我们得到一系列状态 $\mathbf{x}^n = (x_1^n, x_2^n)$ ，其分布将收敛到玻尔兹曼分布。因此，只需计算状态序列 \mathbf{x}^n 的分布，然后取 $n \rightarrow \infty$ 即可得到平衡分布。此过程不使用第20.2节中介绍的结果。

在这个模型中，我们有3个参数， b_1, b_2 ，以及 $w = w_{12} = w_{21}$ 。设 $a_n = P(x_2^n = 1)$ 。在第一阶段，我们将找到序列 a_n 的递推关系，并求出其极限。

在第 $(n+1)$ 步时第二个神经元的状态 x_2^{n+1} ，取决于在第 $(n+1)$ 步时第一个神经元的状态 x_1^{n+1} ，如下所示：

$$x_2^{n+1} = \begin{cases} 1, & \text{with probability } \sigma(wx_1^{n+1} + b_2) \\ 0, & \text{with probability } 1 - \sigma(wx_1^{n+1} + b_2). \end{cases}$$

使用概率链规则⁴，我们可以用条件概率根据第一个神经元的状态来表示第二个神经元状态的概率，如下所示

$$\begin{aligned} P(x_2^{n+1} = 1) &= P(x_2^{n+1} = 1|x_1^{n+1} = 1)P(x_1^{n+1} = 1) \\ &\quad + P(x_2^{n+1} = 1|x_1^{n+1} = 0)P(x_1^{n+1} = 0) \\ &= \sigma(w + b_2)P(x_1^{n+1} = 1) + \sigma(b_2)P(x_1^{n+1} = 0) \end{aligned} \quad (20.5.5)$$

现在，状态 x_1^{n+1} 取决于 x_2^n ，如下所示

$$x_1^{n+1} = \begin{cases} 1, & \text{with probability } \sigma(wx_2^n + b_1) \\ 0, & \text{with probability } 1 - \sigma(wx_2^n + b_1). \end{cases}$$

概率链规则得出

$$\begin{aligned} P(x_1^{n+1} = 1) &= P(x_1^{n+1} = 1|x_2^n = 1)P(x_2^n = 1) \\ &\quad + P(x_1^{n+1} = 1|x_2^n = 0)P(x_2^n = 0) \\ &= \sigma(w + b_1)P(x_2^n = 1) + \sigma(b_1)P(x_2^n = 0). \end{aligned} \quad (20.5.6)$$

将(20.5.6)代入(20.5.5)得到一阶递推式

$$a_{n+1} = \alpha a_n + \beta, \quad (20.5.7)$$

其中

$$\begin{aligned} \alpha &= (\sigma(w + b_1) - \sigma(b_1))(\sigma(w + b_2) - \sigma(b_2)) \\ \beta &= \sigma(w + b_2)\sigma(b_1) + \sigma(b_2)\sigma(-b_1). \end{aligned}$$

由于 σ 是递增的，所以对于 $w \neq 0$ 有 $\alpha > 0$ 。利用中值定理，我们可以估计以下上界：

$$\alpha = \sigma'(c_1)w\sigma'(c_2)w \leq \|\sigma'\|_\infty w^2 = \frac{w^2}{16},$$

其中我们用到了 σ 的最大斜率是 $1/4$ 。由于 $\sigma \in (0, 1)$ ，我们有 $0 < \beta < 2$ 。

⁴ 对于任意两个事件 A 和 B ，我们有

$$P(A) = P(A, B) + P(A, B^c) = P(A | B)P(B) + P(A | B^c)P(B^c).$$

递推式(20.5.7)可以用归纳法求解如下

$$\begin{aligned} a_{n+1} &= \alpha^{n+1}a_0 + \beta(1 + \alpha + \cdots + \alpha^n) \\ &= \alpha^{n+1}a_0 + \beta \frac{1 - \alpha^{n+1}}{1 - \alpha}. \end{aligned}$$

假设 $|w| < 4$ (这意味着稳定性条件)。那么 $0 < \alpha < 1$ ，因此， $\alpha^n \rightarrow 0$ ，因为 $n \rightarrow \infty$ 。所以 $a_n \rightarrow \frac{\beta}{1-\alpha}$ ，或者

$$\lim_{n \rightarrow \infty} P(x_2^n = 1) = \frac{\beta}{1 - \alpha}.$$

下一个目标是找到以下平衡分布：

$$\begin{aligned} p(0, 0) &= \lim_{n \rightarrow \infty} P(x_1^n = 0, x_2^n = 0) \\ p(0, 1) &= \lim_{n \rightarrow \infty} P(x_1^n = 0, x_2^n = 1) \\ p(1, 0) &= \lim_{n \rightarrow \infty} P(x_1^n = 1, x_2^n = 0) \\ p(1, 1) &= \lim_{n \rightarrow \infty} P(x_1^n = 1, x_2^n = 1). \end{aligned}$$

在以下条件概率关系中取极限 $n \rightarrow \infty$

$$\begin{aligned} P(x_1^{n+1} = 0, x_2^n = 0) &= P(x_1^{n+1} = 0 | x_2^n = 0)P(x_2^n = 0) = \sigma(-b_1)P(x_2^n = 0) \\ P(x_1^{n+1} = 0, x_2^n = 1) &= P(x_1^{n+1} = 0 | x_2^n = 1)P(x_2^n = 1) = \sigma(-w - b_1)P(x_2^n = 1) \\ P(x_1^{n+1} = 1, x_2^n = 0) &= P(x_1^{n+1} = 1 | x_2^n = 0)P(x_2^n = 0) = \sigma(b_1)P(x_2^n = 0) \\ P(x_1^{n+1} = 1, x_2^n = 1) &= P(x_1^{n+1} = 1 | x_2^n = 1)P(x_2^n = 1) = \sigma(w + b_1)P(x_2^n = 1) \end{aligned}$$

得到平衡分布

$$\begin{aligned} p(0, 0) &= \sigma(-b_1) \left(1 - \frac{\beta}{1 - \alpha}\right) \\ p(0, 1) &= \sigma(-w - b_1) \frac{\beta}{1 - \alpha} \\ p(1, 0) &= \sigma(b_1) \left(1 - \frac{\beta}{1 - \alpha}\right) \\ p(1, 1) &= \sigma(w + b_1) \frac{\beta}{1 - \alpha}. \end{aligned}$$

接下来我们将介绍两种评估玻尔兹曼流形上信息的方法，一种是使用熵，另一种是评估费希尔信息。

20.6 玻尔兹曼分布的熵

玻尔兹曼分布是在给定平均能量 $\mathbb{E}^p [E(\mathbf{x})] = k$ 的固定值时，状态空间 \mathcal{X} 上具有最大熵的分布。熵可以按如下方式计算：

$$\begin{aligned} H(p) &= - \sum_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}) \ln p(\mathbf{x}) = - \sum_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}) \ln \frac{e^{-\frac{E(\mathbf{x})}{T}}}{Z} \\ &= \frac{1}{T} \sum_{x \in \mathcal{X}} p(\mathbf{x}) E(\mathbf{x}) + \sum_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}) \ln Z \\ &= \frac{1}{T} \mathbb{E}^p [E(\mathbf{x})] + \ln Z = \frac{k}{T} + \ln Z. \end{aligned}$$

由于在物理学中，熵的确定存在一个加法常数，常数项 $\ln Z$ 可以忽略不计。那么熵就变成了平均系统能量 $\mathbb{E}^p [E(\mathbf{x})]$ 与温度 T 的商。

设 p_{unif} 表示 \mathcal{X} 上的均匀分布，即 $p_{unif}(\mathbf{x}) = \frac{1}{N}$ 。它的熵

$$H(p_{unif}) = - \sum_{\mathbf{x}} \frac{1}{N} \ln \frac{1}{N} = \ln N,$$

是 \mathcal{X} 上所有分布熵中最大的。由于

$$\begin{aligned} D_{KL}(p || p_{unif}) &= \sum_{\mathbf{x}} p(\mathbf{x}) \ln \frac{p(\mathbf{x})}{p_{unif}(\mathbf{x})} = -H(p) - \sum_{\mathbf{x}} p(\mathbf{x}) \ln \frac{1}{N} \\ &= \ln N - H(p), \end{aligned}$$

由此可知，最大可能熵与玻尔兹曼分布熵之间的差异由下式给出

$$H(p_{unif}) - H(p) = D_{KL}(p || p_{unif}) > 0.$$

左边的项表示由于对平均能量 $\mathbb{E}^p [E(\mathbf{x})] = k$ 的约束导致的最大熵的减少。右边的项表明这是由库尔贝克 - 莱布勒散度给出的。

20.7 费希尔信息

任何玻尔兹曼机都定义了形式为 $p(\mathbf{x}) = \frac{e^{-E(\mathbf{x})/T}}{Z}$, $\mathbf{x} \in \mathcal{X}$ 的概率分布。反之，任何这种形式的分布都唯一地定义了一台玻尔兹曼机。

因此，玻尔兹曼机族可以与玻尔兹曼分布族等同起来，因此可以由 w_{ij} 和 b_k 进行参数化。具有坐标 (w_{ij}, b_k) 的相关流形上的黎曼度量由费希尔度量给出。使用第20.4节中的计算

$$\begin{aligned}\partial_{w_{ij}} \ln p(\mathbf{x}) &= x_i x_j - \mathbb{E}^p[x_i x_j] \\ \partial_{w_{kl}} \ln p(\mathbf{x}) &= x_k x_l - \mathbb{E}^p[x_k x_l] \\ \partial_{b_k} \ln p(\mathbf{x}) &= x_k - \mathbb{E}^p[x_k] \\ \partial_{b_l} \ln p(\mathbf{x}) &= x_l - \mathbb{E}^p[x_l],\end{aligned}$$

期望算子的线性性质可得

$$\begin{aligned}g_{ij,kl}(w, b) &= \mathbb{E}^p[\partial_{w_{ij}} \ln p(\mathbf{x}) \partial_{w_{kl}} \ln p(\mathbf{x})] \\ &= \mathbb{E}^p[x_i x_j x_k x_l] - \mathbb{E}^p[x_i x_j] \mathbb{E}^p[x_k x_l] \\ &= Cov(x_i x_j, x_k x_l). \\ g_{k,r}(w, b) &= \mathbb{E}^p[\partial_{b_k} \ln p(\mathbf{x}) \partial_{b_r} \ln p(\mathbf{x})] \\ &= \mathbb{E}^p[x_k x_r] - \mathbb{E}^p[x_k] \mathbb{E}^p[x_r] \\ &= Cov(x_k, x_r).\end{aligned}$$

我们注意到，费希尔信息取决于神经元激活 x_j 的相关性，并且与权重和偏差无关，即与神经流形坐标无关。⁵ 这意味着度量系数的导数为零

$$\frac{\partial g_{ij,kl}(w, b)}{\partial w_{ik}} = \frac{\partial g_{ij,kl}(w, b)}{\partial b_r} = \frac{\partial g_{k,r}(w, b)}{\partial w_{ij}} = \frac{\partial g_{k,r}(w, b)}{\partial b_j} = 0.$$

因此，所有克里斯托费尔符号 (13.1.2) 都消失了。那么相关的流形在本质上是平坦的（黎曼曲率张量为零）。在这个流形上的测地线方程 (13.1.1) 变为 $\ddot{c}^\alpha(t) = 0$ ，这意味着测地线分量 $c^\alpha(t)$ 在 t 中是仿射函数。由于流形上初始点和最优点之间的距离不能小于测地线的长度，前面的关系为在这个流形上学习可以多快完成提供了一个下限。这种情况类似于14.4节中描述的与线性神经元相关的神经流形的几何结构。

⁵ 在欧几里得空间中，欧几里得度量也会发生类似的情况， \mathbb{R}^n 。这些类型的度量被称为平移不变的。

出于效率考虑，费舍尔矩阵可与14.8节中介绍的自然梯度学习算法结合使用。在这种情况下，更新规则变为

$$\begin{aligned}\Delta w_{ij} &= \eta \sum_{k,l} g^{ij,kl} \frac{\partial}{\partial w_{kl}} C(w, b) = \eta \sum_{k,l} g^{ij,kl} (\mathbb{E}^q[x_k x_l] - \mathbb{E}^p[x_k x_l]) \\ &= \eta \left(\mathbb{E}^q \left[\sum_{k,l} g^{ij,kl} x_k x_l \right] - \mathbb{E}^p \left[\sum_{k,l} g^{ij,kl} x_k x_l \right] \right) \\ \Delta b_k &= \eta \sum_l g^{kl} \frac{\partial}{\partial b_l} C(w, b) = \eta \sum_l g^{kl} (\mathbb{E}^q[x_l] - \mathbb{E}^p[x_l]) \\ &= \eta \left(\mathbb{E}^q \left[\sum_l g^{kl} x_l \right] - \mathbb{E}^p \left[\sum_l g^{kl} x_l \right] \right).\end{aligned}$$

20.8玻尔兹曼机的应用

玻尔兹曼机主要用于解决组合优化问题和执行学习任务。

1. 分布近似器 正如我们在20.4节中所看到的，玻尔兹曼机能够使用指数形式的分布在状态空间 \mathcal{X} 上学习任何离散分布 $q(\mathbf{x})$ 。

$$p(\mathbf{x}) = \frac{e^{-(\mathbf{x}^T w \mathbf{x} + b^T \mathbf{x})/T}}{Z}. \quad (20.8.8)$$

为了完善这个结果，我们通过类型为(20.8.8)的分布的凸组合来近似分布 $q(\mathbf{x})$ 。

$$q(\mathbf{x}) \approx \sum_{i=1}^m \alpha_i q_i(\mathbf{x}),$$

使用 $q_i(\mathbf{x}) = \frac{e^{-(\mathbf{x}^T w(i)x + b(i)\mathbf{x})/T}}{Z(i)}$ 。由坐标 $(w(i), b(i))$ 定义的玻尔兹曼机 $B(i)$ 学习分布 $q_i(\mathbf{x})$ 。现在考虑一个结合了玻尔兹曼机 $B(i)$ 的神经网络。它的输出， $y(\mathbf{x}) = \sum_{i=1}^m \alpha_i q_i(\mathbf{x})$ 是分布 $q(\mathbf{x})$ 的近似值。备注20.8.1 (i) 已经证明，具有隐藏单元的玻尔兹曼机是离散变量上概率质量函数的通用逼近器，见[103]。

(ii) 对于连续情况，有类似的逼近结果。指数组合类在 $(0, \infty)$ 上的分布集里是众所周知的稠密的。感兴趣的读者可参考[35]。

2. 模拟退火方法 玻尔兹曼机是由相互连接的二元随机神经元组成的网络。当温度参数 $T \searrow 0$ 时，每个随机神经元变成一个常规感知器，见20.1节。因此，玻尔兹曼机趋向于由感知器组成的神经网络，这种网络被称为霍普菲尔德网络。我们将在下面简要介绍这种类型的网络。

霍普菲尔德网络 这种类型的网络是由物理学家约翰·霍普菲尔德⁶ 在1982年引入的，见[57]。其物理重要性在于霍普菲尔德网络与零温度下的磁性伊辛模型同构，见[61]。

一个霍普菲尔德网络由 n 个感知器⁷ 组成，这些感知器在被随机选择进行新的更新之前会保持其各自的状态。感知器是完全耦合的，也就是说，每个神经元都与其他所有神经元相连，但不包括自身。⁸ 第 i 个和第 j 个神经元之间的权重 w_{ij} 是对称的，并且可以用一个对角线元素为零的对称矩阵 w 来建模。一个具有 $n = 6$ 个神经元的霍普菲尔德网络的示例如图20.1 a所示。

网络从初始状态 $\mathbf{x}^0 = (x_1^0, \dots, x_n^0) \in \{0, 1\}^n$ 开始，更新一次发生一个（即它们是异步的）。假设选择第 j 个神经元进行更新。所有其他神经元对第 j 个神经元的影响，包括其偏差，是 $\sum_{i \neq j} w_{ij}x_i + b_j$ 。神经元的值更新为海维

赛德函数 $H\left(\sum_{i \neq j} w_{ij}x_i + b_j\right)$ 的值，其值为0或1。

在状态 \mathbf{x} 下评估的与霍普菲尔德网络相关的能量与玻尔兹曼机情况下的能量相同

$$\begin{aligned} E(\mathbf{x}) &= -\frac{1}{2}\mathbf{x}^T w \mathbf{x} - \mathbf{x}^T b \\ &= -\frac{1}{2} \sum_{i,j} w_{ji}x_jx_i - \sum_i b_i x_i = -\sum_{i < j} w_{ji}x_jx_i - \sum_k b_k x_k. \end{aligned}$$

霍普菲尔德网络的任务是通过更新过程将上述能量最小化。网络收敛到稳定状态，该状态对应于能量函数的局部最小值，如下所示。考虑从状态 $\mathbf{x} = (x_1, \dots, x_k, \dots, x_n)$ 到新状态 $\mathbf{x}' = (x_1, \dots, x'_k, \dots, x_n)$ 的更新。根据练习20.10.1，能量差为

⁶ 自1964年起担任普林斯顿大学教授。

⁷ 在我们的例子中，感知器取值为1和0，而在霍普菲尔德的方法中，它们取值为1和-1，因为他的模型源自一个物理模型，其中粒子的自旋要么是“向上”要么是“向下”。⁸ 这避免了其自身状态值被永久反馈。

$$E(\mathbf{x}') - E(\mathbf{x}) = - \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) (x'_k - x_k).$$

我们考虑两种情况：

(i) 如果第 k 个神经元将其值从 $x_k = 0$ 更新为 $x'_k = 1$ ，那么我们有 $H \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) = 1$ ，这意味着 $\sum_{i=1}^n w_{ki} x_i + b_k > 0$ 。在这种情况下

$$E(\mathbf{x}') - E(\mathbf{x}) = - \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) (1 - 0) < 0.$$

(ii) 如果第 k 个神经元将其值从 $x_k = 1$ 更新为 $x'_k = 0$ ，那么我们有 $H \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) = 0$ ，所以 $\sum_{i=1}^n w_{ki} x_i + b_k < 0$ 。然后

$$E(\mathbf{x}') - E(\mathbf{x}) = - \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) (0 - 1) < 0.$$

然后，每当一个神经元状态被更新时，总能量就会减少。由于只有有限数量的状态可供选择， 2^n ，所以在某个时刻能量不能再进一步降低，这对应于能量最小的状态。

有时，我们会得到一个并非绝对最小值的局部最小值。为了避免陷入能量函数的局部最小值，我们向系统添加噪声。这是通过将霍普菲尔德网络转换为玻尔兹曼机，然后使用带有 $T \searrow 0$ 的模拟退火方法来逼近能量函数的全局最小值来实现的。

示例20.8.2 (n 车问题) 我们现在回到1.3节中引入的问题。我们在这里将其推广一点，要求在一个 $n \times n$ 棋盘上放置 n 个车，使得它们互不威胁。这个优化问题可以使用霍普菲尔德网络来解决。首先，需要最小化的目标函数是和

$$E(x_{11}, \dots, x_{nn}) = \sum_{j=1}^n \left(\sum_{i=1}^n x_{ij} - 1 \right)^2 + \sum_{i=1}^n \left(\sum_{j=1}^n x_{ij} - 1 \right)^2,$$

其中 x_{ij} 是第 (i, j) 个方格的状态。如果在该位置放置了一个车，则状态为1，否则为0。 $E(x_{11}, \dots, x_{nn})$ 的最小值为零，当每行和每列都只有一个车时即可实现。由于 $x_{ij} \in \{0, 1\}$ ，那么 $x_{ij}^2 = x_{ij}$ 。

我们将证明，通过一些代数运算，函数 $E(x_{11}, \dots, x_{nn})$ 可化简为霍普菲尔德网络的能量。我们从展开平方项开始

$$\begin{aligned} \left(\sum_{i=1}^n x_{ij} - 1 \right)^2 &= \left(\sum_{i=1}^n x_{ij} \right)^2 - 2 \sum_{i=1}^n x_{ij} + 1 \\ &= \sum_{i=1}^n x_{ij}^2 + 2 \sum_{k \neq i} x_{ij} x_{kj} - 2 \sum_{i=1}^n x_{ij} + 1 \\ &= 2 \sum_{k \neq i} x_{ij} x_{kj} - \sum_{i=1}^n x_{ij} + 1. \end{aligned}$$

E 的第一项计算如下

$$F_1 = \sum_{j=1}^n \left(\sum_{i=1}^n x_{ij} - 1 \right)^2 = 2 \sum_{j=1}^n \sum_{k \neq i} x_{ij} x_{kj} - \sum_{i,j=1}^n x_{ij} + n$$

类似地， E 的第二项变为

$$F_2 = \sum_{i=1}^n \left(\sum_{j=1}^n x_{ij} - 1 \right)^2 = 2 \sum_{i=1}^n \sum_{k \neq j} x_{ik} x_{ij} - \sum_{i,j=1}^n x_{ij} + n.$$

然后

$$\begin{aligned} E(x_{11}, \dots, x_{nn}) &= F_1(x_{11}, \dots, x_{nn}) + F_2(x_{11}, \dots, x_{nn}) \\ &= 2 \left\{ \sum_{j=1}^n \sum_{k \neq i} x_{ij} x_{kj} + \sum_{i=1}^n \sum_{k \neq j} x_{ik} x_{ij} \right\} - 2 \sum_{i,j=1}^n x_{ij} + 2n \\ &= -\frac{1}{2} \sum_{\alpha, \beta} w_{\alpha, \beta} x_{\alpha} x_{\beta} - \sum_{\alpha} b_{\alpha} x_{\alpha} + 2n, \end{aligned}$$

其中 $b_{\alpha} = 2$ 以及

$$w_{\alpha, \beta} = \begin{cases} -4, & \text{if } \alpha \text{ and } \beta \text{ are placed on the same row or column;} \\ 0, & \text{otherwise.} \end{cases}$$

忽略无关常数 $2n$ ，表达式的其余部分即为霍普菲尔德网络的能量，见图20.1 b。每个单元的偏置为2，并且连接权重仅在水平和垂直连接的神经元之间非零。

棋盘上的每个 n^2 方格都对应一个感知器，因此该网络可以有 2^{n^2} 种状态。如果在该方格中有一个车，则神经元的状态为1，否则为0。

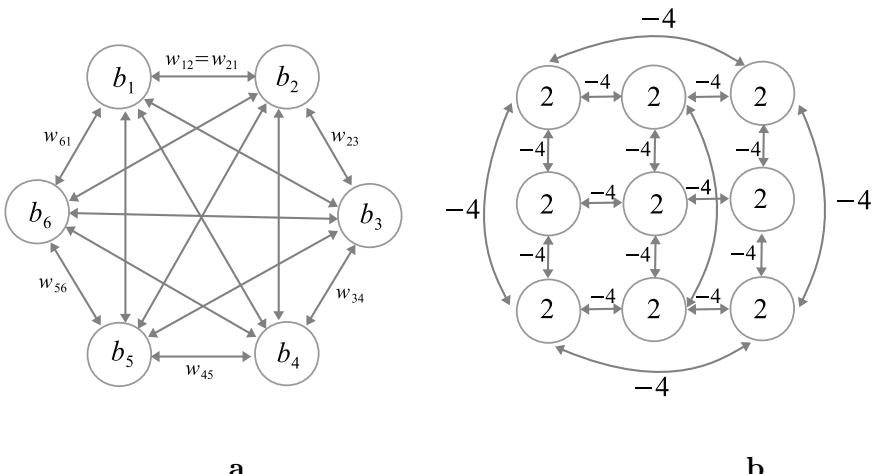


图20.1: a. A 具有 $n = 6$ 个神经元的霍普菲尔德网络; b. 与 3×3 棋盘相关的霍普菲尔德网络仅对垂直或水平连接的单元具有非零连接权重。

每个车的配置， $\mathbf{x} = (x_{11}, \dots, x_{nn})$ 对应于霍普菲尔德网络的一个状态。网络的稳定状态使能量最小化，因此为车的问题提供了一个解决方案。学习算法包括随机选择一个方格并进行更新。值得注意的是，无论霍普菲尔德网络的初始状态如何，问题的最终配置将仅包含 n 个活跃感知器，即有 n 个车解决了问题。

示例20.8.3 一个社区的人们必须对某位候选领导人投赞成票或反对票。我们将把社区行为建模为一个霍普菲尔德网络。投票给候选人的成员状态为1。如果他投票反对，状态为0。第 i 个成员和第 j 个成员之间的相互影响由权重 w_{ij} 给出。每个成员都有自己的信念阈值，用 $-b_i$ 表示。如果其他成员对第 i 个成员的输入信念影响大于他个人的信念阈值，即如果 $\sum_j w_{ij}x_j > -b_i$ ，那么第 i 个个体将投票给候选人，即状态将为1。否则，如果 $\sum_j w_{ij}x_j < -b_i$ ，他将投票反对，即值为0。在这两种情况下，第 i 个个体的状态由感知器 $H\left(\sum_j w_{ij}x_j + b_i\right)$ 的输出给出。因此，每个成员都可以被视为一个感知器，而整个社区则是一个霍普菲尔德网络。

从长远来看，网络状态 \mathbf{x} （它是由0和1组成的序列）正在使二次函数 $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T w\mathbf{x} + \mathbf{x}^T b$ 最大化。因此，如果一个人知道成员之间的相互影响 w_{ij} 以及个体的信念阈值 b_i ，那么他就可以找到群体的状态。

因此，候选人将获得一定数量的 $\sum_i x_i$ 赞成票。如果 $\sum_i x_i > 1 + n/2$ ，那么他赢得选举。这里， n 是社区的规模。

20.9 总结

二元随机神经元是一种输出为在 $\{0, 1\}$ 中取值的随机变量的神经元。取1值的概率由应用于神经元输入的 sigmoid 函数给出。玻尔兹曼机是一种神经网络，它由一组对称连接的随机神经元组成，这些神经元与网络中的所有其他神经元进行复杂的交互。引入能量函数来控制模型的复杂性。玻尔兹曼机的稳定状态由能量函数参数化的平衡分布描述，称为玻尔兹曼分布。当温度参数趋于0时，玻尔兹曼机变成霍普菲尔德网络，即由感知器组成的网络。霍普菲尔德网络对于解决复杂的组合问题很有用。为了避免陷入局部最小值，而是使用玻尔兹曼机与模拟退火方法相结合来获得能量函数的全局最小值。

20.10 练习

练习20.10.1 设 $\mathbf{x} = (x_1, \dots, x_k, \dots, x_n)$ 和 $\mathbf{x}' = (x_1, \dots, x'_k, \dots, x_n)$ 为玻尔兹曼机的两个状态，其相应能量为 $E = E(\mathbf{x})$ 和 $E' = E(\mathbf{x}')$ 。证明

$$E' - E = - \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) (x'_k - x_k).$$

练习20.10.2 考虑例20.3.2中引入的所有符号。如果 $q = (q_1, \dots, q_8)$ 是状态空间 \mathcal{X} 上的一个分布，证明当且仅当玻尔兹曼机能够精确学习分布 q 时

$$q_8 = \frac{q_5 q_6 q_7}{q_2 q_3 q_4}.$$

练习20.10.3 具有 n 个神经元的玻尔兹曼机的状态转移矩阵 $P_T = p_{ij}$ 是一个 $N \times N$ 矩阵，其中 $N = 2^n$ ，其元素 p_{ij} 表示在温度 T 下，从状态 j 到状态 i 单步转移的概率。它由以下定义

$$p_{ij} = \frac{1}{1 + e^{(E_i - E_j)/T}}, \text{ if } i \neq j$$

$$p_{ii} = 1 - \sum_{j \neq i}^N \frac{1}{1 + e^{(E_j - E_i)/T}}.$$

(a) 证明玻尔兹曼分布 $p = \frac{1}{Z} (e^{-E_1/T}, \dots, e^{-E_N/T})^t$ 是 P_T 的一个不动点, 即 $P_T p = p$ (或者等价地, p 是 P_T 的一个特征值等于1的特征向量)。

(b) 证明 P_T 的最大特征值等于1。

(c) 证明对于任何初始状态 q_0 , 由 $q_{n+1} = P_T q_n$ 递归定义的序列 $(q_n)_n$ 在 \mathbb{R}^N 中收敛到玻尔兹曼分布 p 。这有什么物理意义?

练习20.10.4 求出具有2个神经元的玻尔兹曼机情况下费希尔信息矩阵的显式公式。

练习20.10.5 设 w_{ij}, b_k 为玻尔兹曼机上的坐标, 并考虑线性算子⁹

$$A_{w,b} = \frac{1}{2} \sum_{i,j} w_{ij} \frac{\partial}{\partial w_{ij}} + \sum_k \frac{\partial}{\partial b_k}.$$

(a) 证明

$$A_{w,b} p(\mathbf{x}) = (\mathbb{E}^p[E(\mathbf{x})] - E(\mathbf{x})) p(\mathbf{x}).$$

(b) 考虑由坐标 $w_{ij}(t) = w_{ij}e^{\alpha t}, b_k(t) = b_k e^{\alpha t}$ 的指数变换诱导的 $p(\mathbf{x})$ 的光滑变形 $p_t(\mathbf{x})$, 其中 $\alpha > 0$ 。证明

$$\frac{\partial}{\partial t} p_t(\mathbf{x}) = \alpha (\mathbb{E}^p[E(\mathbf{x})] - E(\mathbf{x})) p_t(\mathbf{x}).$$

(c) 考虑玻尔兹曼分布的以下演化方程

$$\begin{aligned} \frac{\partial}{\partial t} p_t(\mathbf{x}) &= \alpha A_{w,b} p(\mathbf{x}), & \alpha > 0 \\ p_0(\mathbf{x}) &= p(\mathbf{x}), \end{aligned}$$

它对应于参数空间中的一条曲线 $(w(t), b(t))$ 。找出这条曲线的分量 $w_{ij}(t), b_k(t)$ 。

练习20.10.6 将8个不同的车随机放置在一个 8×8 棋盘上。求所有车彼此安全的概率。

⁹ 这可以被视为神经流形上的一个向量场。

练习20.10.7 (受限玻尔兹曼机, [114]) 我们考虑更一般的情况, 即玻尔兹曼机的神经元被分为两组, 可见神经元 \mathbf{v} 和隐藏神经元 \mathbf{h} , 并且同一组中的单元之间没有连接 (因此称为“受限”)。因此, 机器的状态为 $\mathbf{x} = (\mathbf{v}, \mathbf{h}) \in \mathcal{V} \times \mathcal{H} = \mathcal{X}$ 。能量定义为

$$E(\mathbf{v}, \mathbf{h}) = -\frac{1}{2}\mathbf{v}^T w\mathbf{h} - b^T \mathbf{v} - c^T \mathbf{h}$$

可见神经元和隐藏神经元的联合概率为

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})},$$

其中 Z 是配分函数。

- (a) 证明给定可见状态时, 隐藏状态 $h_j \in \mathcal{H}$ 是条件独立的;
- (b) 求 $h_j \in \{0, 1\}$ 时的条件概率 $p(h_j | \mathbf{v})$;
- (c) 计算条件对数似然函数 $\ell(\mathbf{h} | \mathbf{v}) = \ln p(\mathbf{h} | \mathbf{v})$ 及其偏导数 $\partial_{b_k} \ell(\mathbf{h} | \mathbf{v}), \partial_{c_k} \ell(\mathbf{h} | \mathbf{v}), \partial_{w_{ij}} \ell(\mathbf{h} | \mathbf{v})$;
- (d) 给定可见状态时, 隐藏状态中包含的关于参数 $\theta = (w, b, c)$ 的信息由费希尔信息度量 $g_{ij}(\theta | \mathbf{v}) = E^{p_{\mathbf{h}|\mathbf{v}}} [\partial_{\theta_i} \ell(\mathbf{h} | \mathbf{v}) \partial_{\theta_j} \ell(\mathbf{h} | \mathbf{v})]$ 描述。计算 $g_{ij}(\theta | \mathbf{v})$;
- (e) 设 $q(\mathbf{h} | \mathbf{v})$ 为给定的条件概率分布。使用与第20.4节中所做计算类似的方法, 给出权重和偏差的学习规则, 以使 $D_{KL}(q(\mathbf{h} | \mathbf{v}) \| p(\mathbf{h} | \mathbf{v}))$ 最小化。

提示与解答

我们挑选了一些习题，并给出了提示或完整解答。鼓励读者基于从章节示例和其他类似的已解习题中获得的专业知识，尝试其他习题。

第1章

习题1.9.1 (a) 该问题可用如图1 a所示的神经元进行建模。如果 $x < b$ ，那么工厂不生产任何东西，所以 $y = 0$ 。如果 $x \geq b$ ，那么收入是 $y = k(x - b)$ ，其中 k 是与生产成本相关的正的常数。考虑以下激活函数，见图1 b：

$$\varphi(x) = \begin{cases} 0, & \text{if } x < 0 \\ kx, & \text{otherwise.} \end{cases}$$

那么收入可以建模为复合函数

$$y = \varphi(x - b) = \varphi\left(\sum_{i=1}^n c_i x_i - b\right).$$

(b) 新出现的学习问题如下：给定数量 x_i ，道路容量 c_i 的值是多少才能满足或非常接近给定的收益值 y ？在这种情况下要最小化的误差函数之一是 $\frac{1}{2}(y - \varphi(x - b))^2$ 。

练习1.9.2 (a) 该问题由如图2所示的神经元建模。输出由下式给出

$$y = \begin{cases} 0, & \text{if } x \leq M \\ k(x - M), & \text{if } x > M. \end{cases}$$

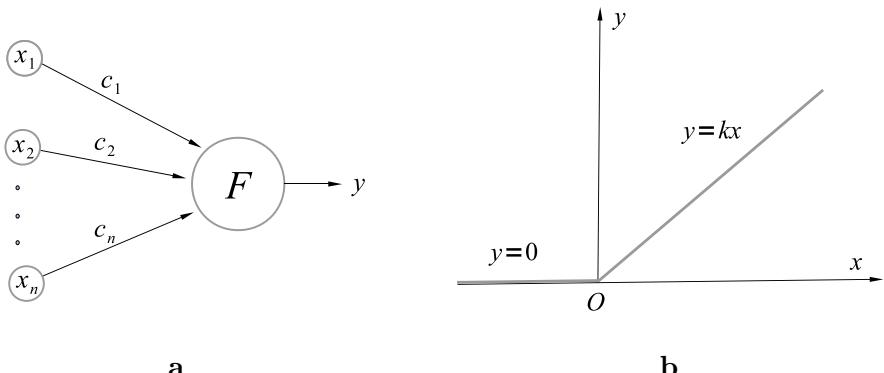


图1：a. 供应商以产能 c_j 向工厂 F 发送产品，该工厂有收入 y 。b. 收入函数 y 是分段线性的。

如果考慮激活函數

$$\varphi(x) = \begin{cases} 0, & \text{if } x \leq 0 \\ kx, & \text{if } x > 0. \end{cases}$$

如图 1b 所示，那么输出变为

$$y = \varphi(x - M) = \varphi(x_1 w_1 + \cdots + x_n w_n - M).$$

(b) 现在学习问题可以表述为：调整投资率 w_i ，以便在规定时间 t 获得关于该基金的先验计划利润 z 。

即使 y 和 z 可能永远不相等，作为变分问题的一个解给出一个可接受的答案

$$\mathbf{w} = \arg \min \frac{1}{2}(z - y)^2 = \arg \min \frac{1}{2}(z - \varphi(\mathbf{w}^T \mathbf{x} - M))^2,$$

其中 $\mathbf{w}^T = (w_1, \dots, w_n)$ 。

练习1.9.3 (a) 我们有 $C(a) = \frac{1}{2} \int_0^1 (ax - f(x))^2 dx$ 和 $C'(a) = a \int_0^1 x^2 dx - \int_0^1 xf(x)dx$, 以及 $C''(a) = \int_0^1 x^2 dx > 0$ 。那么 $a = 3 \int_0^1 xf(x)dx$ 和 $b = f(0)$.

(b) 设 $C(a, b) = \frac{1}{2} \int_0^1 \int_0^1 (ax + by - f(x, y))^2 dx dy$ 。那么方程 $\left(\frac{\partial C}{\partial a}, \frac{\partial C}{\partial b}\right) = (0, 0)$ 变为

$$\begin{aligned}\frac{1}{3}a + \frac{1}{4}b &= \int_0^1 \int_0^1 xf(x, y) dx dy \\ \frac{1}{4}a + \frac{1}{3}b &= \int_0^1 \int_0^1 yf(x, y) dx dy,\end{aligned}$$

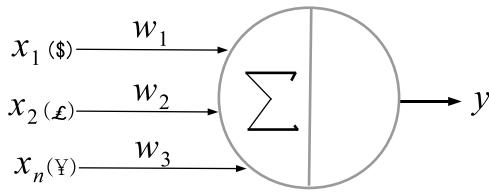


图2：以 w_j 的利率将特定货币的 x_j 金额存入给定基金。

这是一个具有唯一解 a 和 b 的线性系统。最后一个系数是 $c = f(0, 0)$ 。

练习1.9.4 (a) 使用柯西不等式，我们有

$$\det \rho_{ij} = \begin{vmatrix} \int_K x^2 & \int_K xy \\ \int_K xy & \int_K y^2 \end{vmatrix} = \left(\int_K x^2 \right) \left(\int_K y^2 \right) - \left(\int_K xy \right)^2 > 0.$$

该不等式是严格的，因为这些函数不成比例。(b) 在这种情况下，我们可以明确计算

$$\begin{aligned} \rho_{ii} &= \int_{[0,1]^n} x_i^2 dx_1 \cdots dx_n = \int_0^1 x_i^2 dx_i = \frac{1}{3} \\ \rho_{ij} &= \int_{[0,1]^n} x_i x_j dx_1 \cdots dx_n = \int_0^1 x_i dx_i \int_0^1 x_j dx_j = \frac{1}{4}, \quad i \neq j. \end{aligned}$$

第2章

练习 2.5.1 (a) 函数 $f : (0, 1) \rightarrow \mathbb{R}$, $f(t) = -t^2 + t$ 是正的，并且在 $t = 1/2$ 处取得最大值，该最大值等于 $f(1/2) = 1/4$ 。然后利用 sigmoid 性质，我们有 $\sigma' = \sigma(1 - \sigma) = f(\sigma)$ 。因此 $0 < \sigma' \leq 1/4$ 。(b) 由于我们有

$$\sigma'_c(x) = \frac{d}{dx} \sigma(cx) = c\sigma'(cx) = c\sigma(cx)(1 - \sigma(cx)) = cf(\sigma(cx)),$$

由此可得 $0 \leq \sigma'_c(x) \leq -\frac{c}{4}$ 。

练习 2.5.2 (a) 我们有

$$2H(x) - 1 = 2 \begin{cases} 0, & \text{if } x < 0 \\ 1, & \text{if } x \geq 0 \end{cases} - 1 = \begin{cases} -1, & \text{if } x < 0 \\ 1, & \text{if } x \geq 0 \end{cases} = S(x).$$

(b) 从第 (a) 部分求解 $H(x)$, 得到 $H(x) = \frac{1}{2}(S(x) + 1)$ 。然后 $\text{ReLU}(x) = xH(x) = \frac{1}{2}x(S(x) + 1)$.

练习2.5.3 (a) 使用链式法则, 我们有

$$\begin{aligned} sp'(x) &= \left(\ln(1 + e^x) \right)' = \frac{(1 + e^x)'}{1 + e^x} \\ &= \frac{e^x}{e^x(e^{-x} + 1)} = \frac{1}{1 + e^{-x}} = \sigma(x). \end{aligned}$$

(b) 由于 $sp'(x) = \sigma(x) > 0$, 函数 $sp(x)$ 是递增的。其反函数可通过以下方式得到

$$sp(x) = y \iff \ln(1 + e^x) = y \iff 1 + e^x = e^y$$

$$x = \ln(e^y - 1) \iff sp^{-1}(y) = \ln(e^y - 1).$$

(c) 对公式 $sp(x) - sp(-x) = x$ 求导, 我们得到 $sp'(x) + sp'(-x) = 1$ 。利用(a)部分, 我们得到 $\sigma(x) + \sigma(-x) = 1$ 。

练习2.5.4 一次代数运算给出

$$2\sigma(2x) - 1 = \frac{2}{1 + e^{-2x}} - 1 = \frac{2e^{2x}}{e^{2x} - 1} = \frac{e^{2x} - 1}{e^{2x} + 1} = \tanh(x).$$

练习2.5.5 (a) 注意到 $so(-x) = -so(x)$, 即 so 函数是奇函数。首先设 $x > 0$ 。那么 $so(x) = \frac{x}{1+x} = 1 - \frac{1}{1+x}$ 是递增的, 因为 $\frac{1}{1+x}$ 是递减的。因此, so(x) 在 $(0, \infty)$ 上递增。

设 $x_2 < x_1 < 0$ 。利用 so(x) 是奇函数以及 so 在 $(0, +\infty)$ 上递增这一事实, 我们有

$$so(x_1) - so(x_2) = so(-x_2) - so(-x_1) > 0,$$

这意味着 so 在 $(-\infty, 0)$ 上递增。

(b) so(x) 是连续的, 因为 $so(0-) = so(0+) = so(0) = 0$ 。此外, 我们有 $so(\infty+) = 1, so(-\infty) = -1$ 。因此, \mathbb{R} 一对一地应用到 $(-1, 1)$ 上。逆函数是在分支上计算的。如果 $y \in (0, 1)$, 我们有 $so(x) = y \Leftrightarrow \frac{x}{1+x} = y \Leftrightarrow so^{-1}(y) = \frac{y}{1-y}$ 。如果 $y \in (-1, 0)$, 我们有 $so(x) = y \Leftrightarrow \frac{x}{1-x} = y \Leftrightarrow so^{-1}(y) = \frac{y}{1+y}$ 。最后两个表达式意味着 $so^{-1}(y) = \frac{y}{1-|y|}$ 。

(c) 从 $|x + y| \leq |x| + |y|$ 开始, 我们应用 so 是递增的这一条件

$$\begin{aligned} so(|x + y|) &\leq so(|x| + |y|) = \frac{|x| + |y|}{1 + |x| + |y|} \\ &= \frac{|x|}{1 + |x| + |y|} + \frac{|y|}{1 + |x| + |y|} \\ &\leq \frac{|x|}{1 + |x|} + \frac{|y|}{1 + |y|} = so(|x|) + so(|y|). \end{aligned}$$

练习2.5.6 代数运算给出

$$\text{softmax}(y + \mathbf{c})_i = \frac{e^{y_i+c}}{\sum_j e^{y_j+c}} = \frac{e^c \cdot e^{y_i}}{e^c \cdot \sum_j e^{y_j}} = \frac{e^{y_i}}{\sum_j e^{y_j}} = \text{softmax}(y).$$

练习2.5.7 (a) $\sum_i \rho(y)_i = \frac{\sum y_i^2}{\|y\|_2^2} = 1$ 。 (b) $\rho(\lambda y)_i = \frac{\lambda^2 y_i^2}{\lambda^2 \|y\|_2^2} = \rho(y)_i$

练习2.5.8 该函数可以写成两个具有紧支集的函数之和，

$\varphi(x) = \varphi_1(x) + \varphi_2(x)$ ， 其中

$$\begin{aligned}\varphi_1(x) &= \frac{1}{2} \left(1 + \cos\left(x + \frac{3\pi}{2}\right) \right) 1_{[-\frac{\pi}{2}, \frac{\pi}{2}]}(x) \\ \varphi_2(x) &= 1_{(\frac{\pi}{2}, \infty)}(x).\end{aligned}$$

函数 $\varphi_1(x)$ 在 $[-\frac{\pi}{2}, \frac{\pi}{2}]$ 上递增，在其余部分为常数，而函数 $\varphi_2(x)$ 在 $x \geq 1$ 时等于1，在其余部分等于0。显然，对于任何 $x \leq -\frac{\pi}{2}$ 有 $\varphi(x) = 0$ ，对于任何 $x \geq \frac{\pi}{2}$ 有 $\varphi(x) = 1$ 。该函数的图像如图2.11 b所示。

练习2.5.9 (a) 由挤压函数是一个非递增的Sigmoid函数这一事实可得。

(b) 我们选择任何在某个区间上递减的Sigmoid函数。例如，函数

$$\varphi(x) = x\sigma(x)1_{[x_0, \infty)}(x),$$

其中 x_0 是 $\sigma(x) = 1/x$ 的唯一正解，它是非递增的，并且满足 $\varphi(-\infty) = 0$ 和 $\varphi(\infty) = 1$ 。

第3章

练习3.15.1 这是由积分的线性性质以及对数函数的性质得出的。

练习3.15.2 利用 $\ln x \leq x - 1$ ，我们得到

$$\begin{aligned}S(p, q) &= - \int p(x) \ln q(x) dx \geq - \int p(x)(q(x) - 1) dx \\ &= - \int p(x)q(x) dx + \int p(x) dx \\ &= 1 - \int p(x)q(x) dx.\end{aligned}$$

练习3.15.3 每一项都是非负的， $D_{KL}(p \parallel q) \geq 0$ ，

$D_{KL}(q \parallel p) \geq 0$ ，对于 $p = q$ ，它们同时达到最小值，即零。

练习3.15.4 (a) 计算表明

$$\begin{aligned}
 D_{KL}(p_1 || p_2) &= \int_0^\infty p_1(x) \ln \frac{p_1(x)}{p_2(x)} dx \\
 &= \int_0^\infty \ln \frac{\xi^1}{\xi^2} p_1(x) dx + (\xi^2 - \xi^1) \int_0^\infty x p_1(x) dx \\
 &= \ln \frac{\xi^1}{\xi^2} + (\xi^2 - \xi^1) \frac{1}{\xi^1} \\
 &= \frac{\xi^2}{\xi_1} - \ln \frac{\xi^2}{\xi_1} - 1.
 \end{aligned}$$

(b) 设 $f(x) = x - \ln x - 1$ 。由于 $f\left(\frac{\xi^2}{\xi^1}\right) \neq f\left(\frac{\xi^1}{\xi^2}\right)$, 那么 $D_{KL}(p_1 \| p_2) \neq D_{KL}(p_2 \| p_1)$ 。

练习3.15.5 设 $p_i = P(X = x_i), 1 \leq i \leq n$ 。由于 $p_i \in [0, 1]$, 那么 $-\ln p_i \geq 0$, 所以 $H(X) = -\sum p_i \ln p_i \geq 0$ 。

练习3.15.6 由于 $H(X) \geq 0$, 见练习3.15.5, 那么

$$D_{KL}(p || q) = S(p, q) - H(p) \geq S(p, q).$$

练习3.15.7 由于 Z 是 \mathcal{E} 可测的, $\mathbb{E}[Z | \mathcal{E}] = Z$ 。那么误差为

$$\|Z - \mathbb{E}[Z | \mathcal{E}]\| = \|Z - Z\| = 0.$$

这对应于精确学习。

练习3.15.8 映射 $(w, b) \rightarrow f_{w,b}(\mathbf{x})$ 对应于 \mathbb{R}^n 中的一个超平面。最优参数 (w^*, b^*) 对应于目标 \mathbf{z} 在该超平面上的正交投影的坐标。由于几何原因, 此投影是唯一的。正规方程是线性的, 因此, 可以针对 w^* 和 b^* 显式求解。

练习3.15.9 根据洛必达法则, 我们有

$$\begin{aligned}
 \lim_{\alpha \rightarrow 1} H_\alpha(p) &= \lim_{\alpha \rightarrow 1} \frac{1}{1 - \alpha} \ln \int p^\alpha(x) dx = \lim_{t \rightarrow 0} \frac{\ln \int p^{1-t}(x) dx}{t} \\
 &= \lim_{t \rightarrow 0} \frac{d}{dt} \ln \int p^{1-t}(x) dx = \lim_{t \rightarrow 0} \frac{\frac{d}{dt} \int p^{1-t}(x) dx}{\int p^{1-t}(x) dx} \\
 &= \lim_{t \rightarrow 0} \frac{- \int p^{1-t}(x) \ln p(x) dx}{\int p^{1-t}(x) dx} = \frac{- \int p(x) \ln p(x) dx}{\int p(x) dx} \\
 &= - \int p(x) \ln p(x) dx = H(p).
 \end{aligned}$$

练习3.15.10 (a) 这是一个直接的计算，涉及变量变换和配方法

$$\begin{aligned}\phi_\sigma * \phi_\sigma(v) &= \int \phi_\sigma(t)\phi_\sigma(t-v) dt = \frac{1}{2\pi\sigma^2} \int e^{-\frac{t^2}{2\sigma^2}} e^{-\frac{(t-v)^2}{2\sigma^2}} dt \\ &= \frac{1}{2\pi\sigma^2} \int e^{-\frac{(t^2-tv+v^2/2)}{\sigma^2}} dt = \frac{1}{2\pi\sigma^2} e^{-\frac{v^2}{4\sigma^2}} \int e^{-\frac{(t-v^2/2)^2}{\sigma^2}} dt \\ &= \frac{1}{2\pi\sigma} e^{-\frac{v^2}{4\sigma^2}} \int e^{-u^2} du = \frac{1}{2\sqrt{\pi}\sigma} e^{-\frac{v^2}{4\sigma^2}} = \frac{1}{\sqrt{2\pi}\sigma'} e^{-\frac{v^2}{2\sigma'^2}} = \phi_{\sigma'}(v),\end{aligned}$$

对于 $\sigma' = \sigma\sqrt{2}$ 。

(b) 类似的计算表明 $\phi_\sigma * \phi_{\sigma'} = \phi_s$ ，其中 $s = \sqrt{\sigma^2 + \sigma'^2}$ 。

练习3.15.11 (a) 如果 $D_{CS}(p, q) = 0$ ，那么 $\int pq = \sqrt{\int p^2 \int q^2}$ 。由于这意味着施瓦茨不等式中等号成立，所以这些函数必定成比例。那么，存在一个常数 λ 使得 $p(x) = \lambda q(x)$ 。因为 $\int p = \int q = 1$ ，所以可得 $\lambda = 1$ ，进而 $p = q$ 。反之则是显然的。

(b) 根据施瓦茨不等式 $\int pq \leq \sqrt{\int p^2 \int q^2}$ ，所以 \ln 的自变量属于 $(0, 1]$ ，而对数函数在该区间为负。

(c) 显然。 (d) 这可由对数函数的性质和雷尼熵的定义得出。

练习3.15.12 利用 $|\tanh| \leq |x|$ 这个事实。这可由 $\tanh'x = \frac{1 - \tanh^2 x}{1 + \tanh^2 x} \leq 1$ 推出。

第4章

练习4.17.1 (a) 计算拉普拉斯算子

$$\Delta f(x) = \frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} = e^{x_1} \sin x_2 - e^{x_1} \sin x_2 = 0.$$

(b) 由于 $\nabla f(x) = (e^{x_1} \sin x_2, e^{x_1} \cos x_2)$ ，我们有

$$\|\nabla f\| = e^{x_1} \|(\sin x_2, \cos x_2)\| = e^{x_1}.$$

(c) $\nabla f(x) = 0 \Leftrightarrow \|\nabla f\| = 0 \Leftrightarrow e^{x_1} = 0$ ，它没有解。

(d) 由于 f 是调和的（或者，因为 $\nabla f \neq 0$ ），函数 f 在 $[0, 1] \times [0, \frac{\pi}{2}]$ 的边界上达到其最小值和最大值。由于函数 $e^{x_1}, x_1 \in [0, 1]$ 和 $\sin x_2, x_2 \in [0, \pi/2]$ 都是递增的，所以 $f(x)$ 的最大值在 $(x_1, x_2) = (1, \pi/2)$ 处达到，最小值在 $(x_1, x_2) = (0, 0)$ 处达到。此外， $\min f(x) = f(0, 0) = 0$ 和 $\max f(x) = f(1, \pi/2) = e$ 。

练习4.17.2 (a) $\nabla Q(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$ 。

(b) 梯度下降给出的迭代为

$$\begin{aligned}\mathbf{x}^{n+1} &= \mathbf{x}^n - \eta \nabla Q(\mathbf{x}^n) \\ &= \mathbf{x}^n - \eta(A\mathbf{x}^n - b) = (\mathbb{I} - A)\mathbf{x}^n + b.\end{aligned}$$

(c) 海森矩阵为 $H_Q = \frac{1}{2}A$ 。

(d) 牛顿公式给出的迭代为

$$\begin{aligned}\mathbf{x}^{n+1} &= \mathbf{x}^n - H_Q^{-1}(\mathbf{x}^n) \nabla Q(\mathbf{x}^n) \\ &= \frac{1}{2} \mathbf{x}^n \mathbf{x}^n - \frac{1}{2} A^{-1}(A \mathbf{x}^n - b) \\ &= \frac{1}{2} \mathbf{x}^n + \frac{1}{2} A^{-1} b.\end{aligned}$$

假设 $\mathbf{x}^* = \lim_{n \rightarrow \infty} \mathbf{x}^n$, 对上式取极限可得

$$\mathbf{x}^* = \frac{1}{2}\mathbf{x}^* + \frac{1}{2}A^{-1}b,$$

这意味着 $x^* = A^{-1}b$ ，正如预期。极限的存在性由归纳迭代得出

$$\begin{aligned}
 \mathbf{x}^{n+1} &= \frac{1}{2}\mathbf{x}^n + \frac{1}{2}A^{-1}b \\
 &= \frac{1}{2}\left(\frac{1}{2}\mathbf{x}^{n-1} + \frac{1}{2}A^{-1}b\right) + \frac{1}{2}A^{-1}b \\
 &= \dots \\
 &= \frac{1}{2^{n+1}}\mathbf{x}^0 + \left(\frac{1}{2^{n+1}} + \dots + \frac{1}{2}\right)A^{-1}b.
 \end{aligned}$$

第一项趋于零，第二项趋于 $A^{-1}b$ 。

练习4.17.3 (a) 成本函数可以写成

$$\begin{aligned} C(\mathbf{x}) &= \frac{1}{2} \|A\mathbf{x} - b\|^2 = \frac{1}{2} \langle A\mathbf{x} - b, A\mathbf{x} - b \rangle \\ &= \frac{1}{2} \langle A^T A\mathbf{x}, \mathbf{x} \rangle - \langle A\mathbf{x}, b \rangle + \frac{1}{2} \|b\|^2. \end{aligned}$$

梯度为 $\nabla C(\mathbf{x}) = A^T A \mathbf{x} - A b$ ，黑塞矩阵由 $H_C(\mathbf{x}) = A^T A$ 给出。由于 $\det A \neq 0$ 且矩阵 $A^T A$ 是对称的， H_C 有非零实特征值，这些特征值是 A 的特征值的平方。因此， H_C 是正定的。

(b) 我们有

$$\begin{aligned}\mathbf{x}^{n+1} &= \mathbf{x}^n - \eta \nabla C(\mathbf{x}^n) = \mathbf{x}^n - \eta A^T A \mathbf{x}^n - A b \\ &= (I - \eta A^T A) \mathbf{x}^n - A b.\end{aligned}$$

练习4.17.4 (a) 通过归纳法, 我们有

(b) 对等式(4.4.17)取范数可得

$$\|v^{n+1}\| \leq \mu \|v^n\| + \eta \|\nabla f(x^n)\|.$$

考慮序列 $a_n = \|v^n\|$ 。那么 $a_{n+1} \leq \mu a_n + K$ ，其中 $K = \eta M$ 。根据 (a) 部分，序列 $(a_n)_n$ 是有界的。

练习4.17.5 (a) 这由富比尼定理推出。

(b) 对 $|f|$ 和 $|g|$ 应用(a)部分, 我们得到

$$\begin{aligned}
\|f * g\|_1 &= \int |f * g|(x) dx = \int \left| \int f(y)g(x-y) dy \right| dx \\
&\leq \int \int |f(y)| |g(x-y)| dy dx \\
&= \int (|f| * |g|)(x) dx = \int |f|(x) dx \int |g|(x) dx \\
&= \|f\|_1 \|g\|_1.
\end{aligned}$$

(c) 由(b)部分通过取 $g = G_\sigma$ 并使用 $\|G_\sigma\|_1 = \int_{\mathbb{R}} G_\sigma(x) dx = 1$ 可得。更一般地，对于任何 $1 \leq p \leq \infty$ ，我们有 $\|f_\sigma\|_p \leq \|f\|_p \|G_\sigma\|_1 = \|f\|_p$ ，即，如果 $f \in L^p$ ，那么 $f_\sigma \in L^p$ 。特别地，对于 $p = 2$ ，我们得到通过对有限能量信号进行滤波，我们也会得到一个有限能量信号。

练习4.17.6 这是一个如下所示的直接计算：

$$\begin{aligned}(G_{\sigma_1} * G_{\sigma_2})(x) &= \int G_{\sigma_1}(u)G_{\sigma_2}(x-u) du \\ &= \frac{1}{2\pi\sigma_1\sigma_2} \int e^{-\frac{1}{2}\left[\left(\frac{u}{\sigma_1}\right)^2 + \left(\frac{x-u}{\sigma_2}\right)^2\right]} du.\end{aligned}$$

完成配方后，指数可以写成

$$\left(\frac{u}{\sigma_1}\right)^2 + \left(\frac{x-u}{\sigma_2}\right)^2 = \frac{\sigma_1^2 + \sigma_2^2}{\sigma_1^2 \sigma_2^2} \left(u - \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} x\right)^2 + \frac{x^2}{\sigma_1^2 + \sigma_2^2}.$$

改变变量并计算高斯积分，我们得到

$$\begin{aligned}(G_{\sigma_1} * G_{\sigma_2})(x) &= \frac{1}{2\pi\sigma_1\sigma_2} \int e^{-\frac{\sigma_1^2+\sigma_2^2}{2\sigma_1^2\sigma_2^2}\left(u-\frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}x\right)^2} e^{-\frac{x^2}{2(\sigma_1^2+\sigma_2^2)^2}} du \\ &= \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{x^2}{2(\sigma_1^2+\sigma_2^2)}} \int e^{-\frac{\sigma_1^2+\sigma_2^2}{\sigma_1^2\sigma_2^2}\frac{v^2}{2}} dv \\ &= \frac{1}{\sqrt{2\pi}\sqrt{\sigma_1^2+\sigma_2^2}} e^{-\frac{x^2}{2(\sigma_1^2+\sigma_2^2)}} = G_{\sqrt{\sigma_1^2+\sigma_2^2}}(x).\end{aligned}$$

有一种使用傅里叶变换的证明变体如下。如果

$\mathcal{F}(f)(\xi) = \int f(x)e^{-2\pi ix\xi} dx$ 表示 f 的傅里叶变换，利用这些性质

$$\begin{aligned}\mathcal{F}(f * g)(\xi) &= \mathcal{F}(f)(\xi) \mathcal{F}(g)(\xi) \\ \mathcal{F}(e^{-ax^2}) &= \sqrt{\frac{\pi}{a}} e^{-\frac{(\pi a)^2}{a}},\end{aligned}$$

我们有

$$\begin{aligned}\mathcal{F}(G_{\sigma_1} * G_{\sigma_2})(\xi) &= \mathcal{F}(G_{\sigma_1})(\xi) \mathcal{F}(G_{\sigma_2})(\xi) = e^{-(2\pi\sigma_1\xi)^2} e^{-(2\pi\sigma_2\xi)^2} \\ &= e^{-2^2\pi^2(\sigma_1^2+\sigma_2^2)\xi^2} = \mathcal{F}(G_{\sigma}),\end{aligned}$$

对于 $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2}$ 。应用逆变换， \mathcal{F}^{-1} 得到期望的结果。

练习4.17.7 我们考虑带有约束的变分问题

$$L = \frac{1}{2}(\sigma_1^2 + \cdots + \sigma_n^2) - \lambda(\sigma_1 + \cdots + \sigma_n - s),$$

其中 λ 为拉格朗日乘数。最小值满足

$$\frac{\partial L}{\partial \sigma_j} = \sigma_j - \lambda = 0 \implies \sigma_j = \lambda, \quad \forall 1 \leq j \leq n.$$

第5章

练习5.10.1 (a) 映射为 $(0, 0) \rightarrow 0, (0, 1) \rightarrow 0, (1, 0) \rightarrow 1, (1, 1) \rightarrow 0$

◦ 点 $(1, 0)$ 被直线 $y = x - 1/2$ 与其他点分隔开。那么输出为

$y = H(x - y - 1/2)$ 的感知机学习了之前的赋值。我们有

$w_1 = 1, w_2 = -1, b = -1/2$ 。

(b) 对于 (a) 同理。

(c) 一维感知器的输出为 $y = H(-x + 1/2)$ 。权重为 $w = -1$ ，偏差为 $b = 1/2$ 。由于对于 x 布尔变量我们有 $\neg x = 1 - x$ ，所以相关线性神经元的输出为 $y = wx + b$ ，其权重和偏差由 $w = -1$ 和 $b = 1$ 给出。

(d) 我们可以将 $x_1 \wedge x_2 \wedge x_3$ 的学习过程视为将立方体角点 $(1, 1, 1)$ 与其他角点分开。这是由平面 $x_1 + x_2 + x_3 = 5/2$ 完成的。学习 $x_1 \wedge x_2 \wedge x_3$ 的感知机具有权重 $w_1 = w_2 = w_3 = 1$ 、偏差 $b = -5/2$ 和输出 $y = H(x_1 + x_2 + x_3 - 5/2)$ 。学习 $x_1 \vee x_2 \vee x_3$ 的感知机具有输出 $y = H(x_1 + x_2 + x_3 - 1/2)$ 。

练习5.10.2 证明集合 A 和 B 可以被两条不同的直线分开，然后找到一条权重为有理数的中间直线。

练习5.10.3 (a) 矩阵 $A = \mathbb{E}[XX^T]$ 是对角矩阵，其中 $A_{ij} = \sigma_j^2 \delta_{ij}$ 。最优权重是 $\mathbf{w}^* = A^{-1}\mathbf{b}$ ，其中 $w_j^* = b_j/\sigma_j^2 = \mathbb{E}[ZX_j]/\sigma_j^2$ 。

(b) 由于 Z 和 X_j 是独立的，所以 $\mathbb{E}[ZX_j] = \mathbb{E}[Z]\mathbb{E}[X_j] = 0$ 。那么代价函数为 $\xi(\mathbf{w}) = c + \mathbf{w}^T A \mathbf{w}$ ，其中 $c = \mathbb{E}[Z^2]$ 。如果 A 是正定的，其最小值在 $\mathbf{w} = 0$ 处取得。一个与输入无关的零均值目标，会以零权重被学习到；通过这种选择，输入无关紧要。(c) 牛顿迭代为

$$\begin{aligned}\mathbf{w}^{(j+1)} &= \mathbf{w}^{(j)} - H^{-1} \nabla \xi(\mathbf{w}^{(j)}) \\ &= \mathbf{w}^{(j)} - 2A^{-1}(2A\mathbf{w}^{(j)} - 2\mathbf{b}) \\ &= -3\mathbf{w}^{(j)} + 4A^{-1}\mathbf{b}.\end{aligned}$$

练习5.10.4 在线性回归中，在 $\mathbb{R}^n, \mathbf{x}_j$ 中有 N 个向量，对应于 N 个数字， $z_j, 1 \leq j \leq N$ 。我们寻找 $n+1$ 个参数， w_0, w_1, \dots, w_n ，使得平方误差之和

$$C(\mathbf{w}) = \frac{1}{N} \sum_{j=1}^N (w_1 \mathbf{x}_j^1 + \dots + w_n \mathbf{x}_j^n - w_0 - z_j)^2$$

最小。设 X 和 Z 是线性神经元的输入和目标变量。实际上，它们是通过 N 次测量得到的， $\mathbf{x}_j = X(\omega_j)$ 和 $z_j = Z(\omega_j)$ ，对于世界的某种 N 状态 $\omega_j \in \Omega, 1 \leq j \leq N$ 。此外，设 $X_0 = -1$ 。

训练集由 $\mathcal{T} = \{(\mathbf{x}_j, z_j); 1 \leq j \leq N\}$ 给出。成本函数可以用期望的经验评估来表示为

$$C(\mathbf{w}) = \mathbb{E}[(X^T \mathbf{w} - Z)^2] = \frac{1}{N} \sum_{j=1}^N (\mathbf{x}_j^T \mathbf{w} - z_j)^2.$$

因此，线性回归和线性神经元都使相同的成本函数最小化。

练习5.10.5 假设 $x_0 \in (0, 1)$ 。那么

$$y = H\left(\int_0^1 x d\delta_{x_0}(x)\right) = H(x_0) = 1,$$

因为 $x_0 > 0$ 。

假设 $x_0 \notin (0, 1)$ 。那么

$$\begin{aligned} y &= H\left(\int_0^1 x d\delta_{x_0}(x)\right) = H\left(\int_{\mathbb{R}} x 1_{[0,1]}(x) d\delta_{x_0}(x)\right) \\ &= H(x_0 1_{[0,1]}(x_0)) = 1. \end{aligned}$$

练习5.10.6 我们有

$$\langle w_1, \mathbf{x}_{i_0} \rangle = \langle w_0 + \mathbf{x}_{i_0}, \mathbf{x}_{i_0} \rangle = \langle w_0, \mathbf{x}_{i_0} \rangle + \|\mathbf{x}_{i_0}\|^2 = \langle w_0, \mathbf{x}_{i_0} \rangle + 1 > 0,$$

因为 $\langle w_0, \mathbf{x}_{i_0} \rangle \leq \|w_0\| \|\mathbf{x}_{i_0}\| = 1$ 。

由于所有点 P_j 都属于一个半圆，设 w^* 为圆直径的单位法向量，该直径将这些点与圆的其余部分分隔开。用 ρ_m 表示 w^* 与 w_m 之间的夹角，并设 $\delta = \min \{\cos \angle(w^*, \mathbf{x}_i); 1 \leq i \leq n\}$ 。那么

$$\begin{aligned} \langle w^*, w_m \rangle &= \langle w^*, w_{m-1} + \mathbf{x}_{i_{m-1}} \rangle = \langle w^*, w_{m-1} \rangle + \langle w^*, \mathbf{x}_{i_{m-1}} \rangle \\ &\geq \langle w^*, w_{m-1} \rangle + \delta. \end{aligned}$$

对不等式进行迭代，我们得到 $\langle w^*, w_m \rangle \geq \langle w^*, w_0 \rangle + m\delta$ 。那么

$$\cos \rho_m = \frac{\langle w^*, w_m \rangle}{\|w^*\| \|w_m\|} = \langle w^*, w_m \rangle \geq \langle w^*, w_0 \rangle + m\delta.$$

不等式

$$1 \geq \cos \rho_m \geq \langle w^*, w_0 \rangle + m\delta$$

是矛盾的，因为当 $m \rightarrow \infty$ 时右侧项无界增加，而左侧是有界的。因此，该过程应在有限步骤后结束。如果 N 是最大迭代次数，那么 $N\delta \sim 1$ 提供估计 $N \sim 1/\delta$ 。

练习5.10.7 感知机学习算法对于半平面中的点的情况是有效的。原因是如果一条直径分隔了由单位向量 \mathbf{x}_i 定义的点，那么同一条直径分隔任何其他其向量与先前向量方向相同的点。这源于对于所有 $\lambda > 0$ 的等价关系 $\langle w, \mathbf{x}_i \rangle > 0 \Leftrightarrow \langle w, \lambda \mathbf{x}_i \rangle > 0$ 。

练习5.10.9 考虑在步骤 k , $C(w_k) = \frac{1}{2}(z_k - w_k^T x_k)^2$ 计算的均方误差。梯度为 $\nabla_{w_k} C = -(z_k - w_k^T x_k)x_k$ 。梯度下降规则为

$w_{k+1} = w_k - \eta \nabla_{w_k} C = w_k + \eta \epsilon_k x_k$ 。如果令 $c = |x_k|$ 并定义 $\alpha = \eta c^2$, 我们得到 $w_{k+1} = w_k + \alpha \frac{\epsilon_k x_k}{|x_k|^2}$, 这就是 α -LMS更新规则。

第6章

练习6.6.2 (a) 考虑一个单位立方体, 其一个顶点位于原点, 三条边分别沿 \mathbb{R}^3 的坐标轴方向。由于没有单个平面能够分隔立方体顶点 $(0, 0, 0)$ 和 $(1, 1, 1)$, 因此单个感知器无法学习该映射。(b) 尝试一个单隐藏层网络。

练习6.6.3

$$y = H(-2H(x_1 + x_2 + 1/2) - 3/2) - H(H(x_1 + x_2 + 1/2) + 5/2) - 1.$$

练习6.6.4 (a) 根据链式法则 $\frac{\partial y}{\partial w} = x\sigma'(wx + b)$ 和 $\frac{\partial y}{\partial b} = \sigma'(wx + b)$ 。梯度为 $\nabla C = (\frac{\partial C}{\partial w}, \frac{\partial C}{\partial b})$, 其中

$$\begin{aligned}\frac{\partial C}{\partial w} &= (y - z)\frac{\partial C}{\partial w} \\ &= (\sigma(wx + b) - z)x\sigma'(wx + b) \\ \frac{\partial C}{\partial b} &= (y - z)\frac{\partial C}{\partial b} \\ &= (\sigma(wx + b) - z)\sigma'(wx + b).\end{aligned}$$

使用那个 $\sigma' \leq 1/4$ 和 $\sigma < 1$, 我们得到

$$\begin{aligned}\|\nabla C\| &= \sqrt{\left(\frac{\partial C}{\partial w}\right)^2 + \left(\frac{\partial C}{\partial b}\right)^2} \\ &= |\sigma(wx + b) - z|\sigma'(wx + b)\sqrt{1 + x^2} < \frac{1}{4}\sqrt{1 + x^2}(1 + |z|).\end{aligned}$$

(b) 对于一个步骤 $\eta > 0$, 我们有

$$\begin{aligned}w_{n+1} &= w_n - \eta x\sigma'(w_n x + b_n)(\sigma(w_n x + b_n) - z) \\ b_{n+1} &= b_n - \eta \sigma'(w_n x + b_n)(\sigma(w_n x + b_n) - z).\end{aligned}$$

练习6.6.5 (a) 设 $X^{(0)} = X$ 。在灵敏度关系中取范数

$$dY = \phi'(s^{(2)})W^{(2)T}\phi'(s^{(1)})W^{(1)T}dX^{(0)},$$

我们得到

$$\begin{aligned}\|dY\| &= |\phi'(s^{(2)})||W^{(2)T}| |\phi'(s^{(1)})| \|W^{(1)T}\| \|dX^{(0)}\| \\ &\leq \|\phi'\|^2 \|W^{(1)}\| \|W^{(2)}\| \|dX^{(0)}\|\end{aligned}$$

选择 $\eta = \frac{\epsilon}{\|\phi'\|^2 \|W^{(1)}\| \|W^{(2)}\|}$ 就足够了。

(b) 如果输入 X 有噪声, 那么就会出现小的变化 dX 。出于去除噪声的目的, 这些变化对输出变化 dY 的影响应尽可能小。这可以通过两种方式实现: 保持权重矩阵的范数较小, 或者选择具有较小 ϕ' 的激活函数。用于去除噪声的权重正则化可以作为一个约束添加到成本函数中, 例如 $\|W^{(1)}\|^2 + \|W^{(2)}\|^2 < 1$ 。

(c) 我们需要选择导数范数 $\|\phi'\|$ 最小的激活函数, 即饱和度最慢的激活函数。然后使用正切双曲线函数, 它比逻辑 sigmoid 函数饱和得更快。

练习6.6.6 (a) 各层的信号和结果由 $s^{(1)} = w_1x - b_1$, $x^{(1)} = \sigma(s^{(1)}) = \sigma(w_1x - b_1)$, $s^{(2)} = w_2\sigma(s^{(1)}) - b_2$ 以及 $y = x^{(2)} = \sigma(s^{(2)})$ 给出。使用链式法则

$$\begin{aligned}\delta^{(2)} &= \frac{\partial C}{\partial s^{(2)}} = \frac{1}{2}(y - z)\frac{\partial y}{\partial s^{(2)}} = \frac{1}{2}(y - z)\sigma'(s^{(2)}) \\ \delta^{(1)} &= \frac{\partial C}{\partial s^{(1)}} = \frac{\partial C}{\partial s^{(2)}} \frac{\partial s^{(2)}}{\partial s^{(1)}} = \delta^{(2)}w_2\sigma'(s^{(1)}).\end{aligned}$$

(b) 梯度的分量, 即 ∇C , 由以下给出

$$\begin{aligned}\frac{\partial C}{\partial w_2} &= \frac{\partial C}{\partial s^{(2)}} \frac{\partial s^{(2)}}{\partial w_2} = \delta^{(2)}\sigma(s^{(1)}) \\ \frac{\partial C}{\partial w_1} &= \frac{\partial C}{\partial s^{(1)}} \frac{\partial s^{(1)}}{\partial w_1} = \delta^{(1)}x \\ \frac{\partial C}{\partial b_2} &= \frac{\partial C}{\partial s^{(2)}} \frac{\partial s^{(2)}}{\partial b_2} = -\delta^{(2)} \\ \frac{\partial C}{\partial b_1} &= \frac{\partial C}{\partial s^{(1)}} \frac{\partial s^{(1)}}{\partial b_1} = -\delta^{(1)}.\end{aligned}$$

练习6.6.7 (a) 我们将使用逆向归纳法。注意, $W_{ij}^{(\ell)}$ 和 $\delta_i^{(L)}$ 是独立的。然后假设对于任何 $\ell + 1 \leq k \leq L$, $W_{ij}^{(\ell)}$ 和 $\delta_i^{(k)}$ 是独立的。我们需要证明 $W_{ij}^{(\ell)}$ 和 $\delta_i^{(\ell)}$ 是独立的。由于 $\phi'(x) = 1$, 我们可以用第 $(\ell + 1)$ 层的增量来表示第 ℓ 层的增量, 如下所示

$$\delta_i^{(\ell)} = \sum_j \delta_j^{(\ell+1)} W_{ij}^{(\ell+1)}.$$

现在, 根据归纳假设, $W_{ij}^{(\ell)}$ 与 $\delta_j^{(\ell+1)}$ 相互独立, 并且作为不同层中的权重, $W_{ij}^{(\ell)}$ 与 $W_{ij}^{(\ell+1)}$ 相互独立。

因此， $W_{ij}^{(\ell)}$ 独立于组合 $\sum_j \delta_j^{(\ell+1)} W_{ij}^{(\ell+1)}$ ，进而独立于 $\delta_i^{(\ell)}$ 。 (b) 通过归纳法，与 (a) 部分类似。

练习6.6.8 使用关于 $f(x) = \sigma(wx + b)$ 的近似公式

$\text{Var } f(X) \approx f'(\mathbb{E}[X])^2 \text{Var}(X)$ ，我们得到

$$\text{Var } Y \approx \sigma'(w\mathbb{E}[X] + b)^2 \text{Var}(wX + b) = \sigma'(b)^2 w^2 \text{Var}(X) = \sigma'(b)^2 w^2.$$

当偏差取较大的正值或负值时，Sigmoid函数会饱和，其导数会变小；因此 $\text{Var}(Y)$ 会减小。

练习6.6.9 使用与上一个问题中类似的想法，我们有

$$\begin{aligned} \text{Var}(Y) &= \sum_i \alpha_i^2 \text{Var}\sigma(w_i X + b_i) \approx \sum_i \alpha_i^2 \sigma'(b_i)^2 w_i^2 \text{Var}\sigma(X) \\ &= \sum_i \alpha_i^2 \sigma'(b_i)^2 w_i^2. \end{aligned}$$

练习6.6.10 由于 $p(x) = 1/(b-a)$ ，那么

$$H(p) = - \int_a^b p(x) \ln p(x) dx = - \frac{1}{b-a} \int_a^b \ln \frac{1}{b-a} dx = \ln(b-a).$$

练习6.6.11 令 $p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ 。使用关系 $\int_{\mathbb{R}} p(x) dx = 1$ 和 $\int_{\mathbb{R}} (x-\mu)^2 p(x) dx = \sigma^2$ ，我们有

$$\begin{aligned} H(p) &= - \int_{\mathbb{R}} p(x) \ln p(x) dx \\ &= \frac{1}{2} \ln(2\pi) \int_{\mathbb{R}} p(x) dx + \ln \sigma \int_{\mathbb{R}} p(x) dx + \frac{1}{2\sigma^2} \int_{\mathbb{R}} (x-\mu)^2 p(x) dx \\ &= \frac{1}{2} \ln(2\pi) + \ln \sigma + \frac{1}{2} = \ln(\sigma\sqrt{2\pi e}). \end{aligned}$$

第7章

练习7.11.1 由 $|\tanh x| \leq 1$ 可得。

练习7.11.2 (a) 设 $f \in \mathcal{F}$ ，其中 $f(x) = ax + b$ 。那么模不等式得出，对于所有 $x \in [0, 1]$ ，有 $|f(x)| \leq |a||x| + |b| \leq |a| + |b| < 1$ 。族 \mathcal{F} 以 $M = 1$ 一致有界。

修复 $\epsilon > 0$ 并选择 $\eta = \epsilon/|a|$ 。然后对于任何 $x, x' \in [0, 1]$ ，在 $|x - x'| < \eta$ 的情况下，我们有 $|f(x) - f(x')| = |ax + b - ax' - b| = |a||x - x'| < \epsilon$ 。因此，族 \mathcal{F} 是等度连续的。

(b) 应用阿尔泽拉-阿斯科利定理, 我们得到: 在具有一维输入和输出的线性神经元的所有可能结果中, $y = ax + b$, 其满足正则化约束, $|a| + |b| < 1$, 存在一个序列 $y_n = a_n x + b_n$, 它一致收敛于一个仿射函数, $f(x) = ax + b$ 。简而言之, 线性神经元具有学习某个仿射函数的能力。

练习7.11.3 根据微积分基本定理, 我们有 $f(x_2) - f(x_1) = \int_{x_1}^{x_2} f'(x) dx$ 。然后柯西不等式给出

$$\begin{aligned} |f(x_2) - f(x_1)|^2 &= \left| \int_{x_1}^{x_2} f'(x) dx \right|^2 \leq \left(\int_{x_1}^{x_2} 1 dx \right) \left(\int_{x_1}^{x_2} f'(x)^2 dx \right) \\ &\leq M|x_2 - x_1|. \end{aligned}$$

那么对于任何给定的 $\epsilon > 0$, 选择 $\eta = \epsilon^2/M$ 并应用等度连续性的定义。

练习7.11.4 固定 $\epsilon = 1$ 。那么对于任何 $x_0 \in D$, 存在 $\eta > 0$ 使得对于 $x \in (x_0 - \eta, x_0 + \eta)$ 我们有

$$|f(x)| \leq |f(x) - f(x_0)| + |f(x_0)| \leq 1 + M, \quad \forall f \in \mathcal{F}.$$

注意, $\bigcup_{x_0 \in D} (x_0 - \eta, x_0 + \eta) = (a, b)$, 任何 $x \in (a, b)$ 都属于 $(x_0 - \eta, x_0 + \eta)$ 类型的邻域。

练习7.11.5 设 $\phi \in \mathcal{F}$ 。固定 $\epsilon > 0$ 。那么由 \mathcal{F} 的等度连续性, 存在 $\eta_k > 0$ 使得对于 $|x - x'| < \eta_k, 1 \leq j \leq k$ 有 $|f_j(x) - f_j(x')| < \epsilon/k$ 。选择 $\eta = \min_{1 \leq j \leq k} \eta_j$ 。我们有

$$|\phi(x) - \phi(x')| \leq \sum_{j=1}^k |w_j| |f_j(x) - f_j(x')| \leq \sum_{j=1}^k |f_j(x) - f_j(x')| < \epsilon,$$

对于 $|x - x'| < \eta$ 。

练习7.11.6 由迪尼定理应用于递减的正函数序列 $f_n = |f(x) - G_n(x)|$ 可得。

练习7.11.7 周期为 T 的周期函数可被视为半径为 $R = 2\pi/T$ 的圆上的函数, 此圆为紧致集, 记为 K 。因此, $f \in C(K)$ 。所有可写成三角和形式的函数 F 构成 $C(K)$ 的一个子代数, 该子代数能分离点且包含常数。应用斯通 - 魏尔斯特拉斯定理可得所需结果。

练习7.11.8 命题7.7.2证明中的主要成分是 $\|\sigma'\| \leq 1/4$ 。鉴于大多数Sigmoid函数关于原点对称, 我们有 $\|\sigma'\| = \sigma'(0)$ 。

因此，该结果适用于所有满足 $\sigma'(0) < \lambda < 1$ 的递增可微Sigmoid函数。

练习7.11.9 根据阿尔泽拉 - 阿斯克利定理，我们最终可以通过取子序列假设，即 f_j 在 $[a, b]$ 上一致收敛于 f 。然后 f_j^2 在 $[a, b]$ 上一致收敛于 f^2 。接着 $\int_a^b f_j^2(x) dx \rightarrow \int_a^b f^2(x) dx$ 。利用假设，可得 $\int_a^b f^2(x) dx = 0$ 。由于 $f^2 \geq 0$ ，所以 $f = 0$ 。也就是说 $\lim_{j \rightarrow \infty} f_j = 0$ ，是一致的。我们还注意到该证明可以用反证法完成，而无需使用阿尔泽拉 - 阿斯克利定理。

练习7.11.10 (a) 设 $\epsilon > 0$ 为任意固定值。利用函数 $K(s, t)$ (在紧集上的连续函数) 的一致连续性，对于任意 $\rho > 0$ ，存在 $\eta_\rho > 0$ 使得如果 $|s' - s| < \eta_\rho$ ，那么 $|K(s', t) - K(s, t)| < \rho$ 。由于 ρ 是任意的，我们选择它使得 $\rho < \epsilon / \sqrt{M(d - c)}$ 。

对于任意 $g \in \mathcal{F}_M$ ，应用柯西不等式，我们有

$$\begin{aligned} |g(s') - g(s)| &= \left| \int_c^d (K(s', t) - K(s, t)) h(t) dt \right| \\ &\leq \left(\int_c^d |K(s', t) - K(s, t)|^2 dt \right)^{1/2} \left(\int_c^d h(t)^2 dt \right)^{1/2} \\ &\leq \rho \sqrt{d - c} \sqrt{M} < \epsilon. \end{aligned}$$

因此，当 $|s' - s| < \eta$ 时，我们有 $|g(s') - g(s)| < \epsilon$ ，其中 ϵ 和 η 与 g 无关。

(b) 利用柯西不等式，对于任何 $g \in \mathcal{F}_M$ ，我们有

$$\begin{aligned} |g(s)| &= \left| \int_c^d K(s, t) h(t) dt \right| \leq \int_c^d |K(s, t)| |h(t)| dt \\ &\leq \left(\int_c^d K^2(s, t) dt \right)^{1/2} \left(\int_c^d h(t)^2 dt \right)^{1/2} \\ &\leq \left(\int_c^d K^2(s, t) dt \right)^{1/2} M^{1/2}. \end{aligned}$$

函数 $s \rightarrow \int_c^d K^2(s, t) dt$ 在紧致集 $[a, b]$ 上连续，所以它是有界的。由此可知，所有函数 g 都有一个共同的上界。

练习7.11.11 (a) 这只是对代数公理的验证。(b) 它由逼近的斯通 - 魏尔斯特拉斯定理推出。(c) 在 $[a, b]$ 上的任何连续函数都可以由 e^x 中的一系列多项式一致逼近。

第8章

练习8.8.1 (a) 对于任何 $\phi \in C_0^\infty$ ，我们有

$$\begin{aligned} - \int H(x - x_0) \phi'(x) dx &= - \int_{x_0}^{\infty} \phi'(x) dx = -(\phi(\infty) - \phi(x_0)) = \phi(x_0) \\ &= \int \phi \delta(x - x_0) dx. \end{aligned}$$

(b) 对于任何 $\phi \in C_0^\infty$

$$\begin{aligned} - \int \text{ReLU}(x) \phi'(x) dx &= - \int_0^{\infty} x \phi'(x) dx = -\left(x \phi(x)\right|_0^{\infty} - \int_0^{\infty} \phi(x) dx) \\ &= \int_0^{\infty} \phi(x) dx = \int H(x) \phi(x) dx. \end{aligned}$$

其他的类似。

练习8.8.2 通过移动索引和解析求和，我们有

$$\begin{aligned} \sum_{i=0}^{N-1} \alpha_i 1_{[x_i, x_{i+1})}(x) &= \sum_{i=0}^{N-1} \alpha_i [H(x - x_i) - H(x - x_{i+1})] \\ &= \sum_{i=0}^{N-1} \alpha_i H(x - x_i) - \sum_{j=1}^N \alpha_{j-1} H(x - x_j) \\ &= \alpha_0 H(x - x_0) + \sum_{i=1}^{N-1} \alpha_i H(x - x_i) \\ &\quad - \sum_{i=1}^{N-1} \alpha_{i-1} H(x - x_i) - \alpha_{N-1} H(x - x_N) \\ &= \alpha_0 H(x - x_0) + \sum_{i=1}^{N-1} (\alpha_i - \alpha_{i-1}) H(x - x_i) - \alpha_{N-1} H(x - x_N). \end{aligned}$$

与和 $\sum_{i=0}^N c_i H(x - x_i)$ 相等，我们得到以下系数：

$c_0 = \alpha_0, c_i = \alpha_i - \alpha_{i-1}$ 和 $c_N = \alpha_{N-1}$ 。
 练习8.8.4 考虑一个正实数的递减序列， $\epsilon_n \searrow 0$ ，并考虑函数 $f_n(x) = |g(x) - g_{\epsilon_n}(x)|$ 。根据 $g_{\epsilon_n}(x)$ 的构造，对于任何 x ，我们有 $0 \leq f_{n+1}(x) \leq f_n(x)$ 。根据迪尼定理，函数序列 f_n 一致收敛于 0，即 g_{ϵ_n} 一致收敛于 $g(x)$ 。

练习8.8.6 通过变量变换并使用洛必达法则，我们有

$$\begin{aligned}\lim_{\alpha \searrow 0} \varphi_\alpha(x) &= \lim_{\alpha \searrow 0} \alpha \ln(1 + e^{x/\alpha}) = \lim_{\alpha \searrow 0} \frac{\ln(1 + e^{x/\alpha})}{\frac{1}{\alpha}} \\ &= \lim_{t \nearrow \infty} \frac{\ln(1 + e^{tx})}{t} = \lim_{t \nearrow \infty} \frac{(1 + e^{tx})'}{1 + e^{tx}} \\ &= \lim_{t \nearrow \infty} \frac{x e^{tx}}{1 + e^{tx}} = \lim_{t \nearrow \infty} \frac{x}{1 + e^{-tx}} \\ &= \lim_{t \nearrow \infty} x \sigma_t(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0, \end{cases}\end{aligned}$$

这就得到了结果的第一部分。如果令 $t = 1/\alpha > 0$ ，那么

$$\varphi_\alpha(x) = \alpha \ln(1 + e^{x/\alpha}) = \frac{\ln(1 + e^{xt})}{t},$$

对于 α 小 (t 大) 的情况，它的表现如同 $\frac{x}{1+e^{-xt}}$ ，对于 $x > 0$ ，它在 t 中是递增的。

第9章

练习9.8.1 在这种情况下，目标空间是度量空间 (\mathbb{R}, d) ，其距离函数为 $d(x, y) = |x - y|$ 。近似空间是 \mathbb{Q} 。任何实数都可以由有理数来逼近。实际上，实数域的一种构造就是基于这个性质。

练习9.8.2 目标空间是连续函数空间 $C[a, b]$ ，赋予距离

$d(f, g) = \sup_{[a, b]} |f(x) - g(x)|$ 。近似空间是所有多项式 $\mathcal{P}[a, b]$ 的集合。

在一个紧致区间上，任何连续函数都可以由多项式函数学习得到。

练习9.8.3 设 $\mathcal{T} = \{\alpha x_1 + \lambda x_0; \alpha, \lambda \in \mathbb{R}\}$ 为由非共线向量 x_0 和 x_1 生成的空间。定义泛函 $L : \mathcal{T} \rightarrow \mathbb{R}$, $L(t) = \lambda$ ，其中 $t = \alpha x_1 + \lambda x_0$ 。显然 L 是线性的。通过与引理9.3.1证明中使用的过程类似的方法，我们可以证明 L 也是有界的，其范数为 $\|L\| < 1/\delta$ ，其中 δ 是 x_0 与 x_1 的支撑线之间的距离。根据哈恩 - 巴拿赫定理，泛函 L 可以扩展为 \mathcal{X} 上的线性有界泛函，且保持相同的界。我们可以很容易地验证 $L(x_0) = 1$ 和 $L(x_1) = 0$ 。

对 x_0 对称地应用练习9.8.3中使用的构造，然后对 x_1 进行同样的操作，我们在 \mathcal{X} 上得到两个线性有界泛函 L_1 和 L_2 ，使得

$$L_1(x_0) = 0, \quad L_1(x_1) = 1, \quad L_2(x_0) = 1, \quad L_2(x_1) = 0.$$

考慮平均泛函 $L = \frac{1}{2}(L_1 + L_2)$ ，它是线性且有界的。我们有 $L(x_0) = L(x_1) = 1/2$ 。此外，它的范数满足

$$\|L\| = \frac{1}{2}\|L_1 + L_2\| \leq \frac{1}{2}(\|L_1\| + \|L_2\|) \leq \frac{1}{2}\left(\frac{1}{\delta_0} + \frac{1}{\delta_1}\right) = \frac{\delta_0 + \delta_1}{2\delta_0\delta_1}.$$

通过类似的过程，我们可以证明以下更一般的陈述：给定 \mathcal{X} 中线性无关的向量集 $\{x_0, x_1, \dots, x_N\}$ ，在 \mathcal{X} 上存在一个有界线性泛函 L ，使得 $L(x_0) = L(x_1) = \dots = L(x_N)$ 。

练习9.8.5 在练习9.8.4中考虑空间 $\mathcal{X} = C[a, b]$ 和独立向量

$x_0 = \sin t, x_1 = \cos t$ 。那么存在一个有界线性泛函 $L : C[a, b] \rightarrow \mathbb{R}$ 使得 $L(\sin t) = L(\cos t)$ 。根据表示定理，定理E.5.6，在 $[a, b]$ 上存在唯一的有限带号博雷尔测度 μ ，使得

$$L(f) = \int_a^b f(t) d\mu(t), \quad \forall f \in C[a, b].$$

因此，恒等式 $L(\sin t) = L(\cos t)$ 变为

$$\int_a^b \sin t d\mu(t) = \int_a^b \cos t d\mu(t).$$

此外，由于 $\|L\| = |\mu|([a, b])$ 。在练习9.8.4中给出了 $\|L\|$ 的一个上界。

练习9.8.6 (a) 注意， $L(P) = P(1)$ ，所以 L 是一个线性泛函。由于

$$|L(P)| = |P(1)| \leq \sup_{[0,1]} |P(x)| = \|P\|_\infty,$$

由此可得 $\|L\| \leq 1$ 。

(b) 泛函 L 可根据哈恩 - 巴拿赫定理线性扩展到 $C[0, 1]$ ，同时保持界 $\|L\| \leq 1$ 。由于此时 L 是 $C[0, 1]$ 上的有界线性泛函，根据表示定理（定理E.5.6），存在测度 μ 使得对于所有 $f \in C[0, 1]$ ，有

$L(f) = \int_0^1 f(x) d\mu(x)$ 。特别地，对于 $f = P$ ，我们得到

$L(P) = \int_0^1 P(x) d\mu(x)$ ，或者等价地，

$$\int_0^1 P(x) d\mu(x) = a_0 + a_1 + \dots + a_n, \quad \forall P \in \mathcal{P}([0, 1]).$$

练习9.8.7 (b) 函数 $\phi(x) = e^{-x^2} \in L^1(\mathbb{R})$ 和 $\int_{\mathbb{R}} e^{-x^2} dx \neq 0$ 。根据注记9.3.19 的第2点，函数 $\phi(x)$ 在 L^1 意义下具有歧视性。练习9.8.8 (a) 输出为

$$y = \sum_{i=1}^{N_2} \alpha_i \sigma \left(\sum_{j=1}^{N_1} w_{ji} \sigma(\lambda_j x + b_j) + \beta_i \right), \quad (1)$$

使用以下符号表示： b_j 是第一个隐藏层中神经元的偏差， β_i 表示第二个隐藏层中神经元的偏差， λ_j 是从输入到第一层的权重， w_{ji} 是隐藏层之间的权重，并且 α_i 是第二个隐藏层与输出之间的权重。

(b) 将类型(1)的表达式与标量进行加、减和乘法运算，我们得到相同类型但参数不同的结果，最终有些参数等于零。

练习9.8.9 (a) 取连续函数为仿射函数， $f(x) = w^T x + b$ 。(b) 令 \mathcal{U} 表示类型(1)的连续函数集，即所有双隐藏层前馈神经网络的输出集。我们证明 \mathcal{U} 在 $C(I_n)$ 中是稠密的。通过反证法，如果 \mathcal{U} 不是稠密的，根据引理9.3.2，在 $C(I_n)$ 上存在一个非零有界线性泛函 L ，使得 $L|_{\mathcal{U}} = 0$ 。根据表示定理，定理E.5.6，在 I_n 上存在一个符号测度 μ ，使得

$$\int_{I_n} \sum_{i=1}^{N_2} \alpha_i \sigma \left(\sum_{j=1}^{N_1} w_{ji} \sigma(\lambda_j x + b_j) + \beta_i \right) d\mu(x) = 0,$$

对于参数的所有值。特别地，

$$\int_{I_n} \sigma \left(\sum_{j=1}^{N_1} w_{ji} \sigma(\lambda_j x + b_j) + \beta_i \right) d\mu(x) = 0.$$

由于 $f(x) = \sum_{j=1}^{N_1} w_{ji} \sigma(\lambda_j x + b_j) + \beta_i$ 是连续的且 σ 具有强区分性，所以得出 $\mu = 0$ ，这一事实与 $\|L\| = |\mu|(I_n)$ 相矛盾。

第10章

练习10.11.1 (a) 网络输出具有以下形式

$$G(x) = \sum_{i=1}^2 \alpha_i H(\mathbf{w}^T \mathbf{x} + b_i) = \sum_{i=1}^2 \alpha_i H(w_{1i}x_1 + w_{2i}x_2 + b_i).$$

我们做出简化假设 $w_{11} = w_{12} = w_1$ 和 $w_{21} = w_{22} = w_2$ 。然后我们剩下6个参数，它们满足4个方程：

$$\begin{aligned} 0 &= G(0, 0) = \alpha_1 H(b_1) + \alpha_2 H(b_2) \\ 1 &= G(0, 1) = \alpha_1 H(w_2 + b_1) + \alpha_2 H(w_2 + b_2) \\ 1 &= G(1, 0) = \alpha_1 H(w_1 + b_1) + \alpha_2 H(w_1 + b_2) \\ 0 &= G(1, 1) = \alpha_1 H(w_1 + w_2 + b_1) + \alpha_2 H(w_1 + b_2). \end{aligned}$$

一种可能的解决方案是 $\alpha_1 = 1, \alpha_2 = -1, w_1 = -0.5, w_2 = -2, b_1 = 2.25, b_2 = 0.25$ 。因此，一个学习 XOR 函数的输出是

$$G(x_1, x_2) = H(-0.5x_1 - 2x_2 + 2.25) - H(-0.5x_1 - 2x_2 + 0.25).$$

练习10.11.3 显然， $\psi(x) = e^{-x^2} \in L^1(\mathbb{R})$ ，其傅里叶变换为

$\Psi(\xi) = \widehat{\psi}(\xi) = \sqrt{\pi}e^{-\xi^2/4}$ ，所以 $\Psi(1) = \sqrt{\pi}/e^{1/4}$ 。那么Irie-Miyake公式就变成了

$$f(\mathbf{x}) = \frac{e^{1/4}}{(2\pi)^n \sqrt{\pi}} \int_{\mathbb{R}^{n+1}} e^{i\omega_0 - (\mathbf{x}^t \mathbf{w} - w_0)^2} F(\mathbf{w}) d\mathbf{w}.$$

练习10.11.4 考虑函数

$$f(x) = \begin{cases} e^{-\frac{1}{x-\frac{1}{2}}}, & \text{if } \frac{1}{2} < x \leq 1 \\ 0, & \text{if } 0 \leq x \leq \frac{1}{2}. \end{cases}$$

可以证明 f 在 $[0, 1]$ 上连续，但它不是解析的。（如果 f 在 $(0, 1)$ 上是解析的，那么，由于 $f|_{(0,1/2)} = 0$ ，根据解析函数的恒等定理，我们在 $(0, 1)$ 上得到 $f = 0$ ，这是矛盾的。）由于逻辑Sigmoid函数是解析的，那么网络的输出也是解析的，而 f 不是。

练习10.11.5 利用条件 (ii) 变为级数 $\sum_{i \geq 1} \frac{g_i^2}{\lambda_i^{2(n+1)}}$ 的收敛性。

练习10.11.6 对与 $K^{(n)}(t, t) = \sum_{i \geq 1} \lambda_i^n e_i^2(t)$ 相关的式子进行积分可得

$$\int_0^1 K^{(n)}(t, t) dt = \sum_{i \geq 1} \lambda_i^n \int_0^1 e_i^2(t) dt = \sum_{i \geq 1} \lambda_i^n,$$

并利用左边的积分是有限的这一条件。

练习10.11.7 (a) 考虑以1、3、5和7为中心、长度为2的区间。然后选择简单函数

$$c(x) = 1_{[0,2]}(x) + 3.1_{[2,4]}(x) + 2.1_{[4,6]}(x) + 1_{[7,8]}(x),$$

它能学习数据。(b) 由于每个指示函数都可以写成阶梯函数的差，例如， $1_{[2,4]}(x) = H(x - 2) - H(x - 4)$ ，我们得到

$$c(x) = H(x - 0) + 2H(x - 2) - H(x - 4) - H(x - 6) - H(x - 8),$$

并选择这个函数作为 $G(x)$ 。

练习10.11.8 它由复合结合性性质得出。

第11章

练习11.10.1 (a) 由于 $\{\omega; c \leq t\} = \begin{cases} \emptyset, & \text{if } c > t \\ \Omega, & \text{if } c \leq t \end{cases}$, 由 c 生成的 σ -域是由集合 \emptyset 和 Ω 生成的 σ -域, 即平凡代数 $\{\emptyset, \Omega\}$ 。

(b) 如下通过双重包含关系证明: 由于

$$\{\omega; c + X(\omega) \leq t\} = \{\omega; X(\omega) \leq t - c\} \in \mathfrak{S}(X),$$

那么 $\mathfrak{S}(c + X) \subset \mathfrak{S}(X)$ 。类似地, 由于

$$\{\omega; X(\omega) \leq t\} = \{\omega; c + X(\omega) \leq t + c\} \in \mathfrak{S}(c + X),$$

由此可得 $\mathfrak{S}(X) \subset \mathfrak{S}(c + X)$ 。

(c) 假设 $c > 0$ 。利用 $\{\omega; cX(\omega) \leq t\} = \{\omega; X(\omega) \leq t/c\} \in \mathfrak{S}(X)$, 因此 $\mathfrak{S}(cX) \subset \mathfrak{S}(X)$ 。反向包含关系可类似证明。

练习11.10.2 (a) 由于 Y_n 的值由 X_1, \dots, X_n 决定, 那么

$Y_n = F(X_1, \dots, X_n)$, 因此 $\mathfrak{S}(Y_n) \subset \mathfrak{S}(X_1, \dots, X_n)$ 。严格包含关系由 (b) 部分得出。

(b) 由于 $X_n = Y_n - Y_{n-1}$, 那么 $\mathfrak{S}(X_n) \subset \mathfrak{S}(Y_{n-1}, Y_n)$ 。我们有

$$\begin{aligned} \mathfrak{S}(X_1, \dots, X_n) &= \mathfrak{S}\left(S(X_1) \cup \dots \cup \mathfrak{S}(X_n)\right) \\ &\subset \mathfrak{S}\left(\mathfrak{S}(Y_0, Y_1) \cup \dots \cup \mathfrak{S}(Y_{n-1}, Y_n)\right) \\ &\subset \mathfrak{S}(Y_1, \dots, Y_n), \end{aligned}$$

这证明了反向包含关系。

(c) Z_n 的公式可以用归纳法证明。 σ -域的恒等式由 (b) 部分得出, 这意味着

$$\mathfrak{S}(Z_1, \dots, Z_n) = \mathfrak{S}(Y_1, \dots, Y_n) = \mathfrak{S}(X_1, \dots, X_n).$$

练习11.10.3 我们绘制集合 A 和 B 相交的维恩图。我们得到4个区域: $R_1 = A \setminus B, C_2 = A \cap B, R_3 = (A \cup B)^c$ 和 $R_4 = B \setminus A$ 。所需的 σ -代数由划分 $\{R_1, R_2, R_3, R_4\}$ 生成, 即其元素通过取子集的并集得到。有 2^4 个元素。使用德摩根定律我们得到练习中给出的集合。例如,

$$R_1 \cup R_4 = (A \setminus B) \cup (B \setminus A) = (A \cap B^c) \cup (A^c \cap B).$$

练习11.10.4 我们绘制维恩图，其中集合 A, B 和 C 相交。我们得到了 Ω 的一个划分为 8 个不同区域的划分。 \mathfrak{S} -域也将有 $2^8 = 256$ 个元素，太多以至于无法明确写出。

练习11.10.5 (a) 我们有

$$\mathcal{F}_{E_1, E_2} = \{\emptyset, E_1, E_2, E_1 \cup E_2, E_1 \cap E_2, (E_1 \cup E_2)^c, (E_1 \cap E_2)^c, \Omega\}.$$

只需要恢复集合 E_1 和 E_2 。这通过以下方式完成

$$E_1 = \left(\bigcup_{j=2^N} E_j \right)^c, \quad E_2 = \left(\bigcup_{j=1, j \neq 2}^N E_j \right)^c.$$

(b) 由于 $\mathcal{F}_{E_1} = \{\emptyset, E_1, E_1^c, \Omega\}$ ，并且 E_1 可以从其他集合中作为 $E_1 = \bigcup_{i=2}^N E_i$ 恢复，因此 \mathcal{F}_{E_1} 是一个可恢复的信息主体。利用 $\mathcal{F}_{E_1} \subset \mathcal{F}_{E_1, E_2}$ ，可以得出 \mathcal{F}_{E_1} 不是最大的。

练习11.10.6 这是定理C.1.3的一个应用。将通过双重包含关系来证明。设

$\mathcal{P} = \left\{ \bigcap_{i=1}^k \{\omega; X_i(\omega) \leq x_i\}, k \geq 1, x_i \in \mathbb{R} \right\}$ ，它是一个 p -系统（对交集封闭）。对于 $k = 1$ ，我们有 $\{\omega; X_i(\omega) \leq x_i\} \in \mathcal{P}$ ，这意味着 $\mathfrak{S}(\{\omega; X_i(\omega) \leq x_i\}) \in \mathfrak{S}(\mathcal{P})$ ，或者等价地 $\mathfrak{S}(X_i) \subset \mathfrak{S}(\mathcal{P})$ 。取并集得到 $\bigcup_i \mathfrak{S}(X_i) \subset \mathfrak{S}(\mathcal{P})$ ，取 σ -代数我们得到 $\mathfrak{S}\left(\bigcup_i \mathfrak{S}(X_i)\right) \subset \mathfrak{S}(\mathcal{P})$ ，这意味着包含关系 $\mathfrak{S}(X) \subset \mathfrak{S}(\mathcal{P})$ 。接下来，我们证明相反的包含关系。由于 \mathcal{P} 包含在 $\bigcap_i \mathfrak{S}(X_i)$ 中，我们也有 $\mathcal{P} \subset S\left(\bigcup_{i=1}^n \mathfrak{S}(X_i)\right) = \mathcal{D}$ 。作为一个 p -系统， \mathcal{P} ，包含在一个 d -系统， \mathcal{D} 中，根据附录中的定理C.1.3，我们得到

$$\mathfrak{S}(\mathcal{P}) \subset S\left(\bigcup_{i=1}^n \mathfrak{S}(X_i)\right) = \mathfrak{S}(X),$$

这证明了相反的包含关系。

练习11.10.7 (a) 由于 \tilde{Y} 由 Y 决定，所以我们有 $\tilde{Y} = f(Y)$ 。然后，根据附录中的命题 D.5.1，我们有 $\mathfrak{S}(\tilde{Y}) \subset \mathfrak{S}(Y)$ ，或者 $\tilde{\mathcal{E}} \subset \mathcal{E}$ 。那么 $\mathcal{I} \setminus \mathcal{E} \subset \mathcal{I} \setminus \tilde{\mathcal{E}}$ 。在两边取生成的 σ -代数得到 $\mathcal{L} \subset \tilde{\mathcal{L}}$ 。(b) 如果从 ℓ 层中去掉一个神经元，一个逐字的论证表明 $\mathcal{L}^{(\ell)} \subset \tilde{\mathcal{L}}^{(\ell)}$ 。总之，去掉神经元会导致在发生去掉操作的层中出现信息损失。

练习11.10.8 (a) 在这种情况下， Y 取决于 \tilde{Y} ，所以 $S(Y) \subset S(\tilde{Y})$ 。与练习11.10.7 中使用的证明类似，可得包含关系 $\tilde{\mathcal{L}} \subset \mathcal{L}$ 。

练习11.10.9 如果隐藏层中的一个单元足以对每个半平面中的点进行分类。

万一 \mathbf{b} ，我们在隐藏层中需要三个单元，每个单元对应三角形的一条边。每个单元学习一个半平面，一个三角形可以写成三个半平面的交集。类似地，万一 \mathbf{c} ，我们需要4个神经元。使用激活函数 $H(x)$ 会导致三角形和矩形的角很尖锐，而使用 sigmoid 函数会导致角变圆。

练习11.10.10 足以证明两部分都等于sigma域 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ 。这将通过双重包含来完成。

我们首先证明 $(\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ 。由于 $\mathcal{H} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ 以及 $\mathcal{F} \cup \mathcal{G} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ ，所以可得 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G}) \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ ，进而 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G}) \cup \mathcal{H} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ 。在两边取西格玛算子得到 $\mathfrak{S}(\mathfrak{S}(\mathcal{F} \cup \mathcal{G}) \cup \mathcal{H}) \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ ，这就是 $(\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H})$ 。我们现在证明 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H}) \subset (\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H}$ 。从 $\mathcal{F} \cup \mathcal{G} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G})$ 开始，我们有 $\mathcal{F} \cup \mathcal{G} \cup \mathcal{H} \subset \mathfrak{S}(\mathcal{F} \cup \mathcal{G}) \cup \mathcal{H}$ 。取西格玛算子我们得到 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H}) \subset \mathfrak{S}(\mathfrak{S}(\mathcal{F} \cup \mathcal{G}) \cup \mathcal{H})$ ，这就是 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H}) \subset (\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H}$ 。因此， $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H}) = (\mathcal{F} \vee \mathcal{G}) \vee \mathcal{H}$ 。类似地，我们可以证明 $\mathfrak{S}(\mathcal{F} \cup \mathcal{G} \cup \mathcal{H}) = \mathcal{F} \vee (\mathcal{G} \vee \mathcal{H})$ 。

练习11.10.11 假设神经元为Sigmoid类型。我们有

$$\begin{aligned} X_1^{(1)} &= \sigma(w_{11}X_1^{(0)} + b_1) \\ X_2^{(1)} &= \sigma(w_{12}X_1^{(0)} + w_{22}X_2^{(0)} + b_2) \\ X_3^{(1)} &= \sigma(w_{23}X_2^{(0)} + b_3). \end{aligned}$$

由于 $X^{(1)}$ 取决于 $X^{(0)}$ ，我们得到 $\mathcal{I}^{(1)} \subset \mathcal{I}^{(0)}$ 。我们也可以根据 $X^{(1)}$ 求解 $X^{(0)}$ ，这意味着 $\mathcal{I}^{(0)} \subset \mathcal{I}^{(1)}$ 。

练习11.10.12 与练习11.10.11的解法类似。

练习11.10.13 如果 $\mathcal{I} = \mathcal{E}$ ，那么显然 $\mathcal{I} \setminus \mathcal{E} = \emptyset$ ，所以 $\mathcal{L} = \mathfrak{S}(\mathcal{I} \setminus \mathcal{E}) = \mathfrak{S}(\{\emptyset\}) = \{\emptyset, \Omega\}$ 。相反，如果 $\mathcal{L} = \{\emptyset, \Omega\}$ ，那么 $\mathfrak{S}(\mathcal{I} \setminus \mathcal{E}) = \{\emptyset, \Omega\}$ ，所以要么 $\mathcal{I} \setminus \mathcal{E} = \{\emptyset\}$ ，要么 $\mathcal{I} \setminus \mathcal{E} = \{\Omega\}$ ，要么 $\mathcal{I} \setminus \mathcal{E} = \{\emptyset, \Omega\}$ 。第一种情况意味着 $\mathcal{I} = \mathcal{E}$ 。其他两种情况会导致矛盾，因为 Ω 与 \mathcal{E} 相减且不能属于 $\mathcal{I} \setminus \mathcal{E}$ 。

练习11.10.14 用反证法假设 $\mathcal{H} \subseteq \mathcal{E} \subseteq \mathcal{F}$ 。利用命题11.4.5的 (a) 部分，第一个包含关系得出 $\mathcal{I} = \mathcal{L}$ 。类似地，将命题11.4.5的 (b) 部分应用于第二个包含关系意味着 $\mathcal{I} \neq \mathcal{L}$ 。得到的这两个陈述相互矛盾。

练习11.10.15 这是练习11.10.14的一个结果。

练习11.10.16 设 $\mathcal{U} \subset \mathcal{F}$ 为可恢复信息域的一个子域。那么 $\mathcal{I} \setminus \mathcal{F} \subset \mathcal{I} \setminus \mathcal{U}$ 。应用 \mathfrak{S} 算子得到 $\mathcal{I} = \mathfrak{S}(\mathcal{I} \setminus \mathcal{F}) \subset \mathfrak{S}(\mathcal{I} \setminus \mathcal{U})$ 。由此可知该包含关系实际上是等式。

第12章

练习12.13.1 使用不等式

$$H(X) = \int_e^\infty \frac{1}{x(\ln x)^2} (\ln x + 2 \ln \ln x) dx \geq \int_e^\infty \frac{1}{x \ln x} dx = +\infty.$$

练习12.13.2 (a) 令 $\mu = \mathcal{E}[X]$, $\sigma^2 = \text{Var}(X)$, 并考虑正态密度 $q(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ 。然后使用不等式

$$H(X) = - \int_{\mathbb{R}} p(x) \ln p(x) dx \leq - \int_{\mathbb{R}} p(x) \ln q(x) dx = \frac{1}{2} \ln(2\pi e \sigma^2).$$

(b) 由.....得出

$$-H(X) = \int_{\mathbb{R}} p(x) \ln p(x) dx \leq \ln M \int_{\mathbb{R}} p(x) dx = \ln M.$$

练习12.13.3 (a) 我们有

$$\begin{aligned} I(X, Y|Z) &= \int p(x, y, z) \ln p(x|z) dx dy dz + \int p(x, y, z) \ln p(y|z) dx dy dz \\ &\quad - \int p(x, y, z) \ln p(x, y|z) dx dy dz \\ &= \int p(x, y, z) \ln \frac{p(x, y, z)}{p(x|z)p(y|z)} dx dy dz \\ &= D_{KL}[p(x, y, z) || p(x|z)p(y|z)]. \end{aligned}$$

(b) 利用库尔贝克 - 莱布勒散度是非负的这一事实以及.....

$$D_{KL}[p(x, y, z) || p(x|z)p(y|z)] = 0$$

当且仅当 $p(x, y, z) = p(x|z)p(y|z)$, 即当 X 和 Y 相互独立时, 给定 Z 。

练习12.13.4 只需证明该系统不可能有超过两个不同的解。假设有两个不同的解, X_1 和 X_2 , 并考虑它们的差 $X_0 = X_1 - X_2 \neq 0$ 。那么 $AX_0 = AX_1 - AX_2 = b - b = 0$, 所以 X_0 是齐次系统 $AX = 0$ 的一个非零解。对于任何 $\alpha \in \mathbb{R}$, $X = \alpha X_0$ 也是齐次系统的一个解, 因为 $AX = \lambda AX_0 = 0$; 因此解空间

$$\mathcal{S} = \{X \in \mathbb{R}^n; AX = 0\}$$

至少是一维的。另一方面，一个著名的结果（基于同构定理）表明 $\dim \mathcal{S} = m - \text{rank}(A) = m - m = 0$ ，这导致了一个矛盾。由此可得 $X_0 = 0$ ，即 $X_1 = X_2$ 。

练习12.13.5 如果 $m = n$ ，那么 Q 是可逆的，那么

$$p(\mathbf{x}) = Q^{-1}p(\mathbf{y}) = Q^{-1} \text{softmax}(Q^{-1}\mathbf{h}).$$

练习12.13.6 利用自然对数函数是递增的这一性质，我们有 $\ln\left(\sum_j e^{u_j}\right) \geq \ln(e^{u_j}) = u_j$ 。因此，

$$\ln\left(\sum_j e^{u_j}\right) \geq \max_j u_j. \text{ 然后选择 } u_j = \left((Q^T Q)^{-1} Q^T h\right)_j.$$

练习12.13.8 对于 (a) 和 (b)，应用神经流形维度的公式。(c) 在 (b) 处的那个具有更大的容量，因为它有更多的参数。

练习12.13.9 (a) 增加，(b) 减少，(c) 减少。

练习12.13.10 (a) 对于单隐藏层网络，使用与关系(12.11.34)类似的式子，先应用于第一层和第二层，然后应用于第二层和第三层，我们分别得到

$$\mathfrak{S}(X) = \mathfrak{S}(U, X_{101}, \dots, X_{784}), \quad \mathfrak{S}(U) = \mathfrak{S}(\tilde{Y}, U_{11}, \dots, U_{100}).$$

拼接后，得到

$$\mathfrak{S}(X) = \mathfrak{S}(\tilde{Y}, U_{11}, \dots, U_{100}, X_{101}, \dots, X_{784}). \quad (2)$$

(b) 对于零隐藏层网络，将关系(12.11.34)应用于输入 X 和输出 Y ，我们有

$$\mathfrak{S}(X) = \mathfrak{S}(Y, X_{11}, \dots, X_{784}). \quad (3)$$

比较公式(3)和(2)，传递性意味着

$$\mathfrak{S}(Y, X_{11}, \dots, X_{100}, X_{101}, \dots, X_{784}) = \mathfrak{S}(\tilde{Y}, U_{11}, \dots, U_{100}, X_{101}, \dots, X_{784}). \quad (4)$$

练习12.13.11 输入变量 X 的最大熵约为每幅图像9.614比特。给定 U 时， X 的熵等于与 U 的每个条目对应的 X 的平均像素数

$$2^{H(X|U)} = \frac{784}{100},$$

这意味着 $H(X | U) = 2.9708$ 比特。那么 X 和 U 的互信息由下式给出

$$I(X, U) = H(X) - H(X|U) = 9.614 - 2.9708 = 6.643,$$

也就是说，每个图像约6.643比特的信息用于隐藏层的每个输入。根据之前对零隐藏层网络的分析，我们有 $I(X, \tilde{Y}) = 3.322$ 。这些导致了不等式

$$H(X) > I(X, U) > I(X, \tilde{Y}).$$

练习12.13.12 使用 $H(X_h | X_v) = H(X_h, X_v) - H(X_h)$ 和 $H(X_v | X_h) = H(X_v, X_h) - H(X_v)$ ，当且仅当 $H(X_h) = H(X_v)$ 时，可得出 $H(X_h | X_v) = H(X_v | X_h)$ 。

练习12.13.14 根据不变性性质，见命题12.7.5，我们有

$I(X'_h, X'_v) = I(X_h, X_v)$ 。使用公式 $I(X, Y) = H(X) + H(Y) - H(X, Y)$ ，可得当且仅当 $H(X'_h, X'_v) = H(X_h, X_v)$ 时

$$H(X'_h) + H(X'_v) = H(X_h) + H(X_v).$$

练习12.13.16 (a) 根据互信息的定义，我们有

$$\begin{aligned} I(X, Y, Z) &= H(X) + H(Y) + H(Z) - H(X, Y, Z) \\ &= [H(X) + H(Y) - H(X, Y)] + [H(X, Y) + H(Z) - H(X, Y, Z)] \\ &= I(X, Y) + I((X, Y), Z). \end{aligned}$$

(b) 类似地可以证明，对于任何 $1 \leq k \leq n$ ，我们有

$$I(X_1, \dots, X_n) = I(X_1, \dots, X_k) + I((X_1, \dots, X_k), (X_{k+1}, \dots, X_n)).$$

练习12.13.19 根据行列式的定义可得：

$$|\det W| = \left| \sum \epsilon_{i_1 \dots i_n} w_{1i_1} \cdots w_{ni_n} \right| \leq \sum |w_{1i_1}| \cdots |w_{ni_n}| \leq n! c^n.$$

练习12.13.20 由于

$$I(X_1, X_2) = H(X_1) + H(X_2) - H(X_1, X_2)$$

我们需要从右侧计算熵项。根据练习6.6.11，我们有

$$\begin{aligned} H(X_1) &= \frac{1}{2} \ln(2\pi) + \ln \sigma_1 + \frac{1}{2} \\ H(X_2) &= \frac{1}{2} \ln(2\pi) + \ln \sigma_2 + \frac{1}{2}. \end{aligned}$$

利用 (X_1, X_2) 的二元分布由下式给出

$$f(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^{\left\{ -\frac{1}{1-\rho^2} \left[\frac{(x_1-\mu_1)^2}{2\sigma_1^2} - \rho \frac{(x_1-\mu_1)(x_2-\mu_2)}{\sigma_1\sigma_2} + \frac{(x_2-\mu_2)^2}{2\sigma_2^2} \right] \right\}}$$

联合熵项可以计算为

$$\begin{aligned}
 H(X_1, X_2) &= - \iint f(x_1, x_2) \ln f(x_1, x_2) dx_1 dx_2 \\
 &= \ln(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}) \iint f(x_1, x_2) dx_1 dx_2 \\
 &\quad + \frac{1}{1-\rho^2} \left[\iint f(x_1, x_2) \frac{(x_1 - \mu_1)^2}{2\sigma_1^2} dx_1 dx_2 \right. \\
 &\quad - \rho \iint f(x_1, x_2) \frac{(x_1 - \mu_1)(x_2 - \mu_2)}{\sigma_1\sigma_2} dx_1 dx_2 \\
 &\quad \left. + \iint f(x_1, x_2) \frac{(x_2 - \mu_2)^2}{2\sigma_2^2} dx_1 dx_2 \right] \\
 &= \ln(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}) + \frac{1}{1-\rho^2} \left[\frac{1}{2} - \frac{\rho}{\sigma_1\sigma_2} \text{Cov}(X_1, X_2) + \frac{1}{2} \right] \\
 &= 1 + \ln(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}).
 \end{aligned}$$

因此，经过抵消后，我们得到

$$\begin{aligned}
 I(X_1, X_2) &= H(X_1) + H(X_2) - H(X_1, X_2) \\
 &= -\ln\sqrt{1-\rho^2} = -\frac{1}{2}\ln(1-\rho^2).
 \end{aligned}$$

互信息明确地取决于相关系数 ρ 。因此，至少在正态分布的情况下，相关性是变量之间互信息的一种表达。

练习12.13.21 (a) 应用命题12.7.5给出的不变性性质，我们有

$$\begin{aligned}
 I(X_1, X_2) &= I(F_{X_1}(X_1), F_{X_2}(X_2)) = I(U_1, U_2) \\
 &= H(U_1) + H(U_2) - H(U_1, U_2) \\
 &= \int_0^1 \int_0^1 c(u_1, u_2) \ln c(u_1, u_2) du_1 du_2,
 \end{aligned}$$

我们在此处使用了 $H(U_i) = 0$ ，因为 U_i 在 $[0, 1]$ 上均匀分布。(b) 如果 X_1 和 X_2 相互独立，那么Copula为 $C(u_1, u_2) = u_1u_2$ ，其密度为 $c(u_1, u_2) = 1$ 。因此，利用 (a)，我们得到 $I(X_1, X_2) = 0$ 。

第13章

练习13.8.1 相关神经流形的维度为

$$r = 784 \times 200 + 200 \times 100 + 100 \times 50 + 50 \times 10 + 350 = 182,300.$$

练习13.8.2 神经流形的维度是

$$\begin{aligned} f_N(k) &= d^{(0)} \frac{N}{k} + (k-1) \left(\frac{N}{k} \right)^2 + \frac{N}{k} d^{(L)} + N \\ &= (d^{(0)} + d^{(L)}) \frac{N}{k} + \frac{k-1}{k^2} N^2 + N. \end{aligned}$$

如果令 $u = 1/k$ ，上述公式可写为

$$\begin{aligned} \phi_N(u) &= (d^{(0)} + d^{(L)}) N u + (u - u^2) N^2 + N \\ &= -N^2 u^2 + N(d^{(0)} + d^{(L)} + N) u + N. \end{aligned}$$

这是一个关于 u 的二次函数，它在……处取得最大值

$$u = \frac{d^{(0)} + d^{(L)} + N}{2N}.$$

当 N 比 $d^{(0)} + d^{(L)}$ 大很多时，层数为 $k = 2$ 。

练习13.8.3 神经流形的维度是 $r = 784N + 10N = 794N$ 。解不等式 $794N \geq 550000$ ，我们得到 $N \geq 693$ 。因此，当 N 大于700时，网络表现出过拟合效应。

练习13.8.4 解不等式 $h^2 + 796h \geq 550,000$ 。

练习13.8.5 首先假设 u 和 v 是任意向量，因此它们可以分解为

$$u = \sum_{i=1}^r u_i \epsilon_i + u^N N, \quad v = \sum_{j=1}^r v_j \epsilon_j + v^N N,$$

其中 $\{\epsilon_1, \dots, \epsilon_r\}$ 是切空间 $T_y \mathcal{S}$ 中的一个基， N 是在 y 处 \mathcal{S} 的法向单位向量。它们的欧几里得内积为

$$\begin{aligned} \langle u, v \rangle &= \left\langle \sum_{i=1}^r u_i \epsilon_i, \sum_{j=1}^r v_j \epsilon_j \right\rangle + u^N v^N \\ &= \sum_{i=1}^r u_i v_j \langle \epsilon_i, \epsilon_j \rangle + u^N v^N \\ &= \sum_{i=1}^r u_i v_j g_{ij} + u^N v^N. \end{aligned}$$

如果向量 u 和 v 与 \mathcal{S} 相切，它们的法向部分为零，所以 $u^N = v^N = 0$ ，进而 $\langle u, v \rangle = \sum u_i v_j g_{ij}$ 。然后利用 u 和 v 正交当且仅当 $\langle u, v \rangle = 0$ 。
练习13.8.6 \mathbb{R}^n 中的仿射子空间是 \mathbb{R}^n 中的一个 k 平面，其中 $1 \leq k \leq n$ 。这个 k 平面中的任何测地线都是一条直线。

同样 $L = 0$ ，因为当一个向量场 U 关于向量场 V 求导时， U 和 V 都在给定的 k 平面内，方向导数 $\nabla_U V$ 属于同一个 k 平面，所以 $L(U, V) = (\nabla_U V)^\perp = 0$ 。

练习13.8.7 对于任意两个与 \mathcal{S} 相切的向量场 U 和 V ，我们有正交分解

$\nabla_U V = (\nabla_U V)^\parallel + (\nabla_U V)^\perp$ 。对于 \mathcal{S} 中的任意曲线 $c(s)$ ，取 $U = V = \dot{c}(s)$ 并得到运动方程

$$\nabla_{\dot{c}(s)} \dot{c}(s) = (\nabla_{\dot{c}(s)} \dot{c}(s))^\parallel + (\nabla_{\dot{c}(s)} \dot{c}(s))^\perp = D_{\dot{c}(s)} \dot{c}(s) + L(\dot{c}(s), \dot{c}(s)),$$

其中 ∇ 和 D 分别是 \mathcal{M} 和 \mathcal{S} 上的方向导数，并且 L 是 \mathcal{S} 关于 \mathcal{M} 的第二基本形式。我们利用测地线是加速度为零的曲线这一性质。

(a) \Rightarrow (b) 假设 $L = 0$ 。那么 $\nabla_{\dot{c}(s)} \dot{c}(s) = D_{\dot{c}(s)} \dot{c}(s)$ 。如果 $c(s)$ 是 \mathcal{S} 中的一条测地线，那么 $D_{\dot{c}(s)} \dot{c}(s) = 0$ 。那么 $\nabla_{\dot{c}(s)} \dot{c}(s) = 0$ ，所以 $c(s)$ 是 \mathcal{M} 中的一条测地线。

(b) \Rightarrow (a) 如果 $D_{\dot{c}(s)} \dot{c}(s) = 0$ ，那么 $\nabla_{\dot{c}(s)} \dot{c}(s) = 0$ ，因此对于 \mathcal{S} 中的任何测地线 $c(s)$ ，有 $L(\dot{c}, \dot{c}) = 0$ 。由于对于任何给定向量 $v \in T_p \mathcal{S}$ ，存在一条初始速度为 $\dot{c}(0) = v$ 的测地线，所以对于所有 $v \in T_p \mathcal{S}$ ，当 p 在 \mathcal{S} 中任意取值时，有 $L(v, v) = 0$ 。然后使用极化公式

$$L(v, w) = \frac{1}{2}[L(v + w, v + w) - L(v, v) - L(w, w)]$$

得到 $L = 0$ 。

练习13.8.8 (a) 这是由积分中的变量替换公式得出的。根据链式法则， $\gamma'(t) = \dot{c}(s)\phi'(t)$ ，所以取范数，并利用 ϕ 是递增的，我们得到 $\|\gamma'(t)\| = \|\dot{c}(s)\| \phi'(t)$ 。然后

$$\begin{aligned} L(\gamma) &= \int_c^d \|\gamma'(t)\| dt = \int_c^d \|\dot{c}(\phi(t))\| \underbrace{\phi'(t) dt}_{=ds} \\ &= \int_a^b \|\dot{c}(s)\| ds = L(c). \end{aligned}$$

对能量进行类似的计算。

(b) 应用柯西不等式的积分形式

$$\left(\int_a^b f(x)g(x) dx \right)^2 \leq \int_a^b f(x)^2 dx \int_a^b g(x)^2 dx$$

对于 $f = \|\dot{c}\|$ 和 $g = 1$ 。对于匀速曲线，即 $\|\dot{c}(s)\| =$ 为常数时，可得到该恒等式。

(c) 在局部上，长度最小化曲线和能量最小化曲线是等价的。测地线也是能量最小化曲线；沿着测地线加速度为零，因此速度大小是恒定的。

练习13.8.9 (a) 设 $(r(t), z(t))$ 为一条满足 $r(t) > 0$ 的平面曲线。若该曲线绕 z 轴旋转，可得一个旋转曲面。其参数化表示为

$$\phi(t, s) = (r(t) \cos s, r(t) \sin s, z(t)), \quad 0 \leq s \leq 2\pi,$$

其中 t 衡量给定曲线上的位置， s 衡量旋转角度。可以证明（例如，见[85]这本书），旋转曲面的第二基本形式的系数由下式给出

$$L_{ij} = \frac{1}{\sqrt{\dot{r}^2 + \dot{z}^2}} \begin{pmatrix} \dot{r}\ddot{z} - \dot{z}\ddot{r} & 0 \\ 0 & r\dot{z} \end{pmatrix}.$$

我们将把这个应用到单位球面的两种参数化表示上。

(i) 选取 $r(t) = \cos t, z(t) = \sin t, -\pi/2 < t < \pi/2$ 并得到球面的参数化表示

$$\phi(t, s) = (\cos t \cos s, \cos t \sin s, \sin t), \quad 0 \leq s \leq 2\pi, -\pi/2 < t < \pi/2.$$

使用前面的公式可得 $L_{ij} = \begin{pmatrix} 1 & 0 \\ 0 & \cos^2 t \end{pmatrix}$ 。 L 的范数是其最大特征值，所以 $\|L\| = 1$ 。

(ii) 设 $r(t) = \sqrt{1-t^2}, z(t) = t, -1 < t < 1$ 。参数化表示为

$$\phi(t, s) = (\sqrt{1-t^2} \cos s, \sqrt{1-t^2} \sin s, t), \quad 0 \leq s \leq 2\pi, -1 < t < 1.$$

计算表明 $L_{ij} = \begin{pmatrix} \frac{1}{(1-t^2)^2} & 0 \\ 0 & 1 \end{pmatrix}$ 。其范数同样是 $\|L\| = 1$ 。

我们注意到，即使系数 L_{ij} 依赖于参数化表示，但范数 $\|L\|$ 并不依赖。

练习13.8.10 sigmoid神经元的输出是 $y_1 = \sigma(w_1x + b_1)$ 和 $y_2 = \sigma(w_2x + b_2)$ 。它们的组合有输出

$$y = \lambda y_1 + (1-\lambda)y_2 = \lambda\sigma(w_1x + b_1) + (1-\lambda)\sigma(w_2x + b_2).$$

由于 $(w_1, w_2, b_1, b_2, \lambda) \in \mathbb{R}^5$ ，相关的神经流形维度为5。

练习13.8.11 当一个神经元被去除时，所有输入和输出权重以及它的偏差都等于零。因此，神经流形的维度会降低。网络中神经元数量减少会产生一个非线性程度较低的结果。因此，神经流形往往具有较小的嵌入曲率。两者都能减少过拟合，如果去除太多神经元，可能会导致欠拟合。

练习13.8.12 所有公式都是乘积法则和二阶导数对称性的应用。

第14章

练习14.13.1 (a) 考虑展开式 $e_i = \sum_{j=1}^n \alpha_{ij} v_j$ 。那么

$$G_{ik} = e_i^T G e_k = \left(\sum_{j=1}^n \alpha_{ij} v_j \right)^T G \left(\sum_{r=1}^n \alpha_{kr} v_r \right) = \sum_{j,r} \alpha_{ij} \alpha_{kr} \underbrace{v_j^T G v_r}_{=0} = 0,$$

所以 $G = \mathbb{O}_n$ 。 (b) 由线性性质、(a)部分以及考虑 $G = A - B$ 可得

练习14.13.2 设 $Y_1 = u^T X$ 和 $Y_2 = v^T X$ 。联合累积分布函数为

$$\begin{aligned} F_{Y_1 Y_2}(a, b) &= P(Y_1 \leq a, Y_2 \leq b) = P(u^T X \leq a, v^T X \leq b) \\ &= P(u_1 X_1 + u_2 X_2 \leq a, v_1 X_1 + v_2 X_2 \leq b) \\ &= \iint_{\{u^T x \leq a\} \cap \{v^T x \leq b\}} p_{X_1 X_2}(x_1, x_2) dx_1 dx_2. \end{aligned}$$

我们计算这个比率

$$\frac{F_{Y_1 Y_2}(a + \Delta a, b + \Delta b) - F_{Y_1 Y_2}(a, b)}{\Delta a \Delta b} = \frac{1}{\Delta a \Delta b} \iint_{D_{ab}} p_{X_1 X_2}(x_1, x_2) dx_1 dx_2,$$

其中 $D_{ab} = \{a \leq u^T x \leq a + \Delta a\} \cap \{b \leq v^T x \leq b + \Delta b\}$ 是一个矩形区域，其边的方向平行于 u 和 v 。利用 $R^{-1}(D_{ab}) = [a, a + \Delta a, b, b + \Delta b]$ ，进行变量替换，使用富比尼定理以及 X 是旋转不变的这一事实，我们得到

$$\begin{aligned} &\frac{1}{\Delta a \Delta b} \iint_{D_{ab}} p_{X_1 X_2}(x_1, x_2) dx_1 dx_2 \\ &= \frac{1}{\Delta a \Delta b} \iint_{[a, a + \Delta a] \times [b, b + \Delta b]} p_{X_1 X_2}(R(x_1, x_2)) \underbrace{|\det R|}_{=1} dx_1 dx_2 \\ &= \frac{1}{\Delta a} \int_a^{a + \Delta a} p_{X_1}(x_1) dx_1 \frac{1}{\Delta b} \int_b^{b + \Delta b} p_{X_2}(x_2) dx_2. \end{aligned}$$

取极限 $\Delta a \rightarrow 0$ 和 $\Delta b \rightarrow 0$ ，我们得到

$$\frac{\partial^2}{\partial a \partial b} F_{Y_1 Y_2}(a, b) = p_{X_1}(a)p_{X_2}(b)$$

或者 $f_{Y_1 Y_2}(a, b) = f_1(a)f_2(b)$ ，这意味着 Y_1 和 Y_2 是独立的。

(b) $Y_1 = u^T X$ 服从均值为零且方差为 $\text{Var}(Y_1) = |u| = 1$ 的正态分布。因此， $u^T X, v^T X \sim \mathcal{N}(0, 1)$ 。

(c) 这可由三角函数圆推出。我们取 $\phi = \arg u$ ，由于 u 和 v 正交，所以 $\arg v = \phi + \pi/2$ 。然后 $v = (\cos(\phi + \pi/2), \sin(\phi + \pi/2)) = (-\sin \phi, \cos \phi)$ 。

练习14.13.3 (a) 我们有

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \begin{pmatrix} u_1 & u_2 \\ v_1 & v_2 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = MX.$$

根据练习14.13.2， M 是一个旋转矩阵，它保持独立性。(b) 如果 $Y_1 = u^T X$ 和 $Y_2 = v^T X$ 是独立的，那么对于非零常数 α, β ， $Y'_1 = \alpha u^T X$ 和 $Y'_2 = \beta v^T X$ 也是独立的。

练习14.13.4 我们按第一行展开行列式 $D(\lambda) = \det(M - \lambda I_n)$ ，然后按第一列展开每个 $(n-1)$ 子式，得到

$$\begin{aligned} D(\lambda) &= -\lambda \begin{vmatrix} -\lambda & 0 & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & -\lambda & 0 \\ 0 & \cdots & 0 & -\lambda \end{vmatrix} - a_1 \begin{vmatrix} b_1 & 0 & \cdots & 0 \\ b_2 & -\lambda & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ b_n & \cdots & 0 & -\lambda \end{vmatrix} \\ &\quad + a_2 \begin{vmatrix} b_1 & -\lambda & \cdots & 0 \\ b_2 & 0 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ b_n & \cdots & 0 & -\lambda \end{vmatrix} + \cdots + (-1)^{n+1} a_n \begin{vmatrix} b_1 & -\lambda & \cdots & 0 \\ b_2 & 0 & -\lambda & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ b_n & \cdots & 0 & -\lambda \end{vmatrix} \\ &= -\lambda(-\lambda)^n - a_1 b_1(-\lambda)^{n-1} - a_2 b_2(-\lambda)^{n-1} - \cdots + (-1)^{2n+1} a_n b_n(-\lambda)^{n-1} \\ &= (-\lambda)^{n-1} [\lambda^2 - \sum a_i b_i] = (-\lambda)^{n-1} [\lambda^2 - a^T b]. \end{aligned}$$

求解方程 $D(\lambda) = 0$ ，我们得到所需的解。

练习14.13.5 (a) 使用 $\sigma'(x) = \sigma(x)(1 - \sigma(x))$ 。(c) 令 $\|\phi'\| = \sup_x |\phi'(x)|$ 。那么

$$\begin{aligned} 0 \leq g_{00} &= \mathbb{E}^{P_X} [\phi'(w^T X + b)^2] \leq \|\phi'\|^2 \mathbb{E}^{P_X}[1] = \|\phi'\|^2 \\ g_{0k} &= \int x_k \phi'(w^T x + b)^2 p(x) dx \\ &= \int x_k \sqrt{p(x)} \phi'(w^T x + b)^2 \sqrt{p(x)} dx. \end{aligned}$$

那么柯西 - 施瓦茨不等式意味着

$$\begin{aligned} g_{0k}^2 &\leq \int (x_k \sqrt{p(x)})^2 dx \int (\phi'(w^T x + b)^2 \sqrt{p(x)})^2 dx \\ &= \int x_k^2 p(x) dx \int \phi'(w^T x + b)^2 p(x) dx \leq \|\phi'\|^2 \int x_k^2 p(x) dx \\ &= \|\phi'\|^2 \mathbb{E}[X_k^2]. \end{aligned}$$

类似地，我们有

$$\begin{aligned} g_{jk}^2 &\leq \int (x_j x_k \sqrt{p(x)})^2 dx \int (\phi'(w^T x + b)^2 \sqrt{p(x)})^2 dx \\ &= \int x_j^2 x_k^2 p(x) dx \int \phi'(w^T x + b)^2 p(x) dx \leq \|\phi'\|^2 \int x_j^2 x_k^2 p(x) dx \\ &= \|\phi'\|^2 \mathbb{E}[X_j^2 X_k^2]. \end{aligned}$$

为了得到 (b) 部分，使用 $\|\sigma'\| = \frac{1}{4}$ 。

练习14.13.6

$$\begin{aligned} g_{00} &= \mathbb{E}[\phi'(w^T X + b)^2] = \int_{I_n} \phi'(w^T X + b)^2 dx_1 \cdots dx_n \\ &= \frac{1}{w_1 \cdots w_n} \int_0^{w_1} \cdots \int_0^{w_n} \phi'(u_1 + \cdots + u_n + b)^2 du_1 \cdots du_n. \end{aligned}$$

类似的计算适用于其他系数。

练习14.13.8

$$g_{\alpha_j \alpha_k} = \mathbb{E}[\phi(w_j X + b_j) \phi(w_k X + b_k)] = \frac{1}{\sqrt{2\pi}} \int \phi(w_j x + b_j) \phi(w_k x + b_k) e^{-\frac{1}{2}x^2} dx.$$

其他系数情况类似。

练习14.13.10 对关系式取对数

$$p_{X_1, X_2}(x_1, x_2; \theta) = p_{X_1}(x_1; \theta) p_{X_2|X_1}(x_2|x_1; \theta)$$

我们得到 $\ell_{X_1, X_2}(\theta) = \ell_{X_1}(\theta) + \ell_{X_2|X_1}(\theta)$ 。求导，然后乘以 $p_{X_1, X_2}(x_1, x_2; \theta)$ 并在 x_1 和 x_2 上积分，我们得到

$$\mathbb{E}^{P_{X_1 X_2}} [\partial_{\theta_i \theta_j} \ell_{X_1, X_2}(\theta)] = \mathbb{E}^{P_{X_1}} [\partial_{\theta_i \theta_j} \ell_{X_1}(\theta)] + \mathbb{E}^{P_{X_2|X_1}} [\partial_{\theta_i \theta_j} \ell_{X_2|X_1}(\theta)],$$

这里我们使用了 $\int p_{X_2|X_1}(x_2 | x_1; \theta) dx_2 = 1$ 。这意味着

$$g_{ij}(X_1, X_2; \theta) = g_{ij}(X_1; \theta) + g_{ij}(X_2|X_1; \theta).$$

练习14.13.11 (a) 这可由欧几里得标量积的定义推出。

(b) 根据梯度的定义，我们有

$$\begin{aligned} g(\nabla_g f, X) &= \sum_{i,j} g_{ij} (\nabla_g f)^i X^j = \sum_{i,j,k} g_{ij} g^{ik} \frac{\partial f}{\partial \theta_k} X^j \\ &= \sum_{j,k} \delta_j^k \frac{\partial f}{\partial \theta_k} X^j = \sum_{k=1}^N X^k \frac{\partial f}{\partial \theta_k} \end{aligned}$$

(c) 它由 (a) 和 (b) 推出。

练习14.13.12 (a) 我们有

$$\begin{aligned} \|\nabla_g f\|_g^2 &= g(\nabla_g f, \nabla_g f) = \sum_{i,j} g_{ij} (\nabla_g f)^i (\nabla_g f)^j \\ &= \sum_{i,j,k} g_{ij} g^{ik} \frac{\partial f}{\partial \theta_k} g^{jp} \frac{\partial f}{\partial \theta_p} = \sum_{k,p} g^{kp} \frac{\partial f}{\partial \theta_k} \frac{\partial f}{\partial \theta_p} \\ &= (\nabla_{Eu} f)^T g^{-1}(\theta) \nabla_{Eu} f. \end{aligned}$$

(b) 类似地，

$$\|\nabla_{Eu} f\|_g^2 = \sum_{i,j} g_{ij} (\nabla_{Eu} f)^i (\nabla_{Eu} f)^j = (\nabla_{Eu} f)^T g(\theta) \nabla_{Eu} f.$$

(c) 设 $(\nabla_{Eu} f)(p) = 0$ 。由(a)部分我们得到 $\|(\nabla_g f)(p)\|_g^2 = 0$ ，并且由于 g 是非退化的，所以得出 $(\nabla_g f)(p) = 0$ 。反之，现在假设 $(\nabla_g f)(p) = 0$ 。使用公式 $\frac{\partial f}{\partial \theta_p} = \sum_j g_{jp} (\nabla_g f)^j$ 可得 $\frac{\partial f}{\partial \theta_p}(p) = 0$ ，因此 $(\nabla_{Eu} f)(p) = 0$ 。

练习14.13.13 (a) $\mathbb{E}[\hat{\mu}] = \frac{1}{n} \sum_{j=1}^n \mathbb{E}[X_j] = \frac{n\mu}{n} = \mu$ 。

(b) 由于 $p_\mu(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2}}$ ，那么 $\partial_\mu^2 \ln p_\theta = -1$ ，所以

$I(\mu) = -\mathbb{E}[\partial_\mu^2 \ln p_\theta] = 1$ 。 (c) 包含在 n 个独立随机变量 X_1, \dots, X_n 中的费希尔信息是其总和

$$I(X_1, \dots, X_n; \theta) = \sum_{j=1}^n I(X_i; \theta) = n.$$

然后 $\text{Var}(\hat{\mu}) = \frac{1}{n^2} \sum_j \text{Var}(X_i) = \frac{1}{n^2} n \text{Var}(X_1) = \frac{1}{n} = \frac{1}{I(X_1, \dots, X_n; \theta)}$ ，即

达到了克拉美罗界。

练习14.13.14 由于我们知道 $\lambda = \mathbb{E}[X]$ ，自然会考虑估计量 $\hat{\lambda} = \frac{1}{n} \sum_{j=1}^n X_j$ ，

其中 $X_j \sim \text{Pois}(\lambda)$ 是独立随机变量。 n 个独立的泊松分布随机变量 X_1, \dots, X_n 中包含的费希尔信息为

$$\begin{aligned}
I(X_1, \dots, X_n; \lambda) &= \sum_{j=1}^n I(X_j; \lambda) = nI(X; \lambda) = -n\mathbb{E}[\partial_\lambda^2 \ln p_\lambda(X)] \\
&= -n \sum_{k \geq 0} \frac{\lambda^k e^{-\lambda}}{k!} \partial_\lambda^2(k \ln \lambda - \lambda) = ne^{-\lambda} \sum_{k \geq 1} \frac{\lambda^{k-1}}{(k-1)!} \\
&= \frac{n}{\lambda} e^{-\lambda} e^\lambda = \frac{n}{\lambda}.
\end{aligned}$$

估计量 $\hat{\lambda}$ 是无偏的，其方差为

$$Var(\hat{\lambda}) = \frac{1}{n^2} n Var(X_1) = \frac{\lambda}{n} = \frac{1}{I(X_1, \dots, X_n; \lambda)}.$$

练习14.13.15 由于 $\bar{X} \sim \mathcal{N}(\mu, \frac{1}{N})$ ，那么 $p_{\bar{X}}(u; \mu) = \frac{\sqrt{N}}{\sqrt{2\pi}} e^{-\frac{N}{2}(u-\mu)^2}$ ，所以对数似然函数变为

$$\ell_{\bar{X}}(\mu) = \ln \frac{\sqrt{N}}{\sqrt{2\pi}} - \frac{N}{2}(u - \mu)^2.$$

得分函数的平方是 $(\partial_\mu \ell_{\bar{X}}(\mu))^2 = N^2 (\bar{X} - \mu)^2$ 。因此，由 \bar{X} 诱导的费希尔信息是

$$\begin{aligned}
I(\bar{X}) &= \mathbb{E}^{P_{\bar{X}}}[(\partial_\mu \ell_{\bar{X}}(\mu))^2] = \mathbb{E}^{P_{\bar{X}}}[N^2(\bar{X} - \mu)^2] \\
&= N^2 Var(\bar{X}) = N^2 \frac{1}{N} = N.
\end{aligned}$$

从练习14.13.13的第 (b) 部分，我们有 $I(X) = 1$ 。因此， $I(\bar{X}) = NI(X)$ 。

练习14.13.16 (a) 由于 $p_{X_1, X_2}(x_1, x_2; \theta) = p_{X_1}(x_1; \theta)p_{X_2}(x_2; \theta)$ ，对其取对数函数，我们得到 $\ell_{X_1, X_2}(\theta) = \ell_{X_1}(\theta) + \ell_{X_2}(\theta)$ 。求导可得

$$\partial_{\theta_i \theta_j} \ell_{X_1, X_2}(\theta) = \partial_{\theta_i} \ell_{X_1}(\theta) + \partial_{\theta_j} \ell_{X_2}(\theta).$$

然后乘以 $p_{X_1, X_2}(x_1, x_2; \theta)$ 并对 x_1 和 x_2 进行积分，使用 $\int p_{X_1, X_2}(x_1, x_2; \theta) dx_1 = p_{X_2}(x_2; \theta)$ 和 $\int p_{X_1, X_2}(x_1, x_2; \theta) dx_2 = p_{X_1}(x_1; \theta)$ ，得到

$$\mathbb{E}^{P_{X_1 X_2}}[\partial_{\theta_i \theta_j} \ell_{X_1, X_2}(\theta)] = \mathbb{E}^{P_{X_1}}[\partial_{\theta_i} \ell_{X_1}(\theta)] + \mathbb{E}^{P_{X_2}}[\partial_{\theta_j} \ell_{X_2}(\theta)]$$

这等同于

$$g_{ij}(X_1, X_2; \theta) = g_{ij}(X_1; \theta) + g_{ij}(X_2; \theta).$$

(b) 如果 X_1, \dots, X_N 是独立随机变量, 那么

$$g(X_1, \dots, X_N; \theta) = \sum_{j=1}^N g(X_j; \theta).$$

(c) 如果 $X_1, \dots, X_N \sim X$ 是独立同分布的, 根据 (b) 部分我们有

$$g(X_1, \dots, X_N; \theta) = \sum_{j=1}^N g(X_j; \theta) = Ng(X; \theta).$$

因此, 逆是

$$g(X_1, \dots, X_N; \theta)^{-1} = \frac{1}{N}g(X; \theta)^{-1}.$$

(d) 如果对于 $N \rightarrow \infty$, $\hat{\theta}(N)$ 是 θ 的渐近有效估计量

$$\text{Cov}(\hat{\theta}(N)) = \mathbb{E}[(\hat{\theta}(N) - \theta)(\hat{\theta}(N) - \theta)^T] \sim g(X_1, \dots, X_N; \theta)^{-1} = \frac{1}{N}g(X; \theta)^{-1},$$

乘以 N 并取极限, 我们得到

$$\lim_{N \rightarrow \infty} N\mathbb{E}[(\hat{\theta}(N) - \theta)(\hat{\theta}(N) - \theta)^T] = g^{-1}(\theta).$$

第15章

练习15.6.1 (a) 对于任何固定索引 k , 我们有 $\bigcap_i \mathcal{C}_i \subset \mathcal{C}_k$ 。利用 \mathfrak{S} 的单

调性性质可得 $\mathfrak{S}\left(\bigcap_i \mathcal{C}_i\right) \subset \mathfrak{S}(\mathcal{C}_k)$ 。由于 k 是任意的, 所以可得

$\mathfrak{S}\left(\bigcap_i \mathcal{C}_i\right) \subset \bigcap_k \mathfrak{S}(\mathcal{C}_k)$, 这就是所需的关系。

(b) 证明过程类似, 从 $\bigcup_i \mathcal{C}_i \supset \mathcal{C}_k$ 开始。

练习15.6.2 将参与者集合 X 作为输入数据输入到神经网络中。每一轮之后剩下的参与者代表网络的各层。有 $n - 1$ 个隐藏层, 并且每个隐藏层都实现一种最大池化方法, 通过该方法单元数量减少到一半。

练习15.6.3 为了简单起见, 我们假设输入已被划分为两个类别 $\{a_1, a_2, a_3\}, \{b_1, b_2, b_3\}$, 每个类别有三个元素。

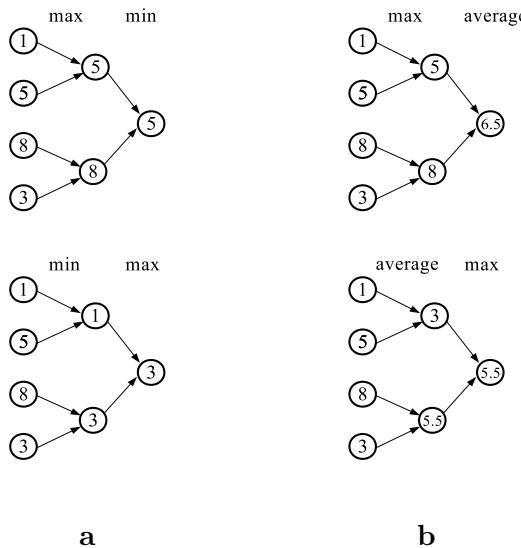


图3：对于练习15.6.5：a. 将最小池化层与最大池化层进行切换会产生不同的输出。b. 将平均池化层与最大池化层进行切换会产生不同的输出。

(a) 由以下得出

$$\max\{\max\{a_1, a_2, a_3\}, \max\{b_1, b_2, b_3\}\} = \max\{a_1, a_2, a_3, b_1, b_2, b_3\}.$$

(c) 平均值的平均数仅计为一个平均数

$$\frac{1}{2}\left(\frac{a_1 + a_2 + a_3}{3} + \frac{b_1 + b_2 + b_3}{3}\right) = \frac{1}{6}(a_1 + a_2 + a_3 + b_1 + b_2 + b_3).$$

(c) 使用关系

$$\min\{\min\{a_1, a_2, a_3\}, \min\{b_1, b_2, b_3\}\} = \min\{a_1, a_2, a_3, b_1, b_2, b_3\}.$$

练习15.6.5 由图3 a、b中提供的反例得出。

第16章

练习16.9.1 滑动内核并进行卷积以获得 $\begin{pmatrix} -3 & 1 \\ 3 & -5 \end{pmatrix}$ 。

练习16.9.2 (a) 定义 $(T_a \circ y)_i = y_{i-a}$ 。那么一维的等变性属性变为 $\mathcal{C}(T_a \circ y) = T_a \circ \mathcal{C}(y)$ 。证明与二维的证明非常相似。(b) 是的。

练习16.9.3 (a) 进行卷积运算后，我们得到

$$\begin{pmatrix} a_{i-1,j-1} & a_{i-1,j} & a_{i-1,j+1} \\ a_{i,j-1} & a_{i,j} & a_{i,j+1} \\ a_{i+1,j-1} & a_{i+1,j} & a_{i+1,j+1} \end{pmatrix} * K$$

$$= (a_{i-1,j-1} - a_{i+1,j-1}) + 2(a_{i-1,j} - a_{i+1,j}) + (a_{i-1,j+1} - a_{i+1,j+1}),$$

其效果是减去第 $(i-1)$ 行和第 $(i+1)$ 行。

(b) 类似地，其效果是减去列。

练习16.9.4 (a) 用这个内核进行卷积会在二维中产生一个移动平均值，由此得到的均匀模糊由

$$\frac{1}{9}(a_{11} + a_{12} + a_{13} + a_{21} + a_{22} + a_{23} + a_{31} + a_{32} + a_{33}).$$

(b) 这是一种更强调中心、然后是中间部分、最后是角落的模糊。所有权重之和等于1。

练习16.9.5 进行卷积运算后，我们得到

$$\begin{pmatrix} a_{i-1,j-1} & a_{i-1,j} & a_{i-1,j+1} \\ a_{i,j-1} & a_{i,j} & a_{i,j+1} \\ a_{i+1,j-1} & a_{i+1,j} & a_{i+1,j+1} \end{pmatrix} * \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

$$= 5a_{ij} - (a_{i-1,j} + a_{i,j-1} + a_{i,j+1} + a_{i+1,j}).$$

这种对比度源于中心像素激活值的5倍与垂直和水平相邻像素激活值之间的差异。

练习16.9.7 内核是 $K = \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ 。一个 3×3 矩阵与 K 的卷积是

$a_{i,j-1} - a_{i,j}$ 。我们也可以考虑等效内核

$$K = \begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix}.$$

练习16.9.8 (a) 利用卷积的等变性和池化对平移的不变性，我们有

$$\mathcal{P} \circ \mathcal{C}(T_{a,b} \circ y) = \mathcal{P} \circ T_{a,b}(\mathcal{C} \circ y) = \mathcal{P} \circ \mathcal{C}(y).$$

(b) 以相反顺序使用的类似属性。

练习16.9.9 仅在全连接层。其他层已经足够稀疏，随机失活不会带来实质性的改进。

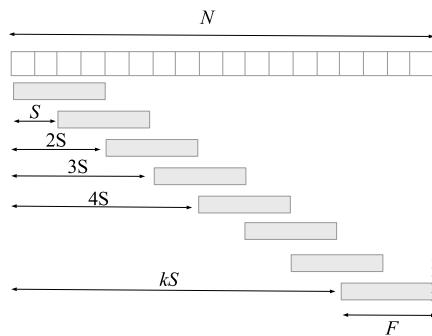


图4：用于练习16.9.11。

练习16.9.10 由于卷积神经网络（CNN）是稀疏的，其权重较少，因此与全连接神经网络相比容量较小。

练习16.9.11 (i) 我们有 $N - F = kS$ ，其中 k 是内核从一端移动到另一端的步数。输出的大小比 k 大1，即 $O = k + 1$ 。求解 k 得到 $O = \frac{N-F}{S} + 1$ 。在填充的情况下，我们在维度 N 上添加 $2P$ （每侧一个 P ），并得到第二个所需公式。(ii) 我们应用与 (i) 部分相同的过程。见图4。

练习16.9.12 (i) 由于 H 是 G 的子群，我们有 $e \in H$ ，然后 $x = xe \in xH$ 。(ii) 假设 $xH \cap yH = \emptyset$ ，我们将使用双重包含来证明 $xH = yH$ 。存在 $z \in xH \cap yH$ 。所以，存在 $h_1, h_2 \in H$ 使得 $z = xh_1 = yh_2$ 。这意味着 $x = yh_2h_1^{-1}$ 。由于 $h_2h_1^{-1} \in H$ ，所以 $x \in yH$ ，由此我们得到 $xH \subseteq yH$ 。类似地， $y = xh_1h_2^{-1} \in xH$ ，然后 $yH \subseteq xH$ 。(iii) 由 $\varphi(h) = xh$ 定义的函数 $\varphi : H \rightarrow xH$ 是一个双射，因此 H 和 xH 具有相同数量的元素。(iv)(a) 由于 $x^{-1}x = e \in H$ ，那么 $x \sim x$ 。(b) 如果 $x \sim y$ ，那么 $y^{-1}x \in H$ ，其逆 $(y^{-1}x)^{-1} \in H$ ，变为 $x^{-1}y \in H$ ，即 $y \sim x$ 。(c) 如果 $x \sim y$ 且 $y \sim z$ ，那么 $y^{-1}x \in H$ 且 $z^{-1}y \in H$ 。由于 H 对乘法封闭，我们有 $z^{-1}x = (z^{-1}y)(y^{-1}x) \in H$ ，这意味着 $x \sim z$ 。

第17章

练习17.10.1 (a) 可以通过双向包含来证明。包含关系“ \subset ”可通过对包含关系应用 \mathfrak{S} 运算符得出

$$\mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3 \subset \mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2) \cup \mathcal{G}_3.$$

包含关系“ \supset ”可由以下内容得出。首先，由于 $\mathcal{G}_3 \subset \mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3$ ，那么 $\mathcal{G}_3 \subset \mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3)$ 。类似地，由 $\mathcal{G}_1 \cup \mathcal{G}_2 \subset \mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3$ 可得出

$$\mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2) \subset \mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3).$$

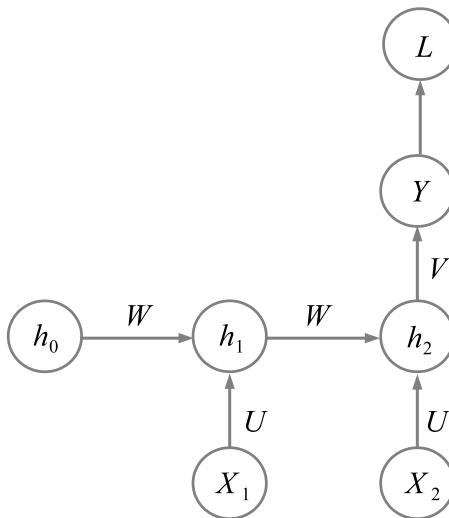


图5：用于练习17.10.3的“2对1”循环神经网络。

从最后两个包含关系我们得到

$$\mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2) \cup \mathcal{G}_3 \subset \mathfrak{S}(\mathcal{G}_1 \cup \mathcal{G}_2 \cup \mathcal{G}_3).$$

在两边应用 \mathfrak{S} 算子，我们得到所需的包含关系。(b) 我们可以通过归纳法证明

$$\mathfrak{S}(\mathcal{G}_1 \cup \dots \cup \mathcal{G}_n) = \mathfrak{S}\left(\mathfrak{S}(\mathcal{G}_1 \cup \dots \cup \mathcal{G}_{n-1}) \cup \mathcal{G}_n\right).$$

$$\mathfrak{S}(\mathcal{G}_1 \cup \dots \cup \mathcal{G}_n) = \mathfrak{S}\left(\mathfrak{S}(\mathcal{G}_1 \cup \dots \cup \mathcal{G}_{n-k}) \cup \mathcal{G}_{n-k+1} \cup \dots \cup \mathcal{G}_n\right).$$

练习17.10.2 (a) W 的特征值是 $\lambda_1 = 1/5$ 和 $\lambda_2 = -1/2$ 。由于两者都小于1，应用命题G.1.2。(b) 由于 $\rho(W) < 1$ ，应用命题G.1.2。

练习17.10.3提示：遵循17.5节中针对图5给出的示例。

练习17.10.4从 $Y = Vh_2 + c$ ，我们得到 $\mathcal{E} \subset \mathcal{H}_2$ 。从

$$\begin{aligned} h_2 &= \tanh(Wh_1 + UX_2 + b) \\ h_1 &= \tanh(Wh_0 + UX_1 + b) \end{aligned}$$

我们可以证明 $\mathcal{H}_2 \subset \mathfrak{S}(\mathcal{I}_1, \mathcal{I}_2)$ 。然后 $\mathcal{E} \subset \mathfrak{S}(\mathcal{I}_1, \mathcal{I}_2)$ 。所以，如果 $\mathfrak{S}(Z) \subset \mathcal{E}$ 那么 $\mathfrak{S}(Z) \subset \mathfrak{S}(\mathcal{I}_1 \cup \mathcal{I}_2)$ 。因此，(a) 成立。

练习17.10.5 转移函数 f 是一个压缩映射。那么 $\lim_{n \rightarrow \infty} h_n$ 是 f 的不动点，即 $\tanh(\theta c) = c$ 。由此可得 $c = 0$ 。

练习17.10.6 由于 $f'(x; \theta) = \theta\sigma'(\theta x) = \theta\sigma(\theta x)(1 - \sigma'(\theta x)) \leq \theta/4 < 1$ ，转移函数 f 仍然是一个压缩映射。不动点满足 $\sigma(\theta c) = c$ 。它是唯一的，但无法用封闭形式确定。

练习17.10.7 如果 θ 很大，那么图 $y = \sin(\theta x)$ 和 $y = x$ 会相交不止一次，每个交点都代表一个固定点。根据 h_0 的情况，系统可能会稳定在这些点中的一个上。

第18章

练习18.11.1 (a) 矩形规则很容易满足，因为如果一个矩形的三个角具有有理坐标，那么第四个角也具有相同的性质。

(b) 不，不是。因为 $(1/\sqrt{2}, 1/\sqrt{2}) \notin (I_1 \times I_1) \cap (\mathbb{Q} \times \mathbb{Q})$, \mathcal{S} 不是自反的。

练习18.11.2 “ \Rightarrow ”用反证法，假设存在两个不同的包含这些点的 $(k-1)$ -超平面。那么这些点属于超平面的交集，而这个交集将是一个维度严格小于 $k-1$ 的超平面，这与这些点处于一般位置这一事实相矛盾。因此，存在唯一的包含这些点的 $(k-1)$ -超平面。

“ ”用反证法，假设这些点不处于一般位置，那么它们属于一个维度为 p 的超平面，该维度严格小于 $k-1$ 。这个 p -超平面总是在两个不同的 $(k-1)$ -超平面的交集内。那么这些点属于两个不同的 $(k-1)$ -超平面，矛盾。由此可得这些点处于一般位置。

练习18.11.3 用反证法，假设这些向量不是线性无关的。如果考虑张成空间 $\mathcal{S} = \text{span} \left\{ \overrightarrow{P_1 P_2}, \dots, \overrightarrow{P_1 P_k} \right\}$ ，那么 $\dim \mathcal{S} = s < k-1$ 。那么点 P_i 属于一个维度为 s 的超平面，这与它们处于一般位置的事实相矛盾。

练习18.11.4 条件 $f(x_1) = 1$ 和 $f(x_2) = 2$ 可以写成线性方程组

$$\begin{aligned} ax_1 + b &= 1 \\ ax_2 + b &= 2. \end{aligned}$$

只要 $x_1 \neq x_2$ ，该系统就有唯一解 (a, b) 。

练习18.11.5(a) 我们需要证明，对于任意两点 $A, B \in \text{hull}(\mathcal{G})$ ，凸组合 $tA + (1-t)B \in \text{hull}(\mathcal{G})$ 。我们有 $A = \sum \alpha_i P_i$ 和 $B = \sum \beta_i P_i$ ，其中 $\sum \alpha_i = \sum \beta_i = 1$ 。

然后

$$\begin{aligned} tA + (1-t)B &= t \sum \alpha_i P_i + (1-t) \sum \beta_i P_i \\ &= \sum (t\alpha_i + (1-t)\beta_i) P_i = \sum r_i P_i. \end{aligned}$$

这是 P_i 的一个凸组合，因为

$$\sum r_i = t \sum \alpha_i + (1-t) \sum \beta_i = t + (1-t) = 1,$$

因此 $tA + (1-t)B \in \text{hull}(\mathcal{G})$ 。

(b) 通过反证法，我们假设在 \mathbb{R}^n 中有一个凸集 K ，使得 $\mathcal{G} \subset K \subset \text{hull}(\mathcal{G})$ 。选取一个点 $Q \in \text{hull}(\mathcal{G}) \setminus K$ 。那么 Q 是 \mathcal{G} 中各点的凸组合，因此属于 K ，矛盾。

(c) 利用凸集的交集总是凸集这一性质。

练习18.11.6 设 $r(t) = r_0 + tv$ 为 \mathcal{G}_1 和 \mathcal{G}_2 的行分隔符。仿射变换 Φ 将这条线变换为线 $\rho(t) = \Phi(r(t)) = \rho_0 + tu$ ，其中 $\rho_0 = wr_0 + b$ 和 $u = Wv$ 。注意 $u \neq 0$ ，因为 $\det W \neq 0$ 。我们将证明 $\rho(t)$ 是 $\Phi(\mathcal{G}_1)$ 和 $\Phi(\mathcal{G}_2)$ 的分隔符。通过反证法，我们假设 $\rho(t)$ 不分离 $\Phi(\mathcal{G}_1)$ 和 $\Phi(\mathcal{G}_2)$ 。因此，存在一些 $g_i \in \mathcal{G}_i$ ，使得 $\rho(t)$ 不分离 $\Phi(g_1)$ 和 $\Phi(g_2)$ 。因此，线段 $\Phi(g_1)\Phi(g_2)$ 不与线 $\rho(t)$ 相交。由于 $r(t)$ 分离 g_1 和 g_2 ，线段 g_1g_2 在某点 p 与线 $r(t)$ 相交。那么 $\Phi(p)$ 是线段 $\Phi(g_1)\Phi(g_2)$ 和线 $\rho(t)$ 的交点，这是一个矛盾。

值得注意的是，只要分隔线的方向向量不在矩阵 W 的核中，即 $Wv \neq 0$ ，该陈述仍然成立。

练习18.11.7 设 $\mathcal{G} = \{A_1, A_2, \dots, A_m\}$ 。那么 $\overrightarrow{OG} = \frac{1}{m} \sum_{j=1}^k \overrightarrow{OA_j} \in \text{hull}(\mathcal{G})$ 作为聚类元素位置向量的凸组合。

练习18.11.9 (a) 设 $p \in A$ 和 $q \in B$ 使得 $\|p_0q_0\| = \inf_{p \in A, q \in B} \|pq\|$ 。选择分隔线垂直于线段 p_0q_0 的中点。(b) 由(a)可得。

第19章

练习19.7.1 (i) 在这种情况下，状态更新满足系统

$$\begin{aligned} x_{n+1} &= x_n - \eta y_n \\ y_{n+1} &= y_n + \eta x_n, \end{aligned}$$

具有 $\eta > 0$ 学习率。这可以写成矩阵形式

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & -\eta \\ \eta & 1 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix}.$$

(ii) 因此

$$\begin{pmatrix} x_n \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & -\eta \\ \eta & 1 \end{pmatrix}^n \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}.$$

(iii) 矩阵 $A = \begin{pmatrix} 1 & -\eta \\ \eta & 1 \end{pmatrix}$ 的特征值等于 $1 \pm i\eta$ ，且谱半径

$\rho(A^T A) = \sqrt{1 + \eta^2}$ 严格大于1。因此，除非 $x_0 = y_0 = 0$ ，序列 (x_n, y_n) 不会收敛。

练习19.7.2 (i) 该系统为 $x'(t) = -a, y'(t) = b$ ，其解为 $x(t) = -at + x_0, y(t) = bt + y_0$ 。 (ii) 该解是 (x, y) 平面上的一条直线，当 $t \rightarrow \infty$ 时它不趋近于任何平衡点。

练习19.7.3 连续情形：相关的微分系统是 $x' = -x, y' = -y$ ，其解为 $x(t) = x_0 e^{-t}, y(t) = y_0 e^{-t}$ 。无论初始值 (x_0, y_0) 是什么，极限都是 $\lim_{t \rightarrow \infty} (x(t), y(t)) = (0, 0)$ 。原点是一个平衡点。离散情形：我们有

$x_{n+1} = x_n - \eta x_n = (1 - \eta)x_n$ 和 $y_{n+1} = y_n - \eta y_n = (1 - \eta)y_n$ ，这意味着 $x_n = (1 - \eta)^n x_0$ 和 $y_n = (1 - \eta)^n y_0$ 。对于任何 x_0, y_0 和 $\eta \in (0, 1)$ ，我们得到 $\lim_{n \rightarrow \infty} (x_n, y_n) = (0, 0)$ 。

练习19.7.4 (i) $\operatorname{div} U = \partial_x U_1 + \partial_y U_2 = -\partial_x^2 V + \partial_y^2 V = 0$ 。 (ii) $\operatorname{div} U = 0$ 当且仅当 $\partial_x^2 V - \partial_y^2 V = 0$ 。通过变量变换 $u = x + y$ 和 $v = x - y$ ，这变为 $\partial_u \partial_v V = 0$ 。它有分离变量的解 $V = F(u) + G(v) = F(x+y) + G(x-y)$ ，其中 F 和 G 是任意光滑函数。 (iii) 如果散度为零，则流动是不可压缩的，因此在长程中不会趋向于任何点。

练习19.7.5 (i) 我们有

$$\begin{aligned} D_G^*(x) &= \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)} = \frac{1}{1 + \left(\frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}\right)^{-1}} = \frac{1}{1 + e^{-\ln \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}}} \\ &= \sigma\left(\ln \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}\right) = \sigma(a(x)), \end{aligned}$$

由此，通过使用 σ 的单射性，然后取指数，我们得到 $e^{a(x)} = \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}$ 。 (ii) 从库尔贝克-莱布勒散度的定义可得

$$\mathbb{E}_{x \sim p_{\text{data}}} [a(x)] = \int p_{\text{data}}(x) \ln \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)} dx = D_{KL}(p_{\text{data}} || p_{\text{model}}).$$

练习19.7.6 我们写出两个期望在最大似然估计处都有临界点的条件

$$\begin{aligned}\partial_{\theta} \mathbb{E}_{x \sim p_{\text{data}}} [\ln p_{\text{model}}(x; \theta)] \Big|_{\theta=\theta_{MLE}} &= 0 \\ \partial_{\theta} \mathbb{E}_{x \sim p_{\text{model}}} [f(x)] \Big|_{\theta=\theta_{MLE}} &= 0.\end{aligned}$$

它们等同于

$$\begin{aligned}\int p_{\text{data}}(x) \partial_{\theta} \ln p_{\text{model}}(x; \theta) \Big|_{\theta=\theta_{MLE}} dx &= 0 \\ \int f(x) \partial_{\theta} p_{\text{model}}(x; \theta) \Big|_{\theta=\theta_{MLE}} dx &= 0.\end{aligned}$$

通过使被积函数相等得到一个满足此性质的可能函数 $f(x)$ 。我们得到

$$f(x) = \frac{p_{\text{data}}(x) \partial_{\theta} \ln p_{\text{model}}(x; \theta)}{\partial_{\theta} p_{\text{model}}(x; \theta)} \Big|_{\theta=\theta_{MLE}} = \frac{p_{\text{data}}(x)}{p_{\text{model}}(x; \theta_{MLE})}.$$

练习19.7.7 (i) 由于(19.4.4)的第一个期望项与 $\theta^{(g)}$ 无关，我们只对第二项求导并交换期望和梯度算子。

(ii) 我们对期望 $\mathbb{E}_{z \sim p_{\text{code}}} [\ln (1 - D(G(z, \theta^{(g)})))]$ 应用梯度并使用链式法则

$$\nabla_{\theta^{(g)}} \ln(1 - D(G(z, \theta^{(g)}))) = \frac{1}{1 - D(G(z))} \frac{\partial D(x)}{\partial x} \Big|_{x=G(z)} \frac{\partial G(z)}{\partial \theta^{(g)}}.$$

(iii) 使用 $\frac{\partial D(x)}{\partial x} = D(x)(1 - D(x)) \frac{da(x)}{dx}$ 。

练习19.7.8 方程 $\frac{\partial L}{\partial y} = \ln \frac{\hat{y}}{1-\hat{y}} = 0$ 得出 $\hat{y} = \frac{1}{2}$ ，而方程 $\frac{\partial L}{\partial \hat{y}} = \frac{y-\hat{y}}{\hat{y}(1-\hat{y})} = 0$ 意味着 $y = \hat{y}$ 。平衡点是 $(y^*, \hat{y}^*) = (\frac{1}{2}, \frac{1}{2})$ 。

练习19.7.9 使用链式法则得出

$$\begin{aligned}\nabla_{\theta^{(g)}} J^{(G)} &= \mathbb{E}_{z \sim p_{\text{code}}} [\nabla_{\theta^{(g)}} \ln D(G(z, \theta^{(g)}))] \\ &= \mathbb{E}_{z \sim p_{\text{code}}} \left[\frac{1}{D(G(z, \theta^{(g)}))} \frac{\partial D(x)}{\partial x} \Big|_{x=G(z)} \frac{\partial G}{\partial \theta^{(g)}} \right].\end{aligned}$$

假设现在鉴别器已优化，那么根据练习19.7.5，我们有

$D(x) = D_G^*(x) = \sigma(a(x))$ ，其中 $a(x) = \ln \frac{p_{\text{data}}(x)}{p_{\text{model}}(x)}$ 。然后使用链式法则

$$\begin{aligned}\frac{\partial D_G^*(x)}{\partial x} &= \sigma'(a(x)) \frac{da(x)}{dx} = \sigma(a(x))(1 - \sigma(a(x))) \frac{da(x)}{dx} \\ &= D_G^*(x)(1 - D_G^*(x)) \frac{da(x)}{dx}.\end{aligned}$$

在前一个表达式中进行替换，得到

$$\nabla_{\theta^{(g)}} J^{(G)} = \mathbb{E}_{z \sim p_{code}} \left[(1 - D(G(z, \theta^{(g)}))) \frac{da(x)}{dx} \frac{\partial G}{\partial \theta^{(g)}} \right],$$

$$\text{with } \frac{da(x)}{dx} = \frac{\partial_x p_{data}(x)}{p_{data}(x)} - \frac{\partial_x p_{model}(x)}{p_{model}(x)}.$$

第20章

练习20.10.1 单独保留第 k 个神经元的贡献，我们有

$$\begin{aligned} E' - E &= E(\mathbf{x}') - E(\mathbf{x}) \\ &= -\frac{1}{2} \sum_{i \neq k} \sum_{j \neq k} w_{ij} x_i x_j - \sum_{i \neq k} b_i x_i - \sum_{i=1}^n w_{ki} x_i x'_k - b_k x'_k \\ &\quad + \frac{1}{2} \sum_{i \neq k} \sum_{j \neq k} w_{ij} x_i x_j + \sum_{i \neq k} b_i x_i + \sum_{i=1}^n w_{ki} x_i x_k + b_k x_k \\ &= \sum_{i=1}^n w_{ki} x_i (x_k - x'_k) + b_k (x_k - x'_k) \\ &= - \left(\sum_{i=1}^n w_{ki} x_i + b_k \right) (x'_k - x_k). \end{aligned}$$

练习20.10.2 通过识别前7个分量，我们有

$$\begin{aligned} q_1 &= \frac{1}{Z} \\ q_2 &= e^{b_1} q_1 \Rightarrow b_1 = \ln \frac{q_2}{q_1} \\ q_3 &= e^{b_2} q_1 \Rightarrow b_2 = \ln \frac{q_3}{q_1} \\ q_4 &= e^{b_3} q_1 \Rightarrow b_3 = \ln \frac{q_4}{q_1} \\ q_5 &= e^{w_{23} + b_2 + b_3} q_1 \Rightarrow w_{23} = \ln \frac{q_5 q_1}{q_3 q_4} \\ q_6 &= e^{w_{13} + b_1 + b_3} q_1 \Rightarrow w_{13} = \ln \frac{q_6 q_1}{q_2 q_4} \\ q_7 &= e^{w_{12} + b_1 + b_2} q_1 \Rightarrow w_{12} = \ln \frac{q_7 q_1}{q_2 q_3}. \end{aligned}$$

我们识别最后一个分量并代入之前得到的值

$$q_8 = e^{w_{12}} e^{w_{13}} e^{w_{23}} e^{b_1} e^{b_2} e^{b_3} q_1 = \frac{q_5 q_6 q_7}{q_2 q_3 q_4}.$$

练习20.10.3 (a) 这可通过直接计算得出。例如，对于第一个分量，我们有

$$\begin{aligned}
 \sum_{j=1}^N p_{1j} p_j &= \frac{1}{Z} \left\{ p_{11} e^{-E_1/T} + \sum_{j=2}^N p_{1j} e^{E_j/T} \right\} \\
 &= \frac{1}{Z} \left\{ \left(1 - \sum_{i=2}^N \frac{1}{1 + e^{(E_i - E_1)/T}} \right) e^{-E_1/T} + \sum_{j=2}^N \frac{e^{-E_j/T}}{1 + e^{(E_1 - E_j)/T}} \right\} \\
 &= \frac{1}{Z} \left\{ e^{-E_1/T} - \sum_{i=2}^N \frac{1}{e^{E_1/T} + e^{E_i/T}} + \sum_{j=2}^N \frac{1}{e^{E_j/T} + e^{E_1/T}} \right\} \\
 &= \frac{1}{Z} e^{-E_1/T} = p_1.
 \end{aligned}$$

(b) 在接下来的计算中，我们将利用 P_T 的所有元素都是非负的，并且每行元素之和等于1这一条件。使用 L^1 范数， $\|v\|_1 = \sum_j |v_j|$ ，我们有

$$\begin{aligned}
 \|P_T v\|_1 &= |\sum_j p_{1j} v_j| + \dots + |\sum_j p_{nj} v_j| \\
 &\leq \sum_j p_{1j} |v_j| + \dots + \sum_j p_{nj} |v_j| = \sum_i \sum_j p_{ij} |v_j| \\
 &= \sum_i \underbrace{\left(\sum_j p_{ij} \right)}_{=1} |v_j| = \sum_j |v_j| = \|v\|_1.
 \end{aligned}$$

如果 λ 是对应于特征向量 v 的特征值，那么 $P_T v = \lambda v$ ，取范数得到 $\|P_T v\|_1 = |\lambda| \|v\|_1$ 。然后利用之前的计算我们得到 $|\lambda| = \frac{\|P_T v\|_1}{\|v\|_1} \leq 1$

◦ (c) 首先，我们证明 q_n 是收敛的。通过迭代，我们得到 $q_n = P_T^n q_0$ 。由于 $P_T^n = (MDM^{-1})^n = MD^n M^{-1}$ ，那么 $q_n = MD^n M^{-1} q_0$ 。由于对角矩阵 D 仅包含绝对值小于或等于1的元素，极限 $L = \lim_{n \rightarrow \infty} D^n$ 是一个只有一个元素等于1的稀疏矩阵。那么以下极限存在：

$$q^* = \lim_{n \rightarrow \infty} q_n = \lim_{n \rightarrow \infty} MD^n M^{-1} q_0 = MLM^{-1} q_0.$$

另一方面，我们在关系 $q_{n+1} = P_T q_n$ 中应用极限并得到 $q^* = P_T q^*$ 。这意味着 q^* 和 p 成比例（我们未经证明地使用了特征值为1的特征空间的维数等于1这一事实）。由于它们元素的和为1，所以可得 $q^* = p$ 。

其物理意义在于，对于任何初始状态，从长远来看，玻尔兹曼机都会趋于平衡分布。

练习20.10.4 玻尔兹曼分布为 $p = \frac{1}{Z} (1, e^{b_1}, e^{b_2}, e^{b_1+b_2+w})$ 。费希尔度量涉及此类计算

$$\begin{aligned}\mathbb{E}^p[x_1x_2] &= \frac{1}{Z}[p(0,0) \cdot 0 \cdot 0 + p(1,0) \cdot 1 \cdot 0 + p(0,1) \cdot 0 \cdot 1 + p(1,1) \cdot 1 \cdot 1] \\ &= \frac{1}{Z}p(1,1) = \frac{1}{Z}e^{b_1+b_2+w}.\end{aligned}$$

其他的读者可以轻松计算出来。

练习20.10.5 (a) 从 $\frac{\partial}{\partial w_{ij}} \ln p(\mathbf{x}) = \frac{1}{p(\mathbf{x})} \frac{\partial}{\partial w_{ij}} p(\mathbf{x})$ 和 $\frac{\partial}{\partial w_{ij}} \ln p(\mathbf{x}) = x_i x_j - \mathbb{E}^p[x_i x_j]$ 可得

$$\frac{\partial}{\partial w_{ij}} p(\mathbf{x}) = (x_i x_j - \mathbb{E}^p[x_i x_j]) p(\mathbf{x}).$$

那么

$$\begin{aligned}\sum_{i,j} w_{ij} \frac{\partial}{\partial w_{ij}} p(\mathbf{x}) &= \left(\sum_{i,j} w_{ij} x_i x_j - \mathbb{E}^p[w_{ij} x_i x_j] \right) p(\mathbf{x}) \\ &= (\mathbf{x}^T w \mathbf{x} - \mathbb{E}^p[\mathbf{x}^T w \mathbf{x}]) p(\mathbf{x}).\end{aligned}$$

类似地，

$$\sum_k b_k \frac{\partial}{\partial b_k} p(\mathbf{x}) = (\mathbf{x}^T b - \mathbb{E}^p[\mathbf{x}^T b]) p(\mathbf{x})$$

因此

$$\begin{aligned}A_{w,b} p(\mathbf{x}) &= \left(\frac{1}{2} \mathbf{x}^T w \mathbf{x} + \mathbf{x}^T b \right) p(\mathbf{x}) - \mathbb{E}^p \left[\frac{1}{2} \mathbf{x}^T w \mathbf{x} + \mathbf{x}^T b \right] p(\mathbf{x}) \\ &= (\mathbb{E}^p[E(\mathbf{x})] - E(\mathbf{x})) p(\mathbf{x}).\end{aligned}$$

(b) 由于 $w'_{ij}(t) = \alpha w_{ij}(t)$ 和 $b'_k(t) = \alpha b_k(t)$ ，我们有

$$\begin{aligned}\partial_t E(x) &= -\partial_t \left[\frac{1}{2} \mathbf{x}^T w(t) \mathbf{x} + \mathbf{x}^T b(t) \right] = -\left[\frac{1}{2} \mathbf{x}^T w'(t) \mathbf{x} + \mathbf{x}^T b(t) \right] \\ &= -\alpha \left[\frac{1}{2} \mathbf{x}^T w(t) \mathbf{x} + \mathbf{x}^T b(t) \right] = \alpha E(\mathbf{x}).\end{aligned}$$

在 $\ln p_t(\mathbf{x}) = -E(\mathbf{x}) - \ln Z(t)$ 中求导，我们得到

$$\begin{aligned}\partial_t \ln p_t(\mathbf{x}) &= -\partial_t E(\mathbf{x}) - \frac{1}{Z} \partial_t Z = -\alpha E(\mathbf{x}) - \frac{1}{Z(t)} \sum_x \partial_t e^{-E(\mathbf{x})} \\ &= -\alpha E(\mathbf{x}) + \sum_x p(\mathbf{x}) \partial_t E(\mathbf{x}) = -\alpha E(\mathbf{x}) + \alpha \sum_x p(\mathbf{x}) E(\mathbf{x}) \\ &= \alpha \left(\mathbb{E}^p[E(\mathbf{x})] - E(\mathbf{x}) \right).\end{aligned}$$

使用 $\partial_t p_t(\mathbf{x}) = \partial_t \ln p_t(\mathbf{x}) p_t(\mathbf{x})$ ，我们得到了所需的公式。

(c) 由 (a) 和 (b) 可知， $w_{ij}(t) = w_{ij}e^{\alpha t}$, $b_k(t) = b_k e^{\alpha t}$ 是该问题的一个解。(局部) 唯一性由解公式 $p_t(\mathbf{x}) = e^{tA_{w,b}} p(\mathbf{x})$. 得出。

练习20.10.6 由于车只沿水平和垂直方向攻击，所以每行每列必须只有一个车。我们从在第一行放置一个车开始。这样做有8种可能性。在下一行放置另一个车。由于它不能与第一个车在同一列，所以还剩下7种可能性。再下一行有6个可能的位置，依此类推，直到在最后一行只有一个可能的位置放置最后一个车。可能性的总数是乘积 $8 \cdot 7 \cdots 2 \cdot 1$ 。因此，有 $8!$ 种可能的方式放置车而不使它们相互威胁。这些是有利选择。所有可能选择的数量是从64个方格中选择8个方格的可能性数量，即二项式系数 $\binom{64}{8}$ 。为了得到所需概率，我们需要将有利选择的数量除以放置车的所有可能方式的数量， $p = \frac{8!}{\binom{64}{8}}$ 。

练习20.10.7 (a) 使用条件概率公式，我们有

$$\begin{aligned} p(h, v) &= \frac{p(v, h)}{p(v)} = \frac{p(v, h)}{\sum_h p(v, h)} = \frac{\frac{1}{Z} e^{-E(v, h)}}{\frac{1}{Z} \sum_h e^{-E(v, h)}} = \frac{e^{-E(v, h)}}{\sum_h e^{-E(v, h)}} \\ &= \frac{e^{v^T w h} e^{b^T v} e^{c^T h}}{\sum_h e^{v^T w h} e^{b^T v} e^{c^T h}} = \frac{e^{v^T w h} e^{c^T h}}{\sum_h e^{v^T w h} e^{c^T h}} = \frac{e^{v^T w h} e^{c^T h}}{Z'} \\ &= \frac{1}{Z'} e^{\sum_j \sum_i v_i w_{ij} h_j} e^{\sum_j c_j h_j} = \frac{1}{Z'} \prod_j e^{\sum_i v_i w_{ij} h_j + c_j h_j} \\ &= \prod_j \frac{1}{Z'_j} e^{(\sum_i v_i w_{ij} + c_j) h_j} = \prod_j p(h_j | v), \end{aligned}$$

对于划分函数 $Z'_k = e^{\left(\sum_i v_i w_{ij} + c_j\right) \cdot 0} + e^{\left(\sum_i v_i w_{ij} + c_j\right) \cdot 1} = 1 + e^{\left(\sum_i v_i w_{ij} + c_j\right)}$ 。由于 $p(h, v) = \prod_j p(h_j | v)$ ，所以在给定可见状态的情况下， h_j 是条件独立的。

(b) 使用前面的公式，我们有

$$p(h_j | v) = \frac{1}{Z'_j} e^{(\sum_i v_i w_{ij} + c_j) h_j} = \frac{e^{(\sum_i v_i w_{ij} + c_j) h_j}}{1 + e^{(\sum_i v_i w_{ij} + c_j)}},$$

这意味着

$$\begin{aligned} p(h_j = 1|v) &= \frac{e^{(\sum_i v_i w_{ij} + c_j)}}{1 + e^{(\sum_i v_i w_{ij} + c_j)}} = \frac{1}{1 + e^{-(\sum_i v_i w_{ij} + c_j)}} = \sigma\left(\sum_i v_i w_{ij} + c_j\right), \\ p(h_j = 0|v) &= 1 - \sigma\left(\sum_i v_i w_{ij} + c_j\right) = \sigma\left(-\sum_i v_i w_{ij} - c_j\right). \end{aligned}$$

(c) 对数似然函数及其偏导数由下式给出

$$\begin{aligned} \ell(h|v) &= \ln p(h|v) = \ln \frac{e^{v^T wh + c^T h}}{Z'} = v^T wh + c^T h - \ln Z' \\ \partial_{b_k} \ell(h|v) &= 0 \\ \partial_{c_k} \ell(h|v) &= h_k - \frac{1}{Z'} \partial_{c_k} Z' = h_k - \frac{1}{Z'} \partial_{c_k} \left(\sum_h e^{v^T wh + c^T h} \right) \\ &= h_k - \frac{1}{Z'} \sum_h e^{v^T wh + c^T h} h_k = h_k - \sum_h p(h|v) h_k \\ &= h_k - \mathbb{E}^{p_{v|h}}[h_k]; \\ \partial_{w_{ij}} \ell(h|v) &= v_i h_j + v_j h_i - \frac{1}{Z'} \partial_{w_{ij}} Z' \\ &= v_i h_j + v_j h_i - \frac{1}{Z'} \sum_h e^{v^T wh + c^T h} (v_i h_j + v_j h_i) \\ &= v_i h_j + v_j h_i - \mathbb{E}^{p_{v|h}}[v_i h_j + v_j h_i]. \end{aligned}$$

(d) 费希尔矩阵的以下元素为零：

$$g_{b_i b_j} = 0, \quad g_{b_i c_j} = 0, \quad g_{b_i w_{ij}} = 0.$$

其他元素由隐藏单元的条件相关性给出

$$\begin{aligned} g_{c_i c_j} &= Cor(h_i, h_j | v) \\ g_{w_{ij} c_{kl}} &= Cor(v_i h_j + v_j h_i, v_k h_l v_l h_k | v). \end{aligned}$$

(e) 为了最小化库尔贝克 - 莱布勒散度， $D_{KL}(q(\mathbf{h} | \mathbf{v}) \| p(\mathbf{h} | \mathbf{v}))$ 等同于最大化成本函数 $C = \sum_h q(h | v) \ln p(h | v)$ 。通过与玻尔兹曼学习类似的计算，我们得到

$$\begin{aligned} \Delta w_{ij} &= \eta \left(\mathbb{E}^{q_{h|v}}[h_i h_j] - \mathbb{E}^{p_{h|v}}[h_i h_j] \right) \\ \Delta c_k &= \eta \left(\mathbb{E}^{q_{h|v}}[h_k] - \mathbb{E}^{p_{h|v}}[h_k] \right) \\ \Delta b &= 0. \end{aligned}$$

附录

为了使本书尽可能自成体系，我们在本附录中简要介绍了全书使用的一些概念。它们是测度论、概率论、线性代数和泛函分析的基本概念。

然而，本附录并非旨在替代上述领域的完整课程；它只是为读者提供足够的信息，以便能够顺利跟上本书的内容。

附录A 集合论

设 $(A_i)_{i \in I}$ 为一个集族。它们的并集和交集将分别记为

$$\bigcup_{i \in I} A_i = \{x; \exists i \in I, x \in A_i\}, \quad \bigcap_{i \in I} A_i = \{x; \forall i \in I, x \in A_i\}.$$

如果 $(A_i)_{i \in I}$ 和 $(B_j)_{j \in J}$ 是两个集族，那么：

$$(a) \left(\bigcup_i A_i \right) \cap \left(\bigcup_j B_j \right) = \bigcup_{i,j} (A_i \cap B_j)$$

$$(b) \left(\bigcap_i A_i \right) \cup \left(\bigcap_j B_j \right) = \bigcap_{i,j} (A_i \cup B_j)$$

这些关系可以推广到任意数量的集族 $(A_{i_1}^1)_{i_1 \in I_1}, (A_{i_2}^2)_{i_2 \in I_2}, \dots, (A_{i_p}^p)_{i_p \in I_p}$ ，如下所示

$$(a') \bigcap_{r=1}^p \left(\bigcup_{i_r} A_{i_r}^r \right) = \bigcup_{i_1, \dots, i_p} \left(\bigcap_{r=1}^p A_{i_r}^r \right)$$

$$(b') \bigcup_{r=1}^p \left(\bigcap_{i_r} A_{i_r}^r \right) = \bigcap_{i_1, \dots, i_p} \left(\bigcup_{r=1}^p A_{i_r}^r \right).$$

在处理西格玛域时，函数的以下性质很有用。它们可以通过双向包含来证明。在下面， \mathcal{X} 和 \mathcal{Y} 表示两个集合。

命题 A.0.1 设 $f : \mathcal{X} \rightarrow \mathcal{Y}$ 为一个函数。那么：

$$(a) f \underset{i}{\cup} A_i = f(A_i), \quad \forall A_i \subset \mathcal{X};$$

- (b) $f\left(\bigcap_i A_i\right) = \bigcap_i f(A_i)$, $\forall A_i \subset \mathcal{X}$;
- (c) $f^{-1}\left(\bigcup_i B_i\right) = \bigcup_i f^{-1}(B_i)$, $\forall B_i \subset \mathcal{Y}$
- ; (d)
- $f^{-1}\left(\bigcap_i B_i\right) = \bigcap_i f^{-1}(B_i)$, $\forall B_i \subset \mathcal{Y}$;
- (e) $f^{-1}(B^c) = (f^{-1}(B))^c$, $\forall B \subset \mathcal{Y}$ 。

一个集合 $A \subset \mathbb{R}^n$ 如果能被包含在一个球内, 即存在一个 $r > 0$ 使得 $A \subset B(0, r)$, 那么就称它是有界的。等价地, 对于所有 $x \in A$, $\|x\| \leq r$ 。

一个集合 $A \subset \mathbb{R}^n$ 如果 A 包含任何收敛序列 $(x_n) \subset A$ 的极限, 即 $\lim_{n \rightarrow \infty} x_n \in A$, 那么就称它是闭集。

\mathbb{R}^n 的一个子集 K 如果有界且封闭, 则称为紧集。等价地, 如果对于任何序列 $(x_n) \subset K$, 我们都能提取出一个收敛子序列 $(x_{n_k})_k$, 那么 K 就是紧集。在本书中, \mathbb{R}^n 中紧集的典型例子是超立方体 $K = I_n = [0, 1] \times \cdots \times [0, 1]$ 和 n 维闭球 $K = B(x_0, r) = \{x; \|x - x_0\| \leq r\}$ 。

命题A.0.2 (康托尔引理) \mathbb{R}^n 中紧致集的递减序列的交集非空。

例如, 如果 $K_n = [-1/n, 1/n]$, 那么 $K_{n+1} \subset K_n \subset \mathbb{R}$ 和 $\bigcap_{k \geq 1} K_n = \{0\}$ 。

非空集合 A 上的二元关系“ \leq ”如果对于任意 $a, b, c \in A$ 都满足以下条件, 则称为序关系

- (i) $a \leq a$ (自反性);
- (ii) $a \leq b$ 且 $b \leq a$, 那么 $a = b$ (反对称性);
- (iii) $a \leq b$ 和 $b \leq c$ 然后 $a \leq c$ (传递性)。

一个定义了序关系“ \leq ”的集合 A 被称为有序集, 并用 (A, \leq) 表示。如果对于任意两个元素 $a, b \in A$, 我们要么有 $a \leq b$ 要么有 $b \leq a$, 那么 (A, \leq) 被称为全序集。如果 $m \leq x$ 意味着 $m = x$, 则元素 $m \in A$ 被称为极大元。考虑一个子集 $B \subset A$ 。如果对于所有 $x \in B$ 都有 $x \leq a$, 那么元素 $a \in A$ 被称为 B 的上界。如果任何子集 $B \subset A$ 都有一个上界, 那么有序集 (A, \leq) 被称为归纳集。

引理A.0.3 (佐恩引理) 任何非空的归纳有序集都至少有一个极大元。

附录B

张量

设 $I_1, \dots, I_n \subset \mathbb{N}$ 为自然数集 n 的子集 \mathbb{N} ，并考虑笛卡尔积

$$I_1 \times \cdots \times I_n = \{(i_1, \dots, i_n); i_k \in I_k, 1 \leq k \leq n\}.$$

一组由多重指标 $(i_1, \dots, i_n) \in I_1 \times \cdots \times I_n$ 索引的对象是一个 n 阶张量。通常，我们用 T_{i_1, \dots, i_n} 表示它。

值得注意的是，无需详述，张量概念源自微分几何和相对论，在那里它被成功用于描述流形曲率、切向量场或某些物理量，如速度、能量动量、力、质量密度等。

神经网络中的许多对象，如输入、权重、偏差、中间表示和输出，都是由张量描述的。例如，一个由 $x = (x_1, \dots, x_d)$ 给出的向量 $x \in \mathbb{R}^d$ 是一阶张量，类型为 d 。一个矩阵 $A = (A_{ij}) \in \mathbb{R}^{d \times r}$ 是二阶张量，类型为 $d \times r$ (d 行和 r 列)，见图1a。一个三阶张量，比如 $t \in \mathbb{R}^{d \times r \times s}$ ，可以看作是一个长度为 s 的矩阵向量，类型为 $d \times r$ 。这个张量的一个通用元素由 t_{ijk} 表示，见图1b。一个标量值可以看作是零阶张量。

示例B.0.1 彩色图像可以表示为类型为 $n \times m \times 3$ 的三阶张量，其中 n 是像素行数， m 是列数，3表示RGB格式中的颜色通道数。

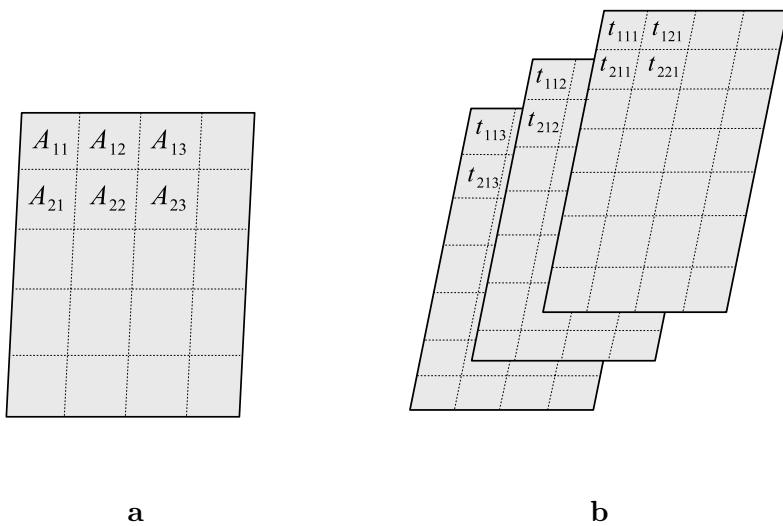


图1：a. 一个二阶张量 $A \in \mathbb{R}^{5 \times 4}$ 。b. 一个三阶张量 $t \in \mathbb{R}^{7 \times 4 \times 3}$ 。

附录C测度论

对测度论更详细阐述感兴趣的读者应查阅参考文献[51]。

C.1信息与 σ -代数

σ -代数的概念对于描述信息结构以及稍后定义可测函数和测度很有用。为了便于理解，将引入 σ -代数的概念作为存储在一组心理概念中的信息。

我们假设大脑是一组 N 神经元，每个神经元被视为一个可以处于开或关状态的装置。这导致总共 2^N 种可能的大脑状态。该集合的任何子集都对应于一个心理概念的表示。所有可能的心理概念的集合将如下定义一个 σ -代数。

假设一个人看着一个物体。那么他/她的思维会进入与那个特定物体相关的某种状态。例如，分别看着一个“苹果”和一个“瓶子”，思维会产生两个分别由 A 和 B 表示的心理概念。然后，我们的日常经验表明，思维能够将“苹果和瓶子”理解为一个由交集 $A \cap B$ 表示的心理概念，其中包含这两个物体的共同特征，如颜色、形状、大小等。

思维还能够将“苹果或瓶子”的复合心理概念理解为并集概念 $A \cup B$ ，其中包含苹果或瓶子的所有特征。思维能够理解苹果不是瓶子，并且一般来说，如果呈现一个苹果，那么它能理解所有不是苹果的物体。

这是由一个用 \bar{A} 表示的概念完成的，称为集合 A 的补集。

心理概念 A 和 B 表示心灵从外部世界获得的两条信息。然后它有能力将它们编译成新的信息，比如 $A \cap B, A \cup B, \bar{A}$ 和 \bar{B} 。

如果所有可能的心理概念都用 \mathcal{E} 表示，那么前面的陈述可以写成

- (i) $A \cup B \in \mathcal{E}, \forall A, B \in \mathcal{E}$
- (ii) $A \cap B \in \mathcal{E}, \forall A, B \in \mathcal{E}$
- (iii) $\bar{A} \in \mathcal{E}, \forall A \in \mathcal{E}$ 。

关系 (i) 可以推广到 n 集合，如下所示：

(iv) 对于任何 $A_1, \dots, A_n \in \mathcal{E}$ ，那么 $\bigcup_{i=1}^n A_i \in \mathcal{E}$ 。

由 (i) – (iv) 定义的 \mathcal{E} 的结构称为代数结构。我们注意到(ii)是其他两个条件的结果，这一事实可由德摩根关系 $\overline{A \cap B} = \bar{A} \cup \bar{B}$ 推出。

现在假设思维有能力选取一个无限可数的信息序列并将其也存储为信息，即 \mathcal{E} 关于可数并集是封闭的：

(v) 对于任何 $A_1, \dots, A_n, \dots \in \mathcal{E}$ ，则 $\bigcup_{i>1} A_i \in \mathcal{E}$ 。

在 \mathcal{E} 也满足条件 (v) 的情况下，该结构称为 \mathfrak{S} -代数。这将是建模信息的基本结构。

注记C.1.1 利用德摩根关系可知， \mathfrak{G} -代数对可数并封闭，即对于任何 $A_1, \dots, A_n, \dots \in \mathcal{E}$ ，有 $\bigcap_{i \geq 1} A_i \in \mathcal{E}$ 。

示例C.1.1 设 E 为有限集。 E 上最小的 \mathfrak{S} -代数是 $\mathcal{E} = \{\emptyset, E\}$ ，而最大的是幂集 $\mathcal{E} = 2^E = \{\mathcal{P}; \mathcal{P} \subseteq E\}$ 。

示例C.1.2 设 \mathcal{C} 是 E 的一个幂集，即 $\mathcal{C} \subset 2^E$ 。那么 E 上包含 \mathcal{C} 的最小 \mathfrak{S} -代数由包含 \mathcal{C} 的所有 \mathfrak{S} -代数 \mathcal{E}_α 的交集给出。

$$\mathfrak{S}(\mathcal{C}) = \bigcap_{\alpha} \mathcal{E}_{\alpha}.$$

在此，我们注意到 \mathfrak{S} -代数的交集也是一个 \mathfrak{S} -代数。 $\mathfrak{S}(\mathcal{C})$ 是由集合 \mathcal{C} 的集合生成的信息结构。可以证明它具有以下性质：

- (i) $\mathcal{C} \subset \mathcal{D} \Rightarrow \mathfrak{S}(\mathcal{C}) \subset \mathfrak{S}(\mathcal{D})$ ；
- (ii) $\mathcal{C} \subset \mathfrak{S}(\mathcal{D}) \Rightarrow \mathfrak{S}(\mathcal{C}) \subset \mathfrak{S}(\mathcal{D})$ ；
- (iii) 如果 $\mathcal{C} \subset \mathfrak{S}(\mathcal{D})$ 且 $\mathcal{D} \subset \mathfrak{S}(\mathcal{C})$ ，那么 $\mathfrak{S}(\mathcal{C}) = \mathfrak{S}(\mathcal{D})$ ；
- (iv) $\mathcal{C} \subset \mathcal{D} \subset \mathfrak{S}(\mathcal{C}) \Rightarrow \mathfrak{S}(\mathcal{C}) = \mathfrak{S}(\mathcal{D})$ 。

示例C.1.3 设 E 为一个拓扑空间。博雷尔 \mathcal{G} -代数 \mathcal{B}_E 是由 E 的所有开集生成的 \mathcal{G} -代数。特别地，如果 $E = \mathbb{R}^n$ ，那么 $\mathcal{B}_{\mathbb{R}^n}$ 是由 \mathbb{R}^n 的所有开球生成的 \mathcal{G} -代数。

示例C.1.4 博雷尔G代数 $\mathcal{B}_{\mathbb{R}}$ 是由以下所有集合生成的G代数：

$$\{(-\infty, x); x \in \mathbb{R}\}, \{(-\infty, x]; x \in \mathbb{R}\}, \{(x, y); x, y \in \mathbb{R}\}, \{(x, y]; x, y \in \mathbb{R}\}, \{(x, \infty); x \in \mathbb{R}\}.$$

定义C.1.2 A 子集的集合 \mathcal{P} 属于 Ω ，如果它在交运算下封闭，则称为p系统，即

$$A, B \in \mathcal{P} \implies A \cap B \in \mathcal{P}.$$

Ω 的子集的集合 \mathcal{D} 在 Ω 上称为d系统，如果：

- (i) $\Omega \in \mathcal{D}$ ；
- (ii) $A, B \in \mathcal{D}$ 和 $B \subset A \Rightarrow A \setminus B \in \mathcal{D}$ ；
- (iii) $(A_n)_n \subset \mathcal{D}$ 和 $A_n \nearrow A \Rightarrow A \in \mathcal{D}$.

定理C.1.3（迪金）如果一个d系包含一个p系，那么它也包含由该p系生成的 σ -代数，即

$$\mathcal{P} \subset \mathcal{D} \implies \mathfrak{S}(\mathcal{P}) \subset \mathcal{D}$$

一个可测空间是一个对 (E, \mathcal{E}) ，其中 E 是一个集合， \mathcal{E} 是 E 上的一个部分集，它是一个 \mathcal{G} -代数。

例如，大脑的配置可以被视为一个可测量的空间。在这种情况下， E 是大脑突触的集合（大脑的状态）。大脑将信息存储在配置（大脑状态的集合）中。大脑的一个配置是突触集合 E 中被激活的一个子集。大脑配置的集合 \mathcal{E} 形成一个 \mathcal{G} -代数，因此，对 (E, \mathcal{E}) 成为一个可测量的空间。

C. 2 可测函数

测量的大致概念似乎很简单：它是一个为每个集合赋予一个数字的过程。这个集合可以是，例如，一条线上或平面上的一组点，或者是具有某种特征的一组人等等。然而，像“有多少人的体重恰好是200磅？”这样的问题并没有明确提出。这是因为为了进行测量，我们需要一个数字所属的区间。可能有很多人的体重在199.9磅和200.01磅之间，并且被视为恰好是200磅。

我们在编写计算机程序时可能会遇到类似的问题。设 L 和 D 表示圆的长度和直径。程序：

$x = L/D$

如果 ($x = \pi$)

打印("它是一个圆")

不会运行，因为商永远不会恰好等于 π 。正确的版本应该包括一个误差容限：

$x = L/D$

如果 ($abs(x - \pi) < 0.0001$)

打印("它可能是一个圆")

因此，正确的测量方法是为测量结果指定一个下限和一个上限。这就是由开区间定义的信息发挥作用的地方，开区间是一种无聊的 \mathcal{G} 代数。

设 $f: E \rightarrow \mathbb{R}$ 为一个函数，其中 E 表示大脑的突触集合。若函数 f 能将任意开区间映射为一个先验给定的大脑配置，则称该函数是可测的。这意味着对于所有 $a, b \in \mathbb{R}$ 有 $f^{-1}(a, b) \in \mathcal{E}$ ，即 $f^{-1}(a, b)$ 是大脑配置之一，这里我们使用了记号

$f^{-1}(a, b) = \{x \in E; f(x) \in (a, b)\}$ 。我们也可以等价地将其写为

$f^{-1}(\mathcal{B}_{\mathbb{R}}) \subset \mathcal{E}$ 。为了明确表明可测性是相对于 \mathcal{G} -代数 $\mathcal{B}_{\mathbb{R}}$ 而言的，函数 f 有时被称为波莱尔可测函数。大脑可以利用可测函数来感知外部世界。

如果 (E, \mathcal{E}) 和 (F, \mathcal{F}) 是两个可测空间，那么函数 $f: E \rightarrow F$ 若对于所有 $B \in \mathcal{F}$ 都满足 $f^{-1}(B) \in \mathcal{E}$ ，则称其为可测函数。或者，等价地说， $f^{-1}(\mathcal{F}) \subset \mathcal{E}$ 。如果 (E, \mathcal{E}) 是与安的大脑相关联的可测空间，而 (F, \mathcal{F}) 是与鲍勃的大脑相关联的，那么函数 f 是可测的这一事实意味着“鲍勃能想到的任何事情，安都能理解”。更确切地说，鲍勃大脑中的任何配置状态 B 都可以通过 f 回归到安大脑中存在的一种配置状态。

示例C.2.1 设 $A \in \mathcal{E}$ 为一个集合，并考虑 A 的指示函数

$$1_A(x) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A. \end{cases}$$

那么 1_A 是可测的（相对于 \mathcal{E} ）。

示例C.2.2 一个函数 $f: E \rightarrow F$ 如果是指示函数的线性组合，则称为简单函数

$$f(x) = \sum_{j=1}^n a_j 1_{A_j}(x), \quad a_j \in \mathbb{R}, A_j \in \mathcal{E}.$$

任何简单函数都是可测的。

给定一些可测函数，可以用它们来构造更多的可测函数，如下所示：

1. 如果 f 和 g 是可测的，那么 $f \pm g, f \cdot g, \min(f, g), \max(f, g)$ 是可测的。
2. 如果 $(f_n)_n$ 是一列可测函数，那么 $\inf f_n, \sup f_n, \liminf f_n, \limsup f_n$ 以及 $\lim f_n$ （如果存在）都是可测的。
3. 如果 f 是可测的，那么 $f^+ = \max(f, 0)$ 和 $f^- = -\min(f, 0)$ 是可测的。

可以证明，任何可测函数都是一列简单函数的极限。如果该函数有界，则同样的界也适用于这些简单函数。

C. 3 测度

测度是评估一组信息的一种方式。这可以通过使用具有以下性质的映射 $\mu : \mathcal{E} \rightarrow \mathbb{R}_+ \cup \{\infty\}$ 来完成：

- (i) $\mu(\emptyset) = 0$ ；
- (ii) $\mu\left(\bigcup_{n>1} A_n\right) = \sum_{n>1} \mu(A_n)$ ，对于 \mathcal{E} 中任何不相交的集合 A_j （可数可加性）。

如果 (E, \mathcal{F}) 分别是与大脑的状态和配置相关联的可测空间，那么测度 μ 就是一个信念评估系统；每个大脑配置 $A \in \mathcal{E}$ 都与一个强度 $\mu(A)$ 相关联。三元组 (E, \mathcal{E}, μ) 被称为测度空间。

示例C.3.1（狄拉克测度）设 $x \in E$ 为一个固定点。那么

$$\delta_x(A) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A, \end{cases} \quad \forall A \in \mathcal{E}$$

是 \mathcal{E} 上的一个测度。

示例C.3.2（计数测度）设 $D \subset E$ 是固定且有限的。定义

$$\mu(A) = \text{card}(A \cap D) = \sum_{x \in D} \delta_x(A), \quad \forall A \in \mathcal{E}.$$

μ 是 \mathcal{E} 上的一个测度。它计算 A 中属于 D 的元素数量。

示例C.3.3（离散测度）设 $D \subset E$ 为固定的离散集。定义 $m(x)$ 为 x 的质量，其中 $m(x) \geq 0$ ，对于所有 $x \in D$ 。考虑

$$\mu(A) = \text{mass}(A) = \sum_{x \in D} m(x) \delta_x(A), \quad \forall A \in \mathcal{E}.$$

μ 是 \mathcal{E} 上的一个测度，它评估 A 的质量。

示例C.3.4 (勒贝格测度) 考虑 $E = \mathbb{R}$ 以及 \mathcal{G} -代数 $\mathcal{E} = \mathcal{B}_{\mathbb{R}}$, μ , 它对开区间的定义为其长度, $\mu(a, b) = |b - a|$ 。可以证明, 存在唯一的测度将 μ 扩展到 $\mathcal{B}_{\mathbb{R}}$, 称为 \mathbb{R} 上的勒贝格测度。类似地, 将长度替换为体积, 开区间替换为开超立方体, 就可以定义 \mathbb{R}^n 上的勒贝格测度。

示例C.3.5 (博雷尔测度) 设 $\mathcal{B}_{\mathbb{R}^n}$ 为由 \mathbb{R}^n 的所有开集生成的 \mathcal{G} 代数。博雷尔测度是一种测度 $\mu : \mathcal{B}_{\mathbb{R}^n} \rightarrow \mathbb{R}$ 。在 $n = 1$, μ 的情况下, 它成为实直线上的博雷尔测度。如果 μ 是有限测度, 我们将其与函数 $F(x) = \mu(-\infty, x]$ 相关联, 该函数称为累积分分布函数, 它是单调递增函数, 满足 $\mu(a, b] = F(b) - F(a)$ 。例如, 勒贝格测度是一种博雷尔测度。

示例C.3.6 (贝尔测度) 设 $K \subseteq \mathbb{R}^n$, 并用 $C^0(K)$ 表示所有具有紧支集的连续实值函数的集合 (这些函数在 K 的紧子集之外消失)。贝尔集的类 \mathcal{B} 被定义为由 $\{x; f(x) \geq a\}$ 生成的 \mathcal{G} -代数, 其中 $f \in C^0(K)$ 。贝尔测度是定义在 \mathcal{B} 上的测度, 使得对于所有紧子集 $C \subset K$, $\mu(C) < \infty$ 。值得注意的是, 对于 $K \subseteq \mathbb{R}^n$, 贝尔集的类与博雷尔集的类相同。特别地, 任何有限博雷尔测度都是贝尔测度。

命题C.3.1 (测度的性质) 设 (E, \mathcal{E}, μ) 为一个测度空间。以下性质成立:

(i) 有限可加性:

$$A \cap B = \emptyset \implies \mu(A \cup B) = \mu(A) + \mu(B), \quad \forall A, B \in \mathcal{E};$$

(ii) 单调性:

$$A \subset B \implies \mu(A) \leq \mu(B), \quad \forall A, B \in \mathcal{E};$$

(iii) 序列连续性:

$$A_n \nearrow A \implies \mu(A_n) \nearrow \mu(A), \quad n \rightarrow \infty;$$

(iv) 布尔不等式:

$$\mu\left(\bigcup_n A_n\right) \leq \sum_n \mu(A_n), \quad \forall A_n \in \mathcal{E}.$$

如果 μ 和 λ 是关于 (E, \mathcal{E}) 的测度，那么 $\mu + \lambda, c\mu$ 和 $c_1\mu + c_2\lambda$ 也是测度，其中 $c, c_i \in \mathbb{R}_+$ 。

设 (E, \mathcal{E}, μ) 为一个测度空间。那么 μ 被称为：

- 有限测度，如果 $\mu(E) < \infty$ ；

- 概率测度，如果 $\mu(E) = 1$ ；

- \mathfrak{S} -有限测度，如果 $\mu(E_n) < \infty$ ，其中 $(E_n)_n$ 是 E 的划分，且 $E_n \in \mathcal{E}$

- \sum -有限测度，若 $\mu = \sum \mu_n$ ，且 μ_n 为有限测度。

例如，勒贝格测度是 \mathfrak{L} -有限的，但它不是有限的。

一个集合 $M \in \mathcal{E}$ 若满足 $\mu(M) = 0$ ，则称其为可忽略的。两个可测函数 $f, g : E \rightarrow \mathbb{R}$ 几乎处处相等，即 $f = g$ a.e.，若存在一个可忽略集合 M 使得对于所有 $x \in E \setminus M$ ，都有 $f(x) = g(x)$ 。

C. 4 测度中的积分

设 (E, \mathcal{E}, μ) 为一个测度空间， $f : E \rightarrow \mathbb{R}$ 为一个可测函数。对象

$$\mu(f) = \int_E f(x) \mu(dx) = \int_E f d\mu$$

通过信念系统 μ 表示对 f 的一种评估，它被称为 f 关于测度 μ 的积分。这由以下步骤序列定义：

(i) 如果 f 是简单且正的，即，如果 $f = \sum_{i=1}^n w_i 1_{A_i}$ ，那么定义

$$\mu(f) = \sum_{i=1}^n w_i \mu(A_i).$$

(ii) 如果 f 是可测且正的，那么存在一个简单且正的函数序列 $(f_n)_n$ ，满足 $f_n \nearrow f$ 。在这种情况下，定义 $\mu(f) = \lim_n \mu(f_n)$ 。(iii) 如果 f 是可测的，那么令 $f = f^+ - f^-$ 并定义

$$\mu(f) = \mu(f^+) - \mu(f^-).$$

如果 $\mu(f) < \infty$ ，那么可测函数 f 被称为可积的。积分的非负性、线性和单调性性质分别由以下给出

1. 对于 $f : E \rightarrow \mathbb{R}_+$ ， $\mu(f) \geq 0$ ；

2. 对于所有 $a, b \in \mathbb{R}$ ， $\mu(af + bg) = a\mu(f) + b\mu(g)$ ；

3. 如果 $f \leq g$ ，那么 $\mu(f) \leq \mu(g)$ 。

示例C.4.1 设 δ_x 为位于 x 的狄拉克测度。关于狄拉克测度的可测函数 f 的积分是 $\delta_x(f) = f(x)$ 。

示例C.4.2 可测函数 f 关于离散测度 $\mu = \sum_{x \in D} m(x) \delta_x$ 的积分由 $\mu(f) = \sum_{x \in D} m(x) f(x)$ 给出。

示例C.4.3 设 $E = \mathbb{R}$, $\mathcal{E} = \mathcal{B}_{\mathbb{R}}$ 和 μ 是 \mathbb{R} 上的勒贝格测度。在这种情况下, $\mu(f) = \int_E f(x) dx$ 称为 f 在 E 上的勒贝格积分。

f 在集合 $A \in \mathcal{E}$ 上的积分定义为

$$\int_A f d\mu = \int_E f 1_A d\mu = \mu(f 1_A).$$

特别地, 我们有

$$\int_A d\mu = \mu(1_A) = \mu(A), \quad \forall A \in \mathcal{E}.$$

我们提供接下来用于交换积分与极限的三个关键工具。

定理C.4.1 (单调收敛定理) 设 $(f_n)_n$ 是 E 上的一列正可测函数, 使得 $f_n \nearrow f$ 。那么

$$\lim_{n \rightarrow \infty} \int f_n d\mu = \int f d\mu.$$

定理C.4.2 (控制收敛定理) 设 $(f_n)_n$ 是一列可测函数, 满足 $|f_n| \leq g$, 且 g 在 E 上可积。如果 $\lim_n f_n$ 存在, 那么

$$\lim_{n \rightarrow \infty} \int f_n d\mu = \int f d\mu.$$

定理C.4.3 (有界收敛定理) 设 $(f_n)_n$ 是 E 上的一个有界可测函数序列, 且 $\mu(E) < \infty$ 。如果 $\lim_{n \rightarrow \infty} f_n$ 存在, 那么

$$\lim_{n \rightarrow \infty} \int f_n d\mu = \int f d\mu.$$

我们注意到, 几乎处处相等的两个可测函数具有相等的积分, 即积分在可忽略集上的变化是不敏感的。

C. 5 图像测度

设 (F, \mathcal{F}) 和 (E, \mathcal{E}) 为两个可测空间， $h : F \rightarrow E$ 为一个可测函数。 (F, \mathcal{F}) 上的任何测度 ν 通过以下方式在 (E, \mathcal{E}) 上诱导出一个测度 μ ：

$$\mu(B) = \nu(h^{-1}(B)), \quad \forall B \in \mathcal{E}.$$

测度 $\mu = \nu \circ h^{-1}$ 称为 ν 在 h 下的像测度。值得注意的是，如果 $f : E \rightarrow \mathbb{R}$ 是可测的，那么上述关系就变成了以下变量替换公式：

$$\int_E f(y) d\mu(y) = \int_F f(h(x)) d\nu(x),$$

前提是积分存在。

备注C.5.1 如果 (F, \mathcal{F}) 和 (E, \mathcal{E}) 分别表示安和鲍勃大脑的（状态，配置）对，那么 h 可测这一事实意味着“鲍勃能想到的任何事情，安都能理解”。测度 μ 是鲍勃的一个信念系统，它由安的信念系统 ν 诱导得出。

C. 6 不定积分

设 (E, \mathcal{E}, μ) 为一个测度空间且 $p : E \rightarrow \mathbb{R}_+$ 可测。那么

$$\nu(A) = \int_A p(x) d\mu(x), \quad A \in \mathcal{E}$$

是 (E, \mathcal{E}) 上的一个测度，称为 p 关于 μ 的不定积分。可以证明，对于任何可测正函数 $f : E \rightarrow \mathbb{R}_+$ ，我们有

$$\int_E f(x) d\nu(x) = \int_E f(x)p(x) d\mu(x).$$

这可以非正式地写为 $p(x)d\mu(x) = d\nu(x)$ 。

C. 7 拉东 - 尼科迪姆定理

设 μ, ν 是 (E, \mathcal{E}) 上的两个测度，使得

$$\mu(A) = 0 \Rightarrow \nu(A) = 0, \quad A \in \mathcal{E}.$$

那么 ν 被称为相对于 μ 绝对连续。以下结果表明存在一个密度函数 p 。

定理C.7.1 设 μ 为 \mathcal{G} -有限，且 ν 相对于 μ 绝对连续。则存在一个可测函数 $p : E \rightarrow \mathbb{R}_+$ ，使得

$$\int_E f(x) d\nu(x) = \int_E f(x)p(x) d\mu(x),$$

对于所有可测函数 $f : E \rightarrow \mathbb{R}_+$ 。

前面的积分也可以非正式地写成 $p(x)d\mu(x) = d\nu(x)$ 。密度函数 $p = \frac{d\nu}{d\mu}$ 称为拉东 - 尼科迪姆导数。 —

注记C.7.2 这涉及对该定理的一种非正式理解。如果将测度 μ, ν 视为两个评估系统， ν 相对于 μ 绝对连续这一事实意味着评估系统 ν 不如系统 μ 严格。这意味着在系统 ν 中可忽略不计的错误，即 $\mu(A) = 0$ ，在系统 μ 中也未被察觉，即 $\nu(A) = 0$ 。在此假设下，拉东 - 尼科迪姆定理表明，更严格的系统可以通过关系 $d\nu = pd\mu$ 按比例缩放为较不严格的系统。密度函数 p 成为评估系统之间的缩放函数。

C. 8 叶戈罗夫定理和卢津定理

叶戈罗夫定理建立了几乎处处收敛和一致收敛之间的关系。

设 (E, \mathcal{E}, μ) 为一个测度空间， $f_n : E \rightarrow \mathbb{R} \cup \{\infty\}$ 为一列扩充实值函数。我们称 f_n 几乎处处收敛于极限 f ，如果当 $n \rightarrow \infty$ 时，对于所有 $x \in E \setminus N$ ，有 $f_n(x) \rightarrow f(x)$ ，其中 N 是一个 μ -可忽略集。

我们称序列 (f_n) 在 A 上一致收敛于 f ，如果存在 $\exists n_0 > 1$ 使得

$$|f_n(x) - f(x)| < \epsilon, \quad \forall n > n_0, \forall x \in A.$$

定理C.8.1 (叶戈罗夫) 设 E 为一个测度有限的可测集，且 f_n 为一列几乎处处有限值的可测函数，它们在 E 上几乎处处收敛于一个有限的可测函数 f 。那么对于任意 $\epsilon > 0$ ，存在 E 的一个可测子集 F ，使得 $\mu(F) < \epsilon$ 且 (f_n) 在 $E \setminus F$ 上一致收敛于 f 。

粗略地说，任何几乎处处逐点收敛的可测函数列在其几乎所有定义域上是一致收敛的。

这个结果的一个推论是卢津定理，该定理表明一个可测函数在其定义域上几乎处处连续，因此一个连续函数可以由可测函数逼近。

在 $(\mathbb{R}^n, \mathcal{B}_{\mathbb{R}^n})$ 上的一个测度 μ 如果满足以下条件，则称为正则的：

$$\mu(A) = \inf \{\mu(U); A \subset U, U \text{ open in } \mathbb{R}^n\} = \sup \{\mu(V); V \subset A, V \text{ open in } \mathbb{R}^n\}.$$

这意味着测度结构与空间的拓扑结构相关。

定理C.8.2 如果 $f: I_n = [0, 1]^n \rightarrow \mathbb{R}$ 是一个可测函数，那么对于任意的 $\epsilon > 0$ ，在 I_n 中存在一个紧集 K ，使得 $\mu(I_n \setminus K) < \epsilon$ 并且 f 在 K 上连续，其中 μ 是一个正则博雷尔测度。

C. 9 符号测度

两个测度的差一般不是一个测度，因为它不一定是非负的。在下面我们将处理这个概念。

可测空间 (E, \mathcal{E}) 上的带号测度是一个映射

$$\nu: \mathcal{E} \rightarrow \mathbb{R} \cup \{\pm\infty\}$$

使得

(i) ν 至多取 $-\infty, +\infty$ 中的一个值；

(ii) $\nu(\emptyset) = 0$ ；

(iii) 对于 \mathcal{E} 中任意不相交集合的序列 (A_i) （即， ν 是可数可加的），

$$\nu\left(\bigcup_{i \geq 1} A_i\right) = \sum_{i \geq 1} \nu(A_i).$$

例C.9.1 任何测度都是带号测度，但反之一般不成立。

示例C.9.2 两个测度的任何差， $\mu = \nu_1 - \nu_2$ 都是一个带号测度。正如我们即将看到的，此陈述的逆命题也成立。

设 ν 为一个带号测度。 \mathcal{E} 中的集合 G 若其任何可测子集的测度均非负，则称其相对于 ν 为正集，即

$$\nu(G \cap A) \geq 0, \quad \forall A \in \mathcal{E}.$$

\mathcal{E} 中的集合 F 若其任何可测子集的测度均非正，则称其相对于 ν 为负集，即

$$\nu(F \cap B) \leq 0, \quad \forall B \in \mathcal{E}.$$

相对于一个带号测度既是正集又是负集的集合称为零集。我们注意到任何零集的测度为零，而反之一般不成立。

命题C.9.3 (哈恩分解定理) 设 ν 为可测空间 (E, \mathcal{E}) 上的一个符号测度。那么存在 E 的一个划分，分为正集 A 和负集 B ，即 $E = A \cup B, A \cap B = \emptyset$ 。

值得注意的是，哈恩分解不是唯一的。然而，对于 E 的任意两个不同分解 $\{A_1, B_1\}$ 和 $\{A_2, B_2\}$ ，可以证明

$$\nu(F \cap A_1) = \nu(F \cap A_2), \quad \nu(F \cap B_1) = \nu(F \cap B_2), \quad \forall F \in \mathcal{E}.$$

这建议为 E 的任意哈恩分解 $\{A, B\}$ 定义两个测度 ν^+, ν^- ，使得

$$\nu^+(F) = \nu(F \cap A), \quad \nu^-(F) = -\nu(F \cap B), \quad \forall F \in \mathcal{E}.$$

我们注意到 $\nu = \nu^+ - \nu^-$ 。我们还有 $\nu^+(B) = \nu^-(A) = 0$ ，其中 $\{A, B\}$ 是 E 的可测划分。具有此属性的一对测度 ν^+, ν^- 称为相互奇异的。

命题C.9.4 (约旦测度分解) 设 ν 是可测空间 (E, \mathcal{E}) 上的一个符号测度。那么在 (E, \mathcal{E}) 上存在两个测度 ν^+ 和 ν^- ，使得

$\nu = \nu^+ - \nu^-$ 。此外，如果 ν^+ 和 ν^- 是相互奇异的测度，则分解是唯一的。

ν^+ 和 ν^- 这两个测度分别称为 ν 的正部和负部。由于 ν 仅取 $\pm\infty$ 中的一个值，所以其中一部分必定是有限测度。测度 $|\nu| = \nu^+ + \nu^-$ 的和是一个测度，称为 ν 的绝对值。

关于带号测度 ν 的可测函数 f 的积分定义为

$$\int f d\nu = \int f d\nu^+ - \int f d\nu^-,$$

前提是 f 关于 $|\nu|$ 是可积的。此外，如果 $|f| < C$ ，那么

$$\left| \int_E f d\nu \right| \leq C|\nu|(E).$$

例C.9.1 设 g 是测度空间 (E, \mathcal{E}, ν) 上的一个可积函数。测度

$$\nu(F) = \int_F g(x) d\nu(x), \quad \forall F \in \mathcal{E}$$

是一个有限带号测度。分解 $g = g^+ - g^-$ 产生 $\nu = \nu^+ - \nu^-$ ，其中

$$\nu^+(F) = \int_F g^+(x) d\nu(x), \quad \nu^-(F) = \int_F g^-(x) d\nu(x).$$

注记C.9.5 若 μ 是可测空间 (Ω, \mathcal{F}) 上的带号有限测度，则将其全变差定义为

$$\|\mu\|_{TV} = \sup \sum_{i=1}^n |\mu(A_i)|$$

对所有由互不相交集合 $\Omega = \bigcup_{i=1}^n A_i$ 构成的有限划分而言。这组具有范数 $\|\cdot\|_{TV}$ 的测度构成一个巴拿赫空间。

附录D概率论

对概率感兴趣的读者可查阅[23]这本书以获取详细信息。

D.1 一般定义

概率空间是一个测度空间 $(\Omega, \mathcal{H}, \mathbb{P})$ ，其中 \mathbb{P} 是一个概率测度，即一个满足 $\mathbb{P}(\Omega) = 1$ 的测度。集合 Ω 是样本空间，有时也被视为世界的状态；它代表了一个实验的结果集。 σ -代数 \mathcal{H} 是历史信息；每个集合 $H \in \mathcal{H}$ 都被称为一个事件。概率测度 \mathbb{P} 评估事件发生的可能性。对于每个 $H \in \mathcal{H}$ ，数字 $\mathbb{P}(H)$ 是 H 发生的概率。

随机变量是一个映射 $X : \Omega \rightarrow \mathbb{R}$ ，它是 $(\mathcal{H}, \mathcal{B}_{\mathbb{R}})$ 可测的，即 $f^{-1}(\mathcal{B}_{\mathbb{R}}) \subset \mathcal{H}$ 。这意味着对于每个实验结果 $\omega \in \Omega$, X ，都分配一个数字 $X(\omega)$ 。

概率测度 \mathbb{P} 通过 $X, \mu = \mathbb{P} \circ X^{-1}$ 的像，是 $(\mathbb{R}, \mathcal{B}_{\mathbb{R}})$ 上的一个测度，称为随机变量 X 的分布。更准确地说，我们有

$$\mu(A) = \mathbb{P}(X \in A) = \mathbb{P}(\omega; X(\omega) \in A) = \mathbb{P}(X^{-1}(A)).$$

测度 μ 描述了 X 是如何分布的。函数

$$F(x) = \mu(-\infty, x] = \mathbb{P}(X \leq x)$$

称为 X 的分布函数。

设 $(\Omega, \mathcal{H}, \mathbb{P})$ 为一个概率空间， $(E, \mathcal{E}), (F, \mathcal{F})$ 为两个可测空间。若 $X : \Omega \rightarrow E$ 和 $f : E \rightarrow F$ 是可测的，考虑它们的合成 $Y = f \circ X$ 。若 μ 是 X 的分布，则 $\nu = \mu \circ f^{-1}$ 是 Y 的分布。我们可以写成

$$\begin{aligned}\nu(B) &= \mathbb{P}(Y \in B) = \mathbb{P}(f \circ X \in B) = \mathbb{P}(X \in f^{-1}(B)) \\ &= \mu(f^{-1}(B)) = (\mu \circ f^{-1})(B), \quad \forall B \in \mathcal{F}.\end{aligned}$$

D. 2 示例

伯努利分布 一个随机变量 X 如果满足 $X \in \{0, 1\}$ ，其中 $P(X = 1) = p$ 且 $P(X = 0) = 1 - p$ ，则称其服从参数为 $p \in [0, 1]$ 的伯努利分布。其均值和方差分别为

$$\mathbb{E}[X] = 1 \cdot P(X = 1) + 0 \cdot P(X = 0) = p,$$

$\text{Var}(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2 = 1^2 \cdot P(X = 1) + 0^2 \cdot P(X = 0) - p^2 = p(1 - p)$ 我们将写 $X \sim \text{Bernoulli}(p)$ 。

正态分布 一个随机变量 X 如果 $P(X < x) = \int_{-\infty}^x f(x)dx$ ，则被称为参数为 μ 和 σ 的正态分布，其中

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad x \in \mathbb{R}.$$

均值和方差由下式给出

$$\mathbb{E}[X] = \mu, \quad \text{Var}(X) = \sigma^2.$$

我们将使用符号 $X \sim \mathcal{N}(\mu, \sigma^2)$ 。

D. 3 期望

随机变量 X 的期望是通过概率测度 \mathbb{P} 对 X 的评估，如下所示

$$\mathbb{E}[X] = \mathbb{P}(X) = \int_{\omega} X(\omega) d\mathbb{P}(\omega).$$

如果 μ 是 X 的分布测度，那么变量变换公式给出

$$\mathbb{E}[f(X)] = \int_{\Omega} f(X(\omega)) d\mathbb{P}(\omega) = \int_{\mathbb{R}} f(y) d\mathbb{P}(X^{-1}y) = \int_{\mathbb{R}} f(y) d\mu(y)$$

特别地，如果 μ 关于 \mathbb{R} 上的勒贝格测度 dy 是绝对连续的，那么存在一个非负可测密度函数 $p(y)$ 使得 $d\mu(y) = p(y)dy$ 。因此，前面的公式变为

$$\mathbb{E}[f(X)] = \int_{\mathbb{R}} f(y)p(y)dy.$$

期望算子是非负的、单调的和线性算子，即，

- (i) $X \geq 0 \Rightarrow \mathbb{E}[X] \geq 0$ ；
- (ii) $X \geq Y \Rightarrow \mathbb{E}[X] \geq \mathbb{E}[Y]$ ；
- (iii) 对于 $a, b \in \mathbb{R}$ ， $\mathbb{E}[aX + bY] = a\mathbb{E}[X] + b\mathbb{E}[Y]$ 。

D. 4 方差

设 $\mu = \mathbb{E}[X]$ 为 X 的均值。随机变量 X 的方差定义为

$$Var(X) = \mathbb{E}[(X - \mu)^2].$$

方差是在均方意义上偏离均值的一种度量。如果设 $p(x)$ 表示 X 的概率密度，从物理意义上讲，方差表示曲线 $y = p(x)$ 关于垂直轴 $x = \mu$ 的惯性矩。这是图形 $y = p(x)$ 绕通过其质心 μ 的垂直轴旋转难易程度的一种度量。

一般来说，方差既不是可加的也不是可乘的。然而，在某些特定情况下它是，正如我们接下来将看到的。两个随机变量 X 和 Y 的协方差由 $Cov(X, Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$ 定义。和的方差由下式给出

$$Var(X + Y) = Var(X) + 2Cov(X, Y) + Var(Y).$$

如果 X 和 Y 相互独立，那么 $Var(X + Y) = Var(X) + Var(Y)$ 。方差也是2阶齐次的，即对于所有 $c \in \mathbb{R}$ ， $Var(cX) = c^2 Var(X)$ 。

我们还有关于独立随机变量乘积方差的以下精确表达式，见[49]。

引理D.4.1 (古德曼公式) 如果 X 和 Y 是两个独立随机变量，那么

$$Var(XY) = \mathbb{E}[X]^2Var(Y) + \mathbb{E}[Y]^2Var(X) + Var(X)Var(Y).$$

特别地，如果 $\mathbb{E}[X] = \mathbb{E}[Y] = 0$ ，那么

$$Var(XY) = Var(X)Var(Y).$$

证明：将右边记为

$$R = \mathbb{E}[X]^2Var(Y) + \mathbb{E}[Y]^2Var(X) + Var(X)Var(Y).$$

方差的定义以及一些代数运算，再结合独立性性质，可得

$$\begin{aligned}
 R &= Var(Y)[\mathbb{E}[X]^2 + Var(X)] + \mathbb{E}[Y]^2Var(X) \\
 &= Var(Y)\mathbb{E}[X^2] + \mathbb{E}[Y]^2Var(X) \\
 &= [\mathbb{E}[Y^2] - \mathbb{E}[Y]^2]\mathbb{E}[X^2] + \mathbb{E}[Y]^2[\mathbb{E}[X^2] - \mathbb{E}[X]^2] \\
 &= \mathbb{E}[Y^2]\mathbb{E}[X^2] - \mathbb{E}[Y]^2\mathbb{E}[X]^2 = \mathbb{E}[X^2Y^2] - \mathbb{E}[XY]^2 \\
 &= Var(XY),
 \end{aligned}$$

这就得到了所需表达式的左边。 ■

方差近似 设 $m = \mathbb{E}[X]$ 和 f 为可微函数。关于 $x = m$ 的线性近似可得

$$f(x) = f(m) + f'(m)(x - m) + o(x - m)^2.$$

然后将变量 x 替换为随机变量 X ，得到

$$f(X) \approx f(m) + f'(m)(X - m).$$

对两边取方差并利用其性质，我们得到以下近似公式：

$$Var(f(X)) \approx f'(m)^2Var(X), \quad (\text{D.4.1})$$

前提是 f 二阶可微且 m 的均值和 X 的方差有限。因此，若 f' 有界， X 的小方差会导致 $f(X)$ 的小方差。

D. 5 由随机变量生成的信息

设 $X : \Omega \rightarrow \mathbb{R}$ 为一个随机变量。由 X 生成的信息域是 \mathcal{G} -代数

$$\mathcal{G}(X) = X^{-1}(\mathcal{B}_{\mathbb{R}}).$$

设 $X : \Omega \rightarrow \mathbb{R}$ 为一个随机变量且 $f : \mathbb{R} \rightarrow \mathbb{R}$ 为一个可测函数。考虑随机变量 $Y = f(X)$ 。那么 Y 是 $\mathcal{G}(X)$ -可测的，即， $\mathcal{G}(Y) \subset \mathcal{G}(X)$ 。等价地说，由 X 生成的信息决定了由 Y 生成的信息。反之亦然。以下结果的证明可在Cinlar [23]，命题4.4，第76页中找到。

命题D.5.1 考虑两个随机向量变量 $X, Y : \Omega \rightarrow \mathbb{R}$ 。那么当且仅当 Y 由 X 确定时 $\mathcal{G}(Y) \subset \mathcal{G}(X)$ 成立，即存在一个可测函数 f 使得 $Y = f(X)$ 。

这也可以说表述为当且仅当由 X 生成的信息比由 Y 生成的信息更精细时， Y 由 X 确定。

值得注意的是，先前的函数 f 是通过一个极限过程构造的。其思路如下：对于固定的 n ，我们考虑可测集

$$A_{m,n} = Y^{-1} \left[\frac{m}{2^n}, \frac{m+1}{2^n} \right] \in \mathfrak{S}(Y) \subset \mathfrak{S}(X), \quad m = 0, \pm 1, \pm 2, \dots$$

使得 $A_{m,n} = X^{-1}(B_{m,n})$ ，其中 $B_{m,n}$ 是可测的。构造简单函数 $f_n(x) = \sum_m \frac{m}{2^n} 1_{B_{m,n}}(x)$ 。可以证明

$$f_n(X) \leq Y \leq f_n(X) + \frac{1}{2^n}.$$

然后我们选择可测的 $f = \lim_{n \rightarrow \infty} f_n$ 作为简单函数的极限。

示例 D.5.1 设 $\omega = (\omega_1, \omega_2, \omega_3) \in \Omega$ 并通过以下方式定义随机变量 $X_i, Y_i : \Omega \rightarrow \mathbb{R}, i = 1, 2, 3$

$$X_1(\omega) = \omega_1, \quad X_2(\omega) = \omega_2, \quad X_3(\omega) = \omega_3,$$

$$Y_1(\omega) = \omega_1 - \omega_2, \quad Y_2(\omega) = \omega_1 + \omega_2, \quad Y_3(\omega) = \omega_1 + \omega_2 + \omega_3.$$

然后 $\mathfrak{S}(X_1, X_2) = \mathfrak{S}(Y_1, Y_2)$ 和 $\mathfrak{S}(X_1, X_2, X_3) = \mathfrak{S}(Y_1, Y_2, Y_3)$ ，但是 $\mathfrak{S}(X_2, X_3) \neq \mathfrak{S}(Y_2, Y_3)$ ，因为 Y_3 不能用 X_2 和 X_3 表示。

随机过程是一族由连续或离散参数 t 索引的随机变量 $(X_t)_{t \in T}$ 。随机过程 $(X_t)_{t \in T}$ 生成的信息是最小的 \mathfrak{S} -代数，相对于该代数，每个随机变量 X_t 都是可测的。这可以写成

$$\mathcal{G} = \mathfrak{S}(X_t; t \in T) = \mathfrak{S}_{t \in T} X_t = \mathfrak{S}_{t \in T} (X_t).$$

D.5.1 过滤

设 $(\Omega, \mathcal{H}, \mathbb{P})$ 为一个概率空间。 \mathfrak{S} -代数 \mathcal{H} 可被解释为世界状态 Ω 的完整历史。直到时间 t 时可用的信息由 \mathcal{F}_t 表示。我们注意到信息随时间增长，即，如果 $s < t$ ，那么 $\mathcal{F}_s \subset \mathcal{F}_t$ 。在 $\mathcal{H}, (\mathcal{F}_t)_{t \in T}$ 上的信息流增加，称为一个滤子。

每个随机过程 $(X_t)_{t \in T}$ 定义一个自然滤子

$$\mathcal{F}_t = \mathfrak{S}(X_s; s \leq t),$$

这是直到每个时间实例 t 的过程历史。在这种情况下，每个随机变量 X_t 都是 \mathcal{F}_t 可测的。具有此属性的随机过程称为适应该滤子。

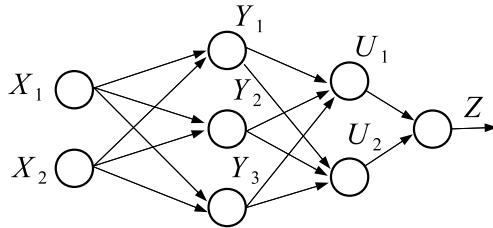


图1：在前馈神经网络中，信息流满足包含关系序列 $\mathcal{F}_Z \subset \mathcal{F}_U \subset \mathcal{F}_Y \subset \mathcal{F}_X$ ，这构成了一个滤链。

示例D.5.2 考虑图1所示的前馈神经网络，它有两个隐藏层，
 $Y = (Y_1, Y_1, Y_3)^T, U = (U_1, U_2)^T$ 。其输入由随机变量
 $X = (X_1, X_2)^T$ 给出，输出由 Z 给出。输入信息是由 X 生成的信息，由 σ 代数 $\mathcal{F}_X = \sigma(X_1, X_2)$ 给出。第一和第二隐藏层中的信息分别由 $\mathcal{F}_Y = \sigma(Y_1, Y_2, Y_3)$ 和 $\mathcal{F}_U = \sigma(U_1, U_2)$ 给出。输出信息是 $\mathcal{F}_Z = \sigma(Z)$ 。由于 Y_j 由 X_i 确定，所以 Y_j 是 \mathcal{F}_X 可测的，这一事实也可以写成 $\mathcal{F}_Y \subset \mathcal{F}_X$ 。类似地，我们有以下一系列包含关系：

$$\mathcal{F}_Z \subset \mathcal{F}_U \subset \mathcal{F}_Y \subset \mathcal{F}_X.$$

这种自然过滤描述了通过网络的信息流。

D.5.2 条件期望

设 X 是概率空间 $(\Omega, \mathcal{H}, \mathbb{P})$ 上的一个随机变量，并考虑一些部分信息 $\mathcal{F} \subset \mathcal{H}$ 。由 \mathcal{F} 确定的随机变量，即 X 的“最佳逼近器”，称为给定 \mathcal{F} 时 X 的条件期望。它被定义为满足以下性质的变量 \tilde{X} ：

(i) \tilde{X} 是 \mathcal{F} 可测的；

(ii) $\int_A X d\mathbb{P} = \int_A \tilde{X} d\mathbb{P}, \forall A \in \mathcal{F}$ 。

可以证明，在平方可积随机变量 X 的情况下，我们有

$$\|X - \tilde{X}\|^2 \leq \|X - Z\|^2, \quad \forall Z \in \mathcal{S}_{\mathcal{F}},$$

其中 $\mathcal{S}_{\mathcal{F}} = \{f \in L^2(\Omega); f \text{ 是关于 } \mathcal{F} \text{ 的}\}$ 可测的。这意味着 \tilde{X} 是 X 在 $\mathcal{S}_{\mathcal{F}}$ 上的正交投影，即，它是在均方意义下用 $\mathcal{S}_{\mathcal{F}}$ 中的元素对 X 的最佳逼近。在前述关系中，范数是由平方的期望诱导的范数， $\|X\|^2 = \mathbb{E}[X^2]$ 。

D.6 收敛类型

在以下内容中， $(\Omega, \mathcal{H}, \mathbb{P})$ 表示一个概率空间， $X_n : \Omega \rightarrow \mathbb{R}$ 是一列随机变量。

D.6.1 依概率收敛

序列 X_n 依概率收敛于 X ，如果 $\lim_{n \rightarrow \infty} \mathbb{P}(|X_n - X| < \epsilon) = 1$ 。这种收敛类型有如下解释。如果 X 表示半径为 ϵ 的目标的中心，而 X_n 表示第 n 次射击的位置，那么 $\{|X_n - X| < \epsilon\}$ 表示第 n 次射击击中目标的事件。依概率收敛意味着从长远来看，射击 X_n 击中以 X 为中心、任意固定半径 ϵ 的任何目标的概率趋近于1。

D.6.2 几乎必然收敛

X_n 几乎必然收敛到 X ，如果 $\lim_{n \rightarrow \infty} \mathbb{P}(\omega; X_n(\omega) \rightarrow X(\omega)) = 1$ 。这意味着对于几乎任何状态 $\omega \in \Omega$ ，序列 $X_n(\omega)$ 作为实数序列收敛到 $X(\omega)$ ，当 $n \rightarrow \infty$ 时。

以下结果提供了几乎必然收敛的必要条件。

命题D.6.1 (波莱尔 - 坎泰利引理I) 假设

$$\sum_{n \geq 1} P(|X_n - X| > \epsilon) < \infty, \quad \forall \epsilon > 0.$$

那么 $X_n \rightarrow X$ 几乎必然。

命题D.6.2 (波莱尔 - 坎泰利引理II) 假设存在一个递减到0的序列 (ϵ_n) ，使得

$$\sum_{n \geq 1} P(|X_n - X| > \epsilon_n) < \infty.$$

那么 $X_n \rightarrow X$ 几乎必然。

值得注意的是，几乎必然收敛意味着依概率收敛。

D.6.3 L^p -收敛

如果 $X_n, X \in L^2(\Omega)$ ，则序列 X_n 在 L^p 意义下收敛于 X 并且

$$\mathbb{E}(|X_n - X|^p) \rightarrow 0, \quad n \rightarrow \infty.$$

值得注意的是马尔可夫不等式

$$P(|X_n - X| > \epsilon) \leq \frac{\mathbb{E}(|X_n - X|^p)}{\epsilon^p}$$

这意味着依 L^p 收敛意味着依概率收敛。

如果 $p = 2$ ，那么在 L^2 中的收敛也称为均方收敛。

一个涉及所有先前收敛类型的经典结果是大数定律：

定理D.6.3 考虑 $X_1, X_2, X_3 \dots$ 个独立同分布的随机变量，其均值为 $\mathbb{E}[X_j] = a$ ，方差为 $\text{Var}[X_j] = b$ ，均为有限值。设 $\bar{X}_n = \frac{1}{n}(X_1 + \dots + X_n)$ 。那么 \bar{X}_n 依均方收敛、几乎必然收敛且依概率收敛于 a 。

D.6.4 弱收敛

考虑随机变量 X_n, X ，其分布测度分别为 μ_n, μ ，并记为

$$\mathbb{C}_b = \{f : \mathbb{R} \rightarrow \mathbb{R}; f \text{ continuous and bounded}\}.$$

如果序列 X_n 依分布收敛于 X ，则

$$\mathbb{E}[f \circ X_n] \rightarrow \mathbb{E}[f \circ X], n \rightarrow \infty \quad \forall f \in \mathbb{C}_b.$$

如果测度 μ_n 弱收敛于 μ ，则

$$\mu_n f \rightarrow \mu f, \quad n \rightarrow \infty \quad \forall f \in \mathbb{C}_b.$$

关系

$$\mu(f) = \int f(x) d\mu(x) = \int_{\Omega} f \circ X d\mathbb{P} = \mathbb{E}[f \circ X]$$

表明随机变量的依分布收敛对应于相关分布测度的弱收敛。

另一种等价的表述如下。如果 $\varphi_X(t) = \mathbb{E}[e^{itX}]$ 表示随机变量 X 的特征函数，那么 X_n 依分布收敛于 X 当且仅当对于所有 $t \in \mathbb{R}$ ， $\varphi_{X_n}(t) \rightarrow \varphi_X(t)$ 成立。如果 μ 表示 X 的分布测度，那么 $\varphi_X(t) = \int_{\mathbb{R}} e^{itx} d\mu(x)$ 。这被称为 μ 的傅里叶变换。此外，如果 μ 关于 dx 是绝对连续的，那么 X 具有概率密度 $p(x)$ ，即 $d\mu(x) = p(x)dx$ ，然后 $\varphi_X(t) = \int_{\mathbb{R}} e^{itx} p(x) dx$ 。这被称为 $p(x)$ 的傅里叶变换。

一个有用的性质是该变换的单射性，即，如果 $\varphi_X = 0$ ，那么 $p = 0$ 。一个启发式的解释基于频率的概念。更准确地说，时域信号 $p(x)$ （即， $p(x)$ 是信号在时间 x 处的幅度）的傅里叶变换是频域中的一个信号，由 $\varphi_X(t)$ 给出（即， $\varphi_X(t)$ 是信号在频率 t 处的幅度）。现在，如果傅里叶变换为零，即对于每个 t ， $\varphi_X(t) = 0$ ，这意味着无论信号频率如何都不存在信号。这一定是零信号， $p = 0$ 。

值得注意的是，前面提到的所有类型的收敛都意味着依分布收敛。下面的经典结果使用了这种收敛类型。

定理D.6.4 (中心极限定理) 设 X_1, X_2, X_3, \dots 为独立同分布随机变量，均值为 $\mathbb{E}[X_j] = a$ ，方差为 $\text{Var}(X_j) = b$ ，二者均为有限值。记 $Z_n = \frac{S_n - na}{\sqrt{nb}}$ ，其中 $S_n = X_1 + \dots + X_n$ 。则 Z_n 依分布收敛于标准正态变量 $\xi \sim N(0, 1)$ 。这也可表述为分布函数的收敛性

$$\lim_{n \rightarrow \infty} \mathbb{P}(Z_n \leq x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du.$$

D.7 对数似然函数

本节给出了对数函数、指数函数和线性函数的一些函数方程。

对于能够定义信息的概念而言，存在一个将数字乘积转换为和的函数是至关重要的。

设 s 为一个概率为 $P(s)$ 的事件。当概率 $P(s)$ 较小时，即当该事件是一个意外时， s 所传递的信息量大。那么 s 中包含的信息由其概率 $g(P(s))$ 的一个函数给出，其中 g 是一个正的递减函数，且 $g(0+) = +\infty$ 和 $g(1-) = 0$ 。这意味着概率为零的事件具有无限信息，而概率为1的事件实际上不包含信息。此外，如果 s_1 和 s_2 是两个分别具有实现概率 $P(s_1) = \pi_1$ 和 $P(s_2) = \pi_2$ 的独立事件，通过启发式推理，两个事件产生的信息必须是每个事件产生的个体信息之和，即

$$g(P(s_1 \cap s_2)) = g(\pi_1 \pi_2) = g(\pi_1) + g(\pi_2).$$

下一个命题表明，满足这些性质的唯一函数是 $g(x) = -\ln x$ ，这里包含负号是为了保证正性。

命题D.7.1 任何满足条件的可微函数 $f : (0, \infty) \rightarrow \mathbb{R}$

$$f(xy) = f(x) + f(y), \quad \forall x, y \in (0, \infty)$$

具有 $f(x) = c \ln x$ 的形式，其中 c 为实常数。

证明：设 $y = 1 + \epsilon$ ，其中 $\epsilon > 0$ 很小。那么

$$f(x + x\epsilon) = f(x) + f(1 + \epsilon). \quad (\text{D.7.2})$$

利用 f 的连续性，取极限

$$\lim_{\epsilon \rightarrow 0} f(x + x\epsilon) = f(x) + \lim_{\epsilon \rightarrow 0} f(1 + \epsilon)$$

得到 $f(1) = 0$ 。这意味着

$$\lim_{\epsilon \rightarrow 0} \frac{f(1 + \epsilon)}{\epsilon} = \lim_{\epsilon \rightarrow 0} \frac{f(1 + \epsilon) - f(1)}{\epsilon} = f'(1).$$

方程(D.7.2)可等价地写为

$$\frac{f(x + x\epsilon) - f(x)}{x\epsilon} = \frac{f(1 + \epsilon)}{x\epsilon}.$$

对 $\epsilon \rightarrow 0$ 取极限可将上述关系转化为一个微分方程

$$f'(x) = \frac{1}{x} f'(1).$$

令 $c = f'(1)$ 。对 $f'(x) = \frac{c}{x}$ 进行积分得到解

$f(x) = c \ln x + K$ ，其中 K 为常数。将其代入初始函数方程，我们得到 $K = 0$. ■

命题D.7.2 任何满足的可微函数 $f : \mathbb{R} \rightarrow \mathbb{R}$

$$f(x + y) = f(x) + f(y), \quad \forall x, y \in \mathbb{R}$$

具有 $f(x) = cx$ 的形式，其中 c 为实常数。

证明：若设 $x = y = 0$ ，方程变为 $f(0) = 2f(0)$ ，因此 $f(0) = 0$ 。若取 $y = \epsilon$ ，则 $f(x + \epsilon) - f(x) = f(\epsilon)$ 。除以 ϵ 并取极限，我们得到

$$\lim_{\epsilon \rightarrow 0} \frac{f(x + \epsilon) - f(x)}{\epsilon} = \lim_{\epsilon \rightarrow 0} \frac{f(\epsilon) - f(0)}{\epsilon},$$

其写为 $f'(x) = f'(0)$ 。积分后，得到 $f(x) = cx + b$ ，其中 $c = f'(0)$ 和 b 为实常数。代入初始方程可得 $b = 0$ 。因此，光滑加法函数是线性的。■

命题D.7.3 任何满足条件的可微函数 $f : \mathbb{R} \rightarrow (0, \infty)$

$$f(x+y) = f(x)f(y), \quad \forall x, y \in \mathbb{R}$$

具有 $f(x) = e^{cx}$ 的形式，其中 c 为实常数。

证明：对给定方程应用对数函数，我们得到 $g(x+y) = g(x) + g(y)$ ，其中 $g(x) = \ln(f(x))$ 。然后命题D.7.1得出 $g(x) = cx$ ，其中 $c \in \mathbb{R}$ 。因此， $f(x) = e^{g(x)} = e^{cx}$ 。 ■

注记D.7.4 可以证明，对于 f 仅为连续这一更为严格的假设，命题 D.7.2 和 D.7.3 成立。

D.8 布朗运动

4.13.2节使用了布朗运动的概念。对于该主题的简要介绍，读者可参考[20]。对于更专业的主题，读者可参考[37]。

定义D.8.1 布朗运动是一个随机过程 $W_t, t \geq 0$ ，它满足：

- (i) $W_0 = 0$ (该过程从原点开始)；
- (ii) 如果 $0 \leq u < t < s$ ，那么 $W_s - W_t$ 和 $W_t - W_u$ 是独立的 (该过程具有独立增量)；
- (iii) $t \rightarrow W_t$ 是连续的；
- (iv) 增量服从正态分布，均值为 $W_t - W_s \sim \mathcal{N}(0, |t-s|)$ 。

还要注意 $\mathbb{E}[W_t] = 0$, $\mathbb{E}[W_t^2] = t$ 和 $\text{Cov}(W_t, W_s) = \min\{s, t\}$ 。
伊藤公式 如果 X_t 是一个满足以下条件的随机过程

$$dX_t = b(X_t)dt + \sigma(X_t)dW_t,$$

其中 b 和 σ 是可测函数，且 $F_t = f(X_t)$ ， f 可微，则

$$dF_t = [b(X_t)f'(X_t) + \frac{1}{2}\sigma(X_t)^2f''(X_t)]dt + \sigma f'(X_t)dW_t.$$

某种逆命题由 Dynkin 公式给出：考虑伊藤扩散

$$dX_t = b(X_t)dt + \sigma(X_t)dW_t, \quad X_0 = x.$$

那么对于任何 $f \in C_0^2(\mathbb{R}^n)$ ，我们有

$$\mathbb{E}^x[f(X_t)] = f(x) + \mathbb{E}^x \left[\int_0^t \mathcal{A}f(X_s) ds \right],$$

条件期望所在之处为

$$\mathbb{E}^x[f(X_t)] = \mathbb{E}[f(X_t)|X_0 = x]$$

并且 A 是 X_t 的无穷小生成元。这意味着

$$\mathcal{A}f(x) = \lim_{t \searrow 0} \frac{\mathbb{E}^x[f(X_t)] - f(x)}{t}.$$

附录E 泛函分析

本节介绍本书所需的泛函分析结果的基本要点。对更多细节感兴趣的读者可参考鲁丁的著作[106]。

E.1 巴拿赫空间

本节讨论一个兼具拓扑结构和代数结构的数学对象。设 $(\mathcal{X}, +, \cdot)$ 为线性向量空间，其中“+”表示 \mathcal{X} 中元素的加法，“.”表示与实标量的乘法。

\mathcal{X} 上的范数是一个实值函数 $\|\cdot\|: \mathcal{X} \rightarrow \mathbb{R}$ ，满足以下性质：

- (i) $\|x\| \geq 0$ ，且 $\|x\| = 0 \Leftrightarrow x = 0$ ；
- (ii) $\|\alpha x\| = |\alpha| \|x\|, \forall \alpha \in \mathbb{R}$ ；

(iii) $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in \mathcal{X}$.

对 $(\mathcal{X}, \|\cdot\|)$ 称为赋范空间。该范数诱导出度量 $d(x, y) = \|x - y\|$ ，因此 (V, d) 成为度量空间。巴拿赫空间是在此度量下完备的赋范向量空间。

值得注意范数的序列连续性：如果 $x_n \rightarrow x$ ，那么 $\|x_n\| \rightarrow \|x\|$ ，对于 $n \rightarrow \infty$ 。

以下给出了一些巴拿赫空间的例子：

1. 配备欧几里得范数 $\|x\| = (x_1^2 + \cdots + x_n^2)^{1/2}$ 的 n 维实向量空间 \mathbb{R}^n 构成一个巴拿赫空间。

2. 设 $K \subset \mathbb{R}^n$ 为一个紧致集。在 K 上所有连续实值函数构成的空间 $C(K)$ ，配备范数 $\|f\| = \max_{x \in K} |f(x)|$ ，形成一个巴拿赫空间。
3. 令 $p \geq 1$ 并考虑 $L^p[0, 1] = \left\{ f; \int_0^1 |f|^p < \infty \right\}$ ，其中 f 表示几乎处处相等的所有可测函数的类。关于函数加法和实数乘法，这是一个向量空间。范数由 $\|f\| = \|f\|_p = \left(\int_0^1 |f|^p \right)^{1/p}$ 给出。 $\|f\| = 0$ 意味着几乎处处 $f = 0$ 这一事实与将该空间定义为几乎处处相等的函数类的空间是一致的。根据里斯 - 费希尔定理， $L^p[0, 1]$ 是一个巴拿赫空间。
4. 考虑 $L^\infty[0, 1]$ 为 $[0, 1]$ 上几乎处处有界的可测函数的空间。关于函数加法和实数乘法，这是一个向量空间。范数由 $\|f\| = \|f\|_\infty = \inf \sup g$ 给出。

E. 2 线性算子

对该主题感兴趣的读者可参考 comprehensive book [36]。设 \mathcal{X} 和 \mathcal{Y} 为两个向量空间。映射 $T : \mathcal{X} \rightarrow \mathcal{Y}$ 若满足以下条件，则称为线性算子：

$$T(a_1x_1 + a_2x_2) = a_1T(x_1) + a_2T(x_2), \quad \forall a_i \in \mathbb{R}, \forall x_i \in \mathcal{X}.$$

现假设 \mathcal{X} 和 \mathcal{Y} 为赋范向量空间。若存在常数 $M > 0$ 使得线性算子 T 满足以下条件，则称其为有界的：

$$\|Tx\| \leq M\|x\|, \quad \forall x \in \mathcal{X}.$$

最小的这样的 M 称为算子的范数，记为 $\|T\|$ 。我们还有等价定义：

$$\|T\| = \sup_{x \in V \setminus \{0\}} \frac{\|Tx\|}{\|x\|} = \sup_{\|x\|=1} \|Tx\| = \sup_{\|x\|\leq 1} \|Tx\|.$$

由于

$$\|Tx_1 - Tx_2\| \leq M\|x_1 - x_2\|, \quad \forall x_1, x_2 \in \mathcal{X},$$

由此可知有界线性算子是一致连续的，因此是连续的。反之，如果线性算子 T 仅在一点连续，那么它是有界的。

可以证明，具有 \mathcal{Y} 巴拿赫空间的有界线性算子空间 $T : \mathcal{X} \rightarrow \mathcal{Y}$ 也是一个巴拿赫空间。

如果空间是 $\mathcal{Y} = \mathbb{R}$ ，线性算子 T 就称为线性泛函。特别地，有界线性泛函空间构成一个巴拿赫空间。

E. 3 哈恩 - 巴拿赫定理

向量空间 V 上的凸泛函是一个函数 $p : \mathcal{X} \rightarrow \mathbb{R}$ ，使得

- (i) $p(x+y) \leq p(x) + p(y)$ ，即 p 是次可加的；
- (ii) 对于每个 $\alpha \geq 0$ ， $p(\alpha x) = \alpha p(x)$ ，即 p 是正齐次的。

示例E.3.1 设 $\mathcal{X} = \mathbb{R}^n$ 并考虑 $p(x) = \max_{1 \leq i \leq n} |x_i|$ ，其中 $x = (x_1, \dots, x_n)$ 。那么 $p(x)$ 是 \mathbb{R}^n 上的一个凸泛函。

如果 $(\mathcal{X}, +, \cdot)$ 是一个线性空间，子集 $\mathcal{X}_0 \subset \mathcal{X}$ 是一个线性子空间当且仅当 \mathcal{X}_0 关于从 \mathcal{X} 赋予的运算封闭。因此， \mathcal{X}_0 关于这些运算成为一个线性空间。

以下结果涉及线性泛函从子空间到整个空间的扩展，使得某些性质得以保留。

定理E.3.1 (哈恩 - 巴拿赫) 设 \mathcal{X} 为线性实向量空间， \mathcal{X}_0 为线性子空间， p 为 \mathcal{X} 上的线性凸泛函，且 $f : \mathcal{X}_0 \rightarrow \mathbb{R}$ 为线性泛函，使得对于所有 $x \in \mathcal{X}_0$ ，有 $f(x) \leq p(x)$ 。

那么存在一个线性泛函 $F : \mathcal{X} \rightarrow \mathbb{R}$ ，使得

- (i) $F_{\mathcal{X}_0} = f$ (F 限制到 \mathcal{X}_0 上为 f)。
- (ii) 对于所有 $x \in \mathcal{X}$ ， $F(x) \leq p(x)$ 。

接下来我们给出哈恩 - 巴拿赫定理的一些应用。

1. 设 $p : \mathcal{X} \rightarrow [0, +\infty)$ 为非负凸泛函， $x_0 \in \mathcal{X}$ 为固定元素。那么在 \mathcal{X} 上存在一个线性泛函 F ，使得对于 \mathcal{X} 中的所有 x ，有 $F(x_0) = p(x_0)$ 且 $F(x) \leq p(x)$ 。
2. 设 x_0 为赋范空间 \mathcal{X} 中的一个元素。那么在 \mathcal{X} 上存在一个有界线性泛函 F ，使得 $F(x_0) = \|F\| \|x_0\|$ 。
3. 设 \mathcal{S} 为赋范线性空间 \mathcal{X} 的一个线性子空间，且 y 为 \mathcal{X} 中的一个元素，其到 \mathcal{S} 的距离至少为 δ ，即

$$\|y - s\| \geq \delta, \quad \forall s \in \mathcal{S}.$$

那么在 \mathcal{X} 上存在一个有界线性泛函 f ，满足 $\|f\| \leq 1, f(y) = \delta$ ，并且对于所有的 $s \in \mathcal{S}$ 都有 $f(s) = 0$ 。

E. 4 希尔伯特空间

一个希尔伯特空间 H 是一个配备了满足以下条件的函数 $(,) : H \times H \rightarrow \mathbb{R}$ 的巴拿赫空间：

- (i) $(x, x) = \|x\|^2$ ；
- (ii) $(x, y) = (y, x)$ ；
- (iii) $(c_1 x_1 + c_2 x_2, y) = c_1 (x_1, y) + c_2 (x_2, y), c_i \in \mathbb{R}, x_i, y \in H$ 。

示例E.4.1 $H = \mathbb{R}^n$ ，带有 $(x, y) = \sum_{i=1}^n x_i y_i$ 。

示例E.4.2 $H = L^2 [0, 1]$ ，带有 $(x, y) = \int_0^1 x(t) y(t) dt$ 。

柯西不等式表明 $|(x, y)| \leq \|x\| \|y\|$ ，所以线性泛函 $g(x) = (x, y)$ 是有界的，因此从 H 到 \mathbb{R} 是连续的。所以，如果 $x_n \rightarrow x$ 在 H 中，那么 $(x_n, y) \rightarrow (x, y)$ ，因为 $n \rightarrow \infty$ 。

如果 $(x, y) = 0$ ，则两个元素 $x, y \in H$ 被称为正交的。如果 \mathcal{U} 中任意两个不同元素都是正交的，那么集合 \mathcal{U} 被称为正交系。

示例 E.4.3 集合 $\{1, \cos t, \sin t, \dots, \cos nt, \sin nt, \dots\}$ 是 $H = L^2 [-\pi, \pi]$ 的正交系。

如果对于所有 $x \in \mathcal{U}$ 都有 $\|x\| = 1$ ，则正交系 \mathcal{U} 称为标准正交系。

示例 E.4.4 以下集合

$$\left\{ \frac{1}{2\pi}, \frac{1}{\sqrt{\pi}} \cos t, \frac{1}{\sqrt{\pi}} \sin t, \dots, \frac{1}{\sqrt{\pi}} \cos nt, \frac{1}{\sqrt{\pi}} \sin nt, \dots \right\}$$

是 $H = L^2 [-\pi, \pi]$ 的正交系。

设 $\{x_1, x_2, \dots\}$ 为 H 中的一个可数正交系。元素 $x \in H$ 关于先前系统的傅里叶系数由 $c_k = (x, x_k)$ 给出。则贝塞尔不等式表明

$$\sum_{k \geq 1} c_k^2 \leq \|x\|^2.$$

希尔伯特空间的线性子空间 \mathcal{X}_0 ，如果它包含 \mathcal{X}_0 中任何收敛序列 (x_n) 的极限，则称其为闭子空间。在这种情况下，对于任何元素 x ，数 $d(x, \mathcal{X}_0) = \inf \{\|x - y\|; y \in \mathcal{X}_0\}$ 称为 x 到子空间 \mathcal{X}_0 的距离。

定理E.4.1 设 \mathcal{X}_0 为希尔伯特空间 \mathcal{X} 的一个闭线性子空间。那么对于 \mathcal{X} 中的任意元素 x ，存在一个元素 $x_0 \in \mathcal{X}_0$ ，使得 $\|x - x_0\|$ 等于 x 到 \mathcal{X}_0 的距离。

元素 x_0 被称为 x 在子空间 \mathcal{X}_0 上的投影。

E. 5表示定理

本节给出了不同空间上线性泛函的表示，这对于证明神经网络的通用逼近性质是必要的。它们中的大多数在文献中通常被引述为“里斯表示定理”。

例如，读者可以在哈尔莫斯的著作[51]或罗伊登的著作[104]中找到证明。

下一个结果是关于希尔伯特空间上有界线性泛函的一种表示。

定理E.5.1 (里斯) 设 $f : H \rightarrow \mathbb{R}$ 是赋有内积(\cdot, \cdot)的希尔伯特空间 H 上的一个有界线性泛函。那么存在唯一的元素 $y \in H$ ，使得对于所有的 $x \in H$ 都有 $f(x) = (x, y)$ 。此外， $\|f\| = \|x\|$ 。

特别地，如果 $F : L^2[0, 1] \rightarrow \mathbb{R}$ 是一个有界线性泛函，那么存在唯一的 $g \in L^2[0, 1]$ 使得 $F(f) = \int_0^1 f(t)g(t)dt$ 。

即使对于 $p \neq 2$ 而言 L^p 不是希尔伯特空间（不过它是一个完备空间），这个结果仍然可以扩展为关于空间 L^p 的一个表示结果。

定理E.5.2 (里斯) 设 F 是 $L^p[0, 1]$ 上的有界线性泛函，满足 $1 < p < \infty$ 。那么存在唯一的函数 $g \in L^q[0, 1]$ ，满足 $\frac{1}{p} + \frac{1}{q} = 1$ ，使得 $F(f) = \int_0^1 f(t)g(t)dt$ 。我们还有 $\|F\| = \|g\|_q$ 。

对于 $p = 1$ 的情况，通过一些小的修改，先前的结果仍然成立。设 $L^\infty[0, 1]$ 为可测且几乎处处有界的函数空间，它配备上确界范数 $\|\cdot\|_\infty$ 后成为一个巴拿赫空间，见E.1节的第4点。

定理E.5.3 设 F 是 $L^1[0, 1]$ 上的有界线性泛函。那么存在唯一的函数 $g \in L^\infty[0, 1]$ ，使得

$$F(f) = \int_0^1 f(t)g(t) dt, \quad \forall f \in L^1[0, 1].$$

我们还有 $\|F\| = \|g\|_\infty$ 。

定义在区间 $[a, b]$ 上的实值函数 g 具有有界变差，如果对于任何划分

$$a = x_0 < x_1 < \cdots < x_n = b,$$

和

$$\sum_{k=0}^{n-1} |g(x_{k+1}) - g(x_k)|$$

小于给定常数。这些和在 $[a, b]$ 的所有可能划分上的上极限称为 g 的全变差，并用 $\bigvee_a^b (g)$ 表示

示例 E.5.1 在 $[a, b]$ 上的递增函数 g 具有有界变差，且 $\bigvee_a^b (g) = g(b) - g(a)$ 。

以下结果是针对非连续函数的阿尔泽拉 - 阿斯克利定理的类似情况，可用于从给定集合中提取一个收敛的函数序列。

定理 E.5.4 (赫利) 设 \mathcal{K} 是从 $[a, b]$ 到 \mathbb{R} 的无穷函数集，使得：

(i) \mathcal{K} 一致有界，即存在 $\exists C > 0$ 使得 $\sup_{x \in [a, b]} |f(x)| < C$ ，对所有 $f \in \mathcal{K}$ ；

(ii) $\exists V > 0$ 使得对于所有的 $f \in \mathcal{K}$ 都有 $\bigvee_a^b (f) \leq V$ 。

那么我们可以在 \mathcal{K} 中选择一个函数序列 $(f_n)_n$ ，它在每个点 $x \in [a, b]$ 处都收敛。

回想一下， $C[0, 1]$ 是定义在 $[0, 1]$ 上的实值连续函数空间，并且相对于范数 $\|\cdot\|_\infty$ 它成为一个巴拿赫空间。 $C([0, 1])$ 上最简单的连续线性泛函是 $F(f) = f(t_0)$ ，它在固定点 t_0 处赋予 f 其值。下一个结果表明，这些类型泛函的一般形式是通过上述特定函数的斯蒂尔杰斯组合得到的。

定理E.5.5 设 F 是 $C([0, 1])$ 上的连续线性泛函。存在一个有界变差函数 $g : [0, 1] \rightarrow \mathbb{R}$ ，使得

$$F(f) = \int_0^1 f(t) dg(t), \quad \forall f \in C([0, 1]).$$

我们也有 $\|F\| = \bigvee_0^1 (f)$ 。

我们回顾一下，斯蒂尔杰斯积分 $\int_0^1 f(t) dg(t)$ 被定义为黎曼型和的极限

$$\sum_{k=0}^{m-1} f(x_k)[g(x_{k+1}) - g(x_k)]$$

当分割 $0 = x_0 < x_1 < \dots < x_m = 1$ 的范数趋于零时。

下一个结果是前一个结果的推广。设 K 表示 \mathbb{R}^n 中的一个紧集，并用 $C(K)$ 表示 K 上实值连续函数的集合。

定理E.5.6 设 F 是 $C(K)$ 上的有界线性泛函。则在 K 上存在唯一的有限带号博雷尔测度 μ ，使得

$$F(f) = \int_K f(x) d\mu(x), \quad \forall f \in C(K).$$

此外， $\|F\| = |\mu|(K)$ 。

下一个结果将泛函的有界性条件替换为正性。在这种情况下，带号测度变为测度。

定理E.5.7 设 L 是 $C(K)$ 上的正线性泛函。则在 K 上存在唯一的有限博雷尔测度 μ ，使得

$$L(f) = \int_K f(x) d\mu(x), \quad \forall f \in C(K).$$

E.6 不动点定理

设 (M, d) 是一个度量空间， $T : M \rightarrow M$ 是从 M 到其自身的一个映射。如果 T ，则 $T : M \rightarrow M$ 称为收缩映射

$$d(T(x), T(x')) \leq \lambda d(x, x'), \quad \forall x, x' \in M$$

对于某个小于1的正的常数 λ 。如果度量是由 M 上的一个范数诱导的，即，如果 $d(x, x') = \|x - x'\|$ ，收缩条件可以写成
 $\|T(x) - T(x')\| \leq \lambda \|x - x'\|$ 。

度量空间 (M, d) 中的一个点的序列 (x_n) 被称为柯西序列，如果对于任意的 $\epsilon > 0$ ，存在一个 $N > 1$ 使得 $d(x_n, x_m) < \epsilon$ ，对于所有的 $n, m > N$ 。

度量空间 (M, d) 如果 M 中的任何柯西序列 (x_n) 都是收敛的，即存在 $x^* \in M$ ，使得对于所有 $\epsilon > 0$ ，存在 $N > 1$ ，使得对于所有 $n \geq N$ ， $d(x_n, x^*) < \epsilon$ ，则称其为完备的。

例E.6.1 配备欧几里得距离的空间 \mathbb{R}^n 构成一个完备度量空间。

示例E.6.2 赋予由范数 $\|L\| = \sup_{x \neq 0} \frac{\|Lx\|}{\|x\|}$ 诱导的度量的线性算子空间

$\{L; L : \mathbb{R}^n \rightarrow \mathbb{R}^n\}$ 是一个完备度量空间。

示例E.6.3 连续的 } 的空间 $C[a, b] = \{f : [a, b] \rightarrow \mathbb{R}; f$ 配备度量

$d(f, g) = \sup_{x \in [a, b]} |f(x) - g(x)|$ 是一个完备度量空间。

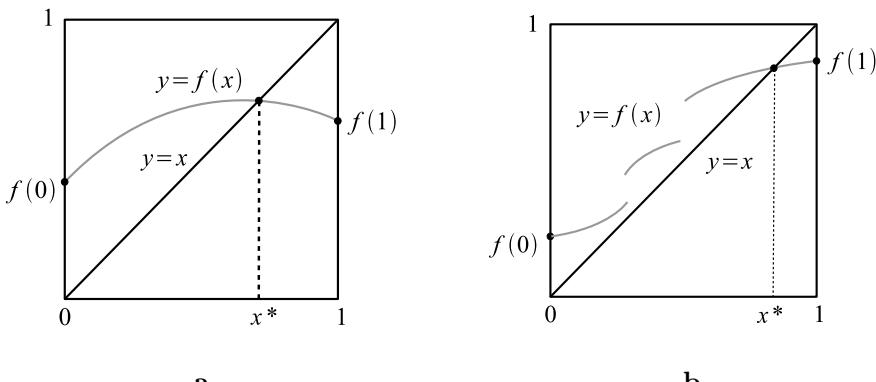


图1: a. 连续函数 $f : [0, 1] \rightarrow [0, 1]$ 的一个不动点。b. 递增函数 $f : [0, 1] \rightarrow [0, 1]$ 的 A 不动点。

如果 $T(x^*) = x^*$ ，则点 $x^* \in M$ 称为应用 $T : M \rightarrow M$ 的不动点。

示例E.6.4 任何连续函数 $f : [0, 1] \rightarrow [0, 1]$ 至少有一个不动点。从几何角度来看，这是因为连接位于正方形相对两侧的任意两点的任何连续曲线都会与对角线 $y = x$ 相交，见图1 a。值得注意的是，克纳斯特证明了在函数 f 单调递增而非连续的情况下，不动点性质也成立，见图1 b。

定理E.6.5 完备度量空间 (M, d) 到自身的压缩映射 T 有唯一的不动点。

对于任意点 $x_0 \in M$ ，由 $x_{n+1} = T(x_n)$ 定义的序列 (x_n) 收敛于不动点 $x_n \rightarrow x^*$ 。此外，我们有如下估计

$$d(x_n, x^*) \leq \frac{\lambda^n}{1-\lambda} d(x_0, x_1).$$

附录F 实分析

F.1 反函数定理

反函数定理指出，从 \mathbb{R}^n 到 \mathbb{R}^n 的连续可微函数在某点满足非零雅可比条件时，该函数在局部是可逆的。我们回忆一下，函数 $F = (F_1, \dots, F_n)$ 的雅可比矩阵是偏导数的 $n \times n$ 矩阵

$$J_F(x) = \left(\frac{\partial F_i(x)}{\partial x_j} \right)_{i,j}.$$

定理F.1.1 设 $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ 为连续可微函数， $p \in \mathbb{R}^n$ 为满足 $\det J_F(p) \neq 0$ 的点。那么存在两个分别包含点 p 和 $q = F(p)$ 的开集 \mathcal{U} 和 \mathcal{V} ，使得 $F|_{\mathcal{U}} : \mathcal{U} \rightarrow \mathcal{V}$ 是可逆的，其逆函数连续可微。逆函数的雅可比矩阵由 $J_{F^{-1}}(q) = [J_F(p)]^{-1}$ 给出。

该定理可以用数学术语重新表述为：在某点处具有非奇异雅可比矩阵的连续可微函数在该点周围是局部微分同胚。

表述该定理的另一种方式是用非线性方程组。考虑具有 n 个未知数的 n 个方程的系统

$$\begin{aligned} F_1(x_1, \dots, x_n) &= y_1 \\ \cdots &\quad \cdots \quad \cdots = \cdots \\ F_n(x_1, \dots, x_n) &= y_n, \end{aligned}$$

并假设存在一个点 $x^0 \in \mathbb{R}^n$ 使得 $\det\left(\frac{\partial F_i}{\partial x_j}\right)(x^0) \neq 0$ 。那么存在关于 x^0 和 $y^0 = F(x^0)$ 的两个开集 \mathcal{U} 和 \mathcal{V} ，只要 $x \in \mathcal{U}$ 和 $y \in \mathcal{V}$ ，该系统就能以唯一解求解。这意味着存在 n 个连续可微函数 $G_i : \mathcal{V} \rightarrow \mathbb{R}$ 使得

$$\begin{aligned} x_1 &= G_1(y_1, \dots, y_n) \\ \dots &= \dots \quad \dots \quad \dots \\ x_n &= G_n(y_1, \dots, y_n) \end{aligned}$$

对于任何 $y = (y_1, \dots, y_n) \in \mathcal{V}$ 。

我们注意到这是一个存在性结果，它并没有明确构造出该系统的解。然而，在函数为线性的特殊情况下，即 $F(x) = Ax$ ，其中 A 为非奇异方阵时，线性系统 $Ax = y$ 有唯一解 $x = A^{-1}y$ 。在这种情况下，解是全局的，因为雅可比矩阵 $J_F(x) = A$ 在任何地方的行列式都不为零。

F.2 广义意义下的微分

我们如何对一个并非处处可微的函数进行微分呢？如果我们有一个分段可微的函数，对其进行分段微分并不总是能得到正确结果，因为我们在接触点处找不到导数。一个不可微函数的“导数”有时可能在所谓的广义意义下存在。设 $f : \mathbb{R} \rightarrow \mathbb{R}$ 为一个函数。如果满足以下条件，我们就说 g 是函数 f 在广义意义下的导数：

$$\int_{\mathbb{R}} g(x)\varphi(x) dx = - \int_{\mathbb{R}} f(x)\varphi'(x) dx, \quad (\text{F.2.1})$$

对于任何具有紧支集的光滑函数 φ 。我们注意到广义微分是经典微分的一种扩展，因为前面的关系变成了熟悉的分部积分公式

$$\int_{\mathbb{R}} f'(x)\varphi(x) dx = - \int_{\mathbb{R}} f(x)\varphi'(x) dx.$$

例F.2.1 设 $f(x) = H(x)$ 为海维赛德阶跃函数。那么它的导数由狄拉克函数 $f'(x) = \delta(x)$ 给出。接下来我们将通过计算关系式(F.2.1)的两边来验证它：

$$\int_{\mathbb{R}} \delta(x)\varphi(x) dx = \int_{\mathbb{R}} \varphi(x)\delta(dx) = \varphi(0)$$

$$-\int_{\mathbb{R}} f(x)\varphi'(x) dx = - \int_{\mathbb{R}} H(x)\varphi'(x) dx = - \int_0^{\infty} \varphi'(x) dx = \varphi(0) - \varphi(\infty) = \varphi(0).$$

类似地， $H(x-a)$ 的导数是 $\delta_a(x)$ ，其中 $\delta_a(x) = \delta(x-a)$ 。

示例 F.2.2 $\text{ReLU}(x)$ 的导数是海维赛德函数 $\text{ReLU}'(x) = H(x)$ 。我们可以使用关系式 (F.2.1) 来验证：

$$\begin{aligned}\int_{\mathbb{R}} \text{ReLU}'(x)\varphi(x) dx &= - \int_{\mathbb{R}} \text{ReLU}(x)\varphi'(x) dx = - \int_0^{\infty} x\varphi'(x) dx \\ &= \int_0^{\infty} x'\varphi(x) dx = \int_0^{\infty} \varphi(x) dx = \int_{\mathbb{R}} H(x)\varphi(x) dx,\end{aligned}$$

对于任意的 $\varphi \in C_0^{\infty}(\mathbb{R})$ 。因此，从广义上讲， $\text{ReLU}'(x) = H(x)$ 。

F.3 函数序列的收敛性

设 $f_n : \mathbb{R} \rightarrow \mathbb{R}$ 为一个函数序列。那么 f_n 可以通过几种方式逼近函数 f 。

1. 若对于任意 $x \in \mathbb{R}$ ， $f_n(x)$ 收敛于 $f(x)$ ，则函数序列 $(f_n)_n$ 逐点收敛于 f 。
2. 设 $f \in L^2(\mathbb{R})$ 。若当 $n \rightarrow \infty$ 时， $\|f_n - f\|_2 \rightarrow 0$ ，则函数序列 $(f_n)_n L^2$ 收敛于 f 。这意味着

$$\lim_{n \rightarrow \infty} \int_{\mathbb{R}} |f_n(x) - f(x)|^2 dx = 0.$$

3. 若序列 $(f_n)_n$ 弱收敛于 f ，则

$$\lim_{n \rightarrow \infty} \int_{\mathbb{R}} f_n(x)\varphi(x) dx = \int_D f(x)\varphi(x) dx, \quad \forall \varphi \in C_0^{\infty}(\mathbb{R}).$$

值得注意的是， L^2 收敛意味着逐点收敛和弱收敛。后者可由柯西不等式的应用得出

$$\begin{aligned}\left| \int_{\mathbb{R}} f_n \varphi - \int_{\mathbb{R}} f \varphi \right| &= \left| \int_{\mathbb{R}} (f_n - f) \varphi \right| \leq \int_{\mathbb{R}} |f_n - f| |\varphi| \\ &= \|f_n - f\|_2 \|\varphi\|_2\end{aligned}$$

以及夹逼定理。

附录G 线性代数

G.1 特征值、范数和逆矩阵

考虑一个具有 n 行和 m 列的矩阵 $A \in \mathcal{M}_{n,m}$ ，由 $A = (a_{ij})$ 给出。矩阵 A 的转置，记为 A^T ，由 $A^T = (a_{ji})$ 给出，并满足 $(A^T)^T = A$, $(AB)^T = B^T A^T$ 。如果 $A = A^T$ ，则该矩阵称为对称矩阵。对称的一个必要条件是 A 必须是方阵，即 $n = m$ 。如果 $AA^T = \mathbb{I}$ （单位矩阵），则该矩阵称为正交矩阵。

一个数 λ （实数或复数）被称为方阵 A 的一个特征值，如果存在一个非零向量 \mathbf{x} 使得 $A\mathbf{x} = \lambda\mathbf{x}$ 。向量 \mathbf{x} 被称为矩阵 A 的一个特征向量。特征值是多项式方程 $\det(A - \lambda\mathbb{I}) = 0$ 的解。

命题G.1.1 设 A 为对称矩阵。那么 A 有 n 个实特征值，不一定不同， $\lambda_1, \dots, \lambda_n$ ，以及 n 个特征向量， $\mathbf{x}_1, \dots, \mathbf{x}_n$ ，它们在 \mathbb{R}^n 中构成一个正交基。

类型为 $n \times 1$ 的矩阵称为向量。如果 w 和 b 是两个向量，那么 $(w^T b) = (wb^T) = \langle w, b \rangle$ ，其中 $\langle \cdot, \cdot \rangle$ 是 \mathbb{R}^n 上的欧几里得标量积。

可以在 \mathbb{R}^n 上定义几种范数。设 $\mathbf{x}^T = (x_1, \dots, x_n)$ 为一个向量。那么

$$\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|, \quad \|\mathbf{x}\|_2 = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}, \quad \|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|$$

本书中使用了三种范数。可以证明，存在两个常数 $C_1, C_2 > 0$ ，使得

$$C_1\|\mathbf{x}\|_1 \leq \|\mathbf{x}\|_2 \leq C_2\|\mathbf{x}\|_\infty, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

从几何角度来看，这意味着任何 $\|\cdot\|_2$ 球都可以同时包含在一个 $\|\cdot\|_\infty$ 球中，并且还包含一个 $\|\cdot\|_1$ 球。

前面的每个范数都为平方矩阵 A 诱导出一个范数。受线性算子范数的启发，我们定义

$$\|A\| = \sup_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}, \quad (\text{G.1.1})$$

其中 $\|\cdot\|$ 是 \mathbb{R}^n 上上述任何一种范数。我们注意到对于任何其他方阵 B ，不等式 $\|A\mathbf{x}\| \leq \|A\| \|\mathbf{x}\|$ 和 $\|AB\| \leq \|A\| \|B\|$ 成立。由前面三种范数诱导出的范数是

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|, \quad \|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|, \quad \|A\|_2 = \sqrt{\rho(A^T A)},$$

其中 ρ 表示矩阵的谱半径，由 $\rho(A) = \max |\lambda_j|$ 给出，其中 λ_j 表示 A 的特征值。矩阵 $B = A^T A$ 是对称的，根据命题G.1.1，它有实特征值。由于 A 和 A^T 有相等的特征值， $A^T A$ 的特征值是 λ_i^2 ，因此 $A^T A$ 是正定的。所以， $\rho(A^T A) = \max \lambda_i^2$ ，进而 $\|A\|_2 = \max |\lambda_i|$ 是矩阵 A 特征值的最大绝对值。

值得注意的是，如果 A 是对称的，那么 $\|A\|_2 = \rho(A)$ 。此外， $\|A\|_2$ 是由公式(G.1.1)生成的所有范数中最小的，即 $\|A\|_2 \leq \|A\|$ 。这可以如下证明。设 λ 是绝对值最大的特征值， \mathbf{x} 是长度为1的特征向量。那么

$$\|A\|_2 = |\lambda| = \|\lambda\mathbf{x}\| = \|A\mathbf{x}\| \leq \|A\| \|\mathbf{x}\| \leq \|A\|.$$

我们还有 $\|A\|_2 \geq \frac{1}{n}|\text{Tr}(A)|$ 和 $\|A\|_2 \geq |\det A|^{1/n}$ 。

命题G.1.2 设 A 是一个方阵。

(i) 当 $m \rightarrow \infty$ 时，幂矩阵 A^m 收敛于零矩阵，当且仅当 $\rho(A) < 1$ 。

(ii) 如果 $\rho(A) < 1$ ，那么 $\mathbb{I} - A$ 是可逆的，并且我们有

$$(\mathbb{I} - A)^{-1} = \mathbb{I} + A + A^2 + \cdots + A^m + \cdots \quad (\text{G.1.2})$$

(iii) 如果几何级数(G.1.2)收敛，那么 $\rho(A) < 1$ 。

推论G.1.3 设 A 为方阵， $\|\cdot\|$ 为范数。如果 $\|A\| < 1$ ，那么 $\mathbb{I} - A$ 是可逆的，其逆由(G.1.2)给出，并且

$$\|(\mathbb{I} - A)^{-1}\| < \frac{1}{1 - \|A\|}.$$

在下面我们将处理求两个方阵之和的逆矩阵的过程。我们首先在一维情况下解释这个想法。假设 $a_1, a_2 \in \mathbb{R} \setminus \{0\}$ 是两个实数，且 $a_1 + a_2 \neq 0$ 。一个简单的代数计算得出

$$\begin{aligned}\frac{1}{a_1 + a_2} &= \frac{1}{a_2} - \frac{1}{a_1 + a_2} \frac{a_1}{a_2} \\ &= \frac{1}{a_1} - \frac{1}{a_1 + a_2} \frac{a_2}{a_1}.\end{aligned}\tag{G.1.3}$$

我们构造两个由 $f(x) = -\frac{a_1}{a_2}x + \frac{1}{a_2}$ 和 $g(x) = -\frac{a_2}{a_1}x + \frac{1}{a_1}$ 给出的线性函数 $f, g : \mathbb{R} \rightarrow \mathbb{R}$ ，并考虑两种情况：

1. 如果 $|a_1| < \lambda |a_2|$ ，且 $0 < \lambda < 1$ ，那么

$$|f(x) - f(x')| = \left| \frac{a_1}{a_2} \right| |x - x'| < \lambda |x - x'|.$$

因此， f 是完备度量空间 $(\mathbb{R}, |||)$ 到自身的一个压缩映射，因此它有唯一的不动点，见定理E.6.5。不动点 x^* 满足 $f(x^*) = x^*$ ，并由 $x^* = \frac{1}{a_1 + a_2}$ 给出。它的近似序列是 (x_n) ，由递推式 $\overline{x_{n+1}} = f(x_n)$ 定义，且 $x_0 = 0$ 。误差估计由 $|x_n - x^*| < \frac{\lambda^n}{1-\lambda} |x_1 - x_0| = \frac{\lambda^n}{(1-\lambda)} \frac{1}{|a_2|}$ 给出。

2. 如果 $|a_2| < \lambda |a_1|$ ，且 $0 < \lambda < 1$ ，那么 $|g(x) - g(x')| < \lambda |x - x'|$ ，所以 g 是 \mathbb{R} 的一个压缩映射，因此它有唯一的不动点，即 $\frac{1}{a_1 + a_2}$ 。细节与第一种情况类似。

现在考虑两个可逆的 $n \times n$ 矩阵， A_1, A_2 。我们断言

$$(A_1 + A_2)^{-1} = A_2^{-1} - (A_1 + A_2)^{-1} A_1 A_2^{-1},\tag{G.1.4}$$

这是矩阵情形下 (G.1.3) 的类似关系。如下通过在两边简单地乘以 $(A_1 + A_2)$ 即可证明：

$$\begin{aligned}\mathbb{I} &= (A_1 + A_2) A_2^{-1} - A_1 A_2^{-1} \iff \\ \mathbb{I} &= A_1 A_2^{-1} + \mathbb{I} - A_1 A_2^{-1}.\end{aligned}$$

假设 $\|A_1 A_2^{-1}\| < 1$ ，由此可得 $\mathbb{I} + A_1 A_2^{-1}$ 是可逆的，见推论G.1.3。然后从关系式(G.1.4)中求解 $(A_1 + A_2)^{-1}$ 可得

$$(A_1 + A_2)^{-1} = A_2^{-1} (\mathbb{I} + A_1 A_2^{-1})^{-1}.\tag{G.1.5}$$

这个关于两个矩阵之和的逆的闭式公式在实际中不能使用。出于计算原因，我们考虑以下两种方法：

1. 考虑映射 $f : \mathcal{M}_{n \times n} \rightarrow \mathcal{M}_{n \times n}$, $f(M) = A_2^{-1} - MA_1A_2^{-1}$ 。由于

$$\|f(M) - f(M')\| = \|(M' - M)A_1A_2^{-1}\| \leq \|M' - M\|\|A_1A_2^{-1}\| < \lambda\|M - M'\|,$$

那么 f 是 $\mathcal{M}_{n \times n}$ 到自身的一个压缩映射。空间 $\mathcal{M}_{n \times n}$ 是完备的，因为任何矩阵都与一个线性算子相关联，并且 \mathbb{R}^n 上的线性算子空间是完备的。根据不动点定理，映射 f 有一个唯一的不动点 M^* ，即 $f(M^*) = M^*$ 。由(G.1.4)可得 $M^* = (A_1 + A_2)^{-1}$ 。这个逆可以由矩阵序列 (M_n) 逼近， (M_n) 由 $M_{n+1} = f(M_n)$, $M_0 = \mathbb{O}$ 给出。误差估计如下

$$\|M_n - M^*\| < \frac{\lambda^n}{1 - \lambda} \|M_1 - M_0\| = \frac{\lambda^n}{(1 - \lambda)} \|A_2^{-1}\|.$$

2. 另一种近似 $(A_1 + A_2)^{-1}$ 的方法是将(G.1.4)展开为级数，见命题G.1.2

$$(A_1 + A_2)^{-1} = A_2^{-1}(\mathbb{I} + A_1A_2^{-1})^{-1} = A_2^{-1} \sum_{k \geq 0} (-1)^k (A_1A_2^{-1})^k.$$

之前的计算是在 $\|A_1A_2^{-1}\| < 1$ 条件下进行的。这意味着对于所有 $i \in 1, \dots, n$ ，有 $\rho(A_1A_2^{-1}) < 1$ ，或 $\rho(A_1) < \rho(A_2)$ ，或 $\lambda_i(A_1) < \lambda_i(A_2)$ ，即矩阵 A_1 的特征值分别比 A_2 小。

值得注意的是，由于对称性原因， A_1 和 A_2 的角色可以互换，并且如果假设 A_1 可逆，也可以得到类似的公式。

以下结果的证明仅通过简单乘法即可得出。

引理G.1.4 (矩阵求逆引理) 设 $A, B \in \mathcal{M}_{m \times n}$ 为正定矩阵， $C \in \mathcal{M}_{m \times n}$, $D \in \mathcal{M}_{n \times n}$ 为正定矩阵。如果

$$A = B^{-1} + CD^{-1}C^T$$

那么

$$A^{-1} = B - BC(D + C^TBC)^{-1}C^T B.$$

G. 2 摩尔 - 彭罗斯伪逆

如果一个线性系统的方程数量多于未知数数量，则称其为超定系统。通常，这类系统没有任何解。摩尔 - 彭罗斯伪逆方法提供了一个近似解，[89, 97]，从所有实际应用的角度来看，在本节末尾指定的某种意义上是很好的。

我们从考虑矩阵形式的线性系统 $AX = b$ 开始，其中 A 是一个 $m \times n$ 矩阵， $m > n$ （行数多于列数）， X 是一个 n 维未知向量， b 是一个 m 维已知向量。由于 A 不是方阵，在这种情况下逆矩阵 A^{-1} 没有意义。然而，方阵 $A^T A$ 很有可能是可逆的。¹ 例如，如果 A 具有满秩， $\text{rank } A = n$ ，那么秩 $A^T A = \text{rank } A = n$ ，所以 $n \times n$ 矩阵 $A^T A$ 具有最大秩，因此 $\det A^T A \neq 0$ ，即 $A^T A$ 是可逆的。

然后将方程乘以转置矩阵 A^T ，从左边进行运算，我们得到 $A^T A X = A^T b$ 。假设 $A^T A$ 是可逆的，我们得到解

$$X = (A^T A)^{-1} A^T b。A\text{ 的伪逆由 }n \times m\text{ 矩阵定义}$$

$$A^+ = (A^T A)^{-1} A^T。 \quad (\text{G.2.6})$$

在这种情况下，超定系统 $AX = b$ 的摩尔 - 彭罗斯伪逆解由

$$X = A^+ b \text{ 给出。}$$

当 A 可逆时，我们有 $A^+ = A^{-1}$ ，即伪逆是矩阵逆的一种推广。

值得注意的是，如果矩阵 A 的列数多于行数，即 $n > m$ ，那么 $A^T A$ 没有逆矩阵，因为 $\det A^T A = 0$ 。这可由 n 维矩阵 $A^T A$ 的秩的评估得出。

$$\text{rank } A^T A = \text{rank } A \leq \min\{n, m\} = m < n.$$

在这种情况下，伪逆矩阵 A^+ 即使总是存在，也不能用显式公式 (G.2.6) 表示。

几何意义 考虑线性映射 $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ， $F(X) = AX$ ，其中 $n < m$ 。假设矩阵 A 具有满秩， $\text{rank } A = n$ 。在这种情况下， F 的值域是 \mathbb{R}^m 的以下线性子空间：

$$\mathcal{R} = \{AX; X \in \mathbb{R}^n\},$$

维度为 $\dim \mathcal{R} = \text{rank } A = n$ 。

现在，给定一个向量 $b \in \mathbb{R}^m$ ，不一定包含在空间 \mathcal{R} 中，我们尝试使用最小范数解来近似求解线性系统 $AX = b$ 。这是一个向量 $X^* \in \mathbb{R}^n$ ，它使差值 $AX - b$ 的 L^2 范数最小，即

$$X^* = \arg \min_{X \in \mathbb{R}^n} \|AX - b\|_2. \quad (\text{G.2.7})$$

¹这是因为满足代数方程 $\det A^T A = 0$ 的矩阵 A 在 $m \times n$ 矩阵集合中构成一个可忽略的集合。

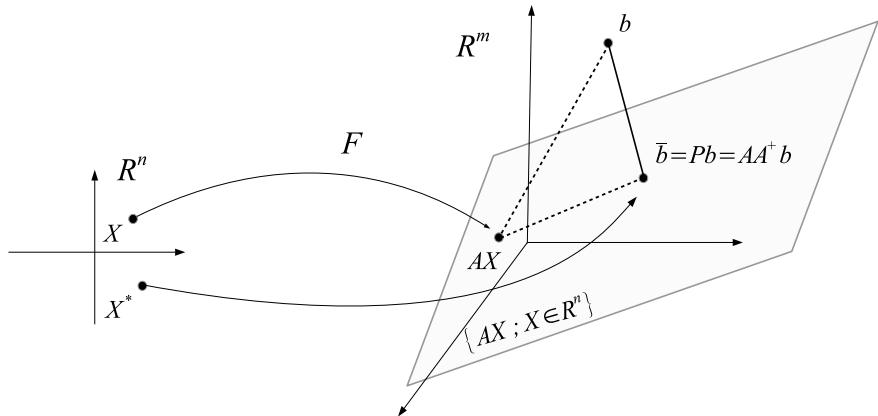


图1：伪逆解 $X^* = A^+b$ 的几何解释。

从几何角度来看，这意味着 AX^* 是空间 \mathcal{R} 中最接近 b 的点，即它是 b 在 \mathcal{R} 上的正交投影，见图1。

令 \bar{b} 表示 b 在空间 \mathcal{R} 上的正交投影，并考虑线性系统 $AX = \bar{b}$ 。由于 $\bar{b} \in \mathcal{R}$ ，该系统确实有解。唯一性由矩阵 A 的最大秩条件得出。因此，存在唯一的向量 $X^* \in \mathbb{R}^n$ 使得 $AX^* = \bar{b}$ 。这就是方程(G.2.7)所声称的解。等价地，这意味着

$$\|AX - b\|_2 \geq \|AX^* - b\|_2 = \|\bar{b} - b\|_2, \quad \forall X \in \mathbb{R}^n.$$

接下来我们关注投影 \bar{b} 的表达式。可以证明 $\bar{b} = AA^+b$ 。这是由于由 $P = AA^+ = A(A^T A)^{-1}A^T$ 给出的线性算子 $P: \mathbb{R}^m \rightarrow \mathbb{R}^m$ 是 \mathbb{R}^m 到子空间 \mathcal{R} 上的正交投影算子。这个结果由以下三个性质推出，这三个性质可以通过直接计算来验证： $P^2 = P$, $P^T = P$ ，以及 $PA = A$ ；前者意味着 P 是一个投影算子，后者意味着 P 是一个正交投影算子，最后一个意味着空间 \mathcal{R} 在 P 作用下是不变的。由于 $X^* = A^+b$ 满足方程 $AX = \bar{b}$ ，即

$$AX^* = A(A^+b) = (AA^+)b = Pb = \bar{b},$$

那么 X^* 代表系统 $AX = b$ 的伪逆解。

摩尔 - 彭罗斯伪逆的第一个应用是在坐标平面 (x_1, y_1) 、 $(x_2, y_2), \dots, (x_m, y_m)$ 中给定的点找到 m 的最佳拟合线。

如果直线的方程为 $y = ax + b$ ，我们将写出以下由 m 个方程组成的超定方程组：

$$\begin{array}{rcl} ax_1 + b & = & y_1 \\ \cdots & = & \cdots \\ ax_m + b & = & y_m, \end{array}$$

它可以写成等价的矩阵形式

$$\underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_m & 1 \end{pmatrix}}_A \begin{pmatrix} a & b \end{pmatrix} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}}_{=Y}$$

我们注意到在这种情况下 $n = 2$ ，因为只有两个参数需要确定。直接计算表明

$$A^T A = \begin{pmatrix} \|x\|^2 & \sum x_i \\ \sum x_i & n \end{pmatrix}, \quad (A^T A)^{-1} = \frac{1}{n\|x\|^2 - (\sum x_i)^2} \begin{pmatrix} n & -\sum x_i \\ -\sum x_i & \|x\|^2 \end{pmatrix}.$$

由于

$$A^T Y = \begin{pmatrix} \sum x_i y_i \\ \sum y_i \end{pmatrix},$$

伪逆解为

$$\begin{pmatrix} a \\ b \end{pmatrix} = A^+ Y = (A^T A)^{-1} A^T Y = \frac{1}{n\|x\|^2 - (\sum x_i)^2} \begin{pmatrix} n \sum x_i y_i - \sum x_i \sum y_i \\ \|x\|^2 \sum y_i - \sum x_i \sum x_i y_i \end{pmatrix}.$$

这给出了回归直线系数的常见表达式

$$a = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}, \quad b = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n \sum x_i^2 - (\sum x_i)^2}.$$

值得注意的是，类似的方法可以应用于多项式回归。

命题G.2.1 设 A 为秩为 n 的 $m \times n$ 矩阵。那么

(i) $A^T A$ 是正定且可逆的；

(ii) $\lim_{t \rightarrow \infty} e^{-A^T A t} = \mathbb{O}_n$ 。

证明： (i) 因为对于任何 $x \in \mathbb{R}^n$ 我们有

$$\langle A^T Ax, x \rangle = x^T A^T Ax = \|Ax\|^2 \geq 0,$$

矩阵 $A^T A$ 是正定的。利用矩阵秩的性质，

$\text{rank}(A^T A) = \text{rank}(A) = n$ ，所以矩阵 $A^T A$ 具有最大秩，因此它是可逆的。

(ii) 部分 (i) 可以表述为矩阵 $A^T A$ 具有正的非零特征值，

$\alpha_j > 0, 1 \leq j \leq n$ 。设 M 是一个可逆的 $n \times n$ 矩阵，它将 $A^T A$ 对角化，即 $A^T A = M \text{Diag}(\alpha_j) M^{-1}$ 。那么 $(A^T A)^k = M \text{Diag}(\alpha_j^k) M^{-1}$ ，因此

$$\begin{aligned} e^{-A^T At} &= \sum_{k \geq 0} (-1)^k (A^T A)^k \frac{t^k}{k!} = M \sum_{k \geq 0} (-1)^k (\text{Diag}(\alpha_j))^k \frac{t^k}{k!} M^{-1} \\ &= M \sum_{k \geq 0} (-1)^k \text{Diag}(\alpha_j^k) \frac{t^k}{k!} M^{-1} = M \text{Diag}\left(\sum_{k \geq 0} (-1)^k \alpha_j^k \frac{t^k}{k!}\right) M^{-1} \\ &= M \text{Diag}(e^{-\alpha_j t}) M^{-1}. \end{aligned}$$

使用 $\lim_{t \rightarrow \infty} e^{-\alpha_j t} = 0$ ，由此可得 $\lim_{t \rightarrow \infty} e^{-A^T At} = \mathbb{O}_n$ 。 ■

参考文献

- [1] E. Aarts, J. Korst, *Simulated Annealing and Boltzmann Machines* (John Wiley, Chichester, UK, 1989)
- [2] D.H. Ackley, G.E. Hinton, T.J. Sejnowski, A learning algorithm for boltzmann machines. *Cogn. Sci.* **9**, 147–169 (1985)
- [3] S. Amari, Theory of adaptive pattern classifiers. *IEEE Trans. Comput.* **EC-16**(3), 299–307 (1967)
- [4] S. Amari, *Differential-Geometrical Methods in Statistics*. Lecture Notes in Statistics, vol. 28 (Springer, Berlin, 1985)
- [5] S. Amari, Information geometry of the EM and em algorithms for neural networks. *Neural Netw.* **8**(9), 1379–1408 (1995)
- [6] S. Amari, Natural gradient works efficiently in learning. *Neural Comput.* **10**(2), 251–276 (1998)
- [7] S. Amari, *Information Geometry and Its Applications*. Applied Mathematical Sciences Book, vol. 194, 1st edn. (Springer, New York, 2016)
- [8] S. Amari, H. Park, F. Fukumizu, Adaptive method of realizing natural gradient learning for multilayer perceptrons. *Neural Comput.* **12**, 1399–1409 (2000)
- [9] V.I. Arnold, On functions of three variables. *Dokl. Akad. Nauk SSSR* **114**, 953–965 (1957)
- [10] R. Arora, A. Basu, P. Mianjy, A. Mukherjee, Understanding deep neural networks with rectified linear units. *ICLR* (2018)

- [11] R.B. Ash, *Information Theory* (Dover Publications, New York, 1990)
- [12] A. Barron, Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inf. Theory* **39**, (1993)
- [13] G. Bartok, C. Szepesvári, S. Zilles, Models of active learning in grouped-structured state spaces. *Inf. Comput.* **208**(4), 364–384 (2010)
- [14] Y. Bengio, P. Frasconi, P. Simard, The problem of learning long-term dependencies in recurrent networks, in *IEEE International Conference on Neural Networks*, San Francisco (IEEE Press, 1993), pp. 1183–1195
- [15] Y. Bengio, P. Frasconi, P. Simard, Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* (1994)
- [16] J. Bergstra, G. Desjardins, P. Lamblin, Y. Bengio, Quadratic polynomials learn better image features. Technical Report 1337 (Département d’Informatique et de Recherche Opérationnelle, Université de Montréal, 2009)
- [17] H. Bohr, Zur theorie der fastperiodischen funktionen i. *Acta Math.* **45**, 29–127 (1925)
- [18] A.E. Bryson, A gradient method for optimizing multi-stage allocation processes, in *Proceedings of the Harvard University Symposium on Digital Computers and Their Applications*, April 1961
- [19] P.C. Bush, T.J. Sejnowski, *The Cortical Neuron* (Oxford University Press, Oxford, 1995)
- [20] O. Calin, *An Informal Introduction to Stochastic Calculus with Applications* (World Scientific, Singapore, 2015)
- [21] O. Calin, Entropy maximizing curves. *Rev. Roum. Math. Pures Appl.* **63**(2), 91–106 (2018)
- [22] O. Calin, C. Udriște, *Geometric Modelling in Probability and Statistics* (Springer, New York, 2014)
- [23] E. Çinlar, *Probability and Stochastics*. Graduate Texts in Mathematics, vol. 261 (Springer, New York, 2011)
- [24] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *IEEE Trans. Neural Netw.* (2014), [arXiv:1406.1078](https://arxiv.org/abs/1406.1078)

- [25] M.A. Cohen, S. Grossberg, Absolute stability of global pattern information and parallel memory storage by competitive neural networks. *IEEE Trans. Syst. Man Cybern.* **SMC-13**, 815–826 (1983)
- [26] T.S. Cohen, M. Geiger, J. Köhler, M. Welling, Spherical CNNs. ICRL (2018), <https://openreview.net/pdf?id=Hkbd5xZRb>
- [27] T.S. Cohen, M. Welling, Group equivariant convolutional networks (2016), <https://arxiv.org/abs/1602.07576>
- [28] J.M. Corcuera, F. Giummolé, A characterization of monotone and regular divergences. *Ann. Inst. Stat. Math.* **50**(3), 433–450 (1998)
- [29] R. Courant, D. Hilbert, *Methods of Mathematical Physics*, 2nd edn. (Interscience Publishers, New York, 1955)
- [30] G. Cybenko, Approximation by superposition of a sigmoidal function. *Math. Control Signals Syst.* **2**, 303–314 (1989)
- [31] K. Diederik, J. Ba, Adam: a method for stochastic optimization (2014), [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
- [32] R.J. Douglas, C. Koch, K.A. Martin, H.H. Suarez, Recurrent excitation in neocortical circuits. *Science* **269**(5226), 981–985 (1995). <https://doi.org/10.1126/science.7638624>
- [33] S.E. Dreyfus, The numerical solutions of variational problems. *J. Math. Anal. Appl.* **5**, 30–45 (1962)
- [34] J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization. *JMLR* **12**, 2121–2159 (2011)
- [35] D. Dufresne, Fitting combinations of exponentials to probability distributions. *Appl. Stoch. Model Bus. Ind.* **23**(1), (2006). <https://doi.org/10.1002/asmb.635>
- [36] N. Dunford, J.T. Schwartz, *Linear Operators*. Pure and Applied Mathematics, vol. 1 (Interscience Publishers, New York, 1957)
- [37] E.B. Dynkin, *Markov Processes I, II* (Springer, Berlin, 1965)
- [38] A. Einstein, *Investigations on the Theory of Brownian Movement* (Dover Publications, Mineola, 1956) translated by A.D. Cowper
- [39] B.R. Frieden, *Science from Fisher Information*, 2nd edn. (Cambridge University Press, Cambridge, 2004)

- [40] B.R. Frieden, Extreme physical information as a principle of universal stability, in *Information Theory and Statistical Learning*, ed. by F. Emmert-Streib, M. Dehmer (Springer, Boston, 2009)
- [41] B.R. Frieden, B.H. Soffer, Lagrangians of physics and the game of fisher-information transfer. Phys. Rev. E **52**, 2274–2286 (1995)
- [42] F.A. Gers, J. Schmidhuber, LSTM recurrent networks learn simple context free and context sensitive languages. IEEE Trans. Neural Netw. **12**, 1333–1340 (2001)
- [43] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in *AISTATS'2010* (2010)
- [44] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics 2010*, Chia Laguna resort, Sardinia, Italy. JMLR, vol. 9 (2010)
- [45] X. Glorot, A. Borders, Y. Bengio, Deep sparse rectifier neural networks, in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics 2011*, Fort Lauderdale, FL, USA (2011)
- [46] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, Cambridge, 2016), <http://www.deeplearningbook.org>
- [47] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, in *NIPS* (2014)
- [48] I.J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, Y. Bengio, Maxout networks, in *ICML'13*, ed. by S. Dasgupta, D. McAllester (2013), pp. 1319–1327
- [49] L. Goodman, On the exact variance of products. J. Am. Stat. Assoc. **55**(292), 708–713 (1960). <https://doi.org/10.2307/2281592>., JSTOR 2281592
- [50] A. Graves, A. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in *ICASSP* (2013), pp. 6645–6649
- [51] P.R. Halmos, *Measure Theory*. The University Series in Higher Mathematics, 7th edn. (Van Nostrand Company, Princeton, 1961)
- [52] B. Hanin, Universal function approximation by deep neural nets with bounded width and relu activations (2017), [arXiv:1708.02691](https://arxiv.org/abs/1708.02691)

- [53] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*, 2nd edn. (Springer, New York, 2017)
- [54] D. Hilbert, *Grundzuge einer allgemeinen theorie der linearen integralgleichungen i.* Gott. Nachrichten, math.-phys. K1 (1904), pp. 49–91
- [55] S. Hochreiter, Untersuchungen zu dynamischen neuronalen netzen, Diploma thesis, Technische Universität München, 1991
- [56] S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997)
- [57] J. Hopfield, Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci.* **79**, 2554–2558 (1982)
- [58] J.J. Hopfield, Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci.* **81**, 3088–3092 (1984)
- [59] K. Hornik, M. Stinchcombe, H. White, Multilayer feed-forward networks are universal approximators. *Neural Netw.* **2**, 359–366 (1989)
- [60] B. Irie, S. Miyake, Capabilities of three-layered perceptrons, in *IEEE International Conference on Neural Networks*, vol. 1 (1988), pp. 641–648
- [61] E. Ising, Beitrag zur theorie des ferromagnetismus. *Z. für Phys.* **31**, 253 (1925)
- [62] H.J. Kelley, Gradient theory of optimal flight paths. *ARS J.* **30**(10), 947–954 (1960). <https://doi.org/10.2514/8.5282>
- [63] S. Kirkpatrick, C. Gelatt, M. Vecchi, Optimization by simulated annealing. *Science* **220**, 671–680 (1983)
- [64] A.N. Kolmogorov, On the representation of continuous functions of many variables by superposition of continuous functions in one variable and addition. *Dokl. Akad. Nauk. SSSR* **144**, 679–681 (1957). American Mathematical Society Translation, **28**, 55–59 (1963)
- [65] R. Kondor, *Group Theoretical Methods in Machine Learning* (Columbia University, New York, 2008)
- [66] R. Kondor, S. Trivedi, On the generalization of equivariance and convolution in neural networks to the action of compact groups (2018), <https://arxiv.org/abs/1802.03690>

- [67] B. Kosko, Bidirectional associative memories. *IEEE Trans. Syst. Man Cybern.* **18**, 49–60 (1988)
- [68] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in *NIPS'2012* (2012)
- [69] S. Kullback, R.A. Leibler, On information and sufficiency. *Ann. Math. Stat.* **22**, 79 (1951)
- [70] S. Kullback, R.A. Leibler, *Information Theory and Statistics* (Wiley, New York, 1959)
- [71] S. Kullback, R.A. Leibler, Letter to the editor: the Kullback-Leibler distance. *Am. Stat.* **41**(4), (1987)
- [72] L.D. Landau, E.M. Lifshitz, *Statistical Physics. Course of Theoretical Physics*, vol. 5, translated by J.B. Sykes, M.J. Kearsley (Pergamon Press, Oxford, 1980)
- [73] Y. LeCun, Modèles connexionistes de l'apprentissage, Ph.D. thesis, Université de Paris VI, 1987
- [74] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in *Proceedings of the IEEE*, November 1998
- [75] Y. LeCun, K. Kavukcuoglu, C. Farabet, Convolutional networks and applications in vision, in *Proceedings of 2010 IEEE International Symposium Circuits and Systems (ISCAS)*, pp. 253–256
- [76] Y. Li, K. Swersky, R.S. Zemel, Generative moment matching networks. *CoRR* (2015), [arXiv:abs/1502.02761](https://arxiv.org/abs/1502.02761)
- [77] S. Linnainmaa, The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors, Master's Thesis (in Finnish), University of Helsinki, pp. 6–7, 1970
- [78] S. Linnainmaa, Taylor expansion of the accumulated rounding error. *BIT Numer. Math.* **16**(2), 146–160 (1976). <https://doi.org/10.1007/bf01931367>
- [79] Z. Lu, H. Pu, , F. Wang, Z. Hu, L. Wang, The expressive power of neural networks: a view from the width, in *Neural Information Processing Systems* (2017), pp. 6231–6239

- [80] M.E. Hoff Jr., Learning phenomena in networks of adaptive circuits. Ph.D. thesis, Tech Rep. 1554-1, Stanford Electron. Labs., Standford, CA, July 1962
- [81] D.J.C. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, 2003)
- [82] W.S. McCulloch, W. Pitts, A logical calculus of idea immanent in nervous activity. Bull. Math. Biophys. **5**, 115–133 (1943)
- [83] J. Mercer, Functions of positive and negative type and their connection with the theory of integral equations. Philos. Trans. R. Soc. Lond. Ser. A **209**, 415–446 (1909)
- [84] T. Mikolov, Statistical language models based on neural networks. Ph.D. thesis, Brno University of Technology, 2012
- [85] R.S. Millman, G.D. Parker, *Elements of Differential Geometry* (Prentice-Hall, Englewoods Cliffs, 1977)
- [86] M. Minsky, Neural nets and the brain: model problem. Dissertation, Princeton University, Princeton, 1954
- [87] M.L. Minsky, S.A. Papert, *Perceptrons* (MIT Press, Cambridge, 1969)
- [88] M. Mohri, A. Rostamizadeh, A. Talwalkar, *Foundations of Machine Learning*, 2nd edn. (MIT Press, Boston, 2018)
- [89] E.H. Moore, On the reciprocal of the general algebraic matrix. Bull. Am. Math. Soc. **26**(9), 394–95 (1920). <https://doi.org/10.1090/S0002-9904-1920-03322-7>
- [90] V. Nair, G.E. Hinton, Rectified linear units improve restricted Boltzmann machines, in *Proceedings of the 27th International Conference on Machine Learning 2010* (2010)
- [91] Y.A. Nesterov, A method of solving a convex programming problem with convergence rate $o(1/\sqrt{k})$. Sov. Math. Dokl. **27**, 372–376 (1983)
- [92] M. Nielsen, *Neural Networks and Deep Learning* (2017), <http://www.neuralnetworksanddeeplearning.com>
- [93] D.B. Parker, *Learning-Logic* (MIT, Cambridge, 1985)
- [94] E. Parzen, On the estimation of a probability density function and its mode. Ann. Math. Stat. **32**, 1065–1076 (1962)

- [95] P. Pascanu, Ç. Gülçehre, K. Cho, Y. Bengio, How to construct deep recurrent neural networks, in *ICLR* (2014)
- [96] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in *ICML*. Neural Computation (2013)
- [97] R. Penrose, A generalized inverse for matrices. Proc. Camb. Philos. Soc. **51**(3), 406–413 (1955). <https://doi.org/10.1017/S0305004100030401>
- [98] B.T. Polyak, Some methods of speeding up the convergence of iteration methods. USSR Comput. Math. Math. Phys. **4**(5), 1–17 (1964)
- [99] M. Rattray, D. Saad, S. Amari, Natural gradient descent for one-line learning. Phys. Rev. Lett. **81**, 5461–5465 (1998)
- [100] S. Ravanbakhsh, J. Schneider, B. Póczos, Equivariance through parameter-sharing, in *Proceedings of International Conference on Machine learning (ICML)* (2016), <https://arxiv.org/pdf/1702.08389.pdf>
- [101] R. Rojas, *Neural Networks a Systemic Introduction* (Springer, Berlin, 1996)
- [102] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain. Psychol. Rev. **65**, 386–408 (1958). Reprinted in: [Anderson and Rosenfeld 1988]
- [103] N. Le Roux, Y. Bengio, Deep belief networks are compact universal approximators. Neural Comput. **22**(8), 2192–2207 (2010)
- [104] H.L. Royden, *Real Analysis*, 6th edn. (The Macmillan Company, New York, 1966)
- [105] H.L. Royden, P.M. Fitzpatrick, *Real Analysis* (Prentice Hall, 2010)
- [106] W. Rudin, *Functional Analysis* (International Series in Pure and Applied Mathematics (McGraw-Hill, New York, 1991)
- [107] D. Rumelhart, G.E. Hinton, J.R. Williams, *Learning internal representations*, in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Foundations* (MIT Press, Cambridge, 1986)
- [108] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986). <https://doi.org/10.1038/323533a0>. Bibcode:1986Natur.323..533R

- [109] J. Schmidhuber, Deep learning in neural networks: an overview. *J. Math. Anal. Appl.* (2014), <https://arxiv.org/pdf/1404.7828.pdf>
- [110] S. Shalev-Shwartz, S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, Cambridge, 2014)
- [111] C. Shannon, A mathematical theory of communication. *Bell Syst. Tech. J.* **379–423**, 623–656 (1948)
- [112] K. Sharp, F. Matschinsky, Translation of ludwig boltzmann's paper “on the relationship between the second fundamental theorem of the mechanical theory of heat and probability calculations regarding the conditions for thermal equilibrium”. *Entropy* **17**, 1971–2009 (2015). <https://doi.org/10.3390/e17041971>
- [113] Xingjian Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.K. Wong, W.C. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in *Proceedings of the 28th International Conference on Neural Information Processing Systems* (2015), pp. 802–810
- [114] P. Smolensky, Information processing in dynamical systems: foundations of harmony theory, in *Parallel Distributed Processing*, vol. 1, ed. by D.E. Rumelhart, J.L. McClelland (MIT Press, Cambridge, 1986), pp. 194–281
- [115] D. Sprecher, On the structure of continuous functions of several variables. *Trans. Am. Math. Soc.* **115**, 340–355 (1964)
- [116] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014)
- [117] K. Steinbuch, *Automat und Mensch: Kybernetische Tatsachen und Hypothesen* (Springer, Berlin, 1965)
- [118] T. Tielemans, G. Hinton, Lecture 6.5—rmsprop, coursera: neural networks for machine learning. Technical report (2012)
- [119] N. Tishby, F.C. Pereira, W. Bialek, The information bottleneck method, in *The 37th Annual Allerton Conference on Communication, Control, and Computing* (1999), pp. 368–377
- [120] D. Wackerly, W. Mendenhall, R. Scheaffer, *Mathematical Statistics with Applications*, 7th edn. (Brooks/Cole Cengage Learning, 2008)

- [121] D. Wagenaar, *Information Geometry for Neural Networks* (Centre for Neural Networks; King's College London, 1998)
- [122] S. Wang, X. Sun, Generalization of hinging hyperplanes. *IEEE Trans. Inf. Theory* **51**(12), 4425–4431 (2005)
- [123] P.J. Werbos, Beyond regression: New tools for prediction and analysis in the behavioral sciences, Harvard University, 1975
- [124] P.J. Werbos, Applications of advances in nonlinear sensitivity analysis, in *Proceedings of the 10th IFIP Conference, 31.8–4.9 NYC* (1981), pp. 762–770
- [125] B. Widrow, An adaptive “adaline” neuron using chemical “memistors”. Technical Report 1553-2 (Office of Naval Research Contract, October 1960)
- [126] B. Widrow, Generalization and information storage in networks of adaline neurons, in *Self-Organizing Systems*, ed. by M. Yovitz, G. Jacobi, G. Goldstein (Spartan Books, Washington, 1962), pp. 435–461
- [127] B. Widrow, M.A. Lehr, 30 years of adaptive neural networks: perceptron, madaline ad backpropagation. *Proc. IEEE* **78**(9), 1415–1442 (1990)
- [128] N. Wiener, Tauberian theorems. *Ann. Math.* **33**(1), 1–100 (1932)
- [129] H.R. Wilson, J.D. Cowan, Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys. J.* **12**, 115–143 (1972)
- [130] H.H. Yang, S. Amari, Complexity issues in natural gradient descent method for training multilayer perceptrons. *Neural Comput.* **10**, 2137–2157 (1998)
- [131] R.W. Yeung, *First Course in Information Theory* (Kluwer, Dordrecht, 2002)
- [132] Y.T. Zhou, R. Chellappa, Computation of optical flow using a neural network, in *IEEE 1988 International Conference on Neural Networks* (1988), pp. 71–78

索引

- 符号
 F -可分的, 574
 $\sum \Pi$ -类网络, 260
 \sum -类网络, 260
 \sum -有限测度, 701
 d -密集, 253
 d -系统, 697
 d_μ -收敛, 323
 p -系统, 697
 ϵ -封闭神经网络, 220
- A
绝对连续, 703
绝对收敛, 304
抽象神经元, 133
动作, 530
激活函数, 21, 217, 342
自适应梯度算法, 100
自适应线性神经元, 157
亚当, 103
阿达马克, 104
改编的, 713
自适应实现, 493
代数, 209
几乎周期的, 224
解析函数, 260, 300
AND, 136
逼近序列, 176
逼近空间, 253
反正切函数, 29
面积, 433
- 阿尔泽拉 - 阿斯科利定理, 203, 205, 207, 208, 726
递增序列, 272
自相关, 154
自动编码器, 360
函数平均值, 109
- 平均池化, 507
轴突, 9
- B
反向传播, 173, 177, 180, 237, 483, 550
贝尔测度, 257, 700
巴拿赫空间, 721
吸引盆, 78, 126
批次, 188
伯努利分布, 710
- 随机变量, 459
贝塞尔不等式, 303, 724
偏差, 5
二元分类器, 148
双极阶跃函数, 22
体细胞, 9
玻尔条件, 224
玻尔兹曼常数,
 107
 分布, 109, 614
 学习, 621
 机器, 611, 617
 概率, 107

- 布尔不等式, 700
 博雷尔
 \mathcal{G} -代数, 36
 \mathcal{G} -域, 254, 268
 可测的, 268
 可测函数, 254
 测量, 700, 705, 727
 集合, 565
 波莱尔 - 坎泰利引理, 715
 瓶颈, 393
 有界泛函, 33
 线性泛函, 255
 有界收敛定理, 229, 702
 布朗运动, 118,
 719
 内置频率, 212
 bump函数, 229
 凸状函数, 30
 C
 康托尔引理, 202, 692
 容量, 381
 柯西序列, 215, 727
 柯西不等式, 205, 216, 307
 中心极限定理, 188, 453, 717
 链式法则, 173
 特征函数, 716
 乔列斯基分解, 119
 克里斯托费尔符号, 424, 449, 627
 CIFAR数据集, 63
 循环搜索, 122
 分类函数, 31
 聚类分类, 140
 换向器, 426
 紧致的, 692
 区间, 207
 度量空间, 202
 集合, 239, 253
 集合, 202
 支持信号, 518 - 520
 完成, 727
 压缩信息, 394
 可压缩性条件, 342, 344
 压缩因子, 380
 无压缩层, 376
 计算机断层扫描, 300
 条件的
 熵, 363
 期望, 44, 439
 期望, 714
 模型密度函数, 48
 条件熵, 378, 548
 包含常数, 209, 260
 连续输入神经元,
 159
 神经元数量, 300
 收缩, 577, 728
 收缩原理, 215, 544
 收敛
 几乎必然, 715
 在 L^p , 715 中
 依分布, 324
 在均方意义上, 716
 依概率, 271, 715
 收敛条件, 98
 凸泛函, 723
 包, 580
 可分性, 575
 集合, 571
 卷积, 229, 518
 内核, 242
 运算符, 521
 系列, 98
 冷却过程, 105
 余弦, 211
 余弦压缩器, 36

- 成本函数, 41, 49, 173, 185
计数测度, 699
协方差, 711
协变导数, 425
克拉美 - 罗不等式, 470
互相关, 518, 519
交叉熵, 46, 146, 185, 609
 误差, 147
晶体结构, 105
电流, 5
曲率张量, 627
- D
阻尼系数, 94
决策边界曲线, 587
- 函数, 565
 行, 150
 映射, 561, 576
 地图, 585
解码器, 395
深度学习, 215
深度神经网络, 300
默认概率, 145
退化核, 306
增量, 181
树突, 9
密集的, 256
稠密集, 208
相依树, 175
行列式, 342
对角化过程, 273
微分熵, 353
扩散矩阵, 118
迪尼定理, 201, 245
狄拉克函数, 22
 测度, 160, 228, 699, 701
方向导数, 81
离散熵, 353
离散测度, 160, 699
歧视性的
 函数, 258
 在 L^1 意义下, 266
 在 L^2 意义下, 262
歧视性函数, 32
离散度, 118
分布, 709
分布测度, 323
散度, 92, 94
控制收敛定理, 702
双指数, 30, 213
漂移, 118
辍学, 454
Dynkin公式, 118,
 719
 定理, 697
- E
叶戈罗夫定理, 704
特征函数, 303
特征值, 303, 734
特征值, 113
特征向量, 113
逐元素乘积, 183
ELU, 24
经验的
 均值, 152
 概率, 146
编码器, 395
能量函数, 618
熵, 195, 198, 351, 615
 二进制, 411
变化, 355
流, 356
 关于一个划分, 563
时期, 177
等度连续的, 204
平衡点, 91
等价类, 562
 关系, 561
等变性, 521, 533, 537

- 误差函数, 41
 错误信息, 319
 估计器, 46
 欧几里得
 球, 272
 距离, 43, 215, 727
 梯度, 493
 内积, 75
 长度, 441
 结构, 450
 欧几里得距离, 254
 欧几里得梯度, 488
 精确学习, 285, 300, 439, 620
 精确解, 153
 期望, 710
 梯度爆炸, 554
 扩展的
 输入, 134
 权重, 134
 外在的, 441
- F
 特征, 377
 前馈神经网络, 178, 281
- 费马定理, 69
 滤波器, 518
 过滤, 327, 713
 有限能量信号, 517, 519,
 520
 信号, 517, 519
 有限测度, 701
 第一基本形式, 424, 441
 第一层, 168
 费希尔信息, 626
- 费希尔信息, 465, 470矩
 阵, 471
 费希尔度量, 472, 499, 612
 不动点, 216, 219, 699
 定理, 727
 最平坦流形, 445
- 前向传播,
 179
 传播公式, 182
 傅里叶变换, 34, 214, 264,
 268, 298, 716
 功能独立性, 343
- G
 门, 554
 高斯, 30, 214, 300
 噪声, 460
 泛化误差, 60
 广义导数, 238
 生成矩匹配网络, 606
- 测地线, 442子流
 形, 446
 全局最小值, 69
 古德曼公式, 193, 711
 谷歌街景图像, 63
 GPU系统, 177
 梯度上升, 613
 梯度裁剪, 554
 梯度下降, 43, 151, 154, 173,
 186
 算法, 73
 方法, 82, 147
 梯度估计, 156
 组, 531
- H
 哈尔测度, 535
 哈达玛积, 184, 185, 459
 哈恩分解定理, 706
 哈恩函数, 29
 哈恩 - 巴拿赫定理, 256, 261, 723
 半空间, 33
 哈密顿函数, 91, 92
 调和函数, 73
 海维赛德函数, 22, 227, 230
 套期保值应用, 241
 海森堡群, 532

- 黑利定理, 37
赫林格距离, 55
黑利定理, 726
黑塞矩阵
 矩阵, 71, 113, 115
 方法, 113
隐藏层, 167, 178,
 205
隐藏层, 334
希尔伯特空间, 44,
 45, 723, 724
希尔伯特第十三问题, 296
曲棍球棒函数, 23
同胚, 574
齐性空间, 531
霍普菲尔德网络, 611, 629
水平渐近线, 32
双曲正切, 28, 181, 344
超参数, 59, 93, 193, 212
超平面, 33
超曲面, 43
- I
恒等函数, 21
像测度, 703
不可压缩的, 93
不定积分, 703
独立层, 375
指示函数, 33, 228,
 320, 698
无穷小算子, 118
信息评估, 351
- 瓶颈, 393
压缩, 355
压缩机, 381
字段, 329
损失, 356
路径, 399
平面, 398
信息几何, 466
- 初始化, 176
内积, 44
输入, 41, 321
 熵, 49
 信息, 325
 变量, 48
输入信息, 318
输入-输出函数, 360,
 578
输入-输出映射, 171, 218
积分核, 304
积分变换, 207, 225
交集, 691
内在的, 441
不变性, 537
不变性性质, 366
反函数定理, 78, 616, 729
- 入江和三宅的积分公式, 297
迭代公式, 155
伊藤过程, 118
伊藤的
 扩散, 719
 公式, 719
- J
雅可比行列式, 78, 344, 354, 360, 548, 616
杰弗里距离, 55
詹森 - 香农散度, 51
联合密度, 48
联合熵, 353
若尔当测度分解, 706
跳跃间断点, 143
- K
核, 207, 518, 520
动能, 90
基尔霍夫定律, 6
柯尔莫哥洛夫定理, 296
库尔贝克-莱布勒散度, 48, 50, 365,
 397, 468, 550, 592, 620

L

拉格朗日乘数, 58
 朗之万方程, 120
 拉普拉斯势, 214
 拉普拉斯算子, 73
 大数定律, 716
 层, 178
 学习连续函数, 255

决策映射, 578
 有限支撑函数, 285
 可积函数, 266
 可测函数, 268
 常微分方程的解, 280
 平方可积函数, 261
 带有信息, 438
 带有ReLU, 237
 带有Softplus, 242
 学习率, 152, 155, 189
 最小均方, 152
 勒贝格测度, 161, 207, 214,
 261, 352, 354, 564, 566,
 700 - 702
 左搜索, 121
 水平曲线,
 105
 超曲面, 73
 集合, 73
 水平曲线, 63
 列维 - 奇维塔联络, 426
 线搜索方法, 86
 线性函数, 21

 泛函, 256
 变换, 169
 线性联络, 426
 线性神经元, 21, 152, 171,
 184, 333, 360, 476, 574
 线性算子, 722

线性无关, 569
 利普希茨连续, 75
 局部最小值, 71
 对数似然, 352, 359
 对数似然函数, 149
 逻辑的
 函数, 27, 143, 181, 185,
 216, 229, 265, 300
 回归, 145
 对数单位, 28
 损失函数, 41, 549
 丢失信息, 325
 LSTM, 554
 卢津定理, 704

 M
 玛德琳, 158
 流形, 417
 马尔可夫
 链, 368
 不等式, 716
 性质, 369
 主方程, 183
 矩阵形式, 183
 最大池化, 507
 极大值的
 元素, 327
 秩, 345
 最大似然, 50, 150
 最大均值差异, 52
 麦卡洛克 - 皮茨神经元, 38
 均方误差, 47
 中值定理, 204, 544
 有意义的信息, 394
 可测的
 函数, 699
 集合, 320, 328
 空间, 254, 697
 度量, 33, 34, 254
 默瑟定理, 308, 311
 度量空间, 202, 254, 269,
 721, 727

- 结构, 439
最小池化, 508
小批量, 156
极小浸入子流形, 429
MNIST数据集, 62, 400, 437, 514
模型平均, 453
- 组合, 454
矩匹配, 606
动量法, 93
单调收敛定理, 702
摩尔 - 彭罗斯伪逆, 389, 450,
 736
移动平均线, 518
乘性噪声, 459
互信息, 364, 365
- N
 自然梯度, 487
 可忽略的, 701
 邻域搜索, 121
 涅斯捷罗夫加速梯度, 100
 神经网络, 42, 167
 神经网络, 167
 神经流形, 474, 498
 神经元, 9, 167
 牛顿法, 115
 牛顿-拉弗森法, 116
 噪声去除, 173
 无噪声编码定理, 378
 噪声神经元, 466
 非退化, 154
 非负定, 308
 不可恢复信息, 327, 328
- 范数, 59, 721
 算子的, 722
 正则化, 445
正态分布, 193, 710
正规方程, 43
- 归一化初始化, 195
 赋范线性空间, 255
 赋范空间, 721
 空集, 705
- O
 支持率, 144
 欧姆定律, 6
 单隐层神经网络, 198, 207, 217,
 239, 259, 311
 独热向量, 31, 401, 567, 576
 在线学习, 156
 最优的
 参数值, 443
 点, 442
OR, 136
轨道, 531
奥恩斯坦-乌伦贝克, 62, 120
正交的, 261
 投影, 436
 正交投影, 43
 标准正交系, 724
输出, 41, 178
 信息, 318, 325, 329
输出信息, 438
超定系统, 737
过拟合, 381
过拟合, 59, 417, 454
 去除, 64
- P
 参数, 42
 分区, 563
 配分函数, 616
 帕曾窗, 55
 感知机, 136, 171, 227, 254,
 325, 565
 网络, 237
 容量, 390
 学习算法, 163
 模型, 135
周期函数, 211, 223, 224

- 皮卡-林德洛夫定理, 76
 分段线性, 30
 平台期, 116, 187
 逐点收敛, 201
 折线, 87
 多项式回归, 448
 汇集, 355, 507
 层, 513
 正定, 154
 潜力
 能量, 90
 函数, 59
 预测器, 44
 参数修正线性单元, 23
 压力, 4
 原函数, 134
 先验, 195
 概率
 度量, 701
 空间, 709
 概率测度, 161
 邻近度, 41
- Q**
 二次的
 误差函数, 153
 形式, 72
 函数, 72
 二次Renyi熵, 55
 商集, 562
- R**
 拉东 - 尼科迪姆定理,
 161,703
 秩, 88,128
 可恢复信息, 329
 整流定理, 587
 递归神经网络, 217
 正则交叉熵函数, 185
 正则化, 58, 205, 216, 443
 正则化成本函数, 63
- 修正线性单元, 23, 187, 215, 239
 雷尼熵, 55
 电阻器, 5
 受限玻尔兹曼机, 635
 黎曼的
 距离, 424
 流形, 424, 468
 度量, 423, 440, 450, 472
 里斯定理, 725
 右搜索, 121
 RMSProp, 101
 循环神经网络, 543, 546
 车问题, 7, 630
 旋转群, 531
- S**
 样本估计, 55
 第二基本形式, 427, 445
 SELU, 24
 可分性, 570
 分离点, 209, 260
 顺序连续性, 700
 香农熵, 48 - 50
- Sigmoid激活函数, 232
- 函数, 26, 204
 神经网络, 236
 神经元, 143, 206, 224, 333, 450
 西格蒙德函数, 32, 34, 259, 272
 信号, 168
 带符号测度, 705
 简单函数, 698
 单纯形算法, 73
 模拟退火方法, 104, 499, 613, 629
- 正弦, 211
 SLU, 24
 softmax, 31, 389, 582
 softplus函数, 25, 242
 软sign函数, 29

谱半径, 734
施普雷彻定理, 297
平方可积函数, 45
平方误差逼近, 277
挤压函数, 36, 286
状态系统, 543
统计流形, 466
最速下降法, 79, 82
阶梯函数, 21, 36
随机梯度下降法, 189

过程, 713
搜索, 116
球形搜索, 124
随机神经元, 611
股票, 212
斯通 - 魏尔斯特拉斯定理,
 208, 260, 306
步长, 522
子流形, 426
平方误差之和, 254
监督学习, 69
SVHN数据集, 63
对称差, 328
突触, 9

T
切线, 70, 116
切平面, 437
切向量场, 423
目标函数, 41, 176, 218, 251
目标信息, 318
出租车度量, 254
泰勒近似, 70, 114
温度, 105
张量, 693
测试误差,
 59
热力学系统,
 107
热力学系统, 618

阈值阶跃函数, 22
托普利茨
 矩阵, 385
 性质, 387
全压缩, 380
总能量, 91
完全测地线, 429
训练
 数据, 58
 分布, 48, 466
 偶数, 320
 测量, 320
 集合, 60
晶体管, 6
转移函数, 544
平移不变性, 510

算子, 521
转置, 733
三角不等式, 218

U
未压缩层, 340, 342
欠拟合, 66, 378, 381
基础分布, 65
一致收敛, 303
一致收敛序列, 203
均匀分布, 198
一致有界, 203, 205
一致连续, 230
并集, 691
单变量高斯分布, 157
通用逼近器, 251

V
验证集, 60, 193
梯度消失, 187, 545, 554
方差, 191, 711
 近似值, 712
电压, 5

W

弱收敛, 324, 716
权重, 178
初始化, 191
维纳陶伯定理, 213,
214, 268

X

Xavier初始化, 193
异或, 138, 171, 286
Z
佐恩引理, 327, 692