

MATHEMATICS FOR MACHINE LEARNING

Marc Peter Deisenroth
A. Aldo Faisal
Cheng Soon Ong

内容

前言

11

第一部分 数学基础	9
 介绍和动机	11
1.1.1 为直觉寻找词语	12
1.2 阅读本书的两种方	13
式 1.3 练习和反馈	16
 线性代数 2.1 线性	17
方程组 2.2 矩阵 2.3 求解线性方程组 2.4 向量	19
空间 2.5 线性独立	22
2.6 基和秩 2.7 线性映射 2.8 仿射空间 2.9 延伸阅读练习	27
	35
	40
	44
	48
	61
	63
	64
 解析几何 3.1 范数 3.2 内	70
积 3.3 长度和距	71
离 3.4 角和正交性 3.5 正交	72
基 3.6 正交补 3.7 函数内积 3.8 正交	75
投影 3.9 旋转 3.10 进一步阅读练习	76
	78
	79
	80
	81
	91
	94
	96
 4 矩阵分解 4.1 行列式和迹	98
	99

11

4.2 特特征值和特征向量	4.3 Cholesky 分解	105		
4.4 特特征分解和对角化	4.5 奇异值分	114		
解	4.6 矩阵近似	4.7 矩阵系统发育	4.8 进一步阅读练习	115
				119
				129
				134
				135
				137
向量微积分				139
5.1 单				
变量函数的微分	5.2 偏微分和梯度	5.3 向量值函数的		141
梯度	5.4 矩阵的梯度	5.5 计算梯度的有用恒等式		146
	5.6 反向传播和自动微分	5.7 高阶导数	5.8 线性化和	149
	多元泰勒级数	5.9 进一步阅读练习		155
				158
				159
				164
				165
				170
				170
6 概率与分布				172
6.1 概率空间的构造	6.2 离散概			172
率和连续概率	6.3 求和法则、乘法法则和贝叶斯定			178
理	6.4 汇總統计和独立性	6.5 高斯分布	6.6 共轭和指	
				183
数族	6.7 变量变化/逆变换	6.8 进一步阅读练习		186
				197
				205
				214
				221
				221
7 连续优化				225
7.1 使用梯度下降优				227
化	7.2 约束优化和拉格朗日乘数	7.3 凸优化	7.4 进一	
				233
步阅读练习				236
				246
				247
第二部分中央机器学习问题				249
8 当模型满足数据时	8.1 数据、模型和学			251
习	8.2 经验风险最小化	8.3 参数估计	8.4	
				251
				258
概率建模和推理	8.5 有向图模型			265
				272
				278

内容	三
8.6 模型选择	283
9 线性回归	289
9.1 问题公式	289
9.2 参数估计	291
9.3 贝叶斯线性回归	292
9.4 最大似然正交投影	292
9.5 进一步阅读	303
10 使用主成分分析进行降维	317
10.1 问题设置	318
10.2 最大方差视角	320
10.3 投影视角	325
10.4 特征向量计算和低秩逼近	333
10.5 高维 PCA	335
10.6 PCA 在实践中的关键步骤	336
10.7 潜在变量视角	339
10.8 延伸阅读	343
11 使用高斯混合模型进行密度估计	348
11.1 高斯混合模型	348
11.2 通过最大似然法进行参数学习	349
11.3 EM 算法	350
11.4 潜在变量视角	360
11.5 进一步阅读	363
12 使用支持向量机进行分类	370
12.1 分离超平面	370
12.2 原始支持向量机	372
12.3 双支持向量机	374
12.4 内核	383
12.5 数值解	388
12.6 进一步阅读	390
参考	395

前言

机器学习是将人类知识和推理提炼成适合构建机器和工程自动化系统的形式的长期尝试中的最新成果。随着机器学习变得越来越普遍,其软件包也越来越易于使用,低级技术细节被抽象出来并向从业者隐藏起来是自然而可取的。然而,这带来了从业者不知道设计决策的危险,因此也不知道机器学习算法的局限性。

有兴趣了解更多关于成功的机器学习算法背后的魔力的热心从业者目前面临着一组令人生畏的先决知识:

- 编程语言和数据分析工具
- 大规模计算及相关框架
- 数学和统计学以及机器学习如何建立在它的基础上

在大学里,机器学习的入门课程往往会在课程的早期部分涵盖其中一些先决条件。由于历史原因,机器学习课程往往在计算机科学系教授,学生通常在前两个知识领域接受培训,但在数学和统计学方面的培训较少。

当前的机器学习教科书主要侧重于机器学习算法和方法,并假设读者具有数学和统计学能力。因此,这些书只用一到两章来介绍背景数学,要么放在书的开头,要么作为附录。我们发现许多想要深入研究基本机器学习方法基础的人都在努力学习阅读机器学习教科书所需的数学知识。在大学教授本科和研究生课程后,我们发现高中数学与阅读标准机器学习教科书所需的数学水平之间的差距对许多人来说太大了。

本书突出了机器学习基本概念的数学基础,并将信息收集在一个地方,从而缩小甚至消除了这种技能差距。

1个

该材料由剑桥大学出版社出版,名为Marc Peter Deisenroth、A. Aldo Faisal 和 Cheng Soon Ong的机器学习数学(2020)。此版本可免费查看和下载,仅供个人使用。不得重新分发、转售或用于衍生作品。© MP Deisenroth、AA Faisal 和 CS Ong,2021年。<https://mml-book.com>

为什么要写另一本关于机器学习的书？

机器学习建立在数学语言的基础上,以表达直观上显而易见但难以形式化的概念。一旦适当形式化,我们就可以深入了解我们想要解决的任务。全球数学学生的一个普遍抱怨是,所涵盖的主题似乎与实际问题无关。我们认为机器学习是人们学习数学的明显而直接的动力。

“在大众心目中,数学与恐惧症和焦虑症联系在一起。你会认为我们在讨论蜘蛛。”
(Strogatz,2014年,第 281 页)

本书旨在成为构成现代机器学习基础的大量数学文献的指南。我们通过直接指出它们在基本机器学习问题的背景下的有用性来激发对数学概念的需求。为了使本书简短,省略了许多细节和更高级的概念。了解了这里介绍的基本概念,以及它们如何适应更大的机器学习背景,读者可以找到大量资源进行进一步研究,我们在各章末尾提供了这些资源。对于具有数学背景的读者,本书提供了对机器学习的简要但准确表述的一瞥。与其他专注于机器学习方法和模型的书籍相比 (MacKay,2003 年;Bishop,2006 年;Alpaydin,2010 年;Barber, 2012 年;Murphy,2012 年;Shalev-Shwartz 和 Ben-David,2014 年;Rogers 和 Girolami, 2016) 或机器学习的程序方面 (Muller 和 Guido,2016;Raschka 和 Mirjalili,2017;Chollet 和 Allaire,2018), 我们仅提供机器学习算法的四个代表性示例。相反,我们专注于模型本身背后的数学概念。我们希望读者能够更深入地理解机器学习中的基本问题,并将机器学习使用中出现的实际问题与数学模型中的基本选择联系起来。

我们的目标不是写一本经典的机器学习书籍。相反,我们的目的是提供应用于四个主要机器学习问题的数学背景,以便更容易阅读其他机器学习教科书。

谁是目标受众?

随着机器学习在社会上的广泛应用,我们相信每个人都应该对其基本原理有所了解。本书以学术数学风格编写,使我们能够准确了解机器学习背后的概念。我们鼓励不熟悉这种看似简洁的风格的读者坚持下去,并牢记每个主题的目标。我们在全文中加入了评论和评论,希望它能提供有关大局的有用指导。

本书假定读者具有普遍的数学知识

前言

3个

包括高中数学和物理。例如,读者应该以前看过导数和积分,以及二维或三维的几何向量。从那里开始,我们概括了这些概念。因此,本书的目标读者包括大学生、晚间学习者和参加在线机器学习课程的学习者。

与音乐类比,人们与机器学习有三种交互方式:敏锐的聆听者管道的具体情况。用户可以专注于使用现成的工具从数据中提

取见解。这使得不懂技术的领域专家能够从机器学习中受益。这类似于听音乐;用户能够在不同类型的机器学习之间进行选择和辨别,并从中受益。更多有经验的用户就像音乐评论家一样,提出有关机器学习在社会中的应用的重要问题,例如道德、公平和个人隐私。我们希望本书能为思考机器学习系统的认证和风险管理提供一个基础,并允许他们利用自己的领域专业知识来构建更好的机器学习系统。

经验丰富的艺术家机器学习的熟练从业者可以将不同的工具和库插入并运行到分析管道中。立体典型的从业者是数据科学家或工程师,他们了解机器学习接口及其用例,并且能够根据数据进行出色的预测。这类似于演奏音乐的演奏家,技艺高超的演奏者可以将现有的乐器带入生活并为听众带来乐趣。使用此处介绍的数学作为入门,从业者将能够了解他们最喜欢的方法的优点和局限性,并扩展和推广现有的机器学习算法。我们希望本书能为机器学习方法的更严格和更有原则的发展提供动力。

初出茅庐的 Composer随着机器学习应用于新的领域,机器学习的开发者需要开发新的方法并扩展现有的算法。他们通常是需要了解机器学习的数学基础并揭示不同任务之间关系的研究人员。这类似于音乐作曲家,他们在音乐理论的规则和结构内创作出令人惊叹的新作品。

我们希望本书能为那些想成为机器学习作曲家的人提供其他技术书籍的高级概述。社会非常需要能够提出和探索新方法来应对从数据中学习的许多挑战的新研究人员。

致谢我们感谢许多看过本

书早期草稿并经历过痛苦的概念阐述的人。我们试图实现他们的想法，我们并没有强烈反对。我们要特别感谢 Christfried Webers 仔细阅读了本书的许多部分，以及他对结构和介绍提出的详细建议。许多朋友和同事也非常友好地为每一章的不同版本提供了他们的时间和精力。

我们很幸运能够受益于在线社区的慷慨，他们通过<https://github.com> 提出了改进建议，这大大改进了这本书。

以下人员通过<https://github.com>发现了错误、提出了澄清并建议了相关文献或个人通讯。他们的名字按字母顺序排列。

Abdul-Ganiy Usman	艾伦布罗德
Adam Gaier	风狂天竺
Adele Jackson	菲奥娜·康登
Aditya Menon	乔治斯·西奥多罗
Alasdair Tran	何欣
Aleksandar Krnjaic	艾琳·赖萨·卡梅尼
Alexander Makrigiorgos	雅库布·纳巴格洛
Alfredo Canziani	詹姆斯·亨斯曼
Ali Shafti	杰米刘
Amr Khalifa	让·卡杜尔
Andrew Tanggara	让-保罗·埃贝耶
Angus Gruen	杰瑞强
Antal A. Buss	吉特什·辛德哈尔
Antoine Toisoul Le Cann	约翰劳埃德
Areg Sarvazyan	乔纳斯·纳维
Artem Artemev	乔恩·马丁
阿尔乔姆·斯捷潘诺夫	贾斯汀·希
比尔克罗米达斯	凯阿鲁库马兰
鲍勃·威廉姆森	卡米尔·德雷茨科夫斯基
文炳林	王莉莉
晁渠	Lionel Tondji Ngoupeyou
李成	莉迪亚·努芬 (Lydia Knufing)
克里斯夏洛克	马哈茂德阿斯兰
克里斯托弗格雷	马克·哈尔滕斯坦
丹尼尔麦克纳马拉	马克范德威尔克
丹尼尔伍德	马库斯黑格兰
达伦西格尔	马丁休因
大卫约翰斯顿	马修阿尔杰
陈大伟	马修·李

前言

5个

马克西姆斯·麦肯	沙基尔穆罕默德
张梦艳	肖恩贝瑞
迈克尔贝内特	谢赫·阿卜杜勒·拉希姆·阿里
迈克尔·佩德森	盛雪
申敏正	斯里达尔·蒂加拉詹
穆罕默德·马勒扎德	赛诺曼哈沙尼
纳文·库马尔	西蒙布赖奇
尼科·蒙塔利	托马斯·布勒
奥斯卡·阿马斯	帖木儿·沙拉波夫
帕特里克亨里克森	汤姆梅拉米德
帕特里克·维绍莱克	文森特·亚当
帕塔拉瓦乔迈	文森特·杜托多尔
保罗·凯利	武明
佩特罗斯·赫里斯托杜鲁	瓦西姆·阿夫塔
彼得亚努谢夫斯基	文志
普拉纳夫苏布拉马尼	沃伊切赫·斯托科维茨
曲玉孔	小南冲
拉吉卜扎曼	张晓薇
张蕊	郝亚洲
瑞安-里斯·格里菲斯	罗义成
所罗门卡邦戈	年轻的李
塞缪尔·奥贡莫拉	玉露
桑迪普马瓦迪亚	程运
萨维什·尼昆布	黄玉晓
塞巴斯蒂安·拉施卡	扎克·克兰科
Senanayak Sesh Kumar Karri	曹子建
白承宪	佐伊诺兰
沙赫巴兹·乔杜里	

通过 GitHub 贡献者,其真实姓名未列在他们的
GitHub 简介,有:

SamDataMad	伤心的	empet
bumptiousmonkey	HorizonP	victorBigand
idoamihai	cs-maillist	17SKYE
deepakiim	kudo23	jessjing1995

我们也非常感谢 Parameswaran Raman 和剑桥大学出版社组织的许多匿名审稿人,他们阅读了手稿早期版本的一个或多个章节,并提供了建设性的批评,从而导致了相当大的改进。特别值得一提的是我们的 LATEX 支持人员 Dinesh Singh Negi,他提供有关 LATEX 相关问题的详细而及时的建议。最后但同样重要的是,我们非常感谢我们的编辑 Lauren Cowles,她在本书的酝酿过程中一直耐心地指导我们。

符号表

符号a,b、	典型含义
c,a,β、γ	标量是小写的
x, y, z	向量是粗体小写
甲、乙、丙	矩阵是粗体大写
$x \perp A$	向量或矩阵的转置
A^{-1}	逆矩阵 x, y
	x 和 y 的内积 y
	x 和 y 的点积
$B = (b_1, b_2, b_3)$	(有序)元组
$B = [b_1, b_2, b_3]$	水平堆叠的列向量矩阵
$B = \{b_1, b_2, b_3\}$	向量集 (无序)
Z, N	整数和自然数, 分别为实数和复数, 分别为实数的n维向量
右,中	空间 全称量词: 对于所有 x 存在量词: 存在 x a 定义为 b
R^n	定义为 a 与 b 成正比, 即 $a = \text{常数} \cdot b$ 函数组合: “g after f”
$\forall x$	
$\exists x a :=$	
$ba :=$	
$ba \propto bg \circ f$	
\iff	当且仅当 implies
\implies	Sets a is
A, C	an
$a \in A$	element of set A由d = 1, 索引。..., D数
\emptyset	据点数; 由n = 1, 索引。..., N大小为m × m 的单位矩阵 大小为m
$A \setminus B$	
丁	
否	$\times n$ 的零矩阵 大小为m × n 的矩阵 标准/规范向量 (其中i是1 的分量)
Im	
$0_{m,n}$	
$1_{m,n}$	
ei	
dim	向量空间的维数 矩阵A的秩 线性映射
$rk(A)$	中的图像
$Im(\Phi)$	
$ker(\Phi)$	线性映射中的核 (零空间)
跨度[b1]	B1的跨度 (生成集) Trace of
$tr(A)$	A
$det(A) $	Determinant of A
$ $	Absolute value or determinant (depending on context)
$\ \cdot\ $	规范; 欧几里得, 除非指定特征值或拉格朗日乘数
λ	对应于特征值 λ 的特征空间

符号x ⊥	典型含义 向量x和y是
y	正交向量空间 向量空间V的正交补 x_n 的
V	和: $x_1 + \dots + x_N$
V^\perp	$x_1 \cdot \dots \cdot x_N$ 的乘积: $x_1 \cdot \dots \cdot x_N$
否 $n=1 \times n$ 否 $n=1 \times n \theta$	
$\frac{\partial f}{\partial x}$	f关于x的偏导数f关于x 的全导数 ∇f
$\frac{df}{dx}$	$= \min_x f(x)$
	f的最小函数值 $x \in \arg \min_x f(x)$ 最小化
f的值x (注: $\arg \min$ 返回一组值)	
LL	Lagrangian
负对数似然二项式系数, n 选k x关于随机变量X的方差x关	
$\begin{matrix} n \\ k \end{matrix}$	于随机变量X的期望x和y之间的协方差。
$\mathbb{V}[x]$	
$\mathbb{E}[x]$	
$\text{Cov}[x, y]$	
$X \perp \perp Y Z$	X在给定Z的情况下与Y条件独立
X p	
$N(\mu, \Sigma)$	
误码率(μ)	
$\text{Bin}(N, \mu)$	
贝塔 (α, β)	

缩略语表

缩略词	含义
例如	Exempli gratia (拉丁语:例如)
GMM	高斯混合模型即
id est	(拉丁语:这意味着) iid
独立同分布	MAP 最大后验最大似然估计/估计
正交基	
最大学习效率	
ONB	
主成分分析	主成分分析
聚碳酸酯	概率主成分分析
参考文献	行梯形
浪涌保护器	对称, 正定
支持向量机	支持向量机

第一部分

数学基础

介绍和动机

机器学习是关于设计自动从数据中提取有价值信息的算法。这里的重点是“自动”，即机器学习关注的是可以应用于许多数据集的通用方法，同时产生有意义的东西。机器学习的核心是三个概念：数据、模型和学习。

由于机器学习本质上是数据驱动的，数据是核心数据
机器学习。机器学习的目标是设计通用方法以从数据中提取有价值的模式，理想情况下不需要太多特定领域的专业知识。例如，给定大量文档（例如，许多图书馆中的书籍），机器学习方法可用于自动查找跨文档共享的相关主题（Hoffman 等人，2010 年）。为了实现这一目标，我们设计了通常与生成数据的过程相关的模型，类似于为我们给定的数据集建模。例如，在回归设置中，模型将描述一个将输入映射到实值输出的函数。套用 Mitchell (1997) 的话：如果模型在给定任务上的性能在考虑数据后得到改善，则称该模型从数据中学习。

目标是找到能够很好地泛化到未见数据的良好模型，我们将来可能会关心这些数据。学习可以理解为通过优化模型的参数，自动发现数据中的模式和结构的一种学习方式。

虽然机器学习已经有许多成功案例，而且软件也很容易用于设计和训练丰富而灵活的机器学习系统，但我们认为机器学习的数学基础对于理解更复杂的机器学习系统所依据的基本原理非常重要被建造。理解这些原则有助于创建新的机器学习解决方案、理解和调试现有方法，以及了解我们正在使用的方法的固有假设和局限性。

1.1 为直觉寻找词语

我们在机器学习中经常面临的一个挑战是概念和词语很滑，机器学习系统的特定组件可以抽象为不同的数学概念。例如，“算法”一词在机器学习的上下文中至少有两种不同的含义。在第一种意义上，我们使用短语“机器学习算法”来表示基于输入数据进行预测的系统。我们将这些算法称为预测器。在第二种意义上，我们使用完全相同的短语“机器学习算法”来表示一个系统，该系统会调整预测器的某些内部参数，以便它在未来看不见的输入数据上表现良好。在这里，我们将这种适应称为训练系统。

预测器

训练

本书不会解决歧义问题，但我们想预先强调，根据上下文，相同的表达方式可能有不同的含义。然而，我们试图使上下文足够清晰，以减少歧义。

本书的第一部分介绍了讨论机器学习系统的三个主要组成部分所需的数学概念和基础：数据、模型和学习。我们将在这里简要概述这些组件，一旦我们讨论了必要的数学概念，我们将在第8章再次讨论它们。

虽然并非所有数据都是数字，但考虑数字格式的数据通常很有用。在本书中，我们假设数据已经适当地转换为适合读入计算机程序的数字表示形式。因此，我们将数据视为向量。作为单词多么微妙的另一个例子，有（至少）三种不同的方式来思考向量：作为数字数组的向量（计算机科学观点），作为具有方向和大小的箭头的向量（a物理视图），以及作为对象的向量，它服从加法和缩放（数学视图）。

数据作为向量

模型

模型通常用于描述生成数据的过程，类似于手头的数据集。因此，好的模型也可以被认为是真实（未知）数据生成过程的简化版本，捕获与数据建模相关的方面并从中提取隐藏模式。然后可以使用一个好的模型来预测现实世界中会发生什么，而无需进行真实世界的实验

评论。

学习

我们现在来到问题的关键，机器学习的学习部分。假设我们有一个数据集和一个合适的模型。

训练模型意味着使用可用数据来优化模型的某些参数，这些参数与效用函数相关，该效用函数评估模型对训练数据的预测程度。大多数训练方法都可以被认为是一种类似于爬山到达顶峰的方法。

在这个类比中，山顶对应于一些最大值

1.2 阅读本书的两种方式

期望的性能测量。然而，在实践中，我们感兴趣的是模型在看不见的数据上表现良好。在我们已经看到的数据（训练数据）上表现良好可能只意味着我们找到了一种记忆数据的好方法。然而，这可能无法很好地泛化到看不见的数据，并且在实际应用中，我们经常需要将我们的机器学习系统暴露在它以前没有遇到过的情况下。

让我们总结一下本书中涵盖的机器学习的主要概念：

- 我们将数据表示为向量。
- 我们使用概率或优化视图选择合适的模型。
- 我们通过使用数值优化方法从可用数据中学习，目的是使模型在未用于训练的数据上表现良好。

1.2 阅读本书的两种方式我们可以考虑

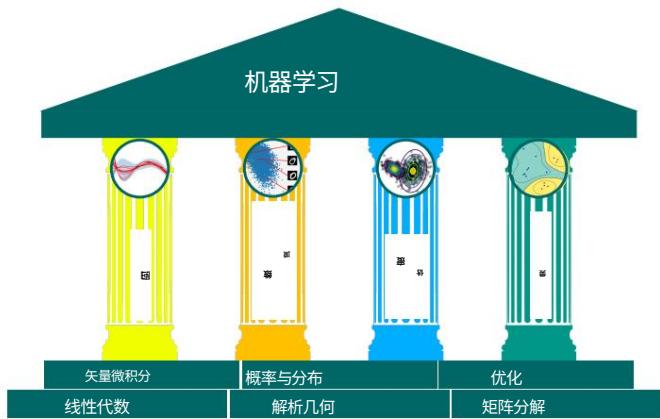
两种理解机器学习数学的策略：

- **自下而上**:从基础到高级构建概念。这通常是更多技术领域的首选方法，例如数学。这种策略的优点是，读者在任何时候都能够依赖他们以前学过的概念。不幸的是，对于从业者来说，许多基础概念本身并不是特别有趣，缺乏动力意味着大多数基础定义很快就会被遗忘。
- **自上而下**:从实际需求向下钻取到更基本的需求。这种以目标为导向的方法的优点是，读者始终知道为什么他们需要研究特定概念，并且有一条清晰的知识路径。这种策略的缺点是知识建立在可能不稳定的基础上，读者必须记住一组他们无法理解的单词。

我们决定以模块化的方式编写本书，将基础（数学）概念与应用程序分开，以便可以两种方式阅读本书。本书分为两部分，第一部分奠定了数学基础，第二部分将第一部分的概念应用于一组基本的机器学习问题，这些问题构成了机器学习的四大支柱，如图 1.2 所示：回归、维度减少、密度估计和分类。第 I 部分的章节主要建立在前面的章节之上，但如果需要，可以跳过一章并向后学习。第二部分中的章节只是松散耦合的，可以按任何顺序阅读。有很多向前和向后的指针

图 1.2 基础和

机器学习的四大支柱。



在本书的两个部分之间,将数学概念与机器学习算法联系起来。

当然,阅读这本书的方式不止两种。大多数读者结合使用自上而下和自下而上的方法进行学习,有时会在尝试更复杂的概念之前建立基本的数学技能,但也会根据机器学习的应用选择主题。

第一部分是关于数学

我们在本书中介绍的机器学习的四大支柱(见图 1.2)需要扎实的数学基础,这在第一部分中有所阐述。

我们将数值数据表示为向量,并将此类数据的表格表示为矩阵。向量和矩阵的研究称为线性代数,我们在第 2 章介绍过。向量的集合作为矩阵也在那里描述。

线性代数

给定两个代表现实世界中两个对象的向量,我们想对它们的相似性做出陈述。这个想法是我们的机器学习算法(我们的预测器)应该预测相似的向量具有相似的输出。为了形式化向量间相似性的概念,我们需要引入将两个向量作为输入并返回表示它们相似性的数值的操作。相似度和距离的构造是解析几何的核心,将在第 3 章中讨论。

解析几何

在第 4 章中,我们介绍了一些关于矩阵和矩阵分解的基本概念。一些矩阵运算在机器学习中非常有用,它们允许对数据进行直观的解释和更有效的学习。

矩阵 分解

我们通常认为数据是对某些真实潜在信号的嘈杂观察。我们希望通过应用机器学习,我们可以从噪声中识别出信号。这需要我们有一种语言来量化“噪音”的含义。我们通常也希望有预测器

1.2 阅读本书的两种方式

15

允许我们表达某种不确定性,例如,量化我们对特定测试数据点的预测值的信心。不确定性的量化是概率论的领域,概率论在第 6 章中介绍。

为了训练机器学习模型,我们通常会找到可以最大化某些性能指标的参数。许多优化技术需要梯度的概念,它告诉我们搜索解决方案的方向。第 5 章是关于向量微积分的,详细介绍了梯度的向量微积分概念,我们随后在第 7 章中使用了梯度,我们在第 7 章中讨论了优化以找到函数的最大值/最小值。

优化

第二部分是关于机器学习本书的第二部

分介绍了机器学习的四大支柱,如图 1.2 所示。我们说明了本书第一部分中介绍的数学概念如何成为每个支柱的基础。

从广义上讲,章节按难度排序(升序)。

在第 8 章中,我们以数学方式重申了机器学习的三个组成部分(数据、模型和参数估计)。此外,我们还提供了一些构建实验设置的指南,以防止对机器学习系统进行过于乐观的评估。回想一下,我们的目标是构建一个对未见数据表现良好的预测器。

在第 9 章中,我们将仔细研究线性回归,其中我们的线性回归目标是找到将输入 $x \in \mathbb{R}^D$ 映射到相应观察到的函数值 $y \in \mathbb{R}$ 的函数,我们可以将其解释为它们各自输入的标签。我们将通过最大似然和最大后验估计以及贝叶斯线性回归讨论经典模型拟合(参数估计),我们在其中整合参数而不是优化它们。

第 10 章使用主成分分析重点介绍降维,这是图中的第二个支柱。1.2 降维的关键目标是找到高维数据 $x \in \mathbb{R}^D$ 的紧凑、低维表示,这通常比原始数据更容易分析。与回归不同,关注数据建模没有与数据点 x 关联的标签。

在第 11 章中,我们将转向第三个支柱:密度估计。密度估计的目标是找到描述给定数据集的概率分布。为此,我们将重点关注高斯混合模型,并将讨论寻找该模型参数的迭代方案。与降维一样,没有与数据点 $x \in \mathbb{R}^D$ 相关联的标签。但是,我们不寻求数据的低维表示。相反,我们对描述数据的密度模型感兴趣。

第 12 章以对第四个问题的深入讨论结束本书。

分类

支柱：分类。我们将在支持向量机的背景下讨论分类。与回归（第 9 章）类似，我们有输入 x 和相应的标签 y 。然而，与标签为实值的回归不同，分类中的标签是整数，这需要特殊的

关心。

1.3 练习与反馈

我们在第一部分提供了一些练习，这些练习大部分可以用笔和纸来完成。对于第二部分，我们提供了编程教程（jupyter 笔记本）来探索我们在本书中讨论的机器学习算法的一些属性。

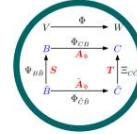
我们感谢剑桥大学出版社大力支持我们实现教育和学习民主化的目标，并通过以下网址免费提供本书下载

<https://mml-book.com>

可以在其中找到教程、勘误表和其他材料。可以使用上述 URL 报告错误并提供反馈。

2个

线性代数



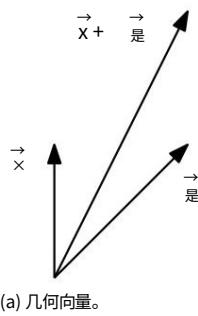
在形式化直观概念时,一种常见的方法是构造一组对象(符号)和一组操作这些对象的规则。这被称为代数。线性代数是研究向量和某些代数规则来处理向量的学科。我们很多人从学校知道的向量称为“几何向量”,通常用字母上方的小箭头表示,例如 \vec{x} 和 \vec{y} 。

在本书中,我们讨论了向量的更一般概念,并使用粗体字母来表示它们,例如, x 和 y 。

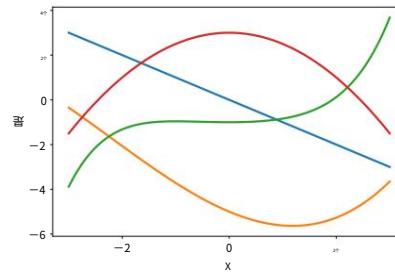
通常,向量是特殊对象,可以将它们加在一起并乘以标量以产生另一个同类对象。从抽象的数学角度来看,任何满足这两个性质的对象都可以认为是向量。以下是此类矢量对象的一些示例:

1. 几何向量。这个向量的例子可能在高中数学和物理中很熟悉。几何向量 见图 2.1(a) 是有向线段,可以绘制(至少在两个维度上)。两个几何向量是另一个几何向量。此外,乘以标量 λx , $\lambda \in \mathbb{R}$,也是一个几何向量。实际上,它是原始向量乘以 λ 。因此,几何向量是前面介绍的向量概念的实例。将向量 \vec{x} , 可以添加 \vec{y} ,这样 $\vec{x} + \vec{y} = \vec{z}$ 解释为几何向量使我们能够使用我们对方向和大小的直觉来推理数学运算。

2. 多项式也是向量;见图 2.1(b):两个多项式可以



(a) 几何向量。



(b) 多项式。

图 2.1 不同类型的载体。矢量可以是令人惊讶的对象,包括 (a) 几何载体 (b) 多项式。

相加,得到另一个多项式;它们可以乘以一个标量 $\lambda \in \mathbb{R}$,结果也是一个多项式。因此,多项式是(相当不寻常的)向量实例。

请注意,多项式与几何向量非常不同。几何向量是具体的“绘图”,而多项式是抽象概念。然而,它们都是前述意义上的载体。

3. 音频信号是矢量。音频信号表示为一系列数字。我们可以把音频信号加在一起,它们的总和就是一个新的音频信号。如果我们缩放一个音频信号,我们也会得到一个音频信号。

因此,音频信号也是一种矢量。

4. \mathbb{R}^n 的元素(n 个实数的元组)是向量。 \mathbb{R}^n 比多项式更抽象,也是我们在本书中重点关注的概念。例如,

$$\begin{matrix} & & & 1\text{维} \\ & \text{一个}= & 2\text{维} & \in \mathbb{R}^3 \\ & & & 3\text{维} \end{matrix} \quad (2.1)$$

是三元组数字的一个例子。添加两个向量 $a, b \in \mathbb{R}^n$ component-wise 得到另一个向量: $a + b = c \in \mathbb{R}^n$ 。此外,将 $a \in \mathbb{R}^n$ 乘以 $\lambda \in \mathbb{R}$ 会得到缩放向量 $\lambda a \in \mathbb{R}^n$ 。

仔细检查
在计算机上实现时,
数组运算是否实际执行矢
量运算。

将向量视为 \mathbb{R}^n 的元素还有一个额外的好处,即它松散地对应于计算机上的实数数组。许多编程语言都支持数组运算,这样可以方便地实现涉及向量运算的算法。

线性代数侧重于这些向量概念之间的相似性。

我们可以将它们加在一起并乘以标量。我们将主要关注 \mathbb{R}^n 中的向量,因为线性代数中的大多数算法都是在 \mathbb{R}^n 中计算的。我们将在第 8 章中看到,我们经常考虑将数据表示为 \mathbb{R}^n 中的向量。

在本书中,我们将关注有限维向量空间,在这种情况下,任何类型的向量与 \mathbb{R}^n 之间存在1:1 的对应关系。在方便的时候,我们将使用关于几何向量的直觉并考虑基于数组的算法。

数学中的一个主要思想是“闭包”的思想。这是一个问题:我提出的操作可能产生的所有事物的集合是什么?在向量的情况下:从一小组向量开始,然后将它们相加并缩放它们可以得到的向量集是什么?这导致向量空间(第 2.4 节)。向量空间的概念及其属性是大部分机器学习的基础。图 2.2 总结了本章介绍的概念。

本章主要基于 Drumm 和 Weil (2001)、Strang (2003)、Hogben (2013)、Liesen 和 Mehrmann (2015) 的讲义和书籍,以及 Pavel Grinfeld 的线性代数系列。其他优秀

2.1 线性方程组

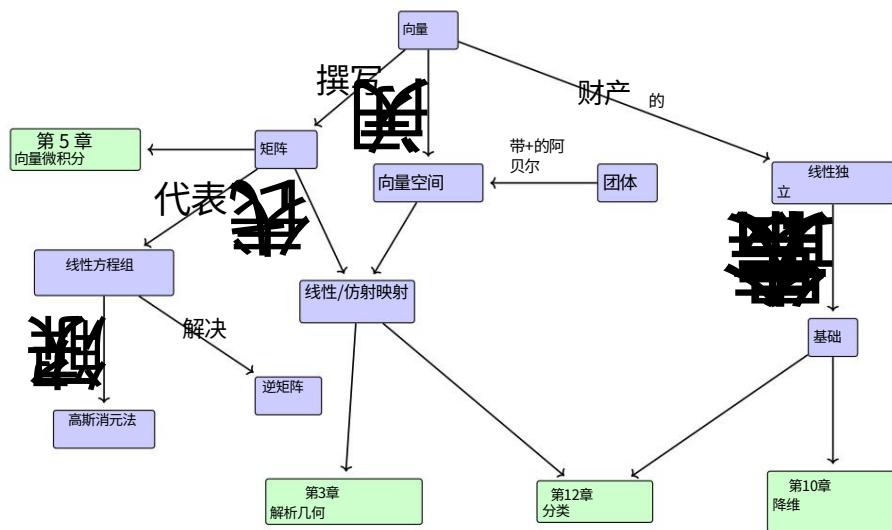


图 2.2 本章介绍的概念及其在本书其他部分中的使用位置的思维导图。

资源是 Gilbert Strang 在麻省理工学院的线性代数课程和 3Blue1Brown 的线性代数系列。

线性代数在机器学习和普通数学中扮演着重要的角色。本章介绍的概念在第 3 章中进一步扩展，包括几何概念。在第 5 章中，我们将讨论向量微积分，其中矩阵运算的基本知识是必不可少的。在第 10 章中，我们将使用投影（将在第 3.8 节中介绍）通过主成分分析 (PCA) 进行降维。在第 9 章中，我们将讨论线性回归，其中线性代数在解决最小二乘问题中起着核心作用。

2.1 线性方程组

线性方程组是线性代数的核心部分。许多问题可以表述为线性方程组，线性代数为我们提供了解决这些问题的工具。

示例 2.1 Nn 用

单位的 R_1, \dots, R_m ，于哪些资源和生产产品 N_1, \dots, N_m 是必需的。生产单位产品 N_j ， a_{ij} 想情况 $i = 1, \dots, m$ ，目标是找到一个最优生产计划，即如果总共有 b_i 单位的资源 R_i 可用并且（理想下）没有剩余资源，则应该生产多少单位 x_j 的产品 N_j 。

如果我们生产 x_1, \dots, x_n 个单位对应的产品，我们需要

总共

$$a_{11}x_1 + \dots + a_{1n}x_n = b_1 \quad (2.2)$$

许多单位的资源日。
以下方程组：

$$\begin{aligned} a_{11}x_1 + \dots + a_{1n}x_n &= b_1 \\ &\vdots \\ a_{m1}x_1 + \dots + a_{mn}x_n &= b_m \end{aligned} \quad (2.3)$$

其中 $a_{ij} \in \mathbb{R}$ 和 $b_i \in \mathbb{R}$ 。

线性方程组

方程(2.3)是一个线性方程组的一般形式， x_n 是这个系统的未知数。每个 n 元组 (x_1, \dots, x_n)
 x_1, \dots, \in
 满足 (2.3) 的 \mathbb{R}^n 是线性方程组的解。

解决方案

例 2.2 线性方程
组

$$\begin{aligned} x_1 + x_2 + x_3 &= 3 & (1) \\ x_1 - x_2 + 2x_3 &= 2 & (2) \\ 2x_1 + 3x_3 &= 5 & (3) \end{aligned} \quad (2.4)$$

无解：将前两个方程相加得到 $2x_1 + 3x_3 = 5$ ，这与第三个方程 (3) 相矛盾。

让我们看一下线性方程组

$$\begin{aligned} x_1 + x_2 + x_3 &= 3 & (1) \\ x_1 - x_2 + 2x_3 &= 2 & (2) \\ x_3 &= 1 & (3) \end{aligned} \quad (2.5)$$

从第一个和第三个等式，可以得出 $x_1 = 1$ 。从 (1)+(2)，我们得到 $2x_1 + 3x_3 = 5$ ，即 $x_3 = 1$ 。从 (3)，我们可以得到 $x_2 = 1$ 。

因此， $(1, 1, 1)$ 是唯一可能的解（通过插入验证 $(1, 1, 1)$ 是解）。

作为第三个例子，我们考虑

$$\begin{aligned} x_1 + x_2 + x_3 &= 3 & (1) \\ x_1 - x_2 + 2x_3 &= 2 & (2) \\ 2x_1 + 3x_3 &= 5 & (3) \end{aligned} \quad (2.6)$$

由于 $(1) + (2) = (3)$ ，我们可以省略第三个等式（冗余）。从 (1) 和 (2)，我们得到 $2x_1 = 5 - 3x_3$ 和 $2x_2 = 1 + x_3$ 。我们将 $x_3 = a \in \mathbb{R}$ 定义为自由变量，使得任何三元组

$$\frac{x_1}{2} - \frac{x_2}{2} - \frac{3x_3}{2} = \frac{5}{2} - \frac{3a}{2} - \frac{1}{2} + \frac{a}{2}, \quad a \in \mathbb{R} \quad (2.7)$$

2.1 线性方程组

21

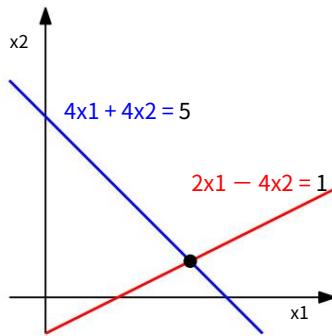


图 2.1 具有两个变量的两个线性方程组的解空间可以在几何上解释为两个的交集
线。每个线性方程表示一条线。

是线性方程组的一个解,即我们得到一个包含无穷多个解的解集。

通常,对于实值线性方程组,我们要么没有解,要么只有一个解,要么有无穷多个解。当我们无法求解线性方程组时,线性回归(第9章)解决了示例2.1的一个版本。

备注(线性方程组的几何解释)。在具有两个变量 x_1 、 x_2 的线性方程组中,每个线性方程在 x_1x_2 平面上定义一条直线。由于线性方程组的解必须同时满足所有方程,因此解集是这些线的交点。这个交集可以是一条线(如果线性方程描述的是同一条线)、一个点或空(当线平行时)。图2.1给出了该系统的说明

$$\begin{aligned} 4x_1 + 4x_2 &= 5 \\ 2x_1 - 4x_2 &= 1 \end{aligned} \tag{2.8}$$

其中解空间是点 $(x_1, x_2) = (1,)$ 。类似地,对于三个变量,每个线性方程确定三维空间中的一个平面。当我们把这些平面相交时,即同时满足所有的线性方程组,我们可以得到一个解集,它是平面、线、点或空的(当平面没有公共交点时)。◇

对于求解线性方程组的系统方法,我们将介绍一个有用的紧凑符号。我们将系数 a_{ij} 收集到向量中,并将向量收集到矩阵中。换句话说,我们将(2.3)中的系统写成以下形式:

$$\begin{array}{cccccc} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ \vdots & x_1 + & \vdots & x_2 + \cdots + & \vdots & x_n = & \vdots \\ a_{m1} & a_{m2} & & \cdots & a_{mn} & b_m \end{array} \tag{2.9}$$

$$\Leftrightarrow \begin{array}{ccc} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{array} \begin{array}{c} x_1 \\ \vdots \\ x_n \end{array} = \begin{array}{c} b_1 \\ \vdots \\ b_m \end{array} \quad (2.10)$$

下面,我们将仔细研究这些矩阵并定义精细的计算规则。我们将在 2.3 节中返回求解线性方程。

2.2 矩阵

矩阵在线性代数中起着核心作用。它们可用于紧凑地表示线性方程组,但它们也表示线性函数(线性映射),我们将在后面的 2.7 节中看到。在我们讨论其中一些有趣的话题之前,让我们首先定义什么是矩阵以及我们可以对矩阵进行哪些操作。我们将在第 4 章看到更多矩阵的性质。

矩阵

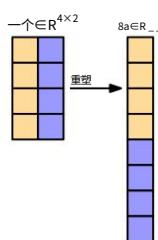
定义 2.1 (矩阵)。当 $m, n \in \mathbb{N}$ 时,实值 (m, n) 矩阵 A 是元素 a_{ij} 的 $m \times n$ 元组, $i = 1, \dots, m, j = 1, \dots, n$, 根据由 m 行 n 列组成的矩形方案排序:

$$\text{一个} = \begin{array}{c} a_{11} a_{12} \cdots a_{1n} \\ a_{21} a_{22} \cdots a_{2n} \\ \vdots \quad \vdots \quad \vdots \\ a_{m1} a_{m2} \cdots a_{mn} \end{array}, a_{ij} \in \mathbb{R}. \quad (2.11)$$

排
柱子
行向量
列向量
图 2.2 通过堆叠其列,矩
阵 A 可以表示为长
向量 a 。

按照惯例, $(1, n)$ -矩阵称为行, $(m, 1)$ -矩阵称为列。这些特殊矩阵也称为行/列向量。

$\mathbb{R}^{m \times n}$ 是所有实值 (m, n) 矩阵的集合。 $A \in \mathbb{R}^{m \times n}$ 可以等价表示为 $a \in \mathbb{R}^{mn}$, 将矩阵的所有 n 列堆叠成一个长向量; 见图 2.2。



2.2.1 矩阵加法和乘法

两个矩阵 $A \in \mathbb{R}^{m \times n}$ 的和,即 $B \in \mathbb{R}^{m \times n}$ 定义为元素

$$\text{一个} + \text{乙} := \begin{array}{c} a_{11} + b_{11} \quad \cdots \quad a_{1n} + b_{1n} \\ \vdots \quad \vdots \quad \vdots \\ a_{m1} + b_{m1} \quad \cdots \quad a_{mn} + b_{mn} \end{array} \in \mathbb{R}^{m \times n}. \quad (2.12)$$

对于矩阵 $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{n \times k}$, 产品的元素 c_{ij}
 $C = AB \in \mathbb{R}^{m \times k}$ 计算为

$$c_{ij} = \sum_{l=1}^n a_{il} b_{lj}, \quad i = 1, \dots, m, j = 1, \dots, k. \quad (2.13)$$

这意味着,为了计算元素 c_{ij} ,我们将A的第*i*行的元素与B的第*j*列相乘并将它们相加。稍后在3.2节中,我们将其称为相应行和列的点积。在某些情况下,我们需要明确表示我们正在执行乘法,我们使用符号 $A \cdot B$ 来表示乘法(明确显示“.”)。

在A和n行中
B这样我们就可以

为 $l = 1$,计算 $a_{il}b_{lj}$ 。...,

通常,两个向量

量a,b之间的点积表示为

$a \cdot b$ 或 a, b 。

评论。矩阵只有在它们的“相邻”维度匹配时才能相乘。例如, $n \times k$ 矩阵A可以与 $k \times m$ 矩阵B相乘,但只能从左侧开始:

$$\begin{array}{ccc} A & \cdot & B \\ n \times k & \cdot & m \times m \\ & & n \times m \end{array} = C \quad (2.14)$$

如果 $m = n$,则乘积 BA 未定义,因为相邻维度不匹配。◇备注。矩阵乘法未定义为矩阵元素上的元素操作,即,即使A,B的大小也适当)。当我们将在(多维)数组彼此相乘时,这种逐元素乘法经常出现在编程语言中,称为Hadamard乘积。

◇阿达玛产品

例 2.3

对于 $A = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} \in R^{2 \times 3}$, $B = \begin{pmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \in R^{3 \times 2}$, 我们获得

$$AB = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 3 \\ 2 & 5 \end{pmatrix} \in R^{2 \times 2}, \quad (2.15)$$

$$B \cdot A = \begin{pmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 6 & 4 & 2 \\ -2 & 0 & 2 \\ 3 & 2 & 1 \end{pmatrix} \in R^{3 \times 3}. \quad (2.16)$$

图 2.3即使两个矩阵

从这个例子中,我们已经可以看出矩阵乘法不是交换律,即 $AB \neq BA$;另请参见图2.3的说明。

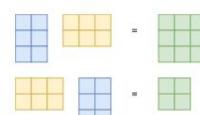
乘法 AB 和 BA 是

定义 2.2 (单位矩阵)。在 $R^{n \times n}$, 我们定义单位矩阵

定义,尺寸的

结果可能不同。

$$\text{在 } \mathbb{I} := \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ & & & 0 & \cdots & 1 & \cdots & 0 \\ & & & & & & \ddots & \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \cdots \end{pmatrix} \in R^{n \times n} \quad (2.17)$$



单位矩阵

作为 $n \times n$ 矩阵,在对角线上包含1 ,在其他任何地方都包含0 。

现在我们定义了矩阵乘法、矩阵加法和单位矩阵,让我们看看矩阵的一些性质:

结合性

■ 结合性:

$$\forall A \in R^{* \times n}, B \in R^{n \times p}, C \in R^{p \times q}: (AB)C = A(BC) \quad (2.18)$$

分配率

■ 分布性:

$$\forall A, B \in R^{* \times n}, C, D \in R^{n \times p}: (A + B)C = AC + BC \quad (2.19a)$$

$$A(C + D) = AC + AD \quad (2.19b)$$

■ 与单位矩阵相乘:

$$\forall A \in R^{* \times n} : I_m A = A I_n = A \quad (2.20)$$

请注意,对于 $m = n$, $I_m = I_n$ 。

2.2.2 逆与转置

方阵具有相同的列数
和行数。

定义 2.3 (逆)。考虑一个方阵 $A \in R^{n \times n}$ 。令矩阵 $B \in R^{n \times n}$ 具有 $AB = I_n = BA$ 的性质。B 称为 A 的逆,记为 A^{-1} 。

逆
常规的
可翻转的
非奇异的单数

不可逆的

备注 (2×2 矩阵的逆矩阵的存在性)。考虑一个矩阵

$$\text{一个} := \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \in R^{2 \times 2}. \quad (2.21)$$

如果我们将A乘以

$$A' := \begin{pmatrix} a_{22} - a_{12} \\ -a_{21} & a_{11} \end{pmatrix} \quad (2.22)$$

我们获得

$$AA' = \begin{pmatrix} a_{11}a_{22} - a_{12}a_{21} & 0 \\ 0 & a_{11}a_{22} - a_{12}a_{21} \end{pmatrix} = (a_{11}a_{22} - a_{12}a_{21})I_2. \quad (2.23)$$

所以,

$$A^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{pmatrix} a_{22} - a_{12} \\ -a_{21} & a_{11} \end{pmatrix} \quad (2.24)$$

当且仅当 $a_{11}a_{22} - a_{12}a_{21} \neq 0$ 。在第 4.1 节中,我们将看到 $a_{11}a_{22} - a_{12}a_{21} \neq 0$

$a_{12}a_{21}$ 是 2×2 矩阵的行列式。此外,我们通常可以使用行列式来检查矩阵是否可逆。

例 2.4 (逆矩阵)
矩阵

$$\begin{array}{c} \text{一个}= \begin{array}{ccc} 1 & 2 & 1 \\ 4 & 4 & 5 \\ 6 & 7 & 7 \end{array} \quad \text{乙}= \begin{array}{ccc} -7 & -7 & 6 \\ 2 & 1 & -1 \\ 4 & 5 & -4 \end{array} \end{array}, \quad (2.25)$$

由于 $AB = I = BA$,因此彼此相反。

定义 2.4 (转置)。对于 $A \in R^{m \times n}$ 矩阵 $B \in R^{n \times m}$ 有
 $b_{ij} = a_{ji}$ 称为A的转置。我们写 $B = A^T$

一般来说, A 可以通过将A的列写为行来获得的。以下是逆和转置的重要属性:

$$AA^{-1} = I = A^{-1}A \quad (2.26)$$

转置
矩阵A的主对角线(有时称为“principal diagonal”、“primary diagonal”、“leading diagonal”或“major diagonal”)是条目的集合

$$(AB)^{-1} = B^{-1}A^{-1} \quad (2.27)$$

$$(甲+乙)^{-1} = \overline{\text{一个}} + \overline{\text{乙}}^{-1} \quad (2.28)$$

A_{ij} 其中 $i = j$ 。
(2.28) 的标量情况是

$$(A) = \overline{\text{一个}} \quad (2.29)$$

$$(甲+乙) = \overline{\text{一个}} + \overline{\text{乙}} \quad (2.30)$$

$$\overline{\frac{1}{12+4}} = \overline{16} = \overline{12} + \overline{14}$$

$$(AB) = B \quad A \quad (2.31)$$

定义 2.5 (对称矩阵)。如果对称矩阵 $A = A^T$,则矩阵 $A \in R^{n \times n}$ 是对称的

请注意,只有 (n, n) -矩阵可以是对称的。通常,我们称 (n, n) -矩阵也为方阵,因为它具有相同的行数和列数。此外,如果A是可逆的,则A (A Remark (Sum and Product of Symmetric Matrices)是可逆的。对称矩阵 $A, B \in R^{n \times n}$ 的和总是对称的。, 和然 $\overline{(-1)} = (-\overline{1})^{-1} = \overline{(-1)}$ 。

而,尽管它们的乘积总是定义的,但它通常不对称:

$$\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 1 \end{array} = \begin{array}{cc} 1 & 1 \\ 0 & 0 \end{array} \quad (2.32)$$



2.2.3 标量乘法

让我们看看当矩阵乘以标量 $\lambda \in R$ 时会发生什么。令 $A \in R^{m \times n}$ 和 $\lambda \in R$,然后 $\lambda A = K$, $K_{ij} = \lambda a_{ij}$ 。

实际上, λ 缩放A的每个元素。对于 $\lambda, \psi \in R$,以下成立:

结合性

- 结合性: $(\lambda\psi)C = \lambda(\psi C)$, $C \in R^{m \times n}$
- $= B(\lambda C) = (\lambda B)C$, $B \in R^{m \times n}$ 请注意, 这允许我们移动标量值左 , $C \in R^{n \times k}$ 右。

分配率

- $(\lambda C) = C \quad \lambda = C \quad \lambda = \lambda C$ 因为 $\lambda = \lambda$ 对于所有 $\lambda \in R$ 。
 - 分配性: $(\lambda + \psi)C = \lambda C + \psi C$, $C \in R^{m \times n}$
- $$\lambda(B + C) = \lambda B + \lambda C, \quad B, C \in R^{m \times n}$$

示例 2.5 (分配率)

如果我们定义

$$\text{丙} := \begin{matrix} 1 & 2 & 3 \\ 4 & & \end{matrix}, \quad (2.33)$$

然后对于任何 $\lambda, \psi \in R$ 我们得到 $(\lambda + \psi)C =$

$$\begin{aligned} (\lambda + \psi)C &= \begin{matrix} \psi & 1 & (\lambda + \psi)2 & (\lambda + \psi)3 \\ (\lambda + \psi)4 & \psi & 2\psi & 3\psi & 4\psi \end{matrix} = \begin{matrix} \lambda + \psi & 2\lambda + 2\psi & 3\lambda + 3\psi \\ 4\lambda + 4\psi & & \end{matrix} \end{aligned} \quad (2.34a)$$

$$= \begin{matrix} \lambda & 2\lambda & 3\lambda \\ 4\lambda & & \end{matrix} + \begin{matrix} & & \\ & & \end{matrix} = \lambda C + \psi C. \quad (2.34b)$$

2.2.4 线性方程组的紧凑表示

如果我们考虑线性方程组

$$\begin{aligned} 2x_1 + 3x_2 + 5x_3 &= 1 \\ 4x_1 - 2x_2 - 7x_3 &= \\ 8x_1 + 5x_2 - 3x_3 &= 2 \end{aligned} \quad (2.35)$$

并使用矩阵乘法的规则, 我们可以将这个方程组写成更紧凑的形式

$$\begin{matrix} 2 & 3 & 5 \\ 4 & -2 & -7 \\ 9 & 5 & -3 \end{matrix} \begin{matrix} x_1 \\ x_2 \\ x_3 \end{matrix} = \begin{matrix} 1 \\ 2 \\ 3 \end{matrix}. \quad (2.36)$$

请注意, x_1 缩放第一列, x_2 缩放第二列, x_3 缩放第三列。通常, 线性方程组可以用矩阵形式紧凑地表示为 $Ax = b$; 参见 (2.3), 乘积 Ax 是 A 的列的 (线性) 组合。我们将在中讨论线性组合

在第 2.5 节中有更多详细信息。

2.3 求解线性方程组

27

2.3 求解线性方程组

在 (2.3) 中, 我们引入了方程组的一般形式, 即

$$\begin{aligned} a_{11}x_1 + \dots + a_{1n}x_n &= b_1 \\ &\vdots \\ a_{m1}x_1 + \dots + a_{mn}x_n &= b_m \end{aligned} \quad (2.37)$$

其中 $a_{ij} \in \mathbb{R}$ 和 $b_i \in \mathbb{R}$ 是已知常数, x_j 是未知数, n 到目前为止, 我们看到矩阵可以用式, 这样我们就可以写出 $Ax = b$ ($i = 1, \dots, m, j = 1, \dots, n$) 一种表达线性方程组的紧凑方式, 见 (2.10)。此外, 我们定义了基本的矩阵运算, 例如矩阵的加法和乘法。在下文中, 我们将专注于求解线性方程组, 并提供一种求逆矩阵的算法。

2.3.1 特殊和通用解决方案

在讨论如何一般地求解线性方程组之前, 让我们看一个例子。考虑方程组

$$\begin{array}{cccc|c} & & x_1 & & \\ 1 & 0 & 8 & -4 & 0 & 1 & 2 & 1 & 2 & 1 & 2 \\ & x_2 & = & 42 \\ & x_3 & & & & \downarrow & & & & & \\ & x_4 & & & & & 8 \text{ 个} & & & & & \end{array} \quad (2.38)$$

该系统有两个方程和四个未知数。因此, 通常我们会期望有无限多个解。这个方程组的形式特别简单, 其中前两列由 1 和 0 组成。请记住, 我们要找到标量 x_1, \dots, x_4 , 使得 $\sum x_i c_i = b$, 其中我们将 c_i 定义为矩阵的第 i 列, 将 b 定义为 (2.38) 的右侧。通过将第一列乘以 42 次, 将第二列乘以 8 次, 可以立即找到 (2.38) 中问题的解决方案, 以便

$$b = \begin{matrix} 42 \\ 8 \end{matrix} = 42 \begin{matrix} 1 \\ 0 \end{matrix} + 8 \begin{matrix} 0 \\ 1 \end{matrix} \quad (2.39)$$

因此, 解是 $[42, 8, 0, 0]$ 。该溶液称为特定的特定溶液或特殊溶液。然而, 这并不是这个线性方程组的特殊解的唯一解。为了捕获所有其他解决方案, 我们需要创造性地使用矩阵的列以非平凡的方式生成 0: 将 0 添加到我们的特殊解决方案不会改变特殊解决方案。为此, 我们使用前两列 (非常简单的形式) 表示第三列

$$\begin{matrix} 8 \\ 2 \end{matrix} = 8 \begin{matrix} 1 \\ 0 \end{matrix} + 2 \begin{matrix} 0 \\ 1 \end{matrix} \quad (2.40)$$

因此 $0 = 8c_1 + 2c_2 - 1c_3 + 0c_4$ 和 $(x_1, x_2, x_3, x_4) = (8, 2, -1, 0)$ 。事实上，通过 $\lambda_1 \in \mathbb{R}$ 对该解进行任何缩放都会产生 0 向量，即，

$$\begin{array}{rcccl} & & 8\text{个} & & \\ 1 & 0 & 8 & -4 & \\ 0 & 1 & 2 & 12 & \\ & & & -1 & \\ & & & 0 & \\ \lambda_1 & & & & \\ & & & & = \lambda_1(8c_1 + 2c_2 - c_3) = 0. \end{array} \quad (2.41)$$

按照同样的推理，我们使用前两列表示 (2.38) 中矩阵的第四列，并生成另一组 0 的非平凡版本作为

$$\begin{array}{rcccl} & & -4 & & \\ 1 & 0 & 8 & -4 & 0 & 1 & 2 & 12 \\ 12 & & & & 0 & \\ & & & & -1 & \\ \lambda_2 & & & & & \\ & & & & & = \lambda_2(-4c_1 + 12c_2 - c_4) = 0 \end{array} \quad (2.42)$$

对任意 $\lambda_2 \in \mathbb{R}$ 。综合起来，我们得到 (2.38) 方程组的所有解，称为通解，作为集合

一般解决方案

$$x \in \mathbb{R}^4 : x = \begin{matrix} 42 \\ 0 \\ 0 \\ 0 \end{matrix} + \lambda_1 \begin{matrix} 8 \\ -1 \\ 0 \\ 0 \end{matrix} + \lambda_2 \begin{matrix} -4 \\ 12 \\ 0 \\ -1 \end{matrix}, \lambda_1, \lambda_2 \in \mathbb{R}. \quad (2.43)$$

评论。我们遵循的一般方法包括以下三个步骤：

1. 求 $Ax = b$ 的特解。
2. 找出 $Ax = 0$ 的所有解。
3. 将步骤 1. 和 2. 的解决方案合并为通用解决方案。

一般解和特殊解都不是唯一的。 ◇ 前面例子中的线性方程组很容易求解，因为

(2.38) 中的矩阵具有这种特别方便的形式，它使我们能够通过检查找到特解和通解。然而，一般方程系统不是这种简单的形式。

幸运的是，存在一种建设性的算法方法可以将任何线性方程组转换为这种特别简单的形式：高斯消元法。高斯消去法的关键是线性方程组的初等变换，将方程组变换为简单形式。然后，我们可以将这三个步骤应用到我们刚刚在 (2.38) 示例的上下文中讨论的简单形式。

2.3.2 初等变换求解线性方程组的关键

初等变换

是初等变换，它保持解集相同，但将方程系统转换为更简单的形式：

2.3 求解线性方程组

29

- 交换两个方程 (矩阵中的行代表方程组)

- 方程 (行)与常数 $\lambda \in \mathbb{R} \setminus \{0\}$ 的乘积
- 添加两个方程 (行)

示例 2.6 对于

$a \in \mathbb{R}$, 我们寻求以下方程组的所有解:

$$\begin{aligned} -2x_1 + 4x_2 - 2x_3 - x_4 + 4x_5 &= -3 \\ 3x_4 + x_5 &= 2 \\ x_1 - 2x_2 + x_3 - x_4 + x_5 &= 0 \\ 3x_4 + 4x_5 &= a \end{aligned} \quad . \quad (2.44)$$

我们首先将这个方程组转换为紧凑的矩阵符号 $Ax = b$ 。我们不再明确提及变量 x 并构建增广矩阵 (以 $A | b$ 的形式)

增广矩阵

$$\left[\begin{array}{ccccc|c} -2 & 4 & -2 & -1 & 4 & -3 \\ 1 & 1 & -2 & 1 & -1 & 1 \\ 1 & -2 & 1 & -1 & 2 & 0 \\ 3 & 4 & 0 & -3 & 4 & a \end{array} \right] \quad \begin{matrix} A & & & & & \\ & \text{与 R3 交换} & & & & \\ & & \text{与 R1 交换} & & & \\ & & & & & \end{matrix}$$

在 (2.44) 中, 我们使用垂直线将左侧与右侧分开。我们使用 $|$ 表示使用初等变换的增广矩阵变换。

交换第 1 行和第 3 行导致

$$\left[\begin{array}{ccccc|c} 1 & -2 & 1 & -1 & 0 & 0 \\ 4 & -8 & -2 & 3 & -3 & 2 \\ -2 & -1 & 4 & 3 & 4 & -4R1 \\ 1 & -2 & 0 & -3 & A & +2R1 \\ & & & & & -R1 \end{array} \right]$$

增广矩阵 $A | b$ 紧凑地表示线性方程组 $Ax = b$ 。

当我们现在应用指定的转换 (例如, 从第 2 行中减去第 1 行四次) 时, 我们得到

$$\left[\begin{array}{ccccc|c} 1 & -2 & 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 & -3 & 2 \\ 0 & 0 & 0 & -3 & 6 & -3 \\ 0 & 0 & -1 & -2 & 3 & A \\ 1 & -2 & 0 & 1 & -1 & -R2 - R3 \\ -3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0+1 \end{array} \right] \quad \begin{matrix} & & & & & \\ & \cdot(-1) & & & & \\ & \cdot(-1) & & & & \\ & & & & & \end{matrix}$$

行梯形

此 (增广)矩阵采用方便的形式,即行阶梯形式(REF)。使用我们寻找的变量将这种紧凑的表示法恢复为显式表示法,我们得到

$$\begin{aligned} x_1 - 2x_2 + x_3 - x_4 + x_5 &= 0 \\ x_3 - x_4 + 3x_5 &= -2 \\ x_4 - 2x_5 &= 1 \end{aligned} \quad . \quad (2.45)$$

特解

只有对于 $a = -1$ 这个系统才能解决。一种特殊的解决方案是

$$\begin{array}{ccccc} x_1 & & 2\downarrow & & \\ x_2 & & 0 & & \\ \hline x_3 & = & & & \\ x_4 & & 1\downarrow & & \\ x_5 & & 0 & & \end{array} \quad . \quad (2.46)$$

一般解决方案

捕获所有可能解决方案的集合的一般解决方案是

$$\begin{array}{ccccc} & 2\downarrow & & 2\downarrow & 2\downarrow \\ & 0 & & 1\downarrow & 0 \\ x \in \mathbb{R}^5 : x = & + \lambda_1 & \cdot & + \lambda_2 & , \lambda_1, \lambda_2 \in \mathbb{R} \\ & 1\downarrow & & 0 & 2\downarrow \\ & 0 & & 0 & 1\downarrow \end{array} \quad . \quad (2.47)$$

在下文中,我们将详细介绍一种构造性的方法来获得线性方程组的特解和通解。

枢

备注 (枢轴和楼梯结构)。一行的前导系数 (从左边开始的第一个非零数) 称为主元,并且始终严格位于其上方行的主元的右侧。因此,任何行阶梯形式的方程系统总是具有“阶梯”结构。 ◇

行梯形

定义 2.6 (行梯队形式)。一个矩阵是行梯形的,如果

- 所有只包含零的行都在矩阵的底部;相应地,包含至少一个非零元素的所有行位于仅包含零的行的顶部。
- 仅查看非零行,从左边开始的第一个非零数字 (也称为主元或前导系数) 始终严格位于其上方行的主元的右侧。

pivot

leading coefficient

在其他文本中,有时要求 pivot 为 1。

基本变量
自由变量

备注 (基本和自由变量)。行阶梯形式中与枢轴对应的变量称为基本变量,其他变量为自由变量。例如,在 (2.45) 中, x_1, x_3, x_4 是基本变量,而 x_2, x_5 是自由变量。

备注 (获得特定解决方案)。行梯队形式使得

2.3 求解线性方程组

31

当我们需要确定特定的解决方案时,我们的生活会更轻松。为此,我们使用主元列表示方程系统的右侧,例如 $b = \lambda_i p_i$,其中 p_i , $i = 1, \dots, P$,是枢轴列。如果我们从最右边的主元列开始并向左移动,则 λ_i 最容易确定。

在前面的示例中,我们将尝试找到 λ_1 、 λ_2 、 λ_3 ,以便

$$\begin{array}{ccccccccc} & & & & -1 & & 0 \\ \lambda_1 & 0 & + \lambda_2 & 0 & + \lambda_3 & 0 & = & -2 \\ & 0 & & 0 & & 0 & & . \end{array} \quad (2.48)$$

从这里,我们相对直接地发现 $\lambda_3 = 1$, $\lambda_2 = -1$, $\lambda_1 = 2$ 。当我们把所有东西放在一起时,我们不能忘记我们将系数隐式设置为0的非主元列。因此,我们得到特解 $x = [2, 0, -1, 1, 0]$ 。 ◇备注(减少行梯队形式)。如果行阶梯形式——

- 它是行梯队形式。
- 每个枢轴都是1。
- 主元是其列中唯一的非零条目。



简化的行阶梯形式将在后面的 2.3.3 节中发挥重要作用,因为它允许我们以直接的方式确定线性方程组的通解。

高斯
备注(高斯消除)。高斯消去法是一种消去法执行初等变换以将线性方程组简化为行阶梯形式的算法。 ◇

示例 2.7 (减少行阶梯形)

验证以下矩阵是否为简化的行阶梯形式(主元以粗体显示)：

$$\begin{array}{ccccccccc} & & & & 1 & 3 & 0 & 0 & 3 \\ \text{一个}= & & & & 0 & 0 & 1 & 0 & 9 \\ & & & & 0 & 0 & 0 & 1 & -4 \\ & & & & & & & & . \end{array} \quad (2.49)$$

找到 $Ax = 0$ 的解的关键思想是查看非主元列,我们需要将其表示为主元列的(线性)组合。简化的行阶梯形式使这相对简单,我们用它们左侧的主元列的总和倍数来表示非主元列:第二列是第一列的3倍(我们可以忽略主元第二列右侧的列)。因此,要获得0,我们需要减去

第二列是第一列的三倍。现在,我们查看第五列,这是我们的第二个非数据透视列。第五列可以表示为第一个主元列的3倍,第二个主元列的9倍,第三个主元列的-4倍。我们需要跟踪数据透视列的索引并将其转换为第一列的3倍,第二列(非数据透视列)的0倍,第三列(第二个数据透视列)的9倍),和-4乘以第四列(这是第三个主元列)。然后我们需要减去第五列得到0。最后,我们仍然在求解一个齐次方程组。

总而言之, $Ax = 0, x \in R^5$ 的所有解由下式给出

$$x \in R^5 : x = \lambda_1 \begin{pmatrix} 3 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0 \\ 0 \\ 9 \\ -4 \\ -1 \end{pmatrix}, \lambda_1, \lambda_2 \in R \quad (2.50)$$

2.3.3 减一技巧

下面,我们介绍一个实用技巧,用于读出齐次线性方程组 $Ax = 0$ 的解 x ,其中

$A \in R^{k \times n}$, $x \in R^n$ 。

首先,我们假设 A 是简化的行梯形形式,没有任何只包含零的行,即

$$\begin{array}{ccccccccc} 0 & \cdots & 0 & 1 & * & \cdots & * & 0 & * \cdots * \\ \vdots & & & \vdots & 0 & 0 & \cdots & 0 & 1 & * & \cdots & * & \vdots & \vdots & \vdots \\ \text{一个}= & \vdots & & \vdots & \vdots & \vdots & & \vdots & 0 & & \vdots & & \vdots & \vdots & \vdots \\ \vdots & & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & 0 & 0 & \vdots & \vdots \\ \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & * & \cdots & * & & & & \end{array}, \quad (2.51)$$

其中*可以是任意实数,每行的第一个非零条目必须为1,并且相应列中的所有其他条目必须为0。列 $1, \dots, j_k$ 和枢轴(在 $e_k \in R^k$ 中标记)我们将这个矩阵扩展为粗体)是标准单位向量 e_1, \dots ,通过添加 $n - k$ 行的形式到 $n \times n$ 矩阵 A

$$0 \cdots 0 -1 0 \cdots 0 \quad (2.52)$$

使得增广矩阵 A 的对角线包含1或-1。

那么,包含-1作为主元的 A 的列是

2.3 求解线性方程组

33

齐次方程组 $Ax = 0$ 。更准确地说,这些列构成 $Ax = 0$ 的解空间的基础 (第 2.6.1 节),我们稍后将其称为核或零空间 (参见第 2.7.3 节)。

核心
零空间

例 2.8 (减一技巧)

让我们重新审视 (2.49) 中的矩阵,它已经在 REF 中:

$$\begin{array}{r} 1 \ 3 \ 0 \ 0 \ 3 \\ \text{一个=} \quad 0 \ 0 \ 1 \ 0 \ 9 \\ \quad \quad \quad 0 \ 0 \ 0 \ 1 \ -4 \end{array} \quad (2.53)$$

我们现在通过在对角线上的枢轴缺失的地方添加形式为 (2.52) 的行来将该矩阵扩充为 5×5 矩阵,并获得

$$\begin{array}{r} 1 \ 3 \ 0 \ 0 \ 3 \\ 0 \ -1 \ 0 \ 0 \\ A \sim = \quad 0 \ 0 \ 1 \ 0 \ 9 \\ \quad 0 \ 0 \ 0 \ 1 \ -4 \\ \quad 0 \ 0 \ 0 \ 0 \ -1 \end{array} \quad (2.54)$$

从这个形式,我们可以通过取 $A \sim$ 的列立即读出 $Ax = 0$ 的解,在对角线上包含 -1 :

$$\begin{array}{ccccc} & \overset{3\uparrow}{-1} & & \overset{3\uparrow}{0} & \\ 5 \times \in R : x = \lambda_1 & 0 & + \lambda_2 & 9 & -4 \\ & 0 & & -4 & \\ & 0 & & & -1 \end{array}, \lambda_1, \lambda_2 \in R, \quad (2.55)$$

这与我们通过“洞察力”获得的 (2.50) 中的解决方案相同。

计算逆

计算满足 $AX = I_n$ 的逆 A^{-1} 。然后, $A \in R^n \times n$ 我们需要找到一个矩阵 $X = A^{-1}$ 一组联立线性方程 $AX = I_n$, 其中我们求解 X^{-1} 。我们可以把它写成 $= [x_1 | \cdots | x_n]$ 。我们使用增广矩阵符号来紧凑表示这组线性方程组并获得

$$A | I_n \quad \cdots \quad I_n | A^{-1} \quad (2.56)$$

这意味着,如果我们将增广方程组转化为简化的行阶梯形式,我们就可以读出方程组右侧的逆矩阵。因此,确定矩阵的逆矩阵等同于求解线性方程组。

示例 2.9 (通过高斯消元法计算逆矩阵)

确定的倒数

$$\text{一个} = \begin{array}{r} 1020 \\ 1100 \\ 1100 \\ 1201 \\ 1111 \end{array} \quad (2.57)$$

我们写下增广矩阵

$$\left| \begin{array}{cccccc|c} 1 & 0 & 2 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ \hline & & & & & & & \end{array} \right|$$

并使用高斯消去法将其变为简化的行梯形

$$\left| \begin{array}{cccccc|c} 1 & 0 & 0 & -1 & 2 & -2 & 2 \\ 0 & 1 & 0 & 1 & -1 & 2 & -2 \\ 0 & 0 & 1 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 1 & -1 & 0 & -1 \\ \hline & & & & & & \end{array} \right|,$$

这样所需的逆作为其右侧给出：

$$A^{-1} = \begin{array}{r} -1 & 2 & -2 & 2 \\ 1 & -1 & 2 & -2 \\ 1 & -1 & 1 & -1 \\ -1 & 0 & -1 & 2 \end{array} \quad (2.58)$$

我们可以通过执行乘法 AA^{-1} 并观察我们恢复 I_4 来验证 (2.58) 确实是逆的。

2.3.4 求解线性方程组的算法

下面,我们将简要讨论求解 $Ax = b$ 形式的线性方程组的方法。我们假设存在解决方案。如果没有解,我们需要求助于近似解,本章不涉及。解决近似问题的一种方法是使用线性回归方法,我们将在第 9 章中详细讨论。

在特殊情况下,我们可以确定逆 A^{-1} , 这样的只有当 A 是方阵且可逆时, $Ax = b$ 的解才可能为 $x = A^{-1}b$ 。然而,这是但通常情况并非如此。否则,在温和的假设下(即 A 需要具有线性独立的列)我们可以使用变换

$$Ax = b \Leftrightarrow A^{-1}Ax = A^{-1}b \Leftrightarrow x = (A^{-1}A)^{-1}A^{-1}b \quad (2.59)$$

并使用求解 $Ax = b$ 的 Moore-Penrose 伪逆 $(A^T A)^{-1} A^T b$ (2.59) ,这也 确定Moore-Penrose 伪逆 对应于最小范数最小二乘解。这种方法的一个缺点是它需要对矩阵矩阵乘积进行多次计算并计算 $A^T A$ 的倒数。此外,出于数值精度的原因,通常不建议计算逆或伪逆。

因此,在下文中,我们将简要讨论求解线性方程组的替代方法。

高斯消元在计算行列式 (第 4.1 节)、检查一组向量是否线性独立 (第 2.5 节)、计算矩阵的逆 (第 2.2.2 节)、计算矩阵的秩 (2.6.2 节) ,以及确定向量空间的基 (2.6.1 节) 。高斯消元法是求解具有数千个变量的线性方程组的一种直观且有建设性的方法。然而,对于具有数百万个变量的系统,这是不切实际的,因为所需的算术运算数量与联立方程的数量呈三次方关系。

在实践中,许多线性方程组通过固定迭代方法间接求解,例如 Richardson 方法、Jacobi 方法、Gauß-Seidel 方法和逐次超松弛法,或 Krylov 子空间方法,例如作为共轭梯度、广义最小残差或双共轭梯度。我们参考了 Stoer 和 Burlirsch (2002 年)、Strang (2003 年) 以及 Liesen 和 Mehrmann (2015 年) 的书籍以了解更多详细信息。

令 x^* 为 $Ax = b$ 的解。这些迭代方法的关键思想就是设置一个迭代的形式

$$x^{(k+1)} = Cx(k) + d \quad (2.60)$$

对于合适的 C 和 d ,在每次迭代中减少残差 $\|x^{(k+1)} - x\|$ 并收敛到 x^* 。我们将在 3.1 节中介绍范数 $\|\cdot\|$,它允许我们计算向量之间的相似性。

2.4 向量空间到目前为止,

我们已经研究了线性方程组以及如何求解它们 (第 2.3 节)。我们看到线性方程组可以用矩阵向量表示法 (2.10) 紧凑地表示。在下文中,我们将仔细研究向量空间,即向量存在的结构化空间。

在本章的开头,我们非正式地将向量描述为可以加在一起并乘以标量的对象,并且它们仍然是同一类型的对象。现在,我们准备好将其形式化,我们将从介绍组的概念开始,它是一组元素和定义在这些元素上的操作,以保持集合的某些结构不变。

2.4.1 群组

组在计算机科学中扮演着重要的角色。除了为集合操作提供基本框架外，它们还大量用于密码学、编码理论和图形学。

定义 2.7 (组)。考虑一个集合 G 和一个操作 $\otimes : G \times G \rightarrow G$ 定义在 G 上。然后 $G := (G, \otimes)$ 被称为一个群如果满足以下条件：

团体
关闭
结合性
中性元素
逆元

1. G 在 \otimes 下的闭包： $\forall x, y \in G : x \otimes y \in G$
2. 结合性： $\forall x, y, z \in G : (x \otimes y) \otimes z = x \otimes (y \otimes z)$
3. 中性元： $\exists e \in G \forall x \in G : x \otimes e = x$ and $e \otimes x = x$
4. 逆元： $\forall x \in G \exists y \in G : x \otimes y = e$ and $y \otimes x = e$, 其中 e 表示逆元
中性元素。我们经常写 x^{-1} of x 。

评论。逆元素是相对于操作 \otimes 定义的，并不一定意味着 \diamond 如果另外 $\forall x, y \in G : x \otimes y = y \otimes x$, 则 $G = (G, \otimes)$ 是阿贝尔群（交换群）。

阿贝尔群

示例 2.10 (组)

让我们看一些具有关联操作的集合示例，看看它们是否是组：

$\text{否 } 0 := N \cup \{0\}$

- $(Z, +)$ 是阿贝尔群。
 - $(N \setminus \{0\}, +)$ 不是一个群：虽然 $(N \setminus \{0\}, +)$ 有一个中性元素 (0) , 但缺少逆元素。
 - (Z, \cdot) 不是一个群：虽然 (Z, \cdot) 包含一个中性元素 (1) , 但缺少任何 $z \in Z, z \neq \pm 1$ 的逆元素。
 - (R, \cdot) 不是群，因为 0 不具有逆元素。
 - $(R \setminus \{0\}, \cdot)$ 是阿贝尔矩阵。
 - $(R^n, +), (Z^n, +), n \in N$ 是 Abelian 如果 $+$ 是按分量定义的，即 $(x_1, \dots, x_n) + (y_1, \dots, y_n) = (x_1 + y_1, \dots, x_n + y_n)$ 。 (2.61)
- 那么， $(x_1, \dots, x_n)^{-1} := (-x_1, \dots, -x_n)$ 是逆元， $e = (0, \dots, 0)$ 是中性元。
- $(R^m \times N, +)$, $m \times n$ 矩阵的集合是阿贝尔矩阵（具有如 (2.61) 中定义的逐分量加法）。
 - 让我们仔细看看 $(R^n \times n, \cdot)$, 即 (2.13) 中定义的具有矩阵乘法的 $n \times n$ 矩阵集。

-闭包和结合律直接来自矩阵乘法的定义。

-中性元素：单位矩阵 I_n 是中性元素
关于 $(R^n \times n, \cdot)$ 中的矩阵乘法 “ \cdot ”。

-逆元:如果逆元存在(A是正则的),则A的逆元 $\in R^{n \times n}$ 并且此时 $(R^{n \times n}, \cdot)$ 是 $^{-1}$ 个群,称为一般线性群。

定义 2.8 (一般线性群)。正则(可逆)矩阵集 $A \in R^{n \times n}$ 是关于矩阵乘法的一个群,如(2.13)中定义的,称为一般线性群 $GL(n, R)$ 。然而,一般线性群由于矩阵乘法不可交换,所以该群不是阿贝尔群。

2.4.2 向量空间

当我们讨论组时,我们查看了集合G和G上的内部操作,即仅对G中的元素进行操作的映射 $G \times G \rightarrow G$ 。在下文中,我们将考虑除了内部操作+乘法之外的集合向量 $x \in G$ 还包含一个外部运算·,一个标量 $\lambda \in R$ 。我们可以将内部运算视为一种加法形式,将外部运算视为一种缩放形式。请注意,内部/外部操作与内部/外部产品无关。

定义 2.9 (向量空间)。一个实值向量空间 $V = (V, +, \cdot)$ 是一个有两个操作的集合V的向量空间

$$+: V \times V \rightarrow V \quad (2.62)$$

$$\cdot: R \times V \rightarrow V \quad (2.63)$$

在哪里

1. $(V, +)$ 是阿贝尔群 2. 分配性:

1. $\forall \lambda \in R, x, y$

$$\begin{aligned} & \in V : \lambda \cdot (x + y) = \lambda \cdot x + \lambda \cdot y \\ & V : (\lambda + \psi) \cdot x = \lambda \cdot x + \psi \cdot x \end{aligned}$$

3. 结合性(外运算): $\forall \lambda, \psi \in R, x \in V : \lambda \cdot (\psi \cdot x) = (\lambda \psi) \cdot x$ 4. 外运算的中性元素: $\forall x \in V : 1 \cdot x = x$

元素 $x \in V$ 称为向量。 $(V, +)$ 的中性元素是向量零向量 $0 = [0, \dots, 0]$,内层运算+称为向量向量相加加法。元素 $\lambda \in R$ 称为标量,外运算标量·是与标量的乘法。请注意,标量乘积是不同的乘积,我们将在第3.2节中介绍

标量

评论。“向量乘法” $ab, a, b \in R^n$ 理论上,我们可以定义逐元,没有定义。定理
素乘法,使得 $c = ab$ 且 $c_j = a_j b_j$ 。这种“数组乘法”在许多编程语言中都很常见,但使用矩阵乘法的标准规则在数学上的意义有限。通过将向量视为 $n \times 1$ 矩阵

(我们通常这样做),我们可以使用(2.13)中定义的矩阵乘法。但是,向量的维度不匹配。仅定义了以下向量乘法: $ab \in \mathbb{R}^{n \times n}$ (外积), $a, b \in \mathbb{R}$ (内积/标量/点积)。

◇

外积

例 2.11 (向量空间)

让我们看一些重要的例子:

- $V = \mathbb{R}^n, n \in \mathbb{N}$ 是一个向量空间,其运算定义如下: -加法: $x+y = (x_1, \dots, x_n) + (y_1, \dots, y_n) = (x_1+y_1, \dots, x_n+y_n)$ 对于所有 $x, y \in \mathbb{R}^n$ -乘以标量: $\lambda x = \lambda(x_1, \dots, x_n) = (\lambda x_1, \dots, \lambda x_n)$ 对于所有 $\lambda \in \mathbb{R}, x \in \mathbb{R}^n$,
- $V = \mathbb{R}^{m \times n}, m, n \in \mathbb{N}$ 是向量空间

$$\begin{array}{lll} a_{11} + b_{11} & \cdots & a_{1n} + b_{1n} \\ \text{-加法: } A + B = & \vdots & \vdots \quad \text{定义元素} \\ am_1 + bm_1 & \cdots & am_n + bn \\ \text{对所有 } A, B \in V & & \lambda a_{11} \cdots \lambda a_{1n} \\ \text{-乘以标量: } \lambda A = & & \vdots \quad \vdots \quad \text{如定义} \\ \lambda am_1 & \cdots & \lambda am_n \end{array}$$

第 2.2 节。请记住, $\mathbb{R}^{m \times n}$ 等同于 \mathbb{R}^{mn} 。

- $V = \mathbb{C}$, 符合复数加法的标准定义。

评论。下面,当+和·是标准向量加法和标量乘法时,我们将用V表示一个向量空间($V, +, \cdot$)。

此外,我们将对V中的向量使用符号 $x \in V$ 以简化符号。 ◇

评论。向量空间 \mathbb{R}^n 我们写向量。下面不, $\mathbb{R}^{n \times 1}, \mathbb{R}^{1 \times n}$ 只是方式不同区分 \mathbb{R}^n 和 $\mathbb{R}^{n \times 1}$

列向量

, 这允许我们将 n 元组写为列向量

$$x = \begin{matrix} x_1 \\ \vdots \\ x_n \end{matrix} \quad (2.64)$$

这简化了关于向量空间操作的符号。但是,我们确实区分了 $\mathbb{R}^{n \times 1}$ 和 $\mathbb{R}^{1 \times n}$ (行向量)以避免与矩阵乘法混淆。默认情况下,我们写 x 来表示 x 的转置。 ◇ umn 向量, 行向量由 x 表示

转置

2.4.3 向量子空间下面介绍

向量子空间。直观上,它们是包含在原始向量空间中的集合,具有这样的性质:当我们对该子空间内的元素执行向量空间操作时,我们将永远不会离开它。从这个意义上说,它们是“封闭的”。向量子空间是机器学习中的一个关键思想。例如,第 10 章演示了如何使用向量子空间进行降维。

定义 2.10 (向量子空间)。令 $V = (V, +, \cdot)$ 为向量空间且 $U \subseteq V, U \neq \emptyset$ 。那么 $U = (U, +, \cdot)$ 被称为 V 的向量子空间(或向量子空间线性子空间)如果 U 是一个向量空间,向量空间操作+线性子空间并且限制为 $U \times U$ 和 $\mathbb{R} \times U$ 。我们写 $U \subseteq V$ 来表示 V 的子空间 U 。

如果 $U \subseteq V$ 和 V 是一个向量空间,那么 U 自然会直接从 V 继承许多属性,因为它们对所有 $x \in V$,特别是对所有 $x \in U \subseteq V$ 都成立。这包括阿贝尔群属性,分布ativity、结合律和中性元素。要确定 $(U, +, \cdot)$ 是否是 V 的子空间,我们仍然需要证明

1. $U = \emptyset$, 特别是: $0 \in U$
2. U 的闭包:

A. 关于外操作: $\forall \lambda \in \mathbb{R} \forall x \in U : \lambda x \in U$. b. 关于内运算: $\forall x, y \in U : x + y \in U$.

例 2.12 (向量子空间)

让我们看一些例子:

- 对于每个向量空间 V 只有图 , 平凡子空间是 V 本身和 $\{0\}$ 。
- 2.1 中的示例 D 是 \mathbb{R}^2 的子空间(具有通常的内部/外部操作)。在 A 和 C 中, 闭包性质被违反; B 不包含 0。
- 齐次线性方程组的解集 $Ax = 0$ 有 n 个未知数 $x = [x_1, \dots, x_n]$ 是 \mathbb{R}^n 的子空间。
- 非齐次线性方程组 $Ax = b$, $b \neq 0$ 的解不是 \mathbb{R}^n 的子空间。
- 任意多个子空间的交集本身就是一个子空间。

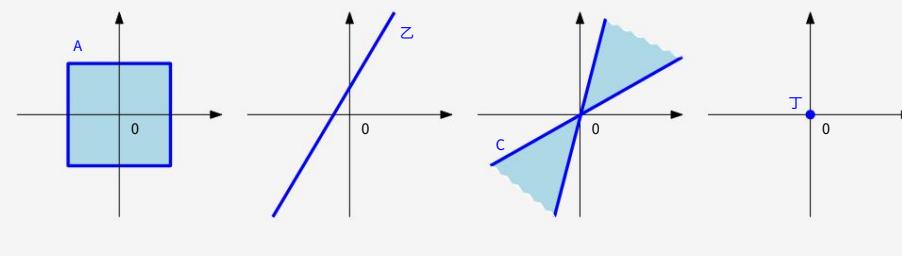


图 2.1 并非 \mathbb{R}^2 的所有子集都是子空间。在 A 和 C 中, 闭包性质被违反; B 不包含 0。只有 D 是子空间。

评论。每个子空间 $U \subseteq (\mathbb{R}^n, +, \cdot)$ 是一个同族 \diamond 线性方程组 $Ax = 0$ 的解空间, 其中 $x \in \mathbb{R}^n$ 。

2.5 线性独立

在下文中, 我们将仔细研究我们可以用向量 (向量空间的元素) 做什么。特别是, 我们可以将向量相加并将它们与标量相乘。闭包属性保证我们最终会在同一个向量空间中得到另一个向量。有可能找到一组向量, 我们可以通过将它们加在一起并缩放来表示向量空间中的每个向量。这组向量是一个基础, 我们将在 2.6.1 节中讨论它们。在我们到达那里之前, 我们需要介绍线性组合和线性独立性的概念。

定义 2.11 (线性组合)。考虑向量空间 V 和有限数量的向量 $x_1, \dots, x_k \in V$ 。然后, 每个 $v \in V$ 的形式

$$v = \lambda_1 x_1 + \dots + \lambda_k x_k = \sum_{i=1}^k \lambda_i x_i \in V \quad (2.65)$$

与 $\lambda_1, \dots, \lambda_k \in \mathbb{R}$ 是向量 x_1, \dots, x_k 的线性组合。

0 向量始终可以写成 k vec x_k 的线性组合, 因为 $0 = 0x_i$ 始终为真。在下文中, tor x_1, \dots, x_k , 我们感兴趣的是一组表示 $\sum_{i=1}^k \lambda_i x_i = 0$ 的向量的非平凡线性组合, 即向量 x_1, \dots, x_k 的线性组合。其中并非(2.65) 中的所有系数 λ_i 都为 0。

定义 2.12 (线性 (独立) 依赖)。让我们考虑一个向量空间。如果存在一个非平凡的线性组合 $0 = \sum_{i=1}^k \lambda_i x_i$ 且至少有一个 $\lambda_i \neq 0$, $x_1, \dots, x_k \in V$ 。com V , 其中 $k \in \mathbb{N}$ 和 x_1, \dots, x_k 二元化, 使向量 x_k 是线性相关的。如果仅存在平凡解, 即 $\lambda_1 = \dots = \lambda_k = 0$ 向量 x_1, \dots, x_k 是线性无关的。

线性独立性是线性代数中最重要的概念之一。直观地说, 一组线性无关的向量由没有冗余的向量组成, 即, 如果我们从集合中删除这些向量中的任何一个, 我们就会丢失一些东西。在接下来的部分中, 我们将更多地形式化这种直觉。

示例 2.13 (线性相关向量)

一个地理例子可能有助于阐明线性独立的概念。内罗毕 (肯尼亚) 的一个人在描述基加利 (卢旺达) 的位置时可能会说, “你可以先向西北方向行驶 506 公里到达 Kam pala (乌干达), 然后向西南方向行驶 374 公里才能到达基加利。”。这是足够的信息

来描述基加利的位置,因为地理坐标系可以被认为是一个二维向量空间（忽略高度和地球曲面）。此人可能会补充说,“它位于此处以西约751公里处。”尽管这最后一个陈述是正确的,但根据先前的信息,没有必要找到基加利（参见图 2.2 的说明）。在这个例子中,“506 km Northwest”向量（蓝色）和“374 km Southwest”向量（紫色）是线性无关的。这意味着不能用西北矢量来描述西南矢量,反之亦然。然而,第三个“751 km West”向量（黑色）是其他两个向量的线性组合,它使向量集线性相关。等价地,给定“西751公里”和“西南374公里”可以线性组合得到“西北506公里”。



图 2.2 线性相关向量的地理示例（粗略近似于基本方向）

二维的空间（平面）。

评论。以下属性可用于确定向量是否线性无关：

- k 个向量要么是线性相关的,要么是线性独立的。没有第三种选择。
- 如果至少有一个向量 x_1, \dots, x_k 为 0 那么它们是线性相关的。如果两个向量相同,则同样成立。
- 向量 $\{x_1, \dots, x_k : x_i = 0, i = 1, \dots, k\}$, $k \geq 2$, 是线性相关的当且仅当 (至少) 其中一个是其他的线性组合。特别地,如果一个向量是另一个向量的倍数,即 $x_i = \lambda x_j, \lambda \in \mathbb{R}$ 则集合 $\{x_1, \dots, x_k : x_i = 0, i = 1, \dots, k\}$ 是线性相关的。
- 检查向量 $x_1, \dots, x_k \in V$ 线性无关就是用高斯消去法:把所有的向量写成矩阵 A 的列,进行高斯消去,直到矩阵为行阶梯形 (这里不需要简化的行阶梯形) :

- 枢轴列表示向量, 它们与左侧的向量线性无关。请注意, 在构建矩阵时存在向量的排序。

- 非枢轴列可以表示为其左侧枢轴列的线性组合。例如, 行阶梯形式

$$\begin{matrix} 1 & 3 & 0 & 0 & 0 \\ 2 & & & & \end{matrix} \quad (2.66)$$

告诉我们第一列和第三列是数据透视列。第二列是非数据透视列, 因为它是第一列的三倍。

当且仅当所有列都是主元列时, 所有列向量都是线性无关的。如果至少有一个非主元列, 则这些列 (以及相应的向量) 是线性相关的。

◇

示例2.14考虑 R4

$$x_1 = \begin{matrix} 1\downarrow \\ -3 \\ 4\downarrow \end{matrix}, \quad x_2 = \begin{matrix} 1\downarrow \\ 0 \\ 2\downarrow \end{matrix}, \quad x_3 = \begin{matrix} 1\downarrow \\ -2 \\ 1\downarrow \end{matrix}. \quad (2.67)$$

为了检查它们是否线性相关, 我们按照一般方法求解

$$\lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 = \lambda_1 \begin{matrix} 1\downarrow \\ -3 \\ 4\downarrow \end{matrix} + \lambda_2 \begin{matrix} 1\downarrow \\ 0 \\ 2\downarrow \end{matrix} + \lambda_3 \begin{matrix} 1\downarrow \\ -2 \\ 1\downarrow \end{matrix} = 0 \quad (2.68)$$

对于 $\lambda_1, \dots, \lambda_3$, 我们将向量 $x_i i = 1, 2, 3$ 写为矩阵的列, 并应用基本行操作, 直到我们确定主元列:

$$\begin{matrix} 1 & 1 & -1 \\ 2 & 1 & -2 \\ -3 & 0 & 1 \\ 4 & 2 & 1 \end{matrix} \cdots \begin{matrix} 1 & 1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{matrix} \quad (2.69)$$

在这里, 矩阵的每一列都是一个数据透视列。因此, 不存在非平凡解, 我们要求 $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$ 来求解方程组。因此, 向量 x_1, x_2, x_3 是线性无关的。

2.5 线性独立

43

评论。考虑具有 k 个线性无关向量 b_1, \dots, b_k 的向量空间 V 。 \dots, b_k 和 m 线性组合

$$\begin{aligned} x_1 &= \sum_{i=1}^k \lambda_{i1} b_i, \\ &\vdots \\ x_m &= \sum_{i=1}^k \lambda_{im} b_i. \end{aligned} \quad (2.70)$$

定义 $B = [b_1, \dots, b_k]$ 作为列为线性无关向量 b_1, \dots, b_k 的矩阵。我们可以写

$$x_j = B \lambda_j, \quad \lambda_j = \begin{pmatrix} \lambda_{1j} \\ \vdots \\ \lambda_{kj} \end{pmatrix}, \quad j = 1, \dots, m, \quad (2.71)$$

以更紧凑的形式。

我们想测试是否 x_1, \dots, x_m 是线性无关的。为此, 我们遵循 $\sum_{j=1}^m \psi_j x_j = 0$ 时的一般测试方法。

利用 (2.71), 我们得到

$$\sum_{j=1}^m \psi_j x_j = \sum_{j=1}^m \psi_j B \lambda_j = B \sum_{j=1}^m \psi_j \lambda_j. \quad (2.72)$$

这意味着 $\{x_1, \dots, x_m\}$ 是线性独立的当且仅当列向量 $\{\lambda_1, \dots, \lambda_m\}$ 是线性无关的。

◇

评论。在向量空间中, 如果 $m > k$, k 个向量的 m 个线性组合 x_1, \dots, x_k ◇
 k , 则 V 是线性相关的。

例 2.15 考虑一组线性无关

的向量 $b_1, b_2, b_3, b_4 \in \mathbb{R}^n$ 和

$$\begin{aligned} x_1 &= b_1 - 2b_2 + b_3 - b_4 \\ x_2 &= -4b_1 - 2b_2 + 4b_4 \\ x_3 &= 2b_1 + 3b_2 \\ x_4 &= -b_3 - 3b_4 \end{aligned} \quad (2.73)$$

向量 $x_1, \dots, x_4 \in \mathbb{R}^n$ 线性无关? 为了回答这个问题, 我们调查列向量是否

$$\begin{matrix} 1 & -2 & , & -1 \\ -4 & -2 & , & -1 \\ 2 & 0 & , & 1 \\ 17 & -10 & , & 11 \\ -1 & 4 & -3 & \end{matrix} \quad (2.74)$$

是线性独立的。具有系数矩阵的相应线性方程组的简化行阶梯形式

$$\text{一个} = \begin{array}{r} 1 -4 2 17 \\ -2 -2 3 -10 \\ \downarrow 0 -1 11 \\ -1 4 -3 1 \end{array} \quad (2.75)$$

给出为

$$\begin{array}{r} 1 0 0 -7 \\ 0 1 0 -15 \\ 0 0 1 -18 \\ 0 0 0 \end{array} . \quad (2.76)$$

我们看到相应的线性方程组是非平凡可解的:最后一列不是主元列,并且 $x_4 = -7x_1 - 15x_2 - 18x_3$ 。
因此, x_1, \dots, x_4 是线性相关的,因为 x_4 可以表示为 x_1, \dots, x_3 的线性组合。

2.6 依据与排名

在向量空间 V , 我们特别感兴趣的向量集 A 具有这样的性质, 即任何向量 $v \in V$ 都可以通过 A 中向量的线性组合得到。这些向量是特殊向量, 下面我们将对它们进行表征。

2.6.1 发电机组和基础定义 2.13 (发

电机组和跨度)。考虑向量空间 $V = (V, +, \cdot)$ 和向量集 $A = \{x_1, \dots, x_k\} \subseteq V$ 。如果每个向量 $v \in V$ 都可以表示为 x_1 的线性组合, ..., x_k , A 称为 V 的生成集。 A 中向量的所有线性组合的集合是我们写的 $V = \text{span}[A]$

发电机组
跨度

称为 A 的跨度。如果 A 跨越向量空间 V 或 $V = \text{span}[x_1, \dots, x_k]$ 。

生成集是跨越向量(子)空间的向量集, 即每个向量都可以表示为生成集中向量的线性组合。现在, 我们将更具体地描述跨越向量(子)空间的最小生成集。

最小的
基础

定义 2.14 (基础)。考虑向量空间 $V = (V, +, \cdot)$ 和 $A \subseteq V$ 。如果不存在跨越 V 的更小集合 $A' \subsetneq A \subseteq V$, 则称 V 的生成集 A 是最小的。 V 的每个线性独立生成集

是最小的, 称为 V 的基础。

令 $V = (V, +, \cdot)$ 为向量空间且 $B \subseteq V, B = \emptyset$ 。那么,下面的语句是等价的:

基础是最小的
生成集和最大线性独立集

- B 是 V 的基础。
- B 是最小生成集。
- B 是 V 中的最大线性独立向量集,即,向该集合添加任何其他向量将使其线性相关。
- 每个向量 $x \in V$ 都是来自 B 的向量的线性组合,并且每个线性组合都是唯一的,即

$$x = \sum_{i=1}^k \lambda_i b_i = \sum_{i=1}^k \psi_i b_i \quad (2.77)$$

和 $\lambda_i, \psi_i \in R, b_i \in B$ 它遵循 $\lambda_i = \psi_i, i = 1, \dots, k$.

例 2.16

- 在 R^3 , 规范/标准基础是

规范基础

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2.78)$$

- R^3 中的不同碱基是

$$B_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0.5 & 1.8 & -2.2 \\ 0.8 & 0.3 & -1.3 \\ 0.4 & 0.3 & 3.5 \end{pmatrix}. \quad (2.79)$$

- 套装

$$\text{一个} = \begin{pmatrix} 1 & -1 & 1 \\ 2 & 0 & 0 \\ 3 & 0 & 0 \\ 4 & 2 & -4 \end{pmatrix}. \quad (2.80)$$

是线性独立的,但不是 R^4 的生成集(也没有基) :例如,向量 $[1, 0, 0, 0]$ 不能通过 A 中元素的线性组合获得。

评论。每个向量空间 V 都有一个基 B 。前面的例子表明向量空间 V 可以有很多基,即没有唯一的基。但是,所有基都具有相同数量的元素,即基向量。

◇ 基向量

我们只考虑有限维向量空间 V 。
 V 的维数是 V 的基向量的数量
如果 $U \subseteq V$ 是 V 的子空间 , 然后 $\dim(U) \leq \dim(V)$ 且 $\dim(U) =$

$\dim(V)$ 当且仅当 $U = V$ 。直观上，一个向量空间的维数可以认为是这个向量空间中独立方向的个数。

的维度

向量空间对应其
基数

向量。

评论。向量空间的维数不一定是个数

向量中的元素。例如，向量空间 $V = \text{span}[x]$ 是一维的，虽然基向量有两个元素。 ◇ 备注。子空间 $U = \text{span}[x_1, \dots, x_m] \subseteq \mathbb{R}^n$ 可以通过执行以下步骤找到：

1. 将生成向量写成矩阵 A 的列 2. 确定 A 的行阶梯形式。

3. 与数据透视列关联的生成向量是
 U 。

◇

示例 2.17 (确定基数)

对于向量子空间 $U \subseteq \mathbb{R}^5$ ，由向量生成

$$\begin{array}{ccccccccc} & 1\uparrow & & 2\uparrow & & 3\uparrow & & & -1 \\ & 2\uparrow & & & & & & & \\ & & -1 & & & -4 & & & \\ x_1 = & -1 & , & x_2 = & 1\uparrow & , & x_3 = & 3\uparrow & 5 \in \mathbb{R}, \quad (2.81) \\ & -1 & & & 2\uparrow & & 5\uparrow & & -6 \\ & -1 & & -2 & & -3 & & & \\ & & & & & & & & 1\uparrow \end{array}$$

我们有兴趣找出哪些向量 x_1, \dots, x_4 是 U 的基础。

为此，我们需要检查 x_1, \dots, x_4 是线性无关的。

因此，我们需要解决

$$\sum_{i=1}^4 \lambda_i x_i = 0, \quad (2.82)$$

这导致具有矩阵的齐次方程组

$$\begin{array}{ccccccccc} & 1 & 2 & 3 & -1 & & & & \\ & 2 & -1 & -4 & 8 & & & & \\ x_1, x_2, x_3, x_4 = & & & & -1 & 1 & 3 & -5 & . \quad (2.83) \\ & -1 & 2 & 5 & -6 & & & & \\ & -1 & -2 & -3 & 1 & & & & \end{array}$$

利用线性方程组的基本变换规则，我们得到行阶梯形

$$\begin{array}{ccccccccc} & 2 & 3 & -1 & 2 & -1 & & & \\ & -4 & 8 & & & & & & \\ & -1 & 1 & 3 & -5 & & & & \\ & -1 & 2 & 5 & -6 & & & & \\ & -1 & -2 & -3 & 1 & & & & \end{array} \quad \begin{array}{ccccccccc} & 1 & 2 & 3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ & 0 & & & & 1 & & & & & \\ & & & & & 2 & -2 & & & & \\ & & & & & 0 & 0 & 0 & & & \\ & & & & & & & & 1 & & \\ & & & & & & & & & 1 & \\ & & & & & & & & & & . \end{array}$$

由于主元列表示哪一组向量是线性独立的,我们从行阶梯形式可以看出 x_1, x_2, x_4 是线性独立的(因为线性方程组 $\lambda_1x_1 + \lambda_2x_2 + \lambda_4x_4 = 0$ 只能求解其中 $\lambda_1 = \lambda_2 = \lambda_4 = 0$)。因此, $\{x_1, x_2, x_4\}$ 是U的基。

2.6.2 排名

矩阵 $A \in \mathbb{R}^{m \times n}$ 的线性独立列数等于线性独立行数,称为A的秩 rank,记为 $\text{rk}(A)$ 。

评论。矩阵的秩有一些重要的性质:

- $\text{rk}(A) = \text{rk}(A^T A)$, 即列秩等于行秩。
- $\in \mathbb{R}^{m \times n}$ 的列跨越一个子空间 $U \subseteq \mathbb{R}^m$ 且 $\dim(U) = \text{rk}(A)$ 。稍后我们称这个子空间为图像或范围。 U 的基可以是通过对 A 应用高斯消去法来识别数据透视列。
- $A \in \mathbb{R}^{m \times n}$ 的行跨越子空间 $W \subseteq \mathbb{R}^n$, 其中 $\dim(W) = \text{rk}(A)$ 。 W 的基础可以通过应用高斯消元来找到 A^T 。
- 对于所有 $A \in \mathbb{R}^{n \times n}$, 当且仅当 $\text{rk}(A) = n$ 时, 它认为 A 是正则的(可逆的)。
- 对于所有 $A \in \mathbb{R}^{m \times n}$ 和所有 $b \in \mathbb{R}^m$, 它认为线性方程组 $Ax = b$ 可以求解当且仅当 $\text{rk}(A) = \text{rk}(A|b)$, 其中 $A|b$ 表示增广系统。
- 对于 $A \in \mathbb{R}^{m \times n}$, $Ax = 0$ 的解的子空间具有维数 $n - \text{rk}(A)$ 。稍后, 我们将这个子空间称为内核或空内核空间。
- 如果矩阵 $A \in \mathbb{R}^{m \times n}$ 的秩等于相同维度矩阵的最大可能满秩秩, 则该矩阵具有满秩。这意味着满秩矩阵的秩是行数和列数中较小的一个, 即 $\text{rk}(A) = \min(m, n)$ 。如果一个矩阵不是满秩的, 则称它是秩亏的。



示例 2.18 (等级)

- 一个=
$$\begin{matrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{matrix}$$
。
- A 有两个线性独立的行/列,因此 $\text{rk}(A) = 2$ 。

■ 一个 = $\begin{array}{r} 21 \\ -2-31 \\ \hline 50 \end{array}$
我们使用高斯消去法来确定排名：

$$\begin{array}{r} 121 \\ -2-31 \\ \hline 350 \end{array} \quad \dots \quad \begin{array}{r} 121 \\ 013 \\ \hline 000 \end{array} \quad . \quad (2.84)$$

在这里，我们看到线性无关的行数和列数都是 2，这样 $\text{rk}(A) = 2$ 。

2.7 线性映射

在下文中，我们将研究保留其结构的向量空间上的映射，这将使我们能够定义坐标的概念。

在本章开头，我们说过向量是可以加在一起乘以标量的对象，结果对象仍然是向量。我们希望在应用映射时保留此属性：考虑两个实向量空间 V, W ，映射 $\Phi : V \rightarrow W$ 保留向量空间的结构，如果

$$\Phi(x + y) = \Phi(x) + \Phi(y) \quad (2.85)$$

$$= \lambda\Phi(x) \quad (2.86)$$

对于所有 $x, y \in V$ 和 $\lambda \in \mathbb{R}$ 。我们可以将其总结为以下定义：

定义 2.15（线性映射）。对于向量空间 V, W ，映射 $\Phi : V \rightarrow W$ 称为线性映射（或向量空间同态/线性变换），如果

线性映射向量空间

同态

线性的
转型

$$\forall x, y \in V \forall \lambda, \psi \in \mathbb{R} : \Phi(\lambda x + \psi y) = \lambda\Phi(x) + \psi\Phi(y). \quad (2.87)$$

事实证明，我们可以将线性映射表示为矩阵（第 2.7.1 节）。回想一下，我们还可以收集一组向量作为矩阵的列。使用矩阵时，我们必须牢记矩阵代表什么：线性映射或向量集合。我们将在第 4 章看到更多关于线性映射的内容。在继续之前，我们将简要介绍特殊映射。

定义 2.16（单射、满射、双射）。考虑一个映射 $\Phi : V \rightarrow W$ ，其中 V, W 可以是任意集合。则称 Φ _

单射的
满射双射

- 单射若 $\forall x, y \in V : \Phi(x) = \Phi(y) \Rightarrow x = y$ 。
- 如果 $\Phi(V) = W$ ，则为满射。
- 如果它是单射和满射，则为双射。

2.7 线性映射

如果 Φ 是满射的,则W中的每个元素都可以使用 Φ 从V“到达”。双射 Φ 可以是“未完成的”,即存在映射 $\Psi : W \rightarrow V$ 使得 $\Psi \circ \Phi(x) = x$ 。这个映射 Ψ 然后被称为 Φ 的逆并且通常用 Φ^{-1} 表示

通过这些定义,我们引入了以下向量空间V和W之间线性映射的特殊情况:

- | | |
|---|---|
| <ul style="list-style-type: none"> ■ 同构: $\Phi : V \rightarrow W$ 线性和双射自同构: $\Phi : V \rightarrow V$ ■ 线性自同构: $\Phi : V \rightarrow V$ 线性和双射 ■ 我们在V中定义$\text{id}_V : V \rightarrow V$ 自同构。 ■ $x \rightarrow x$ 作为恒等映射或恒等恒等映射 | <small>同构</small>
<small>自同态</small>
<small>自同构</small>

<small>恒等自同构</small> |
|---|---|

例 2.19 (同态)

映射 $\Phi : \mathbb{R}^2 \rightarrow \mathbb{C}$, $\Phi(x) = x_1 + ix_2$,是一个同态:

$$\begin{aligned}
 \Phi & \begin{matrix} x_1 \\ x_2 \end{matrix} + \begin{matrix} y_1 \\ y_2 \end{matrix} = (x_1 + y_1) + i(x_2 + y_2) = x_1 + ix_2 + y_1 + iy_2 \\
 & = \Phi \begin{matrix} x_1 \\ x_2 \end{matrix} + \Phi \begin{matrix} y_1 \\ y_2 \end{matrix} \\
 \Phi \lambda & \begin{matrix} x_1 \\ x_2 \end{matrix} = \lambda x_1 + \lambda ix_2 = \lambda(x_1 + ix_2) = \lambda \Phi \begin{matrix} x_1 \\ x_2 \end{matrix}.
 \end{aligned} \tag{2.88}$$

这也证明了为什么复数可以在 \mathbb{R}^2 中表示为元组:存在一个双射线性映射,它将 \mathbb{R}^2 中元组的逐元素加法转换为具有相应加法的复数集。请注意,我们只显示了线性,但没有显示双射。

定理 2.17 (Axler (2015) 中的定理 3.59)。当且仅当 $\dim(V) = \dim(W)$ 时,有限维向量空间V和W是同构的。

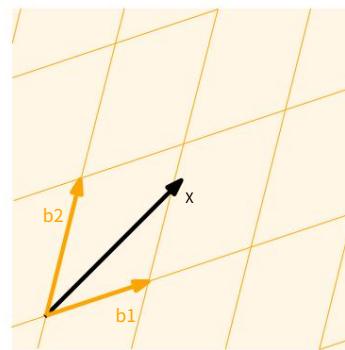
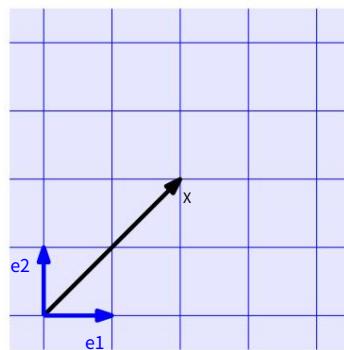
定理 2.17 指出在两个相同维度的向量空间之间存在线性双射映射。直观上,这意味着相同维度的向量空间是同一种东西,因为它们可以相互转换而不会产生任何损失。

定理 2.17 也为我们提供了将 $\mathbb{R}^{m \times n}$ ($m \times n$ 矩阵的向量空间)和 \mathbb{R}^{mn} (长度为 mn 的向量的向量空间)视为相同的理由,因为它们的维数是 mn ,并且存在线性,将一个转换为另一个的双射映射。

评论。考虑向量空间V、W、X。然后:

- 对于线性映射 $\Phi : V \rightarrow W$ 和 $\Psi : W \rightarrow X$,映射 $\Psi \circ \Phi : V \rightarrow X$ 也是线性的。
- 如果 $\Phi : V \rightarrow W$ 是同构,则 $\Phi^{-1} : W \rightarrow V$ 是同分异构体

图 2.1 两个不同的坐标
由两组基定义的系统
向量 x 有不同的
协调
表示取决于选择的
坐标系。



- 如果 $\Phi : V \rightarrow W$, $\Psi : V \rightarrow W$ 是线性的, 那么 $\Phi + \Psi$ 和 $\lambda\Phi$, $\lambda \in \mathbb{R}$, 也是线性的。

◇

2.7.1 线性映射的矩阵表示任何 n 维向量空间都同构于 \mathbb{R}^n (定理 2.17)。我们考虑一个基础 $\{b_1, \dots, b_n\}$ 的 n 维向量空间 V 。

在下文

中, 基向量的顺序将很重要。因此, 我们写

$$B = (b_1, \dots, b_n) \quad (2.89)$$

有序基础

并将此 n 元组称为 V 的有序基。

备注 (符号)。我们正处于符号变得有点棘手的地步。

因此, 我们在这里总结了一些部分。 $B = (b_1, \dots, b_n)$ 是有序基, $B = \{b_1, \dots, b_n\}$ 是一个 (无序的) 基, $B = [b_1, \dots, b_n]$ 是一个 \diamond 矩阵, 其列是向量 b_1, \dots, b_n 。

定义 2.18 (坐标)。考虑向量空间 V 和 V 的有序基 $B = (b_1, \dots, b_n)$ 。对于任何 $x \in V$, 我们获得唯一表示 (线性组合)

$$x = \alpha_1 b_1 + \dots + \alpha_n b_n \quad (2.90)$$

协调

x 相对于 B 。然后 $\alpha_1, \dots, \alpha_n$ 是 x 相对于 B 的坐标, 向量

$$\alpha = \begin{matrix} \alpha_1 \\ \vdots \\ \alpha_n \end{matrix} \in \mathbb{R}^n \quad (2.91)$$

坐标向量
协调
表示

是 x 相对于有序基 B 的坐标向量/坐标表示。

2.7 线性映射

一个基础有效地定义了一个坐标系。我们熟悉二维笛卡尔坐标系，它由规范基向量 e_1 、 e_2 跨越。在这个坐标系中，向量 $x \in R^2$ 有一个表示，告诉我们如何将 e_1 和 e_2 线性组合以获得 x 。然而， R^2 的任何基都定义了一个有效的坐标系，并且之前的相同向量 x 在 (b_1, b_2) 基中可能具有不同的坐标表示。在图 2.1 中， x 相对于标准基 (e_1, e_2) 的坐标为 $[2, 2]$ 。然而，对于基 (b_1, b_2) ，相同的向量 x 表示为 $[1.09, 0.72]$ ，即 $x = 1.09b_1 + 0.72b_2$ 。在接下来的部分中，我们将发现如何获得这种表示。

示例 2.20 让我

们看一下坐标为 $[2, 3]$ 图 2.2 关于 R^2 的标准基 (e_1, e_2) 的几何向量 $x \in R^2$ 。这意味着，我们可以

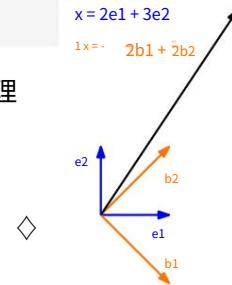
写成 $x = 2e_1 + 3e_2$ 。然而，我们不必选择标准基来表示这个向量。如果我们使用基向量 $b_1 = [1, -1]$ ， $b_2 = [1, 1]$ 来表示相同的向量，我们将获得关于 (b_1, b_2) （见图 2.2）。

向量 x 的不同坐标表示，取决于对

基础。

$$x = 2e_1 + 3e_2$$

$$x = 2b_1 + 2b_2$$



评论。对于 n 维向量空间 V 和 V 中 $i = 1, \dots, n$ 的有序基 $B = (b_1, \dots, b_n)$ 是线性的（并且由于定理 2.17 的推论），其中 (e_1, \dots, e_n) 是 R^n 的标准基。

现在我们准备好在矩阵和有限维向量空间之间的线性映射。

定义 2.19 (转换矩阵)。考虑具有对应 (有序) 基数 $B = (b_1, \dots, b_n)$ 和 $C = (c_1, \dots, c_m)$ 的向量空间 V, W 。

此外，我们考虑线性映射 $\Phi : V \rightarrow W$ 。对于 $j \in \{1, \dots, n\}$ ，

$$\Phi(b_j) = a_{1j}c_1 + \dots + a_{mj}c_m = \sum_{i=1}^{*} a_{ij}c_i \quad (2.92)$$

是 $\Phi(b_j)$ 关于 C 的唯一表示。然后，我们称 $m \times n$ -矩阵 $A\Phi$ ，其元素由下式给出

$$A\Phi(i, j) = a_{ij}, \quad (2.93)$$

Φ 的变换矩阵（相对于 V 变换的有序基 B 和 W 的 C ）。

矩阵

$\Phi(b_j)$ 相对于 W 的有序基 C 的坐标是 $A\Phi$ 的第 j 列。考虑具有有序基 B, C 和线性映射 $\Phi : V \rightarrow W$ 的 (有限维) 向量空间 V, W

变换矩阵 $A\Phi$ 。如果 x^\wedge 是 $x \in V$ 相对于 B 的坐标向量， y^\wedge 是 $y = \Phi(x) \in W$ 相对于 C 的坐标向量，则

$$y^\wedge = A\Phi x^\wedge \quad (2.94)$$

这意味着变换矩阵可用于将 V 中有序基的坐标映射到 W 中有序基的坐标。

示例 2.21 (转换矩阵)

考虑同态 $\Phi : V \rightarrow W$ 和 V 的有序基 $B = (b_1, \dots, b_3)$ 和 W 的 $C = (c_1, \dots, c_4)$ 。

$$\begin{aligned}\Phi(b_1) &= c_1 - c_2 + 3c_3 - c_4 \\ \Phi(b_2) &= 2c_1 + c_2 + 7c_3 + 2c_4 \\ \Phi(b_3) &= 3c_2 + c_3 + 4c_4\end{aligned} \quad (2.95)$$

关于 B 和 C 的变换矩阵 $A\Phi$ 满足 $\Phi(b_k) = \sum_{i=1}^4 \alpha_{ik} c_i$ for $k = 1, \dots, 3$ 并且给出为

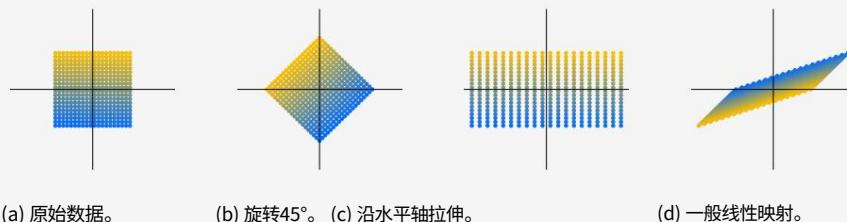
$$A\Phi = [\alpha_{11}, \alpha_{12}, \alpha_{13}] = \begin{matrix} 1 & 2 & 0 \\ -1 & 1 & 3 \\ 3 & 7 & 1 \\ -1 & 2 & 4 \end{matrix}, \quad (2.96)$$

其中 α_{ij} , $j = 1, 2, 3$ 是 $\Phi(b_j)$ 的坐标向量到 C 。

例 2.22 (向量的线性变换)

图 2.3 线性变换的三个例子

在 (a) 中显示为点的向量；(b) 旋转 45° ；(c) 将水平坐标拉伸 2；(d) 反射、旋转和拉伸的组合。



我们考虑使用变换矩阵对 R^2 中的一组向量进行三个线性变换

$$A1 = \begin{pmatrix} \cos(\pi) & -\sin(\pi) \\ \sin(\pi) & \cos(\pi) \end{pmatrix}, \quad A2 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad A3 = \begin{pmatrix} 3 & -1 \\ 1 & -1 \end{pmatrix}. \quad (2.97)$$

图 2.3 给出了一组向量的线性变换的三个例子。图 2.3(a)通过在相应(x_1, x_2)坐标处的一个点显示了 \mathbb{R}^2 中的400 个向量。这些向量排列在一个正方形中。当, 每一个都有代表我们使用 (2.97)中的矩阵 A_1 对这些向量中的每一个进行线性变换时, 我们得到图 2.3 (b)中的旋转正方形。如果我们应用由 A_2 表示的线性映射, 我们将获得图 2.3(c) 中的矩形, 其中每个 x_1 坐标都被拉伸2。图 2.3(d) 显示了使用 A_3 进行线性变换时来自图 2.3(a) 的原始正方形, 它是反射、旋转和拉伸的组合。

2.7.2 基变化

下面, 我们将仔细研究线性映射 $\Phi : V \rightarrow W$ 如果我们改变 V 和 W 中的基, 其变换矩阵是如何变化的。考虑两个有序基

$$B = (b_1, \dots, b_n), B' = (\sim b_1, \dots, \sim b_n) \quad (2.98)$$

V 和两个有序碱基

$$C = (c_1, \dots, c_m), C' = (c'_1, \dots, c'_m) \text{ 是 } W \text{ 的。此} \quad (2.99)$$

外, $A\Phi \in \mathbb{R}^{m \times n}$ 是线性映射 $\Phi : V \rightarrow W$ 相对于基 B 和 C , $A'\sim\Phi \in \mathbb{R}^{m \times n}$ 是相对于 B' 和 C' 的对应变换映射。

在下文中, 我们将研究 A 和 A' 之间的关系, 即如果我们选择执行基础 C' , 我们如何/是否可以将 $A\Phi$ 转换为 A' 。由 B, C 改为 B' 、备注。我们有效地获得了恒等映射 id_V 的不同坐标表示。

在图 2.2 的上下文中, 这意味着将关于 (e_1, e_2) 的坐标映射到关于 (b_1, b_2) 的坐标, 而不改变向量 x 。通过改变基础和相应的向量表示, 关于这个新基础的变换矩阵可以有一个特别简单的形式, 允许直接计算。 ◇

示例 2.23 (基差变化)
考虑一个变换矩阵

$$\begin{matrix} \text{一个} \\ \text{一个} \end{matrix} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \quad (2.100)$$

关于 \mathbb{R}^2 中的规范基础。

如果我们定义一个新的基础

$$B' = \left(\begin{array}{cc} 1 & 1 \\ 1 & -1 \end{array} \right) \quad (2.101)$$

我们得到一个对角变换矩阵

$$A' = \begin{matrix} 3 & 0 \\ 0 & 1 \end{matrix} \quad (2.102)$$

关于B,它比A更容易使用。

在下文中,我们将研究将关于一个基的坐标向量转换为关于不同基的坐标向量的映射。我们将首先陈述我们的主要结果,然后提供解释。

定理 2.20 (基差变化)。对于线性映射 $\Phi : V \rightarrow W$,有序碱基

$$B = (b_1, \dots, b_n), B' = (\sim b_1, \dots, \sim b_n) \quad (2.103)$$

V 和 $_W$

$$C = (c_1, \dots, c_m), C' = (c'_1, \dots, c'_m) \quad (2.104)$$

W 的变换矩阵 $A\Phi$ 的 Φ 相对于 B 和 C , 相应的变换矩阵 A 相对于基 B' 和 C' 给出为

Φ

$$A_\Phi = T - 1 A \Phi S. \quad (2.105)$$

其中, $S \in R^{n \times n}$ 是 id_V 将 B' 坐标映射到 B 坐标的变换矩阵, $T \in R^{m \times m}$ 是 id_W 将 C' 坐标映射到 C 坐标的变换矩阵。

证明按照 Drumm 和 Weil (2001), 我们可以将 V 的新基 B' 的向量写为 B 的基向量的线性组合,

这样

$$\sim b_j = s_{1j} b_1 + \dots + s_{nj} b_n = \sum_{i=1}^n s_{ij} b_i, j = 1, \dots, n. \quad (2.106)$$

类似地, 我们将 W 的新基向量 C' 写为 C 的基向量的线性组合, 从而产生

$$c'_k = t_{1k} c_1 + \dots + t_{mk} c_m = \sum_{l=1}^m t_{lk} c_l, k = 1, \dots, m. \quad (2.107)$$

我们将 $S = ((s_{ij})) \in R^{n \times n}$ 定义为将关于 B' 的坐标映射到关于 B 的坐标的变换矩阵, 定义 $T = ((t_{lk})) \in R^{m \times m}$ 作为映射坐标的变换矩阵关于 C' 到关于 C 的坐标上。特别地, S 的第 j 列是 $\sim b_j$ 相对于 B 的坐标表示, 并且

T的第k列是 c_k 相对于C的坐标表示。注意S和T都是正则的。

我们将从两个角度来看 $\Phi(\sim b_j)$ 。首先，应用映射 Φ ，我们得到所有 $j = 1, \dots, n$

$$\Phi(\sim b_j) = \sum_{k=1}^n a_{kj} c_k \stackrel{(2.107)}{=} \sum_{k=1}^n a_{kj} m_l k l c_l = \sum_{m_l=1}^m \sum_{k=1}^n a_{kj} m_l k l c_l, \quad (2.108)$$

其中我们首先将新的基向量 $c_k \in W$ 表示为基向量 $c_l \in W$ 的线性组合，然后交换求和的顺序。

或者，当我们把 $\sim b_j \in V$ 表示为线性组合时
 $b_j \in V$ ，我们到达

$$\Phi(\sim b_j) \stackrel{(2.106)}{=} \sum_{i=1}^n s_{ij} b_i = \sum_{i=1}^n s_{ij} \Phi(b_i) = \sum_{i=1}^n \sum_{l=1}^n s_{ij} m_l k l c_l, \quad \text{阿里克} \quad (2.109a)$$

$$= \sum_{m_l=1}^m \sum_{i=1}^n s_{ij} m_l k l c_l, \quad j = 1, \dots, n, \quad (2.109b)$$

我们利用了 Φ 的线性度。比较(2.108)和(2.109b)，它遵循所有 $j = 1, \dots, n$ 和 $m_l = 1, \dots, m$ 那

$$\sum_{k=1}^n t_l k a_{kj} = \sum_{i=1}^n s_{ij} m_l k l, \quad \text{阿里西} \quad (2.110)$$

因此，

$$TA \sim_\phi = A\Phi S \in R^{* \times n}, \quad (2.111)$$

这样

$$A_\phi = T - 1A\Phi S, \quad (2.112)$$

这证明了定理 2.20。 \square

定理 2.20 告诉我们，随着 V （ B 替换为 $B\sim$ ）和 W （ C 替换为 $C\sim$ ）的基础变化，线性映射 $\Phi : V \rightarrow W$ 的变换矩阵 $A\Phi$ 被等效矩阵 A 替换~ Φ 和

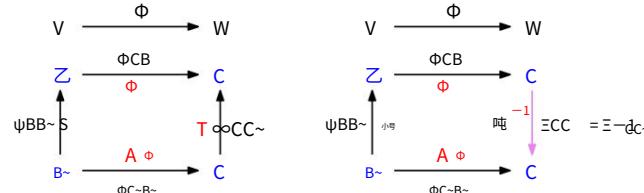
$$A_\phi = T - 1A\Phi S. \quad (2.113)$$

图 2.2 说明了这种关系：考虑同态 $\Phi : V \rightarrow W$ 和有序基 B ， V 和 C 的 $B\sim$ ， W 的 $C\sim$ 。映射 Φ 是 Φ 的一个实例，并将 B 的基向量映射到 $C\sim$ 的基向量。假设我们知道 Φ 相对于有序基 B 、 C 的变换矩阵 $A\Phi$ 。当我们在 V 中执行从 B 到 $B\sim$ 以及在 W 中从 C 到 $C\sim$ 的基更改时，我们可以确定这

图 2.2 对于同态 $\Phi : V \rightarrow W$ 和

有序的基 B, V 和 C 的 $B \sim$,
 W 的 $C \sim$ (蓝色标记), 我们可以将映射 $\Phi C \sim B \sim$ 相对于基 $C \sim$ 等价地表示为 $B \sim$ 同态 $\Phi C \sim B$ 的组合 $= \Xi C \sim \circ \Phi C \circ \Psi B B \sim$ 关于下标中的碱基。相应的变换

向量空间



相应的变换矩阵 $A \sim \Phi$ 如下: 首先, 我们找到线性映射 $\Psi B B \sim : V \rightarrow V$ 的矩阵表示, 它将关于新基 $B \sim$ 的坐标映射到关于“旧”基础 B (在 V 中)。然后, 我们使用 $\Phi C B$ 的变换矩阵 $A \Phi : V \rightarrow W$ 将这些坐标映射到 W 中相对于 C 的坐标。最后, 我们使用线性映射 $\Xi C C \sim : W \rightarrow W$ 将坐标映射到相对于 C 的坐标到关于 $C \sim$ 的坐标上。因此, 我们可以将线性映射 $\Phi C \sim B \sim$ 表示为包含“旧”基的线性映射的组合:

矩阵是红色的。

$$\Phi C \sim B \sim = \Xi C C \sim \circ \Phi C B \circ \Psi B B \sim = \Xi C C \sim \circ \Phi C B \circ \Psi B B \sim \quad (2.114)$$

具体来说, 我们使用 $\Psi B B \sim = id_V$ 和 $\Xi C C \sim = id_W$, 即, 将向量映射到自身的恒等映射, 但相对于不同的基础。

相等的

定义 2.21 (等价)。如果存在正则矩阵 $S \in Rn \times n$ 和 $T \in Rm \times m$, 使得 $A \sim = T - 1AS$, 则两个矩阵 $A, A \sim \in Rm \times n$ 是等价的。

相似的

定义 2.22 (相似性)。两个矩阵 $A, A \sim \in Rn \times n$ 存在正则矩阵 $S \in Rn \times n$ 且 $A \sim = S - 1AS$

评论。相似矩阵总是等价的。然而, 等价矩阵不一定相似。 ◇ 备注。考虑向量空间 V, W, X , 根据定理 2.17 后面的注释, 我们已经知道对于线性映射 $\Phi : V \rightarrow W$ 和 $\Psi : W \rightarrow X$, 映射 $\Psi \circ \Phi : V \rightarrow X$ 也是线性的。有了相应映射的变换矩阵 $A\Phi$ 和 $A\Psi$, 总的变换矩阵为 $A\Psi \circ \Phi = A\Psi A\Phi$ 。 ◇ 鉴于这句话, 我们可以从角度来看基差变化

组成线性映射的想法:

- $A\Phi$ 是线性映射 $\Phi C B : V \rightarrow W$ 相对于基 B, C 的变换矩阵。
- $A \circ$ 是线性映射 $\Phi C \sim B \sim : V \rightarrow W \sim C \sim$ 的变换矩阵。关于基 B , S 是线性映射 $\Psi B B \sim$ 的变换矩阵: $V \rightarrow V$ (自同构), 表示 $B \sim$ 在 B 方面。通常, $\Psi = id_V$ 是 V 中的恒等映射。

- T 是线性映射 $\exists C \subset W \rightarrow W$ (自同构)的变换矩阵,表示 C 关于 C_0 。通常, $\exists = id_W$ 是 W 中的恒等映射。

如果我们 (非正式地)仅根据基写下变换,则 $A\Phi : B \rightarrow C$, $A \sim \Phi : B \sim \rightarrow C \sim$, $S : B \sim \rightarrow B$, $T : C \sim \rightarrow C$ 和

$\Phi^{-1} : C \rightarrow C \sim$, 和

$$B \sim \rightarrow C \sim = B \sim \rightarrow B \rightarrow C \rightarrow C \sim = T \quad (2.115)$$

$$A \sim \Phi^{-1} A \Phi S. \quad (2.116)$$

请注意,(2.116) 中的执行顺序是从右到左,因为向量在右侧相乘,因此 $x \rightarrow Sx \rightarrow A\Phi(Sx) \rightarrow T$

$$\Phi^{-1} A\Phi(Sx) = A \sim \Phi x.$$

示例 2.24 (基差变化)

考虑一个线性映射 $\Phi : R^3 \rightarrow R^4$ 其变换矩阵是

$$A\Phi = \begin{pmatrix} 1 & 2 & 0 \\ -1 & 1 & 3 \\ 3 & 7 & 1 \\ -1 & 2 & 4 \end{pmatrix} \quad (2.117)$$

关于标准碱

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (2.118)$$

我们求变换矩阵 $A \sim$

Φ 相对于新基地

$$B \sim = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad C \sim = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (2.119)$$

然后,

$$S = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \Phi = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.120)$$

其中 S 的第 i 列是根据 B 的基向量表示的 $\sim b_i$ 的坐标表示。由于 B 是标准基,因此坐标表示很容易找到。对于一般基础 B ,我们需要求解线性方程组以找到 λ_i ,使得

$\lambda_i b_i = \sim b_j, j = 1, \dots, 3$. 类似地, T 的第 j 列是 c_j 以 C 的基向量表示的坐标表示。

因此, 我们得到

$$A_\phi = T - 1A\Phi S = \begin{array}{r} & \begin{array}{c} 1 & -1 & -1 \\ - & \begin{array}{c} 1 & 1 & -1 & 1 & -1 \\ 2 & -1 & 1 & & 1 \\ & 1 & 0 & 0 & 2 \end{array} \\ & \begin{array}{c} 3 & 2 & 1 \\ 0 & 4 & 2 \\ 10 & 8 & 4 \\ 1 & 6 & 3 \end{array} \end{array} \quad (2.121a)$$

$$= \begin{array}{r} & \begin{array}{c} -4 & -4 & -2 & 6 & 0 \\ & 4 & 8 & 6 & 0 \\ & & & 4 & . \end{array} \\ & \begin{array}{c} 1 & \\ 3 & \end{array} \end{array} \quad (2.121b)$$

在第 4 章中, 我们将能够利用基变化的概念来找到一个基, 关于该基, 自同态的变换矩阵具有特别简单的 (对角线) 形式。在第 10 章中, 我们将研究数据压缩问题, 并找到一个方便的基础, 我们可以在该基础上投影数据, 同时最小化压缩损失。

2.7.3 镜像和内核

线性映射的图像和内核是具有某些重要属性的向量子空间。在下文中, 我们将更仔细地描述它们。

核心
零空间

对于 $\Phi : V \rightarrow W$, 我们定义内核/零空间

$$\ker(\Phi) := \Phi^{-1}(0W) = \{v \in V : \Phi(v) = 0W\} \quad (2.122)$$

图像
范围

$$\text{Im}(\Phi) := \Phi(V) = \{w \in W \mid \exists v \in V : \Phi(v) = w\}. \quad (2.123)$$

我们也分别称 V 和 W 为 Φ 的域和辅域。

直观上, 内核是向量集 $v \in V$, Φ 映射到中性元素 $0W \in W$ 。像是向量集 $w \in W$, Φ 可以从 V 中的任何向量 “到达”。

图 2.2 给出了一个说明。

评论。考虑一个线性映射 $\Phi : V \rightarrow W$, 其中 V, W 是向量空间。

- 它始终认为 $\Phi(0V) = 0W$, 因此 $0V \in \ker(\Phi)$ 。特别地, 零空间永远不会是空的。
- $\text{Im}(\Phi) \subseteq W$ 是 W 的子空间, $\ker(\Phi) \subseteq V$ 是 V 的子空间。

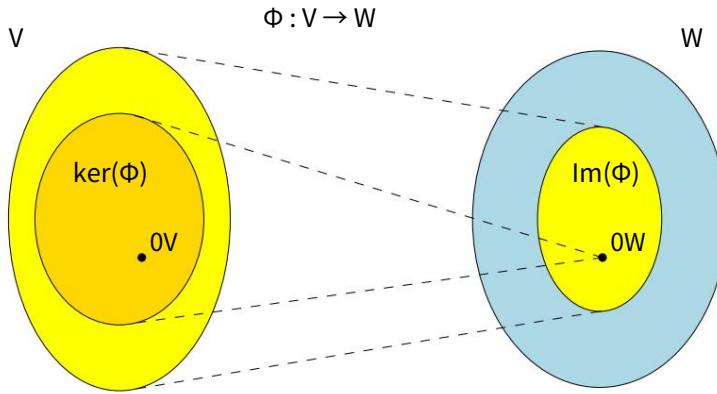


图 2.2 线性映射的内核和图像

$$\phi: V \rightarrow W_0$$

- 当且仅当 $\ker(\Phi) = \{0\}$ 时， Φ 是单射的（一对一）。

◇

备注（零空间和列空间）。让我们考虑 $A \in \mathbb{R}^{m \times n}$ 和线性映射 $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $x \mapsto Ax$ 。

- 对于 $A = [a_1, \dots, a_n]$, 其中 a_i 是 A 的列, 我们得到

$$\text{Im}(\phi) = \{Ax : x \in R^n\} = \underbrace{x_1, \dots, x_n}_{\substack{\text{我=1} \\ \vdots}} \in R^n \quad (2.124a)$$

即图像是A的列的跨度,列空间也叫列空间。因此,列空间(图像)是 \mathbb{R}^m 的子空间,其中m是矩阵的“高度”。 $\text{rk}(A) = \dim(\text{Im}(\Phi))$ 。

- 内核/零空间 $\ker(\Phi)$ 是齐次线性方程组 $Ax = 0$ 的通解，并捕获 R^n 中产生 $0 \in R^m$ 的元素的所有可能线性组合。
 - 内核是 R^n 的一个子空间内核关注列，其中n是矩阵的“宽度”。
 - 之间的关系，我们可以用它来确定我们是否/如何将一列表示为其他列的线性组合。

◆

示例 2.25 (线性映射的图像和内核)
映射

$$4 \Phi: R \xrightarrow{\text{2} \rightarrow \text{右}} \begin{array}{r} x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \rightarrow \begin{array}{r} x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} = \begin{array}{l} x_1 + 2x_2 - x_3 x_1 \\ + x_4 \end{array} \quad (2.125a)$$

$$= x_1 \begin{smallmatrix} 1 \\ 1 \\ 1 \end{smallmatrix} + x_2 \begin{smallmatrix} 2 \\ 0 \\ 0 \end{smallmatrix} + x_3 \begin{smallmatrix} -1 \\ 0 \\ 0 \end{smallmatrix} + x_4 \begin{smallmatrix} 0 \\ 1 \\ 1 \end{smallmatrix} \quad (2.125b)$$

是线性的。为了确定 $\text{Im}(\Phi)$, 我们可以取变换矩阵列的跨度并获得

$$\text{Im}(\Phi) = \text{跨度} \left[\begin{smallmatrix} 1 \\ 1 \\ 1 \end{smallmatrix}, \begin{smallmatrix} 2 \\ 0 \\ 0 \end{smallmatrix}, \begin{smallmatrix} -1 \\ 0 \\ 0 \end{smallmatrix}, \begin{smallmatrix} 0 \\ 1 \\ 1 \end{smallmatrix} \right]. \quad (2.126)$$

要计算 Φ 的核 (零空间), 我们需要求解 $Ax = 0$, 即我们需要求解齐次方程组。为此, 我们使用高斯消去法将 A 转换为简化的行阶梯形式:

$$\begin{array}{ccccccc} 1 & 2 & -1 & 0 & \cdots & 1 & 0 \\ & 1 & 0 & 1 & -2 & & \\ & & & & & - & - \\ & & & & & & \overline{1} \\ & & & & & & 2 \end{array}. \quad (2.127)$$

这个矩阵是简化的行阶梯形式, 我们可以使用 Minus 1 Trick 来计算内核的基础 (参见第 2.3.3 节)。或者, 我们可以将非枢轴列 (第 3 列和第 4 列) 表示为枢轴列 (第 1 列和第 2 列) 的线性组合。第三列 a_3 等价于 - 以同样的方式, 我们看到 $a_4 = a_1$ 。总的来说, 这给了我们内核 (零空间) 作为

$$\begin{array}{c} \begin{array}{l} \text{乘以第二列 } a_2, \text{ 因此, } 0 = a_3 + \\ \text{--- } a_2, \text{ 因此 } 0 = a_1 - \end{array} & \begin{array}{l} \text{--- } a_2, \text{ 在} \\ \text{T } a_2 - a_4, \text{ 2 个} \end{array} \end{array}$$

$$\ker(\Phi) = \text{跨度} \left[\begin{smallmatrix} 0 & -1 \\ \frac{1}{2} & 0 \\ 1 & 1 \end{smallmatrix} \right]. \quad (2.128)$$

秩零定理

定理 2.24 (秩无效定理)。对于向量空间 V, W 和线性映射 $\Phi : V \rightarrow W$ 它认为

$$\dim(\ker(\Phi)) + \dim(\text{Im}(\Phi)) = \dim(V). \quad (2.129)$$

线性基本定理

秩零定理也称为线性映射的基本定理 (Axler, 2015, 定理 3.22)。以下是定理 2.24 的直接结果:

映射

- 如果 $\dim(\text{Im}(\Phi)) < \dim(V)$, 则 $\ker(\Phi)$ 是非平凡的, 即内核包含超过 0 且 $\dim(\ker(\Phi)) \geq 1$ 。
- 如果 $A\Phi$ 是 Φ 相对于有序基的变换矩阵并且 $\dim(\text{Im}(\Phi)) < \dim(V)$, 则线性方程组 $A\Phi x = 0$ 有无穷多个解。
- 如果 $\dim(V) = \dim(W)$, 则以下三向等价成立: - Φ 是单射的 - Φ 是满射的 - Φ 是双射的, 因为 $\text{Im}(\Phi) \subseteq W$ 。

2.8 仿射空间在下文

中,我们将仔细研究偏离原点的空间,即不再是向量空间的空间。此外,我们将简要讨论这些仿射空间之间映射的性质,类似于线性映射。

评论。在机器学习文献中,线性和仿射之间的区别有时并不明确,因此我们可以找到将仿射空间/映射称为线性空间/映射的引用。 ◇

2.8.1 仿射子空间

定义 2.25 (仿射子空间)。令 V 为向量空间, $x_0 \in V$ 且 $U \subseteq V$ 为子空间。然后是子集

$$L = x_0 + U := \{x_0 + u : u \in U\} \quad (2.130a)$$

$$= \{v \in V \mid \exists u \in U : v = x_0 + u\} \subseteq V \quad (2.130b)$$

称为 V 的仿射子空间或线性流形。 U 称为方向或仿射子空间方向空间, x_0 称为支撑点。在第 12 章中,我们将这样的子空间称为超平面。

线性流形
方向
方向空间支撑点超
平面

请注意,如果 $x_0 \in / U$, 则仿射子空间的定义不包括 0。
因此,对于 $x_0 \in / U$, 仿射子空间不是 V 的(线性)子空间(向量空间)。

仿射子空间的示例是 R^3 中的点、线和平面, 哪个
不要(必须)通过原点。

评论。考虑向量空间 V 的两个仿射子空间 $L = x_0 + U$ 和 $L' = x_1 + U'$ 。那么, $L \subseteq L'$
当且仅当 $U \subseteq U'$ 和 $x_0 - x_1 \in U$ 。

仿射子空间通常由参数描述: 考虑一个 k 维 If (b_1, \dots, b_k) 是一个有序的仿射空间 $L = x_0 + U$
of V 的基础。
 U , 那么每个元素 $x \in L$ 可以被唯一描述为

$$x = x_0 + \lambda_1 b_1 + \dots + \lambda_k b_k, \quad (2.131)$$

其中 $\lambda_1, \dots, \lambda_k \in R$ 。这种表示称为参数方程参数方程 λ 。 ◇ L 的参数, 方向向量 b_1, \dots, b_k 和参数 $\lambda_1, \dots,$

例 2.26 (仿射子空间)

- 一维仿射子空间称为线,可以写成线 $y = x_0 + \lambda b_1$, 其中 $\lambda \in R$ 且 $U = \text{span}[b_1] \subseteq R^n$ 是 R^n 的一维子空间。这意味着一条线由支撑点 x_0 和定义方向的矢量 b_1 定义。参见图 2.2 的说明。

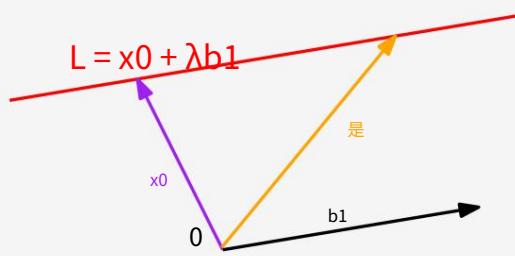
飞机

- Rⁿ的二维仿射子空间称为平面。平面的参数方程是 $y = x_0 + \lambda_1 b_1 + \lambda_2 b_2$, 其中 $\lambda_1, \lambda_2 \in \mathbb{R}$ 且 $U = \text{span}[b_1, b_2] \subseteq R^n$ 。这意味着一个平面由一个支撑点 x_0 和两个跨越方向空间的线性独立向量 b_1, b_2 定义。

超平面

- 在 R^n , $(n - 1)$ 维仿射子空间称为超平面, $\lambda_i b_i$ 相应的参数方程是 $y = x_0 + \sum_{i=1}^{n-1} \lambda_i b_i$ 构成一个 $(n - 1)$ 维子空间的基础, 其中 b_1, \dots, b_{n-1} 跨越方向空间 U 的 R^n 。这意味着超平面由支持点 x_0 和 $(n - 1)$ 个线性独立向量 b_1, \dots, b_{n-1} 定义。在 R^2 中, 一条线也是一个超平面。在 R^3 中, 一个平面也是一个超平面。

图 2.2 线是仿射子空间。
直线 $x_0 + \lambda b_1$ 上的向量 y 位于具有支撑点 x_0 和
方向 b_1 的仿射子空
间 L 中。



备注 (线性方程和仿射子空间的非齐次系统)。

对于 $A \in R^{m \times n}$ 和 $x \in R^m$, 线性方程组 $A\lambda = x$ 的解要么是空集, 要么是 R^n 的维数 $n - rk(A)$ 的仿射子空间。特别是线性方程 $\lambda_1 b_1 + \dots + \lambda_n b_n = x$, 其中 $(\lambda_1, \dots, \lambda_n) = (0, \dots, 0)$, 是 R^n 中的一个超平面。

在 R^n , 每个 k 维仿射子空间是一个 inho 的解
齐次线性方程组 $Ax = b$, 其中 $A \in R^{m \times n}$ 且 $rk(A) = n - k$ 。回想一下齐次方程组 $Ax = 0$ 的解, 是一个向量子空间, 我们也可以将其视为具有支撑点 $x_0 = 0$ 的特殊仿射空间。◇

2.8.2 仿射映射

类似于我们在 2.7 节中讨论的向量空间之间的线性映射, 我们可以定义两个仿射空间之间的仿射映射。

线性映射和仿射映射密切相关。因此, 我们从线性映射中已知的许多性质, 例如, 线性映射的组合是线性映射, 也适用于仿射映射。

定义 2.26 (仿射映射)。对于两个向量空间 V, W , 线性

映射 $\Phi : V \rightarrow W$, 并且 $a \in W$, 映射

$$: V \rightarrow W \quad (2.132)$$

$$x \rightarrow a + \Phi(x) \quad (2.133)$$

是从 V 到 W 的仿射映射。向量 a 称为 Φ 的平移仿射映射向量。

[翻译向量](#)

- 每个仿射映射 $: V \rightarrow W$ 也是线性映射 $\Phi : V \rightarrow W$ 和平移 $\tau : W \rightarrow W$ 在 W 中的组合,使得
 $= \tau \circ \Phi$ 。映射 Φ 和 τ 是唯一确定的。
- 组合 $: W \rightarrow X$ 是仿射的。仿射映射的 $: V \rightarrow W$,
- 仿射映射保持几何结构不变。它们还保留维度和平行度。

2.9 延伸阅读

学习线性代数的资源很多,包括 Strang (2003)、Golan (2007)、Axler (2015) 以及 Liesen 和 Mehrmann (2015) 的教科书。我们在本章的介绍中也提到了一些在线资源。我们在这里只介绍了高斯消元法,但还有许多其他求解线性方程组的方法,我们参考了 Stoer 和 Burlirsch (2002 年)、Golub 和 Van Loan (2012 年) 以及 Horn 和 Johnson 的数值线性代数教科书 (2013) 进行深入讨论。

在本书中,我们区分了线性代数的主题 (例如,向量、矩阵、线性独立性、基) 和与向量空间的几何相关的主题。在第 3 章中,我们将介绍导出范数的内积。这些概念允许我们定义角度、长度和距离,我们将用于正交投影。投影被证明是许多机器学习算法的关键,例如线性回归和主成分分析,我们将分别在第 9 章和第 10 章中介绍这两种算法。

练习

2.1 我们考虑($\mathbb{R} \setminus \{-1\}$, \oplus),其中

$$a \oplus b := ab + a + b, \quad a, b \in \mathbb{R} \setminus \{-1\} \quad (2.134)$$

A.证明($\mathbb{R} \setminus \{-1\}$, \oplus)是阿贝尔群。 b.解决

$$3 \oplus x = x = 15$$

在阿贝尔群($\mathbb{R} \setminus \{-1\}$, \oplus)中,其中在(2.134)中定义。

2.2 设n在 $\mathbb{N} \setminus \{0\}$ 中。令 k, x 在 \mathbb{Z} 中。我们定义整数k的同余类 \bar{k} 为集合

$$\begin{aligned} \bar{k} &= \{x \in \mathbb{Z} \mid x - k = 0 \pmod{n}\} \\ &= \{x \in \mathbb{Z} \mid \exists a \in \mathbb{Z}: (x - k = n \cdot a)\}. \end{aligned}$$

我们现在将 $\mathbb{Z}/n\mathbb{Z}$ (有时写作 \mathbb{Z}_n)定义为所有同余类对 n 取模的集合。欧氏除法意味着这个集合是一个包含n个元素的有限集合:

$$\mathbb{Z}_n = \{0, 1, \dots, n-1\}$$

对于所有 $\bar{a}, \bar{b} \in \mathbb{Z}_n$,我们定义

$$\bar{a} \oplus \bar{b} := \overline{a + b}$$

A.证明(\mathbb{Z}_n, \oplus)是一个群。是阿贝尔吗? b.我们现在为 \mathbb{Z}_n 中的所有a和b定义另一个操作 \otimes 为

$$\bar{a} \otimes \bar{b} = \overline{a \times b}, \quad (2.135)$$

其中 $a \times b$ 表示 \mathbb{Z} 中的通常乘法。

令 $n = 5$ 。画出 $\mathbb{Z}_5 \setminus \{0\}$ 中元素在 \otimes 下的乘法表,即对 $\mathbb{Z}_5 \setminus \{0\}$ 中的所有 \bar{a} 和 \bar{b} 计算乘积 $\bar{a} \otimes \bar{b}$ 。

因此,证明 $\mathbb{Z}_5 \setminus \{0\}$ 在 \otimes 下是封闭的并且拥有 \otimes 的中性元素。在 \otimes 下显示 $\mathbb{Z}_5 \setminus \{0\}$ 中所有元素的逆。

得出($\mathbb{Z}_5 \setminus \{0\}$, \otimes)是阿贝尔群。

C.证明($\mathbb{Z}_8 \setminus \{0\}$, \otimes)不是群。 d.我们记得 B'ezout

定理指出两个整数a和b互质 (即 $\gcd(a, b) = 1$)当且仅当存在两个整数u和v使得 $au + bv = 1$ 。

显示($\mathbb{Z}_n \setminus \{0\}$, \otimes)是一个群当且仅当 $n \in \mathbb{N} \setminus \{0\}$ 是素数。

2.3 考虑定义如下的 3×3 矩阵的集合G :

$$G = \begin{matrix} & & 1 \\ & 0 & 1 \\ 0 & 0 & 1 \end{matrix} \in \mathbb{R}^{3 \times 3} \quad x, y, z \in \mathbb{R}$$

我们将·定义为标准矩阵乘法。

(G, ·)是群吗?如果是,是阿贝尔吗?证明你的答案。

2.4 如果可能,计算以下矩阵乘积:

A.

$$\begin{array}{r} 12 \\ 45 \\ 78 \end{array} \quad \begin{array}{r} 110 \\ 011 \\ 101 \end{array}$$

b.

$$\begin{array}{r} 123 \\ 456 \\ 789 \end{array} \quad \begin{array}{r} 110 \\ 011 \\ 101 \end{array}$$

C.

$$\begin{array}{r} 110 \\ 011 \\ 101 \end{array} \quad \begin{array}{r} 123 \\ 456 \\ 789 \end{array}$$

d.

$$\begin{array}{r} 03 \\ 121 \\ 41-1-4 \end{array} \quad \begin{array}{r} 1-1 \\ 21 \\ 52 \end{array}$$

e.

$$\begin{array}{r} 03 \\ 1-1 \\ 21 \\ 52 \end{array} \quad \begin{array}{r} 12 \\ 41-1-4 \end{array}$$

2.5 求下列非齐次线性方程组 $Ax = b$ 在 x 中所有解的集合 S , 其中 A 和 b 定义如下:

A.

$$\text{一个} = \begin{array}{r} 1-1-1 \\ 25-7-5 \\ 2-11 \\ 52-42 \end{array}, \quad b = \begin{array}{r} -2 \\ 4 \\ 6 \end{array}$$

b.

$$\text{一个} = \begin{array}{r} 1-100 \\ 10-30 \\ 2-101-1 \\ -120-2-1 \end{array}, \quad b = \begin{array}{r} 3 \\ 6 \\ 5 \\ -1 \end{array}$$

2.6 利用高斯消去法求出非齐次方程的所有解

化系统 $Ax = b$ 与

$$\text{一个} = \begin{array}{r} 010010 \\ 000110 \\ 010001 \end{array}, \quad b = \begin{array}{r} -1 \\ 2 \\ 0 \end{array}$$

2.7 求 $x =$ 中的所有解
 $\in R^3$ 方程组 $Ax = 12x$,
 x_1
 x_2
 x_3

在哪里

$$\begin{array}{r} 643 \\ -\text{一个}= \quad 609 \\ \quad 080 \end{array}$$

和 $\sum_i x_i = 1$

2.8 如果可能,确定下列矩阵的逆矩阵:

a.

$$\begin{array}{r} 234 \\ -\text{一个}= \quad 345 \\ \quad 456 \end{array}$$

b.

$$\begin{array}{r} 1010 \\ -\text{一个}= \quad 0110 \\ \quad 1101 \\ \quad 1110 \end{array}$$

2.9 以下哪些集合是 R^3 的子空间? $\lambda - \mu, 0) | \lambda \in R\}$

a. $A = \{(\lambda, \lambda + \mu, \lambda^2) | \lambda, \mu \in R\}$

b. $B = \{(\lambda, \lambda^2, 2-\lambda) | \lambda \in R\}$

设 y 在 R 中。

$C = \{(\xi_1, \xi_2, \xi_3) \in R^3 | \xi_1 - 2\xi_2 + 3\xi_3 = y\}$ d. $D = \{(\xi_1, \xi_2, \xi_3) \in R^3 | \xi_2 \in Z\}$

2.10 以下向量组线性无关吗?

a.

$$\begin{array}{rrr} x_1 = & -1 & x_2 = & 1 \\ & 2 & & 2 \\ & , & & , \end{array} \quad \begin{array}{rrr} x_3 = & -3 & \\ & 3 & \\ & , & \end{array}$$

b.

$$\begin{array}{rrr} x_1 = & \begin{matrix} 1 \\ 2 \\ 0 \\ 0 \end{matrix} & x_2 = & \begin{matrix} 1 \\ 2 \\ 1 \\ 1 \end{matrix} & x_3 = & \begin{matrix} 1 \\ 0 \\ 1 \\ 1 \end{matrix} \\ & , & & , & & , \end{array}$$

2.11 写入

$$y = \begin{matrix} 1 \\ 2 \\ 0 \\ 0 \end{matrix}$$

作为线性组合

$$\begin{array}{rrr} x_1 = & \begin{matrix} 1 \\ 2 \\ 0 \\ 0 \end{matrix} & x_2 = & \begin{matrix} 1 \\ 2 \\ 1 \\ 1 \end{matrix} & x_3 = & \begin{matrix} 1 \\ 0 \\ 1 \\ 1 \end{matrix} \\ & , & & , & & , \end{array}$$

2.12 考虑 \mathbb{R}^4 的两个子空间：

$$U_1 = \text{跨度} \left[\begin{array}{cccc} 1 & 2 & -1 & -1 \\ 1 & -1 & 1 & -2 \\ -3 & 0 & -1 & 0 \\ 1 & -1 & 1 & 0 \end{array} \right], \quad U_2 = \text{跨度} \left[\begin{array}{cccc} 2 & -1 & -3 \\ 2 & -2 & 0 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{array} \right].$$

确定 $U_1 \cap U_2$ 的一个基。2.13 考虑两个子空间 U_1 和 U_2 ,其中 U_1 是齐次方程组 $A_1x = 0$ 的解空间, U_2 是齐次方程组 $A_2x = 0$ 的解空间,其中

$$A_1 = \begin{matrix} 1 & 0 & 3-3 & 0 \\ 1 & -2 & -1 & \\ 2 & 1 & 7-5 & 2 \\ 1 & 0 & 3-1 & 2 \end{matrix}, \quad A_2 = \begin{matrix} 1 & 2 & 3 \\ 2 & 1 & 7-5 \\ 1 & 0 & 2 \end{matrix}.$$

- A。确定 U_1 、 U_2 的尺寸。
B。确定 U_1 和 U_2 的基数。
C。确定 $U_1 \cap U_2$ 的一个基。

2.14 考虑两个子空间 U_1 和 U_2 ,其中 U_1 由 A_1 和 U_2 由 A_2 的列跨越

$$A_1 = \begin{matrix} 1 & 0 & 3-3 & 0 \\ 1 & -2 & -1 & \\ 2 & 1 & 7-5 & 2 \\ 1 & 0 & 3-1 & 2 \end{matrix}, \quad A_2 = \begin{matrix} 1 & 2 & 3 \\ 2 & 1 & 7-5 \\ 1 & 0 & 2 \end{matrix}.$$

- A。确定 U_1 、 U_2 的尺寸
B。确定 U_1 和 U_2 的基数
C。确定 $U_1 \cap U_2$ 的基

2.15 令 $F = \{(x, y, z) \in \mathbb{R}^3 \mid x+y-z=0\}$ 和 $G = \{(a-b, a+b, a-3b) \mid a, b \in \mathbb{R}\}$ 。 A。证

- 明 F 和 G 是 \mathbb{R}^3 的子空间。
B. 在不求助于任何基向量的情况下计算 $F \cap G$ 。
C. 为 F 找到一个基,为 G 找到一个基,使用之前找到的基向量计算 $F \cap G$,并用上一题检查你的结果。

2.16 以下映射是线性的吗?

- A。令 $a, b \in \mathbb{R}$ 。

$$\Phi: \text{大号 } ([a, b]) \rightarrow \mathbb{R}$$

$$f \rightarrow \Phi(f) = \int_a^b f(x) dx,$$

其中大号 $\int_{[a, b]}$ 表示 $[a, b]$ 上的可积函数集。
b.

$$C \rightarrow C^{1,0} \Phi:$$

$$f \rightarrow \Phi(f) = f,$$

其中对于 $k \geq 1$, $C^{k,0}$ 表示 k 次连续不同的集合 0 表示连续函数的集合。
tiable 函数和 C

C_0

$$\Phi : \mathbb{R} \rightarrow \mathbb{R}$$

$$x \mapsto \Phi(x) = \text{余弦}(x)$$

d.

$$\mathbb{R} \rightarrow \mathbb{R}^3 \Phi :$$

$$\begin{array}{ccc} & 1 & 2 & 3 \\ \rightarrow & - & & \\ & 1 & 4 & 3 \end{array} \quad x$$

e. 设 θ 在 $[0, 2\pi]$ [和

$$\Phi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$$

$$\begin{array}{ccc} & \cos(\theta) \sin(\theta) - \sin(\theta) & x \\ \rightarrow & - & \\ & \cos(\theta) & \end{array}$$

2.17 考虑线性映射

$$\mathbb{R} \rightarrow \mathbb{R}^3 \Phi :$$

$$\begin{array}{ccc} & 3x_1 + 2x_2 + x_3 & \\ \Phi & \begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array} & = \begin{array}{c} x_1 + x_2 + x_3 \\ x_1 - 3x_2 \\ 2x_1 + 3x_2 + x_3 \end{array} \end{array}$$

- 找到变换矩阵 $A\Phi$ 。
- 确定 $\text{rk}(A\Phi)$ 。
- 计算 Φ 的核和像。什么是 $\dim(\ker(\Phi))$ 和 $\dim(\text{Im}(\Phi))$?

2.18 令 E 为向量空间。设 f 和 g 是 E 上的两个自同构,使得 $f \circ g = \text{id}_E$ (即, $f \circ g$ 是恒等映射 id_E)。证明 $\ker(f) = \ker(g \circ f)$, $\text{Im}(g) = \text{Im}(g \circ f)$ 并且 $\ker(f) \cap \text{Im}(g) = \{0_E\}$ 。

2.19 考虑一个自同态 $\Phi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ 其变换矩阵 (相对于 \mathbb{R}^3 中的标准基础)是

$$A\Phi = \begin{pmatrix} & & 1 & 0 \\ & & 1 & -1 & 0 \\ & 2 & & 2 & & 2 \\ & & & & & \ddots \end{pmatrix}$$

A. 确定 $\ker(\Phi)$ 和 $\text{Im}(\Phi)$ 。
b. 确定变换
矩阵 A_{\sim} 关于基础

$$A_{\sim} = \begin{pmatrix} 2 & 2 & 2 \\ 2 & 2 & 0 \\ 2 & 0 & 0 \end{pmatrix},$$

即,对新的基础 B 执行基础更改。以标准基础表示

2.20 让我们将 \mathbb{R}^2 的 b_1, b_2, z_1, z_2 的 \mathbb{R}^2 的 4 个向量
视为

$$b_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \quad z_1 = \begin{pmatrix} 2 \\ -2 \end{pmatrix}, \quad z_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

让我们定义两个有序基数 $B = (b_1, b_2)$ 和 $B' = (z_1, z_2)$ 的 \mathbb{R}^2 。

A. 表明B和B
 B c执行基础变化的矩阵P 1。我们考虑标准基础中定义的
 c1、c2、c3、R3的三个向量
 R3作为_

$$c1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \quad c2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \quad c3 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

我们定义C = (c1, c2, c3)。
 (i) 证明C是R3的基础,例如,通过使用行列式 (参见第 4.1 节)。

3) R3的标

础变化的矩阵P 2的数据。确定(ii) 让我们称C = (c执行从C到C d 的基
 考虑同态Φ : R2 → R3

, 这样

$$\Phi(b1 + b2) = c2 + c3 \quad \Phi(b1 - b2) = 2c1 - c2 + 3c3$$

其中B = (b1, b2)和C = (c1, c2, c3)分别是R2和R3的有序基。

确定Φ的变换矩阵AΦ相对于或 dered 基B和C。e. 确定A B 和C

, Φ相对于基数的变换矩阵

F. 让我们考虑坐标在B中的向量x ∈ R2

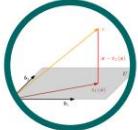
换句话说, x = 2b (i) 计 1 + 3b 2.

算x在B 中的坐标。 (ii) 在此基础上,计算用C
 表示的Φ(x)的坐标。 (iii) 然后,将Φ(x)写成C

(iv) 直接使用B结果中x的表示。 2, 换取矩
 阵A 来找到这个

3个

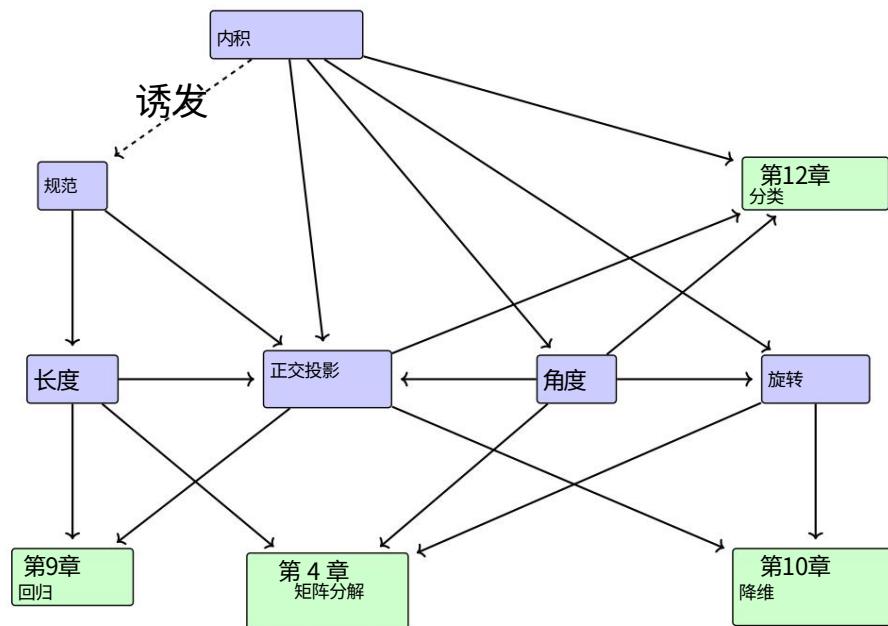
解析几何



在第 2 章中,我们在一般但抽象的层次上研究了向量、向量空间和线性映射。在本章中,我们将为所有这些概念添加一些几何解释和直觉。特别是,我们将查看几何向量并计算它们的长度和两个向量之间的距离或角度。为了能够做到这一点,我们为向量空间配备了一个内积,它可以导出向量空间的几何形状。内积及其相应的范数和度量捕获了相似性和距离的直观概念,我们在第 12 章中使用它们来开发支持向量机。然后我们将使用向量之间的长度和角度的概念来讨论正交投影,这将发挥当我们在第 10 章中讨论主成分分析和在第 9 章中通过最大似然估计进行回归时,它扮演着核心角色。图 3.1 概述了本章中的概念如何相关以及它们与本书其他章节的联系。

图 3.1 本文介绍概念的思维导图

章,以及它们在本书其他部分中的使用时间。



3.1 规范

71

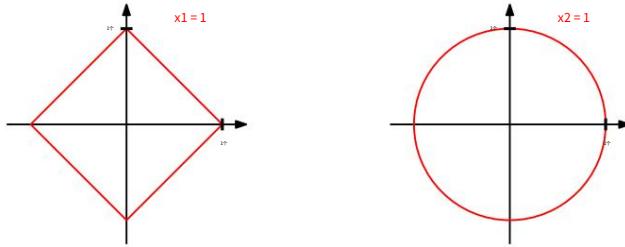


图 3.1 对于不同的范数,红线表示向量集

范数 1。左:曼哈顿范数;
右:欧氏距离。

3.1 规范

当我们想到几何向量时,即从原点开始的有向线段,那么直觉上向量的长度就是该有向线段的“末端”与原点的距离。下面,我们将使用范数的概念来讨论向量长度的概念。

定义 3.1 (标准)。向量空间V上的范数是一个函数

规范

$$\|\cdot\| : V \rightarrow \mathbb{R} \quad , \quad (3.1)$$

$$x \mapsto \|x\| \quad , \quad (3.2)$$

它为每个向量 x 分配其长度 $\|x\| \in \mathbb{R}$,使得对于所有 $\lambda \in \mathbb{R}$ 长度和 $x, y \in V$,以下成立:

- 绝对齐次: $\|\lambda x\| = |\lambda| \|x\|$
- 三角不等式: $\|x+y\| \leq \|x\| + \|y\|$
- 正定: $\|x\| = 0$ 和 $\|x\| = 0 \iff x = 0$

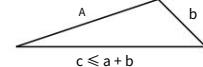
绝对同质

三角不等式

肯定的

在几何术语中,三角形不等式表示对于任何三角形,任意两条边的长度之和必须大于或等于剩余边的长度;参见图 3.2 的说明。

图 3.2 三角不等式。

定义 3.1 是根据一般向量空间V (第 2.4 节),但在本书中我们将只考虑有限维向量空间 \mathbb{R}^n 。

回想一下,对于向量 $x \in \mathbb{R}^n$,我们使用下标表示向量的元素,即 x_i 是向量 x 的第 i 个元素。

示例 3.1 (曼哈顿范数)

 \mathbb{R}^n 上的曼哈顿范数对于 $x \in \mathbb{R}^n$ 定义为

曼哈顿常态

$$\|x\|_1 := \sum_{i=1}^n |x_i| \quad , \quad (3.3)$$

哪里 $| \cdot |$ 是绝对值。图 3.1 的左面板显示了所有向量 $x \in \mathbb{R}^2$ 且 $\|x\|_1 = 1$ 。曼哈顿范数也称为 ℓ_1 范数

规范。

欧氏范数

示例 3.2 (欧几里德范数)
 $x \in R^n$ 的欧几里德范数定义为

$$\|x\|_2 := \sqrt{\sum_{i=1}^n x_i^2} \quad (3.4)$$

欧氏距离并计算 x 与原点的欧氏距离。图 3.1 的右图显示了所有向量 $x \in R^2$, 且 $\|x\|_2 = 1$ 。欧几里德范数也称为 ℓ_2 范数。

 ℓ_2 范数

评论。在本书中,如果没有特别说明,我们将默认使用欧几里德范数 (3.4)。 ◇

3.2 内积

内积允许引入直观的几何概念,例如向量的长度和两个向量之间的角度或距离。内积的一个主要目的是确定向量是否相互正交。

标量积点积

我们可能已经熟悉一种特定类型的内积,即 R^n 中的标量积/点积
 , 这是由

$$x \cdot y = \sum_{i=1}^n x_i y_i \quad (3.5)$$

在本书中,我们将这个特定的内积称为点积。然而,内积是具有特定属性的更一般的概念,我们现在将介绍它。

3.2.2 一般内积

双线性映射

回想一下 2.7 节中的线性映射,我们可以根据标量的加法和乘法重新排列映射。双线性映射 Ω 是具有两个参数的映射,并且它在每个参数中都是线性的,即,当我们查看向量空间 V 时,它认为对于所有 $x, y, z \in V, \lambda, \psi \in R$

$$\Omega(\lambda x + \psi y, z) = \lambda \Omega(x, z) + \psi \Omega(y, z) \quad (3.6)$$

$$\Omega(x, \lambda y + \psi z) = \lambda \Omega(x, y) + \psi \Omega(x, z) \quad (3.7)$$

这里,(3.6) 在第一个参数中断言 Ω 是线性的,并且 (3.7) 在第二个参数中断言 Ω 是线性的 (另请参见 (2.87)) 。

定义 3.2。设 V 是一个向量空间， $\Omega : V \times V \rightarrow \mathbb{R}$ 是一个双线性映射，它采用两个向量并将它们映射到一个实数上。然后

- 如果 $\Omega(x, y) = \Omega(y, x)$ 对于所有 $x, y \in V$ ，则 Ω 称为对称的，即参数的对称顺序无关紧要。 Ω 称为正定的，如果

■

肯定的

$$\forall x \in V \setminus \{0\} : \Omega(x, x) > 0, \Omega(0, 0) = 0. \quad (3.8)$$

定义 3.3。设 V 是一个向量空间， $\Omega : V \times V \rightarrow \mathbb{R}$ 是一个双线性映射，它采用两个向量并将它们映射到一个实数上。然后

- 正定对称双线性映射 $\Omega : V \times V \rightarrow \mathbb{R}$ 称为 V 上的内积。我们通常写 x, y 而不是 $\Omega(x, y)$ 。
- 对 (V, \cdot, \cdot) 称为内积空间或 (实) 向量空间内积空间与内积。如果我们使用 (3.5) 中定义的点积，我们称 (V, \cdot, \cdot) 为欧氏向量空间。

在本书中，我们将这些空间称为内积空间。

内积

具有内积的向量空间

欧氏向量空间

示例 3.3 (不是点积的内积)

考虑 $V = \mathbb{R}^2$ 。如果我们定义

$$x, y := x_1 y_1 - (x_1 y_2 + x_2 y_1) + 2x_2 y_2 \quad (3.9)$$

那么 \cdot, \cdot 是内积，但不同于点积。证明将是一个练习。

3.2.3 对称正定矩阵

对称正定矩阵在机器学习中起着重要作用，它们是通过内积定义的。在 4.3 节中，我们将在矩阵分解的上下文中返回到对称正定矩阵。对称半正定矩阵的思想是内核定义的关键（第 12.4 节）。

考虑具有内积 $\cdot, \cdot : V \times V \rightarrow \mathbb{R}$ （参见定义 3.3）和 V 的有序基 $B = (b_1, \dots, b_n)$ 的 n 维向量空间 V 。回想一下 2.6.1 节，任何向量 $x, y \in V$ 都可以写成基向量的线性组合，使得 $x = \sum_{i=1}^n \psi_i b_i \in V$ 和 $y = \sum_{j=1}^n \lambda_j b_j \in V$ 对于合适的 $\psi_i, \lambda_j \in \mathbb{R}$ 。由于 y 的双线性=内积，它对所有 $x, y \in V$ 成立

$$x, y = \sum_{i=1}^n \psi_i b_i, \sum_{j=1}^n \lambda_j b_j = \sum_{i=1}^n \sum_{j=1}^n \psi_i b_i, b_j \lambda_j = x, Ay, \quad (3.10)$$

其中 $A_{ij} := b_i, b_j$ 和 x^T, y^T 是 x 和 y 相对于基 B 的坐标。这意味着内积 \cdot, \cdot 由 A 唯一确定。内积的对称性产品也意味着

是对称的。此外，内积的正定性意味着

$$\forall x \in V \setminus \{0\} : x^T A x > 0. \quad (3.11)$$

定义 3.4 (对称正定矩阵)。满足 (3.11) 的对称矩阵 $A \in \mathbb{R}^{n \times n}$ 称为对称矩阵、正定矩阵或简称为正定矩阵。如果只有 在 (3.11) 中成立，则称 A 是对称半正定的。

正定对称,半正定

例 3.4 (对称正定矩阵)

考虑矩阵

$$A_1 = \begin{pmatrix} 9 & 6 & 6 \\ 5 & 6 & 5 \\ 6 & 5 & 9 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 9 & 6 & 6 \\ 3 & 6 & 3 \\ 3 & 3 & 9 \end{pmatrix}. \quad (3.12)$$

A_1 是正定的，因为它是对称的并且

$$x^T A_1 x = x_1^2 + 12x_1 x_2 + 5x_2^2 \geq 9x_1^2 + 12x_1 x_2 + 5x_2^2 = (3x_1 + 2x_2)^2 \geq 0 \quad (3.13a)$$

$$2 = 9x_1^2 + 12x_1 x_2 + 5x_2^2 = (3x_1 + 2x_2)^2 \geq 0 \quad (3.13b)$$

对于所有 $x \in V \setminus \{0\}$ 。相反， A_2 是对称的但不是正定的，因为 $x^T A_2 x = 9x_1^2 + 12x_1 x_2 + 3x_2^2$ 可以小于 0，例如，对于 $x = [2, -3]^T$ 。因此 $x^T A_2 x = 9x_1^2 + 12x_1 x_2 + 3x_2^2 = 9(2)^2 + 12(2)(-3) + 3(-3)^2 = 36 - 72 + 27 = -9 < 0$

如果 $A \in \mathbb{R}^{n \times n}$ 是对称的、正定的，则

$$x^T A x = x^T A^T x = x^T A x \quad (3.14)$$

定义关于有序基 B 的内积，其中 x^\wedge 和 y^\wedge 是 $x, y \in V$ 相对于 B 的坐标表示。

定理 3.5。对于实值有限维向量空间 V 和 a_n ，它认为 $\cdot, \cdot : V \times V \rightarrow \mathbb{R}$ 是内积，前提是 V 的有序基 B 且仅当存在对称正定矩阵 $A \in \mathbb{R}^{n \times n}$ 与

$$x, y = x^\wedge \quad Ay^\wedge. \quad (3.15)$$

如果 $A \in \mathbb{R}^{n \times n}$ 是对称且正定的，则以下属性成立：

- A 的零空间 (内核) 仅包含 0，因为对于所有 $x = 0$ ， $x^T A x = 0$ 。这意味着如果 $x = 0$ ，则 $Ax = 0$ 。
- A 的对角线元素 a_{ii} 为正，因为 $a_{ii} = e_i^T A e_i$ 其中 e_i 是 \mathbb{R}^n 中标准基的第 i 个向量。 $e_i^T A e_i > 0$ ，

3.3 长度和距离在 3.1 节中,我们已

经讨论了可以用来计算向量长度的范数。内积和规范密切相关,因为任何内积都可以导出规范

内积诱导规范。

$$\|x\| := \sqrt{x, x} \quad (3.16)$$

以一种自然的方式,这样我们就可以使用内积来计算向量的长度。然而,并不是每一个范数都是由内积引起的。曼哈顿范数 (3.3) 是没有相应内积的范数示例。在下文中,我们将重点关注由内积导出的范数,并介绍几何概念,例如长度、距离和角度。

备注 (Cauchy-Schwarz 不等式)。对于内积向量空间 (V, \cdot, \cdot) , 导出范数 $\|\cdot\|$ 满足 Cauchy-Schwarz 不等式

柯西-施瓦茨不等式

$$|x, y| \leq \|x\| \|y\|. \quad (3.17)$$



例 3.5 (使用内积的向量长度)

在几何学中,我们通常对向量的长度感兴趣。我们现在可以使用内积来使用 (3.16) 来计算它们。让我们取 $x = [1, 1] \in \mathbb{R}^2$ 。

如果我们使用点积作为内积,我们得到 (3.16)

$$\|x\| = \sqrt{x \cdot x} = \sqrt{1 \cdot 1 + 1 \cdot 1} = \sqrt{2} \quad (3.18)$$

作为 x 的长度。现在让我们选择一个不同的内积:

$$x, y := x - y \cdot y = x_1 y_1 - 2 - (x_1 y_2 + x_2 y_1) + x_2 y_2. \quad (3.19)$$

如果我们计算向量的范数,那么如果 x_1 和 x_2 具有相同的符号 (且 $x_1 x_2 > 0$), 则此内积返回的值小于点积;否则,它返回比点积更大的值。有了这个内积,我们得到

$$x, x = x \cdot x - x_1 x_2 + x_2 x_1 = 1 - 1 + 1 = 1 \Rightarrow \|x\| = \sqrt{1} = 1, \quad (3.20)$$

这样 x 与这个内积比与点积的“更短”。

定义 3.6 (距离和公制)。考虑一个内积空间 (V, \cdot, \cdot) 。然后

$$d(x, y) := \|x - y\| = \sqrt{x - y, x - y} \quad (3.21)$$

称为 x 和 y 之间的距离,其中 $x, y \in V$ 。
么这个距离就叫做欧式距离。欧式距离

如果我们用点距积作为内积,那

映射

$$d : V \times V \rightarrow \mathbb{R} \quad (3.22)$$

$$(x, y) \mapsto d(x, y) \quad (3.23)$$

公制

称为度量。

评论。类似于向量的长度，向量之间的距离不需要内积：范数就足够了。如果我们有一个由内积导出的范数，则距离可能会根据内积的 \diamond 选择而变化。

度量 d 满足以下条件：

肯定的

1. d 是正定的，即对于所有 $x, y \in V$ 和 $d(x, y) = 0 \iff x = y, d(x, y) > 0$ 。 2. d 是对称的，即对于所

对称三角不等式

有 $x, y \in V$ ， $d(x, y) = d(y, x)$ 。

3. 三角不等式： $d(x, z) \leq d(x, y) + d(y, z)$ 对于所有 $x, y, z \in V$ 。

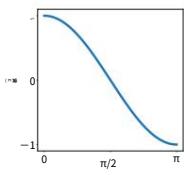
评论。乍一看，内积和度量的属性列表看起来非常相似。然而，通过比较定义 3.3 和定义 3.6，我们观察到 x, y 和 $d(x, y)$ 的行为方向相反。

非常相似的 x 和 y 将导致内积值大而度量值小。 \diamond

图 3.2 当限制为 $[0, \pi]$
时， $f(\omega) = \cos(\omega)$ 返回
一个唯一的数字

3.4 角度和正交性除了能够定义向量的长度

以及两个向量之间的距离之外，内积还通过定义两个向量之间的角度 ω 来捕获向量空间的几何形状。我们使用 Cauchy-Schwarz 不等式 (3.17) 来定义两个向量 x, y 之间的内积空间中的角 ω ，这个概念与我们在 R2 和 R3 中的直觉一致。假设 $x = 0, y = 0$ 。那么

区间 $[-1, 1]$ 。

$$\text{因此, 存在唯一的 } \omega \in [0, \pi], \text{ 如图 3.2 所示, 其中} \quad (3.24)$$

$$\text{余弦 } \omega = \frac{x \cdot y}{\|x\| \|y\|}. \quad (3.25)$$

角度

数 ω 是向量 x 和 y 之间的角度。直观地，两个向量之间的角度告诉我们它们的方向有多相似。例如，使用点积， x 和 y 之间的角度 $= 4x$ ，即 y 是 x 的缩放版本，为 0：它们的方向相同。

3.4 角度和正交性

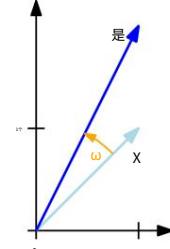
示例 3.6 (向量之间的角度)

让我们计算 $x = [1, 1] \in \mathbb{R}^2$ 和 $y = [1, 2] \in \mathbb{R}^2$ 之间的角度; 图 3.3 见图 3.3, 这里我们使用点积作为内积。然后我们使用内积计算两个向量 x, y 之间的角度 ω 。

$$\text{余弦 } \omega = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|} = \frac{\mathbf{x} \cdot \mathbf{y}}{\sqrt{\mathbf{x} \cdot \mathbf{x}} \sqrt{\mathbf{y} \cdot \mathbf{y}}} = \frac{\mathbf{x} \cdot \mathbf{y}}{\sqrt{\mathbf{x} \cdot \mathbf{x}} \sqrt{\mathbf{y} \cdot \mathbf{y}}} = \frac{3}{\sqrt{10}}, \quad (3.26)$$

并且两个向量之间的角度是 $\arccos(\frac{3}{\sqrt{10}}) \approx 0.32$ 弧度, 其中

内积的一个关键特征是它还允许我们表征正交的向量。



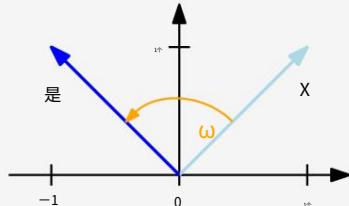
正交的

定义 3.7 (正交性)。两个向量 x 和 y 是正交的当且仅当 $x \cdot y = 0$ 时正交, 我们记为 $x \perp y$ 。如果另外 $\|x\| = \|y\| = 1$, 即向量是单位向量, 则 x 和 y 是正交的。

这个定义的含义是 0 向量正交于向量空间中的每个向量。

评论。正交性是将垂直性概念推广到不必是点积的双线性形式。在我们的上下文中, 从几何上讲, 我们可以将正交向量视为与特定内积成直角。 ◇

例 3.7 (正交向量)

图 3.1 之间的夹角 ω

两个向量 x, y 可以根据内部变化

产品。

考虑两个向量 $x = [1, 1], y = [-1, 1] \in \mathbb{R}^2$; 见图 3.1。我们有兴趣使用两个不同的内积来确定它们之间的角度 ω 。使用点积作为内积在 x 和 y 之间产生 90° 的角 ω , 使得 $x \perp y$ 。但是, 如果我们选择内积

$$\mathbf{x}, \mathbf{y} = \mathbf{x} \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad (3.27)$$

我们得到x和y之间的角度 ω 由下式给出

$$\text{余弦} \omega = \frac{\mathbf{x}, \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|} = -\frac{1}{3} \Rightarrow \omega \approx 1.91 \text{ 弧度} \approx 109.5^\circ, \quad (3.28)$$

x 和 y 不正交。因此,与一个内积正交的向量不一定与不同的内积正交。

定义 3.8 (正交矩阵)。一个方阵 $A \in \mathbb{R}^{n \times n}$ 是一个正交矩阵当且仅当它的列是正交的使得

正交矩阵

$$AA^T = I = A^T A, \quad (3.29)$$

这意味着

$$A^{-1} = A^T, \quad (3.30)$$

这是惯例
称这些矩阵为“正
交”,但更精确的描
述是

是“正交”的。

转换
与正交矩阵保持距离
和角度。

即,通过简单地转置矩阵获得逆。

正交矩阵的变换是特殊的,因为当使用正交矩阵 A 对其进行变换时,向量 x 的长度不会改变。对于点积,我们得到 $(Ax) \cdot (Ax) = x \cdot A^T A x = x \cdot \|x\|^2 = \|x\|^2$

此外,任意两个向量 x 、 y 之间的角度,由它们的内积测量,在使用正交矩阵 A 对它们进行变换时也保持不变。假设点积为内积,则图像 Ax 和 Ay 的角度给出为

$$\text{余弦} \omega = \frac{(Ax) \cdot (Ay)}{\|Ax\| \|Ay\|} = \frac{\mathbf{x} \cdot A^T A \mathbf{y}}{\mathbf{x} \cdot A^T A \mathbf{y}} = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|}, \quad (3.32)$$

它给出了 x 和 y 之间的精确角度。这意味着正交矩阵 A 和 A^T 保留了角度和距离。事实证明,正交矩阵定义的变换是旋转的(有翻转的可能性)⁻¹。在第 3.9 节中,我们将讨论有关旋转的更多细节。

3.5 正交基

在 2.6.1 节中,我们刻画了基向量的性质,发现在一个 n 维向量空间中,我们需要 n 个基向量,即 n 个线性无关的向量。在 3.3 和 3.4 节中,我们使用内积来计算向量的长度和向量之间的角度。在下文中,我们将讨论基向量彼此正交且每个基向量的长度为 1 的特殊情况。我们将此基称为标准正交基。

让我们更正式地介绍一下。

定义 3.9 (正交基)。考虑一个 n 维向量空间 V 和一个基 $\{b_1, \dots, b_n\}$ 的 V 。

如果

$$b_i, b_j = 0 \text{ 对于 } i = j \quad b_i, b_i = 1 \quad (3.33)$$

$$(3.34)$$

对于所有 $i, j = 1, \dots, n$ 则该基称为正交基(ONB)。正交基如果仅满足 (3.33), 则该基称为正交基。请注意 (3.34) 意味着每个基向量的长度/范数为1。
ONB
正交基

回想一下 2.6.1 节, 我们可以使用高斯消去法为由一组向量跨越的向量空间找到一个基。假设我们有一个集合 $\{\sim b_1, \dots, \sim b_n\}$ 的非正交和非归一化基向量。我们将它们连接成一个矩阵 $B \sim = [\sim b_1, \dots, \sim b_n]$ 并将高斯消除应用于增广矩阵 (第 2.3.2 节) $[B \sim | B \sim]$ 正交基。这种以迭代方式构建正交基础 $\{b_1, \dots, b_n\}$ 称为 Gram-Schmidt 过程(Strang, 2003)。

$|B \sim|$ 得到一个

例 3.8 (正交基)

欧几里得向量空间 R^n 的规范/标准基是一个或正态基, 其中内积是向量的点积。载体

在 R^2 ,

$$b_1 = \sqrt{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad b_2 = \sqrt{2} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad (3.35)$$

形成标准正交基, 因为 $b_1 \cdot b_2 = 0$ 且 $\|b_1\| = \|b_2\| = \sqrt{2}$ 。

在第 12 章和第 10 章讨论支持向量机和主成分分析时, 我们将利用正交基的概念。

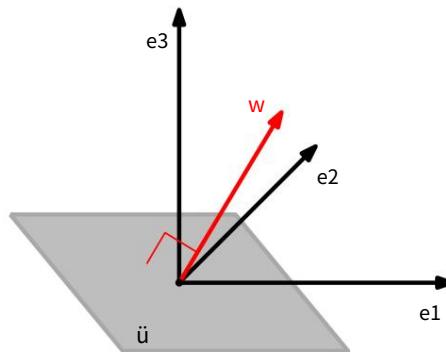
3.6 正交补定义了正交性之后, 我们现在来看

一下相互正交的向量空间。这将在第 10 章中发挥重要作用, 届时我们将从几何角度讨论线性降维。

考虑一个 D 维向量空间 V 和一个 M 维子空间

空间 $U \subseteq V$ 。那么它的正交补集 V 的 U 子空间并且包含 V 中所有正交¹ 是 $(D - M)$ 维正交补集于 every $= \{0\}$ 的向量, 因此任何向量 $x \in V$ 都可以是 U 中的向量。此外, $U \cap U^\perp$

图 3.1 A 平面
你在一个
三维向量空间可以用它
的法向量来描述,它跨越它
的正交补集 U^\perp 。



唯一地分解成

$$x = \sum_{m=1}^M \lambda_m b_m + \sum_{j=1}^{D-M} \psi_j b_j^\perp, \quad \lambda_m, \psi_j \in \mathbb{R}, \quad (3.36)$$

其中 (b_1, \dots, b_M) 是 U 和 (b) 的基, 因此正交补 $(b_1^\perp, \dots, b_{D-M}^\perp)$ 是 U^\perp 上的基。
也可以用来描述三维向量空间中的一个平面 U (二维子空间)。

更具体地说, 与平面 U 正交的 $\|w\| = 1$ 的向量 w 是 U 上的基向量。图 3.1 说明了此设置。所有与 w 正交的向量必须 (通过构造) 位于平面 U 中。向量 w 称为 U 的法向量。

法向量

通常, 正交补集可用于描述 n 维向量和仿射空间中的超平面。

3.7 函数的内积

到目前为止, 我们研究了内积的属性来计算长度、角度和距离。我们专注于有限维向量的内积。在下文中, 我们将看一个不同类型向量的内积示例: 函数的内积。

到目前为止我们讨论的内积是为具有有限个条目的向量定义的。我们可以将向量 $x \in \mathbb{R}^n$ 视为具有 n 个函数值的函数。内积的概念可以推广到具有无限个条目 (可数无限) 和连续值函数 (不可数无限) 的向量。然后向量的各个分量的总和 (例如参见等式 (3.5)) 变成一个积分。

两个函数 $u : \mathbb{R} \rightarrow \mathbb{R}$ 和 $v : \mathbb{R} \rightarrow \mathbb{R}$ 的内积可以是
定义为定积分

$$u, v := \int_a^b u(x)v(x)dx \quad (3.37)$$

3.8 正交投影

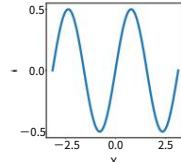
对于下限和上限 $a, b < \infty$, 分别。与我们通常的内积一样, 我们可以通过查看内积来定义范数和正交性。如果 (3.37) 的计算结果为 0, 则函数 u 和 v 是正交的。为了使前面的内积在数学上精确, 我们需要注意度量和积分的定义, 从而导致希尔伯特空间的定义。此外, 与有限维向量的内积不同, 函数的内积可能发散 (具有无限值)。所有这些都需要深入研究实数分析和泛函分析的一些更复杂的细节, 我们在本书中没有涉及这些细节。

例 3.9 (函数的内积)

如果我们选择 $u = \sin(x)$ 和 $v = \cos(x)$, 则被积函数 $f(x) = u(x)v(x)$ 图 3.2 $f(x) = (3.37)$, 如图 3.2 所示。我们看到这个函数是奇函数, 即 $f(-x) = -f(x)$ 。因此, 该乘积的极限 $a = -\pi, b = \pi$ 的积分计算结果为 0。因此, \sin 和 \cos 是正交函数。

评论。它还认为函数的集合

$$\{1, \cos(x), \cos(2x), \cos(3x), \dots\} \quad (3.38)$$



如果我们从 $-\pi$ 到 π 积分, 则它是正交的, 即任何一对函数彼此正交。(3.38) 中的函数集合跨越 $[-\pi, \pi]$ 上的偶数周期性函数的大子空间, 将函数投影到该子空间是傅里叶级数背后的基本思想。
◇ 在 6.4.6 节中, 我们将了解第二种非常规内积: 随机变量的内积。

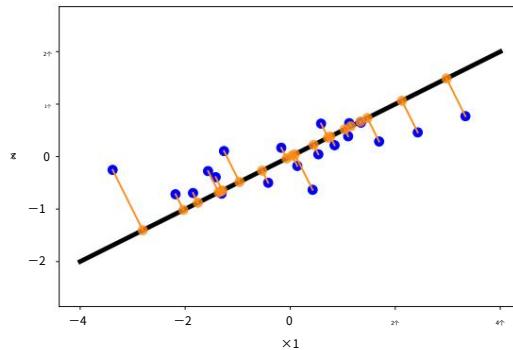
3.8 正交投影

投影是一类重要的线性变换 (除旋转和反射外), 在图形、编码理论、统计和机器学习中起着重要作用。在机器学习中, 我们经常处理高维数据。高维数据通常难以分析或可视化。然而, 高维数据往往具有只有少数几个维度包含最多信息的特性, 而大多数其他维度对于描述数据的关键属性不是必需的。当我们压缩或可视化高维数据时, 我们会丢失信息。为了最小化这种压缩损失, 我们理想地找到数据中信息量最大的维度。正如第 1 章所讨论的, “特征”是数据可以表示为向量的一种常见表达方式, 而在本章中, 我们将针对数据讨论一些数据压缩的基本工具。更具体地说, 我们可以将原始高维数据投影到低维特征空间, 并在这个低维空间中工作以了解更多关于数据集的信息并提取相关模式。例如, 机

表示。

图 3.1 二维数据集 (蓝点) 的正交投影 (橙点)

到一个
一维子空间 (直线)。



学习算法,例如 Pearson (1901) 和 Hotelling (1933) 的主成分分析 (PCA) 和深度神经网络 (例如,深度自动编码器 (Deng et al., 2010)) ,大量利用了维度的概念减少。在下文中,我们将重点关注正交投影,我们将在第 10 章中使用它进行线性降维,在第 12 章中使用它进行分类。甚至我们在第 9 章讨论的线性回归也可以使用正交投影来解释。对于给定的低维子空间,高维数据的正交投影保留尽可能多的信息,并最小化原始数据与对应投影之间的差异/误差。图 3.1 给出了这种正交投影的图示。在我们详细说明如何获得这些投影之前,让我们先定义投影实际上是什么。

定义 3.10 (预测)。令 V 为向量空间, $U \subseteq V$ 为 V 的子空间。

投影

线性映射 $\pi : V \rightarrow U$ 称为投影,如果

$$\pi_2 \circ \pi = \pi \quad \pi = \pi_0$$

由于线性映射可以用变换矩阵表示 (见第 2.7 节),前面的定义同样适用于一种特殊的变换矩阵,投影矩阵 P 表现出 P

投影矩阵

$$\pi^* = P \downarrow_{\pi_0}$$

π_0

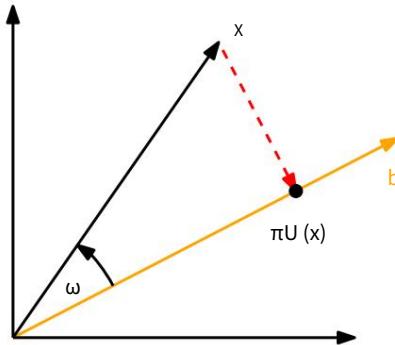
面,我们将导出内积空间 (R^n , \cdot, \cdot) 中的向量在子空间上的正交投影。我们将从一维子空间开始,也称为线。如果没有另外说明,我们假设点积 $x, y = x \cdot y$ 作为内积。

线

3.8.1 投影到一维子空间 (线)

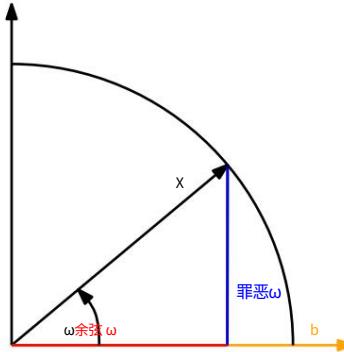
假设给定一条通过原点的直线 (一维子空间), 基向量 $b \in R^n$ 。该线是由 b 跨越的一维子空间 $U \subseteq R^n$ 。当我们把 $x \in R^n$ 投影到 U 上时, 我们寻找最接近 x 的向量 $\pi_U(x) \in U$ 。使用几何参数, 让我们

3.8 正交投影

(a) 将 $x \in \mathbb{R}^2$ 投影到具有基向量 b 的子空间 U 上。

83

图 3.2—维子
空间的投影示
例。

(b) 将 $\|x\| = 1$ 的二维向量 x 投影到由 b 跨越的一维子空
间上。表征投影 $\pi_U(x)$ 的一些性质 (图3.2(a)作为说明) :

- 投影 $\pi_U(x)$ 最接近 x , 其中 “最接近” 意味着距离 $\|x - \pi_U(x)\|$ 是最小的。由此得出, 从 $\pi_U(x)$ 到 x 的线段 $\pi_U(x) - x$ 与 U 正交, 因此 U 的基向量 b 。正交条件产生 $\pi_U(x) - x, b = 0$, 因为角度向量之间通过内积定义。
- x 在 U 上的投影 $\pi_U(x)$ 必须是 U 的一个元素, 因此是跨越 U 的基向量 b 的倍数。因此, 对于某些 $\lambda \in \mathbb{R}$, $\pi_U(x) = \lambda b$ 。

λ 则为
 $\pi_U(x)$ 相对于 b 的坐标。

在以下三个步骤中, 我们确定坐标 λ 、投影 $\pi_U(x) \in U$ 和投影矩阵 P π 将任何 $x \in \mathbb{R}^n$ 映射到 U 上:1. 求坐标 λ 。正交条件产生

$$x - \pi_U(x), b = 0 \Leftrightarrow \frac{\pi_U(x)}{\|b\|} = \lambda \Rightarrow x - \lambda b, b = 0. \quad (3.39)$$

我们现在可以利用内积的双线性并得出

$$b \cdot x - \lambda b \cdot b = 0 \Leftrightarrow \lambda = \frac{x \cdot b}{b \cdot b} = \frac{b \cdot x}{\|b\|^2}. \quad (3.40)$$

对于一般内积, 如果
 $\|b\| = 1$, 我们
得到 $\lambda = x \cdot b$ 。

在最后一步中, 我们利用了内积是对称的这一事实。如果我们选择 \cdot, \cdot 作为点积, 我们得到

$$\lambda = \frac{b \cdot x}{\|b\|^2} = \frac{x \cdot b}{\|b\|^2}. \quad (3.41)$$

如果 $\|b\| = 1$, 则投影的坐标 λ 由 b 给出 x 。

2. 求投影点 $\pi U(x) \in U$ 。由于 $\pi U(x) = \lambda b$, 我们用 (3.40) 立即得到

$$b - \pi U(x) = \lambda b = b = \frac{x,}{\|b\|^2} \frac{b}{\|b\|^2}, \quad (3.42)$$

最后一个等式仅适用于点积。我们还可以通过定义 3.1 计算 $\pi U(x)$ 的长度为

$$\|\pi U(x)\| = \|\lambda b\| = |\lambda| \|b\|. \quad (3.43)$$

因此, 我们的投影长度为 λ 乘以 b 的长度。这也增加了直觉, 即 λ 是 $\pi U(x)$ 相对于跨越我们的一维子空间 U 的基向量 b 的坐标。

如果我们使用点积作为内积, 我们得到

$$\|\pi U(x)\| \stackrel{(3.42)}{=} \frac{\|b\| |x|}{\|b\|^2} \stackrel{(3.25)}{=} \frac{\|b\|}{\|b\|^2} = |\cos \omega| \frac{\|x\|}{\|b\|} = |\cos \omega| \frac{\|x\|}{\|b\|}. \quad (3.44)$$

这里, ω 是 x 和 b 之间的角度。这个等式在三角学中应该很熟悉: 如果 $\|x\| = 1$, 则 x 位于单位圆上。由此可见, b 所跨越的水平轴上的投影正好是 $\cos \omega$, 对应向量的长度 $\pi U(x) = |\cos \omega|$ 。图 3.2(b) 给出了一个说明。

横轴是一维的

子空间。

3. 求投影矩阵 P_{π} 。我们知道投影是线性映射 (见定义 3.10)。因此, 存在一个投影使得 $\pi U(x) = P_{\pi} x$ 。以点积作为内矩阵 P 积和

$$\pi U(x) = \lambda b = b\lambda = b \frac{\|x\|}{\|b\|} = \frac{bb}{\|b\|^2} x, \quad (3.45)$$

我们立即看到

$$P_{\pi} = \frac{bb}{\|b\|^2}. \quad (3.46)$$

投影矩阵总是对称的。

请注意, bb (因此, $P_{\pi} P_{\pi}$) 是一个对称矩阵 (秩为 1), 并且 $\|b\| = b$, b 是一个标量。

投影矩阵 P_{π} 将任何向量 $x \in R^n$ 投影到通过原点且方向为 b 的直线上 (等效地, 由 b 跨越的子空间 U)。

评论。投影 $\pi U(x) \in R^n$ 仍然是一个 n 维向量而不是标量。然而, 我们不再需要 n 个坐标来表示投影, 如果我们想表示它相对于 \diamond 跨越子空间 U 的基向量 b : λ , 则只需要一个坐标。

3.8 正交投影

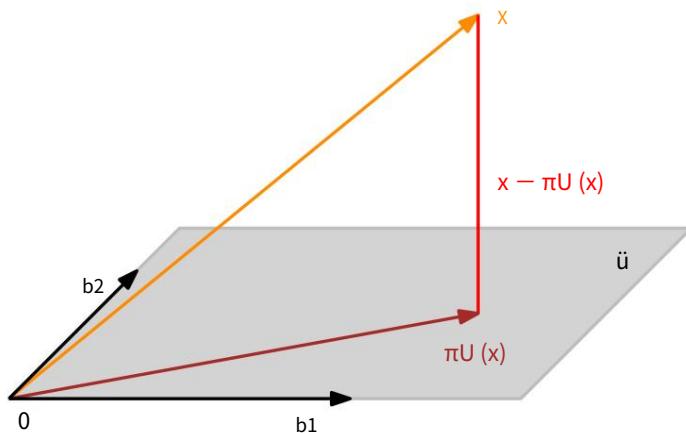


图 3.1 投影到具有基 b_1, b_2 的二维子空间 U 。 $x \in \mathbb{R}^3$ 在 U 上的投影 $\pi_U(x)$ 可以表示为线性组合

b_1, b_2 和位移矢量 $x - \pi_U(x)$ 与 b_1 和 b_2 正交。

例 3.10 (投影到一条线上)

通过 $b = [1 \ 2 \ 2]$ 子空间 (通过原点) 到通过原点的直线上 b 是方向和一维的基础的线) 找到投影矩阵 P 。

利用 (3.46), 我们得到

$$P_{\pi} = \frac{bb}{b^T b} = \frac{1 \ 2 \ 2}{1 \ 2 \ 2} \begin{matrix} 1 \\ 2 \\ 2 \end{matrix} = \frac{1 \ 2 \ 2}{1 \ 2 \ 2} = 9 \quad \begin{matrix} 1 & 2 & 2 \\ 2 & 4 & 4 \\ 2 & 4 & 4 \end{matrix} \quad (3.47)$$

现在让我们选择一个特定的 x 并查看它是否位于 b 所跨越的子空间中。对于 $x = [1 \ 1 \ 1]^T$, 投影是

$$\pi_U(x) = P_{\pi}x = 9 \begin{matrix} 1 & 2 & 2 \\ 2 & 4 & 4 \\ 2 & 4 & 4 \end{matrix} = \begin{matrix} 1 \\ 2 \\ 2 \end{matrix} \begin{matrix} 1 \\ 2 \\ 2 \end{matrix} = \begin{matrix} 1 \\ 2 \\ 2 \end{matrix} \in \text{跨度}[\begin{matrix} 1 \\ 2 \\ 2 \end{matrix}, \begin{matrix} 2 \\ 4 \\ 4 \end{matrix}] \quad (3.48)$$

$\pi_U(x)$ 。这是意料之中的, 因为根据定义 π_{π} 对 $\pi_U(x)$ 没有任何改变, 即注意 PP 的应用 $\pi\pi_U(x) = 3.10$, 我们知道投影矩阵 P

$$\pi_{\pi} \text{ 满足 } P \text{ 对于所有 } x, \pi_{\pi}x = P_{\pi}x.$$

评论。根据第 4 章的结果, 我们可以证明 $\pi_U(x)$ 是 $P \diamond$ 的特征向量
 π_{π} 对应的特征值为 1。

3.8.2 一般子空间上的投影下面, 我们看一下向量

$x \in \mathbb{R}^n$ 到低维子空间 $U \subseteq \mathbb{R}^n$ 上的正交投影, 其中 $\dim(U) = m - 1$ 。图 3.1 给出了说明。

如果 U 由一组不是基础的生成向量给出, 请确保确定基础 b_1, \dots, b_m 在继续之前。

假设 (b_1, \dots, b_m) 是 U 的有序基。任何在 U 上的投影 $\pi_U(x)$ 必然是 U 的一个元素。因此, 它们可以表示

基向量
形成 $B \in R^{n \times m}$ 的列,
其中 $B = [b_1, \dots, b_m]$ 。

作为基向量 b_1, \dots, b_m 的线性组合。... b_m of U , 使得 $\pi_U(x) =$ 与一维情况一样, 我们遵循三步程序
来找到投影 $\pi_U(x)$ 和投影矩阵 P_{π_U} :

1. 找到坐标 $\lambda_1, \dots, \lambda_m$ 的投影 (相对于 U 的基础), 使得线性组合

$$\pi_U(x) = \sum_{i=1}^m \lambda_i b_i = B\lambda, \quad (3.49)$$

$$B = [b_1, \dots, b_m] \in R^{n \times m}, \lambda = [\lambda_1, \dots, \lambda_m] \in R^m, \quad (3.50)$$

最接近 $x \in R^n$ 。在一维情况下, “最接近”意味着“最小距离”, 这意味着连接 $\pi_U(x) \in U$ 和 $x \in R^n$ 的向量必须与 U 的所有基向量正交。因此, 我们得到 m 个同时条件 (假设点积作为内积)

$$b_1, x - \pi_U(x) = b_1(x - \pi_U(x)) = 0 \quad (3.51)$$

⋮

$$b_m, x - \pi_U(x) = b_m(x - \pi_U(x)) = 0 \quad (3.52)$$

其中, $\pi_U(x) = B\lambda$, 可以写成

$$b_1(x - B\lambda) = 0 \quad (3.53)$$

⋮

$$b_m(x - B\lambda) = 0 \quad (3.54)$$

这样我们就得到了齐次线性方程组

$$\begin{matrix} & 1 \\ & \vdots \\ b_* & \end{matrix} x - B\lambda = 0 \iff B(x - B\lambda) = 0 \quad (3.55)$$

$$\iff B B\lambda = B x. \quad (3.56)$$

正规方程

最后一个表达式称为正规方程。自 b_1 以来, \dots, b_m 是 U 的基, 因此是线性无关的, $B \in R^{m \times n}$ 是正则的, 可以倒置。这使我们能够求解系数/坐标

$$\lambda = (B B)^{-1} B x. \quad (3.57)$$

伪逆

矩阵 $(B B)^{-1} B$ 也称为 B 的伪逆, 可以针对非方阵 B 计算。它只要求 B 是正定的, 如果 B 是满秩就是这种情况。在实际应用中 (例如, 线性回归), 我们经常添加一个“抖动项” ϵI

$B^\top B$ 以保证增加的数值稳定性和正定性。这个“脊”可以使用贝叶斯推理严格推导出来。

详见第 9 章。

2. 找到投影 $\pi_U(x) \in U$ 。我们已经确定 $\pi_U(x) = B\lambda$ 。因此,与 (3.57)

$$\pi_U(x) = B(B^\top B)^{-1}Bx \quad (3.58)$$

3. 求投影矩阵 P

由(3.58)式可知,求解 $P\pi x = \pi_U(x)$ 的投影矩阵

必为

$$P_\pi = B(B^\top B)^{-1}B \quad (3.59)$$

评论。投影到一般子空间的解决方案包括作为特殊情况的 1D 情况:如果 $\dim(U) = 1$,则 $B^\top B \in \mathbb{R}$ 是标量,我们可以重写 (3.59) 中的投影矩阵

$$P_\pi = \frac{BB}{B^\top B}, \text{ 是 (3.46) 中的投影矩阵。} \quad \diamond$$

示例 3.11 (投影到二维子空间)

对于子空间 $U = \text{span}[[1, 1, 0], [1, 1, 1], [2, 1, 2]] \subseteq \mathbb{R}^3$ 和 $x = [0, 0, 6]$ 找到

根据子空间 U 、投影点 $\pi_U(x)$ 和投影矩阵 P 坐标 x 的坐标入

首先,我们看到 U 的生成集是一个基 (线性独立

dence) 并将 U 的基向量写入矩阵 $B =$

$$\begin{matrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{matrix}.$$

其次,我们计算矩阵 $B^\top B$ 和向量 Bx

x 作为

$$B^\top B = \begin{matrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 2 \end{matrix} = \begin{matrix} 1 & 0 \\ 3 & 3 \\ 5 \end{matrix}, \quad Bx = \begin{matrix} 1 & 1 & 1 & 0 & 1 \\ 2 \end{matrix} = \begin{matrix} 6 \\ 0 \end{matrix}. \quad (3.60)$$

第三,我们求解正规方程 $B^\top B\lambda = Bx$

x 找到入:

$$\begin{matrix} 3 & 3 \\ 3 & 5 \end{matrix} \begin{matrix} \lambda_1 \\ \lambda_2 \end{matrix} = \begin{matrix} 6 \\ 0 \end{matrix} \Leftrightarrow \lambda = \begin{matrix} 5 \\ -3 \end{matrix}. \quad (3.61)$$

四、 x 在 U 上的投影 $\pi_U(x)$, 即投影到列空间 B , 可以通过直接计算

$$\pi_U(x) = B\lambda = \begin{matrix} 5 \\ 2 \\ -1 \end{matrix}. \quad (3.62)$$

投射误差

投影误差也称为重构误差。

对应的投影误差是原始向量与其在 U 上的投影之差向量的范数,即

$$\|x - \pi_U(x)\| = \sqrt{1 - 21} = \sqrt{6}。 \quad (3.63)$$

第五,投影矩阵 (对于任何 $x \in R^3$) 由下式给出

$$P_{\pi} = B(B^T B)^{-1}B^T = \begin{pmatrix} 5 & 2 & -1 \\ 2 & 2 & 2 \\ -1 & 2 & 5 \end{pmatrix} \quad (3.64)$$

为了验证结果,我们可以 (a) 检查位移向量 $\pi_U(x) - x$ 是否正交于 U 的所有基向量,以及 (b) 验证

$P_{\pi} = P_{\pi}^T$ (见定义 3.10)。

评论。投影 $\pi_U(x)$ 仍然是 R^n 中的向量,尽管它们位于 m 维子空间 $U \subseteq R^n$ 中。然而,为了表示投影矢量,我们只需要 m 个坐标 $\lambda_1, \dots, \lambda_m$ 关于基向量 b_1, \dots, b_m ◇备注。在具有一般内积的向量空间中,我们在计算角度和距离时要注意,它们由内积的 ◇均值 定义。

我们可以找
的近似解

使用投影的不可解
的线性方程系统。

投影让我们可以看到线性系统 $Ax = b$ 没有解的情况。回想一下,这意味着 b 不在 A 的范围内,即向量 b 不在 A 的列所跨越的子空间中。鉴于线性方程无法精确求解,我们可以找到一个近似解。这个想法是在 A 的列所跨越的子空间中找到最接近 b 的向量,即我们计算 b 在 A 的列所跨越的子空间上的正交投影。这个问题在实践中经常出现,并且解称为超定系统的最小二乘解 (假设点积为内积)。这将在第 9.4 节中进一步讨论。使用重构误差 (3.63) 是推导主成分分析的一种可能方法 (第 10.3 节)。

最小二乘法

评论。我们刚刚研究了向量 x 到具有基向量 $\{b_1, \dots, b_m\}$ 。如果这个基是一个 ONB, 即满足 (3.33) 和 (3.34), 则投影方程 (3.58) 大大简化为 $\pi_U(x) = Bx$

$$x \quad (3.65)$$

因为 $B^T B = I$ 坐标

$$\lambda = B^T x。 \quad (3.66)$$

这意味着我们不再需要计算 (3.58) 的逆函数, ◇从而节省了计算时间。

3.8.3 Gram-Schmidt 正交化投影是 Gram-Schmidt 方

法的核心,它允许我们建设性地将n维向量空间V的任何基(b_1, \dots, b_n)转换为正交/正交基(u_1, \dots, u_n)的V。这个基础总是存在的(Liesen and Mehrmann, 2015)和span[b₁, ..., b_n]=跨度[u₁, ..., u_n]。Gram - Schmidt 正交化方法迭代地从V的任意基(b_1, \dots, b_n)构建正交基(u_1, \dots, u_n),如下所示:

正交化

$$u_1 := b_1 \quad (3.67)$$

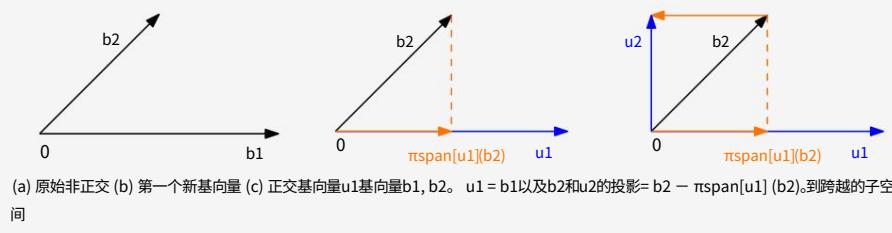
$$u_k := b_k - \pi_{\text{span}[u_1, \dots, u_{k-1}]}(b_k), k = 2, \dots, n \quad (3.68)$$

在(3.68)中,第k个基向量 b_k 被投影到由前 $k-1$ 个构造的正交向量 u_1, \dots, u_{k-1} 跨越的子空间。... $, u_{k-1}$,请参阅第3.8.2节。然后从 b_k 中减去该投影,得到一个向量 u_k ,它与 u_1, \dots, u_{k-1} 跨越的(k-1)维子空间正交。... $, u_{k-1}$ 。对所有n个基向量 b_1, \dots, b_n ,重复此过程。 \dots, b_n 产生V的正交基(u_1, \dots, u_n)。

如果我们对 u_k 进行归一化,我们将获得一个ONB,其中 $\|u_k\| = 1$ 对于 $k = 1, \dots, n$,

名词

示例 3.12 (Gram-Schmidt 正交化)



間

考慮R2的基础(b_1, b_2)

, 在哪里

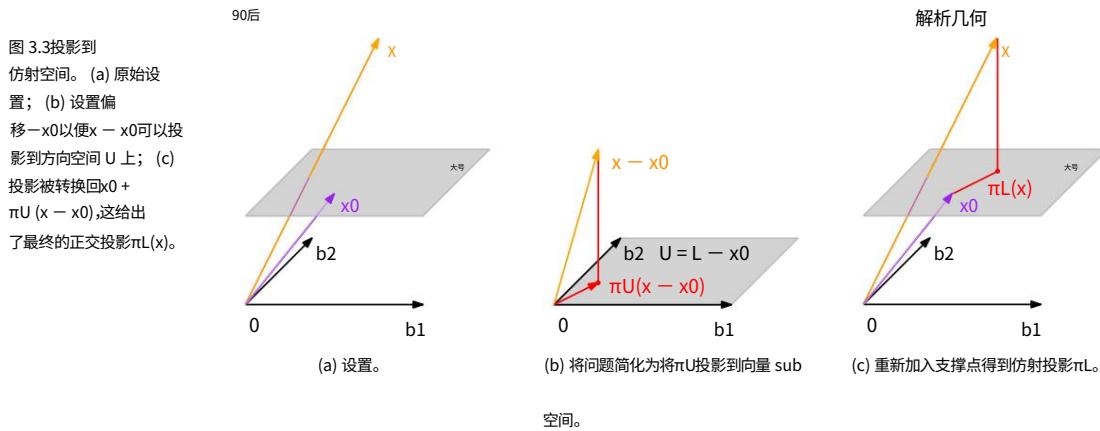
$$b_1 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}; \quad (3.69)$$

另请参见图3.2(a)。使用Gram-Schmidt方法,我们构造R2的正交基(u_1, u_2)如下(假设点积为内积):

$$u_1 := b_1 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \quad (3.70)$$

$$u_2 := b_2 - \pi_{\text{span}[u_1]}(b_2) \stackrel{(3.45)}{=} b_2 - \frac{\langle b_2, u_1 \rangle}{\|u_1\|^2} u_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \frac{1 \cdot 2}{\sqrt{4+0}} \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \frac{2}{2} \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (3.71)$$

图 3.2
Gram-Schmidt
正交化。(a) R2的
非正交基(b_1, b_2) ;
(b) 首先构造基向量 u_1
和 b_2 在span[u₁]上的
正交投影; (c) R2的
正交基(u_1, u_2)。



这些步骤如图 3.2(b) 和 (c) 所示。我们立即看到 u_1 和 u_2 是正交的, 即 $u_1 \cdot u_2 = 0$ 。

3.8.4 投影到仿射子空间

到目前为止, 我们讨论了如何将向量投影到低维子空间 U 。下面, 我们提供将向量投影到仿射子空间的解决方案。

考虑图 3.3(a) 中的设置。给定一个仿射空间 $L = x_0 + U$, 其中 b_1 、 b_2 是 U 的基向量。为了确定 x 在 L 上的正交投影 $\pi_L(x)$, 我们将问题转化为一个我们知道如何解决的问题: 到向量子空间的投影。为了到达那里, 我们从 x 和 L 中减去支撑点 x_0 , 因此 $L - x_0 = U$ 恰好是向量子空间 U 。我们现在可以在子空间上使用正交投影, 我们在 3.8.2 节中讨论过, 并且获得投影 $\pi_U(x - x_0)$, 如图 3.3(b) 所示。

现在可以通过添加 x_0 将该投影转换回 L , 这样我们就可以得到仿射空间 L 上的正交投影为

$$\pi_L(x) = x_0 + \pi_U(x - x_0), \quad (3.72)$$

其中 $\pi_U(\cdot)$ 是在子空间 U 上的正交投影, 即 L 的方向空间; 见图 3.3(c)。

从图 3.3 中也可以明显看出 x 与仿射的距离
空间 L 等于 $x - x_0$ 到 U 的距离, 即

$$d(x, L) = \|x - \pi_L(x)\| = \|x - (x_0 + \pi_U(x - x_0))\| \quad (3.73a)$$

$$= d(x - x_0, \pi_U(x - x_0)) = d(x - x_0, U)。 \quad (3.73b)$$

我们将在 12.1 节中使用到仿射子空间的投影来推导分离超平面的概念。

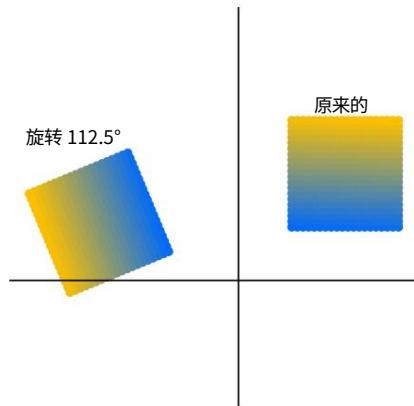


图 3.2 旋转旋转

关于原点的平面内的物体。如果旋转角度为正,我们旋转逆时针方向。

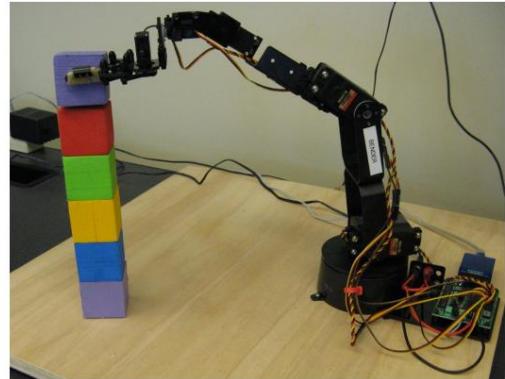


图3.1
机械臂需要
旋转它的关节以拾取物
体或正确放置它们。

图取自
(Deisenroth et al.,
2015)。

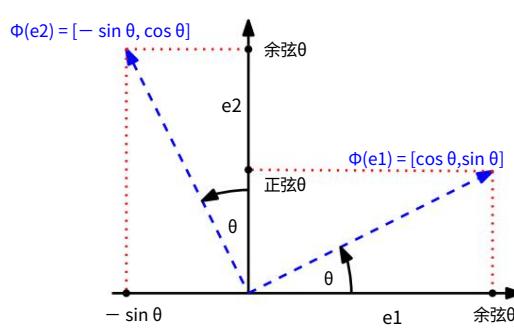
3.9 旋转

如第 3.4 节所述,长度和角度保持是具有正交变换矩阵的线性映射的两个特征。在下文中,我们将仔细研究描述旋转的特定正交变换矩阵。

旋转是线性映射 (更具体地说,旋转的自同构一个欧几里得向量空间), 将一个平面绕原点旋转一个角度 θ , 即原点是一个固定点。对于正角 $\theta > 0$, 按照惯例, 我们逆时针方向旋转。示例如图 3.2 所示, 其中变换矩阵为

$$R = \begin{pmatrix} -0.38 & -0.92 \\ 0.92 & -0.38 \end{pmatrix}. \quad (3.74)$$

旋转的重要应用领域包括计算机图形学和机器人技术。例如, 在机器人技术中, 了解如何旋转机械臂的关节以拾取或放置物体通常很重要, 请参见图 3.1。

图 3.2 标准基在 R2 中旋转角度 θ 。

3.9.1 R2 中的旋转

考虑标准基础 $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ R2 的 , 它定义了

R2 中的标准坐标系。我们的目标是将此坐标系旋转角度 θ , 如图 3.2 所示。请注意 , 旋转后的向量仍然线性无关 , 因此是 R2 的基础。

这意味着旋转执行基础更改。

旋转中是线性映射 , 因此我们可以用旋转矩阵 $R(\theta)$ 来表示它们。三角学 (见图 3.2) 允许我们确定旋转轴 (Φ 的图像) 相对于 R2 中标准基的坐标。我们获得

$$\Phi(e_1) = \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix}, \Phi(e_2) = \begin{pmatrix} -\sin \theta \\ \cos \theta \end{pmatrix}. \quad (3.75)$$

因此 , 将基变换为旋转坐标 $R(\theta)$ 的旋转矩阵为

$$R(\theta) = \Phi(e_1) \Phi(e_2) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}. \quad (3.76)$$

3.9.2 R3 中的旋转

与 R2 的情况相反 , 在 R3 中我们可以围绕一维轴旋转任何二维平面。指定一般旋转矩阵的最简单方法是指定标准基 e_1 、 e_2 、 e_3 的图像应该如何旋转 , 并确保这些图像 e_1 、 e_2 、 e_3 彼此正交。然后我们可以通过组合标准基的图像来获得一般旋转矩阵 R 。

要获得有意义的旋转角度 , 我们必须定义在二维以上操作时 “逆时针”的含义。当我们 “头朝上 , 从末端朝向原点” 看轴时 , 我们使用绕轴的 “逆时针” (平面) 旋转是指绕轴旋转的约定。在关于三个标准基向量的 R3 旋转中 (见图 3.2) :

，因此有三个 (平面)

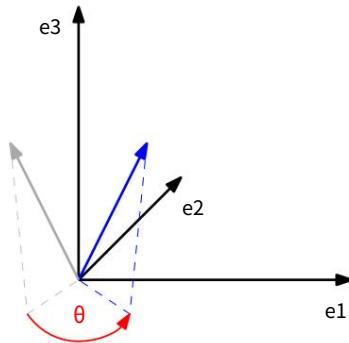


图 3.2 R3中的矢量（灰色）
绕e3 轴旋转角度 θ 。

旋转矢量显示为蓝色。

■ 绕e1 轴旋转

$$R1(\theta) = \Phi(e1) \Phi(e2) \Phi(e3) = \begin{matrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{matrix} \quad (3.77)$$

这里固定e1坐标,在e2e3平面逆时针旋转。

■ 绕e2 轴旋转

$$R2(\theta) = \begin{matrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{matrix} \quad (3.78)$$

如果我们围绕e2轴旋转e1e3平面,我们需要从其“尖端”向原点观察e2轴。

■ 绕e3 轴旋转

$$R3(\theta) = \begin{matrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{matrix} \quad (3.79)$$

图 3.2 说明了这一点。

3.9.3 n维旋转

从 2D 和 3D 到n 维欧几里得向量空间的旋转推广可以直观地描述为固定n-2维
并将旋转限制在n 维空间中的二维平面上。与三维情况一样,我们可以旋转任何
平面 (R_n 的二维子空间) 。

定义 3.11 (给定旋转)。令 V 为n 维欧几里得向量空间,且 $\Phi : V \rightarrow V$ 为具有变
换 ma- 的自同构

三联

$$l_{ij} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$$

$$R_{ij}(\theta) := \begin{pmatrix} 1 & & & & & & \\ & \cos \theta & -\sin \theta & & & & \\ & \sin \theta & \cos \theta & & & & \\ & 0 & 0 & \ddots & & & \\ & & & \cdots & \ddots & & \\ & & & & & 0 & \\ & & & & & & 1 & \end{pmatrix}_{n \times n}, \quad (3.80)$$

吉文斯旋转

对于 $1 \leq i < j \leq n$ 和 $\theta \in \mathbb{R}$, 那么 $R_{ij}(\theta)$ 称为 Givens 旋转。本质上, $R_{ij}(\theta)$ 是单位矩阵 I_n

$$r_{ii} = \cos \theta, r_{ij} = -\sin \theta, r_{ji} = \sin \theta, r_{jj} = \cos \theta. \quad (3.81)$$

在二维 (即 $n = 2$) 中, 我们得到 (3.76) 作为特例。

3.9.4 旋转的性质

旋转展示了许多有用的属性, 可以通过将它们视为正交矩阵 (定义 3.8) 来导出这些属性:

- 旋转保持距离, 即 $\|x - y\| = \|R\theta(x) - R\theta(y)\|$ 。换句话说, 旋转在变换后保持任意两点之间的距离不变。
- 旋转保留角度, 即 $R\theta x$ 和 $R\theta y$ 之间的角度等于 x 和 y 之间的角度。
- 三个 (或更多) 维度的旋转通常不可交换。因此, 应用旋转的顺序很重要, 即使它们围绕同一点旋转。只有在二维向量旋转是可交换的, 使得 $R(\phi)R(\theta) = R(\theta)R(\phi)$ 对于所有 $\phi, \theta \in [0, 2\pi]$ 。仅当它们围绕同一点 (例如, 原点) 旋转时, 它们才形成阿贝尔群 (具有乘法)。

3.10 延伸阅读

在本章中, 我们简要概述了解析几何的一些重要概念, 我们将在本书后面的章节中使用这些概念。

为了更广泛、更深入地概述我们提出的一些概念, 我们参考了以下优秀书籍: Axler (2015) 以及 Boyd 和 Vandenberghe (2018)。

内积允许我们使用 Gram-Schmidt 方法确定向量 (子) 空间的特定基, 其中每个向量与所有其他向量 (正交基) 正交。这些基础在求解线性方程系统的优化和数值算法中很重要。例如, Krylov 子空间方法, 例如共轭梯度或广义最小残差法 (GMRES), 可最大限度地减少彼此正交或正交的残差 (Stoer 和 Burlirsch, 2002 年)。

在机器学习中, 内积在以下情况下很重要

3.10 延伸阅读

95

核方法 (Scholkopf 和 Smola 事实上,许多线, 2002)。内核方法利用性算法可以纯粹通过内积计算来表达。然后, “核技巧”允许我们在 (可能是无限维的) 特征空间中隐含地计算这些内积,甚至不知道这个特征空间明确。这允许机器学习中使用的许多算法的“非线性化”,例如用于降维的内核 PCA (Scholkopf 等人,1997 年)。高斯过程 (Rasmussen 和 Williams,2006 年) 也属于核方法的类别,是概率回归 (将曲线拟合到数据点) 的最新技术。第 12 章将进一步探讨核的概念。

,

投影通常用于计算机图形,例如,生成阴影。在优化中,正交投影通常用于 (迭代地) 最小化残差。这在机器学习中也有应用,例如,在线性回归中,我们想要找到一个 (线性) 函数来最小化残差,即数据在线性函数上的正交投影的长度 (Bishop,2006)。我们将在第 9 章对此进行进一步研究。PCA (Pearson,1901 年;Hotelling,1933 年) 也使用投影来降低高维数据的维数。

我们将在第 10 章中更详细地讨论这个问题。

练习

3.1 证明对所有 $x = [x_1, x_2]$ 定义了 \cdot, \cdot $\in \mathbb{R}^2$ 和 $y = [y_1, y_2]$ $\in \mathbb{R}^2$ 通过

$$x, y := x_1 y_1 - (x_1 y_2 + x_2 y_1) + 2(x_2 y_2)$$

是内积。

3.2 考虑 \mathbb{R}^2 , 为 \mathbb{R}^2 中的所有 x 和 y 定义 \cdot, \cdot 为

$$\begin{aligned} y &:= x_1 2 & 20 && x, \\ && & \text{是的。} \\ & & \overline{-} & - \\ & & = & - \end{aligned}$$

\cdot, \cdot 是内积吗?

3.3 计算之间的距离

$$\begin{array}{ccc} & \scriptstyle 1\wedge & -1 \\ x = & \scriptstyle 2\wedge & -1 \\ & \scriptstyle 3\wedge & , y = 0 \end{array}$$

用—

$$\text{个。 } x, y := x - y$$

$$\begin{array}{ccc} & \scriptstyle 2\wedge & \scriptstyle 1\wedge & 0 \\ b. & \scriptstyle 3\wedge & \scriptstyle 1\wedge & -1 \\ & 0 - 1 & 2 \end{array}$$

3.4 计算夹角

$$\begin{array}{ccc} & \scriptstyle 1\wedge & -1 \\ x = & \scriptstyle 2\wedge & -1 \\ & \scriptstyle 3\wedge & , y = \end{array}$$

用—

$$\text{个。 } x, y := x - y b.$$

$$\begin{array}{ccc} & \scriptstyle 2\wedge & 2 \\ x, y := x - By, & \scriptstyle 1\wedge & 13 \end{array}$$

3.5 考虑点积的欧几里得向量空间 \mathbb{R}^5 。一个子空间

$U \subseteq \mathbb{R}^5$ 和 $x \in \mathbb{R}^5$ 由下式给出

$$\begin{array}{cccccc} 0 & \scriptstyle 1\wedge & -3 & -1 & & -1 \\ & \scriptstyle 2\wedge & -3 & 4\wedge & & -9 \\ U = \text{跨度}[& , & , & , & , &] & , & x = & . \\ & \scriptstyle 2\wedge & -1 & 2\wedge & 0 & 4\wedge \\ & 0 & & & 7 & 1\wedge \end{array}$$

A. 确定 x 在 U 上的正交投影 $U(x)$ 。确定距离 $d(x, U)$

3.6 考虑带内积的 \mathbb{R}^3

$$\begin{array}{ccc} & \scriptstyle 2\wedge & 0 \\ x, y := x & \scriptstyle 1\wedge & 2 - 1 \\ & \scriptstyle 0 - 1 & 2 & y. \end{array}$$

此外, 我们将 e_1 、 e_2 、 e_3 定义为 \mathbb{R}^3 中的标准/规范基础。

A. 确定 e_2 的正交投影 $\pi_U(e_2)$ 到

$$U = \text{跨度 } [e_1, e_3]。$$

提示: 正交性是通过内积定义的。 b. 计算距离 $d(e_2, U)$ 。 C. 绘制场景: 标准基向量和 $\pi_U(e_2)$

3.7 令 V 为向量空间, π 为 V 的自同态。

A. 证明 π 是一个投影当且仅当 $\text{id}_V - \pi$ 是一个投影, 其中 id_V 是 V 上的身份自同态。

b. 现在假设 π 是一个投影。计算 $\text{Im}(\text{id}_V - \pi)$ 和 $\text{ker}(\text{id}_V - \pi)$ 作为 $\text{Im}(\pi)$ 和 $\text{ker}(\pi)$ 的函数。

3.8 使用 Gram-Schmidt 方法, 将二维子空间 $U \subseteq \mathbb{R}^3$ 的基 $B = (b_1, b_2) = (b_1, b_2)$ 转化为 U 的 ONB $C = (c_1, c_2)$, 其中

$$\begin{aligned} b_1 := & \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad b_2 := \begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix}, \\ & , \quad \end{aligned}$$

3.9 令 $n \in \mathbb{N}$ 并令 $x_1, \dots, x_n > 0$ 为 n 个正实数, 使得 $x_1 + \dots + x_n = 1$. 使用 Cauchy-Schwarz 不等式并证明

$$\begin{aligned} A. \quad & \frac{n}{n} \sum_{i=1}^n x_i^2 \leq 1 \\ b. \quad & \sqrt{\sum_{i=1}^n x_i^2} \leq n \end{aligned}$$

提示: 考虑 \mathbb{R}^n 上的点积。然后, 选择特定向量 $x, y \in \mathbb{R}^n$ 并应用 Cauchy-Schwarz 不等式。

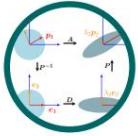
3.10 旋转向量

$$x_1 := \begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix}, \quad x_2 := \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}$$

30° 。

4个

矩阵分解



在第 2 章和第 3 章中,我们研究了操纵和测量向量、向量投影和线性映射的方法。向量的映射和变换可以方便地描述为矩阵执行的操作。此外,数据通常也以矩阵形式表示,例如,矩阵的行代表不同的人,而列描述人的不同特征,例如体重、身高和社会经济地位。在本章中,我们介绍了矩阵的三个方面:如何总结矩阵,如何分解矩阵,以及如何将这些分解用于矩阵逼近。

我们首先考虑允许我们仅使用几个表征矩阵整体属性的数字来描述矩阵的方法。我们将在线性代数 (第 4.1 节) 和特征值 (第 4.2 节) 部分针对重要的方阵特殊情况进行此操作。这些特征数具有重要的数学后果,使我们能够快速掌握矩阵具有哪些有用的属性。从这里我们将进行矩阵分解的方法: 矩阵分解类比为数的因式分解,例如将 21 分解为素数 7·3。因此矩阵分解也就是矩阵分解,通常称为矩阵分解。矩阵分解用于通过使用可解释矩阵的因子的不同表示来描述矩阵。

我们将首先介绍对称正定矩阵的类平方根运算,即 Cholesky 分解 (第 4.3 节)。从这里开始,我们将研究将矩阵因式分解为规范形式的两种相关方法。第一个称为矩阵对角化 (第 4.4 节),如果我们选择合适的基,它允许我们使用对角变换矩阵来表示线性映射。第二种方法,奇异值分解 (第 4.5 节),将这种因式分解扩展到非方矩阵,它被认为是线性代数中的基本概念之一。这些分解很有用,因为表示数字数据的矩阵通常非常大且难以分析。我们以矩阵类型的系统概述和以矩阵分类法的形式区分它们的特征 (第 4.7 节) 来结束本章。

我们在本章中介绍的方法将变得很重要

98

该材料由剑桥大学出版社出版,名为 Marc Peter Deisenroth、A. Aldo Faisal 和 Cheng Soon Ong 的机器学习数学 (2020)。此版本可免费查看和下载,仅供个人使用,不得重新分发、转售或用于衍生作品。© MP Deisenroth、AA Faisal 和 CS Ong, 2021 年。<https://mml-book.com>

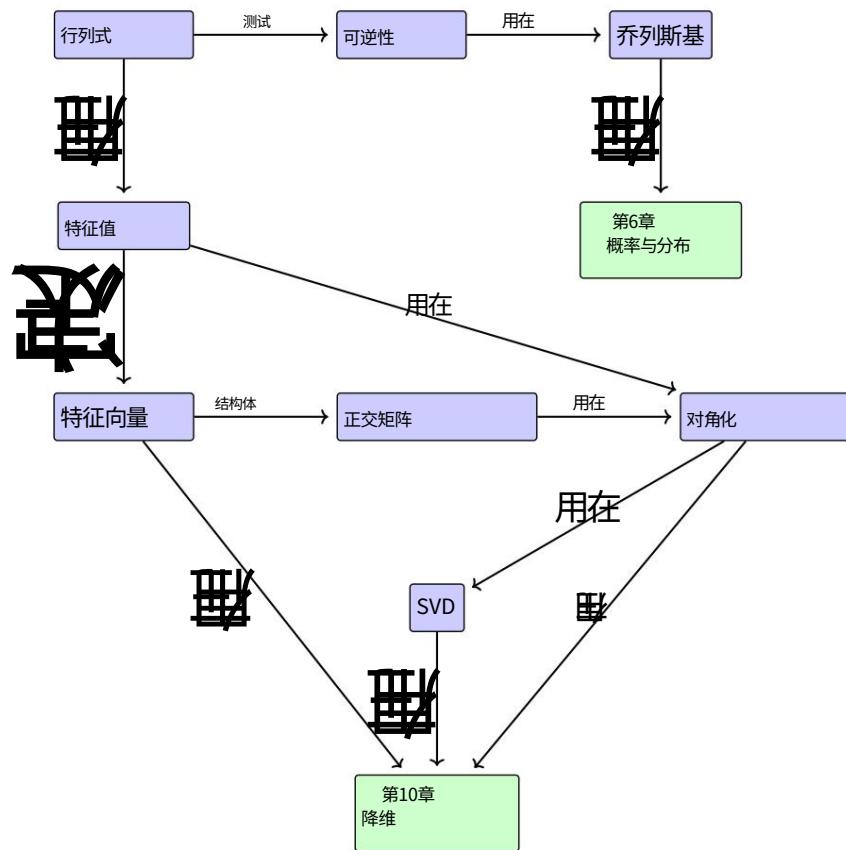


图 4.2 本章介绍的概念及其在本书其他部分中的使用位置的思维导图。

包括后续的数学章节,例如第 6 章,以及应用章节,例如第 10 章的降维或第 11 章的密度估计。
本章的整体结构如图 4.2 的思维导图所示。

4.1 行列式和迹

行列式是线性代数中的重要概念。行列式是线性方程组分析和求解中的数学对象。行列式仅对方阵 $A \in \mathbb{R}^{n \times n}$, 即具有相同行数和列数的矩阵定义。在本书中,我们将行列式写为 $\det(A)$ 或有时写为 $|A|$ 以便

行列式
符号 $|A|$ 不能混淆
与绝对值。

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} \quad (4.1)$$

方阵 $A \in \mathbb{R}^{n \times n}$ 的行列式是映射 A 行列式的函数

到一个实数上。在为一般 $n \times n$ 矩阵提供行列式的定义之前,让我们先看一些有启发性的例子,并为一些特殊矩阵定义行列式。

例 4.1 (矩阵可逆性检验)

让我们从探索方阵 A 是否可逆开始 (参见第 2.2.2 节)。对于最小的情况,我们已经知道矩阵何时是可逆的。若 A 为 1×1 矩阵,即为标量数,则 $A = a \Rightarrow A^{-1} = \frac{1}{a}$ 。

对于 2×2 矩阵,根据逆的定义 (定义 2.3),我们知道 $AA^{-1} = I_2$ 。然后,根据 (2.24), A 的倒数是

$$A^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}. \quad (4.2)$$

因此, A 是可逆的当且仅当

$$a_{11}a_{22} - a_{12}a_{21} \neq 0. \quad (4.3)$$

这个量是 $A \in \mathbb{R}^{2 \times 2}$ 的行列式,即

$$\det(A) = \frac{a_{11}a_{22} - a_{12}a_{21}}{a_{21}a_{22}} = a_{11}a_{22} - a_{12}a_{21}. \quad (4.4)$$

示例 4.1 已经指出了行列式与逆矩阵的存在性之间的关系。下一个定理陈述了 $n \times n$ 矩阵的相同结果。

定理 4.1。 对于任何方阵 $A \in \mathbb{R}^{n \times n}$, 当且仅当 $\det(A) \neq 0$ 时, A 是可逆的。

对于小的决定因素,我们有明确的 (封闭形式) 表达式矩阵的元素方面的矩阵。对于 $n = 1$,

$$\det(A) = \det(a_{11}) = a_{11}. \quad (4.5)$$

对于 $n = 2$,

$$\det(A) = \frac{a_{11}a_{22} - a_{12}a_{21}}{a_{21}a_{22}} = a_{11}a_{22} - a_{12}a_{21}, \quad (4.6)$$

我们在前面的例子中观察到了。

对于 $n = 3$ (称为 Sarrus 规则),

$$\begin{aligned} & a_{11}a_{12}a_{13} \\ & a_{21}a_{22}a_{23} = a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} \\ & a_{31}a_{32}a_{33} \end{aligned} \quad (4.7)$$

$$- a_{31}a_{22}a_{13} - a_{11}a_{32}a_{23} - a_{21}a_{12}a_{33}.$$

为了帮助记忆 Sarrus 规则中的乘积项,请尝试追踪矩阵中三重乘积的元素。

如果 $T_{ij} = 0$ 对于上三角矩阵,我们称方阵 T 为上三角矩阵
 $i > j$,即矩阵在其对角线下方为零。类似地,我们将下三角矩阵定义为其对角线上有零的矩阵。
对于三下三角矩阵 $T \in R^{n \times n}$ 个元素,即
, 行列式是对角线的乘积

$$\det(T) = \sum_{\text{我}=1}^n \text{。} \quad (4.8)$$

示例 4.2 (行列式作为体积的度量)

当我们把行列式视为来自跨越 R^n 中对象的一组 n 个向量的映射时,行列式的概念很自然。
事实证明,行列式 $\det(A)$ 是由矩阵 A 的列构成的 n 维平行六面体的带符号体积。

对于 $n = 2$,矩阵的列形成一个平行四边形;见图 4.3。随着向量之间的角度变小,平行四边形的面积也会缩小。考虑构成矩阵 $A = [b, g]$ 的列的两个向量 b, g 。那么, A 的行列式的绝对值就是顶点为 $0, b, g, b+g$ 的平行四边形的面积。特别是,如果 b, g 是线性相关的,使得对于某些 $\lambda \in R$, $b = \lambda g$,它们不再形成二维平行四边形。因此,对应的面积为 0。反之,若 b, g 线性无关且为

规范基向量 e_1, e_2 那么它们可以写成 $b =$

$$g = \begin{matrix} 0 \\ G \end{matrix}, \text{决定因素是 } \begin{matrix} b & 0 \\ 0 & g \end{matrix} \text{ 行} = \text{背景} - 0 = \text{背景}.$$

行列式的符号表示生成向量 b, g 相对于标准基 (e_1, e_2) 的方向。在我们的图中,翻转 ping 到 g 的顺序, b 交换 A 的列并反转阴影区域的方向。这变成了熟悉的公式:面积=高度 \times 长度。这种直觉延伸到更高的维度。在 R^3 中,三个向量 $r, b, g \in R^3$ 跨越平行六面体的边,即具有平行四边形面的实体(见图 4.4)。ab-3 $\times 3$ 矩阵 $[r, b, g]$ 的行列式的溶质值的符号为固体的体积。因此,行列式用作测量由矩阵中组成的列,我们认为向量形成的带符号体积的函数。

考虑三个线性无关的向量 $r, g, b \in R^3$ 给定为

$$r = \begin{matrix} 2 \\ 0 \\ -8 \end{matrix}, \quad g = \begin{matrix} 6 \\ 1 \\ 0 \end{matrix}, \quad b = \begin{matrix} 1 \\ 4 \\ -1 \end{matrix}. \quad (4.9)$$

行列式是
由列的列形成的平行六面
体的符号体积

矩阵。

图 4.3 向量 b 和 g 所跨越的平
行四边形(阴影区域)的面
积为 $|\det([b, g])|$ 。

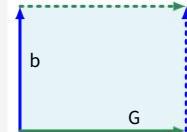
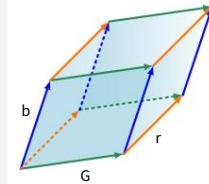


图 4.4 由向量 r 、 b 、
 g 跨越的平行六面
体(阴影体积)的体
积为 $|\det([r, b, g])|$ 。



行列式表示的方
向

跨越向量。

将这些向量写成矩阵的列

$$A = [r, g, b] = \begin{matrix} 2 & 6 & 1 \\ 0 & 1 & 4 \\ -8 & 0 & -1 \end{matrix} \quad (4.10)$$

允许我们将所需的体积计算为

$$V = |\det(A)| = 186。 \quad (4.11)$$

计算 $n \times n$ 矩阵的行列式需要一个通用算法来解决 $n > 3$ 的情况，我们将在下面探讨。下面的定理 4.2 将计算 $n \times n$ 矩阵的行列式的问题简化为计算 $(n-1) \times (n-1)$ 矩阵的行列式。通过递归地应用拉普拉斯展开（定理 4.2），我们因此可以通过最终计算 2×2 矩阵的行列式来计算 $n \times n$ 矩阵的行列式。

拉普拉斯展开

定理 4.2（拉普拉斯展开）。考虑一个矩阵 $A \in R^{n \times n}$ 。然后，对于所有 $j = 1, \dots, n$ ，

名词：

$\det(A_{k,j})$ 称为小调，并且

$(-1)^{k+j} \det(A_{k,j})$ 一个辅助因子。

1. 沿 j 列展开

$$\det(A) = \sum_{k=1}^n (-1)^{k+j} a_{kj} \det(A_{k,j}) \quad (4.12)$$

2. 沿第 j 行扩展

$$\det(A) = \sum_{k=1}^n (-1)^{k+j} a_{jk} \det(A_{j,k}) \quad (4.13)$$

这里 $A_{k,j} \in R^{(n-1) \times (n-1)}$ 是我们在删除 k 行和 j 列时得到的 A 的子矩阵。

例 4.3（拉普拉斯展开）

让我们计算的行列式

$$\begin{matrix} 1 & 2 & 3 \\ \text{一个} = & 3 & 1 & 2 \\ & 0 & 0 & 1 \end{matrix} \quad (4.14)$$

沿第一行使用拉普拉斯展开。应用 (4.13) 产生

$$\begin{aligned} \begin{matrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 1 \end{matrix} &= (-1)^{1+1} \cdot \begin{matrix} 1 & 2 \\ 1 & 0 & 1 \end{matrix} \\ &+ (-1)^{1+2} \cdot 2 \begin{matrix} 3 & 2 \\ 0 & 1 \end{matrix} + (-1)^{1+3} \cdot 3 \begin{matrix} 3 & 1 \\ 0 & 0 \end{matrix} \end{aligned} \quad (4.15)$$

我们使用 (4.6) 来计算所有 2×2 矩阵的行列式并获得

$$\det(A) = 1(1 - 0) - 2(3 - 0) + 3(0 - 0) = -5。 \quad (4.16)$$

为了完整起见,我们可以将此结果与使用 Sarrus 规则 (4.7) 计算行列式进行比较:

$$\det(A) = 1 \cdot 1 \cdot 1 + 3 \cdot 0 \cdot 3 + 0 \cdot 2 \cdot 2 - 0 \cdot 1 \cdot 3 - 1 \cdot 0 \cdot 2 - 3 \cdot 2 \cdot 1 = 1 - 6 = -5。 \quad (4.17)$$

对于 $A \in \mathbb{R}^{n \times n}$, 行列式具有以下性质:

- 矩阵乘积的行列式是相应行列式的乘积, $\det(AB) = \det(A)\det(B)$ 。
- 行列式对于转置是不变的,即 $\det(A) = \det(A)$ 如果 A 是正则的 (可逆的), 则 $\det(A^{-1}) = \frac{\det(A)}{\det(A)}$ 。
- 行列式对于相似的矩阵 (定义 2.22) 具有相同的行 $\det(A\Phi) = \det(\Phi)$ 。
- 行列式因此,对于线性映射 $\Phi : V \rightarrow V$ 所有的变换矩阵 $A\Phi$ 具有相同的行列式, 因此行列式对于线性映射的基的选择是不变的。
- 将一列/行的倍数添加到另一列/行不会更改 $\det(A)$ 。
- 列/行与 $\lambda \in \mathbb{R}$ 的乘积按 λ 缩放 $\det(A)$ 。特别地, $\det(\lambda A) = \lambda^n \det(A)$ 。
- 交换两行/两列会改变 $\det(A)$ 的符号。

由于最后三个性质,我们可以使用高斯消元法 (参见第 2.1 节) 通过将 A 带入行阶梯形式来计算 $\det(A)$ 。

当 A 为三角形,其中对角线以下的元素均为 0 时,我们可以停止高斯消元法。回想一下 (4.8), 三角矩阵的行列式是对角线元素的乘积。

定理 4.3。 方阵 $A \in \mathbb{R}^{n \times n}$ 有 $\det(A) = 0$ 当且仅当 $\text{rk}(A) < n$ 。换句话说, A 是可逆的当且仅当它是满秩的。

当数学主要由手工完成时,行列式计算被认为是分析矩阵可逆性的基本方法。然而,现代机器学习方法使用直接数值方法取代了行列式的显式计算。例如,在第 2 章中,我们了解到可以通过高斯消元法计算逆矩阵。因此,高斯消去法可用于计算矩阵的行列式。

行列式将在接下来的章节中发挥重要的理论作用,尤其是当我们通过特征多项式了解特征值和特征向量 (第 4.2 节) 时。

定义 4.4。 方阵 $A \in \mathbb{R}^{n \times n}$ 的迹定义为

痕迹

$$\text{tr}(A) := \sum_{i=1}^n \text{爱}_i, \quad (4.18)$$

IE_n, 迹是A的对角线元素之和。

跟踪满足以下属性：

- $\text{tr}(A + B) = \text{tr}(A) + \text{tr}(B)$ 对于 $A, B \in \mathbb{R}^{n \times n}$
- $\text{tr}(\alpha A) = \alpha \text{tr}(A)$, $\alpha \in \mathbb{R}$ 对于 $A \in \mathbb{R}^{n \times n}$
- $\text{tr}(I_n) = n$
- $\text{tr}(AB) = \text{tr}(BA)$ 对于 $A \in \mathbb{R}^{n \times k}, B \in \mathbb{R}^{k \times n}$

可以证明只有一个函数同时满足这四个属性 轨迹 (Gohberg et al., 2012)。

微量矩阵乘积的性质更为普遍。具体
理论上,轨迹在循环排列下是不变的,即

$$\text{tr}(AKL) = \text{tr}(KLA) \quad (4.19)$$

对于矩阵 $A \in \mathbb{R}^{a \times k}, K \in \mathbb{R}^{k \times l}, L \in \mathbb{R}^{l \times a}$ 。此属性推广到任意数量矩阵的乘积。作为 (4.19) 的特例,对于两个向量 $x, y \in \mathbb{R}^n$ $\text{tr}(xy^\top) = \text{tr}(y^\top x) = y^\top x \in \mathbb{R}$ 。

$$(4.20)$$

给定线性映射 $\Phi : V \rightarrow V$, 其中 V 是向量空间, 我们使用 Φ 的矩阵表示的迹来定义该映射的迹。对于给定的 V 矩阵 A 的基, 则 Φ 的迹就是 A 的迹。对于 V 的不同基, 我们可以通过变换来描述 Φ

, 它认为 Φ 的相应变换矩阵 B 可以通过 $S - 1AS$ 形式的基础变化为合适的 S 获得 (见第 2.7.2 节)。对于 Φ 的相应轨迹, 这意味着

$$\text{tr}(B) = \text{tr}(S - 1AS) \stackrel{(4.19)}{=} \text{tr}(ASS - 1) = \text{tr}(A). \quad (4.21)$$

因此, 虽然线性映射的矩阵表示依赖于基, 但线性映射 Φ 的迹与基无关。

在本节中, 我们介绍了行列式和迹作为表征方阵的函数。结合我们对行列式和迹的理解, 我们现在可以定义一个用多项式描述矩阵 A 的重要方程, 我们将在以下各节中广泛使用它。

定义 4.5 (特征多项式)。对于 $\lambda \in \mathbb{R}$ 和方阵 $A \in \mathbb{R}^{n \times n}$

$$p_A(\lambda) := \det(A - \lambda I) = c_0 + c_1 \lambda \quad (4.22a)$$

$$+ c_2 \lambda^2 + \dots + c_{n-1} \lambda^{n-1} + (-1)^n \lambda^n, \quad (4.22b)$$

特征多项式 $c_0, \dots, c_{n-1} \in \mathbb{R}$, 是 A 的特征多项式。特别地,

$$c_0 = \det(A), \quad (4.23)$$

$$cn - 1 = (-1)^{n-1} \operatorname{tr}(A). \quad (4.24)$$

特征多项式 (4.22a) 将允许我们计算特征值和特征向量,这将在下一节中介绍。

4.2 特征值和特征向量我们现在将了解一种

表示矩阵及其相关线性映射的新方法。回想一下 2.7.1 节,每个线性映射都有一个给定有序基的唯一变换矩阵。我们可以通过执行“特征”分析来解释线性映射及其相关的变换矩阵。正如我们将看到的,线性特征值是一个德国耳朵映射将告诉我们一组特殊的向量,即特征向量,是如何通过线性映射进行转换的。

意思是“特征”,“自我”或“自己”的词。

定义 4.6。令 $A \in \mathbb{R}^{n \times n}$ 为方阵。那么 $\lambda \in \mathbb{R}$ 是 A 的一个特征值并且 $x \in \mathbb{R}^n \setminus \{0\}$ 是 A 的相应特征向量如果

$$\text{轴} = \lambda x. \quad (4.25)$$

我们称 (4.25) 为特征值方程。

特征值特征向

量

特征值方程

评论。在线性代数文献和软件中,通常约定特征值按降序排列,因此最大的特征值和关联的特征向量称为第一特征值及其关联的特征向量,第二大的称为第二特征值和其关联的特征向量,等等。然而,教科书和出版物可能有不同的顺序概念或没有。如果没有明确说明,我们不想假定本书中的顺序。 ◇ 以下语句是等价的:

- λ 是 $A \in \mathbb{R}^{n \times n}$ 的特征值。
- 存在一个 $x \in \mathbb{R}^n \setminus \{0\}$ 且 $Ax = \lambda x$, 或者等价地, $(A - \lambda I_n)x = 0$ 可以非平凡地求解, 即 $x = 0$ 。
 $\operatorname{rk}(A - \lambda I_n) < n$ 。 $\det(A - \lambda I_n) = 0$ 。
-
-

定义 4.7 (共线性和共向)。指向同一方向的两个向量称为同向向量。如果两个向量共同指向相同或相反的方向,则它们是共线的。

共线

备注 (特征向量的非唯一性)。如果 x 是与特征值 λ 关联的 A 的特征向量, 则对于任何 $c \in \mathbb{R} \setminus \{0\}$, 它认为 cx 是 A 的特征向量, 具有相同的特征值, 因为

$$A(cx) = cAx = c\lambda x = \lambda(cx). \quad (4.26)$$

因此, 与 x 共线的所有向量也是 A 的特征向量。

◇

定理 4.8。 $\lambda \in \mathbb{R}$ 是 $A \in \mathbb{R}^{n \times n}$ 的特征值 当且仅当 λ 是 A 的特征多项式 $p_A(\lambda)$ 的根。

代数重数

定义 4.9。设方阵 A 具有特征值 λ_i 。 λ_i 的代数重数是根在特征多项式中出现的次数。

本征空间

定义 4.10 (本征空间和本征谱)。对于 $A \in \mathbb{R}^{n \times n}$, 与特征值 λ 相关联的 A 的所有特征 , 集合向量跨越 \mathbb{R}^n 的一个子空间, 该子空间称为 A 关于 λ 的特征空间, 记为 E_λ 。 A 的所有特征值的集合称为 A 的特征谱, 或简称为谱。

谱

光谱

如果 λ 是 $A \in \mathbb{R}^{n \times n}$ 的特征值, 则对应的特征空间 E_λ 是齐次线性方程组 $(A - \lambda I)x = 0$ 的解空间。在几何上, 对应于非零特征值的特征向量指向一个方向被线性映射拉伸。

特征值是它被拉伸的因子。如果特征值为负, 则拉伸方向反转。

例 4.4 (单位矩阵的情况)

单位矩阵 $I \in \mathbb{R}^{n \times n}$ 具有特征多项式 $p_I(\lambda) = \det(I - \lambda I) = (1 - \lambda)^n = 0$, 只有一个特征值 $\lambda = 1$ 出现 n 次。此外, $Ix = \lambda x = x$ 对所有向量 $x \in \mathbb{R}^n \setminus \{0\}$ 成立。

因此, 单位矩阵的唯一特征空间 E_1 跨越 n 维, \mathbb{R}^n 的所有 n 个标准基向量都是 I 的特征向量。

有关特征值和特征向量的有用属性包括:

- 矩阵 A 及其转置 A^T 具有相同的特征值, 但不一定具有相同的特征向量。

- 本征空间 E_λ 是 $A - \lambda I$ 的零空间, 因为

$$Ax = \lambda x \iff Ax - \lambda x = 0 \quad (4.27a)$$

$$\iff (A - \lambda I)x = 0 \iff x \in \ker(A - \lambda I) \quad (4.27b)$$

- 相似的矩阵 (见定义 2.22) 具有相同的特征值。

因此, 线性映射中具有与其变换矩阵的基选择无关的特征值。这使得特征值与行列式和迹一起成为线性映射的关键特征参数, 因为它们在基础变化下都是不变的。

- 对称的正定矩阵总是具有正的实数特征值。

例 4.5 (计算特征值、特征向量和特征空间)

让我们找到 2×2 矩阵的特征值和特征向量

$$\text{一个} = \begin{matrix} 4 & 2 \\ 1 & 3 \end{matrix} \quad (4.28)$$

第 1 步 : 特征多项式。根据我们对特征向量 $x = 0$ 和 A 的特征值 λ 的定义, 将有一个向量使得 $Ax = \lambda x$, 即 $(A - \lambda I)x = 0$ 。由于 $x = 0$, 这需要内核 $A - \lambda I$ 的 (零空间) 包含的元素多于 0。这意味着 $A - \lambda I$ 不可逆, 因此 $\det(A - \lambda I) = 0$ 。因此, 我们需要计算特征多项式 (4.22a) 的根) 找到特征值。

第 2 步 : 特征值。特征多项式是

$$p_A(\lambda) = \det(A - \lambda I) \quad (4.29a)$$

$$= \det \begin{matrix} \lambda & 0 & 0 \\ - & \lambda & \\ & 2 & 3 - \lambda \end{matrix} = \frac{(4 - \lambda)(1 - \lambda)}{(2 - \lambda)} \quad (4.29b)$$

$$= (4 - \lambda)(3 - \lambda) - 2 \cdot 1. \quad (4.29c)$$

我们分解特征多项式并获得

$$p(\lambda) = (4 - \lambda)(3 - \lambda) - 2 \cdot 1 = 10 - 7\lambda + \lambda^2 = (2 - \lambda)(5 - \lambda) \quad (4.30)$$

给出根 $\lambda_1 = 2$ 和 $\lambda_2 = 5$ 。

第 3 步 : 特征向量和特征空间。我们找到特征向量
通过查看向量 x 对应于这些特征值, 使得

$$\begin{matrix} 4 - \lambda \\ \downarrow \\ 2 & 3 - \lambda \end{matrix} \quad x = 0. \quad (4.31)$$

对于 $\lambda = 5$, 我们得到

$$\begin{matrix} 4 - 5 & 2 & 1 & 3 - 5 \\ x_1 & = & -1 & 2 \\ x_2 & & 1 & -2 \end{matrix} \quad \begin{matrix} x_1 \\ x_2 \end{matrix} = 0. \quad (4.32)$$

我们解决这个齐次系统并获得解决方案空间

$$E5 = \text{跨度} \left[\begin{matrix} 2 \\ 1 \end{matrix} \right]. \quad (4.33)$$

该特征空间是一维的, 因为它具有单个基向量。

类似地, 我们通过求解齐次方程组找到 $\lambda = 2$ 的特征向量

$$\begin{matrix} 4 - 2 & & & 2 & 2 \\ \downarrow & & & 1 & 1 \end{matrix} \quad x = \begin{matrix} 2 \\ 1 \end{matrix} \quad x = 0. \quad (4.34)$$

这意味着任何向量 $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, 其中 $x_2 = -x_1$, 例如 $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$, 是一个特征值为 2 的特征向量。相应的特征空间为

$$E_2 = \text{跨度}[\begin{pmatrix} 1 \\ -1 \end{pmatrix}]。 \quad (4.35)$$

示例 4.5 中的两个特征空间 E_5 和 E_2 是一维的, 因为它们每个都由一个向量跨越。然而, 在其他情况下, 我们可能有多个相同的特征值 (见定义 4.9) 并且特征空间可能有不止一维。

几何多重性

定义 4.11。令 λ_i 为方阵 A 的特征值。则 λ_i 的几何重数是与 λ_i 相关联的线性独立特征向量的数量。

换句话说, 它是由与 λ_i 关联的特征向量所跨越的特征空间的维数。

评论。特定特征值的几何重数必须至少为 1, 因为每个特征值至少有一个关联的特征向量。特征值的几何重数不能超过其代数重数, 但可以更低。 ◇

例 4.6

矩阵 $A = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}$ 有两个重复的特征值 $\lambda_1 = \lambda_2 = 2$ 和

的代数重数。然而, 特征值只有一个不同的

单位特征向量 $x_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ 因此, 几何多重性 1。

二维图形直觉让我们使用不同的线性映射获得行列

式、特征向量和特征值的一些直觉。图 4.2 描绘了五个变换矩阵 A_1, \dots, A_5 及其对以原点为中心的方格点的影响:

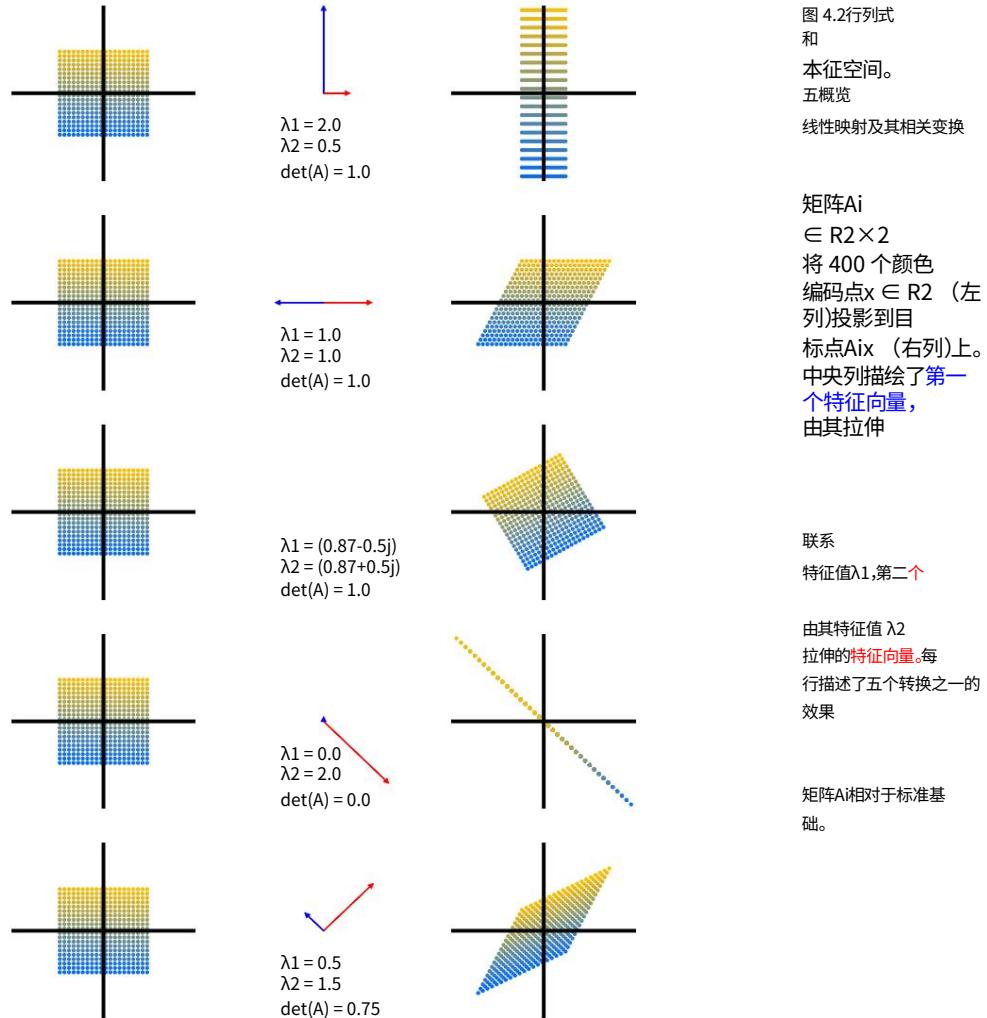
在几何学中, 这种平行于轴的剪切的面积保持特性也被称为平行四边形的卡瓦列里等面积原理 (Katz, 2004)。

- $A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$. 两个特征向量的方向对应于

R^2 中的规范基向量, 即到两个基轴。垂直轴扩展 2 倍 (特征值 $\lambda_1 = 2$), 水平轴压缩 2 倍 (特征值 $\lambda_2 = 1$)。映射是区域保留 ($\det(A_1) = 1 = 2 \cdot 1 \neq 0$)。

$$\begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}.$$

- $A_2 = \begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix}$ 对应于一个剪切映射, 即, 如果它们在正方向上, 它将沿着水平轴向右剪切点



垂直轴的一半,反之亦然。此映射保留区域($\det(A_2) = 1$)。重复特征值 $\lambda_1 = 1 = \lambda_2$ 并且特征向量共线(此处绘制是为了强调两个相反的方向)。这表明映射仅沿一个方向(水平轴)起作用。 $\sqrt{3} - 1 \cos(\pi) - \sin(\pi) \sqrt{3}$

- $A_3 = \begin{pmatrix} \cos(\pi) & -\sin(\pi) \\ \sin(\pi) & \cos(\pi) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ — 矩阵 A_3 旋转
 点乘 π 值, 反 $\text{rad} = 30^\circ$ 逆时针并且只有复数特征
 映射是一个旋转(因此,没有绘制特征向量)。旋转必须是体积保持的,所以行列式是1。关于旋转的更多细节,我们参考第3.9.1 - 1 - 1 1
- $A_4 = \dots$ 表示标准基础上的映射,即 col
 将二维域移到一维上。由于一个特征-

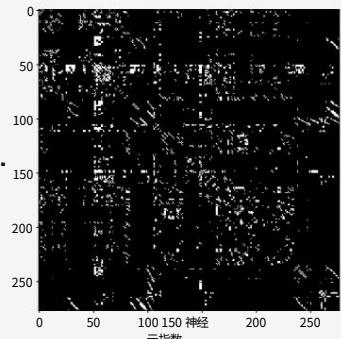
值为0,对应于 $\lambda_1 = 0$ 的 (蓝色) 特征向量方向的空间坍塌,而正交 (红色) 特征向量将空间拉伸因子 $\lambda_2 = 2$ 。因此,图像的面积为0。

■ $A_5 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ 是一种将空间缩放75%的剪切拉伸映射

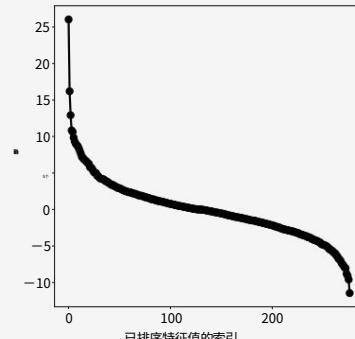
自确定 (A_5) $=$ 它沿 λ_2^2 的 (红色) 特征向量将空间拉伸1.5倍,并沿正交 (蓝色) 特征向量将空间压缩0.5 倍。

例 4.7 (生物神经网络的特征谱)

图 4.3
秀丽
线虫神经网络
(Kaiser and Hilgetag,
2006).(a)
Symmetrized
连接矩阵; (b) 特征谱。



(a) 连接矩阵。



(b) 特征谱。

分析和学习网络数据的方法是机器学习方法的重要组成部分。理解网络的关键是网络节点之间的连通性,尤其是两个节点是否相互连接。在数据科学应用程序中,研究捕获此连接数据的矩阵通常很有用。

我们构建了蠕虫 *C. elegans* 的完整神经网络的连通性/邻接矩阵 $A \in \mathbb{R}^{277 \times 277}$ 。每行/每列代表该蠕虫大脑的277个神经元之一。如果神经元*i*通过突触与神经元*j*对话,则连接矩阵A的值为 $a_{ij} = 1$,否则 $a_{ij} = 0$ 。连接矩阵不是对称的,这意味着特征值可能不是实数。因此,我们计算 a 。这

连接矩阵的对称版本为 $Asym := A + 新矩阵 Asym$ 如图 4.3(a) 所示,当且仅当两个神经元连接 (白色像素) 时具有非零值 a_{ij} ,与连接方向无关。在图 4.3(b) 中,我们显示了 $Asym$ 的相应特征谱。水平轴显示特征值的索引,按降序排列。垂直轴显示相应的特征值。这种特征谱的 S 形是许多生物神经网络的典型特征。造成这种情况的潜在机制是活跃的神经科学研究领域。

定理 4.12。特征向量 x_1, \dots, x_n 矩阵 $A \in \mathbb{R}^{n \times n}$ 具有 n 个不同的特征值 $\lambda_1, \dots, \lambda_n$ 是线性无关的。

该定理指出具有 n 个不同特征值的矩阵的特征向量构成 \mathbb{R}^n 的基。

定义 4.13。如果方阵 $A \in \mathbb{R}^{n \times n}$ 有缺陷的少于 n 个线性独立的特征向量，则它是有缺陷的。

无缺陷矩阵 $A \in \mathbb{R}^{n \times n}$ 不一定需要 n 个不同的特征值，但它确实需要特征向量构成 \mathbb{R}^n 的基。查看有缺陷矩阵的特征空间，可以得出特征空间的维数之和小于 n 。具体来说，一个有缺陷的矩阵至少有一个特征值 λ_i ，其代数重数 $m > 1$ 且几何重数小于 m 。

评论。有缺陷的矩阵不能有 n 个不同的特征值，因为不同的特征值具有线性独立的特征向量（定理 4.12）。 ◇

定理 4.14。给定矩阵 $A \in \mathbb{R}^{m \times n}$ 度量，半正定，我们总能得到一个符号矩阵 $S \in \mathbb{R}^{n \times n}$ 通过定义

$$S := A^\top A. \quad (4.36)$$

评论。如果 $\text{rk}(A) = n$ ，则 $S := A^\top A$ 是对称的，正定的。 ◇ 理解为什么定

理 4.14 成立对于我们如何能够

使用对称矩阵：对称性要求 $S = S$ 并且通过插入 ing (4.36) 我们得到 $S = A^\top A = A = (A^\top A)^\top$ 。以上，半正定性（第 3.2.3 节）要求 $x^\top (AS)x \geq 0$ 和插入 $= \text{小号}$ 。更多的 (4.36) 我们得到 $x^\top Sx = x^\top A^\top Ax = (x^\top A)(Ax) = (Ax)^\top (Ax) \geq 0$ ，因为点积计算平方和（它们本身是非消极的）。

谱定理

定理 4.15（谱定理）。如果 $A \in \mathbb{R}^{n \times n}$ 是对称的，则存在由 A 的特征向量组成的对应向量空间 V 的正交基，且每个特征值都是实数。

谱定理的直接含义是对称矩阵 A 的特征分解存在（具有实特征值），并且我们可以找到特征向量的 ONB 使得 $A = PDP^{-1}$ 其中 D 是对角线并且 P 的列包含特征向量。

例 4.8 考虑矩阵

$$\begin{array}{c} 3 & 2 & 2 \\ \text{一个}= & 2 & 3 & 2 \\ & 2 & 2 & 3 \end{array} \quad . \quad (4.37)$$

A的特征多项式为

$$p_A(\lambda) = -(\lambda - 1)^2 (\lambda - 7), \quad (4.38)$$

从而我们得到特征值 $\lambda_1 = 1$ 和 $\lambda_2 = 7$, 其中 λ_1 是重复的特征值。按照我们计算特征向量的标准程序, 我们获得了特征空间

$$E_1 = \text{跨度} \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, E_7 = \text{跨度} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (4.39)$$

$=:x_1 \quad =:x_2 \quad =:x_3$

我们看到 x_3 与 x_1 和 x_2 都正交。然而, 由于 x

$= 0$, 它们不正交。谱定理 (定理 4.15) 指出存在正交基, 但我们拥有的基不是正交的。但是, 我们可以构建一个。

为了构建这样的基础, 我们利用了 x_1 、 x_2 是与相同特征值 λ 关联的特征向量这一事实。因此, 对于任何 $\alpha, \beta \in \mathbb{R}$, 它认为

$$A(\alpha x_1 + \beta x_2) = Ax_1\alpha + Ax_2\beta = \lambda(\alpha x_1 + \beta x_2), \quad (4.40)$$

即, x_1 和 x_2 的任何线性组合也是与 λ 关联的 A 的特征向量。Gram-Schmidt 算法 (第 3.8.3 节) 是一种使用此类线性组合从一组基向量迭代构建正交/正交基的方法。因此, 即使 x_1 和 x_2 不正交, 我们也可以应用 Gram-Schmidt 算法并找到与 $\lambda_1 = 1$ 相关联且彼此正交 (以及 x_3) 的特征向量。在我们的示例中, 我们将获得

$$x_1 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}, \quad (4.41)$$

它们彼此正交, 与 x_3 正交, 并且 A 的特征向量与 $\lambda_1 = 1$ 相关联。

在我们结束对特征值和特征向量的考虑之前, 将这些矩阵特征与行列式和迹的概念联系在一起是很有用的。

定理 4.16。矩阵 $A \in \mathbb{R}^{n \times n}$ 的行列式是其特征值的乘积, 即,

$$\det(A) = \prod_{i=1}^n \lambda_i, \quad (4.42)$$

其中 $\lambda_i \in \mathbb{C}$ 是 (可能重复的) A 的特征值。

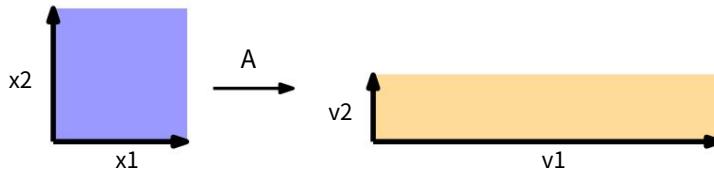


图 4.1 几何

特征值的解释。A 的特征向量被相应的特征值拉伸。单位面积

定理 4.17。矩阵 $A \in R^{n \times n}$ 的迹是其特征值之和,即

$$\text{tr}(A) = \sum_{i=1}^n \lambda_i, \quad (4.43)$$

正方形变化 $|\lambda_1\lambda_2|$, 周长变化因数

$$\frac{1}{12}(|\lambda_1| + |\lambda_2|).$$

其中 $\lambda_i \in C$ 是 (可能重复的) A 的特征值。

让我们提供这两个定理的几何直觉。考虑一个矩阵 $A \in R^{2 \times 2}$, 它具有两个线性独立的特征向量 x_1, x_2 。对于这个例子, 我们假设 (x_1, x_2) 是 R^2 的 ONB, 因此它们是正交的, 并且它们所跨过的正方形的面积是 1; 见图 4.1。

从 4.1 节中, 我们知道行列式计算变换 A 下单位正方形面积的变化。在这个例子中, 我们可以明确地计算面积变化: 使用 A 映射特征向量得到向量 $v_1 = Ax_1 = \lambda_1 x_1$ 和 $v_2 = Ax_2 = \lambda_2 x_2$, 即新向量 v_i 是特征向量 x_i 的缩放版本, 缩放因子是相应的特征值 λ_i 。 v_1, v_2 仍然是正交的, 它们所跨过的矩形的面积是 $|\lambda_1\lambda_2|$ 。

鉴于 x_1, x_2 (在我们的示例中) 是正交的, 我们可以直接计算单位正方形的周长为 $2(1+1)$ 。使用 A 映射特征向量会创建一个周长为 $2(|\lambda_1| + |\lambda_2|)$ 的矩形。

因此, 特征值的绝对值之和告诉我们单位正方形的周长在变换矩阵 A 下如何变化。

示例 4.9 (Google 的 PageRank 网页作为特征向量)

Google 使用对应于矩阵 A 的最大特征值的特征向量来确定搜索页面的排名。拉里佩奇和谢尔盖布林于 1996 年在斯坦福大学开发的 PageRank 算法的想法是, 任何网页的重要性都可以通过链接到它的页面的重要性来近似。为此, 他们将所有网站写成一个巨大的有向图, 显示哪个页面链接到哪个页面。PageRank 通过计算指向 a_i 的页面数量来计算网站 a_i 的权重 (重要性) $x_i > 0$ 。此外, PageRank 会考虑链接到 a_i 的网站的重要性。用户的导航行为然后通过此图的转换矩阵 A 建模, 该矩阵告诉我们某人最终会以什么 (点击) 概率结束

网页排名

在不同的网站上。矩阵A具有以下属性：对于网站的任何初始排名/重要性向量 x ，序列 x, Ax, A^2x, \dots 收敛到向量 x^* 。该向量称为PageRank并满足 $Ax^* = x^*$ ，即它是这样的特征向量（具有相应的特征值1） $\|x^*\| = 1$ ，我们可以更多细节和不同观点可以^{*}，解释条目A。在将 x 归一化为概率之后。有关PageRank的在原始技术报告中找到（Page等人，1999）。

4.3 乔列斯基分解

有很多方法可以分解我们在机器学习中经常遇到的特殊类型的矩阵。在正实数中，我们有平方根运算，可以将数字分解为相同的分量，例如 $9 = 3 \cdot 3$ 。对于矩阵，我们需要小心计算类似平方根的运算对正量进行操作。对于对称的正定矩阵（参见第3.2.3节），我们可以从许多平方根等价运算中进行选择。Cholesky分解/Cholesky分解提供了对对称正定矩阵的平方根等效运算，这在实践中很有用。

乔列斯基
分解
乔列斯基
分解

定理 4.18 (Cholesky 分解)。对称正定矩阵A可以因式分解为乘积 $A = LL^T$ ，其中L是具有正对角线元素的下三角矩阵：

$$\begin{matrix} a_{11} \cdots a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} \cdots a_{nn} \end{matrix} = \begin{matrix} |11 \cdots 0 \\ \vdots & \ddots & \vdots \\ |n1 \cdots |nn \end{matrix} \begin{matrix} |11 \cdots |n1 \\ \vdots & \ddots & \vdots \\ 0 \cdots |nn \end{matrix} \quad (4.44)$$

胆囊因子

L称为A的Cholesky因子，L是唯一的。

例 4.10 (Cholesky 分解)

考虑一个对称的正定矩阵 $A \in R^{3 \times 3}$ 。我们有兴趣找到它的 Cholesky 分解 $A = LL^T$ ，即

$$\begin{matrix} a_{11} a_{21} a_{31} \\ a_{21} a_{22} a_{32} \\ a_{31} a_{32} a_{33} \end{matrix} = LL^T = \begin{matrix} |11 0 0 \\ |21 |22 0 \\ |31 |32 |33 \end{matrix} \begin{matrix} |11 |21 |31 \\ 0 |22 |32 \\ 0 0 |33 \end{matrix} \quad (4.45)$$

乘以右边的收益率

$$\begin{matrix} \text{一个=} & \begin{matrix} \frac{|11}{|21|11} & |21|11 & |31|11 \\ |21|11 & \frac{|21}{|22} + |22|0 & |31||21 + |32||22 \\ |31||11 |31||21 + |32||22 | & \frac{|31}{|32} + |32|0 + |33|0 \end{matrix} \end{matrix} \quad (4.46)$$

比较 (4.45) 的左侧和 (4.46) 的右侧表明在对角线元素 $|ii|$ 中存在一个简单的模式：

$$|11| = \sqrt{a_{11}} \quad , \quad |22| = a_{22} - |11|^2, \quad |33| = a_{33} - (|11|^2 + |22|^2). \quad (4.47)$$

类似地,对于对角线下方的元素 ($|ij|$,其中 $i > j$) ,也存在重复模式：

$$|21| = |11| - a_{21}, \quad |31| = |11| - a_{31}, \quad |32| = (a_{32} - |31||21|). \quad (4.48)$$

因此,我们为任何对称的正定 3×3 矩阵构造了 Cholesky 分解。关键实现是我们可以反向计算 L 的组件 $|ij|$ 应该是什么,给定 A 的值 a_{ij} 和先前计算的 $|ij|$ 值。

Cholesky 分解是机器学习基础数值计算的重要工具。这里,对称正定矩阵需要频繁操作,例如,多元高斯变量的协方差矩阵(参见第 6.5 节)是对称的正定矩阵。这个协方差矩阵的 Cholesky 分解允许我们从高斯分布中生成样本。它还允许我们执行随机变量的线性变换,这在深度随机模型中计算梯度时被大量利用,例如变分自动编码器(Jimenez Rezende 等人,2014 年;Kingma 和 Welling,2014 年)。Cholesky 分解还允许我们非常有效地计算行列式。给定 Cholesky 分解 $A = LL^T$ 知道 $\det(A) = \det(L) \det(L^T)$ 行列式只是其对角线项的乘积,因此 $\det(A) =$ 因此,许多数值软件包使用 Cholesky 分解以提高计算效率。

, 我们

$= \det(\text{大号})$.因为 L 是三角形

$$= \frac{\partial}{\partial}.$$

4.4 特征分解和对角化

对角矩阵是在所有非对角元素对角矩阵上都具有零值的矩阵,即它们的形式

$$D = \begin{matrix} c_1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots \\ 0 & \cdots & c_n \end{matrix} \quad (4.49)$$

它们允许快速计算行列式、幂和逆。行列式是其对角线项的乘积,矩阵幂 D^k 由每个对角线元素的 k 次幂给出,如果所有对角线元素都不为零,则逆 D^{-1} 是其对角线元素的倒数。

在本节中,我们将讨论如何将矩阵转换为对角线

形式。这是我们在第 2.7.2 节中讨论的基变化和第 4.2 节中的特征值的重要应用。

回想一下,如果存在可逆矩阵 P ,则两个矩阵 A, D 相似 (定义 2.22)
 $D = P^{-1}AP$ 。更具体地说,我们将研究类似于对角矩阵 D 的矩阵 A ,后者在对角线上包含 A 的特征值。

可对角化的

定义 4.19 (可对角化)。如果矩阵 $A \in R^{n \times n}$ 类似于对角矩阵,即矩阵 $A \in R^{n \times n}$ 是可对角化的,即如果存在可逆矩阵 $P \in R^{n \times n}$ 使得 $D = P^{-1}AP$

在下文中,我们将看到对角化矩阵 $A \in R^{n \times n}$ 是表达相同线性映射的一种方式,但在另一个基础上 (参见第 2.6.1 节),这将成为一个由特征组成的基础 A 的向量。

令 $A \in R^{n \times n}$, 量。我们定义 $\lambda_1, \dots, \lambda_n$ 是一组标量,令 p_1, \dots, p_n 设 $\lambda_1, \dots, \lambda_n$ 中的一组向量。
 $P := [p_1, \dots, p_n]$ 并设 $D \in R^{n \times n}$ 为对角矩阵,对角元素为 $\lambda_1, \dots, \lambda_n$ 。然后我们可以证明

$$AP = P D \quad (4.50)$$

当且仅当 $\lambda_1, \dots, \lambda_n$ 是 A 和 p_1, \dots, p_n 的特征值。 $\lambda_1, \dots, \lambda_n$ 是 A 的相应特征向量。

我们可以看到这个说法成立,因为

$$AP = A[p_1, \dots, p_n] = [Ap_1, \dots, AP] \quad , \quad (4.51)$$

$$\begin{matrix} & \lambda_1 & & 0 \\ & \vdots & & \ddots \\ & 0 & & \lambda_n \end{matrix}$$

$$PD = [p_1, \dots, p_n] \quad \begin{matrix} & \lambda_1 & & 0 \\ & \vdots & & \ddots \\ & 0 & & \lambda_n \end{matrix} = [\lambda_1 p_1, \dots, \lambda_n p_n] . \quad (4.52)$$

因此,(4.50) 意味着

$$Ap_1 = \lambda_1 p_1 \quad (4.53)$$

⋮

$$Ap_n = \lambda_n p_n . \quad (4.54)$$

因此, P 的列必须是 A 的特征向量。

我们对角化的定义要求 $P \in R^{n \times n}$ 是可逆的,即 P 具有满秩 (定理 4.3)。这要求我们有 n 个线性独立的特征向量 p_1, \dots, p_n , 即 p_i 构成 R^n 的基。

定理 4.20 (本征分解)。方阵 $A \in R^{n \times n}$ 可以因式分解为

$$A = PDP^{-1}, \quad (4.55)$$

其中 $P \in R^{n \times n}$ 并且 D 是对角矩阵,其对角线元素是 A 的特征值,当且仅当 A 的特征向量构成 R^n 的基时。

4.4 特征分解和对角化

117

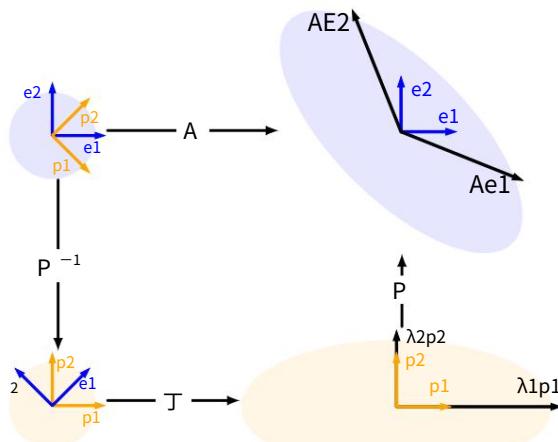


图 4.1 背后的直觉

作为顺序变换的特征分解。

左上角到
角: P^{-1} 左下执行基础变化 (此处
绘制在 R2 中并描述为类似
旋转的)

操作) 从标准基础变成

本征基础。
左下到右下角: D 沿着重新
映射的正交特征向量执
行缩放, 这里用一个被拉伸
成椭圆的圆描
绘。从右下角到右
上角: P 撤销基础

定理 4.20 意味着只有无缺陷矩阵可以对角化并且 P 的列是 A 的 n 个特征向量。对于对称矩阵, 我们可以获得更强的特征值分解位置的结果。

定理 4.21 对称矩阵 $S \in \mathbb{R}^{n \times n}$ 总是可以对角化的。

定理 4.21 直接来自谱定理 4.15。此外, 谱定理指出我们可以找到 \mathbb{R}^n 的特征向量的 ONB。这使得 P 成为正交矩阵, 使得 $D = P^{-1}AP$ 备注。矩阵的 Jordan 范式提供了一种适用于有缺陷矩阵的分解 (Lang, 1987), 但超出了本书的范围。 ◇

改变 (描绘为反向旋转) 并
恢复原始坐标系。

特征分解的几何直觉我们可以如下解释矩阵的特征分解 (另

请参见图 4.1) : 令 A 为具有执行基的线性映射的变换矩阵

关于标准基础 e_i (蓝色箭头)。 P 从标准基础变为特征基础。⁻¹

然后, 对角线 D 按特征值 λ_i 缩放沿这些轴的向量。最后, P 将这些缩放向量转换回标准/规范坐标, 产生 $\lambda_i p_i$ 。

例 4.11 (特征分解)

让我们计算 A 的特征分解

$$\begin{matrix} 5 & -2 \\ -1 & 2 \end{matrix} \quad -2 \quad 5 \quad \text{第} \quad \cdot$$

1 步: 计算特征值和特征向量。特点

A的多项式是

$$\det(A - \lambda I) = \det \begin{vmatrix} -\lambda & -1 \\ 5 & 2 - \lambda \end{vmatrix} = \frac{5}{2} - \lambda \quad (4.56a)$$

$$= (\frac{5}{2} - \lambda)^2 - 1 = \lambda^2 - 5\lambda + \frac{21}{4} = (\lambda - \frac{7}{2})(\lambda - \frac{3}{2}). \quad (4.56b)$$

因此，A的特征值是 $\lambda_1 = \frac{7}{2}$ （和 $\lambda_2 = \frac{3}{2}$ 是特征多项式的根），关联获得归一化的特征向量通过以下方式

$$p_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, p_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \quad (4.57)$$

这产生

$$p_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, p_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \quad (4.58)$$

第2步：检查是否存在。特征向量 p_1 和 p_2 构成R2的基础。

因此，A可以对角化。

步骤3：构造矩阵P以对角化A。我们收集P中A的特征向量，使得

$$P = [p_1, p_2] = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \quad (4.59)$$

然后我们得到

$$P^{-1}AP = \begin{pmatrix} \frac{7}{2} & 0 \\ 0 & \frac{3}{2} \end{pmatrix} = D. \quad (4.60)$$

图4.1 可视化特征分解
等价地，我们得到（利用本例中的P p1和p2形成ONB） $P^{-1} = P$ 因为特征向量

一个=-25作为
一个序列
线性变
换。

$$\begin{matrix} 5 & -2 \\ -2 & 5 \end{matrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \frac{7}{2} & 0 \\ 0 & \frac{3}{2} \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}. \quad (4.61)$$

- 对角矩阵D可以有效地提升为一个幂。因此，我们可以通过特征值分解（如果存在）找到矩阵 $A \in \mathbb{R}^{n \times n}$ 的矩阵幂，使得

$$A^k = (PDP^{-1})^k = P D^k P^{-1}. \quad (4.62)$$

计算 D^k 是高效的，因为我们将此操作单独应用于任何对角线元素。

- 假设特征分解 $A = PDP^{-1}$ 存在。然后，

$$\det(A) = \det(PDP^{-1}) = \det(P) \det(D) \det(P^{-1}) \quad (4.63a)$$

$$= \det(D) = \text{迪} \quad (4.63b)$$

允许有效计算A的行列式。

特征值分解需要方阵。对一般矩阵进行分解会很有用。在下一节中，我们将介绍一种更通用的矩阵分解技术，即奇异值分解。

4.5 奇异值分解

矩阵的奇异值分解 (SVD) 是线性代数中的一种中心矩阵分解方法。它被称为“线性代数基本定理”(Strang, 1993)，因为它可以应用于所有矩阵，而不仅仅是方矩阵，而且它始终存在。

此外，正如我们将在下文中探讨的那样，表示线性映射 $\Phi : V \rightarrow W$ 的矩阵 A 的 SVD 量化了这两个向量空间的基础几何形状之间的变化。我们推荐 Kalman (1996) 和 Roy and Banerjee (2014) 的工作，以便更深入地概述 SVD 的数学。

定理 4.22 (SVD 定理)。令 $A \in \mathbb{R}^{m \times n}$ 为秩为 $r \in [0, \min(m, n)]$ 的矩形矩阵。 A 的 SVD 是以下形式的分解

奇异值分解定理

SVD

奇异值分解

$$\begin{array}{c} n \\ * \end{array} \xrightarrow{\text{一个}} \begin{array}{c} * \\ \square \end{array} \xrightarrow{*} \begin{array}{c} n \\ \Sigma \end{array} \xrightarrow{*} \begin{array}{c} n \\ V \\ - \\ u \end{array} \quad (4.64)$$

具有正交矩阵 $U \in \mathbb{R}^{m \times m}$ ，列向量 $u_i | i = 1, \dots, m$ ，和一个正交矩阵 $V \in \mathbb{R}^{n \times n}$ ，列向量为 $v_j | j = 1, \dots, n$ 。此外， Σ 是 $m \times n$ 矩阵， $\Sigma_{ii} = \sigma_i > 0$ 且 $\Sigma_{ij} = 0, i \neq j$ 。

名词

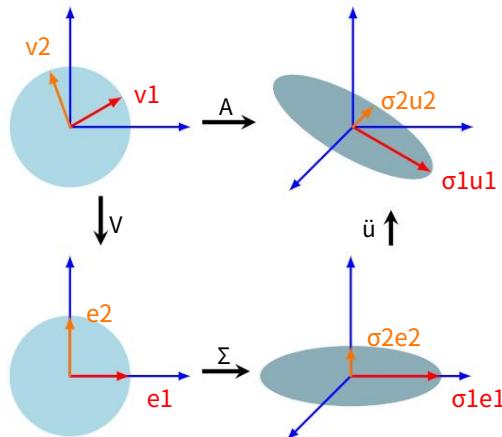
对角线项 σ_i ， Σ 的 r 称为奇异值， $i = 1, \dots, r$ 。
 u_i 称为左奇异向量， v_j 称为右奇异向量。按照惯例，奇异值是有序的，即 $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$ 。

右奇异向量

奇异值矩阵 Σ 是唯一的，但需要注意。奇异值
观察到 $\Sigma \in \mathbb{R}^{m \times n}$ 是矩形的。特别是， Σ 的大小与 A 相同。这意味着 Σ 具有包含奇异值的对
角子矩阵，并且需要额外的零填充。具体来说，如果 $m > n$ ，则矩阵 Σ 具有直到第 n 行的对角线结
构，然后由

图 4.1 矩阵 $A \in R^{3 \times 2}$
作为顺序变换的 SVD
背后的直觉。

从左上角
到左下角: V
在 R^2 中执行基
础更改。
从左下角到右
下角: Σ 从 R^2 缩
放和映射到 R^3 。
右下角的椭圆位于 R^3
中。第三个维
度是



正交于表面

椭圆盘。
从右下角到右上
角: U 在 R^3
内执行基础更改。

① 从下面的 $n+1$ 到 m 的行向量,使得

$$\sigma_1 \ 0 \ 0$$

$$0 \ \ddots \ 0$$

$$\Sigma = \begin{matrix} & & & 0 & 0 & \sigma_n & & \cdot \\ & & & 0 & \dots & 0 & & \\ & & & \vdots & & \vdots & & \\ & & & 0 & \dots & 0 & & \end{matrix} \quad (4.65)$$

如果 $m < n$, 则矩阵 Σ 具有直到 m 列的对角线结构以及由 $m+1$ 到 n 中的 0 组成的列:

$$\Sigma = \begin{matrix} \sigma_1 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & ; & & ; \\ 0 & 0 & \ddots & 0 & \dots & 0 \end{matrix} \quad (4.66)$$

评论。SVD 存在于任何矩阵 $A \in R^{m \times n}$ 。



4.5.1 SVD 的几何直觉

SVD 提供了描述变换矩阵 A 的几何直觉。在下文中,我们将讨论 SVD 作为在基上执行的顺序线性变换。在示例 4.12 中,我们将 SVD 的变换矩阵应用于 R^2 中的一组向量,这使我们能够更清楚地可视化每个变换的效果。

矩阵的 SVD 可以解释为相应线性映射 (回忆第 2.7.1 节) $\Phi: R^n \rightarrow R^m$ 分解为三个操作;见图 4.1。SVD 的直觉表面上遵循与我们的特征分解直觉相似的结构,见图 4.1:广义上讲,SVD 通过 V 执行基础变化,然后通过奇异的维度进行缩放和增加 (或减少)

4.5 奇异值分解

价值矩阵 Σ 。最后,它通过 U 执行第二次基础更改。SVD 包含许多重要的细节和注意事项,这就是为什么我们将更详细地回顾我们的直觉。

假设我们得到一个线性映射 $\Phi : \mathbf{R}^n \rightarrow \mathbf{R}^m$ 的变换矩阵,分别相对于 \mathbf{R}^n 和 \mathbf{R}^m 的标准基B和C。此外,假设 \mathbf{R}^n 的第二基 B_{\sim} 和 \mathbf{R}^m 的 C_{\sim} 。然后 1. 矩阵V在域 \mathbf{R}^n 中从 B_{\sim} (由图 4.1 左上角的红色和橙色向量 v_1 和 v_2 表示)到标准基B执行基变化。V执行基变化从B到 B_{\sim} 。红色和橙色向量现在与图 4.1 左下角的规范基对齐。

复习很有用
基变化 (第 2.7.2 节)、正交矩阵 (定义 3.8) 和正交基 (第 3.5 节)。

$$= V^{-1}$$

2. 将坐标系改为 B_{\sim} 后, Σ 将新坐标按奇异值 σ_i 进行缩放 (并增删维数),即 Σ 为 Φ 相对于 B_{\sim} 和 C_{\sim} 的变换矩阵,表示为红色和橙色矢量被拉伸并位于 e_1-e_2 平面中,该平面现在嵌入在图 4.1 右下角的三维空间中。

3. U 执行从 C_{\sim} 到 \mathbf{R}^m 的规范基的共域 \mathbf{R}^m 中的基变化,由 e_1-e_2 平面外的红色和橙色矢量旋转表示。如图 4.1 的右上角所示。

SVD 表示域和辅域中的基础变化。

这与在相同向量空间内运行的特征分解形成对比,在相同向量空间中应用相同的基础变化然后取消。SVD 的特别之处在于这两个不同的基同时由奇异值矩阵 Σ 连接起来。

例 4.12 (向量和 SVD)

考虑一个正方形网格的向量 $X \in \mathbf{R}^2$ 的映射,它适合以原点为中心的大小为 2×2 的盒子。使用标准基础,我们映射这些向量使用

$$\text{一个=} \begin{pmatrix} 1 & -0.8 \\ 0 & 1 \end{pmatrix} = U \Sigma V \quad (4.67a)$$

$$= \begin{pmatrix} -0.79 & 0 & -0.62 & 1.62 \\ 0.38 & -0.78 & -0.49 & 0 \\ -0.48 & -0.62 & 0.62 & 0 \end{pmatrix} \begin{pmatrix} 0.62 & 0 \\ 0 & 1.0 \end{pmatrix} \begin{pmatrix} 0.62 & 0 \\ -0.62 & -0.78 \end{pmatrix} \quad (4.67b)$$

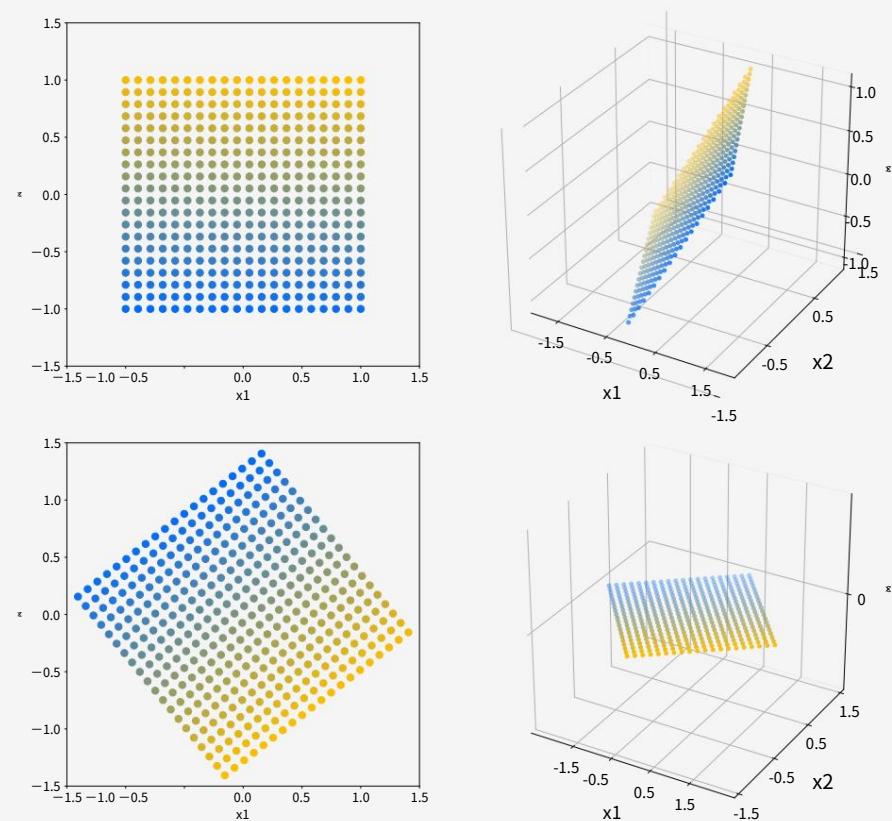
我们从一组排列在网格中的向量 X (彩色点;参见图 4.2 的左上图)开始。然后我们应用旋转 X 的 V。旋转的矢量显示在图 4.2 的左下面板中。我们现在使用奇异值矩阵 Σ 将这些向量映射到陪域 \mathbf{R}^3 (参见图 4.2 中的右下面板)。请注意,所有向量都位于

$x_1 - x_2$ 平面。第三个坐标始终为 0。 $x_1 - x_2$ 平面中的向量已被奇异值拉伸。

通过 A 将向量 X 直接映射到辅域 R_3 等于通过 $U\Sigma V$ 对 X 进行变换，其中 U 在辅域 R_3 内执行旋转，以便映射的向量不再局限于 $x_1 - x_2$ 平面；它们仍然在一个平面上，如图 4.2 的右上面板所示。

图 4.2 SVD 和矢量映射（用圆盘表示）。面板遵循相同的逆时针方向

的结构
图 4.1。



4.5.2 SVD 的构造

我们接下来将讨论为什么存在 SVD 并详细说明如何计算它。一般矩阵的 SVD 与方阵的特征分解有一些相似之处。

评论。比较 SPD 矩阵的特征分解

$$\text{小号} = \text{小号} = PDP \quad (4.68)$$

与相应的 SVD

$$S = U\Sigma V^T \quad . \quad (4.69)$$

如果我们设置

$$U = P = V \quad , \quad D = \Sigma \quad , \quad (4.70)$$

我们看到 SPD 矩阵的 SVD 是它们的特征分解。 ◇在下文中,我们将探讨为什么定理 4.22 成

立以及 SVD 是如何构建的。计算 $A \in \mathbb{R}^{m \times n}$ 的 SVD 等价于找到余域 \mathbb{R}^m 和域 \mathbb{R}^n 的两组正交基 $U = (u_1, \dots, u_m)$ 和 $V = (v_1, \dots, v_n)$, 分别。从这些有序的基础,我们将构建矩阵 U 和 V 。

我们的计划是从构造正交右奇异向量 $v_1, \dots, v_n \in \mathbb{R}^n$ 开始。然后我们构建左奇异向量的正交集 $u_1, \dots, u_m \in \mathbb{R}^m$ 。此后,我们将两者联系起来,并要求在 A 的变换下保持 v_i 的正交性。这很重要,因为我们知道图像 $A v_i$ 形成一组正交向量。然后我们将通过标量因子对这些图像进行归一化,这将成为奇异值。

让我们从构造右奇异向量开始。谱定理 (定理 4.15) 告诉我们,对称矩阵的特征向量构成一个 ONB,这也意味着它可以对角化。此外,根据定理 4.14,我们总能从任意矩形矩阵 $A \in \mathbb{R}^{m \times n}$ 构造对称半正定矩阵 $A^T A \in \mathbb{R}^{n \times n}$ 。因此,我们总是可以对角化 $A^T A$ 并得到

$$A^T A = P D P^T = P \begin{matrix} \lambda_1 & & & \\ & \ddots & & \\ & & \lambda_n & \\ & 0 & \cdots & 0 \end{matrix} P^T, \quad (4.71)$$

其中 P 是正交矩阵,由正交特征基组成。 $\lambda_i \geq 0$ 是 $A^T A$ 的特征值。让我们假设 A 的 SVD 存在并将 (4.64) 注入 (4.71)。这产生 $= V \Sigma^T U^T U \Sigma V^T$ 其中 U, V 是正交矩阵。因此,对于 $U^T U = I$ 我们得到

$$A^T A = (U \Sigma V^T)^T (U \Sigma V) = U \Sigma^T V^T V \Sigma U^T = U \Sigma^2 U^T, \quad (4.72)$$

由

$$A^T A = V \Sigma^2 V^T = V \begin{matrix} \sigma_1^2 & & & \\ & \ddots & & \\ & & \sigma_n^2 & \\ & 0 & \cdots & 0 \end{matrix} V^T, \quad (4.73)$$

现在比较 (4.71) 和 (4.73), 我们确定

$$V = P, \quad (4.74)$$

$$\sigma_i = \sqrt{\lambda_i}. \quad (4.75)$$

因此,构成P的A A的特征向量是A的右奇异向量V (见 (4.74))。A A的特征值是Σ的奇异值的平方 (见 (4.75))。

为了获得左奇异向量U,我们遵循类似的过程。

我们从计算对称矩阵AA ∈ Rm × m (而不是之前的A A ∈ Rn × n)的 SVD 开始。A的 SVD 产生AA = (UΣV)

$$(U\Sigma V) = U\Sigma V \quad (4.76a)$$

$$\begin{matrix} & & 0 & 0 \\ \text{你} & 0 & \ddots & 0.20 & \ddots & \cdot \\ 0 & \sigma & * & & & \end{matrix} \quad (4.76b)$$

谱定理告诉我们AA = SDS可以对角化,我们可以找到AA S的特征向量的 ONB。AA的正交特征向量是左奇异向量U,收集在

并在 SVD 的余域中形成正交基。

这就留下了矩阵 Σ 的结构问题。由于AA和A A具有相同的非零特征值 (参见第 106 页),因此非零两种情况下 SVD 中的Σ矩阵条目必须相同。

最后一步是将我们目前接触到的所有部分联系起来。我们在V中有一组正交右奇异向量。为了完成 SVD 的构造,我们将它们与正交向量 U 连接起来。为了达到这个目标,我们利用A下的vi的图像也必须是正交的这一事实。我们可以使用第 3.4 节的结果来证明这一点。

我们要求Avi和Avj之间的内积必须为0,因为i = j。对于任意两个正交特征向量vi i = j,我们认为, vj,

$$(Avi) \cdot (Avj) = v_i \cdot (A A)vj = v_i \cdot i(\lambda_j v_j) = \lambda_j v_i \cdot v_j = 0. \quad (4.77)$$

对于m = r的情况,它认为{Av1, …, Avr}是 Rm 的 r 维子空间的基。

为了完成 SVD 构造,我们需要正交的左奇异向量:我们将右奇异向量Avi的图像归一化并获得

$$\frac{\text{阿维}}{\| \text{阿维} \|} = \frac{1}{\lambda_i} Av_i = \sqrt{\frac{1}{\sigma_i^2}} \text{阿维}, \quad (4.78)$$

其中最后一个等式是从 (4.75) 和 (4.76b) 中获得的,向我们展示了AA的特征值使得 $\sigma_i^2 = \lambda_i$ 。

因此, A A的特征向量,我们知道的是右和它们在A 下的归一化图像,左奇异形成奇异向量vi,两个自治的 ONB,它们通过向量ui奇异,值矩阵Σ。

奇异值方程

让我们重新排列 (4.78) 以获得奇异值方程

$$Av_i = \sigma_i u_i, \quad i = 1, \dots, r. \quad (4.79)$$

该方程与特征值方程 (4.25) 非常相似,但左侧和右侧的向量不同。

对于 $n < m$, (4.79) 仅适用于 $i \leq n$, 但 (4.79) 对 $i > n$ 的 u_i 没有任何说明。然而, 我们通过构造知道它们是正态的。相反, 对于 $m < n$, (4.79) 仅在 $i \leq m$ 时成立。对于 $i > m$, 我们有 $A v_i = 0$ 并且我们仍然知道 v_i 形成一个正交集。

这意味着 SVD 还提供 A 的核 (零空间) 的正交基, 即 $Ax = 0$ 的向量 x 的集合 (参见第 2.7.3 节)。

连接 v_i 作为 V 的列和 u_i 作为
产量

$$AV = U\Sigma \quad , \quad (4.80)$$

其中 Σ 具有与 A 相同的维度和第 1 行的对角线结构, ...。因此, 右乘 V 是 A 的 SVD。
收益率 $A = U\Sigma V^T$

例 4.13 (计算 SVD)

让我们找到奇异值分解

$$\begin{array}{c} \text{一个} = \\ \begin{array}{ccccc} 1 & 0 & 1 & -2 & 1 \\ 0 & & & & \end{array} \end{array} \quad . \quad (4.81)$$

SVD 要求我们计算右奇异向量 v_j 、奇异值 σ_k 和左奇异向量 u_i 。

步骤 1: 右奇异向量作为 $A^T A$ 的特征基。

我们从计算开始

$$\begin{array}{c} \text{一个一个} = \\ \begin{array}{ccccc} 1 & -2 & & & \\ 0 & 1 & 1 & 0 & -2 \\ 1 & 0 & 0 & & \end{array} \end{array} = \begin{array}{ccccc} 5 & -2 & 1 & & \\ -2 & 1 & 0 & & \\ 1 & & 0 & 1 & \end{array} \quad . \quad (4.82)$$

我们通过 $A^T A$ 的特征值分解来计算奇异值和右奇异向量 v_j , 给出为

$$\begin{array}{c} \text{一个一个} = \\ \begin{array}{ccccc} \frac{\sqrt{30}}{\sqrt{-230}} & 0 & \frac{\sqrt{-16}}{\sqrt{-26}} & 6 & 0 \\ \frac{\sqrt{-230}}{\sqrt{30}} & \frac{\sqrt{-16}}{\sqrt{-26}} & \frac{\sqrt{-16}}{\sqrt{-26}} & 0 & 1 \\ \frac{\sqrt{30}}{\sqrt{30}} & \frac{\sqrt{-16}}{\sqrt{-26}} & \frac{\sqrt{-16}}{\sqrt{-26}} & 0 & 0 \end{array} \end{array} = P D P^T \quad , \quad (4.83)$$

并且我们获得右奇异向量作为 P 的列, 以便

$$V = P = \begin{array}{ccccc} \frac{\sqrt{30}}{\sqrt{-230}} & 0 & \frac{\sqrt{-16}}{\sqrt{-26}} & & \\ \frac{\sqrt{-230}}{\sqrt{30}} & \frac{\sqrt{-16}}{\sqrt{-26}} & \frac{\sqrt{-16}}{\sqrt{-26}} & & \\ \frac{\sqrt{30}}{\sqrt{30}} & \frac{\sqrt{-16}}{\sqrt{-26}} & \frac{\sqrt{-16}}{\sqrt{-26}} & & \end{array} \quad . \quad (4.84)$$

第 2 步: 奇异值矩阵。

由于奇异值 σ_i 是特征值的平方根

$A = A$ 我们直接从 D 得到它们。由于 $\text{rk}(A) = 2$, 因此只有两个非零奇异值: $\sigma_1 = \sqrt{6}$ 和 $\sigma_2 = 1$ 。奇异值矩阵必须与 A 大小相同, 我们得到

$$\Sigma = \begin{matrix} \sqrt{6} & 0 & 0 \\ 0 & 1 & 0 \end{matrix} . \quad (4.85)$$

第三步: 左奇异向量作为右图的归一化图像
奇异向量。

我们通过计算 A 下右奇异向量的图像并通过将它们除以相应的奇异值对其进行归一化来找到左奇异向量。我们获得

$$u_1 = \frac{1}{\sigma_1} Av_1 = \frac{1}{\sqrt{6}} \begin{matrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{matrix} = \frac{\begin{matrix} \frac{1}{\sqrt{30}} \\ \frac{-2}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \end{matrix}}{\frac{1}{\sqrt{30}}} = \begin{matrix} \frac{1}{\sqrt{5}} \\ -\frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{matrix}, \quad (4.86)$$

$$u_2 = \frac{1}{\sigma_2} Av_2 = \frac{1}{1} \begin{matrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{matrix} = \frac{0}{\frac{1}{\sqrt{5}}} = \begin{matrix} 2 \\ \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{matrix}, \quad (4.87)$$

$$U = [u_1, u_2] = \sqrt{5} \begin{matrix} 1 & 2 \\ -2 & 1 \end{matrix}. \quad (4.88)$$

请注意, 在计算机上, 此处说明的方法具有较差的数值行为, 并且 A 的 SVD 通常是在不求助于 $A = A$ 的特征值分解的情况下计算的。

4.5.3 特征值分解与奇异值分解

让我们考虑特征分解 $A = P D P^{-1}$ SVD $A = U \Sigma V^T$ 并回顾过去章节的核心要素。

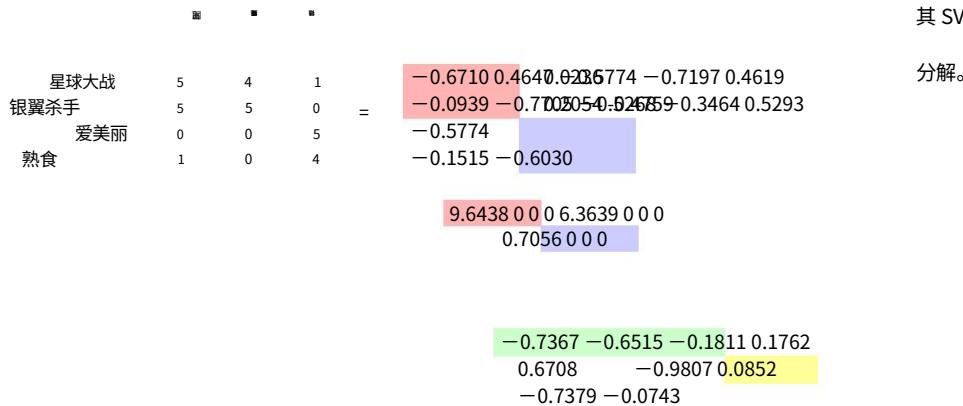
- SVD 对于任何矩阵 $R^{m \times n}$ 总是存在的。特征分解仅针对方阵 $R^{n \times n}$ 定义, 并且仅当我们能够找到 R^n 的特征向量的基时才存在。
- 特征分解矩阵 P 中的向量不一定是正交的, 即基的变化不是简单的旋转和缩放。

另一方面, SVD 中的矩阵 U 和 V 中的向量是正交的, 因此它们确实表示旋转。

- 特征分解和 SVD 都是三个线性映射的组合: 1. 域中基的变化 2. 每个新基向量的独立缩放和
从主域到辅域的映射

3. codomain 基础的改变

图 4.1三人对四部电影的电影评分及
其 SVD



特征分解和 SVD 之间的一个关键区别是,在 SVD 中,域和余域可以是不同维度的向量空间。

- 在 SVD 中,左奇异向量矩阵和右奇异向量矩阵U和V通常不是彼此的逆矩阵 (它们在不同的向量空间中执行基变化)。在特征分解中,基变化 ma 互为倒数。
- 在 SVD 中,对角矩阵 Σ 中的元素都是实数且非负,这对于特征分解中的对角矩阵通常不是这样。
- SVD 和特征分解通过它们的投影密切相关- A的左奇异向量是AA的特征向量- A的右奇异向量是A A^T 的特征向量。
 - A的非零奇异值是非零奇异值的平方根
 - AA和A A^T 的特征值。
- 对于对称矩阵 $A \in R^{n \times n}$,特征值分解和 SVD 是一回事,这是从谱定理 4.15 得出的。

示例 4.14 (在电影评级和消费者中查找结构)

让我们通过分析关于人和他们喜欢的电影的数据来添加对 SVD 的实际解释。假设三个观众 (阿里、比阿特丽克斯、钱德拉)给四部不同的电影 (星球大战、银翼杀手、天使爱美丽、熟食店)打分。他们的评分是0 (最差)和5 (最好)之间的值,并编码在数据矩阵 $A \in R^{4 \times 3}$ 中,如图 4.1 所示。每行代表一部电影,每列代表一个用户。因此,电影评级的列向量 (每个观众一个)是 x_{Ali} 、 x_{Beatrix} 、 x_{Chandra} 。

使用 SVD 分解 A 为我们提供了一种捕捉人们如何评价电影的关系的方法,特别是如果存在将哪些人喜欢哪些电影联系起来的结构。将 SVD 应用于我们的数据矩阵 A 需要做出一些假设:

1. 所有观众都使用相同的线性映射一致地对电影进行评分。
2. 评级中没有错误或噪音。
3. 我们将左奇异向量 u_i 解释为典型电影,将右奇异向量 v_j 解释为典型观众。

然后我们假设任何观众的特定电影偏好都可以表示为 v_j 的线性组合。同样,任何电影的好感度都可以表示为 u_i 的线性组合。因此,SVD 域中的向量可以解释为刻板观众“空间”中的观众,SVD 余域中的向量将这两个“空间”对应为刻板“空间”中的电影电影。让我们只检查电影用户矩阵的 SVD。第一个左奇异向量 u_1 对于两部科幻电影来说具有很大的绝对值,并且第一个奇异值的跨度很大(图 4.1 中的红色阴影)。因此,如果用户具有一组特定的电影(科幻主题),则这会将一种类型的相应观众和电影数据分组。类似地,数据本身覆盖的第一个右单数 v_1 显示 Ali 和 Beatrix 的绝对值很大,足够的多样性给科幻电影带来了高评级(图 4.1 中的绿色阴影)。观众和这表明 v_1 反映了科幻爱好者的概念。

电影。

同样, u_2 似乎抓住了法国艺术电影的主题,而 v_2 表明 Chandra 接近于此类电影的理想化爱好者。理想化的科幻爱好者是纯粹主义者,只喜欢科幻电影,因此科幻爱好者 v_1 对除科幻主题以外的所有内容都给予零评级。奇异值矩阵 Σ 的对角子结构暗示了这一逻辑。因此,一部特定的电影由它如何(线性地)分解成它的刻板电影来表示。同样,一个人将通过他们如何分解(通过线性组合)成电影主题来表示。

有必要简要讨论 SVD 术语和约定,因为文献中使用了不同的版本。虽然这些差异可能令人困惑,但数学对它们来说仍然是不变的。

- 为了符号和抽象的方便,我们使用 SVD 符号,其中 SVD 被描述为具有两个正方形左和右奇异向量矩阵,但是一个非正方形奇异值矩阵。我们对 SVD 的定义(4.64)有时称为全 SVD。
- 一些作者对 SVD 的定义略有不同,并侧重于方奇异矩阵。然后,对于 $A \in \mathbb{R}^{m \times n}$ 和 $m > n$,

$$A_{m \times n} = U_{m \times n} \Sigma_{n \times n} V_n^T \quad . \quad (4.89)$$

有时,此公式称为缩减 SVD (例如,Datta (2010))缩减 SVD或SVD (例如,Press 等人 (2007))。这种替代格式仅改变了矩阵的构造方式,但保留了 SVD 的数学结构不变。这个替代公式的便利之处在于 Σ 是对角线的,就像在特征值分解中一样。

- 在 4.6 节中,我们将学习使用 SVD (也称为截断 SVD)的矩阵逼近技术。

截断的 SVD
- 可以定义秩为 r 的矩阵A的SVD ,使得U为 $m \times r$ 矩阵, Σ 为对角矩阵 $r \times r$, V 为 $r \times n$ 矩阵。
这种结构与我们的定义非常相似,并确保对角矩阵 Σ 沿对角线只有非零元素。这种替代符号的主要便利之处在于 Σ 是对角线的,就像在特征值分解中一样。
- A的 SVD仅适用于 $m > n$ 的 $m \times n$ 矩阵的限制实际上是不必要的。当 $m < n$ 时,SVD 分解将产生零列多于行的 Σ ,因此产生奇异值 $\sigma_{m+1}, \dots, \sigma_n$ 为0。

SVD 用于机器学习的各种应用,从曲线拟合中的最小二乘问题到求解线性方程组。这些应用程序利用 SVD 的各种重要属性、它与矩阵秩的关系,以及它用较低秩矩阵逼近给定秩矩阵的能力。用 SVD 代替矩阵通常具有使计算对数值舍入误差更稳健的优点。正如我们将在下一节中探讨的那样,SVD 以一种原则性的方式用“更简单”的矩阵来近似矩阵的能力开辟了从降维和主题建模到数据压缩和聚类的机器学习应用。

4.6 矩阵近似

我们将 SVD 视为一种将 $A = U\Sigma V^T \in \mathbb{R}^{m \times n}$ 分解为三个矩阵乘积的方法,其中 $U \in \mathbb{R}^{m \times m}$ 和 $V \in \mathbb{R}^{n \times n}$ 是或正交矩阵,并且 Σ 在其主对角线上包含奇异值。我们现在将研究 SVD 如何让我们将矩阵A表示为更简单 (低阶)矩阵 A_i 的总和,而不是进行完整的 SVD 分解,这适用于矩阵近似方案,其计算成本低于完整矩阵奇异值分解。

我们构造一个 rank-1 矩阵 $A_i \in \mathbb{R}^{m \times n}$ 为

$$A_i := u_i v_i^T \quad \text{我,} \quad (4.90)$$

它由U和V的第 i 个正交列向量的外积形成。图 4.2 显示了巨石阵的图像,可以用矩阵 $A \in \mathbb{R}^{1432 \times 1910}$ 和 (4.90)中定义的一些外积 A_i 来表示。

, 作为

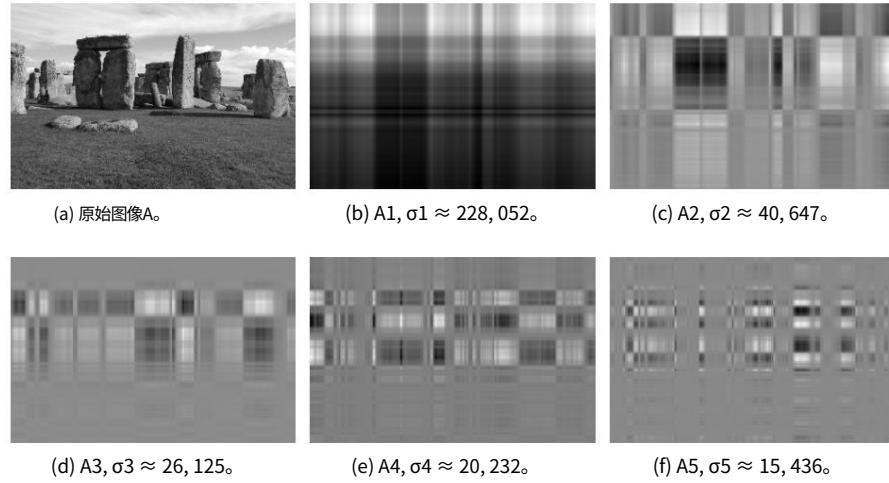
130

矩阵分解

图 4.2 使用 SVD 进行图像处理。 (a) 原始灰度图像是一个 $1,432 \times 1,910$ 的值矩阵

介于 0 (黑色) 和 1 (白色) 之间。 (b) – (f) Rank-1 矩阵 A_1, A_2, \dots, A_5 和它们对应的奇异值 $\sigma_1, \dots, \sigma_5$ 。

每个 rank-1 矩阵的网格状结构是由左和



右奇异向量。

秩为 r 的矩阵 $A \in R^{m \times n}$ 可以写成秩为 1 的矩阵之和

这样

$$\text{一个} = \sum_{i=1}^r \sigma_i u_i v_i^T = \sum_{i=1}^r \sigma_i A_i, \quad (4.91)$$

其中外积矩阵 A_i 由第 i 个奇异值 σ_i 加权。我们可以看出为什么 (4.91) 成立：奇异值矩阵 Σ 的对角线结构只乘以匹配的左奇异值和右奇异值，并按相应的奇异值 σ_i 对它们进行缩放。

向量 $u_i v_i^T$ 对于 $i = 1, \dots, r$

全部 $\neq 0$ ，因为 Σ 是对角矩阵。任何项 $\Sigma_j u_i v_i^T$ 由于 $j \neq i$ 都消失了，因为对应的奇异值为 0。

在 (4.90) 中，我们引入了秩为 1 的矩阵 A_i 。我们将各个 rank-1 矩阵中的 r 求和得到一个 rank- r 矩阵 A ；见 (4.91)。如果总和没有遍历所有矩阵 A_i ，但只到 $i = 1, \dots, k$ ，中间值 $k < r$ ，我们得到一个 rank- k 近似值

k 级
近似

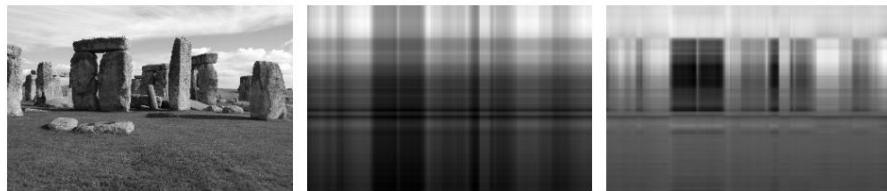
$$\text{一个} (k) := \sum_{i=1}^k \sigma_i u_i v_i^T = \sum_{i=1}^k \sigma_i A_i \quad (4.92)$$

A 与 $\text{rk}(A(k)) = k$ 。图 4.3 显示了巨石阵原始图像 A 的低秩近似值 $A(k)$ 。在 5 阶近似中，岩石的形状变得越来越明显和清晰可辨。

虽然原始图像需要 $1,432 \times 1,910 = 2,735,120$ 个数字，但 rank-5 近似只需要我们存储五个奇异值和五个左右奇异向量 ($1,432$ 和 $1,910$ -dimensional each) 总共有 $5 \cdot (1,432 + 1,910 + 1) = 16,715$ 个数，略高于原始数的 0.6%。

为了衡量 A 与其秩 k 近似值 $A(k)$ 之间的差异（误差），我们需要范数的概念。在 3.1 节中，我们已经使用

4.6 矩阵近似



(a) 原始图像A。

(b) Rank-1 近似A (1)。(c) Rank-2 近似A (2)。



(d) Rank-3 近似A (3)。(e) Rank-4 近似A (4)。(f) Rank-5 近似A (5)。

图 4.3 使用 SVD 进行
图像重建。 (A)

原始图像。 (b)–
(f) 使用 SVD 的
低秩近似重建图像,其中
秩 k

近似值由 $A(k) =$

$$\sum_{i=1}^k \sigma_i u_i v_i^T$$

衡量向量长度的向量范数。以此类推,我们也可以在矩阵上定义范数。

定义 4.23 (矩阵的谱范数)。对于 $x \in \mathbb{R}^n \setminus \{0\}$, 矩阵 $A \in \mathbb{R}^{m \times n}$ 的谱范数范数定义为

$$\|A\|_2 := \max_x \frac{\|Ax\|_2}{\|x\|_2}. \quad (4.93)$$

我们在矩阵范数 (左侧) 中引入下标符号,类似于向量的欧几里得范数 (右侧),其下标为 2。谱范数 (4.93) 决定了任何向量 x 的长度最多乘以 A 。

定理 4.24。 A 的谱范数是它的最大奇异值 σ_1 。

我们把这个定理的证明留作练习。

Young 定理 4.25 (Eckart-Young 定理 (Eckart 和 Young, 1936 年))。定理是一个矩阵 $A \in \mathbb{R}^{m \times n}$, 阶数为 r , 令 $B \in \mathbb{R}^{m \times n}$ 为阶数为 k 的矩阵。对于任何 $k \leq r$ 且 $A(k) = \sum_{i=1}^k \sigma_i u_i v_i^T$

$$\|A - A(k)\|_2 = \arg \min_{B \in \mathbb{R}^{m \times n}} \|A - B\|_2, \quad (4.94)$$

$$\|A - A(k)\|_2 = \sigma_{k+1}. \quad (4.95)$$

Eckart-Young 定理明确说明我们通过使用秩 k 近似值来近似 A 会引入多少误差。我们可以将用 SVD 获得的秩 k 近似解释为满秩矩阵 A 到秩至多 k 矩阵的低维空间的投影。在所有可能的投影中,SVD 最小化 A 和任何秩 k 近似之间的误差 (相对于谱范数)。

我们可以追溯一些步骤来理解为什么 (4.95) 应该成立。

我们观察到 $A - A(k)$ 之间的差异是包含剩余秩 1 矩阵之和的矩阵

$$A - A(k) = \sum_{i=k+1}^r \sigma_i u_i v_i^\top \quad (4.96)$$

根据定理 4.24, 我们立即得到 σ_{k+1} 作为差分矩阵的谱范数。让我们仔细看看 (4.94)。如果我们假设存在另一个矩阵 B 且 $\text{rk}(B) = k$, 使得

$$\|A - B\|_F^2 < \|A - A(k)\|_F^2, \quad (4.97)$$

则存在一个至少 $(n - k)$ 维的零空间 $Z \subseteq \mathbb{R}^n$, 其中 $x \in Z$ 意味着 $Bx = 0$ 。那么它遵循 , 这样的

$$\|Ax\|_2^2 = \|(A - B)x\|_2^2, \quad (4.98)$$

并通过使用包含矩阵范数的 Cauchy-Schwartz 不等式 (3.17) 的一个版本, 我们得到

$$\|Ax\|_2^2 = \|A - B\|_F^2 \|x\|_2^2 < \sigma_{k+1}^2 \|x\|_2^2. \quad (4.99)$$

但是, 存在一个 $(k + 1)$ 维子空间, 其中 $\|Ax\|_2^2 = \sigma_{k+1}^2 \|x\|_2^2$, 它由 A 的右奇异向量 $v_j, j = k + 1$ 跨越。这两个空间产生一个大于 n 的数, 因为两个空间中必须有一个非零向量。这与 2.7.3 节中的秩无效定理 (定理 2.24) 相矛盾。

Eckart-Young 定理意味着我们可以使用 SVD 以原则上的最优 (在谱范数意义上) 方式将 rank- r 矩阵 A 简化为 rank- k 矩阵 A 。我们可以将秩为 k 的矩阵对 A 的近似解释为有损压缩的一种形式。因此, 矩阵的低秩近似出现在许多机器学习应用中, 例如, 图像处理、噪声过滤和不适定问题的正则化。此外, 它在降维和主成分分析中起着关键作用, 我们将在第 10 章中看到。

示例 4.15 (在电影评级和消费者中查找结构 (续))

回到我们的电影评级示例, 我们现在可以应用低秩近似的概念来近似原始数据矩阵。

回想一下, 我们的第一个奇异值捕捉了电影和科幻爱好者中科幻主题的概念。因此, 通过在电影评级矩阵的秩 1 分解中仅使用第一个奇异值项, 我们获得了预测评级

$$A_1 = u_1 v_1^\top = \begin{pmatrix} -0.6710 \\ -0.7197 \\ -0.0939 \\ -0.1515 \end{pmatrix} \begin{pmatrix} -0.7367 & -0.6515 & -0.1811 \end{pmatrix} \quad (4.100a)$$

$$\begin{aligned}
 & 0.4943 \ 0.4372 \ 0.1215 \\
 = & 0.5302 \ 0.4689 \ 0.1303 \\
 & 0.0692 \ 0.0612 \ 0.0170 \\
 & 0.1116 \ 0.0987 \ 0.0274
 \end{aligned} \quad (4.100b)$$

这个一级近似值 A1 很有见地: 它告诉我们 Ali 和 Beatrix 喜欢科幻电影, 例如星球大战和银翼杀手 (条目的值 > 0.4), 但未能捕捉到 Chandra 对其他电影的收视率。这并不奇怪, 因为钱德拉的电影类型没有被第一个奇异值捕获。第二个奇异值为那些电影主题爱好者提供了更好的 rank-1 近似值:

$$\begin{aligned}
 & 0.0236 \\
 & 0.2054 \\
 A2 = u2v^* = & -0.7705 \quad 0.0852 \ 0.1762 \ -0.9807 \\
 & -0.6030
 \end{aligned} \quad (4.101a)$$

$$\begin{aligned}
 & 0.0020 \ 0.0042 \ -0.0231 \\
 = & 0.0175 \ 0.0362 \ -0.2014 \\
 & -0.0656 \ -0.1358 \ 0.7556 \\
 & -0.0514 \ -0.1063 \ 0.5914
 \end{aligned} \quad (4.101b)$$

在第二个 rank-1 近似值 A2 中, 我们很好地捕获了 Chandra 的收视率和电影类型, 但不是科幻电影。这导致我们考虑 rank-2 近似值 A (2), 其中我们组合前两个 rank-1 近似值

$$\begin{aligned}
 & 4.7801 \ 4.2419 \quad 1.0244 \\
 & 5.2252 \ 4.7522 \ -0.0250 \\
 A(2) = \sigma_1 A1 + \sigma_2 A2 = & 0.2493 \ -0.2743 \ 4.9724 \\
 & 0.7495 \ 0.2756 \ 4.0278
 \end{aligned} \quad (4.102)$$

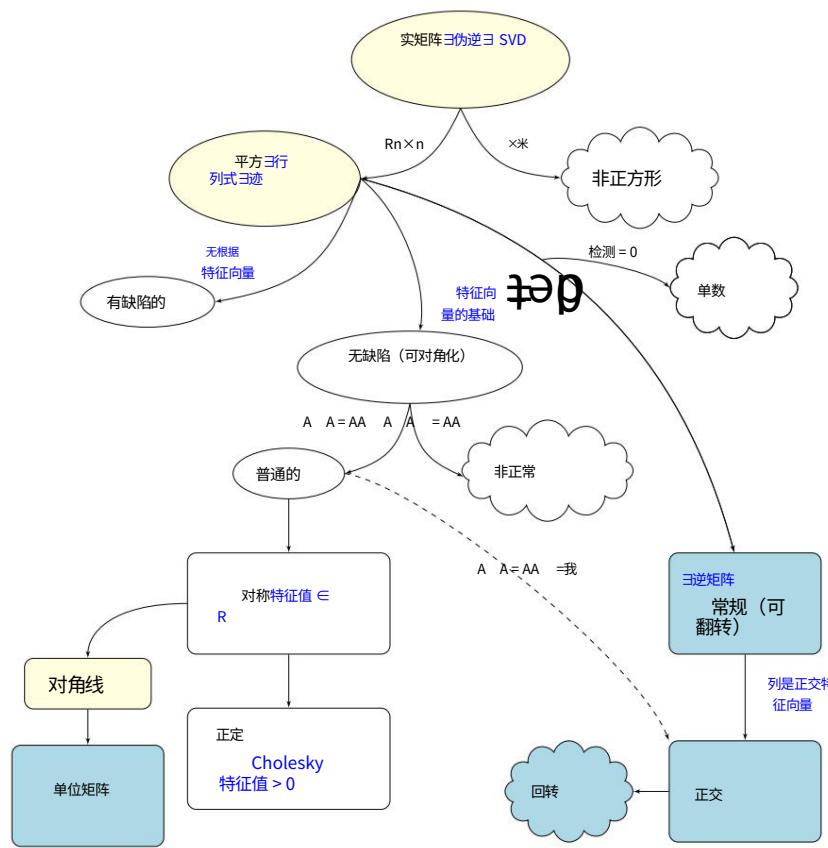
A (2) 类似于原来的电影评分表

$$\begin{array}{r}
 5 \ 4 \ 1 \\
 5 \ 5 \ 0 \\
 \text{一个=} \quad 0 \ 0 \ 5 \\
 \hline
 1 \ 0 \ 4
 \end{array}, \quad (4.103)$$

这表明我们可以忽略 A3 的贡献。我们可以对此进行解释, 以便在数据表中没有第三个电影主题/电影爱好者类别的证据。这也意味着我们示例中的电影主题/电影爱好者们的整个空间是一个由科幻小说和法国艺术电影和爱好者跨越的二维空间。

图 4.2 泛函

矩阵的系统发育
机器学习中遇到的。



4.7 矩阵系统发育在第 2

章和第 3 章中,我们介绍了线性代数和解析几何的基础知识。在本章中,我们研究了矩阵和线性映射的基本特征。图 4.2 描绘了不同类型矩阵之间关系的系统发育树

(黑色箭头表示“是的子集”)以及我们可以对它们执行的覆盖操作(蓝色)。我们考虑所有实矩阵 $A \in R^{n \times m}$ 。对于非方阵(其中 $n \neq m$),SVD 始终存在,正如我们在本章中看到的那样。关注方阵 $A \in R^{n \times n}$,行列式告诉我们方阵是否具有逆矩阵,即它是否属于规则的可逆矩阵类。如果 $n \times n$ 方阵具有 n 个线性无关的特征向量,则该矩阵是无缺陷的,并且存在特征分解(定理 4.12)。我们知道重复的特征值可能会导致有缺陷的矩阵,无法对角化。

“系统发育”这个词描述了我们如何捕捉个体之间的关系或

群体,源自希腊语中的“部落”一词和“来源”。

非奇异矩阵和非缺陷矩阵是不一样的。例如,旋转矩阵将是可逆的(行列式非零)但在实数中不可对角化(不保证特征值是实数)。

我们进一步研究无缺陷方形 $n \times n$ 矩阵的分支。如果条件 $A^T A = AA^T = I$ 成立，则 A 是正规的。此外，如果更严格的条件满足 $A^T A = AA^T = I$ ，则 A 称为正交矩阵（见定义 3.8）。正交矩阵集是规则（可逆）矩阵的子集并满足 $A^{-1} = A^T$ 。

正规矩阵有一个经常遇到的子集，即对称的。对称矩阵只有矩阵 $S \in \mathbb{R}^{n \times n}$ 个实数，满足 $S = S^T$ 。特征值。对称矩阵的子集由正定矩阵 P 组成，对于所有 $x \in \mathbb{R}^n \setminus \{0\}$ ，满足 $x^T P x > 0$ 的条件。在这种情况下，存在唯一的 Cholesky 分解（定理 4.18）。正定矩阵只有正特征值并且总是可逆的（即，具有非零行列式）。

对称矩阵的另一个子集由对角矩阵 D 组成。对角矩阵在乘法和加法下是封闭的，但不一定形成一个群（只有当所有对角元素都非零以便矩阵可逆时才会出现这种情况）。一个特殊的对角矩阵是单位矩阵 I 。

4.8 进一步阅读本章的大部

分内容建立了基础数学并将它们与研究映射的方法联系起来，其中许多是机器学习的核心，在支持软件解决方案和几乎所有机器学习理论的构建块级别。使用行列式、特征谱和特征空间的矩阵表征为矩阵的分类和分析提供了基本特征和条件。这扩展到所有形式的数据表示和涉及数据的映射，以及判断此类矩阵上计算操作的数值稳定性（Press 等人，2007 年）。

行列式是“手动”求逆矩阵和计算特征值的基本工具。然而，对于除最小实例之外的几乎所有实例，高斯消元法的数值计算都优于行列式（Press et al., 2007）。尽管如此，行列式仍然是一个强大的理论概念，例如，根据行列式的符号获得关于基础方向的直觉。特征向量可用于执行基础更改以将数据转换为有意义的正交特征向量的坐标。同样，当我们计算或模拟随机事件时，矩阵分解方法（例如 Cholesky 分解）经常出现（Rubinstein 和 Kroese, 2016 年）。因此，Cholesky 分解使我们能够计算重新参数化技巧，其中我们希望对随机变量执行连续微分，例如，在变分自动编码器中（Jimenez Rezende 等人, 2014 年；Kingma 和 Welling, 2014 年）。

特征分解是使我们能够提取表征线性映射的有意义且可解释的信息的基础。

因此,特征分解是一类通用的机器学习算法的基础,称为谱方法,该算法执行正定核的特征分解。这些频谱分解方法包括统计数据分析的经典方法,例如:

主成分分析

- 主成分分析 (PCA (Pearson,1901 年) ,另见第 10 章) ,其中寻求解释数据中大部分可变性的低维子空间。

Fisher 判别分析

- Fisher 判别分析,旨在确定用于数据分类的分离超平面 (Mika et al., 1999)。

多维缩放

- 多维尺度(MDS) (Carroll 和 Chang,1970) 。

这些方法的计算效率通常来自于找到对称半正定矩阵的最佳秩 k 近似。更多现代谱方法的例子有不同的起源,但它们每个都需要计算正定核的特征向量和特征值,例如 Isomap (Tenenbaum 等人,2000 年)、拉普拉斯特征图 (Belkin 和 Niyogi,2003 年)、Hessian 特征图 (Donoho 和 Grimes,2003 年) 和谱聚类 (Shi 和 Malik,2000 年)。正如我们在这里通过 SVD 遇到的那样,这些核心计算通常由低秩矩阵近似技术 (Belabbas 和 Wolfe,2009) 提供支持。

等值图

拉普拉斯

特征图

Hessian 特征图谱

聚类

SVD 允许我们发现一些与

特征分解。但是,SVD 更普遍适用于非方阵和数据表。当我们想要通过近似执行数据压缩时,例如,而不是存储 $n \times m$ 值,而只是存储 $(n+m)k$ 值,或者当我们想要执行数据预处理,例如,去相关设计矩阵的预测变量 (Ormoneit 等人,2001 年)。SVD 对矩阵进行运算,我们可以将其解释为具有两个索引 (行和列) 的矩形数组。将类矩阵结构扩展到高维数组称为张量。事实证明,SVD 是对此类张量进行操作的更一般的分解族的特例 (Kolda 和 Bader,2009)。类似 SVD 的操作和张量的低秩近似是,例如, Tucker 分解(Tucker, 1966) 或 CP 分解(Carroll 和 Chang, 1970)。

塔克分 解

CP 分解

出于计算效率的原因,SVD 低秩近似经常用于机器学习。这是因为它减少了内存用量和我们需要对可能非常大的数据矩阵执行的非零乘法运算 (Trefethen 和 Bau III,1997)。此外,低秩近似用于对可能包含缺失值的矩阵进行运算,以及用于有损压缩和降维 (Moonen 和 De Moor,1995; Markovsky,2011)。

练习

4.1 使用拉普拉斯展开式计算行列式（使用第一行）
和 Sarrus 规则

$$\begin{array}{cccccc} & 1 & 3 & 5 & 2 & 4 & 6 \\ \text{一个=} & & & & & & \\ & 0 & 2 & 4 & & & . \end{array}$$

4.2 有效计算以下行列式：

$$\begin{array}{cccccc} & 0 & 1 & 2 & 0 & & \\ \text{2个} & -1 & 0 & 1 & 0 & & \\ & & 2 & & & 1 & 2 \\ \text{2个} & & & & & & . \\ -2 & 0 & 2 & -1 & 2 & 0 & 0 & 1 \\ \text{2个} & & & & & & 2 & \\ & & & & & & & 2 \end{array}$$

4.3 计算的特征空间

A.

$$\begin{array}{ccc} & 1 & 0 \\ \text{一个=} & & 1 & 1 \end{array}$$

b.

$$\begin{array}{ccc} & -2 & 2 \\ \text{乙=} & & 2 & 1 \end{array}$$

4.4 计算的所有特征空间

$$\begin{array}{cccccc} & 0 & -1 & 1 & & & \\ \text{一个=} & -1 & 1 & -2 & 3 & & \\ & 2 & -1 & 0 & 0 & & \\ & 1 & -1 & 1 & 0 & & . \end{array}$$

4.5 矩阵的对角化性与其可逆性无关。确定以下四个矩阵是否可对角化和/或可逆

$$\begin{array}{cccccc} 1 & 0 & & 1 & 0 & & \\ 0 & 1 & , & 0 & 0 & , & 1 & 1 & , & 0 & 1 & , & 0 & 0 \end{array}$$

4.6 计算下列变换矩阵的特征空间。它们可对角化吗？

A. 为了

$$\begin{array}{ccc} & 2 & 3 & 0 \\ \text{一个=} & 1 & 4 & 3 \\ & 0 & 0 & 1 \end{array}$$

b. 为了

$$\begin{array}{ccc} & 1 & 1 & 0 & 0 \\ \text{一个=} & 0 & 0 & 0 & 0 \\ & 0 & 0 & 0 & 0 \\ & 0 & 0 & 0 & 0 \end{array}$$

4.7 下列矩阵是否可对角化?如果是,确定它们的对角形式和变换矩阵对角的基。如果不是,请给出它们不可对角化的原因。

A.

$$\text{一个}= \begin{matrix} & & 1 \\ & 0 & -8 \\ -1 & & 4 \end{matrix}$$

b.

$$\begin{matrix} & 1 & 1 & 1 \\ \text{一个}= & 1 & 1 & 1 \\ & 1 & 1 & 1 \end{matrix}$$

C.

$$\begin{matrix} & 5 & 4 & 2 & 1 \\ \text{一个}= & 0 & 1 & -1 & -1 \\ & -1 & -1 & 3 & 0 \\ & & & 1 & -1 & 2 \end{matrix}$$

d.

$$\begin{matrix} & 5 & -6 & -6 \\ \text{一个}= & -1 & 4 & & 2 \\ & 3 & -6 & -4 \end{matrix}$$

4.8 求矩阵的SVD

$$\text{一个}= \begin{matrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{matrix}$$

4.9 求奇异值分解

$$\text{一个}= \begin{matrix} 2 & 2 \\ -1 & 1 \end{matrix}$$

4.10 求 rank-1 近似值

$$\text{一个}= \begin{matrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{matrix}$$

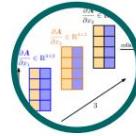
4.11 证明对于任何 $A \in \mathbb{R}^{m \times n}$, 矩阵 $A^T A$ 和 AA^T 具有相同的非零特征值。4.12 证明对于 $x = 0$ 定理 4.24 成立, 即证明

$$\max_{\|x\|=2} \frac{\|Ax\|_2}{\|x\|_2} = \sigma_1, \quad ,$$

其中 σ_1 是 $A \in \mathbb{R}^{m \times n}$ 的最大奇异值。

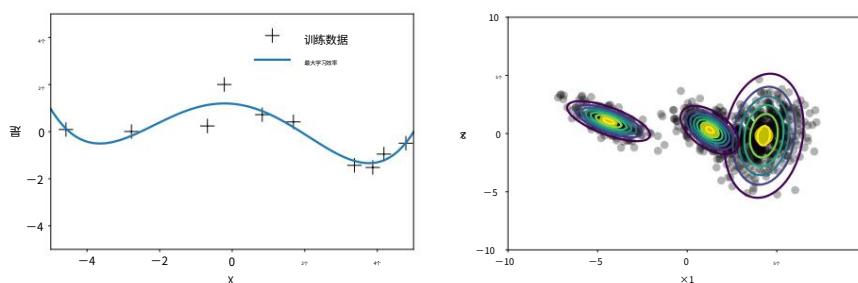
5个

矢量微积分



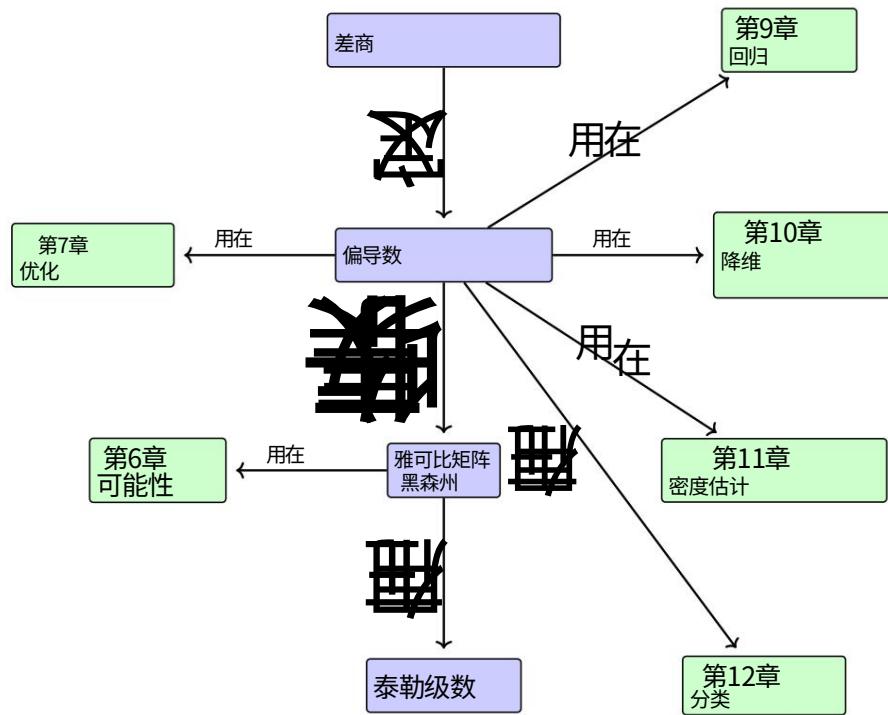
机器学习中的许多算法根据一组所需的模型参数优化目标函数,这些模型参数控制模型解释数据的程度:寻找好的参数可以表述为优化问题(参见第8.2节和第8.3节)。示例包括:(i)线性回归(见第9章),我们研究曲线拟合问题并优化线性权重参数以最大化可能性;(ii)用于降维和数据压缩的神经网络自动编码器,其中参数是每一层的权重和偏差,我们通过重复应用链式规则来最小化重构误差;(iii)用于建模数据分布的高斯混合模型(见第11章),我们优化每个混合成分的位置和形状参数以最大化模型的可能性。图5.1说明了其中一些问题,我们通常通过使用利用梯度信息的优化算法来解决这些问题(第7.1节)。图5.2概述了本章中的概念是如何相关的,以及它们如何与本书的其他章节联系起来。

本章的核心是函数的概念。函数 f 是使两个量相互关联的量。在本书中,这些量通常是输入 $x \in \text{RD}$ 和目标(函数值) $f(x)$,如果没有另外说明,我们假设它们是实值。这里RD是 f 的定义域,函数值 $f(x)$ 是 f 的像/余域。domain image/codomain图5.1矢量微积分在(a)回归(曲线拟合)和(b)密度估计(即数据分布建模)中起着核心作用。



(a) 回归问题:找到参数,使曲线很好地解释观察结果(交叉点)。
(b) 使用高斯混合模型进行密度估计:找到均值和协方差,以便可以很好地解释数据(点)。

图 5.2 本章介绍的概念的思维导图,以及它们在本书其他部分的使用时间。



2.7.3 节在线性函数的上下文中提供了更详细的讨论。我们经常写

$$f: \mathbb{R}^n \rightarrow \mathbb{R} \quad (5.1a)$$

$$\rightarrow f(x) \quad (5.1b)$$

指定一个函数,其中 (5.1a) 指定 f 是从 \mathbb{R}^n 到 \mathbb{R} 的映射,而 (5.1b) 指定输入 x 到函数值 $f(x)$ 的显式赋值。函数 f 恰好为每个输入 x 分配一个函数值 $f(x)$ 。

示例 5.1 回忆

点积作为内积的一个特例 (第 3.2 节)。

在前面的表示法中,函数 $f(x) = x \cdot x$, $x \in \mathbb{R}^2$ 将被指定为

$$f: \mathbb{R}^2 \rightarrow \mathbb{R} \quad (5.2a)$$

$$\rightarrow x_1 x_2 \circ x_1 x_2 \quad (5.2b)$$

在本章中,我们将讨论如何计算函数的梯度,这对于促进机器学习模型的学习通常是非常必不可少的,因为梯度指向最陡上升方向。所以,

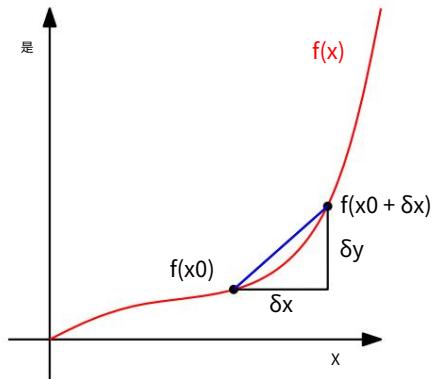


图 5.1 函数 f 在 x_0 和 $x_0 + \delta x$ 之间的平均斜率是通过 $f(x_0)$ 和 $f(x_0 + \delta x)$ 的割线 (蓝色) 的斜率,由 $\delta y / \delta x$ 给出。

矢量微积分是我们在机器学习中需要的基本数学工具之一。在本书中,我们假设函数是可微的。通过一些我们未在此处涵盖的其他技术定义,所提出的许多方法都可以扩展到子微分 (连续但在某些点不可微的函数)。我们将在第 7 章中查看对具有约束的函数情况的扩展。

5.1 单变量函数的微分

下面,我们简要回顾一下单变量函数的微分,这可能是高中数学所熟悉的。我们从单变量函数 $y = f(x)$, $x, y \in \mathbb{R}$ 的差商开始,我们随后将使用它来定义导数。

定义 5.1 (差商)。差商

差商

$$\frac{\delta y}{\delta x} := \frac{f(x + \delta x) - f(x)}{\delta x} \quad (5.3)$$

计算通过 f 的图形上两点的割线的斜率。在图 5.1 中,这些点的 x 坐标为 x_0 和 $x_0 + \delta x$ 。

如果我们假设 f 是一个线性函数,差商也可以被认为是 f 在 x 和 $x + \delta x$ 之间的平均斜率。在 $\delta x \rightarrow 0$ 的极限下,如果 f 是可微的,我们得到 f 在 x 处的正切。切线是 f 在 x 处的导数。

定义 5.2 (导数)。更正式地说,对于 $h > 0$, f 导数在 x 处的导数定义为极限

$$\frac{df}{dx} \lim_{h \rightarrow 0} h \frac{f(x + h) - f(x)}{h} := , \quad (5.4)$$

而图 5.1 中的正割变为切线。

f 的导数指向 f 的最陡上升方向。

例 5.2 (多项式的导数)

我们想计算 $f(x) = x^n \in N$ 的导数。我们可能已经知道答案是 nx^{n-1} , 但我想使用导数的定义作为差商的极限来推导这个结果。

使用 (5.4) 中导数的定义, 我们得到

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (5.5a)$$

$$= \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} \quad (5.5b)$$

$$= \lim_{h \rightarrow 0} \frac{\sum_{i=0}^n x^n - ih - x^n}{h} \quad (5.5c)$$

我们看到 x 并且我 $= \sum_{i=0}^n x^n - ih$ 通过从 1 开始求和, x^n 术语取消, 我们得到

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{\sum_{i=1}^n x^n - ih}{h} \quad (5.6a)$$

$$= \lim_{h \rightarrow 0} \sum_{i=1}^n x^n - ih \quad (5.6b)$$

$$= \lim_{h \rightarrow 0} \sum_{i=1}^n x^{n-1} + \sum_{i=2}^n x^{n-i-1} \quad (5.6c)$$

$$= \frac{n!}{1!(n-1)!x^{n-1}} = nx^{n-1} \quad (5.6d)$$

5.1.1 泰勒级数

泰勒级数将函数 f 表示为项的无穷和。这些项是使用在 x_0 处计算的 f 的导数确定的。

泰勒多项式
我们为所有 $t^0 := 1$
 $\in R$ 定义 t 。

定义 5.3 (泰勒多项式)。 $f: R \rightarrow R$ 在 x_0 的 n 次泰勒多项式定义为

$$T_n(x) := \sum_{k=0}^n \frac{f(k)(x_0)}{k!} (x - x_0)^k, \quad (5.7)$$

其中 $f(k)(x_0)$ 是 f 在 x_0 处的第 k 个导数 (我们假设存在) 和
 $\frac{(k)}{(k)}(x_0)$ 是多项式的系数。 $k!$

定义 5.4 (泰勒级数)。对于平滑函数 $f \in C^\infty, f: R \rightarrow R$, f 在 x_0 处的泰勒级数定义为

泰勒级数

$$T\infty(x) = \sum_{k=0}^{\infty} \frac{f(k)(x_0)(x - x_0)}{k!}^k. \quad (5.8)$$

对于 $x_0 = 0$, 我们得到麦克劳林级数作为 $f \in C^\infty$ 的特例, 即泰勒级数。如果 $f(x) = T\infty(x)$, 则 f 称为解析的。

f 是连续可微的

评论。通常, n 次泰勒多项式是函数的近似值, 不一定是多项式。泰勒多项式类似于 x_0 附近的 f 。 无限多次。

但是, n 次泰勒多项式是 $k = n$ 次多项式 f 的精确表示, 因为所有导数 $f(i)$ \diamond

麦克劳林级数
分析的

, $i > k$ 消失。

例 5.3 (泰勒多项式)

我们考虑多项式

$$f(x) = x \quad (5.9)$$

并寻找在 $x_0 = 1$ 处计算的泰勒多项式 T_6 。我们首先计算 $k = 0$ 的系数 $f(k)(1), \dots, 6$:

$$f(1) = 1 f \quad (5.10)$$

$$(1) = 4 f' \quad (5.11)$$

$$'(1) = 12 f(3) \quad (5.12)$$

$$(1) = 24 f(4)(1) \quad (5.13)$$

$$= 24 f(5)(1) = \quad (5.14)$$

$$0 f(6)(1) = 0 \quad (5.15)$$

$$(5.16)$$

因此, 所需的泰勒多项式是

$$T_6(x) = \sum_{k=0}^{6} \frac{f(k)(x_0)(x - x_0)}{k!}^k \quad (5.17a)$$

$$= 1 + 4(x - 1) + 6(x - 1)^2 + 4(x - 1)^3 + (x - 1)^4 + 0. \quad (5.17b)$$

乘以并重新安排收益率

$$T_6(x) = (1 - 4 + 6 - 4 + 1) + x(4 - 12 + 12 - 4) \quad (5.18a)$$

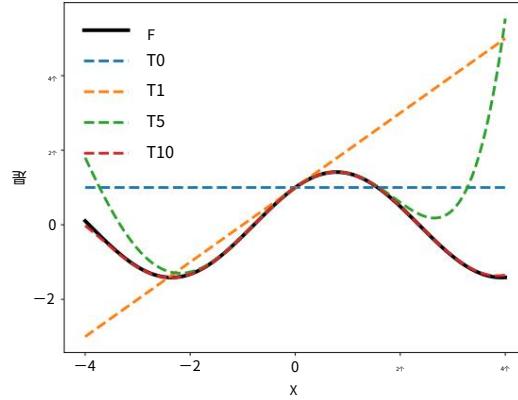
$$2 + x(6 - 12 + 6) + x = x(4 - 4) + x \quad (5.18b)$$

$$4 = x f(x), \quad (5.18b)$$

即, 我们获得了原始函数的精确表示。

图 5.2 泰勒多项式。
 原始函数 $f(x) = \sin(x) + \cos(x)$ (黑色实线)由 $x_0 = 0$ 附近的泰勒多项式 (虚线)逼近。

高阶泰勒多项式可以
更好、更全局
地逼近函数 f 。



T_{10} 已经与
 $[-4, 4]$ 中的 f
相似。

例 5.4 (泰勒级数)

考虑图 5.2 中给出的函数

$$f(x) = \sin(x) + \cos(x) \in C^\infty. \quad (5.19)$$

我们寻求 f 在 $x_0 = 0$ 处的泰勒级数展开, 这是 f 的麦克劳林级数展开。我们得到以下导数:

$$f(0) = \sin(0) + \cos(0) = 1 \quad (5.20)$$

$$f'(0) = \cos(0) - \sin(0) = 1 \quad f''(0) = \quad (5.21)$$

$$-\sin(0) - \cos(0) = -1 \quad f(3)(0) = - \quad (5.22)$$

$$\cos(0) + \sin(0) = -1 \quad f(4)(0) = \sin(0) + \quad (5.23)$$

$$\cos(0) = f(0) = 1 \quad (5.24)$$

⋮

我们可以在这里看到一个模式: 我们的泰勒级数中的系数只有 ± 1 (因为 $\sin(0) = 0$) , 每个系数在切换到另一个系数之前出现两次。此外, $f(k+4)(0) = f(k)(0)$ 。

因此, f 在 $x_0 = 0$ 处的完整泰勒级数展开由 $f(k)(x_0)(x - x_0)^k$ 给出!

$$T_\infty(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(0)}{k!} x^k \quad (5.25a)$$

$$= 1 + x - \frac{1}{2!x} + \frac{1}{3!x} - \frac{1}{4!x} + \frac{1}{5!x} - \dots \quad (5.25b)$$

$$= 1 - \frac{1}{2!x} + \frac{1}{4!x} - \frac{1}{6!x} + \dots + x - \frac{1}{3!x} + \frac{1}{5!x} - \dots \quad (5.25c)$$

$$= \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!x} + \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!x} \quad (5.25d)$$

$$= \cos(x) + \sin(x), \quad (5.25e)$$

我们使用幂级数表示的地方

幂级数表示

$$\text{余弦 } (x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}, \quad (5.26)$$

$$\text{罪恶 } (x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!}. \quad (5.27)$$

图 5.2 显示了 $n = 0, 1, 5, 10$ 时相应的第一泰勒多项式 T_n 。

评论。泰勒级数是幂级数的特例

$$f(x) = \sum_{k=0}^{\infty} a_k (x-c)^k \quad (5.28)$$

其中 a_k 是系数, c 是常数, 具有定义 5.4 中的特殊形式。 ◇

5.1.2 微分规则

在下文中, 我们简要说明基本的微分规则, 其中我们用 f' 表示 f 的导数

产品规则: $(f(x)g(x))' = f'(x)g(x) + f(x)g'(x)$ (5.29)

商法则: $\frac{f(x)}{g(x)}' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$ (5.30)

求和规则: $(f(x) + g(x))' = f'(x) + g'(x)$ (5.31)

链式规则: $(g \circ f)'(x) = g'(f(x))f'(x)$ (5.32)

里, $g \circ f$ 表示函数组合 $x \rightarrow f(x) \rightarrow g(f(x))$ 。

例 5.5 (链式法则)

让我们计算函数 $h(x) = (2x + 1)^4$ 的导数, 使用链式规则。和

$$h(x) = (2x + 1)^4 = g(f(x)), f(x) = 2x \quad (5.33)$$

$$+ 1, \quad (5.34)$$

$$g(f) = f^4, \quad (5.35)$$

我们得到 f 和 g 的导数为

$$F'(x) = 2, \quad (5.36)$$

$$G(f) = 4f^3, \quad (5.37)$$

使得 h 的导数为

$$H'(x) = \text{克}(f)f'(x) = (4f''(x)) \cdot 2^{(5.34)} 4(2x+1)3 \cdot 2 = 8(2x+1)3, \quad (5.38)$$

其中我们使用链式法则 (5.32) 并代入 $f'(f)$ 的定义。在 (5.34) 在克

5.2 偏微分和梯度

5.1 节中讨论的微分适用于标量变量 $x \in R$ 的函数 f 。在下文中, 我们考虑函数 f 取决于一个或多个变量 $x \in R^n$ 的一般情况, 例如, $f(x) = f(x_1, x_2)$ 。导数对多变量函数的推广就是梯度。

我们通过一次改变一个变量并保持其他变量不变来找到函数 f 相对于 x 的梯度。梯度就是这些偏导数的集合。

定义 5.5 (偏导数) 对于函数 $f: R^n \rightarrow R$, $x \mapsto f(x)$, $x \in R^n$ of n variables x_1, \dots, x_n 我们将偏导数定义为

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2, \dots, x_n) - f(x_1, x_2, \dots, x_n)}{h} \\ &\vdots \\ \frac{\partial f}{\partial x_n} &= \lim_{h \rightarrow 0} \frac{f(x_1, \dots, x_{n-1}, x_n + h) - f(x_1, \dots, x_{n-1}, x_n)}{h} \end{aligned} \quad (5.39)$$

并将它们收集到行向量中

$$\nabla_x f = \text{grad} f = dx \quad \frac{df}{dx} = \frac{\partial f(x)}{\partial x_1} \quad \frac{\partial f(x)}{\partial x_2} \quad \dots \quad \frac{\partial f(x)}{\partial x_n} \quad \in R^{1 \times n}, \quad (5.40)$$

其中 n 是变量的数量, 1是 f 的图像/范围/余域的维度。在这里, 我们定义了列向量 $x = [x_1, \dots, x_n]$

(5.40) 中的行向量称为 f 的梯度或Jacobian $\in R^n$, 是 5.1 节导数的推广。

坡度

雅可比矩阵

评论。雅可比行列式的这个定义是向量值函数的雅可比行列式一般定义的一个特例, 作为偏导数的集合。我们将在 5.3 节中回到这一点。 ◇

我们可以使用标量的结果

微分: 每个偏导数都是关于标量的导数。

例 5.6 (使用链式法则的偏导数)

对于 $f(x, y) = (x + 2y)$, 我们得到偏导数 $\frac{\partial f(x, y)}{\partial x} = 2(x + 2y)$

$$\frac{\partial}{\partial x}(x + 2y) = 2(x + 2y), \quad (5.41)$$

$$\frac{\partial f(x, y)}{\partial x} = 2(x + 2y) \quad \frac{\partial f}{\partial y} = 2(x + 2y) \quad (5.42)$$

其中我们使用链式法则 (5.32) 来计算偏导数。

备注 (渐变作为行向量)。将梯度向量定义为列向量在文献中并不少见,遵循向量通常是列向量的约定。我们将梯度向量定义为行向量的原因有两个:首先,我们可以一致地将梯度推广到向量值函数 $f: R^n \rightarrow R^m$ (然后梯度变成矩阵)。其次,我们可以立即应用多元链式法则,而无需关注梯度的维度。我们将在 5.3 节中讨论这两点。 ◇

例 5.7 (渐变)

对于 $f(x_1, x_2) = x \in R$, 偏导数 x (即对 x_1 和 x_2 的导数) 为 $\frac{\partial f(x_1, x_2)}{\partial x_1} = 2x_1x_2 + x$ $\frac{\partial f(x_1, x_2)}{\partial x_2} = 2x_1 + 3x_1x_2$

$$(5.43)$$

$$2 = x_1 + 3x_1x_2 \quad (5.44)$$

然后梯度是

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x_1, x_2)}{\partial x_1} & \frac{\partial f(x_1, x_2)}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 2x_1x_2 + x \\ 2x_1 + 3x_1x_2 \end{pmatrix} \in R^{1 \times 2}. \quad (5.45)$$

5.2.1 偏微分的基本规则

在多变量情况下,我们从学校知道的 $x \in R^n$ (例如, 基本微分规则)

求和法则、乘法法则、链式法则(另见第 5.1.2 节)仍然适用。然而,当我们计算关于向量 $x \in R^n$ 的导数时,我们需要注意:我们的梯度现在涉及向量和矩阵,矩阵乘法是不可交换的(第 2.2.1 节),即顺序很重要。

乘积法则: (fg)

$= f'g + fg'$,

求和法则: $(f$

$+ g) = f' + g'$,

链式法则:

$(g(f))' = g'(f'f)$

以下是一般乘积法则、求和法则和链式法则:

$$\text{产品规则: } \frac{\partial}{\partial x} f(x) \frac{\partial f}{\partial x} \frac{\partial g}{\partial x} = g(x) + f(x) \frac{\partial g}{\partial x} \quad (5.46)$$

$$\text{求和规则: } \frac{\partial}{\partial x} \frac{\partial f}{\partial x} \frac{\partial g}{\partial x} \quad (5.47)$$

$$\text{链式规则: } \frac{\partial}{\partial x} (g \circ f)(x) = \partial x \frac{\partial}{\partial f} g(f(x)) = \frac{\partial g}{\partial f \partial x} \quad (5.48)$$

这只是一种直觉，但在数学上并不正确，因为偏导数不是分数。

让我们仔细看看链式法则。链式法则 (5.48) 在某种程度上类似于矩阵乘法的规则，其中我们说相邻维度必须匹配才能定义矩阵乘法；请参阅第 2.2.1 节。如果我们从左到右，链式法则表现出相似的性质： ∂f 出现在第一个因素的“分母”和第二个因素的“分子”中。如果我们将这些因子相乘，乘法是有定义的，即 ∂f 的维度匹配，而 ∂f “抵消”，这样 $\partial g / \partial x$ 仍然存在。

5.2.2 链式法则

考虑两个变量 x_1, x_2 的函数 $f: R^2 \rightarrow R$ 。此外， $x_1(t)$ 和 $x_2(t)$ 本身就是 t 的函数。为了计算 f 相对于 t 的梯度，我们需要对多元函数应用链式法则 (5.48) 作为

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial t} = \frac{\partial f \partial x_1}{\partial x_1 \partial t} + \frac{\partial f \partial x_2}{\partial x_2 \partial t}, \quad (5.49)$$

其中 d 表示梯度和 ∂ 偏导数。

例 5.8 考虑 $f(x_1, x_2) = x_1 \sin t + 2x_2$ ，其中 $x_1 = \sin t$ 和 $x_2 = \text{成本}$ ，则

$$\frac{\partial f}{\partial t} = \frac{\partial f \partial x_1}{\partial t \partial x_1} \frac{\partial f \partial x_2}{\partial t \partial x_2} + \frac{\partial f \partial x_1}{\partial t \partial x_1} \frac{\partial x_1}{\partial t} \quad (5.50a)$$

$$\frac{\text{成本}}{\partial t} = 2 \sin t + 2 \frac{\partial x_2}{\partial t} \quad (5.50b)$$

$$\frac{\partial t}{\partial t} = 2 \sin t \text{ 成本} - 2 \sin t = 2 \sin t(\text{成本} - 1) \quad (5.50 \text{ 摄氏度})$$

是 f 关于 t 的相应导数。

如果 $f(x_1, x_2)$ 是 x_1 和 x_2 的函数，其中 $x_1(s, t)$ 和 $x_2(s, t)$ 本身是两个变量 s 和 t 的函数，则链式法则产生偏导数

$$\frac{\partial f}{\partial s} = \frac{\partial f \partial x_1}{\partial s \partial x_1} \frac{\partial f \partial x_2}{\partial s \partial x_2} + \frac{\partial x_1}{\partial s} \frac{\partial x_2}{\partial s}, \quad (5.51)$$

$$\frac{\partial f}{\partial t} = \frac{\partial f \partial x_1}{\partial t \partial x_1} \frac{\partial f \partial x_2}{\partial t \partial x_2} + \frac{\partial x_1}{\partial t} \frac{\partial x_2}{\partial t}, \quad (5.52)$$

5.3 向量值函数的梯度

149

梯度是通过矩阵乘法得到的

$$\begin{aligned} \frac{df}{d(s,t)} &= \frac{\partial f}{\partial x} \frac{\partial x}{\partial (s,t)} = \underbrace{\frac{\partial f}{\partial x_1}}_{=} \underbrace{\frac{\partial f}{\partial x_2}}_{=} \begin{matrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial s} & \frac{\partial x_2}{\partial t} \end{matrix} . \quad (5.53) \\ &= \frac{\partial f}{\partial x} \begin{matrix} \frac{\partial s}{\partial x} & \frac{\partial t}{\partial x} \\ \end{matrix} = \frac{\partial x}{\partial (s,t)} \end{aligned}$$

这种仅将链式法则写为矩阵乘法的紧凑方式如果将梯度定义为行向量,则链式法则才有意义。否则,我们将需要开始转置梯度以匹配矩阵维度。

写成
矩阵
乘法。

只要梯度是向量或矩阵,这可能仍然很简单;然而,当梯度变成张量时(我们将在下面讨论),转置不再是一件小事。

备注(验证梯度实现的正确性)。当在计算机程序中对梯度的正确性进行数值检查时,可以利用偏导数的定义作为相应差商的极限(见(5.39)):当我们计算梯度检查梯度并实现它们时,我们可以使用有限差分以数值方式测试我们的计算和实现:我们选择较小的值 h (例如, $h = 10^{-4}$)并将(5.39)的有限差分近似值与我们的(解析)梯度实现进行比较。如果误差很小,我们的梯度实现可能是正确的。“小” $< 10^{-6}$ 可能意味着

$\frac{=(dhi-dfi)}{=(dhi+dfi)}$, 其中 dhi 是有限差分近似值, dfi 是 f 相对于第 i 个变量 x_i 的解析梯度。 ◇

5.3 向量值函数的梯度

到目前为止,我们讨论了函数 $f: R^n \rightarrow R$ 映射到实数的偏导数和梯度。下面,我们将梯度的概念推广到向量值函数(向量场) $f: R^n \rightarrow R^m$,其中 $n \geq 1$ 且 $m > 1$ 。

对于函数 $f: R^n \rightarrow R^m$ 和向量 $x = [x_1, \dots, x_n]^\top \in R^n$,这相应的函数值向量为

$$f(x) = \begin{matrix} f_1(x) \\ \vdots \\ f_m(x) \end{matrix} \in R^m . \quad (5.54)$$

以这种方式编写向量值函数允许我们将向量值函数 $f: R^n \rightarrow R^m$ 视为函数向量 $[f_1, \dots, f_m]$, $f_i: R^n \rightarrow R$ 映射到 R 。每个 f_i 的微分规则正是我们在5.2节中讨论的规则。

因此,向量值函数 f 的偏导数: $R^n \rightarrow R^m$ 关于 $x_i \in R$, $i = 1, \dots, n$, 作为向量给出

$$\frac{\partial f}{\partial x_i} = \begin{matrix} \frac{\partial f_1}{\partial x_i} \\ \vdots \\ \frac{\partial f_m}{\partial x_i} \end{matrix} = \begin{matrix} h \\ \vdots \\ \text{生命值} \rightarrow 0 \end{matrix} \begin{matrix} f_1(x_1, \dots, x_{i-1}, x_i + h, x_{i+1}, \dots, x_n) - f_1(x) \\ \vdots \\ f_m(x_1, \dots, x_{i-1}, x_i + h, x_{i+1}, \dots, x_n) - f_m(x) \end{matrix} \in R^m.$$
(5.55)

由(5.40)可知, f 关于向量的梯度是偏导数的行向量。在 (5.55) 中, 每个偏导数 $\partial f / \partial x_i$ 本身就是一个列向量。因此, 我们通过收集这些偏导数获得 $f: R^n \rightarrow R^m$ 相对于 $x \in R^n$ 的梯度:

$$df(x) = \partial f(x) dx \begin{matrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{matrix} \quad (5.56a)$$

$$= \begin{matrix} \frac{\partial f_1(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f_m(x)}{\partial x_1} \end{matrix} \dots \begin{matrix} \frac{\partial f_1(x)}{\partial x_n} \\ \vdots \\ \frac{\partial f_m(x)}{\partial x_n} \end{matrix} \in R^{m \times n}. \quad (5.56b)$$

定义 5.6 (雅可比矩阵) 向量值函数 $f: R^n \rightarrow R^m$ 的所有一阶偏导数的集合称为 Jacobian。Jacobian J 是一个 $m \times n$ 矩阵, 我们定义和排列如下:

雅可比矩阵
函数的梯度

$f: R^n \rightarrow R^m$ 是大小矩阵
 $m \times n$

$$J = \nabla_x f = dx \begin{matrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{matrix} = \begin{matrix} \frac{\partial f_1(x)}{\partial x_1} & \dots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(x)}{\partial x_1} & \dots & \frac{\partial f_m(x)}{\partial x_n} \end{matrix} \quad (5.57)$$

$$= \begin{matrix} \frac{\partial f_1(x)}{\partial x_1} & \dots & \frac{\partial f_1(x)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(x)}{\partial x_1} & \dots & \frac{\partial f_m(x)}{\partial x_n} \end{matrix}, \quad (5.58)$$

$$x = \begin{matrix} x_1 \\ \vdots \\ x_n \end{matrix}, J(i, j) = \frac{\partial f_i}{\partial x_j}. \quad (5.59)$$

作为 (5.58) 的特例, 函数 $f: R^n \rightarrow R^1$ 向量 $x \in R^n$ 到一个标量上 (例如, $f(x) =$ 那是一个 $\sum_{i=1}^n$ 拥有一个雅可比行列式个行向量 (维度 $1 \times n$ 的矩阵); 参见 (5.40))。

分子布局

评论。在本书中, 我们使用导数的分子布局, 即 $f \in R^m$ 对 $x \in R^n$ 的导数 df/dx 是一个 $m \times n$ 矩阵, 其中的元素定义行和 x 的元素定义相应雅可比行列式的列; 参见 (5.58)。那里

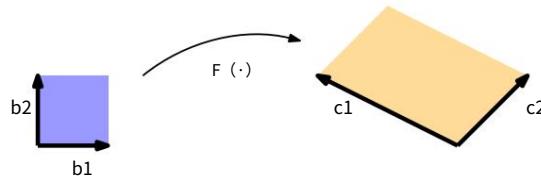


图 5.1 的行列式

f 的雅可比矩阵可用于计算蓝色之间的放大镜和橙色区域。

也存在分母布局,它是分子分母布局布局的转置。在本书中,我们将使用分子布局。



我们将在第 6.7 节中看到如何在概率分布的变量变化方法中使用雅可比行列式。由于变量转换而导致的缩放量由行列式提供。

在 4.1 节中,我们看到行列式可用于计算平行四边形的面积。如果给定两个向量 $b_1 = [1, 0]$, $b_2 = [0, 1]$ 作为单位正方形 (蓝色; 见图 5.1) 的边, 则该正方形的面积为

$$\text{检测 } \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = 1。 \quad (5.60)$$

如果我们取一个平行四边形, 其边为 $c_1 = [-2, 1]$, $c_2 = [1, 1]$ (图 5.1 中的橙色), 则其面积为行列式的绝对值 (参见第 4.1 节)

$$\text{检测 } \begin{vmatrix} -2 & 1 \\ 1 & 1 \end{vmatrix} = |-3| = 3, \quad (5.61)$$

也就是说, 它的面积恰好是单位正方形面积的三倍。

我们可以通过找到一个将单位正方形转换为另一个正方形的映射来找到这个比例因子。在线性代数术语中, 我们有效地执行了从 (b_1, b_2) 到 (c_1, c_2) 的变量转换。在我们的例子中, 映射是线性的, 并且这个映射的行列式的绝对值恰好为我们提供了我们正在寻找的比例因子。

我们将描述两种识别此映射的方法。首先, 我们利用映射是线性的, 这样我们就可以使用第 2 章中的工具来识别这个映射。其次, 我们将使用本章讨论的工具找到使用偏导数的映射。

方法一 为了开始使用线性代数方法, 我们将 $\{b_1, b_2\}$ 和 $\{c_1, c_2\}$ 都确定为 R^2 的基 (请参阅第 2.6.1 节进行回顾)。我们有效执行的是基从 (b_1, b_2) 到 (c_1, c_2) 的变化, 我们正在寻找实现基变化的变换矩阵。使用第 2.7.2 节的结果, 我们将所需的基础变化矩阵确定为

$$J = \begin{vmatrix} -2 & 1 \\ 1 & 1 \end{vmatrix}, \quad (5.62)$$

这样 $Jb_1 = c_1$ 和 $Jb_2 = c_2$ 确定的绝对值

J 的 nant , 它产生我们正在寻找的比例因子, 被给出为 $|\det(J)| = 3$, 即 (c_1, c_2) 所跨越的正方形的面积是 (b_1, b_2) 所跨越的面积的三倍。

性变换 (在第 6.7 节中变得相关) , 我们遵循使用偏导数的更通用的方法。

对于这种方法, 我们考虑执行变量转换的函数 $f: R^2 \rightarrow R^2$ 。在我们的示例中, f 将任何向量 $x \in R^2$ 相对于 (b_1, b_2) 的坐标表示映射到坐标表示 $y \in R^2$ 相对于 (c_1, c_2) 。我们想要识别映射, 以便我们可以计算面积 (或体积) 在被 f 变换时如何变化。为此, 我们需要找出如果稍微修改 x , $f(x)$ 会如何变化。雅可比矩阵 $df \in R^2 \times 2$ 正好回答了这个问题。因为我们可以写

\bar{dx}

$$y_1 = -2x_1 + x_2 \quad (5.63)$$

$$y_2 = x_1 + x_2 \quad (5.64)$$

我们得到 x 和 y 之间的函数关系, 这使我们能够得到偏导数 $\partial y_1 / \partial x_1$

$$\frac{\partial y_1}{\partial x_2} = -2, \quad \frac{\partial y_1}{\partial x_1} = 1, \quad \frac{\partial y_2}{\partial x_1} = 1, \quad \frac{\partial y_2}{\partial x_2} = 1 \quad (5.65)$$

并将雅可比行列式组成为 ∂y

$$J = \begin{vmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} \end{vmatrix} = \begin{vmatrix} -2 & 1 \\ 1 & 1 \end{vmatrix}. \quad (5.66)$$

在几何上, 雅可比矩阵代表我们正在寻找的坐标变换。如果坐标变换是线性的 (如我们的例子), 则它是精确的, 并且 (5.66) 准确地恢复 (5.62) 中的基变化矩阵。如果坐标变换是非线性的, 则雅可比行列式在局部用线性变换逼近该非线性变换。雅可比行列式的绝对值 $|\det(J)|$ 是坐标变换时面积或体积缩放的系数。我们的案例产生 $|\det(J)| = 3$ 。

面积或体积。

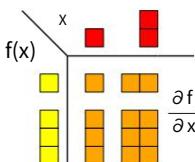
雅可比行列

式

雅可比行列式和变量变换将变为

当我们转换随机变量和概率分布时, 与第 6.7 节相关。在使用重新参数化技巧 (也称为无限扰动分析) 训练深度神经网络的背景下, 这些转换与机器学习极为相关。

图 5.2 (偏)
导数的维数。



在本章中, 我们遇到了函数的导数。图 5.2 总结了这些导数的维度。如果 $f: R \rightarrow R$ 梯度只是一个标量 (左上角的条目)。对于 $f: RD \rightarrow R$, 梯度是一个 $1 \times D$ 行向量 (右上角的条目)。对于 $f: R \rightarrow RE$, 梯度是一个 $E \times 1$ 列向量, 对于 $f: RD \rightarrow RE$, 梯度是一个 $E \times D$ 矩阵。

例 5.9 (向量值函数的梯度)

我们被赋予

$$f(x) = \text{轴} \quad , \quad f(x) \in \mathbb{R}^M, A \in \mathbb{R}^{M \times N}, \quad x \in \mathbb{R}^n.$$

为了计算梯度 df/dx , 我们首先确定 df/dx 的维度: 因为 $f: \mathbb{R}^N \rightarrow \mathbb{R}^M$, 所以 $df/dx \in \mathbb{R}^{M \times N}$ 。其次, 为了计算梯度, 我们确定 f 对每个 x_j 的偏导数:

$$\text{菲}(x) = \begin{matrix} \text{否} \\ j=1 \end{matrix} A_{ij} x_j \Rightarrow \frac{\partial f_i}{\partial x_j} = A_{ij} \quad (5.67)$$

我们收集雅可比矩阵中的偏导数并获得梯度

$$\frac{\partial f}{\partial x} = \begin{matrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & & \vdots \\ \frac{\partial f_M}{\partial x_1} & \cdots & \frac{\partial f_M}{\partial x_N} \end{matrix} = \begin{matrix} A_{11} & \cdots & A_{1N} \\ \vdots & & \vdots \\ A_{M1} & \cdots & A_{MN} \end{matrix} = A \in \mathbb{R}^{M \times N}. \quad (5.68)$$

例 5.10 (链式法则)

考虑函数 $h: \mathbb{R} \rightarrow \mathbb{R}$, $h(t) = (f \circ g)(t)$ 其中

$$f: \mathbb{R} \rightarrow \mathbb{R} \quad (5.69)$$

$$R \rightarrow R \quad f(x) = x^2 \quad (5.70)$$

$$\exp(x_1 x_2), \quad (5.71)$$

$$x = \begin{matrix} x_1 \\ x_2 \end{matrix} = \begin{matrix} \text{成本} \\ \text{清爽} \end{matrix} = \begin{matrix} \text{克 (吨)} \\ \text{清爽} \end{matrix} \quad (5.72)$$

并计算 h 相对于 t 的梯度。由于 $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ 和 $g: \mathbb{R} \rightarrow \mathbb{R}^2$ 我们注意到 $\partial f \in \mathbb{R}^{2 \times 1}$

$$\frac{\partial f}{\partial t} \in \mathbb{R}^{1 \times 2}, \quad \frac{\partial g}{\partial t} \in \mathbb{R}^{2 \times 1}. \quad (5.73)$$

所需的梯度是通过应用链式法则计算的:

$$\frac{\partial h}{\partial t} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial t} = \begin{matrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{matrix} \begin{matrix} \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial t} \end{matrix} = \begin{matrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{matrix} \begin{matrix} \frac{\partial \exp(x_1 x_2)}{\partial x_1} \\ \frac{\partial \exp(x_1 x_2)}{\partial x_2} \end{matrix} \quad (5.74a)$$

$$= \exp(x_1 x_2) \begin{matrix} x_1^2 & 2x_1 x_2 \end{matrix} \frac{\partial \exp(x_1 x_2)}{\partial x_1} \frac{\partial \exp(x_1 x_2)}{\partial x_2} \quad (5.74b)$$

$$= \exp(x_1 x_2) \times 2(\cos t - \sin t) + 2x_1 x_2 (\sin t + t \cos t), \quad (5.74c)$$

其中 $x_1 = t$ 成本和 $x_2 = \sin t$; 见 (5.72)。

我们将在第 9 章的线性回归背景下更详细地讨论这个模型,其中我们需要最小二乘损失 L 关于参数 θ 的导数。

最小二乘损失

示例 5.11 (线性模型中最小二乘损失的梯度)

让我们考虑线性模型

$$y = \Phi\theta \quad , \quad (5.75)$$

其中 $\theta \in RD$ 是参数向量, $\Phi \in RN \times D$ 是输入特征, $y \in RN$ 是相应的观测值。我们定义函数

$$L(e) := \|e\|_2^2, \quad (5.76)$$

$$e(\theta) := y - \Phi\theta. \quad (5.77)$$

我们寻找 $\frac{\partial L}{\partial \theta}$, 为此, 我们将使用链式法则。L 称为最小二乘损失函数。

在开始计算之前, 我们将梯度的维数确定为

$$\frac{\partial L}{\partial \theta} \in R^{1 \times \text{深}}. \quad (5.78)$$

链式法则允许我们将梯度计算为

$$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial e} \frac{\partial e}{\partial \theta}, \quad (5.79)$$

`dLdtheta =`

其中第 d 个元素由下式给出

$$\frac{\partial L}{\partial \theta}[1, d] = \sum_{n=1}^N \frac{\partial L}{\partial e[n]} \frac{\partial e[n]}{\partial \theta} \quad (5.80)$$

我们知道 $\|e\|_2^2 = e^\top e$ (见第 3.2 节) 并确定

$$\frac{\partial L}{\partial e} = 2e \in R^{1 \times N}. \quad (5.81)$$

此外, 我们得到

$$\frac{\partial e}{\partial \theta} = -\Phi \in R^{N \times D}, \quad (5.82)$$

这样我们想要的导数是

$$\frac{\partial L}{\partial \theta} = -2e^\top \Phi \partial \theta \stackrel{(5.77)}{=} -2(y - \Phi\theta)^\top \Phi \in R^{1 \times \text{深}}. \quad (5.83)$$

评论。如果不使用链式法则, 我们会通过立即查看函数来获得相同的结果

$$L2(\theta) := \|y - \Phi\theta\|_2^2 = (y - \Phi\theta)^\top (y - \Phi\theta). \quad (5.84)$$

这种方法对于像 L2 这样的简单函数仍然实用, 但对于深度函数组合变得不切实际。 ◇

5.4 矩阵的梯度

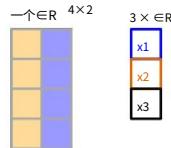
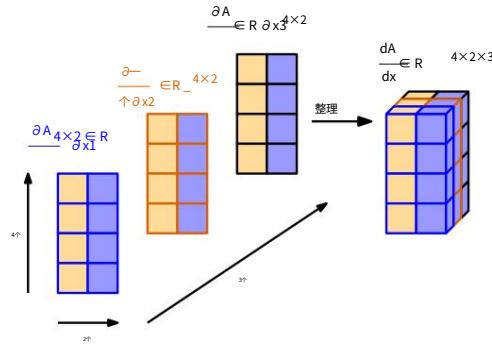


图 5.3 矩阵相
对于向量的梯度计算
的可视化。
我们有兴趣

偏导数：

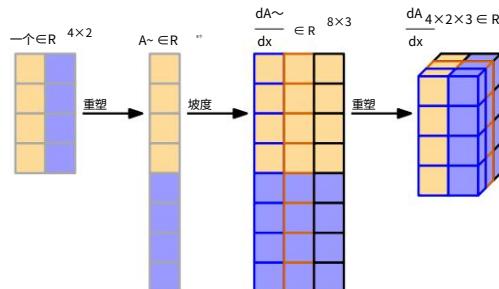
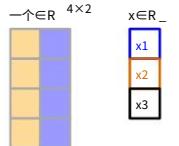


(a) 方法 1: 我们计算偏导数 $\frac{\partial A}{\partial A}$, 其中每个都是一个 4×2 矩阵, col $\frac{\partial x_1}{\partial x_2}$, $\frac{\partial x_3}$, 将它们放在一个 $4 \times 2 \times 3$ 张量中。

计算 $A \in \mathbb{R}^{4 \times 2}$ 相
对于向量 $x \in \mathbb{R}^3$ 的梯度。我们知道
梯度 $\in \mathbb{R}^{4 \times 2 \times 3}$ 。我们

$\frac{dA}{dx}$
遵循两种等效
的方法到达那
里：(a) 将偏导数整理成雅
可比行列式

张量；
(b) 将矩阵展平为向量, 计
算雅可比矩阵, 重新整形为
雅可比张量。



(b) 方法 2: 我们将 $A \in \mathbb{R}^{4 \times 2}$ 重塑 (展平) 为向量 dA or $A \in \mathbb{R}^8$ 。然后, 我们
计算梯度 $\in \mathbb{R}^{8 \times 3}$ 。如上所述, 我们通过重塑这个梯度来获得梯度张量。

5.4 矩阵的梯度

我们会遇到需要对向量 (或其他矩阵) 求矩阵梯度的情况, 这会导致多维张量。我们可以把这个张量想象成一个多维数组

我们可以想到一个
张量作为多维
大批。

收集偏导数。例如,如果我们计算一个 $m \times n$ 矩阵 A 相对于一个 $p \times q$ 矩阵 B 的梯度,得到的雅可比矩阵将是 $(m \times n) \times (p \times q)$,即一个四维张量 J,其项为 $J_{ijkl} = \partial A_{ij} / \partial B_{kl}$ 。

由于矩阵表示线性映射,我们可以利用在 $m \times n$ 矩阵的空间 $R^{m \times n}$ 和 mn 向量的空间 R^{mn} 之间存在向量空间同构(线性,可逆映射)这一事实。

因此,我们可以将矩阵重新整形为长度分别为 mn 和 pq 的向量。使用这些 mn 个向量的梯度产生大小为 $mn \times pq$ 的雅可比矩阵。图 5.3 形象化了这两种方法。在实际应用中,通常需要将矩阵重新整形为向量并继续使用此雅可比矩阵:链式法则(5.48)归结为简单的矩阵乘法,而在雅可比张量的情况下,我们将需要更多关注我们需要哪些维度

矩阵可以转化为
通过堆叠的列向量

矩阵

(“扁平化”)

总结一下。

例 5.12 (向量相对于矩阵的梯度)

让我们考虑以下示例,其中

$$f \in R^M, A \in R^M, f = Ax^{\text{否}}, x \in R^{\text{否}} \quad (5.85)$$

以及我们在哪里寻找梯度 df/dA 。让我们重新开始确定梯度的维度为

$$\frac{df}{dA} \in R^{M \times (M \times N)} \quad (5.86)$$

根据定义,梯度是偏导数的集合:

$$\frac{df}{dA} = \begin{bmatrix} \frac{\partial f_1}{\partial A} \\ \vdots \\ \frac{\partial f_M}{\partial A} \end{bmatrix}, \quad \frac{\partial f_i}{\partial A} \in R^{1 \times (M \times N)} \quad . \quad (5.87)$$

要计算偏导数,显式写出矩阵向量乘法会很有帮助:

$$f = \sum_{j=1}^M x_j A_{j \cdot} \quad , \quad i = 1, \dots, M \quad , \quad (5.88)$$

然后偏导数给出为 $\partial f_i = x_q \circ \partial A_{iq}$

$$= x \in R^{1 \times 1 \times N} \quad , \quad (5.89)$$

这允许我们计算 f_i 相对于 A 的一行的偏导数,给出为 $\partial f_i / \partial A_i$:

$$= x \in R^{1 \times 1 \times N} \quad , \quad (5.90)$$

$$\frac{\partial f_i}{\partial A_k} = 0 \in R^{1 \times 1 \times N} \quad (5.91)$$

我们必须注意正确的维度。由于 f_i 映射到 R 并且 A 的每一行大小为 $1 \times N$, 我们得到一个 $1 \times 1 \times N$ 大小的张量作为 f_i 相对于 A 的一行的偏导数。

我们堆叠偏导数 (5.91) 并通过 (5.87) 获得所需的梯度

$$\begin{matrix} & & 0 \\ & & \vdots \\ \frac{\partial f_i}{\partial A} = & \begin{matrix} & \ddots & \\ & \vdots & \\ & \ddots & \end{matrix} & \in R^{1 \times (M \times N)} . \end{matrix} \quad (5.92)$$

0

例 5.13 (矩阵相对于矩阵的梯度)

考虑一个矩阵 $R \in R^{M \times N}$ 和 $f: R^{M \times N} \rightarrow R^{N \times N}$ 有

$$f(R) = R \quad R =: K \in R^{N \times N}, \quad (5.93)$$

我们在这里寻找梯度 dK/dR 。

为了解决这个难题, 让我们先写下我们已经知道的: 梯度有维度

$$\frac{dK}{dR} \in R^{(N \times N) \times (M \times N)}, \quad (5.94)$$

这是一个张量。而且,

$$\frac{dK_{pq}}{dR_{ij}} \in R^{1 \times M \times N} \quad (5.95)$$

对于 $p, q = 1, \dots, N$, 其中 K_{pq} 是 $K = f(R)$ 的第 (p, q) 个条目。用 r_i 表示 R 的第 i 列, K 的每个条目由 R 的两列的点积给出, 即

$$K_{pq} = r_p^T r_q = \sum_{i=1}^M R_{pi} R_{qi} . \quad (5.96)$$

当我们现在计算偏导数 ∂K_{pq} 时, 我们得到

$$\frac{\partial K_{pq}}{\partial R_{ij}} = \sum_{i=1}^M \frac{\partial}{\partial R_{ij}} R_{pi} R_{qi} = \partial_{pqij}, \quad (5.97)$$

$$\partial p_{qij} = \begin{cases} Riq & \text{如果 } j = p, p = q \\ Rip & \text{如果 } j = q, p = q \\ 2Riq & \text{如果 } j = p, p = q \\ 0 & \text{否则} \end{cases}. \quad (5.98)$$

从 (5.94) 中, 我们知道所需的梯度具有维度 $(N \times N) \times (M \times N)$, 并且该张量的每个条目由 (5.98) 中的 ∂p_{qij} 给出, 其中 $p, q, j = 1, \dots, N$ 和 $i = 1, \dots, M$.

5.5 计算梯度的有用恒等式

在下文中, 我们列出了机器学习环境中经常需要的一些有用梯度 (Petersen 和 Pedersen, 2012)。这里, 我们使用 $\text{tr}(\cdot)$ 作为迹 (见定义 4.4), $\det(\cdot)$ 作为行列式 (见 4.1 节), $f(X)$ 作为 $f(X)$ 的倒数, 假设它存在。

-1

$$\frac{\partial}{\partial X} f(X) \frac{\partial X}{\partial X} = \frac{\partial f(X)}{\partial X} \quad (5.99)$$

$$\frac{\partial}{\partial} \text{tr}(f(X)) = \text{tr} \frac{\partial f(X)}{\partial X} \frac{\partial X}{\partial} \quad (5.100)$$

$$\frac{\partial}{\partial} \det(f(X)) = \det(f(X)) \text{tr} f(X) \frac{\partial f(X)}{\partial X} \frac{\partial X}{\partial} \quad (5.101)$$

$$\frac{\partial}{\partial a} f(X) \frac{\partial X}{\partial X}^{-1} = -f(X) \quad \frac{\partial f(X)}{\partial X} \frac{\partial X}{\partial X}^{-1} \quad (5.102)$$

$$\frac{\partial}{\partial x} \frac{x-1}{x} b = -(x-1) ab \frac{(x-1)}{\partial X} \frac{\partial X}{\partial} \quad (5.103)$$

$$\frac{\partial}{\partial x} \frac{a}{x} = -\frac{a}{x^2} \quad (5.104)$$

$$\frac{\partial}{\partial x} \frac{a}{x} = -\frac{a}{x^2} \quad (5.105)$$

$$\frac{\partial}{\partial x} \frac{ab}{x} = ab \frac{\partial}{\partial X} \quad (5.106)$$

$$\frac{\partial}{\partial x} \frac{Bx = x}{x} (B + B) \quad (5.107)$$

$$\frac{\partial}{\partial s} (x - As) W(x - As) = -2(x - As) \quad WA \text{ 对于对称 } W \quad (5.108)$$

评论。在本书中, 我们只介绍矩阵的迹和转置。

然而, 我们已经看到导数可以是高维多元, 在这种情况下, 通常的迹和转置没有定义。在这些情况下, $D \times D \times E \times F$ 张量的迹将是 $E \times F$ 维矩阵。这是张量收缩的特例。同样, 当我们

“转置”一个张量,我们的意思是交换前两个维度。具体来说,在(5.99)到(5.102)中,当我们使用多元函数 $f(\cdot)$ 并计算关于矩阵的导数时,我们需要进行与张量相关的计算(并且选择不对它们进行矢量化,如第5.4节中所讨论的)。

5.6 反向传播和自动微分在许多机器学习应用中,我们通过执行梯度下降(第

7.1节)找到好的模型参数,这依赖于我们可以计算学习目标相对于模型参数的梯度这一事实。对于给定的目标函数,我们可以使用微积分并应用链式法则获得关于模型参数的梯度;请参阅第5.2.2节。当我们查看关于线性回归模型参数的平方损失的梯度时,我们已经在第5.3节中有所了解。

关于反向传播和链式
法则的
精彩讨论可在Tim
Vieira的博客中找到,网
址为<https://tinyurl.com/yckm2yrw>.

考虑函数

$$f(x) = x^{2^0} + \exp(x^{2^1}) + \sin(x^{2^2}) + \exp(x^{2^3}) \quad (5.109)$$

通过应用链式法则,并注意到微分是线性的,我们计算梯度

$$\begin{aligned} \frac{df}{dx} &= \frac{2x + 2x \exp(x^{2^0}) - \sin x^{2^2} + \exp(x^{2^3}) 2x + 2x \text{表达式}(x^{2^1})}{2 2^0 x + \exp(x^{2^1})} \\ &= 2x \frac{1^0}{2 2^0 x + \exp(x^{2^1})} - \sin x^{2^2} + \exp(x^{2^3}) \frac{1 + \text{指数}(x^{2^1})}{2 2^1 x + \exp(x^{2^2})}. \end{aligned} \quad (5.110)$$

以这种显式方式写出梯度通常是不切实际的,因为它通常会导致导数的表达式非常冗长。实际上,这意味着,如果我们不小心,梯度的实现可能比计算函数要昂贵得多,这会带来不必要的开销。为了训练深度神经网络模型,反向传播算法(Kelley,1960年;Bryson,1961年;Dreyfus,1962年;Rumelhart等人,1986年)是计算误差函数相对于参数的梯度的有效方法该模型。

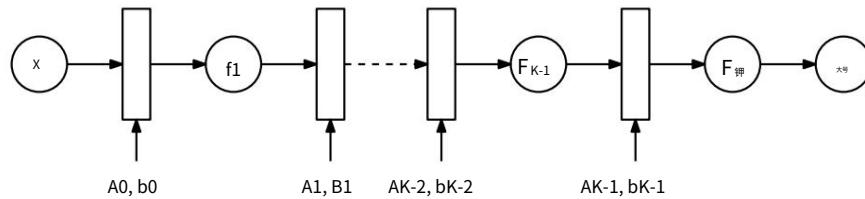
5.6.1 深度网络中的梯度

链式法则被极端使用的领域是深度学习,其中函数值 y 被计算为多级函数组合

$$y = (f_K \circ f_{K-1} \circ \dots \circ f_1)(x) = f_K(f_{K-1}(\dots(f_1(x))\dots)), \quad (5.111)$$

其中 x 是输入(例如图像), y 是观察值(例如类标签),每个函数 f_i , $i = 1, \dots, K$,拥有自己的参数。

图 5.2 多层神经网络中的正向传递计算损失 L 作为输入 x 和参数 A_i、b_i 的函数。



我们讨论案例,其中激活
每个功能都相同
层到整洁的符号。

在具有多层的神经网络中,我们在第 i 层中有函数 $f_i(x_{i-1}) = \sigma(A_{i-1}x_{i-1} + b_{i-1})$ 。这里 x_{i-1} 是第 $i-1$ 层的输出, σ 是激活函数,例如 logistic sigmoid tanh 或整流线性单元 (ReLU)。为了训练这些模型,我们需要关于所有模型参数 A_j 和 b_j 的损失函数 L 的梯度, 对于 $j = 1, \dots, K$ 。这也需要我们计算 L 相对于每一层输入的梯度。例如,如果我们有输入 x 和观察值 y 以及定义的网络结构

$$f_0 := x \quad (5.112)$$

$$f_i := \sigma(A_{i-1}f_{i-1} + b_{i-1}), i = 1, \dots, K, \quad (5.113)$$

另请参见图 5.2 的可视化,我们可能有兴趣找到 A_j, b_j for $j = 0, \dots, K-1$, 这样平方损失

$$L(\theta) = \|y - f_K(\theta, x)\|^2 \quad (5.114)$$

被最小化,其中 $\theta = \{A_0, b_0, \dots, A_{K-1}, b_{K-1}\}$ 。

为了获得关于参数集 θ 的梯度,我们需要 L 相对于每层 $j = 0$ 的参数 $\theta_j = [A_j, b_j]$ 的偏导数, ...。链式法则允许我们将偏导数确定为

关于神经网络梯度的
更深入的讨论可以在
Justin 中找到

多姆克的演讲
笔记
<https://tinyurl.com/yalcxgtv>

$$\frac{\partial L}{\partial \theta_{K-1}} = \frac{\partial L}{\partial f_K} \frac{\partial f_K}{\partial \theta_{K-1}} \quad (5.115)$$

$$\frac{\partial L}{\partial \theta_{K-2}} = \frac{\partial L}{\partial f_K} \left[\frac{\partial f_K}{\partial \theta_{K-1}} \frac{\partial f_{K-1}}{\partial \theta_{K-2}} \right] \quad (5.116)$$

$$\frac{\partial L}{\partial \theta_{K-3}} = \frac{\partial L}{\partial f_K} \left[\frac{\partial f_K}{\partial \theta_{K-1}} \frac{\partial f_{K-1}}{\partial \theta_{K-2}} \frac{\partial f_{K-2}}{\partial \theta_{K-3}} \right] \quad (5.117)$$

$$\frac{\partial L}{\partial \theta_i} = \frac{\partial L}{\partial f_K} \left[\dots \frac{\partial f_K}{\partial \theta_{i+1}} \frac{\partial f_{i+1}}{\partial \theta_i} \right] \quad (5.118)$$

橙色项是层输出相对于其输入的偏导数,而蓝色项是层输出相对于其参数的偏导数。假设,我们已经计算了偏导数 $\partial L / \partial \theta_{i+1}$,那么大部分计算可以重新用于计算 $\partial L / \partial \theta_i$ 。我们的附加条款

5.6 反向传播和自动微分

161

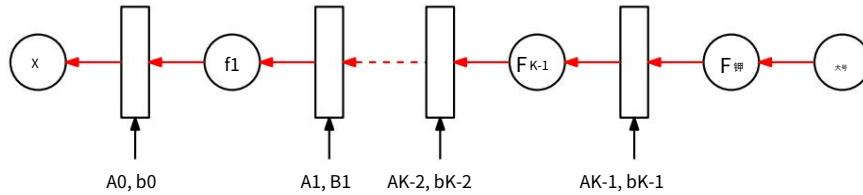


图 5.2 在多层神经网络中向后传递以计算损失函数的梯度。

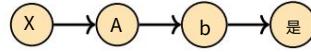


图 5.1 说明数据从 x 到 y 通过一些中间体的流动的简单图表

需要计算的由方框表示。图 5.2 可视化梯度通过网络向后传递。

变量 a, b。

5.6.2 自动微分

事实证明,反向传播是数值分析中一种称为自动微分的通用技术的特例。我们可以将自动微分视为一组技术,通过使用中间变量并应用链式法则,以数值方式(与符号方式相反)评估函数的精确(达到机器精度)梯度。自动微分应用一系列初等算术自动运算,例如加法和乘法以及初等函数,例如 \sin, \cos, \exp, \log 。通过将链式法则应用于这些操作,可以自动计算相当复杂的函数的梯度。

差异化不同于

符号微分和

自动微分适用于通用计算机程序,有正向和反向两种方式。贝丁等人。(2018) 对机器学习中的自动微分进行了很好的概述。

梯度的数值
近似,例如,通过使用有限差分。

图 5.1 显示了一个简单的图表,表示通过一些中间变量 a、b 从输入 x 到输出 y 的数据流。
如果我们要计算导数 dy/dx ,我们将应用链式法则并获得

$$\frac{dy}{dx} = \frac{dy}{db} \frac{db}{da} \frac{da}{dx}. \quad (5.119)$$

直观上,正向和反向模式在乘法顺序上有所不同 一般情况下,是阳离子。由于矩阵乘法的结合性,我们可以选择我们合作

雅可比矩阵,可以是向量、矩阵或张量。

$$\frac{dy}{dx} = \frac{dy}{db} \frac{db}{da} \frac{da}{dx}, \quad (5.120)$$

$$\frac{dy}{dx} = \frac{db}{da} \frac{da}{dx} \frac{dy}{db}. \quad (5.121)$$

等式 (5.120) 将是反向模式,因为梯度是反向模式通过图形向后门控的,即与数据流反向。方程 (5.121) 将是正向模式,其中梯度以正向模式流动,数据从左到右通过图形。

下面重点介绍反向模式的自动微分,也就是反向传播。在神经网络的上下文中,输入维数通常远高于标签的维数,反向模式在计算上比正向模式便宜得多。让我们从一个有启发性的例子开始。

例 5.14 考虑函数

$$f(x) = x^{2^x} + \exp(x^2) + \cos(x^{-2}) \quad (5.122)$$

来自 (5.109)。如果我们要在计算机上实现一个函数 f , 我们可以通过使用中间变量来节省一些计算量:

$$a = x^2, \quad (5.123)$$

$$b = \exp(a), c \quad (5.124)$$

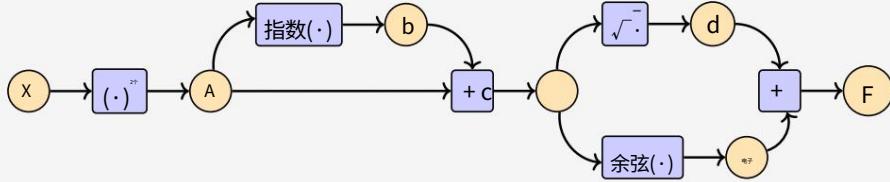
$$= a + bd = , \quad (5.125)$$

$$\sqrt{c} = . \quad (5.126)$$

$$\cos(c), f = d \quad (5.127)$$

$$+ e. \quad (5.128)$$

图 5.1 具有输入 x 、函数值 f 和中间值的计算图
变量 a, b, c, d, e



这与应用链式法则时发生的思维过程相同。请注意,前面的方程组比 (5.109) 中定义的函数 $f(x)$ 的直接实现需要更少的运算。图 5.1 中相应的计算图显示了获得函数值 f 所需的数据流和计算。

包括中间变量的方程组可以被认为是一个计算图,一种广泛用于神经网络软件库实现的表示。通过回忆初等函数导数的定义,我们可以直接计算中间变量相对于它们相应输入的导数。我们得到以下信息:

$$\frac{\partial a}{\partial x} = 2x \quad (5.129)$$

$$\frac{\partial b}{\partial c} = \exp(a) \frac{\partial a}{\partial x} \quad (5.130)$$

$$\frac{\partial c}{\partial a} = 1 = \frac{\partial b}{\partial a} \quad (5.131)$$

$$\frac{\partial d}{\partial c} = -\frac{1}{2\sqrt{-}} \quad (5.132)$$

$$\frac{\partial e}{\partial c} = -\sin(c) \quad (5.133)$$

$$\frac{\partial f}{\partial e} \frac{\partial f}{\partial d} = 1 = \frac{\partial d}{\partial e} . \quad (5.134)$$

通过查看图 5.1 中的计算图,我们可以通过从输出反向计算 $\partial f / \partial x$ 并获得

$$\frac{\partial f}{\partial c} = \frac{\partial f \partial d \partial f \partial e}{\partial c} \quad \text{--- ---} \quad (5.135)$$

$$\frac{\partial f}{\partial b} = \dots \quad (5.136)$$

$$\frac{\partial f}{\partial a} = \text{---} - \text{---} \quad (5.137)$$

$$\frac{\partial f}{\partial x} = \dots \quad (5.138)$$

请注意,我们隐含地应用了链式法则来获得 $\partial f / \partial x_0$ 。代入初等函数导数的结果,我们得到

$$\frac{\partial f}{\partial c} + \underbrace{1}_{\frac{\partial f}{\partial f}} \cdot \left(-\frac{\sin(c)}{\exp(a)} \right) = 1 \cdot \partial c \cdot 2 \sqrt{c} \cdot \partial f \quad (5.139)$$

$$\frac{\partial f}{\partial b} = \frac{+ \cdot}{\partial b}_1 \quad (5.140)$$

$$\frac{\partial f}{\partial a} = \underline{\partial c} \cdot \partial f \cdot 2x_0 \cdot \underline{\partial a} \quad (5.141)$$

$$\frac{\partial f}{\partial x} = \dots \quad (5.142)$$

通过将上述每个导数视为一个变量,我们观察到计算导数所需的计算与函数本身的计算具有相似的复杂性。这是非常违反直觉的,因为导数 ∂f (5.110) 的数学表达式比(5.109) 中函数 $f(x)$ 的数学表达式复杂得多。

$$+ 1, \dots, D : x_1 - g_1(x_1 p_d(x_1)), x_1 \in I - d$$

可以使用链式法则逐步计算函数的导数。回想一下,根据定义 $f = xD$,因此

$$\frac{\partial f}{\partial x_D} = 1. \quad (5.144)$$

对于其他变量 x_i ,我们应用链式法则 $\partial f / \partial x_j$

$$\frac{\partial f}{\partial x_i} = \underbrace{\frac{\partial x_j}{\partial x_i}}_{x_j : x_i \in Pa(x_j)} \underbrace{\frac{\partial f}{\partial g_j}}_{g_j : x_i \in Pa(x_j)}, \quad (5.145)$$

其中 $Pa(x_j)$ 是计算图中 x_j 的父节点集合。

自微分方程(5.143)是函数的前向传播,而(5.145)是梯度通过计算图的反向传播。

在反向模式下需要解

析
树。

对于神经网络训练,我们反向传播关于标签的预测误差。

每当我们有一个可以表示为计算图的函数时,上面的自动微分方法就有效,其中基本函数是可微的。事实上,函数甚至可能不是数学函数,而是计算机程序。然而,并不是所有的计算机程序都可以自动微分,例如,如果我们找不到微分初等函数。编程结构,例如for循环和if语句,也需要更加小心。

5.7 高阶导数

到目前为止,我们已经讨论了梯度,即一阶导数。有时,我们对高阶导数感兴趣,例如,当我们想使用牛顿法进行优化时,这需要二阶导数(Nocedal and Wright, 2006)。在5.1.1节中,我们讨论了使用多项式逼近函数的泰勒级数。在多变量情况下,我们可以做完全相同的事情。在下文中,我们将完全做到这一点。但是让我们从一些符号开始。

考虑一个函数 $f: R^2 \rightarrow R$ 的两个变量 x, y 。我们对高阶偏导数(和梯度)使用以下符号:

- $\frac{\partial^2 f}{\partial x^2}$ 是 f 关于 x 的二阶偏导数。 $\frac{\partial x^n}{\partial x^n}$ 是 f 关于 x 的第 n 个偏导数。
- $\frac{\partial^n f}{\partial x^n}$ 是一次偏微分 $\frac{\partial y}{\partial x}$ 得到的偏导数
- $\frac{\partial^n}{\partial y^n} = \frac{\partial}{\partial y} \frac{\partial f}{\partial x}$
entiating 相对于 x ,然后相对于 y 。
- $\frac{\partial^n}{\partial x \partial y}$ 是通过第一次偏微分获得的偏导数
 y 然后 x 。

Hessian是所有二阶偏导数的集合。

如果 $f(x, y)$ 是二次(连续)可微函数,则 $\frac{\partial^2 f}{\partial x \partial y}$

$$= \frac{\partial^2 f}{\partial y \partial x}, \quad (5.146)$$

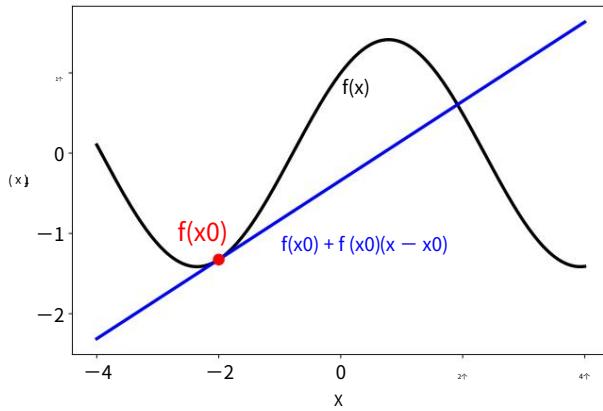


图 5.1 函数的线性逼近。
使用一阶泰勒级数展开将
原始函数 f 在 $x_0 =$
 -2 处线性化。

即微分的顺序无关紧要,对应的Hessian矩阵

海森矩阵

$$\text{高} = \begin{array}{c} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \hline \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{array} \quad (5.147)$$

是对称的。Hessian 表示为 $\nabla^2 f(x, y)$ 。通常,对于 $x \in \mathbb{R}^n$ 和 $f: \mathbb{R}^n \rightarrow \mathbb{R}$, Hessian 是一个 $n \times n$ 矩阵。Hessian 在 (x, y) 周围局部测量函数的曲率。

备注 (矢量场的 Hessian)。如果 $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ 是矢量场,则 Hessian 矩阵是 $(m \times n \times n)$ -张量。◇

5.8 线性化和多元泰勒级数函数 f 的梯度 ∇f 通常用于 f 在 x_0 附近的局部线性逼近:

$$f(x) \approx f(x_0) + (\nabla f)(x_0)(x - x_0). \quad (5.148)$$

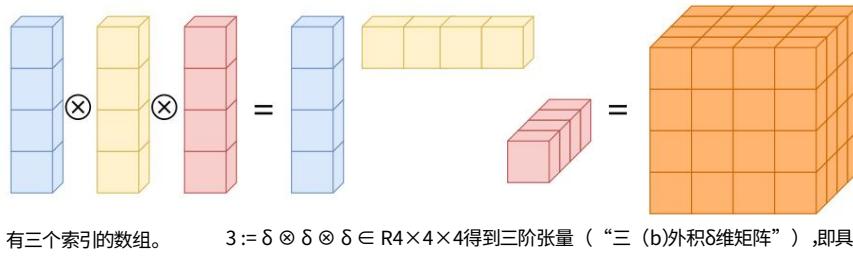
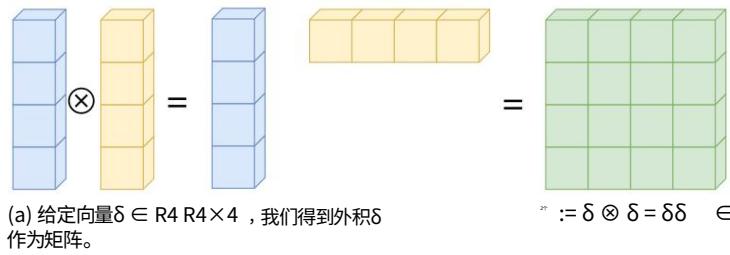
这里 $(\nabla f)(x_0)$ 是 f 相对于 x 的梯度,在 x_0 处计算。

图 5.1 说明了函数 f 在输入 x_0 处的线性逼近。原函数用直线逼近。这个近似值在局部是准确的,但是我们离 x_0 越远,近似值就越差。方程 (5.148) 是 f 在 x_0 处的多元泰勒级数展开的特例,其中我们只考虑前两项。我们在下面讨论更一般的情况,这将允许更好的近似。

定义 5.7 (多元泰勒级数)。我们考虑一个函数

$$f: \mathbb{R}^D \rightarrow \mathbb{R} \quad (5.149)$$

图 5.1 可视化外积。向量的外积每项将数组的维数增加 1。(a) 两个向量的外积产生一个矩阵；(b) 三个向量的外积产生一个三阶张量。



$$x \rightarrow f(x), x \in \mathbb{R}^D, \quad (5.150)$$

在 x_0 处是平滑的。当我们定义差分向量 $\delta := x - x_0$, multivariate Taylor 时, f 在 (x_0) 处的多

元泰勒级数定义为
系列

$$f(x) = \sum_{k=0}^{\infty} \frac{D^k_x f(x_0)}{k!} \delta^k, \quad (5.151)$$

其中 $D^k_x f(x_0)$ 是 f 关于 x 的第 k 个 (全) 导数, evaluated 在 x_0 。

泰勒多项式

定义 5.8 (泰勒多项式)。 f 在 x_0 处的 n 次泰勒多项式包含 (5.151) 中级数的前 $n+1$ 个分量, 定义为

$$T_n(x) = \sum_{k=0}^n \frac{D^k_x f(x_0)}{k!} \delta^k. \quad (5.152)$$

在 (5.151) 和 (5.152) 中, 我们使用了稍微草率的 δ 符号, 它没有为向量 $x \in \mathbb{R}^D$ 、 $D > 1$ 和 $k > 1$ 定义。注意 $D^k_x f$ 和 δ 都是 k 阶张量, 即, k 维数组。这

向量可以实现为一维
数组, 一个二维矩阵
大批。

x 向量 $\delta \in \mathbb{R}^D$ 的 k 阶张量 $\delta \in \overbrace{\mathbb{R}^D \times \mathbb{R}^D \times \dots \times \mathbb{R}^D}^{k \text{ 次}}$ 作为 k 次外积获得,

$$\delta := \delta \otimes \delta = \delta\delta, \quad \delta[i, j] = \delta[i]\delta[j] \quad (5.153)$$

$$\delta := \delta \otimes \delta \otimes \delta, \quad \delta[i, j, k] = \delta[i]\delta[j]\delta[k]. \quad (5.154)$$

图 5.1 可视化了两个这样的外积。一般来说, 我们得到

条款

$$D_k f(x_0) \delta = \sum_{i_1=1}^{\infty} \cdots \sum_{i_k=1}^{\infty} D_k^{\vec{i}} f(x_0)[i_1, \dots, i_k] \delta[i_1] \cdots \delta[i_k] \quad (5.155)$$

在泰勒级数中, $D_k^{\vec{x}} f(x_0) \delta$ 包含 k 阶多项式。

现在我们已经为向量场定义了泰勒级数, 让我们明确地为写下第一项 $D_k^{\vec{x}} f(x_0) \delta$

$k = 0, \dots, 3$ 和 $\delta := \vec{x} - \vec{x}_0$:

$$k=0: D_0^{\vec{x}} f(x_0) \delta^0 = f(x_0) \in \mathbb{R} \quad (5.156)$$

$$k=1: D_1^{\vec{x}} f(x_0) \delta^1 = \nabla \vec{x} f(x_0) \underbrace{\delta}_{1 \times \text{深}} = \sum_{i=1}^{\infty} \nabla x f(x_0)[i] \delta[i] \in \mathbb{R} \quad (5.157)$$

$$k=2: D_2^{\vec{x}} f(x_0) \delta^2 = \text{tr } H(x_0) \underbrace{\delta}_{\text{深} \times \text{深}} \underbrace{\delta}_{\text{长} \times 1} = \delta^T H(x_0) \delta \quad (5.158) \quad \text{np.einsum('i,i', Df1,d) np.einsum('ij,j', Df2,c)}$$

$$= \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} H[i, j] \delta[i] \delta[j] \in \mathbb{R} \quad (5.159)$$

$$k=3: D_3^{\vec{x}} f(x_0) \delta^3 = \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} D_3^{\vec{x}} f(x_0)[i, j, k] \delta[i] \delta[j] \delta[k] \in \mathbb{R} \quad (5.160)$$

这里, $H(x_0)$ 是在 x_0 处计算的 f 的 Hessian。

示例 5.15 (具有两个变量的函数的泰勒级数展开)

考虑函数

$$f(x, y) = x + 2xy + y^3. \quad (5.161)$$

我们想要计算 f 在 $(x_0, y_0) = (1, 2)$ 处的泰勒级数展开。

在我们开始之前, 让我们讨论一下会发生什么: (5.161) 中的函数是 3 次多项式。我们正在寻找泰勒级数展开, 它本身是多项式的线性组合。因此, 我们不期望泰勒级数展开包含四次或更高次的项来表示三次多项式。这意味着确定 (5.151) 的前四项对于 (5.161) 的精确替代表示应该足够了。

为了确定泰勒级数展开, 我们从常数项和一阶导数开始, 它们由下式给出

$$f(1, 2) = 13 \quad \partial f \quad (5.162)$$

$$\begin{aligned} &= -2x + 2y \Rightarrow \partial_x \\ &\frac{\partial f}{\partial x}(1, 2) = 6 \end{aligned} \quad (5.163)$$

$$\frac{\partial f}{\partial y} = 2x + 3y \quad \Rightarrow \quad \frac{\partial f}{\partial y}(1, 2) = 14. \quad (5.164)$$

因此,我们得到

$$D_1 x, y f(1, 2) = \nabla x, y f(1, 2) = \begin{matrix} \frac{\partial f}{\partial x}(1, 2) & \frac{\partial f}{\partial y}(1, 2) = 6 \\ 1 & 14 \end{matrix} \in \mathbb{R}^{1 \times 2} \quad (5.165)$$

这样

$$\frac{D_1 x, y f(1, 2)}{1!} \delta = 6 \begin{matrix} x-1 \\ y-2 \end{matrix} = 6(x-1) + 14(y-2). \quad (5.166)$$

请注意, $D_1 x, y f(1, 2) \delta$ 仅包含线性项,即一阶多项式米勒斯。

二阶偏导数由下式给出

$$\frac{\partial^2 f}{\partial x^2} = 2 \Rightarrow \frac{\partial^2 f}{\partial x^2}(1, 2) = 2 \quad (5.167)$$

$$\frac{\partial^2 f}{\partial y^2} = 6 \Rightarrow \frac{\partial^2 f}{\partial y^2}(1, 2) = 6 \quad (5.168)$$

$$\frac{\partial^2 f}{\partial x \partial y} = 2 \Rightarrow \frac{\partial^2 f}{\partial x \partial y}(1, 2) = 2 \quad (5.169)$$

$$\frac{\partial^2 f}{\partial y \partial x} = 2 \Rightarrow \frac{\partial^2 f}{\partial y \partial x}(1, 2) = 2. \quad (5.170)$$

当我们收集二阶偏导数时,我们得到海森

$$H = \begin{matrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{matrix} = \begin{matrix} 2 & 2 & 2 \\ 2 & 6 \end{matrix}, \quad (5.171)$$

这样

$$H(1, 2) = \begin{matrix} 2 & 2 & 2 \\ 2 & 12 \end{matrix} \in \mathbb{R}^{2 \times 2}. \quad (5.172)$$

因此,泰勒级数展开的下一项为

$$\frac{D_2 x, y f(1, 2)}{2!} \delta^2 = \frac{1}{2} \delta H(1, 2) \delta \quad (5.173a)$$

$$= \frac{1}{2} \begin{matrix} 2 & 2 & 2 \\ x-1 & y-2 \end{matrix} \quad (5.173b)$$

$$= (x-1)^2 + 2(x-1)(y-2) + 6(y-2)^2. \quad (5.173c)$$

这里, $D_2 x, y f(1, 2) \delta^2$ 仅包含二次项,即二阶多项式

三阶导数为

$$D3 x,yf = \begin{matrix} \frac{\partial^3 f}{\partial x^3} & \frac{\partial^3 f}{\partial x^2 \partial y} \\ \frac{\partial^3 f}{\partial x \partial y^2} & \frac{\partial^3 f}{\partial y^3} \end{matrix} \in R^{2 \times 2}, \quad (5.174)$$

$$D3 x,yf[:, :, 1] = \begin{matrix} \frac{\partial^3 f}{\partial x^3} \\ \frac{\partial^3 f}{\partial x^2 \partial y} \\ \frac{\partial^3 f}{\partial x \partial y^2} \end{matrix}, \quad (5.175)$$

$$D3 x,yf[:, :, 2] = \begin{matrix} \frac{\partial^3 f}{\partial y^3} \\ \frac{\partial^3 f}{\partial y^2 \partial x} \\ \frac{\partial^3 f}{\partial y \partial x^2} \end{matrix}. \quad (5.176)$$

由于 (5.171) 中 Hessian 的大多数二阶偏导数是常数,唯一的非零三阶偏导数是

$$\frac{\partial^3 f}{\partial y^3} = 6 \Rightarrow \frac{\partial^3 f}{\partial y^3}(1, 2) = 6. \quad (5.177)$$

高阶导数和 3 阶混合导数 (例如 ∂f_3) 消失,使得 $\partial x^2 \partial y$

$$D3 x,yf[:, :, 1] = \begin{matrix} 0 & 0 & 0 \\ 0 & & \end{matrix}, \quad D3 x,yf[:, :, 2] = \begin{matrix} 0 & 0 & 0 \\ 6 & & \end{matrix} \quad (5.178)$$

和

$$\frac{D3 x,yf(1, 2)}{3!} \delta^{**} = (y - 2)3, \quad (5.179)$$

它收集了泰勒级数的所有立方项。总的来说, (确切的)

f 在 $(x_0, y_0) = (1, 2)$ 处的泰勒级数展开是 $D2 x,yf(1, 2) f(x) = f(1,$

$$2)\delta + \frac{f''(1, 2)}{2!} + \frac{D3 x,yf(1, 2)}{3!} \delta^{**} \quad (5.180a)$$

$$1) + (y - 2) \frac{\partial f(1, 2)}{\partial x} \frac{\partial f(1, 2)}{\partial y} = f(1, 2) + (x - 1)2 + (y - 2)3$$

$$\frac{2f(1, 2)}{6} \frac{\partial^3 f(1, 2)}{\partial y^3} = \frac{1}{6} \frac{(y - 2)3}{(x - 1)(y - 2) + 2}$$

$$+ (x - 1)2 + 6(y - 2)2 + 2(x - 1)(y - 2) + (y - 2)3. \quad (5.180c)$$

在这种情况下,我们得到了 (5.161) 中多项式的精确泰勒级数展开,即 (5.180c) 中的多项式与 (5.161) 中的原始多项式相同。在这个特定的例子中,这个结果并不令人惊讶,因为原始函数是一个三阶多项式,我们通过 (5.180c) 中的常数项、一阶、二阶和三阶多项式的线性组合来表示它)。

5.9 延伸阅读

可以在 Magnus 和 Neudecker (2007) 中找到有关矩阵微分的更多详细信息以及对所需线性代数的简短回顾。

自动微分有着悠久的历史,我们参考了 Griewank 和 Walther (2003)、Griewank 和 Walther (2008) 以及 Elliott (2009) 及其中的参考文献。

在机器学习 (和其他学科) 中,我们经常需要计算期望值,即我们需要求解以下形式的积分

$$\text{例如} [f(x)] = f(x)p(x)dx \quad (5.181)$$

即使 $p(x)$ 是一种方便的形式 (例如,高斯),这个积分通常也无法解析求解。 f 的泰勒级数展开是一种求近似解的方法:假设 $p(x) = N(\mu, \Sigma)$ 是高斯分布,则围绕 μ 的一阶泰勒级数展开局部线性化非线性函数 f 。对于线性函数,如果 $p(x)$ 服从高斯分布 (请参阅第 6.5 节),我们可以准确地计算均值 (和协方差)。扩展卡尔曼滤波器 (Maybeck, 1979) 大量利用了此属性,用于非线性动态系统 (也称为“状态空间模型”) 中的在线状态估计。其他确定性方法无须变换来逼近 (5.181) 中的积分是无须变换 (Julier and Uhlmann, 1997),它不需要任何梯度,或者 Laplace Laplace approximation approximation (MacKay, 2003; Bishop, 2006; Murphy, 2012),它使用二阶泰勒级数展开 (需要 Hessian) 对其模式周围的 $p(x)$ 进行局部高斯近似。

扩展卡尔曼
筛选

练习

5.1 计算导数 f'

(x) 对于

$$f(x) = \frac{d}{dx}(x^4 + x^3).$$

5.2 计算导数 f'

(x) 逻辑 sigmoid

$$f(x) = 1 + \frac{e^{-x}}{\exp(-x)}.$$

5.3 计算导数 f'

(x) 的功能

$$f(x) = \exp(-\frac{1}{2\sigma^2}(x - \mu)^2),$$

其中 $\mu, \sigma \in \mathbb{R}$ 是常数。

5.4 计算泰勒多项式 T_n , $n = 0, \dots, 5$ of $f(x) = \sin(x) + \cos(x)$ 在 $x_0 = 0$ 。

5.5 考虑以下功能:

$$\begin{aligned} f_1(x) &= \sin(x_1) \cos(x_2), x \in \mathbb{R} \\ f_2(x, y) &= x^y, x, y \in \mathbb{R} \\ f_3(x) &= xx, x \in \mathbb{R} \end{aligned}$$

A。 $\partial f / \partial b$ 的维数是多少? 计算雅可比矩阵 $\frac{\partial f}{\partial x}$?

5.6 微分 f 关于 t 和 g 关于 X , 其中 $f(t) = \sin(\log(t - t))$, $g(X) = \text{tr}(AXB)$, $A \in \mathbb{R}^{D \times E}$, $X \in \mathbb{R}^E$

$$t \in \mathbb{R}^D, B \in \mathbb{R}^{E \times D},$$

其中 $\text{tr}(\cdot)$ 表示迹线。

5.7 使用链式法则计算下列函数的导数 df/dx 。提供每个偏导数的维数。详细描述你的步骤。

A。

$$f(z) = \log(1 + z), z = x \in \mathbb{R}, x \in \mathbb{R}^T$$

b.

$$f(z) = \sin(z), z = Ax + b, A \in \mathbb{R}^{E \times D}, x \in \mathbb{R}^D, b \in \mathbb{R}^E$$

其中 $\sin(\cdot)$ 应用于 z 的每个元素。

5.8 计算下列函数的导数 df/dx 。详细描述你的步骤。A。使用链式法则。提供每个偏导数的维度

略去。

$$\begin{aligned} f(z) &= \exp(-z) \\ -1z &= g(y) = yyy = h(x) = \\ &x - \mu \end{aligned}$$

其中 $x, \mu \in \mathbb{R}^D, S \in \mathbb{R}^{D \times D}$ 。

b.

$$f(x) = \text{tr}(xx^T), x \in \mathbb{R}^{2 \times D}$$

这里 $\text{tr}(A)$ 是 A 的迹, 即对角线元素 A_{ii} 之和。

提示: 显式写出外积。

C。使用链式法则。提供每个偏导数的维数。您不需要显式计算偏导数的乘积。

$$\begin{aligned} f &= \tanh(z) \in \mathbb{R}^M \\ z &= Ax + b, x \in \mathbb{R}^E, a \in \mathbb{R}^{M \times N}, b \in \mathbb{R}^M. \end{aligned}$$

在这里, \tanh 应用于 z 的每个分量。

5.9 我们定义

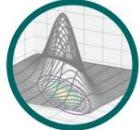
$$g(x, z, v) := \log p(x, z) - \log q(z, v) z := t(\varepsilon, v)$$

对于可微函数 p, q, t 和 $x \in \mathbb{R}^D, z \in \mathbb{R}^E, v \in \mathbb{R}^F$ 使用链式法则, 计算梯度

$$\frac{d}{dv} g_{dv}(x, z,$$

6个

概率与分布



粗略地说,概率涉及对不确定性的研究。概率可以被认为是事件发生的次数的分数,或者是对事件的信念程度。然后我们想使用这个概率来衡量某事在实验中发生的可能性。正如第1章所述,我们经常量化数据中的不确定性、机器学习模型中的不确定性以及模型产生的预测中的不确定性。量化不确定性需要随机变量的概念,它是一个将随机实验的结果映射到我们感兴趣的一组属性的函数。与随机变量相关的是一个函数,它测量特定结果(或集合)出现的概率结果;这称为概率分布。

随机变量

概率分布

概率分布用作其他概念的构建块,例如概率建模(第8.4节)、图形模型(第8.5节)和模型选择(第8.6节)。在下一节中,我们将介绍定义概率空间的三个概念(样本空间、事件和事件的概率)以及它们与称为随机变量的第四个概念的关系。由于严谨的演示可能会掩盖概念背后的直觉,因此演示文稿故意略带波浪形。图6.2显示了本章中提出的概念的概要。

6.1 概率空间的构建概率论旨在定义一个数学结构来描述实验的随机结果。例如,当抛一枚硬币时,我们无法确定结果,但通过多次抛硬币,我们可以观察到平均结果的规律性。

使用概率的这种数学结构,目标是执行自动推理,从这个意义上说,概率概括了逻辑推理(Jaynes,2003)。

6.1.1 哲学问题在构建自动推理系统时,经典布尔逻辑不允许我们表达某些形式的合理推理。考虑

172

该材料由剑桥大学出版社出版,名为Marc Peter Deisenroth、A. Aldo Faisal 和 Cheng Soon Ong的机器学习数学(2020)。此版本可免费查看和下载,仅供个人使用,不得重新分发、转售或用于衍生作品。© MP Deisenroth、AA Faisal 和 CS Ong,2021年。<https://mml-book.com>

6.1 概率空间的构造

173

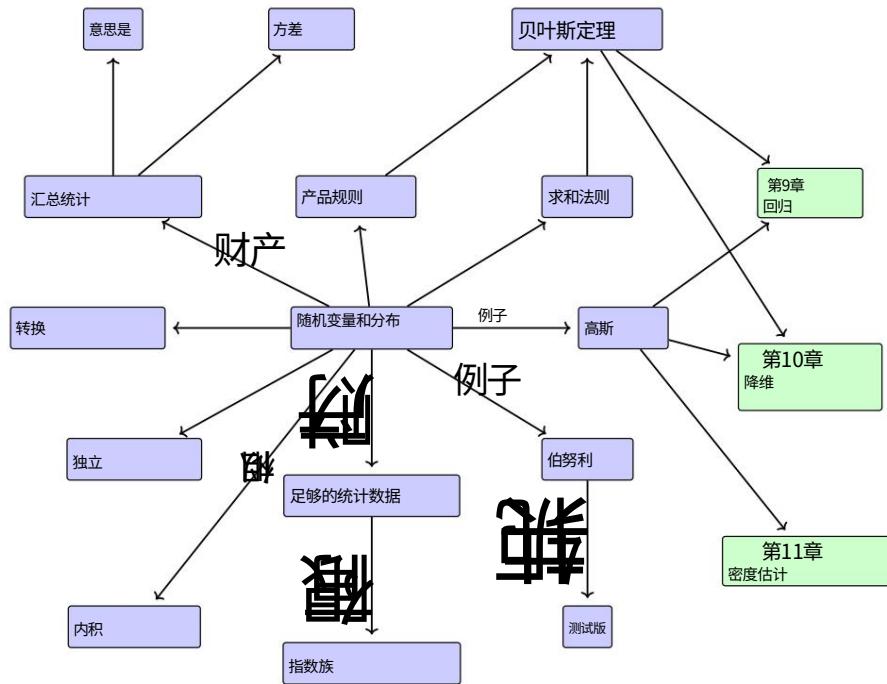


图 6.2 与随机变量和相关概念的思维导图

概率分布,如
本章所述。

以下场景:我们观察到A为假。我们发现B变得不太合理,尽管不能从经典逻辑中得出结论。

我们观察到B为真。似乎A变得更合理了。我们每天都在使用这种推理形式。我们在等一个朋友,考虑三种可能:H1,她准时; H2,她被交通耽误了;而H3,她已经被外星人绑架了。当我们观察到我们的朋友迟到了时,我们必须从逻辑上排除 H1。我们也倾向于认为 H2 更有可能,尽管我们在逻辑上并不需要这样做。最后,我们可能认为 H3 是可能的,但我们仍然认为它不太可能。

我们如何断定 H2 是最合理的答案?这样看来,“似是而非”的概率论可以被认为是布尔逻辑的推广。在推理中有必要扩展机器学习的上下文,它通常以这种方式应用来形式化离散真实和自动推理系统的设计。在 Pearl (1988) 中可以找到关于概率论如何成为推理系统基础的更多论点。

真假值连续

似是而非”

(杰恩斯,2003 年)。

Cox (Jaynes,2003) 研究了概率的哲学基础以及它应该如何与我们认为应该是真的(在逻辑意义上)相关联。另一种思考方式是,如果我们对我们的常识很精确,我们最终会构建概率。

ET Jaynes (1922–1998) 确定了三个必须适用于所有合理性的数学标准:

1. 可信度用实数表示。
2. 这些数字必须基于常识规则。

3. 得出的推理必须是一致的，“一致”一词具有以下三个含义：

- (a) 一致性或不矛盾：当可以通过不同的方式达到相同的结果时，必须在所有情况下找到相同的似真值。
- (b) 诚实：必须考虑所有可用数据。
- (c) 可再现性：如果我们将两个问题的知识状态相同，那么我们必须为这两个问题分配相同程度的似真性。

Cox-Jaynes 定理证明这些合理性足以定义适用于合理性 p 的通用数学规则，直至通过任意单调函数进行转换。至关重要的是，这些规则是概率规则。

评论。在机器学习和统计学中，有两种主要的概率解释：贝叶斯解释和频率解释（Bishop, 2006 年；Efron 和 Hastie, 2016 年）。贝叶斯解释使用概率来指定用户对事件的不确定程度。它有时被称为“主观概率”或“置信度”。

频率论者的解释考虑的是感兴趣事件的相对频率与发生的事件总数之比。事件的概率定义为当一个人拥有无限数据时，该事件在极限内的相对频率。 ◇一些关于概率模型的机器学习文本使用懒惰的符号和行话，这令人困惑。本文也不例外。多个不同的概念都被称为“概率分布”，读者不得不经常将其含义与上下文区分开来。帮助理解概率分布的一个技巧是检查我们是在尝试对某些事物建模（离散随机变量）还是某些事物连续（连续随机变量）。我们在机器学习中解决的问题类型与我们考虑的是分类模型还是连续模型密切相关。

6.1.2 概率和随机变量在讨论概率时，经常混淆三种截然不同的概念。

首先是概率空间的概念，它使我们能够量化概率的概念。然而，我们大多不直接使用这个基本概率空间。相反，我们使用随机变量（第二个想法），将概率转移到更方便的（通常是数字）空间。第三个想法是与随机变量相关的分布或规律的想法。我们将在本节中介绍前两个想法，并在 6.2 节中扩展第三个想法。

现代概率基于 Kolmogorov 提出的一组公理

(Grinstead and Snell, 1997; Jaynes, 2003),介绍了样本空间、事件空间和概率测度三个概念。
概率空间模拟真实世界的过程（称为实验）

随机结果。

样本空间 Ω

样本空间是实验所有可能结果的集合,样本空间通常用 Ω 表示。例如,两次连续抛硬币的样本空间为{hh, tt, ht, th},其中“h”表示“正面”,“t”表示“反面”。

活动空间A

事件空间是实验可能结果的空间。如果在实验结束时我们可以观察到特定结果 $\omega \in \Omega$ 是否在A中,则样本空间 Ω 的事件空间子集A在事件空间A中。事件空间A是通过考虑子集的集合获得的 Ω ,以及离散概率分布(第6.2.1节)

A通常是 Ω 的幂集。

概率P对于每个事件 $A \in$

A ,我们关联一个数字 $P(A)$ 来衡量事件发生的概率或信念程度。 $P(A)$ 称为A的概率。

可能性

单个事件的概率必须在区间[0, 1]内,样本空间 Ω 中所有结果的总概率必须为1,即 $P(\Omega) = 1$ 。给定一个概率空间 (Ω, A, P) ,我们想用它来模拟一些真实世界的现象。在机器学习中,我们经常避免明确提及概率空间,而是提及感兴趣数量的概率,我们用T表示。

在本书中,我们将T称为目标空间,并

将T的元素称为状态。我们引入了一个目标空间函数 $X : \Omega \rightarrow T$,它采用 Ω 的一个元素(结果)并返回一个特定的兴趣量x,即T中的一个值。这种从 Ω 到T的关联/映射称为随机变量。例如,在抛随机变量两枚硬币并计算正面朝上的次数的情况下,随机变量X映射到三种可能的结果: $X(hh) = 2$ 、 $X(ht) = 1$ 、 $X(th) = 1$ 、 $X(tt) = 0$ 。在这种特殊情况下, $T = \{0, 1, 2\}$,我们感兴趣的是T的元素的概率。对于有限样本空间 Ω 和名称“随机finite T the function corresponding to a random variable is essentially a variable”是查找表的重要来源。对于任何子集 $S \subseteq T$,我们将 $P(X(S)) \in [0, 1]$ (概率)关联到对应于随机变量X发生的特定事件。示例6.1提供了术语的具体说明

,

,

误解,因为它既不是

随机的,也不是一个变量。
它是一个
功能。

狂欢。

评论。不幸的是,前面提到的样本空间 Ω 在不同的书中有不同的名称。 Ω 的另一个常见名称是“状态空间”(Jacod和Protter,2004年),但状态空间有时保留用于指代动力系统中的状态(Hasselblatt和

卡托克,2003 年)。有时用来描述 Ω 的其他名称是：“样本◇描述空间”、“可能性空间”和“事件空间”。

示例 6.1这个玩

具示例是我们假设读者已经熟悉计算概率,本质上是事件集的交集和并集。对硬币翻转示例的更温和的介绍。在 Walpole 等人的第 2 章中可以找到许多例子的概率。(2011)。

考虑一个统计实验,我们对一个游乐场游戏进行建模,该游戏包括从袋子中取出两个硬币(有放回)。袋子里有来自美国(表示为 \$)和英国(表示为 £)的硬币,由于我们从袋子中抽出两枚硬币,因此总共有四种结果。

那么这个实验的状态空间或样本空间 Ω 就是 $(\$, \$), (\$, £), (£, \$), (£, £)$ 。让我们假设一袋硬币的组成是抽奖随机返回 \$ 的概率为 0.3。

我们感兴趣的事件是重复抽奖返回 \$ 的总次数。让我们定义一个随机变量 X ,它将样本空间 Ω 映射到 T , T 表示我们从包中抽出 \$ 的次数。我们可以从前面的样本空间中看到我们可以获得零个\$、一个\$或两个\$,因此 $T = \{0, 1, 2\}$ 。随机变量 X (函数或查找表)可以表示为如下表:

$$X((\$, \$)) = 2 \quad (6.1)$$

$$X((\$, £)) = 1 \quad (6.2)$$

$$X((£, \$)) = 1 \quad (6.3)$$

$$X((£, £)) = 0. \quad (6.4)$$

由于我们在抽取第二个之前返回抽取的第一个硬币,这意味着两次抽取彼此独立,我们将在第 6.4.5 节中讨论。请注意,有两个实验结果映射到同一事件,其中只有一个抽奖返回 \$。

因此, X 的概率质量函数(第 6.2.1 节)由下式给出

$$\begin{aligned} P(X = 2) &= P((\$, \$)) \\ &= P(\$) \cdot P(\$) \\ &= 0.3 \cdot 0.3 = 0.09 \end{aligned} \quad (6.5)$$

$$\begin{aligned} P(X = 1) &= P((\$, £) \cup (£, \$)) \\ &= P((\$, £)) + P((£, \$)) = 0.3 \\ &\quad \cdot (1 - 0.3) + (1 - 0.3) \cdot 0.3 = 0.42 \end{aligned} \quad (6.6)$$

$$\begin{aligned} &= P((£, £)) \\ &= P(£) \cdot P(£) = \\ &(1 - 0.3) \cdot (1 - 0.3) = 0.49. \end{aligned} \quad (6.7)$$

在计算中,我们把两个不同的概念等同起来, X 输出的概率和 Ω 中样本的概率。例如,在(6.7)中我们说 $P(X = 0) = P((\varepsilon, \varepsilon))$ 。考虑随机变量 $X : \Omega \rightarrow T$ 和子集 $S \subseteq T$ (例如, T 的单个元素,例如抛两枚硬币时获得正面朝上的结果)。

,

令 $X^{-1}(S)$ 为 X 对 S 的原像,即 Ω 中映射到 X 下 S 的元素集; $\{\omega \in \Omega : X(\omega) \in S\}$ 。通过随机变量 X 理解 Ω 中事件的概率转换的一种方法是将其与 S 的原像概率相关联 (Jacod 和 Protter,2004)。对于 $S \subseteq T$

, 我们有符号

$$P(X(S) = P(X \in S) = P(X^{-1}(S)) = P(\{\omega \in \Omega : X(\omega) \in S\}) \quad (6.8)$$

(6.8) 的左侧是我们感兴趣的一组可能结果的概率 (例如, $S = 1$ 的数量)。通过将状态映射到结果的随机变量 X ,我们在右侧看到-(6.8)的一侧,这是具有属性 (例如, $\$f, f\$$)的一组状态 (以 Ω 为单位)的概率。我们说随机变量 X 是按照特定的概率分布 P_X 分布的,它定义了事件与随机变量结果概率之间的概率映射。换句话说,函数 P_X 或等价的 $P \circ X^{-1}$ 是随机变量 X 的规律或分布。

法律
分配

评论。目标空间,即随机变量 X 的取值范围 T ,用来表示概率空间的种类,即 T 随机变量。

当 T 有限或可数无限时,这称为离散随机变量 (第 6.2.1 节)。对于连续随机变量 (第 6.2.2 节), 我们只考虑 $T = \mathbb{R}$ 或 $T = \mathbb{R}^D$ 。 ◇

6.1.3 统计

概率论和统计学经常一起出现,但它们涉及不确定性的不同方面。对比它们的一种方法是通过所考虑的问题的种类。使用概率,我们可以考虑一些过程的模型,其中潜在的不确定性由随机变量捕获,我们使用概率规则来推导发生的事情。在统计学中,我们观察到某事已经发生,并试图找出解释观察结果的潜在过程。从这个意义上说,机器学习在其目标上接近统计学,即构建一个模型来充分表示生成数据的过程。我们可以使用概率规则为某些数据获得“最佳拟合”模型。

机器学习系统的另一个方面是我们对泛化误差感兴趣 (见第 8 章)。这意味着我们实际上对系统在未来我们将观察到的实例上的性能感兴趣,这些实例与我们拥有的实例不同

到目前为止看到。这种对未来表现的分析依赖于概率和统计,其中大部分超出了本章将要介绍的范围。

鼓励有兴趣的读者阅读 Boucheron 等人的书籍。(2013) 和 Shalev-Shwartz 和 Ben-David (2014)。我们将在第 8 章中看到更多关于统计的信息。

6.2 离散和连续概率

让我们将注意力集中在描述第 6.1 节中介绍的事件概率的方法上。根据目标空间是离散的还是连续的,指代分布的自然方式是不同的。

当目标空间 T 是离散的时,我们可以指定随机变量 X 取特定值 $x \in T$ 的概率,表示为 $P(X = x)$ 。

概率质量函数
离散随机变量 X 的表达式 $P(X = x)$ 称为概率质量函数。当目标空间 T 是连续的,例如实数 R 时,更自然地指定随机变量 X 在区间内的概率,用 $P(a < X < b)$ 表示,其中 $a < b$ 。按照惯例,我们指定随机变量 X 小于特定值 x 的概率,用 $P(X < x)$ 表示。连续随机变量 X 的表达式 $P(X < x)$ 称为累积分布函数。我们将在第 6.2.2 节中讨论连续随机变量。

累积的

我们将重新审视术语并对比离散和连续
第 6.2.3 节中的随机变量。

单变量
评论。我们将使用短语单变量分布来指代单个随机变量的分布(其状态由非粗体 x 表示)。我们将多个随机变量的分布称为多元分布,并且通常会考虑随机变量的向量(其状态用粗体 x 表示)。 ◇

多元的

6.2.1 离散概率

当目标空间是离散的时,我们可以将多个随机变量的概率分布想象成填充一个(多维)数字数组。图 6.1 显示了一个示例。联合概率的目标空间是每个随机变量的目标空间的笛卡尔积。我们将联合概率定义为两个值的联合输入

联合概率

$$P(X = x_i, Y = y_j) = \begin{cases} \text{奈杰} & , \\ \text{否} & , \end{cases} \quad (6.9)$$

其中 n_{ij} 是状态为 x_i 和 y_j 的事件数, N 是事件总数。联合概率是两个事件相交的概率,即 $P(X = x_i \cap Y = y_j)$ 。图 6.1 说明了离散概率分布的概率质量函数(pmf)。对于两个随机变量 X 和 Y, 概率

$$, Y = y_j) = P(X = x_i \cap Y = y_j).$$

概率质量函数

			词	
	y1			
是				
	y2		奈杰	
	y3			
	x1	x2	x3	x4
			x5	
		X		

图 6.1 a 的可视化
具有随机变量 X 的离
散双变量概率质量函
数
和 Y。这
图改编自 Bishop (2006)。

$X = x$ 和 $Y = y$ (惰性地) 写为 $p(x, y)$ 并称为联合概率。可以将概率视为接受状态 x 和 y 并返回实数的函数, 这就是我们编写 $p(x, y)$ 的原因。

无论随机变量 Y 的值边际概率如何, X 取值 x 的边际概率 (惰性地) 写为 $p(x)$ 。我们写 $X \sim p(x)$ 来表示随机变量 X 是根据 $p(x)$ 分布的。如果我们只考虑 $X = x$ 的实例, 那么 $Y = y$ 的实例分数 (条件概率) 被 (懒惰地) 写为 $p(y | x)$ 。条件概率

例 6.2 考虑两个

随机变量 X 和 Y , 其中 X 有五种可能的状态, Y 有三种可能的状态, 如图 6.1 所示。我们用 n_{ij} 表示状态 $X = x_i$ 和 $Y = y_j$ 的事件数, 并用 N 表示事件总数。值 c_i 是第 i 列的各个频率的总和, 即 $c_i = \sum_{j=1}^3 n_{ij}$ 。同样, 值 r_j 是行总和, 即 $r_j = \sum_{i=1}^5 n_{ij}$ 。使用这些定义, 我们可以简洁地表达 X 和 Y 的分布

每个随机变量的概率分布, 即边际概率, 可以看作是一行或一列的总和

$$P(X = x_i) = \frac{\sum_{j=1}^3 n_{ij}}{N} \quad (6.10)$$

和

$$P(Y = y_j) = \frac{\sum_{i=1}^5 n_{ij}}{N}, \quad (6.11)$$

其中 c_i 和 r_j 分别是概率表的第 i 列和第 j 行。按照惯例, 对于具有有限数量事件的离散随机变量, 我们假设概率之和为 1, 即

$$\sum_{i=1}^5 P(X = x_i) = 1 \quad \text{和} \quad \sum_{j=1}^3 P(Y = y_j) = 1. \quad (6.12)$$

条件概率是 $p(x_i | y_j)$ 中行或列的分数

细胞。例如,给定X的Y的条件概率是

$$P(Y = y_j | X = x_i) = \frac{\text{奈杰}}{\text{词}}, \quad (6.13)$$

给定Y的X的条件概率是

$$P(X = x_i | Y = y_j) = \frac{\text{奈杰}}{r_j}. \quad (6.14)$$

在机器学习中,我们使用离散概率分布来模拟分类变量categorical variables,即取一组有限无序值的变量。它们可以是分类特征,例如用于预测一个人的薪水时在大学获得的学位,或分类标签,例如用于手写识别时的字母表中的字母。

离散分布也常用于构建组合有限数量的连续分布的概率模型(第11章)。

6.2.2 连续概率

我们在本节中考虑实值随机变量,即我们考虑作为实线R区间的目空间。在本书中,我们假装我们可以对实随机变量执行操作,就像我们有离散概率空间一样有限状态。然而,这种简化在两种情况下并不精确:当我们无限次地重复某件事时,以及当我们想从一个区间中画出一个点时。第一种情况出现在我们讨论机器学习中的泛化错误时(第8章)。当我们想要讨论连续分布时会出现第二种情况,例如高斯分布(第6.5节)。出于我们的目的,缺乏精确性允许对概率进行更简短的介绍。

评论。在连续空间中,还有两个违反直觉的额外技术细节。首先,所有子集的集合(用于定义第6.1节中的事件空间A)表现不够好。A需要被限制在集补集、集交集和集并集下表现良好。其次,集合的大小(在离散空间中可以通过计算元素来获得)变得很棘手。集合的大小称为其测度。例如,离散集的基数、R中区间的长度和Rd中区域的体积都是度量。在集合运算下表现良好并且另外具有拓扑的集合称为Borel σ-代数。Betancourt详述了从集合论中仔细构造概率空间的过程,而没有陷入技术细节的泥潭;请参阅<https://tinyurl.com/yb3t6mfd>。对于更精确的结构,我们参考Billingsley(1995)和Jacod和Protter(2004)。

措施

Borel σ-代数

在本书中,我们考虑实值随机变量及其相关系数

响应 Borel σ -代数。我们将 RD 中具有值的随机变量视为实值随机变量的向量。 ◇

定义 6.1 (概率密度函数) 函数 $f: RD \rightarrow R$ 称为概率密度函数(pdf),如果

概率密度函数

$$1. \forall x \in R \quad 2. \int_{-\infty}^{\infty} f(x) dx = 1$$

其积分存在且

$$\int_{-\infty}^{\infty} f(x) dx = 1. \quad (6.15)$$

对于离散随机变量的概率质量函数(pmf),将 (6.15) 中的积分替换为求和 (6.12)。

观察到概率密度函数是非负且积分为 1 的任何函数 f 。我们通过以下方式将随机变量 X 与函数 f 相关联

$$P(a \leq X \leq b) = \int_a^b f(x) dx, \quad (6.16)$$

其中 $a, b \in R$ 和 $x \in R$ 是连续随机变量 X 的结果。状态 $x \in RD$ 通过考虑 $x \in R$ 的向量类似地定义。这种关联 (6.16) 称为法则或分布法则随机变量 X 。

评论。与离散随机变量相反,连续随机变量取特定值 $P(X = x)$ 的概率为零。
 $P(X = x)$ 是一组
 测量零。

这就像试图在 (6.16) 中指定一个区间,其中 $a = b$ 。 ◇

定义 6.2 (累积分布函数) 具有状态 $x \in RD$ 的多元实值随机变量 X 的累积分布函数(cdf)由下式给出
 $P(X \leq x)$ 分配功能

$$F_X(x) = P(X_1 \leq x_1, \dots, X_D \leq x_D), \text{ 其中 } X = [X_1, \dots, X_D], \quad (6.17)$$

$x = [x_1, \dots, x_D]$, 右边表示随机变量 X 取值小于或等于 x_i 的概率。

cdf 也可以表示为概率密度函数 $f(x)$ 的积分,因此

有cdfs,没有对应的
 pdf。

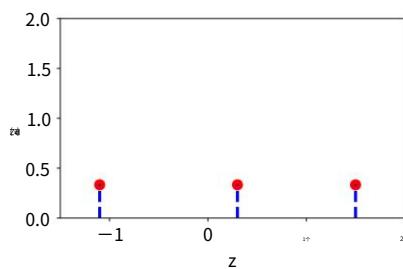
$$F_X(x) = \int_{-\infty}^{x_1} \dots \int_{-\infty}^{x_D} f(z_1, \dots, z_D) dz_1 \dots dz_D. \quad (6.18)$$

评论。我们重申,在谈论分布时实际上有两个不同的概念。首先是 pdf (由 $f(x)$ 表示)的概念,它是一个总和为 1 的非负函数。其次是随机变量 X 的规律,即随机变量 X 与 pdf $f(x)$ 的关联。 ◇

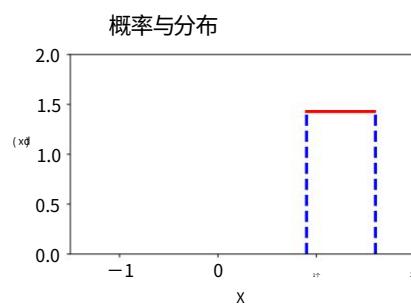
182

图 6.2 (a) 离散和 (b) 连续均匀的示例

分布。有关详细信息,请参见示例
6.3
分布。



(a) 离散分布



(b) 持续分发

对于本书的大部分内容,我们不会使用符号 $f(x)$ 和 $F(x)$,因为我们大多不需要区分 pdf 和 cdf。但是,我们需要注意第 6.7 节中的 pdf 和 cdf。

6.2.3 对比离散和连续分布

回想第 6.1.2 节,概率为正,总概率总和为 1。对于离散随机变量(见 (6.12)) ,这意味着每个状态的概率必须位于区间 $[0, 1]$ 内。

然而,对于连续随机变量,归一化(见 (6.15))并不意味着密度值对于所有值都小于或等于 1。我们在图 6.2 中使用离散和连续随机变量的均匀分布来说明这一点。

均匀分布

示例 6.3 我们考

虑两个均匀分布的示例,其中每个状态出现的可能性相同。这个例子说明了离散概率分布和连续概率分布之间的一些差异。

令 Z 为具有三个状态的离散均匀随机变量 $\{z = -1.1, z = 0.3, z = 1.5\}$ 。概率质量

作为概率值表:

	-1.1	0.3	1.5
$P(Z = z)$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

或者,我们可以将其视为一个图形(图 6.2(a)),其中我们使用状态可以位于 x 轴上的事实,而 y 轴表示特定状态的概率。图 6.2(a) 中的 y 轴被有意延长,因此它与图 6.2(b) 中的相同。

设 X 是取值范围为 $[0, 1]$ 的连续随机变量

$X \sim U[0, 1]$,如图 6.2(b) 所示。观察到高度

类型	“点概率”	“区间概率”	表 6.1 的命名法 概率分布。
离散的	$P(X = x)$ 概率质量函数	不适用	
连续的	$p(x)$ 概率密度函数 累积分布函数	$P(X \in x)$	

密度可以大于1。但是,它需要保持

$$\int_{0.9}^{1.6} p(x)dx = 1. \quad (6.19)$$

评论。关于离散概率分布还有一个额外的微妙之处。状态 z_1, \dots, z_d 原则上没有任何结构,即通常无法比较它们,例如 $z_1 = \text{red}$, $z_2 = \text{green}$, $z_3 = \text{blue}$ 。然而,在许多机器学习应用中,离散状态采用数值,例如, $z_1 = -1.1$, $z_2 = 0.3$, $z_3 = 1.5$, 我们可以说 $z_1 < z_2 < z_3$ 。离散状态作为 sume 数值特别有用,因为我们经常考虑随机变量的期望值(第 6.4.1 节)。 ◇ 不幸的是,机器学习文献使用的符号和命名法隐藏了样本空间 Ω 、目标空间 T 和随机变量 X 之间的区别。对于随机变量 X 的一组可能结果的值 x , 即 $x \in T$, $p(x)$ 表示概率我们认为随机变量 X 具有结果 x 的能力。对于离散随机变量,这写为 $P(X = x)$, 称为概率质量函数。 pmf 通常被称为“分布”。对于连续变量, $p(x)$ 称

为概率密度函数(通常称为密度)。更复杂的是,累积分布函数 $P(X \leq x)$ 通常也称为“分布”。在本章中,我们将使用符号 X 来指代单变量和多元随机变量,并分别用 x 和 x 表示状态。我们在表 6.1 中总结了术语。

结果 x 作为导致结果的参数

概率 $p(x)$ 。

评论。我们将不仅对离散概率质量函数而且对连续概率密度函数都使用“概率分布”这个表达式,尽管这在技术上是不正确的。与大多数机器学习文献一样,我们也依靠上下文来区分短语概率分布的不同用途。 ◇

6.3 求和规则、乘积规则和贝叶斯定理我们认为概率论是逻辑推理的扩展。正如我们在第 6.1.1 节中讨论的那样,这里介绍的概率规则遵循

自然而然地从满足需求 (Jaynes, 2003, 第 2 章)。概率建模 (第 8.4 节) 为设计机器学习方法提供了原则性基础。一旦我们定义了对应于数据和问题的不确定性的概率分布 (第 6.2 节), 就会发现只有两个基本规则, 求和规则和乘积规则。

回想一下 (6.9) 中的 $p(x, y)$ 是两个随机变量 x, y 的联合分布。分布 $p(x)$ 和 $p(y)$ 是相应的边际分布, $p(y | x)$ 是给定 x 时 y 的条件分布。鉴于第 6.2 节中离散和连续随机变量的边际概率和条件概率的定义, 我们现在可以提出概率论中的两个基本规则。

这两条规定

出现

自然地 (Jaynes, 2003)

从我们在

第 6.1.1 节。
求和规则

边缘化属性

第一条规则, 求和规则, 指出

$p(x, y)$ 如果 y 是离散的

$$p(x) = \sum_{y \in Y} p(x, y) \quad , \quad (6.20)$$

$p(x, y) dy$ 如果 y 是连续的
是

其中 Y 是随机变量 Y 的目标空间的状态。这意味着我们对随机变量 Y 的状态集 y 求和 (或积分)。求和规则也称为边缘化属性。

求和规则将联合分布与边际分布联系起来。通常, 当联合分布包含两个以上的随机变量时, 可以将求和规则应用于随机变量的任何子集, 从而导致可能存在多个随机变量的边际分布。更具体地说, 如果 $x = [x_1, \dots, x_D]$, 我们得到边际

$$p(x_i) = p(x_1, \dots, x_D) dx \setminus i \quad (6.21)$$

通过重复应用求和规则, 我们对除 x_i 之外的所有随机变量进行积分/求和, 用 $\setminus i$ 表示为“除了 i 之外的所有变量”。

评论。概率建模的许多计算挑战都归因于求和规则的应用。当有多个变量或离散变量有多个状态时, 求和规则归结为进行高维求和或积分。执行高维求和或积分通常在计算上很困难, 因为没有已知的多项式时间算法来精确计算它们。 ◇第二条规则, 称为乘积规则, 涉及联合分布

产品规则

通过条件分布

$$p(x, y) = p(y | x)p(x) \quad (6.22)$$

乘积规则可以解释为两个随机变量的每个联合分布都可以因式分解 (写为乘积)

其他两个分布。这两个因素是第一个随机变量 $p(x)$ 的边际分布，以及给定第一个随机变量 $p(y | x)$ 的第二个随机变量的条件分布。由于 $p(x, y)$ 中随机变量的顺序是任意的，乘积规则也意味着 $p(x, y) = p(x | y)p(y)$ 。准确地说，(6.22) 是用离散随机变量的概率质量函数表示的。对于连续随机变量，乘积规则用概率密度函数表示（第 6.2.3 节）。

在机器学习和贝叶斯统计中，我们通常有兴趣在观察到其他随机变量的情况下对未观察到的（潜在的）随机变量进行推断。让我们假设我们有一些关于未观察到的随机变量 x 的先验知识 $p(x)$ 以及 x 和我们可以观察到的第二个随机变量 y 之间的某种关系 $p(y | x)$ 。如果我们观察 y ，我们可以使用贝叶斯定理在给定观察到的 y 值的情况下得出一些关于 x 的结论。贝叶斯定理（也称为贝叶斯定理、贝叶斯法则或贝叶斯定律）

$$\frac{p(x | y)}{\text{后部}} = \frac{p(y | x)p(x)}{p(y)} \quad (6.23)$$

可能性的
贝叶斯规则
事先的
贝叶斯定律
证据

是 (6.22) 中乘积规则的直接结果，因为

$$p(x, y) = p(x | y)p(y) \quad (6.24)$$

和

$$p(x, y) = p(y | x)p(x) \quad (6.25)$$

以便

$$| y)p(y) = p(y | x)p(x) \Leftrightarrow p(x | y) = \frac{p(y | x)p(x)}{p(y)} . \quad (6.26)$$

在 (6.23) 中， $p(x)$ 是先验，它封装了我们的主观先验先验。在观察任何数据之前了解未观察到的（潜在）变量 x 。我们可以选择任何对我们有意义的先验，但关键是要确保先验对所有似是而非的 x 具有非零 pdf（或 pmf），即使它们非常罕见。

可能性 $p(y | x)$ 描述了 x 和 y 是如何相关的，并且在可能性中。在离散概率分布的情况下，如果我们知道潜在变量 x ，则它是数据 y 的概率。请注意，可能性不是 x 中的分布，而仅是 y 中的分布。我们将 $p(y | x)$ 称为“ x （给定 y ）的可能性”或“给定 x 的 y 的概率”，但绝不是 y 的可能性（MacKay, 2003）。

可能性有时也是
称为“测量
模型”。

后验 $p(x | y)$ 是贝叶斯后验统计中感兴趣的数量。因为它准确地表达了我们感兴趣的东西，即我们在观察 y 之后对 x 的了解。

数量

$$p(y) := p(y | x)p(x)dx = \text{EX}[p(y | x)] \quad (6.27)$$

边际可能性
证据

是边际可能性/证据。(6.27) 的右侧使用了我们在 6.4.1 节中定义的期望算子。根据定义，边际似然对 (6.23) 中关于潜在变量 x 的分子进行积分。因此，边际似然与 x 无关，它确保后验 $p(x | y)$ 被归一化。边际似然也可以解释为我们对先验 $p(x)$ 进行期望的预期似然。除了后验归一化之外，边际似然在贝叶斯模型选择中也起着重要作用，我们将在第 8.6 节中讨论。由于 (8.44) 中的积分，证据通常难以计算。

贝叶斯定理也被称
为
“概率逆。”

贝叶斯定理 (6.23) 允许我们反转由似然给出的 x 和 y 之间的关系。因此，贝叶斯定理有时也被称为概率逆。我们将在 8.4 节进一步讨论贝叶斯定理。

概率逆

评论。在贝叶斯统计中，后验分布是感兴趣的数量，因为它封装了先验和数据中的所有可用信息。可以关注后验的一些统计量，例如后验的最大值，我们将在第 8.3 节中讨论，而不是四处携带后验。然而，关注一些后验统计会导致信息丢失。如果我们在更大的背景下思考，那么后验可以在决策系统中使用，并且拥有完整的后验可能非常有用，并且会导致对干扰具有鲁棒性的决策。例如，在基于模型的强化学习的背景下，Deisenroth 等人。(2015) 表明，使用似是而非的转换函数的完整后验分布会导致非常快的（数据/样本有效）学习，而关注后验的最大值会导致一致的失败。因此，拥有完整的后验对于下游任务非常有用。在第 9 章中，我们将在线性回归的背景下继续讨论。◇

6.4 汇总统计和独立性

我们通常对总结随机变量集和比较随机变量对感兴趣。随机变量的统计量是该随机变量的确定性函数。分布的汇总统计提供了一个关于随机变量行为方式的有用视图，顾名思义，它提供了汇总和表征分布的数字。我们描述了均值和方差，这是两个众所周知的汇总统计数据。然后我们讨论比较一对随机变量的两种方法：第一，如何说两个随机变量是独立的；其次，如何计算它们之间的内积。

6.4.1 均值和协方差

均值和(协)方差通常可用于描述概率分布的属性(期望值和分布)。我们将在6.6节中看到,有一个有用的分布族(称为指数族),其中随机变量的统计量包含所有可能的信息。

期望值的概念是机器学习的核心,概率本身的基本概念可以从期望值中推导出来(Whittle, 2000)。

定义6.3(预期值)。函数 g 的期望值: $R \rightarrow$ 单变量连续随机变量 $X \quad p(x)$ 的期望值 R 由下式给出

$$E[X[g(x)] = \int_x g(x)p(x)dx. \quad (6.28)$$

相应地,离散随机变量 $X \quad p(x)$ 的函数 g 的期望值由下式给出

$$E[X[g(x)] = \sum_{x \in X} g(x)p(x), \quad (6.29)$$

其中 X 是随机变量 X 的可能结果集(目标空间)。

在本节中,我们考虑具有数值结果的离散随机变量。这可以通过观察函数 g 将实数作为输入来看出。

评论。我们将多元随机变量 X 视为单变量随机变量 $[X_1, \dots, X_D]$ 。对于多元随机变量,我们明智地定义期望值元素

a函数的期望值

随机变量有时被称为

作为无意识的法则

$$\begin{aligned} E[X_1[g(x_1)] & \\ E[X[g(x)] = & \vdots \quad \in R^D, \\ E[X_D[g(x_D)] & \end{aligned} \quad (6.30)$$

其中下标 Ex_d 表示我们正在获取关于向量 x 的第 d 个元素的期望值。 ◇

定义6.3将符号 E 的含义定义为运算符,指示我们应该对概率密度(对于连续分布)或所有状态的总和(对于离散分布)进行积分。均值的定义(定义6.4)是期望值的一个特例,通过选择 g 作为恒等函数获得。

定义6.4(平均值)。具有状态均值的随机变量 X 的均值

$x \in \mathbb{R}^D$ 是平均值, 定义为

$$\begin{aligned} & \text{EX}_1[x_1] \\ \text{EX}[x] = & \vdots \quad D \in \mathbb{R}_+, \\ & \text{EX}_D[x_D] \end{aligned} \tag{6.31}$$

在哪里

$$\text{EX}_d[x_d] := \begin{cases} \int_{\mathcal{X}} x_d p(x_d) dx_d & \text{如果 } X \text{ 是连续随机变量} \\ \sum_{x_i \in \mathcal{X}} x_i p(x_d = x_i) & \text{如果 } X \text{ 是离散随机变量} \end{cases} \tag{6.32}$$

对于 $d = 1, \dots, D$, 其中下标 d 表示 x 对应的维度。积分和求和是在随机变量 X 的目标空间中的状态 X 上。

中位数 在一个维度上, “平均”还有另外两个直观的概念, 即中位数和众数。中位数是“中间”值, 如果

模式 我们对值进行排序, 即 50% 的值大于中值, 50% 的值小于中值。通过考虑 cdf (定义 6.2) 为 0.5 的值, 可以将这个想法推广到连续值。

对于不对称或长尾的分布, 中位数提供了一个典型值的估计值, 它比平均值更接近人类的直觉。此外, 中位数比均值对异常值更稳健。将中位数推广到更高维度并非易事, 因为没有明显的方法可以在多维中“排序”(Hallin 等人, 2010 年; Kong 和 Mizera, 2012 年)。模式是最常出现的值。对于离散随机变量, 模式定义为具有最高出现频率的 x 值。对于连续随机变量, 众数定义为密度 $p(x)$ 中的峰值。一个特定的密度 $p(x)$ 可能有不止一种模式, 而且在高维分布中可能有非常多的模式。因此, 找到分布的所有模式在计算上可能具有挑战性。

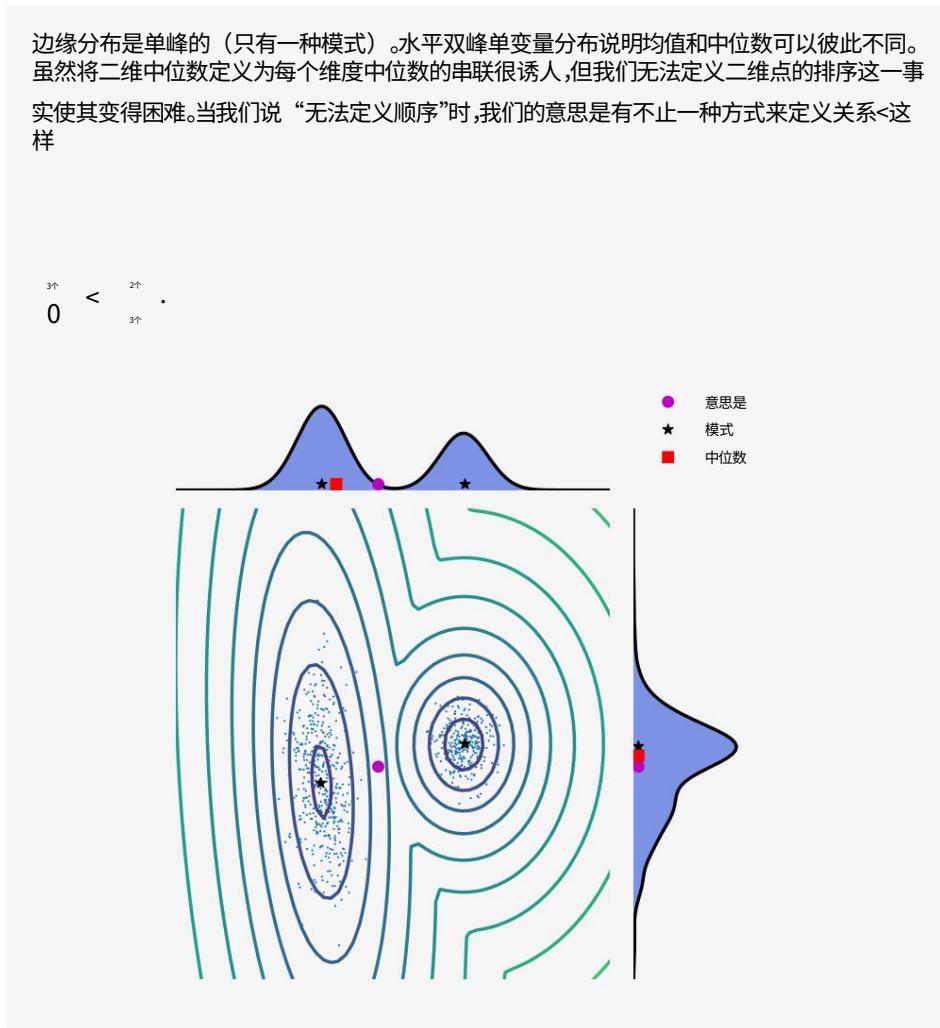
示例 6.4 考虑

图 6.2 中所示的二维分布:

$$p(x) = 0.4 N(x; \begin{pmatrix} 10 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}) + 0.6 \text{ 牛顿} \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 8.4 & 2.0 \\ 2.0 & 1.7 \end{pmatrix}. \tag{6.33}$$

我们将在第 6.5 节中定义高斯分布 $N(\mu, \sigma^2)$, 还显示了其在每个维度中相应的边缘分布。观察到分布是双峰的 (有两种模式), 但其中一种

边缘分布是单峰的（只有一种模式）。水平双峰单变量分布说明均值和中位数可以彼此不同。虽然将二维中位数定义为每个维度中位数的串联很诱人，但我们无法定义二维点的排序这一事实使其变得困难。当我们说“无法定义顺序”时，我们的意思是有不止一种方式来定义关系 \prec 这样



评论。期望值（定义 6.3）是一个线性算子。例如，给定实值函数 $f(x) = ag(x) + bh(x)$ 其中 $a, b \in \mathbb{R}$ 且 $x \in \mathbb{R}^d$ ，我们得到

$$\mathbb{E}[f(x)] = f(x)p(x)dx \quad (6.34a)$$

$$= [ag(x) + bh(x)]p(x)dx \quad (6.34b)$$

$$= ag(x)p(x)dx + bh(x)p(x)dx \quad (6.34c)$$

$$= a\mathbb{E}[g(x)] + b\mathbb{E}[h(x)] \quad (6.34d)$$



对于两个随机变量，我们可能希望表征它们对应的

互相拒绝。协方差直观地表示随机变量相互依赖程度的概念。

协方差

定义 6.5 (协方差 (单变量))。两个单变量随机变量 $X, Y \in R$ 之间的协方差由它们与各自均值的偏差的预期乘积给出,即

$$\text{Cov}_X, Y [x, y] := E[X, Y (x - E[X])(y - E[Y])]. \quad (6.35)$$

术语:的协方差

多元随机变量 $\text{Cov}[x, y]$ 有时称为

评论。当与期望或协方差相关的随机变量的参数明确时,下标通常被抑制 (例如, $E[X]$ 通常写为 $E[x]$)。 ◇利用期望的线性,定义 6.5 中的表达式可以改写为乘积的期望值减去期望值的乘积,即,

交叉协方差,具有协方差

参考 $\text{Cov}[x, x]$ 。

方差

标准偏差

变量与其自身 $\text{Cov}[x, x]$ 的协方差称为方差,记为 $V[X]$ 。方差的平方根称为标准差,通常用 $\sigma(x)$ 表示。协方差的概念可以推广到多元随机变量。

协方差

定义 6.6 (协方差 (多变量))。如果我们考虑两个分别具有状态 $x \in RD$ 和 $y \in RE$ 的多元随机变量 X 和 Y ,则 X 和 Y 之间的协方差定义为

$$\text{Cov}[x, y] = E[xy] - E[x]E[y] \quad = \text{冠状病毒}[y, x] \in R_{-}^{n \times n}. \quad (6.37)$$

定义 6.6 可以在两个论点中应用相同的多元随机变量,从而产生一个有用的概念,可以直观地捕捉随机变量的“分布”。对于多元随机变量,方差描述了随机变量的各个维度之间的关系。

方差

定义 6.7 (方差)。具有状态 $x \in RD$ 和均值向量 $\mu \in RD$ 的随机变量 X 的方差定义为

$$V[X] = \text{Cov}[X, X] \quad (6.38a)$$

$$= E[(x - \mu)(x - \mu)] = E[xx] - E[x]E[x] \quad (6.38b)$$

$$\begin{aligned} &= \text{Cov}[x_1, x_1] \text{ Cov}[x_1, x_2] \dots \text{冠状病毒}[x_1, x_D] \\ &= \text{Cov}[x_2, x_1] \text{ Cov}[x_2, x_2] \dots \text{冠状病毒}[x_2, x_D] \\ &\vdots \quad \vdots \quad \ddots \quad \vdots \quad \vdots \end{aligned} \quad (6.38c)$$

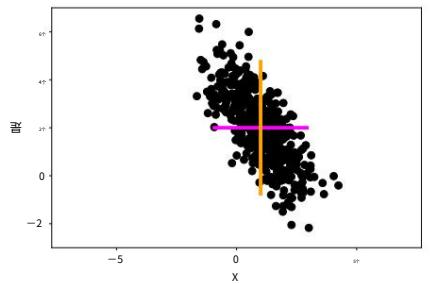
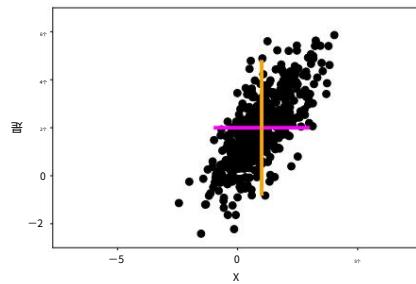
$$\text{冠状病毒}[x_D, x_1] \quad \cdots \quad \text{冠状病毒}[x_D, x_D]$$

协方差矩阵

(6.38c) 中的 $D \times D$ 矩阵称为多元随机变量 X 的协方差矩阵。协方差矩阵是对称的半正定矩阵,它告诉我们一些关于数据分布的信息。在其对角线上,协方差矩阵包含边缘的方差

边缘

6.4 汇总统计和独立性

(a) x 和 y 负相关。(b) x 和 y 正相关。图 6.3 二维数
据集

沿每个轴 (彩色线) 的均值
和方差相同但不同

协方差。

$$p(x_i) = p(x_1, \dots, x_D) dx \setminus i, \quad (6.39)$$

其中 “\i” 表示 “除 i 之外的所有变量”。非对角线项是 $i, j = 1$ 的交叉协方差项 $\text{Cov}[x_i, x_j]$ 。互协方差

评论。在本书中,我们通常假设协方差矩阵是正定的,以便更好地理解。因此,我们不讨论导致半正定(低秩)协方差矩阵的极端情况。当我们想要比较不同对随机变量之间的协方差时,结果发现每个随机变量的方差影响协方差的值。协方差的标准化版本称为相关性。

定义 6.8 (相关性)。两个随机变量 X, Y 之间的相关性由下式给出

$$\text{校正 } [x, y] = \frac{\text{Cov}[x, y]}{\sqrt{V[x]V[y]}} \in [-1, 1]. \quad (6.40)$$

相关矩阵是标准化随机变量的协方差矩阵, $x/\sigma(x)$ 。换句话说,每个随机变量除以相关矩阵中的标准偏差(方差的平方根)。

协方差(和相关性)表明两个随机变量是如何相关的;见图 6.3。正相关 $\text{corr}[x, y]$ 意味着当 x 增长时, y 也有望增长。负相关意味着随着 x 增加, y 减少。

6.4.2 经验均值和协方差

第 6.4.1 节中的定义通常也称为总体均值、总体均值和协方差,因为它指的是总体的真实统计数据。在机器学习中,我们需要从数据的经验观察中学习。考虑一个随机变量 X 。有两个概念 和协方差
步骤可以从

人口统计要实现实证统计。首先,我们利用我们有一个有限数据集 (大小为N)这一事实来构建一个经验统计量,该统计量是有限数量的相同随机变量 X_1, \dots, X_N 的函数。 \dots, X_N 。其次,我们观察数据,即我们查看每个随机变量的实现 x_N 并应用经验值 x_1, \dots, x_N ,统计。

经验均值样本均值 具体来说,对于均值 (定义 6.4),给定一个特定的数据集,我们可以获得均值的估计值,称为经验均值或样本均值。这同样适用于经验协方差。

经验平均值 定义 6.9 (经验均值和协方差)。经验均值向量是每个变量观测值的算术平均值,定义为

$$\bar{x} := \frac{1}{n} \sum_{n=1}^N x_n, \quad (6.41)$$

其中 $x_n \in RD$ 。

经验协方差 与经验均值类似,经验协方差矩阵是一个 $D \times D$ 矩阵

$$\Sigma := \frac{1}{n} \sum_{n=1}^N (x_n - \bar{x})(x_n - \bar{x})^\top. \quad (6.42)$$

在整本书中,我们使用经验协方差,这是一种有偏估计。

要计算特定数据集的统计数据,我们将使用实现 (观察) x_1, \dots, x_N 并使用 (6.41) 和 (6.42)。经验协方差矩阵是对称的、半正定的 (见第 3.2.3 节)。

无偏 (有时称为校正) 协方差具有

6.4.3 方差的三个表达式

分母中的因子 $N - 1$

我们现在关注单个随机变量 X ,并使用前面的经验公式推导出方差的三个可能表达式。

而不是 N 。

以下推导对于总体方差是相同的,只是我们需要注意积分。方差的标准定义,对应于协方差的定义 (定义 6.5),是随机变量 X 与其期望值 μ 的平方偏差的期望,即,

推导是
本章末尾的练习。

$$VX[x] := E[X((x - \mu)^2)]. \quad (6.43)$$

(6.43) 中的期望值和平均值 $\mu = EX(x)$ 是使用 (6.32) 计算的,具体取决于 X 是离散随机变量还是连续随机变量。(6.43) 中表示的方差是新随机变量 $Z := (X - \mu)$ 的均值

当根据经验估计 (6.43) 中的方差时,我们需要求助于一种两次通过的算法:第一次使用 (6.41) 通过数据计算平均值 μ ,然后第二次使用该估计值 μ 计算

方差。事实证明,我们可以通过重新排列项来避免两次传递。(6.43)中的公式可以转化为方差的所谓raw-score raw-score formula公式:

方差

$$VX[x] = EX[x^2] - (EX[x])^2 \quad (6.44)$$

(6.44)中的表达式可以记为“平方的平均值减去平均值的平方”。可以一次通过经验计算

通过数据,因为我们可以同时累积 x_i (计算平均值)和 x^2 ,其中 x_i 是第 i 个观察值。不幸的是,

如果以这种方式实现这两项,它可能在数值上不稳定。方差的原始分数版本在机器学习中很有用,例如,在推导偏差-方差分解时 (Bishop, 2006)。

在 (6.44) 中是巨大的
并且近似相等,我们可
能会遇到

理解方差的第三种方法是它是成对差异的总和

浮点运算中不必要的数值
精度损失。

所有对观察之间的参考。考虑样本 x_1, \dots, x_N ,随机变量 X 的实现,我们计算 x_i 和 x_j 之间 的平方差。通过扩大平方,我们可以证明 N^2 个成对差异的总和是观测值的经验方差:

$$\frac{1}{N^2} \sum_{i,j=1}^{N^2} (x_i - x_j)^2 = \frac{1}{N^2} \sum_{i=1}^{N^2} x_i^2 - \frac{1}{N^2} \sum_{i=1}^{N^2} x_i^2 \quad (6.45)$$

我们看到 (6.45) 是原始分数表达式 (6.44) 的两倍。这意味着我们可以将成对距离之和(其中有 N^2 个)表示为与均值(其中有 N 个)的偏差之和。从几何学上讲,这意味着成对距离与距点集中心的距离之间存在等价性。从计算的角度来看,这意味着通过计算均值(求和中的 N 项),然后计算方差(求和中的 N 项),我们可以获得一个表达式((6.45) 的左侧)有 N^2 项。

6.4.4 随机变量的求和和变换

我们可能想要对教科书分布无法很好解释的现象建模(我们在第 6.5 和 6.6 节中介绍了一些),因此可能会对随机变量进行简单的操作(例如添加两个随机变量变量)。

考虑两个状态为 $x, y \in RD$ 的随机变量 X, Y ,然后:

$$E[x + y] = E[x] + E[y] \quad (6.46)$$

$$E[x-y] = E[x]-E[y] \quad (6.47)$$

$$V[x+y] = V[x] + V[y] + Cov[x, y] + Cov[y, x] \quad (6.48)$$

$$V[x-y] = V[x] + V[y] - Cov[x, y] - Cov[y, x]. \quad (6.49)$$

当涉及到随机变量的仿射变换时,均值和(协)方差表现出一些有用的特性。考虑一个随机变量

X 具有均值 μ 和协方差矩阵 Σ 以及 x 的(确定性)仿射变换 $y = Ax + b$ 。那么 y 本身就是一个随机变量,其均值向量和协方差矩阵由下式给出

$$E[y] = E[Ax + b] = AEX[x] + b = A\mu + b \quad , \quad (6.50)$$

$$VY[y] = VX[Ax + b] = VX[Ax] = AVX[x]A = A\Sigma A \quad , \quad (6.51)$$

这可以通过使用均值和的
定义直接显示

分别。此外, $Cov[x, y] =$

$$E[x(Ax + b)] - E[x]E[Ax + b] = \mu b - \mu\mu A \quad (6.52a)$$

$$= E[x]b + E[xx]A \quad (6.52b)$$

$$= \mu b - \mu b + E[xx] - \mu\mu A \quad (6.52c)$$

$$\stackrel{(6.38b)}{=} \Sigma A \quad , \quad (6.52d)$$

其中 $\Sigma = E[xx] - \mu\mu$ 是 X 的协方差。

6.4.5 统计独立性

统计独立性

定义 6.10 (独立性)。两个随机变量 X, Y 是统计独立的当且仅当

$$p(x, y) = p(x)p(y) \quad (6.53)$$

直观地说,如果 y 的值(一旦已知)不添加任何关于 x 的附加信息(反之亦然),则两个随机变量 X 和 Y 是独立的。如果 X, Y 是(统计上)独立的,那么

- $p(y | x) = p(y)$
- $p(x | y) = p(x)$
- $VX, Y[x + y] = VX[x] + VY[y]$
- $CovX, Y[x, y] = 0$

最后一点可能不成立,即两个随机变量的协方差可以为零,但在统计上并不独立。要理解原因,请回想一下协方差仅衡量线性相关性。因此,非线性相关的随机变量可能具有协方差

零。

示例 6.5 考虑均

值为零的随机变量 X ($EX[x] = 0$) 和 $EX[x]$ (因此, Y 依赖于 X)并考虑 X 和 Y 之间的协方差(6.36)。但这给了 $= 0$ 。让 $y = x$

$$Cov[x, y] = E[xy] - E[x]E[y] = E[x^2] - E[x]^2 = 0 \quad (6.54)$$

在机器学习中,我们经常考虑可以模化的问题
 elel 作为独立同分布(iid)随机变量,独立且 X_1, \dots, X_N 。对于两个以上的随机变量,“独立”一词 (定义 6.10)通常是指相互独立的随机变量,其中所有子集都是独立的 (参见 Pollard (2002, 同分布 第 4 章) 和 Jacob 和 Protter (2004, 第 3 章))。短语 “同分布”意味着所有随机变量 $\stackrel{iid}{\sim}$ 都来自同一分布。

机器学习中另一个重要的概念是条件独立性。

定义 6.11 (条件独立)。两个随机变量X和Y条件独立给定Z当且仅当

条件独立

$$p(x, y | z) = p(x | z)p(y | z) \text{ 对于所有 } z \in Z \quad , \quad (6.55)$$

其中Z是随机变量 Z 的状态集。我们写 $X \perp\!\!\!\perp Y | Z$ 表示X在给定Z的情况下条件独立于Y。

定义 6.11 要求 (6.55) 中的关系必须对每个z值都成立。(6.55) 的解释可以理解为“给定关于 z 的知识, x和y因式分解的分布”。如果我们写 $X \perp\!\!\!\perp Y |$,则独立性可以作为条件独立性的特例。∅.利用概率的乘积法则(6.22),我们可以展开(6.55)的左边得到

$$p(x, y | z) = p(x | y, z)p(y | z) \quad (6.56)$$

通过比较 (6.55) 和 (6.56) 的右侧,我们看到 $p(y | z)$ 出现在它们两个中,因此

$$p(x | y, z) = p(x | z) \quad (6.57)$$

方程 (6.57) 提供了条件独立性的另一种定义,即 $X \perp\!\!\!\perp Y | Z$ 。这种替代表述提供了“假设我们知道z,关于y的知识不会改变我们对x 的知识”的解释。

6.4.6 随机变量的内积

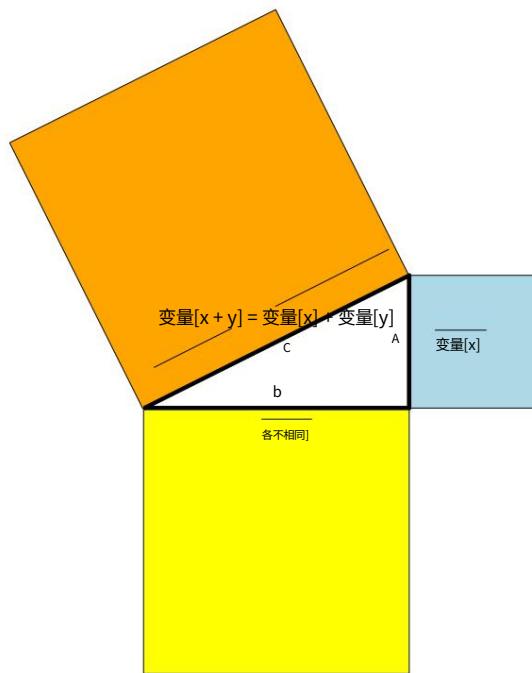
回忆一下 3.2 节中内积的定义。我们可以定义随机变量之间的内积,我们将在本节中简要介绍。如果我们有两个不相关的随机变量X, Y
 之间
 , 然后
 多元随机变量可以是

$$V[x + y] = V[x] + V[y] \quad (6.58) \quad \text{以类似的方式对待}$$

由于方差是以平方为单位测量的,这看起来很像直角三角形c + b的毕达哥拉斯定理。在下文中,我们看看是否可以在 (6.58) 中找到不相关随机变量的方差关系的几何解释。

图 6.1 随机
变量的几何。如
果随机变量 X

和 Y 互不相
关，在对应的向量空间
是正交向量，符合
勾股定理。



随机变量可以被认为是向量空间中的向量，我们可以定义内积来获得随机变量的几何特性（Eaton, 2007）。如果我们定义

$$X, Y := \text{Cov}[x, y] \quad (6.59)$$

对于零均值随机变量 X 和 Y ，我们得到一个内积。我们 $\text{Cov}[x, x] = 0$

\Leftrightarrow 看到协方差是对称的、正定的和线性的
 $x = 0$ 争论。随机变量的长度是
 $\text{Cov}[ax + z, y] = a$
 $\text{Cov}[x, y] +$
 $\text{Cov}[z, y]$ 对于 $a \in \mathbb{R}$ 。

$$\|X\| = \text{Cov}[x, x] = V[x] = \sigma[x], \quad (6.60)$$

即，它的标准偏差。随机变量“越长”，它的不确定性就越大；长度为 0 的随机变量是确定性的。

如果我们观察两个随机变量 X, Y 之间的角度 θ ，我们得到

$$Y \cos \frac{\theta}{\theta = \|X\| \|Y\|} = \frac{\text{冠状病毒}[x, y]}{V[x]V[y]}, \quad (6.61)$$

这是两个随机变量之间的相关性（定义 6.8）。这意味着当我们从几何角度考虑它们时，我们可以将相关性视为两个随机变量之间夹角的余弦值。我们从定义 3.7 知道 $X \perp Y \Leftrightarrow \text{Cov}[x, y] = 0$ 。在我们的例子中，这意味着当且仅当 $\text{Cov}[x, y] = 0$ 时 X 和 Y 是正交的，即它们是不相关的。图 6.1 说明了这种关系。

评论。虽然很想使用欧几里得距离（构造

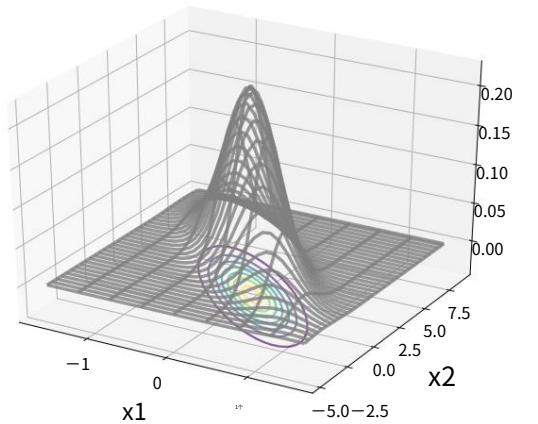


图 6.1 高斯分布
两个随机变量 x_1 和 x_2 的分布。

从前面的内积定义)来比较概率分布,不幸的是,这不是获得分布之间距离的最佳方法。回想一下,概率质量(或密度)是正的并且需要加起来为 1。这些约束意味着分布存在于称为统计流形的东西上。对这种概率分布空间的研究称为信息几何。计算分布之间的距离通常使用 Kullback-Leibler 散度来完成,它是考虑统计流形属性的距离的概括。就像欧几里德距离是度量的特例(第 3.3 节)一样,Kullback-Leibler 散度是两个更一般的散度类的特例,称为 Bregman 散度和 f -散度。散度的研究超出了本书的范围,我们可以参考信息几何领域的创始人之一阿马里(Amari,2016 年)最近出版的一本书来了解更多细节。 ◇

6.5 高斯分布

高斯分布是对连续值随机变量研究最深入的概率分布。它也被称为正态正态分布。它的最重要性源于这样一个事实,即它具有许多计算高斯方便的属性,我们将在下面讨论这些属性。特别是,我们将使用它来定义线性回归的似然和先验(第 9 章),并考虑混合高斯分布来进行密度估计(第 11 章)。

分配出现
很自然地,当我们考虑独立同分布随机数的总和时

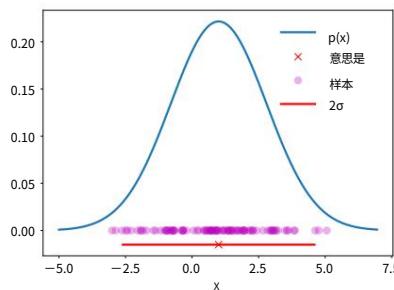
机器学习的许多其他领域也受益于使用高斯分布,例如高斯过程、变分推理和强化学习。它还广泛用于其他应用领域,例如信号处理(例如,卡尔曼滤波器)、控制(例如,线性二次调节器)和统计(例如,假设检验)。

变量。这被称为
中心极限定理
(Grinstead 和 Snell,
1997)。

198

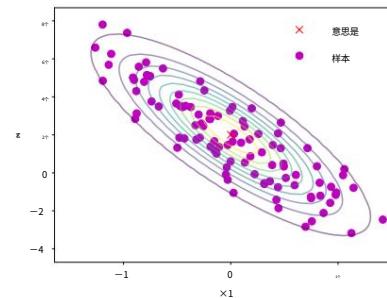
图 6.2 高斯分
布

覆盖 100 个样本。(a)
一维情况; (b) 二维
案件。



(a) 单变量 (一维)高斯分布;
红色十字显示平均值和红色
线显示方差的程度。

概率与分布



(b) 多元 (二维)高斯分布,从顶部看。红叉表示平均值,彩色线
表示密度的轮廓线。

对于单变量随机变量,高斯分布的密度为

$$p(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right). \quad (6.62)$$

多元高斯分布完全由均值向量 μ 和协方差矩阵 Σ 表征, 定义为

多元的
高斯分布均值向量
协方差矩阵

也称为多元正态分布
分配。

其中 $x \in \mathbb{R}^d$ 。我们写 $p(x) = N(x | \mu, \Sigma)$ 或 $X \sim N(\mu, \Sigma)$ 。图 6.1 显示了一个双变量高斯 (网格), 以及相应的轮廓图。图 6.2 显示了具有相应样本的单变量高斯和双变量高斯。具有零均值和恒等协方差的高斯分布的特殊情况, 即 $\mu = 0$ 和 $\Sigma = I$, 称为标准正态分布。

标准正常
分配

高斯分布广泛用于统计估计和机器学习, 因为它们具有边际分布和条件分布的封闭形式表达式。在第 9 章中, 我们将这些封闭形式的表达式广泛用于线性回归。使用高斯随机变量建模的一个主要优点是通常不需要变量转换 (第 6.7 节)。由于高斯分布完全由其均值和协方差指定, 我们通常可以通过将变换应用于随机变量的均值和协方差来获得变换后的分布。

6.5.1 高斯的边缘和条件是高斯

在下文中, 我们介绍了多元随机变量的一般情况下的边缘化和调节。如果这在第一次阅读时令人困惑, 建议读者改为考虑两个单变量随机变量。设 X 和 Y 是两个多元随机变量, 可能有

不同的维度。为了考虑应用概率求和规则的效果和调节的效果，我们根据级联状态 $[x, y]$ 显式地写出高斯分布，

$$p(x, y) = N \frac{\mu_x}{\mu_y}, \frac{\Sigma_{xx} \Sigma_{xy} \Sigma_{yx}}{\Sigma_{yy}}. \quad (6.64)$$

其中 $\Sigma_{xx} = \text{Cov}[x, x]$ 和 $\Sigma_{yy} = \text{Cov}[y, y]$ 分别是 x 和 y 的边际协方差矩阵， $\Sigma_{xy} = \text{Cov}[x, y]$ 是 x 和 y 之间的交叉协方差矩阵。

条件分布 $p(x | y)$ 也是高斯分布的（如图所示 figure 6.3(c)）并由（源自 Bishop, 2006 年的第 2.3 节）给出

$$p(x | y) = N \mu_x | y, \Sigma_x | \text{是} \quad (6.65)$$

$$\text{是} = \mu_x + \Sigma_{xy} \Sigma_{yy}^{-1} (y - \mu_y) \quad (6.66)$$

$$\Sigma_x | y = \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx}. \quad (6.67)$$

请注意，在 (6.66) 中计算均值时， y 值是一个观测值，不再是随机的。

评论。条件高斯分布出现在很多地方，我们对后验分布感兴趣：

- 卡尔曼滤波器 (Kalman, 1960) 是信号处理中最重要的状态估计算法之一，它除了计算联合分布的高斯条件外什么都不做 (Deisenroth 和 Ohlsson, 2011 年; Sarkka 等, 2013 年)。
- 高斯过程 (Rasmussen 和 Williams, 2006 年)，它是函数分布的实际实现。在高斯过程中，我们假设随机变量具有联合高斯性。通过对观察到的数据进行 (高斯) 调节，我们可以确定函数的后验分布。
- 潜在线性高斯模型 (Roweis 和 Ghahramani, 1999 年; Murphy, 2012 年)，其中包括概率主成分分析 (PPCA) (Tipping 和 Bishop, 1999 年)。我们将在 10.7 节中更详细地研究 PPCA。



联合高斯分布 $p(x, y)$ (见 (6.64)) 的边际分布 $p(x)$ 本身是高斯分布，通过应用求和规则 (6.20) 计算并由下式给出

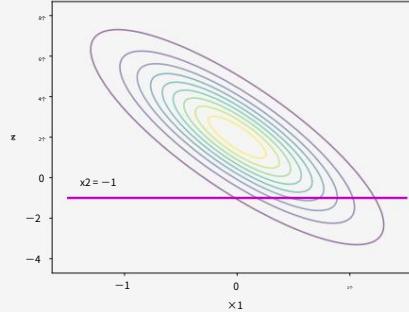
$$p(x) = p(x, y) dy = N x | \mu_x, \Sigma_{xx}. \quad (6.68)$$

相应地结果适用于 $p(y)$ ，它是通过对 x 进行边缘化而获得的。直观地，查看 (6.64) 中的联合分布，我们忽略 (即积分掉) 我们不感兴趣的所有内容。如图 6.3(b) 所示。

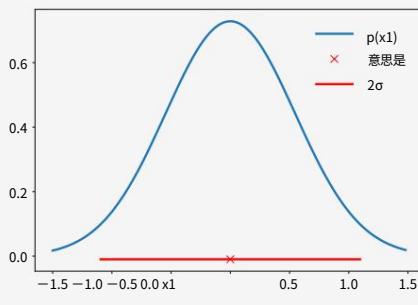
例 6.6

图 6.3 (a) 双变量高斯分布；
(b) 联合高斯的边缘

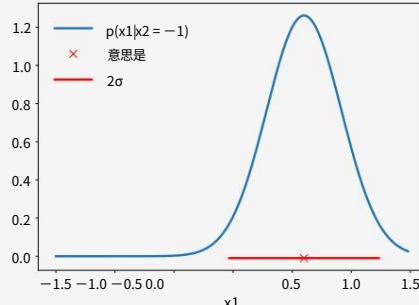
分布是高斯分布；(c) 有条件的
一个分布高斯也是高斯。



(a) 双变量高斯。



(b) 边际分布。



(c) 有条件的分配。

考虑双变量高斯分布（如图 6.3 所示）：

$$p(x_1, x_2) = N \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 0.3 & -1 \\ -1 & 1.5 \end{pmatrix}. \quad (6.69)$$

我们可以计算单变量高斯的参数，以 $x_2 = -1$ 为条件，通过应用 (6.66) 和 (6.67) 分别获得均值和方差。从数字上看，这是

$$x_2 = -1 = 0 + (-1) \cdot 0.2 \cdot (-1 - 2) = 0.6 \mu_{x1} | \quad (6.70)$$

和

$$\sigma_{x1 | x2 = -1}^2 = 0.3 - (-1) \cdot 0.2 \cdot (-1) = 0.1. \quad (6.71)$$

因此，条件高斯由下式给出

$$p(x_1 | x_2 = -1) = N(0.6, 0.1). \quad (6.72)$$

相反，边缘分布 $p(x_1)$ 可以通过应用 (6.68) 获得，它本质上是使用随机变量 x_1 的均值和方差，给我们

$$p(x_1) = N(0, 0.3). \quad (6.73)$$

6.5.2 高斯密度乘积

对于线性回归（第 9 章）,我们需要计算高斯似然。此外,我们可能希望假设一个高斯先验（第 9.3 节）。

我们应用贝叶斯定理来计算后验,这导致似然和先验的乘法,即两个高斯密度的乘法。两个高斯分布的乘积 $N(x|a, b) \cdot N(x|c, d)$ 是由 $a, c \in \mathbb{R}$ 缩放的高斯分布,由 $c = N(x|a + c, A + B)$ 给出,与

本章末尾的练习。

$$C = (A^{-1} + B^{-1})^{-1} \quad (6.74)$$

$$c = C (A^{-1} + B^{-1})^{-1} \quad (6.75)$$

$$c = (2\pi)^{-\frac{D}{2}} |A+B|^{-\frac{1}{2}} \exp(-\frac{1}{2}(a-b)^T (A+B)^{-1} (a-b)) \quad (6.76)$$

缩放常数 c 本身可以写成 a 或 b 中的高斯密度形式,并带有“膨胀的”协方差矩阵 $A + B$,即 $c = N(a|b, A + B) = N(b|a, A + B)$ 。为了符号方便,我们有时会使用 $N(x|m, S)$ 来描述高斯密度的函数形式,即使 x 不是随机变量。我们刚刚在前面的演示中 .

完成了这一点,当时我们写了

$$c = N(a|b, A + B) = N(b|a, A + B) \quad (6.77)$$

这里, a 和 b 都不是随机变量。然而,以这种方式编写 c 比 (6.76) 更紧凑。 ◇

6.5.3 求和和线性变换

如果 X, Y 是独立的高斯随机变量 (即联合分布为 $p(x, y) = p(x)p(y)$) ,其中 $p(x) = N(x|\mu_x, \Sigma_x)$ 和 $p(y) = N(y|\mu_y, \Sigma_y)$,那么 $x + y$ 也是高斯分布的并且由下式给出

$$p(x + y) = N(\mu_x + \mu_y, \Sigma_x + \Sigma_y) \quad (6.78)$$

知道 $p(x + y)$ 是高斯矩阵,可以使用 (6.46) 到 (6.49) 的结果立即确定均值和协方差矩阵。

当我们考虑作用于随机变量的独立同分布高斯噪声时,这个属性将很重要,就像线性回归的情况一样 (第 9 章)。

例 6.7 由于期望
是线性运算,我们可以得到独立高斯随机变量的加权和

$$p(ax + by) = N(a\mu_x + b\mu_y, a^2\Sigma_x + b^2\Sigma_y) \quad (6.79)$$

评论。在第 11 章中有用的一个例子是高斯密度的加权和。这不同于高斯 ◇ 随机变量的加权和。

在定理 6.12 中,随机变量 x 来自一个密度,它是两个密度 $p_1(x)$ 和 $p_2(x)$ 的混合,由 α 加权。该定理可以推广到多元随机变量的情况,因为期望的线性也适用于多元随机变量。但是,平方随机变量的想法需要用 xx 替代。

定理 6.12。考虑两个单变量高斯密度的混合

$$p(x) = \alpha p_1(x) + (1 - \alpha)p_2(x), \quad (6.80)$$

其中标量 $0 < \alpha < 1$ 是混合权重, $p_1(x)$ 和 $p_2(x)$ 是具有不同参数的单变量高斯密度 (方程(6.62)) ,即 $(\mu_1, \sigma_1^2) = (\mu_2, \sigma_2^2)$ 。

然后混合密度 $p(x)$ 的均值由每个随机变量均值的加权和给出:

$$E[x] = \alpha\mu_1 + (1 - \alpha)\mu_2. \quad (6.81)$$

混合密度 $p(x)$ 的方差由下式给出

$$V[x] = \alpha\sigma_1^2 + (1 - \alpha)\sigma_2^2 + \alpha\mu_1^2 + (1 - \alpha)\mu_2^2 - [\alpha\mu_1 + (1 - \alpha)\mu_2]^2. \quad (6.82)$$

证明混合密度 $p(x)$ 的均值由每个随机变量均值的加权和给出。我们应用均值的定义 (定义 6.4) ,并代入我们的混合函数 (6.80),得到

$$E[x] = \int_{-\infty}^{\infty} xp(x)dx \quad (6.83a)$$

$$= \int_{-\infty}^{\infty} (\alpha p_1(x) + (1 - \alpha)p_2(x)) dx \quad (6.83b)$$

$$= \alpha \int_{-\infty}^{\infty} xp_1(x)dx + (1 - \alpha) \int_{-\infty}^{\infty} xp_2(x)dx \quad (6.83c)$$

$$= \alpha\mu_1 + (1 - \alpha)\mu_2. \quad (6.83d)$$

为了计算方差,我们可以使用 (6.44) 中方差的原始分数版本,它需要平方随机变量的期望表达式。这里我们使用随机变量函数 (平方) 期望的定义 (定义 6.3) ,

$$\text{前任 } [x^2] = \int_{-\infty}^{\infty} x^2 p(x)dx \quad (6.84a)$$

$$= \int_{-\infty}^{\infty} \alpha x^2 p_1(x) + (1 - \alpha)x^2 p_2(x) dx \quad (6.84b)$$

$$= \alpha \int_{-\infty}^{\infty} x^2 p_1(x) dx + (1 - \alpha) \int_{-\infty}^{\infty} x^2 p_2(x) dx \quad (6.84c)$$

$$= \alpha (\mu_1^{2+\sigma} + (1 - \alpha)(\mu_2^{2+\sigma})) \quad (6.84d)$$

在最后一个等式中,我们再次使用给出 σ 的方差 (6.44) 的原始分数版本。这是重新排列,使得
 $E[x^2] = E[x^2] - \mu^2$
 平方随机变量的期望是均方差和方差的和。

因此,方差由 (6.84d) 减去 (6.83d) 得出,

$$V[x] = E[x^2] - (E[x])^2 \quad (6.85a)$$

$$= \alpha(\mu_1^{2+\sigma} + (1 - \alpha)(\mu_2^{2+\sigma}) - (\alpha\mu_1 + (1 - \alpha)\mu_2)^2) \quad (6.85b)$$

$$+ \alpha\sigma^2 + (1 - \alpha)\sigma^2 - [\alpha\mu_1 + (1 - \alpha)\mu_2]^2 \quad (6.85c)$$

□

评论。前面的推导适用于任何密度,但由于高斯分布完全由均值和方差决定,混合密度可以用封闭形式确定。 ◇对于混合密度,各个成分可以被认为是条件分布(以成分身份为条件)。

方程 (6.85c) 是条件方差公式的一个例子,也称为总方差定律,它一般表示对于变量X和Y的两个总方差随机定律,它认为 $V[X] = E[Y] [V[X|Y]] + V[Y] [E[X|Y]]$, 即X的(总)方差是期望的条件方差加上条件均值的方差。

我们在示例 6.17 中考虑了一个双变量标准高斯随机变量X并对其执行了线性变换Ax。结果是均值为零且协方差为AA的高斯随机变量。观察到添加一个常数向量会改变分布的均值,而不影响其方差,也就是说,随机变量 $x + \mu$ 是具有均值 μ 和恒等协方差的高斯分布。因此,高斯随机变量的任何线性/仿射变换都是高斯分布的。

考虑一个高斯分布随机变量 $X \sim N(\mu, \Sigma)$ 。对于适当形状的给定矩阵A,设Y为随机变量,使得 $y = Ax$ 是x的变换版本。我们可以通过利用期望是线性算子 (6.50) 来计算y的均值,如下所示:

a 的任何线性/仿射变换
 高斯随机变量也是
 高斯分布。

$$E[y] = E[Ax] = AE[x] = A\mu \quad (6.86)$$

类似地,可以使用 (6.51) 找到y的方差:

$$V[y] = V[Ax] = AV[x]A^T = A\Sigma A^T \quad (6.87)$$

这意味着随机变量y服从

$$p(y) = N(y | A\mu, A\Sigma A^T) \quad (6.88)$$

现在让我们考虑逆向变换:当我们知道
随机变量的均值是另一个的线性变换

随机变量。对于给定的满秩矩阵 $A \in \mathbb{R}^{M \times N}$ 令 $y \in \mathbb{R}^M$ 为均值为 Ax 的高斯随, 其中 $M = N$,
机变量,即,

$$p(y) = N(y | \text{均值}, \Sigma). \quad (6.89)$$

对应的概率分布 $p(x)$ 是多少?如果 A 是可逆的,那么我们可以写成 $x = A^{-1}y$ 并应用上一段中的转换。然而,一般来说 A 是不可逆的,我们使用类似于伪逆 (3.57) 的方法。也就是说,我们预先然后反转 $A - A$,这是对称的,两边都乘以 A 并且是正定的,给我们关系

$$y = Ax \iff (A - A) - 1A = x. \quad (6.90)$$

因此, x 是 y 的线性变换,我们得到

$$p(x) = N(x | (A - A) - 1A, (A - A) - 1A - \Sigma(A - A) - 1) \quad (6.91)$$

6.5.4 从多元高斯分布中抽样我们不会解释计算机上随机抽样的微妙之处,有兴趣的读者可以参考 Gentle (2004)。在多变量高斯分布的情况下,这个过程包括三个阶段:首先,我们需要一个伪随机数源,在 $[0,1]$ 区间内提供均匀样本;其次,我们使用非线性变换,例如 Box-Muller 变换 (Devroye, 1986) 从单变量高斯分布中获取样本;第三,我们整理这些样本的向量,从多元标准正态 $N(0, I)$ 中获取样本

对于一般的多元高斯分布,即均值非零且协方差不是单位矩阵的情况,我们使用高斯随机变量的线性变换的性质。假设我们 $j = 1, \dots, n$, 来自多元

有兴趣生成具有均值 μ 和协方差矩阵 Σ 的样本 x_i 高斯,
分布。我们想从一个采样器构建样本,该采样器提供来自多元标准正态 $N(0, I)$ 的样本 从多元正态 $N(\mu, \Sigma)$ 中获取样本

计算
a 的
Cholesky 分解
矩阵,要求矩阵是
对称且正定 (第
3.2.3 节)。
协方差矩阵具有此属
性。

，我们可以用
高斯随机变量线性变换的性质:然后 $y = Ax + \mu$,其中 $AA^T = \Sigma$ 是高斯分布如果 $x \sim N(0, I)$, 我用均值 μ
Cholesky 分解 (第 4.3 节), 和协方差矩阵 Σ 表示。 A 的一种方便选择是使用协方差矩阵 $\Sigma = AA^T$ 的
Cholesky 分解的优点是 A 是三角形的,从而实现高效计算。

6.6 共轭和指数族

我们在统计学教科书中找到的许多“有名称”的概率分布被发现是为了模拟特定类型的现象。

例如,我们在6.5节中看到了高斯分布。这些分布还以复杂的方式相互关联(Leemis和McQueston,2008年)。对于该领域的初学者来说,弄清楚要使用哪个发行版可能会让人不知所措。此外,许多这样的分布是在过去使用铅笔和纸进行统计和计算的“计算机”时发现的。很自然地会问什么是有意义的概念作为职位描述。在计算时代(Efron和Hastie,2016年)。在上一节中,我们看到当分布为高斯分布时,可以方便地计算推理所需的许多操作。在这一点上值得回顾一下在机器学习环境中操纵概率分布的必要条件:

1. 在应用概率规则时有一些“封闭性”,例如贝叶斯定理。通过闭包,我们的意思是应用特定操作返回相同类型的对象。

2. 随着我们收集更多的数据,我们不需要更多的参数来描述分布。

3. 由于我们对从数据中学习感兴趣,所以我们想要参数es是时候表现得很好了。

事实证明,称为指数族指数族的分布类提供了适当的通用性平衡,同时保留了有利的计算和推理属性。在我们介绍指数族之前,让我们看看“命名”概率分布的另外三个成员,伯努利分布(例6.8)、二项分布(例6.9)和Beta分布(例6.10)。

示例 6.8 伯努利

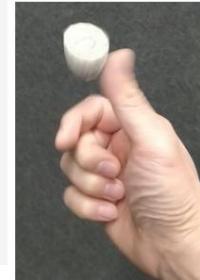
分布是状态 $x \in \{0, 1\}$ 的单个二进制随机伯努利变量X的分布。它由表示 $X = 1$ 概率的单个连续参数 $\mu \in [0, 1]$ 控制。伯努利分布Ber(μ)定义为 $p(x | \mu) = \mu^x (1 - \mu)^{1-x}$

$$p(x | \mu) = \mu^x (1 - \mu)^{1-x}, \quad x \in \{0, 1\}, \quad (6.92)$$

$$E[x] = \mu, \quad (6.93)$$

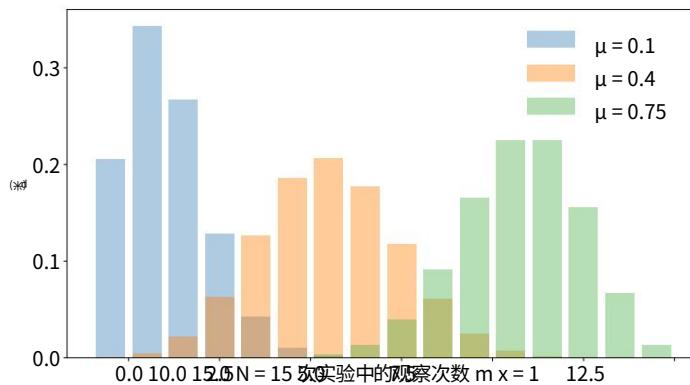
$$V[x] = \mu(1 - \mu), \quad (6.94)$$

其中 $E[x]$ 和 $V[x]$ 是二进制随机变量X的均值和方差。



可以使用伯努利分布的一个例子是当我们有兴趣对抛硬币时“正面”的概率进行建模时。

图 6.1 $\mu \in \{0.1, 0.4, 0.75\}$ 和 $N = 15$ 的二项分布示例。



评论。上面对伯努利分布的改写,我们使用布尔变量作为数值0或1,并用指数表示,这是机器学习教科书中经常使用的技巧。另一种情况是在表示多项式分布时。 ◇

二项分布

例 6.9 (二项分布)

二项分布是伯努利分布对整数分布的推广(如图 6.1 所示)。特别是,二项式可用于描述在伯努利分布的一组 N 个样本中观察到 m 次 $X = 1$ 的概率,其中 $p(X = 1) = \mu \in [0, 1]$ 。二项分布 $\text{Bin}(N, \mu)$ 定义为

$$p(m | N, \mu) = \begin{cases} \mu^m (1 - \mu)^{N-m} & m \leq N \\ 0 & m > N \end{cases}, \quad (6.95)$$

$$E[m] = N\mu, \quad (6.96)$$

$$V[m] = N\mu(1 - \mu), \quad (6.97)$$

其中 $E[m]$ 和 $V[m]$ 分别是 m 的均值和方差。

一个可以使用二项式的例子是,如果我们想描述在 N 个抛硬币实验中观察到 m 个“正面”的概率,如果在单个实验中观察到正面朝上的概率是 μ 。

贝塔分布

例 6.10 (Beta 分布)

我们可能希望在有限区间上对连续随机变量建模。

Beta 分布是连续随机变量 $\mu \in [0, 1]$ 上的分布,通常用于表示某些二元事件的概率(例如,控制伯努利分布的参数)。Beta 分布 $\text{Beta}(\alpha, \beta)$ (如图 6.2 所示)本身由两个

参数 $\alpha > 0, \beta > 0$ 且定义为

$$p(\mu | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \mu^{\alpha-1} (1 - \mu)^{\beta-1} \quad (6.98)$$

$$E[\mu] = \frac{\alpha}{\alpha + \beta}, \quad V[\mu] = \frac{\alpha\beta}{(1 + \beta)(\alpha + \beta)^2(1 + \beta + 1)} \quad (6.99)$$

其中 $\Gamma(\cdot)$ 是 Gamma 函数, 定义为

$$\Gamma(t) := \int_0^\infty x^{t-1} \exp(-x) dx, \quad t > 0. \quad (6.100)$$

$$\Gamma(t+1) = t\Gamma(t). \quad (6.101)$$

请注意 (6.98) 中 Gamma 函数的分数对 Beta 分布进行了归一化。

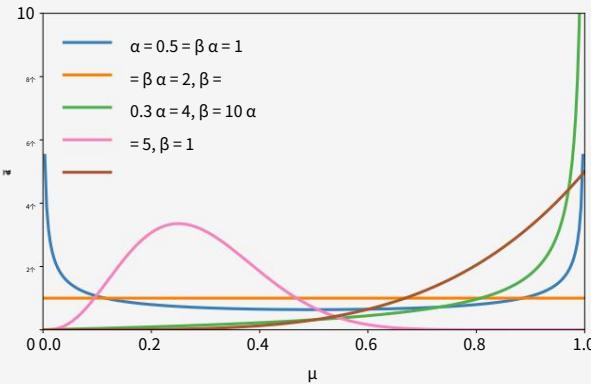


图 6.2 Beta
分布示例
 α 和 β 的不同值。

直觉上, α 将概率质量移向 1, 而 β 将概率移向能力质量接近 0。有一些特殊情况 (Murphy, 2012):

- 对于 $\alpha = 1 = \beta$, 我们得到均匀分布 $U[0, 1]$ 。
- 对于 $\alpha, \beta < 1$, 我们得到尖峰在 0 和 1 处的双峰分布。
- 对于 $\alpha, \beta > 1$, 分布是单峰的。
- 对于 $\alpha, \beta > 1$ 和 $\alpha = \beta$, 分布是单峰的、对称的, 并且以区间 $[0, 1]$ 为对称中心, 即众数/均值位于

评论。有一个完整的带有名称的分布动物园, 它们以不同的方式相互关联 (Leemis 和 McQueston, 2008)。

值得记住的是, 每个命名发行版都是出于特定原因而创建的, 但可能有其他应用程序。了解创建特定发行版背后的原因通常可以深入了解如何最好地使用它。我们介绍了前面的三个分布, 以便能够说明共轭 (第 6.6.1 节) 和指数族 (第 6.6.3 节) 的概念。 ◇

6.6.1 共轭

根据贝叶斯定理 (6.23), 后验与先验和似然的乘积成正比。由于两个原因, 先验的规范可能很棘手: 首先, 先验应该在我们看到任何数据之前封装我们关于问题的知识。这通常很难描述。其次, 通常不可能分析地计算后验分布。然而, 有一些先验在计算上很方便: 共轭先验。

先验共轭

共轭

定义 6.13 (共轭先验)。如果后验与先验具有相同的形式/类型, 则先验对于似然函数是共轭的。

共轭特别方便, 因为我们可以更新先验分布的参数来代数计算后验分布。

评论。在考虑概率分布的几何时, 共轭先验保留与似然相同的距离结构 (Agarwal 和 Daum 'e III, 2010)。为了介绍共轭先验的具体示例, 我们在示例 6.11 中描述了二项分布 (定义在离散随机变量上) 和 Beta 分布 (定义在连续随机变量上)。

示例 6.11 (Beta-二项式共轭)

考虑一个二项式随机变量 $x \sim \text{Bin}(N, \mu)$ 其中

$$p(x | N, \mu) = \sum_{x=0}^N \mu^x (1 - \mu)^{N-x}, \quad x = 0, 1, \dots, N, \quad (6.102)$$

是在 N 次掷硬币中找到 x 次结果“正面”的概率, 其中 μ 是“正面”的概率。我们在参数 μ 上放置一个 Beta 先验, 即 $\mu \sim \text{Beta}(\alpha, \beta)$, 其中 $\Gamma(\alpha + \beta)$

$$p(\mu | \alpha, \beta) = \frac{1}{\Gamma(\alpha)\Gamma(\beta)} \mu^{\alpha-1} (1 - \mu)^{\beta-1}. \quad (6.103)$$

如果我们现在观察到一些结果 $x = h$, 也就是说, 我们在 N 次抛硬币中看到 h 次正面朝上, 我们计算 μ 上的后验分布为

$$p(\mu | x = h, N, \alpha, \beta) \propto p(x | N, \mu) p(\mu | \alpha, \beta) (1 - \mu)^{\beta-1}. \quad (6.104a)$$

$$\propto \mu^{h-\alpha-1} (1 - \mu)^{N-h+\beta-1} \quad (6.104b)$$

$$= \mu^{h+\alpha-1} (1 - \mu)^{(N-h)+\beta-1} \quad (6.104c)$$

$$\propto \text{Beta}(h + \alpha, N - h + \beta), \quad (6.104d)$$

即, 后验分布是 Beta 分布, 因为先验分布, 即

可能性	先验共轭	后部
伯努利	测试版	测试版
二项式	测试版	测试版
高斯	高斯/逆伽马	高斯/逆伽马
高斯	高斯/逆 Wishart	高斯/逆 Wishart
多项狄利克雷		狄利克雷

表 6.2 常见似然函数的
共轭先验示例。

Beta 先验与二项式似然函数中的参数 μ 共轭。

在下面的示例中,我们将导出类似于 Beta-Binomial 共轭结果的结果。这里我们将证明 Beta 分布是伯努利分布的共轭先验。

示例 6.12 (Beta-Bernoulli 共轭)

令 $x \in \{0, 1\}$ 服从参数为 $\theta \in [0, 1]$ 的伯努利分布, 即 $p(x = 1 | \theta) = \theta$ 。这也可以表示为 $p(x | \theta) = \theta^x (1 - \theta)^{1-x}$ 。令 θ 服从 Beta 分布

$$\theta^{\alpha-1} (1 - \theta)^{\beta-1}$$

将 Beta 分布和伯努利分布相乘, 我们得到

$$p(\theta | x, \alpha, \beta) = p(x | \theta)p(\theta | \alpha, \beta) \quad (6.105a)$$

$$\propto \theta^x (1 - \theta)^{1-x} \theta^{\alpha-1} (1 - \theta)^{\beta-1} \quad (6.105b)$$

$$\propto \theta^{\alpha+x-1} (1 - \theta)^{\beta+(1-x)} \quad (6.105c)$$

$$\propto p(\theta | \alpha + x, \beta + (1 - x)) \quad (6.105d)$$

最后一行是参数为 $(\alpha + x, \beta + (1 - x))$ 的 Beta 分布。

表 6.2 列出了一些参数的共轭先验示例

概率建模中使用的标准似然。诸如 The Gamma prior is Multinomial, inverse Gamma, inverse Wishart 和 Dirichlet 等分布可以在任何统计文本中找到, 例如, 在 Bishop (2006) 中有描述。

单变量高斯中精度
(逆方差)的共轭

Beta 分布是二项式和伯努利似然中参数 μ 的共轭先验。对于高斯似然函数, 我们可以在均值上放置一个共轭高斯先验。高斯似然在表中出现两次的原因是我们需要区分单变量和多变量的情况。在单变量 (标量) 情况下, 逆 Gamma 是方差的共轭先验。

似然,Wishart 先验
与多元高斯似然
中的精度矩阵 (逆
协方差矩阵) 共轭。

在多变量情况下, 我们使用共轭逆 Wishart 分布作为协方差矩阵的先验。Dirichlet 分布是多项式似然函数的先验共轭。有关详细信息, 请参阅 Bishop (2006)。

6.6.2 充分统计

回想一下,随机变量的统计量是该随机变量的确定性函数。例如,如果 $x = [x_1, \dots, x_N]$ 是单变量高斯随机变量向量,即 $x_n \sim N(\mu, \sigma^2)$ ($x_1 + \dots + x_N$)是一个统计量。Ronald Fisher 爵士 dis sample mean $\bar{x} = \frac{1}{N} \sum x_i$ =涵盖了充分统计量,然后从与正在考虑的分布 $p(x|\theta)$ 的概念:存在包含所有可用信息的统计信息,这些信息可以从相对应的数据中推断出来。换句话说,充分统计量包含了对总体进行推断所需的所有信息,也就是说,它们是足以表示分布的统计量。

足够的统计数据

对于由 θ 参数化的一组分布,假设 X 是一个随机变量,其分布为 $p(x|\theta_0)$,给定一个未知的 θ_0 。如果统计向量 $T(x)$ 包含关于 θ_0 的所有可能信息,则称为 θ_0 的充分统计量。更正式地说“包含所有可能的信息”,这意味着给定 θ 的 x 的概率可以分解为不依赖于 θ 的部分和仅通过 $T(x)$ 依赖于 θ 的部分。

Fisher-Neyman 分解定理形式化了这个概念,我们在定理 6.14 中陈述了这个概念,但没有证明。

定理 6.14 (Fisher-Neyman)。 [Lehmann 和 Casella (1998) 中的定理 6.5] 令 X 具有概率密度函数 $p(x|\theta)$ 。那么当且仅当 $p(x|\theta)$ 可以写成以下形式时,统计数据 $T(x)$ 对于 θ 是充分的

$$p(x|\theta) = h(x)g(\theta)(T(x)), \quad (6.106)$$

其中 $h(x)$ 是独立于 θ 的分布, $g(\theta)$ 通过充分的统计量 $T(x)$ 捕获对 θ 的所有依赖性。

如果 $p(x|\theta)$ 不依赖于 θ ,则 $T(x)$ 对于任何函数 g 来说都是一个充分的统计量。更有趣的情况是 $p(x|\theta)$ 仅依赖于 $T(x)$ 而不是 x 本身。在这种情况下, $T(x)$ 是 θ 的充分统计量。

在机器学习中,我们考虑来自分布的有限数量的样本。可以想象,对于简单分布(例如示例 6.8 中的伯努利分布),我们只需要少量样本即可估计分布的参数。我们还可以考虑相反的问题:如果我们有一组数据(来自未知分布的样本),哪个分布最合适?一个自然而然的问题是,随着我们观察到更多的数据,我们是否需要更多的参数 θ 来描述分布?事实证明,一般来说答案是肯定的,这在非参数统计中得到了研究(Wasserman, 2007)。一个逆向的问题是考虑哪一类分布具有有限维的充分统计量,即描述它们所需的参数个数不会任意增加。答案是指数族分布,如下一节所述。

6.6.3 指数族

在考虑（离散或连续随机变量的）分布时，我们可以有三种可能的抽象层次。在第一级（光谱最具体的一端），我们有一个具有固定参数的特定命名分布，例如具有零均值和单位方差的单变量高斯 $N(0, 1)$ 。在机器学习中，我们经常使用第二层抽象，即我们固定参数形式（单变量高斯）并从数据中推断参数。例如，我们假设一个单变量高斯分布 $N(\mu, \sigma^2)$ ， σ^2 具有未知均值 μ 和未知方差 σ^2 并使用最大似然拟合来确定最佳参数 (μ, σ^2) 。我们将在第 9 章考虑线性回归时看到一个这样的例子。第三个抽象层次是考虑分布族，在本书中，我们考虑指数族。单变量高斯是指数族成员的一个例子。许多广泛使用的统计模型，包括表 6.2 中所有“命名”模型，都是指数家族的成员。它们都可以统一为一个概念（Brown, 1986）。

评论。一个简短的历史轶事：与数学和科学中的许多概念一样，指数族是由不同的研究人员同时独立发现的。1935-1936 年，塔斯马尼亚的埃德温·皮特曼、巴黎的乔治·达莫瓦和纽约的伯纳德·库普曼独立表明，指数族是唯一在重复 ◇ 独立抽样下享有有限维充分统计的族（Lehmann 和 Casella，1998）。

指数族是概率分布族，由 $\theta \in \mathbb{R}^d$ 的参数化指数族，形式为

$$p(x | \theta) = h(x) \exp(-\theta^\top \phi(x) - A(\theta)), \quad (6.107)$$

其中 $\phi(x)$ 是充分统计量的向量。一般来说，任何内积（第 3.2 节）都可以用在 (6.107) 中，为了具体起见，我们将在此处使用标准点积 $(\theta, \phi(x)) = \theta^\top \phi(x)$ 。请注意，指数族的形式本质上是 Fisher-Neyman 定理（定理 6.14）中 $g(\theta)(\phi(x))$ 的特定表达式。

通过向充分统计向量 $\phi(x)$ 添加另一个条目 $(\log h(x))$ 并约束相应的参数 $\theta_0 = 1$ ，可以将因子 $h(x)$ 吸收到点积项中。项 $A(\theta)$ 是确保分布总和或积分为一的归一化常数，称为对数分区函数。通过忽略这两项并将指数族视为以下形式的分布，可以获得指数族的良好直观无对数划分

功能

$$p(x | \theta) \propto \exp \theta^\top \phi(x). \quad (6.108)$$

对于这种形式的参数化，参数 θ 称为自然参数

参数。乍一看,指数族似乎是将指数函数添加到点积结果的普通变换。然而,基于我们可以在 (x) 中捕获有关数据的信息这一事实,存在许多允许方便建模和高效计算的含义。

例 6.13 (作为指数族的高斯函数)

考虑单变量高斯分布 $N(\mu, \sigma^2)$ 。让 $\phi(x) =$

$x_{2,x}$

然后通过使用指数族的定义,

$$p(x | \theta) \propto \exp(\theta_1 x + \theta_2 x^2). \quad (6.109)$$

环境

$$\theta = \frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2} \quad (6.110)$$

代入 (6.109), 我们得到

$$p(x | \theta) \propto \exp \left(\frac{\mu x}{\sigma^2} - \frac{x^2}{2\sigma^2} \right) \propto \exp \left(-2\sigma \frac{1}{2} (x - \mu)^2 \right). \quad (6.111)$$

因此,单变量高斯分布是世博会的成员

具有足够统计量的基本家庭 $(x) =$

$x_{2,x}$ 和自然参数

由 (6.110) 中的 θ 给出。

示例 6.14 (伯努利作为指数族)

回忆一下例 6.8 中的伯努利分布

$$p(x | \mu) = \mu^x (1 - \mu)^{1-x}, \quad x \in \{0, 1\}. \quad (6.112)$$

这可以写成指数族形式

$$p(x | \mu) = \exp[\log \mu x + (1 - \mu)^{1-x}] = \exp[x \log \mu + (1 - x) \log(1 - \mu)] \quad (6.113a)$$

$$= \exp[x \log \mu + (1 - x) \log(1 - \mu)]. \quad (6.113b)$$

$$= \exp[x \log \mu + (1 - x) \log(1 - \mu)]. \quad (6.113c)$$

$$= \exp[x \log \mu + (1 - x) \log(1 - \mu)]. \quad (6.113d)$$

最后一行 (6.113d) 可以通过观察确定为指数族形式 (6.107)

$$h(x) = 1 \quad (6.114)$$

$$\theta = \log \mu \quad (6.115)$$

$$\phi(x) = x \quad (6.116)$$

$$A(\theta) = -\log(1 - \mu) = \log(1 + \exp(\theta)). \quad (6.117)$$

θ 和 μ 之间的关系是可逆的,因此

$$\mu = \frac{1}{1 + \exp(-\theta)}. \quad (6.118)$$

关系 (6.118) 用于获得 (6.117) 的右等式。

评论。原始伯努利参数 μ 与自然参数 θ 之间的关系称为 sigmoid 函数或逻辑函数。Ob-sigmoid 满足 $\mu \in (0, 1)$ 但 $\theta \in \mathbb{R}$, 因此 sigmoid 函数将实数值压缩到范围 $(0, 1)$ 中。此属性在机器学习中很有用,例如它用于逻辑回归 (Bishop, 2006, 第 4.3.2 节), 以及神经网络 ◇ 作品中的非线性激活函数 (Goodfellow 等人, 2016, 章节 6)。

如何找到特定分布 (例如, 表 6.2 中的那些) 的共轭分布的参数形式通常并不明显。

指数族提供了一种查找共轭分布对的便捷方法。考虑随机变量 X 是指数族 (6.107) 的成员:

$$p(x | \theta) = h(x) \exp(-\theta - A(\theta)). \quad (6.119)$$

指数族的每个成员都有一个共轭先验 (Brown, 1986)

$$p(\theta | \gamma) = h_c(\theta) \exp\left(\frac{\gamma_1}{\gamma_2} \theta - A_c(\gamma)\right), \quad (6.120)$$

其中 $\gamma = \frac{\gamma_1}{\gamma_2}$ 维度为 $\dim(\theta) + 1$ 。

共轭先验是 $\theta - A(\theta)$ 。通过使用一般知识

指数族的共轭先验形式,我们可以导出对应于特定分布的共轭先验的函数形式。

示例 6.15 回顾伯努利分布的指数族形式 (6.113d)

$$p(x | \mu) = \exp x \log \mu + \log(1 - \mu). \quad (6.121)$$

规范共轭先验具有以下形式

$$+ \alpha) \log(1 - \mu) - \frac{\mu \mu p(\mu | \alpha, \beta)}{A_c(\gamma)}, \exp \alpha \log \frac{1 - \mu}{\mu} \quad (6.122)$$

其中我们定义了 $\gamma := [\alpha, \beta + \alpha]$ 和 $hc(\mu) := \mu/(1 - \mu)$ 。平等
(6.122) 然后简化为

$$p(\mu | \alpha, \beta) = \exp [(\alpha - 1) \log \mu + (\beta - 1) \log(1 - \mu) - Ac(\alpha, \beta)] . \quad (6.123)$$

将其置于非指数家庭形式中

$$p(\mu | \alpha, \beta) \propto \mu^{\alpha-1} (1 - \mu)^{\beta-1} , \quad (6.124)$$

我们将其确定为 Beta 分布 (6.98)。在示例 6.12 中, 我们假设 Beta 分布是伯努利分布的共轭先验, 并证明它确实是共轭先验。在这个例子中, 我们通过查看指数族形式的伯努利分布的典型共轭先验来推导出 Beta 分布的形式。

如前一节所述, 指数族的主要动机是它们具有有限维的充分统计量。

此外, 共轭分布很容易记下来, 共轭分布也来自指数族。从推理的角度来看, 最大似然估计表现得很好, 因为充分统计的经验估计是充分统计的总体值的最优估计 (回想一下高斯的均值和协方差)。从优化的角度来看, 对数似然函数是凹的, 允许应用有效的优化方法 (第 7 章)。

6.7 变量的变化/逆变换看起来似乎有很多已知的分布,

但实际上我们有名字的分布集是非常有限的。因此, 了解转换后的随机变量是如何分布的通常很有用。例如, 假设 X 是一个随机变量, 服从单变量正态分布 $N(0, 1)$ 。 X_2 的分布是什么? 另一个在机器学习中很常见的例子是, 假设 X_1 和 X_2 是单变量标准正态分布, $(X_1 + X_2)$? 计算分布的一个选项是什么

如我们在 6.4.4 节中看到的, 当我们考虑随机变量的仿射变换时, 我们可以计算结果随机变量的均值和方差。但是, 我们可能无法获得变换下分布的函数形式。此外, 我们可能对具有封闭形式的随机变量的非线性变换感兴趣

表达式不是现成的。

备注（符号）。在本节中，我们将明确说明随机变量及其取值。因此，回想一下，我们使用大写字母X、Y表示随机变量，使用小写字母x、y表示随机变量在目标空间T中的取值。我们将显式地将离散随机变量X的 pmfs 写为 $P(X = x)$ 。对于连续随机变量X（第 6.2.2 节），pdf 写为 $f(x)$ ，cdf 写为 $F_X(x)$ 。 ◇ 我们将研究两种获取随机变量变换分布的方法：使用累积分布函数定义的直接方法和使用微积分链式法则的变量变化方法（第 5.2.2 节）。变量变化 app-Moment 生成方法被广泛使用，因为它提供了一个“配方”，用于尝试计算由于转换而产生的结果分布。我们将解释单变量随机变量的技术，并仅简要提供多变量随机变量一般情况下的结果。

函数也可以用来研究
随机变换

离散随机变量的变换可以理解为 di

正确地。假设有一个离散随机变量X，其 pmf $P(X = x)$ （第 6.2.1 节）和一个可逆函数 $U(x)$ 。考虑变换后的随机变量 $Y := U(X)$ ，其中 pmf $P(Y = y)$ 。然后

变量（Casella 和
Berger,2002 年,第 2
章）。

$$\begin{aligned} P(Y = y) &= P(U(X) = y) && \text{兴趣转换(6.125a)} \\ &= P(X = U^{-1}(y)) && \text{逆转换(6.125b)} \end{aligned}$$

我们可以观察到 $x = U^{-1}(y)$ 。因此，对于离散随机变量，变换直接改变单个事件（概率适当变换）。

6.7.1 分布函数技术分布函数技术可以追溯到第一原理，

并使用 $cdf F_X(x) = P(X \leq x)$ 的定义及其微分是 pdf $f(x)$ （Wasserman, 2004 年, 第 2 章）。对于随机变量X和函数U，我们找到随机变量 $Y := U(X)$ 的 pdf

1. 找到 cdf：

$$F_Y(y) = P(Y \leq y) \quad (6.126)$$

2. 对 cdf $F_Y(y)$ 求微分得到 pdf $f(y)$ 。

$$df(y) = F'_Y(y) dy \quad (6.127)$$

我们还需要记住，由于U的变换，随机变量的域可能已经改变。

例 6.16 设 X 为概率

密度函数为 $0 < x < 1$ 的连续随机变量

$$f(x) = 3x \quad . \quad (6.128)$$

我们有兴趣找到 $Y = X^2$ 的 pdf

函数 f 是 x 的增函数, 因此得到
 y 的值在 $[0, 1]$ 区间内。我们获得

$$F_Y(y) = P(Y \leq y) \quad \text{cdf 的定义(6.129a)} \quad (6.129b)$$

$= P(X^2 \leq y)$ 兴趣的转变

$$= P(X \leq \sqrt{y}) \quad \text{逆} \quad (6.129c)$$

$= \text{外汇}(y^{\frac{1}{2}})$ cdf 的定义(6.129d)

$$= \int_0^{y^{\frac{1}{2}}} 3t \, dt \quad \text{作为定积分的 cdf (6.129e)}$$

$$= \int_{t=0}^{y^{\frac{1}{2}}} 3t \, dt = y^{\frac{3}{2}} \quad \text{整合的结果} \quad (6.129f)$$

$$= y^{\frac{3}{2}}, \quad 0 \leq y \leq 1. \quad (6.129 \text{ 克})$$

因此, Y 的 cdf 是

$$F_Y(y) = y^{\frac{3}{2}} \quad (6.130)$$

对于 $0 < y < 1$ 。为了获得 pdf, 我们对 cdf 进行微分

$$f_Y(y) = \frac{d}{dy} F_Y(y) = \frac{3}{2} y^{\frac{1}{2}} \quad (6.131)$$

对于 $0 < y < 1$ 。

在例 6.16 中, 我们考虑了一个严格单调递增的函数。这意味着我们可以计算反函数。

tion $f(x) = 3x$ 一般来

具有的功能
 逆函数称为双射函数 (第
 2.7 节)。

说, 我们要求感兴趣的函数 $y = U(x)$ 有一个 $\text{in}(y)$ 。通过考虑随机变量 X 的累积分布函数 $F_X(x)$, 并将诗节 $x = U^{-1}(y)$ 其用作变换 $U(x)$, 可以获得有用的结果。这导致以下定理。

定理 6.15。[Casella 和 Berger (2002) 中的定理 2.1.10] 令 X 为具有严格单调累积分布函数 $F_X(x)$ 的连续随机变量。那么随机变量 Y 定义为

$$Y := \text{外汇}(X) \quad (6.132)$$

具有均匀分布。

定理 6.15 被称为概率积分变换, 它用于推导通过变换从分布中抽样的算法

从均匀随机变量中抽样的结果 (Bishop, 2006)。

该算法的工作原理是首先从均匀分布中生成样本,然后通过逆 cdf (假设这是可用的)对其进行转换,以获得所需分布中的样本。概率积分变换也用于假设检验样本是否来自特定分布 (Lehmann 和 Romano,2005) 。 cdf 的输出给出均匀分布的想法也构成了 copulas 的基础 (Nelsen, 2006)。

6.7.2 变量的变化第 6.7.1 节中的分

布函数技术源自第一原理,基于 cdfs 的定义并使用逆、微分和积分的特性。这一来自第一性原理的论证依赖于两个事实:

1. 我们可以将Y的 cdf 转换为X的 cdf 表达式。
2. 我们可以对cdf进行微分得到pdf。

让我们逐步分解推理,目的是理解定理 6.16 中更一般的变量变化方法。

评论。“变量变换”这个名字来源于在遇到困难的积分时改变积分变量的想法。对于单变量函数,我们使用积分的替换规则,

概率变量的变化依赖于
微积分中的变量变化方
法
(Tandra,2014)。

$$f(g(x))g'(x)dx = f(u)du, \quad \text{其中 } u = g(x). \quad (6.133)$$

该规则的推导是基于微积分的链式法则 (5.32)并通过应用微积分基本定理的两倍。微积分的基本定理形式化了积分和微分在某种程度上是彼此“逆”的事实。通过(松散地)考虑方程 $u = g(x)$ 的微小变化(微分),即通过将 $\Delta u = g'(x)\Delta x$ 视为u的微分,可以获得对规则的直观理解=克(X)。通过代入 $u = g(x)$, (6.133) 右侧积分内的自变量变为 $f(g(x))$ 。通过假定 du 项可以近似为 $du \approx \Delta u = g'(x)\Delta x$,并且 $dx \approx \Delta x$,我们得到 (6.133)。 ◇考虑一个单变量随机变量X 和一个可逆函数U,它给我们另一个随机变量Y = U(X)。我们假设随机变量X具有状态 $x \in [a, b]$ 。根据 cdf 的定义,我们有

$$F_Y(y) = P(Y \leq y). \quad (6.134)$$

我们对随机变量的函数U感兴趣

$$P(Y \leq y) = P(U(X) \leq y), \quad (6.135)$$

其中我们假设函数U是可逆的。区间上的可逆函数要么严格递增,要么严格递减。在U严格递增的情况下,其逆 U^{-1} 也严格递增。对于 $P(U(X) \leq y)$ 的参数,我们通过应用逆 U^{-1}

$$P(U(X) \leq y) = P(U^{-1}(U(X)) \leq U^{-1}(y)) = P(X \leq U^{-1}(y)). \quad (6.136)$$

(6.136) 中最右边的项是X的cdf的表达式。回想一下 cdf 在 pdf 方面的定义

$$P(X \leq U^{-1}(y)) = \int_{-\infty}^{U^{-1}(y)} f(x) dx. \quad (6.137)$$

现在我们用x表示Y的cdf:

$$F(y) = \int_{-\infty}^{U^{-1}(y)} f(x) dx. \quad (6.138)$$

为了获得pdf,我们对(6.138)关于y进行微分:

$$f(y) = F'(y) = \frac{d}{dy} \int_{-\infty}^{U^{-1}(y)} f(x) dx. \quad (6.139)$$

是关于x的,但我们需要关于y的积分,因为我们是关于y微分的。特别地,我们使用(6.133)来得到替换

$$f(U^{-1}(y))U'^{-1}(y)dy = f(x)dx \text{ 其中 } x = U^{-1}(y). \quad (6.140)$$

在(6.139)的右侧使用(6.140)可以得到

$$f(y) = \frac{d}{dy} \int_{-\infty}^{U^{-1}(y)} f(x) dx = f(U^{-1}(y))U'^{-1}(y). \quad (6.141)$$

然后我们回想起微分是一个线性算子,我们使用(y)是x的函数而不是下标x提醒我们自己 $f_x(U^{-1}(y))$ 。再次调用微积分基本定理给我们

$$f(y) = f(U^{-1}(y)) \cdot \frac{d}{dy} U^{-1}(y). \quad (6.142)$$

回想一下,我们假设U是严格递增的函数。对于递减函数,当我们遵循相同的推导时,结果证明我们有一个负号。我们引入微分的绝对值,使U的增加和减少都具有相同的表达式:

$$f(y) = f(U^{-1}(y)) \cdot \frac{d}{dy} U^{-1}(y). \quad (6.143)$$

评论。与 (6.125b) 中的离散情况相比,我们有一个额外的因素

$\frac{d}{dy} U^{-1}(y)$ 。连续情况需要更加小心,因为对于所有 y , $P(Y = y) = 0$ 。概率密度函数 $f(y)$ 没有描述为涉及 y 的事件的概率。到目前为止,在本节中,我们一直在研究变量的单变量变化。多元随机变量的情况类似,但由于绝对值不能用于多元函数而变得复杂。相反,我们使用雅可比矩阵的行列式。回想一下 (5.58),雅可比矩阵是偏导数矩阵,非零行列式的存在表明我们可以反转雅可比矩阵。回想一下 4.1 节中的讨论,行列式的出现是因为我们的微分(体积的立方体)被雅可比行列式变换为平行六面体。让我们在下面的定理中总结之前的讨论,它为我们提供了变量的多元变化的方法。

定理 6.16。[Billingsley (1995) 中的定理 17.2] 令 $f(x)$ 为多元连续随机变量 X 的概率密度值。

如果向量值函数 $y = U(x)$ 对于 x 域内的所有值都是可微且可逆的,那么对于 y 的相应值, $Y = U(X)$ 的概率密度由下式给出

$$f(y) = f_x(U^{-1}(y)) \cdot \det \frac{\partial U^{-1}}{\partial y}(y) \quad (6.144)$$

这个定理乍一看很吓人,但关键是多元随机变量的变量变化遵循单变量变量变化的过程。首先我们需要计算逆变换,并将其代入 x 的密度。然后我们计算雅可比矩阵的行列式并将结果相乘。以下示例说明了双变量随机变量的情况。

例 6.17

考虑一个双变量随机变量 X ,其状态为 $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ 和概率

能力密度函数

$$F \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \frac{1}{2\pi} \exp \left(-\frac{x_1^2 + x_2^2}{2} \right) \quad (6.145)$$

我们使用定理 6.16 中的变量变化技术来推导随机变量的线性变换(第 2.7 节)的效果。

考虑矩阵 $A \in \mathbb{R}^{2 \times 2}$ 定义为

$$\text{一个} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \quad (6.146)$$

我们有兴趣找到状态为 $y = Ax$ 的转换后的双变量随机变量Y的概率密度函数。

回想一下,对于变量的变化,我们需要将x作为y的函数进行逆变换。由于我们考虑线性变换,逆变换由矩阵求逆给出(参见第2.2.2节)。

对于 2×2 矩阵,我们可以明确地写出公式,由下式给出

$$\begin{matrix} x_1 \\ x_2 \end{matrix} = \begin{matrix} -1 & y_1 \\ & y_2 \end{matrix} = \frac{1}{\det(A)} \begin{matrix} d-b \\ -c \end{matrix} \begin{matrix} y_1 \\ y_2 \end{matrix}. \quad (6.147)$$

观察到 $\det(A) - bc$ 是A的行列式(第4.1节)。相应的概率密度函数由下式给出

$$f(x) = f(A^{-1}y) = \frac{1}{2\pi} \exp(-\frac{1}{2}y^T A^{-1} y). \quad (6.148)$$

矩阵乘以向量相对于向量的偏导数就是矩阵本身(第5.5节),因此

$$\frac{\partial}{\partial y} \det(A^{-1}) = -\frac{1}{2\pi} y^T A^{-1} y. \quad (6.149)$$

回忆一下4.1节,逆行列式是行列式的倒数,因此雅可比矩阵的行列式是

$$\text{检测 } \frac{\partial}{\partial y} \det(A^{-1}) = \frac{1}{\det(A)}. \quad (6.150)$$

我们现在可以通过将(6.148)乘以(6.150)来应用定理6.16中的变量变化公式,得到

$$f(y) = f(x) \det \frac{\partial}{\partial y} \det(A^{-1}) \quad (6.151a)$$

$$= \frac{1}{2\pi} \exp(-\frac{1}{2}y^T A^{-1} y) | \det(A) |^{-1}. \quad (6.151b)$$

虽然示例6.17基于双变量随机变量,这使我们能够轻松计算矩阵逆,但前面的关系适用于更高的维度。

评论。我们在6.5节中看到,(6.148)中的密度 $f(x)$ 实际上是标准高斯分布,变换后的密度 $f(y)$ 是协方差为 $\Sigma = AA^T$ 的二元高斯分布

我们将使用本章中的思想在8.4节中描述概率建模,并在8.5节中介绍图形语言。我们将在第9章和第11章中看到这些想法的直接机器学习应用。

6.8 进一步阅读本章有时相

当简洁。 Grinstead 和 Snell (1997) 以及 Walpole 等人。 (2011) 提供更轻松的演示文稿,适合自学。对概率的更多哲学方面感兴趣的读者应该考虑 Hacking (2001),而 Downey (2014) 提出了一种与软件工程更相关的方法。

可以在 Barndorff-Nielsen (2014) 中找到指数族的概述。我们将在第 8 章中看到更多关于如何使用概率分布对机器学习任务建模的信息。具有讽刺意味的是,最近对神经网络的兴趣激增导致了对概率模型的更广泛理解。例如,标准化流量的想法 (Jimenez Rezende 和 Mohamed,2015 年) 依赖于变量的变化来转换随机变量。 Goodfellow 等人在该书的第 16 至 20 章中描述了应用于神经网络的变分推理方法的概述。 (2016)。

我们通过避免测量理论问题 (Billingsley, 1995; Pollard, 2002),并通过假设我们有实数,以及在实数上定义集合的方法,避免了连续随机变量的大部分困难作为它们适当的出现频率。这些细节确实很重要,例如,在连续随机变量 x, y 的条件概率 $p(y | x)$ 的指定中 (Proschan 和 Presnell,1998)。惰性符号隐藏了我们想要指定 $X = x$ (这是一组测度零) 的事实。此外,我们感兴趣的是 y 的概率密度函数。更精确的表示法必须是 $E[y| \sigma(x)]$,其中我们对以 x 的 σ -代数为条件的测试函数 f 的 y 进行期望。对概率论的细节感兴趣的更技术性的观众有很多选择 (Jaynes,2003 年;MacKay,2003 年;Jacod 和 Protter,2004 年;Grimmett 和 Welsh,2014 年),包括一些非常技术性的讨论 (Shiryayev,1984 年;Lehmann 和 Casella,1998 年;Dudley,2002 年;Bickel 和 Doksum,2006 年;Cinlar,2011 年)。处理概率的另一种方法是从期望的概念开始,然后“逆向工作”以推导出概率空间的必要属性 (Whittle,2000)。由于机器学习允许我们对越来越复杂的数据类型进行更复杂的分布建模,因此概率机器学习模型的开发人员必须了解这些更多的技术方面。以概率建模为重点的机器学习文本包括 MacKay (2003) 的书籍;主教 (2006);拉斯穆森和威廉姆斯 (2006);理发师 (2012);墨菲 (2012)。

练习

6.1 考虑以下两个离散随机变量 X 和 Y 的双变量分布 $p(x, y)$

		y1	0.01	0.02	0.03	0.1	0.1		
是	y2		0.05	0.1	0.05	0.07	0.2		
	y3		0.1	0.05	0.03	0.05	0.04		

x1 x2 x3 x4 x5
X

计算：

A. 边际分布 $p(x)$ 和 $p(y)$ 。 b. 条件分布 $p(x|y=y_1)$ 和 $p(y|x=x_3)$ 。

6.2 考虑两种高斯分布的混合（如图 6.2 所示），

$$\begin{array}{ccccccccc} & 10 & & 1 & 0 & 0 & & 0 & & 8.4 & 2.0 & 2.0 \\ \begin{matrix} 0.4 \text{牛顿} \\ \text{z} \end{matrix} & , & 1 & & & & + 0.6 \text{牛} & & 0 & , & 1.7 \end{array}$$

A. 计算每个维度的边际分布。 b. 计算每个边际分布的均值、众数和中位数。 C.
计算二维分布的均值和众数。

6.3 你编写了一个有时编译有时不编译的计算机程序（代码不变）。您决定使用带参数 μ 的伯努利分布对编译器的表观随机性（成功与不成功） x 建模：

$$p(x|\mu) = \mu^x (1-\mu)^{1-x}, \quad x \in \{0, 1\}.$$

为伯努利似然选择一个共轭先验并计算后验分布 $p(\mu | x_1, \dots, x_N)$ 。

6.4 有两个袋子。第一个袋子里有四个芒果和两个苹果；这

第二个袋子里有四个芒果和四个苹果。

我们还有一枚有偏差的硬币，它显示“正面”的概率为 0.6，“反面”的概率为 0.4。如果硬币显示“正面”。我们从袋子 1 中随机挑选一个水果；否则我们从袋子 2 中随机挑选一个水果。

你的朋友抛硬币（你看不到结果），从相应的袋子里随机挑选一个水果，送给你一个芒果。

从袋子 2 中摘下芒果的概率是多少？

提示：使用贝叶斯定理。

6.5 考虑时间序列模型

$$\begin{aligned} x_{t+1} &= A x_t + w, \quad w \sim N(0, Q) \quad v \sim N \\ &+ v, \quad 0, R, \end{aligned}$$

其中 w, v 是 iid 高斯噪声变量。此外，假设 $p(x_0) = N(\mu_0, \Sigma_0)$ 。A. $p(x_0, x_1, \dots, x_T)$ 的形式是什么？证明你的答案（你不必明

确计算联合分布）。b. 假设 $p(x_t | y_1, \dots, y_t) = N(\mu_t, \Sigma_t)$ 。

1. 计算 $p(x_{t+1} | y_1, \dots, y_t)$ 。

2. 计算 $p(x_{t+1}, y_{t+1} | y_1, \dots, y_t)$ 。
 3. 在时间 $t+1$, 我们观察到值 $y_{t+1} = y^*$ 。计算条件分布 $p(x_{t+1} | y_1, \dots, y_{t+1})$ 。

6.6 证明 (6.44) 中的关系, 它将方差的标准定义与方差的原始分数表达式联系起来。

6.7 证明 (6.45) 中的关系, 该关系将数据集中示例之间的成对差异与方差的原始分数表达式相关联。

6.8 用指数族的自然参数形式表示伯努利分布, 见 (6.107)。

6.9 将二项式分布表示为指数族分布。也表示 Beta 分布是一个指数族分布。证明 Beta 和二项分布的乘积也是指数族的成员。

6.10 推导 6.5.2 节中的关系有两种方式:

A. 通过完成方块 b. 通过以指数族形式表示高斯函数两个高斯函数 $N(x | \mu, \Sigma)$ 的乘积一个, 一个 $N(x | \mu, \Sigma)$, Σ 是一个非归一化的高斯分布 $c N(x | \mu, \Sigma)$ 与

$$\begin{aligned} C &= (\text{一个 } -\frac{1}{2} \text{ } \Sigma^{-1})^{-1} \\) c &= C(\mu - \frac{1}{2} \Sigma^{-1} \mu) \\ c &= (2\pi)^{-\frac{D}{2}} |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2} \text{tr}(\Sigma^{-1}(\mu - \mu)^T)} \end{aligned}$$

请注意, 归一化常数 c 本身可以被认为是 (归一化的)
 a 或 b 中的高斯分布具有 “膨胀的” 协方差矩阵 $\Sigma + \Sigma$, 即 $c = N(a | \mu, \Sigma + \Sigma)$, $\Sigma + \Sigma = N(b | \mu, \Sigma + \Sigma)$ 6.11 迭代期望。

考虑两个具有联合分布 $p(x, y)$ 的随机变量 x, y 。显示

$$E[X] = E[Y] E[X | Y]。$$

这里, $E[X | Y]$ 表示 x 在条件分布 $p(x | Y)$ 下的期望值。

6.12 高斯随机变量的操作

考虑一个高斯随机变量 $x \sim N(x | \mu_x, \Sigma_x)$, 其中 $x \in \mathbb{R}^D$ 。
 此外, 我们有

$$y = Ax + b + w,$$

其中 $y \in \mathbb{R}^E$, $A \in \mathbb{R}^{E \times D}$, $b \in \mathbb{R}^E$, $w \sim N(w | 0, Q)$ 是独立的高斯噪声。“独立”意味着 x 和 w 是独立的随机变量并且 Q 是对角线的。A. 记下可能性 $p(y | x)$ 。b. 分布 $p(y) = p(y | x)p(x)dx$ 是高斯分布。计算平均值

μ_y 和协方差 Σ_y 。详细推导你的结果。C. 随机变量 y 正在根据测量映射进行变换

$$z = Cy + v,$$

其中 $z \in RF$ sian , $C \in RF \times E$,且 $v \sim N(0, R)$, R 为独立高斯
(测量)噪声。

- 记下 $p(z | y)$ 。
- 计算 $p(z)$,即平均 μ_z 和协方差 Σ_z 。详细推导你的结果。

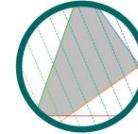
d.现在,测量值 y^{\wedge} 。计算后验分布 $p(x | y^{\wedge})$ 。

解决方案提示:这个后验也是高斯分布的,即我们只需要确定它的均值和协方差矩阵。
从显式计算联合高斯 $p(x, y)$ 开始。这也需要我们计算交叉协方差 $Cov_{x,y}[x, y]$ 和
 $Cov_{y,x}[y, x]$ 。然后应用高斯调节规则。

6.13概率积分变换给定连续随机变量 X ,cdf $F_X(x)$,
证明随机变量 $Y := F_X(X)$ 服从均匀分布 (定理 6.15) 。

7

持续优化



由于机器学习算法是在计算机上实现的,因此数学公式表示为数值优化方法。本章描述了训练机器学习模型的基本数值方法。训练机器学习模型通常归结为找到一组好的参数。“好”的概念由目标函数或概率模型决定,我们将在本书的第二部分看到相关示例。给定目标函数,使用优化算法找到最佳值。

本章涵盖连续优化的两个主要分支(图 7.2):无约束和约束优化。我们将在本章中假设我们的目标函数是可微的(见第 5 章),因此我们可以访问空间中每个位置的梯度来帮助我们找到最优值。按照惯例,机器学习中的大多数目标函数都是为了最小化,即最佳值是最小值。凭直觉找到最佳值就像找到目标函数的谷,梯度指向我们上坡。思路是下坡(与坡度相反),希望找到最深点。对于无约束优化,这是我们唯一需要的概念,但是有几种设计选择,我们将在第 7.1 节中讨论。对于约束优化,我们需要引入其他概念来管理约束(第 7.2 节)。我们还将介绍一类特殊的问题(第 7.3 节中的凸优化问题),我们可以在其中做出关于达到全局最优的陈述。

由于我们考虑数据和模

型

RD,我们面
临的优化问题是
连续的

优化问题,而不是
组合问题

离散变量的优化
问题。

考虑图 7.2 中的函数。该函数具有全局最小值 global minimum $x = -4.5$ 左右,函数值约为 -47。由于该函数是“平滑的”,因此可以使用梯度来指示我们应该向右还是向左迈出一步,从而帮助找到最小值。

这假设我们在正确的碗中,因为在 $x = 0.7$ 附近存在另一个局部最小值。回想一下,我们可以通过计算函数的导数并将其设置为零来求解函数的所有驻点。对于固定点

$$\ell(x) = x^4 + 7x^3 + 5x^2 - 17x + 3 \quad (7.1)$$

是真正的根源
导数,即具有零梯度的点。

我们得到相应的梯度 $d\ell(x) = 4x$

$$\frac{d}{dx} + 21x^2 + 10x - 17. \quad (7.2)$$

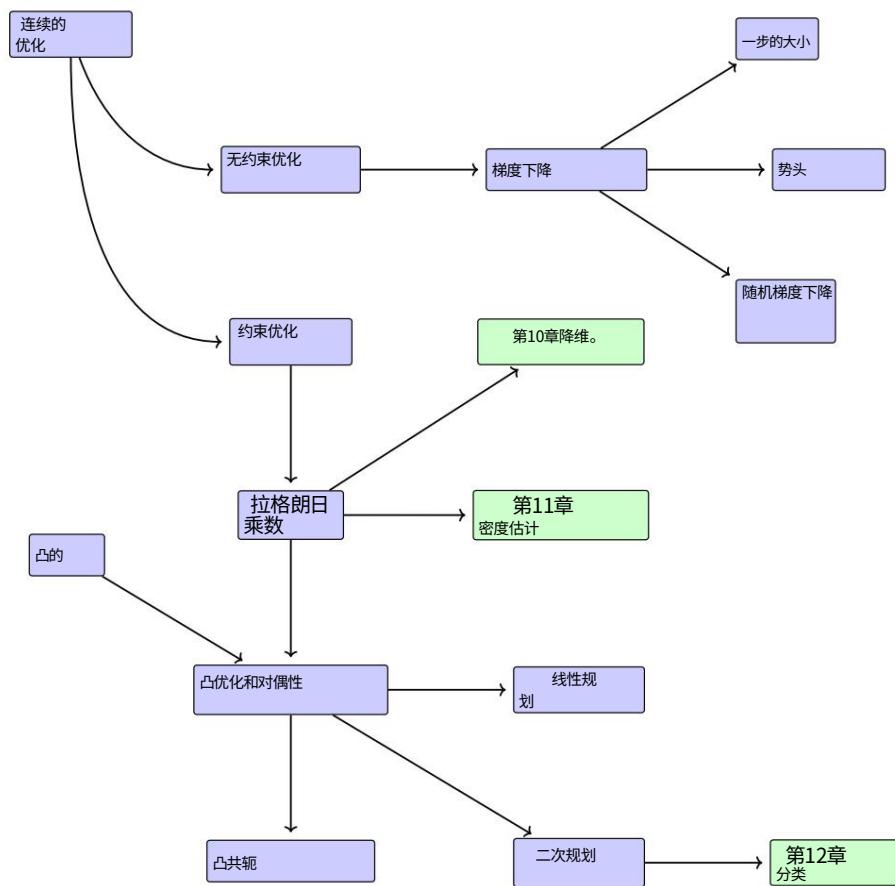
225

226

持续优化

图 7.2 相关概念的思维导图

优化,如本章所述。有两个主要思想:梯度下降和凸优化。



由于这是一个三次方程,当设置为零时,它通常有三个解。在示例中,其中两个是最小值,一个是最值 (大约 $x = -1.4$)。要检查驻点是最小值还是最大值,我们需要再次求导并检查驻点处的二阶导数是正还是负。在我们的例子中,二阶导数是

$$\frac{d^2 \ell(x)}{dx^2} = 12x + 42 < 0 \quad (7.3)$$

通过代入我们的视觉估计值 $x = -4.5, -1.4, 0.7$, 我们将观察到正如预期的那样, 中间点是最大值, 而其他两个静止点是最小值。

请注意,我们在前面的讨论中避免了对 x 的值进行解析求解,尽管对于低阶多项式(例如前面的内容)我们可以这样做。一般来说,我们无法找到解析解,因此我们需要从某个值开始,比如 $x_0 = -6$, 然后遵循负梯度。负梯度表示我们应该去

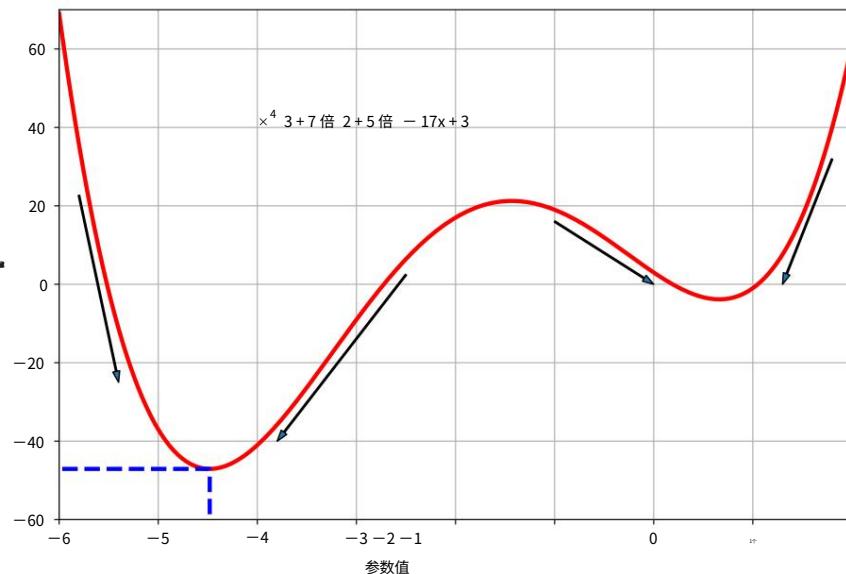


图 7.2示例目标函数。

负梯度用箭头表示,全局最小值用蓝色虚线表示。

正确,但不是多远 (这称为步长)。此外,如果我们根据已从右侧开始 (例如, $x_0 = 0$),则负梯度会导致我们到达错误的最小值。图 7.2 说明了这样一个事实,即对于 $x > -1$, 负梯度指向图中右侧的最小值,其具有更大的目标值。

Abel-Ruffini 定理,一般没有代数解

在 7.3 节中,我们将了解一类称为凸函数的函数,它们不会表现出对优化算法起点的这种棘手的依赖性。对于凸函数,所有局部最小值都是全局最小值。事实证明,许多机器学习目标对于凸函数,函数被设计为凸函数,我们将在第 12 章中看到一个示例。

5 次或以上的多项式 (Abel, 1826)。

所有局部最小值都是全局最小值。

到目前为止,本章的讨论是关于一维函数的,在这里我们能够将梯度、下降方向和最优值的概念形象化。在本章的其余部分,我们将在高维度上发展相同的想法。不幸的是,我们只能在一个维度上对概念进行可视化,但有些概念并不能直接推广到更高的维度,因此在阅读时需要格外小心。

7.1 使用梯度下降优化

我们现在考虑求解实值函数的最小值的问题

$$\text{分钟 } f(x), \quad (7.4)$$

其中 $f: \mathbb{R}^d \rightarrow \mathbb{R}$ 是捕获当前机器学习问题的目标函数。我们假设我们的函数 f 是可微分的，并且我们无法分析地找到封闭形式的解决方案。

梯度下降是一种一阶优化算法。要使用梯度下降找到函数的局部最小值，需要采取与函数在当前点的梯度负值成比例的步数。

我们使用
行向量约定
梯度。

回想一下 5.1 节，梯度指向最陡峭的上升方向。另一个有用的直觉是考虑函数处于某个值的线集 $\{f(x) = c\}$ 对于某个值 $c \in \mathbb{R}$ ，这些线被称为等高线。梯度指向与我们希望优化的函数的等高线正交的方向。

让我们考虑多元函数。想象一个表面（由函数 $f(x)$ 描述），球从特定位置 x_0 开始。当球被释放时，它会朝最陡峭的方向下坡。梯度下降利用了以下事实：如果一个人从 x_0 沿负梯度 $-(\nabla f)(x_0)$ 的方向移动，则 $f(x_0)$ 下降最快。我们在本书中假设函数是可微分的，并建议读者参考第 7.4 节中更一般的设置。那么，如果

$$x_1 = x_0 - \gamma (\nabla f)(x_0) \quad (7.5)$$

对于小步长 $\gamma > 0$ ，则 $f(x_1) < f(x_0)$ 。请注意，我们对梯度使用转置，否则尺寸将不起作用。

这一观察使我们能够定义一个简单的梯度下降算法：如果我们想找到函数 f 的局部最优值 $f(x^*)$ ：
 $R^n \rightarrow R, x \rightarrow f(x)$ ，我们从我们希望优化的参数的初始猜测 x_0 开始，然后根据

$$x_{i+1} = x_i - \gamma_i (\nabla f)(x_i) \quad (7.6)$$

对于合适的步长 γ_i ，序列 $f(x_0) \geq f(x_1) \geq \dots$ 收敛到局部最小值。

例 7.1 考虑一个二维的二次函数

$$F \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} x_1 & x_2 \\ x_2 & 20 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} 5 \\ 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad (7.7)$$

带渐变

$$\nabla F \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 \\ x_2 & 120 \end{pmatrix} - \begin{pmatrix} 5 \\ 3 \end{pmatrix}. \quad (7.8)$$

从初始位置 $x_0 = [-3, -1]^T$ 开始，我们迭代应用 (7.6) 以获得收敛于最小值的估计序列

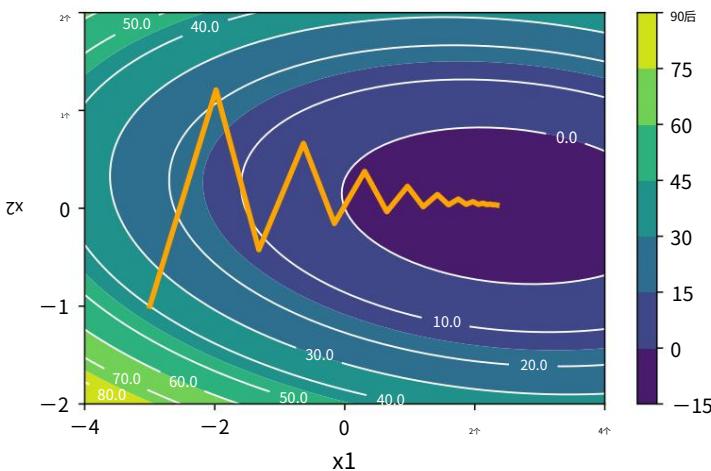


图 7.2 上的梯度下降

二维二次曲面（显示为热图）。有关说明，请参见示例 7.1。

（如图 7.2 所示）。我们可以看到（从图中和将 x_0 代入 (7.8) 且 $\gamma = 0.085$ ） x_0 处的负梯度指向北和东，导致 $x_1 = [-1.98, 1.21]$ 。重复该论证得到 $x_2 = [-1.32, -0.42]$ ，等等。

评论。梯度下降在接近最小值时可能相对较慢：它的渐近收敛速度不如许多其他方法。使用球从山上滚下的类比，当地表是一个细长的山谷时，问题的条件就很差（Trefethen 和 Bau III, 1997）。

对于条件较差的凸问题，随着梯度几乎垂直于指向最小点的最短方向，梯度下降越来越“之字形”；见图 7.2。 ◇

7.1.1 步长

如前所述，选择一个好的步长在梯度下降中很重要。如果步长太小，梯度下降会很慢。如果步长也称为学习步长选择太大，则梯度下降可能会过冲、无法收敛甚至发散。我们将在下一节讨论动量的消除梯度更新数据的不稳定行为并抑制振荡的方法。

速度。

自适应梯度方法在每次迭代时重新缩放步长，具体取决于函数的局部属性。有两种简单的启发式抽动法 (Toussaint, 2012)：

- 当函数值在梯度步长后增加时，步长太大。撤销步骤并减小步长。
- 当函数值减小时，步长可能会更大。尝试增加步长。

尽管“撤消”步骤似乎是一种资源浪费,但使用这种启发式方法可以保证单调收敛。

例 7.2 (求解线性方程组)

当我们求解 $Ax = b$ 形式的线性方程时,实际上我们通过找到最小化平方误差的 x^* 近似求解
 $Ax - b = 0 = (Ax - b)^\top (Ax - b)$

$$\|Ax - b\|^2 \quad (7.9)$$

如果我们使用欧几里得范数。 (7.9) 关于 x 的梯度为

$$\nabla x = 2(Ax - b)^\top A \quad (7.10)$$

我们可以直接在梯度下降算法中使用这个梯度。然而,对于这种特殊的特殊情况,事实证明存在解析解,可以通过将梯度设置为零来找到解析解。我们将在第 9 章中看到更多关于解决平方误差问题的内容。

评论。当应用于求解线性方程组 $Ax = b$ 时,梯度下降可能收敛较慢。梯度下降的收敛速度取决于条件数 $\kappa = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}$,它是 A 的最大奇异值与最小奇异值 (第 4.5 节) 之比。条件数本质上衡量的是最弯曲方向与最小弯曲方向的比率弯曲的方向,这对应于我们的形象 条件数。问题是细长的山谷:它们在一个方向上非常弯曲,但在另一个方向上非常平坦。不是直接求解 $Ax = b$,而是求解 $PP^{-1}x = P^{-1}b$ 称为预条件子。目标是设计 P 具有更好的条件数,但同时 P^{-1} 具有更好的梯度下降、预处理和收敛的更多信息,请参阅 Boyd 和 Vandenberghe (2004 年,第 9 章)。 ◇

条件数

预处理器

$$(P^{-1}A)^{-1}(P^{-1}b) = 0, \text{ 其中 } P^{-1} \text{ 很容易沟通}$$

7.1.2 动量梯度下降

如图 7.2 所示,如果优化曲面的曲率使得某些区域缩放不佳,则梯度下降的收敛速度可能会非常慢。曲率使得梯度下降步骤在谷壁之间跳跃,并以小步骤接近最佳值。建议的改善收敛性的调整是给梯度下降一些记忆。

Goh (2017) 写了一篇关于梯度下降的直观博客文章
势头。

动量梯度下降法 (Rumelhart 等人,1986 年) 是一种引入附加项以记住上一次迭代中发生的事情的方法。该内存抑制振荡并平滑梯度更新。继续用球类比,动量项模拟重球不愿改变方向的现象。这个想法是用内存进行梯度更新来实现

移动平均线。基于动量的方法记住每次迭代的更新 Δx_i 并将下一次更新确定为当前和先前梯度的线性组合

$$x_{i+1} = x_i - \gamma_i ((\nabla f)(x_i)) + \alpha \Delta x_i \quad (7.11)$$

$$- x_{i-1} = \alpha \Delta x_i - 1 - \gamma_{i-1} ((\nabla f)(x_{i-1})) , \quad (7.12)$$

其中 $\alpha \in [0, 1]$ 。有时我们只会近似地知道梯度。在这种情况下,动量项很有用,因为它可以对梯度的不同噪声估计进行平均。获得近似梯度的一种特别有用的方法是使用随机近似,我们将在下面讨论。

7.1.3 随机梯度下降

计算梯度可能非常耗时。然而,通常可以找到梯度的“廉价”近似值。近似梯度仍然有用,只要它指向与真实梯度大致相同的方向。

随机梯度下降

随机梯度下降 (通常简称为 SGD) 是梯度下降法的随机近似,用于最小化写为可微函数之和的目标函数。这里的随机这个词指的是我们承认我们并不精确地知道梯度,而是只知道一个有噪声的近似值。通过约束近似梯度的概率分布,我们仍然可以从理论上保证 SGD 会收敛。

在机器学习中,给定 $n = 1, \dots, N$ 个数据点,我们经常考虑的目标函数是每个样本 n 招致的损失 L_n 之和。在数学符号中,我们有以下形式

$$L(\theta) = \sum_{n=1}^N L_n(\theta), \quad (7.13)$$

其中 θ 是感兴趣参数的向量,即,我们想要找到最小化 L 的 θ 。回归 (第 9 章) 的一个例子是负对数似然,它表示为单个例子的对数似然之和,因此

$$L(\theta) = - \sum_{n=1}^N \log p(y_n | x_n, \theta), \quad (7.14)$$

其中 $x_n \in RD$ 是训练输入, y_n 是训练目标, θ 是回归模型的参数。

标准梯度下降,如前所述,是一种“批量”优化方法,即使用完整的训练集进行优化

通过根据更新参数向量

$$\theta_{i+1} = \theta_i - \gamma_i (\nabla L(\theta_i)) = \theta_i - \gamma_i \sum_{n=1}^N (\nabla L_n(\theta_i)) \quad (7.15)$$

对于合适的步长参数 γ_i 。评估总梯度可能需要对来自所有单个函数 L_n 的梯度进行昂贵的评估。当训练集很大和/或不存在简单公式时，评估梯度之和变得非常昂贵。

考虑这个词 (7.15) 中 $\sum_{n=1}^N (\nabla L_n(\theta_i))$ 。我们可以通过对较小的 L_n 集合求和来减少计算量。与批量梯度下降相反，它使用所有 L_n for $n = 1, \dots, N$ ，我们随机选择 L_n 的一个子集进行小批量梯度下降。在极端情况下，我们只随机选择一个 L_n 来估计梯度。为什么采用数据子集是明智的关键见解是要认识到梯度下降要收敛，我们只需要梯度是真实梯度的 $\sum_{n=1}^N (\nabla L_n(\theta_i))$ 无偏估计。事实上，(7.15) 中的项是梯度期望值（第 6.4.1 节）的经验估计。因此，期望值的任何其他无偏经验估计，例如使用数据的任何子样本，都足以使梯度下降收敛。

否

评论。当学习率以适当的速度下降时，并根据相对温和的假设，随机梯度下降几乎肯定会收敛到局部最小值 (Bottou, 1998)。 ◇为什么要考虑使用近似梯度？一个主要原因是实际实施限制，例如中央处理器 (CPU)/图形处理单元 (GPU) 内存的大小或计算时间的限制。我们可以用与估计经验均

值时考虑样本大小相同的方式来考虑用于估计梯度的子集的大小（第 6.4.1 节）。大的 mini-batch 大小将提供梯度的准确估计，减少参数更新中的方差。此外，大型 mini-batches 在成本和梯度的矢量化实现中利用高度优化的矩阵运算。方差的减少导致更稳定的收敛，但每次梯度计算都会更昂贵。

相比之下，小的 mini-batches 估计起来很快。如果我们保持 mini-batch 大小较小，梯度估计中的噪声将使我们能够摆脱一些糟糕的局部最优，否则我们可能会陷入其中。

在机器学习中，优化方法用于通过最小化训练数据上的目标函数来进行训练，但总体目标是提高泛化性能（第 8 章）。由于机器学习中的目标不一定需要精确估计目标函数的最小值，因此使用小批量方法的近似梯度已被广泛使用。随机梯度下降在大规模机器学习问题中非常有效 (Bottou 等人, 2018) ，

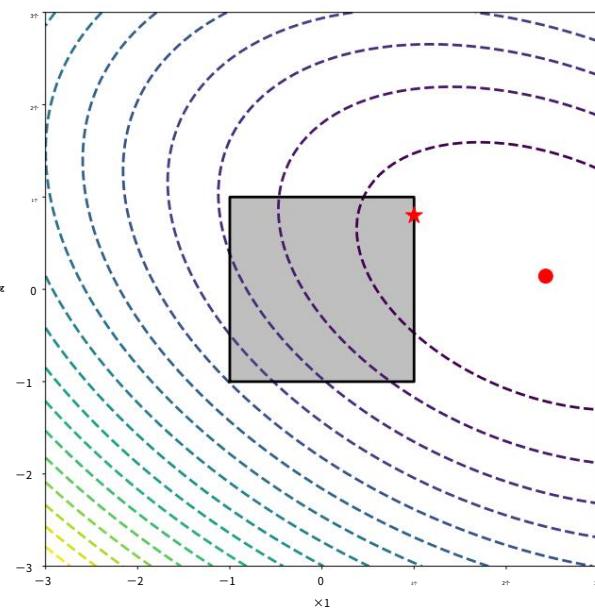


图 7.1 图示

约束优化。无
约束问题（由等高线
表示）在

右侧（用圆圈表示）。框
约束 ($-1 \leq x_1 \leq 1$
和 $-1 \leq x_2 \leq 1$)
要求最优解在框内，
从而得到由

星星。

例如在数百万图像上训练深度神经网络 (Dean et al., 2012)、主题模型 (Hoffman et al., 2013)、强化学习 (Mnih et al., 2015) 或训练大规模高斯过程模型 (Hensman 等人, 2013 年;Gal 等人, 2014 年)。

7.2 约束优化和拉格朗日乘子

在上一节中,我们考虑了求解函数最小值的问题

$$\underset{x}{\text{分钟}} \quad f(x), \quad (7.16)$$

其中 $f : RD \rightarrow R$ 。

在本节中,我们有额外的限制。也就是说,对于实值函数 $g_i : RD \rightarrow R$ for $i = 1, \dots, m$, 我们考虑约束优化问题 (见图 7.1 的说明)

$$\underset{x}{\text{分钟}} \quad f(x) \quad (7.17)$$

服从 $g_i(x) \leq 0$ 对于所有 $i = 1, \dots, m$ 。

值得指出的是,函数 f 和 g_i 通常可以是非凸函数,我们将在下一节中考虑凸函数。

将约束问题 (7.17) 转换为无约束问题的一种明显但不太实用的方法是使用指示函数

$$J(x) = f(x) + \sum_{i=1}^m I(g_i(x)), \quad (7.18)$$

其中 $\mathbf{1}(z)$ 是无限阶跃函数

$$\mathbf{1}(z) = \begin{cases} 0 & \text{如果 } z \geq 0 \\ \infty & \text{否则} \end{cases}. \quad (7.19)$$

如果不满足约束,这将给出无限惩罚,因此将提供相同的解决方案。然而,这个无限阶梯函数同样难以优化。我们可以通过引入 Lagrange multiplier 来克服这个困难。拉格朗日乘子的思想是用线性函数代替阶跃函数。

拉格朗日量

我们通过分别引入对应于每个不等式约束的拉格朗日乘数 $\lambda_i \geq 0$ 将拉格朗日问题 (7.17) 关联起来 (Boyd 和 Vandenberghe, 2004 年, 第 4 章), 使得

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) \quad (7.20a)$$

$$= f(x) + \lambda^T g(x), \quad (7.20b)$$

在最后一行中, 我们将所有约束 $g_i(x)$ 连接到向量 $g(x)$ 中, 并将所有拉格朗日乘数连接到向量 $\lambda \in \mathbb{R}^m$ 中。

我们现在介绍拉格朗日对偶的概念。一般来说, 优化中的对偶性是将一组变量 x (称为原始变量) 中的优化问题转换为另一组不同变量 λ (称为对偶变量) 中的优化问题。我们介绍两种不同的对偶性方法: 在本节中, 我们讨论拉格朗日对偶性; 在 7.3.3 节中, 我们讨论 Legendre-Fenchel 对偶性。

定义 7.1。(7.17) 中的问题

$$\underset{x}{\text{分钟}} \quad f(x) \quad (7.21)$$

$$\text{服从 } g_i(x) \leq 0 \text{ 对于所有 } i = 1, \dots, m$$

原始问题

被称为原始问题, 对应于原始变量 x 。

拉格朗日对偶问题

相关的拉格朗日对偶问题由下式给出

$$\underset{\lambda \in \mathbb{R}^m}{\text{最大}} \quad D(\lambda) \quad (7.22)$$

$$\text{受制于 } \lambda \geq 0,$$

其中 λ 是对偶变量, $D(\lambda) = \min_{x \in \mathbb{R}^d} L(x, \lambda)$ 。

评论。在定义 7.1 的讨论中, 我们使用了两个同样具有独立意义的概念 (Boyd 和 Vandenberghe, 2004 年)。

极小极大不等式

首先是极小极大不等式, 它表示对于具有两个参数 $\phi(x, y)$, $\max_{y \in Y} \min_{x \in X} \phi(x, y) \leq \min_{x \in X} \max_{y \in Y} \phi(x, y)$, 即

$$\max_{y \in Y} \min_{x \in X} \phi(x, y) \leq \min_{x \in X} \max_{y \in Y} \phi(x, y). \quad (7.23)$$

这个不等式可以通过考虑不等式来证明

$$\text{对于所有 } x, y \quad \underset{x}{\text{最小}} \phi(x, y) \underset{\lambda}{\text{最大}} \phi(x, y)。 \quad (7.24)$$

请注意,取(7.24)左侧y的最大值可以保持不等式,因为不等式对所有y都成立。同样,我们可以取(7.24)右边x上的最小值得到(7.23)。

第二个概念是弱对偶性,它用(7.23)来证明弱对偶性
原始值总是大于或等于对偶值。这在(7.27)中有更详细的描述。

回想一下(7.18)中的J(x)和(7.20b)中的拉格朗日函数之间的区别在于我们将指示函数放宽为线性函数。因此,当 $\lambda = 0$ 时,拉格朗日L(x,λ)是J(x)的下界。因此,L(x,λ)相对于λ的最大值是J(x) = max L(x,λ)。

$$\underset{\lambda=0}{\text{最小最大}} x \in \mathbb{R}^d L(x, \lambda) \quad (7.25)$$

回想一下,最初的问题是最小化J(x), L(x,λ)。

$$\underset{\lambda=0}{\text{最小最大}} x \in \mathbb{R}^d L(x, \lambda) \quad (7.26)$$

根据minimax不等式(7.23),可以得出交换最小值和最大值的顺序会导致较小的值,即,

$$\underset{\lambda=0}{\text{最小最大}} x \in \mathbb{R}^d L(x, \lambda) \underset{\lambda=0}{\text{最大值}} \underset{x \in \mathbb{R}^d}{\text{最小}} L(x, \lambda)。 \quad (7.27)$$

这也称为弱对偶性。请注意,右弱对偶手边的内部是对偶目标函数D(λ),定义如下。

与具有约束的原始优化问题相反, $\min_{x \in \mathbb{R}^d} L(x, \lambda)$ 是给定λ值的无约束优化问题。如果求解 $\min_{x \in \mathbb{R}^d} L(x, \lambda)$ 很容易,那么整个问题就很容易求解。我们可以从(7.20b)中观察到L(x,λ)对λ是仿射的。因此 $\min_{x \in \mathbb{R}^d} L(x, \lambda)$ 是λ的仿射函数的逐点最小值,因此D(λ)是凹的,即使f(·)和gi(·)可能是非凸的。外部问题,即λ上的最大化,是凹函数的最大值,可以有效计算。

假设f(·)和gi(·)可微,我们通过对拉格朗日量对x求微分,将微分设为零,求解最优值,从而找到拉格朗日对偶问题。我们将在7.3.1和7.3.2节中讨论两个具体示例,其中f(·)和gi(·)是

凸的。

备注(平等约束)。考虑带有附加等式约束的(7.17)

$$\begin{array}{ll} \text{分钟} & f(x) \\ \underset{x}{\text{最小}} & \\ \text{服从} & g_i(x) = 0 \quad i = 1, \dots, * \\ & h_j(x) = 0 \quad \text{对于所有} j = 1, \dots, m \end{array} \quad (7.28)$$

我们可以通过用两个不等式约束替换它们来对等式约束进行建模。也就是说，对于每个等式约束 $h_j(x) = 0$ ，我们等效地用两个约束 $h_j(x) \geq 0$ 和 $h_j(x) \leq 0$ 替换它。事实证明，由此产生的拉格朗日乘子是不受约束的。

因此，我们将 (7.28) 中不等式约束对应的拉格朗日乘数约束为非负，不约束等式约束对应的拉格朗日乘数。 ◇

7.3 凸优化

我们将注意力集中在一类特别有用的优化问题上，我们可以在其中保证全局最优性。当 $f(\cdot)$ 为凸函数，且 $g(\cdot)$ 和 $h(\cdot)$ 的约束为凸集时，凸优化称为凸优化问题。在这种情况下，我们具有强对偶性：对偶问题的最优解与原始问题的强对偶性最优解相同。凸函数和凸集之间的区别在机器学习文献中往往没有严格的表述，但人们通常可以从上下文中推断出隐含的含义。

凸集

定义 7.2。集合 C 是一个凸集，如果对于任何 $x, y \in C$ 和任何标量 $\theta \in [0, 1]$ ，我们有

$$\theta x + (1 - \theta)y \in C. \quad (7.29)$$

图 7.2 凸集示例。



图 7.3 非凸集示例。

凸集是这样的集合，即连接集合中任意两个元素的直线位于集合内部。图 7.2 和 7.3 分别说明了凸集和非凸集。

凸函数是函数中任意两点之间的直线位于函数上方的函数。图 7.2 显示了一个非凸函数，图 7.2 显示了一个凸函数。另一个凸函数如图 7.2 所示。

凸函数
凹函数

题词

定义 7.3。设函数 $f: RD \rightarrow R$ 是定义域为凸集的函数。函数 f 是一个凸函数，如果对于 f 域中的所有 x, y 以及具有 $0 \leq \theta \leq 1$ 的任何标量 θ ，我们有

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y). \quad (7.30)$$

评论。凹函数是凸函数的负数。

◇ (7.28) 中涉及 $g(\cdot)$ 和 $h(\cdot)$ 的约束在标量值处截断函数，从而产生集合。凸函数和凸集之间的另一个关系是考虑通过“填充”凸函数获得的集合。凸函数是一个碗状的物体，我们想象将水倒入其中以将其填满。这个生成的填充集称为凸函数的题记，是一个凸集。

如果函数 $f: R^n \rightarrow R$ 是可微的，我们可以指定凸性

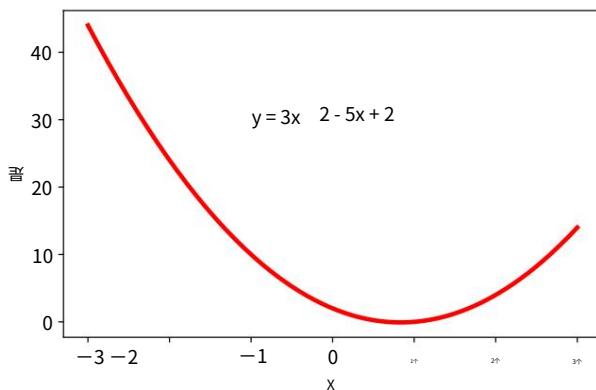


图 7.2 凸面示例

功能。

其梯度 $\nabla f(x)$ 的项（第 5.2 节）。函数 $f(x)$ 是凸函数当且仅当对于任意两点 x, y 它认为

$$f(y) \geq f(x) + \nabla f(x) \cdot (y - x) \quad (7.31)$$

如果我们进一步知道函数 $f(x)$ 是二次可微的，即 Hessian (5.147) 对于 x 域中的所有值都存在，那么函数 $f(x)$ 是凸函数当且仅当 $\nabla^2 f(x)$ 是半正定的 (Boyd 和 Vandenberghe, 2004)。

$\nabla^2 f(x)$ 是半正定的 (Boyd 和 Vandenberghe, 2004)。

示例 7.3 对于 $x > 0$

负熵 $f(x) = x \log_2 x$ 是凸函数。该函数的可视化如图 7.2 所示，我们可以看到该函数是凸函数。为了说明前面的凸性定义，让我们检查两个点 $x = 2$ 和 $x = 4$ 的计算。请注意，要证明 $f(x)$ 的凸性，我们需要检查所有点 $x \in \mathbb{R}$ 。

回顾定义 7.3。考虑两点中间的一个点（即 $\theta = 0.5$ ）；那么左边是 $f(0.5 \cdot 2 + 0.5 \cdot 4) = 3 \log_2 3 \approx 4.75$ 。右边是 $0.5(2 \log_2 2) + 0.5(4 \log_2 4) = 1 + 4 = 5$ 。因此满足定义。

由于 $f(x)$ 是可微的，我们可以替代地使用 (7.31)。计算中 $f(x)$ 的导数，我们得到

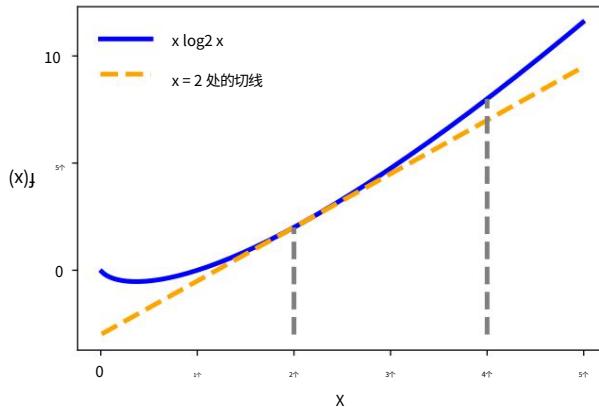
$$\nabla f(x \log_2 x) = 1 \cdot \log_2 x + x \cdot \frac{1}{x} = \log_2 x + 1 \quad (7.32)$$

(7.31) 的左侧由下式给出 $f(4) = 8$ 。右边是

$$f(x) + \nabla f(x)(y - x) = f(2) + \nabla f(2) \cdot (4 - 2) \quad (7.33a)$$

$$= 2 + (1 +) \cdot 2 \approx 6.9 \quad (7.33b)$$

图 7.2 负熵函数
(凸函数)及其在
 $x = 2$ 处的正切。



我们可以通过回顾定义来根据第一性原理检查一个函数或集合是否是凸的。在实践中,我们经常依靠保留凸性的操作来检查特定函数或集合是否为凸性。尽管细节有很大不同,但这又是我们在第 2 章中针对向量空间介绍的闭包思想。

示例 7.4 凸函

数的非负加权和是凸函数。观察到如果 f 是凸函数,并且 $\alpha \geq 0$ 是非负标量,则函数 αf 是凸函数。我们可以通过将 α 乘以定义 7.3 中的等式两边来看到这一点,并且回想一下,乘以一个非负数并不会改变不等式。

如果 f_1 和 f_2 是凸函数,那么根据定义我们有

$$f_1(\theta x + (1 - \theta)y) \leq \theta f_1(x) + (1 - \theta)f_1(y) \quad (7.34)$$

$$+ (1 - \theta)y \leq \theta f_2(x) + (1 - \theta)f_2(y) \quad (7.35)$$

总结双方给我们

$$f_1(\theta x + (1 - \theta)y) + f_2(\theta x + (1 - \theta)y)$$

$$\theta f_1(x) + (1 - \theta)f_1(y) + \theta f_2(x) + (1 - \theta)f_2(y), \quad (7.36)$$

右侧可以重新排列为

$$\theta(f_1(x) + f_2(x)) + (1 - \theta)(f_1(y) + f_2(y)), \quad (7.37)$$

完成凸函数之和为凸的证明。

结合前面两个事实,我们看到 $\alpha f_1(x) + \beta f_2(x)$ 对于 $\alpha, \beta \geq 0$ 是凸的。对于两个以上的凸函数的非负加权和,可以使用类似的参数来扩展这个闭包属性。

7.3 凸优化

评论。(7.30) 中的不等式有时称为 Jensen 不等式。Jensen 不等式事实上,一整类对凸函数取非负加权和的不等式都称为 Jensen 不等式。 ◇

总之,约束优化问题称为凸优化-凸优化
化问题如果 问题

$$\begin{array}{ll} \underset{x}{\text{最小}} f(x) \\ \text{服从 } g_i(x) \leq 0 & i = 1, \dots, m \\ h_j(x) = 0 & \text{对于所有 } j = 1, \dots, n, \end{array} \quad (7.38)$$

其中所有函数 $f(x)$ 和 $g_i(x)$ 都是凸函数,并且所有 $h_j(x) = 0$ 都是凸集。在下文中,我们将描述两类广泛使用且易于理解的凸优化问题。

7.3.1 线性规划考虑所有前面的函数都

是线性的特殊情况,即

$$\begin{array}{ll} \underset{x \in \mathbb{R}^d}{\text{最小}} c^T x \\ \text{服从 } Ax \leq b \end{array}, \quad (7.39)$$

其中 $A \in \mathbb{R}^{m \times d}$ 且 $b \in \mathbb{R}^m$ 。这被称为线性程序。它有 d 个线性程序变量和 m 个线性约束。拉格朗日由线性程序给出,是工业中使用最广泛的方法之一。

$$L(x, \lambda) = c^T x + \lambda^T (Ax - b), \quad (7.40)$$

其中 $\lambda \in \mathbb{R}^m$ 是非负拉格朗日乘数的向量。重新排列对应于 x 的项

$$L(x, \lambda) = (c + A^T \lambda)^T x - \lambda^T b. \quad (7.41)$$

取 $L(x, \lambda)$ 关于 x 的导数并将其设置为零得到我们

$$c + A^T \lambda = 0. \quad (7.42)$$

因此,对偶拉格朗日量为 $D(\lambda) = -\lambda^T b$ 。回想一下我们想要
导数为零的约束外,我们还有 $\lambda \geq 0$ 的事实,导致以下对偶优化问题

$$\begin{array}{ll} \underset{\lambda \in \mathbb{R}^m}{\text{最大}} & -b^T \lambda \\ \text{服从 } & c^T + A^T \lambda = 0 \quad \lambda \geq 0. \end{array} \quad (7.43)$$

最小化原始和最大
化对偶是惯例。

这也是一个线性规划,但有 m 个变量。我们可以选择求解原始 (7.39) 或对偶 (7.43) 程序,具体取决于

m 或 d 是否较大。回想一下， d 是变量数， m 是原始线性规划中的约束数。

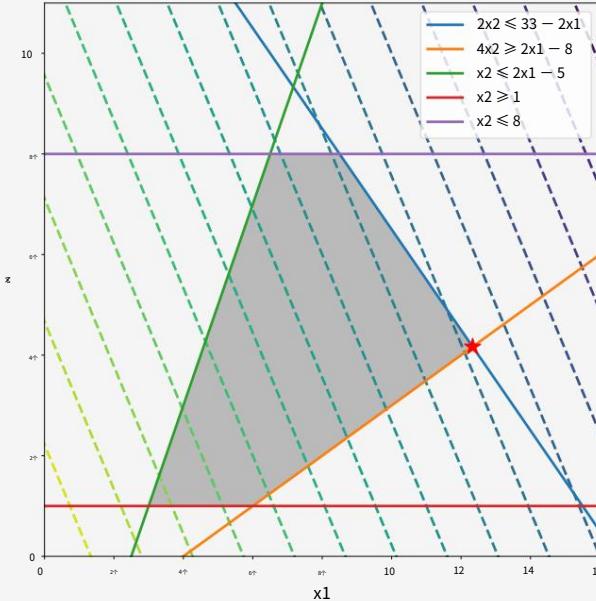
例 7.5 (线性规划) 考虑线性程序

$$\begin{array}{ll}
 \text{最小} & -5 \quad x_1 \\
 x \in R^2 & 3 \quad x_2 \\
 \\
 \text{受制于} & z \geq 33 \\
 & 2x_2 - 4x_1 & 8 \quad (7.44) \\
 & -2x_1 & 5 \\
 & 0 & -1 \\
 & 0 & 8 \text{个}
 \end{array}$$

有两个变量。该程序也如图 7.1 所示。目标函数是线性的，产生线性轮廓线。标准形式的约束集被翻译成图例。最优值必须位于阴影（可行）区域，并用星号表示。

图 7.1 a 的
图示
线性程序。无约束问题
(由等高线表
示)在右侧具有最小
值。给定约束的
最优值是

由明星展示。



7.3.2 二次规划

考虑凸二次目标函数的情况,其中约束是仿射的,即

$$\underset{x \in \mathbb{R}^d}{\text{最小}} \quad -x^\top Qx + c^\top x \quad (7.45)$$

服从 $Ax = b$,

其中 $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^d$ 。正方形对称矩阵 $Q \in \mathbb{R}^{d \times d}$ 是正定的,因此目标函数是凸的。

这被称为二次规划。观察它有 d 个变量和 m 个线性约束。

示例 7.6 (二次程序)

考虑二次规划

$$\underset{x \in \mathbb{R}^2}{\text{最小}} \quad -\frac{x_1^2}{2} - \frac{x_2^2}{4} + \frac{x_1}{x_2} + \frac{5}{3} \quad (7.46)$$

$$\begin{array}{ll} \text{受制于} & \begin{array}{ll} 0 & -1 \\ 0 & 0 \\ 1 & 1 \\ x_1 & x_2 \\ x_2 & x_2 \\ 0 & -1 \end{array} \end{array} \quad (7.47)$$

的两个变量。该程序也如图 7.1 所示。目标函数是具有半正定矩阵 Q 的二次函数,从而产生椭圆等高线。最优值必须位于阴影 (可行) 区域,并用星号表示。

拉格朗日量由

$$L(x, \lambda) = -x^\top Qx + c^\top x + \lambda^\top (Ax - b) \quad (7.48a)$$

$$= -x^\top Qx + (c + A^\top \lambda)^\top x - \lambda^\top b, \quad (7.48b)$$

我们再次重新排列了术语。取 $L(x, \lambda)$ 关于 x 的导数并将其设置为零给出

$$Qx + (c + A^\top \lambda) = 0. \quad (7.49)$$

假设 Q 是可逆的,我们得到

$$x = -Q^{-1}(c + A^\top \lambda). \quad (7.50)$$

将 (7.50) 代入原始拉格朗日 $L(x, \lambda)$,我们得到对偶拉格朗日

$$D(\lambda) = -2 - (c + A^\top \lambda)^\top Q^{-1}(c + A^\top \lambda) - \lambda^\top b. \quad (7.51)$$

因此,对偶优化问题由下式给出

$$\underset{\lambda \in \mathbb{R}^m}{\text{最大}} \quad - \frac{1}{2} (c + A - \lambda)^T Q^{-1} (c + A - \lambda) - \lambda^T b \quad (7.52)$$

服从 $\lambda \geq 0$ 。

我们将在第 12 章看到二次规划在机器学习中的应用。

7.3.3 Legendre-Fenchel 变换和凸共轭让我们重新审视 7.2 节中的对偶性概

念,而不考虑约束。关于凸集的一个有用的事是它可以被它的支持超平面等价地描述。如果超平面与凸集相交,并且凸集仅包含在它的一侧,则超平面称为凸集的支持超平面。回想一下,我们可以填充一个凸函数来获得题词,这是一个凸集。因此,我们也可以用支持超平面来描述凸函数。此外,观察支撑超平面刚好接触凸函数,实际上是该点函数的切线。

支持超平面

回想一下,函数 $f(x)$ 在给定点 x_0 处的正切是该函数在该点处的梯度的求值摘要,因为凸集可以由它们的支持超平面等效地描述,凸函数可以等效地描述通过它们梯度的勒让德变换 $\frac{df(x)}{dx} \Big|_{x=x_0}$ 。在勒让德变换函数。勒让德变换形式化了这个概念。

物理专业的学生经常被介绍给勒让德变换,因为它与拉格朗日量和哈密顿量有关

我们从最一般的定义开始,不幸的是它有一种违反直觉的形式,然后查看特殊情况以将定义与上一段中描述的直觉联系起来。Legendre-Fenchel 变换是从凸微分函数 $f(x)$ 到依赖于切线 $s(x) = \nabla_x f(x)$ 的函数的变换(在傅里叶变换的意义上)。值得强调的是,这是函数 $f(\cdot)$ 的变换,而不是变量 x 或在 x 处求值的函数。

经典力学。
勒让德-芬歇尔变换
凸共轭

Legendre-Fenchel 变换也称为凸共轭(原因我们很快就会看到)并且与对偶性密切相关(Hiriart-Urruty 和 Lemaréchal,2001 年,第 5 章)。

凸共轭

定义 7.4。函数 f 的凸共轭: $\mathbb{R}^D \rightarrow \mathbb{R}$ 是函数 f 定义为

$$f^*(s) = \sup_{x \in \mathbb{R}^D} (s^T x - f(x)) \quad (7.53)$$

请注意,前面的凸共轭定义不需要函数 f 是凸的或可微的。在定义 7.4 中,我们使用了一般内积(第 3.2 节),但在本节的其余部分我们

将考虑有限维向量($s, x = s - x$)之间的标准点积以避免过多的技术细节。

要以几何方式理解定义 7.4,请考虑一个很好的推导:

简单的一维凸可微函数,例如 $f(x) = x$ 超平面缩减为一条直线。考虑一条直线 $y = sx + c$ 。

回想一下, .请注意,由于我们正在研究一维问题,

我们能够通过支持超平面来描述凸函数,所以让我们尝试通过支持线来描述这个函数 $f(x)$ 。固定直线 $s \in \mathbb{R}$ 的梯度,对于 f 的图上的每个点 $(x_0, f(x_0))$,找到 c 的最小值,使得直线仍然与 $(x_0, f(x_0))$ 相交。请注意, c 的最小值是斜率为 s 的直线“恰好接触”函数 $f(x) = x$ 且梯度为 s 的位置,由下式给出

通过绘制推
理最容易理解

进步。

².通过 $(x_0, f(x_0))$ 的直线

$$y - f(x_0) = s(x - x_0). \quad (7.54)$$

这条线的 y 轴截距是 $-sx_0 + f(x_0)$ 。因此, $y = sx + c$ 与 f 的图形相交的 c 的最小值是

$$\text{息}_{x_0} - sx_0 + f(x_0)。信 \quad (7.55)$$

前面的凸共轭按照惯例定义为 this 的负数。本段中的推理并不依赖于我们选择一维凸可微函数这一事实,并且对 $f: \mathbb{R}^d \rightarrow \mathbb{R}$ 成立,它们是非凸且不可微的。

经典的
勒让德变换是在凸面上定义的
 \mathbb{R}^d 中的可微函数。

评论。凸可微函数,例如示例 $f(x) = x$ 是一个很好的特例,其中不需要上确界,并且函²是数与其 Legendre 变换之间存在一对一的对应关系。让我们从第一原理中得出这一点。对于凸可微函数,我们知道在 x_0 处切线与 $f(x_0)$ 相交,因此

$$f(x_0) = sx_0 + c. \quad (7.56)$$

回想一下,我们想用梯度 $\nabla f(x)$ 来描述凸函数 $f(x)$,并且 $s = \nabla f(x_0)$ 。我们重新排列以获得 $-c$ 的表达式以获得

$$-c = sx_0 - f(x_0). \quad (7.57)$$

请注意, $-c$ 随 x_0 变化,因此随 s 变化,这就是为什么我们可以将其视为 s 的函数,我们称之为

$$f^*(s) := sx_0 - f(x_0). \quad (7.58)$$

比较 (7.58) 和定义 7.4,我们看到 (7.58) 是一个特例(没有上确界)。◆共轭函数有很好的性质;例如,对于凸函数,再次应用勒让德变换可以让我们回到原来的函数。

同理 $f(x)$ 的斜率为 s , $f^*(s)$ 的斜率

是 x 。以下两个示例展示了凸共轭在机器学习中的常见用法。

例 7.7 (凸共轭)

为了说明凸共轭的应用,考虑二次函数

$$f(y) = \frac{\lambda}{2} y^T K^{-1} y \quad (7.59)$$

基于正定矩阵 $K \in \mathbb{R}^{n \times n}$ 。我们将原始变量表示为 $y \in \mathbb{R}^n$,将对偶变量表示为 $\alpha \in \mathbb{R}^n$ 。

应用定义 7.4,我们得到函数

$$f^*(\alpha) = \sup_{y \in \mathbb{R}^n} y^T \alpha - \frac{\lambda}{2} y^T K^{-1} y. \quad (7.60)$$

由于函数是可微分的,我们可以通过取导数并将 y 设置为零来找到最大值。

$$\frac{\partial}{\partial y} (y^T \alpha - \frac{\lambda}{2} y^T K^{-1} y) = (\alpha - \lambda K^{-1} y) \quad (7.61)$$

因此,当梯度为零时,我们有 $y = (7.60)$ 收益率 $\lambda K \alpha$ 。代入

$$(7.62)$$

$$(7.62)$$

$$(7.62)$$

示例 7.8 在机器

学习中,我们经常使用函数求和;例如,训练集的目标函数包括训练集中每个示例的损失总和。在下文中,我们推导出损失总和的凸共轭 $\ell(t)$,其中 $\ell : \mathbb{R} \rightarrow \mathbb{R}$ 。这也说明了凸共轭在矢量情况下的应用。让 $L(t) = \ell_i(t_i)$ 。

然后,

$$L^*(z) = \sup_{t \in \mathbb{R}^n} z^T t - \sum_{i=1}^n \ell_i(t_i) \quad (7.63a)$$

$$= \sup_{t \in \mathbb{R}^n} \sum_{i=1}^n z_i t_i - \ell_i(t_i) \quad \text{点积的定义(7.63b)}$$

$$= \sum_{i=1}^n \text{支持}_{t \in \mathbb{R}^n} z_i t_i - \ell_i(t_i) \quad (7.63c)$$

$$= \sum_{i=1}^n \ell_i^*(z_i) \quad \text{共轭的定义(7.63d)}$$

回想一下，在7.2节中，我们使用拉格朗日乘数推导出了一个对偶优化问题。此外，对于凸优化问题，我们具有很强的对偶性，即原始问题和对偶问题匹配的解决方案。此处描述的 Legendre-Fenchel 变换也可用于推导对偶优化问题。此外，当函数是凸函数且可微时，上确界是唯一的。为了进一步研究这两种方法之间的关系，让我们考虑一个线性等式约束凸优化问题。

例 7.9 令 $f(y)$ 和
 $g(x)$ 为凸函数， A 为适当维数的实数矩阵，使得 $Ax = y$ 。然后

$$\underset{x}{\text{分钟}} f(Ax) + g(x) = \text{最小值} \quad f(y) + g(x) \quad (7.64)$$

通过为约束 $Ax = y$ 引入拉格朗日乘数 u ，

$$\underset{\substack{x \\ \text{轴}=y}}{\text{分钟}} f(y) + g(x) = \text{最小值} \quad \underset{\substack{\text{坐标轴} \\ y}}{\text{最大限度}} f(y) + g(x) + (Ax - y) \quad u \quad (7.65a)$$

$$\underset{\substack{x,y \\ \text{轴}=y}}{\text{最大}} f(y) + g(x) + (Ax - y) \quad u, \quad (7.65b)$$

最后一步交换 \max 和 \min 是因为 $f(y)$ 和 $g(x)$ 是凸函数。通过拆分点积项并收集 x 和 y ，
 $f(y) + g(x) + (Ax - y) \quad u$

$$\underset{\substack{x,y \\ \text{轴}=y}}{\text{最大}} f(y) + g(x) + (Ax - y) \quad u \quad (7.66a)$$

$$= \underset{x}{\text{最大}} -y \quad \underset{x}{\text{最小}} f(y) + \min_x (Ax) \quad u + g(x) \quad (7.66b)$$

$$= \underset{x}{\text{最大}} -y \quad \underset{x}{\text{最小}} f(y) + \min_x A_x \quad u + g(x) \quad (7.66c)$$

回想一下凸共轭（定义 7.4）和点积 - 对于一般内积， A 被伴随 A^* 替代
 $ucts$ 是对称的，

$$\underset{y}{\text{最大限度}} \underset{x}{\text{分钟}} -y \quad u + f(y) + \min_x A_x \quad u + g(x) \quad (7.67a)$$

$$= \underset{y}{\text{最大}} -f(u) - g(-A_y) \quad (7.67b)$$

因此，我们已经证明

$$\underset{x}{\text{分钟}} f(Ax) + g(x) = \max_u (u) - g(-A - f) \quad (7.68)$$

事实证明,Legendre-Fenchel 共轭对于可以表示为凸优化问题的机器学习问题非常有用。特别是,对于独立应用于每个示例的凸损失函数,共轭损失是推导对偶问题的一种便捷方式。

7.4 进一步阅读持续优化是一个活跃的研究领域,我们并不试图全面介绍最近的进展。

从梯度下降的角度来看,有两个主要的弱点,每个都有自己的一套文献。第一个挑战是梯度下降是一阶算法,不使用有关曲面曲率的信息。当有长谷时,梯度垂直于感兴趣的方向。动量的概念可以推广到一类一般的加速方法(Nesterov, 2018)。共轭梯度法通过考虑先前的方向避免了梯度下降所面临的问题(Shewchuk, 1994)。牛顿法等二阶方法使用Hessian矩阵提供有关曲率的信息。许多选择步长的选择和动量等想法都是通过考虑目标函数的曲率而产生的(Goh, 2017年; Bottou 等人, 2018年)。L-BFGS等拟牛顿方法尝试使用更便宜的计算方法来近似 Hessian (Nocedal 和 Wright, 2006)。最近,人们对计算下降方向的其他指标产生了兴趣,从而产生了镜像下降(Beck 和 Teboulle, 2003 年)和自然梯度(Toussaint, 2012 年)等方法。

第二个挑战是处理不可微函数。当函数中存在扭结时,梯度方法的定义不明确。

在这些情况下,可以使用次梯度法(Shor, 1985)。有关优化不可微函数的更多信息和算法,请参阅 Bertsekas (1999) 的书。有大量关于数值求解连续优化问题的不同方法的文献,包括约束优化问题的算法。欣赏这些文献的良好起点是 Luenberger (1969) 和 Bonnans 等人的著作。(2006)。Bubeck (2015) 提供了最近关于持续优化的调查。

Hugo Gonçalves
的博客也是一个很好的
资源,可以更容易
地进行介绍
到 Legendre-Fenchel
变换:

<https://tinyurl.com/ydaal7hj>

机器学习的现代应用通常意味着数据集的大小禁止使用批量梯度下降,因此随机梯度下降是当前大规模机器学习方法的主力。最近的文献调查包括 Hazan (2015) 和 Bottou 等人。(2018)。

对于对偶性和凸优化,Boyd 和 Vandenberghe (2004) 的书包括在线讲座和幻灯片。Bertsekas (2009) 提供了一种更数学化的处理方法,最近的一本书由一位

优化领域的主要研究人员是 Nesterov (2018)。凸优化基于凸分析,对凸函数的更多基础结果感兴趣的读者可以参考 Rock afellar (1970)、Hiriart-Urruty 和 Lemar'echal (2001),以及 Borwein 和 Lewis (2006)。 Legendre-Fenchel 变换也包含在上述关于凸分析的书籍中,但 Zia 等人提供了对初学者更友好的介绍。(2009)。 Polyak (2016) 调查了 Legendre-Fenchel 变换在凸优化算法分析中的作用。

练习

7.1 考虑单变量函数

$$f(x) = x^3 + 6x^2 - 3x - 5.$$

找到它的固定点并指出它们是最大、最小、极小还是鞍点。

7.2 考虑随机梯度下降的更新方程 (方程 (7.15))。

记下我们使用 1 的小批量时的更新。

7.3 考虑下列陈述是否正确:

- A. 任意两个凸集的交集都是凸的。 b. 任意两个凸集的并集都是凸的。 C. 凸集 A 与另一个凸集 B 的区别是凸的。

7.4 考虑下列陈述是否正确:

- A. 任意两个凸函数之和是凸的。 b. 任意两个凸函数的差是凸的。 C. 任何两个凸函数的乘积都是凸的。 d. 任意两个凸函数的最大值是凸的。

7.5 将下列优化问题表示为标准线性规划

矩阵符号

$$\max_{x \in \mathbb{R}^2, \xi \in \mathbb{R}} p$$

受 $\xi \geq 0$ 、 $x_0 \leq 0$ 和 $x_1 \leq 3$ 的约束。

7.6 考虑图 7.1 所示的线性程序,

最小	-	x_1	
$x \in \mathbb{R}^2$		x_2	
		$\begin{matrix} 5 \\ 3 \end{matrix}$	
			$\begin{matrix} 2 \\ 3 \end{matrix}$
			$\begin{matrix} 2 \\ 1 \end{matrix}$
		x_1	$\begin{matrix} 8 \\ 5 \end{matrix}$
		x_2	
			-1
			$\begin{matrix} 1 \\ 8 \end{matrix}$
			0

使用拉格朗日对偶性推导对偶线性程序。

7.7 考虑图 7.1 所示的二次规划，

$$\begin{array}{ll}
 \text{分} & \frac{1}{2} x_1^2 + \frac{2}{3} x_1 x_2 + \frac{5}{3} x_2^2 \\
 \text{受制于} & \begin{array}{l}
 \frac{1}{2} x_1 + x_2 \leq 0 \\
 -x_1 + x_2 \leq 0 \\
 x_1, x_2 \geq 0
 \end{array}
 \end{array}$$

使用拉格朗日对偶性推导对偶二次规划。

7.8 考虑以下凸优化问题

$$\min_{w \in RD} \frac{1}{2} w^T w$$

服从 $w^T x = 1$ 。

通过引入拉格朗日乘数入 推导拉格朗日对偶。

7.9 考虑 $x \in RD$ 的负熵，

$$f(x) = \sum_{d=1}^D \log x_d$$

推导凸共轭函数 f^* 乘积。

(s) , 通过假设标准点

提示: 采用适当函数的梯度并将梯度设置为零。

7.10 考虑函数

$$f(x) = \frac{1}{2} x^T A x + b^T x + c, \quad ,$$

其中 A 是严格正定的, 这意味着它是可逆的。推导 $f(x)$ 的凸共轭。

提示: 采用适当函数的梯度并将梯度设置为零。

7.11 链损失 (支持向量机使用的损失) 是

由

$$L(a) = \max\{0, 1 - a\},$$

如果我们对应用 L-BFGS 等梯度方法感兴趣, 并且不想求助于次梯度方法, 我们需要平滑链损失中的扭结。计算链损失 $L(\beta)$ 的凸共轭是对偶变量。添加 ℓ_2 近项, 并计算结果函数的共轭

(β) 其中

$$\text{大号}^*(\beta) + \beta 2 \frac{\gamma}{2},$$

其中 γ 是给定的超参数。

第二部分

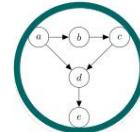
中央机器学习问题

249

该材料由剑桥大学出版社出版,名为Marc Peter Deisenroth、A. Aldo Faisal 和 Cheng Soon Ong的机器学习数学(2020)。此版本可免费查看和下载,仅供个人使用。不得重新分发、转售或用于衍生作品。© MP Deisenroth、AA Faisal 和 CS Ong,2021年。<https://mml-book.com>。

8个

当模型遇到数据



在本书的第一部分,我们介绍了构成许多机器学习方法基础的数学。希望读者能够从第一部分学习数学语言的基本形式,我们现在将用它来描述和讨论机器学习。本书的第二部分介绍了机器学习的四大支柱:

- 回归 (第 9 章)
- 降维 (第 10 章)
- 密度估计 (第 11 章)
- 分类 (第 12 章)

本书这一部分的主要目的是说明如何使用本书第一部分中介绍的数学概念来设计机器学习算法,这些算法可用于解决四大支柱职权范围内的任务。我们无意介绍高级的机器学习概念,而是提供一套实用的方法,让读者能够应用他们从本书第一部分获得的知识。它还为已经熟悉数学的读者提供了通往更广泛的机器学习文献的途径。

8.1 数据、模型和学习

在这一点上值得停下来考虑一下机器学习算法旨在解决的问题。如第 1 章所述,机器学习系统包含三个主要组件:数据、模型和学习。机器学习的主要问题是“好的模型是什么意思?”。模型这个词有很多微妙之处,我们将在本章中多次重温。如何客观地定义“好”这个词也不是很明显。机器学习的指导原则之一是好的模型应该在看不见的数据上表现良好。这需要我们定义一些性能指标,例如准确性或与真实情况的距离,以及找出在这些性能指标下表现出色的方法。本章涵盖一些常用的数学和统计语言的必要点点滴滴

表 8.1 来自 a 的示例数据

虚构的人力资源数据
库
那不是数字格式。

姓名	性别	学历	邮编	年龄	年薪	89563	123543	23989	138769	113888
阿迪亚		硕士	W21BG			36				
鲍勃		硕士博士			EC1A1BA	47				
克洛伊	F		BEcon	SW1A1BH	26	68	33			
大辅	理学士				SE207AT					
伊丽莎白		工商管理硕士	SE10AA							

过去常常谈论机器学习模型。通过这样做，我们简要概述了当前训练模型的最佳实践，以便生成的预测器在我们尚未看到的数据上表现良好。

正如第 1 章所述，我们在两种不同的意义上使用“机器学习算法”一词：训练和预测。我们将在本章中描述这些想法，以及在不同模型之间进行选择的想法。我们将在 8.2 节介绍经验风险最小化的框架，在 8.3 节介绍最大似然原理，在 8.4 节介绍概率模型的思想。我们在第 8.5 节简要概述了一种用于指定概率模型的图形语言，最后在第 8.6 节讨论了模型选择。本节的其余部分将扩展机器学习的三个主要组成部分：数据、模型和学习。

假设数据为
采用整洁的格
式（Wickham,2014
年；Codd,1990 年）。

8.1.1 数据作为向量

我们假设我们的数据可以被计算机读取，并以数字格式充分表示。假设数据是表格形式的（图 8.1），我们认为表格的每一行代表一个特定的实例或示例，每一列代表一个特定的特征。近年来，机器学习已应用于许多不明显以表格数字格式出现的数据类型，例如基因组序列、网页的文本和图像内容以及社交媒体图表。

我们不讨论识别良好特征的重要和具有挑战性的方面。其中许多方面取决于领域专业知识，需要精心设计，近年来，它们已被归入数据科学的范畴（Stray,2016 年；Adhikari 和 DeNero,2018 年）。

即使我们有表格格式的数据，仍然需要做出选择以获得数字表示。例如，在表 8.1 中，性别列（分类变量）可以转换为数字 0 表示“男性”，1 表示“女性”。或者，性别可以分别用数字 -1、+1 表示（如表 8.2 所示）。此外，在构建表示时使用领域知识通常很重要，例如知道大学学位从学士到硕士学位，或者意识到提供的邮政编码不仅仅是一串字符，而是实际上对伦敦的一个区域进行编码。在表 8.2 中，我们将表 8.1 中的数据转换为数字格式，每个邮政编码表示为两个数字，

性别身份学位		纬度 (以 度为单位)	经度 (以 度为单位)	年龄	年薪 (千)	89.563	123.543
				51.5073	23.989	138.769	
-1	2	0.1290	51.5074	0.1275	36	113.888	
-1	3	51.5071	0.1278	51.5075	47		
+1	1	0.1281	51.5074	0.1278	26		
-1	1				68		
+1	2				33		

表 8.2 来自 a 的示例数据
虚构的人力资源数据
库
(见表 8.1), 转换为
数字格式。

纬度和经度。即使是可能直接读入机器学习算法的数值数据,也应该仔细考虑单位、缩放和约束。在没有额外信息的情况下,应该移动和缩放数据集的所有列,使它们的经验平均值为0,经验方差为1。为了本书的目的,我们假设领域专家已经适当地转换了数据,即,每个输入 x_n 是实数的D维向量,称为特征、属性或协变量。我们认为数据集的特征形式如表 8.2 所示。请注意,我们在新的数字表示中删除了表 8.1 的名称列。这有两个主要原因:

(1) 我们不希望标识符(名称)为机器学习任务提供信息; (2) 我们可能希望将数据匿名化以帮助保护员工的隐私。

属性
协变量

在本书的这一部分中,我们将使用N来表示数据集中示例的数量,并使用小写字母 $n = 1, \dots, N$ 对示例进行索引。

我们假设给定了一组数值数据,表示为向量数组(表 8.2)。每行都是一个特定的个体 x_n ,在机器学习中通常被称为示例或数据点。下标示例n表示这是数据集中总共N个数据点示例中的第n个示例。每列代表一个关于示例的特定特征,我们将这些特征索引为 $d = 1, \dots, D$ 。回想一下,数据表示为向量,这意味着每个示例(每个数据点)都是一个D维向量。表的方向起源于数据库社区,但对于某些机器学习算法(例如,第 10 章),将示例表示为列向量更方便。

让我们根据表 8.2 中的数据考虑根据年龄预测年薪的问题。这称为监督学习问题,其中我们有一个标签 y_n (工资)与每个示例 x_n 标签(年龄)相关联。标签 y_n 有各种其他名称,包括目标、响应变量和注释。数据集被写成一组示例标签对 $\{(x_1, y_1), \dots, (x_n, y_n), \dots, (x_N, y_N)\}$ 。样本表 x_N 通常是串联的,写为 $X \in \mathbb{R}^{N \times D}$ 。图 8.2 说明了由表 8.2 最右边的两列组成的数据集,其中 $x = \text{age}$ 和 $y = \text{salary}$ 。

我们使用本书第一部分介绍的概念来形式化

图 8.2 用于线性回归的玩具

数据。

从最右边的两列中以

(xn, yn) 对的形式训练数

据

表 8.2。我们感兴趣的

是

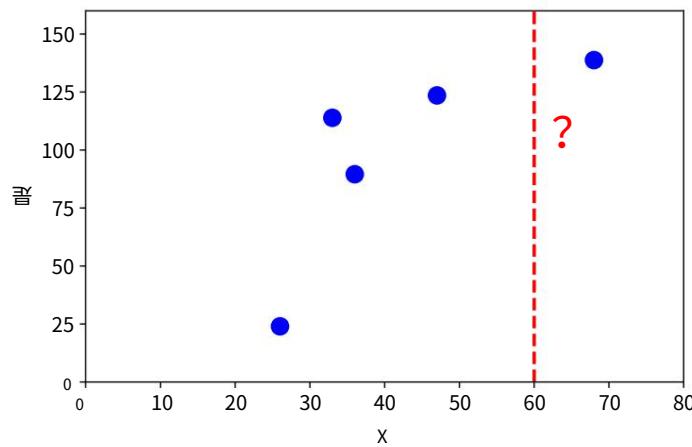
一个六十岁的人的工

资 ($x = 60$)

说明为

垂直的红色虚线,它不是

训练数据的一部分。



机器学习问题,例如上一段中的问题。

将数据表示为向量 x_n 允许我们使用线性代数中的概念 (在第 2 章中介绍)。在许多机器学习算法中,我们还需要能够比较两个向量。正如我们将在第 9 章和第 12 章中看到的那样,计算两个示例之间的相似性或距离使我们能够形式化直觉,即具有相似特征的示例应该具有相似的标签。两个向量的比较需要我们构造一个几何图形 (在第 3 章中解释),并允许我们使用第 7 章中的技术优化产生的学习问题。

由于我们有数据的矢量表示,我们可以操纵数据以找到可能更好的数据表示。我们将讨论以两种方式找到好的表示:找到原始特征向量的低维近似值,以及使用原始特征向量的非线性高维组合。在第 10 章中,我们将看到一个通过查找主成分来查找原始数据空间的低维近似值的示例。寻找主成分与第 4 章介绍的特征值和奇异值分解的概念密切相关。对于高维表示,我们将看到一个显式特征映射-维度表示 (x_n)。高维表示的主要动机是我们可以将新特征构造为原始特征的非线性组合,这反过来可能会使学习问题更容易。我们将在 9.2 节中讨论特征图,并在 12.4 节中展示这个特征图如何产生内核。近年来,深度学习方法 (Goodfellow et al., 2016) 在利用数据本身学习新的良好特征方面显示出前景,并在计算机视觉、语音识别和自然语言处理等领域取得了巨大成功。我们不会在本书的这一部分介绍神经网络,但读者可以参考

特征图

核心

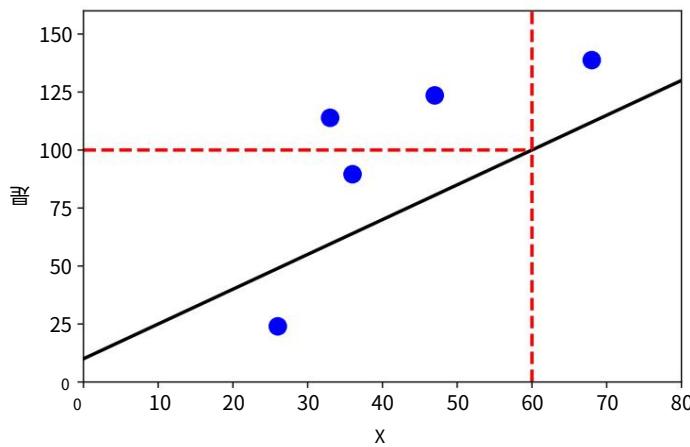


图 8.1示例函数 (黑色实心斜线)及其在 $x = 60$ 处的预测,即 $f(60) = 100$ 。

5.6 节对反向传播进行了数学描述,这是训练神经网络的一个关键概念。

8.1.2 模型作为函数

一旦我们以适当的向量表示形式获得数据,我们就可以开始构建预测函数 (称为预测器)的业务。预测器在第 1 章中,我们还没有关于模型的精确语言。

使用本书第一部分的概念,我们现在可以介绍“模型”的含义。我们在本书中介绍了两种主要方法:作为函数的预测器和作为概率模型的预测器。我们在这里描述前者,在下一节中描述后者。

预测器是一个函数,当给定一个特定的输入示例 (在我们的例子中是一个特征向量)时,它会产生一个输出。现在,将输出视为单个数字,即实数值标量输出。这可以写成

$$f: \mathbb{R}^D \rightarrow \mathbb{R}, \quad (8.1)$$

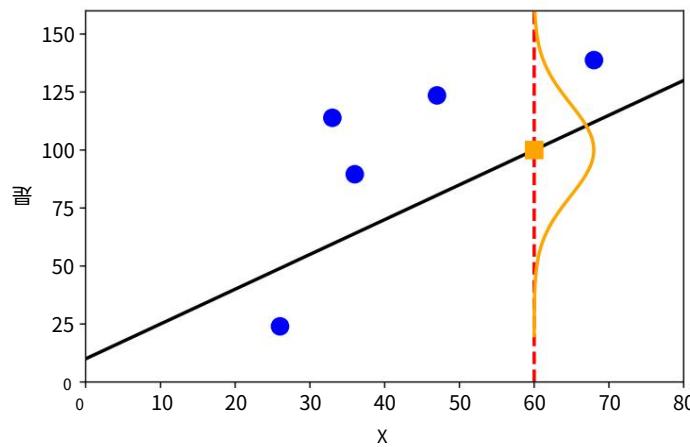
其中输入向量 x 是 D 维的 (具有 D 个特征),然后应用于它的函数 f (写为 $f(x)$) 返回一个实数。图 8.1 说明了一个可能的函数,可用于计算输入值 x 的预测值。

在本书中,我们不考虑所有功能的一般情况,这将涉及功能分析的需要。相反,我们考虑线性函数的特殊情况

$$f(x) = \theta_0 + \theta_1 x \quad (8.2)$$

对于未知的 θ 和 θ_0 。此限制意味着第 2 章和第 3 章的内容足以准确说明非概率预测器的概念 (与接下来描述的概率观点相反)

图 8.1 示例函数（黑色实心对角线）及其在 $x = 60$ 处的预测不确定性（绘制为高斯分布）。



机器学习的观点。线性函数在可以解决的问题的普遍性和所需的背景数学量之间取得了很好的平衡。

8.1.3 作为概率分布的模型

我们通常认为数据是对某些真实潜在影响的嘈杂观察，并希望通过应用机器学习，我们可以从噪声中识别出信号。这需要我们有一种语言来量化噪声的影响。我们通常还希望有表达某种不确定性的预测变量，例如，量化我们对特定测试数据点的预测值的置信度。正如我们在第 6 章中看到的，概率论提供了一种量化不确定性的语言。图 8.1 说明了作为高斯分布的函数的预测不确定性。

我们可以将预测变量视为概率模型，即描述可能函数分布的模型，而不是将预测变量视为单个函数。我们在本书中将自己限制在具有有限维参数的分布的特殊情况下，这使我们能够在不需要随机过程和随机测量的情况下描述概率模型。对于这种特殊情况，我们可以将概率模型视为多元概率分布，它已经允许丰富的模型类别。

我们将在第 8.4 节中介绍如何使用概率（第 6 章）中的概念来定义机器学习模型，并在第 8.5 节中介绍一种以紧凑的方式描述概率模型的图形语言。

8.1.4 学习就是寻找参数

学习的目标是找到一个模型及其相应的参数,以便生成的预测器在未见数据上表现良好。

在讨论机器学习算法时,在概念上分为三个不同的算法阶段:

1. 预测或推断
2. 训练或参数估计
3. 超参数调整或模型选择

预测阶段是我们对以前未见过的测试数据使用训练有素的预测器。换句话说,参数和模型选择已经固定,预测器应用于表示新输入数据点的新向量。正如第 1 章和上一小节所述,我们将在本书中考虑两种机器学习流派,对应于预测变量是函数还是概率模型。当我们有一个概率模型(在第 8.4 节中进一步讨论)时,预测阶段称为推理。

评论。不幸的是,没有就不同算法阶段的命名达成一致。“推理”一词有时也用于表示概率模型的参数估计,较少也用于表示非概率模型的预测。 \diamond 训练或参数估计阶段是我们根据训练数据调整预测模型的时候。我们希望找到给定训练数据的良好预测器,这样做有两种主要策略:根据某种质量度量(有时称为找到点估计)找到最佳预测器,或使用贝叶斯推理。找到点估计可以应用于两种类型的预测变量,但贝叶斯推理需要概率模型。

对于非概率模型,我们遵循经验风险最小化原则

,我们在第 8.2 节中对此进行了描述。经验风险最小化直接提供了寻找好的参数的优化问题。对于统计模型,最大似然原理是使用最大似然来找到一组好的参数(第 8.3 节)。我们还可以使用概率模型对参数的不确定性进行建模,我们将在第 8.4 节中详细介绍。

最小化

我们使用数值方法来找到“适合”数据的良好参数,并且大多数训练方法可以被认为是爬山方法来找到目标的最大值,例如可能性的最大值。为了应用爬山方法,我们使用第 5 章约定中描述的梯度,并实施 Chap 优化的数值优化方法是为了最小化目标。之三 7。

因此,通常如第 1 章所述,我们有兴趣学习一个基于数据额外减号的模型,以便它在未来的数据上表现良好。这对于机器学习目标来说是不够的。

交叉验证

模型只适合训练数据,预测器需要在看不见的数据上表现良好。我们使用交叉验证(第 8.2.4 节)模拟预测器对未来未见数据的行为。正如我们将在本章中看到的那样,为了实现在未见过的数据上表现良好的目标,我们需要在很好地拟合训练数据和找到对现象的“简单”解释之间取得平衡。这种权衡是通过使用正则化(第 8.2.3 节)或通过添加先验(第 8.3.2 节)来实现的。在哲学上,这被认为既不是归纳也不是演绎,而是被称为溯因。根据斯坦福哲学百科全书,溯因推理是对最佳解释进行推理的过程(Douven, A good movie title is 2017)。

绑架

“人工智能绑架”。

超参数

我们经常需要对预测变量的结构做出高级建模决策,例如要使用的组件数量或要考虑的概率分布类别。组件数量的选择是超参数的一个例子,这种选择会显着影响模型的性能。在不同模型之间进行选择的问题称为模型选择,我们在第 8.6 节中对此进行了描述。对于非概率模型,模型选择通常使用嵌套交叉验证完成,如第 8.6.1 节所述。我们还使用模型选择来选择模型的超参数。

选型

嵌套交
叉验证

评论。参数和超参数之间的区别有些随意,主要是由可以进行数值优化的内容与需要使用搜索技术的内容之间的区别所驱动。

考虑区别的另一种方法是将参数视为概率模型的显式参数,并将超参数(更高级别的参数)视为控制这些显式参数分布的参数。◊在接下来的几节中,我们将研究机器学习的三种风格:经验风险最小化(第 8.2 节)、最大似然原理(第 8.3 节)和概率建模(第 8.4 节)。

8.2 经验风险最小化

在掌握了所有的数学知识之后,我们现在可以介绍学习的意义了。机器学习的“学习”部分归结为根据训练数据估计参数。

在本节中,我们考虑作为函数的预测变量的情况,并在 8.3 节中考虑概率模型的情况。我们描述了经验风险最小化的想法,它最初是通过支持向量机的提议(在第 12 章中描述)而普及的。

然而,它的一般原则是广泛适用的,并且允许我们在不明确构建概率模型的情况下提出学习什么的问题。有四种主要的设计选择,我们将在以下小节中详细介绍:

第 8.2.1 节我们允许预测器采用的函数集是什么?

第 8.2.2 节我们如何衡量预测器在

训练数据?

第 8.2.3 节我们如何仅根据训练数据构建预测变量

在看不见的测试数据上表现良好?

第 8.2.4 节搜索 mod 空间的过程是什么

埃尔斯?

8.2.1 函数的假设类 假设我们给定了N 个例子

$x_n \in RD$ 和相应的标量标签 $y_n \in R$ 。我们考虑监督学习设置,其中我们获得对 $(x_1, y_1), \dots, (x_N, y_N)$ 。给定这些数据,我们想估计一个预测变量 $f(\cdot, \theta) : RD \rightarrow R$,由 θ 参数化。我们希望能够找到一个好的参数 θ

* 这样我们就可以很好地拟合数据,也就是说,

$$f(x_n, \theta^*) \approx y_n \text{ 对于所有 } n = 1, \dots, N. \quad (8.3)$$

在本节中,我们使用预测变量的符号 $y^* = f(x_n, \theta)$ 。 * 来表示输出

评论。为了便于演示,我们将根据监督学习(我们有标签)来描述经验风险最小化。这简化了假设类和损失函数的定义。在机器学习中选择参数化函数类也很常见,例如仿射函数。

◇

示例 8.1 我们引

入普通最小二乘回归问题来说明经验风险最小化。第 9 章给出了更全面的回归说明。当标签 y_n 是实值时,预测变量的函数类的流行选择是仿射函数集。我们通过连接附加的 $(0) x(2)$ 单元特征 x 选择仿射函数是仿射函数的更紧凑的符号

在机器学习中通常被称为线性函数。

= 1 到 x_n , 即 $x_n = [1, x(1), \dots, x(D)]$ 。参数向量对应为 $\theta = [\theta_0, \theta_1, \theta_2, \dots, \theta_D]$, 允许我们将预测变量写成线性函数

$$f(x_n, \theta) = \theta_0 + \theta_1 x(1) + \dots + \theta_D x(D). \quad (8.4)$$

此线性预测器等效于仿射模型

$$f(x_n, \theta) = \theta_0 + \sum_{d=1}^D \theta_d x(d). \quad (8.5)$$

预测器将表示单个示例 x_n 的特征向量作为输入,并产生实值输出,即 $f : RD+1 \rightarrow R$ 。

本章前面的图有一条直线作为预测变量,这意味着我们假设了一个仿射函数。

我们可能希望将非线性函数视为预测变量,而不是线性函数。神经网络的最新进展允许更复杂的非线性函数类的有效计算。

给定函数类别,我们想要搜索一个好的预测器。

我们现在继续讨论经验风险最小化的第二个要素:如何衡量预测变量与训练数据的拟合程度。

8.2.2 训练损失函数

考虑一个特定示例的标签 y_n ;以及我们基于 x_n 做出的相应预测 \hat{y}_n 。为了很好地拟合数据意味着什么,我们需要指定一个损失函数 $\ell(y_n, \hat{y}_n)$,它将地面真值标签和预测作为输入并产生一个非负数(称为损失)表示我们在这个特定预测上犯了多少错误。我们找到一个好的参数向量 θ 的目标是最小化N个训练示例集的平均损失。

损失函数

“错误”这个词
经常被使用
意味着损失。

独立同分布

* 机器学习中通常做出的一个假设是示例集 $(x_1, y_1), \dots, (x_N, y_N)$ 是独立同分布的。独立一词(第6.4.5节)表示两个数据点 (x_i, y_i) 和 (x_j, y_j) 在统计上不相互依赖,这意味着经验平均值是对总体平均值的良好估计(第6.4节).1.这意味着我们可以使用训练数据损失的经验平均值。对于给定的训练集 $\{(x_1, y_1), \dots, (x_N, y_N)\}$,我们引入示例矩阵 $X := [x_1, \dots, x_N]$

训练集

\in

$R^{N \times D}$ 和标签向量 $y := [y_1, \dots, y_N] \in R^N$ 。使用此矩阵表示法,平均损失由下式给出

$$R_{\text{emp}}(f, X, y) = \frac{1}{N} \sum_{n=1}^N \ell(y_n, \hat{y}_n), \quad (8.6)$$

经验风险

其中 $\hat{y}_n = f(x_n, \theta)$ 。等式(8.6)称为经验风险,取决于三个参数,即预测变量 f 和数据 X 、 y 。这种一般的学习策略称为经验风险最小化。

经验风险最小化

例 8.2 (最小二乘损失)

继续最小二乘回归的例子,我们指定我们使用平方损失 $\ell(y_n, \hat{y}_n) = (y_n - \hat{y}_n)^2$ 来衡量训练期间犯错误的成本

²⁸. 我们希望最小化经验风险(8.6),

这是数据损失的平均值

$$\frac{1}{N} \sum_{n=1}^N \ell(y_n - f(x_n, \theta)), \quad (8.7)$$

我们将预测变量替换为 $y_n = f(x_n, \theta)$ 。通过使用我们选择的线性预测器 $f(x_n, \theta) = \theta^\top x_n$, 我们得到优化问题

$$\min_{\theta \in \mathbb{R}^D} \frac{1}{N} \sum_{n=1}^N (y_n - \theta^\top x_n)^2. \quad (8.8)$$

该方程可以等效地表示为矩阵形式

$$\min_{\theta \in \mathbb{R}^D} \frac{1}{N} \|y - X\theta\|^2. \quad (8.9)$$

这被称为最小二乘问题。通过求解正规方程, 存在一个封闭形式的最小二乘解, 我们将在第 9.2 节中讨论。

我们对仅在训练数据上表现良好的预测器不感兴趣。相反, 我们寻找一个在看不见的测试数据上表现良好 (风险低) 的预测器。更正式地说, 我们有兴趣找到一个最小化预期风险的预测变量 f (参数固定)

预期风险

$$R_{\text{true}}(f) = \mathbb{E}_{x,y}[\ell(y, f(x))], \quad (8.10)$$

其中 y 是标签, $f(x)$ 是基于示例 x 的预测。

符号 $R_{\text{true}}(f)$ 表示如果我们可以访问无限量的数据, 这就是真正的风险。期望超过所有的 (无限) 集合另一个常用于可能的数据和标签的短语。预期风险会引起两个实际问题, 即我们希望将预期风险降至最低, 我们将在以下“总体风险”中解决这个问题。两个小节:

- 我们应该如何改变我们的训练程序才能很好地泛化?
- 我们如何根据 (有限) 数据估计预期风险?

评论。许多机器学习任务都指定了相关的性能度量, 例如预测的准确性或均方根误差。性能测量可能更复杂, 对成本敏感, 并捕获有关特定应用程序的详细信息。原则上, 用于经验风险最小化的损失函数的设计应直接对应于机器学习任务指定的性能度量。在实践中, 损失函数的设计和性能指标之间经常存在不匹配。这可能是由于诸如易于实施或优化效率等问题。 ◇

8.2.3 减少过度拟合的正则化

本节描述了经验风险最小化的补充,使其能够很好地概括(近似最小化预期风险)。回想一下,训练机器学习预测器的目的是让我们能够在未见过的数据上表现良好,即预测器能够很好地泛化。我们通过保留整个数据集的一部分来模拟这些看不见的数据。

测试集
即使只知道预测器在
测试集泄漏上的表现

信息(Blum 和
Hardt,2015 年)。

该保持集称为测试集。为预测变量 f 提供足够丰富的函数类别,我们基本上可以记住训练数据以获得零经验风险。虽然这对于最小化训练数据的损失(以及风险)非常有用,但我们不希望预测器能够很好地泛化到看不见的数据。实际上,我们只有一组有限的数据,因此我们将数据分成训练集和测试集。训练集用于拟合模型,测试集(机器学习算法在训练期间看不到)用于评估泛化性能。重要的是,用户在观察完测试集后不要循环回到新一轮的训练。我们使用下标 train 和 test 分别表示训练集和测试集。

过拟合

事实证明,经验风险最小化会导致过度拟合,即预测器与训练数据的拟合过于紧密,并且不能很好地泛化到新数据(Mitchell,1997)。这种在训练集上平均损失很小而在测试集上平均损失很大的普遍现象往往发生在我们的数据很少和假设类很复杂的情况下。对于特定的预测器 f (参数固定),当来自训练数据 $\text{Remp}(f, X_{\text{train}}, y_{\text{train}})$ 的风险估计低估了预期风险 $\text{Rtrue}(f)$ 时,就会出现过拟合现象。

正则化

由于我们通过使用测试集 $\text{Remp}(f, X_{\text{test}}, y_{\text{test}})$ 上的经验风险来估计预期风险 $\text{Rtrue}(f)$,如果测试风险远大于训练风险,这表明存在过度拟合。我们在第 8.3.3 节中重新讨论过拟合的概念。

因此,我们需要通过引入惩罚项以某种方式偏向于寻找经验风险的最小值,这使得优化器更难返回过于灵活的预测器。在机器学习中,惩罚项被称为正则化。正则化是一种在经验风险最小化的准确解决方案与解决方案的大小或复杂性之间进行折衷的方法。

示例 8.3(正则化最小二乘法)

正则化是一种阻止优化问题的复杂或极端解决方案的方法。最简单的正则化策略是

取代最小二乘问题

$$\text{分钟} \frac{1}{\theta} \|y - X\theta\|^2. \quad (8.11)$$

在前面的“正则化”问题示例中,添加了一个仅涉及 θ 的惩罚项:

$$\text{分钟} \frac{1}{\theta} \|y - X\theta\|^2 + \lambda \|\theta\|^2. \quad (8.12)$$

附加项 $\|\theta\|^2$ 是正则化参数。称为正则化器,参数正则化器数。正则化参数权衡正则化以最小化训练集的损失和参数 θ 的大小。如果我们遇到过度拟合,通常会范围发生参数值的大小变得相对较大的情况(Bishop,2006)。

正则化项有时称为惩罚项,即双惩罚项使向量 θ 更接近原点。正则化的思想也作为参数的先验概率出现在概率模型中。

回忆一下6.6节,为了使后验分布与先验分布具有相同的形式,先验和似然需要共轭。我们将在第8.3.2节中重新讨论这个想法。我们将在第12章中看到,正则化器的思想等同于大间距的思想。

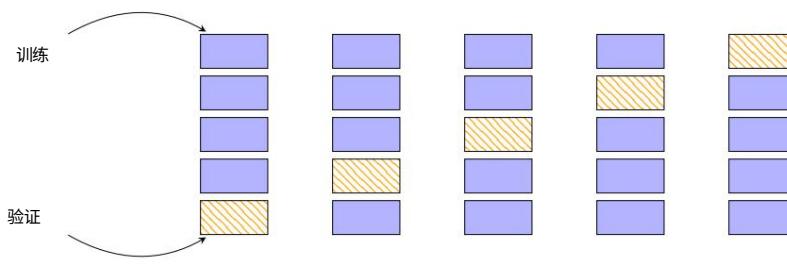
8.2.4 评估泛化性能的交叉验证我们在上一节中提到,我们通过在测试数据上应用预测器来估计泛化误差来衡量泛化误差。此数据有时也称为验证集。验证集是我们保留的可用训练数据的子验证集。这种方法的一个实际问题是数据量有限,理想情况下我们会使用尽可能多的可用数据来训练模型。这将要求我们保持我们的验证集 V 较小,这将导致预测性能的噪声估计(具有高方差)。这些相互矛盾的目标(大型训练集、大型验证集)的一种解决方案是使用交叉验证。 K 折交叉验证有效地将交叉验证数据划分为 K 个块,其中 $K-1$ 个构成训练集 R ,最后一个块用作验证集 V (类似于之前概述的想法)。交叉验证迭代(理想情况下)块分配给 R 和 V 的所有组合;参见图8.2。对验证集的所有 K 个选择重复此过程,并对 K 次运行的模型性能进行平均。

我们将我们的数据集分成两组 $D = R \cup V$,这样它们就不会重叠($R \cap V = \emptyset$),其中 V 是验证集,并在 R 上训练我们的模型。训练后,我们评估上的预测变量 f

图 8.2 K 折交叉验证。

数据集分为 $K = 5$ 块,其中 $K - 1$ 个用作

训练集 (蓝色) 和一
个作为
验证集 (橙色影
线)。



验证集 V (例如,通过计算训练模型在验证集上的均方根误差 (RMSE))。更准确地说,对于每个分区 k ,训练数据 $R(k)$ 生成一个预测变量 $f(k)$,然后将其应用于验证集 $V(k)$ 以计算经验风险 $R(f(k))$,

$, V(k)$)。我们循环遍历所有可能的验证集和训练集分区,并计算预测器的平均泛化误差。交叉验证近似于预期的泛化误差

$$EV[R(f, V)] \approx \frac{1}{K} \sum_{k=1}^K R(f^{(k)}, V(k)), \quad (8.13)$$

其中 $R(f(k), V(k))$ 是验证集 $V(k)$ 上的风险 (例如, RMSE) 测变量 $f(k)$ 近似值有两个来源:首先,由于有限的训练集,这导致不是最好的 $f(k)$;其次,由于验证集有限,导致对风险 $R(f(k))$ 的估计不准确

$, V(k)$)。K 折交叉验证的一个潜在缺点是训练模型 K 次的计算成本,如果训练成本在计算上很昂贵,这可能会很麻烦。在实践中,仅查看直接参数通常是不够的。例如,我们需要探索多个复杂性参数 (例如,多个正则化参数),这些参数可能不是模型的直接参数。

根据这些超参数评估模型的质量,可能会导致模型参数数量呈指数增长的训练运行次数。可以使用嵌套交叉验证 (第 8.6.1 节) 来搜索好的超参数。

尴尬地平行

然而,交叉验证是一个令人尴尬的并行问题,即只需很少的努力就可以将问题分解为多个并行任务。给定足够的计算资源 (例如,云计算、服务器场),交叉验证不需要比单个性能更长的时间

评估。

在本节中,我们看到经验风险最小化基于以下概念:函数的假设类、损失函数和正则化。在 8.3 节中,我们将看到使用概率分布代替损失函数和正则化思想的效果。

8.2.5 延伸阅读

由于实证风险最小化的最初发展 (Vapnik,1998)是用浓重的理论语言表达的,因此许多后续发展都是理论性的。研究领域称为统计学习理论 (Vapnik,1999 年;Evgeniou 等人,2000 年;统计学习Hastie 等人,2001 年;von Luxburg 和 Scholkopf “ 2011) 。最近一本建立在理论基础上并开发高效学习算法的机器学习教科书是 Shalev-Shwartz 和 Ben-David (2014)。

,

正则化的概念起源于诗句不适用问题的解决方案 (Neumaier, 1998)。这里介绍的方法称为Tikhonov 正则化,还有一个密切相关的约束版本Tikhonov 称为 Ivanov 正则化。Tikhonov 正则化与偏差-方差权衡和特征选择有很深的关系正则化 (Buhlmann 和 Van De Geer, 2011) 。交叉验证的替代方法是 bootstrap 和 jackknife (Efron 和 Tibshirani,1993 年;Davidson 和 Hinkley,1997 年;Hall,1992 年) 。

将经验风险最小化 (第 8.2 节)视为“无概率”是不正确的。存在控制数据生成的潜在未知概率分布 $p(x, y)$ 。但是,经验风险最小化方法与该分布选择无关。

这与明确要求了解 $p(x, y)$ 的标准统计方法形成对比。此外,由于分布是样本 x 和标签 y 的联合分布,因此标签可能是不确定的。与标准统计相比,我们不需要指定标签 y 的噪声分布。

8.3 参数估计

在 8.2 节中,我们没有使用概率分布明确地对我们的问题建模。在本节中,我们将看到如何使用概率分布来模拟由于观察过程和预测变量参数的不确定性引起的不确定性。在第 8.3.1 节中,我们引入了似然,它类似于经验风险最小化中损失函数的概念 (第 8.2.2 节)。先验的概念 (第 8.3.2 节)类似于正则化的概念 (第 8.2.3 节)。

8.3.1 最大似然估计

最大似然估计(MLE)背后的想法是定义参数的最大似然函数,使我们能够找到与数据很好地拟合的模型。估计问题集中在似然函数上,或者更准确地说是似然函数的负对数。对于由随机变量 x 表示的数据和由 θ 参数化的概率密度族 $p(x | \theta)$,负对数似然由下式给出

负对数似然

$$L(x|\theta) = -\log p(x|\theta). \quad (8.14)$$

符号 $L(x|\theta)$ 强调参数 θ 是变化的而数据 x 是固定的。我们经常在写负对数似然时省略对 x 的引用,因为它实际上是 θ 的函数,当表示数据不确定性的随机变量从上下文中清楚时,我们将其写为 $L(\theta)$ 。

让我们解释一下概率密度 $p(x|\theta)$ 是针对固定值 θ 建模的。它是对给定参数设置的数据不确定性建模的分布。对于给定的数据集 x ,似然允许我们表达对参数 θ 不同设置的偏好,我们可以选择更“可能”生成数据的设置。

从互补的角度来看,如果我们认为数据是固定的(因为它已被观察到),并且我们改变参数 θ ,那么 $L(\theta)$ 告诉我们什么?它告诉我们 θ 的特定设置对于观察值 x 的可能性有多大。

基于第二种观点,最大似然估计器为我们提供了数据集最可能的参数 θ 。

我们考虑监督学习设置,其中我们获得对 $(x_1, y_1), \dots, (x_N, y_N)$ with $x_n \in \mathcal{R}^D$ 和标签 $y_n \in \mathcal{R}$ 对于 x_n ,我们想要标签 y_n 的概率分布。换句话说,我们在给定特定参数设置 θ 的示例的情况下指定标签的条件概率分布。

示例 8.4 经常使

用的第一个示例是指定给定示例的标签的条件概率是高斯分布。换句话说,我们假设我们可以通过具有零均值的独立高斯噪声(参见第 6.5 节) $\epsilon_n \sim N(0, \sigma^2)$ 来解释我们的观测不确定性。我们进一步假设线性模型 x 预测。这意味着我们为每个示例标签对 (x_n, y_n) 指定高斯似然,

θ 用于
 n

$$p(y_n | x_n, \theta) = N(y_n | x_n, \theta, \sigma^2). \quad (8.15)$$

图 8.1 显示了给定参数 θ 的高斯似然图。我们将在第 9.2 节中看到如何根据高斯分布显式扩展前面的表达式。

独立同分布

我们假设样本集 $(x_1, y_1), \dots, (x_N, y_N)$ 是独立的同分布(iid)。“独立”一词(第 6.4.5 节)意味着涉及整个数据集 ($Y = \{y_1, \dots, y_N\}$) 和 $X = \{x_1, \dots, x_N\}$) 的似然因式分解为的可能性

每个单独的例子

$$p(Y | X, \theta) = \prod_{n=1}^N p(y_n | x_n, \theta), \quad (8.16)$$

其中 $p(y_n | x_n, \theta)$ 是特定分布（在例 8.4 中是高斯分布）。 “同分布”是指乘积 (8.16) 中的每一项都服从相同的分布，并且它们都共享相同的参数。从优化的角度来看，计算可以分解为更简单函数之和的函数通常更容易。

因此，在机器学习中我们经常考虑负对数似然 $\log(ab) = \log(a) + \log(b)$

$$L(\theta) = -\sum_{n=1}^N \log p(y_n | x_n, \theta). \quad (8.17)$$

虽然很容易解释 θ 在 $p(y_n | x_n, \theta)$ (8.15) 中条件的右边，因此应该被解释为观察到的和固定的，但这种解释是不正确的。负对数似然 $L(\theta)$ 是 θ 的函数。因此，要找到一个很好的参数向量 θ 来解释数据 $(x_1, y_1), \dots, (x_N, y_N)$ 那么，最小化关于 θ 的负对数似然 $L(\theta)$ 。

评论。(8.17) 中的负号是一个历史产物，这是由于我们想要最大化似然的约定，但数值优化文献倾向于研究函数的最小化。 ◇

示例 8.5 继续我

们的高斯似然示例 (8.15)，负对数似然可以重写为

$$L(\theta) = -\sum_{n=1}^N \log p(y_n | x_n, \theta) = -\sum_{n=1}^N \text{记录 } N y_n | x_n, \theta, \sigma^2 \quad (8.18a)$$

$$= -\sum_{n=1}^N \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp -\frac{(y_n - x_n \theta)^2}{2\sigma^2} \quad (8.18b)$$

$$= -\sum_{n=1}^N \log \exp -\frac{(y_n - x_n \theta)^2}{2\sigma^2} - \sum_{n=1}^N \log \frac{1}{\sqrt{2\pi\sigma^2}} \quad (8.18c)$$

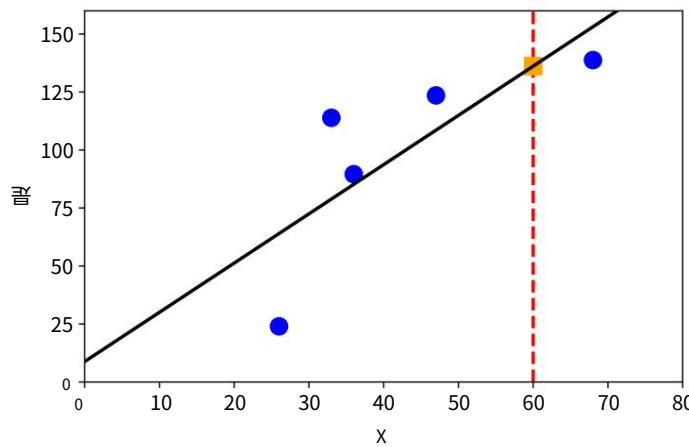
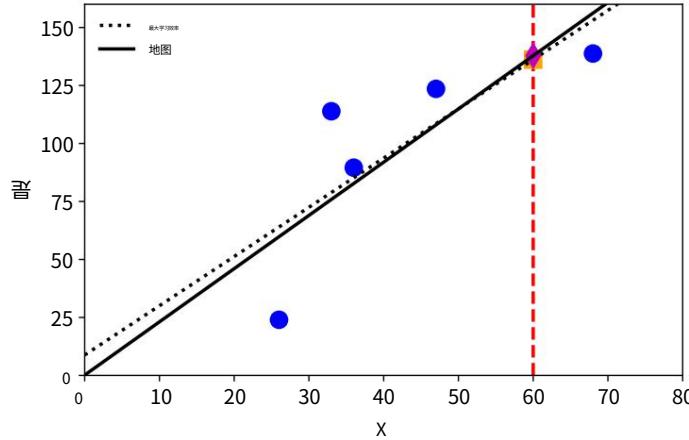
$$= \frac{1}{2\sigma^2} \sum_{n=1}^N (y_n - x_n \theta)^2 - \sum_{n=1}^N \log \frac{1}{\sqrt{2\pi\sigma^2}}. \quad (8.18d)$$

由于 σ 给定，(8.18d) 中的第二项是常数，最小化 $L(\theta)$ 对应于解决第一项表示的最小二乘问题（与 (8.8) 比较）。

事实证明，对于高斯似然，得到的优化

图 8.2 对于给定的数据, 最大似然估计

参数结果为黑色

对角线。这
橙色方块显示最大
值可能性
预测于
 $x = 60$ 。图 8.1 比较预
测与最大似然估计和MAP 估计在
 $x = 60$ 。先验使斜率变得
不那么陡峭, 截距更接近
于零。在此示例中, 使截距更
接近零的偏差实际
上增加了斜率。

最大似然估计对应的问题有一个封闭形式的解决方案。我们将在第 9 章中看到更多详细信息。图 8.2 显示了回归数据集和由最大似然参数导出的函数。最大似然估计可能会过度拟合（第 8.3.3 节），类似于非正则化经验风险最小化（第 8.2.3 节）。对于其他似然函数，即如果我们用非高斯分布对噪声建模，最大似然估计可能没有封闭形式的解析解。在这种情况下，我们求助于第 7 章中讨论的数值优化方法。

8.3.2 最大后验估计

如果我们有关于参数 θ 分布的先验知识，我们可以将一个附加项乘以似然。这个附加项是参数 $p(\theta)$ 的先验概率分布。对于给定的先验，后

观察一些数据 x , 我们应该如何更新 θ 的分布? 换句话说, 我们应该如何表示我们在观察数据 x 后对 θ 有了更具体的了解这一事实? 如第 6.3 节所述, 贝叶斯定理为我们提供了更新随机变量概率分布的原则性工具。它允许我们从一般先验陈述 (先验分布) $p(\theta)$ 和先验链接函数 $p(x | \theta)$ 的参数 θ 上计算后验分布后验 $p(\theta | x)$ (更具体的知识) 参数 θ 和观测数据 x (称为似然) :

$$| x) = p(x) \frac{p(x | \theta)p(\theta)}{p(\theta | x)}. \quad (8.19)$$

回想一下, 我们有兴趣找到使后验最大化的参数 θ 。由于分布 $p(x)$ 不依赖于 θ , 我们可以忽略优化的分母值并获得

$$p(\theta | x) \propto p(x | \theta)p(\theta). \quad (8.20)$$

前面的比例关系隐藏了数据 $p(x)$ 的密度, 这可能很难估计。我们现在估计负对数后验的最小值, 而不是估计负对数似然的最小值, 这被称为最大后验估计 (MAP 估计)。图 8.1 显示了添加零均值高斯先验的效果图。

可能性

事后的
估计
MAP 估计

示例 8.6 除了前

面示例中的高斯似然假设外, 我们还假设参数向量分布为零均值的多元高斯分布, 即 $p(\theta) = N(0, \Sigma)$ 其中 Σ 是协方差矩阵 (第 6.5 节)。请注意, 高斯分布的共轭先验也是高斯分布 (第 6.6.1 节), 因此我们期望后验分布也是高斯分布。我们将在第 9 章看到最大后验估计的细节。

包含关于良好参数所在位置的先验知识的想法在机器学习中很普遍。我们在第 8.2.3 节中看到的另一种观点是正则化的思想, 它引入了一个附加项, 使结果参数偏向于接近原点。

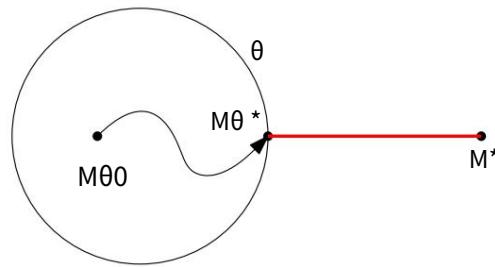
最大后验估计可以被认为是连接非概率和概率世界的桥梁, 因为它明确承认需要先验分布, 但它仍然只产生参数的点估计。

评论。最大似然估计 θ_{ML} 具有以下属性 (Lehmann 和 Casella, 1998; Efron 和 Hastie, 2016) :

- 渐近一致性: MLE 收敛于真值

图 8.1 模型拟合。在模型的参数化类 $M\theta$ 中，我们优化模型参数 θ 以最小化与真实模型的距离

(未知) 模型 M^* 。



无限多个观测值的极限,加上一个近似正态的随机误差。

- 实现这些特性所需的样本量可能非常大。
- 误差的方差以 $1/N$ 衰减,其中 N 是数据点的数量。
- 特别是,在“小”数据范围内,最大似然估计会导致过度拟合。



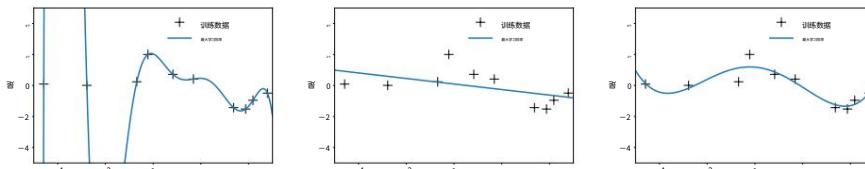
最大似然估计 (和最大后验估计) 的原理使用概率建模来推理数据和模型参数中的不确定性。然而,我们还没有充分利用概率建模。在本节中,由此产生的训练过程仍然会产生预测器的点估计,即训练返回代表最佳预测器的一组参数值。在 8.4 节中,我们将认为参数值也应被视为随机变量,而不是估计该分布的“最佳”值,我们将在进行预测时使用完整的参数分布。

8.3.3 模型拟合

考虑给定数据集的设置,我们有兴趣将参数化模型拟合到数据中。当我们谈论“拟合”时,我们通常指的是优化/学习模型参数,以便它们最小化某些损失函数,例如负对数似然。通过最大似然 (第 8.3.1 节) 和最大后验估计 (第 8.3.2 节),我们已经讨论了两种常用的模型拟合算法。

模型的参数化定义了一个我们可以操作的模型类 $M\theta$ 。例如,在线性回归设置中,我们可以将输入 x 和 (无噪声) 观察值 y 之间的关系定义为 $y = ax + b$, 其中 $\theta := \{a, b\}$ 是模型参数。在这种情况下,模型参数 θ 描述了仿射函数族,即具有斜率 a 的直线,其从 0 偏移 b 。假设数据来了

8.3 参数估计



(a) 过拟合

(b) 欠拟合。

(c) 拟合良好。

图 8.2 将不同的模型类别（按最大似然）拟合到回归数据集。

来自模型 M^* ，这是我们不知道的。对于给定的训练数据集，其中“关闭我们优化 θ 使得 $M\theta$ 尽可能接近 M^* ”由我们优化的目标函数定义（例如，训练数据的平方损失）。图 8.1 说明了一个设置，其中我们有一个小模型类（由圆圈 $M\theta$ 表示），并且数据生成模型 M 位于所考虑模型的集合之外。我们从 $M\theta=0$ 开始我们的参数搜索。优化后，即当我们获得最佳可能参数 θ 时，我们区分三种不同的情况：(i) 过度拟合，(ii) 欠拟合，以及 (iii) 拟合良好。我们将对这三个概念的含义给出一个高层次的直觉。

*

粗略地说，过拟合是指副过拟合的情况

度量化模型类太丰富，无法对 M 生成的数据集进行建模，即 $M\theta$ 可以对更复杂的数据集进行建模。例如，如果数据集是由线性函数生成的，并且我们将 $M\theta$ 定义为七阶多项式的类，那么我们不仅可以建模线性函数，还可以建模二阶、三阶等多项式。 fit 通常有大量的参数。我们经常用一种方法来检测 make 的一个观察是，过度灵活的模型类 $M\theta$ 使用其所有建模能力来减少训练误差。如果训练数据有噪声，它会因此在噪声本身中找到一些有用的信号。当我们远离训练数据进行预测时，这将导致巨大的问题。图 8.2(a) 给出了回归上下文中过度拟合的示例，其中模型参数是通过最大似然法学习的（参见第 8.3.1 节）。

实践中的过度拟合是观察模型具有低

交叉验证期间存在训练风险但测试风险高（第 8.2.4 节）。

我们将在第 9.2.2 节中更多地讨论回归中的过度拟合。

当我们遇到欠拟合时，我们遇到了相反的问题 underfitting。其中模型类 $M\theta$ 不够丰富。例如，如果我们的数据集是由正弦函数生成的，但 θ 仅对直线进行参数化，那么最好的优化过程不会让我们接近真实模型。但是，我们仍然优化参数并找到对数据集建模的最佳直线。图 8.2(b) 显示了一个因不够灵活而欠拟合的模型示例。欠拟合的模型通常参数很少。

第三种情况是参数化模型类是正确的。

然后，我们的模型拟合得很好，即既没有过拟合也没有欠拟合。这意味着我们的模型类足够丰富来描述我们给定的数据集。

图 8.2(c) 显示了一个非常适合给定数据集的模型。理想情况下，

这是我们想要使用的模型类,因为它具有良好的泛化属性。

在实践中,我们经常会定义非常丰富的模型类 $M\theta$,参数很多,比如深度神经网络。为了减轻过度拟合的问题,我们可以使用正则化(第8.2.3节)或先验(第8.3.2节)。

我们将在8.6节讨论如何选择模型类。

8.3.4 延伸阅读

在考虑概率模型时,最大似然估计原理概括了线性模型的最小二乘回归思想,我们将在第9章中详细讨论。输出,即

$$p(y_n|x_n, \theta) = \phi(\theta - x_n), \quad (8.21)$$

我们可以考虑其他预测任务的其他模型,例如二进制分类或建模计数数据(McCullagh和Nelder,1989)。另一种观点是考虑来自指数族的可能性(第6.6节)。参数和数据之间具有线性相关性并具有潜在的非线性变换 ϕ (称为链接函数)的一类模型被称为广义线性模型(Agresti,2002年,第4章)。

链接功能
广义线性模型

最大似然估计具有悠久的历史,最初由罗纳德·费希尔爵士在1930年代提出。我们将在8.4节中扩展概率模型的概念。使用概率模型的研究人员之间的一场争论是贝叶斯统计和频率统计之间的讨论。如6.1.1节所述,归结为概率的定义。回想一下6.1节,人们可以将概率视为逻辑推理的概括(通过允许不确定性)(Cheeseman,1985年;Jaynes,2003年)。最大似然估计的方法本质上是频率论者,感兴趣的读者可以参考Efron和Hastie(2016)以平衡贝叶斯和频率论者统计的观点。

有一些概率模型可能无法进行最大似然估计。读者可以参考更高级的统计教科书,例如Casella和Berger(2002),了解矩量法、M估计和估计方程等方法。

8.4 概率建模与推理

在机器学习中,我们经常关注数据的解释和分析,例如,预测未来事件和决策制定。为了使这项任务更容易处理,我们经常构建模型来描述生成观察数据的生成过程。

生成过程

例如,我们可以分两步描述抛硬币实验(“正面”或“反面”)的结果。首先,我们定义一个参数 μ ,它描述了“正面”的概率作为伯努利分布的参数(第6章);其次,我们可以从伯努利分布 $p(x|\mu) = \text{Ber}(\mu)$ 中采样结果 $x \in \{\text{head}, \text{tail}\}$ 。参数 μ 产生一个特定的数据集 X 并取决于所使用的硬币。由于 μ 事先未知并且永远无法直接观察到,因此我们需要一种机制来了解 μ 给定的抛硬币实验的观察结果。在下文中,我们将讨论如何将概率建模用于此目的。

8.4.1 概率模型

概率模型将实验的不确定方面表示为概率分布。使用概率模型的好处是它们提供了一组统一且一致的概率论工具(第6章),用于建模、推理、预测和模型选择。

概率模型由所有随机变量的联合分布指定。

在概率建模中,观测变量 x 和隐藏参数 θ 的联合分布 $p(x, \theta)$ 至关重要:它封装了以下信息:

- 先验和可能性(产品规则,第6.3节)。
- 边际似然 $p(x)$ 将在模型选择(第8.6节)中发挥重要作用,可以通过采用联合分布和对参数进行积分来计算(求和规则,第6.3节)。
- 后验,可以通过将关节除以边缘似然获得。

只有联合分布具有此属性。因此,概率模型由其所有随机变量的联合分布指定。

8.4.2 贝叶斯推理机器学习中的一个

关键任务是采用模型和数据来揭示给定观测变量 x 的模型隐藏变量 θ 的值。在第8.3.1节中,我们已经讨论了使用最大似然估计或最大后验估计估计模型参数 θ 的两种方法。在这两种情况下,我们都获得了 θ 的单一最佳值,因此参数估计的关键算法问题是解决优化问题。一旦这些点估计 θ 已知,我们就可以使用它们进行预测。更具体地说,预测分布将是 $p(x|\theta)$

范围
估计可以表述为优化问题。

我们在哪里使用 θ^* 在似然函数中。

如第6.3节所述,仅关注后验分布的某些统计量(例如最大化后验的参数 θ^*)会导致信息丢失,这在系统中可能是至关重要的

贝叶斯推理是关于学习分布
随机变量。
贝叶斯推理

使用预测 $p(x | \theta)$ 系统通常具有与^{*}做出决定。这些决策似然、平方误差损失或错误分类误差不同的目标函数。因此，具有完整的后验分布可能非常有用，并导致更稳健的决策。贝叶斯推理是关于找到这个后验分布 (Gelman et al., 2004)。对于数据集 X 一个似然函数，后验

$, p(\theta)$ 之前的参数，以及

$$p(\theta | X) = \frac{p(X | \theta)p(\theta)}{p(X)} = \frac{p(X | \theta)p(\theta)d\theta}{p(X)} , \quad (8.22)$$

贝叶斯推理颠倒了参数和数据之间的关系。

通过应用贝叶斯定理获得。关键思想是利用贝叶斯定理反转参数 θ 和数据 X 之间的关系（由似然给出）以获得后验分布 $p(\theta | X)$ 。

参数具有后验分布的含义是它可用于将不确定性从参数传播到数据。更具体地说，对于参数的分布 $p(\theta)$ ，我们的预测将是

$$p(x) = p(x | \theta)p(\theta)d\theta = E_\theta[p(x | \theta)] , \quad (8.23)$$

并且它们不再依赖于已被边缘化/整合掉的模型参数 θ 。等式 (8.23) 表明预测是所有合理参数值 θ 的平均值，其中合理性由参数分布 $p(\theta)$ 封装。

在第 8.3 节中讨论了参数估计和此处的贝叶斯推理之后，让我们比较这两种学习方法。通过最大似然或 MAP 估计的参数估计产生参数的一致点估计 θ ，要解决的关键计算问题是优化。相反，贝叶斯推理产生（后验）分布，要解决的关键计算问题是积分。点估计的预测很简单，而贝叶斯框架中的预测需要解决另一个积分问题；见 (8.23)。然而，贝叶斯推理为我们提供了一种合并先验知识、考虑辅助信息和合并结构知识的原则性方法，所有这些在参数估计的背景下都不容易完成。此外，在数据高效学习的背景下，参数不确定性对预测的传播对于风险评估和探索的决策系统可能很有价值 (Deisenroth 等人, 2015 年; Kamthe 和 Deisenroth, 2018 年)。

虽然贝叶斯推理是一个用于学习参数和进行预测的数学原理框架，但由于我们需要解决的集成问题，它也带来了一些实际挑战；参见 (8.22) 和 (8.23)。更具体地说，如果我们不在参数上选择共轭先验（第 6.6.1 节），则 (8.22) 和 (8.23) 中的积分在分析上不易处理，我们无法计算 $p(x)$

后验、预测或封闭形式的边际似然。在这些情况下，我们需要求助于近似值。在这里，我们可以使用随机近似，例如马尔可夫链蒙特卡洛 (MCMC) (Gilks 等人,1996 年)，或确定性近似，例如拉普拉斯逼近 (Bishop,2006;Barber,2012;Murphy,2012)、变分推理 (Jordan et al., 1999; Blei et al., 2017) 或期望传播 (Minka, 2001a)。

尽管存在这些挑战，贝叶斯推理已成功应用于各种问题，包括大规模主题建模 (Hoff man 等人,2013 年)、点击率预测 (Graepel 等人,2010 年)、数据控制系统 (Deisenroth 等人,2015 年)、在线排名系统 (Herbrich 等人,2007 年) 和大规模推荐系统中的有效强化学习。有通用工具，例如贝叶斯优化 (Brochu 等人,2009 年; Snoek 等人,2012 年； Shahriari 等人,2016 年)，它们对于有效搜索模型或算法的元参数非常有用。

评论。在机器学习文献中，(随机)“变量”和“参数”之间可能存在某种程度的任意分离。在估计参数时（例如，通过最大似然），变量通常被边缘化。在本书中，我们对这种分离并没有那么严格，因为原则上我们可以对任何参数放置先验并将其积分出来，然后根据上述分离将参数变成随机变量。 ◇

8.4.3 潜在变量模型

在实践中，有时将额外的潜在变量 z 潜在变量（除了模型参数 θ ）作为模型的一部分很有用 (Moustaki 等人, 2015 年)。这些潜在变量与模型参数 θ 不同，因为它们没有明确地参数化模型。潜在变量可以描述数据生成过程，从而有助于模型的可解释性。它们还经常简化模型的结构，并允许我们定义更简单和更丰富的模型结构。模型结构的简化通常伴随着较少数量的模型参数 (Paquet, 2008; Murphy, 2012)。可以使用期望最大化 (EM) 算法 (Dempster 等人,1977 年； Bishop,2006 年) 以一种有原则的方式来学习潜在变量模型（至少通过最大似然）。此类潜在变量有用的示例包括用于降维的主成分分析（第 10 章）、用于密度估计的高斯混合模型（第 11 章）、隐马尔可夫模型 (Maybeck,1979) 或动力系统 (Ghahramani 和 Roweis, 1999 年; Ljung,1999 年) 用于时间序列建模、元学习和任务泛化 (Hausman 等人,2018 年; Sæmundsson 等人,2018 年)。虽然引入这些潜在变量

可能会使模型结构和生成过程更容易,但潜变量模型的学习通常很难,我们将在第 11 章中看到。

由于潜变量模型还允许我们定义从参数生成数据的过程,让我们来看看这个生成过程。用 x 表示数据,用 θ 表示模型参数,用 z 表示潜在变量,我们得到条件分布

$$p(x | z, \theta) \quad (8.24)$$

这使我们能够为任何模型参数和潜在变量生成数据。鉴于 z 是潜在变量,我们在它们上面放置了先验 $p(z)$ 。

正如我们之前讨论的模型,具有潜在变量的模型可用于我们在第 8.3 节和 8.4.2 节中讨论的框架内的参数学习和推理。为了促进学习(例如,通过最大似然估计或贝叶斯推理),我们遵循两步程序。首先,我们计算模型的可能性 $p(x | \theta)$,它不依赖于潜在变量。其次,我们将这种可能性用于参数估计或贝叶斯推理,其中我们分别使用与第 8.3 节和第 8.4.2 节中完全相同的表达式。

由于似然函数 $p(x | \theta)$ 是给定模型参数的数据的预测分布,我们需要边缘化潜在变量,以便

$$p(x | \theta) = p(x | z, \theta)p(z)dz, \quad (8.25)$$

其中 $p(x | z, \theta)$ 在 (8.24) 中给出, $p(z)$ 是潜在变量的先验。请注意,似然性不能依赖于潜在变量 z ,而只是数据 x 和模型参数 θ 的函数。

可能性是数据的函数

和模型
参数,但独立于潜在变
量。

(8.25) 中的似然直接允许通过最大似然进行参数估计。如第 8.3.2 节中所讨论的,MAP 估计也很简单,带有模型参数 θ 的附加先验。

此外,潜在变量模型中的似然 (8.25) 贝叶斯推理(第 8.4.2 节)以通常的方式工作:我们将先验 $p(\theta)$ 放在模型参数上并使用贝叶斯定理获得后验分布

$$p(\theta | X) = \frac{p(X | \theta)p(\theta)}{p(X)} \quad (8.26)$$

在给定数据集 X 的模型参数上。(8.26) 中的后验可用于贝叶斯推理框架内的预测;见 (8.23)。

我们在这个潜在变量模型中面临的一个挑战是,相似性 $p(X | \theta)$ 需要根据 (8.25) 对潜在变量进行边际化。除非我们为 $p(x | z, \theta)$ 选择共轭先验 $p(z)$,否则 (8.25) 中的边缘化在分析上不易处理,我们需要求助于近似值(Bishop, 2006; Paquet, 2008; Murphy, 2012 年; 穆斯塔基等人,2015 年)。

类似于参数后验 (8.26) 我们可以计算后验
根据潜在变量

$$p(z | X) = \frac{p(X | z)p(z)}{p(X | z)p(z, \theta)p(\theta)d\theta}, p(X) \quad (8.27)$$

其中 $p(z)$ 是潜在变量的先验, $p(X | z)$ 要求我们对模型参数 θ 进行积分。

考虑到解析求解积分的难度,很明显,同时对潜在变量和模型参数进行边际化通常是不可能的 (Bishop, 2006 年; Murphy, 2012 年)。一个更容易计算的量是潜在变量的后验分布,但以模型参数为条件,即

$$p(z | X, \theta) = \frac{p(X | z, \theta)p(z)}{p(X | \theta)}, \quad (8.28)$$

其中 $p(z)$ 是潜在变量的先验, $p(X | z, \theta)$ 在 (8.24) 中给出。

在第 10 章和第 11 章中,我们分别推导了 PCA 和高斯混合模型的似然函数。此外,我们计算 PCA 和高斯混合模型的潜在变量的后验分布 (8.28)。

评论。在接下来的章节中,我们可能不会在潜在变量 z 和不确定的模型参数 θ 之间做出如此明确的区分,并将模型参数称为“潜在”或“隐藏”,因为它们是不可观察的。在第 10 章和第 11 章中,我们将使用潜在变量 z ,我们将注意差异,因为我们有两类不同类型的隐藏变量:模型参数 θ 和潜在变量 z 。◇我们可以利用概率模型的所有元素都是随机变量这一事实来定义表示它们的统一语言。在 8.5 节中,我们将看到一种简洁的图形语言来表示概率模型的结构。我们将使用这种图形语言来描述后续章节中的概率模型。

8.4.4 延伸阅读

机器学习中的概率模型 (Bishop, 2006 年; Barber, 2012 年; Murphy, 2012 年) 为用户提供了一种以有原则的方式捕捉数据和预测模型不确定性的方法。Ghahramani (2015) 简要回顾了机器学习中的概率模型。给定一个概率模型,我们可能足够幸运能够分析地计算感兴趣的参数。然而,一般来说,解析解很少见,计算方法如抽样 (Gilks et al., 1996; Brooks et al., 2011) 和变分推理 (Jordan et al., 1999; Blei et al.,

2017)被使用。穆斯塔基等人。(2015) 和 Paquet (2008) 很好地概述了潜在变量模型中的贝叶斯推理。

近年来,已经提出了几种旨在将软件中定义的变量视为随机变量的编程语言

对应于概率分布。目标是能够编写复杂的概率分布函数,而在底层,编译器会自动处理贝叶斯推理规则。

概率规划

这个快速变化的领域被称为概率编程。

8.5 有向图模型

有向图模型

在本节中,我们介绍一种用于指定概率模型的图形语言,称为有向图模型。它提供了一种简洁明了的方式来指定概率模型,并允许读者直观地解析随机变量之间的依赖关系。图形模型直观地捕捉了将所有随机变量的联合分布分解为仅取决于这些变量子集的因子乘积的方式。在第 8.4 节中,我们将概率模型的联合分布确定为感兴趣的关键量,因为它包含有关先验、似然和后验的信息。

有向图模型也是
称为贝叶斯网络。

然而,联合分布本身可能非常复杂,它并没有告诉我们任何有关概率模型结构特性的信息。例如,联合分布 $p(a, b, c)$ 没有告诉我们任何关于独立关系的信息。这就是图形模型发挥作用的地方。本节依赖于独立性和条件独立性的概念,如第 6.4.5 节所述。

图解模型

在图形模型中,节点是随机变量。在图 8.3(a) 中,节点代表随机变量 a, b, c , 边表示变量之间的概率关系,例如,条件概率。

评论。并非每个分布都可以用特定选择的图形模型来表示。Bishop (2006) 中对此进行了讨论。 ◇ 概率图形模型有一些方便的属性:

- 它们是可视化概率模型结构的简单方法。
- 它们可用于设计或激发新型统计模型。
- 仅检查图形就可以让我们深入了解属性,例如,条件独立性。
- 统计模型中用于推理和学习的复杂计算可以用图形操作来表示。

有向图模型/贝叶斯
网络

斯网络是一种在概率模型中表示条件依赖的方法。他们提供了一个视觉

8.5.1 图语义有向图模型/贝叶

条件概率的描述,因此,为描述复杂的相互依存关系提供了一种简单的语言。With additional 的模块化描述还需要计算简化。假设之间有向链接(箭头),箭头可以用来表示两个节点(随机变量)的条件概率。例如,图 8.3(a) 中 a 和 b 之间的箭头给出了给定 a 时 b 的条件概率 $p(b | a)$ 。

来表示因果关系
(Pearl,2009)。

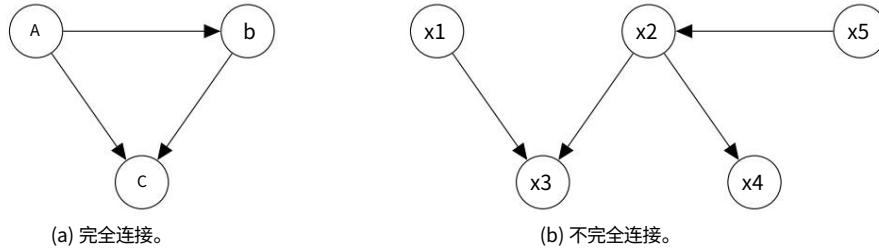


图 8.3 有向图
模型示例。

如果我们对它们的分解有所了解,则有向图模型可以从联合分布中导出。

例 8.7 考虑联合 分布

$$p(a, b, c) = p(c | a, b)p(b | a)p(a) \quad (8.29)$$

三个随机变量 a、b、c。(8.29) 中联合分布的因式分解告诉我们一些关于随机变量之间关系的信息:

- c 直接依赖于 a 和 b。b 直接依赖于
- a。a 既不依赖于 b 也不依赖
- 于 c。

对于(8.29)中的因式分解,我们得到图 8.3 (a) 中的有向图模型。

一般来说,我们可以构造相应的有向图模型
从分解联合分布如下:

1. 为所有随机变量创建一个节点。
2. 对于每个条件分布,我们从对应于分布条件的变量的节点向图形添加一个有向链接
(箭头)。

图形布局取决于
联合分布。

图形布局取决于

我们讨论了如何从联合分布的已知因式分解得到相应的有向图模型。现在,我们将做

联合分布。

恰恰相反,描述了如何从给定的图形模型中提取一组随机变量的联合分布。

示例 8.8 查看

图 8.3(b) 中的图形模型,我们利用两个属性关系:

- 我们寻求的联合分布 $p(x_1, \dots, x_5)$ 是一组条件的乘积,图中的每个节点都有一个。在这个特定的例子中,我们需要五个条件。
- 每个条件仅取决于图中相应节点的父节点。例如, x_4 将以 x_2 为条件。

这两个属性产生了所需的联合分布分解

$$p(x_1, x_2, x_3, x_4, x_5) = p(x_1)p(x_5)p(x_2 | x_5)p(x_3 | x_1, x_2)p(x_4 | x_2)。 \quad (8.30)$$

通常,联合分布 $p(x) = p(x_1, \dots, x_K)$ 为

$$p(x) = \prod_{k=1}^K p(x_k | P_{ak}), \quad (8.31)$$

其中 P_{ak} 表示 “ x_k 的父节点” 。 x_k 的父节点是具有指向 x_k 的箭头的节点。

我们以抛硬币实验的具体例子来结束本小节。考虑一个伯努利实验 (示例 6.8), 其中该实验的结果 x 为 “正面”的概率为

$$p(x | \mu) = \text{Ber}(\mu)。 \quad (8.32)$$

我们现在重复这个实验 N 次并观察结果 x_1, \dots, x_N , 这样我们就得到了联合分布

$$p(x_1, \dots, x_N | \mu) = \prod_{n=1}^N p(x_n | \mu)。 \quad (8.33)$$

右侧的表达式是每个单独结果的伯努利分布的乘积,因为实验是独立的。回顾第 6.4.5 节,统计独立性意味着分布分解。为了写下这个集合的图形模型,我们区分了未观察到/潜在变量和观察到的变量。在图形上,观察到的变量用阴影节点表示,这样我们就可以得到图 8.4(a) 中的图形模型。我们看到单个参数 μ 对于所有 x_n 都是相同的, $n = 1, \dots, N$ 因为结果 x_n 是同分布的。图 8.4(b) 给出了此设置的更紧凑但等效的图形模型,其中我们使用

8.5 有向图模型

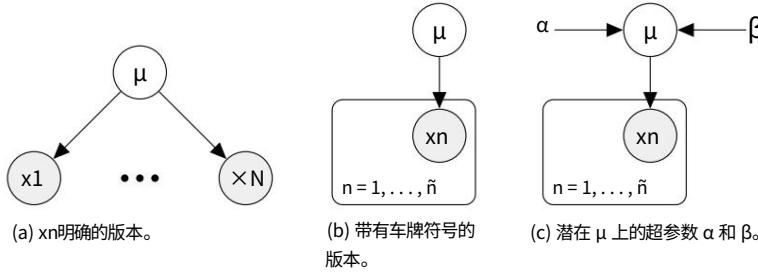


图 8.4 重复伯努利实验的图形模型。

板符号。盘子(盒子)重复里面的所有东西(在这种情况下,盘子观察值 x_n) N 次。因此,两个图形模型是等价的,但车牌符号更紧凑。图模型立即允许我们在 μ 上放置一个超先验。超先验是第一层先验参数的先验分布的第二层超先验。图 8.4(c)在潜在变量 μ 上放置了 Beta(α , β) 先验。如果我们将 α 和 β 视为确定性参数,即不是随机变量,我们将省略围绕它的圆圈。

8.5.2 条件独立性和d-分离

有向图模型允许我们仅通过查看图就可以找到联合分布的条件独立性(第 6.4.5 节)关系属性。称为 d-separation(Pearl, 1988)的概念是其中的关键。

考虑一个一般有向图,其中 A、B、C 是任意不相交的节点集(其并集可能小于图中的完整节点集)。我们希望确定一个特定的条件独立性陈述,“给定 C,A 在条件上独立于 B”,表示为

d-分离

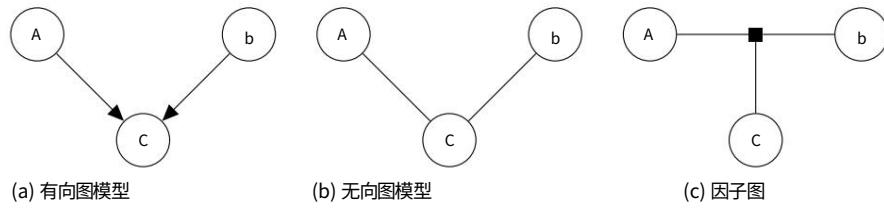
$$A \perp\!\!\!\perp B \mid C, \quad (8.34)$$

由给定的有向无环图暗示。为此,我们考虑了从 A 中的任何节点到 B 中的任何节点的所有可能路径(忽略箭头方向的路径)。如果任何此类路径包含任何节点,则称该路径被阻塞,从而满足以下任一条件:真的:

- 路径上的箭头在节点处从头到尾或尾到尾相交,并且该节点在集合 C 中。
- 箭头在节点处头对头相遇,并且该节点及其任何后代都不在集合 C 中。

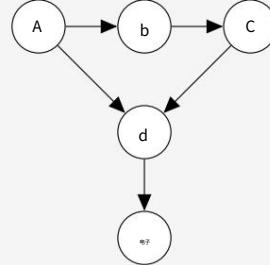
如果所有路径都被阻塞,则称 A 与 B 被 C d-分离,并且图中所有变量的联合分布将满足 $A \perp\!\!\!\perp B \mid C$ 。

图 8.1 三种类型的图模型：(a) 有向图模型（贝叶斯网络）；
(b) 无向图模型（马尔可夫随机场）；(c)
因子图。



例 8.9 (条件独立)

图 8.1 D-分
离示例。



考虑图 8.1 中的图形模型。视觉检查给我们

$$b \perp\!\!\!\perp d \mid a, c \quad (8.35)$$

$$\perp\!\!\!\perp c \mid b \quad (8.36)$$

$$\perp\!\!\!\perp d \mid \quad (8.37)$$

$$ca \quad \perp\!\!\!\perp c \mid \emptyset, \emptyset \quad (8.38)$$

有向图模型允许概率模型的紧凑表示，我们将在第 9、10 和 11 章中看到有向图模型的示例。这种表示以及条件独立的概念允许我们分解相应的概率模型转换成更容易优化的表达式。

概率模型的图形表示使我们能够直观地看到我们所做的设计选择对模型结构的影响。我们经常需要对模型的结构做出高层次的假设。这些建模假设（超参数）会影响预测性能，但不能使用我们目前所见的方法直接选择。我们将在 8.6 节中讨论选择结构的不同方法。

8.5.3 延伸阅读

在 Bishop (2006 年,第 8 章) 中可以找到对概率图形模型的介绍,在 Koller 和 Friedman (2009 年) 的书中可以找到对不同应用程序和相应算法含义的广泛描述。概率图模型主要分为三种类型:

- 有向图形模型 (贝叶斯网络);见图 8.1(a)
- 无向图形模型 (马尔可夫随机场);见图 8.1(b)
- 因子图;见图 8.1(c)

定向图形
模型
贝叶斯网络无向图模型

马尔可夫随机场
因子图

图模型允许基于图的算法进行推理和学习,例如,通过本地消息传递。应用范围包括在线游戏排名 (Herbrich 等人,2007 年) 和计算机视觉 (例如,图像分割、语义标记、图像去噪、图像恢复 (Kittler 和 Foglein,1984 年;Sucar 和 Gillies,1994 年;Shotton 等人) al., 2006; Szeliski et al., 2008)) 到编码理论 (McEliece et al., 1998),求解线性方程系统 (Shental et al., 2008),以及信号处理中的迭代贝叶斯状态估计 (Bickson et al., 2008) 等人,2007 年;Deisenroth 和 Mohamed,2012 年)。

结构化预测的概念 (Bakir 等人,2007 年;Nowozin 等人,2014 年) 是我们在本书中未讨论的在实际应用中特别重要的一个主题,它允许机器学习模型处理结构化预测,例如序列、树和图。神经网络模型的流行使得可以使用更灵活的概率模型,从而导致结构化模型的许多有用应用 (Goodfellow 等人,2016 年,第 16 章)。近年来,由于图形模型在因果推理中的应用,人们对它们重新产生了兴趣 (Pearl,2009 年;Imbens 和 Rubin,2015 年;Peters 等人,2017 年;Rosenbaum,2017 年)。

8.6 模型选择

在机器学习中,我们经常需要做出对模型性能有重大影响的高级建模决策。我们所做的选择 (例如,似然的函数形式) 影响模型中自由参数的数量和类型,从而也影响模型的灵活性和表现力。更复杂的模型在多项式中更灵活,因为它们可以用来描述更多的数据集。例如, $y = a_0 + a_1x + a_2x^2$ 也可以描述 1 次多项式 (直线 $y = a_0 + a_1x$) 只能用于描述输入 x 和观测值 y 之间的线性关系。2 次多项式还可以描述输入和观察之间的二次关系。

线性函数通过设置 $a_2 = 0$,即它比一阶多项式更严格地表达。

现在人们会认为非常灵活的模型通常比简单模型更可取,因为它们更具表现力。一般问题

图 8.1 嵌套交叉验证。我们执行两个级别的 K 折交叉验证。



是在训练时我们只能使用训练集来评估模型的性能并学习其参数。然而，训练集上的表现并不是我们真正感兴趣的。在 8.3 节中，我们已经看到最大似然估计会导致过度拟合，尤其是当训练数据集较小时。理想情况下，我们的模型（也）在测试集（训练时不可用）上运行良好。因此，我们需要一些机制来评估模型如何泛化到看不见的测试数据。模型选择正是关注这个问题。

嵌套交叉验证

8.6.1 嵌套交叉验证

我们已经看到了一种可用于模型选择的方法（第 8.2.4 节中的交叉验证）。回想一下，交叉验证通过将数据集重复拆分为训练集和验证集来提供泛化误差的估计。我们可以再次应用这个想法，即对于每个拆分，我们可以执行另一轮交叉验证。这有时被称为嵌套交叉验证；见图 8.1。内部级别用于估计特定选择的模型或超参数在内部验证集上的性能。外层用于估计内层循环选择的最佳模型选择的泛化性能。我们可以在内循环中测试不同的模型和超参数选择。为了区分这两个级别，用于估计泛化性能的集合通常称为测试集，用于选择最佳模型的集合称为验证集。内循环通过使用验证集上的经验误差对其进行近似来估计给定模型 (8.39) 的泛化误差的预期值，即，

测试集
验证集

标准误
定义为 $\sqrt{\sigma_K}$
其中 K 是
实验次数， σ
是风险的标准差

$$EV [R(V | M)] \approx \frac{1}{K} \sum_{k=1}^K R(V^{(k)} | M), \quad (8.39)$$

每个实验。

其中 $R(V | M)$ 是模型 M 的验证集 V 上的经验风险（例如，均方根误差）。我们对所有模型重复此过程并选择表现最佳的模型。请注意，交叉验证不仅为我们提供了预期的泛化误差，而且我们还可以获得高阶统计量，例如，标准误差，对不确定性的估计。

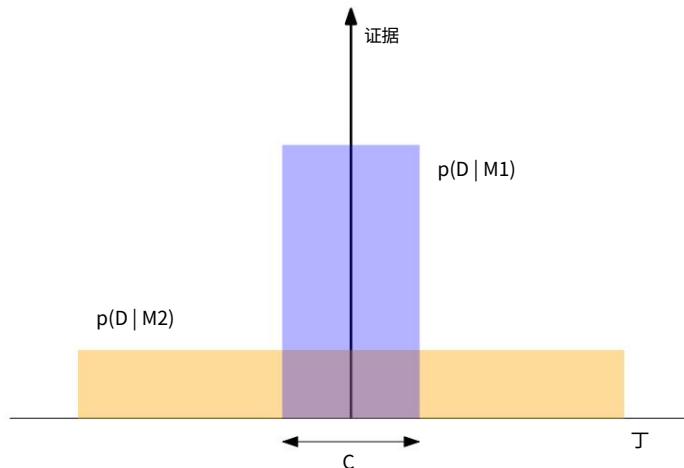


图 8.1 贝叶斯推理体现了奥卡姆剃刀原则。这

横轴描述了所有可能数据集的空间 D 。证据（纵轴）评估模型的好坏

预测可用数据。自从

$p(D | M_i)$ 需要积分为 1, 我们应该选择具有

最大的证据。
改编自麦凯
(2003)。

平均估计是。选择模型后,我们可以评估测试集的最终性能。

8.6.2 贝叶斯模型选择

法,本节介绍了其中的一些方法。通常,它们都试图权衡模型复杂性和数据拟合。我们假设更简单的模型比复杂模型更不容易过度拟合,因此模型选择的目标是找到能够合理很好地解释数据的最简单模型。这个概念也被称为奥卡姆剃刀。

奥卡姆剃刀

评论。如果我们将模型选择视为一个假设检验问题,我们就是在寻找与数据一致的最简单假设 (Murphy, 2012)。 ◇ 可以考虑在有利于更简单模型的模型上放置一个先验。

然而,没有必要这样做:“自动奥卡姆剃刀”定量地体现在贝叶斯概率的应用中 (Smith 和 Spiegelhalter, 1980; Jefferys 和 Berger, 1992; MacKay, 1992)。图 8.1, 改编自 MacKay (2003), 给了我们基本的直觉,为什么复杂和非常有表现力的模型可能被证明是对给定数据集 D 建模的不太可能的选择。让我们考虑水平轴这些预测被量化为 a 表示所有可能的数据集 D 的空间。如果我们对给定数据 D 的模型 M_i 的归一化后验概率 $p(M_i | D)$ 感兴趣,我们可以概率性地使用贝叶斯定理。假设在 D 上的所有模分布上有一个统一的先验 $p(M)$, else, 贝叶斯定理奖励模型与它的先验程度成正比,即,它需要口述发生的数据。给定模型对数据的预测积分/总和为 1., $p(D | M_i)$, 称为 M_i 的证据。

M_i 一个简单的模型 M_1 只能对少量数据集进行证据预测,如 $p(D | M_1)$ 所示;例如,比 M_1 具有更多自由参数的更强大的模型 M_2 能够

预测更多种类的数据集。然而,这意味着M2不能预测区域C和M1中的数据集。假设已将相等的先验概率分配给两个模型。然后,如果数据集落入区域C,则功能较弱的模型M1是更可能的模型。

在本章的前面,我们认为模型需要能够解释数据,即应该有一种方法可以从给定模型生成数据。

此外,如果模型已经从数据中适当地学习,那么我们期望生成的数据应该与经验数据相似。为此,将模型选择描述为层次推理问题是有所帮助的,这使我们能够计算模型的后验分布。

让我们考虑有限数量的模型 $M = \{M_1, \dots, M_K\}$, 其中每个模型 M_k 具有参数 θ_k 。在贝叶斯模型选择中,我们将先验 $p(M)$ 放在模型集上。允许我们从该模型生成数据的相应生成过程是

贝叶斯模型选择
生成过程图 8.2 分层说明

贝叶斯模型选择中的
生成过程。我们将先验
 $p(M)$ 放在模型集上。对
于每个模型,在相应的模
型参数上都有一个分
布 $p(\theta | M)$,

$$M_k \quad p(M) \quad (8.40)$$

$$\theta_k \quad p(\theta | M_k) \quad (8.41)$$

$$D \quad p(D | \theta_k) \quad (8.42)$$

如图 8.2 所示。给定训练集 D, 我们应用贝叶斯定理并计算模型的后验分布为

$$p(M_k | D) \propto p(M_k)p(D | M_k) \quad (8.43)$$

请注意, 此后验不再依赖于模型参数 θ_k , 因为它们已被集成到贝叶斯设置中, 因为

这是用来
生成数据 D.

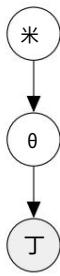
$$p(D | M_k) = p(D | \theta_k)p(\theta_k | M_k)d\theta_k, \quad (8.44)$$

其中 $p(\theta_k | M_k)$ 是模型 M_k 的模型参数 θ_k 的先验分布。术语 (8.44) 称为模型证据或边际似然。从 (8.43) 中的后验, 我们确定 MAP 估计

$$M^* = \underset{\text{马克}}{\text{最大参数}} \quad p(M_k | D). \quad (8.45)$$

使用统一的先验 $p(M_k) =$ 给每个模型相等的 (先验) 概率, 确定模型的 MAP 估计相当于选择最大化模型证据的模型 (8.44)。

模型证据
边际可能性



备注 (可能性和边际可能性)。似然和边际似然 (证据) 之间有一些重要的区别: 虽然似然容易过度拟合, 但边际似然通常不会, 因为模型参数已被边缘化 (即, 我们不再需要拟合参数)。此外, 边际似然自动体现了模型复杂性和数据拟合之间的权衡 (奥卡姆剃刀)。 ◇

8.6.3 用于模型比较的贝叶斯因子考虑比较两个概率模型M1、M2的问题,给定数据集D。如果我们计算后验概率 $p(M1 | D)$ 和 $p(M2 | D)$,我们可以计算比率后验概率

$$\frac{p(M1 | D)}{p(M2 | D)} = \frac{\frac{p(D | M1)p(M1)}{p(D)}}{\frac{p(D | M2)p(M2)}{p(D)}} = \frac{p(M1) p(D | M1) p(M2)}{p(D | M2) p(M2)} . \quad (8.46)$$

后验概率 先验概率 贝叶斯因子

后验概率的比率也称为后验概率。(8.46)右侧的第一个分数后验概率,即先验概率,衡量我们的先验(初始)信念有多少先验概率支持M1而不是M2。边际似然度的比率(右侧的第二个分数)称为贝叶斯因子,用于衡量M1与M2相比预测数据D的好坏程度。

评论。Jeffreys-Lindley悖论指出“贝叶斯因子总是Jeffreys-Lindley倾向于更简单的模型,因为具有扩散先验的复杂模型下数据的概率非常小”(Murphy,2012年)。这里,扩散先验是指不偏向于特定模型的先验,即许多模型在该先验下是先验似是而非的。◊如果我们选择统一先验模型,则(8.46)中的先验概率项为1,即后验概率为边际似然比(贝叶斯因子)

$$\frac{p(D | M1)}{p(D | M2)} . \quad (8.47)$$

如果贝叶斯因子大于1,我们选择模型M1,否则选择模型M2。与常客统计类似,在结果“显著性”之前应该考虑比率大小的指导方针(Jeffreys,1961)。

备注(计算边际似然)。边际似然在模型选择中起着重要作用:我们需要计算贝叶斯因子(8.46)和模型的后验分布(8.43)。

不幸的是,计算边际似然需要我们求解一个积分(8.44)。这种积分通常在分析上难以处理,我们将不得不求助于近似技术,例如,数值积分(Stoer 和 Burlirsch,2002年)、使用蒙特卡罗的随机近似(Murphy,2012年)或贝叶斯蒙特卡罗技术(O'Hagan, 1991年;Rasmussen 和 Ghahramani,2003年)。

但是,有些特殊情况我们可以解决。在第6.6.1节中,我们讨论了共轭模型。如果我们选择先验共轭参数 $p(\theta)$,我们可以计算封闭形式的边际似然。在第9章中,我们将在线性回归的背景下做这件事。◊我们已经在本章中看到了对机器学习基本概念的简要介绍。对于本书这一部分的其余部分,我们将看到

8.2、8.3 和 8.4 节中的三种不同学习风格如何应用于机器学习的四大支柱（回归、降维、密度估计和分类）。

8.6.4 进一步阅读我们在本节

开头提到了影响模型性能的高级建模选择。示例包括以下内容：

- 回归设置中多项式的次数
- 混合模型中的组件数量
- (深度)神经网络的网络架构
- 支持向量机中的内核类型
- PCA 中潜在空间的维度
- 优化算法中的学习率 (时间表)

在参数模型中，参数的数量通常与模型类的复杂性。

Rasmussen 和 Ghahramani (2001) 表明，自动奥卡姆剃刀不一定会惩罚模型中的参数数量，但它在函数的复杂性方面很活跃。他们还表明，自动奥卡姆剃刀也适用于具有许多参数的贝叶斯非参数模型，例如高斯过程。

如果我们关注最大似然估计，则存在许多模型选择启发式方法可以防止过度拟合。它们被称为信息标准，我们选择具有最大价值的模型。这

赤池信息准则(AIC) (赤池,1974 年)
标准

$$\log p(x | \theta) - M \quad (8.48)$$

通过添加惩罚项来纠正最大似然估计的偏差，以补偿具有大量参数的更复杂模型的过度拟合。这里，M 是模型参数的数量。AIC 估计给定模型丢失的相关信息。

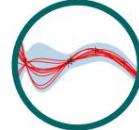
贝叶斯信息
标准

贝叶斯信息准则(BIC) (Schwarz,1978 年)

$$1 \log p(x) = \log p(x | \theta)p(\theta)d\theta \approx \log p(x | \theta) - M \log N \quad (8.49)$$

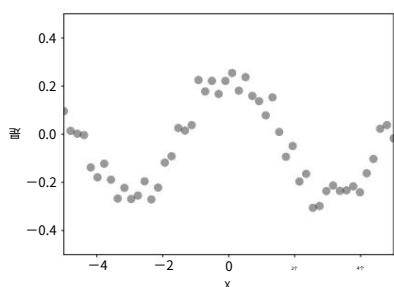
可用于指数族分布。这里，N 是数据点的数量，M 是参数的数量。BIC 比 AIC 更严重地惩罚模型复杂性。

线性回归

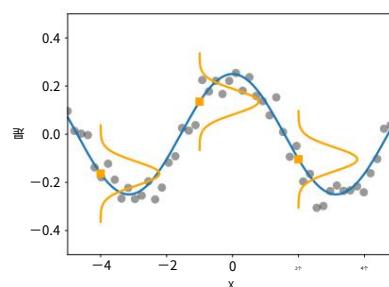


下面,我们将应用第 2.5.6 和 7 章中的数学概念来解决线性回归(曲线拟合)问题。在回归中,我们的目标是找到一个函数 f , 它将输入 $x \in \mathbb{R}^D$ 映射到相应的回归响应函数值 $f(x) \in \mathbb{R}$ 。我们假设给定一组训练输入 x_n 和相应的噪声观测值 $y_n = f(x_n) + \varepsilon$, 其中 ε 是一个 iid 随机变量, 它描述测量/观察噪声和潜在的未建模过程(我们不会在本章中进一步考虑)。在本章中, 我们假设零均值高斯噪声。我们的任务是找到一个函数, 它不仅可以对训练数据建模, 而且可以很好地泛化以预测不属于训练数据的输入位置的函数值(参见第 8 章)。图 9.1 给出了这种回归问题的说明。图 9.1(a) 给出了典型的回归设置: 对于某些输入值 x_n , 我们观察到(噪声)函数值 $y_n = f(x_n) + \varepsilon$ 。任务是推断生成数据并很好地泛化为新输入位置处的函数值的函数 f 。图 9.1(b) 给出了一种可能的解决方案, 其中我们还显示了以表示数据中噪声的函数值 $f(x)$ 为中心的三个分布。

回归是机器学习中的一个基本问题, 回归问题出现在各种各样的研究领域和应用中



(a) 回归问题: 观察到的噪声函数值, 我们希望从中推断出生成数据的基础函数。



(b) 回归解决方案: 可以生成数据(蓝色)的可能函数, 并指示测量噪声

对应输入 x 的函数值(橙色分布)。

图 9.1 (a) 数据集; (b) 回归问题的可能解决方案。

化,包括时间序列分析(例如,系统识别)、控制和机器人技术(例如,强化学习、正向/逆向模型学习)、优化(例如,线搜索、全局优化)和深度学习应用(例如,计算机游戏、语音到文本的翻译、图像识别、自动视频注释)。回归也是分类算法的关键组成部分。寻找回归函数需要解决各种问题,包括:

通常,噪音的类型也可能
是“模型选择”,但我们
将噪声固定为高斯
分布
这一章。

- 模型(类型)的选择和回归函数的参数化。给定一个数据集,哪些函数类(例如,多项式)是对数据建模的良好候选者,我们应该选择什么特定的参数化(例如,多项式的次数)?
如第8.6节所述,模型选择允许我们比较各种模型以找到能够合理很好地解释训练数据的最简单模型。
- 寻找好的参数。选择了回归函数的模型后,我们如何找到好的模型参数?在这里,我们将需要查看不同的损失/目标函数(它们决定什么是“好的”拟合)和使我们能够最小化这种损失的优化算法。
- 过度拟合和模型选择。当回归函数“太好”地拟合训练数据但不能泛化到看不见的测试数据时,就会出现过度拟合问题。如果底层模型(或其参数化)过于灵活和表现力过大,通常会发生过度拟合;参见第8.6节。我们将研究根本原因,并讨论在线性回归的背景下减轻过度拟合影响的方法。
- 损失函数和参数先验之间的关系。损失函数(优化目标)通常由概率模型激发和诱导。我们将研究损失函数与导致这些损失的潜在先验假设之间的联系。
- 不确定性建模。在任何实际设置中,我们只能访问有限的、可能大量的(训练)数据来选择模型类和相应的参数。鉴于这种有限数量的训练数据并未涵盖所有可能的场景,我们可能希望描述剩余的参数不确定性,以获得模型在测试时预测的置信度度量;训练集越小,不确定性建模越重要。不确定性的一致建模为模型预测配备了置信区间。

在下文中,我们将使用第3、5、6和7章中的数学工具来解决线性回归问题。我们将讨论最大似然和最大后验(MAP)估计以找到最佳模型参数。使用这些参数估计,我们将简要了解泛化错误和过度拟合。在本章末尾,我们将讨论贝叶斯线性回归,它允许我们在更高层次上推理模型参数,从而消除最大似然和MAP估计中遇到的一些问题。

9.1 问题表述

由于观察噪声的存在,我们将采用概率方法并使用似然函数对噪声进行显式建模。

更具体地说,在本章中,我们考虑了具有似然函数的回归问题

$$p(y | x) = N(y | f(x), \sigma^2). \quad (9.1)$$

这里, $x \in RD$ 是输入, $y \in R$ 是有噪声的函数值(目标)。

利用(9.1), x 和 y 之间的函数关系为

$$y = f(x) + \varepsilon, \quad (9.2)$$

其中 $\varepsilon \sim N(0, \sigma^2)$ 是独立的同分布(iid)高斯测量噪声,均值为0,方差为 σ^2

^{2*}.我们的目标是找到一

个与生成数据的未知函数 f 接近(相似)并且泛化良好的函数。

在本章中,我们重点关注参数模型,即我们选择一个参数化函数并找到对数据建模“效果很好”的参数 θ 。目前,我们假设噪声方差 σ^2 已知,并专注于学习模型参数 θ 。在线性回归中,我们考虑参数 θ 在我们的模型中线性出现的特殊情况。线性回归的一个例子是”

$$p(y | x, \theta) = N(y | x\theta, \sigma^2) \quad (9.3)$$

$$\Leftrightarrow y = x\theta + \varepsilon \sim N(0, \sigma^2), \quad (9.4)$$

其中 $\theta \in RD$ 是我们寻找的参数。(9.4) 描述的函数类是通过原点的直线。在(9.4)中,我们选择了参数化 $f(x) = x\theta$ 。

(9.3) 中的似然是在 $x = \theta$ 处评估的 y 的概率密度函数。请注意,不确定性的唯一来源来自观察噪声(因为 x 和 θ 假定在(9.3)中已知)。没有观察噪声, x 和 y 之间的关系将是确定性的,并且(9.3) 将是一个 Dirac delta。

狄拉克 delta (delta 函数)除了一个点外,处处为零,其积分为 1。

可以考虑
 σ 的 2 极限中的高斯分
布 $\rightarrow 0$ 。
可能性

例 9.1 对于 x, θ

$\in R$ (9.4) 中的线性回归模型描述的是直线(线性函数),参数 θ 是直线的斜率。图 9.2(a) 显示了不同 θ 值的一些示例函数。

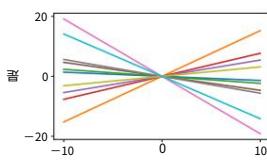
线性回归指的是线性的
模型

参数。

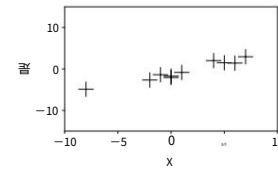
(9.3)-(9.4) 中的线性回归模型不仅在参数上是线性的,而且在输入 x 上也是线性的。图 9.2(a) 显示了非线性反式的示例 $(x)\theta$ 这样的功能。稍后我们会看到 $y = \theta$ 地层也是一个线性回归模型,因为“线性回归”

292

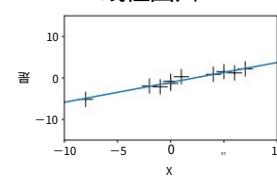
图 9.2 线性回归示例。
(a) 属于此类的示例功能；
(b) 训练集；
(c) 最大似然估计。



(a) 可以使用 (9.4) 中的线性模型描述的示例函数（直线）。



(b) 训练集。



(c) 最大似然估计
伴侣。

指的是“参数线性”的模型,即通过输入特征的线性组合来描述函数的模型。这里,“特征”是输入 x 的表示 (x) 。

在下文中,我们将更详细地讨论如何找到好的参数 θ 以及如何评估参数集是否“有效”。

暂时,我们假设噪声方差 σ^2

²⁹ 众所周知。

9.2 参数估计

考虑线性回归设置 (9.4) 并假设我们有一个训练集 $D := \{(x_1, y_1), \dots, (x_N, y_N)\}$ 由 N 个输入 $x_n \in \mathbb{R}^D$ 和相应的观察/目标 $y_n \in \mathbb{R}$, $n = 1, \dots, N$. 相应的图形模型如图 9.3 所示。请注意, y_i 和 y_j 在给定它们各自的输入 x_i 、 x_j 的情况下是条件独立的, 因此似然因式分解为

训练集图 9.3

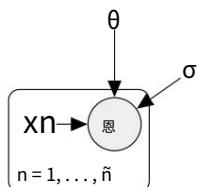
概率

线性回归的图形模型。

观察到的随机变量是

阴影,确定
性/已知值是

没有圆圈。



我们定义 $X := \{x_1, \dots, x_N\}$ 和 $Y := \{y_1, \dots, y_N\}$ 分别作为训练输入和相应目标的集合。由于噪声分布, 似然度和因子 $p(y_n | x_n, \theta)$ 呈高斯分布; 见 (9.3)。

下面, 我们将讨论如何找到最优参数 θ^* 是线性回归模型 (9.4) 的 \mathbb{R}^D 。一旦找到参数 θ^* , 我们就可以通过使用 (9.4) 中的参数 θ^* 来预测函数值, 以便在任意测试输入 x 对应目标 y 的分布是

$$p(y^* | x^*, \theta^*) = N(y^* | x^*, \theta^*, \sigma^2) \quad (9.6)$$

在下文中, 我们将通过最大化似然来了解参数估计, 这是我们在第 8.3 节中已经在一定程度上讨论过的主题。

9.2.1 最大似然估计

一种广泛使用的寻找所需参数 θ_{ML} 的方法是最大似然估计,我们在其中找到使似然最大化的参数 θ_{ML} (9.5b)。直觉上,最大化似然意味着在给定模型参数的情况下最大化训练数据的预测分布。
我们获得最大似然参数为

可能性意味着最大化预测分布

$$\theta_{ML} \in \arg \max_{\theta} p(Y | X, \theta). \quad (9.7)$$

给定参数的(训练)数据。

评论。似然 $p(y | x, \theta)$ 不是 θ 中的概率分布:它只是参数 θ 的函数,但不积分为1(即未归一化),甚至可能不可积分 θ 。

可能性不大中的概率分布

然而,(9.7)中的似然是 y 中的归一化概率分布 \diamond 。

参数。

为了找到使似然最大化的所需参数 θ_{ML} ,我们通常执行梯度上升(或对负似然进行梯度下降)。然而,在我们这里考虑的线性回归的情况下,由于对数存在封闭形式的解决方案,因此不需要迭代梯度下降。在实践中,我们不是直接最大化似然,而是将对数变换应用于似然函数并最小化负对数似然。

是一个(严格)单调递增的函数,函数 f 的最优值是

备注(对数转换)。由于似然(9.5b)是 N 个高斯分布的乘积,对数变换很有用,因为(a)它不会受到数值下溢的影响,并且(b)微分规则会变得更简单。更具体地说,当我们乘以 N 个概率时,数值下溢将成为一个问题,其中 N 是数据点的数量,因为我们不能表示非常小的数字,例如 10^{-256} 。

与 $\log f$ 的最佳值。

此外,对数变换会将乘积转换为对数概率之和,使得相应的梯度是各个梯度的和,而不是重复应用乘积规则(5.46)来计算 N 项乘积的梯度。 \diamond 为了找到线性回归问题的最优参数 θ_{ML} ,我们最小化负对数似然

$$-\log p(Y | X, \theta) = -\sum_{n=1}^N \log p(y_n | x_n, \theta), \quad (9.8)$$

由于我们对训练集的独立性假设,我们利用似然(9.5b)分解数据点的数量。

在线性回归模型(9.4)中,似然是高斯分布的(由于高斯加性噪声项),因此我们得出

$$\log p(y_n | x_n, \theta) = -(y_n - \bar{x}_n)^2 / (2\sigma^2) + \text{常量}, \quad (9.9)$$

其中常数包括所有独立于 θ 的项。在(9.9)中使用

负对数似然 (9.8), 我们得到 (忽略常数项)

$$L(\theta) := \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - x_n^\top \theta)^2 \quad (9.10a)$$

$$= \frac{1}{2\sigma^2} \|y - X\theta\|^2, \quad (9.10b)$$

负对数似然函数
也称为误差函数。
设计矩阵 平方误差
通常用作

中我们定义设计矩阵 $X := [x_1, \dots, x_N]$ 作为 $\in \mathbb{R}^N$ 作为一个向量
训练输入和 $y := [y_1, \dots, y_N]$ 收集所有训练目标。请注意, 设
计矩阵 X 中的第 n 行对应于训练输入 x_n 。在 (9.10b) 中, 我们使用了观测值 y_n 与相应模型
预测值 x 之间的误差平方和

距离的度量。
召回自
第 3.1 节说 $2 =$
 $\|x\|_2^2$ x 如果我们
选择点积作为内积。

θ 等于 y 和 $X\theta$ 之间的平方距离。
通过 (9.10b), 我们现在有了需要优化的负对数似然函数的具体形式。我们立即看到
(9.10b) 是 θ 的二次方。这意味着我们可以找到一个唯一的全局解 θ_{ML} 来最小化负对数似
然 L 。我们可以通过计算 L 的梯度, 将其设置为 0 并求解 θ 来找到全局最优解。

使用第 5 章的结果, 我们计算 L 相对于参数的梯度为

$$\frac{\partial L}{\partial \theta} = \frac{d}{d\theta_1} \frac{1}{2\sigma^2} (y - X\theta)^\top (y - X\theta) \quad (9.11a)$$

$$= \frac{2}{2\sigma^2} \frac{d}{d\theta_1} y^\top y - 2y^\top X\theta + \theta^\top X^\top X\theta \quad (9.11b)$$

$$= \frac{1}{\sigma^2} (-y^\top X + \theta^\top X) \in \mathbb{R}^{1 \times D}. \quad (9.11c)$$

最大似然估计器 θ_{ML} 求解恶意条件), 我们得到 $\frac{dL}{d\theta} = 0$ (必要的优化

忽略重复数据点
的可能性, 如果
 $N > D$, 则 $\text{rk}(X) = D$, 即我们没有比数据点更
多的参数。

$$\frac{dL}{d\theta} \xrightarrow{(9.11c)} \theta^\top X^\top X = y^\top y \quad (9.12a)$$

$$\Leftrightarrow \theta^\top X^\top X = y^\top y \quad (9.12b)$$

$$\Leftrightarrow \theta_{ML} = (X^\top X)^{-1} X^\top y. \quad (9.12c)$$

我们可以将第一个方程右乘以 $(X^\top X)^{-1}$, 因为如果 $\text{rk}(X) = D$, $X^\top X$ 是正定的, 其中
 $\text{rk}(X)$ 表示 X 的秩。

评论。将梯度设置为 0 是充分必要条件, 并且我们获得了全局最小值, 因为 Hessian $\nabla^2 L(\theta) = X^\top X \in \mathbb{R}^{D \times D}$ 是正定的。 ◇

评论。 (9.12c) 中的最大似然解要求我们求解形式为 $A\theta = b$ 的线性方程组, 其中 $A = (X^\top X)$ 和 $b = X^\top y$ 。

例 9.2 (拟合线)

让我们看一下图 9.2,我们的目标是使用最大似然估计将直线 $f(x) = \theta x$ (其中 θ 是未知斜率) 拟合到数据集。该模型类中的函数示例 (直线) 如图 9.2(a) 所示。对于图 9.2(b) 所示的数据集,我们使用 (9.12c) 找到斜率参数 θ 的最大似然估计,并获得图 9.2(c) 中的最大似然线性函数。

特征的最大似然估计

到目前为止,我们考虑了 (9.4) 中描述的线性回归设置,它允许我们使用最大似然估计将直线拟合到数据。然而,当线性回归拟合更有趣的数据时,直线的表现力不够。幸运的是,线性回归为我们提供了一种在线性回归框架内拟合非线性函数的方法:由于“线性回归”仅指“参数中的线性”,我们可以对输入 x 执行任意非线性变换 $\phi(x)$,然后线性组合此转换的组件。对应的线性回归模型为

指的是“线性参数”回归模型,但输入可以进行任何非线性变换。

$$\begin{aligned} p(y | x, \theta) &= N(y | \phi(x)\theta, \sigma^2) \\ \Leftrightarrow y &= (\phi(x)\theta + \varepsilon)_{k=0}^{K-1}, \end{aligned} \quad (9.13)$$

其中 $\phi: RD \rightarrow RK$ 是输入 x 的(非线性)变换, $k: RD \rightarrow R$ 是特征向量的第 k 个分量。请注意,特征向量模型参数 θ 仍然仅线性出现。

例 9.3 (多项式回归)

我们关注回归问题 $y = \phi(x)\theta + \varepsilon$, 其中 $x \in RD$ 和 $\theta \in RK$ 。在这种情况下经常使用的转换是

$$\phi(x) = \begin{matrix} \phi_0(x) \\ \phi_1(x) \\ \vdots \\ \phi_{K-1}(x) \end{matrix} = \begin{matrix} 1 \\ x \\ x^2 \\ \vdots \\ x^{K-1} \end{matrix} \quad K \in R_+ . \quad (9.14)$$

这意味着我们将原始的一维输入空间“提升”为由所有单项式 x 组成的 K 维特征空间,其中 $k = K - 1$ 。利用这些特征,我们可以对 0 次多项式建模, ..., ..., $K-1$ 在线性回归的框架内:一次多项式

$K - 1$ 是

$$f(x) = \sum_{k=0}^{K-1} \theta_k x^k = \phi(x)\theta, \quad (9.15)$$

其中在 (9.14) 中定义且 $\theta = [\theta_0, \dots, \theta_{K-1}]$ (线性)参数 $\theta_k \in R$ 包含

现在让我们看一下线性回归模型 (9.13) 中参数 θ 的最大似然估计。我们考虑训练输入 $x_n \in RD$ 和目标 $y_n \in R, n = 1, \dots, N$, 定义特征矩阵 (设计矩阵) 为

特征矩阵

设计矩阵

$$\Phi := \begin{matrix} \Phi & (x_1) \\ \vdots & \\ \Phi & (x_N) \end{matrix} = \begin{matrix} \phi_0(x_1) & \cdots & \phi_{K-1}(x_1) \\ \phi_0(x_2) & \cdots & \phi_{K-1}(x_2) \\ \vdots & & \vdots \\ \phi_0(x_N) & \cdots & \phi_{K-1}(x_N) \end{matrix}, \quad K \in R^{N \times K}, \quad (9.16)$$

其中 $\Phi_{ij} = j(x_i)$ 和 $j: RD \rightarrow R$

例 9.4 (二阶多项式的特征矩阵)

对于一个二阶多项式和 N 个训练点 $x_n \in R, n = N$, 特征矩阵为 $1, \dots,$

$$\Phi = \begin{matrix} 1 \times 1 \times & \vdots \\ 1 \times 2 \times & \vdots \\ \vdots & \vdots \\ 1 \times N \times & \vdots \end{matrix}. \quad (9.17)$$

利用 (9.16) 中定义的特征矩阵 Φ , 线性回归模型 (9.13) 的负对数似然可写为

$$-\log p(Y | X, \theta) = 2\sigma_2^{-1} (y - \Phi\theta)^T (y - \Phi\theta) + \text{const.} \quad (9.18)$$

将 (9.18) 与 (9.10b) 中“无特征”模型的负对数似然相比较, 我们立即发现我们只需要用 Φ 替换 X 。

由于 X 和 Φ 都与我们希望的参数 θ 无关

最大似然优化, 我们立即得出最大似然估计
估计

$$\theta_{ML} = (\Phi^T \Phi)^{-1} \Phi^T y \quad (9.19)$$

对于在 (9.13) 中定义的具有非线性特征的线性回归问题。

评论。当我们在没有特征的情况下工作时, 我们要求 X 是可逆的, 即 $\text{rk}(X) = D$ 时的情况, 即 X 的列

是线性独立的。因此，在(9.19)中，我们要求 $\Phi \in \mathbb{R}^{K \times K}$ 是可逆的。当且仅当 $\text{rk}(\Phi) = K$ 时才会出现这种情况。 ◇

例 9.5 (最大似然多项式拟合)

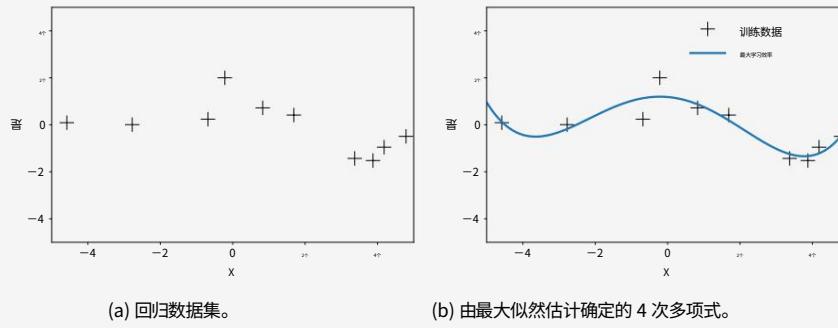


图 9.4 多项式回归。
(a) 由 (x_n, y_n) 对组成的数据集， $n = 1, \dots, 10$ ；
(b) 4 次最大似然多项式。

考虑图 9.4(a) 中的数据集。数据集由 $N = 10$ 对 (x_n, y_n) 组成，其中 $x_n \sim U[-5, 5]$ 且 $y_n = -\sin(x_n/5) + \cos(x_n) + \epsilon$ ，其中 $\epsilon \sim N(0, 0.2)$ 。我们使用最大似然估计拟合4次多项式，即参数 θ ML在(9.19)中给出。在 \dots 。

任何测试位置 x 的最大似然估计 $(x)_{\theta \text{ML}}$ 。结果是

产生图 9.4(b) 所示的函数值 \dots 。

估计噪声方差到目前为止，我们假设噪

声方差 σ^2 已知。然而，我们也可以利用最大似然估计的原理来获得噪声方差的最大似然估计量 $\hat{\sigma}^2$ 。为此，我们遵循标准程序：写下对数似然，计算其关于 $\sigma > 0$ 的导数，将其设置为0，然后求解。对数似然由下式给出

$$\log p(Y | X, \theta, \sigma^2) = \sum_{n=1}^N \log N(y_n | \phi(x_n)\theta, \sigma^2) \quad (9.20a)$$

$$= -\frac{1}{2} \log(2\pi) - \frac{1}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2 \quad (9.20b)$$

$$= -\frac{N}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2 + \text{常量。} \quad (9.20c)$$

对数似然关于 σ $\partial \log p(Y|X, \theta, \sigma^2) / \partial \sigma^2$ 的偏导数

然后是

$$\frac{\partial}{\partial \sigma^2} = -\frac{1}{2\sigma^4} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2 \quad (9.21a)$$

$$\Leftrightarrow \frac{\partial}{\partial \sigma^2} = \frac{1}{2\sigma^4} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2 \quad (9.21b)$$

以便我们识别

$$\frac{\partial}{\partial \sigma^2} = \frac{1}{\sigma^2} = \frac{1}{n} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2 \quad (9.22)$$

因此,噪声方差的最大似然估计是无噪声函数值 y_n 之间的平方距离的经验平均值。

$(x_n)\theta$ 和输入 x_n 处的相应噪声观测值 y_n

9.2.2 线性回归中的过度拟合

我们刚刚讨论了如何使用最大似然估计来将线性模型（例如，多项式）拟合到数据。我们可以通过计算产生的错误/损失来评估模型的质量。这样做的一种方法是计算负对数似然 (9.10b)，我们将其最小化以确定最大似然估计量。或者，鉴于噪声参数 σ 不是自由模型参数，我们可以忽略 $1/\sigma^2$ 的缩放，因此我们最终得到平方误差损失函数 // $y - \phi\theta$ // 平方误差(RMSE)

均方根
错误
均方根误差

. 我们通常不使用这个平方损失，而是使用均方根

$$\sqrt{\frac{1}{n} \|y - \phi\theta\|^2} = \sqrt{\frac{1}{n} \sum_{n=1}^N (y_n - \phi(x_n)\theta)^2}, \quad (9.23)$$

RMSE 已归一化。

其中 (a) 允许我们比较不同大小的数据集的误差，并且 (b) 与观察到的函数值 y_n 具有相同的尺度和相同的单位。例如，如果我们拟合一个将邮政编码 (x 以纬度、经度给出) 映射到房价 (y 值是欧元) 的模型，则 RMSE 也以欧元衡量，而平方误差以 EUR² 给出。

负对数似然是无量纲的。

如果我们选择包括原始负对数似然 (9.10b) 中的因子 σ ，那么我们最终会得到一个无单位的目标，即在前面的示例中，我们的目标将不再以 EUR 或 EUR² 为单位。

对于模型选择（参见第 8.6 节），我们可以使用 RMSE（或负对数似然）通过找到使目标最小化的多项式次数 M 来确定多项式的最佳次数。鉴于多项式次数是一个自然数，我们可以执行强力搜索并枚举 M 的所有（合理）值。对于大小为 N 的训练集，测试 $0 \leq M \leq N - 1$ 就足够了。对于 $M < N$ ，最大似然估计是唯一的。对于 $M = N$ ，我们有更多的参数

9.2 参数估计

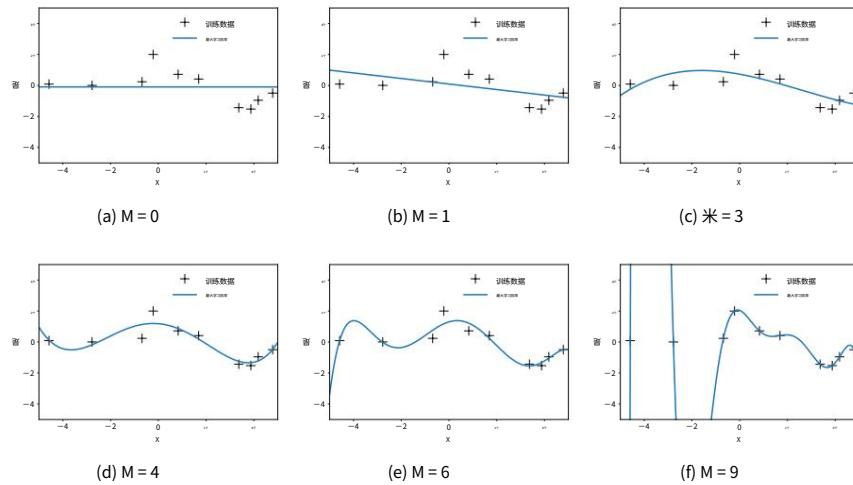


图 9.5 最大似然拟合

不同的多项式次数 M 。

比数据点，并且需要求解欠定的线性方程组（(9.19) 中的 $\Phi^\top \Phi$ 也将不再可逆），因此存在无限多个可能的最大似然估计量。

图 9.5 显示了由图 9.4(a) 中的数据集的最大似然确定的多项式拟合数，其中 $N = 10$ 个观测值。

我们注意到，低次多项式（例如，常数($M = 0$) 或线性($M = 1$)）与数据的拟合很差，因此不能很好地表示真实的基础函数。对于度数 $M = 3, \dots, 6$ ，拟合看起来合理并且平滑地插入数据。当我们进入更高阶多项式的情况下时，我们注意到它们越来越适合数据。在 $M = N - 1 = 9$ 的极端情况下，该函数将通过每个数据点。然而，这些高次多项式剧烈振荡并且不能很好地表示生成数据的基础函数，因此我们会遭受过度拟合的困扰。

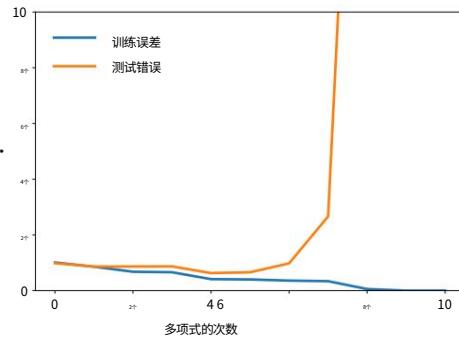
$M = N - 1$ 是
极端的意思是，否则相应的线性方程组的零空间将是平凡的，并且我们将有无限多个线性回归问题的最优解。过拟合

请记住，目标是通过对新（未见）数据进行准确预测来实现良好的泛化。通过考虑包含 200 个数据点的单独测试集，我们获得了一些关于泛化性能对 M 次多项式依赖性的定量洞察，这些测试集使用与生成训练集完全相同的程序生成。作为测试输入，我们选择了 $[-5, 5]$ 区间内 200 个点的线性网格。对于 M 的每个选择，我们评估训练数据和测试数据的 RMSE (9.23)。

请注意噪音方差 $\sigma^2 > 0$ 。

现在看一下测试误差，它是相应多项式的泛化特性的定性度量，我们注意到最初测试误差减小；参见图 9.2（橙色）。对于四阶多项式，测试误差相对较低并且在 5 次之前保持相对恒定。但是，从 6 次开始测试误差显着增加，并且高阶多项式具有非常差的泛化性质。在这个特定的例子中，这也从相应的

图 9.2 训练和测试误差。



训练误差
最大似然拟合在图 9.5 中。请注意,当多项式的次数增加时,训练误差(图 9.2 中的蓝色曲线)永远不会增加。在我们的示例中,最佳泛化(最小测试误差点)是针对 $M = 4$ 的多项式获得的。

测试错误

9.2.3 最大后验估计

我们刚刚看到最大似然估计容易过度拟合。

我们经常观察到,如果我们遇到过度拟合,参数值的大小会变得相对较大(Bishop, 2006)。

为了减轻巨大参数值的影响,我们可以在参数上放置一个先验分布 $p(\theta)$ 。先验分布明确编码了哪些参数值是合理的(在看到任何数据之前)。

例如,单个参数 θ 上的高斯先验 $p(\theta) = N(0, 1)$ 编码参数值预期位于区间 $[-2, 2]$ (围绕平均值的两个标准差)内。一旦数据集 XY 可用,我们就不会寻求最大化后验分布 $p(\theta | X, Y)$ 的参数,而不是最大化似然。此过程称为最大后验(MAP)估计。

最大一个

事后的
地图

参数 θ 的后验,给定训练数据 X ,是
通过应用贝叶斯定理(第 6.3 节)获得 $p(Y | X, \theta)p(\theta)p(\theta | X,$

$$Y) = p(Y | X) \quad (9.24)$$

由于后验明确地取决于参数先验 $p(\theta)$,因此先验将对我们发现的作为后验最大化器的参数向量产生影响。我们将在下文中更明确地看到这一点。最大化后验(9.24)的参数向量 θ_{MAP} 是 MAP 估计。

为了找到 MAP 估计,我们遵循与最大似然估计类似的步骤。我们从对数变换开始并将对数后验计算为

$$\log p(\theta | X, Y) = \log p(Y | X, \theta) + \log p(\theta) + \text{const} \quad (9.25)$$

其中常数包含与 θ 无关的项。我们看到 (9.25) 中的对数后验是对数似然 $p(Y|X, \theta)$ 和对数先验对数 $p(\theta)$ 的总和,因此 MAP 估计将是先验 (我们在观察数据之前对合理参数值的建议) 和数据相关的可能性。

为了找到 MAP 估计 θ_{MAP} , 我们最小化关于 θ 的负对数后验分布, 即, 我们求解

$$\arg \min_{\theta} \{-\log p(Y|X, \theta) - \log p(\theta)\} \quad \theta_{MAP} \in \quad (9.26)$$

负对数后验关于 θ 的梯度是 $d \log p(Y|X, \theta) / d\theta$

$$-\frac{d \log p(\theta | X, Y)}{d\theta} = -\frac{\frac{1}{2\sigma^2} (y - \Phi\theta)^T (y - \Phi\theta) + \frac{1}{2b^2} \theta^T \theta}{2\sigma^2} \quad (9.27)$$

其中我们将右侧的第一项确定为 (9.11c) 的负对数似然的梯度。

使用 (共轭) 高斯先验 $p(\theta) = N(0, bI)$, 参数 θ 上的 bI , 线性回归设置 (9.13) 的负对数后验, 我们获得负对数后验

$$-\log p(\theta | X, Y) = \frac{1}{2\sigma^2} (y - \Phi\theta)^T (y - \Phi\theta) + \frac{1}{2b^2} \theta^T \theta + \text{常量。} \quad (9.28)$$

这里, 第一项对应于对数似然的贡献, 第二项来自对数先验。然后, 关于参数 θ 的对数后验梯度为 $d \log p(\theta | X, Y) / d\theta$

$$-\frac{\partial}{\partial \theta} \left[\frac{1}{2\sigma^2} (y - \Phi\theta)^T (y - \Phi\theta) + \frac{1}{2b^2} \theta^T \theta \right] = \frac{1}{2\sigma^2} \Phi^T (y - \Phi\theta) + \frac{1}{b^2} \theta \quad (9.29)$$

我们将通过将此梯度设置为 0 求解 θ_{MAP} 来找到 MAP 估计 θ_{MAP} 。我们获得 和

$$\frac{1}{2\sigma^2} (\Phi^T y - \Phi^T \Phi \theta) + \frac{1}{b^2} \theta = 0 \quad (9.30a)$$

$$\Leftrightarrow \theta = \frac{1}{2\sigma^2} \Phi^T y - \frac{1}{b^2} \Phi \theta \quad (9.30b)$$

$$\Leftrightarrow \theta = \frac{1}{2\sigma^2} \Phi^T y - \frac{1}{b^2} \Phi \theta \quad (9.30c)$$

$$\Leftrightarrow \theta = \frac{1}{2\sigma^2} \Phi^T y - \frac{1}{b^2} \Phi \theta \quad (9.30 \text{ 天})$$

使得 MAP 估计是 (通过调换最后一个等式的两边) $\Phi^T \Phi$ 是对称的,

$$\theta_{MAP} = \Phi^T y - \frac{1}{b^2} \Phi \theta \quad \Phi^T \Phi \text{ 是对称的。} \quad (9.31)$$

半正定。附加条款

在 (9.31) 中是严格正定的, 因此逆存在。

将 (9.31) 中的 MAP 估计与 (9.19) 中的最大似然估计进行比较, 我们看到两个解之间的唯一区别是逆矩阵中的附加项! 该术语确保

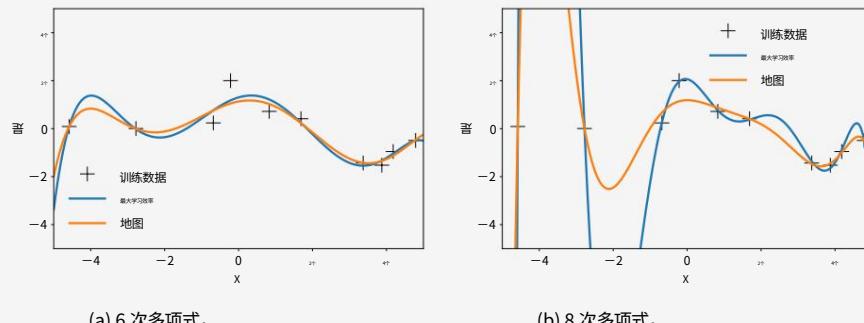
$$\frac{1}{b^2}$$

$\Phi + \lambda I$ 是对称且严格正定的（即它的逆存在，MAP 估计是线性方程组的唯一解）。此外，它反映了正则化器的影响。

例 9.6 (多项式回归的 MAP 估计)

在第 9.2.1 节的多项式回归示例中，我们在参数 θ 上放置高斯先验 $p(\theta) = N(0, I)$ 并根据 (9.31) 确定 MAP 估计。在图 9.1 中，我们显示了 6 次 (左) 和 8 次 (右) 多项式的最大似然估计和 MAP 估计。先验 (正则化器) 对低阶多项式没有显著影响，但对高阶多项式保持函数相对平滑。虽然 MAP 估计可以突破过拟合的界限，但它并不是解决这个问题的通用方法，因此我们需要一种更有原则的方法来解决过拟合问题。

图 9.1 多项式
回归：最大似
然和 MAP 估
计。(a) 6 次多项式；
(b) 8 次多项式。



9.2.4 作为正则化的 MAP 估计

除了在参数 θ 上放置先验分布之外，还可以通过正则化惩罚参数的幅度来减轻过度拟合的影响。在正则化最小二乘法中，我们考虑损失函数

最小正则化

正方形

$$\|y - \Phi\theta\|^p + \lambda \|\theta\|^2 \quad (9.32)$$

数据拟合项
不合适的术语
regularizer
正则化参数

我们将 θ 最小化（参见第 8.2.3 节）。这里，第一项是数据拟合项（也称为失配项），它与负对数似然成正比；参见 (9.10b)。第二项称为正则化器，正则化参数 $\lambda > 0$ 控制正则化的“严格性”。

评论。我们可以在 (9.32) 中选择任意 p 范数 $\|\cdot\|_p$ 替代欧几里得范数 $\|\cdot\|_2$ 。实际上， p 值越小，解越稀疏。这里，“稀疏”是指许多参数值 $\theta_d = 0$ ，这也是

对变量选择很有用。对于 $p = 1$, 正则化项称为 LASSO LASSO (最小绝对收缩和选择算子), 由 Tibshirani (1996) 提出。

(9.32) 中的正则项 $\|\theta\|_1$ 可以解释为负对数高斯先验, 我们在 MAP 估计中使用它; 见 (9.26)。更具体地说, 使用高斯先验 $p(\theta) = N(0, b^2)$ 我们获得负对数高斯先验

$$-\log p(\theta) = \frac{1}{2} \theta^\top \theta / 2 - \frac{1}{2} \log(2\pi b^2) + \text{常量} \quad (9.33)$$

因此对于 $\lambda = \frac{1}{2b^2}$ 正则项和负对数高斯先验是相同的。

鉴于 (9.32) 中的正则化最小二乘损失函数由与负对数似然和负对数先验密切相关的项组成, 当我们最小化此损失时, 我们得到一个解也就不足为奇了这与 (9.31) 中的 MAP 估计非常相似。更具体地说, 最小化正则化最小二乘损失函数收益率

$$\theta_{RLS} = (\Phi^\top \Phi + \lambda I)^{-1} \Phi^\top y, \quad (9.34)$$

这与 (9.31) 中的 MAP 估计相同, 因为 $\lambda = \sigma_b^2$ 其中 σ_b^2 是 (各向同性) 高斯先验的方差

$p(\theta) = N(0, b^2)$ 。

到目前为止, 我们已经介绍了使用最大似然估计和 MAP 估计的参数估计, 其中我们发现了优化目标函数 (似然或后验) 的点估计 $\hat{\theta}$ 。我们看到最大似然估计和 MAP 估计都会导致过度拟合。在下一节中, 我们将讨论贝叶斯线性回归, 其中我们使用贝叶斯推理 (第 8.4 节) 找到未知参数的后验分布, 我们随后使用它进行预测。

与分布不同, 点估计是单个特定参数值
超过合理的参数设置。

更具体地说, 对于预测, 我们将对所有可能的参数集进行平均, 而不是关注点估计。

9.3 贝叶斯线性回归之前, 我们研究了线性回归模

型, 其中我们估计模型参数 θ , 例如, 通过最大似然或 MAP 估计。我们发现 MLE 会导致严重的过度拟合, 尤其是在小数据领域。MAP 通过在扮演正则化角色的参数上放置一个先验来解决这个问题。

贝叶斯线性

贝叶斯线性回归推动了步进回归之前的参数思想
进一步甚至不尝试计算参数的点估计, 而是在进行预测时考虑参数的完整后验分布。这意味着我们不适合任何参数, 但我们计算所有合理参数设置的平均值 (根据后验)。

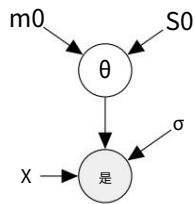
9.3.1 型号

在贝叶斯线性回归中,我们考虑模型

$$\text{似然 } p(\theta) = N(m_0, S_0 \text{ 先验}) , \\ p(y | x, \theta) = N(y | \phi(x)\theta, \sigma^2) , \quad (9.35)$$

图 9.2 贝叶斯
线性回归的图形模型。

我们现在明确地在 θ 上放置一个高斯先验 $p(\theta) = N(m_0, S_0)$,这将参数向量变成了一个随机变量。这使我们能够在图 9.2 中写下相应的图形模型,其中我们明确了 θ 上高斯先验的参数。完整的概率模型,即观察到的和未观察到的随机变量 y 和 θ 的联合分布分别为



$$p(y, \theta | x) = p(y | x, \theta)p(\theta) . \quad (9.36)$$

9.3.2 先前的预测

实际上,我们通常对参数值 θ 本身不太感兴趣。相反,我们的重点通常在于我们对这些参数值所做的预测。在贝叶斯设置中,我们在进行预测时采用参数分布并对所有合理的参数设置进行平均。更具体地说,为了对输入 x 进行预测,我们对 θ 进行积分并获得

$$p(y^* | x) = p(y^* | x, \theta)p(\theta)d\theta = E_\theta[p(y^* | x, \theta)] , \quad (9.37)$$

我们可以将其解释为 y^* 的平均预测 x , θ 对于根据先验分布 $p(\theta)$ 的所有似是而非的参数 θ 。请注意,使用先验分布的预测只需要我们指定输入 x^* ,而不需要训练数据。

在我们的模型 (9.35) 中,我们选择了 θ 上的共轭(高斯)先验,以便预测分布也是高斯分布(并且可以以封闭形式计算): 对于先验分布 $p(\theta) = N(m_0, S_0)$,我们得到预测分布为

$$p(y^* | x^*) = N((x^*)m_0, \phi((x^*)S_0)(x^*) + \sigma^2) , \quad (9.38)$$

我们利用了(i)由于共轭性(见第 6.6 节)和高斯的边缘化特性(见第 6.5 节),预测是高斯分布的,(ii)高斯噪声是独立的,因此

$$V[y^*] = V[\theta(x^*)] + V[\epsilon] , \quad (9.39)$$

(iii) y 是 θ 的线性变换,因此我们可以应用 $(x^*)S_0(x^*)$ 分析计算预测的均值和协方差的规则

分别使用 (6.50) 和 (6.51)。在 (9.38) 中,预测方差中的项明确说明了相关的不确定性

参数为 θ ,而 σ 为测量噪声。

\hat{y} 是由于的不确定性贡献

如果我们有兴趣预测无噪声函数值 $f(x^*) = \phi(x^*)\theta$ 而不是我们得到的被噪声破坏的目标 y

$$p(f(x^*)) = N(x^*)m_0, \phi(x^*)s_0(x^*), \quad (9.40)$$

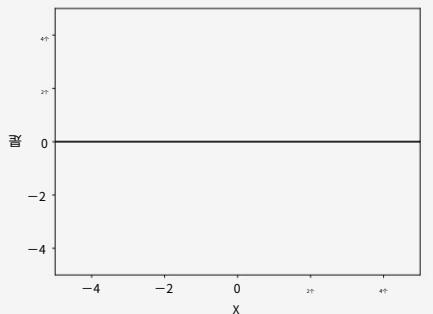
它与 (9.38) 的唯一不同在于省略了噪声方差 σ^2 预测方差。

\hat{y} 在

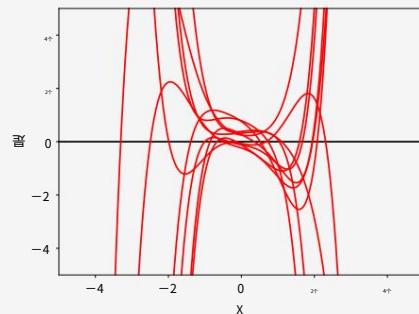
备注 (函数分布)。由于我们可以使用一组样本 θ_i 来表示分布参数 $p(\theta)$,并且每个样本 θ_i 都会产生一个函数 $f_i(\cdot) = \theta_i$ 会在函数上产生一个分布 $p(f_i(\cdot))$ 。这里我们使用符号 (\cdot) 来明确表示函数关系。◇
示函数关系。 \diamond 分布 $p(\theta)$

功能。

例 9.7 (优先于函数)



(a) 功能的优先分配。



(b) 来自函数先验分布的样本。

图 9.3先验函数。

(a) 职能分配

用均值函数表示

(黑线)和边缘

不确定性
(阴影) ,
分别代表 67% 和
95% 的置信区

间; (b) 来自先验函数
的样本,这些样
本是由来自参数先
验的样本引起的。

让我们考虑具有 5 次多项式的贝叶斯线性回归问题。我们选择先验参数 $p(\theta) = N(0, 1)$ 。图 9.3 可视化了由该参数先验诱导的函数的诱导先验分布 (阴影区域深灰色:67% 置信区间;浅灰色:95% 置信区间),包括来自该先验的一些函数样本。

通过首先对参数向量 (\cdot) 进行采样来获得函数样本。我们使用了200 个输入 $l_{\theta_i} p(\theta)$ 然后计算 $f_i(\cdot) = \theta_i x_i \in [-5, 5]$ 。我们应用特征函数 (\cdot) 。图 9.3 中的不确定性 (由阴影区域表示)完全是由参数不确定性引起的,因为我们考虑了无噪声预测分布 (9.40)。

到目前为止,我们研究了使用参数先验 $p(\theta)$ 来计算预测。然而,当我们有一个后验参数 (给定一些训练数据 X, Y)时,与 (9.37) 中的预测和推理原理相同 我们只需要将先验 $p(\theta)$ 替换为后验

$p(\theta | X, Y)$ 。在下文中，我们将在使用它进行预测之前详细推导后验分布。

9.3.3 后验分布

给定一组训练输入 $x_n \in RD$ 和相应的观测值 N ，我们计算参数 $y_n \in R, n = 1, \dots$ 的后验。…，使用贝叶斯定理作为

$$p(\theta | X, Y) = \frac{p(Y | X, \theta)p(\theta)}{p(Y | X)}, \quad (9.41)$$

其中 X 是训练输入集， Y 是相应训练目标的集合。此外， $p(Y | X, \theta)$ 是可能性， $p(\theta)$ 是先验参数，以及

$$p(Y | X) = p(Y | X, \theta)p(\theta)d\theta = E_\theta[p(Y | X, \theta)] \quad (9.42)$$

marginal likelihood 边际似然/证据，它独立于参数 θ 并确保后验被归一化，即它积分为 1。我们可以将边际似然视为所有可能参数设置的平均似然（相对于先验分布 $p(\theta)$ ）。

边际似然是

参数先验下的预期可能性。

定理 9.1（后验参数）。在我们的模型(9.35)中，参数后验(9.41)可以以封闭形式计算为

$$p(\theta | X, Y) = N \theta | mN, \quad SN, \quad (9.43a)$$

$$SN = (\text{小号}_0^{-1} + \sigma^2 - 2\Phi \Phi) - 1, \quad (9.43b)$$

$$mN = SN (S_0^{-1} m_0 + \sigma^2 - 2\Phi y), \quad (9.43c)$$

其中下标 N 表示训练集的大小。

证明贝叶斯定理告诉我们后验 $p(\theta | X, Y)$ 与似然 $p(Y | X, \theta)$ 和先验 $p(\theta)$ 的乘积成正比：
 $p(Y | X, \theta) p(\theta)$

$$\text{后验 } p(\theta | X, Y) = p(Y | X, \theta) = \frac{1}{N y | \Phi \theta, \sigma^2 |} \quad (9.44a)$$

$$m_0, S_0. \quad \text{似然 } p(Y | X) p(\theta) = N \theta | \quad (9.44b)$$

$$\text{先验的} \quad (9.44c)$$

我们可以将问题转换为对数空间，并通过完成平方来求解后验的均值和协方差，而不是查看先验和似然的乘积。

$$\text{对数先验和对数似然之和为 } \log N y | \Phi \theta, \sigma^2 | + \log N \theta | m_0, S_0 \quad (9.45a)$$

$$= -\frac{1}{2\sigma^2} (y - \Phi \theta)^T (y - \Phi \theta) + (\theta - m_0)^T S_0^{-1} (\theta - m_0) + \text{const} \quad (9.45b)$$

其中常数包含独立于 θ 的项。下面我们将忽略常量。我们现在对(9.45b)进行因式分解,得到

$$-\frac{1}{2} \sigma^2 - 2y \cdot y - 2\sigma^2 - 2y \cdot \Phi\theta + \theta^2 = \sigma^2 - 2\Phi \cdot \Phi\theta + \theta^2 - S^{-1}\theta \quad (9.46a)$$

$$= -\frac{1}{2} \theta^2 - (\sigma^2 - 2\Phi \cdot \Phi + S^{-1})\theta - 2(\sigma^2 - 2\Phi \cdot y + S^{-1}m_0) \cdot \theta + \text{const} \quad , \quad (9.46b)$$

其中常数包含(9.46a)中的黑色项,它们与 θ 无关。橙色项是 θ 的线性项,蓝色项是 θ 的二次项。检查(9.46b),我们发现这个方程是 θ 的二次方程。非归一化对数后验分布是(负)二次型这一事意味着后验分布是高斯分布的,即

$$p(\theta | X, Y) = \exp(\log p(\theta | X, Y)) \propto \exp(\log p(Y | X, \theta) + \log p(\theta)) \quad (9.47a)$$

$$\propto \exp -\frac{1}{2} \theta^2 - (\sigma^2 - 2\Phi \cdot \Phi + S^{-1})\theta - 2(\sigma^2 - 2\Phi \cdot y + S^{-1}m_0) \cdot \theta + \text{const} \quad , \quad (9.47b)$$

我们在最后一个表达式中使用(9.46b)的地方。

剩下的任务是将这个(未归一化的)高斯函数转化为与 $N \theta | mN, SN$ 成正比的形式,即,我们需要确定均值 mN 和协方差矩阵 SN 。为此,我们使用完成正方形的概念。所需的对数后验是

$$\log N \theta | mN, \quad SN = -\frac{1}{2} (\theta - mN)^T S N (\theta - mN) + \text{const} \quad (9.48a)$$

$$= -\frac{1}{2} \theta^T S^{-1} \theta - 2m^T S^{-1} \theta - 2m^T S^{-1} m + \text{const} \quad (9.48b)$$

在这里,我们分解了二次形式 $(\theta - mN)^T S$ 项,它在 θ 中是二次的(蓝 $\bar{N}^1(\theta - mN)$ 转化为 a 因为 $p(\theta | X, Y) = N$ 色),在 θ 中是线性的(橙色),和常数项(黑色)。这允许我们现在通过匹配(9.46b)和(9.48b)中的彩色表达式来找到 SN 和 mN ,这产生

$$\frac{1}{2} \theta^T S^{-1} \theta = \Phi^T I \Phi + S^{-1} \theta^T \theta \quad (9.49a)$$

$$\Leftrightarrow SN = (\sigma^2 - 2\Phi^T I \Phi + S^{-1})^{-1} \quad (9.49b)$$

和

$$m^T S^{-1} \theta = (\sigma^2 - 2\Phi^T I \Phi + S^{-1})^{-1} \theta^T \theta \quad (9.50a)$$

$$\Leftrightarrow mN = SN (\sigma^2 - 2\Phi^T I \Phi + S^{-1})^{-1} \theta^T \theta \quad (9.50b)$$

□

备注（完成方块的一般方法）。如果给我们一个等式

$$x - Ax - 2a = x + \text{const1} , \quad (9.51)$$

其中A是对称且正定的，我们希望将其带入该形式

$$(x - \mu) = \Sigma(x - \mu) + \text{const2} , \quad (9.52)$$

我们可以通过设置来做到这一点

$$\Sigma := \text{一个} , \quad (9.53)$$

$$\mu := \Sigma^{-1} \uparrow \quad (9.54)$$

$$\text{const2} = \text{const1} - \mu - \Sigma\mu . \quad \diamond$$

我们可以看到 (9.47b) 中的指数项是
表格 (9.51) 与

$$A := \sigma - 2\Phi \Phi + S^{-1} \quad (9.55)$$

$$a := \sigma - 2\Phi y + \text{小号} S^{-1} \uparrow 0 . \quad (9.56)$$

由于A,a在像 (9.46a) 这样的方程式中可能很难识别，因此将这些方程式转化为 (9.51) 的形式非常有帮助，这种形式可以解耦二次项、线性项和常数，从而简化寻找所需解的过程。

9.3.4 后验预测

在 (9.37) 中，我们使用参数先验 $p(\theta)$ 计算了 y 在测试输入 x 处的预测分布。原则上，考虑到在我们的共轭模型中，先验和后验都是高斯分布的（具有不同的参数），使用参数后验 $p(\theta | X, Y)$ 进行预测并没有根本的不同。因此，通过遵循与第 9.3.2 节相同的推理，我们获得（后验）预测分布

$$p(y^* | X, Y, x^*) = p(y^* | x^*, \theta) p(\theta | X, Y) d\theta \quad (9.57a)$$

$$= N y^* | \Phi(x^*) \theta, \sigma^2 N \theta | mN, SN d\theta \quad (9.57b)$$

$$= N y^* | \Phi(x^*) mN, (x^*) SN \phi(x^*) + \sigma^2 . \quad (9.57c)$$

$E[y^* | X, Y, x^*] =$
 $(x^*) mN =$
 $(x^*) \theta_{MAP}$

带有参数 θ 的 $(x^*) SN \phi(x^*)$ 反映了相关的后验不确定性项。请注意， SN 取决于训练输入 $(x^*) mN$ 与通过 Φ ；参见 (9.43b)。预测均值是用 MAP 估计 θ_{MAP} 做出的预测。

备注 (边际似然和后验预测分布)。通过替换 (9.57a) 中的积分,预测分布可以等价地写为期望 $E\theta | X, Y [p(y | x, \theta)]$, 其中期望值是关于参数后验 $p(\theta | X, Y)$ 的。

以这种方式编写后验预测分布突出显示与边际似然 (9.42) 非常相似。边际似然和后验预测分布之间的主要区别是 (i) 边际似然可以被认为是预测训练目标 y 而不是测试目标 y^* , 以及 (ii) 边际似然关于参数的平均值先验而不是后验参数。 ◇ 备注 (无噪声函数值的均值和方差)。在许多情况下, 我们对 (嘈杂的) 观察值 y 的预测分布 $p(y | X, Y, x)$ 不感兴趣。相反, 我们希望通过利用均值和方差的特性来获得 (无噪声) 函数值 $f(x^*) = \text{对应矩的分布}$, 这会产生

$(x^*)\theta$ 。我们确定

$$\begin{aligned} E[f(x^*) | X, Y] &= E\theta[\phi(x^*)\theta | X, Y] = \phi(x^*)E\theta[\theta | X, Y] \\ &= \phi(x^*)mN = mN(x^*), \end{aligned} \quad (9.58)$$

$$\begin{aligned} V\theta[f(x^*) | X, Y] &= V\theta[\phi(x^*)\theta | X, Y] \\ &= \phi(x^*)V\theta[\theta | X, Y](x^*) \\ &= \phi(x^*)SN\phi(x^*). \end{aligned} \quad (9.59)$$

我们看到预测均值与噪声观察的预测均值相同, 因为噪声均值为 0, 预测方差仅相差 σ^2 , 即测量噪声的方差: 当我们预测噪声函数值时, 我们需要将 σ^2 作为不确定性来源, 但无噪声预测不需要此项。在这里, 唯一剩下的不确定性来自后验参数。 ◇ 备注 (函数分布)。我们对参数 θ 进行积分的事实导致函数分布: 如果我们从参数后验中采样 $\theta_i \sim p(\theta | X, Y)$, 我们将获得单个函数实现 θ

整合出参数会导致
分布

功能。

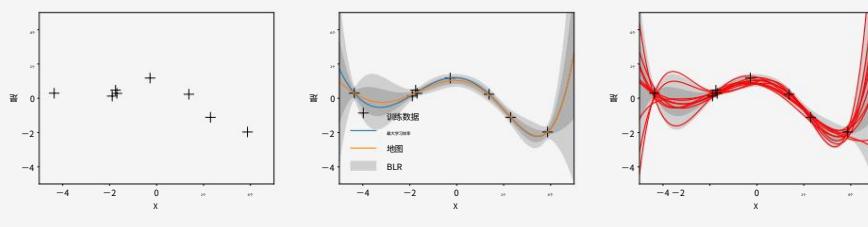
例 9.8 (函数后验)
让我们重新审视具有 5 次多项式的贝叶斯线性回归问题。我们选择先验参数 $p(\theta) = N(0, I)$, 图 9.3 可视化了由参数先验和来自该先验的示例函数引起的先验函数。

图 9.4 显示了我们通过贝叶斯线性回归获得的后验函数。训练数据集显示在面板 (a) 中;面板 (b) 显示了函数的后验分布,包括我们将通过最大似然和 MAP 估计获得的函数。

我们使用 MAP 估计获得的函数也对应于贝叶斯线性回归设置中的后验均值函数。面板 (c) 显示了在该函数后置函数下的一些似是而非的函数实现 (样本)。

图 9.4 贝叶斯
线性回归和后验函
数。(a) 训练数
据; (b) 后验分
布

功能; (c) 来
自后验函数的样本。



(a) 训练数据。

(b) 后验函数 rep
(c) 来自后验函数的样本被边缘不确定函数所反对,这些样本
处于污点 (阴影)中,显示了来自 67% 和 95% 预测 con 参数后验样本的推
断。置信区间、最大似然估计 (MLE) 和 MAP 估计 (MAP),后者与后验均值函数
相同。

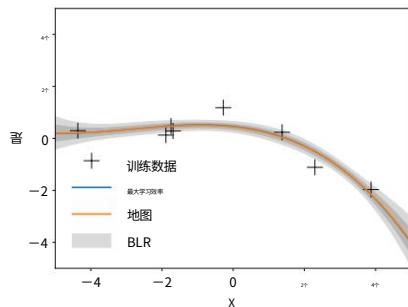
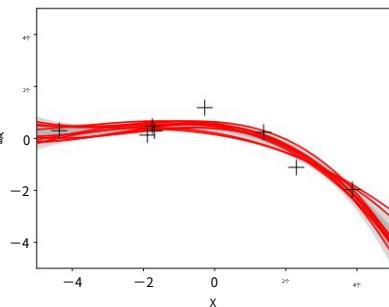
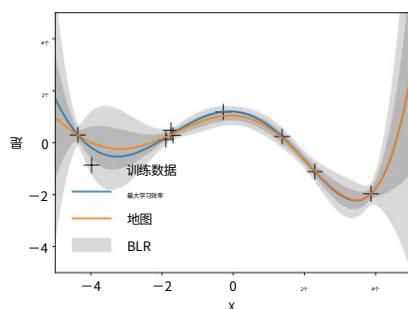
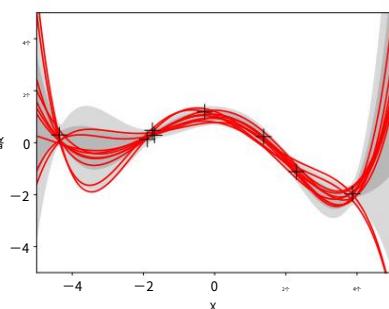
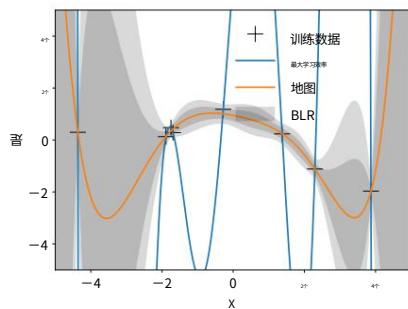
图 9.5 显示了一些由参数后验引起的函数的后验分布。对于不同的多项式次数 M , 左侧面板显示最大似然函数 $\theta_{ML}(\cdot)$, MAP 函数

$\theta_{MAP}(\cdot)$ (与后验均值函数相同), 以及通过贝叶斯线性回归获得的 67% 和 95% 预测置信区间,由阴影区域表示。

右侧面板显示来自后验函数的样本:在这里,我们从参数后验中采样参数 θ_i 并计算函数 $\theta(x_i)$, 这是函数后验分布下的函数的单一实现。对于低阶多项式,后验参数不允许参数变化太大:采样函数几乎相同。当我们通过添加更多参数使模型更灵活时(即,我们以高阶多项式结束),这些参数没有充分地受到后验约束,并且采样函数可以很容易地在视觉上分开。我们还在左侧的相应面板中看到不确定性如何增加,尤其是在边界处。

尽管对于七阶多项式,MAP 估计产生合理的拟合,但贝叶斯线性回归模型还告诉我们

9.3 贝叶斯线性回归

(a) $M = 3$ 次多项式的后验分布 (左)和来自后验函数的样本 (右)。(a) $M = 3$ 次多项式的后验分布 (左)和来自后验函数的样本 (右)。(b) $M = 5$ 次多项式的后验分布 (左)和来自后验函数的样本 (右)。(b) $M = 5$ 次多项式的后验分布 (左)和来自后验函数的样本 (右)。(c) $M = 7$ 次多项式的后验分布 (左)和来自后验函数的样本 (右)。图 9.5 贝叶斯
线性回归。左面板:阴
影区域表示

67% (深灰色)和
95% (浅灰色)
预测置信区间。

的平均值
贝叶斯线性回归模
型与 MAP 估计一致。
这

预测不确定
性是噪声项和

后验参数不确定性,这取决
于测试输入的位置。右
图:来自后验分布的
采样函数。

后验不确定性很大。当我们在决策系统中使用这些预测时,这些信息可能很重要,
在决策系统中,错误的决策可能会产生重大后果 (例如,在强化学习或机器人技
术中)。

9.3.5 计算边际似然在 8.6.2 节中,我们强调了边际似然对

贝叶斯模型选择的重要性。在下文中,我们使用参数的共轭高斯先验计算贝叶斯线性回归的边际似然,即,正是我们在本章中讨论的设置。

回顾一下,我们考虑以下生成过程:

$$\theta \sim N(m_0, S_0) \quad (9.60a)$$

$$y | x_n, \theta \sim N(x_n, \sigma^2), \quad (9.60b)$$

边际似然可以是

N. 边际似然由 $n = 1$, 给出。 . . ,

解释为先验下的预期似然,即 $E[\theta | p(Y | X, \theta)]$ 。

$$p(Y | X) = p(Y | X, \theta)p(\theta)d\theta \quad (9.61a)$$

$$= \int \theta | X \sim N(m_0, S_0) d\theta, \quad (9.61b)$$

我们整合了模型参数 θ 。我们分两步计算边际似然:首先,我们证明边际似然是高斯分布的(作为 y 中的分布);其次,我们计算这个高斯分布的均值和协方差。

1. 边际似然是高斯分布的:从 6.5.2 节我们知道 (i) 两个高斯随机变量的乘积是一个(未归一化的)

高斯分布,以及 (ii) 高斯随机变量的线性变换是高斯分布的。在 (9.61b) 中,我们需要一个线性变换来使 $N(y | X\theta, \sigma^2)$ 化为 $N(\theta | \mu, \Sigma)$ 。一旦完成,积分就可以以封闭形式求解。

结果是两个高斯分布的乘积的归一化常数。归一化常数本身具有高斯形状;参见 (6.76)。

2. 均值和协方差。我们通过利用随机变量仿射变换的均值和协方差的标准结果来计算边际似然的均值和协方差矩阵;请参阅第 6.4.4 节。边际似然的均值计算如下

$$E[Y | X] = E[\theta + \epsilon | X] = XE[\theta | X] = Xm_0. \quad (9.62)$$

请注意, $\epsilon \sim N(0, \sigma^2)$ 是 iid 随机变量的向量。协方差矩阵为

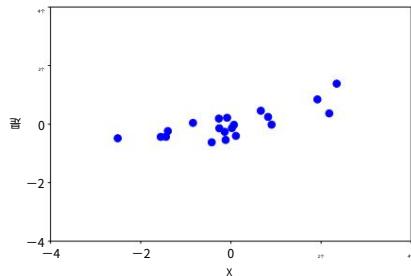
$$\text{Cov}[Y | X] = \text{Cov}[\theta + \epsilon | X] = \text{Cov}[\theta | X] + \text{Cov}[\epsilon | X] \quad (9.63a)$$

$$= X \text{Cov}[\theta | X] X^T + \sigma^2 I \quad (9.63b)$$

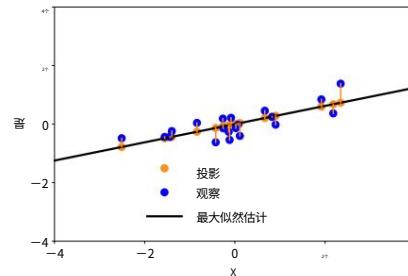
因此,边际似然是

$$p(Y | X) = (2\pi)^{-n/2} \det(XS_0X^T + \sigma^2 I)^{-1/2} \exp -\frac{1}{2} (y - Xm_0)^T (XS_0X^T + \sigma^2 I)^{-1} (y - Xm_0) \quad (9.64a)$$

9.4 作为正交投影的最大似然法



(a) 回归数据集由输入位置 x_n 处函数值 $f(x_n)$ 的噪声观测值 y_n (蓝色)组成。



(b) 橙色点是噪声观测值 (蓝色点) 在 $\theta_{ML}x$ 线上的投影。线性回归问题的最大似然解找到一个子空间 (线) , 在该子空间 (线) 上观测值的整体投影误差 (橙色线) 最小化。

图 9.6 最小二乘法的几何解释。(a) 数据集; (b) 最大似然解
解释为投影。

$$= \text{否} | X\theta_0, X\theta_0X + \sigma^2 \text{ 我。} \quad (9.64b)$$

鉴于与后验预测分布的密切联系 (参见本节前面关于边际似然和后验预测分布的备注) , 边际似然的函数形式应该不会太令人惊讶。

9.4 作为正交投影的最大似然法在通过大量代数推导出最大似然

法和 MAP 估计之后,我们现在将提供最大似然估计的几何解释。让我们考虑一个简单的线性回归设置

$$y = x\theta + \varepsilon, \varepsilon \sim N(0, \sigma^2) \quad (9.65)$$

其中我们考虑通过原点的线性函数 $f: \mathbb{R} \rightarrow \mathbb{R}$ (为了清楚起见, 我们在这里省略了特征) 。参数 θ 决定直线的斜率。图 9.6(a) 显示了一个一维数据集。

使用训练数据集 $\{(x_1, y_1), \dots, (x_N, y_N)\}$ 我们回顾第 9.2.1 节的结果并获得斜率参数的最大似然估计量为

$$\theta_{ML} = (X^T X)^{-1} X^T y = \frac{\bar{X}^T \bar{y}}{\bar{X}^T \bar{X}} \in \mathbb{R}, \quad (9.66)$$

其中 $X = [x_1, \dots, x_N] \in \mathbb{R}^N$, $y = [y_1, \dots, y_N] \in \mathbb{R}^N$ 。

这意味着对于训练输入 X , 我们获得训练目标的最优 (最大似然) 重建为

$$X\theta_{ML} = X \frac{\bar{X}^T \bar{y}}{\bar{X}^T \bar{X}} = \frac{X\bar{X}^T}{\bar{X}^T \bar{X}} y, \quad (9.67)$$

即,我们获得了 y 和 $X\theta$ 之间具有最小最小二乘误差的近似值。

当我们正在寻找 $y = X\theta$ 的解时,我们可以将线性回归视为求解线性方程组的问题。因此,我们可以将我们在第 2 章和第 3 章中讨论的线性代数和解析几何的概念联系起来。特别是,仔细观察 (9.67),我们发现(9.65)示例中的最大似然估计量 θ_{ML} 有效地完成了 y 在 X 所跨的一维子空间上的正交投影。回顾 $X\theta$ 在 X 上的结果

线性回归可以被认为是一种求解线性方程组的方法。

最大限度
似然线性回归执行
正交投影。

来自第 3.8 节的正交投影,我们将 θ_{ML} 标识为投影矩阵, θ_{ML} 是在 X 所跨越的 RN 的一维子空间上的投影坐标, $X\theta_{ML}$ 是 y 在该子空间上的正交投影。

因此,最大似然解也提供了几何最优解,方法是在 X 所跨越的子空间中找到“最接近”相应观测值 y 的向量,其中“最接近”是指函数值的最小(平方)距离 y_n 到 $x_n\theta$ 。这是通过正交投影实现的。图 9.6(b) 显示了噪声观察到子空间的投影,该子空间最小化原始数据集与其投影之间的平方距离(注意 x 坐标是固定的),这对应于最大似然解。

在一般的线性回归情况下

$$y = \Phi(x)\theta + \varepsilon, \varepsilon \sim N(0, \sigma^2) \quad (9.68)$$

使用向量值特征 $(x) \in RK$,我们可以再次解释最大似然结果

$$y \approx \Phi\theta_{ML}, \quad (9.69)$$

$$\theta_{ML} = (\Phi^\top \Phi)^{-1} \Phi^\top y \quad (9.70)$$

作为特征矩阵中的列在 RN 的 K 维子空间上的投影;请参阅第 3.8.2, 这是跨越节。

如果我们用来构造特征矩阵 Φ 的特征函数 k 是正交的(见第 3.7 节), 我们得到一个特殊情况, 其中 Φ 的列形成一个正交基(见第 3.5 节), 这样 $\Phi^\top \Phi = I$ 。这将导致投影

$$\Phi(\Phi^\top \Phi)^{-1} \Phi^\top y = \Phi \Phi^{-1} \Phi^\top y = \sum_{k=1}^K \phi_k \phi_k^\top y \quad (9.71)$$

因此最大似然投影只是 y 在各个基向量 ϕ_k 上的投影之和, 即 Φ 的列。此外, 由于基的正交性, 不同特征之间的耦合已经消失。信号处理中许多流行的基函数, 例如小波和傅里叶基, 都是正交基函数。

当基不正交时,可以使用Gram-Schmidt过程将一组线性无关的基函数转换为正交基;参见第3.8.3节和(Strang,2003年)。

9.5 进一步阅读在本章中,我

们讨论了高斯似然的线性回归和模型参数的共轭高斯先验。这允许封闭形式的贝叶斯推理。然而,在某些应用中,我们可能希望选择不同的似然函数。例如,在二元分类设置中,我们仅观察到两种可能的(分类)分类结果,而高斯似然在此设置中是不合适的。相反,我们可以选择将返回预测标签概率为1(或0)的伯努利似然。我们参考了Barber(2012)、Bishop(2006)和Murphy(2012)的书籍,以深入介绍分类问题。非高斯可能性很重要的另一个例子是计数数据。计数是非负整数,在这种情况下,二项式或泊松似然比高斯似然是更好的选择。

所有这些示例都属于广义线性模型的类别,这是线性回归的灵活广义线性泛化,允许具有除高斯分布之外的误差分布的响应变量。GLM Generalized linear通过允许线性模型通过平滑且可逆的函数 $\sigma(\cdot)$ 与观测值相关联来概括线性回归,该函数可能是非线性的,因此 $y = \sigma(f(x))$,其中 $f(x) = \theta^T x$ 是(9.13)中的线性回归模型。因此,我们可以根据函数组合 $y = \sigma \circ f$ 来考虑广义线性模型,其中 f 是线性回归模型, σ 是激活函数。请注意,虽然我们谈论的是“广义线性模型”,但输出 y 在参数 θ 中不再是线性的。在逻辑回归中,我们选择逻辑回归 logistic sigmoid $\sigma(f) = \frac{1}{1 + \exp(-f)}$,可以解释为 logistic sigmoid 观察到伯努利随机变量 $y \in \{0, 1\}$ 的 $y = 1$ 的概率。

$$\frac{1}{1 + \exp(-f)}$$

函数 $\sigma(\cdot)$ 称为传递函数或激活函数,其传递函数的逆函数称为规范链接函数。从这个角度来看,激活函数也很清楚,广义线性模型是(深度)前馈神经网络的构建块:如果我们考虑广义线性模型 $y = \sigma(Ax + b)$,其中 A 是权重矩阵, b 作为偏置向量,我们将这个广义线性模型识别为具有激活函数 $\sigma(\cdot)$ 的单层神经网络。我们现在可以递归地组合这些函数

对于普通的线性回归,激活函数就是恒等式。
规范链接功能

通过

$$\begin{aligned} x_{k+1} &= f_k(x_k) \\ f_k(x_k) &= \sigma_k(A_k x_k + b_k) \end{aligned} \tag{9.72}$$

对于 $k = 0, \dots, K - 1$,其中 x_0 是输入特征, $x_K = y$ 是观察到的输出,因此 f 是 K 层深度神经网络。因此,这个深度神经网络的构建块是 $f = f_{K-1} \circ \dots \circ f_1$

一篇关于GLM和深度网络之间关系的好文章可以在

<https://tinyurl.com/glm-dnn>

(9.72) 中定义的广义线性模型。神经网络 (Bishop, 1995 年; Goodfellow 等人, 2016 年) 比线性回归模型更具表现力和灵活性。然而, 最大似然参数估计是一个非凸优化问题, 完全贝叶斯设置中的参数边缘化在分析上是难以处理的。

高斯过程 我们简要地暗示了一个事实, 即参数的分布推导了回归函数的分布。高斯过程 (Rasmussen 和 Williams, 2006 年) 是回归模型, 其中函数分布的概念是核心。高斯过程不是在参数上放置分布, 而是直接在函数空间上放置分布, 而无需通过参数 “绕行”。为此, 高斯过程利用了内核技巧 (Schölkopf 和 Smola, 2002 年), 它允许我们仅通过查看相应的输入 x_i 和 x_j 来计算两个函数值 $f(x_i)$ 、 $f(x_j)$ 之间的内积, $\langle f(x_i), f(x_j) \rangle$ 。该过程与贝叶斯线性回归和支持向量回归密切相关, 但也可以解释为具有单个隐藏层的贝叶斯神经网络, 其中单元数趋于无穷大 (Neal, 1996; Williams, 1997)。在 MacKay (1998) 和 Rasmussen 和 Williams (2006) 中可以找到对高斯过程的精彩介绍。

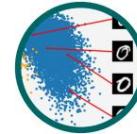
内核技巧

我们在本章的讨论中重点关注高斯参数先验, 因为它们允许在线性回归模型中进行闭式推理。然而, 即使在具有高斯似然的回归设置中, 我们也可以选择非高斯先验。考虑一个设置, 其中输入为 $x \in \mathbb{R}^D$ 并且我们的训练集很小且大小为 $N \ll D$ 。这意味着回归问题是不确定的。在这种情况下, 我们可以选择一个强制稀疏性的先验参数, 即尝试将尽可能多的参数设置为 0 的变量选择先验 (变量选择)。这个先验提供了比高斯先验更强的正则化器, 这通常会导致模型的预测准确性和可解释性提高。拉普拉斯先验是经常用于此目的的一个例子。在参数上具有拉普拉斯先验的线性回归模型等效于具有 L1 正则化 (LASSO) 的线性回归 (Tibshirani, 1996)。拉普拉斯分布在零处有一个尖锐的峰值 (它的一阶导数是不连续的) 并且它的概率质量比高斯分布更接近于零, 这鼓励参数为 0。因此, 非零参数与回归问题相关, 这这就是为什么我们也说“变量选择”的原因。

索引

10

主降维 成分分析



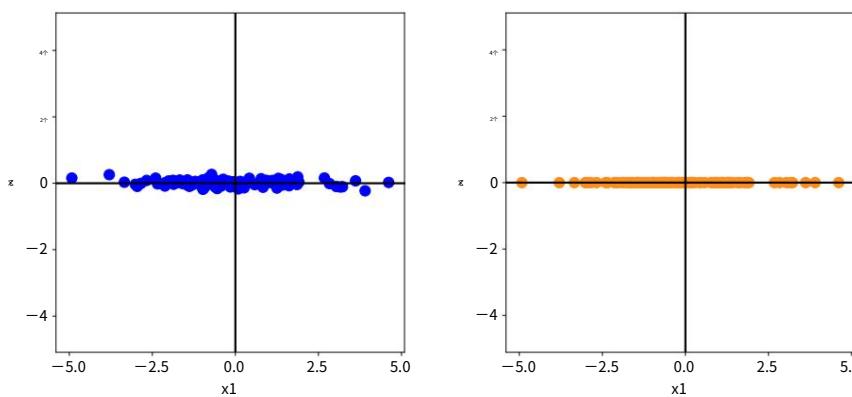
直接处理图像等高维数据会带来 640×480 像素的一些困难:难以分析、解释困难、可视化几乎不可能,并且(从实用的角度来看)存储数据向量可能很昂贵。然而,高维数据通常具有我们可以利用的属性。例如,高维数据通常是过度完备的,即许多维度是冗余的,可以通过其他维度的组合来解释。此外,高维数据中的维度通常是相关的,因此数据具有内在的低维结构。降维利用结构和相关性,使我们能够处理更紧凑的数据表示,理想情况下不会丢失信息。我们可以把降维看做是一种压缩技术,类似于jpeg或者mp3,都是图像和音乐的压缩算法。

彩色图像是百万维空间中的数据点,其中每个像素响应三个维度,一个对应一种颜色

通道(红、绿、蓝)。

在本章中,我们将讨论主成分分析(PCA),一种主成分线性降维算法。由 Pearson (1901) 和 Hotelling (1933) 提出的 PCA 已经存在了100多年,仍然是最常用的数据压缩和数据可视化技术之一。它还用于识别高维数据的简单模式、潜在因素和结构。在里面

分析
主成分分析
降维



(a) 具有x1和x2坐标的的数据集。

(b) 只有x1坐标相关的压缩数据集。

图 10.1说明:

降维。(a) 原始数据集沿x2方向变化不大。(b) (a) 中的数据可以单独使用x1坐标表示,几乎没有损失。

Karhunen-Lo`eve
转换

信号处理社区,PCA 也称为Karhunen-Lo`eve 变换。在本章中,我们根据对基和基变化(第 2.6.1 节和 2.7.2 节)、投影(第 3.8 节)、特征值(第 4.2 节)、高斯分布(第 6.5 节)的理解,从第一性原理推导出 PCA 和约束优化(第 7.2 节)。

降维通常利用高维数据(例如图像)的特性,即它通常位于低维子空间中。

图 10.1 给出了一个二维的示例。虽然图 10.1(a) 中的数据并不完全位于一条直线上,但数据在 x_2 方向上变化不大,因此我们可以将其表示为好像在一条直线上几乎没有损失;见图 10.1(b)。为了描述图 10.1(b) 中的数据,只需要 x_1 坐标,数据位于 \mathbb{R}^2 的一维子空间中。

10.1 问题设置在 PCA 中,我

们感兴趣的是找到尽可能类似于原始数据点的数据点 x_n 的投影 x_n ,但其内在维度要低得多。图 10.1 说明了这可能是什么样子。

更具体地说,我们考虑一个 iid 数据集 $X = \{x_1, \dots, x_N\}$, $x_n \in \mathbb{R}^D$ 均值为 0 具有数据协方差矩阵(6.42)

$$\text{小号} = \frac{1}{N} \sum_{n=1}^{N} x_n x_n^T. \quad (10.1)$$

此外,我们假设存在一个低维压缩表示(代码)

$$z_n = B x_n \in \mathbb{R}^M \quad (10.2)$$

x_n ,我们在这里定义投影矩阵

$$B := [b_1, \dots, b_M] \in \mathbb{R}^{D \times M}. \quad (10.3)$$

我们假设 B 的列是正交的(定义 3.7),因此

$b_i \cdot b_j = 0$ 当且仅当 $i = j$ and $b_i \cdot b_i = 1$ 。我们寻求 M 维子空间 $U \subseteq \mathbb{R}^D$, $\dim(U) = M < D$ 我们将数据投影到其上。我们用 $x_n \in U$ 表示投影数据,用 z_n 表示它们的坐标(相对于 U 的 b_M)。我们的目标是找到基向量 b_1, \dots, b_M ,使它们与原始影 $\dots, x_n \in \mathbb{R}^D$ (或等价于代码 z_n 和基向量数据 x_n 相似,并使压缩损失最小化)。

示例 10.1 (坐标表示/代码)

考虑具有规范基础 $e_1 = [1, 0]$, $e_2 = [0, 1]$ 的 \mathbb{R}^2 。从

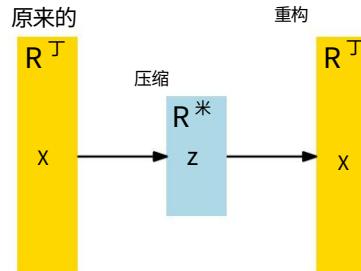


图 10.1 PCA 的图形说明。

在 PCA 中,我们找到原始数据 x 的压缩版本 z 。
压缩后的数据可以重构为

$x\sim$,它存在于原始数据空间中,但具有内在的低维

第 2 章,我们知道 $x \in R^2$ 可以表示为这些基向量的线性组合,例如,

$$\begin{matrix} 5 \\ 3 \end{matrix} = 5e_1 + 3e_2。 \quad (10.4)$$

然而,当我们考虑形式的向量时

$$x\sim = \begin{matrix} 0 \\ z \end{matrix} \in R^{\sim}, \quad z \in R^-, \quad (10.5)$$

它们总是可以写成 $0e_1 + ze_2$ 。为了表示这些向量,记住/存储 $x\sim$ 相对于 e_2 向量的坐标/代码 z 就足够了。

更准确地说, x 向量集 (具有标准向量加法和标量乘法) 形成一个向量子空间 U (参见第 2.4 节), 其中 $\dim(U) = 1$, 因为 $U = \text{span}[e_2]$ 。

表示比
 $x\sim$

的维度
向量空间对应其
基数
矢量 (见第
2.6.1 节)。

在 10.2 节中,我们将找到保留尽可能多的信息并最小化压缩损失的低维表示。

PCA 的另一种推导在第 10.3 节中给出, 我们将在其中研究最小化原始数据 x_n 及其投影 $x\sim_n$ 之间的平方重建误差 $\|x_n - x\sim_n\|^2$ 。

图 10.1 说明了我们在 PCA 中考虑的设置, 其中 z 表示压缩数据 $x\sim$ 的低维表示, 并扮演瓶颈的角色, 它控制了多少信息可以在 x 和 $x\sim$ 之间流动。在 PCA 中, 我们考虑原始数据 x 与其低维编码 z 之间的线性关系, 使得 $z = Bx \approx = Bz$ 对于合适的矩阵 B 。基于将 PCA 视为一种数据压缩技术的动机, 我们可以将图 10.1 中的箭头解释为一对表示编码器和解码器的操作。

x 和

B 表示的线性映射可以看作是一个解码器, 它将低维编码 $z \in RM$ 映射回原始数据空间 RD 。类似地, B 可以被认为是一个编码器, 它将原始数据 x 编码为低维 (压缩) 代码 z 。

在本章中, 我们将使用 MNIST 数字数据集作为重新生成的数据集。

图 10.2 来自
MNIST 的手写
数字示例

数据集。 <http://yann.lecun.com/exdb/mnist/>。



发生的例子,其中包含 60,000 个手写数字 0 到 9 的例子。每个数字都是一个大小为 28×28 的灰度图像,即它包含 784 个像素,因此我们可以将此数据集中的每个图像解释为向量 $x \in \mathbb{R}^{784}$ 。这些数字的示例如图 10.2 所示。

10.2 最大方差透视图 10.1 给出了一个二维数据

集如何使用单个坐标表示的示例。在图 10.1(b) 中,我们选择忽略数据的 x_2 坐标,因为它没有添加太多信息,因此压缩后的数据类似于图 10.1(a) 中的原始数据。我们本可以选择忽略 x_1 坐标,但压缩后的数据与原始数据非常不同,数据中的很多信息都会丢失。

如果我们将数据中的信息内容解释为数据集的“空间填充”程度,那么我们可以通过查看数据的分布来描述数据中包含的信息。从 6.4.1 节我们知道方差是数据分布的指标,我们可以推导出 PCA 作为一种降维算法,它在数据的低维表示中最大化方差以保留尽可能多的信息可能的。图 10.2 说明了这一点。

考虑到第 10.1 节中讨论的设置,我们的目标是找到一个矩阵 B (见 (10.3)), 它通过将数据投影到列 b_1, \dots, b_M 所跨越的子空间来压缩数据时保留尽可能多的信息。 \dots, b_M of B 。数据压缩后保留大部分信息相当于在低维代码中捕获了最大的方差量 (Hotelling, 1933)。

评论。(居中数据) 对于 (10.1) 中的数据协方差矩阵, 我们假设数据居中。我们可以在不失一般性的情况下做出这个假设: 让我们假设 μ 是数据的平均值。使用我们在第 6.4.4 节中讨论的方差属性, 我们得到

$$\nabla z[z] = \nabla x[B(x - \mu)] = \nabla x[Bx - B\mu] = \nabla x[Bx] - \nabla x[B\mu], \quad (10.6)$$

即低维码的方差不依赖于数据的均值。因此, 我们不失一般性地假设本节剩余部分的数据均值为 0。在此假设下, 低维代码的均值也为 0, 因为 $Ez[z] = Ex[Bx] - Ex[\mu] = 0$ 。

$$x] = \diamond$$

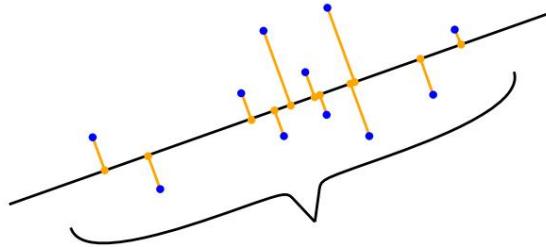


图 10.2 PCA 找到一个当数据（蓝色）投影到该子空间（橙色）时，保持尽可能多的方差（数据的分布）的低维子空间（线）。

10.2.1 方差最大的方向

我们使用顺序方法最大化低维代码的方差。我们首先寻找一个向量 $b_1 \in \mathbb{R}^D$ 来最大化向量 b_1 将投影数据的方差，即我们的目标是最大化 $z \in \mathbb{R}^M$ 的第一个坐标 z_1 的方差，使得

是矩阵 B 的第一列并且

因此第一个
 M 正交
跨越低维子空间的基向
量。

$$V_1 := V[z_1] = \frac{\text{否}}{\sum_{n=1}^{1^n} z_{1n}^2} \quad (10.7)$$

被最大化，我们利用数据的 iid 假设并将 z_{1n} 定义为 $x_n \in \mathbb{R}^D$ 的低维表示 $z_n \in \mathbb{R}^M$ 的第一个坐标。请注意， z_n 的第一个分量由下式给出

$$z_{1n} = b_1 \times n, \quad (10.8)$$

即，它是 x_n 在 b_1 所跨越的一维子空间上的正交投影坐标（第 3.8 节）。我们将 (10.8) 代入 (10.7)，得到

$$V_1 = \frac{\text{否}}{\sum_{n=1}^{1^n} (b_1 \times n)^2} = \frac{\text{否}}{\sum_{n=1}^{1^n} b_1 \times n \times b_1} \quad (10.9a)$$

$$= b_1 \frac{\text{否}}{\sum_{n=1}^{1^n} x_n \times b_1} = b_1 \times S b_1, \quad (10.9b)$$

其中 S 是 (10.1) 中定义的数据协方差矩阵。在 (10.9a) 中，我们使用了两个向量的点积关于其参数对称的事实，即 $b_1 \times n = x_n \times b_1$ 。

任意增加向量 b_1 的幅度会使 V_1 增加，也就是说，向量 b_1 长两倍可能导致 V_1 可能大四倍。因此，我们将所有解限制为 $\|b_1\| = 1$ ，这导致约束优化问题 $\Leftrightarrow \|b_1\| = 1$ 。我们寻找数据变化最大的方向。

通过将解空间限制为单位向量，可以通过以下方式找到指向最大方差方向的向量 b_1

约束优化问题

$$\begin{aligned} & \text{最大 } b_1^T S b_1 \\ & \text{受制于 } \|b_1\|_2 = 1。 \end{aligned} \quad (10.10)$$

按照第 7.2 节, 我们得到拉格朗日量

$$L(b_1, \lambda) = b_1^T S b_1 + \lambda(1 - b_1^T b_1) \quad (10.11)$$

来解决这个约束优化问题。 L 关于 b_1 和 λ 的偏导数是

$$\frac{\partial L}{\partial b_1} = 2b_1^T S - 2\lambda b_1, \quad \frac{\partial L}{\partial \lambda} = 1 - b_1^T b_1, \quad (10.12)$$

分别。将这些偏导数设置为 0 给出了关系

$$Sb_1 = \lambda b_1, \quad (10.13)$$

$$b_1^T b_1 = 1. \quad (10.14)$$

通过将其与特征值分解的定义 (第 4.4 节) 进行比较, 我们看到 b_1 是数据协方差矩阵 S 的特征向量, 而拉格朗日乘子 λ 扮演相应特征值的角色。此特征向量属性 (10.13) 允许我们将方差目标 (10.10) 重写为

量 $\sqrt{\lambda}$ 也称为单位向量
 b_1 的载荷, 表示

主子空间 $\text{span}[b_1]$ 所
占数据的标准
差。

$$V_1 = b_1^T S b_1 = \lambda b_1^T b_1 = \lambda, \quad (10.15)$$

即, 投影到一维子空间的数据的方差等于与跨越该子空间的基向量 b_1 相关联的特征值。
因此, 为了最大化低维码的方差, 我们选择与数据协方差矩阵的最大特征值相关联的基向量。这个特征向量被称为第一主成分。我们可以通过将坐标 $z_1 n$ 映射回数据空间来确定主成分 b_1 在原始数据空间中的作用/贡献, 这为我们提供了投影数据点

主成分

$$x_n = b_1 z_1 n = b_1 b_1^T n \in \mathbb{R}^m \quad (10.16)$$

在原始数据空间中。

评论。虽然 x_n 是一个 D 维向量, 但它只需要一个坐标 $z_1 n$ 来表示它相对于基向量 $b_1 \in \mathbb{R}^D$ 。 ◇

10.2.2 具有最大方差的 M 维子空间

假设我们已经找到前 $m - 1$ 个主成分作为 S 的 $m - 1$ 个特征向量, 它们与最大的 $m - 1$ 个特征值相关联。

由于 S 是对称的, 谱定理 (定理 4.15) 指出我们可以使用这些特征向量来构造一个正交特征基

RD 的 $(m - 1)$ 维子空间。一般情况下,第 m 个主成分可以通过从数据中减去第 $m-1$ 个主成分 b_{m-1} 的影响得到,从而求出主成分 b_1, \dots, b_{m-1} ,压缩剩余信息的组件。然后我们到达新的数据矩阵

$$X^{\wedge} := X - \sum_{i=1}^{m-1} b_i b_i^T = X - B_{m-1} X \quad (10.17)$$

子空间的投影矩阵。 $\dots, b_{m-1}, x_N \in RD \times N$ 包含数据点作为列矩阵 $X^{\wedge} :=$ 其中 $X = [x_1, \dots, x_N]$ 向量和 $B_{m-1} :=$ 是投影到 b_1 所跨过的
 $[x^{\wedge} 1, \dots, x^{\wedge} N] \in RD \times N$ 在 (10.17) 中包含

备注 (符号)。在本章中,我们没有遵循收集数据 x_1, \dots, x_N 作为数据矩阵的行,但我们将它们定义为 X 的列。这意味着我们的数据矩阵

数据中尚未压缩的信息。

matrix X 是一个 $D \times N$ 矩阵,而不是常规的 $N \times D$ 矩阵。这

我们选择的原因是代数运算可以顺利进行,无需转置矩阵或将向量重新定义为左乘到矩阵的行向量。 ◇

为了找到第 m 个主成分,我们最大化方差

$$V_m = V[z_m] = \frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2 = \frac{1}{n} \sum_{i=1}^n (b_m x_i^T)^2 = b_m^T b_m, \quad (10.18)$$

服从 // $b_m // = 1$, 我们遵循与 (10.9b) 中相同的步骤并将 S^{\wedge} 定义为转换数据集 $X^{\wedge} := \{x^{\wedge} 1, \dots, x^{\wedge} N\}$ 。如前所述,当我们单独查看第一个主成分时,我们解决了一个约束优化问题,并发现最优解 b_m 是与 S^{\wedge} 的最大特征值相关联的 S^{\wedge} 的特征向量。

事实证明 b_m 也是 S 的特征向量。更一般地, S 和 S^{\wedge} 的特征向量集是相同的。由于 S 和 S^{\wedge} 都是对称的,我们可以找到特征向量的ONB (谱定理4.15), 即 S 和 S^{\wedge} 都存在 D 个不同的特征向量。接下来, 我们证明 S 的每个特征向量都是 S^{\wedge} 的特征向量。假设我们已经有 S^{\wedge} 的 b_{m-1} 。考虑 S 的特征向量 b_i , 找到特征向量 b_1, \dots, b_{m-1} , 即 $S b_i = \lambda_i b_i$ 。

一般来说,

$$\begin{aligned} S b_m &= \frac{1}{n} X^{\wedge} X^{\wedge T} b_m = \frac{1}{n} (X - B_{m-1} X)(X - B_{m-1} X)^T b_m \\ &= (S - S B_{m-1} - B_{m-1} S + B_{m-1} S B_{m-1}) b_m. \end{aligned} \quad (10.19a)$$

我们区分两种情况。如果 $i < m$, 即 b_i 是不在前 $m-1$ 个主成分中的特征向量, 则 b_i 与前 $m-1$ 个主成分正交且 $B_{m-1} b_i = 0$ 。如果 $i > m$, 即, b_i 是前 $m-1$ 个主成分中的一个, 则 b_i 是一个基向量

B_{m-1} 投射到的主子空间。自 b_1 以来， \dots, b_{m-1} 是这个主子空间的ONB，我们得到 $B_{m-1}b_i = b_i$ 。这两种情况可以概括如下：

$$B_{m-1}b_i = b_i \text{ 如果 } i < m, \quad B_{m-1}b_i = 0 \text{ 如果 } i = m. \quad (10.20)$$

在 $i < m$ 的情况下，通过在 (10.19b) 中使用 (10.20)，我们得到 $Sb^*_{\perp} = (S - B_{m-1}S)b_i = Sb_i = \lambda_i b_i$ ，即， b_i 也是 S^* 的特征向量，具有特征值 λ_i 。具体来说，

$$Sb^*_{\perp} = Sb_m = \lambda_m b_m. \quad (10.21)$$

方程 (10.21) 表明 b_m 不仅是 S 的特征向量，还是 S^* 的特征向量。具体来说， λ_m 是 S^* 的最大特征值， λ_m 是 S 的第 m 个最大特征值，并且两者都有关联的特征向量 b_m 。

在 $i < m$ 的情况下，通过在 (10.19b) 中使用 (10.20)，我们得到 $Sb^*_{\perp} = (S - SB_{m-1} - B_{m-1}S + B_{m-1}SB_{m-1})b_i = 0 = 0b_i$ (10.22) b_{m-1}

跨越 S^* 的零空间。与特征 b_i 也是 S^* 的特征向量，但它们为这意味着 b_1, \dots, b_{m-1} 是 S^* 的特征向量。但是，如果 S 相关的向量是 $\{m \dots, \text{总的来说， } S\}$ 的每个特征向量也 $- 1\}$ 维主子空间的一部分，则 S^* 的相关特征值

这个推导表明有

之间的亲密联系

为 0。

将关系 (10.21) 和 b 注入第 m 个主成分 $mb_m = 1$ ，数据的方差

是

$$Vm = b_m Sb_m \stackrel{(10.21)}{=} \lambda_m b_m \quad mb_m = \lambda_m. \quad (10.23)$$

这意味着当投影到 M 维子空间时，数据的方差等于与数据协方差矩阵的对应特征向量关联的特征值之和。

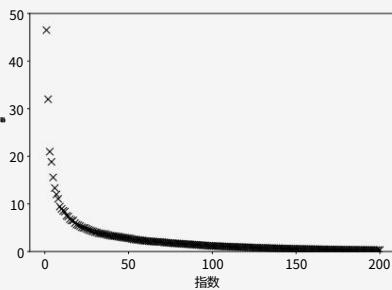
连接在
第 10.4 节。

示例 10.2 (MNIST “8”的特征值)

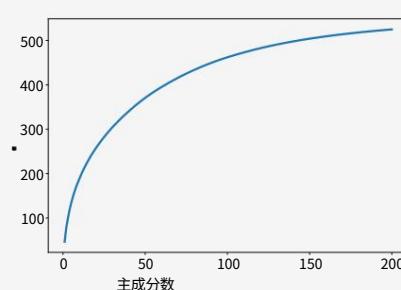
图 10.3

MNIST “8”训练数据的属性。(A)

特征值按降序排列；(b) 与最大特征值相关的主成分捕获的方差。



(a) MNIST 训练集中所有数字“8”的数据协方差矩阵的特征值（降序排列）。



(b) 主成分捕获的差异。

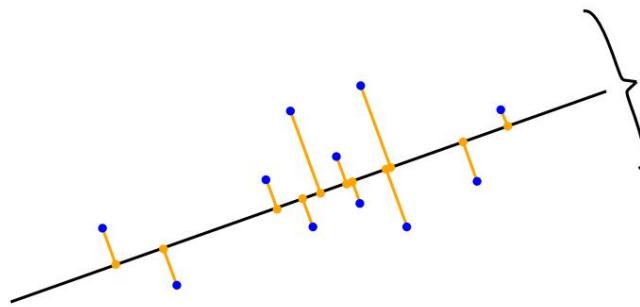


图 10.1 的图示

投影方法:找
到一个子空间(线),
使投影(橙色)和原始(蓝色)
数据之间的差
向量的长度最
小化。

取 MNIST 训练数据中的所有数字“8”,我们计算数据协方差矩阵的特征值。图 10.3(a) 显示了数据协方差矩阵的 200 个最大特征值。我们看到其中只有少数具有与 0 显著不同的值。因此,当将数据投影到相应特征向量所跨越的子空间时,大部分方差仅由少数几个主成分捕获,如图所示 10.3(b)。

总体而言,为了找到保留尽可能多信息的 RD 的 M 维子空间,PCA 告诉我们选择 (10.3) 中矩阵 B 的列作为与 M 相关联的数据协方差矩阵 S 的 M 个特征向量最大的特征值。PCA 可以用前 M 个主成分捕获的最大方差量是

$$\text{虚拟机} = \sum_{m=1}^M \lambda_m, \quad (10.24)$$

其中 λ_m 是数据协方差矩阵 S 的 M 个最大特征值。因此,通过 PCA 进行数据压缩所损失的方差为

$$\text{杰姆} := \sum_{j=M+1}^D \lambda_j = VD - VM. \quad (10.25)$$

除了这些绝对数量,我们可以将捕获的相对方差定义为,压缩损失的相对方差定义为 $1 - \frac{\text{虚拟机}}{VD}$,

10.3 投影透视

在下文中,我们将推导 PCA 作为一种直接最小化平均重构误差的算法。这种观点使我们能够将 PCA 解释为实现最佳线性自动编码器。我们将大量借鉴第 2 章和第 3 章的内容。

在上一节中,我们通过最大化投影空间中的方差来导出 PCA,以保留尽可能多的信息。在里面

326

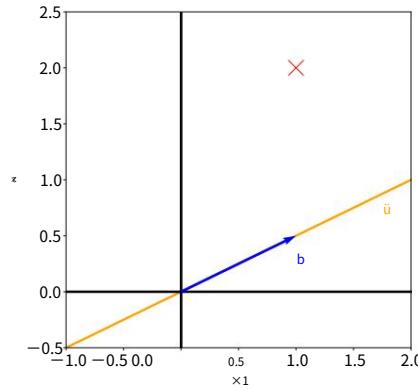
通过主成分分析降维

图 10.2 简化的
投影设置。

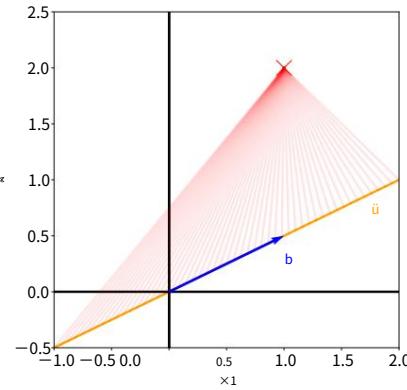
(a) 向量 $x \in R^2$ (红叉)
应投影到由 b 跨越的一维
子空间 $U \subseteq R^2$ 。(b) 显
示差异

x 和 some 之间的向量

候选人 x_{\sim}



(a) 设置。

(b) 50 个不同的 $x_{\sim i}$ 的差异 $x - x_{\sim i}$ 用红线表示。

接下来,我们将查看原始数据 x_n 和它们的重建 $x_{\sim n}$ 之间的差异数量,并最小化这个距离,
使 x_n 和 $x_{\sim n}$ 尽可能接近。图 10.1 说明了此设置。

10.3.1 设置和目标假设一个 (有序

的) 正交基 (ONB) $B = (b_1, \dots, b_D)$ of R^D , 即 b_j 从第 2.5 节我们知道对于基 (b_1, \dots, b_D) 的 $b_j = 1$ 当且仅当 $j = i$ 否则为 0。

R^D 中任何 $x \in R^D$ 可以写成 R^D 的基向量的线性组合, 即

向量 $x_{\sim} \in U$ 可以
是一个向量
 R^3 中的平面。平面的维
数是 2, 但是向量仍然有

三个坐标相对于 R^3 的
标准基础。

$$x = \sum_{d=1}^D \zeta_d b_d = \sum_{m=1}^M \zeta_m b_m + \sum_{j=M+1}^D \zeta_j b_j \quad (10.26)$$

对于合适的坐标 $\zeta_d \in R$ 。

我们感兴趣的是找到向量 $x_{\sim} \in R^D$, 它存在于低维子空间 $U \subseteq R^D$, $\dim(U) = M$,
因此

$$x_{\sim} = \sum_{m=1}^M z_m b_m \in U \subseteq R^D \quad (10.27)$$

尽可能与 x 相似。请注意, 此时我们需要假设 x 的坐标 z_m 和 x 的 ζ_m 不相同。

在下文中, 我们正是使用 x 的这种表示来找到最佳坐标 z 和基向量 b_1, \dots, b_M 使得
 x_{\sim} 尽可能与原始数据点 x 相似, 即我们的目标是最小化 (欧几里得) 距离 $\|x - x_{\sim}\|$ 。
图 10.2 说明了此设置。

不失一般性, 我们假设数据集 $X = \{x_1, \dots, x_N\}$, $x_n \in R^D$, 以 0 为中心, 即 $E[X] = 0$ 。没
有零均值假设

化,我们会得到完全相同的解决方案,但符号会更加混乱。

我们感兴趣的是找到 x 在 R^D 的低维子空间 U 上的最佳线性投影,其中 $\dim(U) = M$ 和正交基向量 b_1, \dots, b_M 。我们将这个子空间 U 称为主子空间。主子空间数据点的投影表示为

$$x_{\sim n} := \sum_{m=1}^{M} z_m b_m = B z_n \in R^D, \quad (10.28)$$

于基础 (b_1, \dots, b_M) 。更具体地说, $z \in R^M$ 是 $x_{\sim n}$ 的坐标向量,其中 $z_n := [z_{1n}, \dots, z_{Mn}]^T$ 。我们感兴趣的是让 $x_{\sim n}$ 尽可能与 x_n 相似。

我们在下文中使用的相似性度量是平方距离(欧几里得范数) $\|x - x_{\sim}\|^2$ 作为最小化平均平方欧几里得距离(重建误差)在 x 和 x_{\sim} 之间。因此,我们定义我们的 ob 误差(Pearson, 1901)

$$\text{杰姆} := \frac{1}{N} \sum_{n=1}^{N} \|x_n - x_{\sim n}\|^2, \quad (10.29)$$

其中我们明确表示我们将数据投影到的子空间的维数是 M 。为了找到这个最优线性投影,我们需要找到主子空间的正交基和投影的坐标 $z_n \in R^M$ 尊重这个基础。

为了找到主子空间的坐标 z_n 和 ONB, 我们采用两步法。首先, 我们针对给定的 ONB (b_1, \dots, b_M) 优化坐标 z_n ; 其次, 我们找到最佳的 ONB。

10.3.2 寻找最佳坐标

让我们从找到最佳坐标 z_{1n} 开始。 \dots, z_{Mn} 的投影 $x_{\sim n}$ for $n = 1, \dots, N$ 。考虑图 10.2(b), 其中主子空间由单个向量 b 跨越。从几何学上讲, 找到最佳坐标 z 对应于找到线性投影 x_{\sim} 相对于 b 的表示, 使 $x_{\sim} - x$ 之间的距离最小化。从图 10.2(b) 可以清楚地看出这将是正交投影, 下面我们将准确地展示这一点。

我们假设 $U \subseteq R^D$ 的ONB (b_1, \dots, b_M) 。为了找到关于这个基础的最佳坐标 z_m , 我们需要偏导数

$$\frac{\partial J_M}{\partial z_{in}} = \frac{\partial J_M}{\partial x_{\sim n}} \frac{\partial x_{\sim n}}{\partial z_{in}}, \quad (10.30a)$$

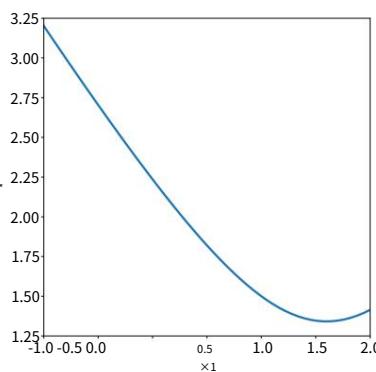
$$\frac{\partial J_M}{\partial x_{\sim n}} = -\frac{2}{N} (x_n - x_{\sim n}) \in R^{1 \times M}, \quad (10.30b)$$

328

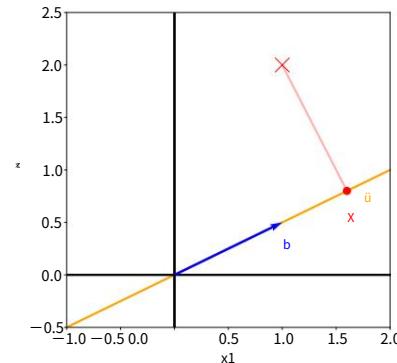
通过主成分分析降维

图 10.3 向量 $x \in R^2$ 的最优投影
到一维

子空间 (从
图 10.2 继续)。 (a)
距离 $\|x - x\|$ 对于一
些 $x \in U$ 。 (b) 正
交投影
和最优坐标。



(a) 距离 $\|x - z\|$ 对于某些 $z \in \text{span}[b]$; 有关设置, 请参见面板 (b)。



(b) 使面板 (a) 中的距离最小化的向量 z 是它在 U 上的正交投影。投影 x 相对于跨越 U 的基向量 b 的坐标是我们需要按顺序缩放 b 的因子 “到达” x 。

$$\frac{\partial \|x - z\|^2}{\partial z_{in}} \stackrel{(10.28)}{=} -\frac{\partial}{\partial z_{in}} \stackrel{*}{=} z_{mn} b_m = b_i \quad (10.30c)$$

对于 $i = 1, \dots, M$, 这样我们得到

$$\frac{\partial J_M}{\partial z_{in}} \stackrel{(10.30b)}{=} \sum_{i=1}^M (x_{in} - z_{in}) b_i \stackrel{(10.28)}{=} \sum_{i=1}^M x_{in} - z_{mn} b_m \quad (10.31a)$$

$$\sum_{i=1}^M (x_{in} - z_{in}) b_i = 0 \quad (10.31b)$$

的坐标
 x_n 相对于基向
量的最优投影

因为 $b_i = 1$, 将此偏导数设置为 0 会立即产生
最佳坐标

$$z_{in} = x_{in} \quad (10.32)$$

b_1, \dots, b_M 是坐标
 x_n 到主子空间
的正交投影。

对于 $i = 1, \dots, M$ 和 $n = 1, \dots, N$, 这意味着投影 x_n 的最佳坐标 z_{in} 是原始数据点 x_n 到由 b_i 跨越的一维子空间的正交投影 (见第 3.8 节) 的坐标。最后:

- x_n 的最优线性投影 x_{in} 是正交投影。
- x_{in} 相对于基 (b_1, \dots, b_M) 的坐标是 x_n 在主子空间上的正投影坐标。
- 正交投影是给定目标 (10.29) 的最佳线性映射。
- (10.26) 中 x 的坐标 z_m 和 (10.27) 中 x_{in} 的坐标 z_{in}

于 $m = 1$, 必须相同。 \dots, M 因为 U 是 $U = \text{span}[b_1, \dots, b_M]$ 。

$\perp = \text{跨度 } [b_{M+1}, \dots, b_D]$ 对

备注 (正交投影与正交基向量)。让我们简要回顾一下 3.8 节中的正交投影。如果 (b_1, \dots, b_D) 是 RD 的正交基, 则

b_x 是

$$x \sim = b_j (b_j^T b_j)^{-1} b_j \quad x = b_j b_j^T D \quad x \in R \quad (10.33)$$

x 到 b_j 跨越的子空间的正交投影。

是 x 在第 j 个基向量所跨越的子空间上的正交投影, 并且 $z_j = b_j$ 是跨越该子空间的基向量 b_j , 因为 $z_j b_j = x \sim$ 。图 10.3(b) 说明 $x \sim$ 是该投影相对于这个设置。

更一般地, 如果我们的目标是投影到 RD 的 M 维子空间, 我们将获得 x 到 M 维子空间的正交投影, 其正交基向量为 b_1, \dots, b_M 作为

$$x \sim = B(B^T B)^{-1} B \quad x = BB^T x, \quad (10.34)$$

我们定义 $B := [b_1, \dots, b_M] \in RD \times M$ 。该投影相对于有序基 (b_1, \dots, b_M) 的坐标为 $z := B^T x$

如第 3.8 节所述。

我们可以将坐标视为投影矢量在由 (b_1, \dots, b_M) 定义的新坐标系中的表示。请注意, 尽管 $x \in RD$, 我们只需要 M 个坐标 z_1, \dots, z_M 代表这个向量; 其他 $D - M$ 相对于基向量 (b_{M+1}, \dots, b_D) 的坐标始终为 0。投影到主子空间。

下面, 我们将确定最佳基础是什么。

10.3.3 求主子空间的基

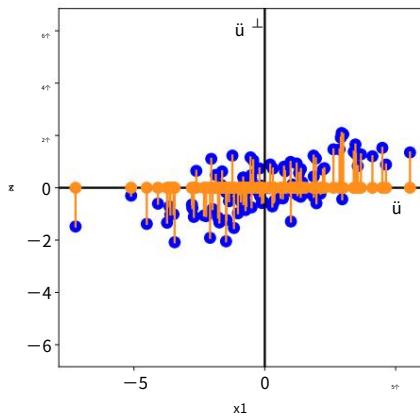
确定基向量 b_1, \dots, b_M 的主子空间, 我们使用目前的结果重新表述损失函数 (10.29)。这将使查找基向量变得更容易。为了重新制定损失函数, 我们利用之前的结果并获得

$$x \sim n = \sum_{m=1}^M z_m b_m b_m^T \stackrel{(10.32)}{=} (X \text{ 大地}) \text{ 大地。} \quad (10.35)$$

我们现在利用点积的对称性, 它产生

$$x \sim n = \sum_{m=1}^M b_m b_m^T \times \times \times \quad (10.36)$$

图 10.1 正交投影和位移矢量。当将数据点 x_n (蓝色) 投影到子空间 U_1 时, 我们得到 x_n (橙色)。位移向量 $x_n - x_n$ 完全位于 U_1 的正交补 U_2 中。



由于我们通常可以将原始数据点 x_n 写成所有基向量的线性组合, 因此认为

$$x_n = \sum_{d=1}^D z_d b_d b_d^\top \quad (10.37a)$$

$$= \sum_{m=1}^{M+1} b_m b_m^\top x_n + \sum_{j=M+1}^D b_j b_j^\top x_n, \quad (10.37b)$$

其中我们将具有 D 项的总和拆分为 M 上的总和和 $D - M$ 项上的总和。有了这个结果, 我们发现位移向量 $x_n - x_n$, 即原始数据点与其投影之间的差值向量, 是

$$x_n - x_n = \sum_{j=M+1}^D b_j b_j^\top x_n \quad (10.38a)$$

$$= \sum_{j=M+1}^D (x_n b_j) b_j^\top. \quad (10.38b)$$

这意味着差异恰好是数据点在主子空间的正交补集上的投影: 我们确定矩阵

$\sum_{j=M+1}^D b_j b_j^\top$ 在 (10.38a) 中作为执行此投影的投影矩阵。因此, 位移向量 $x_n - x_n$ 位于与主子空间正交的子空间中, 如图 10.1 所示。

备注 (低秩近似)。在 (10.38a) 中, 我们看到将 x 投影到 x 的投影矩阵由下式给出

$$\sum_{m=1}^M b_m b_m^\top = BB^\top. \quad (10.39)$$

通过构造为秩一矩阵 $b_m b_m^\top$ 的总和

我们看到 BB^\top 是

10.3 投影透视

331

对称且秩为M。因此,平均平方重建误差也可以写为

$$\frac{1}{n} \sum_{n=1}^N \|x_n - x_n\|_2^2 = \frac{1}{n} \sum_{n=1}^N \|x_n - BB_n x_n\|_2^2 \quad (10.40a)$$

$$= \frac{1}{n} \sum_{n=1}^N \|I - BB_n\|_2^2 x_n \quad (10.40b)$$

寻找正交基向量 b_1, \dots, b_M , 它最小化差异PCA 找到原始数据 x_n 和它们的投影 x_n 之间的最佳值, 相当于找到单位矩阵 I 的最佳等级 M 近似值 BB (见第 4.6 节)。 ◇
单位矩阵
的秩 M 近似。

现在我们拥有了重新制定损失函数 (10.29) 的所有工具。

$$JM = \frac{1}{n} \sum_{n=1}^N \|x_n - x_n\|_2^2 \stackrel{(10.38b)}{=} \frac{1}{n} \sum_{n=1}^N \|x_n - (b_j x_n) b_j\|_2^2 \quad (10.41)$$

我们现在明确地计算平方范数并利用 b_j 形成 ONB 的事实, 从而产生

$$JM = \frac{1}{n} \sum_{n=1}^N \sum_{j=M+1}^D (b_j x_n)_n^2 = \frac{1}{n} \sum_{n=1}^N \sum_{j=M+1}^D b_j x_n b_j x_n \quad (10.42a)$$

$$= \frac{1}{n} \sum_{n=1}^N \sum_{j=M+1}^D b_j x_n x_n^T b_j, \quad (10.42b)$$

我们在最后一步中利用点积的对称性来写 $b_j x_n = x$

$n b_j$ 。我们现在交换总和并获得

$$JM = \sum_{j=M+1}^D b_j \underbrace{\sum_{n=1}^N x_n x_n^T}_{{\text{小号}}} b_j = \sum_{j=M+1}^D b_j S b_j \quad (10.43a)$$

$$= \sum_{j=M+1}^D \text{tr}(b_j S b_j) = \sum_{j=M+1}^D \text{tr}(S b_j b_j^T) = \sum_{j=M+1}^D \underbrace{\text{tr}(b_j b_j^T)}_{\text{投影矩阵}}, \quad (10.43b)$$

我们利用了跟踪运算符 $\text{tr}(\cdot)$ (见 (4.18)) 是线性的并且对其参数的循环排列不变的属性。由于我们假设我们的数据集是居中的, 即 $E[X] = 0$, 我们将 S 确定为数据协方差矩阵。由于 (10.43b) 中的投影矩阵构造为秩一矩阵 $b_j b_j^T$ 的和 等式 (10.43a) 意味着我们可以将平均平方重建误差等效地表示为数据的协方差矩阵,

$\sum_j b_j b_j^T$ 它本身的等级为 $D - M$ 。

最小化平均平方

重构误差

等同于最小化数
据协方差的投影

矩阵到主子空间
的正交补集
上。

最小化平均平方
重建误差等同于
最大化投影数据的方
差。

投影到主子空间的正交补集上。因此，最小化平均平方重建误差等同于最小化投影到我们忽略的子空间（即主子空间的正交补集）上的数据方差。等价地，我们最大化我们在主子空间中保留的投影的方差，这将投影损失立即与第 10.2 节中讨论的 PCA 的最大方差公式联系起来。但这也意味着我们将获得与最大方差视角相同的解决方案。因此，我们省略了与第 10.2 节中介绍的推导相同的推导，并根据投影视角总结了前面的结果。

投影到 M 上时的平均平方重建误差
维度主子空间，是

丁

$$JM = \sum_{j=M+1}^D \lambda_j, \quad (10.44)$$

其中 λ_j 是数据协方差矩阵的特征值。因此，为了最小化 (10.44) 我们需要选择最小的 $D - M$ 特征值，这意味着它们对应的特征向量是主子空间的正交补集的基础。因此，这意味着主子空间的基础包括特征向量 b_1, \dots, b_M 与数据协方差矩阵的最大 M 个特征值相关联。

示例 10.3 (MNIST 数字嵌入)

图 10.1
MNIST 数字 0
(蓝色) 和 1 (橙
色) 在二维图
像中的嵌入

使用 PCA 的主子空
间。主子空间中数
字“0”和“1”的四
个嵌入以红色突出显
示，并带有相
应的原始数
字。

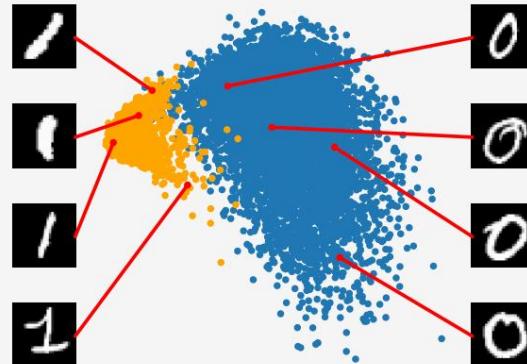


图 10.1 可可视化了 MMIST 数字“0”和“1”的训练数据嵌入到由前两个主成分跨越的向量子空间中。我们观察到“0”s (蓝点) 和“1”s (橙色点) 之间相对清晰的分离，我们看到每个人内部的差异

簇。主子空间中数字“0”和“1”的四个嵌入以红色突出显示，并带有相应的原始数字。该图显示“0”组内的变化明显大于“1”组内的变化。

10.4 特征向量计算和低秩逼近

在前面的章节中，我们得到了主子空间的基作为与数据

协方差矩阵的最大特征值相关联的特征向量

$$\text{小号} = \frac{\text{否}}{\text{nn}} \sum_{n=1}^N x_n x_n^\top = \frac{\text{否}}{\text{否}} X X^\top, \quad (10.45)$$

$$X = [x_1, \dots, x_N] \in \mathbb{R}^{D \times N}. \quad (10.46)$$

请注意， X 是一个 $D \times N$ 矩阵，即它是“典型”数据矩阵的转置 (Bishop, 2006; Murphy, 2012)。要获得 S 的特征值（和相应的特征向量），我们可以采用两种方法：

- 我们执行特征分解（参见第 4.2 节）并直接计算 S 的特征值和特征向量。
- 我们使用奇异值分解（参见第 4.5 节）。由于 S 是对称的并且分解为 $\frac{\text{否}}{\text{否}} X X^\top$ （忽略因子），因此 S 的特征值是 X 的奇异值的平方。

使用
特征分解或 SVD 来计算
特征向量。

更具体地说， X 的 SVD 由下式给出

$$X = U \Sigma V^\top, \quad (10.47)$$

其中 $U \in \mathbb{R}^{D \times D}$ 和 $V \in \mathbb{R}^{N \times N}$ 是正交矩阵， $\Sigma \in \mathbb{R}^{D \times N}$ 是一个矩阵，其唯一的非零元素是奇异值 $\sigma_{ii} > 0$ 。然后得出

$$\text{小号} = \frac{\text{否}}{\text{否}} X X^\top = \frac{\text{否}}{\text{否}} U \Sigma V^\top V \Sigma^\top U^\top = \frac{\text{否}}{\text{否}} U \Sigma \Sigma^\top U^\top. \quad (10.48)$$

根据 4.5 节的结果，我们得到 U 的列是 X 的 N 个特征向量的列。此外， S 的特征值 λ 通过以下方式与 X 的奇异值相关

$$\lambda = \frac{\text{否}}{\text{否}} \sigma_i^2. \quad (10.49)$$

S 的特征值和 X 的奇异值之间的这种关系提供了最大方差视图（第 10.2 节）和奇异值分解之间的联系。

10.4.1 PCA 使用低秩矩阵近似

为了最大化投影数据的方差（或最小化平均平方重构误差），PCA 选择 (10.48) 中 U 的列作为与数据协方差矩阵 S 的 M 个最大特征值相关的特征向量，以便我们确定 U 作为 (10.3) 中的投影矩阵 B ，它将原始数据投影到 M 维的低维子空间上。Eckart-Young 定理（4.6 节中的定理 4.25）提供了一种直接估计低维表示的方法化。考虑最好的等级- M 近似

埃卡特-杨定理

$$X \underset{\text{深} \times \text{深}}{=} \text{argmin}_{\text{rk}(A)} \|M - X - A\|_F^2 \in \mathbb{R}^{D \times D} \quad (10.50)$$

X 的，其中 $\|\cdot\|_F$ 是 (4.93) 中定义的谱范数。Eckart-Young 定理指出 $X \sim$ 是通过截断 top- M 奇异值处的 SVD 给出的。换句话说，我们得到

$$X \underset{\text{深} \times M}{=} U \underset{M \times M}{=} \Sigma \underset{M \times N}{=} V^T \quad (10.51)$$

正交矩阵 $UM := [u_1, \dots, u_M] \in \mathbb{R}^{D \times M}$ 和 $V^T = [v_1, \dots, v_M]^T \in \mathbb{R}^{N \times M}$ 和对角矩阵 $\Sigma \in \mathbb{R}^{M \times M}$ 其对角线元素是 X 的 M 个最大奇异值。

10.4.2 实践方面

查找特征值和特征向量在其他需要矩阵分解的基本机器学习方法中也很重要。理论上，正如我们在 4.2 节中讨论的那样，我们可以求解特征值作为特征多项式的根。然而，对于大于 4×4 的矩阵，这是不可能的，因为我们需要找到 5 次或更高次多项式的根。然而，Abel-Ruffini 定理（Ruffini, 1799 年；Abel, 1826 年）指出，对于 5 次或以上的多项式，此问题不存在代数解。因此，在实践中，我们使用迭代方法求解特征值或奇异值，所有现代线性代数包中都实现了这些方法。

阿贝尔-鲁菲尼
定理

`np.linalg.eigh`
或者
`np.linalg.svd`

在许多应用中（例如本章介绍的 PCA），我们只需要几个特征向量。计算完整的分解，然后丢弃特征值超出前几个特征值的所有特征向量将是一种浪费。事实证明，如果我们只对前几个特征向量（具有最大特征值）感兴趣，那么直接优化这些特征向量的迭代过程在计算上比完整的特征分解（或 SVD）更有效。在只需要第一个特征向量的极端情况下，称为幂迭代的简单方法非常有效。幂迭代选择一个不在

幂迭代

S的零空间并跟随迭代

$$x_{k+1} = \frac{Sx_k}{\|Sx_k\|}, \quad k = 0, 1, \dots \quad (10.52)$$

这意味着向量 x_k 在每次迭代中都乘以 S , 然后如果 S 是可逆的, 则它被归一化, 即, 我们总是有 $\|x_k\| = 1$ 。这个向量序列收敛到与 S 的最大特征值相关联的特征向量。最初的 Google PageRank 算法 (Page et al., 1999) 使用这种算法根据超链接对网页进行排名。

10.5 高维PCA为了进行PCA, 我们需要

计算数据协方差矩阵。在 D 维中, 数据协方差矩阵是 $D \times D$ 矩阵。计算该矩阵的特征值和特征向量在计算上非常昂贵, 因为它在 D 中按三次方缩放。因此, 正如我们之前讨论的那样, PCA 在非常高的维度上是不可行的。例如, 如果我们的 x_n 是具有 10,000 像素的图像 (例如, 100×100 像素图像), 我们将需要计算 $10,000 \times 10,000$ 协方差矩阵的特征分解。下面, 我们针对数据点远少于维度的情况 (即 $N < D$) 提供此问题的解决方案。

假设我们有一个居中数据集 x_1, \dots, x_N , 数据协方差矩阵为 $\Sigma \in \mathbb{R}^{N \times N}$, 其列是数据点。

$$\text{小号} = \frac{\frac{1}{N} \mathbf{X} \mathbf{X}^T}{\text{否}} \in \mathbb{R}^{D \times D}, \quad (10.53)$$

其中 $\mathbf{X} = [x_1, \dots, x_N]$ 是一个 $D \times N$ 矩阵, 其列是数据点。

我们现在假设 $N \ll D$, 即数据点的数量小于数据的维数。如果没有重复数据点, 则协方差矩阵 S 的秩为 N , 因此它有 $D - N + 1$ 个为 0 的特征值。直观上, 这意味着存在一些冗余。

下面, 我们将利用这一点, 将 $D \times D$ 协方差矩阵转化为特征值为正的 $N \times N$ 协方差矩阵。

在 PCA 中, 我们最终得到了特征向量方程

$$Sb_m = \lambda_m b_m, \quad m = 1, \dots, M, \quad (10.54)$$

其中 b_m 是主子空间的基向量。让我们稍微改写这个等式: 在 (10.53) 中定义 S , 我们得到

$$\text{体重指数} = \frac{\frac{1}{N} \mathbf{X} \mathbf{X}^T}{\text{否}} b_m = \lambda_m b_m. \quad (10.55)$$

我们现在从左侧乘以 $\mathbf{X} \in \mathbb{R}^{N \times D}$, 得到

$$\frac{\frac{1}{N} \mathbf{X} \mathbf{X}^T}{\text{否}} \mathbf{X} b_m = \lambda_m \mathbf{X} b_m \Leftrightarrow \frac{\frac{1}{N} \mathbf{X} \mathbf{X}^T}{\text{否}} \mathbf{X} c_m = \lambda_m c_m, \quad (10.56)$$

并且我们得到一个新的特征向量/特征值方程: λm 仍然是特征值,这证实了我们在 4.5.3 节中的结果,即 XX^T 的非零特征值等于 $X^T X$ 的非零特征值。我们得到 $X^T X \in \mathbb{R}^{N \times N}$ 与 λm 相关联作为矩阵 $Cm := X^T b_m$ 的特征向量。假设我们没有重复的数据点,也意味着 (非零) 特征值作为数据协方差矩阵 $X^T X$ 具有相同的秩 N 并且是可逆的。这方差矩阵 S 。但现在这是一个 $N \times N$ 矩阵,因此我们可以比原始 $D \times D$ 数据协方差矩阵更有效地计算特征值和特征向量。

现在我们有了特征向量 $X^T X$,我们将重新覆盖原始特征向量,我们仍然需要 PCA。目前, $X^T X$ 。如果我们左乘我们的特征值/我们知道特征向量 $X^T X$ 方程的特征向量与 X ,我们得到

$$\cancel{\underline{\underline{X}}^T X} \quad X^T Cm = \lambda m X^T Cm \quad (10.57)$$

小号

我们再次恢复数据协方差矩阵。这现在也意味着我们恢复 $X^T Cm$ 作为 S 的特征向量。

评论。如果我们想应用我们在第 10.6 节中讨论的 PCA 算法,我们需要对 S 的特征向量 $X^T Cm$ 进行归一化,使它们具有范数 1。 ◇

10.6 实践中 PCA 的关键步骤下面,我们将

使用一个运行示例来完成 PCA 的各个步骤,如图 10.2 所示。给定一个二维数据集 (图 10.2(a)), 我们想使用 PCA 将其投影到一维子空间。

1. 均值减法 我们首先通过计算数据集的均值 μ 并从每个数据点中减去它来使数据居中。

这确保了数据集的均值为 0 (图 10.2(b))。均值减法并非绝对必要,但可以降低出现数值问题的风险。

2. 标准化 将数据点除以数据集的标准偏差 σ_d , 对于每个维度 $d = 1, \dots, D$ 。现在数据是无单位的,并且它沿每个轴的方差为 1, 如图 10.2(c) 中的两个箭头所示。这一步完成了数据的标准化。

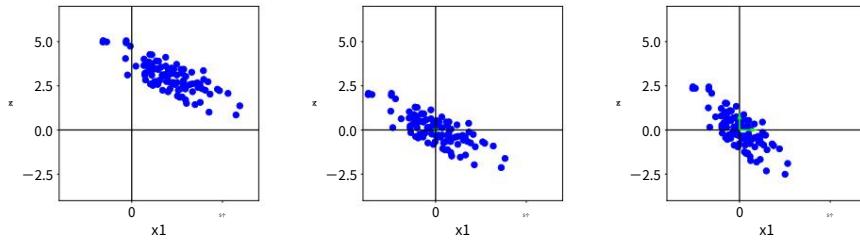
标准化

3. 协方差矩阵的特征分解 计算数据协方差矩阵及其特征值和对应的特征向量。

由于协方差矩阵是对称的,谱定理 (定理 4.15) 指出我们可以找到特征向量的 ONB。在图 10.2(d) 中, 特征向量按相关的大小缩放

10.6 PCA在实践中的关键步骤

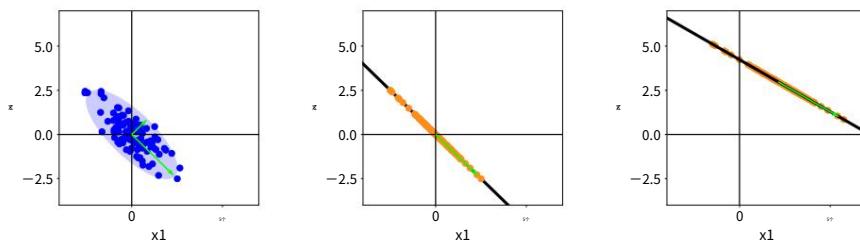
337



(a) 原始数据集。

(b) 第1步:通过从每个数据点减去平均值来居中。

(c) 第2步:除以标准偏差,使数据单元自由。资料有沿每个轴的方差1。



(d) 步骤 3:计算数据协方差矩阵的特征值和特征向量 (箭头) (椭圆)。

(e) 第 4 步:将数据投影到主子空间。

(f) 撤消标准化并将投影数据从 (a) 移回原始数据空间。

响应特征值。较长的向量跨越主子空间,我们用 U 表示。数据协方差矩阵由椭圆表示。

4. 投影我们可以将任意数据点 $x \in RD$ 投影到主子空间;为了做到这一点,我们需要分别使用第 d 维训练数据的均值 μ_d 和标准差 σ_d 对 x 进行标准化,以便

$$x_{*}^{(d)} \leftarrow \frac{x^{(d)} - \mu_d}{\sigma_d}, \quad d = 1, \dots, D, \quad (10.58)$$

其中 x 是 x^* 的第 d 个分量。我们得到的投影为

$$x = BB^T x^* \quad (10.59)$$

有坐标

$$z^* = B^T x^* \quad (10.60)$$

关于主子空间的基础。这里, B 是包含与数据协方差矩阵的最大特征值关联的特征向量作为列的矩阵。PCA 返回坐标 (10.60),而不是投影 x^* 。

图 10.2 PCA 的步骤。
 (a) 原始数据集; (b) 居中;
 (c) 除以标准偏差;
 (d) 特征分解;
 (e) 预测;
 (f) 映射回原始数据空间。

标准化我们的数据集后,(10.59) 仅在标准化数据集的上下文中产生投影。为了获得我们在原始数据空间 (即标准化之前) 的投影,我们需要取消标准化 (10.58) 并乘以标准差,然后再添加均值,以便我们获得 $\mathbf{x}_{(d)} \leftarrow \mathbf{x}_{(d)} \sigma_d + \mu_d$,

$$\mathbf{x}_{(d)} \leftarrow \mathbf{x}_{(d)} \sigma_d + \mu_d \quad d = 1, \dots, D. \quad (10.61)$$

图 10.2(f) 说明了原始数据空间中的投影。

示例 10.4 (MNIST 数字:重构)

下面,我们将PCA应用于MNIST数字数据集,其中包含60,000个手写数字0到9的示例。每个数字都是大小为 28×28 的图像,即包含784个像素,以便我们可以解释每个图像这个数据集作为向量 $\mathbf{x} \in \mathbb{R}^{784}$ 。这些数字的示例如图 10.2 所示。

图 10.1 增加校长人数的影响
重建的组成部分。



出于说明目的,我们将 PCA 应用于 MNIST 挖掘的一个子集,我们关注数字“8”。我们使用了 5,389 个数字“8”的训练图像,并确定了本章详述的主子空间。然后我们使用学习到的投影矩阵来重建一组测试图像,如图 10.1 所示。图 10.1 的第一行显示了一组来自测试集的四个原始数字。下面几行分别显示了使用维数为 1、10、100 和 500 的主要子空间时这些数字的重建。我们看到,即使使用一维主子空间,我们也能对原始数字进行半途而废的重建,然而,这是模糊和通用的。

随着主成分 (PC) 数量的增加,重建变得更加清晰,并且考虑了更多细节。有 500 个原则

cipal 组件,我们有效地获得了近乎完美的重建。如果我们选择 784 个 PC,我们将恢复准确的数字而不会造成任何压缩损失。

图 10.2 显示了平均平方重建误差,它是

$$\frac{1}{n} \sum_{n=1}^N \|x_n - \hat{x}_n\|^2 = \lambda_i, \quad (10.62)$$

作为主成分数 M 的函数。我们可以看到主成分的重要性迅速下降,增加更多的 PC 只能获得边际收益。这与我们在图 10.3 中的观察结果完全吻合,我们发现投影数据的大部分方差仅由少数几个主成分捕获。有了大约 550 台 PC,我们基本上可以完全重建包含数字“8”的训练数据(边界周围的一些像素在数据集中没有变化,因为它们总是黑色的)。

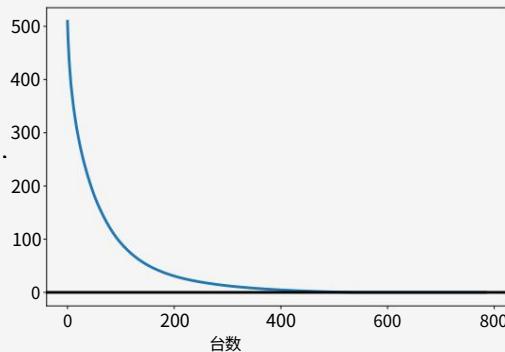


图 10.2 作为
函数的平均平方重
建误差
主成分的数量。平均平
方重建误差
是主子空间的正交
补集的特征值之和。

10.7 潜在变量视角

在前面的章节中,我们使用最大方差和投影视角在没有任何概率模型概念的情况下推导了 PCA。一方面,这种方法可能很有吸引力,因为它使我们能够避开概率论带来的所有数学困难,但另一方面,概率模型将为我们提供更多的灵活性和有用的见解。更具体地说,概率模型将

- 带有似然函数,我们可以明确处理嘈杂的观察结果(我们之前甚至没有讨论过)
- 允许我们通过第 8.6 节中讨论的边际似然进行贝叶斯模型比较 将 PCA 视为生成模型,这使我们能够模拟新数据
-

- 允许我们与相关算法建立直接联系 通过应用贝叶斯定理处理随机缺失的数据
- 维度 给我们一个新数据点新颖性的概念 给我们一个扩展模型的原则方法,例如 PCA 模型 将我们
- 在前面部分推导出的 PCA 作为特例 允许通过边缘化模型
- 参数进行完全贝叶斯处理
-
-

通过引入连续值潜在变量 $z \in \mathbb{R}^M$,可以将 PCA 表述为概率潜在变量模型。

Tipping 和 Bishop (1999) 提出这种潜在变量模型作为概率 PCA (PPCA)。

概率主成分分析

聚类数据

PPCA 解决了上述大部分问题,我们通过最大化投影空间中的方差或通过最小化重建误差获得的 PCA 解决方案是作为无噪声设置中最大似然估计的特例获得的。

10.7.1 生成过程和概率模型

在 PPCA 中,我们明确地写下了线性降维的概率模型。为此,我们假设一个连续的潜在变量 $z \in \mathbb{R}^M$,具有标准正态先验 $p(z) = N(0)$,和潜在变量与观察到的 x 数据之间的线性关系,其中

$$x = Bz + \mu + \varepsilon \in \mathbb{R}^D, \quad (10.63)$$

其中 $\varepsilon \sim N(0, \sigma^2)$ 是高斯观测噪声, $B \in \mathbb{R}^{D \times M}$ 和 $\mu \in \mathbb{R}^D$ 描述了从潜在变量到观测变量的线性/仿射映射。因此,PPCA 通过

$$p(x|z, B, \mu, \sigma^2) = N(x | Bz + \mu, \sigma^2 I). \quad (10.64)$$

总的来说,PPCA 诱导了以下生成过程:

$$z_n \sim N(z | 0, I) \text{, 我们} \quad (10.65)$$

$$| z_n \sim N(x | Bz_n + \mu, \sigma^2 I) \quad (10.66)$$

为了生成给定模型参数的典型数据点,我们遵循祖先采样方案:我们首先从 $p(z)$ 中采样一个潜在变量 z_n 。然后我们使用(10.64)中的 z_n 来采样一个以采样的 z_n 为条件的数据点,即 $x_n \sim p(x | z_n, B, \mu, \sigma^2)$ 。

这个生成过程允许我们写下概率模型
(即所有随机变量的联合分布;参见第 8.4 节)作为

$$p(x, z | B, \mu, \sigma^2) = p(x|z, B, \mu, \sigma^2)p(z), \quad (10.67)$$

使用第 8.5 节的结果立即产生图 10.2 中的图形模型。

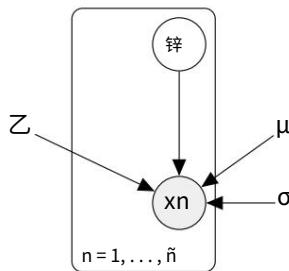


图 10.2 概率
PCA 的图形模型。

观测值 x_n 明确依赖于相应的
潜在变量
 $z_n \sim N(0, I)$ 。模型参数 B 、
 μ 和似然

评论。注意连接潜在变量 z 和观测数据 x 的箭头方向：箭头从 z 指向 x ，这意味着
PPCA 模型假设高维观测值 x 的低维潜在原因 z 。最后，根据一些观察，我们显然
有兴趣找到关于 z 的一些东西。为了到达那里，我们将应用贝叶斯推理来隐式地
“反转”箭头，并从观察到潜在变量。 ◇

参数 σ 共享
数据集。

示例 10.5（使用潜在变量生成新数据）

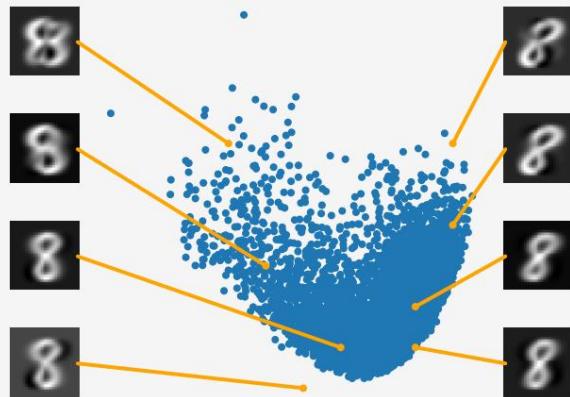


图 10.1 生成新的
MNIST 数字。潜在
变量 z

可用于生成新数据
 $x \sim Bz$ 。我们离训练数据
越近，生成的数据就越真实。

图 10.1 显示了 PCA 在使用二维主成分空间（蓝点）时发现的 MNIST 数字“8”的潜在坐标。我们可以查询这个潜在空间中的任何向量 z 并生成类似于数字“8”的图像 $x = Bz$ 。我们展示了八张这样生成的图像及其相应的潜在空间表示。根据我们查询潜在空间的位置，生成的图像看起来不同（形状、旋转、大小等）。如果我们远离训练数据查询，我们会看到越来越多的人工制品，例如，左上角和右上角的数字。请注意，这些生成的图像的固有维度只有两个。

10.7.2 似然与联合分布

可能性不依赖于潜在变量 z 。

使用第 6 章的结果,我们通过对潜在变量 z (参见第 8.4.3 节)进行积分来获得该概率模型的可能性,以便

$$p(x | B, \mu, \sigma^2) = p(x | z, B, \mu, \sigma^2) p(z) dz \quad (10.68a)$$

$$= N(x | Bz + \mu, \sigma^2) dz. \quad (10.68b)$$

从 6.5 节我们知道这个积分的解是一个均值为

$$E[x] = Ez[Bz + \mu] + E[z] = \mu \quad (10.69)$$

和协方差矩阵

$$V[x] = Vz[Bz + \mu] + V[z] = Vz[Bz] + \sigma^2 \quad \text{我} \quad (10.70a)$$

$$= BVz[z]B + \sigma^2 I = BB^T + \sigma^2 I. \quad \text{我} \quad (10.70b)$$

(10.68b) 中的似然可用于模型参数的最大似然或 MAP 估计。

评论。我们不能将 (10.64) 中的条件分布用于最大似然估计,因为它仍然取决于潜在变量。最大似然(或 MAP)估计所需的似然函数应该只是数据 x 和模型参数 θ 的函数,但不得依赖于潜在变量。

从 6.5 节中,我们知道高斯随机变量 z 和它的线性/仿射变换 $x = Bz$ 是联合高斯分布的。我们已经知道边缘 $p(z) = N(z | 0, I)$ 和 $p(x) = N(x | \mu, BB^T + \sigma^2 I)$ 。缺失的互协方差给出为

$$\text{Cov}[x, z] = \text{Cov}[Bz + \mu, z] = B \text{Cov}[z, z] = B. \quad (10.71)$$

因此,PPCA 的概率模型,即潜在随机变量和观察到的随机变量的联合分布由下式明确给出

$$p(x, z | B, \mu, \sigma^2) = N\begin{pmatrix} x \\ z \end{pmatrix} \begin{pmatrix} \mu \\ 0 \end{pmatrix}, \begin{pmatrix} BB^T + \sigma^2 I & B \\ B & I \end{pmatrix}, \quad (10.72)$$

具有长度为 $D + M$ 的均值向量和大小为 $(D + M) \times (D + M)$ 的协方差矩阵。

10.7.3 后验分布

(10.72) 中的联合高斯分布 $p(x, z | B, \mu, \sigma^2)$ 允许我们通过应用

6.5.1 节中的高斯调节规则。给定观察值 x 的潜在变量的后验分布是

$$p(z | x) = N(z | \mu, C) \quad (10.73)$$

$$\mu = \bar{x} + \sigma^{-1} (x - \bar{x}) \quad (10.74)$$

$$C = I - B(B^T + \sigma^{-2}I)^{-1}B \quad (10.75)$$

请注意,后验协方差不依赖于观察到的数据 x 。对于数据空间中的新观测值 x^* ,我们使用 (10.73) 来确定相应潜在变量 z^* 的后验分布。协方差矩阵 C 使我们能够评估嵌入的置信度。具有小行列式 (测量体积) 的协方差矩阵 C 告诉我们潜在嵌入 z 是相当确定的。如果我们获得一个方差很大的后验分布 $p(z^* | x^*)$,我们可能会遇到异常值。然而,我们可以探索这个后验分布来理解在这个后验分布下还有哪些其他数据点 x 是合理的。为此,我们利用了 PPCA 的生成过程,它允许我们通过生成在该后验条件下合理的新数据来探索潜在变量的后验分布:

1. 从潜在变量 (10.73) 的后验分布中采样潜在变量 $z \sim p(z | x)$ 。

2. 从 (10.64) 中抽取重构向量 $x \sim p(x | z, B, \mu, \sigma^2)$ 。

如果我们多次重复这个过程,我们可以探索潜在变量 z 的后验分布 (10.73) 及其对观察数据的影响。采样过程有效地假设了数据,这在后验分布下是合理的。

10.8 延伸阅读

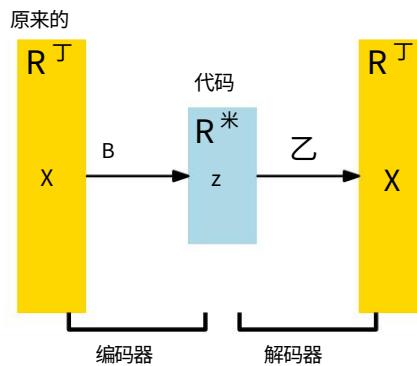
我们从两个角度得出 PCA:(a) 最大化投影空间中的方差; (b) 最小化平均重构误差。然而,PCA 也可以从不同的角度进行解释。让我们回顾一下我们所做的:我们采用高维数据 $x \in RD$ 并用于寻找低维表示 $z \in RM$ 。 B 的矩阵 B 列是与最大特征值相关联的数据协方差矩阵 S 的特征向量。一旦我们有了低维表示 z ,我们就可以获得它的高维版本 (在阵。原始数据空间中),如 $x \approx z = Bz = BB^T x \in RD$,其中 BB^T 是投影矩

我们也可以将 PCA 视为线性自动编码器,如图所示 - auto-encoder

10.2. 自动编码器对数据 $x_n \in R^n$ 进行编码,并将其解码为类似 \hat{x} 到代码 $z_n \in RM$ 代码。对于 x_n 的 $x \sim n$ 。从数据到代码的映射称为编码器,从代码到原始编码器原始数据空间的映射称为解码器。如果我们考虑线性映射,其中解码器

图 10.2 PCA 可以看作是一个线性自动编码器。它编码将高维数据 x 转化为低维表示（代码） $z \in RM$ 并使用解码器对 z 进行解码。这

解码向量 $x \sim$ 是原始数据 x 在 M 维主空间上的正交投影。



代码由 $z_n = B$ 给出数据 x_n 与其重构 \tilde{x}_n 之间 $x_n \in RM$ 我们对 minimiz 感兴趣的平均平方误差，我们得到 $x_n = Bz_n, n = 1, \dots,$

$$\frac{1}{n} \sum_{n=1}^N \|x_n - \tilde{x}_n\|^2 = \frac{1}{n} \sum_{n=1}^N \|x_n - BB^T x_n\|^2. \quad (10.76)$$

这意味着我们最终得到与我们在 10.3 节中讨论的 (10.29) 中相同的目标函数，以便我们在最小化平方自动编码损失时获得 PCA 解决方案。如果我们将 PCA 的线性映射替换为非线性映射，我们将得到一个非线性自动编码器。

一个突出的例子是深度自动编码器，其中线性函数被深度神经网络取代。在这种情况下，编码器识别网络也称为识别网络或推理网络，而

解码器也称为生成器。

PCA 的另一种解释与信息论有关。我们可以将代码视为原始数据点的缩小版或压缩版。当我们使用代码重建原始数据时，我们并没有得到确切的数据点，而是它的一个轻微扭曲或嘈杂的版本。这意味着我们的压缩是“有损的”。直观上，我们希望最大化原始数据和低维代码之间的相关性。更正式地说，这与互信息有关。

该代码是原始数据的压缩版本。

然后，我们将通过最大化互信息（信息论中的核心概念（MacKay, 2003））获得与 10.3 节中讨论的相同的 PCA 解决方案。

在我们对 PPCA 的讨论中，我们假设模型的参数，即 B 、 μ 和似然参数 σ 是已知的。Tipping 和 Bishop (1999) 描述了如何在 PPCA 设置中推导这些参数的最大似然估计（请注意，我们在本章中使用了不同的符号）。最大似然参数，当 pro

将D维数据投射到M维子空间,是

$$\mu ML = \frac{1}{\sqrt{M}} \sum_{n=1}^N x_n, \quad (10.77)$$

$$BML = T(\Lambda - \sigma^2 I)^{-1} R, \quad (10.78)$$

$$\sigma^2 = \frac{1}{D-M} \sum_{j=M+1}^D, \quad (10.79)$$

其中 $T \in RD \times M$ 包含数据协方差矩阵的M个特征向量,矩阵 $\Lambda - \sigma^2 I$ 是一个对角矩阵,其特征值与上的主轴相关联其对角线, $R \in RM \times M$ 是任意正交矩阵。最大似然解 BML 对于任意正交变换是唯一的,例如,我们可以将 BML 与任何旋转矩阵 R 右乘,使得 (10.78) 本质上是奇异值分解 (参见第 4.5 节)。Tipping 和 Bishop (1999) 给出了证明的概要。

在 (10.78) 作
为数据协方差的最小
特征值保证是半正定的

矩阵从下方以2噪声方
差 σ 为界

(10.77) 中给出的 μ 的最大似然估计是数据的样本均值。(10.79) 中给出的观测值的最大似然估计是主子空间正交补的平均方差,即我们不能用前 M 个主成分捕获的平均剩余方差被视为观测噪声。

噪声方差 σ

在 $\sigma \rightarrow 0$ 的无噪声极限,PPCA 和 PCA 提供相同的解决方案: 由于数据协方差矩阵 S 是对称的,它可以对角化 (见第 4.4 节),即存在 S 的特征向量的矩阵 T 所以那

$$S = T \Lambda T^{-1}. \quad (10.80)$$

在 PPCA 模型中,数据协方差矩阵是高斯似然 $p(x | \mu, \sigma^2)$ 的协方差矩阵,即 $BB^T + \sigma^2 I$,见(10.70b)。

对于 $\sigma \rightarrow 0$,我们获得 BB^T ,因此该数据协方差必须等于 PCA 数据协方差 (及其在 (10.80) 中给出的因素分解),以便

$$\text{Cov}[X] = T \Lambda T^{-1} = BB^T \iff B = T \Lambda^{-1/2} R, \quad (10.81)$$

即,我们在 (10.78) 中获得了 $\sigma = 0$ 的最大似然估计。

从 (10.78) 和 (10.80) 可以清楚地看出,(P)PCA 执行数据协方差矩阵的分解。

在数据按顺序到达的流设置中,建议使用迭代期望最大化 (EM) 算法进行最大似然估计 (Roweis, 1998)。

为了确定潜在变量的维数 (代码的长度,我们将数据投影到的低维子空间的维数),Gavish 和 Donoho (2014) 建议启发式方法,如果我们可以估计噪声方差 σ 的数据,我们应该

丢弃所有小于 $\frac{3}{\sqrt{2}}$ 的奇异值。或者,我们可以使用 (嵌套) 交叉验证 (第 8.6.1 节) 或贝叶斯模型选择标准 (在第 8.6.2 节中讨论) 来确定对数据内在维度的良好估计 (Minka, 2001b)。

类似于我们在第 9 章中对线性回归的讨论,我们可以在模型的参数上放置一个先验分布并将它们积分出来。通过这样做,我们 (a) 避免了参数的点估计和这些点估计带来的问题 (见第 8.6 节) 和 (b) 允许自动选择潜在空间的适当维数 M 。在这个由 Bishop (1999) 提出的贝叶斯 PCA 中,先验 $p(\mu, B, \sigma^2)$ 被放置在模型参数上。生成过程允许我们整合模型参数而不是对它们进行调节,从而解决了过度拟合问题。由于这种积分在分析上难以处理,Bishop (1999) 建议在推理方法中使用近似,例如 MCMC 或变分推理。我们参考了 Gilks 等人的工作。(1996) 和 Blei 等人。(2017) 有关这些近似推理技术的更多详细信息。

贝叶斯主成分分析

因子分析

过于灵活的可能性是

能够解释的不仅仅是噪音。

在 PPCA 中,我们考虑了线性模型 $p(x_n | z_n) = N(x_n | Bz_n + \mu, \sigma^2 I)$ with prior $p(z_n) = N(0, I)$, 其中所有观测维度都受到相同数量噪声的影响。如果我们允许每个观察维度 d 具有不同的方差 σ_d^2 , 我们将获得因子分析(FA) (Spearman, 1904; Bartholomew 等人, 2011)。这意味着 FA 比 PPCA 给予似然更多的灵活性, 但仍然强制数据由模型参数 B, μ 解释。但是, FA 不再允许封闭形式的最大似然解, 因此我们需要使用一个迭代方案, 例如期望最大化算法, 来估计模型参数。虽然在 PPCA 中所有固定点都是全局最优的, 但这不再适用于 FA。与 PPCA 相比, 如果我们缩放数据, FA 不会改变, 但如果我们将数据旋转, 它会返回不同的解决方案。

国际科学院

盲源分离

(Hyvärinen et al., 2001)。与 PCA 也密切相关的一种算法是独立成分分析(ICA)。从潜在变量角度重新开始 $p(x_n | z_n) = N(x_n | Bz_n + \mu, \sigma^2 I)$ 我们现在将 z_n 的先验更改为非高斯分布。ICA 可用于盲源分离。想象一下你在一个繁忙的火车站, 很多人在聊天。你的耳朵扮演着麦克风的角色, 它们在火车站里线性混合不同的语音信号。盲源分离的目标是识别混合信号的组成部分。

正如之前在 PPCA 的最大似然估计上下文中所讨论的, 原始 PCA 解决方案对于任何旋转都是不变的。因此, PCA 可以识别信号所在的最佳低维子空间, 但不能识别信号本身 (Murphy, 2012)。ICA 通过修改潜在源上的先验分布 $p(z)$ 来解决这个问题。

需要非高斯先验 $p(z)$ 。我们参考了 Hyvarinen 等人的书籍。(2001) 和 Murphy (2012) 了解更多关于 ICA 的细节。

PCA、因子分析和 ICA 是使用线性模型进行降维的三个示例。Cunningham 和 Ghahramani (2015) 对线性降维进行了更广泛的调查。

我们在此讨论的 (P)PCA 模型允许几个重要的扩展。在 10.5 节中，我们解释了当输入维数 D 明显大于数据点数 N 时如何进行 PCA。通过利用 PCA 可以通过计算（许多）内积来执行的洞察力，可以通过考虑无限维特征将这个想法推向极端。内核技巧是内核内核技巧 PCA 的基础，允许我们隐式计算无限内核 PCA 维度特征之间的内积 (Scholkopf 等人)。

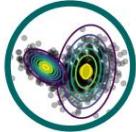
从 PCA 派生的非线性降维技术 (Burges (2010) 提供了很好的概述)。我们之前在本节中讨论的 PCA 的自动编码器视角可用于将 PCA 呈现为深度自动编码器的特例。在 deep auto-encoder 深度自编码器中，encoder 和 decoder 都是用多层前馈神经网络来表示的，它们本身就是非线性映射。如果我们将这些神经网络中的激活函数设置为恒等式，则该模型就等同于 PCA。另一种非线性降维方法是 Lawrence (2005) 提出的高斯过程隐变量高斯过程模型 (GP-LVM)。GP-LVM 从我们用来推导 PPCA 的潜变量视角开始，并用高斯过程 (GP) 替换潜变量 z 和观测值 x 之间的线性关系。GP-LVM 不是估计映射的参数（就像我们在 PPCA 中所做的那样），而是边缘化模型参数并对潜在变量 z 进行点估计。与 Bayesian PCA 类似，Titsias 和 Lawrence Bayesian GP-LVM (2010) 提出的 Bayesian GP-LVM 维护了潜在变量 z 的分布，并使用近似推理将它们整合出来。

GP-LVM

11

高斯混合的密度估计

楷模



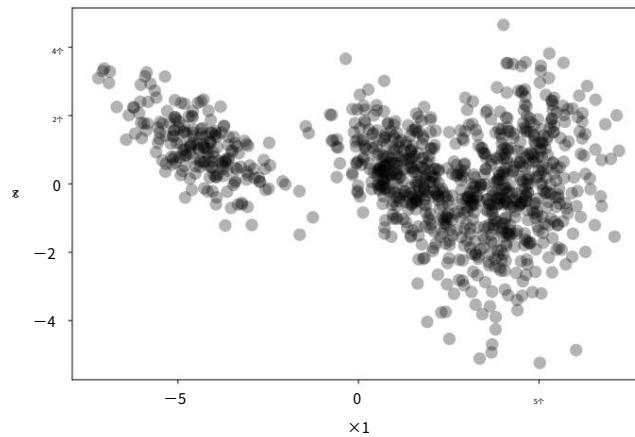
在前面的章节中,我们已经讨论了机器学习中的两个基本问题:回归(第9章)和降维(第10章)。在本章中,我们将了解机器学习的第三个支柱:密度估计。在我们的旅程中,我们介绍了重要的概念,例如期望最大化(EM)算法和混合模型密度估计的潜在变量视角。

当我们把机器学习应用于数据时,我们通常旨在以某种方式表示数据。一种直接的方法是将数据点本身作为数据的表示;示例请参见图11.1。

但是,如果数据集很大或者我们对表示数据的特征感兴趣,则这种方法可能无济于事。在密度估计中,我们使用来自参数族的密度来紧凑地表示数据,例如,高斯分布或Beta分布。例如,我们可能正在寻找数据集的均值和方差,以便使用高斯分布紧凑地表示数据。可以使用我们在第8.3节中讨论的工具找到均值和方差:最大似然估计或最大后验估计。然后我们可以使用这个高斯分布的均值和方差来表示数据的分布,也就是说,如果我们要从中采样,我们认为数据集是这个分布的典型实现。

图 11.1 二维

不能用高斯有意义地表示的数据集。



在实践中,高斯分布 (或类似的我们目前遇到的所有其他分布)的建模能力有限。例如,生成图 11.1 中数据的密度的高斯近似是一个较差的近似。在下文中,我们将研究一个更具表现力的分布族,我们可以将其用于密度估计:混合模型。

混合模型

混合模型可用于通过凸函数描述分布 $p(x)$
K 简单 (基本) 分布的组合

$$p(x) = \sum_{k=1}^K \pi_k p_k(x) \quad (11.1)$$

$$0 \leq \pi_k \leq 1, \quad \sum_{k=1}^K \pi_k = 1, \quad (11.2)$$

其中分量 p_k 是基本分布族的成员,例如高斯分布、伯努利分布或伽马分布,而 π_k 是混合权重。混合权重混合模型比相应的基本分布更具表现力,因为它们允许多模式数据表示,即它们可以描述具有多个“集群”的数据集,如图 11.1 中的示例。

我们将专注于高斯混合模型 (GMM),其中基本分布是高斯分布。对于给定的数据集,我们的目标是最大化模型参数训练 GMM 的可能性。为此,我们将使用第 5 章、第 6 章和第 7.2 节的结果。然而,与我们之前讨论的其他应用 (线性回归或 PCA) 不同,我们不会找到封闭形式的最大似然解。相反,我们将得到一组相关的联立方程,我们只能迭代求解。

11.1 高斯混合模型

高斯混合模型是一种密度模型,我们将 K 个高斯分布 $N(x|\mu_k, \Sigma_k)$ 的有限高斯混合数组合在一起。 μ_k, Σ_k 使得模型

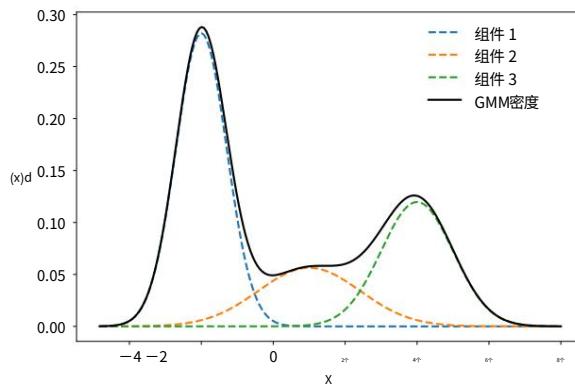
$$p(x | \theta) = \sum_{k=1}^K \pi_k N(x | \mu_k, \Sigma_k) \quad (11.3)$$

$$0 \leq \pi_k \leq 1, \quad \sum_{k=1}^K \pi_k = 1, \quad (11.4)$$

我们定义 $\theta := \{\mu_k, \Sigma_k, \pi_k : k = 1, \dots, K\}$ 作为模型所有参数的集合。与简单的高斯分布 (我们从 (11.3) 中恢复 $K = 1$) 相比,高斯分布的这种凸组合为我们建模复杂密度提供了更大的灵活性。图 11.1 给出了一个说明,显示加权

高斯混合模型的密度估计

图 11.1 高斯混合模型。高斯混合分布（黑色）由一个凸组合的高斯分布是比任何单个组件都更具表现力。虚线表示加权高斯成分。



成分和混合物密度,给出为 $0.2N(x|1,2) + 0.3N(x|4,1)$ 。

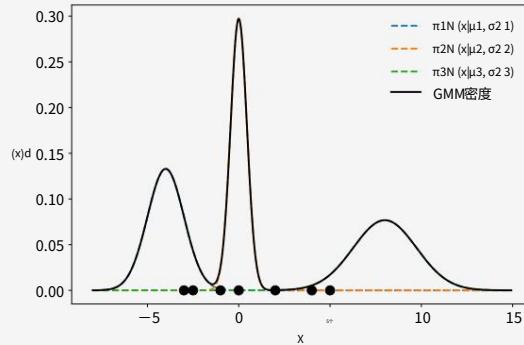
$$p(x|\theta) = 0.5N(x|-2, 1) + 0.5N(x|2, 1) \quad (11.5)$$

11.2 通过最大似然法学习参数

假设我们有一个数据集 $X = \{x_1, \dots, x_N\}$, 其中 $x_n, n = N$, 是从未知分布 $p(x)$ 中独立同分布地抽取的。我们的目标是通过具有 K 个混合分量的找到该未知分布 $p(x)$ 的良好近似/表示。GMM 的参数是 K 均值 μ_k 、协方差 Σ_k 和混合权重 π_k 。我们在 $\theta := \{\pi_k, \mu_k, \Sigma_k : k = 1, \dots, K\}$ 。

例 11.1 (初始设置)

图 11.1 初始设置:
GMM (黑色)
和混合物三
混合物成分 (虚线) 和
七个数据点 (圆盘)。



在本章中,我们将有一个简单的运行示例来帮助我们说明和可视化重要概念。

我们考虑一个由七个数据点组成的一维数据集 $X = \{-3, -2.5, -1, 0, 2, 4, 5\}$ 并希望找到一个具有 $K = 3$ 个分量的 GMM 来模拟数据的密度。我们将混合成分初始化为

$$p_1(x) = N(x | -4, 1) \quad (11.6)$$

$$p_2(x) = N(x | 0, 0.2) \quad (11.7)$$

$$p_3(x) = N(x | 8, 3) \quad (11.8)$$

并赋予它们相等的权重 $\pi_1 = \pi_2 = \pi_3 = \frac{1}{3}$ 。相应的模型（和数据点）如图 11.1 所示。

下面，我们将详细介绍如何获得模型参数 θ 的最大似然估计 θ_{ML} 。我们首先写下似然概率，即给定参数的训练数据的预测分布。我们利用我们的 iid 假设，这导致因式分解的可能性

$$p(X | \theta) = \prod_{n=1}^N p(x_n | \theta), p(x_n | \theta) = \prod_{k=1}^K \pi_k N(x_n | \mu_k, \Sigma_k), \quad (11.9)$$

其中每个单独的似然项 $p(x_n | \theta)$ 是高斯混合密度。然后我们得到对数似然为

$$\log p(X | \theta) = \sum_{n=1}^N \log p(x_n | \theta) = \sum_{n=1}^N \sum_{k=1}^K \pi_k N(x_n | \mu_k, \Sigma_k). \quad (11.10)$$

我们的目标是找到使(11.10) 中定义的对数似然 L 最大化的参数 θ 。我们的“正常”程序是计算关于模型参数 θ 的对数似然的梯度 $dL/d\theta$ ，将其设置为 0，然后求解 θ 。然而，与我们之前的最大似然估计示例不同（例如，当我们在第 9.2 节中讨论线性回归时），我们无法获得封闭形式的解决方案。然而，我们可以利用迭代方案来找到好的模型参数 θ_{ML} ，这将成为 GMM 的 EM 算法。关键思想是一次更新一个模型参数，同时保持其他模型参数不变。

评论。如果我们将单个高斯视为所需的密度，则(11.10) 中 k 的和消失，并且对数可以直接应用于高斯分量，这样我们得到

$$\text{对数 } N(x | \mu, \Sigma) = -\frac{1}{2} \log (2\pi)^d - \frac{1}{2} \log \det(\Sigma) - \frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu). \quad (11.11)$$

这个简单的形式允许我们找到 μ 和 Σ 的闭式最大似然估计，正如第 8 章所讨论的那样。在(11.10) 中，我们不能移动

\log 到 k 的总和中,这样我们就无法获得简单的封闭形式 \diamond 最大似然解。

函数的任何局部最优都表现出其相对于参数的梯度必须消失的性质(必要条件);见第7章。在我们的例子中,当我们优化(11.10)中关于GMM参数 μ_k 、 Σ_k 、 π_k 的对数似然时,我们获得了以下必要条件:

$$\frac{\partial L}{\partial \mu_k} = 0 \Leftrightarrow \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \mu_k} = 0, \quad (11.12)$$

$$\frac{\partial L}{\partial \Sigma_k} = 0 \Leftrightarrow \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \Sigma_k} = 0, \quad (11.13)$$

$$\frac{\partial L}{\partial \pi_k} = 0 \Leftrightarrow \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \pi_k} = 0. \quad (11.14)$$

对于所有三个必要条件,通过应用链式法则(参见第5.2.2节),我们需要以下形式的偏导数

$$\frac{\partial \log p(x_n | \theta)}{\partial \theta} = \frac{\frac{\partial p(x_n | \theta)}{\partial \theta}}{p(x_n | \theta)}, \quad (11.15)$$

$\{\mu_k, \Sigma_k, \pi_k, k = 1, \dots, K\}$ 是模型参数和

$$\frac{\partial p(x_n | \theta)}{\partial \theta} = \sum_{j=1}^K \frac{\pi_j N(x_n | \mu_j, \Sigma_j)}{p(x_n | \theta)}. \quad (11.16)$$

下面,我们将计算(11.12)到(11.14)的偏导数。但在此之前,我们先介绍一个将在本章剩余部分发挥核心作用的量:责任。

11.2.1 职责

我们定义数量

$$\text{恩克} := \frac{\pi_k N(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j N(x_n | \mu_j, \Sigma_j)} \quad (11.17)$$

作为第 k 个混合成分对第 n 个数据点的责任。

第 k 个混合分量对数据点 x_n 的责任 r_{nk} 与似然成正比

$$p(x_n | \pi_k, \mu_k, \Sigma_k) = \pi_k N(x_n | \mu_k, \Sigma_k) \quad (11.18)$$

服从玻尔兹曼/吉布斯分布。

给定数据点的混合成分。因此,当数据点可能是来自该混合物成分的合理样本时,混合物成分对数据点负有高度责任。请注意 $r_n := [r_{n1}, \dots, r_{nK}] \in \mathbb{R}^K$ 是一个(标准化的)概率向量,即,

$k \text{ rnk} = 1$ 且 $\text{rnk} = 0$ 。该概率向量在 K 个混合成分之间分配概率质量,我们可以将 rnk 视为 x_n 到 K 个混合成分的“软分配”。因此,(11.17) 中的责任 $\text{responsibility rnk}$ 表示 x_n 已由第 k 个混合成分生成的概率。

rnk 是第 k 个混合的概率

组件生成了
第 n 个数据点。

示例 11.2 (职责)

对于图 11.1 中的示例,我们计算责任 rnk

$$\begin{matrix} 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ & 0.057 & 0.943 & 0.0 \\ & 0.001 & 0.999 & 0.0 \\ & & 0.0 & 0.066 & 0.934 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 \end{matrix} \in \mathbb{R}_{+}^{N \times K}. \quad (11.19)$$

这里第 n 行告诉我们所有混合成分对 x_n 的责任。数据点的所有 K 责任之和(每行的总和)为 1。第 k 列为我们提供了第 k 个混合成分责任的概述。我们可以看到第三个混合成分(第三列)不负责前四个数据点中的任何一个,但对其余数据点负有很大责任。一列所有条目的总和为我们提供了值 N_k ,即第 k 个混合物成分的总责任。在我们的示例中,我们得到 $N_1 = 2.058$ 、 $N_2 = 2.008$ 、 $N_3 = 2.934$ 。

在下文中,我们确定了给定职责的模型参数 μ_k 、 Σ_k 、 π_k 的更新。我们将看到更新方程都依赖于责任,这使得最大似然估计问题的封闭形式的解决方案成为不可能。然而,对于给定的职责,我们将一次更新一个模型参数,同时保持其他模型参数不变。在此之后,我们将重新计算责任。迭代这两个步骤最终会收敛到一个局部最优,是 EM 算法的具体实例。我们将在 11.3 节中更详细地讨论这个问题。

11.2.2 更新均值

定理 11.1 (GMM 均值的更新)。更新均值 μ_k 为 $k = 1, \dots, GMM$ 的, K , 由参数 μ_k 给出,

$$\mu_k^{\text{新的}} = \frac{\sum_{n=1}^N \text{rnk}_n x_n}{\sum_{n=1}^N \text{rnk}_n}, \quad (11.20)$$

其中职责 rnk 在(11.17)中定义。

评论。(11.20) 中单个混合分量的均值 μ_k 的更新取决于所有均值、协方差矩阵 Σ_k 和通过(11.17) 中给出的 r_{nk} 的混合权重 π_k 。因此, 我们无法一次获得所有 μ_k 的闭式解。 \diamond

证明从 (11.15) 中, 我们看到对数似然的梯度相对于平均参数 μ_k , $k = 1, \dots, K$, 要求我们计算偏导数

$$\frac{\partial p(x_n | \theta)}{\partial \mu_k} = \sum_{j=1}^K \pi_j \frac{\partial N(x_n | \mu_j, \Sigma_j)}{\partial \mu_k} = \pi_k \frac{\partial N(x_n | \mu_k, \Sigma_k)}{\partial \mu_k} \quad (11.21a)$$

$$= \pi_k (x_n - \mu_k) \Sigma_k^{-1} N(x_n | \mu_k, \Sigma_k), \quad (11.21b)$$

我们利用只有第 k 个混合成分取决于 μ_k 的地方。

我们在 (11.15) 中使用 (11.21b) 的结果并将所有内容放在一起因此, L 相对于 μ_k 的所需偏导数为

$$\frac{\partial L}{\partial \mu_k} = \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \mu_k} = \sum_{n=1}^N \frac{1}{\pi_k} \frac{\partial p(x_n | \theta)}{\partial \mu_k} p(x_n | \theta) \quad (11.22a)$$

$$= \sum_{n=1}^N (x_n - \mu_k) \Sigma_k^{-1} \boxed{\frac{\pi_k N(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j N(x_n | \mu_j, \Sigma_j)}} \quad (11.22b)$$

$$= \sum_{n=1}^N r_{nk} (x_n - \mu_k) \Sigma_k^{-1} \quad (11.22c)$$

这里我们使用 (11.16) 的恒等式和 (11.21b) 中的偏导数的结果得到 (11.22b)。值 r_{nk} 是我们在 (11.17) 中定义的职责。

$$\begin{aligned} \text{我们现在求解 (11.22c) 的 } \mu_k & \text{ 使得 } \frac{\partial L(\mu_k)}{\partial \mu_k} = 0 \text{ 并获得} \\ \text{否} & \text{ } \end{aligned} \quad (11.23)$$

$$\begin{aligned} r_{nk} x_n &= \sum_{n=1}^N r_{nk} \mu_k \quad \Leftrightarrow \mu_k = \frac{\sum_{n=1}^N r_{nk} x_n}{\sum_{n=1}^N r_{nk}} = \frac{1}{N_k} \sum_{n=1}^N r_{nk} x_n \quad \text{恩恩,} \\ \text{否} & \text{ } \end{aligned}$$

我们定义的地方

$$N_k := \sum_{n=1}^N r_{nk} \quad (11.24)$$

作为整个数据集的第 k 个混合组件的总责任。定理 11.1 的证明到此结束。 \square

直观地,(11.20) 可以解释为均值的重要性加权蒙特卡洛估计, 其中数据点 x_n 的重要性权重是第 k 个簇对 x_n 的责任 r_{nk} , $k = 1, \dots, K$.

因此,平均 μ_k 被拉向具有强度的数据点 x_n 图 11.2 更新由 r_{nk} 给出。均值被更强地拉向相应混合成分具有高责任(即高可能性)的数据点。图 11.2 说明了这一点。我们还可以将(11.20)中的平均更新解释为在给定分布下所有数据点的期望值

GMM 中混合分量的平均参数。这

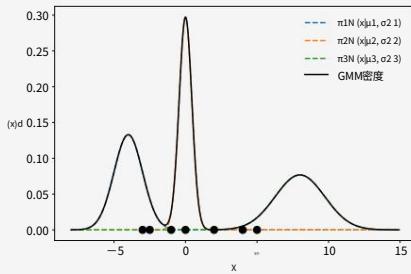
$$r_k := [r_{1k}, \dots, r_{Nk}] / N_k, \quad (11.25)$$

意味着 μ 被拉向具有相应职责赋予的权重的单个数据点。

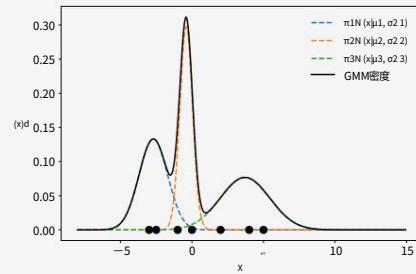
这是归一化概率向量,即

$$\mu_k \leftarrow E_{\mu_k}[X]. \quad (11.26)$$

例 11.3 (平均更新)



(a) 更新平均值之前的 GMM 密度和单个组件。



(b) 更新平均值后的 GMM 密度和单个组件。

在图 11.1 的示例中,平均值更新如下:

$$\mu_1 : -4 \rightarrow -2.7 \quad (11.27)$$

$$\mu_2 : 0 \rightarrow -0.4 \quad (11.28)$$

$$\mu_3 : 8 \rightarrow 3.7 \quad (11.29)$$

在这里,我们看到第一个和第三个混合分量的均值向数据状态移动,而第二个分量的均值变化不大。图 11.3 说明了这种变化,其中图 11.3(a) 显示了更新平均值之前的 GMM 密度,图 11.3(b) 显示了更新平均值 μ_k 之后的 GMM 密度。

图 11.3 更新 a 中的平均值的效果

GMM。(a) 更新平均值之前的 GMM; (b) 在保留方差和混合权重的同时更新平均值 μ_k 之后的 GMM。

11.20

中平均参数的更新看起来相当直接。但是,请注意,对于所有 $j = 1, \dots, K$, 责任 r_{nk} 是 π_j 、 μ_j 、 Σ_j 的函数。使得(11.20)中的更新取决于 GMM 的所有参数,并且无法获得我们在第 9.2 节中为线性回归或第 10 章中的 PCA 获得的封闭形式的解决方案。

11.2.3 更新协方差

定理 11.2 (GMM 协方差的更新)。协方差参数的更新 Σ_k , $k = 1, \dots, K$ 由下式给出

$$\sum k^{\text{新的}} = \frac{1}{Nk} \sum_{n=1}^{Nk} \text{rank}(x_n - \mu_k)(x_n - \mu_k)^T, \quad (11.30)$$

其中 r_{nk} 和 N_k 分别在(11.17)和(11.24)中定义。

证明为了证明定理 11.2,我们的方法是计算对数似然 L 关于协方差 Σ_k 的偏导数,将它们设置为 0,然后求解 Σ_k 。我们从我们的一般方法开始。

$$\frac{\partial L}{\partial \Sigma_k} = \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \Sigma_k} = \sum_{n=1}^N \frac{p(x_n | \theta)}{\partial \Sigma_k} \frac{\partial p(x_n | \theta)}{\partial \Sigma_k} \text{.} \quad (11.31)$$

我们已经从(11.16)中知道 $1/p(xn | \theta)$ 。为了获得剩余的偏导数 $\partial p(xn | \theta) / \partial \Sigma k$, 我们写下高斯分布 $p(xn | \theta)$ 的定义(见(11.9))并删除除第k项以外的所有项。然后我们得到

$$\frac{\partial p(x_n | \theta)}{\partial \Sigma_k} \quad (11.32a)$$

$$= \frac{\partial}{\partial \Sigma k} \quad \pi k (2\pi) - \frac{J}{2} \det(\Sigma k) - 2 \exp \left(- \frac{J^2}{2\pi} (\chi n - \mu k) \right) \quad \Sigma \quad k^{-1} (\chi n - \mu k)$$

(11.32b)

$$= \pi k(2\pi) - \frac{2}{\frac{\partial}{\partial \Sigma k}} \det(\Sigma k) - 2^{\frac{17}{2}} \exp(-\frac{17}{2}(\chi_n - \mu_k)) \quad \sum_k^{-1} (\chi_n - \mu_k)$$

$$+ \det(\Sigma_k) = \frac{1}{2^{\frac{n}{2}} \det(\Sigma_k)} \exp(-\frac{1}{2} (x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k)) \quad . \quad (11.32c)$$

我们现在使用身份

$$\frac{\partial}{\partial \Sigma k} \det(\Sigma k) = \frac{1}{2k} (5.101) \quad \text{1} \det(\Sigma k) - 2^{\frac{1}{k}-1} 2^{\Sigma} \quad (11.33)$$

$$\frac{\partial}{\partial \Sigma k} (x_n - \mu_k) \quad \Sigma \quad {}_{k=1}^{-1} (x_n - \mu_k) \quad \stackrel{(5.103)}{=} \Sigma k \quad {}_{k=1}^{-1} (x_n - \mu_k) (x_n - \mu_k) \quad \Sigma \quad {}_{k=1}^{-1}$$

(11.34)

并获得（经过一些重新排列后）(11.31)中所需的偏导数为

$$\frac{\partial p(x_n | \theta)}{\partial \theta} = \pi_k N x_n | \mu_k, \Sigma_k \partial \Sigma_k$$

$$= -\frac{n}{\pi_k} (\sum_k^{-1} - \sum_k^{-1} (x_n - \mu_k)(x_n - \mu_k)^T) \Sigma_k^{-1} \quad (11.35)$$

把所有东西放在一起,对数似然的偏导数

11.2 通过最大似然法学习参数

357

关于 Σk 由下式给出

$$\frac{\partial L}{\partial \Sigma k} = \sum_{n=1}^N \frac{\partial \log p(x_n | \theta)}{\partial \Sigma k} = \sum_{n=1}^N \frac{\partial p(x_n | \theta)}{p(x_n | \theta) \partial \Sigma k} \quad (11.36a)$$

$$= \sum_{n=1}^N \frac{\pi_{kN} x_n | \mu_k, \Sigma_k}{\sum_{j=1}^K \pi_{jN} x_n | \mu_j, \Sigma_j} \quad (11.36b)$$

$$= -\frac{1}{2} \left(\sum_{k=1}^K \left(\sum_{n=1}^N (x_n - \mu_k)(x_n - \mu_k)^T \right) - \sum_{k=1}^K \left(\sum_{n=1}^N (x_n - \mu_k)(x_n - \mu_k)^T \right)^{-1} \right) \quad (11.36c)$$

$$= -\frac{1}{2} \sum_{k=1}^K \left(\sum_{n=1}^N (x_n - \mu_k)(x_n - \mu_k)^T \right) - \sum_{k=1}^K \sum_{n=1}^N (x_n - \mu_k)(x_n - \mu_k)^T \quad (11.36d)$$

(11.36 天)

我们看到责任 r_{nk} 也出现在这个偏导数中。

将这个偏导数设置为0,我们得到了必要的最优条件

$$Nk \sum_{k=1}^K \left(\sum_{n=1}^N r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T \right) = \sum_{k=1}^K \sum_{n=1}^N r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T \quad (11.37a)$$

$$\Leftrightarrow NkI = \sum_{k=1}^K \sum_{n=1}^N r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T \quad (11.37b)$$

通过求解 Σk ,我们得到

$$\Sigma k^{\text{新的}} = \frac{1}{Nk} \sum_{k=1}^K \sum_{n=1}^N r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T, \quad (11.38)$$

其中 r_{nk} 是 (11.25) 中定义的概率向量。这为我们提供了一个简单的 Σk 更新规则,其中 $k = 1, \dots, K$ 并证明定理 11.2。 \square 类似于 (11.20) 中 μ_k 的更新,我们可以将 (11.30) 中协方差的更新解释为中心数据平方的重要性加权期望值 $X_k := \{x_1 - \mu_k, \dots, x_N - \mu_k\}$ 。

例 11.4 (方差更新)

在图 11.1 的示例中,方差更新如下:

$$z_{o1} : 1 \rightarrow 0.14 \quad (11.39)$$

$$z_{o2} : 0.2 \rightarrow 0.44 \quad (11.40)$$

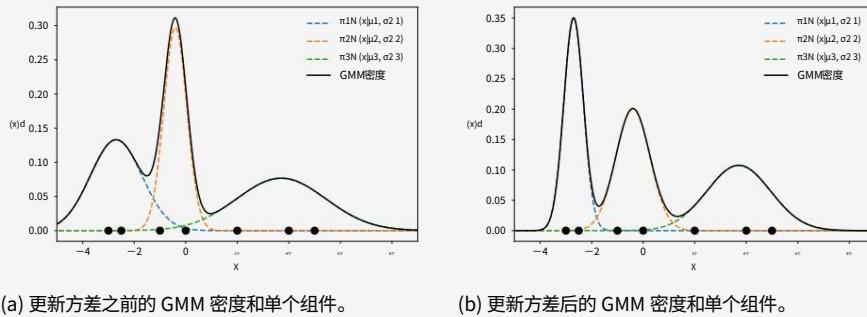
$$z_{o3} : 3 \rightarrow 1.53 \quad (11.41)$$

在这里我们看到第一个和第三个分量的方差显着缩小,而第二个分量的方差略有增加。

图 11.4 说明了此设置。图 11.4(a) 与图 11.3(b) 相同 (但放大了), 显示了 GMM 密度及其在更新方差之前的各个分量。图 11.4(b) 显示了更新方差后的 GMM 密度。

图 11.4 在 GMM 中更新方差的效果。
(a) 更新方差之前的 GMM; (b) 更新方差后的 GMM

保留手段和混合物权重。



类似于均值参数的更新,我们可以将 (11.30) 解释为与第 k 个混合分量相关的数据点 x_n 的加权协方差的蒙特卡罗估计,其中权重是责任 r_{nk} 。与平均参数的更新一样,此更新取决于所有 $\pi_j, \mu_j, \Sigma_j, j = 1, \dots, K$, 通过责任 r_{nk} , 它禁止封闭形式的解决方案。

11.2.4 更新混合权重

定理 11.3 (GMM 混合权重的更新)。GMM 的混合权重更新为

$$\pi_k^{\text{新的}} = \frac{N_k}{\text{否}} , \quad k = 1, \dots, K , \quad (11.42)$$

其中 N 是数据点的数量, N_k 在 (11.24) 中定义。

证明求对数似然关于权重参数 π_k 的偏导数, $k = 1, \dots, K$, 我们通过使用拉格朗日乘数来解释约束 $\sum_k \pi_k = 1$ (见第 7.2 节)。拉格朗日量是

$$\text{大号} = \text{大号} + \lambda \sum_{k=1}^K \pi_k - 1 \quad (11.43a)$$

$$= \prod_{n=1}^{\text{否}} \prod_{k=1}^{\text{钾}} \frac{\pi_k N x_n | \mu_k, \Sigma_k + \lambda}{\sum_{j=1}^N \pi_j N x_n | \mu_j, \Sigma_j} \prod_{k=1}^{\text{钾}} \pi_k - 1 , \quad (11.43b)$$

其中 L 是 (11.10) 的对数似然, 第二项编码所有混合权重需要总和为 1 的等式约束。我们获得关于 π_k 的偏导数为

$$\frac{\partial L}{\partial \pi_k} = \prod_{n=1}^{\text{否}} \frac{N x_n | \mu_k, \Sigma_k}{\sum_{j=1}^N \pi_j N x_n | \mu_j, \Sigma_j} + \lambda \quad (11.44a)$$

$$= \prod_{n=1}^{\text{否}} \frac{\sum_{j=1}^N \pi_j N x_n | \mu_j, \Sigma_j}{\sum_{j=1}^N \pi_j N x_n | \mu_j, \Sigma_j} + \lambda = \lambda \frac{N_k}{\sum_{j=1}^N \pi_j N x_n | \mu_j, \Sigma_j} = \lambda \frac{N_k}{N_k} = 1 , \quad (11.44b)$$

和关于拉格朗日乘数 λ 的偏导数为

$$\frac{\partial L}{\partial \lambda} = \prod_{k=1}^{\text{钾}} \pi_k - 1 . \quad (11.45)$$

将两个偏导数都设置为 0 (最优的必要条件) 产生方程组

$$\pi_k = - \frac{N_k}{\lambda} , \quad (11.46)$$

$$1 = \prod_{k=1}^{\text{钾}} \pi_k . \quad (11.47)$$

将 (11.46) 代入 (11.47) 并求解 π_k , 我们得到

$$\prod_{k=1}^{\text{钾}} \pi_k = 1 \Leftrightarrow - \prod_{k=1}^{\text{钾}} \frac{N_k}{\lambda} = 1 \Leftrightarrow - \lambda \lambda \frac{N}{\lambda} = 1 \Leftrightarrow \lambda = -N . \quad (11.48)$$

这允许我们用 $-N$ 替代 (11.46) 中的 λ 以获得

$$\pi_k^{\text{新的}} = \frac{N_k}{\sum_{j=1}^N N_j} , \quad (11.49)$$

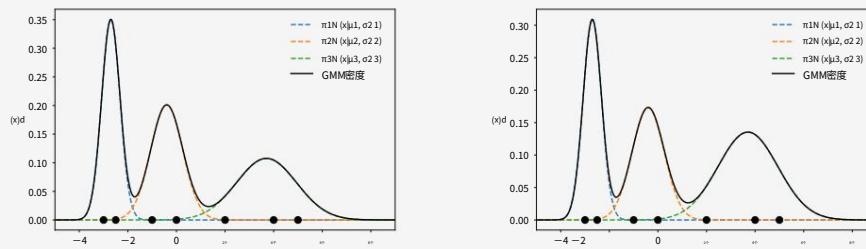
这为我们提供了权重参数 π_k 的更新并证明了定理 11.3。 \square

我们可以将 (11.42) 中的混合权重确定为第 k 个簇的总责任与数据点数的比率。由于 N_k , 数据点的数量也可以解释为 $N = \sum_{j=1}^N N_j$ = 所有混合成分一起的总责任, 因此 π_k 是数据集的第 k 个混合成分的相对重要性。

评论: 由于 $N_k = \text{true}$ 权重 π_k 也取决于 $i=1 \dots N_k$, 通过 $\text{re } \diamond$ 更新混合 K 的方程式 (11.42)
所有 $\pi_j, \mu_j, \Sigma_j, j = 1, \dots, N$, 负责 N_k 。

图 11.5 在 GMM 中更新混合权重的效果。
(a) 更新混合权重之前的 GMM; (b) 在保留均值和方差的同时更新混合权重后的 GMM。注意垂直轴的不同比例。

例 11.5 (权重参数更新)



(a) 更新混合权重之前的 GMM 密度和单个组件。

(b) 更新混合权重后的 GMM 密度和单个组件。

在图 11.1 的运行示例中,混合权重向上日期如下:

$$\pi_1: \frac{1}{3} \rightarrow 0.29 \quad (11.50)$$

$$\pi_2: \frac{1}{3} \rightarrow 0.29 \quad (11.51)$$

$$\pi_3: \frac{1}{3} \rightarrow 0.42 \quad (11.52)$$

在这里,我们看到第三个组件的权重/重要性更高,而其他组件的重要性略有下降。图 11.5 说明了更新混合权重的效果。图 11.5(a) 与图 11.4(b) 相同,显示了更新混合权重之前的 GMM 密度及其各个分量。图 11.5(b) 显示了更新混合权重后的 GMM 密度。

总的来说,更新均值、方差和权重一次后,我们得到如图 11.5(b) 所示的 GMM。与图 11.1 所示的初始化相比,我们可以看到参数更新导致 GMM 密度将其部分质量移向数据点。

在均值、方差和权重更新一次后,图 11.5(b) 中的 GMM 拟合已经明显好于图 11.1 中的初始化。对似然值也证明了这一点,在一个完整的更新周期后,该值从 28.3 (初始化) 增加到 14.4。

11.3 EM 算法

不幸的是,(11.20)、(11.30) 和 (11.42) 中的更新不构成混合模型参数 μ_k 、 Σ_k 、 π_k 更新的封闭形式解决方案,因为责任 r_{nk} 取决于这些参数以一种复杂的方式。然而,结果提出了一种简单的迭代方案,用于通过最大似然法找到参数估计问题的解决方案。期望最大化算法 (EM algo

rithm)是由 Dempster 等人提出的。(1977),是混合模型中学习参数(最大似然或 MAP)的通用迭代方案,更一般地说,是潜在变量模型。

在我们的高斯混合模型示例中,我们为 μ_k 、 Σ_k 、 π_k 和交替选择初始值,直到收敛于

- E步:评估责任 r_{nk} (数据点n属于混合成分k的后验概率)。
- M-step:使用更新后的责任重新估计参数 μ_k 、 Σ_k 、 π_k 。

EM 算法中的每一步都会增加对数似然函数(Neal 和 Hinton,1999)。为了收敛,我们可以直接检查对数似然或参数。用于估计 GMM 参数的 EM 算法的具体实例如下:

1. 初始化 μ_k 、 Σ_k 、 π_k 。

2. E步:使用 cur 为每个数据点 x_n 评估责任 r_{nk}

租金参数 π_k , μ_k ,
 Σ_k :

$$r_{nk} = \frac{\pi_k N(x_n | \mu_k, \Sigma_k)}{\sum_j N(x_n | \mu_j, \Sigma_j)} . \quad (11.53)$$

μ_k , bilities r_{nk} (来自 E-step) :

前响应 3. M-step:重新估计参数 π_k ,

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N_k} r_{nk} x_n , \quad (11.54)$$

更新了
在 (11.54) 中
表示 μ_k ,它们随后在
(11.55) 中用于更新相应
的协方差。

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N_k} r_{nk} (x_n - \mu_k)(x_n - \mu_k)^T , \quad (11.55)$$

$$\pi_k = \frac{N_k}{N} . \quad (11.56)$$

示例 11.6 (GMM 拟合)

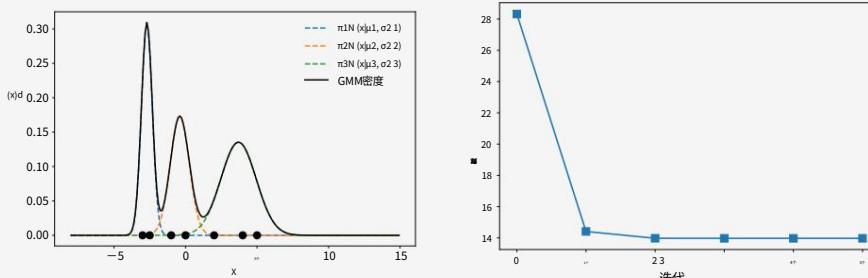


图 11.6 应用于图 11.1 中 GMM 的 EM 算法。(A)

最终 GMM 拟合; (b) 作为 EM 函数的负对数似然迭代。

362

高斯混合模型的密度估计

图 11.7 用于拟合
高斯混合模型的 EM 算
法图示

二维的三个组成部分

数据集。 (a) 数据集；
(b) 作为 EM
函数的负对数似然
然 (越低越好)

迭代。红点表示

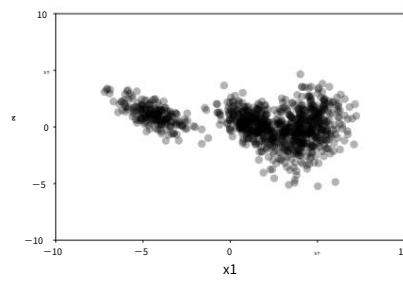
混合的迭代

相应 GMM 拟合的组件显
示在 (c) 到 (f) 中。黄色圆盘
表示

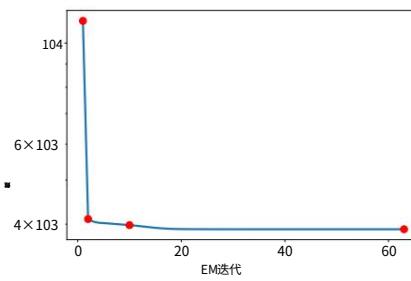
高斯混合

成分。

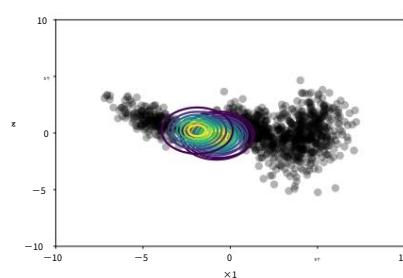
图 11.8(a) 显示
了最终的 GMM
拟合。



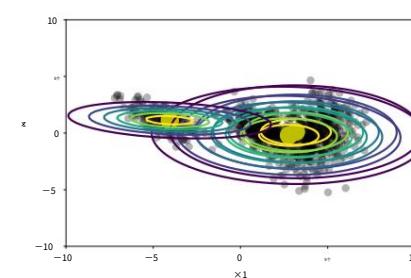
(a) 数据集。



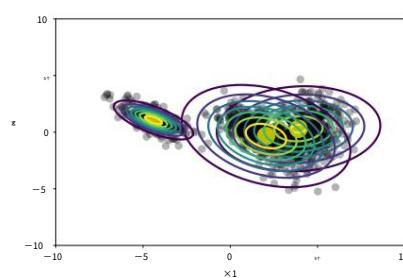
(b) 负对数似然。



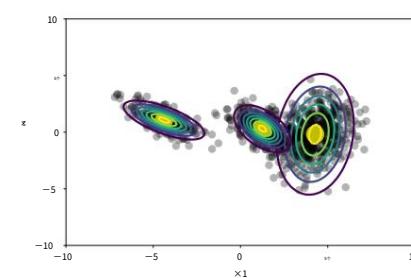
(c) EM 初始化。



(d) 一次迭代后的 EM。



(e) 10 次迭代后的 EM。



(f) 62 次迭代后的 EM。

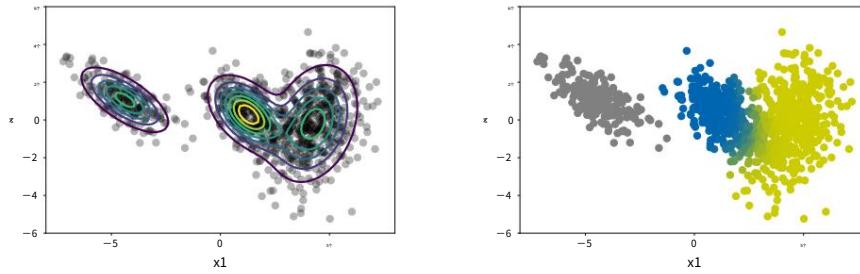
当我们对图 11.1 中的示例运行 EM 时 ,我们在五次迭代后获得如图 11.6(a) 所示的最
终结果 ,图 11.6(b) 显示负对数似然如何作为 EM 迭代的函数演变。最终的 GMM 为

$$p(x) = 0.29N(x \mid -2.75, 0.06) + 0.28N(x \mid -0.50, 0.25) + 0.43N(x \mid 3.64, 1.63) \quad (11.57)$$

我们将 EM 算法应用于图 11.1 所示的二维数据集 ,其中 $K = 3$ 个混合成分。图 11.7 说
明了 EM 算法的一些步骤 ,并显示了作为 EM 迭代函数的负对数似然 (图 11.7(b)) 。图
11.8(a) 显示了

11.4 潜在变量视角

363



(a) 62 次迭代后的 GMM 拟合。

(b) 数据集根据混合成分的责任着色。

图 11.8 EM 收敛时的
GMM
拟合和责任。(a)
当 EM 收敛时 GMM 拟
合; (b) 每个数据点
都根据混合物的
责任进行着色

成分。

相应的最终 GMM 拟合。图 11.8(b) 可可视化了混合成分对数据点的最终响应。当 EM 收敛时,数据集根据混合组件的职责进行着色。虽然单个混合组件显然对左侧的数据负责,但右侧两个数据集群的重叠可能是由两个混合组件生成的。很明显,有些数据点无法唯一分配给单个组件 (蓝色或黄色),因此这两个集群对这些点的责任约为 0.5。

11.4 潜在变量视角

我们可以从离散隐变量模型的角度来看待 GMM,即隐变量 z 只能获得一组有限的值。这与 PCA 形成对比,其中潜在变量是 RM 中的连续值数字。

概率观点的优点是 (i) 它将证明我们在前面部分中做出的一些临时决定, (ii) 它允许将责任具体解释为后验概率,以及 (iii) 迭代算法用于更新模型参数的算法可以原则上推导为潜在变量模型中用于最大似然参数估计的 EM 算法。

11.4.1 生成过程和概率模型

要推导 GMM 的概率模型,考虑生成过程很有用,即允许我们使用概率模型生成数据的过程。

我们假设一个具有 K 个成分的混合模型,并且数据点 x 可以由一个混合成分生成。我们引入了一个二元指示变量 $z_k \in \{0, 1\}$,它有两个状态 (见第 6.2 节),指示第 k 个混合成分是否生成了该数据点

以便

$$p(x | z_k = 1) = N(x | \mu_k, \Sigma_k). \quad (11.58)$$

$z \in \mathbb{R}^K$ 作为一个概率向量,包含我们定义 $z := [z_1, \dots, z_{K-1}, 0]$ 和一个 1。例如,对于 $K = 3$,有效的 z 将是 $[0, 1, 0]$,这将选择第二个混合分量 $z = [z_1, z_2, z_3]$ 因为 $z_2 = 1$ 。

评论。有时,这种概率分布被称为“多努利”,这是伯努利分布对两个以上值的概括 (Murphy, 2012)。 \diamond z 的属性意味着编码 (也: 1-of- K 表示)。

一次性编码
1-of- K
表示

$\sum_{k=1}^K z_k = 1$. 因此, z 是一个 one-hot

到目前为止,我们假设指示变量 z_k 是已知的。如何
曾经,在实践中,情况并非如此,我们放置了一个先验分布

$$p(z) = \pi = [\pi_1, \dots, \pi_K], \quad \sum_{k=1}^K \pi_k = 1, \quad (11.59)$$

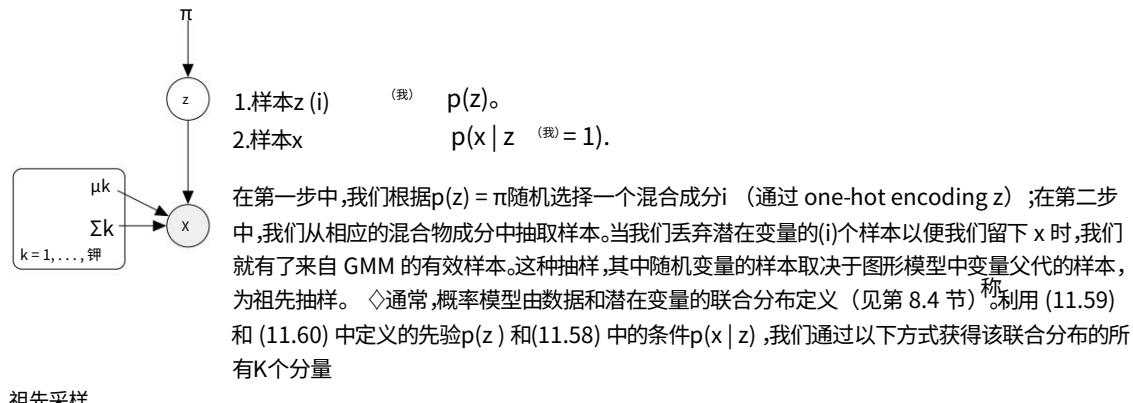
关于潜在变量 z 。然后是第 k 个条目

$$\pi_k = p(z_k = 1) \quad (11.60)$$

该概率向量的描述了第 k 个混合分量生成数据点 x 的概率。

图 11.9 具有
单个数据点的 GMM
的图形模型。

备注 (从 GMM 采样)。这个潜在变量模型的构造 (参见图 11.9 中相应的图形模型) 使其可以通过非常简单的采样过程 (生成过程) 来生成数据:



祖先采样

$$p(x, z_k = 1) = p(x | z_k = 1)p(z_k = 1) = \pi_k N(x | \mu_k, \Sigma_k) \quad (11.61)$$

11.4 潜在变量视角

365

对于 $k = 1, \dots, K$, 所以

$$\begin{aligned} p(x, z_1=1) &= \pi_{1N} x | \mu_1, \Sigma_1 \\ p(x, z) &\vdots \quad = \vdots \\ p(x, z_K=1) &= \pi_{KN} x | \mu_K, \Sigma_K \end{aligned}, \quad (11.62)$$

它完全指定了概率模型。

11.4.2 可能性

为了在潜在变量模型中获得似然 $p(x | \theta)$, 我们需要边缘化潜在变量 (参见第 8.4.3 节)。在我们的例子中, 这可以通过对 (11.62) 中的联合 $p(x, z)$ 的所有潜在变量求和来完成, 这样

$$p(x | \theta) = \sum_z p(x | \theta, z) p(z | \theta), \theta := \{\mu_k, \Sigma_k, \pi_k : k = 1, \dots, K\}. \quad (11.63)$$

我们现在明确地以概率模型的参数 θ 为条件, 这是我们之前省略的。在 (11.63) 中, 我们对 z 的所有 K 个可能的热编码求和, 记为 \sum_z 。由于每个 z 中只有一个非零单个条目, 因此只有 K 个可能的 z 配置/设置。例如, 如果 $K = 3$, 则 z 可以具有以下配置

$$\begin{matrix} & 0 & 0 \\ 0 & & 0 \\ 0 & , & 0 \end{matrix}, \quad (11.64)$$

对 (11.63) 中 z 的所有可能配置求和等同于查看 z 向量的非零项并写出

$$p(x | \theta) = \sum_z p(x | \theta, z) p(z | \theta) \quad (11.65a)$$

$$= \sum_{k=1}^K p(x | \theta, z_k=1) p(z_k=1 | \theta) \quad (11.65b)$$

使得所需的边际分布为

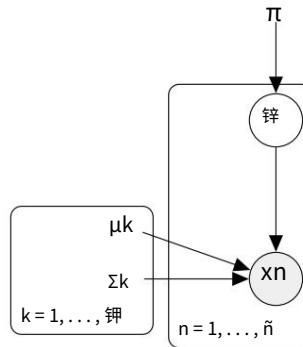
$$p(x | \theta) \stackrel{(11.65b)}{=} \sum_{k=1}^K p(x | \theta, z_k=1) p(z_k=1 | \theta) \quad (11.66a)$$

$$= \sum_{k=1}^K \pi_{kN} x | \mu_k, \Sigma_k, \quad (11.66b)$$

我们将其识别为 (11.3) 中的 GMM 模型。给定数据集 X 立即获得似然

$$p(X | \theta) = \prod_{n=1}^N p(x_n | \theta) \stackrel{(11.66b)}{=} \prod_{n=1}^N \sum_{k=1}^K \pi_{kN} x_n | \mu_k, \Sigma_k, \quad (11.67)$$

图 11.1 具有 N 个数据点的 GMM 的图形模型。



这正是 (11.9) 的 GMM 似然。因此,具有潜在指标 z_k 的潜在变量模型是考虑高斯混合模型的等价方式。

11.4.3 后验分布

让我们简要看一下潜在变量 z 的后验分布。根据贝叶斯定理,生成数据点 x 的第 k 个分量的后验

$$p(z_k = 1 | x) = p(x) \frac{p(z_k = 1)p(x | z_k = 1)}{\sum_{j=1}^K p(z_j = 1)p(x | z_j = 1)}, \quad (11.68)$$

其中边际 $p(x)$ 在 (11.66b) 中给出。这产生了第 k 个指标变量 z_k 的后验分布

$$p(z_k = 1 | x) = \frac{p(z_k = 1)p(x | z_k = 1)}{\sum_{j=1}^K p(z_j = 1)p(x | z_j = 1)} = \frac{\pi_k p(x | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j p(x | \mu_j, \Sigma_j)}, \quad (11.69)$$

我们将其确定为数据点 x 的第 k 个混合分量的责任。请注意,我们省略了 GMM Σ_k 的显式条件,其中 $k = 1, \dots, K$ 。参数 π_k, μ_k ,

11.4.4 扩展到完整数据集

到目前为止,我们只讨论了数据集仅包含单个数据点 x 的情况。但是先验和后验的概念可以直接推广到 N 个数据点 $X := \{x_1, \dots, x_N\}$ 。

在 GMM 的概率解释中,每个数据点 x_n pos sesses 自己的潜在变量

$$z_n = [z_{n1}, \dots, z_K] \in \mathbb{R}^K. \quad (11.70)$$

以前(当我们只考虑单个数据点 x 时),我们省略了索引 n ,但现在这变得很重要。

我们在所有潜在变量 z_n 上共享相同的先验分布 π 。
相应的图形模型如图 11.1 所示,我们在其中使用了板符号。

条件分布 $p(x_1, \dots, x_N | z_1, \dots, z_N)$ 分解数据点并给出如下

$$p(x_1, \dots, x_N | z_1, \dots, z_N) = \prod_{n=1}^N p(x_n | z_n). \quad (11.71)$$

为了获得后验分布 $p(z_{nk} = 1 | x_n)$, 我们遵循与 11.4.3 节相同的推理并应用贝叶斯定理获得

$$p(z_{nk} = 1 | x_n) = \frac{p(x_n | z_{nk} = 1)p(z_{nk} = 1)}{\prod_{j=1}^K p(x_n | z_{nj} = 1)p(z_{nj} = 1)} \quad (11.72a)$$

$$= \frac{\pi_{nk} x_n | \mu_k, \Sigma_k}{\prod_{j=1}^K \pi_{nj} x_n | \mu_j, \Sigma_j} = r_{nk}. \quad (11.72b)$$

这意味着 $p(z_k = 1 | x_n)$ 是第 k 个混合分量生成数据点 x_n 的 (后验) 概率, 对应于我们在 (11.17) 中引入的责任 r_{nk} 。现在, 责任不仅具有直观的解释, 而且具有数学上合理的解释作为后验概率。

11.4.5 重访 EM 算法我们作为最大似然

估计的迭代方案引入的 EM 算法可以从潜在变量的角度以原则性方式推导出来。给定模型参数的当前设置 $\theta(t)$, E-step 计算预期的对数似然

$$Q(\theta | \theta(t)) = E_z [\log p(x, z | \theta)] \quad (11.73a)$$

$$= \log p(x, z | \theta) p(z | x, \theta(t)) dz, \quad (11.73b)$$

其中 $\log p(x, z | \theta)$ 的期望值是关于潜在变量的后验 $p(z | x, \theta(t))$ 的。M 步通过最大化 (11.73b) 选择一组更新的模型参数 $\theta(t+1)$ 。

尽管 EM 迭代确实增加了对数似然, 但不能保证 EM 收敛到最大似然解。

EM 算法可能会收敛到对数似然的局部最大值。可以在多个 EM 运行中使用参数 θ 的不同初始化, 以降低以糟糕的局部最优结束的风险。我们在这里不做进一步的详细介绍, 而是参考 Rogers 和 Girolami (2016) 以及 Bishop (2006) 的精彩阐述。

11.5 延伸阅读

GMM 可以被认为是一种生成模型,因为它可以直接使用祖先采样生成新数据 (Bishop, 2006)。对于给定的 GMM 参数 $\pi_k, \mu_k, \Sigma_k, k = 1, \dots, K$, 我们从概率向量 $[\pi_1, \dots, \pi_K]$ 然后采样一个数据点 $x \sim N(\mu_k, \Sigma_k)$, 如果我们重复这个N次, 我们得到一个数据集 Σ_k 。由 GMM 生成。图 11.1 是使用此过程生成的。

在本章中, 我们假设组件的数量K是已知的。实际上, 情况往往并非如此。然而, 我们可以使用嵌套交叉验证, 如第 8.6.1 节中所讨论的, 来找到好的模型。

高斯混合模型与K 均值聚类算法密切相关。K-means 还使用 EM 算法将数据点分配给集群。如果我们将 GMM 中的均值视为聚类中心并忽略协方差 (或将它们设置为I), 我们将得到K 均值。正如 MacKay (2003) 也很好地描述的那样, K-means 将数据点“硬”分配给聚类中心 μ_k , 而 GMM 通过责任进行“软”分配。

我们只谈到了 GMM 和 EM 算法的潜在变量视角。请注意, EM 可用于一般潜在变量模型中的参数学习, 例如非线性状态空间模型 (Ghahramani 和 Roweis, 1999 年; Roweis 和 Ghahramani, 1999 年) 以及 Barber (2012 年) 讨论的强化学习。因此, GMM 的潜在变量视角有助于以原则性方式推导相应的 EM 算法 (Bishop, 2006; Barber, 2012; Murphy, 2012)。

我们只讨论了寻找 GMM 参数的最大似然估计 (通过 EM 算法)。最大似然的标准批评也适用于此:

- 与线性回归一样, 最大似然可能会遭受严重的过度拟合。在 GMM 的情况下, 当混合分量的均值与数据点相同并且协方差趋于0 时, 就会发生这种情况。然后, 似然趋近于无穷大。Bishop (2006) 和 Barber (2012) 详细讨论了这个问题。
- 对于 $k = 1, \dots, K$, 我们仅获得参数 π_k, μ_k, Σ_k 的点估计。它没有给出参数值不确定性的任何指示。贝叶斯方法会在参数上放置先验, 这可用于获得参数的后验分布。这个后验允许我们计算模型证据 (边际似然), 它可以用于模型比较, 这给了我们一个原则性的方法来确定混合成分的数量。不幸的是, 在这种情况下不可能进行封闭式推理, 因为该模型没有共轭先验。但是, 可以使用变分推理等近似值来获得近似后验概率 (Bishop, 2006)。

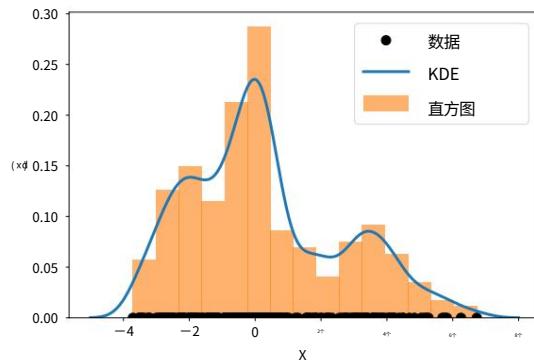


图 11.1 直方图
(橙色条) 和核密度估计
(蓝线)。核密度估计
器产生底层密度的平滑估
计,而直方图是一个
未平滑的计数度量
如何

许多数据点 (黑
色) 落入一个箱
子。

在本章中,我们讨论了用于密度估计的混合模型。
有大量可用的密度估计技术。在实践中,我们经常使用直方图和核密度估计。

直方图

直方图提供了一种表示连续密度的非参数方法,由 Pearson (1895) 提出。直方图是通过“分箱”数据空间和计数(每个分箱中有多少数据点)构建的,然后在每个 bin 的中心绘制一个条形,条形的高度与该 bin 内的数据点数成正比。bin 大小是一个关键的超参数,错误的选择会导致过度拟合和欠拟合。如第 8.2.4 节所述,交叉验证可用于确定合适的 bin 大小。

核密度估计是 Rosenblatt (1956) 和 Parzen (1962) 独立提出的一种非参数密度估计方法。给定 N iid 个样本,核密度估计器将基础分布表示为

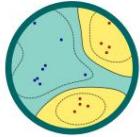
内核密度
估计

$$p(x) = \frac{1}{Nh} \sum_{n=1}^N k\left(\frac{x - x_n}{h}\right), \quad (11.74)$$

其中 k 是核函数,即对 1 求积分的非负函数, $h > 0$ 是平滑/带宽参数,其作用类似于直方图中的 bin 大小。请注意,我们在数据集中的每个数据点 x_n 上放置了一个内核。常用的核函数有均匀分布和高斯分布。核密度估计与直方图密切相关,但通过选择合适的核,我们可以保证密度估计的平滑性。图 11.1 说明了对于包含 250 个数据点的给定数据集,直方图和核密度估计器(具有高斯形核)之间的差异。

12

使用支持向量机进行分类



在许多情况下,我们希望我们的机器学习算法能够预测许多(离散)结果中的一个。例如,电子邮件客户端将邮件分为个人邮件和垃圾邮件,这有两种结果。另一个例子是望远镜,它可以识别夜空中的物体是星系、恒星还是行星。通常只有少数结果,更重要的是这些结果通常没有额外的结构。在本章中,我们考虑输出二进制值的预测变量,即只有两种可能的结果。此机器学习任务称为二进制分类。这与第9章相反,在第9章中我们考虑了具有连续值输出的预测问题。

结构的一个例子是
如果
结果是有序的,就像
在小型、中型和大型的情
况下
 T 息。
二元分类

对于二元分类,标签/输出可以获得的一组可能值是二元的,在本章中我们将它们表示为 $\{+1, -1\}$ 。换句话说,我们考虑形式的预测变量

$$f: \mathbb{R}^D \rightarrow \{+1, -1\}. \quad (12.1)$$

输入示例 x_n 也可以称为
输入、数据点、特征或实例。

回想一下第8章,我们将每个示例(数据点) x_n 表示为 D 个实数的特征向量。这些标签通常分别称为正类和负类。应该注意不要推断+1类的积极性的直观属性。例如,在癌症检测任务中,癌症患者通常被标记为+1。原则上,可以使用任何两个不同的值,例如{True, False}、{0, 1}或{red, blue}。二元分类问题得到了很好的研究,我们将对其他方法的调查推迟到第12.6节。

班级
对于概率模型,使用数
学上方便

{0, 1} 作为二进制表示;
见例 6.12 后的注释。

我们提出了一种称为支持向量机(SVM)的方法,它可以解决二元分类任务。与回归一样,我们有一个监督学习任务,其中我们有一组示例 $x_n \in \mathbb{R}^D$ 以及它们对应的(二进制)标签 $y_n \in \{+1, -1\}$ 。给定一个由示例-标签对组成的训练数据集 $\{(x_1, y_1), \dots, (x_N, y_N)\}$,我们想估计模型的参数,使分类误差最小。与第9章类似,我们考虑一个线性模型,并在示例(9.13)的变换中隐藏非线性。

我们将在12.4节中重新讨论。

SVM在许多应用中提供了最先进的结果,具有可靠的理论保证(Steinwart 和 Christmann, 2008年)。我们选择使用说明二元分类有两个主要原因

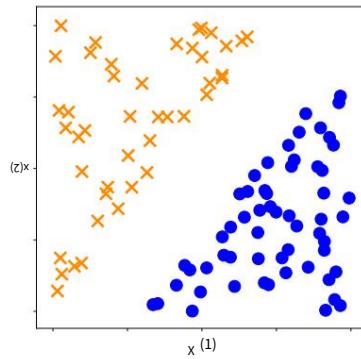


图 12.1示例
2D 数据 ,说明数据的直
觉

我们在哪里可以找到一个
线性分类器
将橙色十字与蓝色圆
盘分开。

支持向量机。首先,SVM 允许以几何方式思考监督机器学习。在第 9 章中,我们从概率模型的角度考虑了机器学习问题,并使用最大似然估计和贝叶斯推理对其进行了攻击,而在这里我们将考虑另一种方法,即对机器学习任务进行几何推理。它在很大程度上依赖于概念,例如我们在第 3 章中讨论的内积和投影。我们发现 SVM 具有指导意义的第二个原因是,与第 9 章相比,SVM 的优化问题不承认解析解,因此我们需要求助于第 7 章中介绍的各种优化工具。

机器学习的 SVM 观点与第 9 章的最大似然观点略有不同。最大似然观点提出了一个基于数据分布概率观点的模型,从中推导出了一个优化问题。相比之下,SVM 视图首先根据几何直觉设计要在训练期间优化的特定函数。我们已经在第 10 章中看到过类似的东西,我们从几何原理中推导出 PCA。在 SVM 案例中,我们首先设计一个损失函数,该损失函数将根据经验风险最小化原则 (第 8.2 节)在训练数据上最小化。

让我们推导出对应于在样本-标签对上训练 SVM 的优化问题。直观地,我们想象二元分类数据,它可以被一个超平面分开,如图 12.1 所示。

这里,每个示例 x_n (2 维向量)是一个二维位置 ($x(1)$ 和 $x(2)$),对应的二进制标签 y_n 是两个不同符号 (橙色圆盘)之一。“超平面”是机器学习中常用的词,我们在 2.8 节中已经遇到过超平面。超平面是维数为 $D - 1$ 的仿射子空间 (如果相应的向量空间为维数 D)。

这些示例由两个类 (有两个可能的标签)组成,它们具有特征 (表示示例的向量的分量),其排列方式允许我们通过绘制一条直线来分离/分类它们。

在下文中,我们将寻找两个类的线性分隔符的想法形式化。我们引入边距的概念,然后扩展线性分隔符以允许示例落在“错误”的一侧,从而导致分类错误。我们提出了两种形式化 SVM 的等效方法:几何视图(第 12.2.4 节)和损失函数视图(第 12.2.5 节)。我们使用拉格朗日乘数推导出 SVM 的对偶版本(第 7.2 节)。对偶 SVM 允许我们观察形式化 SVM 的第三种方式:根据每个类的示例的凸包(第 12.3.2 节)。最后,我们通过简要描述内核以及如何数值求解非线性内核-SVM 优化问题来得出结论。

12.1 分离超平面

给定两个表示为向量 x_i 和 x_j 的示例,计算它们之间相似度的一种方法是使用内积 $x_i \cdot x_j$ 。回想一下 3.2 节,内积与两个向量之间的角度密切相关。两个向量之间的内积值取决于每个向量的长度(范数)。此外,内积允许我们严格定义几何概念,例如正交性和投影。

许多分类算法背后的主要思想是在 RD 中表示数据,然后对该空间进行分区,理想情况下,具有相同标签(且没有其他示例)的示例位于同一分区中。

在二元分类的情况下,空间将分为两部分,分别对应正类和负类。我们考虑一种特别方便的划分,即使用超平面将空间(线性)划分为两半。假设 $x \in RD$ 是数据空间的一个元素。考虑一个函数

$$f: R^d \rightarrow \mathbb{R} \quad (12.2a)$$

$$x \mapsto f(x) := w \cdot x + b, \quad (12.2b)$$

由 $w \in RD$ 和 $b \in \mathbb{R}$ 参数化。回想一下 2.8 节,超平面是仿射子空间。因此,我们将二元分类问题中将两个类分开的超平面定义为

$$x \in R^d : f(x) = 0. \quad (12.3)$$

超平面的图示如图 12.2 所示,其中向量 w 是垂直于超平面的向量, b 是截距。我们可以通过对超平面上选择任意两个样本 x_a 和 x_b 并证明它们之间的向量正交于 w 来推导 w 是 (12.3) 中超平面的法向量。以方程的形式,

$$f(x_a) - f(x_b) = w \cdot x_a + b - (w \cdot x_b + b) \quad (12.4a)$$

$$= w \cdot x_a - w \cdot x_b, \quad (12.4b)$$

12.1 分离超平面

373

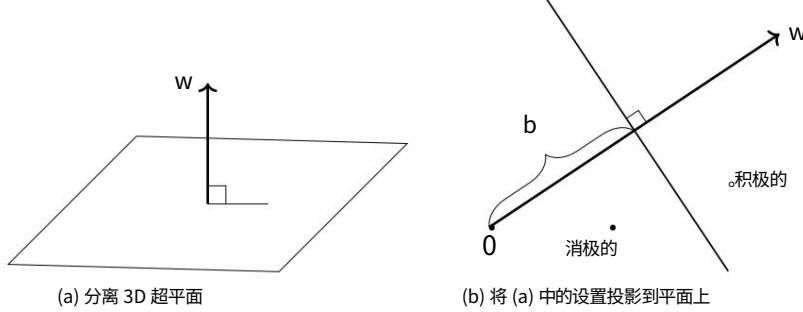


图 12.2 分离超平面方程 (12.3)。(a) 在 3D 中表示方程的标准方法。(b) 为了画图方便,我们看一下超平面的边在。

其中第二行是通过内积的线性度得到的(第3.2节)。由于我们选择了 x_a 和 x_b 在超平面上,这意味着 $f(x_a) = 0$ 和 $f(x_b) = 0$,因此 $w \cdot x_a - w \cdot x_b = 0$ 。

回想一下,当两个向量的内积为零时,它们是正交的。 w 与上的任意向量正交因此,我们得到 w 与超平面上的任意向量正交。超平面。

评论。回想一下第2章,我们可以用不同的方式来思考向量。在本章中,我们将参数向量 w 看成一个指示方向的箭头,即我们将 w 看成一个几何向量。相反,我们将示例向量 x 视为数据点(如其坐标所示),即,我们将 x 视为向量相对于标准基的坐标。◇当出现测试示例时,我们根据它出现在超平面的一侧将示例分类为正例或负例。注意(12.3)不仅定义了一个超平面;它还定义了方向。换句话说,它定义了超平面的正面和负面。因此,为了对测试示例 x_{test} 进行分类,我们计算函数 $f(x_{test})$ 的值,如果 $f(x_{test}) > 0$ 则将示例分类为+1,否则分类为-1。从几何角度考虑,正例位于超平面“上方”,而负例位于超平面“下方”。

在训练分类器时,我们要确保样本与正标签位于超平面的正侧,即

$$w \cdot x_n + b \geq 0 \text{ 当 } y_n = +1 \quad (12.5)$$

带有负标签的例子是负面的,即

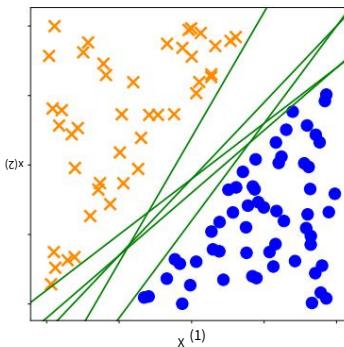
$$w \cdot x_n + b < 0 \text{ 当 } y_n = -1 \text{ 时。} \quad (12.6)$$

参考图12.2的正例和负例的几何直觉。这两个条件通常出现在一个方程式中

$$y_n(w \cdot x_n + b) \geq 0. \quad (12.7)$$

当我们把(12.5)和(12.6)的两边分别乘以 $y_n = 1$ 和 $y_n = -1$ 时,等式(12.7)等价于(12.5)和(12.6)。

图 12.1 可能的分离超平面。有许多线性分类器（绿线）将橙色十字与蓝色圆盘分开。



12.2 原始支持向量机基于点到超平面的距离

的概念，我们现在可以讨论支持向量机。对于数据集 $\{(x_1, y_1), \dots, (x_N, y_N)\}$ 是线性可分的，我们有无限多的候选超平面（参见图 12.1），因此分类器可以解决我们的分类问题而没有任何（训练）错误。为了找到一个唯一的解决方案，一个想法是选择最大化正例和负例之间的间隔的分离超平面。换句话说，我们希望正面和负面的例子之间有很大的距离（第 12.2.1 节）。在下文中，我们计算示例和超平面之间的距离来导出边距。回想一下超平面上到给定点（示例 x_n ）的最近点是通过正交投影获得的（第 3.8 节）。

结果证明，边缘较大的分类器可以很好地泛化
(Steinwart 和 Christmann, 2008)。

12.2.1 保证金的概念

个分离超平面到数据集中最接近的示例。或更接近超平面的例子。

margin 的概念直观上很简单：它是 margin 的距离。假设数据集是线性可分的，则可以有两个超平面到数据集中最接近的示例。然而，当试图将这个距离形式化时，存在一个可能令人困惑的技术问题。技术上的问题是需要定义一个尺度来测量距离。一个潜在的尺度是考虑数据的尺度，即 x_n 的原始值。这有问题，因为我们可以更改 x_n 的测量单位并更改 x_n 中的值，从而更改到超平面的距离。正如我们很快就会看到的，我们根据超平面 (12.3) 本身的方程来定义尺度。

考虑一个超平面 $w \cdot x + b$ 和一个示例 x_a ，如图 12.2 所示。不失一般性，我们可以认为样本 x_a 在超平面的正侧，即 $w \cdot x_a + b > 0$ 。我们想计算 x_a 到超平面的距离 $r > 0$ 。我们通过考虑 x_a 到上的正交投影（第 3.8 节）来做到这一点。由于 w 正交于

超平面，我们用 x 表示

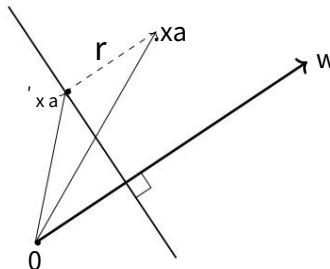


图 12.2 矢量加法表示到超平面的距离：

$$xa = x - \frac{r}{\|w\|} w$$

超平面,我们知道距离r只是这个向量w 的缩放。

如果w的长度已知,那么我们可以使用这个比例因子r factor 计算出xa和x之间的绝对距离
为了方便,

我们选择使用单位长度的向量 (其范数为1) ,并通过将w除以其范数 $\|w\|$ 来获得。使用向量加法 (第 2.4 节) ,我们得到

$$xa = x - r \frac{w}{\|w\|}. \quad (12.8)$$

另一种思考r的方式是,它是xa在 $w/\|w\|$ 所跨过的子空间中的坐标。我们现在已经将xa到超平面的距离表示为r,如果我们选择xa作为离超平面最近的点,这个距离r就是margin。

回想一下,我们希望正样本距超平面的距离大于r ,而负样本距超平面的距离大于r (负方向)。类似于将 (12.5)和 (12.6)组合成 (12.7) ,我们将这个目标表述为

$$y_n(w, x_n + b) \geq r. \quad (12.9)$$

换句话说,我们将样本距离超平面 (在正负方向上)至少r的要求组合成一个不等式。

由于我们只对方向感兴趣,因此我们在模型中添加了一个假设,即参数向量w具有单位长度,即 $\|w\| = 1$,其中我们使用欧几里得范数 $\|w\| = \sqrt{w \cdot w}$ (第3.1).我们将看到其他假设也允许对距离r (12.8) 进行更直观的解释,因为它是长度为 1 的向量的比例因子。

内心的选择

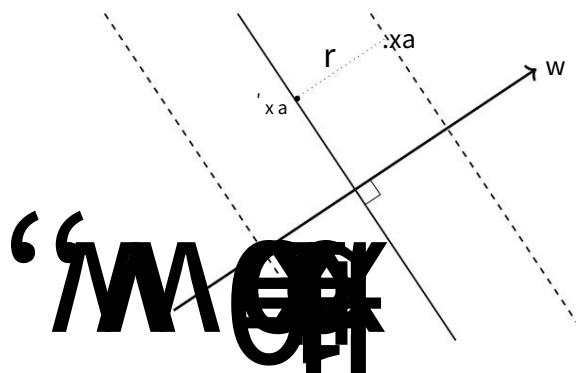
12.4 节中
的产品 (第 3.2
节)。

评论。如果 SVM 是由 Scholkopf 和 Smola (2002) 提供的,那么熟悉其他边距表示的读者会注意到我们对 $\|w\| = 1$ 的定义不同于标准表示。在第 12.2.3 节中,我们将展示这两种方法的等价性。 ◇

将三个要求收集到一个约束优化中

图 12.1 的推导

$$\text{保证金 } r = \frac{\|w\|}{\|w\|}.$$



问题，我们得到目标

$$\begin{array}{c} \text{最大} \\ w, b, r \\ \text{利润} \\ \text{服从} y_n(w, x_n + b) \end{array} \quad \frac{r}{\|w\| = 1} \quad \begin{array}{c} \text{数据拟合} \\ \text{———} \end{array}, \quad r > 0, \quad \begin{array}{c} \text{正常化} \\ \text{———} \end{array} \quad (12.10)$$

这表示我们想要最大化边距 r ，同时确保数据位于超平面的正确一侧。

评论。边缘的概念在机器学习中非常普遍。Vladimir Vapnik 和 Alexey Chervonenkis 使用它来表明当边缘较大时，函数类的“复杂性”较低，因此学习是可能的（Vapnik, 2000）。事实证明，该概念可用于从理论上分析泛化误差的各种不同方法（Steinwart 和 Christmann, 2008 年；Shalev-Shwartz 和 Ben-David, 2014 年）。◇

12.2.2 边距的传统推导

在上一节中，我们通过观察只对 w 的

方向而不是它的长度感兴趣，从而得出了 (12.10)，这导致了 $\|w\| = 1$ 的假设。在这个部分，我们通过不同的假设推导了边距最大化问题。我们没有选择对参数向量进行归一化，而是为数据选择一个尺度。

我们选择这个尺度，使得预测变量 $w, x + b$ 的值在最接近的示例中为 1。我们还用 x_A 表示数据集中最接近超平面的示例。

回想一下我们
目前考虑线性可分数据。

图 12.1 与图 12.2 相同，只是现在我们重新缩放了轴，这样示例 x_A 正好位于边缘上，即 $w, x_A + b = 1$ 。因为 x 是 x_A 在超平面上的正交投影，根据定义它必须位于超平面上，即

A

$$w, x_A + b = 0. \quad (12.11)$$

通过将 (12.8) 代入 (12.11), 我们得到

$$w, xa - r \frac{w}{\|w\|} + b = 0. \quad (12.12)$$

利用内积的双线性 (见第 3.2 节), 我们得到

$$w, xa + b - r \frac{w, w}{\|w\|} = 0. \quad (12.13)$$

根据我们的比例假设, 观察到第一项为 1, 即 $w, xa + b = 1$ 。从 3.1 节的 (3.16), 我们知道 $w, w = \|w\|^2$ 。第二项简化为 $r = \|w\|$ 。使用这些简化, 我们得到

$$r = \frac{\|w\|}{\|w\|} = 1. \quad (12.14)$$

这意味着我们根据法向量 w 推导出距离 r

的超平面。乍一看, 这个等式是违反直觉的, 因为我们也可以认为似乎已经根据向量 w 的长度导出了与超平面的距离, 但我们还不知道这个向量。考虑它的一种方法是将距离 r 视为我们仅用于此推导的临时变量。因此, 对于本节的其余部分, 我们将用在第 12.2.3 节中表示到超平面的距离,

$$\frac{1}{\|w\|}$$

我们会看到, margin 等于 1 的选择等同于我们之前在 12.2.1 节中假设 $\|w\| = 1$ 。

与获得 (12.9) 的论证类似, 我们希望正例和负例与超平面的距离至少为 1, 从而产生条件

$$y_n(w, x_n + b) \geq 1. \quad (12.15)$$

将边距最大化与示例需要位于超平面的正确一侧 (基于它们的标签) 这一事实相结合, 我们得到了

$$\text{最大 } w, b \frac{1}{\|w\|} \quad (12.16)$$

$$\text{服从 } y_n(w, x_n + b) \geq 1 \text{ 对于所有 } n = 1, \dots, N. \quad (12.17)$$

我们不是像 (12.16) 那样最大化范数的倒数, 而是经常最小化平方范数。我们还经常包括一个常数, 它不会影响最优的 w, b , 但在我们计算梯度时会产生更整洁的形式。那么, 我们 $\frac{1}{2} \|w\|^2$ 平方范数导致 SVM 的凸二次规划问题 (第 12.5 节)。

$$\text{最小 } w, b \frac{1}{2} \|w\|^2 \quad (12.18)$$

$$\text{受制于 } y_n(w, x_n + b) \geq 1 \text{ 对于所有 } n = 1, \dots, N. \quad (12.19)$$

方程 (12.18) 称为硬间隔 SVM。硬边际 SVM 表达“硬”的原因是该公式不允许任何违反边际条件的情况。我们将在第 12.2.4 节中看到, 这

如果数据是
不是线性可分的。

12.2.3 为什么我们可以将边距设置为1在

12.2.1 节中,我们讨论了我们想要最大化某个值 r ,它表示最近示例到超平面的距离。

在第 12.2.2 节中,我们对数据进行了缩放,使得最近的示例与超平面的距离为1。在本节中,我们将这两个推导联系起来,并证明它们是等价的。

定理 12.1。最大化边距 r ,其中我们考虑(12.10)中的归一化权重,

$$\begin{array}{c} \text{最大 } r \\ w, b \\ \text{利润} \\ \text{服从 } y_n(w, x_n + b) - \frac{r}{\|w\|} = 1 \\ \text{数据拟合} \end{array}, \quad r > 0, \quad (12.20)$$

—— —— 正常化

相当于缩放数据,使得边距是统一的:

$$\begin{array}{c} \text{最小 } \frac{1}{\|w\|^2} \\ w, b \\ \text{利润} \\ \text{受制于 } y_n(w, x_n + b) - \frac{1}{\|w\|^2} \end{array}. \quad (12.21)$$

数据拟合

证明考虑 (12.20)。由于平方是非负参数的严格单调变换,因此如果我们在目标中考虑 r ,则最大值保持不变。由于 $\|w\| = 1$,我们可以使用新的权重向量 w' 重新参数化方程,该向量未通过显式使用 $\|w'\|$ 进行归一化。我们获得

$$\begin{array}{c} \text{最大 } 2r \\ w', b, r \\ \text{受制于 } y_n \frac{w'}{\|w'\|}, x_n + b - r, r > 0. \end{array} \quad (12.22)$$

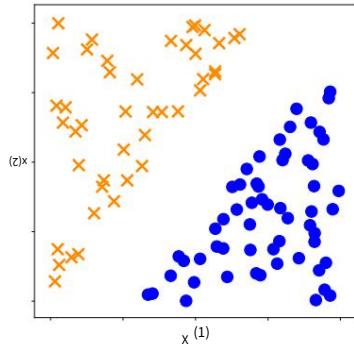
等式 (12.22) 明确指出距离 r 是正的。因此,我们可以将第一个约束条件除以 r ,得到

注意 $r > 0$ 因为我们

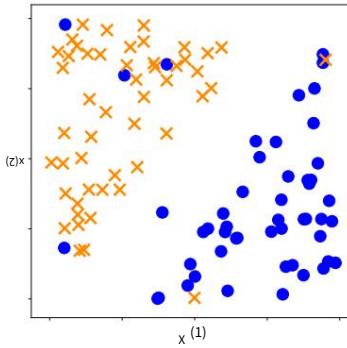
假定线性可分性,
因此除以 r 没有问题。

$$\begin{array}{c} \text{最大 } 2r \\ w', b, r \\ \text{受制于 } y_n \frac{w'}{\|w'\|/r}, x_n + \frac{b}{r} - 1, r > 0 \end{array} \quad (12.23)$$

12.2 原始支持向量机



(a) 线性可分的数据,具有较大的margin



(b) 非线性可分数据

图 12.2 (a) 线性可分和 (b) 非线性可分数据。

将参数重命名为 w' 和 b' 。由于 $\|w''\| = \|w'\| / r$, 重新排列 r 得到 $\frac{w'}{\|w'\|}$

$$\|w''\| = \frac{\sqrt{w' \cdot w'}}{\|w'\|} = \frac{\sqrt{r^2}}{r} \cdot \frac{\sqrt{w' \cdot w'}}{\|w'\|} = \frac{\sqrt{r^2}}{r} \cdot \frac{w'}{\|w'\|} = \frac{w'}{\|w'\|}. \quad (12.24)$$

将这个结果代入(12.23),我们得到

$$\frac{\max(w')}{\|w'\|} - \frac{\min(w')}{\|w'\|} \geq \frac{1}{r} \quad (12.25)$$

服从 $y_n(w'', x_n) + b'' \geq 1$ 。

1最后一步是观察最大化会产生相同的解 $\|w''\|$,从而得出定理 12.1 的证明。

作为最小化 $\frac{1}{2} \|w''\|^2$,

□

12.2.4 Soft Margin SVM: 几何视图在数据不可线性分离的情况下

我们可能希望允许一些示例落在边缘区域内,或者甚至位于超平面的错误一侧,如图 12.2 所示。

允许一些分类错误的模型称为 soft margin SVM。保证金支持向量机。在本节中,我们使用几何参数推导了由此产生的优化问题。在 12.2.5 节中,我们将使用损失函数的思想推导一个等价的优化问题。使用 Lagrange 乘数(第 7.2 节),我们将在第 12.3 节中推导 SVM 的对偶优化问题。这个对偶优化问题使我们能够观察到 SVM 的第三种解释:作为一个超平面,它将对应于正负数据示例的凸包之间的线一分为二(第 12.3.2 节)。

关键的几何思想是引入一个松弛变量 ξ_n 对应的松弛变量
每个示例-标签对 (x_n, y_n) 允许特定示例位于超平面的边界内或甚至错误的一侧(请参阅

图 12.2 Soft margin
SVM 允许样本位于边界
内或超平面的错误
一侧。松弛变量 ξ 衡量

正例 x^+ 到正边
缘超平面 w 的距
离, $x^+ + b = 1$ 当
 x^+ 在错误的一侧时。

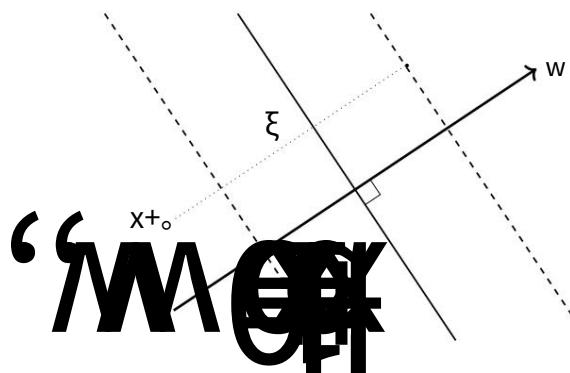


图 12.2)。我们从边距中减去 ξ_n 的值,将 ξ_n 约束为非负数。为了鼓励样本的正确分类,我们将 ξ_n 添加到目标中

$$\text{最小 } \frac{1}{2} \|w\|^2 + C \sum_{n=1}^N \xi_n \quad \text{否} \quad (12.26a)$$

$$\text{受制于 } y_n(w, x_n + b) - 1 - \xi_n \geq 0 \quad (12.26b)$$

$$(12.26c)$$

软间隔支持向量机
正则化参数
正则化器

对于 $n = 1, \dots, N$, 与硬间隔 SVM 的优化问题 (12.18) 相比,这个问题称为软间隔 SVM。参数 $C > 0$ 权衡了保证金的大小和我们拥有的松弛总量。该参数称为正则化参数,因为正如我们将在下一节中看到的那样,目标函数 (12.26a) 中的间隔项是正则化项。边缘项 $\|w\|$ 称为正则项,在许多关于数值优化的书籍中,正则化参数与此项相乘(第 8.2.3 节)。这与我们在本节中的表述形成对比。此处较大的 C 值意味着较低的正则化,因为我们赋予松弛变量更大的权重,因此更优先考虑不位于边缘正确一侧的示例。

此正则化有
替代参数化,
即
为什么 (12.26a) 也常被
称为
C-支持向量机。

评论。在 soft margin SVM (12.26a) 的公式中, w 被正则化,但 b 没有被正则化。我们可以观察正则化项不包含 b 来看到这一点。非正则化项 b 使理论分析变得复杂 (Steinwart 和 Christmann,2008 年,第 1 章) 并降低了计算效率 (Fan 等人,2008 年)。◇

12.2.5 Soft Margin SVM 损失函数观点让我们考虑一种不同的方法来推导 SVM,遵循经验风险最小化的原则(第 8.2 节)。对于 SVM,我们

选择超平面作为假设类,即

$$f(x) = w \cdot x + b. \quad (12.27)$$

我们将在本节中看到边距对应于正则化项。剩下的问题是,什么是损失函数?在第 9 章的损失函数中,我们考虑回归问题(预测器的输出是实数),在本章中,我们考虑二元分类问题(预测器的输出是两个标签之一{+1, -1})。因此,每个样本-标签对的误差/损失函数需要适用于二元分类。例如,用于回归(9.10b)的平方损失不适用于二元分类。

评论。二进制标签之间的理想损失函数是计算预测和标签之间的不匹配数。这意味着对于应用于示例 x_n 的预测变量 f ,我们将输出 $f(x_n)$ 与标签 y_n 进行比较。如果它们匹配,我们将损失定义为零;如果它们不匹配,我们将损失定义为一。这由 $\ell(f(x_n)) = y_n$ 表示,称为零一损失。不幸的是,零一损失导致了寻找最佳参数 w 、 b 的组合零一损失优化问题。组合优化问题(与第 7 章中讨论的连续优化问题相反)通常更难解决。◇ SVM 对应的损失函数是什么?考虑预测器 $f(x_n)$ 的输出与标签 y_n 之间的误差。损失描述了在训练数据上产生的错误。推导(12.26a)的等效方法是使用铰链损失

铰链损失

$$\ell(t) = \max\{0, 1 - t\} \text{ 其中 } t = y f(x) = y(w \cdot x + b). \quad (12.28)$$

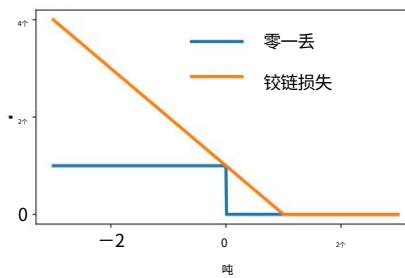
如果 $f(x)$ 在超平面的正确一侧(基于相应的标签 y),并且比距离 1 远,这意味着 $t > 1$ 并且铰链损失返回零值。如果 $f(x)$ 在正确的一侧但太靠近超平面($0 < t < 1$),则示例 x 在边界内,铰链损失返回正值。当样本位于超平面的错误一侧($t < 0$)时,铰链损失返回一个更大的值,该值呈线性增加。换句话说,一旦我们比超平面的边缘更近,我们就会付出代价,即使预测是正确的,惩罚也会线性增加。表达铰链损失的另一种方法是将其视为两个线性部分

$$\ell(t) = \begin{cases} 0 & \text{如果 } t \geq 1 \\ 1 - t & \text{如果 } t < 1 \end{cases}, \quad (12.29)$$

如图 12.2 所示。hard margin SVM 12.18 对应的 loss 定义为

$$\ell(t) = \begin{cases} 0 & \text{如果 } t \geq 1 \\ \infty & \text{如果 } t < 1 \end{cases}. \quad (12.30)$$

图 12.2 铰链损失是零一损失的凸上界。



这种损失可以解释为不允许任何样本位于边距内。

对于给定的训练集 $\{(x_1, y_1), \dots, (x_N, y_N)\}$, 我们寻求最小化总损失, 同时使用 ℓ_2 -正则化对目标进行正则化 (参见第 8.2.3 节)。使用铰链损失 (12.28) 给出了无约束优化问题

$$\begin{array}{ll} \text{最小} & \frac{1}{2} \|w\|^2 + C \max_{n=1}^N \{0, 1 - y_n(w \cdot x_n + b)\} \\ w, b & \end{array} \quad (12.31)$$

正则化器 错误项

正则化
损失期限
错误项

(12.31) 中的第一项称为正则化项或正则化器 (见第 8.2.3 节), 第二项称为损失项或误差项。回想一下 12.2.4 节, 该术语直接来自边缘。换句话说, 边距最大化可以解释为正则化。

$$\frac{1}{2} \|w\|^2$$

正则化

原则上, (12.31) 中的无约束优化问题可以直接用 7.1 节中描述的 (子) 梯度下降法求解。要看出 (12.31) 和 (12.26a) 是等价的, 观察铰链损失 (12.28) 本质上由两个线性部分组成, 如 (12.29) 所示。考虑单个示例标签对 (12.28) 的铰链损失。

我们可以等效地用具有两个约束的松弛变量 ξ 的最小化替换 t 上铰链损失的最小化。在方程式中,

$$\min_t \max_{n=1}^N \{0, 1 - t\} \quad (12.32)$$

相当于

$$\begin{array}{ll} \text{最小} & \xi \\ \xi, t & \end{array} \quad (12.33)$$

服从 $\xi \geq 0, \xi \leq 1 - t$ 。

通过将此表达式代入 (12.31) 并重新排列其中一个约束, 我们准确地获得了软间隔 SVM (12.26a)。

评论。让我们将本节中损失函数的选择与

第 9 章中线性回归的损失函数。回想一下第 9.2.1 节, 为了找到最大似然估计, 我们通常最小化

负对数似然。此外,由于具有高斯噪声的线性回归的似然项是高斯分布的,因此每个示例的负对数似然是平方误差函数。平方误差函数是寻找最大似然解时最小化的损失函数。 ◇

12.3 双支持向量机

前面几节中关于变量 w 和 b 的 SVM 的描述被称为原始 SVM。回想一下,我们考虑具有 D 个特征的输入 $x \in \mathbb{R}^D$ 。由于 w 与 x 具有相同的维度,这意味着优化问题的参数数量(w 的维度)随特征数量线性增长。

在下文中,我们考虑一个与特征数量无关的等效优化问题(所谓的对偶视图)。相反,参数的数量随着训练集中示例的数量而增加。我们在第10章中看到了类似的想法,我们以一种不随特征数量缩放的方式来表达学习问题。这对于我们拥有的特征多于训练数据集中示例数量的问题很有用。双SVM还具有额外的优势,即它可以轻松地应用内核,正如我们将在本章末尾看到的那样。“对偶”一词经常出现在数学文献中,在这种特殊情况下,它指的是凸对偶性。以下小节本质上是凸对偶性的应用,我们在第7.2节中对此进行了讨论。

12.3.1 通过拉格朗日乘数的凸对偶性回想一下原始软间隔

SVM (12.26a)。我们称原始SVM对应的变量 w 、 b 、 ξ 为原始变量。我们使用 α_n 在第7章中,我们将0作为样本被正确分类的约束 (12.26b)对应的拉格朗日乘子, γ_n 非负性约束对应的拉格朗日乘子;参见 (12.26c)。然后给出拉格朗日量

使用 λ 作为拉格朗日乘数。在本节中，我们遵循 SVM 文献中常用的符号，并使用 α 和 γ 。

通过分别对三个原始变量 w 、 b 和 ξ 对拉格朗日量 (12.34) 进行微分, 我们得到

$$\frac{\partial L}{\partial w} = w - \sum_{n=1}^N \alpha_n y_n x_n, \quad (12.35)$$

$$\frac{\partial L}{\partial b} = -\sum_{n=1}^N \alpha_n, \quad (12.36)$$

$$\frac{\partial L}{\partial \xi_n} = C - \alpha_n - \gamma_n. \quad (12.37)$$

我们现在通过将这些偏导数中的每一个设置为零来找到拉格朗日量的最大值。通过将 (12.35) 设置为零, 我们发现

$$w = \sum_{n=1}^N \alpha_n y_n x_n, \quad (12.38)$$

代表定理是代表定理的一个特例 (Kimeldorf 和 Wahba, 1970)。等式 (12.38) 指出, 表示者定理中的最优权重向量实际数据的范围。此上是原始的, 是示例 x_n 的线性组合。回想第 2.6.1 节的集合, 这意味着优化问题的解决方案在于训练外, 通过将 (12.36) 设置为零获得的约束意味着最佳权重向量是示例的仿射组合。事实证明, 代表定理适用于正则化经验风险最小化的非常一般的设置 (Hofmann 等人, 2008 年; Argyriou 和 Dinuzzo, 2014 年)。该定理有更一般的版本定理说 (Scholkopf et al., 2001), 其存在的充分必要条件可以在 Yu et al. (2013)。

最小化经验

风险在于示例定义的
子空间 (第
2.4.3 节)。

评论。表示定理 (12.38) 也提供了对“支持向量机”这个名称的解释。对应参数 $\alpha_n = 0$ 的示例 x_n 对解 w 根本没有贡献。其他示例, 其中 $\alpha_n > 0$, 称为支持向量, 因为它们“支持”超平面。 ◇

支持向量

通过将 w 的表达式代入拉格朗日量 (12.34), 我们
获得双

$$D(\xi, \alpha, \gamma) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j - \sum_{i=1}^N \alpha_i y_i + \sum_{i=1}^N \alpha_i \xi_i - \sum_{i=1}^N \alpha_i \xi_i - \sum_{i=1}^N \alpha_i \xi_i. \quad (12.39)$$

请注意, 不再有任何涉及原始变量 w 的项。

通过将 (12.36) 设置为零, 我们得到 $\sum_{n=1}^N \alpha_n = 0$ 。因此, 涉及 b 的项也消失了。回想一下, 内积是对称的并且

双线性（见第 3.2 节）。因此，(12.39) 中的前两项针对相同的对象。这些项（蓝色）可以简化，我们得到拉格朗日

$$D(\xi, \alpha, \gamma) = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j x_i \cdot x_j + \sum_{i=1}^n \alpha_i + (C - \sum_{i=1}^n \alpha_i - \gamma_i) \xi_i. \quad (12.40)$$

该等式的最后一项是包含松弛变量 ξ_i 的所有项的集合。通过将 (12.37) 设置为零，我们看到 (12.40) 中的最后一项也为零。此外，通过使用相同的方程式并回顾拉格朗日乘数 γ_i 是非负的，我们得出结论 $\alpha_i \leq C$ 。

我们现在得到 SVM 的对偶优化问题，它专门用拉格朗日乘数 α_i 表示。回忆拉格朗日对偶性（定义 7.1），我们最大化了对偶问题。

这相当于最小化负对偶问题，这样我们最终得到对偶 SVM

$$\begin{aligned} & \text{分钟 } \alpha \quad \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j x_i \cdot x_j - \sum_{i=1}^n \alpha_i \\ & \text{受制于 } y_i \alpha_i = 0 \\ & \quad 0 \leq \alpha_i \leq C \text{ 对于所有 } i = 1, \dots, n. \end{aligned} \quad (12.41)$$

(12.41) 中的等式约束是通过将 (12.36) 设置为零获得的。不等式约束 $\alpha_i \geq 0$ 是施加在不等式约束的拉格朗日乘数上的条件（第 7.2 节）。上一段讨论了不等式约束 $\alpha_i \leq C$ 。

SVM 中的不等式约束集合称为“框约束”，因为它们将拉格朗日乘数的向量 $\alpha = [\alpha_1, \dots, \alpha_n] \in \mathbb{R}^n$ 限制在每个轴上由 0 和 C 定义的框内。这些轴对齐的框在数值求解器中特别有效（Dostal, 2009, 第 5 章）。

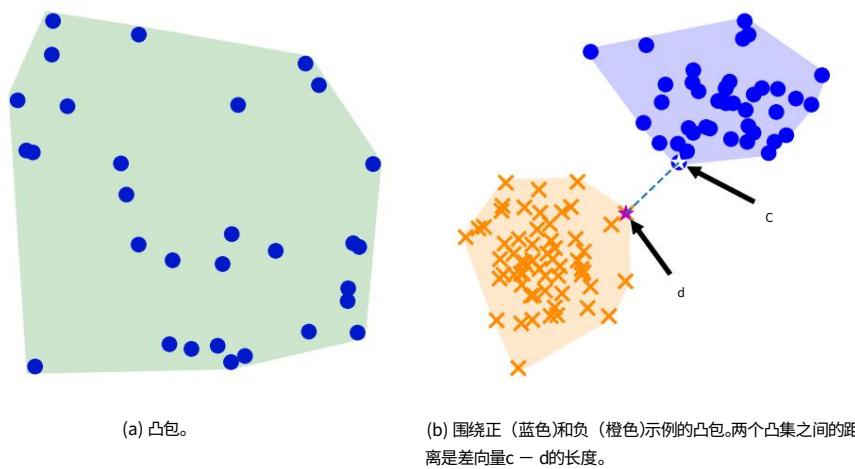
一旦我们获得对偶参数 α ，我们就可以使用表示定理 (12.38) 恢复原始参数 w 。让我们称最佳原始参数为 w^* 。然而，如何获得参数 b 仍然是一个问题。考虑一个正好位于边距边界上的示例 x_n ，即 $w^* \cdot x_n + b = y_n$ 。回想一下， y_n 是 +1 或 -1。因此，唯一的未知数是 b ，可以通过以下方式计算

$$b^* = y_n - w^* \cdot x_n. \quad (12.42)$$

评论。原则上，可能没有完全位于边缘的示例。在这种情况下，我们应该计算 $|y_n - w^* \cdot x_n|$ 对于所有的支持向量，取这个绝对值差的中值为

事实证明
恰好位于边缘的示例
是双参数严格位于
框约束内的示
例， $0 < \alpha_i < C$ 。这是使
用 Karush Kuhn Tucker
条件得出的，例如
Scholkopf 和
Smola (2002)。

图 12.3 凸包。(a) 点的凸包, 其中一些位于边界内; (b) 正例和负例周围的凸包。



$b^* \in \mathbb{R}$. 可以在 <http://fouryears.net/~tjones/mml/> 中找到它的推导。 ◇
2012/06/07/the-svm-bias-term-conspiracy/ 的值。

12.3.2 双SVM: 凸壳视图

获得对偶 SVM 的另一种方法是考虑替代几何参数。考虑具有相同标签的示例集 x_n 。

我们想构建一个包含所有示例的凸集,使其成为可能的最小集。这称为凸包,如图 12.3 所示。

让我们首先建立一些关于点的凸组合的直觉。

考虑两个点 x_1 和 x_2 以及相应的非负权重 $\alpha_1, \alpha_2 \geq 0$, 使得 $\alpha_1 + \alpha_2 = 1$ 。等式 $\alpha_1 x_1 + \alpha_2 x_2$ 描述了 x_1 和 x_2 之间直线上的每个点。考虑当我们添加第三个点 x_3 以及权重 $\alpha_3 \geq 0$ 使得 $\alpha_1 + \alpha_2 + \alpha_3 = 1$ 时会发生什么。

凸包

这三个点 x_1, x_2, x_3 的凸组合跨越一个二维区域。该区域的凸包是由每对点对应的边形成的三角形。随着我们添加更多的点,并且点的数量变得大于维度的数量,一些点将在凸包内部,如图 12.3(a) 所示。

一般来说,构建一个凸包可以通过引入对应于每个示例 x_n 的非负权重 $\alpha_n \geq 0$ 来完成。那么凸包可以描述为集合

$$\text{转换}(X) = \left\{ \sum_{n=1}^N \alpha_n x_n \mid \sum_{n=1}^N \alpha_n = 1 \text{ 且 } \alpha_n \geq 0, n=1, \dots, N \right\}, \quad (12.43)$$

对于所有 $n = 1, \dots, N$, 如果正负类对应的两个点云是分开的,那么凸包不重叠。给定训练数据 $(x_1, y_1), \dots, (x_N, y_N)$, 我们形成两个凸包,分别对应正类和负类。

我们选择一个点 c , 它位于正例集的凸包中, 并且最接近负类分布。类似地, 我们在负样本集的凸包中选择一个点 d , 并且最接近正类分布; 见图 12.3(b)。我们将 d 和 c 之间的差异向量定义为

$$w := c - d。 \quad (12.44)$$

像前面的情况那样选取点 c 和 d , 并要求它们彼此最接近, 相当于最小化 w 的长度/范数, 所以我们最终得到相应的优化问题

$$\text{小参数}_{\frac{1}{w}} \quad \|w\| = \text{最小参数}_{\frac{1}{w}} \quad \frac{\|w\|^2}{2}。 \quad (12.45)$$

由于 c 必须在正凸包中, 所以可以表示为正例的凸组合, 即对于非负系数

$$c = \sum_{n:y_n=+1} \alpha_n^+ x_n。 \quad (12.46)$$

在 (12.46) 中, 我们使用符号 $n : y_n = +1$ 来表示 $y_n = +1$ 的下标集 n 。类似地, 对于带有负标签的例子, 我们得到

$$d = \sum_{n:y_n=-1} \alpha_n^- x_n。 \quad (12.47)$$

将(12.44)、(12.46)和(12.47)代入(12.45), 我们得到目标

$$\frac{1}{2} \sum_{n:y_n=+1} \alpha_n^+ x_n - \sum_{n:y_n=-1} \alpha_n^- x_n。 \quad (12.48)$$

令 α 为所有系数的集合, 即 α^+ 和 α^- 的串联。
回想一下, 我们要求每个凸包的系数之和为 1,

$$\sum_{n:y_n=+1} \alpha_n^+ = 1 \text{ 和 } \sum_{n:y_n=-1} \alpha_n^- = 1。 \quad (12.49)$$

这意味着约束

$$\sum_{n=1}^N y_n \alpha_n = 0。 \quad (12.50)$$

通过将各个类相乘可以看出此结果

$$\begin{aligned} \text{否} \\ \sum_{n=1}^N y_n \alpha_n &= \sum_{n:y_n=+1} (+1) \alpha_n^+ - \sum_{n:y_n=-1} (-1) \alpha_n^- \quad (12.51a) \end{aligned}$$

$$= \sum_{n:y_n=+1} \alpha_n^+ - \sum_{n:y_n=-1} \alpha_n^- = 1 - 1 = 0. \quad (12.51b)$$

目标函数 (12.48) 和约束 (12.50), 以及 $\alpha \geq 0$ 的假设, 给我们一个约束 (凸) 优化问题。这个优化问题可以证明与对偶硬间隔 SVM 的问题相同 (Bennett 和 Bredensteiner, 2000a)。

评论。为了获得软边距对偶, 我们考虑了缩小的船体。缩减包类似于凸包, 但对系数 α 的大小有上限。 α 元素的最大可能值限制了凸包的大小。换句话说, α 上的界限将凸包缩小到更小的体积 (Bennett 和 Bredensteiner, 2000b)。 ◇

减少船体

12.4 内核

考虑对偶 SVM (12.41) 的公式。请注意, 目标中的内积仅出现在示例 x_i 和 x_j 之间。

示例和参数之间没有内积。

因此, 如果我们考虑一组特征 (x_i) 来表示 x_i , 对偶 SVM 的唯一变化将是替换内积。这种模块化, 其中分类方法 (SVM) 的选择和特征表示 (x) 的选择可以分开考虑, 为我们独立探索这两个问题提供了灵活性。在本节中, 我们将讨论表示形式 (x) 并简要介绍核的概念, 但不涉及技术细节。

由于 (x) 可能是一个非线性函数, 我们可以使用 SVM (假设一个线性分类器) 来构造样本 x_n 中非线性的分类器。除了 soft margin 之外, 这为用户提供了第二条途径来处理非线性可分的数据集。事实证明, 有许多算法和统计方法具有我们在对偶 SVM 中观察到的这种特性: 唯一的内积是那些发生在样本之间的内积。我们没有明确定义非线性特征映射 (\cdot) 并计算结果内积, 而是在示例 x_i 和 x_j 之间、 x_i 和 x_j 之间定义相似度函数 $k(x_i, x_j)$ 。对于某一类相似性函数, 称为内核, 相似性函数隐式定义了一个非线性特征映射 (\cdot) 。

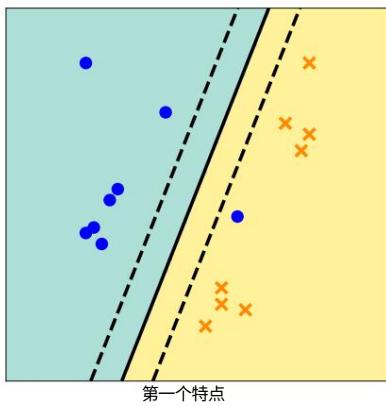
核心

内核的输入 X 是定义函数 $k : X \times X \rightarrow \mathbb{R}$ 存在
核函数可以非常通用, 不一定局限于 \mathbb{R}^d 。一个希尔伯特空间 H 和一个特征图使得

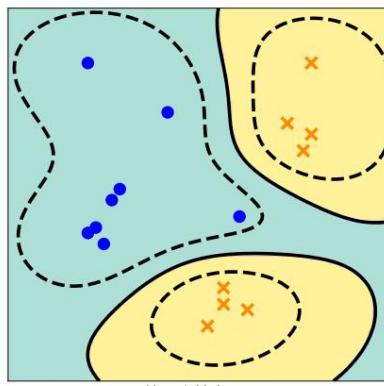
$$k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle_H. \quad (12.52)$$

12.4 内核

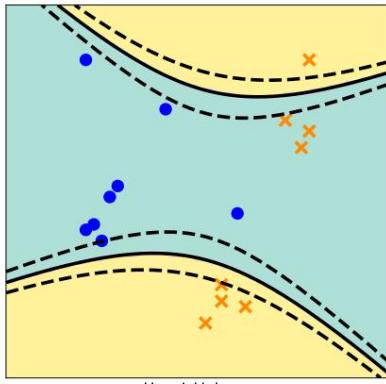
389



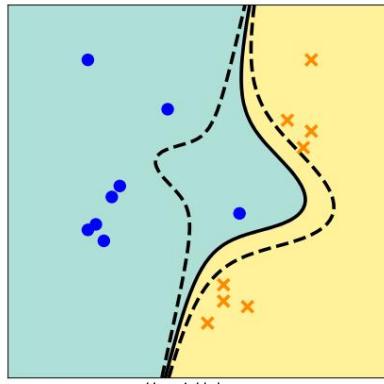
(a) 具有线性内核的 SVM



(b) 具有 RBF 内核的 SVM



(c) 具有多项式 (2 次) 内核的 SVM



(d) 具有多项式 (3 次) 内核的 SVM

图 12.4 具有不同内核的 SVM。注意

虽然决策边界是非线性的，但要解决的潜在问题是线性分离超平面（尽管具有非线性核）。

有一个与每个内核 k 相关联的唯一再现内核希尔伯特空间 (Aronszajn, 1950; Berlinet 和 Thomas-Agnan, 2004)。在这个唯一关联中， $\phi(x) = k(\cdot, x)$ 被称为规范特征图。规范特征从内积到核函数 (12.52) 的泛化被称为核技巧 (Scholkopf 和 Smola, 2002; Shawe-Taylor 和 Cristianini, 2004)，因为它隐藏了显式非线性特征图。

矩阵 $K \in \mathbb{R}^{N \times N}$, 由内积或应用

$k(\cdot, \cdot)$ 到数据集的阳离子，称为 Gram 矩阵，通常只是 Gram 矩阵，称为核矩阵。内核必须是对称且正的内核矩阵半定函数，以便每个内核矩阵 K 都是对称的且半正定的（第 3.2.3 节）：

$$\forall z \in \mathbb{R}^N : z^T K z \geq 0. \quad (12.53)$$

多元实值数据 $x_i \in \mathcal{X}$ 的一些流行示例

RD 是多项式核、高斯径向基函数核和有理二次核 (Scholkopf 和 Smola,

, 2002; 拉斯穆森

和威廉姆斯,2006 年)。图 12.4 说明了不同内核对示例数据集上分离超平面的影响。请注意,我们仍在求解超平面,即函数的假设类仍然是线性的。非线性表面归因于核函数。

评论。不幸的是,对于初出茅庐的机器学习者来说,“内核”这个词有多种含义。在本章中,“核”一词来自于再生核希尔伯特空间 (RKHS) 的概念 (Aron szajn,1950;Saitoh, 1988)。我们已经讨论了线性代数中核的概念 (第 2.7.3 节),其中核是零空间的另一个词。机器学习中“核”一词的第三个常见用法是核密度估计中的平滑核 (第 11.5 节)。由于显式表示 (x) 在数学上等同于核表示 $k(x_i, x_j)$,因此从业者通常会设计核函数,使其比显式特征图之间的内积计算更有效。例如,考虑多项式内核 (Scholkopf 和 Smola,2002 年),当输入维数很大时,显式扩展中的项数增长非常快 (即使对于低阶多项式也是如此)。内核函数只需要对每个输入维度进行一次乘法运算,这可以显著节省计算量。另一个例子是高斯径向基函数核 (Scholkopf 和 Smola,2002;Rasmussen 和 Williams, 2006),其中对应的特征空间是无限维的。在这种情况下,我们不能显式地表示特征空间,但仍然可以使用内核计算一对示例之间的相似性。

的选择

内核以及内核的参数通常使用嵌套交叉验证来选择 (第 8.6.1 节)。

内核技巧的另一个有用方面是原始数据不需要已经表示为多元实值数据。请注意,内积是在函数 (\cdot) 的输出上定义的,但不限制输入为实数。因此,函数 (\cdot) 和核函数 $k(\cdot, \cdot)$ 可以在任何对象上定义,例如集合、序列、字符串、图形和分布 (Ben-Hur et al., 2008;, 2008; Shi 等人,2009 年;Sriperumbudur 等人,2010 年;Vishwanathan Gartner 等人,2010 年)。

12.5 数值解

我们通过查看如何根据第 7 章中介绍的概念来表达本章中导出的问题来结束对 SVM 的讨论。我们考虑了两种不同的方法来寻找 SVM 的最优解。首先,我们考虑 SVM 8.2.2 的损失视图,并将其表示为无约束优化问题。然后我们将原始和对偶 SVM 的约束版本表示为标准形式 7.3.2 中的二次规划。

考虑 SVM (12.31) 的损失函数视图。这是一个凸无约束优化问题,但 hinge loss (12.28) 不是 diff

可推理的。因此，我们采用次梯度方法来解决它。

然而，铰链损失几乎在任何地方都是可微的，除了铰链 $t = 1$ 处的一个点。此时，梯度是一组介于 0 和 -1 之间的可能值。因此，铰链损失的次梯度 g 由下式给出

$$g(t) = \begin{cases} -1 & t < 1 \\ [-1, 0] & t = 1 \\ 0 & t > 1 \end{cases} \quad (12.54)$$

使用这个次梯度，我们可以应用第 7.1 节中介绍的优化方法。

原始 SVM 和对偶 SVM 都会导致凸二次规划问题（约束优化）。请注意，(12.26a) 中的原始 SVM 具有优化变量，其大小为输入示例的维度 D 。(12.41) 中的对偶 SVM 具有大小为 N 个示例的优化变量。

为了以二次规划的标准形式 (7.45) 表示原始 SVM，让我们假设我们使用点积 (3.5) 作为内积。我们重新排列原始 SVM (12.26a) 的方程式，回忆一下，优化变量都在右边，约束的不等式与标准形式匹配。这产生了优化

第 3.2 节我们使用短语点积来表示上的内积

$$\begin{array}{lll} \text{最小} & \frac{1}{2} \|w\|^2 + C \sum_{n=1}^N \xi_n & \text{否} \\ w, b, \xi & \text{受制于} & \text{欧氏向量} \\ & -y_n x_n^T w - y_n b - \xi_n & \text{空间。} \\ & -\xi_n & 0 \end{array} \quad (12.55)$$

我们得到软间隔 SVM 的以下矩阵形式：

$$\begin{array}{lll} \text{最小} & \frac{1}{2} \begin{matrix} w \\ b \\ \xi \end{matrix}^T \begin{matrix} w \\ b \\ \xi \end{matrix} + C \begin{matrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{matrix} \begin{matrix} w \\ b \\ \xi \end{matrix} & \text{w} \\ w, b, \xi & \text{受制于} & b \\ & -YX - y - IN & -1 \\ & 0N, D+1 & 0N, 1 \end{array} \quad (12.56)$$

在前面的优化问题中，最小化是在参数 $[w, b, \xi]$ 上进行的，我们使用符号：
 I_m 表示大小为 $m \times m$ 的单位矩阵。
 $0_{m,n}$ 表示大小为零的矩阵 $m \times n$ ，和 $1_{m,n}$ 表示大小为 1×1 的矩阵。
 $Y = \text{diag}(y)$ 表示大小为 $m \times m$ 的矩阵， y 是标签的向量 $[y_1, \dots, y_N]^T$ 。

是一个 $N \times N$ 矩阵, 其中对角线的元素来自 y , $X \in \mathbb{R}^{N \times D}$ 是通过连接所有示例获得的矩阵。

我们可以类似地为 SVM (12.41) 的双重版本执行一组术语。为了以标准形式表达对偶 SVM, 我们首先必须表达内核矩阵 K , 使得每个条目都是 $K_{ij} = k(x_i, x_j)$ 。如果我们有一个明确的特征表示 x 那么我们定义 $K_{ij} = x_i^T x_j$ 。

为了方便表示, 我们引入了一个矩阵, 除了在对角线上 (我们存储标签的地方) 外, 其他地方都为零, 即 $Y = \text{diag}(y)$ 。
双 SVM 可以写成

$$\begin{array}{ll} \text{分钟} & \frac{1}{2} \alpha^T Y K Y \alpha - 1 \\ \alpha & \text{是} \\ \text{受制于} & -y^T \alpha - 0 \\ & -I_N \alpha - C_1 N, 1 \\ & \text{在} \end{array} \quad (12.57)$$

评论。在 7.3.1 和 7.3.2 节中, 我们介绍了约束的标准形式是不等式约束。我们将对偶 SVM 的等式约束表示为两个不等式约束, 即

$$Ax = b \text{ 替换为 } Ax - b \text{ 和 } Ax + b. \quad (12.58)$$

凸优化方法的特定软件实现可以提供表达等式约束的能力。 ◇由于 SVM 有许多不同的可能视图, 因此有许多方法可以解决由此产生的优化问题。这里介绍的方法以标准凸优化形式

表达 SVM 问题, 在实践中并不经常使用。SVM 求解器的两个主要实现是 Chang 和 Lin (2011) (开源) 和 Joachims (1999)。由于 SVM 有一个明确且定义明确的优化问题, 因此可以应用许多基于数值优化技术的方法 (Nocedal 和 Wright, 2006 年) (Shawe-Taylor 和 Sun, 2011 年)。

12.6 延伸阅读

SVM 是研究二元分类的众多方法之一。

其他方法包括感知器、逻辑回归、Fisher 判别、最近邻、朴素贝叶斯和随机森林 (Bishop, 2006 年; Murphy, 2012 年)。在 Ben-Hur 等人中可以找到关于离散序列的支持向量机和内核的简短教程。 (2008)。SVM 的发展与 8.2 节中讨论的经验风险最小化密切相关。

因此, SVM 具有强大的理论属性 (Vapnik, 2000 年; Steinwart 和 Christmann, 2008 年)。关于内核方法的书 (Scholkopf 和 Smola, 2002 年) 包括支持向量机的许多细节和

如何优化它们。一本关于内核方法的更广泛的书 (Shawe Taylor 和 Cristianini,2004 年) 也包括许多针对不同机器学习问题的线性代数方法。

可以使用 Legendre–Fenchel 变换的思想获得对偶 SVM 的另一种推导 (第 7.3.3 节)。推导分别考虑 SVM (12.31) 的无约束公式的每一项,并计算它们的凸共轭 (Rifkin and Lippert, 2007)。对 SVM 的功能分析观点 (也是正则化方法观点) 感兴趣的读者可以参考 Wahba (1990) 的著作。内核的理论阐述 (Aronszajn,1950 年;Schwartz,1964 年; Saitoh,1988 年;Manton 和 Amblard,2015 年) 需要线性算子的基本基础 (Akhiezer 和 Glazman,1993 年)。内核的概念已推广到 Banach 空间 (Zhang et al., 2009) 和 Krein 空间 (Ong et al., 2004; Loosli et al., 2016)。

观察铰链损失具有三个等价表示,如 (12.28) 和 (12.29) 所示,以及 (12.33) 中的约束优化问题。在将 SVM 损失函数与其他损失函数进行比较时,经常使用公式 (12.28) (Steinwart, 2007)。

两段公式 (12.29) 便于计算子梯度,因为每一段都是线性的。第三个公式 (12.33),如第 12.5 节所示,支持使用凸二次规划 (第 7.3.2 节) 工具。

由于二元分类是机器学习中一项经过充分研究的任务,因此有时也会使用其他词,例如辨别、分离和决策。此外,二元分类器可以输出三个量。首先是线性函数本身的输出 (通常称为得分),它可以取任何实数值。此输出可用于对示例进行排名,二元分类可以被认为是在排名示例上选择一个阈值 (Shawe-Taylor 和 Cristianini,2004)。通常被认为是二元分类器输出的第二个量是在通过非线性函数将其值限制在有界范围内 (例如区间 [0, 1]) 后确定的输出。一个常见的非线性函数是 sigmoid 函数 (Bishop, 2006)。当非线性导致经过良好校准的概率时 (Gneiting 和 Raftery,2007 年;Reid 和 Williamson,2011 年),这称为类别概率估计。二元分类器的第三个输出是最终的二元决策 {+1, -1},这是最常被假定为分类器输出的一个。

SVM 是一种二元分类器,它本身并不适用于概率解释。有几种方法可以将线性函数的原始输出 (分数) 转换为校准类概率估计 ($P(Y=1|X=x)$),这些方法涉及额外的校准步骤 (Platt, 2000; Zadrozny 和 Elkan), 2001 年; Lin 等人, 2007 年)。

从训练的角度来看,有很多相关的概率方法。我们在第 12.2.5 节末尾提到有一个重新

损失函数和似然之间的关系（也比较第 8.2 和 8.3 节）。在训练期间对应于良好校准转换的最大似然方法称为逻辑回归，它来自一类称为广义线性模型的方法。从这个角度出发的逻辑回归的详细信息可以在 Agresti (2002 年, 第 5 章) 和 McCullagh 和 Nelder (1989 年, 第 4 章) 中找到。

自然地，可以通过使用贝叶斯逻辑回归估计后验分布来对分类器输出采取更贝叶斯的观点。贝叶斯观点还包括先验规范，其中包括设计选择，例如具有可能性的共轭（第 6.6.1 节）。此外，人们可以将潜在函数视为先验函数，这会导致高斯过程分类 (Rasmussen 和 Williams, 2006 年, 第 3 章)。

参考

Abel, Niels H. 1826. Démonstration de l'Impossibilité de la Résolution Algébrique des équations G Adhikari, érales qui Passent le Quatrième Degré. Grøndahl 和 Søn.

Ani 和 DeNero, John. 2018. 计算和推理思维: 数据科学的基础。Gitbooks。

Agarwal, Arvind 和 Daumé III, Hal. 2010. 共轭先验的几何视图。

机器学习, 81(1), 99–113。

Agresti, A. 2002. 分类数据分析。威利。

赤池, Hirotugu. 1974. 统计模型识别的新视角。IEEE 自动控制汇刊, 19(6), 716–723。

Akhiezer, Naum I. 和 Glazman, Izrail M. 1993. 希尔伯特空间中的线性算子理论。多佛出版社。

阿尔帕丁, 埃瑟姆。 2010. 机器学习简介。麻省理工学院出版社。

阿玛利, 顺一。 2016. 信息几何及其应用。施普林格。

Argyriou, Andreas 和 Dinuzzo, Francesco. 2014. 表示定理的统一观点。在: 国际机器学习会议论文集。

Aronszajn, 纳赫曼。 1950. 再生核理论。美国数学学会汇刊, 68, 337–404。

阿克斯勒, 谢尔顿。 2015. 线性代数做对了。施普林格。 ..

Bakir, Gökhan, Hofmann, Thomas, Schölkopf, Bernhard, Smola, Alexander J., Taskar, Ben 和 Vishwanathan, SVN (编)。 2007. 预测结构化数据。麻省理工学院出版社。

理发师, 大卫。 2012. 贝叶斯推理和机器学习。剑桥大学
按。

Barndorff-Nielsen, Ole. 2014. 信息和指数族: 在统计中
论。威利。

Bartholomew, David, Knott, Martin 和 Moustaki, Irini. 2011. 潜变量模型
和因素分析: 统一方法。威利。

Baydin, Atılım G., Pearlmutter, Barak A., Radul, Alexey A. 和 Siskind, Jeffrey M.
2018. 机器学习中的自动微分: 一项调查。机器学习研究杂志, 18, 1–43。

贝克、阿米尔和特布勒、马克。 2003. 用于凸优化的镜像下降法和非线性投影子梯度法。运筹学快报, 31(3), 167–175。

Belabbas, Mohamed-Ali 和 Wolfe, Patrick J. 2009. 机器学习中的光谱方法和超大型数据集的新策略。美国国家科学院院
刊, 0810600105。

Belkin, Mikhail 和 Niyogi, Partha. 2003. 用于降维和数据表示的拉普拉斯特征图。神经计算, 15(6), 1373–1396。

Ben-Hur, Asa, Ong, Cheng Soon, Sonnenburg, Sören, Schölkopf, Bernhard 和 Ratsch, Gunnar. 2008. 计算生物学
的支持向量机和内核。
PLoS 计算生物学, 4(10), e1000173。

- Bennett, Kristin P. 和 Bredensteiner, Erin J. 2000a。 SVM 分类器中的对偶性和几何。在 :国际机器学习会议论文集。
- Bennett, Kristin P. 和 Bredensteiner, Erin J. 2000b。学习中的几何。第 132-145 页 :工作中的几何。美国数学协会。
- Berlinet,Alain 和 Thomas-Agnan,Christine。 2004.在概率和统计中再现核希尔伯特空间。施普林格。
- Bertsekas, Dimitri P. 1999.非线性规划。雅典娜科学。
- Bertsekas, Dimitri P. 2009.凸优化理论。雅典娜科学。
- Bickel, Peter J. 和 Doksum, Kjell。 2006.数理统计、基本思想与选题。卷。 1. 学徒堂。
- Bickson,Danny,Dolev,Danny,Shental,Ori,Siegel,Paul H. 和 Wolf, Jack K. 2007。
通过置信度传播的线性检测。在 :关于通信、控制和计算的阿勒顿年度会议论文集。
- 比林斯利,帕特里克。 1995.概率与测量。威利。
- Bishop, Christopher M. 1995.模式识别的神经网络。克拉伦登
按。
- Bishop, Christopher M. 1999.贝叶斯 PCA。在 :神经信息专业进展
处理系统。
- Bishop, Christopher M. 2006.模式识别和机器学习。施普林格。
- Blei, David M.,Kucukelbir, Alp 和 McAuliffe, Jon D. 2017.变分推理:统计学家评论。美国统计协会杂志, 112(518),859-877。
- Blum,Arvind 和 Hardt,Moritz。 2015. The Ladder:机器学习竞赛的可靠排行榜。在 :国际机器学习会议。
- Bonnans, J. Fr ed eric, Gilbert, J. Charles, Lemar echal, Claude, and Sagastizabal, Claudia A. 2006.数值优化:理论和实践方面。施普林格。
- Borwein, Jonathan M. 和 Lewis, Adrian S. 2006.凸分析和非线性优化。第二版。加拿大数学学会。
- Bottou,L'eon。 1998. 在线算法和随机逼近。第 9-42 页 :在线学习和神经网络。剑桥大学出版社。
- Bottou,L'eon,Curtis,Frank E. 和 Nocedal,Jorge。 2018. 大规模机器学习的优化方法。 SIAM 评论, 60(2),223-311。
- Boucheron,Stephane,Lugosi,Gabor 和 Massart,Pascal。 2013.集中在平等:独立的非渐近理论。牛津
大学出版社。
- Boyd,Stephen 和 Vandenberghe, Lieven。 2004.凸优化。剑桥
大学出版社。
- Boyd,Stephen 和 Vandenberghe, Lieven。 2018.应用线藻导论
胸罩。剑桥大学出版社。
- Brochu,Eric,Cora,Vlad M. 和 de Freitas,Nando。 2009.昂贵成本函数的贝叶斯优化教程,以及在主动用
户建模和分层强化学习中的应用。技术。回复TR-2009-023。不列颠哥伦比亚大学计算机科学系。
- Brooks,Steve,Gelman,Andrew,Jones,Galin L. 和 Meng, Xiao-Li (编)。 2011.马尔可夫链蒙特卡
洛手册。查普曼和霍尔/CRC。
- Brown, Lawrence D. 1986.统计指数族的基础:在统计决策理论中的应用。数理统计研究所。
- Bryson, Arthur E. 1961.优化多阶段分配过程的梯度法。在 :哈佛大学数字计算机及其应用研讨会论文集。
- 布贝克,塞巴斯蒂安。 2015. 凸优化:算法和复杂性。机器学习的基础和趋势, 8(3-4),231-357。
- Bühlmann,Peter 和 Van De Geer,Sara。 2011. 高维数据统计。
施普林格。

- 伯吉斯,克里斯托弗。 2010. 降维:导览。基金会和机器学习趋势, 2(4),275–365。
- Carroll, J Douglas 和 Chang, Jih-Jie。 1970. 通过 “Eckart-Young”Decom 位置的 N 向概括分析多维尺度中的个体差异。心理测量学, 35(3),283-319。
- Casella, George 和 Berger, Roger L. 2002. 统计推断。达克斯伯里。
- Cinlar, Erhan. 2011. 概率与随机。施普林格。
- Chang, Chih-Chung 和 Lin, Chih-Jen。 2011. LIBSVM:支持向量机库。 ACM 智能系统和技术交易, 2,27:1–27:27。
- 奶酪人,彼得。 1985. 为概率辩护。在:国际人工智能联合会议论文集。
- Chollet,Francois 和 Allaire,JJ 2018. 深度学习与R. Manning 出版物。
- Codd, Edgar F. 1990. 数据库管理的关系模型。艾迪生卫斯理 朗文出版社。
- Cunningham, John P. 和 Ghahramani, Zoubin。 2015. 线性降维:调查、洞察和概括。机器学习研究杂志, 16,2859–2900。
- Datta, Biswa N. 2010. 数值线性代数和应用。暹。
- Davidson, Anthony C. 和 Hinkley, David V. 1997. Bootstrap方法及其应用。剑桥大学出版社。
- Dean,Jeffrey,Corrado,Greg S.,Monga,Rajat 和 Chen 等人。 2012. 大规模分布式深度网络。在:神经信息处理系统的进展。
- Deisenroth, Marc P. 和 Mohamed, Shakir。 2012. 高斯过程动力系统中的期望传播。第 2618–2626 页:神经信息处理系统的进展。
- Deisenroth, Marc P. 和 Ohlsson, Henrik。 2011. 高斯滤波和平滑的一般观点:解释当前和推导新算法。在:美国控制会议论文集。
- Deisenroth, Marc P.,Fox,Dieter 和 Rasmussen, Carl E. 2015. 机器人和控制中数据高效学习的高斯过程。 IEEE 模式分析和机器智能汇刊, 37(2), 408–423。
- Dempster, Arthur P.,Laird, Nan M. 和 Rubin, Donald B. 1977. 通过 EM 算法从不完整数据中得出最大似然。皇家统计学会杂志, 39(1), 1-38。
- Deng, Li, Seltzer, Michael L., Yu, Dong, Acero, Alex, Mohamed, Abdel-rahman, and Hinton, Geoffrey E. 2010. 使用深度自动编码器对语音频谱图进行二进制编码。在: Interspeech 的会议记录。
- 德罗耶,吕克。 1986. 非均匀随机变量生成。施普林格。
- Donoho, David L. 和 Grimes, Carrie。 2003. Hessian Eigenmaps:高维数据的局部线性嵌入技术。美国国家科学院院刊, 100(10),5591-5596。
- Dostal, Zdenek。 2009. 最优二次规划算法:变分不等式的应用。施普林格。
- 杜文,伊戈尔。 2017. 绑架。在:斯坦福哲学百科全书。元 斯坦福大学物理研究实验室。
- Downey, Allen B. 2014. Think Stats:探索性数据分析。第二版。奥赖利 媒体。
- 德雷福斯,斯图尔特。 1962. 变分问题的数值解。数学分析与应用杂志, 5(1),30-45。
- Drumm,Volker 和 Weil,Wolfgang。 2001. 线性代数和分析几何。 讲义,卡尔斯鲁厄大学 (TH)。
- Dudley, Richard M. 2002. 真实分析与概率。剑桥大学出版社。

- Eaton, Morris L. 2007. 多元统计 : 向量空间方法。数理统计研究所讲义。
- Eckart, Carl 和 Young, Gale. 1936. 一个矩阵被另一个矩阵逼近
较低的等级。心理测量学, 1(3), 211–218。
- Efron, Bradley 和 Hastie, Trevor. 2016. 计算机时代统计推断 : 算法、证据和数据科学。剑桥大学出版社。
- Efron, Bradley 和 Tibshirani, Robert J. 1993. Bootstrap简介。Chap man 和 Hall/CRC。
- 埃利奥特, 科纳尔。2009. 美丽的分化。在 : 函数式编程国际会议。
- Evgeniou, Theodoros, Pontil, Massimiliano 和 Poggio, Tomaso. 2000. 统计学习理论 : 入门。国际计算机视觉杂志, 38(1), 9–13。
- Fan, Rong-En, Chang, Kai-Wei, Hsieh, Cho-Jui, Wang, Xiang-Rui 和 Lin, Chih-Jen. 2008. LIBLINEAR : 大型线性分类库。机器学习研究杂志, 9, 1871–1874。
- Gal, Yarin, van der Wilk, Mark 和 Rasmussen, Carl E. 2014. 稀疏高斯过程回归和潜在变量模型中的分布式变分推理。在 : 神经信息处理系统的进展。
- 加特纳, 托马斯。2008. 结构化数据的内核。世界科学。
- Gavish, Matan 和 Donoho, David L. 2014. 奇异值的最佳硬阈值是 $4\sqrt{3}$ 。IEEE 信息论汇刊, 60(8), 5040–5053。
- Gelman, Andrew, Carlin, John B., Stern, Hal S. 和 Rubin, Donald B. 2004. 贝叶斯数据分析。查普曼和霍尔/CRC。
- Gentle, James E. 2004. 随机数生成和蒙特卡洛方法。
施普林格。
- Ghahramani, Zoubin. 2015. 概率机器学习和人工智能。
自然, 521, 452–459。
- Ghahramani, Zoubin 和 Roweis, Sam T. 1999. 使用 EM 算法学习非线性动力系统。在 : 神经信息处理系统的进展。
- 麻省理工学院出版社。
- Gilks, Walter R., Richardson, Sylvia, and Spiegelhalter, David J. 1996. 马尔可夫链
蒙特卡洛实践。查普曼和霍尔/CRC。
- Gneiting, Tilman 和 Raftery, Adrian E. 2007. 严格正确的评分规则、预测和估计。美国统计协会杂志, 102(477), 359–378。
- 哦, 加布里埃尔。2017. 为什么 Momentum 真的有效。蒸馏。
- Gohberg, Israel, Goldberg, Seymour 和 Krupnik, Nahum. 2012. 线性算子的踪迹和行列式。伯克豪泽。
- Golan, Jonathan S. 2007. 刚入门的研究生应该学习的线性代数
知道。施普林格。
- Golub, Gene H. 和 Van Loan, Charles F. 2012. 矩阵计算。JHU出版社。
- Goodfellow, Ian, Bengio, Yoshua 和 Courville, Aaron. 2016. 深度学习。麻省理工学院
按。
- Graepel, Thore, Candela, Joaquin Quinonero-Candela, Borchert, Thomas 和 Herbrich, Ralf. 2010. 微软 Bing 搜索
引擎中赞助搜索广告的网络规模贝叶斯点击率预测。在 : 国际机器学习会议论文集。
- Griewank, Andreas 和 Walther, Andrea. 2003. 自动微分导论。在 : 应用数学和力学论文集。
- Griewank, Andreas 和 Walther, Andrea. 2008. 评估衍生品、原则和
算法微分技术。暹。
- Grimmett, Geoffrey R. 和 Welsh, Dominic. 2014. 概率 : 简介。牛津
大学出版社。

- Grinstead, Charles M. 和 Snell, J. Laurie。 1997. 概率导论。美国人
数学学会。
- 黑客,伊恩。 2001. 概率和归纳逻辑。剑桥大学出版社。
- 霍尔,彼得。 1992. Bootstrap 和 Edgeworth 扩展。施普林格。
- Hallin,Marc,Paindaveine,Davy 和 Siman,Miroslav。 2010. 多元分位数和多输出回归分位数:从 ℓ_1 优化到半空间深度。统
计年鉴, 38,635–669。
- Hasselblatt,Boris 和 Katok,Anatole。 2003. 第一门动力学课程与最新发展全景。剑桥大学出版社。
- Hastie,Trevor,Tibshirani,Robert 和 Friedman,Jerome。 2001. Sta 元素
统计学习 数据挖掘、推理和预测。施普林格。
- Hausman,Karol,Springenberg,Jost T.,Wang,Ziyu,Heess,Nicolas 和 Riedmiller,Martin。 2018. 学习可转移机器人
技能的嵌入空间。在:国际学习代表会议论文集。
- 哈赞,埃拉德。 2015. 在线凸优化简介。优化的基础和趋势, 2(3–4), 157–325。
- Hensman,James,Fusi,Nicolo 和 Lawrence, Neil D. 2013. 大数据的高斯过程。在:人工智能不确定性会议论文集。
- Herbrich,Ralf,Minka,Tom 和 Graepel,Thore。 2007. TrueSkill(TM):贝叶斯技能评级系统。在:神经信息处理系统的进
展。
- Hiriart-Urruty,Jean-Baptiste 和 Lemaréchal,Claude。 2001. 凸分析基础。施普林格。
- Hoffman, Matthew D., Blei, David M., 和 Bach, Francis。 2010. 潜在狄利克雷分配的在线学习。神经信息处理系统的进
展。
- Hoffman, Matthew D., Blei, David M., Wang, Chong 和 Paisley, John。 2013. 随机变分推理。机器学习研究杂志,
14(1),1303–1347。
- Hofmann,Thomas,Scholkopf,Bernhard 和 Smola,Alexander J. 2008. Kernel Meth-
机器学习中的 ods。统计年鉴, 36(3),1171–1220。
- 霍格本,莱斯利。 2013. 线性代数手册。查普曼和霍尔/CRC。
- Horn, Roger A. 和 Johnson, Charles R. 2013. 矩阵分析。剑桥大学
按。
- 霍特林,哈罗德。 1933. 将复杂的统计变量分析成主成分。教育心理学杂志, 24, 417–441。
- Hyvärinen,Aapo,Oja,Erkki 和 Karhunen,Juha。 2001. 独立成分分析
分析。威利。
- Imbens, Guido W. 和 Rubin, Donald B. 2015. 统计、社会和生物医学科学的因果推理。剑桥大学出版社。
- Jacob, Jean 和 Protter, Philip。 2004. 概率要点。施普林格。
- Jaynes, Edwin T. 2003. 概率论:科学的逻辑。剑桥大学
按。
- Jefferys, William H. 和 Berger, James O. 1992. 奥卡姆剃刀和贝叶斯分析。美国科学家, 80, 64–72。
- 杰弗里斯,哈罗德。 1961. 概率论。牛津大学出版社。
- Jimenez Rezende,Danilo 和 Mohamed,Shakir。 2015. 规范化流的变分推理。在:国际机器学习会议论文集。
- Jimenez Rezende,Danilo,Mohamed,Shakir 和 Wierstra,Daan。 2014. 深度生成模型中的随机反向传播和近似推理。
在:国际机器学习会议论文集。
- 约阿希姆斯,托尔斯滕。 1999. 内核方法的进展 支持向量学习。麻省理工学院出版社。第一章 Making
Large-Scale SVM Learning Practical,第 169–184 页。
- Jordan, Michael I., Ghahramani, Zoubin, Jaakkola, Tommi S., and Saul, Lawrence K.
1999. 图形模型变分方法简介。机器学习, 37,183–233。

- Julier, Simon J. 和 Uhlmann, Jeffrey K. 1997。卡尔曼滤波器在非线性系统中的新扩展。在：AeroSense 航空航天/国防传感、模拟和控制研讨会论文集。
- Kaiser, Marcus 和 Hilgetag, Claus C. 2006。由于神经系统中的长距离投射，组件放置非最佳，但处理路径较短。 PLoS 计算生物学， 2(7),e95。
- 卡尔曼,丹。 1996. 一个特别有价值的分解:矩阵的 SVD。大学数学杂志， 27 (1) ,2-23。
- Kalman, Rudolf E. 1960。线性过滤和预测问题的新方法。 ASME 汇刊 - 基础工程杂志， 82 (D 系列) ,35-45。
- Kamthe, Sanket 和 Deisenroth, Marc P. 2018。具有概率模型预测控制的数据高效强化学习。在：人工智能和统计国际会议论文集。
- Katz, Victor J. 2004。数学史。皮尔逊/艾迪生 - 卫斯理。
- Kelley, Henry J. 1960。最佳飞行路径的梯度理论。艺术杂志， 30 (10) , 947-954。
- Kimeldorf, George S. 和 Wahba, Grace。 1970. 贝叶斯随机过程估计与样条平滑之间的对应关系。数理统计年鉴， 41(2),495-502。
- Kingma, Diederik P. 和 Welling, Max。 2014. 自动编码变分贝叶斯。在：国际学习代表会议论文集。
- Kittler, Josef 和 Foglein, Janos。 1984. 多光谱像素的上下文分类 数据。图像和视觉计算， 2(1),13-29。
- Kolda, Tamara G. 和 Bader, Brett W. 2009。张量分解和应用。 SIAM 评论， 51(3),455-500。
- 科勒,达芙妮和弗里德曼,尼尔。 2009.概率图形模型。麻省理工学院出版社。
- Kong, Linglong 和 Mizera, Ivan。 2012. 分位数层析成像:使用分位数 多元数据。中国统计学, 22, 1598-1610.
- 朗,塞尔。 1987.线性代数。施普林格。
- Lawrence, Neil D. 2005。使用高斯过程潜变量模型进行概率非线性主成分分析。机器学习研究杂志， 6 (十一月) ,1783-1816。
- Leemis, Lawrence M. 和 McQueston, Jacquelyn T. 2008。单变量分布 关系。美国统计学家， 62(1), 45-53。
- Lehmann, Erich L. 和 Romano, Joseph P. 2005。检验统计假设。 施普林格。
- Lehmann, Erich Leo 和 Casella, George。 1998.点估计理论。施普林格。
- Liesen, Jorg 和 Mehrmann, Volker。 2015. 线性代数。施普林格。
- Lin, Hsuan-Tien, Lin, Chih-Jen 和 Weng, Ruby C. 2007。关于支持向量机的 Platt 概率输出的注释。机器 学习， 68, 267-276。
- 荣格,伦纳特。 1999.系统识别:用户理论。学徒堂。
- Loosli, Gaëlle, Canu, Stéphane 和 Ong, Cheng Soon。 2016. 在 \mathbb{K}^n 空间中学习 SVM。 IEEE 模式分析和机器智能汇刊， 38(6), 1204-1216。
- Luenberger, David G. 1969。通过向量空间方法优化。威利。
- MacKay, David JC 1992。贝叶斯插值法。神经计算， 4, 415-447。
- MacKay, David JC 1998。高斯过程简介。第 133-165 页 :Bishop, CM (ed), 神经网络和机器学习。施普林格。
- MacKay, David JC 2003。信息论、推理和学习算法。 剑桥大学出版社。
- Magnus, Jan R. 和 Neudecker, Heinz。 2007.矩阵微分与 Appli 统计学和计量经济学中的阳离子。威利。

- Manton, Jonathan H. 和 Amblard, Pierre-Olivier。 2015. 重现内核希尔伯特空间入门。信号处理的基础和趋势, 8(1-2), 1-126。
- 马尔可夫斯基,伊万。 2011.低秩近似:算法、实现、应用。施普林格。
- Maybeck, Peter S. 1979.随机模型、估计和控制。学术出版社。
- McCullagh,Peter 和 Nelder,John A. 1989。广义线性模型。 CRC出版社。
- McEliece,Robert J.,MacKay,David JC 和 Cheng, Jung-Fu。 1998. Turbo 解码作为 Pearl 的“置信度传播”算法的实例。 IEEE 通讯选定领域期刊, 16(2), 140-152。
- Mika,Sebastian,Ratsch,Gunnar,Weston,Jason,Sch olkopf,Bernhard 和 Muller," Klaus-Robert. 1999. Fisher 核判别分析。第 41-48 页 :信号处理神经网络研讨会论文集。
- Minka, Thomas P. 2001a.近似贝叶斯推理的一系列算法。
博士论文,麻省理工学院。
- 明卡,汤姆。 2001b. PCA 维数的自动选择。在 :神经信息处理系统的进展。
- 米切尔,汤姆。 1997.机器学习。麦格劳-希尔。
- Mnih,Volodymyr,Kavukcuoglu,Koray 和 Silver,David 等。 2015. 通过深度强化学习进行人类水平的控制。自然, 518,529-533。
- Moonen, Marc 和 De Moor, Bart。 1995. SVD 和信号处理,III:算法、架构和应用。爱思唯尔。
- Moustaki,Irini,Knott,Martin 和 Bartholomew,David J. 2015.潜在变量模型
鹅岭。美国癌症协会。第 1-10 页。
- Muller, Andreas C. 和 Guido, Sarah。 2016. Python 机器学习简介 :数据科学家指南。奥莱利
出版社。
- Murphy, Kevin P. 2012.机器学习 :概率视角。麻省理工学院出版社。
- Neal, Radford M. 1996.神经网络的贝叶斯学习。博士论文,出发
多伦多大学计算机科学专业。
- Neal, Radford M. 和 Hinton, Geoffrey E. 1999.证明增量、稀疏和其他变体合理的 EM 算法的观点。第
355-368 页 :学习图形模型。麻省理工学院出版社。
- 尼尔森,罗杰。 2006. Copulas 简介。施普林格。
- 内斯特罗夫,尤里。 2018.凸优化讲座。施普林格。
- 纽迈尔,阿诺德。 1998. 解决病态和奇异线性系统 :正则化教程。 SIAM 评论, 40,636-666。
- Nocedal,Jorge 和 Wright,Stephen J. 2006.数值优化。施普林格。
- Nowozin,Sebastian,Gehler,Peter V.,Jancsary,Jeremy 和 Lampert,Christoph H. (编)。 2014.
高级结构化预测。麻省理工学院出版社。
- 奥哈根,安东尼。 1991. Bayes-Hermite 正交。统计规划杂志
和推理, 29,245-260。
- Ong, Cheng Soon, Mary, Xavier, Canu, St'ephane, and Smola, Alexander J. 2004. Learning with
Non-Positive Kernels,在 :国际机器学习会议论文集。
- Ormoneit,Dirk,Sidenbladh,Hedvig,Black,Michael J. 和 Hastie,Trevor。 2001.
学习和跟踪循环人体运动。在 :神经信息处理系统的进展。
- Page,Lawrence,Brin,Sergey,Motwani,Rajeev 和 Winograd,Terry。 1999. PageRank 引文排名 :为网
络带来秩序。技术。回复斯坦福信息实验室。
- 帕奎特,乌尔里希。 2008.潜在变量模型的贝叶斯推理。博士论文,大学
剑桥大学。
- 帕岑,伊曼纽尔。 1962. 关于概率密度函数和模式的估计。
数理统计年鉴, 33(3),1065-1076。

- 珍珠,朱迪亚。 1988.智能系统中的概率推理:似是而非的网络
推理。摩根考夫曼。
- 珍珠,朱迪亚。 2009.因果关系:模型、推理和推理。第二版。剑桥
大学出版社。
- 皮尔逊,卡尔。 1895. 对进化数学理论的贡献。二。均质材料中的偏斜变化。英国皇家学会哲学汇刊 A:数学、
物理和工程科学, 186,343–414。
- 皮尔逊,卡尔。 1901. 关于最适合空间点系统的线和平面。
哲学杂志, 2(11), 559–572。
- Peters, Jonas, Janzing, Dominik 和 Scholkopf, Bernhard。 2017. 因果推理的要素:基础
和学习算法。麻省理工学院出版社。
- Petersen, Kaare B. 和 Pedersen, Michael S. 2012.矩阵食谱。技术。回复
丹麦技术大学。
- Platt, John C. 2000.支持向量机的概率输出和与正则化似然法的比较。在:大利润分类器的进展。
- 波拉德,大卫。 2002.测量理论概率的用户指南。剑桥
大学出版社。
- Polyak, Roman A. 2016.现代优化中的勒让德变换。第 437–507 页:Goldengorin, B. (编) ,优化及其在
控制和数据科学中的应用。施普林格。
- Press, William H., Teukolsky, Saul A., Vetterling, William T. 和 Flannery, Brian P.
2007.数值食谱:科学计算的艺术。剑桥大学出版社。
- Proschan, Michael A. 和 Presnell, Brett。 1998. 期待康迪的意外
期望值。美国统计学家, 52(3), 248–252。
- Raschka, Sebastian 和 Mirjalili, Vahid。 2017. Python 机器学习:使用 Python、scikit-learn 和
TensorFlow 进行机器学习和深度学习。 Packt Publishing。
- Rasmussen, Carl E. 和 Ghahramani, Zoubin。 2001. 奥卡姆剃刀。在:神经信息处理系统的进展。
- Rasmussen, Carl E. 和 Ghahramani, Zoubin。 2003. 贝叶斯蒙特卡洛。在:广告
神经信息处理系统的进展。
- Rasmussen, Carl E. 和 Williams, Christopher KI 2006.机器学习的高斯过程。麻省理工学院出版社。
- Reid, Mark 和 Williamson, Robert C. 2011.二元实验的信息、分歧和风险。机器学习研究杂志, 12,731–
817。
- Rifkin, Ryan M. 和 Lippert, Ross A. 2007.价值正则化和 Fenchel 对偶性。
机器学习研究杂志, 8, 441–479。
- Rockafellar, Ralph T. 1970.凸分析。普林斯顿大学出版社。
- 罗杰斯、西蒙和吉罗拉米、马克。 2016.机器学习第一门课程。 Chap man 和 Hall/CRC。
- Rosenbaum, Paul R. 2017.观察与实验:因果推理简介。哈佛大学出版社。
- 罗森布拉特,穆雷。 1956. 关于密度函数的一些非参数估计的评论。数理统计年鉴, 27(3),832–837。
- Roweis, Sam T. 1998. PCA 和 SPCA 的 EM 算法。第 626–632 页:神经信息处理系统的进展。
- Roweis, Sam T. 和 Ghahramani, Zoubin。 1999. 线性高斯的统一回顾
楷模。神经计算, 11(2),305–345。
- Roy, Anindya 和 Banerjee, Sudipto。 2014.统计的线性代数和矩阵分析。查普曼和霍尔/CRC。
- Rubinstein, Reuven Y. 和 Kroese, Dirk P. 2016.模拟和蒙特卡洛
方法。威利。

- 鲁菲尼,保罗。 1799. Teoria Generale delle Equazioni,在 cui si Dimostra Impossibile la Soluzione Algebraica delle Equazioni Generali di Grado Superiore al Quarto 中。 Stamperia di S. Tommaso d' Aquino。
- Rumelhart, David E., Hinton, Geoffrey E. 和 Williams, Ronald J. 1986。通过反向传播误差学习表示。自然, 323 (6088), 533-536。
- Sæmundsson, Steindor, Hofmann, Katja 和 Deisenroth, Marc P. 2018。元强化学习与潜在变量高斯过程。在:人工智能不确定性会议论文集。
- 斋藤,三郎。 1988. 再生核理论及其应用。朗文科技。
- Sarkk "贝叶斯过滤和平滑。剑桥大学出版社。一个,西莫。 2013.
- Scholkopf, Bernhard 和 Smola, Alexander J. 2002。 Learning with Kernels 支持向量机、正则化、优化等。麻省理工学院出版社。
- Scholkopf, Bernhard, Smola, Alexander J. 和 Muller, Klaus-Robert。 1997. 内核主成分分析。在:国际人工神经网络会议论文集。
- Scholkopf, Bernhard, Smola, Alexander J. 和 Muller, Klaus-Robert。 1998. 作为内核特征值问题的非线性分量分析。神经计算, 10(5), 1299-1319。
- Scholkopf, Bernhard, Herbrich, Ralf 和 Smola, Alexander J. 2001。广义表示定理。在:国际计算学习理论会议论文集。
- 施瓦茨,洛朗。 1964. Sous Espaces Hilbertiens des Espaces Vectoriels Topologiques et Noyaux Associés. Journal d'Analyse Mathématique, 13, 115-256.
- Schwarz, Gideon E. 1978。估计模型的维度。统计年鉴, 6(2), 461-464。
- Shahriari, Bobak, Swersky, Kevin, Wang, Ziyu, Adams, Ryan P. 和 De Freitas, Nando。 2016. 让人类脱离循环:贝叶斯优化回顾。IEEE 会刊, 104(1), 148-175。
- Shalev-Shwartz, Shai 和 Ben-David, Shai。 2014. 了解机器学习:从理论到算法。剑桥大学出版社。
- Shawe-Taylor, John 和 Cristianini, Nello。 2004. 模式分析的内核方法。剑桥大学出版社。
- Shawe-Taylor, John 和 Sun, Shiliang。 2011. 优化方法回顾。在支持向量机中。神经计算, 74(17), 3609-3618。
- Shental, Ori, Siegel, Paul H., Wolf, Jack K., Bickson, Danny 和 Dolev, Danny。 2008. 线性方程组的高斯置信传播求解器。第 1863-1867 页:国际信息论研讨会论文集。
- Shewchuk, Jonathan R. 1994. 没有痛苦痛苦的共轭梯度法简介。
- Shi, Jianbo, 和 Malik, Jitendra。 2000. 归一化切割和图像分割。IEEE 模式分析和机器智能汇刊, 22(8), 888-905。
- Shi, Qinfeng, Petterson, James, Dror, Gideon, Langford, John, Smola, Alexander J. 和 Vishwanathan, SVN 2009。结构化数据的哈希内核。机器学习研究杂志, 2615-2637。
- Shiryayev, Albert N. 1984。概率。施普林格。
- Shor, Naum Z. 1985。不可微分函数的最小化方法。施普林格。
- Shotton, Jamie, Winn, John, Rother, Carsten 和 Criminisi, Antonio。 2006. Texton Boost:用于多类对象识别和分割的联合外观、形状和上下文建模。在:欧洲计算机视觉会议论文集。
- Smith, Adrian FM 和 Spiegelhalter, David。 1980. 线性模型的贝叶斯因素和选择标准。英国皇家统计学会杂志B, 42(2), 213-220。

Snoek,Jasper,Larochelle,Hugo 和 Adams,Ryan P. 2012。机器学习算法的实用贝叶斯优化。在:神经信息处理系统的进展。

斯皮尔曼,查尔斯。 1904. “一般情报”,客观确定和测量
确定。美国心理学杂志, 15(2), 201–292。

Sriperumbudur,Bharath K.,Gretton,Arthur,Fukumizu,Kenji,Scholkopf,Bernhard 和 Lanckriet,Gert RG 2010。希尔伯特空间嵌入和概率度量指标。机器学习研究杂志, 11,1517–1561。

斯坦瓦特,英戈。 2007. 如何比较不同的损失函数及其风险。
建设性近似, 26,225–287。

Steinwart,Ingo 和 Christmann,Andreas。 2008.支持向量机。施普林格。

Stoer, Josef 和 Burlirsch, Roland。 2002.数值分析导论。施普林格。

斯特朗,吉尔伯特。 1993. 线性代数基本定理。美国数学月刊, 100(9),848-855。

斯特朗,吉尔伯特。 2003.线性代数导论。韦尔斯利剑桥出版社。

流浪,乔纳森。 2016.好奇的记者数据指南。数字化中心
哥伦比亚大学新闻研究生院的新闻学。

斯特罗加茨,史蒂文。 2014. 为困惑和创伤的人写数学。
美国数学学会通告, 61(3), 286–291。

Sucar, Luis E. 和 Gillies, Duncan F. 1994.高级概率推理
想象。图像和视觉计算, 12(1),42–60。

Szeliski,Richard,Zabih,Ramin 和 Scharstein,Daniel 等人。 2008. 基于平滑先验的马尔可夫随机场能量最
小化方法的比较研究。 IEEE 模式分析和机器智能汇刊, 30(6),1068–1080。

Tandra, Haryono. 2014. 变量定理的变化与勒贝格积分微积分基本定理的关系。数学教学, 17(2), 76–83。

Tenenbaum, Joshua B.,De Silva, Vin 和 Langford, John C. 2000。非线性降维的全球几何框架。科学,
290(5500),2319-2323。

蒂布希拉尼,罗伯特。 1996. 通过套索进行回归选择和收缩。杂志
英国皇家统计学会B, 58(1),267–288。

Tipping, Michael E. 和 Bishop, Christopher M. 1999.概率主成分分析。皇家统计学会杂志: B 系列, 61(3),
611–622。

Titsias, Michalis K. 和 Lawrence, Neil D. 2010.贝叶斯高斯过程潜变量模型。在:人工智能和统计国际会议论文
集。

图森特,马克。 2012.关于梯度下降的一些笔记。 <https://ipvs.informatik.uni-stuttgart.de/mlr/marc/notes/gradientDescent.pdf>。

Trefethen, Lloyd N. 和 Bau III, David。 1997.数值线性代数。暹。

Tucker, Ledyard R. 1966.关于三模因子分析的一些数学笔记。
心理测量学, 31(3),279-311。

Vapnik, Vladimir N. 1998.统计学习理论。威利。

Vapnik, Vladimir N. 1999.统计学习理论概述。 IEEE事务处理
神经网络, 10 (5) ,988-999。

Vapnik, Vladimir N. 2000.统计学习理论的本质。施普林格。

Vishwanathan, SVN, Schraudolph, Nicol N., Kondor, Risi, and Borgwardt, Karsten M. 2010. 图内核。机
器学习研究杂志, 11,1201–1242。

von Luxburg,Ulrike 和 Scholkopf,Bernhard。 2011. 统计学习理论:模型、概念和结果。第 651–706 页:DM
Gabbay,S. Hartmann,J. Woods (编) ,逻辑史手册,卷。 10.爱思唯尔。

- 瓦巴,格蕾丝。 1990.观测数据样条模型。工业和社会应用数学。
- Walpole, Ronald E., Myers, Raymond H., Myers, Sharon L. 和 Ye, Keying。 2011. 工程师和科学家的概率和统计。学徒堂。
- 沃瑟曼,拉里。 2004.所有统计数据。施普林格。
- 沃瑟曼,拉里。 2007.所有非参数统计。施普林格。
- 惠特尔,彼得。 2000.通过期望的概率。施普林格。
- 威克姆,哈德利。 2014. 整理数据。统计软件杂志, 59,1-23。
- Williams, Christopher KI 1997.《无限网络计算》。在:神经信息处理系统的进展。
- Yu, Yaoliang, Cheng, Hao, Schuurmans, Dale 和 Szepesvari, Csaba。 2013. 表征表示定理。在:国际机器学习会议论文集。
- Zadrozny, Bianca 和 Elkan, Charles。 2001. 从决策树和朴素贝叶斯分类器中获得校准概率估计。在: 国际机器学习会议论文集。
- Zhang, Haizhang, Xu, Yuesheng 和 Zhang, Jun. 2009. 为机器学习再现内核 Banach 空间。机器学习研究杂志, 10, 2741–2775。
- Zia, Royce KP, Redish, Edward F. 和 McKay, Susan R. 2009. 了解勒让德变换。美国物理学杂志, 77 (614) , 614-622。

