

V2.0/20221122

数据科学与工程数学基础

数据科学与工程 数学基础

(第1版)
黄定江 编著

数据科学与工程数学基础

华东师范大学
上海

内 容 简 介

本书介绍了数据科学、人工智能和机器学习领域所需的核心数学基础知识，涉及矩阵计算、概率和信息论基础、优化基础。内容按照从模式分析到数据分析再到数学基础的思路来组织，围绕数据分析系统的核心构成：表示、模型和学习形成数据线和数学线两条线。数据线按照数据分析的处理流程、通过大量翔实的案例作为导引，引出所需数学；数学线紧扣数据线，按照知识内容发生的内在自然逻辑顺序展开。两者相辅相成，构成从具体到抽象、从抽象到具体的闭环。本书在数据科学的定位类似于《离散数学》在计算机科学的定位，配有相当数量的习题，可作为数据科学与大数据技术、人工智能、计算机科学和软件工程等相关专业的本科生或研究生的数学基础课程教材或参考书，也可作为学术和工业界科技人员了解和应用数据科学与大数据技术数学基础的参考手册。

数据科学与工程数学基础

序言

数据科学与工程核心课程的系列教材终于要面试了，这是一件鼓舞人心的事。作为华东师范大学数据学院的发起者和见证人，核心课程和系列教材一直是我心心念念的事情。值此教材出版发行之际，我很高兴能被邀请写几句话，做个回顾，分享一些感悟，也展望一下未来。

借着大数据热的东风，依托何积丰院士在 2007 年倡导成立的华东师范大学海量计算研究所，2012 年 6 月在时任 SAP 公司 CTO 史维学博士（Dr. Vishal Sikka）的支持下，我们成立了华东师范大学云计算与大数据研究中心。2013 年 9 月，学校发起成立作为二级独立实体的数据科学与工程研究院，开始软件工程一级学科下自设的数据科学与工程二级学科的博士和硕士研究生培养。在进行研究生培养的探索过程中，我们深切感受到我们的本科生的培养需要反思和改革。因此，到了 2016 年 9 月，研究院改制成数据科学与工程学院，随后就招收了数据科学与工程专业的本科生，第一届本科生已于 2020 年毕业。这是我们学院和专业的简单历史。经过这么几年的实践和思考，我们越发坚信当年对“数据科学与工程”这一名称的选择，“数据学院”和“数据专业”已经得到越来越多的认可，学院的师生也逐渐接受“数据人”这一称呼。

这里我想分享以下几方面的感悟：为什么要办数据专业？怎么办数据专业？教材为什么很重要？对人才培养有什么贡献？

为什么要办数据专业？数据是新能源，这是大家耳熟能详的一句话。说到能源，我们首先想到的是石油，所以大家就习惯把数据比喻成石油。但是，在我们看来，“新能源”对应的英文说法应该是“New Power”。“Data is Power”，这是我们的基本信念，也是我们为什么要办数据学院的根本动机。数据是人类文明史上的第三个重要的 Power，蒸汽能（Steam Power）和电能（Electric Power）引发了第一次和第二次工业革命。如果说蒸汽能和电能造就了从西方开始的两百多年的工业文明，数据能（Data Power）将把人们带入数字文明时代。数据是数字经济发展的重要的生产要素，这个生产要素不同于土地、劳动力，也不同于资本、技术。如果要给数据找一个恰当的比拟，也许只有十九世纪末伟大的发明家尼古拉·特斯拉发明的交流电。数据是新时代的交流电，就像上个世纪，交流电给世界带来的深刻变化一样，因为人们对数据能（Data Power）认识的提高，我们将进入一个“未来已来，一切重构”的时代。数据学院就像一百多年前的电力学院或电气学院。

怎么办数据专业？我们数据学院脱胎于软件工程学院，在此以前还有计算机科学与工程学院，数据相关的研究和偏向管理和图书情报方向的信息系统学科和专业也密切相关，应用数学、概率统计更是数据分析和处理的理论基础，不可或缺。到底什么样的专业才算是数据专业？起初的时候，这对我们来说基本上可以说是一个“灵魂拷问”。为此，我们发起成立了国内十五所高校三十多位知名教授组成立“高校数据科学与工程专业建设协作组”。我们相信，有了先进的理念，再加上集体的力量，数据专业建设的探索之路就能走通。协作组已经召开了四次研讨会，

确定了称为 CST 的专业建设路线图, C 代表 Curriculum (培养计划), S 代表 Syllabus (课程大纲), T 代表 Textbook (教材建设)。在得知我们的工作后, ACM/IEEE 计算机工程学科规范主席 John Impagliazzo 教授邀请我们参与了 ACM/IEEE 数据科学学科规范的制定。协作组达成共识: 专业课程分为基础课、核心课、方向课三类, 核心课是体现专业区分度的一组课程。与数据专业 (DSE) 最相近的专业就是计算机科学与工程 (CSE) 与软件工程 (SE) 两个专业, 我们确定的第一批 DSE 区别于 CSE 和 SE 的 8 门核心课程是: 数据科学与工程导论, 数据科学与工程数学基础, 数据科学与工程算法基础, 应用统计与机器学习、当代人工智能, 云计算系统、分布式计算系统、当代数据管理系统。随后我们又确定两门课纳入这个系列, 分别是: 区块链导论——原理、技术与应用, 数据中台初阶教程。数据专业作为一个新专业, 三类课程的边界还不清晰, 我们重点关注核心课程上面, 核心课有遗漏的知识点可以纳入基础课或方向课。这样可以保证知识体系的完整性, 简单起步, 快速迭代。随着实践和认识的深入, 逐渐明晰三类课程的边界, 形成完善的培养计划。

教材为什么很重要? 建设好一个专业, 培养计划和课程体系固然很重要, 但落实在根本上是教材。一套好的教材是建成一个好的专业的前提, 放眼看去, 无论是国内还是国外, 无论是具体高校还是国家区域层面, 这都是不争的事实, 好的专业都有成体系的好的教材。当然, 现在的教材已经不仅仅指的是单纯的一本教科书, 还有深层次的内容, 比如说具体的教学内容和教学方式。我们都知道, 教材是知识的结晶, 是站到巨人肩膀上去的台阶。在自然科学领域, 确实如此, 一百年前我们民族的仁人志士呼唤“赛先生”, 在中华大地上科学的传播带来了翻天覆地的变化。在更广泛的领域, 教材也还是技术、工艺和文化的传承, 是产业发展的助推器。拿信息技术来举例, 技术的源头和产业的发祥地都在美国和欧洲, 像 IBM、Lucent、Oracle 等跨国企业在我国商业上取得的巨大成功无一不与他们重视教材开发密切相关。试想一下, 我们的学生在课堂上学的都是他们的研究和研发的东西, 等走上工作岗位, 自然会对熟悉的技术和系统有亲近感, 这应该是产业或产品生态最重要的一个环节。本世纪以来, 随着互联网的蓬勃发展, 人们已经深刻认识到, 互联网改变世界。在人类的文明史上, 没有任何一项科研成果像互联网这样深刻地改变人, 改变世界。互联网之所以能改变世界, 是因为它真正发挥了数据的威力。互联网实现了信息技术发展从“以计算为中心”到“以数据为中心”的路径转变。用“昔日王谢堂前燕, 飞入寻常百姓家”来形容很多我们以前甚至当前教材上的一些内容, 可能毫不为过。以互联网为代表的新型产业的发展, 极大地推动了技术的进步, 我们已经到了可以编写自己的教材, 形成自己的技术体系和科学理论体系的时候了。我们是现代科学的后来者, 已经习惯了从科学到技术再到应用的路径, 现在有了成功的应用, 企业也发展出了领先的技术, 学界可以在此基础上发展出技术体系和科学理论体系, 应用、技术和科学的联动才是真正的创新之路。

对人才培养有什么贡献? 在信息技术领域, 迄今为止我们更多的是参考或沿袭了西方发达国家的培养计划和教材体系, 在改革开放以来的四十年, 这种“拿来主义”的做法很有效, 培养了大量的人才, 推动了我国的社会经济发展。但总的来说, 我们的高校在这一领域更像是在培养“驾驶员”, 培养开车的人, 现在到了需要我们来培养自己的造车人的时候了。技术发展趋

势如此，国际形势也对我们提出了这样的要求。我们处在一个大变局的时代，世界充满不确定性，开放和创新是应对不确定性的不二之选。创新成为人才培养的第一性原理，更新观念，变革教育，卓越育人是我们华东师范大学新时期人才培养的基本理念。人才培养是大学的第一要务，科学研究、社会服务和文化传承是大学的另外三大职能，大学通过这三大职能的实现可以更好地服务于人才培养。这也是数据专业核心课程系列教材的建设的指导思想，我们计划久久为功的这一件事是我们践行这一理解的一个小小的行动。

最后，要特别表示感谢，感谢华东师范大学和高等教育出版社的支持和鼓励，感谢数据专业建设协助组的各位老师们的通力协作和辛勤劳动，也要感谢数据学院师生的信任和付出。心有所信，方能行远；因为相信，所以看见。希望作为探路者的艰辛能成为大家学术和职业生涯中的一笔重要财富。

“The best way to predict the future is to invent it” ——Alan Kay

周傲英

华东师范大学

2020年11月于上海

数据科学与工程数学基础

前言

本书主要介绍数据科学、机器学习和人工智能所依赖的数学基础，包括：线性代数、概率与信息论和优化理论。我们知道数据的表示需要向量，机器学习中函数模型的权重可以用矩阵来表示；数据中的不确定性或随机性描述通常由概率来刻画，大数定律为统计机器学习模型的成功提供了理论基础；而优化为最终训练出一套可靠的模型参数提供了强大的数值计算支撑。

尽管线性代数、概率与信息论和优化理论的很多内容研究已经持续了一个世纪以上，但是直到近二十多年来人们才发现它们已然成为数据科学建模求解的核心数学基础，比如，奇异值分解的广泛应用、最大似然和最大后验的成功运用、凸优化方法可靠和迅速的求解等等，使得这些理论和方法足以嵌入到基于计算机程序运行的数据分析和人工智能算法设计之中。

但是就像很多其它学科利用线性代数、概率统计和优化作为基础工具一样，现实世界的数据问题如何转换为一个线性代数计算或概率估计或优化求解问题是不容易的，特别是数据科学领域的问题与其它领域的不同之处在于它对这三部分知识的需求是如此的交错复杂和浑然一体，比如，矩阵既可以用于表示数据，但它也是函数模型变换的一部分；协方差矩阵巧妙的融合了概率和线性代数，把方差和矩阵捏合在一起，从而能用作主成分分析的建模对象；数据科学大部分优化问题是非凸的，判断它是不是凸的或者将某个问题表述为凸优化的形式是比较困难的，这可以部分地借助对称正半定矩阵的概念等来实现。

本书目的

因此，本书的主要目的是帮助读者快速理清和掌握数据科学、机器学习和人工智能领域所需的相关数学知识，即表示数据所需的向量和矩阵的概念与运算，以及数值线性代数的四大核心议题；构建数据概率模型所需的概率基础和相关的统计和信息论准则；判断、描述以及求解凸优化问题的方法和背景知识等等。全书包括四个部分，共 12 章内容。

第一部分：绪论。也即第 1 章，主要介绍数据科学与工程数学基础在数据科学与大数据技术专业中的定位、应用背景、服务学科领域和主要数学内容的构成以及相关的数学基础简史，使读者对本书有初步的了解。这一章，我们会对从图像感知到自然语言处理再到数据分析与机器学习做一个简要的概览，让读者能够从“应用驱动”的角度来了解数据科学所涉及的和所需的数据基础，为全书的数据案例和数学内容展开做好铺垫。

第二部分：数据的低维表示——矩阵分析。涵盖了从第 2 章到第 6 章的主要内容。

第 2 章主要按数据的向量和矩阵表示、数据的向量和矩阵空间、数据空间的关系以及数据空间上代数结构建立的过程来具体介绍数据科学与工程所涉及的向量和矩阵的计算所需的基本知识，包括向量和矩阵基本概念和运算、向量空间、线性映射和线性变换、矩阵的基本特征和矩阵的特征分解等。

第 3 章介绍了线性代数的几何：度量和投影，包括向量的范数和内积、矩阵的范数和内积、

矩阵的四个基本子空间、投影以及特殊的正交矩阵等。这些概念有助于我们从几何的角度来理解线性代数的基本概念以及在数据科学中的应用。如范数和内积将被用作定义数据的各种相似性度量, 以及防止数据模型过拟合的正则化手段; 投影既是一个几何量, 也是一个变换, 在数据科学的降维任务中具有本质的作用。

第 4 章介绍了五种常用的矩阵分解方法, 包括 LU (三角) 分解、QR (正交) 三角分解、谱 (特征) 分解、Cholosky 分解和奇异值分解等。线性代数包含很多有趣的矩阵, 如: 对角阵、三角矩阵、正交矩阵、对称矩阵、置换矩阵、投影矩阵和关联矩阵等等。在这些矩阵当中对称正 (半) 定矩阵是核心, 因为数据科学与机器学习中大部分矩阵都是非方阵, 而非方阵总是可以通过与其自身的转置相乘得到对称正 (半) 定矩阵。对称正 (半) 定矩阵有正 (非负) 的特征值, 并且有正交的特征向量, 它也可以表示成一些秩 1 矩阵的线性组合, 因此可以方便的用于做低秩近似计算。在机器学习中, 我们主要处理的是这些大规模的对称正定矩阵或复杂的非方阵矩阵, 需要借助矩阵分解的技术, 特别是奇异值分解, 把它表示为对角阵、三角阵和正交矩阵的乘积等等, 然后利用这些特殊且简单的矩阵实现复杂矩阵的特征值等矩阵基本特征的快速计算, 并用于数据压缩、数据降维、稀疏优化以及低秩矩阵恢复问题的求解等等, 这对帮助理解原本复杂的高维数据矩阵的结构和性质具有重要的作用。

第 5 章介绍了数值线性代数三大核心主题内容, 包括线性方程组的求解、最小二乘问题和特征值的求解。数据科学中的很多问题最终都归结为上述三类问题的求解, 因此这一章主要介绍线性方程组的类型和解的结构, 引入基于矩阵分解的线性方程组和最小二乘问题的求解方法, 并讨论解的敏感性, 这些内容将与后续优化问题求解、数据科学中的线性回归问题相联系。此外, 还介绍了大规模矩阵求解特征值的一些计算方法, 包括幂迭代法, 这已被广泛应用于数据科学中的搜索技术 pagerank 的矩阵特征值计算。

第 6 章主要介绍向量和矩阵微分。包括向量和矩阵函数, 以及数据科学和统计机器学习中常见的各种函数 (包括模型函数、损失函数和目标函数等)、深度神经网络中函数的构造 (包括模型函数和激活函数等), 梯度和高阶导数的定义和性质、向量值函数和矩阵函数的梯度和求解方法以及用迹微分法求梯度的方法, 并引入深度网络中的反向传播和自动微分求解方法。这一章介绍的函数模型是数据科学中两大类型的模型之一。这些内容将在优化方法和数据科学中的各种优化问题求解中反复使用。

第三部分: 数据的随机表示——概率和信息论。涵盖了从第 7 章至第 9 章的内容。

第 7 章回顾概率论的基本概念, 建立用随机变量和分布来描述数据中的不确定性的思想。包括概率论的基本概念、随机变量及其分布、随机变量的数字特征、概率不等式、大数定律和中心极限定理、随机过程初步等。其中, 概率不等式在机器学习的理论分析, 通常也称为计算学习理论, 如 PAC 可学习性以及算法的泛化界和收敛性分析等方面具有重要的应用。此外, 大数据定律将被推广用于统计学习理论中经验风险最小化准则的建立, 而随机过程则在深度强化学习中具有广泛的应用。

第 8 章介绍香农熵、信息熵、KL 散度和微分熵等信息论基本概念和性质, 并引入基于熵概

念的信息度量准则和数据科学建模原理。信息论与机器学习有着紧密的联系，学习某种意义上就是一个熵减的过程，学习的过程也就是使信息的不确定度下降的过程，因此这些内容可以用于创造和改进学习算法（主要是分类问题），甚至衍生出了一个新方向——信息理论学习。特别可用于数据科学中基于概率和熵的相似性度量，这与第3章中非概率的相似性度量形成对应。

第9章介绍概率模型。包括数据建模的概率思想、模型的参数估计和非参数估计、概率模型的图语言描述和统计决策理论。其中数据建模的概率思想将引出数据科学和机器学习中模型的概率表示和类型等；模型的参数估计和非参数估计重点介绍极大似然、极大后验、直方图估计、核密度估计和非参数估计等；概率模型的图语言描述将给出条件独立性、有向非循环图、无向图、团和势等，这为以后学习朴素贝叶斯、隐马尔科夫等概率图模型内容奠定基础；统计决策理论主要涉及模型参数估计的好坏判断，这与机器学习中建立模型的策略密切相关。这一章介绍的概率模型是数据科学中另一大类型的模型之一，与第6章中的非概率模型，也即函数模型形成对应。

第四部分：数据的数值优化——凸优化。也即第10章至第12章的内容。

第10章介绍优化的基础理论。包括优化问题的分类，凸集和凸函数的定义和判别方法以及保凸运算，引入凸优化问题的定义和标准形式，并介绍数据科学和机器学习中常见的典型优化问题。事实上，机器学习中通过经验风险最小化准则建立的很多问题都可以建模为凸优化问题。

第11章介绍拉格朗日对偶函数和拉格朗日对偶问题，把标准形式（可能是非凸）的优化问题转化为对偶问题进行求解；并引出优化问题的最优化条件；介绍数据科学中各种常见的优化问题的对偶性问题。数据科学和机器学习中的很多问题是非凸的，比如最大割问题，可以通过转换为对偶问题进行有效求解。

第12章介绍无约束优化问题的性质和求解方法，包括直线搜索、梯度下降、最速下降、随机梯度下降方法等零阶和一阶方法；约束优化问题的求解方法，包括可行函数法和罚函数法；凸优化问题求解的高阶算法，包括牛顿法、内点法和拟牛顿法等二阶方法以及深度学习中一些常见的优化技术。这些方法将用于数据科学与机器学习中各种优化问题的求解。

读者范围

本书主要面向“数据科学与大数据技术”、“人工智能”、“计算机科学”等专业的本科生或低年级研究生。对于在工作中需要用到数值线性代数、概率估计和数学优化，或者更一般地说，用到计算数学的科研人员、科学家以及工程师，本书也较为合适。这些人群包括直接从事数据分析、机器学习和人工智能算法的科技工作者，亦包括一些工作在其他科学和工程领域但是需要借助数据科学数学基础的科技工作者，这些领域包括计算科学、经济学、金融、统计学、数据挖掘等。在阅读本书之前，读者只需要掌握现代微积分的基础知识即可。如果读者对一些基本的线性代数和基本的概率论有一定的了解，应能较好地理解本书的所有论证和讨论。当然，我们希望即使没有学过线性代数和概率论的读者也能够理解本书所有的基本思想和内容要点。

使用本书作为教材

我们希望本书能够在不同的课程中作为基本教材或者是参考教材来发挥它的作用，这些课

程包括数据科学的数学基础、人工智能的数学基础、机器学习的数学基础和计算机科学的数学基础（偏应用）等。从 2018 年开始，我们即在华东师范大学数据学院的本科生和低年级研究生的同名课程中使用本书的初稿。我们的经验表明，用 3 个学分，也即 48 学时到 54 学时，可以粗略讲授本书的大部分内容。如果用一个 4 学分的课程时间，也即 64 学时到 72 学时，讲课进度就可以比较从容，也可以增加更多的例子，并且可以更加详尽地讨论有关理论。若能用 5 个学分的课程时间，就可以对奇异值分解、最小二乘问题、特征值的计算、线性规划和二次规划（对于以应用为目的的学生极为重要）这些基本内容进行较广泛的细致讨论，或者加强这些内容对应算法方面的介绍或对学生布置更多的习题训练。本书可以作为线性代数、概率统计、线性优化和非线性优化等基础的参考读物。此外，对于像数学系更关注理论的课程，本书可以作为辅助教材，它提供了一些简单的实际例子。

致谢

本书写作参考了 Gilbert Strange 教授的 *Linear Algebra and Its Application* 和 *Linear Algebra and Learning from data*, Larry Wasserman 教授的 *All of Statistics*, Thomas Cover 教授的 *Elements of Information Theory*, Giuseppe Calafiori 教授和 El Chaoui Laurent 教授的 *Optimization Models*, Stephen Boyd 教授和 Lieven Vandenberghe 教授的 *Convex Optimization* 等经典数学教材以及 Vladimir Vapnik 教授的 *Statistical learning theory*, Hastie Trevor 教授, Tibshirani Robert 教授和 Friedman Jerome 教授的 *The Elements of Statistical Learning*, Ian Goodfellow, Yoshua Bengio, Aaron Courville 的 *Deep Learning* 等经典的机器学习教材。

本书是在华东师范大学周傲英副校长和数据科学与工程学院的大力支持下历时两年多完成的，虽然两年的时间不算短，并且主要内容也在华东师范大学数据学院本科生和低年级研究生的同名课程中使用过并取得了不错的讲授和学习效果，但是作为“数据科学与大数据技术”这样一个崭新的硬专业提供一本适用的教材，这点时间显然是不够的，很多内容还没有得到很好的打磨以适应不同层次水平的学生或相关的科研人员。但我们还是希望能够快速出版以满足日益增长的专业需求和读者们对这一领域持续探索的热情。我们只能期待在使用的过程中不断获得反馈以便快速迭代，从而获得更广泛的使用普遍性。这正如数据科学、人工智能和计算机科学这一领域从业者的行事准则：上线、迭代更新、再迭代，…，直至打磨稳定。我们也计划采用这种方式，所以恳请读者们如果碰到任何书本有关的问题，能及时反馈给我们 djhuang@dase.ecnu.edu.cn，以便我们能够改进，我们将不吝感激。

本书的写作过程中得到了来自由华东师范大学、哈尔滨工业大学、中国人民大学、中山大学、东北大学、西北工业大学、河南大学以及桂林电子科技大学等 15 所高校组成的数据专业协作组以及华东师范大学出版社和高等教育出版社的专家们的反馈和建议，同时也获得了华东师范大学很多同事，我课题组的研究生们以及我课程上的学生们的反馈和建议。篇幅所限，我们无法一一表达我们的感谢，只能在此对大家一并表达诚挚的谢意。

最后要特别感谢我的课题组的研究生们，我的博士生郝珊峰、申弋斌、刘友超和硕士生唐贊喆、赖叶静、张洋、余若男、汤路民、杨康、周雪茗、杨礼孟、王明和李特等同学，他们花

费了很多时间来协助我一起修改、编辑书中的公式、表格和图片等，才使得本书能够快速面世。郝珊峰和唐贊喆也协助我一起制作了与本书配套的同名课程的 MOOC 视频（本课程在融优学堂和超星泛雅 <http://mooc1.chaoxing.com/course/208843967.html> 上线），感谢他们的努力付出。

限于作者的知识水平，书中难免有不妥和错误之处，恳请读者不吝批评和指正。

黄定江

2020 年 10 月

数据科学与工程数学基础

数学符号

下面简要介绍本书所使用的数学符号。如果你不熟悉数学符号所表示的数学概念，可以参考对应的章节。

数据集

\mathbb{X}	输入空间
\mathbb{Y}	输出空间
$\mathbf{x} \in \mathbb{X}$	输入，实例
$y \in \mathbb{Y}$	输出，标记
$T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$	训练数据集
N	样本容量
(\mathbf{x}_i, y_i)	第 i 个训练数据点
$\mathbf{x} = (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)})^T$	输入向量， n 维实数向量
$\mathbf{x}_i^{(j)}$	输入向量 \mathbf{x}_i 的第 j 分量

向量和矩阵

a	标量 (整数或实数)
\mathbf{a}	向量
A	矩阵
\mathbf{A}	张量
\mathbf{I}_n	n 行 n 列的单位矩阵
\mathbf{I}	维度蕴含于上下文的单位矩阵
$\mathbf{e}^{(i)}$	索引 i 处值为 1 其它值为 0 的标准基向量
$\text{diag}(\mathbf{a})$	对角方阵，其中对角元素由 \mathbf{a} 给定
a_i	向量 \mathbf{a} 的第 i 个元素，其中索引从 1 开始
a_{-i}	除了第 i 个元素, \mathbf{a} 的所有元素
$a_{i,j}$	矩阵 \mathbf{A} 的 i 行 j 列元素
$\mathbf{A}_{i,:}$	矩阵 \mathbf{A} 的第 i 行
$\mathbf{A}_{:,i}$	矩阵 \mathbf{A} 的第 i 列
a_i	随机向量 \mathbf{a} 的第 i 个元素
\mathbb{A}	集合
\mathbb{R}	实数集

\mathbb{C}	复数域集
$\mathbb{A} \setminus \mathbb{B}$	差集, 即其元素包含于 \mathbb{A} 但不包含于 \mathbb{B}
\mathbb{R}^n	n 维实向量空间
\mathbb{C}^n	n 维复向量空间
$\dim(\mathbb{V})$	空间 \mathbb{V} 的维数
\mathbf{A}^{-1}	矩阵的逆
\mathbf{A}^T	矩阵 \mathbf{A} 的转置
\mathbf{A}^\dagger	\mathbf{A} 的 Moore-Penrose 伪逆
$\mathbf{A} \odot \mathbf{B}$	\mathbf{A} 和 \mathbf{B} 的逐元素乘积 (Hadamard 乘积)
$ \mathbf{A} $	\mathbf{A} 的行列式
$\text{rank}(\mathbf{A})$	矩阵的秩
A_{ij}	元素 a_{ij} 的代数余子式
\mathbf{A}^*	\mathbf{A} 的伴随矩阵
$\text{Tr}(\mathbf{A})$	矩阵的迹
λ	矩阵的特征值
范数	
$\ \cdot\ $	范数
$\ \mathbf{x}\ _2$	向量的 l_2 范数
$\ \mathbf{x}\ _1$	向量的 l_1 范数
$\ \mathbf{x}\ _\infty$	向量的 l_∞ 范数
$\ \mathbf{X}\ _2$	矩阵 \mathbf{X} 的谱范数
$\text{sim}_{\cos}(\mathbf{x}, \mathbf{y})$	余弦相似度
$\text{vec}(\mathbf{A})$	矩阵的向量化
$\ \mathbf{A}\ _F$	矩阵的 F 范数
$\ \mathbf{A}\ _*$	矩阵的核范数
$\text{Col}(\mathbf{A})$	\mathbf{A} 的列空间
$\text{Row}(\mathbf{A})$	行空间
$\text{Null}(\mathbf{A})$	零空间
$\text{Null}(\mathbf{A}^T)$	左零空间
微分	
$\frac{dy}{dx}$	y 关于 x 的导数
$\frac{\partial y}{\partial x}$	y 关于 x 的偏导
$\nabla_{\mathbf{x}} y$	y 关于 \mathbf{x} 的梯度
$\nabla_{\mathbf{X}} y$	y 关于 \mathbf{X} 的矩阵导数

$\nabla_{\mathbf{X}} y$	y 关于 \mathbf{X} 求导后的张量
$\frac{\partial f}{\partial \mathbf{x}}$	f 对 $\mathbf{x} \in \mathbb{R}^m$ 的 Jacobian 矩阵
$\nabla_{\mathbf{x}}^2 f(\mathbf{x})$ 或 $\mathbf{H}_f(\mathbf{x})$	f 在点 \mathbf{x} 处的 Hessian 矩阵
概率基础	
Ω	样本空间
E	随机试验
A	事件
$P(A)$	事件 A 发生的概率
$P(B A)$	事件 A 发生的情况下, 事件 B 发生的概率
$F_X(x)$	累积分布函数 CDF
$f_X(x)$	概率密度函数
x^+	从右边趋向于 x
$E(X)$	随机变量 X 的期望
$D(X)$	随机变量 X 的方差
$\text{Cov}(X, Y)$	随机变量 X, Y 的协方差
$N(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	均值为 $\boldsymbol{\mu}$ 协方差为 $\boldsymbol{\Sigma}$, \mathbf{x} 的高斯分布
$\Phi_X(t)$	X 的矩母函数, t 为实数
$\mathbb{E}_{x \sim P}[f(x)]$	$f(x)$ 关于 $P(x)$ 的期望
$X_n \xrightarrow{P} X$	X_n 依概率收敛于 X
$X_n \rightsquigarrow X$	X_n 依分布收敛于 X
$X_n \xrightarrow{qm} X$	X_n 均方意义上收敛于 X
$\Phi(z)$	标准正态分布的累积分布函数
$R_{\text{exp}}(f)$	期望风险
$R_{\text{emp}}(f)$	经验风险
信息论基础	
$I(x_i)$	事件 x_i 的自信息
$I(x_i; y_i)$	事件 x_i 和事件 y_i 的互信息
$H(X)$	随机变量 X 的信息熵
$p(x_i)$	事件 x_i 的概率分布
$H(\mathbf{p})$	熵函数
$D_{KL}(P\ Q)$	P 和 Q 的 KL 散度
$H(P, Q)$	P 和 Q 交叉熵
$h(X)$	连续随机变量 X 的微分熵
概率模型和参数估计	

θ	待估参数
Θ	待估参数可能的取值
$L(\theta)$	样本在参数 θ 下的似然函数
$f(x; \theta)$	由 θ 参数化, 关于 x 的函数
$p(\mathcal{D} \theta)$	由 θ 参数化, 获得给定数据 \mathcal{D} 的概率
$l(\theta)$	对数似然函数
$\mathbf{1}_{condition}$	如果条件为真则为 1, 否则为 0
$K(u)$	参数为 u 的核函数
$\Gamma(x)$	伽马函数
$\text{Beta}(a, b)$	Beta 函数
$Pa(x_i)$	随机变量 x_i 的父节点
$\Phi_C(x_C)$	团的势函数, 变量 x_C 属于集合 C
$NLL(\theta)$	模型参数为 θ 的极小负 log 似然损失
\mathbf{w}	权重向量
	优化
aff C	集合 C 的仿射包
relint C	集合 C 的相对内部
conv C	集合 C 的凸包
int C	集合 C 的内部
cl C	集合 C 的闭包
bd C	集合 C 的边界: $\mathbf{bd}C = \mathbf{cl}C \setminus \mathbf{int}C$
I_C	集合 C 的示性函数
S_C	集合 C 的支撑函数
$x \preceq y$	向量 x 和 y 之间的分量不等式
S^n	对称的 $n \times n$ 矩阵
S_+^n, S_{++}^n	对称半正定、正定 $n \times n$ 矩阵
$\mathbb{R}_+, \mathbb{R}_{++}$	非负、正实数
epi f	函数 f 的上镜图
prob S	事件 S 的概率
dom f	函数 f 的定义域
$\lambda_{max}(X), \lambda_{min}(X)$	对称矩阵 X 的最大、最小特征值
$\text{dist}(A, B)$	集合 (或点) A 和 B 之间的距离
∇f	函数的导数
f^*	f 的共轭函数

目录

第一章 绪论	1
1.1 本教材产生的背景和定位	1
1.2 从图像感知到自然语言处理	5
1.2.1 猫、分类和神经网络	5
1.2.2 文本、词向量和朴素贝叶斯	10
1.3 从数据分析到数学基础	18
1.3.1 数据分析和机器学习概览	18
1.3.2 数据	22
1.3.3 模型	25
1.3.4 学习	29
1.3.5 机器学习的应用	35
1.4 数据分析和机器学习所需数学内容框架	37
1.4.1 数值线性代数简介	39
1.4.2 概率与信息论简介	39
1.4.3 最优化简介	39
1.5 数据科学与工程数学的历史	39
1.5.1 早期阶段: 线性代数的诞生	40
1.5.2 概率论的起源	40
1.5.3 优化作为理论工具	40
1.5.4 数值线性代数的出现	41
1.5.5 线性和二次规划的出现	41
1.5.6 凸规划的出现	41
1.5.7 现阶段	42
1.6 本教材的使用建议	42

第二章 向量和矩阵基础	47
2.1 向量与矩阵的概念与运算	48
2.1.1 向量与矩阵的基本概念: 数据表示的观点	48
2.1.2 向量的运算	53
2.1.3 矩阵的运算	53
2.1.4 线性方程组	58
2.2 向量空间	61
2.2.1 向量空间的基本概念: 数据处理空间的出发点	61
2.2.2 向量子空间	63
2.2.3 子空间的交、和、直和	65
2.2.4 线性无关性	66
2.2.5 生成集、基底与坐标	68
2.2.6 秩	71
2.2.7 仿射空间	73
2.3 线性映射与线性变换	74
2.3.1 线性映射: 线性模型的观点	75
2.3.2 线性映射的矩阵表示	78
2.3.3 线性变换	84
2.3.4 仿射映射	88
2.4 矩阵的基本特征	91
2.4.1 行列式	91
2.4.2 迹运算	96
2.4.3 对称矩阵与二次型	97
2.4.4 特征值与特征向量	102
2.5 阅读材料	109
第三章 度量与投影	115
3.1 内积与范数: 数据度量的观点	116
3.1.1 向量范数	117
3.1.2 内积与夹角	122
3.1.3 数据科学中常用的相似性度量	128
3.1.4 矩阵的内积与范数	133
3.1.5 范数在机器学习中的应用	139
3.2 正交与投影	140
3.2.1 矩阵的四个基本子空间	140

3.2.2 四个基本子空间的正交性	144
3.2.3 正交投影	147
3.3 正交基与 Gram-Schmidt 正交化	152
3.3.1 标准正交基	152
3.3.2 Gram-Schmidt 正交化	153
3.4 具有特殊结构和性质的矩阵	154
3.4.1 特殊的正交变换矩阵——旋转	155
3.4.2 反射矩阵	159
3.4.3 信号处理中常见的正交矩阵	162
3.5 阅读材料	171
第四章 矩阵分解	177
4.1 数学中常见的具有特殊结构的矩阵	178
4.2 数据科学中常见的矩阵	184
4.2.1 图的矩阵	184
4.2.2 低秩矩阵	194
4.3 LU 分解	196
4.3.1 LU 分解	196
4.3.2 选主元的 LU 分解	201
4.4 QR 分解	205
4.4.1 基于 Gram-Schmidt 正交化的 QR 分解	205
4.4.2 基于 Householder 变换的 QR 分解	208
4.4.3 基于 Givens 变换的 QR 分解	211
4.5 谱分解与 Cholesky 分解	214
4.5.1 谱分解	214
4.5.2 Cholesky 分解	219
4.6 奇异值分解	221
4.6.1 奇异值分解	222
4.6.2 基于奇异值分解的矩阵性质	230
4.6.3 奇异值分解与低秩表示	235
4.7 阅读材料	240
第五章 矩阵计算问题	245
5.1 线性方程组的直接解法	246
5.1.1 线性方程组问题	246

5.1.2 一般线性方程组解的理论	247
5.1.3 容易求解的线性方程组	250
5.1.4 基于矩阵分解的方阵系统的直接解法	255
5.1.5 非方阵系统的直接求解方法	260
5.1.6 敏度分析与其他方法	266
5.2 最小二乘问题	269
5.2.1 最小二乘问题与线性回归	269
5.2.2 最小二乘问题的求解方法	273
5.2.3 最小二乘问题的变体	277
5.2.4 最小二乘问题的解的敏感性	279
5.3 特征值计算	280
5.3.1 矩阵特征值分布范围的估计	280
5.3.2 幂法	283
5.3.3 反幂法	286
5.3.4 特征值计算的应用: Pagerank 网页排名	290
5.4 阅读材料	292
第六章 向量与矩阵微分	298
6.1 向量函数和矩阵函数	299
6.1.1 函数	299
6.1.2 算子	303
6.1.3 泛函	304
6.1.4 机器学习中的风险泛函	305
6.2 统计机器学习中的非概率型函数模型	307
6.2.1 线性模型中的函数	307
6.2.2 感知机模型中的函数	308
6.2.3 支持向量机	310
6.2.4 降维和主成分分析中函数	318
6.2.5 聚类中的函数	319
6.3 深度神经网络中的函数构造	321
6.3.1 深度神经网络模型函数的构造过程	322
6.3.2 激活函数	325
6.4 向量和矩阵函数的梯度	329
6.4.1 向量函数的梯度	330
6.4.2 矩阵函数的梯度	333

6.5 对矩阵微分	335
6.5.1 矩阵微分与偏导数的联系	336
6.5.2 关于逆矩阵的函数的微分	337
6.5.3 关于行列式函数的微分	337
6.6 迹函数的微分和迹微分法	339
6.7 向量值函数和矩阵值函数的梯度	341
6.7.1 向量值函数的梯度	341
6.7.2 矩阵值函数的梯度	342
6.7.3 向量值函数微分	342
6.8 链式法则	344
6.9 反向传播和自动微分	346
6.9.1 反向传播	346
6.9.2 自动微分	348
6.10 高阶微分和泰勒展开	352
6.10.1 Hessian 矩阵	352
6.10.2 线性化和多元泰勒级数	353
6.11 阅读材料	354
第七章 概率基础	358
7.1 概率论基本概念回顾: 数据不确定性描述的观点	358
7.1.1 概率论基本概念	358
7.1.2 概率论公理	360
7.1.3 独立事件和条件概率	361
7.1.4 贝叶斯公式	362
7.2 随机变量及其分布	363
7.2.1 常用的随机变量及其分布	365
7.2.2 多维随机变量及其分布函数	366
7.3 随机变量的数字特征	370
7.3.1 期望	370
7.3.2 方差	372
7.3.3 一些重要分布的期望和方差	373
7.3.4 协方差和相关系数	375
7.3.5 矩和协方差矩阵	377
7.3.6 条件期望	380
7.3.7 方差的应用: 过拟合与偏差-方差分解	381

7.4 概率不等式	384
7.5 大数定律与中心极限定理	389
7.5.1 引言	389
7.5.2 大数定律	390
7.5.3 中心极限定理	392
7.5.4 大数定律的推广及其在统计学习中的应用	394
7.6 随机过程简介	397
7.6.1 马尔可夫链	398
7.6.2 高斯过程	404
7.7 阅读材料	405
第八章 信息论基础	408
8.1 熵、相对熵和互信息	409
8.1.1 自信息	410
8.1.2 熵及其性质	411
8.1.3 联合熵和条件熵	415
8.1.4 互信息和相对熵	417
8.1.5 熵、相对熵和互信息的链式法则	421
8.1.6 信息不等式	421
8.2 连续分布的微分熵和最大熵	423
8.2.1 连续信源的微分熵	423
8.2.2 连续信源的最大熵	426
8.3 信息论在数据科学中的应用	426
8.3.1 基于信息量的度量	426
8.3.2 其他概率相关的度量	428
8.4 阅读材料	431
第九章 概率模型	434
9.1 从概率到统计	435
9.1.1 统计的基本概念	435
9.1.2 模型、统计推断和学习	438
9.2 概率密度函数的估计	444
9.2.1 概率密度估计引入	444
9.2.2 基于频率观点的参数估计方法	446
9.2.3 贝叶斯推断	452

9.2.4	统计决策与贝叶斯估计	459
9.2.5	非参数估计	473
9.3	概率模型与图表示	485
9.3.1	概率模型的有向图表示	485
9.3.2	概率模型的无向图表示	491
9.4	机器学习中的概率模型	498
9.4.1	机器学习的概率思路	498
9.4.2	机器学习中的概率模型	499
9.4.3	深度学习中的概率模型	507
9.4.4	强化学习中的概率模型	515
9.5	阅读材料	515
第十章 优化基础		523
10.1	优化简介	524
10.1.1	数据科学与机器学习中最优化问题的例子	525
10.1.2	其他常见的优化问题举例	527
10.1.3	优化问题的一般形式	530
10.1.4	优化问题的分类	533
10.2	凸集	536
10.2.1	凸集	536
10.2.2	重要的凸集例子	539
10.2.3	保持凸集的运算	543
10.2.4	分离与支撑超平面	548
10.3	凸函数	550
10.3.1	凸函数的定义和基本性质	551
10.3.2	凸函数举例	552
10.3.3	凸函数的性质	553
10.3.4	凸函数的判定条件	554
10.3.5	保凸运算	560
10.3.6	共轭函数	567
10.3.7	次梯度	570
10.4	凸优化	577
10.4.1	凸优化问题	577
10.4.2	典型凸优化及其在数据科学中应用示例	581
10.5	阅读材料	588

10.6 习题	590
10.7 参考文献	592
第十一章 最优性条件和对偶理论	598
11.1 无约束优化的最优性条件	598
11.2 Lagrange 对偶函数	602
11.2.1 Lagrange 函数与对偶函数	602
11.2.2 常见优化问题目标函数的对偶函数	604
11.2.3 Lagrange 对偶函数与共轭函数的联系	606
11.3 Lagrange 对偶问题	608
11.3.1 Lagrange 对偶问题	608
11.3.2 对偶性质	610
11.3.3 常见优化问题的对偶问题及强对偶性	612
11.3.4 强对偶性定理的证明	613
11.3.5 强弱对偶性的极大极小描述	617
11.4 最优性条件	618
11.4.1 互补松弛条件	618
11.4.2 KKT 最优性条件	618
11.4.3 通过解对偶问题求解原问题	620
11.5 数据科学中常见模型的对偶问题	622
11.5.1 线性可分支持向量机	622
11.5.2 线性支持向量机	625
11.6 阅读材料	626
11.7 习题	627
11.8 参考文献	628
第十二章 优化算法	631
12.1 无约束优化	632
12.1.1 线搜索	635
12.1.2 一阶方法	644
12.1.3 二阶方法	658
12.2 约束优化	672
12.2.1 可行方向法	673
12.2.2 外点法	678
12.2.3 内点法	685

12.3	复合优化算法	690
12.3.1	近似点梯度法	691
12.3.2	分块坐标下降法	697
12.3.3	交替方向乘子法	702
12.4	深度学习常用优化算法	705
12.4.1	随机梯度下降	706
12.4.2	动量梯度下降	708
12.4.3	自适应学习速率	710
12.4.4	应用实例：多层感知机	713
12.5	在线凸优化算法简介	715
12.5.1	在线凸优化模型	715
12.5.2	一阶方法	717
12.5.3	二阶方法	720
12.6	阅读材料	723
12.7	习题	724
12.8	参考文献	726

数据科学与工程数学基础

数据科学与工程数学基础

第七章 概率基础

在第二章我们引入了向量和矩阵来对数据进行确定性表示。然而，在数据科学中，我们所遇到数据问题常具有不确定性和随机性。一个本身就具有随机性的系统、具有随机性（如测量时的四舍五入）的观测行为以及舍弃了某些特征或假设空间的不完全建模都可能引入不确定性和随机性。

粗略的说，概率可以被看作对处理不确定性的研究。在不同的视角下，它可以被认为是一个事件发生的次数在总试验次数中的占比，或者是对一个事件的信任程度。正如第1章中所提到的，我们通常希望量化不确定性：数据中的不确定性、机器学习模型中的不确定性以及模型生成的预测中的不确定性。量化不确定性需要引入随机变量的概念，这是一个将随机试验结果映射到实数的函数。而指明试验结果发生概率的是与随机变量的取值相关联的一组数字，对应于每个可能的结果到实数的映射。这组数字表明了概率分布。

7.1 概率论基本概念回顾：数据不确定性描述的观点

概率是描述不确定性的数学语言。概率论是研究不确定现象统计规律的一门学科。在数据科学中，数据通过采样得到的，具有一定的不确定性，它的结果是通过观测得到的，也具有一定的不确定性。因此，自然地使用概率模型对真实数据统计规律进行建模模拟。

本节将回顾概率论的基本概念。

7.1.1 概率论基本概念

人类在自然界的生产实践中，观察到的现象大致分为确定性现象和不确定现象两类。在中学阶段的物理课程中，我们一般学习研究确定性现象，比如：太阳肯定会东方升起；标准气压下，水在 100°C 会沸腾。

在数据科学中，我们经常研究不确定现象（随机现象），不确定现象在个别试验中呈现不确定结果，大量试验后呈现统计规律。

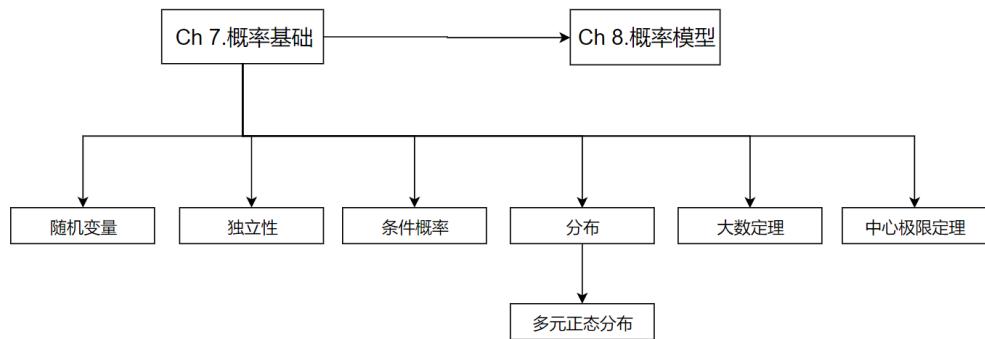


图 7.1: 本章导图

例 7.1.1. 给定一个查询 Q 和文档集 $D = \{d_1, d_2, d_3, \dots, d_n\}$, 从 D 随机抽取一篇文档 d_i , 查看 d_i 是否与查询相关。

例 7.1.2. 观察 A 股票在中午 12 点时的股价。

例 7.1.3. 从一部电影的众多影评中随机抽取一条影评, 观察该影评是正类影评还是负类影评。

以上三个试验都具有可重复、结果多样、结果不可预测这三个特点, 因此也被称为随机试验, 简称试验。

样本空间、样本点和事件

样本空间是随机试验所有可能结果的集合, 记作 Ω 。每一种试验结果称样本空间中的一个样本点。

例7.1.1中, 给定一个查询 Q 和文档集 D , 从文档集中的随机抽取文档 d_j 的试验的样本空间 $\Omega = \{\text{相关, 不相关}\}$ 。即 d_j 要么与查询相关, 要么不相关。“相关”是样本空间中的一个样本点, “不相关”也是一个样本点。

满足某些条件的样本点组成样本空间的子集称为随机事件, 简称事件。例7.1.1中从文档集 D 中抽取与查询 Q 相关的文档是一随机事件。例7.1.2中股票的价格大于 100 是一个随机事件。例7.1.3中影评是正类影评是一个随机事件。需要注意的是:

- 一个样本点也属于一个事件。
- 空集 \emptyset 是样本空间 Ω 的子集, 称为不可能事件。
- Ω 是它自己的子集, 称为必然事件。

事件的关系与运算

事件是样本点的集合，事件之间的关系与运算可以按照集合之间的关系与集运算来规定。

给定一个随机试验， Ω 是试验的样本空间，事件 A, B, C 是 Ω 的子集。下列给出事件之间的 7 种关系。

包含关系 如果 $A \subset B$ 或 $B \supset A$ ，称事件 B 包含事件 A 。它的含义是：若事件 A 发生，则事件 B 必然发生。

相等关系 如果 $A \subset B$ 且 $A \supset B$ ，称事件 B 与事件 A 相等。

事件和 $A \cup B = \{\omega : \omega \in A \text{ 或 } \omega \in B\}$ 称事件 A 与事件 B 的和事件。它的含义是：当且仅当事件 A 与事件 B 中至少一个发生时，事件 $A \cup B$ 发生。

事件积 事件 $A \cap B = \{\omega : \omega \in A \text{ 且 } \omega \in B\}$ 称事件 A 与事件 B 的积事件。它的含义是：当且仅当事件 A 与事件 B 中同时发生时，事件 $A \cap B$ 发生。为了表述方便，有时将 $A \cap B$ 直接简记为 AB 。

事件差 事件 $A - B = \{\omega : \omega \in A \text{ 且 } \omega \notin B\}$ 称事件 A 与事件 B 的差事件。它的含义是：当且仅当事件 A 发生且事件 B 不发生时，事件 $A - B$ 发生。

互斥关系 如果事件 $A \cap B = \emptyset$ 。称事件 A 与事件 B 互斥或不相容。它的含义是：在一次试验后，事件 A 与事件 B 不会同时发生。如果一组事件中任意两个事件互不相容，这组事件两两不相容。

逆事件 事件 $\Omega - A$ 称为事件 A 的逆事件，记作 $\bar{A} = \Omega - A$ 。它的含义是：当且仅当事件 A 不发生时，事件 \bar{A} 发生。于是 $\bar{A} \cap A = \emptyset$ ， $\bar{A} + A = \Omega$ 。由于 A 也是 \bar{A} 的对立事件，因此称事件 A 与 \bar{A} 互逆。

事件运算

与集合论中集合的运算一样，事件之间的运算满足下述定律：

- 交换律： $A \cup B = B \cup A$ $A \cap B = B \cap A$
- 结合律： $A \cup (B \cup C) = (A \cup B) \cup C$ $A \cap (B \cap C) = (A \cap B) \cap C$
- 分配律： $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
- 德·摩根法则： $\bar{A} \cap \bar{B} = \bar{A} \cup \bar{B}$ $\bar{A} \cup \bar{B} = \bar{A} \cap \bar{B}$

以上这些定律都可以扩展到任意多个事件。

7.1.2 概率论公理

有了事件概念的基础，我们就可以定义概率。

定义 7.1.1. 设 E 是随机试验， Ω 是它的样本空间。对于 E 的每一事件 A 赋予一个实数，记为 $P(A)$ ，称为事件 A 的概率，如果集合函数 $P(\cdot)$ 满足下列条件：

- **非负性** 对每一个事件 A , $P(A) \geq 0$.
- **正则性** 对必然事件 Ω , $P(\Omega) = 1$.
- **可列可加性** 设 A_1, A_2, \dots 是可列个两两互不相容的事件。即当 $i \neq j$ 时, $A_i \cap A_j = \emptyset$, 其中 $i, j = 1, 2, \dots$, 有

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$$

历史上,对概率有两种主要的理解方式,分别是频率派概率(frequentist)和贝叶斯概率(Bayesian probability)。

以抛一枚硬币为例,硬币正面向上落下的概率为 0.5,对于频率学派,他们认为多次的重复投掷硬币,他们期望正面向上的次数占总实验次数的一半。而贝叶斯学派认为,概率是对事情不确定性的定量描述,与信息有关,而不需要重复试验,因此硬币正面向上概率为 0.5 的解释是:相信下一次试验中,硬币正面向上的可能性为 0.5。两种解释方式各有优劣。

7.1.3 独立事件和条件概率

独立事件

如果连续两次抛一枚均匀的硬币,则两次都出现正面的概率是 $1/2 \times 1/2$ 。之所以能将二者相乘是因为我们认为这两次抛硬币的事件是独立的。事件独立的定义如下:

定义 7.1.2. 如果下式成立,则事件 A 和 B 是独立的:

$$P(AB) = P(A)P(B)$$

如果等式

$$P\left(\bigcap_{i \in J} A_i\right) = \prod_{i \in J} P(A_i)$$

对于所有的 I 的子集 J 都成立,则事件集 $\{A_i : i \in I\}$ 是独立的。

需要注意的是,假定 A 与 B 是互斥事件,并且每个事件都有正的概率,它们可能独立吗?答案是否定的,因为 $P(A)P(B) > 0$ 而 $P(AB) = P(\emptyset) = 0$ 。

条件概率

假设 $P(B) > 0$, 定义 B 发生情况下 A 的条件概率如下:

定义 7.1.3. 如果 $P(B) > 0$, 则 A 在 B 下的条件概率为

$$P(A|B) = \frac{P(AB)}{P(B)}$$

从条件概率的定义中可以得到下述引理。

引理 7.1.1. 如果 A 与 B 是相互独立的事件则 $P(AB) = P(A)P(B)$ 。对于任意两个事件 A, B 有

$$P(A|B) = P(A)$$

根据引理, 独立性的另一个解释为: 如果事件 A 和事件 B 相互独立, 那么在已知 B 的情况下不会改变 A 的概率。

7.1.4 贝叶斯公式

贝叶斯公式也称为贝叶斯定理。早在 18 世纪, 由英国学者贝叶斯 (1702 - 1763) 提出。它是机器学习算法的基础, 在信息检索、邮件过滤、文本分类等诸多方面有广泛的应用。这个公式涉及到条件概率公式和全概率公式, 其利用在 A 的条件下 B 发生的概率能够反推出在 B 的条件下 A 发生的概率。

在介绍全概率公式前, 引入样本空间划分的定义。

定义 7.1.4. 设 E 是随机试验, Ω 是它的样本空间。若集合 A_1, A_2, \dots, A_n 满足:

- 1) $A_i \cap A_j = \emptyset$, 若 $i \neq j$, 其中 $i, j = 1, 2, \dots, n$
- 2) $A_1 \cup A_2 \cup \dots \cup A_n = \Omega$

则称 A_1, A_2, \dots, A_n 为样本空间 Ω 的一个划分。若 A_1, A_2, \dots, A_n 为样本空间 Ω 的一个划分。每次试验 A_1, A_2, \dots, A_n 中必有一个也仅有一个发生。

定义 7.1.5. 设试验 E 的样本空间是 Ω , B 是 E 的事件, A_1, A_2, \dots, A_n 为样本空间 Ω 的一个划分, 且 $P(A_i) > 0 (i = 1, 2, \dots, n)$, 则

$$P(B) = P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + \dots + P(B|A_n)P(A_n) = \sum_{j=1}^n P(B|A_j)P(A_j)$$

称为全概率公式。

定义 7.1.6. [贝叶斯定理] 令 A_1, \dots, A_k 是 Ω 的一个划分, 对每一个 i 有 $P(A_i) > 0$, 如果 $P(B) > 0$, 则对 $i = 1, 2, \dots, k$ 有

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum_j P(B|A_j)P(A_j)}$$

通常称 $P(A_i)$ 为 A 的先验概率, 称 $P(A_i|B)$ 为 A 的后验概率。

在机器学习中, 无论模型如何复杂, 均可以用最基本的加法规则和乘法规则进行概率推理。

$$\text{加法规则} \quad p(x, y) = \sum_y p(x, y)$$

$$\text{乘法规则} \quad p(x, y) = p(x)p(y|x)$$

加法规则与乘法规则本质上就是概率论中的全概率公式与贝叶斯公式。

7.2 随机变量及其分布

随机变量是将样本空间中的数据和概率联系起来的纽带, 其描述了事件可能状态的取值, 通过概率分布表明了其每种状态的可能性。本节将对随机变量及其分布进行介绍。

随机变量的定义

定义 7.2.1. 随机变量即映射

$$X : \Omega \rightarrow \mathbb{R},$$

该映射对每一个 $\omega \in \Omega$ 赋予实值 $X(\omega)$ 。

例 7.2.1. 给定一个查询 Q 和文档集 D , 从文档集中随机抽取一篇文档 d_i , 查看 d_i 是否与查询相关。该试验结果的样本空间是 {不相关, 相关}。构造映射 $X(\omega)$:

$$X(\omega) = \begin{cases} 0, & \omega = \text{不相关} \\ 1, & \omega = \text{相关} \end{cases}$$

则 $X(\omega)$ 是一个随机变量。

例 7.2.2. 用随机变量 X 表示股票 A 中午 12 点的股价

$$X(\omega) = x, \omega = \text{股票中午 12 点的股价是 } x \text{ 元}$$

例 7.2.3. 从某部电影的 1000 个影评中抽取 5 个影评, 随机变量 X 表示正类影评(对电影持肯定态度)的个数, $X(\omega)$ 定义为

$$X(\omega) = \begin{cases} 0, & \omega = \text{没有正类影评} \\ 1, & \omega = 1 \text{ 个正类影评} \\ 2, & \omega = 2 \text{ 个正类影评} \\ 3, & \omega = 3 \text{ 个正类影评} \\ 4, & \omega = 4 \text{ 个正类影评} \\ 5, & \omega = 5 \text{ 个正类影评} \end{cases}$$

累积分布函数

给定随机变量 X , 定义它的累积分布函数(分布函数)如下:

定义 7.2.2. 累积分布函数, 简记为大写的 CDF(Cumulative Distribution Function), 表示函数 $F_X : \mathbb{R} \rightarrow [0, 1]$, 其定义为:

$$F_X(x) = P(X \leq x)$$

有时用 F 代替 F_X 来表示 CDF。

通过以下两个例子来进一步了解累积分布函数。

例 7.2.4. 从包含 500 个正类与 500 个负类的影视评论集中又放回的抽取 2 次, 随机变量 X 表示正类影评的个数, 则累积分布函数 CDF 为:

$$F_X(x) = \begin{cases} 0, & x < 0 \\ \frac{1}{4}, & 0 \leq x < 1 \\ \frac{3}{4}, & 1 \leq x < 2 \\ 1, & x \geq 2 \end{cases}$$

注: 使用 1000 个影评中正类影评的频率 0.5 来近似估计抽取到正类影评的概率。

概率质量函数和概率密度函数

随机变量分为离散型和连续型随机变量, 对于离散型随机变量, 我们可以通过概率质量函数刻画随机变量在每个取值的概率; 对于连续型随机变量, 我们可以利用概率密度函数刻画随机变量的概率密度。

离散型概率质量函数

定义 7.2.3. 如果 X 取可数个值 $\{x_1, x_2, \dots\}$, 则 X 是离散的, 定义 X 的概率函数或概率质量函数 (probability mass function, 记为小写的 pmf) 为

$$f_X(x) = P(X = x)$$

因此, 对于 $x \in \mathbb{R}$ 有 $f_X(x) \geq 0$ 并且 $\sum_i f_X(x_i) = 1$ 。有时用 f 代替 f_X . X 的累积分布函数 $F_X(x)$ 和 f_X 的关系如下:

$$F_X(x) = P(X \leq x) = \sum_{x_i \leq x} f_X(x_i)$$

连续型概率密度函数

定义 7.2.4. 如果存在某个函数 f_X 对所有 x 有 $f_X(x) \geq 0$, $\int_{-\infty}^{+\infty} f_X(x)dx = 1$ 并且对任意 $a \leq b$ 有

$$P(a < X \leq b) = \int_a^b f_X(x)dx,$$

则随机变量 X 是连续型随机变量, 函数 f_X 称为概率密度函数 (probability density function, 记为小写的 pdf) 或密度函数, 且有

$$F_X(x) = \int_{-\infty}^x f_X(t)dt,$$

并且 $f_X(x) = F'_X(x)$ 在 F_X 可微的点均成立。

有时用 $\int f(x)dx$ 或者 $\int f$ 表示 $\int_{-\infty}^{+\infty} f(x)dx$ 。

例 7.2.5. 假设 X 的 pdf 为

$$f_X(x) = \begin{cases} 1, & 0 \leq x \leq 1 \\ 0, & \text{其他,} \end{cases}$$

显然 $f_X(x) \geq 0$ 且 $\int f_X(x)dx = 1$ 。具有这种密度的随机变量称它服从 $[0,1]$ 均分分布。其含义就是从 0 到 1 之间随机取一点的概率相等。CDF 为

$$F_X(x) = \begin{cases} 0, & x < 0 \\ x, & 0 \leq x \leq 1 \\ 1, & x > 1 \end{cases}$$

7.2.1 常用的随机变量及其分布

单变元离散型随机变量

若随机变量 X 为有限个或可列无限多个时, 称 X 为离散型随机变量。如例 7.2.4 中随机变量 X 只有 3 个取值, 所以是离散型随机变量。

常用的离散型随机变量及其分布

单点分布 仅在一个点 a 上有概率, 记为 $X \sim \delta_a$, 即 $P(X = a) = 1$, 那么

$$F(x) = \begin{cases} 0, & x < a \\ 1, & x \geq a \end{cases}$$

概率密度函数在 $x = a$ 处 $f(x) = 1$, 其他情形下为 0.

离散均匀分布 令 $k > 1$ 为给定的整数, 假设 X 具有如下 pmf:

$$f(x) = \begin{cases} \frac{1}{k}, & x = 1, 2, \dots, k. \\ 0, & \text{其他.} \end{cases}$$

则称 X 在 $\{1, 2, \dots, k\}$ 上服从均匀分布。

伯努利分布 随机变量 X 只取两个值, 一般用 0, 1 表示, 且 pmf 为:

$$f(x) = \begin{cases} p, & x = 1, \\ 1 - p, & x = 0. \end{cases} \quad (7.1)$$

其中 $p \in [0, 1]$, 称 X 服从伯努利分布, 记为 $X \sim \text{Bernoulli}(p)$, pmf 可简写为:

$$f(x) = p^x(1 - p)^{1-x}$$

其中 $x \in [0, 1]$ 。在统计机器学习中的逻辑回归分类模型假设数据服从伯努利分布, 进而对数据进行建模。

二项式分布 假设从若干影视评论中有放回的抽取 n 次, 令随机变量 X 表示抽取到正类影评的次数。假设每次取影评都是独立的且取到正类影评的概率是 p 。概率密度函数:

$$f(x) = \begin{cases} \binom{n}{x} p^x (1 - p)^{n-x}, & x = 0, 1, \dots, n \\ 0, & \text{其他.} \end{cases}$$

具有上述概率密度函数的随机变量称为二项随机变量, 记 $X \sim \text{Binomial}(n, p)$.

几何分布 如果在二项式分布的示例中, 不是有放回的取 n 次, 而是直到取到正类影评为止, 令随机变量 X 表示第一次取得正类影评的次数, 则 X 的密度函数为:

$$P(X = k) = p(1 - p)^{k-1}, k = 0, 1, 2, \dots$$

则 X 服从参数为 $p \in (0, 1)$ 的几何分布, 记为 $X \sim \text{Geom}(p)$ 。对于几何分布有:

$$\sum_{k=0}^{+\infty} P(X = k) = p \sum_{k=0}^{+\infty} (1 - p)^k = \frac{p}{p} = 1$$

泊松分布 如果

$$f(x = k) = e^{-\lambda} \frac{\lambda^x}{x!}, x \geq 0,$$

则随机变量 X 服从参数为 λ 的泊松分布, 记为 $X \sim \text{Poisson}(\lambda)$ 。并且有:

$$\sum_{x=0}^{+\infty} f(x) = e^{-\lambda} \sum_{x=0}^{+\infty} \frac{\lambda^x}{x!} = e^{-\lambda} e^{\lambda} = 1$$

泊松分布常用于罕见事件的计数, 如放射性元素的衰变与交通事故。

常用的连续型随机变量及其分布

单变元均匀分布 设 X 的概率密度函数为:

$$f_X(x) = \begin{cases} 1, & 0 \leq x \leq 1 \\ 0, & \text{其他} \end{cases}$$

显然 $f_X(x) \geq 0$ 且 $\int_{-\infty}^{\infty} f(x) = 1$ 。称具有这种概率密度函数的随机变量 X 服从 $(0, 1)$ 均匀分布。

单变元正态(高斯)分布 如果随机变量 X 的概率密度函数是:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(x - \mu)^2\right\}$$

则 X 服从参数为 μ 和 σ 的正态分布, 记为 $X \sim N(\mu, \sigma^2)$, 其中 $\mu \in \mathbb{R}, \sigma > 0$.

正态分布在概率和统计中扮演者重要角色, 许多自然现象可以用正态分布来近似, 如男性身高。在深度学习中, 也常利用正态分布对参数初始化。

拉普拉斯分布 如果随机变量 X 的概率密度函数是:

$$f(x) = \frac{1}{2\lambda} \exp\left\{-\frac{|x - \mu|}{\lambda}\right\}$$

其中 $\lambda > 0, \mu$ 为常数, 则 X 服从拉普拉斯分布。

7.2.2 多维随机变量及其分布函数

在实际生产与理论研究中, 都常常会遇到这种情况: 需要同时用几个随机变量才能较好地描绘某一试验或现象。例如我们会根据(颜色, 手感, 气味)的特征组判断水果的好坏。航天飞船返航时的位置需要用(经度, 纬度)来确定。

在数据科学中使用多维随机变量来描述一个数据样本。例如在金融反欺诈领域中要决定是否给一人贷款时,要观察此人的(收入,年龄,是否结婚,学历)等等。称 n 个随机变量 x_1, x_2, \dots, x_n 的总体 $X = (x_1, x_2, \dots, x_n)$ 为 n 元随机变量(或 n 维随机变量)。本节重点讨论二维的情形, n 维情况类似。

多元随机变量的分布函数

定义 7.2.5. 设 (X, Y) 是二维随机变量, 对于任意实数 x, y , 二元函数:

$$F(x, y) = P\{(X \leq x) \cap (Y \leq y)\} = P\{X \leq x, Y \leq y\}$$

称为二维随机变量 (X, Y) 的分布函数, 或称为随机变量 X 和 Y 的联合分布函数。

多元离散型随机变量

定义 7.2.6. 如果随机变量 (X, Y) 的取值是有限对或可列无限多对时, 称二维随机变量 (X, Y) 是离散型随机变量。定义其联合密度函数为 $f(x, y) = P(X = x, Y = y)$ 。

设二维离散型随机变量 (X, Y) 所有可能的取值为 $(x_i, y_j), i, j = 1, 2, \dots$, 记 $P(X = x_i, Y = y_j) = f_{i,j}, i, j = 1, 2, \dots$ 。由概率的定义有

$$f_{i,j} \geq 0, \quad \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} f_{i,j} = 1,$$

称 $f_{i,k} (i, k = 1, 2, \dots)$ 为概率质量函数, 此时累积分布函数是:

$$F(x, y) = \sum_{\substack{x_i < x \\ y_k < y}} P(X = x_i, Y = y_k)$$

多元连续型随机变量

对于二维随机变量的分布函数 $F(x, y)$, 如果存在非负函数 $f(x, y)$ 使得对于任意 x, y 有

$$F(x, y) = \int_{-\infty}^x \int_{-\infty}^y f(u, v) dv du$$

则称 (X, Y) 是连续型二维随机变量, 函数 $f(x, y)$ 称为二维随机变量 (X, Y) 的概率密度, 或称为随机变量 X 和 Y 的联合概率密度。

边缘分布

定义 7.2.7. 如果 (X, Y) 具有联合质量函数 $f_{X,Y}$, 则 X 的边缘概率质量函数定义为:

$$f_X(x) = P(X = x) = \sum_y P(X = x, Y = y) = \sum_y f(x, y)$$

Y 的边缘概率质量函数定义为:

$$f_Y(y) = P(Y = y) = \sum_x P(X = x, Y = y) = \sum_x f(x, y)$$

定义 7.2.8. 对于连续型随机变量, 边缘概率密度函数为

$$f_X(x) = \int_{-\infty}^{+\infty} f(x, y) dy$$

$$f_Y(y) = \int_{-\infty}^{+\infty} f(x, y) dx$$

相应的边缘分布函数记为 F_X 和 F_Y 。

独立的随机变量

定义 7.2.9. 如果对于任意 A 和 B 有

$$P(X \in A, Y \in B) = P(X \in A)P(Y \in B)$$

称随机变量 X 和 Y 是独立的

原则上, 为检验两个随机变量 X 和 Y 是否独立, 需要对所有子集 A 和 B 验证等式 $P(X \in A, Y \in B) = P(X \in A)P(Y \in B)$ 。值得庆幸的是, 对于连续型随机变量有如下结论。事实上, 该结论对离散随机变量也是成立的。

定理 7.2.1. 令 X 和 Y 具有联合概率质量函数或联合概率密度函数 $f_{X,Y}$, 则 X 与 Y 相互独立当且仅当 $f_{X,Y}(x, y) = f_X(x)f_Y(y)$ 对于所有 x 和 y 成立。

条件分布

如果 X 和 Y 是离散的, 则可以计算假设已观察到 $Y = y$ 情况下 X 的条件分布。特别地, $P(X = x|Y = y) = P(X = x, Y = y)/P(Y = y)$ 。从而有如下条件概率密度函数的定义:

定义 7.2.10. 如果 $f_Y(y) > 0$, 则条件概率密度函数为

$$f_{X|Y}(x|y) = P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)} = \frac{f_{X,Y}(x, y)}{f_Y(y)}$$

对于离散的情形下, $f_{X|Y}(x|y)$ 表示 $P(X = x|Y = y)$ 。对于连续型随机变量, 采用相同的概念, 有时需要通过积分求得概率。

定义 7.2.11. 对于连续情形, 假设 $f_Y(y) > 0$, 则条件概率密度函数为

$$f_{X|Y}(x|y) = \frac{f_{X,Y}(x, y)}{f_Y(y)}$$

从而

$$P(X \in A|Y = y) = \int_A f_{X|Y}(x|y) dx$$

条件分布在机器学习中的应用

在机器学习中，监督学习的任务就是学习一个模型，应用这一模型，对给定的输入预测相应的输出。模型的一般形式为决策函数：

$$Y = f(X)$$

或者条件概率分布：

$$P(Y|X)$$

监督学习方法又可以分为生成方法 (generative approach) 和判别方法 (discriminative approach)。所学到的模型分别称为生成模型 (generative model) 和判别模型 (discriminative model)。

生成方法由数据学习联合概率分布 $P(X, Y)$ ，然后求出条件概率分布 $P(X|Y)$ 作为预测的模型，即生成模型：

$$P(Y|X) = \frac{P(X, Y)}{P(X)}$$

这样的方法之所以称为生成方法，是因为模型表示了给定输入 X 产生输出 Y 的生成关系。典型的生成模型有：朴素贝叶斯法，隐马尔可夫模型。

判别方法由数据直接学习决策函数 $f(X)$ 或者条件概率分布 $P(Y|X)$ 作为预测的模型，即为判别模型。判别方法关系的是对于给定的输入 X ，应该预测什么样的输出 Y 。典型的判别模型包括： k 近邻法、感知机、决策树、逻辑回归模型、最大熵模型、支持向量机、提升方法和条件随机场等。

在监督学习中，生成方法和判别方法各有优缺点，适合于不同条件下的学习问题。

生成方法的特点：生成方法可以还原出联合概率分布 $P(X, Y)$ ，而判别方法则不能；生成方法的学习收敛速度更快，即当样本容量增加时，学到的模型可以更快地收敛于真实模型；当存在隐变量时，仍可以用生成方法，此时判别方法就不能用。

判别方法的特点：判别方法直接学习的是条件概率 $P(Y|X)$ 或者决策函数 $f(X)$ ，直接面对预测，往往学习的准确率更高；由于直接学习 $P(Y|X)$ ，可以对数据进行各种程度上的抽象、定义特征并使用特征，因此可以简化学习问题。

独立同分布样本

令 $X = (X_1, \dots, X_n)$ ，其中 X_1, \dots, X_n 为随机变量，则称 X 为 n 维随机向量。令 $f(x_1, \dots, x_n)$ 表示概率密度函数，同二维情形一样，可以定义边缘分布、条件分布等。

如果对于任意集合 A_1, \dots, A_n 有：

$$P(X_1 \in A_1, \dots, X_n \in A_n) = \prod_{i=1}^n P(X_i \in A_i).$$

称 X_1, \dots, X_n 是独立的。容易验证 $f(x_1, \dots, x_n) = \prod_{i=1}^n f_{X_i}(x_i)$ 成立。

定义 7.2.12. 如果 X_1, \dots, X_n 独立并且都有相同的累积分布函数 F , 则称 X_1, \dots, X_n 是独立同分布的 (IID, 或 i.i.d.), 记为

$$X_1, \dots, X_n \sim F.$$

如果 F 的密度函数为 f , 也可记为 $X_1, \dots, X_n \sim f$, 有时也称 X_1, \dots, X_n 是来自 F , 样本量为 n 的随机样本。

许多统计理论和事件都建立在 IID 观测的假设基础上, 当讨论统计量的时候将对它作详细研究。

7.3 随机变量的数字特征

随机变量的分布函数描述随机变量的统计规律。但实际上并不知道随机变量的真实分布。有时只关心随机变量的某些方面的数字特征——期望, 方差, 协方差。

期望 $E(X)$ 反映了随机变量的平均取值水平。比如一个班级的平均成绩, 某地区人的平均身高等。在机器学习任务中, 我们的目标是学习到损失函数 (记为 $L(Y, f(X))$) 的期望最小的模型 f 。 $R_{exp}(f) = E[L(Y, f(X))]$ 称为期望损失。我们利用模型在训练集上的期望损失学习模型; 利用模型在测试集上的期望损失衡量模型的泛化性能。

方差 $D(X)$ 反映随机变量与期望的平均偏离程度。在机器学习中方差常用来度量同样大小的训练集的变动所导致模型性能的变化幅度, 是模型性能的评价指标之一。

协方差 $\text{Cov}(X, Y)$ 反映两个随机变量 X, Y 的整体误差。如果两个变量的变化趋势一致, 也就是说如果当 X 大于自身的 $E(X)$ 且 Y 大于自身的 $E(Y)$ 时, 那么两个变量之间的协方差就是正值; 如果两个变量的变化趋势相反, 那么两个变量之间的协方差就是负值。通过协方差可以计算相关系数, 而相关系数反映了不同特征间的线性相关程度。也可作为模型建立中特征的选择建立评价指标。

7.3.1 期望

随机变量 X 期望表示 X 在其分布上的平均值, 其定义如下:

定义 7.3.1. 随机变量 X 的期望值或均值定义为

$$E(X) = \int_{-\infty}^{+\infty} x dF(x) = \begin{cases} \sum_x x f(x), & X \text{ 为离散型随机变量,} \\ \int_x x f(x) dx, & X \text{ 为连续型随机变量。} \end{cases}$$

其中 $F(x)$ 是累积分布函数。加入以上求和 (或积分) 定义明确, 也可使用如下符号表示 X 的期望:

$$E(X) = EX = \int_{-\infty}^{+\infty} x dF(x) = u = u_X.$$

期望是分布的单值概括，可以将 $E(X)$ 看成是 IID 随机样本 X_1, \dots, X_n 的平均 $\sum_{i=1}^n X_i/n$ 。事实上， $E(X) \approx \sum_{i=1}^n X_i/n$ 是正确的而不是主观的推断，这点将在 7.5 节详细说明。

符号 $\int x dF(x)$ 仅仅用来统一符号，而不用将离散形式写成 $\sum_x x f(x)$ ，将连续形式写成 $\int x f(x) dx$ 。

为保证 $E(X)$ 定义明确，如果 $\int_x |x| dF_X(x) < \infty$ ，则称 $E(X)$ 存在。否则称期望不存在。

实际上，我们经常需要求随机变量函数的数学期望，例如飞机机翼受到压力 $W = kV^2$ (V 是风速， $k > 0$ 是常数) 的作用，需要求 W 的数学期望，这里 W 是随机变量 V 的函数。这时，可以通过下面的定理来求 W 的数学期望。

定理 7.3.1. 设 Y 是随机变量 X 的函数： $Y = r(X)$ 。则

$$E(Y) = E(r(X)) = \int r(x) dF_X(x).$$

当我们求 $E(r(X))$ 时，不必算出 $r(X)$ 的概率密度函数，而只需利用 X 的概率密度函数就可以了。

期望的性质

根据期望的定义，我们可以得出许多有关期望的性质。

性质 7.3.1. 设 C 是常数，则有 $E(C) = C$ 。

性质 7.3.2. 设 X 是随机变量， C 是常数，则有

$$E(CX) = CE(X).$$

性质 7.3.3. 设 X 和 Y 是两个随机变量，则有：

$$E(X + Y) = E(X) + E(Y).$$

性质 7.3.4. 设 X 和 Y 是相互独立的两个随机变量，则有：

$$E(XY) = E(X)E(Y).$$

上述性质也可以扩张到 n 个独立的随机变量。若 X_1, X_2, \dots, X_n 为独立的随机变量，则

$$E\left(\prod_{i=1}^n X_i\right) = \prod_{i=1}^n E(X_i).$$

应用举例

期望在机器学习中随处可见。比如在衡量分类模型准确率时我们使用期望来定义：

$$\text{accuracy} = E(I(y = \hat{f}(x))) = \sum I(y = \hat{f}(x)) dF(x)$$

其中 $F(x)$ 是随机变量 x 的累积分布函数, $\hat{f}(x)$ 是训练好的模型, 且:

$$I(y = \hat{f}(x)) = \begin{cases} 1 & \text{如果 } y = \hat{f}(x) \\ 0 & \text{如果 } y \neq \hat{f}(x) \end{cases}$$

但在实际中, 我们并不知道数据真实的概率分布函数, 常取的做法是采集 n 个未在训练集中出现过的样本作为测试集, 使用:

$$\sum_i^n \frac{1}{n} I(y_i = \hat{f}(x_i))$$

来衡量模型的准确率。

此外强化学习中价值函数 (状态价值函数和动作价值函数) 也是由数学期望来定义的。

7.3.2 方差

定义 7.3.2. 令随机变量 X 的均值为 μ , X 的方差记为 σ^2 或 $D(X)$, 定义为

$$\begin{aligned} \sigma^2 &= E(X - \mu)^2 \\ &= \int (x - \mu)^2 dF(X) \\ &= \begin{cases} \sum_x (x - \mu)^2 f(x), & X \text{ 为离散型随机变量,} \\ \int_x (x - \mu)^2 f(x) dx, & X \text{ 为连续型随机变量。} \end{cases} \end{aligned}$$

其中假设期望存在。标准差定义为 $sd(X) = \sqrt{D(X)}$ 也记为 σ 或 σ_X 。

例 7.3.1. 设随机变量 X 具有数学期望 $E(X) = \mu$, 方差 $D(X) = \sigma^2 \neq 0$, 记

$$X^* = \frac{X - \mu}{\sigma},$$

则

$$E(X^*) = \frac{1}{\sigma} E(X - \mu) = \frac{1}{\sigma} [E(X) - \mu] = 0,$$

$$D(X^*) = E((X^*)^2) = \int \left(\frac{x - \mu}{\sigma}\right)^2 f(x) dx = \frac{1}{\sigma^2} \int (x - \mu)^2 f(x) dx = 1.$$

因此对于任何一个具有均值和方差的分布, 我们总可以通过这样的变换将其变为均值为 0, 方差为 1 的分布。

方差的性质

性质 7.3.5. 设 X 是随机变量, 有

$$D(X) = E(X^2) - [E(X)]^2.$$

性质 7.3.6. 设 C 是常数, 则有 $D(C) = 0$.

性质 7.3.7. 设 X 是随机变量, C 是常数, 则有

$$D(CX) = C^2 D(X), \quad D(X + C) = D(X).$$

性质 7.3.8. 设 X 和 Y 是两个随机变量, 则有:

$$D(X + Y) = D(X) + D(Y) + 2E\{(X - E(X))(Y - E(Y))\}.$$

若 X 与 Y 相互独立, 则有:

$$D(X + Y) = D(X) + D(Y).$$

此性质也可扩展到 n 个随机变量的情形。假设 X_1, X_2, \dots, X_n 是随机变量, a_1, a_2, \dots, a_n 是常数, 则

$$D\left(\sum_{i=1}^n a_i X_i\right) = \sum_{i=1}^n a_i^2 D(X_i).$$

如果 X_1, X_2, \dots, X_n 是随机变量, 则定义样本均值

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i,$$

样本方差为

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2.$$

定理 7.3.2. 令 X_1, X_2, \dots, X_n 是独立同分布的随机变量且 $\mu = E(X_i), \sigma^2 = D(X_i)$, 则

$$E(\bar{X}_n) = \mu, D(\bar{X}_n) = \frac{\sigma^2}{n}, E(S_n^2) = \sigma^2.$$

7.3.3 一些重要分布的期望和方差

下面介绍一些重要分布的期望与方差。

伯努利分布的期望与方差 设随机变量 X 服从参数为 p 的伯努利分布:

X	1	0
$P(X)$	p	$1-p$

则 X 的期望:

$$E(X) = 1 \times p + 0 \times (1-p) = p,$$

X 的方差:

$$D(X) = (1-p)^2 \times p + (0-p)^2 \times (1-p) = p(1-p).$$

二项分布的期望和方差 设随机变量 X 服从参数为 n, p 的二项式分布:

X	0	1	2	\dots	n
$P(X)$	$C_n^0 p^0 (1-p)^{n-0}$	$C_n^1 p^1 (1-p)^{n-1}$	$C_n^2 p^2 (1-p)^{n-2}$	\dots	$C_n^n p^n (1-p)^0$

则随机变量 X 的期望

$$E(X) = \sum_{k=0}^n k \times P(X=k) = \sum_{k=0}^n k \times C_n^k p^k (1-p)^{n-k} = np,$$

方差 $D(X)$:

$$\begin{aligned} D(X) &= \sum_{k=0}^n (k - E(X))^2 \times P(X=k) \\ &= \sum_{k=0}^n (k - np)^2 \times C_n^k p^k (1-p)^{n-k} \\ &= np(1-p). \end{aligned}$$

几何分布的期望和方差 设随机变量 X 服从参数为 p 的几何分布, $0 < p < 1$:

X	1	2	3	\dots	n	\dots
$P(X)$	p	$(1-p)p$	$(1-p)^2 p$	\dots	$(1-p)^{n-1} p$	\dots

即:

$$P(X=k) = (1-p)^{k-1} p,$$

随机变量 X 的期望:

$$E(X) = \sum_{k=1}^n k \times P(X=k) = \sum_{k=1}^n k \times (1-p)^{k-1} p = \frac{1}{p},$$

方差 $D(X)$:

$$D(X) = \sum_{k=1}^n (k - E(X))^2 \times P(X=k) = \sum_{k=1}^n (k - \frac{1}{p})^2 \times (1-p)^{k-1} p = \frac{1-p}{p^2}.$$

泊松分布的期望和方差 随机变量 X 服从参数为 λ 的泊松分布:

$$F(x=k) = e^{-\lambda} \frac{\lambda^k}{k!}, \quad k = 0, 1, 2, \dots,$$

则随机变量 X 的期望:

$$E(X) = \sum_{k=0}^{\infty} k e^{-\lambda} \frac{\lambda^k}{k!} = \lambda,$$

方差:

$$D(X) = \lambda.$$

均匀分布的期望和方差 设随机变量 X 服从均匀分布, 记为 $X \sim U(a, b)$, 其概率密度函数:

$$f(x) = \begin{cases} 0, & X < a \\ \frac{1}{b-a}, & a \leq x \leq b \\ 0, & x > b \end{cases}$$

则随机变量 X 的期望 $E(X)$:

$$E(X) = \int_{-\infty}^{+\infty} xf(x)dx = \int_a^b x \frac{1}{b-a} dx = \frac{b+a}{2},$$

方差 $D(X)$:

$$D(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 f(x)dx = \int_a^b (x - \frac{b+a}{2})^2 \frac{1}{b-a} dx = \frac{(b-a)^2}{12}.$$

Laplace 分布的期望和方差 设随机变量 X 服从 Laplace 分布, 其密度函数如下:

$$\text{Laplace}(x; \mu, \gamma) = \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right),$$

则 X 的期望 $E(X)$:

$$E(X) = \int_{-\infty}^{+\infty} xf(x)dx = \int_{-\infty}^{+\infty} x \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right) dx = \mu,$$

方差 $D(X)$:

$$D(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 f(x)dx = \int_{-\infty}^{+\infty} (x - \mu)^2 \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right) dx = 2\gamma^2.$$

高斯分布的期望和方差 设 X 服从参数为 μ 和 σ 的高斯分布, $X \sim N(x; \mu, \sigma)$:

$$N(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

则随机变量的期望 $E(X)$:

$$E(X) = \int_{-\infty}^{+\infty} xf(x)dx = \int_{-\infty}^{+\infty} x \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = \mu,$$

方差 $D(X)$:

$$D(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 f(x)dx = \int_{-\infty}^{+\infty} (x - \mu)^2 \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = \sigma^2.$$

7.3.4 协方差和相关系数

对于二维随机变量 (X, Y) , 除了讨论期望与方差之外, 还需讨论 X 和 Y 之间的相关关系的数字特征。

定义 7.3.3. 令 X 和 Y 是均值分别为 μ_X 和 μ_Y , 标准差分别是 σ_X 和 σ_Y 的随机变量, 定义 X 和 Y 的协方差:

$$\text{Cov}(X, Y) = E\{(X - \mu_X)(Y - \mu_Y)\},$$

相关系数:

$$\rho_{XY} = \frac{\text{Cov}(X, Y)}{\sqrt{D(X)}\sqrt{D(Y)}} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}.$$

由协方差定义可知

$$\text{Cov}(X, Y) = \text{Cov}(Y, X) \quad \text{Cov}(X, X) = D(X)$$

以及对于任意两个随机变量 X 和 Y , 下列等式成立:

$$D(X + Y) = D(X) + D(Y) + 2\text{Cov}(X, Y)$$

将 $\text{Cov}(X, Y)$ 的定义展开, 易得

$$\text{Cov}(X, Y) = E(XY) - E(X)E(Y)$$

我们常常利用这一式子得出以下协方差性质:

性质 7.3.9.

$$\text{Cov}(aX, bY) = ab\text{Cov}(X, Y)$$

性质 7.3.10.

$$\text{Cov}(X_1 + X_2, Y) = \text{Cov}(X_1, Y) + \text{Cov}(X_2, Y)$$

同时相关系数具有以下性质:

性质 7.3.11.

$$|\rho_{XY}| \leq 1$$

当 $\rho = 0$ 时, 称随机变量 X 和 Y 不相关。

性质 7.3.12. $|\rho_{XY}| = 1$ 的充要条件是存在 a, b 使 $P(Y = a + bX) = 1$

例 7.3.2. 设 (X, Y) 服从二维正态分布, 它的概率密度函数

$$f(x, y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left\{ \frac{-1}{2(1-\rho^2)} \left[\frac{(x-\mu_1)^2}{\sigma_1^2} - 2\rho\frac{(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2} + \frac{(y-\mu_2)^2}{\sigma_2^2} \right] \right\}$$

我们可以分别计算出 X 和 Y 的边缘概率分布, 然后分布求出 X 和 Y 的期望, 方差以及相关系数。 $E(X) = \mu_1, E(Y) = \mu_2, D(X) = \sigma_1^2, D(Y) = \sigma_2^2, \text{Cov}(X, Y) = \rho$, 具体过程留给读者自行计算。

这也就是说, 二维正态随机变量 (X, Y) 的概率密度中的参数 ρ 就是 X 和 Y 的相关系数, 因而二维正态随机变量的分布完全可由 X, Y 各自的数学期望, 方差以及它们的相关系数所确定。

定理 7.3.3. 若 (X, Y) 服从二维正态分布, 那么 X 和 Y 相互独立的充要条件是 $\rho = 0$ 。

随机变量的内积

如果两个随机变量 X 和 Y 是不相关的, 那么:

$$D(X+Y) = D(X) + D(Y),$$

因为方差是以平方来衡量的, 这看起来很像勾股定理:

$$c^2 = a^2 + b^2,$$

其中 a, b 是直角三角形的直角边, c 是直角三角形的斜边。

接下来, 我们尝试从几何学的角度来解释不相关随机变量之间的关系。随机变量能被看成向量空间中的向量, 我们能定义内积来获取随机变量的几何性质, 如果我们定义随机变量的内积:

$$\langle X, Y \rangle := \text{Cov}(X, Y).$$

随机变量的长度是:

$$\|X\| = \sqrt{\text{Cov}(X, X)} = \sqrt{D(X)} = \sigma(X).$$

我们知道如果两个向量 $X \perp Y \Leftrightarrow \langle X, Y \rangle = 0$ 。对随机变量我们这种情况意味着 X 与 Y 正交当且仅当 $\text{Cov}(X, Y) = 0$ 或者说 X 与 Y 不相关。

7.3.5 矩和协方差矩阵

本节先介绍随机变量的另外几个数字特征。设 (X, Y) 是二维随机变量。

定义 7.3.4. 设 X 和 Y 是随机变量, 若

$$E(X^k), \quad k = 1, 2, \dots$$

存在, 称它为 X 的 k 阶原点矩, 简称 k 阶矩。

若

$$E\{[X - E(X)]^k\}, \quad k = 2, 3, \dots$$

存在, 称它为 X 的 k 阶中心矩。

若

$$E\{X^k Y^l\}, \quad k, l = 1, 2, 3, \dots$$

存在, 称它为 X 和 Y 的 $k+l$ 阶混合矩。

若

$$E\{[X - E(X)]^k [Y - E(Y)]^l\}, \quad k, l = 1, 2, 3, \dots$$

存在, 称它为 X 和 Y 的 $(k+l)$ 阶混合中心矩。

显然 X 的数学期望 $E(X)$ 是 X 的一阶原点矩, 方差 $D(X)$ 是 X 的二阶中心矩, 协方差 $\text{Cov}(X, Y)$ 是 X 和 Y 的二阶混合中心矩。

下面介绍 n 维随机变量的协方差矩阵。先从二维随机变量讲起。

二维随机变量 (X_1, X_2) 有 4 个二阶中心矩 (假设它们都存在), 分别记为

$$c_{11} = E \{ [X_1 - E(X_1)]^2 \}$$

$$c_{12} = E \{ [X_1 - E(X_1)][X_2 - E(X_2)] \}$$

$$c_{21} = E \{ [X_2 - E(X_2)][X_1 - E(X_1)] \}$$

$$c_{22} = E \{ [X_2 - E(X_2)]^2 \}$$

将它们排成矩阵的形式

$$\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

这个矩阵称为随机变量 (X, Y) 的协方差矩阵。

设 n 维随机变量 (X_1, X_2, \dots, X_n) 的二阶混合中心矩

$$c_{ij} = \text{Cov}(X_i, Y_j) = E[X_i - E(X_i)][X_j - E(X_j)], i, j = 1, 2, \dots, n$$

都存在, 则称矩阵:

$$C = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix}$$

为 n 维随机变量 (X_1, X_2, \dots, X_n) 的协方差矩阵。由于 $c_{ij} = c_{ji} (i \neq j; i, j = 1, 2, \dots, n)$, 因此上述矩阵是一个对称矩阵。

一般情况下 n 维随机变量的分布是不知道的, 或者太过复杂, 以致在数学上不易处理, 因此在实际应用中协方差矩阵就显得重要了。我们以 n 维正态分布为例来介绍 n 维随机变量。在介绍 n 维正态分布的概率密度函数之前, 我们先将二维正态分布的概率密度函数改成另外一种形式, 以便将它推广到 n 维随机变量的场合中去。二维正态随机变量 (X_1, X_2) 的概率密度函数为

$$f(x, y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp \left\{ \frac{-1}{2(1-\rho^2)} \left[\frac{(x-\mu_1)^2}{\sigma_1^2} - 2\rho \frac{(x-\mu_1)(y-\mu_2)}{\sigma_1\sigma_2} + \frac{(y-\mu_2)^2}{\sigma_2^2} \right] \right\}$$

现将上式中花括号内的式子写成矩阵形式, 为此引入下面的列向量

$$X = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

(X_1, X_2) 的协方差矩阵为

$$C = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

它的行列式 $|C| = \sigma_1^2\sigma_2^2(1 - \rho^2)$, C 的逆矩阵为

$$C^{-1} = \frac{1}{|C|} \begin{pmatrix} \sigma_2^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_1^2 \end{pmatrix}$$

经计算可知 (这里矩阵 $(X - \mu)^T$ 是 $(X - \mu)$ 的转置矩阵)

$$(X - \mu)^T C^{-1} (X - \mu) = \frac{1}{|C|} (x_1 - u_1 x_2 - u_2) \begin{pmatrix} \sigma_2^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_1^2 \end{pmatrix} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}$$

$$(X - \mu)^T C^{-1} (X - \mu) = \frac{1}{1 - \rho^2} \left[\frac{(x - \mu_1)^2}{\sigma_1^2} - 2\rho \frac{(x - \mu_1)(y - \mu_2)}{\sigma_1\sigma_2} + \frac{(y - \mu_2)^2}{\sigma_2^2} \right]$$

于是 (X_1, X_2) 的概率密度函数可写成

$$f(x_1, x_2) = \frac{1}{(2\pi)^{2/2}(|C|)^{1/2}} \exp \left\{ -\frac{1}{2} (X - \mu)^T C^{-1} (X - \mu) \right\}$$

上式容易推广到 n 维正态随机变量 (X_1, X_2, \dots, X_n) 的情况。引入列矩阵

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \begin{pmatrix} E(X_1) \\ E(X_2) \\ \vdots \\ E(X_n) \end{pmatrix}$$

n 维正态随机变量 (X_1, X_2, \dots, X_n) 的概率密度函数定义为:

$$f(x_1, x_2, \dots, x_n) = \frac{1}{(2\pi)^{n/2}(|C|)^{1/2}} \exp \left\{ -\frac{1}{2} (X - \mu)^T C^{-1} (X - \mu) \right\}$$

其中 C 是 (X_1, X_2, \dots, X_n) 的协方差矩阵。

n 维正态随机变量具有以下四条重要的性质 (证略):

1. n 维正态随机变量 (X_1, X_2, \dots, X_n) 的每一个分量 $X_i, i = 1, 2, \dots, n$ 都是 n 维正态随机变量; 反之, 若 X_1, X_2, \dots, X_n 都是正态随机变量, 且相互独立, 则 (X_1, X_2, \dots, X_n) 是 n 维正态随机变量。

2. n 维随机变量 X_1, X_2, \dots, X_n 服从 n 维正态分布的充要条件是 X_1, X_2, \dots, X_n 的任意线性组合

$$l_1 X_1 + l_2 X_2 + \dots + l_n X_n$$

服从一维正态分布 (其中 l_1, l_2, \dots, l_n 不全为 0)

3. 若 X_1, X_2, \dots, X_n 服从 n 维正态分布, 设 Y_1, Y_2, \dots, Y_k 是 $X_j (j = 1, 2, \dots, n)$ 的线性函数, 则 (Y_1, Y_2, \dots, Y_k) 也服从多维正态分布。

这一性质称为正态变量的线性变换不变性。

4. 设 (X_1, X_2, \dots, X_n) 服从 n 维正态分布, 则 “ X_1, X_2, \dots, X_n ” 相互独立与 “ X_1, X_2, \dots, X_n ” 两两不相关是等价的。

协方差矩阵的半正定性

本节以二元离散随机变量为例, 来介绍协方差矩阵的半正定性质, 对于 n 元随机变量类似。对于二元随机变量 (x, y) , 其协方差矩阵为:

$$V = \sum_{i,j} p_{ij} V_{i,j} = p_{ij} \begin{bmatrix} (x_i - \mu_x)^2 & (x_i - \mu_x)(y_j - \mu_y) \\ (x_i - \mu_x)(y_j - \mu_y) & (y_j - \mu_y)^2 \end{bmatrix}$$

其中 $p_{ij} = F(X = i, Y = j)$ 的概率。注意矩阵 $V_{i,j}$:

$$V_{i,j} = \begin{bmatrix} x_i - \mu_x \\ y_j - \mu_y \end{bmatrix} \begin{bmatrix} x_i - \mu_x & y_j - \mu_y \end{bmatrix} = \mathbf{u} \mathbf{u}^T$$

其中 $\mathbf{u} = \begin{bmatrix} x_i - \mu_x \\ y_j - \mu_y \end{bmatrix}$, 由于 $p_{ij} > 0$ 且 $V_{i,j}$ 是秩为 1 的半正定矩阵, 故 $p_{ij} V_{i,j}$ 也是半正定矩阵。

根据半正定矩阵之和仍是半正定矩阵的性质, 那么 V 是半正定矩阵。因此协方差矩阵是半正定矩阵。

7.3.6 条件期望

假设 X 和 Y 为随机变量, 当 $Y = y$ 时 X 的均值是多少? 方法跟前面计算 X 的均值一样, 只不过将期望定义中的 $f_X(x)$ 用 $f_{X|Y}(x|y)$ 代替就可以了。

定义 7.3.5. 给定 $Y = y$ 情况下 X 的条件期望为

$$E(X|Y = y) = \begin{cases} \sum x f_{X|Y}(x|y), & \text{离散情形,} \\ \int x f_{X|Y}(x|y) dx, & \text{连续情形} \end{cases}$$

如果 $r(x, y)$ 为 x, y 的函数, 则

$$E(r(X, Y)|Y = y) = \begin{cases} \sum r(x, y) f_{X|Y}(x|y), & \text{离散情形,} \\ \int r(x, y) f_{X|Y}(x|y) dx, & \text{连续情形} \end{cases}$$

注意! 条件期望与期望有一些区别, 期望 $E(X)$ 是一个值, 而条件期望 $E(X|Y = y)$ 是 y 的函数。在观察 y 之前, 并不知道 $E(X|Y = y)$ 的值, 所以它是一个随机变量, 记为 $E(X|Y)$ 。换句话说, $E(X|Y)$ 是随机变量, 当 $Y = y$ 时, 其值为 $E(X|Y = y)$ 。类似的, $E(r(X, Y)|Y)$ 是随机变量, 当 $Y = y$ 时, 其值为 $E(r(X, Y)|Y = y)$ 。这一点很容易引起混淆, 下面举一个例子来说明。

例 7.3.3. 假设 $X \sim \text{Uniform}(0, 1)$, 当观察到 $X = x$ 后, 假设 $Y|X = x \sim \text{Uniform}(x, 1)$, 凭直觉 $E(Y|X = x) = (1 + x)/2$, 事实上 $f_{Y|X}(y|x = 1) = 1/(1 - x)$, 其中 $x < y < 1$, 故

$$E(Y|X = x) = \int_0^1 y f_{Y|X}(y|x = 1) dy = \frac{1}{1-x} \int_x^1 y dy = \frac{1+x}{2}$$

因此 $E(Y|X) = (1 + X)/2$, 它是一个随机变量。当观察到 $X = x$ 后, 其值为 $E(Y|X = x) = (1 + x)/2$

定理 7.3.4. (期望迭代法则) 对随机变量 X 和 Y , 假设期望均存在, 则有

$$E[E(Y|X)] = E(Y), \quad E[E(X|Y)] = E(X)$$

更一般的, 对任意函数 $r(x, y)$ 有

$$E[E(r(X, Y))|X] = E(r(X, Y))$$

证明. 下面证明第一个等式, 利用条件期望的定义和 $f(x, y) = f(x)f(y|x)$

$$\begin{aligned} E[E(Y|X)] &= \int E(Y|X=x)f_X(x)dx = \int \int yf(y|x)dyf(x)dx \\ &= \int \int yf(y|x)f(x)dxdy = \int \int yf(x, y)dxdy = E(Y) \end{aligned}$$

□

例 7.3.4. 回到例 7.3.3 中, 试问怎么计算 $E(Y)$?

解. 一种方法是求出联合密度函数 $f(x, y)$, 然后计算 $E(Y) = \int \int yf(x, y)dxdy$ 。另一种更简单的方式可以分两步来实现。首先计算 $E(Y|X) = (1+X)/2$, 从而

$$\begin{aligned} E(Y) &= E[E(Y|X)] = E((1+X)/2) \\ &= \frac{1+E(X)}{2} = \frac{1+(1/2)}{2} = \frac{3}{4}. \end{aligned}$$

7.3.7 方差的应用: 过拟合与偏差-方差分解

如 1.3 节所述, 在机器学习领域, 我们将数据划分为训练集和测试集并在训练集上训练得到模型 \hat{f} 。如果模型在训练集上表现很好(如对于分类问题, 表现好意味着分类准确率高), 而在测试集上表现很差, 则此时的模型处于过拟合状态。为了避免过拟合, 我们需要在模型的拟合能力与复杂度之间进行权衡。拟合能力强的模型一般复杂度会比较高, 容易导致过拟合。相反, 如果限制模型的复杂度, 降低其拟合能力, 又可能会导致欠拟合。因此, 如何在模型的拟合能力和复杂度之间取得一个较好的平衡, 对一个机器学习算法来讲十分重要。偏差-方差分解 (Bias-Variance Decomposition) 为我们提供一个分析模型泛化能力和拟合能力的工具。

以回归问题为例。假设样本的真实分布是 $p_r(x, y)$, 并采用平方损失函数, 模型 $f(x)$ 的期望误差为:

$$\mathcal{R}(f) = E_{(x, y) \sim p_r(x, y)} [(y - f(x))^2],$$

那么最优模型为:

$$f^*(x) = E_{y \sim p_r(y|x)} [y],$$

其中 $p_r(y|x)$ 为样本的真实条件分布, $f^*(x)$ 为使用平方损失作为优化目标的最优模型, 其损失为:

$$\varepsilon = E_{(x, y) \sim p_r(x, y)} [(y - f^*(x))^2].$$

通常损失 ε 是由样本分布以及噪声引起的，无法通过优化模型来减少。

期望误差可以分解为：

$$\begin{aligned}\mathcal{R} &= E_{(x,y) \sim p_r(x,y)} [(y - f^*(x) + f^*(x) - f(x))^2] \\ &= E_{(x,y) \sim p_r(x,y)} [(y - f^*(x))^2] + E_{(x,y) \sim p_r(x,y)} [(f^*(x) - f(x))^2] \\ &= \varepsilon + E_{x \sim p_r(x)} [(f^*(x) - f(x))^2],\end{aligned}\quad (7.2)$$

其中

$$\begin{aligned}2E_{(x,y) \sim p_r(x,y)} [(y - f^*(x))(f^*(x) - f(x))] &= 2 \int_x \int_y p_r(x,y) (y - f^*(x))(f^*(x) - f(x)) dx dy \\ &= 2 \int_x (f^*(x) - f(x)) dx \int_y p_r(x,y) (y - f^*(x)) dy,\end{aligned}$$

对于给定的 x_0 ：

$$\begin{aligned}\int_y p_r(x_0, y) ((y - f^*(x_0)) dy &= \int_y p_r(x_0, y) (y - f^*(x_0)) dy \\ &= p_r(x_0) \int_y p_r(y|x_0) (y - f^*(x_0)) dy,\end{aligned}$$

由于

$$f^*(x_0) = E_{y \sim p_r(y|x_0)} [y] = \int_y p_r(y|x_0) y dy,$$

故

$$\int_y p_r(x_0, y) ((y - f^*(x_0))(f^*(x_0) - f(x_0)) dy = 0,$$

$$2E_{(x,y) \sim p_r(x,y)} [(y - f^*(x))(f^*(x) - f(x))] = 0.$$

式(7.2)中的第一项是当前训练出的模型与最优模型之间的差距，是机器学习算法可以优化的目标。

在实际训练一个模型 $f(x, y)$ 时，训练集 D 是从真实分布 $p_r(x, y)$ 上独立同分布地采样出来的有限样本集合。不同的训练集会得到不同的模型。令 $f_D(x)$ 表示在训练集 D 学习到的模型，一个机器学习算法（包括模型以及优化算法）的能力可以用不同训练集上的模型的平均性能来评价。

对于单个样本 x ，不同训练集 D 得到模型 $f_D(x)$ 和最优模型 $f^*(x)$ 的期望差距为

$$\begin{aligned}E_D [(f_D(x) - f^*(x))^2] &= E_D [(f_D(x) - E_D [f_D(x)] + E_D [f_D(x)] - f^*(x))^2] \\ &= (E_D [f_D(x)] - f^*(x))^2 + E_D [(f_D(x) - E_D [f_D(x)])^2],\end{aligned}\quad (7.3)$$

式(7.3)中第一项 $(E_D [f_D(x)] - f^*(x))^2$ 称为偏差 (Bias) 的平方，记为 $(bias.x)^2$ ，是指一个模型在不同训练集上的平均性能和最优模型的差异。第二项 $E_D [(f_D(x) - E_D [f_D(x)])^2]$ 称为方差

(Variance), 记为 $variance.x$, 是指一个模型在不同训练集上的差异, 可以用来衡量一个模型是否容易过拟合。

用 $E_D [(f_D(x) - f^*(x))^3]$ 来代替式(7.2)中的 $(f(x) - f^*(x))^2$, 则期望错误可写成:

$$\begin{aligned}\mathcal{R}(f) &= E_{x \sim p_r(x)} [E_D [(f_D(x) - f^*(x))^2]] + \varepsilon \\ &= (bias)^2 + variance + \varepsilon,\end{aligned}\quad (7.4)$$

其中:

$$(bias)^2 = E_x [(E_D[f_D(x)] - f^*(x))^2],$$

$$variance = E_x [E_D[(f_D(x) - E_D[F_D(x)])^2]].$$

所以最小化期望误差等价于最小化偏差与方差之和。

图7.2给出了机器学习模型的四种偏差和方差组合情况。每个图的中心点为最优模型 $f^*(x)$, 蓝点为不同训练集 D 上得到的模型 $f_D(x)$. 图7.2(a) 给出了一种理想情况, 方差和偏差都比较小。图7.2(b) 为高偏差低方差的情况, 表示模型的泛化能力很好, 但拟合能力不足。图7.2(c) 为低偏差高方差的情况, 表示模型的拟合能力很好, 但泛化能力比较差。当训练数据比较少时会导致过拟合。图7.2(d) 为高偏差高方差的情况, 是一种最差的情况。

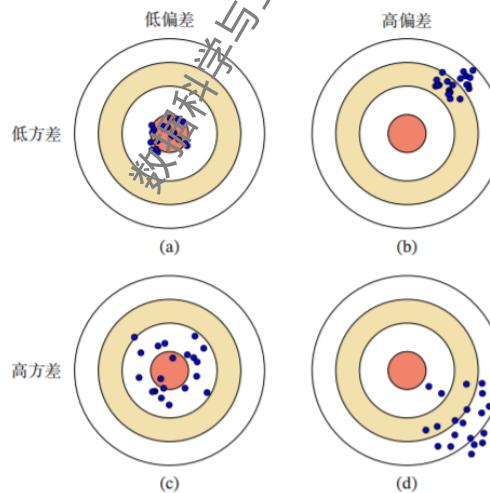


图 7.2: 偏差与方差的组合

方差一般会随着训练样本的增加而减少。当样本比较多时, 方差比较少, 这时可以选择能力强的模型来减少偏差。然而在很多机器学习任务上, 训练集往往都比较有限, 最优的偏差和最优的方差就无法兼顾。

随着模型复杂度的增加，模型的拟合能力变强，偏差减少而方差增大，从而导致过拟合。以结构错误最小化为例，我们可以调整正则化系数来控制模型的复杂度。当正则化系数变大时，模型复杂度会降低，可以有效地减少方差，避免过拟合，但偏差会上升。当正则化系数过大时，总的期望错误反而会上升。因此，一个好的正则化系数需要在偏差和方差之间取得比较好的平衡。图7.3给出了机器学习模型的期望错误、偏差和方差随复杂度的变化情况，其中红色虚线表示最优模型。最优模型并不一定是偏差曲线和方差曲线的交点。

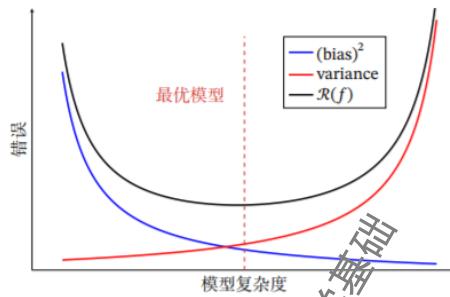


图 7.3: 机器学习模型的期望错误、偏差和方差随复杂度的变化情况

偏差和方差分解给机器学习模型提供了一种分析途径，但在实际操作中难以直接衡量。一般来说，当一个模型在训练集上的错误率比较高时，说明模型的拟合能力不够，偏差比较高。这种情况可以通过增加数据特征、提高模型复杂度、减少正则化系数等操作来改进模型。当模型在训练集上的错误率比较低，但验证集上的错误率比较高时，说明模型过拟合，方差比较高。这种情况可以通过降低模型复杂度、加大正则化系数、引入先验等方法来缓解。此外，还有一种有效降低方差的方法为集成模型，即通过多个高方差模型的平均来降低方差。

7.4 概率不等式

不等式能够对一些难以准确计算的量给出估计的上下界，它也常用于收敛定理的证明，有关收敛定理将在下一节具体讨论。

定理 7.4.1. (马尔可夫不等式) 令 X 为一非负随机变量，假设 $E(X)$ 存在，对任意 $t > 0$ 有

$$P(X > t) \leq \frac{E(X)}{t}.$$

证明. 因为 $X > 0$ ，所以

$$\begin{aligned} E(X) &= \int_0^\infty xf(x)dx = \int_0^t xf(x)dx + \int_t^\infty xf(x)dx \\ &\geq \int_t^\infty xf(x)dx \geq t \int_t^\infty f(x)dx = tP(X > t), \end{aligned}$$

所以

$$P(X > t) \leq \frac{E(X)}{t}.$$

□

定理 7.4.2. (切比雪夫不等式) 令 $\mu = E(X)$, $\sigma^2 = D(X)$, 则

$$P(|X - \mu| \geq t) \leq \frac{\sigma^2}{t^2}, \quad P(|Z| \geq k) \leq \frac{1}{k^2},$$

其中, $Z = (x - \mu)/\sigma$, 特别地, $P(|Z| > 2) \leq 1/4$, $P(|Z| > 3) \leq 1/9$ 。

证明. 利用马尔可夫不等式可得

$$P(|X - \mu| \geq t) = P(|X - \mu|^2 \geq t^2) \leq \frac{E(X - \mu)^2}{t^2} = \frac{\sigma^2}{t^2},$$

第二部分令 $t = k\sigma$ 即得。□

例 7.4.1. 假设检验是一种预测方法, 涉及 n 种检验情形, 以神经网络为例。如果预测错误则令 $X_i = 1$, 反之则令 $X_i = 0$ 。从而 $\bar{X}_n = \sum_{i=1}^n X_i/n$ 是观察到的误差率。每个 X_i 可认为服从未知均值 p 的伯努利分布。想求出未知的真实误差率 p 。从直觉上判断, \bar{X}_n 应与 p 非常接近, \bar{X}_n 不在 p 附近 ε 的范围内的概率为多少? 已知 $D(\bar{X}_n) = D(X_1)/n = p(1-p)/n$, 从而

$$P(|\bar{X}_n - p| > \varepsilon) \leq \frac{D(\bar{X}_n)}{\varepsilon^2} = \frac{p(1-p)}{n\varepsilon^2} \leq \frac{1}{4n\varepsilon^2}.$$

上式利用了不等式 $p(1-p) \leq 1/4$, 对于 $\varepsilon = 0.01$ 和 $n = 250000$, 所求的界为 0.01。也就是说, 当试验 25 万次时, 用 \bar{X}_n 估计 p 的误差不超过 0.01 的概率为 99%。

定理 7.4.3. (霍夫丁不等式) 令 Y_1, \dots, Y_n 为独立观察值, 满足 $E(Y_i) = 0$, 且 $a_i \leq Y_i \leq b_i$ 。令 $\varepsilon > 0$, 则对于任意 $t > 0$ 有

$$P\left(\frac{1}{n} \sum_{i=1}^n Y_i \geq \varepsilon\right) \leq \exp\left(-\frac{2n^2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right)$$

证明. 对任意 $t > 0$, 由马尔可夫不等式得

$$\begin{aligned} P\left(\frac{1}{n} \sum_{i=1}^n Y_i \geq \varepsilon\right) &= P\left(t \sum_{i=1}^n Y_i \geq tn\varepsilon\right) = P\left(e^{t \sum_{i=1}^n Y_i} \geq e^{tn\varepsilon}\right) \\ &\leq e^{-tn\varepsilon} E\left(e^{t \sum_{i=1}^n Y_i}\right) = e^{-tn\varepsilon} \prod_i E\left(e^{tY_i}\right). \end{aligned}$$

因为 $a_i \leq Y_i \leq b_i$, 可将 Y_i 写成 $Y_i = \alpha b_i + (1 - \alpha)a_i$, 其中, $\alpha = (Y_i - a_i) / (b_i - a_i)$, 所以根据 e^{ty} 的凸性得到

$$e^{tY_i} \leq \frac{Y_i - a_i}{b_i - a_i} e^{tb_i} + \frac{b_i - Y_i}{b_i - a_i} e^{ta_i},$$

两边取期望并利用 $E(Y_i) = 0$ 得

$$E(e^{tY_i}) \leq -\frac{a_i}{b_i - a_i} e^{tb_i} + \frac{b_i}{b_i - a_i} e^{ta_i} = e^{g(u)},$$

其中, $u = t(b_i - a_i)$, $g(u) = -\gamma u + \log(1 - \gamma + \gamma e^u)$, $\gamma = -a_i/(b_i - a_i)$, 注意到 $g(0) = g'(0) = 0$ 且对所有 $u > 0$, $g''(u) = \frac{\gamma e^u}{1 - \gamma + \gamma e^u} \left(1 - \frac{\gamma e^u}{1 - \gamma + \gamma e^u}\right) < 1/4$, 根据泰勒定理, 存在 $\xi \in (0, \mu)$ 满足

$$\begin{aligned} g(u) &= g(0) + ug'(0) + \frac{u^2}{2}g''(\xi) \\ &= \frac{u^2}{2}g''(\xi) \leq \frac{u^2}{8} = \frac{t^2(b_i - a_i)^2}{8}, \end{aligned}$$

因此

$$e^{tY_i} \leq e^{t^2(b_i - a_i)^2/8},$$

所以

$$\begin{aligned} P\left(\frac{1}{n} \sum_{i=1}^n Y_i \geq \varepsilon\right) &\leq e^{-tn\varepsilon} \prod_{i=1}^n e^{t^2(b_i - a_i)^2/8} = \exp\left(\left(\sum_{i=1}^n \frac{(b_i - a_i)^2}{8}\right)t^2 - n\varepsilon t\right) \\ &= \exp\left(\frac{1}{8} \left(\sum_{i=1}^n (b_i - a_i)^2 t^2 - 8n\varepsilon t + \left(\frac{4n\varepsilon}{\sqrt{\sum_{i=1}^n (b_i - a_i)^2}}\right)^2\right) - \frac{1}{8} \left(\frac{4n\varepsilon}{\sqrt{\sum_{i=1}^n (b_i - a_i)^2}}\right)^2\right) \\ &\leq \exp\left(-\frac{1}{8} \left(\frac{4n\varepsilon}{\sqrt{\sum_{i=1}^n (b_i - a_i)^2}}\right)^2\right) = \exp\left(-\frac{2n^2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right). \end{aligned}$$

□

定理 7.4.4. 令 X_1, \dots, X_n 服从参数为 p 的伯努利分布, 则对于任意 $\varepsilon > 0$ 有

$$P(|\bar{X}_n - p| > \varepsilon) \leq 2e^{-2n\varepsilon^2},$$

其中, $\bar{X}_n = n^{-1} \sum_{i=1}^n X_i$.

证明. 令 $Y_i = (1/n)(X_i - p)$, 则 $E(Y_i) = 0$, 令 $a = -p/n, b = (1 - p)/n$, 则 $a \leq Y_i \leq b$ 且 $(b - a)^2 = 1/n^2$, 根据霍夫丁不等式得

$$P(\bar{X}_n - p > \varepsilon) = P\left(\sum_i Y_i > \varepsilon\right) \leq e^{-2n\varepsilon^2}.$$

□

例 7.4.2. 令 X_1, \dots, X_n 服从参数为 p 的伯努利分布, 令 $n = 100, \varepsilon = 0.2$, 由切比雪夫不等式可得

$$P(|\bar{X}_n - p| > \varepsilon) \leq 0.0625,$$

由霍夫丁不等式得

$$P(|\bar{X}_n - p| > 0.2) \leq 2e^{-2(100)(0.2)^2} = 0.00067,$$

这比 0.0625 要小很多。

霍夫丁不等式提供了一种建立在参数 p 的二项式分布置信区间的简单方法。固定 $\alpha > 0$ 并令

$$\varepsilon_n = \sqrt{\frac{1}{2n} \log\left(\frac{2}{\alpha}\right)},$$

由霍夫丁不等式可知

$$P(|\bar{X}_n - p| > \varepsilon_n) \leq 2e^{-2n\varepsilon_n^2} = \alpha.$$

令 $C = (\bar{X}_n - \varepsilon_n, \bar{X}_n + \varepsilon_n)$, 则 $P(p \notin C) = P(|\bar{X}_n - p| > \varepsilon_n) \leq \alpha$ 。因此, $P(p \in C) \geq 1 - \alpha$, 也即随机区间 C 包括真实参数 p 的概率为 $1 - \alpha$; 称 C 为 $1 - \alpha$ 置信区间。

下面的不等式对于正态分布随机变量的概率范围确定非常有用。

定理 7.4.5. (Mill 不等式) 令 $Z \sim N(0, 1)$, 则:

$$P(|Z| > t) \leq \sqrt{\frac{2}{\pi}} e^{-t^2/2}$$

证明. 根据正态分布的定义有

$$\begin{aligned} t \cdot P\{Z > t\} &= t \int_t^{\infty} dF(x) \\ &\leq \int_t^{\infty} x dF(x) \\ &= \int_t^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} \\ &= \frac{\sigma}{\sqrt{2\pi}} \exp\left\{-\frac{t^2}{2\sigma^2}\right\} \end{aligned}$$

再根据正态分布的对称性可得

$$\begin{aligned} P\{Z > t\} &\leq \frac{\sigma}{t\sqrt{2\pi}} \exp\left\{-\frac{t^2}{2\sigma^2}\right\} \\ \Rightarrow P\{|Z| > t\} &\leq \sqrt{\frac{2}{\pi}} \frac{\sigma}{t} \exp\left\{-\frac{t^2}{2\sigma^2}\right\} \end{aligned}$$

当 $\sigma^2 = 1$ 时, 即得证。 □

有关期望的不等式

本节介绍有关期望的两个不等式。

定理 7.4.6. (柯西-施瓦兹不等式) 如果 X 和 Y 具有有限方差, 则

$$E|XY| \leq \sqrt{E(X^2)E(Y^2)}$$

定理 7.4.7. (詹森不等式) 如果 g 为凸函数, 则

$$Eg(X) \geq g(EX)$$

如果 g 为凹函数, 则

$$Eg(X) \leq g(EX)$$

证明. 令直线 $L(x) = a + bx$ 与 $g(x)$ 相切于点 $E(X)$, 因为 g 是凸函数, 它位于直线 $L(x)$ 的上方, 所以

$$Eg(X) \geq EL(X) = E(a + bX) = a + bE(X) = L(E(X)) = g(EX)$$

由詹森不等式可知 $E(X^2) \geq (EX)^2$; 如果 X 为正, 则 $E(1/X) \geq 1/E(X)$; 因为对数函数是凹函数, 所以 $E(\log X) \leq \log E(X)$ 。 \square

詹森不等式是一个重要的不等式。很多机器学习定理的证明中, 如 EM 算法收敛性的证明中使用了詹森不等式, 在第八章中有关熵的一些命题中也应用了詹森不等式。

概率不等式在统计机器学习中的应用: 泛化能力分析

定义 7.4.1. (泛化误差) 如果学到的模型是 \hat{f} , 那么这个模型对未知数据预测的误差即为泛化误差 (generalization error):

$$R_{exp}(\hat{f}) = E_p[L(Y, \hat{f}(X))] = \int_{X \times Y} L(y, \hat{f}(x))P(x, y)dxdy$$

泛化误差的概率上界称为泛化误差上界。具体来说就是通过比较两种学习方法的泛化误差上界的大小来比较它们的优劣。泛化误差上界通常具有以下性质:

- 它是样本容量的函数, 当样本容量增加时, 泛化上界趋于 0
- 它是假设空间容量的函数, 假设空间容量越大, 模型就越难学, 泛化误差上界就越大

考虑二分类问题, 已知训练数据集 $\mathbb{T} = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, N 是样本容量, \mathbb{T} 是从联合概率分布 $P(X, Y)$ 独立同分布产生的, $X \in \mathbb{R}^n, Y \in \{-1, +1\}$ 。假设空间是函数的有限集合 $\mathcal{F} = \{f_1, f_2, \dots, f_d\}$, d 是函数个数。设 f 是从 \mathcal{F} 中选取的函数。损失函数是 0-1 损失。关于 f 的期望风险和经验风险分别是

$$R(f) = E[L(Y, f(X))]$$

$$\hat{R}(f) = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i))$$

经验风险最小化函数是

$$f_N = \arg \min_{f \in \mathcal{F}} \hat{R}(f)$$

f_N 依赖训练数据集的样本容量 N 。我们更关心 f_N 的泛化能力

$$R(f_N) = E[L(Y, f_N(X))]$$

接下来我们讨论从有限集合 $\mathcal{F} = \{f_1, f_2, \dots, f_d\}$ 中任意选出的函数 f 的泛化误差上界。

定理 7.4.8. [泛化误差上界] 对于二分类问题, 当假设空间是有限个函数的集合 $\mathcal{F} = \{f_1, f_2, \dots, f_d\}$ 时, 对于任意函数 $f \in \mathcal{F}$, 至少以概率 $1 - \delta$, $0 \leq \delta \leq 1$, 以下不等式成立:

$$R(f) \leq \hat{R}(f) + \varepsilon(d, N, \delta) \quad (7.5)$$

其中

$$\varepsilon(d, N, \delta) = \sqrt{\frac{1}{2N} \left(\log d + \log \frac{1}{\delta} \right)} \quad (7.6)$$

不等式(7.5)左端 $R(f)$ 是泛化误差, 右端即泛化误差的上界。在泛化误差上界中, 第 1 项是训练误差, 训练误差越小, 泛化误差也越小。第 2 项 $\varepsilon(d, N, \delta)$ 是 N 的单调递减函数, 当 N 趋于无穷时趋于 0; 同时它也是 $\sqrt{\log d}$ 阶的函数, 假设空间包含的函数越多, 其值越大。

证明. 对任意函数 $f \in \mathcal{F}$, $\hat{R}(f)$ 是 N 个独立的随机变量 $L(Y, f(X))$ 的样本均值, $R(f)$ 是随机变量 $L(Y, f(X))$ 的期望值。如果损失函数取值于区间 $[0, 1]$, 即对所有 i , $[a_i, b_i] = [0, 1]$, 那么由霍夫丁不等式不难得知, 对 $\varepsilon > 0$, 以下不等式成立:

$$P(R(f) - \hat{R}(f) \geq \varepsilon) \leq \exp(-2N\varepsilon^2).$$

由于 $\mathcal{F} = \{f_1, f_2, \dots, f_d\}$ 是一个有限集合, 故

$$\begin{aligned} P(\exists f \in \mathcal{F} : R(f) - \hat{R}(f) \geq \varepsilon) &= P\left(\bigcup_{f \in \mathcal{F}} \{R(f) - \hat{R}(f) \geq \varepsilon\}\right) \\ &\leq \sum_{f \in \mathcal{F}} P(R(f) - \hat{R}(f) \geq \varepsilon) \\ &\leq d \exp(-2N\varepsilon^2). \end{aligned}$$

令 $\delta = d \exp(-2N\varepsilon^2)$, 则等价地, 对任意 $f \in \mathcal{F}$, 有 $P(R(f) - \hat{R}(f) \geq \varepsilon) \leq \delta$, 或者有 $P(R(f) < \hat{R}(f) + \varepsilon) \geq 1 - \delta$, 即至少以概率 $1 - \delta$ 有 $R(f) < \hat{R}(f) + \varepsilon$, 其中 ε 可从 $\delta = d \exp(-2N\varepsilon^2)$ 中反解, 即定理中的表达式(7.5)。 \square

7.5 大数定律与中心极限定理

7.5.1 引言

概率论最重要的一方面就是关注随机变量序列的趋势, 这部分内容称为大样本理论或极限理论或渐进理论。最基本的问题是: 关于随机变量序列 X_1, X_2, \dots 的极限性质可以作何论断? 因为统计与数据挖掘涉及大量数据, 自然而然地, 也会关心当收集到越来越多的数据时会发生什么。

在积分理论中, 如果对任意 $\varepsilon > 0$, $|x_n - x| < \varepsilon$ 对充分大的 n 都成立, 则称实数序列 x_n 收敛于极限 x 。在概率论中, 假设 X_1, X_2, \dots 为独立同分布随机序列, 服从 $N(0, 1)$ 分布, 因为所有变量具有相同的分布, 所以可以尝试着称 X_n “收敛于” $X \sim N(0, 1)$, 但这种描述并不十分精确, 因为对所有 n , $P(X_n = X) = 0$ (两个连续随机变量相同的概率为 0)。

还有另外一个例子，假设 $X_1, X_2, \dots \sim N(0, 1/n)$ ，从直觉上判断，当 n 很大时， X_n 集中在 0 附近，所以很希望称 X_n 收敛于 0，但是对所有 n ， $P(X_n = 0) = 0$ 。很明显，需要其他工具来讨论更严格意义上的随机变量的收敛。

本节将主要介绍两种思想

1. **大数定律**说明样本均值 $\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$ 依概率收敛于期望 $\mu = E(X_i)$ ，这意味着 \bar{X}_n 以很高的概率趋于 μ 。

2. **中心极限定理**说明 $\sqrt{n}(\bar{X}_n - \mu)$ 依分布收敛于正态分布，这也意味着对很大的 n ，样本均值渐进服从正态分布。

7.5.2 大数定律

依概率收敛和依分布收敛

定义 7.5.1. 令 X_1, X_2, \dots 为随机变量序列， X 为某一随机变量，用 F_n 表示 X_n 的 CDF，用 F 表示 X 的 CDF

1. 如果对任意 $\varepsilon > 0$ ，当 $n \rightarrow \infty$ 时有

$$P(|X_n - X| > \varepsilon) \rightarrow 0$$

则 X_n 依概率收敛于 X ，记为 $X_n \xrightarrow{P} X$ 。

2. 如果对所有的 F 的连续点 t ，有

$$\lim_{n \rightarrow \infty} F_n(t) = F(t)$$

则 X_n 依分布收敛于 X ，记为 $X_n \rightsquigarrow X$

当求 X 服从单点分布时，需要改变一下符号，如果 $P(X = c) = 1$ 且 $X_n \xrightarrow{P} X$ ，则记 $X_n \xrightarrow{P} c$ ，类似地，如果 $X_n \rightsquigarrow X$ ，则记 $X_n \rightsquigarrow c$ 。

这里再介绍另外一种形式的收敛，这种收敛对证明概率中的收敛很有用。

定义 7.5.2. 如果 $n \rightarrow \infty$ 时有

$$E(X_n - X)^2 \rightarrow 0,$$

则称 X_n 均方意义下收敛于 X （也称 L_2 收敛），记为 $X_n \xrightarrow{q.m} X$ 。

同上面类似，如果 X 服从在 c 点的单点分布，则用 $X_n \xrightarrow{q.m} c$ 代替 $X_n \xrightarrow{q.m} X$

例 7.5.1. 令 $X_n \sim N(0, 1/n)$ ，从直觉上判断，当 n 很大时， X_n 集中在 0 附近，所以就希望称 X_n 以概率收敛于 0，那么现在来看一下这是否正确。令 F 为在零点的单点分布的分布函数，注意到 $\sqrt{n}X_n \sim N(0, 1)$ ，令 Z 表示标准正态分布随机变量，对于 $t < 0$ ，因为 $\sqrt{nt} \rightarrow -\infty$ ，所以 $F_n(t) = P(X_n < t) = P(\sqrt{n}X_n < \sqrt{nt}) = P(Z < \sqrt{nt}) \rightarrow 0$ ；对于 $t > 0$ ，因为 $\sqrt{nt} \rightarrow \infty$ ，所以 $F_n(t) = P(X_n < t) = P(\sqrt{n}X_n < \sqrt{nt}) = P(Z < \sqrt{nt}) \rightarrow 1$ 。因此，对所有 $t \neq 0$ 有

$F_n(t) \rightarrow F(t)$, 所以 $X_n \rightsquigarrow 0$ 。注意 $F_n = 1/2 \neq F(1/2) = 1$, 所以在 $t = 0$ 处收敛不成立。这并不影响结果, 因为 $t = 0$ 不是 F 的连续点, 而分布收敛的定义仅需连续的点收敛即可。

现在再考察概率收敛, 对于任意 $\varepsilon > 0$, 使用马尔可夫不等式, 当 $n \rightarrow \infty$ 时有

$$P(|X_n| > \varepsilon) = P(|X_n|^2 > \varepsilon^2) \leq \frac{E(X_n^2)}{\varepsilon^2} = \frac{1/n}{\varepsilon^2} \rightarrow 0,$$

因此 $X_n \xrightarrow{P} 0$ 。

下面给出定理来说明各种收敛类型之间的关系

定理 7.5.1. (a) $X_n \xrightarrow{qm} X$ 意味着 $X_n \xrightarrow{P} X$,

(b) $X_n \xrightarrow{P} X$ 意味着 $X_n \rightsquigarrow X$,

(c) 如果 $X_n \rightsquigarrow X$ 且对于实数 c 有 $P(X = c) = 1$, 则 $X_n \xrightarrow{P} X$ 。

证明. (a) 假设 $X_n \xrightarrow{qm} X$ 成立, 对固定 $\varepsilon > 0$, 利用马尔可夫不等式

$$P(|X_n - X| > \varepsilon) = P(|X_n - X|^2 > \varepsilon^2) \leq \frac{E|X_n - X|^2}{\varepsilon^2} \rightarrow 0.$$

(b) 的证明有些复杂, 这里略去。

(c) 对固定 $\varepsilon > 0$

$$\begin{aligned} P(|X_n - c|) &= P(X_n < c - \varepsilon) + P(X_n > c + \varepsilon) \\ &\leq P(X_n < c - \varepsilon) + P(X_n > c + \varepsilon) \\ &\rightarrow F(c - \varepsilon) + 1 - F(c + \varepsilon) \\ &= 0 + 1 - 1 = 0. \end{aligned}$$

□

下面来说明逆命题并不成立。

依概率收敛不能推出均方意义下收敛 令 $U \sim Uniform(0, 1)$, $X_n = \sqrt{n}I_{0,1/n}(U)$, 则 $P(|X_n| > \varepsilon) = P(\sqrt{n}I_{0,1/n}(U)) > \varepsilon = P(0 \leq U < 1/n) = 1/n \rightarrow 0$ 。因此 $X_n \xrightarrow{P} 0$, 但是对所有 n 有 $E(X_n^2) = n \int_0^{1/n} du = 1$, 所以均方意义下不收敛。

依分布收敛不能推出依概率收敛 令 $X \sim N(0, 1)$, $X_n = -X$, 其中 $n = 1, 2, 3, \dots$; 因此, $X_n \sim N(0, 1)$, 即对所有 n , X_n 与 X 同分布, 所以对所有 x , $\lim_n F_n(x) = F(x)$, 也就是说 $X_n \rightsquigarrow X$ 但是 $P(|X_n - X| > \varepsilon) = P(|2X| > \varepsilon) = P(|X| > \frac{\varepsilon}{2}) \neq 0$, 也即 X_n 不依概率收敛于 X 。

弱大数定律

接下来的讨论议题可以被称为是概率论中最伟大的成果, 它就是大数定律。大数定律指出大量样本的均值近似于分布的均值, 例如, 无数次投掷硬币出现正面的概率趋近于 $1/2$, 下面对该定律简要描述。

令 X_1, X_2, \dots 为 IID 样本, 令 $\mu = E(X_1), \sigma^2 = D(X_1)$ 。则样本均值为 $\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$, $E(\bar{X}_n) = \mu, D(\bar{X}_n) = \sigma^2/n$

定理 7.5.2. (弱大数定律)(the weak law of large numbers(WLLN)) 若 X_1, X_2, \dots, X_n 为 IID 样本, 则 $\bar{X}_n \xrightarrow{P} \mu$ 。

WLLN 的含义: 当 n 逐渐变大时, X_n 的分布靠近 μ , 称 \bar{X}_n 为 μ 的一致估计(一致性), 在定理条件下, 当样本数目 N 无限增加时, 随机样本均值将几乎变成一个常量, 样本方差也依概率收敛于方差 σ^2 。

证明. 假设 $\sigma < \infty$, 该假设并不是必需的, 但有利于简化证明, 利用切比雪夫不等式得:

$$P(|X_n - \mu| > \varepsilon) \leq \frac{D(\bar{X}_n)}{\varepsilon^2} = \frac{\sigma^2}{n\varepsilon^2}$$

当 $n \rightarrow 0$ 时, 上式趋于 0. □

例 7.5.2. 假定抛一枚硬币, 出现正面的概率是 p , 令 X_i 表示每次结果, 因此 $p = P(X_i = 1) = E(X_i)$, 当抛 n 次后正面次数所占比例为 \bar{X}_n 。根据大数定律 X_n 以概率收敛于 p , 它意味着当 n 很大时, X_n 的分布会紧密围绕在 p 的附近。假设 $p = 1/2$, 需要多大的 n 才能使得 $P(0.4 \leq \bar{X}_n \leq 0.6) = 0.7$ 呢? 首先, $E(\bar{X}_n) = p = 1/2$ 且 $D(\bar{X}_n) = \sigma^2/n = p(1-p)/n = 1/(4n)$, 然后便有切比雪夫不等式

$$\begin{aligned} P(0.4 \leq \bar{X}_n \leq 0.6) &= P(|\bar{X}_n - \mu| \leq 0.1) \\ &= 1 - P(|\bar{X}_n - \mu| > 0.1) \\ &\geq 1 - \frac{1}{4n(0.1)^2} = 1 - \frac{25}{n} \end{aligned}$$

当 $n = 84$ 时就能保证上式大于 0.7.

7.5.3 中心极限定理

定义 7.5.3. (中心极限定理 (CLT)) 令 X_1, \dots, X_n 的均值为 μ , 方差为 σ^2 的 IID 序列, 令 $\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$, 则

$$Z_n = \frac{\bar{X}_n - \mu}{\sqrt{D(\bar{X}_n)}} = \frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma} \rightsquigarrow Z$$

其中 $Z \sim N(0, 1)$, 换句话说, 下式成立:

$$\lim_{n \rightarrow \infty} P(Z_n \leq z) = \Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx$$

含义: 有关 \bar{X}_n 概率陈述可以利用正态分布来近似, 注意这仅仅是概率陈述上的近似, 而并不是随机变量本身。

除了 $Z_n \rightsquigarrow N(0, 1)$ 外, 还有其他几个符号可以表示 Z_n 的分布收敛于正态分布, 他们表达的含义本质是一样的, 具体形式如下:

$$Z_n \approx N(0, 1),$$

$$\bar{X}_n \approx N(\mu, \frac{\sigma^2}{n}),$$

$$\bar{X}_n - \mu \approx (0, \frac{\sigma^2}{n}),$$

$$\sqrt{n}(\bar{X}_n - \mu) \approx N(0, \sigma^2),$$

$$\frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma} \approx N(0, 1),$$

例 7.5.3. 假设每个计算机程序产生误差的数量服从均值为 5 的泊松分布, 有 125 个程序, 令 X_1, \dots, X_{125} 分别表示程序中的误差数量, 求 $P(\bar{X}_n < 5.5)$ 。令 $\mu = E(X_1) = \lambda = 5, \sigma^2 = D(X_1) = \lambda = 5$ 则

$$P(\bar{X}_n < 5.5) = P\left(\frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma} < \frac{\sqrt{n}(5.5 - \mu)}{\sigma}\right) \approx P(Z < 2.5) = 0.9938$$

中心极限定理说明 $Z_n = \sqrt{n}(\bar{X}_n - \mu)/\sigma$ 服从 $N(0, 1)$, 然而 σ 值在大部分情况下是未知的, 后面将介绍用 X_1, \dots, X_n 的函数

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

去估计 σ^2 的方法。这又产生了另外一个问题: 如果用 S_n 去代替 σ , 中心极限定理还成立吗? 答案是肯定的。

定理 7.5.3. 假设跟 CLT 相同条件, 则

$$\frac{\sqrt{n}(\bar{X}_n - \mu)}{S_n} \rightsquigarrow N(0, 1)$$

读者或许要问, 正态近似的精度有多大呢? 答案将在 Berry - Esseen 定理中给出。

定理 7.5.4. (Berry-Esseen 定理) 假设 $E|X_1|^3 < \infty$, 则

$$\tilde{|P(Z_n \leq z) - \Phi(z)|} \leq \frac{33}{4} \frac{E|X_1 - \mu|^3}{\sqrt{n}\sigma^3}$$

中心极限定理也存在多元的情形

定理 7.5.5. (多元中心极限定理) 令 X_1, \dots, X_n 为 IID 随机向量, 其中

$$\begin{pmatrix} X_{1i} \\ X_{2i} \\ \vdots \\ X_{ki} \end{pmatrix}$$

其均值为

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix} = \begin{pmatrix} E(X_{1i}) \\ E(X_{2i}) \\ \vdots \\ E(X_{ki}) \end{pmatrix}$$

方差矩阵为 σ , 令

$$\bar{X} = \begin{pmatrix} \bar{X}_1 \\ \bar{X}_2 \\ \vdots \\ \bar{X}_k \end{pmatrix}$$

其中 $\bar{X}_j = \frac{1}{n} \sum_{i=1}^n X_{ji}$, 则

$$\sqrt{n}(\bar{X} - \mu) \rightsquigarrow N(0, \Sigma)$$

7.5.4 大数定律的推广及其在统计学习中的应用

我们前面介绍过在函数集 $Q(z, \alpha), \alpha \in \Lambda$ 上最小化风险泛函的问题

$$R(\alpha) = \int Q(z, \alpha) dF(z), \alpha \in \Lambda$$

其中, 分布函数 $F(z)$ 是未知的, 但给定了依据分布函数抽取的独立同分布数据 z_1, \dots, z_t 。为了求解上述问题, 我们提出了经验风险最小化原则。根据这一原则, 我们用最小化经验风险泛函

$$R_{emp}(\alpha) = \frac{1}{t} \sum_{i=1}^t Q(z_i, \alpha), \alpha \in \Lambda$$

来代替最小化泛函。设

$$Q(z, \alpha_t) = Q(z, \alpha(z_1, \dots, z_t))$$

为最小化泛函的一个函数。经验风险最小化理论的基本问题是描述经验风险最小化原则一致性的条件。下面我们给出一致性的经典定义。

大数定律指出 X_n 的分布会聚集在 u 附近, 这还不能描述 X_n 的概率性质, 为此还需要中心极限定理。

假设 X_1, X_2, \dots, X_n 为均值 μ , 方差 σ^2 的 IID 序列, 中心极限定理 (CLT) 指出 $\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$ 近似服从期望为 μ , 方差为 σ^2/n 的正态分布, 这一结论非常卓越, 因为只需要对 X_i 的分布的均值和方差进行要求, 没有其他别的条件。

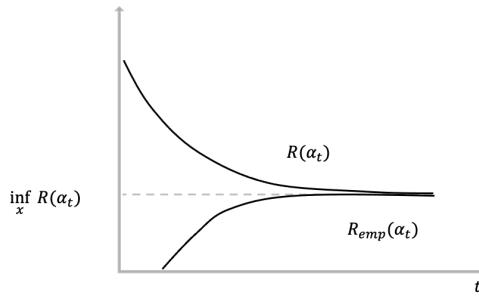


图 7.4: 如果期望风险 $R(\alpha_t)$ 和经验风险 $R_{emp}(\alpha_t)$ 都收敛于风险最小可能值 $\inf_{\alpha \in \Lambda} R(\alpha)$, 则学习过程是一致的

定义 7.5.4. 对于函数集 $Q(z, \alpha), \alpha \in \Lambda$ 和概率分布函数 $F(z)$, 如果下面两个序列依概率收敛于同一极限

$$R(\alpha_t) \xrightarrow[t \rightarrow \infty]{P} \inf R(\alpha), \quad (7.7)$$

$$R_{emp}(\alpha_t) \xrightarrow[t \rightarrow \infty]{P} \inf R(\alpha). \quad (7.8)$$

如图 7.4 所示, 我们称经验风险最小化原则(方法)是一致的。

换句话说, 如果经验风险最小化方法能够提供一个函数序列 $Q(z, \alpha_t), \alpha_t \in \Lambda$ 使得期望风险和经验风险依概率收敛于(对于给定的函数集)最小的可能风险值, 则经验风险最小化方法是一致的。方程(7.7)表明, 对于给定的函数集, 所得风险值序列收敛于最小的可能风险; 方程(7.8)表明, 经验风险序列的极限估计出风险的最小可能值。

统计学习理论的核心问题之一是找到经验风险最小化方法的一致性条件。而经验风险最小化的一致性分析在本质上是与两种经验过程的收敛性分析相联系的。

- 设概率分布函数 $F(z)$ 定义在空间 $z \in \mathbb{R}^n$ 上, $Q(z, \alpha), \alpha \in \Lambda$ 为一个(关于分布 $F(z)$ 的)可测函数集, 又设 z_1, \dots, z_t, \dots 为一个独立同分布的向量序列。考虑随机变量序列

$$\xi^l = \sup_{\alpha \in \Lambda} \left| \int Q(z, \alpha) dF(z) - \frac{1}{l} \sum_{i=1}^l Q(z_i, \alpha) \right|, l = 1, 2, \dots \quad (7.9)$$

我们将这一依赖于测度 $F(z)$ 和函数集 $Q(z, \alpha), \alpha \in \Lambda$ 的随机变量序列称为双边经验过程。

- 我们的问题是要描述一组条件, 在这组条件下上述经验过程依概率收敛于零。

换句话说, 我们的问题是描述一组条件, 使得对于任意正的 ε 下列收敛性成立:

表 7.1: 经典统计学体系和统计学习理论体系的结构

	经典统计学体系	统计学习理论体系
问题的表达	函数的参数估计	利用经验数据最小化期望风险
问题的解决方法	ML 法	ERM 或 SRM 方法
证明	参数估计的有效性	一致大数定律的存在性

定义 7.5.5. 假设测度 $F(z)$ 和函数集 $Q(z, \alpha), \alpha \in \Lambda$, 对于任意正的 ε 有

$$P \left\{ \sup_{\alpha \in \Lambda} \left| \int Q(z, \alpha) dF(z) - \frac{1}{l} \sum_{i=1}^l Q(z, \alpha) \right| > \varepsilon \right\} \xrightarrow[t \rightarrow \infty]{} 0,$$

我们称上述关系式为给定函数集上均值到数学期望的一致收敛性, 或者简称为一致收敛性。

定义 7.5.6. 与经验过程 ξ^l 一起, 我们考虑由随机值序列给出的单边经验过程

$$\xi_+^l = \sup_{\alpha \in \Lambda} \left(\int Q(z, \alpha) dF(z) - \frac{1}{l} \sum_{i=1}^l Q(z, \alpha) \right), l = 1, 2, \dots$$

我们称上述关系式为在给定的函数集上均值到数学期望的一致单边收敛性, 或者简称为一致单边收敛性。

值得注意的是如果函数集 $Q(z, \alpha), \alpha \in \Lambda$ 只包含一个元素, 则由式(7.9)定义的随机变量序列 ξ^l 永远依概率收敛于零。这一事实构成了统计学的主要定律, 即大数定律。随着 l 的增加, 均值序列收敛于随机变量的期望 (如果期望存在)。

将大数定律推广到函数集 $Q(z, \alpha), \alpha \in \Lambda$ 包含有限个元素的情况是容易的。与包含有限个元素情况相比, 对于包含无穷多个元素的函数集 $Q(z, \alpha), \alpha \in \Lambda$ 随机变量序列 ξ^l 不一定收敛于零, 那么就出现一个问题: 描述函数集 $Q(z, \alpha), \alpha \in \Lambda$ 和概率测度 $F(z)$ 的性质, 使得在这些性质下随机变量序列 ξ^l 依概率收敛于零。在这种情况下, 我们称大数定律在函数空间 (函数 $Q(z, \alpha), \alpha \in \Lambda$ 的空间) 中成立, 或者给定函数集上存在均值到期望的一致 (双边) 收敛性。因此, 函数空间中的大数定律 (均值到期望的一致双边收敛性) 的存在性问题可以看成经典大数定律的推广。应该注意的是, 在经典统计学中, 并没有考虑一致单边收敛性的存在性问题。经典统计学体系和统计学习理论体系的结构如表7.1所示。由于关键定理指出了分析 ERM 归纳原则一致性问题的方法, 一致单边收敛性的存在性问题变得重要起来了。一致收敛性意味着, 对于充分大的 l , 在给定函数集的所有函数上, 经验风险泛函一致地逼近于风险泛函。在前面, 我们已经证明, 当存在一致收敛性时, 最小化经验风险的函数给出了接近于最小可能风险的风险值。所以, 一致收敛性给出了经验风险最小化方法一致性的充分条件。在这种情况下, 就会出现一个新问题:

是否有可能认为一致收敛性的要求太强? 是否存在这样一种情况, 经验风险最小化方法是一致的, 但一致收敛性不成立?

事实上, 这样的情况是不可能出现的。可以证明一致单边收敛性不但构成了经验风险最小化方法一致性的充分条件, 而且构成了它的必要条件。

定理 7.5.6. 设存在常数 α 和 A , 使得对于函数集 $Q(z, \alpha), \alpha \in \Lambda$ 中的所有函数和给定的分布函数 $F(z)$, 有下列不等式成立:

$$\alpha \leq \int Q(z, \alpha) dF(z) \leq A, \alpha \in \Lambda,$$

则下面两种表述方式是等价的:

1. 对于给定分布函数 $F(z)$, 经验风险最小化方法在函数集 $Q(z, \alpha), \alpha \in \Lambda$ 上是严格一致的。
2. 对于给定分布函数 $F(z)$, 在函数集 $Q(z, \alpha), \alpha \in \Lambda$ 上出现均值到数学期望的一致单边收敛性的。

推论 7.5.1. 假设存在常数 α 和 A , 使得对于函数集 $Q(z, \alpha), \alpha \in \Lambda$ 中的所有函数和集合 \mathcal{P} 中的所有分布函数 $F(z)$, 有下列不等式成立:

$$\alpha \leq \int Q(z, \alpha) dF(z) \leq A, \alpha \in \Lambda,$$

则下面两种表述方式是等价的:

1. 对于集合 \mathcal{P} 中的任意分布函数 $F(z)$, 经验风险最小化方法在函数集 $Q(z, \alpha), \alpha \in \Lambda$ 上是严格一致的。
2. 对于集合 \mathcal{P} 中的任意分布函数 $F(z)$, 在函数集 $Q(z, \alpha), \alpha \in \Lambda$ 上出现均值到数学期望的一致单边收敛性的。

7.6 随机过程简介

日常生活中的很多例子包括股票的波动、语音信号、身高的变化都可以看作是随机过程。常见的和时间相关的随机过程模型包括伯努利过程、随机游走、马尔可夫过程等, 和空间相关的随机过程通常称为随机场。比如一张二维的图片, 每个像素点(变量)通过空间的位置进行索引, 这些像素就组成了一个随机过程。

定义 7.6.1. 一个随机过程 $\{X_t : t \in T\}$ 是一个随机变量集合, 通常写成 $X(t)$ 而不是 X_t , 其中变量 X_t 在一个被称作状态空间的集合 \mathcal{X} 里取值, 集合 T 被称作指标集, 通常可以视为时间。指标集可以是离散的 $T = \{0, 1, 2, \dots\}$ 或者连续的 $T = [0, \infty)$ 。

例 7.6.1. (IID 观测) 一个 IID 随机变量序列可以写作 $\{X_t : t \in T\}$, 其中 $T = \{1, 2, 3, \dots\}$ 。因此, 一个 IID 随机变量序列就是一个随机过程。

例 7.6.2. (天气) 令 $\mathcal{X} = \{\text{晴, 多云}\}$ 。一个典型的序列(依赖于你住在哪里)为
晴, 晴, 多云, 晴, 多云, 多云...

该过程具有一个离散的状态空间和一个离散的指标集。

例 7.6.3. (经验分布函数) 令 $X_1, \dots, X_n \sim F$ 其中 F 为 $[0, 1]$ 上的某个 CDF。令

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^n I(X_i \leq t)$$

为经验 CDF。对于任意固定值 t , $\hat{F}_n(t)$ 是一个随机变量。但是整个经验 CDF

$$\hat{F}_n(t) : t \in [0, 1]$$

为一个具有连续状态空间和连续指标集的随机过程。

7.6.1 马尔可夫链

离散时间的马尔可夫过程也称为马尔可夫链。如果一个马尔可夫链的条件概率

$$P(X_{t+1} = s | X_t = s') = m_{ss'}$$

只和状态 s 和 s' 相关, 和时间 t 无关, 则称为时间同质的马尔可夫链, 其中 $m_{ss'}$ 称为状态转移概率, 如果状态空间大小 K 是有限的, 状态转移概率可以用一个矩阵 $\mathbf{M} \in \mathbb{R}^{K \times K}$ 表示, 称为状态转移矩阵, 其中元素 m_{ij} 表示状态 s_i 转移到状态 s_j 的概率。

定义 7.6.2. 若

$$P(X_n = x | X_0, \dots, X_{n-1}) = P(X_n = x | X_{n-1})$$

对于所有的 n 和对所有的 $x \in \mathcal{X}$ 成立, 则称过程 $\{X_n : n \in T\}$ 是一个马尔可夫链。

马尔可夫链可以用下面的 DAG 来表示:

$$X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow \dots \rightarrow X_n \rightarrow \dots$$

每个变量具有单个母节点, 即前一个观测。马尔可夫链理论非常丰富且复杂, 主要涉及如下问题:

- 一个马尔可夫链何时“安定”为某种平稳态?
- 如何估计一个马尔可夫链的参数?
- 如何构造一个收敛到既定平稳分布的马尔可夫链和为什么想要那样做?

转移概率

一个马尔可夫链的重要的量为一个状态到另一个状态的概率。若 $P(X_{n+1} = j | X_n = i)$ 不随着时间而变化, 则一个马尔可夫链是时齐的。因此, 对于一个时齐马尔可夫链, $P(X_{n+1} | X_n = i) = P(X_1 = j | X_0 = i)$ 。下面只讨论时齐马尔可夫链

定义 7.6.3. 称

$$p_{ij} = P(X_{n+1} = j | X_n = i)$$

为转移概率, 第 (ij) 个元素为 p_{ij} 的矩阵 P 称作转移矩阵。

注意到 P 具有两个性质 (i) $p_{ij} \geq 0$ 且 (ii) $\sum_i p_{ij} = 1$ 。每行可以看作一个概率密度函数。

例 7.6.4. (带吸收壁的随机游动) 令 $\mathcal{X} = \{1, \dots, N\}$ 。假设你正站在这些点中的一个点上, 以 $P(\text{正面朝上}) = p$ 且 $P(\text{反面朝上}) = q = 1 - p$ 的概率投掷一枚硬币。若是正面朝上, 向右走一步, 若是反面朝上, 向左走一步。若你碰上某个终点, 停止。转移矩阵为

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ q & 0 & p & 0 & \dots & 0 & 0 \\ 0 & q & 0 & p & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & p \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix}$$

例 7.6.5. 假设状态空间为 $\mathcal{X} = \{\text{晴}, \text{多云}\}$, 则 X_1, X_2, \dots 表示一系列日子的天气。今天的天气还明显依赖于昨天的天气。它还可能依赖于前两天的天气, 但是作为第一个近似, 可以假设依赖性只倒退一天。在这种情况下, 天气为一个马尔可夫链且一个典型的转移矩阵为

$$P = \begin{pmatrix} 0.4 & 0.6 \\ 0.8 & 0.2 \end{pmatrix}$$

例如, 若今天是晴天, 则明天有 60% 的可能性是多云。

定义 7.6.4. 令

$$p_{ij}(n) = P(X_{m+n} = j | X_m = i)$$

为在 n 步中从状态 i 转移到状态 j 的概率。令 P_n 表示第 (i, j) 个元素为 $p_{ij}(n)$ 的元素。这些被称为 n 步转移概率。

定理 7.6.1. (Chapman-Kolmogorov 方程) n 步概率满足

$$p_{ij}(m+n) = \sum_k p_{ik}(m)p_{kj}(n)$$

仔细观察上述方程, 这只不过是矩阵乘法公式。因此证明了

$$\mathbf{P}_{m+n} = \mathbf{P}_m \mathbf{P}_n$$

由定义, $\mathbf{P}_1 = \mathbf{P}$ 由上述定理, $\mathbf{P}_2 = \mathbf{P}_{1+1} = \mathbf{P}_1 \mathbf{P}_1 = \mathbf{P} \mathbf{P} = \mathbf{P}^2$ 按该方法继续下去, 可以看到

$$\mathbf{P}_n = \mathbf{P}^n = \mathbf{P} \times \mathbf{P} \times \dots \times \mathbf{P}$$

令 $\mu_n = (\mu_n(1), \dots, \mu_n(N))$ 为行向量, 其中,

$$\mu_n(i) = P(X_n = i)$$

为该链在时刻 n 时处于状态 i 的边缘概率。特别地, μ_0 被称作初始分布。为了模拟一个马尔可夫链, 所要知道的就是 μ_0 和 \mathbf{P} 。模拟步骤应如下:

- 第一步产生 $X_0 \sim \mu_0$ ，因此 $P(X_0 = i) = \mu_0(i)$
- 第二步用 i 表示第一步的输出。产生 $X_1 \sim P$ 。换句话说， $P(X_1 = j | X_0 = i) = p_{ij}$
- 第三步假设第二步的输出为 j 。产生 $X_2 \sim P$ 。换句话说 $P(X_2 = k | X_1 = j) = p_{jk}$
- 继续下去。

理解 μ_n 的含义可能比较困难。想象模拟该链许多次，将所有的链在时刻 n 的输出收集起来。该直方图会近似于 μ_n

定理 7.6.2. 边缘概率可由下式给出

$$\mu_n = \mu_0 P^n$$

状态分类

定义 7.6.5. i 到达 j （或 j 从 i 是可达的）若对于某个 n 有 $p_{ij}(n) > 0$ ，且记作 $i \rightarrow j$ 。若 $i \rightarrow j$ 且 $j \rightarrow i$ 则记作 $i \leftrightarrow j$ ，并且称 i 和 j 互通。

定理 7.6.3. 互通关系满足下面的性质

- $i \leftrightarrow i$
- 若 $i \leftrightarrow j$ 则 $j \leftrightarrow i$
- 若 $i \leftrightarrow j$ 且 $j \leftrightarrow k$ 则 $i \leftrightarrow k$
- 状态集 \mathcal{X} 可以写作不相交的类的并 $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2 \cup \dots$ ，其中，两个状态之间互通当且仅当它们在同一个类中。

注：按互通关系是等价关系，可以把状态空间 \mathcal{X} 划分为若干个不相交的集合（或者说等价类），并称之为状态类。若两个状态互通，则这两个状态属于同一类。任意两个类或不相交或者相同。

定义 7.6.6. 设 C 为状态空间 \mathcal{X} 的一个子集，若对任意的 $i \in C$ 和 $j \notin C$ 有 $p_{ij} = 0$ 则称 C 为闭集。

注：若 C 为闭集，则表示自 C 内任意状态 i 出发，始终不能到达 C 以外的任何状态 j 。显然，整个状态空间构成一个闭集。

定义 7.6.7. 只含有单个状态的闭集称作为吸收态。

注：若状态空间含有吸收态，那么这个吸收态构成一个最小的闭集。

定义 7.6.8. 若除整个状态空间 \mathcal{X} 以外没有其它的闭集，则称此马氏链是不可约的。

如果闭集 C 的状态都是互通的，则称闭集 C 是不可约的。

例 7.6.6. 令 $\mathcal{X} = \{1, 2, 3, 4\}$ 且

$$P = \begin{pmatrix} 1/2 & 2/3 & 0 & 0 \\ 2/3 & 1/3 & 0 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

类为 $\{1, 2\}, \{3\}, \{4\}$ 。状态 4 为一个吸收态。

假设从状态 i 开始一个链。该链会返回状态 i 吗？若如此，称状态 i 为持久的或常返的。

定义 7.6.9. 状态 i 为持久的或常返的，若

$$P(X_n = i \text{ 对于某个 } n \geq 1 | X_0 = i) = 1$$

否则，状态 i 为瞬过的。

定理 7.6.4. 一个状态 i 为常返的当且仅当

$$\sum_n p_{ii}(n) = \infty$$

一个状态为瞬过的当且仅当

$$\sum_n p_{ii}(n) < \infty$$

定理 7.6.5. 关于常返性的事实

- 若状态 i 为常返的且 $i \leftrightarrow j$ ，则 j 是常返的。
- 若状态 i 为瞬过的且 $i \leftrightarrow j$ ，则 j 是瞬过的。
- 一个有限马尔可夫链必然至少有一个常返态。
- 一个有限的不可约马尔可夫链的状态都是常返的。

定理 7.6.6. (分解定理) 状态空间 \mathcal{X} 可以写成不相交集的并

$$\mathcal{X} = \mathcal{X}_T \cup \mathcal{X}_1 \cup \mathcal{X}_2 \dots$$

其中 \mathcal{X}_T 为瞬过态，且每个 \mathcal{X}_i 为一个闭的，不可约的常返态集。

马尔可夫链的收敛性

为了讨论马尔可夫链的收敛性，需要一些定义。

定义 7.6.10. 假设 $X_0 = i$ 定义常返时间

$$T_{ij} = \min\{n > 0 : X_n = j\}$$

假设 X_n 可返回状态 i ，否则定义 $T_{ij} = \infty$ 一个常返态 i 的平均常返时间为

$$m_i = E(T_{ii}) = \sum_n n f_{ii}(n)$$

其中

$$f_{ij}(n) = P(X_1 \neq j, X_2 \neq j, \dots, X_{n-1} \neq j, X_n \neq j | X_0 = i)$$

若 $m_i = \infty$ 称一个常返态是零的，否则称之为非零的或正的。

定理 7.6.7. 若一个状态是零的且是常返的, 则 $p_{ii}^n \rightarrow 0$

定理 7.6.8. 在一个有限的状态的马尔可夫链里, 所有的常返态都是正的。

例 7.6.7. 考虑具有三个状态的马尔可夫链, 其转移矩阵为

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

假设该链的初始状态为 I , 那么将在时刻 $3, 6, 9, \dots$ 到达状态 3 , 这是一个周期链的例子。

定义 7.6.11. 若 $p_{ii}(n) = 0$, 其中 n 不能被 d 整除且 d 是满足该性质的最大整数, 则称状态 i 的周期为 d 。因此, $d = \gcd\{n : p_{ii}(n) > 0\}$, 其中 \gcd 的意思为“最大公约数”。若 $d(i) > 1$, 则称该链的状态 i 是周期的。若 $d(i) = 1$ 是非周期的。周期为 I 的一个状态被称作非周期的。

定理 7.6.9. 若状态 i 具有周期 d 且 $i \leftrightarrow j$, 则 j 也具有周期 d 。

定义 7.6.12. 若一个状态是常返的, 非零的且非周期的, 则称这个状态 i 是遍历的。若其所有状态是遍历的, 则称这一个链是遍历的。

令 $\pi = (\pi_i : i \in \mathcal{X})$ 为一个非负数向量, 且分量和为 1。因此 π 可以视为一个概率密度函数。

定义 7.6.13. 若 $\pi = \pi \mathbf{P}$, 则称 π 是一个平稳(或不变)分布。

这里给出直观的思路。 X_0 服从 π 分布并且假设 π 是一个平稳分布。现在根据马尔可夫链的转移概率来抽取 X_1 , 得到 X_1 的分布为 $\mu_1 = \mu_0 \mathbf{P} = \pi \mathbf{P} = \pi$ 。 X_2 的分布为 $\pi \mathbf{P}^2 = (\pi \mathbf{P}) \mathbf{P} = \pi \mathbf{P} = \pi$, 如此继续下去, 会看到 X_n 的分布为 $\pi \mathbf{P}^n = \pi$ 。换句话说, 若该链在任何时候都具有分布 π , 则它将持续具有分布 π 。

定义 7.6.14. 称一个链具体极限分布 π , 若 $\mathbf{P}^n \rightarrow (\pi, \pi, \dots, \pi)^T$ 对于某个 π , 即 $\pi_j = \lim_{n \rightarrow \infty} P_{ij}^n$ 存在与 i 是独立的。

下面给出收敛性的主要定理, 该定理表明一个遍历链收敛到它的平稳分布。而且, 样本均值收敛到它的平稳分布下的理论期望。

定理 7.6.10. 一个不可约, 遍历的马尔可夫链具有唯一的平稳分布 π 。极限分布存在且等于 π 。若 g 是任意一个有界函数, 则以概率 1

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N g(X_n) \rightarrow E_{\pi}(g) = \sum_j g(j) \pi_j$$

定义 7.6.15. 若

$$\pi_i p_{ij} = \pi_j p_{ji}$$

则 π 满足细致平衡

细致平衡保证了 π 是一个平稳分布。

定理 7.6.11. 若 π 满足细致平衡，则 π 是一个平稳分布。

注意仅仅因为一个链有一个平稳分布并不意味着它收敛。

例 7.6.8. 令

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

令 $\pi = 1/3, 1/3, 1/3$ ，则 $\pi P = \pi$ 所以 π 是一个平稳分布。若该链是从分布 π 开始的，它将停留在该分布里。想象模拟许多链且在每个时刻 n 去验证其边缘分布。它将永远为均匀分布 π 但是该链没有极限。它将继续循环下去。

例 7.6.9. 令 $\mathcal{X} = \{1, 2, 3, 4, 5, 6\}$ ，令

$$P = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 & 0 \\ 1/4 & 3/4 & 0 & 0 & 0 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 \\ 1/4 & 0 & 1/4 & 1/4 & 0 & 1/4 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \end{pmatrix}$$

则 $C_1 = \{1, 2\}$ 且 $C_2 = \{5, 6\}$ 是不可约的闭集。状态 3 和状态 4 是暂留的因为路径为 $3 \rightarrow 4 \rightarrow 6$ 且一旦到达状态 6 就不能返回 3 或 4。因为 $p_{ii}(1) > 0$ ，所有的状态都是非周期的，总之 3 和 4 是暂留的，而 1, 2, 5 和 6 是遍历的。

这里简述一种叫马尔可夫链蒙特卡罗 (MCMC) 的模拟方法的基本思想。

例 7.6.10. (马尔可夫链蒙特卡罗)

令 $f(x)$ 为实轴上的一个概率密度函数且假设 $f(x) = cg(x)$ 其中 $g(x)$ 是一个已知函数且 $c > 0$ 是未知的。原则上可以计算出 c ，因为 $\int f(x)dx = 1$ 意味着 $c = 1/\int g(x)dx$ 。然而，计算该积分可能行不通，而且 c 对下面的计算也没有必要。令 X_0 为一个任意开始值。给定 X_0, \dots, X_i 按下面的方法产生 X_{i+1} 。首先，选取 $W \sim N(X_i, b^2)$ 其中 $b > 0$ 是一个固定的常数。令

$$r = \min \left\{ \frac{g(W)}{g(X_i)}, 1 \right\}$$

选取 $U \sim U(0, 1)$ 且设定

$$X_{i+1} = \begin{cases} W, & U < r \\ X_i & U \geq r \end{cases}$$

在弱条件下， X_0, X_1, \dots 是以一个遍历的马尔可夫链且平稳分布为 f 。因此，可以将选取出来的变量看作来自 f 的一个样本。

7.6.2 高斯过程

高斯过程也是一种应用广泛的随机过程模型，常用于统计建模中。

定义 7.6.16. 假设有一组连续随机变量 X_0, X_1, \dots, X_T ，如果由这组随机变量构成的任一有限集合

$$X_{t_1, \dots, t_N} = [X_{t_1}, \dots, X_{t_N}]^\top, \quad 1 \leq N \leq T$$

都服从一个多元正态分布，那么这组随机变量为一个高斯随机过程，高斯过程也可以定义为：如果 X_{t_1, \dots, t_N} 的任一线性组合都服从一元正态分布，那么这组随机变量为一个高斯过程。

高斯过程回归

高斯过程回归是利用高斯过程来对一个函数分布进行建模。和机器学习中参数化建模（比如贝叶斯线性回归）相比，高斯过程是一种非参数模型，可以拟合一个黑盒函数，并给出拟合结果的置信度。

假设一个未知函数 $f(\mathbf{x})$ 服从高斯函数，且为平滑函数，如果两个样本 $\mathbf{x}_1, \mathbf{x}_2$ 比较接近，那么对应的 $f(\mathbf{x}_1), f(\mathbf{x}_2)$ 也比较接近，假设从函数 $f(\mathbf{x})$ 中采样有限个样本 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ ，这 N 个点服从一个多元正态分布，

$$[f(\mathbf{x}_1), f(\mathbf{x}_2), \dots, f(\mathbf{x}_N)]^\top \sim N(\boldsymbol{\mu}(\mathbf{X}), \mathbf{K}(\mathbf{X}, \mathbf{X}))$$

其中 $\boldsymbol{\mu}(\mathbf{X}) = [\boldsymbol{\mu}(\mathbf{x}_1), \boldsymbol{\mu}(\mathbf{x}_2), \dots, \boldsymbol{\mu}(\mathbf{x}_N)]^\top$ 是均值向量， $\mathbf{K}(\mathbf{X}, \mathbf{X}) = [k(\mathbf{x}_i, \mathbf{x}_j)]_{N \times N}$ 是协方差矩阵， $k(\mathbf{x}_i, \mathbf{x}_j)$ 为核函数，可以衡量两个样本的相似度。

在高斯过程回归中，一个常用的核函数是平方指数函数

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2l^2}\right)$$

其中 l 为超参数，当 \mathbf{x}_i 和 \mathbf{x}_j 越接近，其核函数的值越大，表明 $f(\mathbf{x}_i)$ 和 $f(\mathbf{x}_j)$ 越相关。

假设 $f(\mathbf{x})$ 的一组带噪声的观测值为 $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$ ，其中 $y_n \sim N(f(\mathbf{x}_n), \sigma^2)$ 为 $f(\mathbf{x}_n)$ 的观测值，服从正态分布， σ 为噪声方差。

对于一个新的样本点 \mathbf{x}^* ，我们希望预测 $f(\mathbf{x}^*)$ 观测值 y^* 。令向量 $\mathbf{y} = [y_1, y_2, \dots, y_N]^\top$ 为已有的观测值，根据高斯过程的假设， $[\mathbf{y}; y^*]$ 满足

$$\begin{bmatrix} \mathbf{y} \\ y^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}(\mathbf{X}) \\ \boldsymbol{\mu}(\mathbf{x}^*) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I} & \mathbf{K}(\mathbf{x}^*, \mathbf{X})^\top \\ \mathbf{K}(\mathbf{x}^*, \mathbf{X}) & k(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix}\right)$$

其中 $\mathbf{K}(\mathbf{x}^*, \mathbf{X}) = [k(\mathbf{x}^*, \mathbf{x}_1), \dots, k(\mathbf{x}^*, \mathbf{x}_n)]$

根据上面的联合分布， y^* 的后验分布为

$$p(y^* | \mathbf{X}, \mathbf{y}) = \mathcal{N}(\hat{\boldsymbol{\mu}}, \hat{\sigma}^2)$$

其中均值 $\hat{\boldsymbol{\mu}}$ 和方差 $\hat{\sigma}^2$ 为

$$\hat{\boldsymbol{\mu}} = \mathbf{K}(\mathbf{x}^*, \mathbf{X}) (\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I})^{-1} (\mathbf{y} - \boldsymbol{\mu}(\mathbf{X})) + \boldsymbol{\mu}(\mathbf{x}^*)$$

$$\hat{\sigma}^2 = k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{K}(\mathbf{x}^*, \mathbf{X}) (\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I})^{-1} \mathbf{K}(\mathbf{x}^*, \mathbf{X})^\top$$

从公式可以看出, 均值函数 $\mu(x)$ 可以近似地互相抵消, 在实际应用中, 一般假设 $\mu(x) = 0$, 均值 $\hat{\mu}$ 可以将简化为

$$\hat{\mu} = \mathbf{K}(\mathbf{x}^*, \mathbf{X}) (\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I})^{-1} \mathbf{y}$$

高斯过程回归可以认为是一种有效地贝叶斯优化方法, 广泛地应用于机器学习中。

7.7 阅读材料

机器学习中的概率模型 Bishop (2006); Murphy (2012) 为用户提供了一种以原则方式捕捉数据和预测模型的不确定性的方法。Ghahramani (2015) 简要回顾了机器学习中的概率模型。本章有时候会出现这种情况, Grinstead 和 Snell (1997) 提供了一种适合自学的更轻松的表达方式。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) 提出了更多的软件工程方法。

给定概率模型, 我们可能足够幸运能够分析地计算感兴趣的参数。然而, 一般而言, 分析方法很少用, 而使用的是诸如采样 (Brooks 等人, 2011) 和变分推理 (Blei 等人, 2017) 之类的计算方法。具有讽刺意味的是, 最近对神经网络兴趣的激增导致了对概率模型的更广泛的认识。例如, 流量标准化的想法 (Rezende 和 Mohamed, 2015) 依赖于变量来改变随机变量。应用于神经网络的变分推断方法的概述在 Goodfellow 等人 (2016) 的第 16 至 20 章中描述。

对概率论细节感兴趣的更多读者有很多选择 (Jacod 和 Protter, 2004; Jaynes, 2003; Mackay, 2003), 包括一些非常技术性的讨论 (Dudley, 2002; Shirayev, 1984; Lehmann 和 Casella, 1998; Bickel 和 Doksum, 2006)。我们通过对测量理论问题进行掩饰来回避大部分困难 (Billingsley, 1995; Pollard, 2002), 并不进行构造地假设我们有实数, 以及定义实数集的方法以及它们的适当频率发生。由于机器学习允许我们在移动复杂类型的数据上模拟移动复杂的分布, 因此概率机器学习模型的开发者必须理解这些更多的技术方面。具有概率建模焦点的机器学习书包括 Mackay (2003); Bishop (2006); Murphy (2012); Barber (2012 年); Rasmussen 和 Williams (2006 年)。

习题

习题 7.1. (1) 设 A, B, C 是三个事件, 且 $P(A) = P(B) = P(C) = 1/4, P(AB) = P(BC) = 0, P(AC) = 1/8$, 求 A, B, C 至少有一个发生的概率。

(2) 已知 $P(A) = 1/2, P(B) = 1/3, P(C) = 1/5, P(AB) = 1/10, P(AC) = 1/15, P(BC) = 1/20, P(ABC) = 1/30$, 求 $A \cup B, \overline{AB}, A \cup B \cup C, \overline{ABC}, \overline{ABC}, \overline{AB} \cup C$ 的概率

习题 7.2. (1) 已知 $P(\overline{A}) = 0.3, P(B) = 0.4, P(A\overline{B}) = 0.5$, 求条件概率 $P(B|A \cup \overline{B})$

(2) 已知 $P(A) = 1/4, P(B|A) = 1/3, P(A|B) = 1/2$, 求 $P(A \cup B)$

习题 7.3. 设事件 A, B 的概率均大于零, 说明以下的叙述 (1) 必然对。 (2) 必然错。 (3) 可能对. 并说明理由.

- (1) 若 A 与 B 互不相容, 则它们相互独立.
- (2) 若 A 与 B 相互独立, 则它们互不相容.
- (3) $P(A) = P(B) = 0.6$, 且 A, B 互不相容.
- (4) $P(A) = P(B) = 0.6$, 且 A, B 相互独立.

习题 7.4. (1) 袋中装有 5 只球, 编号为 1, 2, 3, 4, 5。在袋中同时取 3 只, 以 X 表示去除的 3 只中的最大号码, 写出随机变量 X 的分布律。

- (2) 将一颗骰子抛掷两次, 以 X 表示两次中得到的小的点数, 试求 X 的分布律.

习题 7.5. 设随机变量 X 的分布函数为

$$F_x(x) = \begin{cases} 0, & x < 1 \\ \ln x, & 1 \leq x < e \\ 1, & x \geq e \end{cases}$$

- (1) 求 $P\{X < 2\}, P\{0 < X \leq 3\}, P\{2 < X < 5\}$
- (2) 求概率密度 $f_x(x)$

习题 7.6. 设随机变量 X 的概率密度为

$$f(x) = \begin{cases} \frac{2x}{\pi^2}, & x > 0 \\ 0, & \text{其他} \end{cases}$$

求 $Y = \sin X$ 的概率密度。

习题 7.7. 设 $X \sim N(0, 1)$.

- (1) 求 $Y = e^x$ 的概率密度。
- (2) 求 $Y = 2X^2 + 1$ 的概率密度。
- (3) 求 $Y = |X|$ 的概率密度。

习题 7.8. 设二维随机变量 (X, Y) 的概率密度为

$$f(x, y) = \begin{cases} e^{-y}, & 0 < x < y \\ 0, & \text{其他} \end{cases}$$

- (1) 确定常数 c 。
- (2) 求边缘概率密度。

习题 7.9. 设二维随机变量 (X, Y) 的概率密度为

$$f(x, y) = \begin{cases} 1, & |y| < x, 0 < x < 1 \\ 0, & \text{其他} \end{cases}$$

求条件概率密度 $f_{Y|X}(y|x), f_{X|Y}(x|y)$

习题 7.10. 设随机变量 X, Y 的联合密度为

$$f(x, y) = \begin{cases} \frac{1}{y} e^{-(y+x)/y} & x > 0, y > 0 \\ 0 & \text{其他} \end{cases}$$

求 $E(X), E(Y), E(XY)$

习题 7.11. 计算样在进行加法时, 将每个加数舍入最靠近它的整数, 设所有舍入误差相互独立且在 $(-0.5, 0.5)$ 上服从均匀分布.

- (1) 将 1500 个数相加, 问误差总和的绝对值超过 15 的概率是多少?
- (2) 最多可有几个数相加使得误差总和的绝对值小于 10 的概率不小于 0.90?

习题 7.12. 设齐次马氏链的一步转移概率矩阵为

$$P = \begin{bmatrix} q & p & 0 \\ q & 0 & p \\ 0 & q & p \end{bmatrix}, \quad q = 1 - p, 0 < p < 1$$

试证明此链具有遍历性, 并求其平稳分布.

习题 7.13. 设马氏链的一步转移概率矩阵为

$$P = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

试证此链不是遍历的.

第八章 信息论基础

信息论创始人香农对信息的定义为信息是对事物运动状态或存在方式的不确定性的表示。信息论解答了通信理论中的两个基问题：数据压缩的临界（熵）值和通信传输速率的临界值（信道容量）。即香农证明了只要通信速率低于信道容量，那么，在理论上存在一种方法可使信息的输出能够以任意小的差错概率通过信道传输。因此认为信息论是通信的基础理论之一。信息论不仅在通信领域具有深远影响，事实上它在很多学科，如统计物理，计算机科学，统计推断，概率论和统计等学科都具有奠基性的贡献。

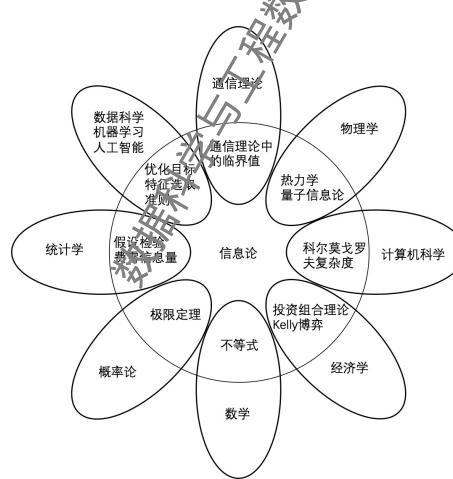


图 8.1: 信息论与其他学科的关系

数据承载着信息，有些信息可以直接从数据中获得，有些信息需要对数据进行一定的运算处理后才能获得。比如一张表示猫的图片，人类可以从图片上看出这是一只猫，计算机却需要将图片经过模型运算后才能知道图片上表示的内容是猫，从而获得图片表示内容的信息。我们把发出数据的一端称作信源，从信源发出一张图片，这个图片是什么样的我们是不知道的。因此，图片数据本身就包含着信息。对于特定一张图片，在将图片数据经过分类模型处理前，计算机是不知道图片表示的物体类别的。当图片经过模型处理，确定了图片所属的类别，也就消除了

不确定性，从而获得图片所属类别这一信息。Watanabe 认为学习就是一个熵减的过程。所以我们可以将机器学习或深度模型中的编码解码模型看做是一个通信系统，输入为信源，这样信息论中的一些度量也可以作为学习算法的度量。

具体来说，机器学习或深度学习除了可以将经验风险、经验误差、经验损失的经验函数作为一类学习准则外，还可以使用基于信息论中熵的函数如信息熵、交叉熵、相对熵、互信息作为学习准则。信息论指导机器学习和深度学习中的很多算法的设计和改进，比如用交叉熵损失作为损失函数，利用互信息进行特征选择等。David MacKay 认为信息论和机器学习就是一枚硬币的两面。以信息理论为基础的机器学习在理论上更具有优势。

在概率论和统计学学科中，信息论中的基本量，熵、相对熵、互信息，定义成概率分布的泛函。它们中的任何一个量都能刻画随机变量长序列的行为特征，使得我们能够估计稀有事件的概率（大偏差理论），并且在假设检验中找到最佳的误差指数。这推动了概率论与统计学学科的发展。同时概率是研究不确定性的工具，我们也可以基于概率对信息量进行度量。下图是本章需要讲解的关于信息论基础知识的导图。

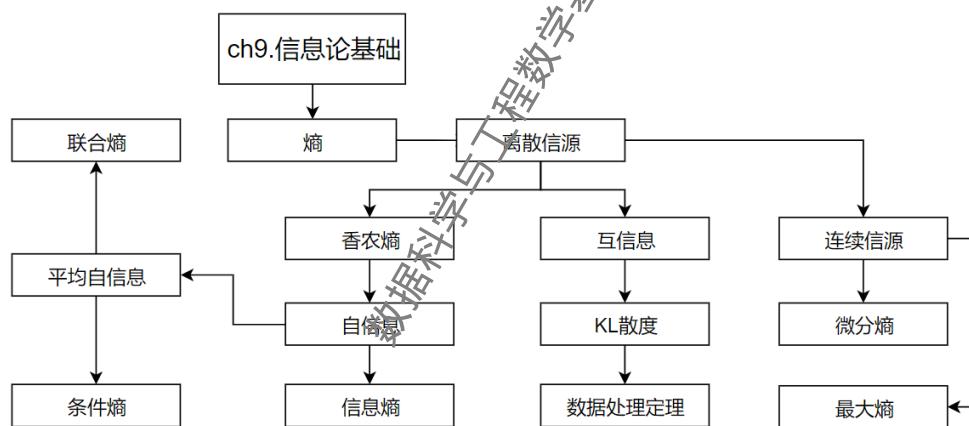


图 8.2: 本章导图

8.1 熵、相对熵和互信息

信息论主要研究的是对一个信号包含信息的多少进行量化。这种量化应该符合这样的直觉即事件的发生概率应该能够反映事件所包含的信息量。不难理解，小概率事件，不确定性大，一旦出现使人感到意外，因此产生的信息量就大，特别是几乎不可能出现的事件一旦出现，必然产生极大的信息量，而大概率事件，是预料之中的事件，不确定性小，即使发生，也没什么信息量，特别是概率为 1 的确定事件发生以后，不会给人以任何信息量。

信息是个相当宽泛的概念，很难用一个简单的定义将其完全准确地把握，然而对于任何一

一个概率分布可以定义一个称为熵的量。它具有许多特性符合度量信息的直观要求，这个概念可以推广到互信息，互信息是一种测度，用来度量一个随机变量包含另一个随机变量的信息量。照此理解，熵恰好变成了度量其本身信息量的度量，称作自信息。相对熵是个更广泛的量，它是刻画两个概率分布之间的距离的一种度量，而互信息又是它的特殊情形，以上所有的这些量密切相关，存在许多简单的共性。

8.1.1 自信息

信源发出的消息(事件)具有不确定性。不确定性小的，即发生概率大的消息含的信息量小。不确定性大的，即发生概率小的消息含的信息量大。如果两个消息是毫无关联的，即发生的概率是相互独立的，那么通过这两个消息获得的信息量应当是两个消息各自信息量的和。

因此随机事件的自信息量 $I(x_i)$ 是该事件发生概率 $P(x_i)$ 的函数，并且 $I(x_i)$ 应该满足以下公理化条件：

(1) $I(x_i)$ 是 $P(x_i)$ 的严格递减函数。当 $P(x_1) < P(x_2)$ 时， $I(x_1) > I(x_2)$ ，概率越小，事件发生的不确定性越大，事件发生以后所包含的自信息量越大。

(2) 极限情况下，当 $P(x_i) = 0$ 时， $I(x_i) \rightarrow \infty$ ；当 $P(x_i) = 1$ 时， $I(x_i) = 0$ 。

(3) 从直观概念上讲，由两个相对独立的不同的消息所提供的信息量应等于它们分别提供的信息量之和，即自信息量满足可加性。

可以证明，满足以上公理化条件的函数形式是对数形式。

定义 8.1.1. 随机事件的自信息量定义为该事件发生概率的对数的负值。设事件 x_i 的概率为 $P(x_i)$ ，则它的自信息量定义为

$$I(x_i) = -\log p(x_i) = \log \frac{1}{p(x_i)}$$

$I(x_i)$ 代表两种含义：在事件 x_i 发生以前，等于事件 x_i 发生的不确定性的大小；在事件 x_i 发生以后，表示事件 x_i 所含有或所能提供的信息量。

自信息量的单位与所用对数的底有关。

(1) 通常取对数的底为 2，信息量的单位为比特 (bit, binary unit)。当 $p(x_i) = 1/2$ 时， $I(x_i) = 1\text{bit}$ ，即概率等于 $1/2$ 的事件具有 1 bit 的自信息量。例如，一枚均匀硬币的任何一种抛掷结果均含有 1 bit 的信息量。比特是信息论中最常用的信息量单位，当取对数的底为 2 时，2 常省略。注意：计算机术语中 bit 是位的单位 (bit, binary digit)，与信息量单位不同，但有联系，1 位的二进制数字最大能提供 1 bit 的信息量。

(2) 若取自然对数 (以 e 为底)，自信息量的单位为奈特 (nat, natural unit)。理论推导中或用于连续信源时用以 e 为底的对数比较方便。

$$1\text{ nat} = \log_2 e \text{ bit} = 1.443 \text{ bit}$$

(3) 工程上用以 10 为底较方便. 若以 10 为对数底, 则自信息量的单位为哈特莱 (Hartley), 用来纪念哈特莱首先提出用对数来度量信息.

$$1 \text{ Hartley} = \log_2 10 \text{ bit} = 3.322 \text{ bit}$$

(4) 如果取以 r 为底的对数 ($r > 1$), 则 $I(xi) = -\log_r p(xi)$, r 进制单位

$$1 r \text{ 进制单位} = \log_2 r \text{ bit}$$

例 8.1.1. (1) 英文字母中 “ a ” 出现的概率为 0.064, “ c ” 出现的概率为 0.022, 分别计算它们的自信息量.

(2) 假定前后字母出现是互相独立的, 计算 “ ac ” 的自信息量.

(3) 假定前后字母出现不是互相独立的, 当 “ a ” 出现以后, “ c ” 出现的概率为 0.04, 计算 “ a ” 出现以后, “ c ” 出现的自信息量.

解. (1) $I(a) = -\log_2 0.064 = 3.96 \text{ bit}$

$$I(c) = -\log_2 0.022 = 5.51 \text{ bit}$$

(2) 由于前后字母出现是互相独立的, “ ac ” 出现的概率为 0.064×0.022 , 所以 $I(ac) = -\log_2(0.064 \times 0.022) = -(\log_2 0.064 + \log_2 0.022) = I(a) + I(c) = 9.47 \text{ bit}$ 即两个相对独立的事件的自信息量满足可加性, 也就是由两个相对独立的事件的积事件所提供的信息量应等于它们分别提供的信息量之和.

(3) “ a ” 出现的条件下 “ c ” 出现的概率变大, 它的不确定性变小.

$$I(c|a) = -\log_2 0.04 = 4.64 \text{ bit}$$

8.1.2 熵及其性质

自信息量是信源发出某一具体消息所含有的信息量, 发出的消息不同它的自信息量就不同, 所以自信息量本身为随机变量, 不能用来表征整个信源的不确定度. 我们用平均自信息量来表征整个信源的不确定度. 平均自信息量又称为信息熵、信源熵, 简称熵.

因为信源具有不确定性, 所以用随机变量来表示信源, 用随机变量的概率分布来描述信源的不确定性. 通常把一个随机变量的所有可能的取值和这些取值对应的概率 $[X, P(X)]$ 称为它的概率空间.

假设随机变量 X 有 q 个可能的取值 $x_i, i = 1, 2, \dots, q$, 各种取值出现的概率为 $p(x_i), i = 1, 2, \dots, q$, 它的概率空间表示为

$$\begin{pmatrix} X \\ P(X) \end{pmatrix} = \begin{pmatrix} X = x_1 & \dots & X = x_i & \dots & X = x_q \\ p(x_1) & \dots & p(x_i) & \dots & p(x_q) \end{pmatrix}$$

这里要注意, $p(x_i)$ 满足概率空间的基本特性: 非负性 $0 \leq p(x_i) \leq 1$ 和完备性 $\sum_{i=1}^q p(x_i) = 1$.

定义 8.1.2. 随机变量 X 的每一个可能取值的自信息 $I(x_i)$ 的统计平均值定义为随机变量 X 的信息熵.

$$H(X) = \mathbb{E}[I(x_i)] = -\sum_{i=1}^q p(x_i) \log p(x_i)$$

这里 q 为 X 的所有可能取值的个数。

熵的单位也是与所取的对数底有关, 根据所取的对数底不同, 可以是比特、奈特、哈特莱或者是 r 进制单位, 通常用比特为单位.

例 8.1.2. 假设随机变量 X 的概率分布为 $p(x_i) = 2^{-i}, i = 1, 2, 3, \dots$, 求 $H(X)$.

解.

$$H(X) = \sum_{i=1}^{\infty} 2^{-i} \log_2 \frac{1}{2^{-i}} = \sum_{i=1}^{\infty} i 2^{-i} = 2 \text{ 比特}$$

熵编码

信息论的研究目标之一是如何用最少的编码表示传递信息. 假设我们要传递一段文本信息, 这段文本中包含的符号都来自于一个字母表 A , 我们就需要对字母表 A 中的每个符号进行编码. 以二进制编码为例, 我们常用的 ASCII 码就是用固定的 8bits 来编码每个字母. 但这种固定长度的编码方案不是最优的. 一种高效的编码原则是字母的出现概率越高, 其编码长度越短. 比如对字母 a, b, c 分别编码为 0, 10, 110.

给定一串要传输的文本信息, 其中字母 x 的出现概率为 $p(x)$, 其最佳编码长度为 $-\log_2 p(x)$, 整段文本的平均编码长度为 $-\sum_x p(x) \log_2 p(x)$, 即底为 2 的熵. 在对分布 $p(x)$ 的符号进行编码时, 熵 $H(p)$ 也是理论上最优的平均编码长度, 这种编码方式称为熵编码 (Entropy Encoding). 由于每个符号的自信息通常都不是整数, 因此在实际编码中很难达到理论上的最优值. 霍夫曼编码 (Huffman Coding) 和算术编码 (Arithmetic Coding) 是两种最常见的熵编码技术.

熵函数的性质

信息熵 $H(X)$ 是随机变量 X 的概率分布的函数, 所以又称为熵函数. 如果把概率分布 $p(x_i), i = 1, 2, \dots, q$, 记为 p_1, p_2, \dots, p_q , 则熵函数又可以写成概率向量 $\mathbf{p} = (p_1, p_2, \dots, p_q)$ 的函数形式, 记为 $H(\mathbf{p})$.

$$H(X) = -\sum_{i=1}^q p(x_i) \log p(x_i) = H(p_1, p_2, \dots, p_q) = H(\mathbf{p})$$

因为概率空间的完备性, 即 $\sum_{i=1}^q p(x_i) = 1$, 所以 $H(\mathbf{p})$ 是 $(q-1)$ 元函数. 当 $q = 2$ 时, 因为 $p_1 + p_2 = 1$, 若令其中一个概率为 p , 则另一个概率为 $(1-p)$, 熵函数可以写成 $H(p)$.

熵函数 $H(\mathbf{p})$ 具有以下性质:

1. 对称性

$$H(p_1, p_2, \dots, p_q) = H(p_2, p_1, \dots, p_q) = \dots = H(p_q, p_1, \dots, p_{q-1})$$

也就是说概率向量 $p = (p_1, p_2, \dots, p_q)$ 各分量的次序可以任意变更, 熵值不变。对称性说明熵函数仅与信源的总体统计特性有关。

2. 确定性

$$H(1, 0) = H(1, 0, 0) = H(1, 0, 0, 0) = \dots = H(1, 0, \dots, 0) = 0$$

在概率向量 $p = (p_1, p_2, \dots, p_q)$ 中, 只要有一个分量为 1, 其他分量必为 0, 它们对熵的贡献均为 0, 因此熵等于 0, 也就是说确定信源的平均不确定度为 0。

3. 非负性

$$H(p) = H(p_1, p_2, \dots, p_q) \geq 0$$

对确定信源, 等号成立.

信源熵是自信息的数学期望, 自信息是非负值, 所以信源熵必定是非负的. 离散信源熵才有这种非负性, 以后会讲到连续信源的微分熵则可能出现负值.

4. 扩展性

$$\lim_{\epsilon \rightarrow 0} H_{q+1}(p_1, p_2, \dots, p_q - \epsilon, \epsilon) = H_q(p_1, p_2, \dots, p_q)$$

这是因为 $\lim_{\epsilon \rightarrow 0} \epsilon \log \epsilon = 0$

这个性质的含义是: 增加一个基本不会出现的小概率事件, 信源的熵保持不变. 虽然小概率事件出现给予收信者的信息量很大, 但在熵的计算中, 它占的比重很小, 可以忽略不计, 这也是熵的总体平均性的体现.

5. 连续性

$$\lim_{\epsilon \rightarrow 0} H(p_1, p_2, \dots, p_{q-1} - \epsilon, p_q + \epsilon) = H(p_1, p_2, \dots, p_q)$$

即信源概率空间中概率分量的微小波动, 不会引起熵的变化.

6. 递增性

$$H(p_1, p_2, \dots, p_{n-1}, q_1, q_2, \dots, q_m) = H(p_1, p_2, \dots, p_n) + p_n H\left(\frac{q_1}{p_n}, \frac{q_2}{p_n}, \dots, \frac{q_m}{p_n}\right)$$

这个性质表明, 假如有一信源的 n 个元素的概率分布为 p_1, p_2, \dots, p_n , 其中某个元素 x_n 又被划分成 m 个元素, 这 m 个元素的概率之和等于元素 x_n 的概率, 这样得到的新信源的熵增加了一项, 增加的一项是由于划分产生的不确定性.

例 8.1.3. 利用递增性计算 $H(1/2, 1/8, 1/8, 1/8, 1/8)$.

解.

$$\begin{aligned}
 & H(1/2, 1/8, 1/8, 1/8, 1/8) \\
 &= H(1/2, 1/2) + \frac{1}{2} \times H(1/4, 1/4, 1/4, 1/4) \\
 &= 1 + \frac{1}{2} \times 2 \\
 &= 2 \text{ 比特}
 \end{aligned}$$

7. 极值性

$$H(p_1, p_2, \dots, p_n) \leq H\left(\frac{1}{n}, \dots, \frac{1}{n}\right) = \log n \quad (8.1)$$

式中 n 是随机变量 X 的可能取值的个数.

极值性表明离散信源中各消息等概率出现时熵最大, 这就是最大离散熵定理. 连续信源的最大熵则还与约束条件有关.

极值性可看成

$$H(p_1, p_2, \dots, p_n) \leq -\sum_{i=1}^n p_i \log_2 q_i \quad (8.2)$$

的特例情况. 下面先证明式(8.2)

证明. 利用 Jensen 不等式, 有

$$\begin{aligned}
 & H(p_1, p_2, \dots, p_n) + \sum_{i=1}^n p_i \log_2 q_i \\
 &= -\sum_{i=1}^n p_i \log_2 p_i + \sum_{i=1}^n p_i \log_2 q_i = \sum_{i=1}^n p_i \log_2 \frac{q_i}{p_i} \leq \log_2 \sum_{i=1}^n \left(p_i \cdot \frac{q_i}{p_i} \right) = 0
 \end{aligned}$$

当 $\frac{q_i}{p_i} = 1$, $i = 1, 2, \dots, n$ 时, 等号成立. 证毕. \square

式(8.2)表明任一随机变量的概率分布 p_i , 对其他概率分布 q_i 定义的自信息 $-\log_2 q_i$ 的数学期望, 必不小于概率分布 p_i 本身定义的熵 $H(p_1, p_2, \dots, p_n)$.

如果取 $q_i = \frac{1}{n}$, $i = 1, 2, \dots, n$ 时, 由式(8.2)就得到

$$H(p_1, p_2, \dots, p_n) \leq H\left(\frac{1}{n}, \dots, \frac{1}{n}\right) = \log n$$

当 $p_i = \frac{1}{n}$, $i = 1, 2, \dots, n$ 时, 等号成立.

当信源输出的消息等概分布时, 信源熵达到最大值——1 比特. 因此当二元数字是由等概的二元信源输出时, 每个二元数字提供 1 bit 的信息量, 否则, 每个二元数字提供的信息量小于 1 bit. 这就是信息量的单位比特和计算机术语中位的单位比特的关系.

8. 上凸性

$H(\mathbf{p})$ 是严格的上凸函数, 设 $\mathbf{p} = (p_1, p_2, \dots, p_q)$, $\mathbf{p}' = (p'_1, p'_2, \dots, p'_q)$, $\sum_{i=1}^q p_i = 1$, $\sum_{i=1}^q p'_i = 1$, 则对于任意小于 1 的正数 α , $0 < \alpha < 1$, 以下不等式成立:

$$H[\alpha \mathbf{p} + (1 - \alpha) \mathbf{p}'] > \alpha H(\mathbf{p}) + (1 - \alpha) H(\mathbf{p}')$$

证明. 因为 $0 \leq p_i \leq 1, 0 \leq p'_i \leq 1$, 且 $0 < \alpha < 1$, 所以 $0 \leq \alpha p_i + (1 - \alpha)p'_i \leq 1$, 并且 $\sum_{i=1}^q (\alpha p_i + (1 - \alpha)p'_i) = 1$, 所以 $\alpha p + (1 - \alpha)p'$ 可以看作是一种新的概率分布。

$$\begin{aligned} H(\alpha p + (1 - \alpha)p') &= -\sum_{i=1}^q (\alpha p_i + (1 - \alpha)p'_i) \log_2 (\alpha p_i + (1 - \alpha)p'_i) \\ &= -\alpha \sum_{i=1}^q p_i \log (\alpha p_i + (1 - \alpha)p'_i) - (1 - \alpha) \sum_{i=1}^q p'_i \log_2 (\alpha p_i + (1 - \alpha)p'_i) \\ &\geq -\alpha \sum_{i=1}^q p_i \log_2 p_i - (1 - \alpha) \sum_{i=1}^q p'_i \log_2 p'_i \\ &\geq \alpha H(p) + (1 - \alpha) H(p') \end{aligned}$$

当 $p \neq p'$ 时, 有 $\frac{\alpha p_i + (1 - \alpha)p'_i}{p_i} \neq 1$, 式(8.2)中等号不成立, 所以

$$H(\alpha p + (1 - \alpha)p') > \alpha H(p) + (1 - \alpha) H(p') \quad (8.3)$$

成立。证毕。 □

上凸函数在定义域内的极值必为极大值, 可以利用熵函数的这个性质证明熵函数的极值性。

直观来看, 随机变量的不确定程度并不都是一样的. 例如, 抛掷一枚均匀硬币结果所得到的信息量会比抛掷一枚偏畸硬币所得到的信息量大; 投掷一颗均匀骰子的试验比抛掷一枚均匀硬币的试验所得到的信息量大. 怎么度量这种不确定性呢? 香农指出, 存在这样的不确定性的度量, 它是随机变量的概率分布的函数, 而且必须满足 3 个公理性条件:

(1) 连续性条件: $f(p_1, p_2, \dots, p_n)$ 应是 $p_i, i = 1, 2, \dots, n$ 的连续函数;

(2) 等概时为单调函数: $f(1/n, 1/n, \dots, 1/n)$ 应是 n 的增函数;

(3) 递增性条件: 当随机变量的取值不是通过一次试验而是若干次试验才最后得到时, 随机变量在各次试验中的不确定性应该可加, 且其和始终与通过一次试验取得的不确定程度相同, 即

$$\begin{aligned} f(p_1, p_2, \dots, p_n) \\ = f((p_1 + p_2 + \dots + p_k), p_{k+1}, \dots, p_n) + (p_1 + p_2 + \dots + p_k) f(p'_1, p'_2, \dots, p'_k) \end{aligned}$$

其中, $p'_k = p_k / (p_1 + p_2 + \dots + p_k)$ 。

香农根据这 3 个公理性条件于 1948 年先提出了熵的概念, 他当时并没有像我们现在这样把熵看成自信息的均值. 后来, Feinstein(范恩斯坦)等人从数学上严格地证明了当满足上述条件时, 信息熵的表达形式是唯一的.

8.1.3 联合熵和条件熵

一个随机变量的不确定性可以用熵来表示, 这一概念可以方便地推广到多个随机变量。

定义 8.1.3. [联合熵] 二维随机变量 XY 的概率空间表示为

$$\begin{bmatrix} XY \\ P(XY) \end{bmatrix} = \begin{bmatrix} x_1y_1 & \cdots & x_iy_j & \cdots & x_ny_n \\ p(x_1y_1) & \cdots & p(x_iy_j) & \cdots & p(x_ny_n) \end{bmatrix}$$

其中, $p(x_iy_j)$ 满足概率空间的非负性和完备性: $0 \leq p(x_iy_j) \leq 1$, $\sum_{i=1}^n \sum_{j=1}^m p(x_iy_j) = 1$ 。

二维随机变量 XY 的联合熵定义为联合自信息的数学期望, 它是二维随机变量 XY 的不确定性的度量。

$$H(XY) = \sum_{i=1}^n \sum_{j=1}^m p(x_iy_j) I(x_iy_j) = - \sum_{i=1}^n \sum_{j=1}^m p(x_iy_j) \log p(x_iy_j)$$

定义 8.1.4. [条件熵] 考虑在给定 $X = x_i$ 的条件下, 随机变量 Y 的不确定性为

$$H(Y|x_i) = - \sum_j p(y_j|x_i) \log p(y_j|x_i)$$

对 $H(Y|x_i)$ 的所有可能值进行统计平均, 就得出给定 X 时 Y 的条件熵 $H(Y|X)$

$$\begin{aligned} H(Y|X) &= \sum_i p(x_i) H(Y|x_i) \\ &= - \sum_i \sum_j p(x_i) p(y_j|x_i) \log p(y_j|x_i) \\ &= - \sum_i \sum_j p(x_iy_j) \log p(y_j|x_i) \end{aligned}$$

性质 8.1.1. 联合熵和条件熵有如下关系:

$$H(XY) = H(X) + H(Y|X)$$

证明.

$$\begin{aligned} H(XY) &= \mathbb{E}(\log \frac{1}{p(xy)}) \\ &= \mathbb{E}(\log \frac{1}{p(x)p(y|x)}) \\ &= \mathbb{E}(\log \frac{1}{p(x)} + \log \frac{1}{p(y|x)}) \\ &= \mathbb{E}(\log \frac{1}{p(x)}) + \mathbb{E}(\log \frac{1}{p(y|x)}) \\ &= H(X) + H(Y|X) \end{aligned}$$

□

推论 8.1.1. 当二维随机变量 X , Y 相互独立时, 联合熵等于 X 和 Y 各自熵之和。

$$H(XY) = H(X) + H(Y)$$

证明. 因为随机变量 X, Y 相互独立, 所以有

$$\begin{aligned} p(x_i y_j) &= p(x_i) p(y_j) \\ H(XY) &= E[-\log_2 p(xy)] \\ &= E[-\log_2 p(x)p(y)] \\ &= E[-(\log_2 p(x) + \log_2 p(y))] \\ &= E[-\log_2 p(x)] + E[-\log_2 p(y)] \\ &= H(X) + H(Y) \end{aligned}$$

证毕. \square

8.1.4 互信息和相对熵

互信息

定义 8.1.5. 一个事件 y_j 所给出关于另一个事件 x_i 的信息定义为互信息, 用 $I(x_i; y_j)$ 表示.

$$I(x_i; y_j) = I(x_i) - I(x_i|y_j) = \log_2 \frac{p(x_i|y_j)}{p(x_i)} \quad (8.4)$$

互信息 $I(x_i; y_j)$ 是已知事件 y_j 后所消除的关于事件 x_i 的不确定性, 它等于事件 x_i 本身的不确定性 $I(x_i)$ 减去已知事件 y_j 后对 x_i 仍然存在的不确定性 $I(x_i|y_j)$. 互信息的引出, 使信息的传递得到了定量的表示.

例 8.1.4. 某地二月份天气出现的概率分别为晴 $1/2$, 阴 $1/4$, 雨 $1/8$, 雪 $1/8$. 某一天有人告诉你: “今天不是晴天”, 把这句话作为收到的消息 y_1 , 求收到 y_1 后, y_1 与各种天气的互信息量.

解. 把各种天气记作 x_1 (晴), x_2 (阴), x_3 (雨), x_4 (雪). 收到消息 y_1 后, 各种天气发生的概率变成了后验概率:

$$p(x_1|y_1) = \frac{p(x_1 y_1)}{p(y_1)} = 0$$

$$p(x_2|y_1) = \frac{p(x_2 y_1)}{p(y_1)} = \frac{1/4}{1/4 + 1/8 + 1/8} = \frac{1}{2}$$

$$p(x_3|y_1) = \frac{p(x_3 y_1)}{p(y_1)} = \frac{1/8}{1/4 + 1/8 + 1/8} = \frac{1}{4}$$

同理

$$p(x_4|y_1) = \frac{1}{4}$$

根据互信息量的定义, 可计算出 y_1 与各种天气之间的互信息:

$$I(x_1; y_1) = \log_2 \frac{p(x_1|y_1)}{p(x_1)} = \infty$$

$$I(x_2; y_1) = \log_2 \frac{p(x_2|y_1)}{p(x_2)} = \log_2 \frac{1/2}{1/4} = 1\text{bit}$$

$$I(x_3; y_1) = \log_2 \frac{p(x_3|y_1)}{p(x_3)} = \log_2 \frac{1/4}{1/8} = 1\text{bit}$$

$$I(x_4; y_1) = \log_2 \frac{p(x_4|y_1)}{p(x_4)} = \log_2 \frac{1/4}{1/8} = 1\text{bit}$$

定义 8.1.6. 定义互信息 $I(x_i; y_j)$ 在 XY 的联合概率空间中的统计平均值为随机变量 X 和 Y 间的平均互信息。

$$I(X; Y) = \sum_x \sum_y p(x, y) I(x_i; y_j)$$

也称为互信息。

互信息有以下性质：

性质 8.1.2. [对称性]

$$I(X; Y) = I(Y; X)$$

证明.

$$\begin{aligned} I(X; Y) &= \sum_{i=1}^n \sum_{j=1}^m p(x_i y_j) \log_2 \frac{p(x_i|y_j)}{p(x_i)} \\ &= \sum_{i=1}^n \sum_{j=1}^m p(x_i y_j) \log_2 \frac{p(x_i y_i)}{p(x_i) p(y_j)} \\ &= \sum_{i=1}^n \sum_{j=1}^m p(x_i y_j) \log_2 \frac{p(y_j|x_i)}{p(y_j)} \\ &= I(Y; X) \end{aligned}$$

证毕。 □

对称性表示从 Y 中获得关于 X 的信息量等于从 X 中获得关于 Y 的信息量。

性质 8.1.3. [非负性]

$$I(X; Y) \geq 0$$

当且仅当 $p(x, y) = p(x)p(y)$ 即 X 与 Y 独立时，互信息为 0

证明.

$$\begin{aligned}
 -I(X;Y) &= \sum_{i=1}^n \sum_{j=1}^m p(x_i y_j) \log_2 \frac{p(x_i) p(y_j)}{p(x_i y_j)} \\
 &\leq \log_2 \sum_{i=1}^n \sum_{j=1}^m p(x_i y_j) \frac{p(x_i) p(y_j)}{p(x_i y_j)} \\
 &= \log_2 \sum_{i=1}^n \sum_{j=1}^m p(x_i) p(y_j) \\
 &= 0
 \end{aligned}$$

所以

$$I(X;Y) \geq 0$$

证毕。 □

平均互信息是非负的, 说明给定随机变量 Y 后, 一般来说总能消除一部分关于 X 的不确定性.

相对熵

相对熵是两个随机分布之间距离的度量。统计学上对应于对数似然比的期望。

定义 8.1.7. 定义同一个随机变量 x 的两个概率密度函数 $p(x)$ 和 $q(x)$ 间的相对熵为:

$$D(p||q) = \mathbb{E}_{x \sim p} \left[\log \frac{p(x)}{q(x)} \right] = \mathbb{E}_{x \sim p} [\log p(x) - \log q(x)]$$

在机器学习中, 相对熵更常用的名称是 KL 散度 (Kullback-Leibler Divergence), 记做 $D_{KL}(p||q)$ 。

KL 散度有很多有用的性质, 首先 KL 散度是非负的并且可以度量两个分布之间的差异。但需要注意的是, 它并不是距离, 因为它不是对称的。其次, 当且仅当 p 和 q 在离散型变量的情况下是相同的分布, 或者在连续型变量的情况下“几乎处处”相同时, KL 散度才为 0. 此外, 联合分布 $p(X, Y)$ 和 $p(X)p(Y)$ 之间的 KL 散度可以作为 X 和 Y 的互信息的另一种定义:

$$I(X;Y) := D_{KL}(p(X, Y) \| p(X)p(Y)) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)}$$

性质 8.1.4. [互信息和熵的关系]

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) = H(X) + H(Y) - H(XY)$$

当 X, Y 统计独立时, $I(X;Y) = 0$.

性质 8.1.5. $H(X) = I(X;X)$

也就是随机变量 X 的熵是自己对自己的互信息。

性质 8.1.6. [极值性]

$$I(X;Y) \leq H(X), I(X;Y) \leq H(Y)$$

由于 $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$ ，而条件熵 $H(X|Y)$ 、 $H(Y|X)$ 是非负的，所以可得到 $I(X;Y) \leq H(X)$ ， $I(X;Y) \leq H(Y)$ 。极值性说明从一个事件获得的关于另一个事件的信息量至多只能是另一个事件的平均自信息量，不会超过另一事件本身所含的信息量。最好的情况是通信后 $I(X;Y) = H(X) = H(Y)$ ，最坏的情况是当 X, Y 相互独立时，从一个事件不能得到另一个事件的任何信息，即 $I(X;Y) = 0$ 。

推论 8.1.2. 条件熵和信息熵的关系

$$H(X|Y) \leq H(X), \quad H(Y|X) \leq H(Y) \quad (8.5)$$

证明。利用式(8.2)先证明式 $H(X|Y) \leq H(X)$

$$\begin{aligned} H(X|Y) &= - \sum_i \sum_j p(x_i y_j) \log_2 p(x_i | y_j) \\ &= - \sum_i \sum_j p(y_j) p(x_i | y_j) \log_2 p(x_i | y_j) \\ &= - \sum_j p(y_j) \sum_i p(x_i | y_j) \log_2 p(x_i | y_j) \\ &\leq - \sum_j p(y_j) \sum_i p(x_i | y_j) \log_2 p(x_i) \\ &= - \sum_j \sum_i p(x_i y_j) \log_2 p(x_i) \\ &= - \sum_i p(x_i) \log_2 p(x_i) = H(X) \end{aligned}$$

当 $p(x_i | y_j) = p(x_i)$ 时等号成立。

类似地，可以证明 $H(Y|X) \leq H(Y)$ 。证毕。 \square

推论 8.1.3. 联合熵和信息熵的关系：

$$H(XY) \leq H(X) + H(Y) \quad (8.6)$$

证明。

$$H(XY) = H(X) + H(Y|X) \leq H(X) + H(Y)$$

当 X, Y 相互独立时等号成立。推广到 N 个随机变量的情况：

$$H(X_1 X_2 \cdots X_N) \leq H(X_1) + H(X_2) + \cdots + H(X_N)$$

当 X_1, X_2, \dots, X_N 相互独立时，等号成立。 \square

8.1.5 熵、相对熵和互信息的链式法则

熵的链式法则

即两个随机变量 X 和 Y 的联合熵等于 X 的熵加上在 X 已知条件下 Y 的条件熵, 这个关系可以方便地推广到 N 个随机变量的情况, 即

$$H(X_1 X_2 \cdots X_N) = H(X_1) + H(X_2 | X_1) + \cdots + H(X_N | X_1 X_2 \cdots X_{N-1})$$

称为熵函数的链规则。

如果 N 个随机变量 X_1, X_2, \dots, X_N 相互独立, 则有

$$H(X_1 X_2 \cdots X_N) = \sum_{i=1}^N H(X_i) \quad (8.7)$$

互信息的链式法则

我们先定义条件互信息:

定义 8.1.8. 随机变量 X 和 Y 在给定随机变量 Z 时的条件互信息为

$$I(X; Y | Z) = H(X | Z) - H(X | Y, Z) = \sum_{i=1}^l \sum_{j=1}^m \sum_{k=1}^n p(x_i, y_j, z_k) \log_2 \frac{p(x_i, y_j | z_k)}{p(x_i | z_k) p(y_j | z_k)}$$

性质 8.1.7. 互信息的链式法则

$$I(X_1, X_2, \dots, X_n; Y) = \sum_{i=1}^n I(X_i; Y | X_{i-1}, X_{i-2}, \dots, X_1)$$

相对熵的链式法则

我们先定义条件相对熵:

定义 8.1.9. 联合概率密度函数 $p(x, y)$ 和 $q(x, y)$ 的条件概率熵 $D(p(y|x)||q(y|x))$ 定义为条件概率密度函数 $p(y|x)$ 和 $q(y|x)$ 间关于 $p(x)$ 的平均相对熵, 即

$$D(p(y|x)||q(y|x)) = \sum_{i=1}^m p(x_i) \sum_{j=1}^n p(y_j|x_i) \log_2 \frac{p(y_j|x_i)}{q(y_j|x_i)} = \sum_{i=1}^m \sum_{j=1}^n p(x_i, y_j) \log_2 \frac{p(y_j|x_i)}{q(y_j|x_i)}$$

性质 8.1.8. 相对熵的链式法则

$$D(p(x, y)||q(x, y)) = D(p(x)||q(x)) + D(p(y|x)||q(y|x))$$

8.1.6 信息不等式

信息不等式即相对熵恒非负性, 其直觉上的理解就是, 同一个随机变量的任意两个概率密度函数之间的距离总是大于等于 0 的, 这也为机器学习或深度学习模型的优化提供了上界。其证明如下所示。

性质 8.1.9. 信息不等式 设 $p(x), q(x)$ 是两个概率密度函数, 则

$$D(p||q) \geq 0$$

当且仅当对任意 x , $p(x) = q(x)$ 时, 等号成立。

证明.

$$\begin{aligned} -D(p||q) &= -\sum_x p(x) \log \frac{p(x)}{q(x)} = \sum_x p(x) \log \frac{q(x)}{p(x)} \\ &\leq \log \sum_x p(x) \frac{q(x)}{p(x)} = \log \sum_x q(x) \\ &= \log 1 = 0 \end{aligned}$$

□

利用信息不等式, 可以导出如下推论

推论 8.1.4. 对任意两个随机变量 X 和 Y ,

$$I(X;Y) \geq 0$$

当且仅当 X 与 Y 相互独立时, 等号成立.

推论 8.1.5. 条件相对熵非负, 即 $D(p(y|x)||q(y|x)) \geq 0$. 当且仅当对任意 y 满足 $p(y|x) = q(y|x)$ 时, 等号成立。

推论 8.1.6. 条件互信息非负, 即 $I(X;Y|Z) \geq 0$, 当且仅当对给定随机变量 Z 时, X 和 Y 是条件独立的, 等号成立。

信息处理定理

下面我们介绍信息处理定理。为了表述数据处理定理, 需要引入三元随机变量 X, Y, Z 的平均条件互信息和平均联合互信息的概念。

定义 8.1.10. [平均条件互信息]

$$I(X;Y|Z) = E[I(x;y|z)] = \sum_x \sum_y \sum_z p(xyz) \log_2 \frac{p(x|yz)}{p(x|z)} \quad (8.8)$$

它表示随机变量 Z 给定后, 从随机变量 Y 所得到的关于随机变量 X 的信息量。

定义 8.1.11. [平均联合互信息]

$$I(X;YZ) = E[I(x;yz)] = \sum_x \sum_y \sum_z p(xyz) \log_2 \frac{p(x|yz)}{p(x)} \quad (8.9)$$

它表示从二维随机变量 YZ 所得到的关于随机变量 X 的信息量。

可以证明

$$\begin{aligned} I(X;YZ) &= \sum_x \sum_y \sum_z p(xyz) \log_2 \frac{p(x|z)p(x|yz)}{p(x)p(x|z)} \\ &= I(X;Z) + I(X;Y|Z) \end{aligned} \quad (8.10)$$

同理

$$I(X;YZ) = I(X;Y) + I(X;Z|Y) \quad (8.11)$$

定理 8.1.1. [数据处理定理] 如果随机变量 X, Y, Z 构成一个马尔可夫链, 则有以下关系成立:

$$I(X;Z) \leq I(X;Y), I(X;Z) \leq I(Y;Z) \quad (8.12)$$

等号成立的条件是对于任意的 x, y, z , 有 $p(x|yz) = p(x|z)$ 和 $p(z|xy) = p(z|x)$ 。

证明. 当 X, Y, Z 构成一个马尔可夫链时, Y 值给定后, X, Z 可以认为是互相独立的 (在第九章概率图模型中, 将会了解到这一点). 所以,

$$I(X;Z|Y) = 0$$

又因为 $I(X;YZ) = I(X;Y) + I(X;Z|Y) = I(X;Z) + I(X;Y|Z)$, 并且 $I(X;Y|Z) \geq 0$, 所以 $I(X;Z) \leq I(X;Y)$ 。

当 $p(x|yz) = p(x|z)$ 时, Z 值给定后, X 和 Y 相互独立, 所以

$$I(X;Y|Z) = 0$$

因此

$$I(X;Z) = I(X;Y)$$

这时 $p(x|yz) = p(x|z) = p(x|y)$ 。 Y, Z 为确定关系时显然满足该条件。

同理可以证明 $I(X;Z) \leq I(Y;Z)$ 并且当 $p(z|xy) = p(z|x)$ 时, 等号成立。

证毕。 □

$I(X;Z) \leq I(X;Y)$ 表明从 Z 所得到的关于 X 的信息量小于等于从 Y 所得到的关于 X 的信息量. 如果把 $Y \rightarrow Z$ 看作数据处理系统, 那么通过数据处理后, 虽然可以满足我们的某种具体要求, 但是从信息量来看, 处理后会损失一部分信息, 最多保持原有的信息, 也就是说, 对接收到的数据 Y 进行处理后, 决不会减少关于 X 的不确定性. 这个定理称为数据处理定理. 数据处理定理与日常生活中的经验是一致的. 比如: 通过别人转述一段话或多或少会有一些失真, 通过书本得到的间接经验总不如直接经验来得详实.

8.2 连续分布的微分熵和最大熵

8.2.1 连续信源的微分熵

连续随机变量的取值是连续的, 一般用概率密度函数来描述其统计特征.

单变量连续信源的数学模型为 $X : \begin{bmatrix} \mathbb{R} \\ p(x) \end{bmatrix}$ ，并且满足 $\int_R p(x)dx = 1$ ， \mathbb{R} 是实数域，表示 X 的取值范围。

对于取值范围有限的单变量连续信源还可以表示成 $X : \begin{bmatrix} (a, b) \\ p(x) \end{bmatrix}$ ，并满足 $\int_a^b p(x)dx = 1$ ， (a, b) 是 X 的取值范围。

通过对连续变量的取值进行量化分层，可以将连续随机变量用离散随机变量来逼近。量化间隔越小，离散随机变量与连续随机变量越接近。当量化间隔趋于 0 时，离散随机变量就变成了连续随机变量。通过对离散随机变量的熵取极限，可以推导出连续随机变量熵的计算公式。

我们把连续随机变量 X 的取值分割成 n 个小区间，各小区间等宽，区间宽度 $\Delta = \frac{b-a}{n}$ ，则变量落在第 i 个小区间的概率为

$$P_r\{a + (i-1)\Delta \leq x \leq a + i\Delta\} = \int_{a + (i-1)\Delta}^{a + i\Delta} p(x)dx = p(x_i)\Delta \quad (8.13)$$

其中， x_i 是 $a + (i-1)\Delta$ 到 $a + i\Delta$ 之间的某一值。当 $p(x)$ 是连续函数时，由中值定理可知，必存在一个 x_i 使式(8.13)成立，这样，连续变量 X 就可用取值为 $x_i, i = 1, 2, \dots, n$ 的离散变量来近似，连续信源就被量化成离散信源，这 n 个取值对应的概率分布为 $p_i = p(x_i)\Delta$ ，这时的离散信源熵是

$$H(X) = - \sum_{i=1}^n p(x_i)\Delta \log_2 [p(x_i)\Delta] = - \sum_{i=1}^n p(x_i)\Delta \log_2 p(x_i) - \sum_{i=1}^n p(x_i)\Delta \log_2 \Delta \quad (8.14)$$

当 $n \rightarrow \infty$ 时， $\Delta \rightarrow 0$ ，如果(8.14)极限存在，离散信源熵就变成了连续信源的熵：

$$\begin{aligned} \lim_{\substack{n \rightarrow \infty \\ \Delta \rightarrow 0}} H(X) &= \lim_{n \rightarrow \infty} - \sum_{i=1}^n p(x_i)\Delta \log_2 p(x_i) - \lim_{n \rightarrow \infty} \sum_{i=1}^n p(x_i)\Delta \log_2 \Delta \\ &= - \int_a^b p(x) \log_2 p(x)dx - \lim_{n \rightarrow \infty} \log_2 \Delta \int_a^b p(x)dx \\ &= - \int_a^b p(x) \log_2 p(x)dx - \lim_{\substack{n \rightarrow \infty \\ \Delta \rightarrow 0}} \log_2 \Delta \end{aligned} \quad (8.15)$$

式(8.15)第一项一般是定值，第二项为无穷大量，因此连续信源的熵实际是无穷大量。这一点是可以理解的，因为连续信源的可能取值是无限多的，所以它的不确定性是无限大的，当确知输出为某值后，所获得的信息量也是无限大。在丢掉第二项后，定义第一项为连续信源的微分熵：

$$h(X) = - \int_R p(x) \log_2 p(x)dx \quad (8.16)$$

微分熵又称为差熵。虽然 $h(X)$ 已不能代表连续信源的平均不确定性，也不能代表连续信源输出的信息量，但是它具有和离散熵相同的形式，也具有离散熵的主要特性，比如可加性，但是不具有非负性。另外，我们在实际问题中常常考虑的是熵差，比如平均互信息，在讨论熵差时，只要两者离散逼近时所取的间隔 Δ 一致，这两个无限大量就将互相抵消，所以熵差具有信息的特性，如非负性。由此可见，连续信源的熵 $h(X)$ 具有相对性。

同样,可以定义两个连续随机变量的联合熵:

$$h(XY) = - \iint_{\mathbb{R}^2} p(xy) \log_2 p(xy) dx dy \quad (8.17)$$

以及条件熵

$$h(X|Y) = - \iint_{\mathbb{R}^2} p(xy) \log_2 p(y|x) dx dy \quad (8.18)$$

$$h(Y|X) = - \iint_{\mathbb{R}^2} p(xy) \log_2 p(x|y) dx dy \quad (8.19)$$

并且它们之间也有与离散随机变量一样的相互关系:

$$h(XY) = h(X) + h(Y|X) = h(Y) + h(X|Y) \quad (8.20)$$

$$h(X|Y) \leq h(X) \quad (8.21)$$

$$h(Y|X) \leq h(Y) \quad (8.22)$$

例 8.2.1. X 是在区间 (a, b) 内服从均匀分布的连续随机变量, 求微分熵.

$$p(x) = \begin{cases} \frac{1}{b-a} & x \in (a, b) \\ 0 & x \notin (a, b) \end{cases}$$

解.

$$h(X) = - \int_a^b p(x) \log_2 p(x) dx = - \int_a^b \frac{1}{b-a} \log_2 \frac{1}{b-a} dx = \log_2(b-a)$$

当 $(b-a) > 1$ 时, $h(X) > 0$

当 $(b-a) = 1$ 时, $h(X) = 0$

当 $(b-a) < 1$ 时, $h(X) < 0$ 这说明连续熵不具有非负性, 失去了信息的部分含义和性质 (但是熵差具有信息的特性)。

例 8.2.2. 求均值为 m , 方差为 σ^2 的高斯分布的随机变量的微分熵.

解. 高斯随机变量的概率密度为

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-m)^2}{2\sigma^2}}$$

微分熵为

$$\begin{aligned} h(X) &= - \int_{-\infty}^{+\infty} p(x) \log_2 p(x) dx \\ &= - \int_{-\infty}^{+\infty} p(x) \log_2 \left[\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}} \right] dx \\ &= - \int_{-\infty}^{+\infty} p(x) \log_2 \frac{1}{\sqrt{2\pi}\sigma} dx - \log_2 e \int_{-\infty}^{+\infty} p(x) \left[-\frac{(x-m)^2}{2\sigma^2} \right] dx \\ &= \log_2 \sqrt{2\pi}\sigma + \log_2 e \int_{-\infty}^{+\infty} p(x) \frac{(x-m)^2}{2\sigma^2} dx \\ &= \log_2 \sqrt{2\pi}\sigma + \frac{1}{2} \log_2 e \\ &= \log_2 \sqrt{2\pi e \sigma} \end{aligned}$$

这里对数以 2 为底, 所得微分熵的单位为比特, 如果对数取以 e 为底, 则得到

$$h(X) = \ln \sqrt{2\pi e} \sigma \text{ 奈特}$$

我们看到, 正态分布的连续信源的微分熵与数学期望 m 无关, 只与方差 σ^2 有关.

8.2.2 连续信源的最大熵

离散信源当信源符号为等概分布时有最大熵. 连续信源微分熵也有极大值, 但是与约束条件有关, 当约束条件不同时, 信源的最大熵不同. 我们一般关心的是下面两种约束下的最大熵.

定理 8.2.1. 在均值一定的情况下, 服从均匀分布的随机变量 X 具有最大熵.

即

$$p(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{其他} \end{cases}$$

$$h(X) = - \int_a^b p(x) \log_2 p(x) dx = - \int_a^b \frac{1}{b-a} \log_2 \frac{1}{b-a} dx = \log_2(b-a)$$

因此对于输出均值受限的连续信源, 当满足均匀分布时达到最大熵. 这个结论与离散信源在等概分布时达到最大熵的结论类似.

定理 8.2.2. 对于固定均值为 μ 和方差为 σ^2 的连续随机变量, 当服从高斯分布 $N(\mu, \sigma^2)$ 时具有最大熵.

证明. 对给定的 $p(x)$, 利用相对熵非负, 有

$$H(p) \leq - \int p(x) \log q(x) dx$$

取 $q(x) = N(\mu, \sigma^2)$, 有

$$\begin{aligned} H(p) &\leq - \int p(x) \log \left(\frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{(x-\mu)^2}{2\sigma^2} \right\} \right) dx = \int p(x) \left\{ \frac{(x-\mu)^2}{2\sigma^2} + \log \sqrt{2\pi}\sigma \right\} dx \\ &= \frac{1}{2\sigma^2} \int p(x)(x-\mu)^2 dx + \log \sqrt{2\pi}\sigma = \frac{1}{2} + \log \sqrt{2\pi}\sigma \end{aligned}$$

当 $p(x) = N(\mu, \sigma^2)$ 时, 可以取等, 证毕. \square

这说明, 当均值和方差一定时, 高斯分布的连续信源的熵最大.

8.3 信息论在数据科学中的应用

8.3.1 基于信息量的度量

我们在日常谈话中常常会说“你的话信息量太大了”, 其中的“信息量”到底指的是什么, 如何进行度量呢? 一个系统中有了新的信息, 系统的不确定性将会发生变化, 不是减少, 就是

增加。不确定性增加或者减少了多少，就是信息的度量。1948年香农借鉴了热力学的概念，把信息中排除了冗余后的平均信息量称为“信息熵”。接下来，我们就来学习具体如何进行信息的度量，以及与信息的度量相关的知识。

信息熵

信息熵：

$$H(X) = \mathbb{E}[I(x_i)] = - \sum_{i=1}^q p(x_i) \log p(x_i)$$

信息熵是对信息不确定性的度量，也可以这样理解，数据信息熵越小，数据就越纯。

互信息

假设带标签 X 的数据集有若干属性 Y_1, Y_2, \dots, Y_n ，我们想通过选择数据集的某个属性判断数据集的标签，那么我们就要根据哪一个属性对标签的信息量最大以选择属性。这时我们就要利用互信息

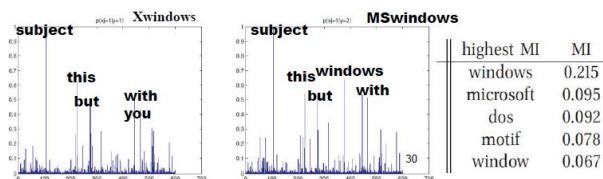
$$\arg \max_i I(X; Y_i) = \arg \max_i (H(X) - H(X|Y_i))$$

属性对标签的互信息可以理解成选择特定属性后，不确定性的下降量，也就是数据纯度的提升量。在机器学习中，这个量等价于信息增益。

关于信息熵和互信息（信息增益）在机器学习的应用，可以参考介绍决策树算法的相关书籍。

例 8.3.1. 在特征选择时，可以通过计算特征与目标之间的互信息，选择与目标互信息最大的那些特征，抛弃与目标关系不大的特征。

- 给定文档分类任务，将文档分成 *class 1 (X windows)* and *class 2 (MS windows)*，特征为 600 个二维特征 (600 个词语分别是否在文档中出现)，令 $p(x_i)$ 为词语在文档中出现的概率， $p(x_i|y_j)$ 为在 y_j 分类下词语在文档中出现的概率。则可计算 $I(X; Y) = H(X) - H(X|Y)$ ，互信息高的词语 (*windows, microsoft*) 更有判别性。



KL 散度

相对熵或称 KL 散度可以衡量同一个随机变量 x 的概率分布 $p(x)$ 和 $q(x)$ 的差异:

$$D_{KL}(p||q) = \mathbb{E}_{x \sim p}[\log p(x) - \log q(x)]$$

并且 KL 散度具有非负性, 当且仅当 p 和 q 在离散型变量的情况下是相同的分布, 或者在连续型变量的情况下是“几乎处处”相同, KL 散度为 0。但 KL 散度并不是距离, 因为它不满足对称性。

交叉熵

一个和 KL 散度密切联系的量是交叉熵 (cross-entropy):

定义 8.3.1. 设关于随机变量 x 的两个分布 $p(x), q(x)$, 关于这两个分布的交叉熵定义为:

$$H(p, q) = -\mathbb{E}_{x \sim p} \log q(x) = H(p) + D_{KL}(p||q) \quad (8.23)$$

由上式可以看出针对 q 最小化交叉熵等价于最小化 KL 散度, 因为 q 并不参与被省略的那一项。且在给定 p 的情况下, 如果 q 和 p 越接近, 交叉熵越小; 如果 q 和 p 越远, 交叉熵就越大。

JS 散度

JS 散度 (Jensen-Shannon Divergence) 是一种对称的衡量两个分布相似度的度量方式。

定义 8.3.2. 设关于随机变量 x 的两个分布 $p(x), q(x)$, 关于这两个分布的 JS 散度定义为:

$$D_{JS}(p, q) = \frac{1}{2}D_{KL}(p, M) + \frac{1}{2}D_{KL}(q, M)$$

其中 $M = \frac{1}{2}(p + q)$

JS 散度是 KL 散度一种改进。但两种散度都存在一个问题, 即如果两个分布 p, q 没有重叠或者重叠非常少时, KL 散度和 JS 散度都很难衡量两个分布的距离。

8.3.2 其他概率相关的度量

本节还将介绍一些其他和概率相关的度量, 作为基于信息论的度量的补充。

马氏距离 (Mahalanobis Distance)

在前面, 我们介绍了一些关于度量两个向量相似度的一些方法。

并且我们提到了闵氏距离 (包括曼哈顿距离、欧氏距离和切比雪夫距离) 存在明显的缺点, 并通过下例进行了说明。

例 8.3.2. 给定二维样本 (身高, 体重), 其中身高范围是 150 ~ 190, 体重范围是 50 ~ 60, 有三个样本: $a(180, 50)$, $b(190, 50)$, $c(180, 60)$ 。

- 通过计算可以得出 ab 之间的闵氏距离等于 ac 之间的闵氏距离, 但是身高的 10cm 不等价于体重的 10kg。

现在我们就来介绍解决这个问题的一种度量相似度的方式。

定义 8.3.3. 马氏距离: 表示点与一个分布之间的距离。有 m 个样本向量 $\mathbf{x}_1, \dots, \mathbf{x}_m$, 协方差矩阵记为 \mathbf{S} , 均值记为向量 $\boldsymbol{\mu}$, 则其中样本向量 \mathbf{x} 到 $\boldsymbol{\mu}$ 的马氏距离表示为:

$$dist(\mathbf{x}, \boldsymbol{\mu}) = \sqrt{(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{S}^{-1} (\mathbf{x} - \boldsymbol{\mu})}$$

而其中向量 \mathbf{x}_i 与 \mathbf{x}_j 之间的马氏距离定义为:

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{S}^{-1} (\mathbf{x}_i - \mathbf{x}_j)}$$

若协方差矩阵是单位矩阵 (各个样本向量之间独立同分布), 则公式就成了:

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}$$

也就是欧氏距离了。

马氏距离的优点: 它不受量纲的影响, 两点之间的马氏距离与原始数据的测量单位无关。马氏距离还可以排除变量之间的相关性的干扰。

皮尔逊相关系数

相关系数是衡量随机变量 \mathbf{x} 与 \mathbf{y} 线性相关程度的一种方法, 一般用 r 表示。 r 的取值范围是 $[-1, 1]$ 。 r 的绝对值越大, 则表明 \mathbf{x} 与 \mathbf{y} 线性相关度越高。当 \mathbf{x} 与 \mathbf{y} 线性相关时, 相关系数取值为 1 (正线性相关) 或 -1 (负线性相关)。

定义 8.3.4. 设随机变量 \mathbf{x}, \mathbf{y} , 皮尔逊相关系数定义为:

$$r(\mathbf{x}, \mathbf{y}) = \frac{\text{Cov}(\mathbf{x}, \mathbf{y})}{\sqrt{D(\mathbf{x})} \sqrt{D(\mathbf{y})}} = \frac{E((\mathbf{x} - E\mathbf{x})(\mathbf{y} - E\mathbf{y}))}{\sqrt{D(\mathbf{x})} \sqrt{D(\mathbf{y})}}$$

其中, $\text{Cov}(\mathbf{x}, \mathbf{y})$ 为 \mathbf{x} 与 \mathbf{y} 的协方差, $\sqrt{D(\mathbf{x})}$ 为 \mathbf{x} 的方差, $\sqrt{D(\mathbf{y})}$ 为 \mathbf{y} 的方差。

Wasserstein 距离

Wasserstein 距离 (Wasserstein Distance) 也用于衡量两个分布之间的距离。

定义 8.3.5. 对于两个分布 q_1, q_2 , p 级 Wasserstein 距离定义为

$$W_p(q_1, q_2) = \left(\inf_{\gamma(x, y) \in \Gamma(q_1, q_2)} E_{(x, y)} \gamma(x, y) [d(x, y)^p] \right)^{\frac{1}{p}}$$

其中 $\Gamma(q_1, q_2)$ 是边际分布为 q_1 和 q_2 的所有可能的联合分布集合, $d(x, y)$ 为 x 和 y 的距离, 比如 l_p 距离等。

Wasserstein 距离相比 KL 散度和 JS 散度的优势在于: 即使两个分布没有重叠或者重叠非常少, Wasserstein 距离仍然能反映两个分布的远近。在生成网络 GAN 中, 为了生成与目标分布接近的分布, 使用 JS 散度时训练起来比较困难, 而使用 Wasserstein 距离使得模型训练更稳定。

例 8.3.3. 对于 \mathbb{R}^n 空间中的两个高斯分布 $p = N(\mu_1, \Sigma_1)$ 和 $q = N(\mu_2, \Sigma_2)$, 它们的 2nd-Wasserstein 距离为

$$W_2(p, q) = \|\mu_1 - \mu_2\|_2^2 + \text{Tr}(\Sigma_1 + \Sigma_2 - 2(\Sigma_2^{\frac{1}{2}} \Sigma_1 \Sigma_2^{\frac{1}{2}})^{\frac{1}{2}})$$

当两个分布的方差为 0 时, 2nd-Wasserstein 距离等价于欧氏距离。

Jaccard 系数

Jaccard 系数又称为 Jaccard 相似系数, 用于比较有限样本集之间的相似性与差异性。Jaccard 系数值越大, 样本相似度越高。

定义 8.3.6. 两个集合 \mathbb{A} 和 \mathbb{B} 的交集元素在 $\mathbb{A} \cup \mathbb{B}$ 的并集中所占的比例, 称为两个集合的 Jaccard 相似系数, 用符号 $J(\mathbb{A}, \mathbb{B})$ 表示。

$$J(\mathbb{A}, \mathbb{B}) = \frac{|\mathbb{A} \cap \mathbb{B}|}{|\mathbb{A} \cup \mathbb{B}|}$$

当集合 \mathbb{A} , \mathbb{B} 都为空时, $J(\mathbb{A}, \mathbb{B})$ 定义为 1。

Jaccard 相似系数是衡量两个集合的相似度一种指标。

对于等概率的随机排列, 两个集合的 minHash 值相同的概率等于两个集合的 Jaccard 相似度。关于 minHash 算法, 可以参考有关数据科学算法的教材如 *Mining of Massive Datasets*。

Jaccard 距离

Jaccard 距离: 与 Jaccard 系数相关的概念是 Jaccard 距离。

定义 8.3.7. Jaccard 距离可用如下公式表示:

$$J_D(\mathbb{A}, \mathbb{B}) = 1 - J(\mathbb{A}, \mathbb{B}) = \frac{|\mathbb{A} \cup \mathbb{B}| - |\mathbb{A} \cap \mathbb{B}|}{|\mathbb{A} \cup \mathbb{B}|}$$

Jaccard 距离用两个集合中不同元素占所有元素的比例来衡量两个集合的区分度。

例 8.3.4. Jaccard 相似系数用在衡量样本的相似度上。

样本 A 与样本 B 是两个 n 维向量, 而且所有维度的取值都是 0 或 1。例如: $A = (0111)$ 和 $B = (1011)$ 。我们将样本看成是一个集合, 1 表示集合包含该元素, 0 表示集合不包含该元素。 p : 样本 A 与 B 都是 1 的维度的个数;

q : 样本 A 是 1, 样本 B 是 0 的维度的个数;

r : 样本 A 是 0, 样本 B 是 1 的维度的个数;

s : 样本 A 与 B 都是 0 的维度的个数。

那么样本 A 与 B 的 Jaccard 相似系数可以表示为:

$$J = \frac{p}{p + q + r}$$

这里 $p + q + r$ 可理解为 A 与 B 的并集的元素个数, 而 p 是 A 与 B 的交集的元素个数。而样本 A 与 B 的 Jaccard 距离表示为:

$$J_D = \frac{q + r}{p + q + r}$$

8.4 阅读材料

熵的概念首先在热力学中引入, 用于表述热力学第二定律。此后, 统计力学告诉我们, 在系统的某个宏观状态中, 热力学熵与微观状态数目的对数之间存在着联系。此项研究工作归功于玻尔兹曼的伟大成就, 他给出了方程 $S = k \ln W$, 该方程式作为墓志铭刻在了他的墓碑上。

20 世纪 30 年代, Hanley 在通信系统中引入了信息的对数度量。这个度量本质上是字母表大小的对数。本章中熵与互信息的定义由香农首先给出。相对熵概念由库尔贝克 (Kullback) 和 Leibler 首先定义, 它有各种各样的命名, 包括 Kullback-Leibler 距离、交叉熵、信息散度、信息判别, 在 Csiszdr 和 Amari 中有详细的论述。

许多简单性质都是由香农发展起来的。费诺不等式的证明见 Fano。充分统计量概念由费希尔 (FiSher) 定义, 而最小充分统计量是由 Lehmann 和 Scheffc 引入的。互信息与充分性关系的解释归功于 Kullback。Brillouin 和 Jaynes 对信息论和热力学之间的关系给予了广泛的讨论。

信息物理学是一门相当新型的学科, 产生于统计力学、量子力学和信息论。讨论的关键问题是如何将信息表示物理化。量子信道容量 (物理系统中可分辨的制备数量的对数) 和量子数据压缩都是定义明确的问题, 利用冯·诺伊曼熵获得了完美的解答。由于量子纠缠的存在, 以及观察到的物理事件的边际分布与任何联合分布均不一致 (没有局部的真实) 这一结论 (体现于贝尔 (Bell) 不等式), 量子信息的研究有了新的课题。Nielsen 和 Chuang 所著的基础文献较为详尽地论述了量子信息论, 同时包含本书中的许多结论的量子形式。人们也试图确定在计算上是否存在本质的物理限制, 这些工作包括 Bennett 以及 Bennett 与 Landauer。

习题

习题 8.1. 同时抛 2 颗骰子, 事件 A, B, C 分别表示: (A) 仅有一个骰子是 3; (B) 至少有一个骰子是 4; (C) 骰子上点数的总和为偶数。是计算事件 A, B, C 发生后所提供的信息量。

习题 8.2. 计算熵函数 $H(1/3, 1/3, 1/6, 1/6)$ 的值。

习题 8.3. X 和 Y 是 $\{0, 1, 2, 3\}$ 上的独立、等概分布的随机变量, 求:

(1) $H(X + Y), H(X - Y), H(X \cdot Y)$

(2) $H(X + Y, X), H(X \cdot Y, X)$

习题 8.4. X, Y, Z 为 3 个随机变量, 证明一下不等式成立并指出等号成立的条件:

(1) $H(XY|Z) \geq H(X|Z)$

(2) $I(XY; Z) \geq I(X; Z)$

(3) $H(XYZ) - H(XY) \leq H(XZ) - H(X)$

(4) $I(X; Z|Y) \geq I(Z; Y|X) - I(Z; Y) + I(X; Z)$

习题 8.5. 找出一个概率分布 $\{p_1, p_2, p_3, p_4, p_5\}$, 并且 $p_i > 0$, 使得 $H(p_1, p_2, p_3, p_4, p_5) = 2$

习题 8.6. 假定 $X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow \cdots \rightarrow X_n$ 形成一个马尔科夫链, 那么 $p(x_1 x_2 \cdots x_n) = p(x_1)p(x_2|x_1) \cdots p(x_n|x_{n-1})$, 化简 $I(X_1; X_2, \cdots, X_n)$

习题 8.7. 假定 X 是一个离散随机变量, $g(X)$ 是 X 的函数, 证明: $H[g(X)] \leq H(X)$ 。

习题 8.8. 三扇门中有一扇门后面藏有一袋金子, 并且三扇门后面藏有金子的可能性相同。如果有人随机打开一扇门并告诉你门后是否藏有金子, 他给了你多少关于金子位置的信息量?

参考文献

- [1] S.Amari.Differential-Geometrical Methods in Statistics.Springer-Vcrlag,New York, 1985.
- [2] C.H.Bennett.Demons, engines and the second law.Sci.Am.259(5):108-116.Nov.1987.
- [3] C.H.Bennett and R.Landauer. The fundamental physical limits of computation. Sci. Am. 255(1):48-56,July 1985.
- [4] I.Csiszar. Information type measures of difference of probability distributions and indirect observations. Stud. Sci. Math. Hung. 2:299-318,1967.
- [5] R Jozsa and B. Schumacher. A new proof of the quantum noiseless coding theorem. J Mod. Opt, pages 2343-2350,1994
- [6] S.Kullback and R.A.Leibler. On information and sufficiency. Ann.Math.Stat.22:79-87,1951.

- [7] D.Lindley.Boltzmann's Atom:The Great Debate That Launched A Revolution in Physics. Free Press,New York,2001.
- [8] M.Nielsen and I.Chang. Quantum Computation and Quantum Information. Cambridge University Press, Cambridge,2000.
- [9] C.E.Shannon.A mathematical theory of communication.Bell Syst.Tech.J.27:379-423,623-656,1948.

数据科学与工程数学基础

第九章 概率模型

在实践中我们所遇到的问题中各种变量之间的关系更多是动态的、不确定的，所以我们无法使用确定的函数表示变量之间的关系，此时我们可以选择概率模型来描述它们。机器学习中有一大类模型是基于概率的。概率分布的模型表达主要分为两种情况。第一种情况，不考虑模型变量间的依赖关系。若我们已经有含参数的公式来描述连续随机变量的概率密度或者离散随机变量的概率，这种分布就称为参数型概率分布，此时我们可以采用极大似然法或极大后验概率法估计出概率分布的参数，从而得到随机变量的概率分布了；若我们无法使用确定的公式来描述随机变量的概率分布，此时可以采用非参数概率模型。第二种情况，若考虑变量间存在依赖关系的情形，我们就可以采用图的方式来表达随机变量之间的结构关系，即概率图模型。本章我们会详细介绍这两种情况的模型构建。并在之后，介绍如何利用统计决策理论，对模型进行评估和选择。

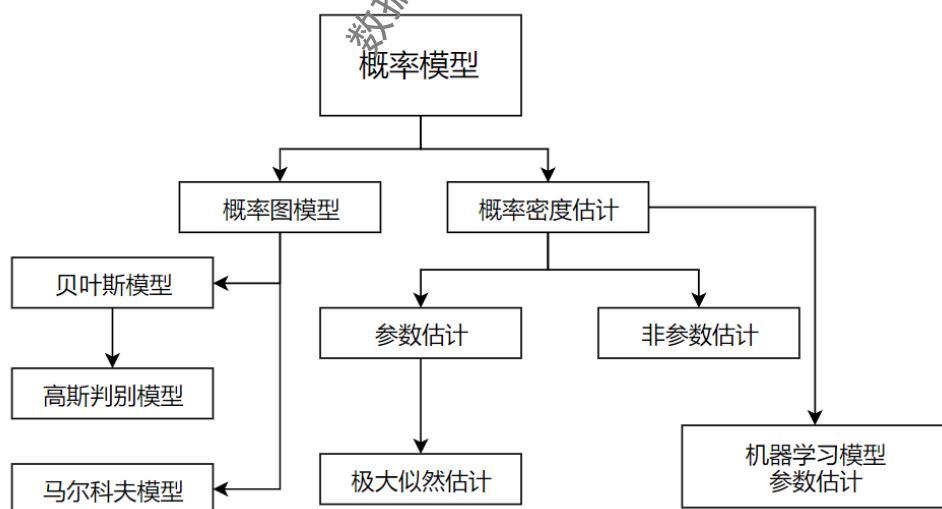


图 9.1: 本章导图

9.1 从概率到统计

从第 7 章内容, 我们知道随机变量及其所伴随的概率分布全面描述了随机现象的统计规律性, 因此要研究一个随机现象首先要知道它的概率分布。在概率论中, 概率分布通常是已知的, 或假设为已知的, 而一切概率计算和推理, 比如求出它的数字特征, 讨论随机变量函数的分布, 介绍各种常用的分布, 就在这已知的基础上得出来。但在实际中, 一个随机现象所服从的分布是什么类型可能完全不知道, 比如电视机的寿命服从什么分布是不知道的; 或者由于现象的某些事实而知道其类型, 但不知其分布函数中所含的参数, 比如一件产品是合格品还是不合格品服从一个二项分布, 但分布中参数 p (不合格品率) 却不知道。为了对这些问题展开研究, 必须知道它们的分布或分布所含的参数。

那么怎样才能知道一个随机现象的分布或参数呢? 这是统计学所要解决的一个首要问题。

9.1.1 统计的基本概念

在统计中我们总是从所要研究的对象全体中抽取一部分进行观测或试验以取得数据或信息, 从而对整体作出推断。由于观测或试验是随机现象, 依据有限个观测或试验对整体所作出的推论不可能绝对准确, 含有一定程度的不确定性, 而这种不确定性用概率的大小来表示比较恰当。概率大, 推断就比较可靠, 概率小, 推断就比较不可靠。所以统计的基本问题就是依据观测或试验所取得的有限信息对整体如何推断的问题。每个推断必须伴随一定的概率以表明推断的可靠程度。这种伴随有一定概率的推断称为统计推断。

我们把研究的对象全体所构成的集合称为总体, 而把组成总体的每一个单元成员称为个体。例如变压器的总体就组成一个总体, 其中每一个变压器就是一个个体。在实际中我们所研究的往往是总体中个体的各种数值指标 X , 例如变压器的寿命指标, 它是一个随机变量。假设 X 的分布函数是 $F(x)$, 有时简记为 F 。如果我们主要关心的只是这个数值指标 X , 为了方便起见, 我们可以把这个数值指标 X 的可能取值的全体看作总体, 并且称这一总体为具有分布函数 $F(x)$ 的总体, 这样就把总体和随机变量联系起来了。这种联系也可以推广到 k 维, 这样就和随机向量联系起来。

在实际实验中, 我们通过观测或试验以取得信息。如果我们按照机会均等的原则随机地选取一些个体进行观测或测试某一指标 X 的数值, 我们把这一过程称为随机抽样。假如我们抽取了 n 个个体, 且这 n 个个体的某一指标为 (X_1, X_2, \dots, X_n) , 我们称这 n 个个体的指标 (X_1, X_2, \dots, X_n) 为一个样本, n 称作这个样本的容量。在重复取样中每个 X_i 是一个随机变量, 从而我们把容量为 n 的样本 (X_1, X_2, \dots, X_n) 看成一个 n 维随机向量。

在一次抽样以后, 观测到 (X_1, X_2, \dots, X_n) 的一组确定的值 (x_1, x_2, \dots, x_n) 称作容量为 n 的样本的观测值或数据。容量为 n 的样本的观测值 (x_1, x_2, \dots, x_n) 可以看作一个随机试验的一个结果, 它的一切可能的结果的全体构成一个样本空间。它可以是 n 维空间, 也可以是其中的

一个子集。而样本的一组观测值 (x_1, x_2, \dots, x_n) 是样本空间的一个点。

实际上, 从总体中抽取样本可以有各种不同的方法。为了使抽到的样本能够对总体作出比较可靠的推断, 就希望它能很好地代表总体, 这就需要对抽样方法提出一些要求。比如: (1) 总体中每一个个体有同等机会选入样本; (2) 样本的分量 X_1, X_2, \dots, X_n 是相互独立的随机变量, 即样本的每一分量有什么观测结果并不影响其它分量有什么观测结果。这样取得的样本称为简单随机样本。例如放回抽样所得的样本就是简单随机样本。

例 9.1.1. 设总体 X 具有分布函数 $F(x)$, (X_1, X_2, \dots, X_n) 为取自这一总体的容量为 n 的样本, 则 (X_1, X_2, \dots, X_n) 的联合分布函数

$$F^*(x_1, x_2, \dots, x_n) = \prod_{i=1}^n F(x_i).$$

又若 X 具有概率密度 f , 则 (X_1, X_2, \dots, X_n) 的概率密度为

$$f^*(x_1, x_2, \dots, x_n) = \prod_{i=1}^n f(x_i).$$

为了研究总体分布的性质, 人们通过试验得到许多观测值, 一般来说, 这些数据是杂乱无章的, 为了利用它们进行统计分析, 要将这些数据加以整理, 还常借助于表格或图形对它们加以描述。例如, 对于连续型随机变量 X 引入频率直方图或数据的箱线图, 它们可以使人们对总体 X 的分布有一个粗略的了解。

另一方面, 我们知道, 样本是总体的反映, 但是样本所含的信息不能直接用于解决我们所要研究的问题, 而需要把样本所含的信息进行数学上的加工, 使其浓缩起来, 从而解决我们的问题。这在统计学当中, 往往通过构造一个合适的依赖于样本的函数——统计量——来达到。

统计量

定义 9.1.1. 设 (X_1, X_2, \dots, X_n) 是来自总体 X 的一个样本, $g(X_1, X_2, \dots, X_n)$ 是 X_1, X_2, \dots, X_n 的函数, 若 g 中不含未知参数, 则称 $g(X_1, X_2, \dots, X_n)$ 是一个统计量。

显然统计量也是一个随机变量。设 (x_1, x_2, \dots, x_n) 是相应于样本 (X_1, X_2, \dots, X_n) 的样本值, 则称 $g(x_1, x_2, \dots, x_n)$ 是 $g(X_1, X_2, \dots, X_n)$ 的观测值。常用的统计量包括:

- 样本均值: $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$;
- 样本方差: $S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2 = \frac{1}{n-1} (\sum_{i=1}^n X_i^2 - n\bar{X}^2)$
- 样本标准差: $S = \sqrt{S^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}$
- 样本 k 阶 (原点) 矩: $A_k = \frac{1}{n} \sum_{i=1}^n X_i^k, k = 1, 2, \dots$
- 样本 k 阶中心矩: $B_k = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^k, k = 2, 3, \dots$

它们的观察值分别为:

- $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$

- $s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{n-1} (\sum_{i=1}^n x_i^2 - n\bar{x}^2)$
- $s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$
- $a_k = \frac{1}{n} \sum_{i=1}^n x_i^k, \quad k = 1, 2, \dots$
- $b_k = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^k, \quad k = 2, 3, \dots$

这些观察值仍分别称为样本均值, 样本方差, 样本标准差, 样本 k 阶 (原点) 矩以及样本 k 阶中心矩。

定理 9.1.1. 若总体 X 的 k 阶矩 $E(X^k)$ ^{记成} μ_k 存在, 则当 $n \rightarrow \infty$ 时, $A_k \xrightarrow{P} \mu_k, k = 1, 2, \dots$ 。

证明. 因为 X_1, X_2, \dots, X_n 独立且与 X 同分布, 所以 $X_1^k, X_2^k, \dots, X_n^k$ 独立且与 X^k 同分布, 故有

$$E(X_1^k) = E(X_2^k) = \dots = E(X_n^k) = \mu_k.$$

从而由辛钦大数定理知, $A_k = \frac{1}{n} \sum_{i=1}^n X_i^k \xrightarrow{P} \mu_k, \quad k = 1, 2, \dots$ □

进而由关于依概率收敛的序列的性质, 可知

$$g(A_1, A_2, \dots, A_k) \xrightarrow{P} g(\mu_1, \mu_2, \dots, \mu_k)$$

其中 g 为连续函数。

设 (x_1, x_2, \dots, x_n) 是取自分布为 $F(x)$ 的总体中一个简单随机样本的观测值。若把样本观测值由小到大进行排列, 得到 $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$, 这里 $x_{(1)}$ 是样本观测值 (x_1, \dots, x_n) 中最小一个, $x_{(i)}$ 是样本观测值中第 i 个小的数等, 则

$$F_n(x) = \begin{cases} 0 & \text{当 } x \leq x_{(1)} \\ \frac{k}{n} & \text{当 } x_{(k)} < x \leq x_{(k+1)}, k = 1, 2, \dots, n-1 \\ 1 & \text{当 } x > x_{(n)} \end{cases}$$

显然, $F_n(x)$ 是一非减左连续函数, 且满足

$$F_n(-\infty) = 0 \text{ 和 } F_n(+\infty) = 1$$

由此可见, $F_n(x)$ 是一个分布函数, 称作经验分布函数 (或子样分布函数)。

对于经验分布函数 $F_n(x)$, 格里汶科 (Glivenko) 在 1933 年证明了以下的结果: 对于任一实数 x , 当 $n \rightarrow \infty$ 时 $F_n(x)$ 以概率 1 一致收敛于分布函数 $F(x)$, 即

$$P \left\{ \lim_{n \rightarrow \infty} \sup_{-\infty < x < \infty} |F_n(x) - F(x)| = 0 \right\} = 1$$

因此, 对于任一实数 x 当 n 充分大时, 经验分布函数的任一个观察值 $F_n(x)$ 与总体分布函数 $F(x)$ 只有微小的差别, 从而在实际上可当作 $F(x)$ 来使用。

抽样分布

在使用统计量进行统计推断时，常需要知道它的分布。我们称统计量的分布为抽样分布。当总体的分布函数已知时，抽样分布是确定的，然而要求出统计量的精确分布，一般来说是困难的。来自正态总体的几个常用的统计量的抽样分布如下：

- χ^2 分布
- t 分布，也称为学生分布
- F 分布

上述三个分布称为统计学的三大分布，它们在数理统计中有着广泛的应用。

9.1.2 模型、统计推断和学习

统计学的基本问题之一是统计推断，在数据科学或机器学习领域，称之为学习，是指利用数据（样本）去推断产生这些数据（样本）的总体分布的过程。一个典型的统计推断问题是：

- 给定样本 $X_1, \dots, X_n \sim F$ ，怎样去推断总体分布 F ？
- 某些情况下，只需推断分布 F 的某种性质，如数字特征，包括均值方差等。

通常把数据服从的一系列分布称为概率模型或统计模型。

本书我们只讨论总体分布是连续型和离散型两种情形。为了简便起见，我们引入一个对两种情形通用的概念——概率函数。我们称随机变量（总体） X 的概率函数为 $f(x)$ 的意思是指：

- 在连续情形时， $f(x)$ 是 $X = x$ 的密度函数值；
- 在离散情形时， $f(x)$ 是 $X = x$ 的概率。

一般地，在实际推断中，我们对样本总体分布情况的了解有两种可能性：一种是其形式已知，并且可以用有限个参数来表示（虽然这些参数可能是未知的）；另一种是其形式未知，或者其形式已知但不能用有限个参数来表示。由此我们引出分布的表示：参数和非参数模型。

参数与非参数模型

定义 9.1.2. 参数模型是指一系列可用有限个参数表示的概率模型 \mathfrak{F} 。

一般地，参数模型可以用一族带参数 θ 的概率函数来表示，具有如下形式：

$$\mathfrak{F} = \{f(x; \theta) : \theta \in \Theta\},$$

其中 $f(x; \theta)$ 是总体（也就是随机变量） X 的概率函数，参数 θ （可能是标量或向量）除了只知道它的可能取值范围为 Θ 外，其它一无所知。今后我们称 Θ 为参数空间。如果 θ 是向量，但仅关心其中的一个元素的时候，则称其它参数为冗余参数。例如 $\{N(\mu; 1) : \mu \in R\}$ 是 μ 取实数值的一族正态分布。

定义 9.1.3. 非参数模型指一些不能用有限个参数表示的概率模型 \mathfrak{F} 。

例如, $\mathfrak{F}_{\text{所有}} = \{\text{所有 CDF}\}$ 就是非参数模型。非参数模型是相对参数模型来说的。

例 9.1.2. (一维参数估计) 令 X_1, \dots, X_n 为相互独立的 $Bernoulli(p)$ 观察值, 问题是如何估计参数 p 。

例 9.1.3. (二维参数估计) 假设 $X_1, \dots, X_n \sim F$ 并假设 PDF $f \in \mathfrak{F}$, 其中 \mathfrak{F} 满足高斯分布。这种情况下就有两个参数 μ 和 σ , 目标是根据数据去估计这两个参数, 如果仅关心估计 μ 的值, 则 μ 就是感兴趣的参数而 σ 就是冗余参数。

例 9.1.4. (CDF 的非参数估计) 令 X_1, \dots, X_n 是来源于 CDF 为 F 的独立观察值, 问题是在假设 $F \in \mathfrak{F}_{\text{所有}} = \{\text{所有 CDF}\}$ 的前提下如何去估计 F 。

例 9.1.5. (非参数密度估计) 令 X_1, \dots, X_n 是来源于 CDF 为 F 的独立观察值, 令 $f = F'$ 为 PDF。假设要估计 PDF f 。如果仅假设 $F \in \mathfrak{F}_{\text{所有}}$ 是不可能估计 f 的, 需要假设 f 的光滑性, 例如, 假设 $f \in \mathfrak{F} = \mathfrak{F}_{\text{DENS}} \cap \mathfrak{F}_{\text{SOB}}$, 其中, $\mathfrak{F}_{\text{DENS}}$ 表示所有密度函数的集合

$$\mathfrak{F}_{\text{SOB}} = \{f: \int (f'(x))^2 dx < \infty\} \quad (9.1)$$

集合 $\mathfrak{F}_{\text{SOB}}$ 称为索伯列夫空间 (Sobolev space), 它表示一系列“波动不大”的函数的集合。

例 9.1.6. (函数的非参数估计) 令 $X_1, \dots, X_n \sim F$ 。假定要在仅假设 μ 存在的条件下估计 $\mu = T(F) = \int x dF(x)$, 通常情况下, 任何 F 的函数称为统计泛函, 其他一些统计泛函的例子有方差 $T(F) = \int x^2 dF(X) - (\int x dF(X))^2$, 中位数 $T(F) = F^{-1}(1/2)$ 。

例 9.1.7. (回归, 预测与分类) 假设有成对的观察值 $(X_1, Y_1), \dots, (X_n, Y_n)$, 如 X_i 表示第 i 个患者的血压, Y_i 表示该患者能活多久。 X 称为预测变量或回归变量或特征变量或自变量, Y 称为输出变量或响应变量或因变量。称 $r(x) = \mathbb{E}(Y | X = x)$ 为回归函数。如果假设 $r \in \mathfrak{F}$, 其中, \mathfrak{F} 是有限维的, 如直线集, 则称模型为参数回归模型, 如果假设 $r \in \mathfrak{F}$, 其中, \mathfrak{F} 不是有限维的, 则称模型为非参数回归模型。对一个新的病人, 根据他的 X 值去预测 Y 称为预测, 如果 Y 是离散的 (例如, 生或死), 则称为分类, 如果目标是估计函数 r , 则称为回归估计或曲线估计, 有时回归模型也记为

$$Y = r(X) + \varepsilon$$

其中, $\mathbb{E}(\varepsilon) = 0$, 通常也用这种方式来描述回归模型, 为进一步理解, 定义 $\varepsilon = Y - r(X)$, 则 $Y = Y + r(X) - r(X) = r(X) + \varepsilon$ 。此外, $\mathbb{E}(\varepsilon) = \mathbb{E}\mathbb{E}(\varepsilon | X) = \mathbb{E}(\mathbb{E}(Y - r(X)) | X) = \mathbb{E}(\mathbb{E}(Y | X) - r(X)) = \mathbb{E}(r(X) - r(X)) = 0$ 。

统计推断的基本概念

有了总体分布的参数和非参数模型表示, 接下来我们需要对总体进行参数和非参数统计推断:

- 对于参数模型: 我们的任务是, 如何根据已知的信息, 在分布族 $\{f(x; \theta) : \theta \in \Theta\}$ 中选定一个分布作为总体的分布。用统计的语言就是根据已知信息估计出未知参数 θ 的值。这样, 就能使总体的分布从不明确变成明确的了。□
- 对于非参数模型: 我们的任务是, 在没有关于总体累积分布函数 F 或者概率函数 $f(x)$ 的任何假设或者仅有一般性假设 (例如连续分布、对称分布等) 的前提下, 作出一个累积分布函数 F 或者一个概率函数 $f(x)$ 的一致估计。

参数模型推断属于参数统计问题, 非参数模型推断属于非参数统计问题。例如, 检验“两个总体有相同分布”这个假设, 若假定两总体的分布分别为正态分布 $N(\mu_1, \sigma_2)$ 和 $N(\mu_2, \sigma_2)$, 则问题只涉及三个实参数 μ_1, μ_2, σ_2 , 这是参数统计问题。若只假定两总体的分布为连续, 此外一无所知, 问题涉及的分布不能用有限个实参数刻画, 则这是非参数统计问题。

本课程我们主要讨论参数统计推断; 对于非参数统计推断, 我们主要限定在对概率密度函数或回归函数的非参数估计讨论。

研究统计推断的方法有多种, 最主要的有两大类方法: 古典的频率统计推断、贝叶斯推断。许多统计推断问题可以归入以下三类: 点估计、置信区间、假设检验。下面对这三类问题做一个简单的介绍。

点估计 点估计是指对感兴趣的某一单个提供“最优估计”。感兴趣的点可以是参数模型、分布函数 F 、概率函数 f 和回归函数 r 等中的某一参数, 或者可以是对某些随机变量的未来值 Y 的预测。

假设总体 X 的分布函数的形式已知, 但它的一个或多个参数未知, 借助于总体 X 的一个样本来估计总体未知参数的值称为参数的点估计。记 θ 的点估计为 $\hat{\theta}$ 或 $\hat{\theta}_n$ 。注意 θ 是固定且未知的, 而估计 $\hat{\theta}$ 依赖于数据, 所以它是随机的。设 X_1, X_2, \dots, X_n 是取自总体 X 的一个样本。我们构造一个统计量 $\hat{\theta} = u(X_1, X_2, \dots, X_n)$ 作为参数 θ 的估计, 称这个统计量 $\hat{\theta}$ 为参数 θ 的一个估计量。若 (x_1, x_2, \dots, x_n) 是样本 (X_1, X_2, \dots, X_n) 的一组观测值, 则 $\hat{\theta} = u(x_1, x_2, \dots, x_n)$ 就是 $\hat{\theta}$ 的一个点估计值或简称估计值。

如果分布簇中含有 k 个未知参数, 即 $\{f(x; \theta_1, \dots, \theta_k) : (\theta_1, \dots, \theta_k) \in \Theta\}$, 则需要构造 k 个统计量 $\hat{\theta}_1 = u_1(X_1, X_2, \dots, X_n), \dots, \hat{\theta}_k = u_k(X_1, X_2, \dots, X_n)$ 分别作为 $\theta_1, \dots, \theta_k$ 的估计量。这种问题又称为多元参数的点估计问题。

由上面看到, 要求参数 θ 的估计值, 必须先构造一个估计量, 然后把样本观测值代入估计量得到一个估计值。寻找估计量是寻找参数 θ 的估计值的一个前提, 绝不是针对一组具体的观测值去定一个估计值, 因为对于一组观测值所决定的估计值是不可能知道这个估计的好坏的, 而

必须从总体出发，在大量重复取样的情况下，才能评价估计的好坏。研究估计的好坏，一个很自然的想法是研究参数 θ 的一个估计量与参数 θ 的真值之间的偏差在统计意义上是大还是小呢？在统计意义上，偏差小的估计量可以认为是较好的估计量。在下一讲介绍估计量的构造方法之前，下面我们先简要介绍估计量的评价。

估计量的评价 对于同一参数，用不同的估计方法求出的估计量可能不相同，原则上任何统计量都可以作为未知参数的估计量，一个自然的问题是，采用哪一个估计量为好？这涉及用什么样的标准来评价估计量的问题。主要有三个评价标准：无偏性、有效性、相合性。

设 X_1, X_2, \dots, X_n 是取自总体的一个样本， $\theta \in \Theta$ 是包含在总体 X 的分布中的待估参数，这里 Θ 是 θ 的取值范围。

无偏性：若估计量 $\hat{\theta} = u(X_1, X_2, \dots, X_n)$ 的数学期望 $E(\hat{\theta})$ 存在，且对任意的 $\theta \in \Theta$ 有 $E(\hat{\theta}) = \theta$ ，则称 $\hat{\theta}$ 是 θ 的无偏估计量。估计量的偏差定义为 $bias(\hat{\theta}) = E(\hat{\theta}) - \theta$ ，称为以 $\hat{\theta}$ 作为 θ 的估计的系统误差。无偏估计的实际意义就是无系统误差。

有效性(风险小)：设 $\hat{\theta}_1 = u(X_1, X_2, \dots, X_n)$ 和 $\hat{\theta}_2 = v(X_1, X_2, \dots, X_n)$ 都是 θ 的无偏估计量，若对于任意的 $\theta \in \Theta$ 有 $D(\hat{\theta}_1) \leq D(\hat{\theta}_2)$ ，且至少对于某一个 $\theta \in \Theta$ 上式中的不等号成立，则称 $\hat{\theta}_1$ 较 $\hat{\theta}_2$ 有效，也即方差越小，越有效。

相合性(一致性)：设 $\hat{\theta} = u(X_1, X_2, \dots, X_n)$ 为参数 θ 的估计量，若对任意的 $\theta \in \Theta$ ，当 $n \rightarrow \infty$ 时 $\hat{\theta} = u(X_1, X_2, \dots, X_n)$ 依概率收敛于 θ ，则称 $\hat{\theta}$ 为 θ 的相合估计量。即对任意的 $\theta \in \Theta$ 都满足：对于任意的 $\varepsilon > 0$ ，有 $\lim_{n \rightarrow \infty} P\{|\hat{\theta} - \theta| \geq \varepsilon\} = 0$ ，则称 $\hat{\theta}$ 为 θ 的相合估计量。

前面讲的无偏性和有效性都是在样本容量 n 固定的前提下提出的，我们自然希望随着样本容量的增大，也即收集的数据越来越多的时候，一个估计量的值稳定于待估参数的真值。因此这个时候我们就需要考虑相合性的要求。

均方误差评价：点估计的质量好坏有时也用均方误差，即 MSE 来评价，均方误差定义为

$$MSE = \mathbb{E}_\theta (\hat{\theta} - \theta)^2$$

要注意 $E_\theta(\cdot)$ 是关于如下分布的期望而不是关于 θ 分布的平均，该分布由数据得来，这是因为误差来自于从总体中随机采样导致的，具体如下：

$$f(x_1, \dots, x_n; \theta) = \prod_{i=1}^n f(x_i; \theta)$$

定理 9.1.2. 均方误差 MSE 可写成如下形式：

$$MSE = bias^2(\hat{\theta}) + D(\hat{\theta})$$

证明. 令 $\bar{\theta} = \mathbb{E}_\theta(\hat{\theta})$, 则

$$\begin{aligned}\mathbb{E}_\theta(\hat{\theta} - \theta)^2 &= \mathbb{E}_\theta(\hat{\theta} - \bar{\theta} + \bar{\theta} - \theta)^2 \\ &= \mathbb{E}_\theta(\hat{\theta} - \bar{\theta})^2 + 2(\bar{\theta} - \theta)\mathbb{E}_\theta(\hat{\theta} - \bar{\theta}) + \mathbb{E}_\theta(\bar{\theta} - \theta)^2 \\ &= (\bar{\theta} - \theta)^2 + \mathbb{E}_\theta(\hat{\theta} - \bar{\theta})^2 \\ &= \text{bias}^2(\hat{\theta}) + D(\hat{\theta}).\end{aligned}$$

□

推导过程中用到了如下事实: $\mathbb{E}_\theta(\hat{\theta} - \bar{\theta}) = \bar{\theta} - \bar{\theta} = 0$

$\hat{\theta}$ 的分布称为抽样分布, $\hat{\theta}$ 的标准差称为标准误差, 记为 se ,

$$se = se(\hat{\theta}) = \sqrt{D(\hat{\theta})}$$

通常标准误差依赖于未知分布 F , 在另外一些情况下, se 是未知量, 但通常去估计它, 估计的标准误差记为 \hat{se} 。

例 9.1.8. 在抛硬币的试验中, 令 $X_1, \dots, X_n \sim \text{Bernoulli}(p)$, $\hat{p}_n = n^{-1} \sum X_i$, 则:

- (1) $\mathbb{E}(\hat{p}_n) = n^{-1} \sum \mathbb{E}(X_i) = p$, 所以 \hat{p}_n 是无偏的。
- (2) 标准误差为 $se = \sqrt{D(\hat{p}_n)} = \sqrt{p(1-p)/n}$, 估计的标准误差为 $\hat{se} = \sqrt{\hat{p}(1-\hat{p})/n}$ 。
- (3) 因为 $\mathbb{E}(\hat{p}_n) = p$, 所以 $\text{bias} = p - p = 0$, $se = \sqrt{p(1-p)/n} \xrightarrow{P} 0$, 因此 $\hat{p}_n \xrightarrow{P} p$, 即 \hat{p}_n 是一致估计量, 是相合的。

今后将要遇到的许多估计量都近似服从正态分布。

定义 9.1.4. 如果 $\frac{\hat{\theta}_n - \theta}{se} \rightsquigarrow N(0, 1)$, 则称估计量 $\hat{\theta}_n$ 是渐进正态的。

那么怎样构造估计量呢? 参数的点估计方法包括:

- 矩估计 (频率学派)
- 极大似然估计 (频率学派)
- 极大后验估计 (贝叶斯学派)
- 贝叶斯估计 (贝叶斯学派)

这些我们将在下一节内容展开介绍。

置信区间 对于未知参数 θ , 除了求出它的点估计 $\hat{\theta}$ 外, 我们还希望估计出一个范围, 并希望知道这个范围包含参数 θ 真值的可信程度。这样的范围通常以区间的形式给出, 同时还给出此区间包含参数 θ 真值的可信程度。这种形式的估计称为区间估计, 这样的区间即所谓的置信区间。

设总体 X 的分布函数 $F(x; \theta)$ 含有一个未知参数 $\theta, \theta \in \Theta$ (Θ 是 θ 可能取值的范围)，对于给定值 α ($0 < \alpha < 1$)，若由来自 X 的样本 X_1, X_2, \dots, X_n 确定的两个统计量 $\underline{\theta} = \underline{\theta}(X_1, X_2, \dots, X_n)$ 和 $\bar{\theta} = \bar{\theta}(X_1, X_2, \dots, X_n)$ ($\underline{\theta} < \bar{\theta}$)，对于任意 $\theta < \Theta$ 满足

$$P\{\underline{\theta}(X_1, X_2, \dots, X_n) < \theta < \bar{\theta}(X_1, X_2, \dots, X_n)\} \geq 1 - \alpha \quad (9.2)$$

则称随机区间 $C_n = (\underline{\theta}, \bar{\theta})$ 是 θ 的置信水平为 $1 - \alpha$ 的置信区间， $\underline{\theta}$ 和 $\bar{\theta}$ 分别称为置信水平为 $1 - \alpha$ 的双侧置信区间的置信下限和置信上限， $1 - \alpha$ 称为置信水平。

当 X 是连续型随机变量时，对于给定的 α ，我们总是按要求 $P\{\underline{\theta} < \theta < \bar{\theta}\} = 1 - \alpha$ 求出置信区间。而当 X 是离散型随机变量时，对于给定的 α ，常常找不到区间 $(\underline{\theta}, \bar{\theta})$ 使得 $P\{\underline{\theta} < \theta < \bar{\theta}\}$ 恰为 $1 - \alpha$ 。此时我们去找区间 $(\underline{\theta}, \bar{\theta})$ 使得 $P\{\underline{\theta} < \theta < \bar{\theta}\}$ 至少为 $1 - \alpha$ ，且尽可能地接近 $1 - \alpha$ 。 C_n 是随机的而 θ 是固定的。如果 θ 是向量，则用置信集（例如球面或者椭圆面）代替置信区间。

式(9.2)的含义如下：若反复抽样多次（各次得到的样本的容量相等，都是 n ）。每个样本值确定一个区间 $(\underline{\theta}, \bar{\theta})$ ，每个这样的区间要么包含 θ 的真值，要么不包含 θ 的真值。按伯努利大数定律，在这么多的区间中，包含 θ 真值的约占 $100(1 - \alpha)\%$ ，不包含 θ 真值的约仅占 $100\alpha\%$ 。例如，若 $\alpha = 0.05$ ，反复抽样 1000 次，则得到的 1000 个区间中不包含 θ 真值的约仅为 50 个。该解释并没有错误，但用处不大，因为人们很少反复地多次重复相同的试验。第 1 次，对于参数 θ ，收集到数据并建立了 95% 的置信区间，第 2 次，对于参数 θ_2 ，收集到数据并建立了 95% 的置信区间，第 3 次，对于参数 θ_3 。收集到数据并建立了 95% 的置信区间，继续这一过程，对一系列不相关参数 $\theta_1, \theta_2, \dots$ 建立置信区间，则这些置信区间有 95% 的概率覆盖真实的参数值，这一解释不需要反复地重复同一试验。

例 9.1.9. 报纸每天都会报道民意调查的结果。例如，报道称“有 83% 的公众对飞行员随身配备真枪飞行的做法表示赞同”，通常你还会看到诸如这样的陈述“该调查有 95% 的概率在 4 个百分点的范围内变动”。意思就是赞同飞行员随身配备真枪飞行的做法的人数所占的比例 p 的 95% 的置信区间是 $83\% \pm 4\%$ ，如果以后都按这种方式建立置信区间，则有 95% 的区间将包括真实的参数值，即使每天估计的量不同（不同的民意测验），这一结论也是正确的。

例 9.1.10. 在抛硬币的试验中，令 $C_n = (\hat{p}_n - \varepsilon, \hat{p}_n + \varepsilon)$ ，其中 $\varepsilon^2 = \log(2/\alpha)/(2n)$ ，由霍夫丁不等式得，对任意 p

$$\mathbb{P}(p \in C_n) \geq 1 - \alpha$$

因此， C_n 是 $1 - \alpha$ 置信区间。

假设检验 统计推断的另一类重要问题是假设检验问题。在总体的分布函数完全未知或只知其形式，但不知其参数的情况。为了推断总体的某些未知特性，提出某些关于总体的假设。例如，提出总体服从泊松分布的假设，又如对于正态总体提出数学期望等于 μ_0 的假设等。我们要根据样本对所提出的假设作出是接受，还是拒绝的决策。假设检验是作出这一决策的过程。

在假设检验中, 从缺省理论, 即原假设开始, 通过数据是否提供显著性证据来支持拒绝该假设, 如果不能拒绝, 则保留原假设。

例 9.1.11. (检验硬币是否均匀) 令 $X_1, \dots, X_n \sim \text{Bernoulli}(p)$ 为 n 次独立的硬币投掷结果, 假设要检验硬币是否均匀, 令 H_0 表示硬币是均匀的假设, H 表示硬币不是均匀的假设, H_0 称为原假设, H_1 称为备择假设, 可以将假设写成

$$H_0: p = 1/2 \text{ 对比 } H_1: p \neq 1/2.$$

如果 $T = |\hat{p}_n - \frac{1}{2}|$ 的值很大, 则有理由拒绝 H_0 , 当详细讨论假设检验的时候, 将会确定出拒绝 H_0 的精确 T 值。

参数估计和非参数估计也分别称为参数推断和非参数推断。除了这两种推断, 统计推断还包括: 独立性推断、因果推断。本节对参数与非参数模型中的参数密度估计、函数 (CDF) 的非参数估计、非参数密度估计, 统计推断的基本概念如点估计、置信区间、假设检验等做了简单的介绍, 关于概率函数的估计方法, 特别是参数估计和非参数估计的方法, 没有涉及, 将在下一讲进行详细介绍!

9.2 概率密度函数的估计

9.2.1 概率密度估计引入

统计机器学习方法按其使用的技巧大致可以分为两类: 核方法和贝叶斯学习。其中贝叶斯学习, 又称为贝叶斯推断, 其主要思想是在概率模型的学习和推理中, 利用贝叶斯定理, 计算在给定数据条件下模型的条件概率, 即后验概率, 并应用这个原理进行模型的估计以及对数据的预测。

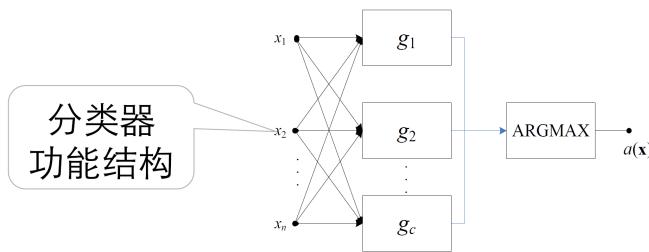


图 9.2: 贝叶斯推断

在设计贝叶斯分类器时, 需要已知先验概率和类条件概率密度, 并按一定的决策规则确定判别函数和决策面。但实际工作中, 类条件概率密度常常是未知的。以鸢尾花分类任务为例, 尽管在数据集中各种鸢尾花所占的比例是相等的, 但是在实际中人们有可能根据所采集鸢尾花的

地域大致判断其类别。例如，在中国的吉林省更有可能采集的是山鸢尾，而不是维吉尼亚鸢尾。这样，结合实际经验使得我们有可能推断先验概率 $P(\omega_i)$ 。另外，通常我们可能获得各类鸢尾花的样本数据，但不能给出类条件概率密度 $p(x|\omega_i)$ 。这就需要我们从所采集的各类鸢尾花样本中去估计出山鸢尾和维吉尼亚鸢尾类条件概率密度。

由上可知，在实际中，我们能收集到的是有限数目的样本，而未知的可能是先验概率 $P(\omega_i)$ 和类条件概率密度 $p(x|\omega_i)$ 。任务是利用样本集设计分类器，一个很自然的想法是把分类器设计分成两步：

1. 利用样本集估计先验概率 $P(\omega_i)$ 和类条件概率密度 $p(x|\omega_i)$ ，分别记为 $\hat{P}(\omega_i)$ 和 $\hat{p}(x|\omega_i)$
2. 然后利用估计的概率密度设计贝叶斯分类器。

这就是基于样本的两步 Bayes 分类器设计。

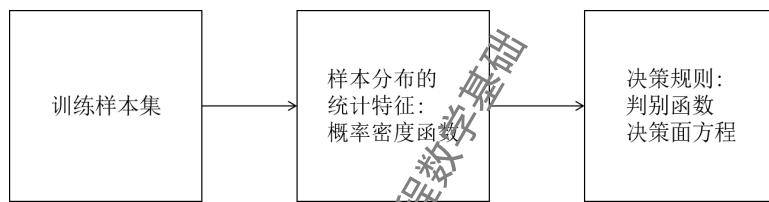


图 9.3: 基于样本的两步 Bayes 分类器设计

理想情况，希望当样本数 $N \rightarrow \infty$ 时，该方法设计的分类器收敛于理论上的最优解。为此目标，则需要

$$\begin{aligned} P(\omega_i) &\xrightarrow{N \rightarrow \infty} P(\omega_i) \\ \hat{p}(\mathbf{x} | \omega_i) &\xrightarrow{N \rightarrow \infty} p(\mathbf{x} | \omega_i) \end{aligned}$$

本节主要内容：研究如何利用样本集估计概率密度函数。

类先验概率 $P(\omega_i)$ 的估计

首先我们来看如何对类先验概率 $P(\omega_i)$ 的估计。这种估计主要依靠经验，用训练数据中各类出现的频率来估计。使用频率估计概率的优点是使估计具有无偏性、相合性且使估计的收敛速度快。

类条件概率密度函数估计

概率密度函数可是满足下面条件的任何函数：

$$p(x) \geq 0, \quad \int p(x) dx = 1$$

估计该函数的方法有参数估计和非参数估计：

- **参数估计**: 已知类条件概率密度函数的形式, 而参数未知。如已知样本总体符合正态分布, 而正态分布的参数均值和方差未知。根据是否已知样本所在类别, 参数估计又分为监督参数估计和非监督参数估计。根据是否使用参数的先验信息, 参数估计的方法又分为基于频率的参数估计方法和基于贝叶斯的参数估计方法。在接下来的三小节中, 我们将主要介绍参数估计方法。
- **非参数估计**: 已知样本所在类别, 未知类条件概率密度函数的形式, 要求直接推断函数本身。一些常见典型的分布形式并不能总是满足实际需求。因此, 有些实际问题需要根据样本推断总体分布。在本节的最后一小节中将会涉及概率密度函数的非参数估计。

在表9.1中, 我们对概率密度函数的参数和非参数估计, 及其所使用的方法进行更直观的展示。

表 9.1: 概率密度估计概览

	样本所属类别	总体概率密度函数的形式	推断	解决方法
监督参数估计	已知	已知	参数	矩估计、极大似然估计; 贝叶斯估计
非监督参数估计	未知	未知		
非参数估计	已知	未知	概率密度函数	直方图、核密度估计、 k 近邻法

9.2.2 基于频率观点的参数估计方法

本节开始主要讨论参数估计的问题。前面我们已经提及如果对待求解的概率密度分布模型形式已经了解或做了合适的假设, 那么问题便转化为模型参数估计的问题。考虑参数模型, 其形式为:

$$\mathfrak{F} = \{f(x; \theta) : \theta \in \Theta\},$$

其中 $\Theta \subset \mathbb{R}^k$ 为参数空间, $\theta = (\theta_1, \dots, \theta_k)$ 为参数。因此推断问题简化为 θ 的参数估计问题。

在统计学习时经常可能被问及的问题: 怎样能确定生成数据的分布是某种参数模型呢? 实际上非常困难。但学习参数模型的方法仍然非常有用。首先, 根据有些案例的背景知识可以假定数据近似服从某种参数模型。例如, 根据先验可以知道交通事故发生的次数服从近似泊松分布。其次, 参数模型的推断为理解非参方法提供了背景知识。

本小节主要介绍基于频率观点(经典学派)的参数估计方法。Pearson、Fisher、Neyman 是该流派的创始人, 该流派的观点是: 概率就是频率, 参数就是参数, 不会变化。该流派的主要方法包括矩估计、极大似然估计等。

矩估计

矩估计的基本思想 讨论的第一种参数估计方法为矩估计法。矩估计的基本思想：上一节提到，由大数定理，我们知道样本矩依概率收敛于总体矩，样本矩的连续函数依概率收敛于总体矩的连续函数；又在许多分布中它们所含的参数都是矩的函数，例如正态分布 $N(\mu, \sigma^2)$ 中的参数 μ 和 σ^2 就是这个分布的一阶原点矩和二阶中心矩；因此很自然的会想到用样本矩作为相应总体矩的一种估计量。这种方法称为矩估计法。矩估计不是最优的，但是最容易计算，它们也可以作为其他需要循环几次的算法的初始值。接下来我们介绍矩估计的具体做法。

假设总体 X 的概率函数为 $f(x; \theta_1, \dots, \theta_k)$ ，其中 $\theta = (\theta_1, \dots, \theta_k)$ 为待估参数， X_1, \dots, X_n 是来自 X 的样本。对于 $1 \leq j \leq k$ ，定义总体 X 的 j 阶矩为

$$\alpha_j \equiv \alpha_j(\theta) = E_\theta(X^j) = \int x^j f(x; \theta_1, \dots, \theta_k) dx,$$

一般来说，它们都是 $\theta_1, \dots, \theta_k$ 的函数。而样本 X_1, \dots, X_n 的 j 阶样本矩定义为

$$\hat{\alpha}_j = \frac{1}{n} \sum_{i=1}^n X_i^j.$$

定义 9.2.1. θ 的矩估计定义为 $\hat{\theta}_n$ ，使得

$$\begin{aligned} \alpha_1(\hat{\theta}_n) &= \hat{\alpha}_1, \\ \alpha_2(\hat{\theta}_n) &= \hat{\alpha}_2, \\ &\vdots \\ \alpha_k(\hat{\theta}_n) &= \hat{\alpha}_k. \end{aligned} \tag{9.3}$$

公式(9.3)定义了带有 k 个未知参数 $\hat{\theta} = (\hat{\theta}_1, \dots, \hat{\theta}_k)$ 的 k 个方程的方程组，从中可以解出参数 $\theta = (\theta_1, \dots, \theta_k)$ 的矩估计量。

例 9.2.1. 令 $X_1, \dots, X_n \stackrel{i.i.d.}{\sim} Bernoulli(p)$. 则 $\alpha_1 = E_p(X) = p$ 且 $\hat{\alpha}_1 = n^{-1} \sum_{i=1}^n X_i$. 让它们相等可以得到估计值

$$\hat{p}_n = \frac{1}{n} \sum_{i=1}^n X_i.$$

例 9.2.2. 令 $X_1, \dots, X_n \stackrel{i.i.d.}{\sim} N(\mu, \sigma^2)$. 则 $\alpha_1 = E_\theta(X) = \mu$ 且 $\alpha_2 = E_\theta(X^2) = D(X) + (E_\theta(X))^2 = \sigma^2 + \mu^2$. 现在需要解下述方程：

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n X_i,$$

$$\hat{\sigma}^2 + \hat{\mu}^2 = \frac{1}{n} \sum_{i=1}^n X_i^2.$$

这是由两个方程组成含有两个未知参数的方程组。它的解为

$$\hat{\mu} = \bar{X},$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2.$$

矩估计的性质 矩估计有一些比较好的性质:

定理 9.2.1. 令 $\hat{\theta}$ 表示矩估计. 在适当的条件下, 下述成立:

1. 矩估计 $\hat{\theta}$ 以接近概率 1 存在.
2. 这个估计是相合的: $\hat{\theta} \xrightarrow{P} \theta$.
3. 这个估计是渐进正态的:

$$\sqrt{n} (\hat{\theta} - \theta) \rightsquigarrow N(0, \Sigma)$$

其中,

$$\Sigma = g E_{\theta} (YY^T) g^T,$$

$$Y = (X, X^2, \dots, X^k)^T, g = (g_1, \dots, g_k), g_j = \partial \alpha_j^{-1}(\theta) / \partial \theta$$

定理最后一条可以用于求标准差和置信区间、然而, 有比这更加简单的方法: Bootstrap 方法. 这里不再额外展开讨论。

极大似然估计

极大似然估计的基本思想 极大似然估计法 (Maximum Likelihood Estimator, MLE) 是求估计的另一种方法, 在参数模型中, 它是最常用的参数估计方法. 极大似然估计法最早由高斯 (G.F. Gauss) 提出, 后来为费歇 (R.A. Fisher) 在 1912 年重新提出, 并且证明了这个方法的一些性质. 极大似然估计这一名称也是费歇给的. 它是建立在极大似然原理的基础上的一个统计方法。

极大似然原理的直观想法是: 一个随机试验如有若干个可能的结果 A, B, C, \dots . 若在一次试验中, 结果 A 出现, 则一般认为试验条件对 A 有利, 也即 A 出现的概率最大. 下面我们来介绍该方法。

首先, 设假设条件为:

- 参数 θ 是确定而未知的量 (非随机量);
- 样本集按类别分开, 样本集 $X^j (j = 1, \dots, c)$ 中样本都是从概率函数为 $f(x|\omega_j)$ 的总体中独立抽取出来 (独立同分布, i.i.d.);
- 概率函数 $f(x|\omega_j)$ 的形式已知, 仅其参数 θ 未知, 用 $f(x|\omega_j; \theta)$ 表示, 对于同类别可简化为 $f(x; \theta)$;
- 各类样本只包含了本类分布的信息。

那么, 在上述假设前提下, 可以分别处理 c 个独立的问题. 独立地按照概率密度 $f(x; \theta)$ 抽取样本集 $X = \{X_1, X_2, \dots, X_n\}$, 用样本集 X 估计未知参数 θ . 下面我们只考虑一类样本的极大似然估计。

定义 9.2.2. 设 X_1, X_2, \dots, X_n 为取自具有概率函数 $\{f(x; \theta) : \theta \in \Theta\}$ 的总体 X 的一个样本。样本 X_1, X_2, \dots, X_n 的联合概率函数在 X_i 取已知观测值 $x_i, i = 1, \dots, n$ 时的值

$$L(\theta) = L(x_1, x_2, \dots, x_n; \theta) = \prod_{i=1}^n f(x_i; \theta), \quad \theta \in \Theta,$$

称作这个样本的似然函数。对数似然函数为

$$H(\theta) = \ln L(\theta) = \sum_{i=1}^n \ln f(x_i; \theta)$$

定义 9.2.3. 极大似然估计 MLE, 记为 $\hat{\theta}$, 是使得 $L(\theta)$ 最大的 θ 值, 也即满足

$$L(x_1, x_2, \dots, x_n; \hat{\theta}) = \max_{\theta \in \Theta} L(x_1, x_2, \dots, x_n; \theta),$$

称 $\hat{\theta}(x_1, x_2, \dots, x_n)$ 为参数 θ 的极大似然估计值, 其相应的统计量 $\hat{\theta}(X_1, X_2, \dots, X_n)$ 为参数 θ 的极大似然估计量。

极大似然估计是典型的频率学派观点, 它的基本意义是: 待估计参数 θ 是客观存在的, 只是未知而已, 当 θ 满足 $\theta = \hat{\theta}$ 时, 该组观测样本 $(X_1, X_2, \dots, X_n) = (x_1, x_2, \dots, x_n)$ 更容易被观测到, 我们就说 $\hat{\theta}$ 是 θ 的极大似然估计值。也即, 估计值 $\hat{\theta}$ 使得事件发生的可能性最大。

接下来, 我们考虑如何去求解极大似然估计。这里由于 $\ln x$ 是单调递增函数, 使得似然函数 $L(\theta)$ 最大的 $\hat{\theta}$ 也使得对数似然函数 $H(\theta)$ 最大, 因此有时我们只要求对数似然最大即可!

$$\begin{aligned} \hat{\theta} &= \arg \max_{\theta} L(\theta) = \arg \max_{\theta} H(\theta) \\ &= \arg \max_{\theta} \sum_{i=1}^n \ln f(x_i; \theta) \end{aligned}$$

其中, $\hat{\theta}$ 是极大似然解的必要条件是: 函数梯度 (导数) 为 0。

常见分布的极大似然估计 本节主要介绍了如何使用极大似然估计法对一些常见分布的参数进行估计。

例 9.2.3. 假设 $X \sim N(\mu, \sigma^2)$, μ, σ^2 为未知参数, x_1, x_2, \dots, x_n 是来自 X 的一个样本值, 求 μ, σ^2 的极大似然估计量。

解. X 的概率密度为

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left[-\frac{1}{2} \left(\frac{x - \mu}{\sigma} \right)^2 \right],$$

此时对应的参数 $\theta = (\theta_1, \theta_2) \triangleq (\mu, \sigma^2)$, 则对数似然函数为

$$H(\theta) = \ln L(\theta) = -\frac{n}{2} \ln 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2$$

由 $\nabla_{\theta} H(\theta) = 0$ 得

$$\begin{cases} \sum_{i=1}^n \frac{1}{\sigma^2} (x_i - \mu) = 0 \\ -\sum_{i=1}^n \frac{1}{\sigma^2} + \sum_{i=1}^n \frac{(x_i - \mu)^2}{(\sigma^2)^2} = 0 \end{cases}$$

求解方程组得

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2$$

因此得 μ, σ^2 的极大似然估计量分别为

$$\hat{\mu} = \bar{X}, \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2.$$

根据该例的计算结果可以验证，均值的极大似然估计量是无偏的

$$E(\hat{\mu}) = E\left(\frac{1}{n} \sum_{i=1}^n x_i\right) = \mu$$

方差的极大似然估计量不是无偏的

$$E(\hat{\sigma}^2) = E\left(\frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2\right) = \frac{n-1}{n} \sigma^2 \neq \sigma^2$$

方差的无偏估计为样本方差

$$\frac{1}{n-1} \sum_{i=1}^n (x_i - \hat{\mu})^2.$$

由上述分析可知：(1) 正态总体均值的极大似然估计即为学习样本的算术平均；(2) 正态总体方差的极大似然估计与样本的方差不同，当 n 较大的时候，二者的差别不大。

例 9.2.4. 类似地，可以求解具有 n 个特征的多元正态分布的极大似然估计。

解.

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i$$

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})(x_i - \hat{\mu})^T$$

由上述估计值可以看出对于多元正态分布：(1) μ 的估计即为学习样本的算术平均；(2) 估计的协方差矩阵是矩阵 $(x_i - \hat{\mu})(x_i - \hat{\mu})^T$ 的算术平均 ($n \times n$ 阵列， $n \times n$ 个值)。

例 9.2.5. 令 $X_1, \dots, X_n \stackrel{i.i.d.}{\sim} \text{Uniform}(0, \theta)$ ，其概率密度函数为 $f(x; \theta) = \begin{cases} 1/\theta, & 0 \leq x \leq \theta \\ 0, & \text{其他} \end{cases}$ ，

求未知参数 θ 的极大似然估计量。

解. 考虑一个固定的 θ 值。假设对于某一个 i , 有 $\theta < x_i$, 则 $f(x_i; \theta) = 0$, 因此

$$L(\theta) = \prod_i f(x_i; \theta) = 0.$$

对任意的 $x_i > \theta$, 则 $L(\theta) = 0$. 因此, 如果 $\theta < x_{(n)}$, 就有 $L(\theta) = 0$, 这里 $x_{(n)} = \max\{x_1, \dots, x_n\}$ 。现在考虑任意 $\theta \geq x_{(n)}$ 。对每一个 x_i , 有 $f(x_i; \theta) = 1/\theta$, 所以

$$L(\theta) = \prod_i f(x_i; \theta) = \theta^{-n}.$$

总之

$$L(\theta) = \begin{cases} \left(\frac{1}{\theta}\right)^n, & \theta \geq x_{(n)} \\ 0, & \theta < x_{(n)} \end{cases}$$

在区间 $[x_{(n)}, \infty)$ 上, $L(\theta)$ 是严格递减的。因此 $\hat{\theta} = x_{(n)}$, 其相应的估计量为 $\hat{\theta} = X_{(n)}$ 。

极大似然函数的性质 在某些条件下, 极大似然估计有很多性质:

- 极大似然估计是相合估计: $\hat{\theta} \xrightarrow{P} \theta_*$, 其中, θ_* 表示参数 θ 的真实值。
- 极大似然估计是同变估计: 如果 $\hat{\theta}$ 是 θ 的极大似然估计, 则 $g(\hat{\theta})$ 是 $g(\theta)$ 的极大似然估计。
- 极大似然估计是渐近正态的: $(\hat{\theta} - \theta_*)/\hat{s}\hat{e} \sim N(0, 1)$ 。同时, 估计的标准差 $\hat{s}\hat{e}$ 可以解出来。
- 极大似然估计是渐近最优或有效的: 这表示, 在所有表现优异的估计中, 极大似然估计的方差最小, 至少对大样本这肯定成立。

非监督极大似然估计 非监督极大似然估计问题表现为: 假设样本集 $X = \{X_1, \dots, X_N\}$ 中的样本分属于 c 个类别, 但未知各样本所属类别; 已知各类先验概率 $P(\omega_i), i = 1, \dots, c$ (有时也可未知, 一起估计) 和类条件概率密度形式 $p(x | \omega_i, \theta_i), i = 1, \dots, c$ 。需估计未知的 c 个参数向量 $\theta_1, \theta_2, \dots, \theta_c$ 。

为求解该问题, 首先给出相应的混合密度函数及其似然函数: 混合密度函数: 分量密度的线性组合

$$p(x | \theta) = \sum_{i=1}^c \underbrace{p(x | \omega_i, \theta_i)}_{\text{分量密度}} \underbrace{P(\omega_i)}_{\text{混合参数}}$$

似然函数和对数似然函数:

$$L(\theta) = p(X | \theta) = \prod_{i=1}^N p(x_i | \theta)$$

$$H(\theta) = \ln[L(\theta)] = \sum_{i=1}^N \ln p(x_i | \theta)$$

则可进一步求得其极大似然估计:

$$\hat{\theta} = \arg \max_{\theta \in \Theta} \prod_{k=1}^N p(x_k | \theta) = \arg \max_{\theta \in \Theta} \sum_{k=1}^N \ln p(x_k | \theta)$$

则问题转化成求解下列微分方程组:

$$\begin{aligned}\nabla_{\theta_i} H(\theta) &= \sum_{k=1}^N \frac{1}{p(x_k | \theta)} \nabla_{\theta_i} \left[\sum_{j=1}^c p(x_k | \omega_j, \theta_j) P(\omega_j) \right] \\ &= \sum_{k=1}^N \frac{1}{p(x_k | \theta)} \nabla_{\theta_i} [p(x_k | \omega_i, \theta_i) P(\omega_i)] \quad (\text{设 } \theta_i, \theta_j \text{ 独立}) \\ &= \sum_{k=1}^N P(\omega_i | x_k, \theta_i) \nabla_{\theta_i} \ln p(x_k | \omega_i, \theta_i)\end{aligned}$$

其中后验概率 $P(\omega_i | x_k, \theta_i) = \frac{p(x_k | \omega_i, \theta_i) P(\omega_i)}{p(x_k | \theta)}$

9.2.3 贝叶斯推断

经典统计学更多关注频率推断, 到目前为止我们讲述的方法都是频率论的估计方法(或经典方法)。频率论方法的观点基于下面的假设:

- 概率指的是相对频率, 是真实世界的客观属性。
- 参数是固定的未知常数。由于参数不会波动, 因此不能对其进行概率描述。
- 统计过程应该具有定义良好的频率稳定性。如: 一个 95% 的置信区间应覆盖参数真实值至少 95% 的频率。

频率推断是根据样本信息对总体分布或总体的特征数进行推断, 这里用到两种信息:

1. **总体信息:** 总体分布提供的信息。
2. **样本信息:** 抽取样本所得观测值提供的信息。

但是基于频率的估计方法, 有其局限性: 基于频率估计方法的优良性, 在大样本情况下有其理论上的保障。但在许多情况下, 我们无法重复大量的试验, 无法得到大量的试验结果, 只能得到少量的试验结果。因此在小样本情况下, 传统方法是否优良, 是没有保障的。设总体 X 的概率密度函数为 $f(x; \theta)$, X_1, \dots, X_n 为来自总体 X 的样本, 当 n 较大时, 用传统方法估计 θ , 估计很准确。 n 较小时, 特别当 $n = 1$ 或 $n = 2$ 时, 传统估计不是很可靠。因而, 人们一直在寻求小样本情况下的优良估计方法。解决的思路是: 用过去的经验, 用人们过去对 θ 的了解(或部分了解), 给出 θ 较可靠、较切合实际的估计。

过去的看法、记忆或经验, 常常支配着我们对事物的判断(估计、评判)。例如, 裁判打分, 对知名运动员的评分总是要偏高, 对新手的评分总是偏低。又如, 对知名产品进行抽样检查, 抽取了少量样品, 如果全合格, 便终止抽样, 得出产品合格的结论。但对一个新厂生产的产品, 则不会依据少量的抽样检查下结论。因此, 对 θ 的了解形成的一种先验信息, 对估计可能是有帮助的。而在传统频率估计方法中, 反映在数学上, 我们把 X 当作随机变量, 而把 θ 当作确定的未知常量, 可能不一定恰当。 $f(x; \theta)$ 提供的知识与信息是关于 X 的, 它反映了 X 取值的规律。

性, 但它没有反映 θ 的变化规律。所以, 我们可以引入反映 θ 变化规律的信息, 这种引入参数 θ 的先验信息的推断思想, 就是贝叶斯推断。

贝叶斯推断的基本思想

机器学习和数据挖掘更偏爱贝叶斯推断。贝叶斯方法基于下面的假设:

- 概率描述的是主观信念的程度, 而不是频率。这样除了对从随机变化产生的数据进行概率描述外, 我们还可以对其他事物进行概率描述。
- 可以对各个参数进行概率描述, 即使它们是固定的常数。
- 为参数生成一个概率分布来对它们进行推导, 点估计和区间估计可以从这些分布得到。

贝叶斯推断除了利用前面频率推断中提到的总体信息和样本信息, 还使用第三种信息:

3. **先验信息**: 即是抽样 (试验) 之前有关统计问题的一些信息。人们在试验之前对要做的问题在经验上和资料上总是有所了解的, 这些信息对统计推断是有益的。一般说来, 先验信息来源于经验和历史资料。先验信息在日常生活和工作中是很重要的。

基于上述三种信息进行统计推断的统计学称为贝叶斯统计学。该学派的带头人是 Bayes, Laplace, Jeffreys, Robbins。Bayes, Thomas(1702—1761)是英国数学家。1702 年生于伦敦, 1761 年 4 月 17 日卒于坦布里奇韦尔斯。他长期担任坦布里奇韦尔斯地方教堂的牧师。1742 年, 贝叶斯被选为英国皇家学会会员。如今在概率、数理统计学中以贝叶斯姓氏命名的有: 贝叶斯公式、贝叶斯风险、贝叶斯决策函数、贝叶斯决策规则、贝叶斯估计量、贝叶斯方法、贝叶斯统计等等。该学派的主要观点是: 频率不只是概率, 存在主观概率, 和实体概率可转化, 参数作为随机变量。其采用的主要方法包括后验均值、贝叶斯估计、最大后验估计等。

贝叶斯统计学与经典统计学的差别就在于是否利用先验信息。贝叶斯统计通过对先验信息的收集、挖掘和加工, 使它数量化, 形成先验分布, 参加到统计推断中来, 以提高统计推断的质量。忽视先验信息的利用, 有时是一种浪费, 有时还会导出不合理的结论。上述利用先验信息形成先验分布的前提是: 总体分布的参数是随机的, 但有一定的分布规律; 参数是某一常数, 但无法知道。目标是充分利用参数的先验信息对未知参数作出更准确的估计。所以贝叶斯方法就是把未知参数视为具有已知分布的随机变量, 将先验信息数字化并利用的一种方法。

贝叶斯推断

贝叶斯推断通常的做法如下:

1. 选择一个概率密度函数 $\pi(\theta)$, 用来表示在观察到数据之前我们对参数的信念 (经验判断), 称之为先验分布。

2. 选择一个统计模型 $q(\mathbf{x}; \theta)$ (在此处记为 $q(\mathbf{x} | \theta)$), 用来反映在给定参数 θ 情况下我们对 \mathbf{x} 的信念 (经验判断)。
3. 当得到观察数据 X_1, X_2, \dots, X_n 后, 改进我们原来的信念 (经验判断), 并且计算后验分布 $h(\theta | X_1, \dots, X_n)$, 从后验分布中得到点估计和区间估计。

下面我们具体地介绍如何实现这些做法。

先验分布和贝叶斯参数统计模型 Bayes 学派认为: 样本分布族中的参数 θ 不是常量, 而是随机变量, 它可能取各种不同的值, 取各种不同值的概率分布 $\Pi(\theta)$ 也是确定的。

例 9.2.6. 考虑某厂每天产品的次品率 p 。关于 p 的算法: 在当天的生产的产品中, 进行产品全检, 计算其次品率 p ; 或者抽取部分产品, 估计其次品率 p 。从当天看, p 是一个单纯的未知常数。但从较长的时间看, 每天都有一个 p 值, 其值因随机因素的作用, 会产生波动, 当天的 p 值可合理地视为随机变量 p 的一个可能值。如果我们有相当长一个时期的检验记录, 则可以相当精确地定出 p 的概率分布。

形式上, 把参数 θ 看成一个随机变量, 并给出 θ 的概率分布 $\Pi(\theta)$, 或概率密度 $\pi(\theta)$, 这个分布 $\Pi(\theta)$ 在抽样前就给出了, 把它称为 θ 的先验分布。

定义 9.2.4. 参数 θ 的参数空间 Θ 上的一个概率分布称为 θ 的先验分布, 其密度族记为

$$\{\pi(\theta) : \theta \in \Theta\}.$$

样本 $\mathbf{X} = (X_1, X_2, \dots, X_n)$ 的条件密度函数族

$$\{q(\mathbf{x} | \theta) : \theta \in \Theta\}, (\mathbf{x} = (x_1, x_2, \dots, x_n))$$

称为样本分布族; 先验分布与样本分布族构成 贝叶斯参数统计模型。

实际上, 有时, 把参数 θ 看成随机变量有其合理性, 但把所有未知参数都视为随机变量则牵强。例如, 要估计某铁矿的含铁量 p , 把 p 看成随机变量, 就要设想这个铁矿是无穷多“类似”铁矿的一个样本, 这是不自然的, 不如把 p 看做一个独立的未知常数。此外, 虽然把参数 θ 看成随机变量有其合理性, 但人们的先验知识没有确切到能用概率分布把 θ 表达出来。于是, 引出了一系列先验分布的确定方法。

后验分布与贝叶斯公式的密度函数形式 设 θ 为随机变量, 总体 X 依赖于参数 θ 的概率密度函数为 $f(x; \theta)$, 在贝叶斯统计中记为 $f(x | \theta)$, 它表示在随机变量 θ 取某个给定值时总体的条件概率密度函数; 根据参数 θ 的先验信息可确定先验分布, θ 的先验概率密度函数记为 $\pi(\theta)$; 从贝叶斯观点看, 来自总体 X 的样本 $\mathbf{X} = (X_1, X_2, \dots, X_n)$ 的产生分两步进行: 首先从先验分布 $\pi(\theta)$ 产生一个样本 θ_0 , 然后从 $f(\mathbf{x} | \theta_0)$ 中产生一组样本, 这时样本的联合条件概率函数为

$$q(\mathbf{x} | \theta_0) = \prod_{i=1}^n f(x_i | \theta_0)$$

其中 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 为样本观测值, 这个分布综合了总体信息和样本信息。

由于 θ_0 是未知的, 它是按先验分布 $\pi(\theta)$ 产生的。为把先验信息综合进去, 不能只考虑 θ_0 , 对 θ 的其它值发生的可能性也要加以考虑, 故要用 $\pi(\theta)$ 进行综合。这样一来, 样本 X_1, \dots, X_n 和参数 θ 的联合概率密度函数为:

$$g(\mathbf{x}; \theta) = q(\mathbf{x} | \theta) \pi(\theta), \quad (9.4)$$

其中 $q(\mathbf{x} | \theta) = \prod_{i=1}^n f(x_i | \theta)$, 这个联合分布把总体信息、样本信息和先验信息三种可用信息都综合进去了。

在没有样本信息时, 人们只能依据先验分布对 θ 作出推断。在有了样本观察值 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 之后, 则应依据 $g(\mathbf{x}; \theta)$ 对 θ 作出推断。由于联合概率密度函数等于条件概率密度函数和边际概率密度函数的乘积, 也即

$$g(\mathbf{x}; \theta) = h(\theta | \mathbf{x}) m(\mathbf{x}), \quad (9.5)$$

其中 $m(\mathbf{x}) = \int_{\Theta} g(\mathbf{x}; \theta) d\theta = \int_{\Theta} q(\mathbf{x} | \theta) \pi(\theta) d\theta$ 是 \mathbf{x} 的联合边际概率密度函数, 它与 θ 无关, 不含 θ 的任何信息。

联立公式(9.4)和(9.5)可知, 能用来对 θ 作出推断的仅是条件概率密度函数 $h(\theta | \mathbf{x})$, 它的计算公式是

$$h(\theta | \mathbf{x}) = \frac{g(\mathbf{x}; \theta)}{m(\mathbf{x})} = \frac{q(\mathbf{x} | \theta) \cdot \pi(\theta)}{\int_{\Theta} q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta}$$

其相应的条件分布称为 θ 的后验分布, 它集中了总体、样本和先验中有关 θ 的一切信息。后验分布 $h(\theta | \mathbf{x})$ 的计算公式就是用密度函数表示的贝叶斯公式。它是用总体和样本对先验分布 $\pi(\theta)$ 作调整的结果, 贝叶斯统计的一切推断都基于后验分布进行。

定义 9.2.5. 在 $\mathbf{X} = \mathbf{x}$ 的条件下, θ 的条件分布 (或条件概率密度) 称为后验分布, 后验分布由后验密度函数 $\{h(\theta | \mathbf{x}) : \theta \in \Theta\}$ 描述, 其计算公式为:

$$h(\theta | \mathbf{x}) = \frac{g(\mathbf{x}; \theta)}{m(\mathbf{x})} = \frac{q(\mathbf{x} | \theta) \pi(\theta)}{\int_{\Theta} q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta}$$

值得说明的是, θ 的先验分布 $\pi(\theta)$ 概括了我们在试验前关于 θ 的认识。经过试验得到样本观测值 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 后, 我们的认识起了变化, $h(\theta | \mathbf{x})$ 是我们重新认识 θ 的基础和根据。特别地, 由于 $\int_{\Theta} q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta$ 不依赖于 θ , 在计算 θ 的后验分布中仅起到一个规范化因子的作用, 若把 $\int_{\Theta} q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta$ 省略, 可将后验密度函数改写为如下等价形式:

$$h(\theta | \mathbf{x}) \propto q(\mathbf{x} | \theta) \pi(\theta)$$

其中符号 “ \propto ” 表示两边仅相差一个不依赖于 θ 的常数因子。 $q(\mathbf{x} | \theta) \pi(\theta)$ 称为后验分布 $h(\theta | \mathbf{x})$ 的核。

例 9.2.7. 设 p 是某厂产品的合格率。在抽样前, 我们可以假定 p 在区间 $(0, 1)$ 之间是均匀分布的 (对 p 的认识不多, 不妨设 p 取各种值的可能性一样大)。抽取了 n 个产品检查发现有 m 个废品, 这时我们会修正对 p 的认识, p 仍有可能取 $(0, 1)$ 区间的任何值, 但机会大小不处处一样了, 在 $p = \frac{m}{n}$ 这一点附近的可能性最大, 而接近 0, 1 处则可能性很小。

现在我们通过下面的例子来说明后验分布的计算：

例 9.2.8. 假设总体 $X \sim N(\mu, \sigma^2)$ (σ^2 已知), X_1, X_2, \dots, X_n 为来自总体 X 的样本, 由过去的经验和知识, 我们可以确定 μ 的取值范围在区间 $[-\mu_0, \mu_0]$ 之内, 但无法得到关于 μ 的更多的信息, 按同等无知的原则, 我们假定 $\mu \sim U[-\mu_0, \mu_0]$, 其概率密度为:

$$\pi(\mu) = \begin{cases} \frac{1}{2\mu_0} & |\mu| \leq \mu_0 \\ 0 & |\mu| > \mu_0 \end{cases}$$

样本分布函数族为

$$q(\mathbf{x} | \mu) = \frac{1}{\sigma^n (2\pi)^{n/2}} \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2 \right]$$

于是

$$h(\mu | \mathbf{x}) = \frac{q(\mathbf{x} | \mu) \cdot \pi(\mu)}{m(\mathbf{x})} = \frac{q(\mathbf{x} | \mu) \cdot \pi(\mu)}{\int_{-\infty}^{+\infty} q(\mathbf{x} | \mu) \cdot \pi(\mu) d\mu}$$
$$\propto \begin{cases} \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2 \right] & |\mu| \leq \mu_0 \\ 0 & |\mu| > \mu_0 \end{cases}$$

消去分子分母中的公共部分, 得:

$$h(\mu | \mathbf{x}) = \begin{cases} \frac{1}{c(\mathbf{x})} \cdot \exp \left[-\frac{n}{2\sigma^2} \cdot (\bar{x} - \mu)^2 \right] & |\mu| \leq \mu_0 \\ 0 & |\mu| > \mu_0 \end{cases}$$

其中

$$c(\mathbf{x}) = \int_{-\mu_0}^{+\mu_0} \exp \left[-\frac{n}{2\sigma^2} \cdot (\bar{x} - \mu)^2 \right] d\mu$$

贝叶斯推断的原则 对贝叶斯统计而言, 样本 X_1, \dots, X_n 的唯一作用在于把对 θ 的认识由先验分布转化成后验分布。因此, 贝叶斯统计推断的原则就是:

- 对参数 θ 所作的任何推断(参数估计、假设检验等)必须基于且只能基于 θ 的后验分布, 即后验密度函数族 $\{h(\theta | \mathbf{x}) : \theta \in \Theta\}$ 。
- 一旦由样本 X_1, \dots, X_n 算出 θ 的后验分布, 就设想我们除了这一后验分布外, 其余的东西(样本值、样本分布、先验分布)全忘记了。这时, 对 θ 的推断的唯一凭借就是这一后验分布。

传统的统计推断原则许多都不能用了。如无偏性原则, $\hat{\theta} = T(X_1, \dots, X_n)$, $E(\hat{\theta}) = \theta$, 完全利用样本(样本的函数)在进行推断没有利用后验分布, 不符合贝叶斯统计推断的原则。

先验分布的确定 贝叶斯推断涉及先验分布的确定。确定先验分布 $\pi(\theta)$ 的方法主要有客观法、主观概率法、同等无知原则以及共轭分布法。

(1) 客观法

以前的资料积累较多, 对 θ 的先验分布能作出较准确的统计或估计。在这种情况下, 分布的确定没有掺杂多少人的主观因素, 故称之为客观法。如果能用客观法确定 θ 的先验分布 $\pi(\theta)$, 对贝叶斯学派持否定态度的统计学者也不反对用贝叶斯方法去作数据处理。在不少情况下, 以往积累的资料并不是直接给出了参数在当时的取值, 而只是一种估计。例如, 某厂产品的废品率, 不可能是全检(可能是破坏性检验)。有些资料不是直接关于 θ 取值分布的记录, 但我们可以利用这些资料对 θ 的先验分布作出经验性的推断。

(2) 主观概率法

按照 Bayes 学派的说法, 这是一种通过“自我反省”去确定先验分布的方法。就是说, 对参数 θ 取某某值的可能性多大, 通过思考, 觉得该如何, 而定下一个值。主观先验分布反映了个人以往对 θ 的了解, 包括经验知识和理论知识, 其中有部分可能是通过他人获取的, 也可能是他人对 θ 的了解。对过去的经验和知识, 必须经过组织和整理。这样提出的先验分布, 在主观上是正确的, 但不能保证合乎某种客观标准。

(3) 同等无知原则

这一原则称为 Bayes 假定。以产品的废品率为例, 当我们对 p 一无所知时, 我们只好先验地认为, p 以同等机会取 $(0, 1)$ 内各种值, 因而以 $(0, 1)$ 内均匀分布 $U(0, 1)$ 作为 p 的先验分布。这一先验分布称为无信息先验分布。这一原则会出现矛盾: 如果我们对 p 无知, 对 p^3 也同样无知。按同等无知原则, 可以取 $U(0, 1)$ 作为 p^3 的分布, 但这时 p 的分布就不是 $U(0, 1)$ 了。

(4) 共轭分布方法

H.Raiffa, R.Schlaifer 提出了先验分布应取共轭分布才合适。

定义 9.2.6. 设样本分布族为 $\{q(\mathbf{x} \mid \theta) : \theta \in \Theta\}$, 若先验分布 $\pi(\theta)$ 与后验分布 $h(\theta \mid \mathbf{x})$ 属于同一分布类型, 则先验分布 $\pi(\theta)$ 称为关于 $q(\mathbf{x} \mid \theta)$ 的共轭分布。确切地说, 若 \mathcal{F} 为 θ 的一个密度函数族, 若任取 $\pi(\theta) \in \mathcal{F}$, 得到样本观测值 \mathbf{x} 后, 由 $\pi(\theta)$ 及 $q(\mathbf{x} \mid \theta)$ 确定的后验密度函数 $h(\theta \mid \mathbf{x}) \in \mathcal{F}$, 则称 \mathcal{F} 是关于 $\{q(\mathbf{x} \mid \theta) : \theta \in \Theta\}$ 的共轭先验分布族, 或称为参数 θ 的共轭先验分布族。

选取共轭先验分布有如下好处: 符合直观, 先验分布和后验分布应该是相同形式的; 可以给出后验分布的解析形式; 可以形成一个先验链, 即现在的后验分布可以作为下一次计算的先验分布, 如果形式相同, 就可以形成一个链条。

例 9.2.9. 设 X_1, X_2, \dots, X_n 是来自正态分布 $N(\theta, \sigma^2)$ 的一个样本, 其中 θ 已知, 求方差 σ^2 的共轭先验分布。

解. $(X_1, X_2, \dots, X_n)^T$ 的联合条件概率函数为

$$q(\mathbf{x} | \sigma^2) = \frac{1}{(\sqrt{2\pi}\sigma)^n} \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \theta)^2 \right] \propto \left(\frac{1}{\sigma^2} \right)^{n/2} \exp \left[-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \theta)^2 \right]$$

所以 σ^2 的共轭先验分布是

$$\pi(\sigma^2) = \frac{\lambda^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2} \right)^{\alpha+1} \exp \left[-\frac{\lambda}{\sigma^2} \right],$$

为倒 Γ 分布。

计算共轭先验分布的方法: 由 $h(\theta | \mathbf{x}) = \pi(\theta)q(\mathbf{x} | \theta)/m(\mathbf{x})$, 其中 $m(\mathbf{x})$ 不依赖于 θ , 先求出 $q(\mathbf{x} | \theta)$, 再选取与 $q(\mathbf{x} | \theta)$ 具有相同形式的分布作为先验分布, 就是共轭分布。

常见分布的共轭先验分布有:

- 二项分布 $b(n, \theta)$ 中的成功概率 θ 的共轭先验分布是贝塔分布 $Be(a, b)$;
- 泊松分布 $P(\theta)$ 中的均值 θ 的共轭先验分布是伽玛分布 $\Gamma(\alpha, \lambda)$;
- 指数分布中均值的倒数的共轭先验分布是伽玛分布 $\Gamma(\alpha, \lambda)$;
- 在方差已知时, 正态均值 θ 的共轭先验分布是正态分布 $N(\mu, \tau^2)$;
- 在均值已知时, 正态方差 σ^2 的共轭先验分布是倒伽玛分布 (Inverse Gamma distribution) $IG(\alpha, \lambda)$ 。

基于后验分布的点估计和区间估计 首先, 可以通过集中后验的中心得到点估计. 通常, 使用后验的均值或众数. 后验均值为

$$\bar{\theta}_n = \int \theta h(\theta | \mathbf{x}) d\theta = \frac{\int \theta q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta}{\int_{\Theta} q(\mathbf{x} | \theta) \cdot \pi(\theta) d\theta}.$$

也可以得到贝叶斯区间估计. 可以求出 a 和 b , 使得

$$\int_{-\infty}^a h(\theta | \mathbf{x}) d\theta = \int_b^{\infty} h(\theta | \mathbf{x}) d\theta = \alpha/2.$$

令 $C = (a, b)$. 则

$$\mathbb{P}(\theta \in C | \mathbf{x}) = \int_a^b h(\theta | \mathbf{x}) d\theta = 1 - \alpha,$$

所以 C 是 $1 - \alpha$ 后验区间。

例 9.2.10. 令 $X_1, \dots, X_n \stackrel{i.i.d.}{\sim} \text{Bernoulli}(p)$. 假设把均匀分布 $\pi(p) = 1$ 或 Beta 分布作为 p 的先验分布. 考虑其后验估计和贝叶斯区间估计。

解. 根据贝叶斯定理, 后验的形式为

$$h(p | \mathbf{x}) \propto \pi(p)q(\mathbf{x} | p) = p^s(1-p)^{n-s} = p^{s+1-1}(1-p)^{n-s+1-1},$$

其中, $s = \sum_{i=1}^n x_i$ 是成功的次数. 回想起如果一个随机变量服从参数为 α 和 β 的 Beta 分布, 其密度为

$$f(p; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}.$$

可以求出 p 的后验分布是参数为 $s + 1$ 和 $n - s + 1$ 的 Beta 分布, 即

$$h(p | \mathbf{x}) = \frac{\Gamma(n+2)}{\Gamma(s+1)\Gamma(n-s+1)} p^{(s+1)-1} (1-p)^{(n-s+1)-1}.$$

将其记为

$$p | \mathbf{x} \sim \text{Beta}(s+1, n-s+1).$$

注意到并没有真正做积分 $\int_p q(\mathbf{x} | p) \cdot \pi(p) dp$ 就求出了归一化系数. 由于 Beta (α, β) 的均值为 $\alpha/(\alpha + \beta)$, 所以贝叶斯估计为

$$\bar{p} = \frac{s+1}{n+2}$$

可以把这个估计改写为

$$\bar{p} = \lambda_n \hat{p} + (1 - \lambda_n) \tilde{p},$$

其中, \hat{p} 是极大似然估计, $\tilde{p} = 1/2$ 是先验均值, $\lambda_n = n/(n+2) \approx 1$. 通过计算 $\int_a^b h(p | \mathbf{x}) dp = 0.95$ 得到 a 和 b , 从而得到一个 95% 的后验区间.

假设先验分布不是用均匀分布, 而是用 $p \sim \text{Beta}(\alpha, \beta)$. 如果重复上述的计算, 可以得到 $p | \mathbf{x} \sim \text{Beta}(\alpha+s, \beta+n-s)$. 扁平先验 (均匀分布) 仅仅是 $\alpha = \beta = 1$ 时的一个特例. 后验均值为

$$\bar{p} = \frac{\alpha+s}{\alpha+\beta+n} = \left(\frac{n}{\alpha+\beta+n} \right) \hat{p} + \left(\frac{\alpha+\beta}{\alpha+\beta+n} \right) p_0,$$

其中, $p_0 = \alpha/(\alpha+\beta)$ 是先验均值.

9.2.4 统计决策与贝叶斯估计

前面已经考虑了几种点估计, 如矩估计、极大似然估计和后验均值. 事实上, 还有许多其它的估计方法. 如何选择这些方法呢? 可以通过决策理论来评价这些方法, 统计决策理论是比较统计过程的正规理论. 20 世纪 40 年代末, 瓦尔德 (Wald) 建立了统计决策理论. 1950 年发表了《统计决策函数》一书, 系统地论述了他的理论. 这一理论对参数估计、区间估计、假设检验等统计问题在统计决策的观点下统一处理. 它通过将统计问题表示成数学最优化问题的解, 引进了各种优良性准则. 这个理论的一些基本观点现在已经不同程度地渗透到各个统计分支, 对数理统计学的发展产生了重大的影响. 统计决策理论是二战后数理统计学发展的重大事件.

统计决策与统计推断是既有联系又有区别的. 统计决策问题要考虑到决策的损失, 而统计推断问题, 一般是指解决一类统计问题的方法, 但不考虑决策的损失问题. 如在参数的点估计中, 矩估计与极大似然估计是进行点估计的统计方法, 属于统计推断的范围, 而讨论点估计的优良性, 则与统计决策有关. 统计决策是统计推断研究的深化. 统计决策方法可以作为产生优良统计推断的手段. 贝叶斯 (Bayes) 估计是贝叶斯统计的主要部分, 它是运用统计决策理论研究参数估计问题. 本小节, 我们先简要介绍统计决策理论, 然后引出贝叶斯估计.

统计决策的基本概念

统计决策三要素 统计决策问题的三个要素是：样本空间和样本分布族、决策（行动）空间、损失函数。

(1) 样本空间和分布族

设总体 X 的分布函数为 $F(x; \theta)$, $\theta \in \Theta$ 是未知参数, Θ 是参数空间。若设 X_1, \dots, X_n 是来自总体 X 的一个样本, 则样本所有可能值组成的集合称为样本空间, 记为 \mathcal{X} 。由于 X_i 的分布函数为 $F(x_i; \theta)$, $i = 1, \dots, n$, 则 X_1, \dots, X_n 的联合分布函数为

$$F(x; \theta) = \prod_{i=1}^n F(x_i; \theta), \theta \in \Theta$$

若记

$$F^* = \left\{ \prod_{i=1}^n F(x_i; \theta) \mid \theta \in \Theta \right\},$$

则称 F^* 为样本 X_1, \dots, X_n 的概率分布族, 称样本分布族。

所谓给定了一个参数统计模型, 实质上是指给定了样本空间和样本分布族。

(2) 决策空间（判决空间）

对于一个统计问题, 如参数的点估计、区间估计以及参数的假设检验问题, 我们常常要给予适当的回答。对参数的点估计, 一个具体的估计值就是一个回答。在假设检验中, 它是一个决定, 即是接受还是拒绝原假设。在统计决策中, 每个具体的回答称为一个决策（或行动）, 一个统计问题中可能选取的全部决策组成的集合称为决策空间, 记为 \mathcal{A} 。一个决策空间至少应有两个决策, 假如 \mathcal{A} 中只含有一个决策, 那么人们就无需选择, 从而也形成不了一个统计决策问题。本书讨论的决策主要集中在点估计。

(3) 损失函数

统计决策的一个基本观点是假设: 每采取一个决策, 必然有一定的后果, 所采取的决策不同, 后果就不同。这种后果必须以某种方式通过损失函数的形式表示出来。这样, 每一决策有优劣之分。统计决策的一个基本思想就是把决策的优劣性以数量的形式表现出来, 其方法是引入一个依赖参数值 $\theta \in \Theta$ 和决策 $d \in \mathcal{A}$ 的二元实值非负 $L(\theta, d) \geq 0$, 称之为损失函数, 它表示当参数真值为 θ 而采取决策 d 时所造成的损失, 决策越正确, 损失就越小。由于在统计问题中人们总是利用样本对总体进行推断, 所以误差是不可避免的, 因而总会带来损失, 这就是损失函数定义为非负函数的原因。

对于不同的统计问题, 可以选取不同的损失函数, 对于参数的点估计问题常见的损失函数有如下几种:

- 线性损失:

$$L(\theta, d) = \begin{cases} k_0(\theta - d) & \text{当 } \theta \geq d \text{ 时,} \\ k_1(d - \theta) & \text{当 } \theta < d \text{ 时} \end{cases}$$

其中 k_0, k_1 是两个非负常数。

- 绝对损失: $L(\theta, d) = |\theta - d|$
- 平方损失: $L(\theta, d) = (\theta - d)^2$
- L_p 损失: $L(\theta, d) = |\theta - d|^p$
- 0-1 损失:

$$L(\theta, d) = \begin{cases} 0 & \text{当 } \theta = d \text{ 时,} \\ 1 & \text{当 } \theta \neq d \text{ 时} \end{cases}$$

- 凸损失: $L(\theta, d) = \lambda(\theta)W(|\theta - d|)$
- 多元二次损失: $L(\theta, d) = (d - \theta)^T A(d - \theta)$
- Kullback-Leibler 损失:

$$L(\theta, d) = \int \log \left(\frac{f(x; \theta)}{f(x; d)} \right) f(x; \theta) dx$$

统计决策函数及其风险函数 假设给定一个统计决策问题的三要素: 样本空间 \mathcal{X} 和样本分布族, 决策空间 \mathcal{A} 及损失函数 $L(\theta, d)$ 。我们的问题是对每一样本观测值 $\mathbf{x} = (x_1, \dots, x_n)$, 即对每一个 $\mathbf{x} \in \mathcal{X}$, 有一个确定的法则, 在 \mathcal{A} 中选取一个决策 d 。这样一个对应关系是定义在样本空间 \mathcal{X} 上, 取值于决策空间 \mathcal{A} 的一个函数 (即由 \mathcal{X} 到 \mathcal{A} 的一个映射) $d(\mathbf{x})$ 。

定义 9.2.7. 定义在样本空间 \mathcal{X} 上, 取值于决策空间 \mathcal{A} 内的函数 $d(\mathbf{x})$, 称为统计决策函数, 简称决策函数。

易见, 决策函数 $d(\mathbf{x})$ 就是一个行动方案, 当有了样本观测值 \mathbf{x} 后, 按既定的方案采取行动 (决策) $d(\mathbf{x})$; 因此 $d(\mathbf{x})$ 本质上就是一个统计量。决策函数 $d(\mathbf{x})$ 就是所给定的统计决策问题的一个解。

给定一个统计决策问题, 若使用决策函数 $d(\mathbf{x})$, 由于样本 $\mathbf{X} = (X_1, \dots, X_n)$ 是随机的, 从而 $d(\mathbf{X})$ 也是随机的, 因而 $L(\theta, d(\mathbf{X}))$ 也是随机的, 它是样本 \mathbf{X} 的函数。当样本取不同的值 \mathbf{x} , 决策 $d(\mathbf{x})$ 可能不同, 所以损失函数值 $L(\theta, d)$ 也不同。因此为了判断一个决策的好坏, 一般从总体上来评价比较决策函数, 也即用 $L(\theta, d(\mathbf{X}))$ 关于样本的数学期望, 代表了取决策函数 $d(\mathbf{x})$ 时在概率意义上的平均风险或损失, 这个平均风险就是统计决策理论中非常重要的风险函数的概念。

定义 9.2.8. 设样本空间和样本分布族分别为 \mathcal{X} 和 $F^* = \{F(\mathbf{x}; \theta) : \theta \in \Theta\}$, 决策空间为 \mathcal{A} , 损失函数为 $L(\theta, d)(\theta \in \Theta, d \in \mathcal{A})$, 则统计决策函数 $d(\mathbf{x})$ 的风险函数定义为

$$R(\theta, d) = E_\theta[L(\theta, d(\mathbf{x}))] = \int_{\mathcal{X}} L(\theta, d(\mathbf{x})) dF(\mathbf{x}; \theta),$$

$R(\theta, d)$ 是 θ 的函数, 当 θ 取定值时, $R(\theta, d)$ 称为决策函数 $d(\mathbf{x})$ 在参数值 θ 时的风险。

风险函数 $R(\theta, d)$ 是统计决策问题当采取决策函数 d 时统计意义上的平均损失。风险函数是 Wald 统计决策理论的基本概念。评价一个决策函数 d 的依据就是其风险函数。

点估计问题在各种损失函数下的风险:

- 设决策函数为 $d(\mathbf{x}) = \hat{\theta}(\mathbf{x})$ (即 θ 的点估计), 则对应平方损失函数的风险函数为

$$R(\theta, \hat{\theta}) = E_\theta[(\theta - \hat{\theta}(\mathbf{x}))^2] = \int_{\mathcal{X}} (\theta - \hat{\theta}(\mathbf{x}))^2 dF(\mathbf{x}; \theta),$$

即为估计量 $\hat{\theta}$ 的均方误差;

- 对应绝对值损失函数的风险函数为

$$R(\theta, \hat{\theta}) = E_\theta[|\theta - \hat{\theta}(\mathbf{x})|] = \int_{\mathcal{X}} |\theta - \hat{\theta}(\mathbf{x})| dF(\mathbf{x}; \theta),$$

即为估计量 $\hat{\theta}$ 的平均绝对误差。

此外, 针对区间估计和假设检验问题, 也可以给出在相应损失函数下的风险函数。

Wald 理论引进统计决策函数及其风险函数, 将各类统计推断问题用统一的观点与方法处理。若要论及统计推断方法的优良性, 必须考虑统计推断所采取决策的损失, 即要考虑风险函数。按照 Wald 的理论, 风险函数越小, 决策函数就越优良。但是对于给定的决策函数, 风险函数仍是参数 θ 的函数。所以, 两个决策函数风险大小的比较, 情况比较复杂, 因此就产生了种种优良性准则。

定义 9.2.9. 设 d_1, d_2 是统计问题中的两个决策函数, 若其风险函数满足不等式

$$R(\theta, d_1) \leq R(\theta, d_2), \forall \theta \in \Theta$$

则称决策函数 d_1 优于 d_2 。若不等号严格成立, 则称决策函数 d_1 一致优于 d_2 ; 若 $R(\theta, d_1) = R(\theta, d_2), \forall \theta \in \Theta$, 则称 d_1, d_2 等价。

定义 9.2.10. 设 $D = \{d(\mathbf{x})\}$ 是一切定义在样本空间 \mathcal{X} 上, 取值于决策空间 \mathcal{A} 上的决策函数全体, 若存在一个决策函数 $d^* \in D$, 使对任意一个 $d(\mathbf{x}) \in \mathcal{A}$ 都有

$$R(\theta, d^*) \leq R(\theta, d), \forall \theta \in \Theta$$

则称 d^* 为一致最小风险决策函数, 或一致最优决策函数。

例 9.2.11. 设总体 $X \sim N(\mu, 1)$, $\mu \in (-\infty, +\infty)$, 估计未知参数 μ 。

解. 选取损失函数为: $L(\mu, d) = (d - \mu)^2$ 则对 μ 的任一估计 $d(\mathbf{X})$, 风险函数为

$$R(\mu, d) = E_\mu[L(\mu, d)] = E_\mu(d - \mu)^2$$

若要求 $d(\mathbf{X})$ 是无偏估计, 即 $E_\mu(d(\mathbf{X})) = \mu$, 则风险函数为:

$$R(\mu, d) = E_\mu(d - Ed)^2 = D_\mu(d(\mathbf{X}))$$

即风险函数为估计量 $d(\mathbf{X})$ 的方差。

若取 $d(\mathbf{X}) = \bar{\mathbf{X}}$, 则 $R(\mu, d) = D\bar{\mathbf{X}} = \frac{1}{n}$

若取 $d(\mathbf{X}) = \mathbf{X}_1$, 则 $R(\mu, d) = D\mathbf{X}_1 = 1$

显然, 当 $n > 1$ 时, 后者的风险比前者大, $\bar{\mathbf{X}}$ 优于 \mathbf{X}_1 。

例 9.2.12. 设总体 $X \sim P(x; \lambda)$, 估计未知参数 λ 。

解. 选取损失函数为:

$$L(\lambda, d) = (d - \lambda)^2$$

则对 λ 的任一估计 $d(\mathbf{X})$, 风险函数为

$$R(\lambda, d) = E_\lambda[L(\lambda, d)] = E_\lambda(d - \lambda)^2$$

若要 $d(\mathbf{X})$ 是无偏估计, 即 $E_\lambda(d(\mathbf{X})) = \lambda$, 则风险函数为:

$$R(\lambda, d) = E_\lambda(d - Ed)^2 = D_\lambda(d(\mathbf{X}))$$

若取 $d(\mathbf{X}) = \bar{\mathbf{X}}$, 则 $R(\lambda, d) = D\bar{\mathbf{X}} = \frac{\lambda}{n}$

若取 $d(\mathbf{X}) = \mathbf{X}_1$, 则 $R(\lambda, d) = D\mathbf{X}_1 = \lambda$

显然, 当 $n > 1$ 时, 风险不同。

在一个统计决策问题中, 可供选择的决策函数往往很多, 自然希望寻找使风险最小的决策函数, 然而在这种意义下的最优决策函数往往是不存在的。这是因为:

1. 风险函数是二元函数, 极值往往不存在或不唯一
2. 在某个区间内的逐点比较不现实 (麻烦)
3. 对应不同参数的, 同一决策函数, 风险值不相等
4. 由统计规律的特性决定不能点点比较

因此必须由一个整体指标来代替点点比较。要解决这个问题, 就要建立一个整体指标的比较准则。贝叶斯方法通过引进先验分布把两个风险函数的点点比较转化为用一个整体指标的比较来代替, 从而可以决定优劣。贝叶斯风险和最大风险就是采用这种形式定义的。

贝叶斯估计

贝叶斯风险 首先我们考虑贝叶斯风险。

定义 9.2.11. 对于给定的统计决策问题, 设 $d(\mathbf{x})$ 为该统计问题的决策函数, 又设 $d(\mathbf{x})$ 的风险函数为 $R(\theta, d)(\theta \in \Theta)$, 设参数 θ 的先验密度函数为 $\pi(\theta)(\theta \in \Theta)$ 。风险函数 $R(\theta, d)$ 的关于 θ 的期望

$$R_B(d) = E(R(\theta, d)) = \int_{\Theta} R(\theta, d) \pi(\theta) d\theta$$

称为决策函数 $d(\mathbf{x})$ 在给定先验分布 $\pi(\theta)$ 下的贝叶斯风险, 简称 $d(\mathbf{x})$ 的贝叶斯风险。

使贝叶斯风险最小的决策规则称为贝叶斯规则, 相应的决策函数称为贝叶斯规则或贝叶斯决策。

定义 9.2.12. 对于给定的统计决策问题, 设总体 X 的分布函数 $F(x, \theta)$ 中参数 θ 为随机变量, $\pi(\theta)$ 为 θ 的先验分布, 若在决策函数类 \mathcal{A} 中存在一个决策函数 $d^*(\mathbf{x})$, 使得

$$R_B(d^*) = \inf_{d \in \mathcal{A}} R_B(d),$$

则称 $d^*(\mathbf{x})$ 是统计决策问题在先验分布 $\pi(\theta)$ 下的贝叶斯规则或贝叶斯决策。

当总体 X 和 θ 都是连续型随机变量时, 设 X 的概率密度函数为 $f(x; \theta)$, θ 的先验概率密度函数为 $\pi(\theta)$, 记 $q(\mathbf{x} | \theta) = \prod_{i=1}^n f(x_i; \theta)$ (此即为样本密度), 则

$$\begin{aligned} R_B(d) &= \int_{\Theta} R(\theta, d) \pi(\theta) d\theta \\ &= \int_{\Theta} \int_{\mathcal{X}} L(\theta, d(\mathbf{x})) q(\mathbf{x} | \theta) \pi(\theta) d\mathbf{x} d\theta \\ &= \int_{\Theta} \int_{\mathcal{X}} L(\theta, d(\mathbf{x})) m(\mathbf{x}) h(\theta | \mathbf{x}) d\mathbf{x} d\theta \\ &= \int_{\mathcal{X}} m(\mathbf{x}) \left\{ \int_{\Theta} L(\theta, d(\mathbf{x})) h(\theta | \mathbf{x}) d\theta \right\} d\mathbf{x} \end{aligned}$$

其中 $m(\mathbf{x}) = \int_{\Theta} q(\mathbf{x} | \theta) \pi(\theta) d\theta$ 为 (\mathbf{X}, θ) 关于 \mathbf{X} 的边缘联合密度函数。对于离散型随机变量: $R_B(d) = \sum_{\mathbf{x}} m(\mathbf{x}) \{ \sum_{\theta} L(\theta, d(\mathbf{x})) h(\theta | \mathbf{x}) \}$ 。

由上式可见, 贝叶斯风险可以看作是对随机损失函数 $L(\theta, d(\mathbf{X}))$ 求两次数学期望而得到的, 第一次先对 θ 的后验分布求数学期望, 第二次是关于样本的边缘分布求数学期望。

定义 9.2.13. 设 $L(\theta, d)(\theta \in \Theta, d \in \mathcal{A})$ 为某一统计决策问题的损失函数, 则称 $L(\theta, d)$ 对后验分布 $h(\theta | \mathbf{x})$ 的数学期望, 记作

$$R(d | \mathbf{x}) = E(L(\theta, d)) = \int_{\Theta} L(\theta, d(\mathbf{x})) h(\theta | \mathbf{x}) d\theta \square$$

为样本观测值为 \mathbf{x} 时, 决策 d 的后验风险。

定理 9.2.2. 任给 $\mathbf{x} \in \mathcal{X}$, 若对任一 $d \in \mathcal{A}$, $R(d | \mathbf{x}) < +\infty$, 又存在决策函数 $d_{\mathcal{X}}$, 使得后验风险达到最小, 即

$$R(d_{\mathcal{X}} | \mathbf{x}) = \min_{d \in \mathcal{A}} R(d | \mathbf{x})$$

则由下式定义的决策函数

$$d^*(\mathbf{x}) = d_{\mathcal{X}}, \quad \mathbf{x} \in \mathcal{X}$$

是在后验风险准则下的最优决策函数, 即贝叶斯估计。

证明. 设样本的分布为 $\{q(\mathbf{x} | \theta) : \theta \in \Theta\}$, 参数 θ 的先验密度为 $\pi(\theta)$, $d(\mathbf{x})$ 为一决策函数, 则 $d(\mathbf{x})$ 的贝叶斯风险为

$$\begin{aligned} R_B(d) &= E_{\pi}[R(\theta, d)] = \int_{\Theta} R(\theta, d) \pi(\theta) d\theta \\ &= \int_{\mathcal{X}} m(\mathbf{x}) \int_{\Theta} L(\theta, d(\mathbf{x})) h(\theta | \mathbf{x}) d\theta d\mathbf{x} \\ &= \int_{\mathcal{X}} R(d | \mathbf{x}) m(\mathbf{x}) d\mathbf{x} \end{aligned} \quad (9.6)$$

由于, 对任意的 $\mathbf{x} \in \mathcal{X}$, 有

$$R(d^*(\mathbf{x}) | \mathbf{x}) = \min_{d \in \mathcal{A}} R(d | \mathbf{x}) \leq R(d(\mathbf{x}) | \mathbf{x})$$

从而有, $R_B(d^*) = \int_{\mathcal{X}} R(d^*(\mathbf{x}) | \mathbf{x}) m(\mathbf{x}) d\mathbf{x} \leq \int_{\mathcal{X}} R(d(\mathbf{x}) | \mathbf{x}) m(\mathbf{x}) d\mathbf{x} = R_B(d)$
即 d^* 贝叶斯决策函数。 \square

下面给出各种损失函数下的贝叶斯估计。

定理 9.2.3. 设 θ 的先验分布为 $\pi(\theta)$, 损失函数为 $L(\theta, d) = (\theta - d)^2$, 则 θ 的贝叶斯估计是

$$d^*(\mathbf{x}) = E(\theta | \mathbf{X} = \mathbf{x}) = \int_{\Theta} \theta \cdot h(\theta | \mathbf{x}) d\theta$$

其中 $h(\theta | \mathbf{x})$ 为参数 θ 的后验概率密度函数。

证明. 由于最小化

$$R_B(d) = \int_{\mathcal{X}} m(\mathbf{x}) \left\{ \int_{\Theta} [\theta - d(\mathbf{x})]^2 h(\theta | \mathbf{x}) d\theta \right\} d\mathbf{x}$$

与最小化 $\int_{\Theta} (\theta - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta$ 等价。

而

$$\begin{aligned} \int_{\Theta} (\theta - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta &= \int_{\Theta} (\theta - E(\theta | \mathbf{x}) + E(\theta | \mathbf{x}) - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta \\ &= \int_{\Theta} (\theta - E(\theta | \mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta + \int_{\Theta} (E(\theta | \mathbf{x}) - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta \\ &\quad + 2 \int_{\Theta} (\theta - E(\theta | \mathbf{x}))(E(\theta | \mathbf{x}) - d(\mathbf{x})) h(\theta | \mathbf{x}) d\theta \end{aligned}$$

其中 $E(\theta | \mathbf{x}) = \int_{\Theta} \theta \cdot h(\theta | \mathbf{x}) d\theta$, 又

$$\begin{aligned} \int_{\Theta} (\theta - E(\theta | \mathbf{x})) (E(\theta | \mathbf{x}) - d(\mathbf{x})) h(\theta | \mathbf{x}) d\theta &= (E(\theta | \mathbf{x}) - d(\mathbf{x})) \int_{\Theta} (\theta - E(\theta | \mathbf{x})) h(\theta | \mathbf{x}) d\theta \\ &= (E(\theta | \mathbf{x}) - d(\mathbf{x})) (E(\theta | \mathbf{x}) - E(\theta | \mathbf{x})) = 0 \end{aligned}$$

所以

$$\begin{aligned} &\int_{\Theta} (\theta - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta \\ &= \int_{\Theta} (\theta - E(\theta | \mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta + \int_{\Theta} (E(\theta | \mathbf{x}) - d(\mathbf{x}))^2 h(\theta | \mathbf{x}) d\theta \end{aligned}$$

显然, 当 $d^*(\mathbf{x}) = E(\theta | \mathbf{x})$ 时, $R_B(d)$ 达到最小。 \square

一般地, 我们常说的贝叶斯估计是指平方损失函数下用后验分布的均值作为 θ 的点估计, 也称为后验期望估计。实际上, 我们可以对该定理的结论做一个更为直观的解释。设 θ 的后验分布为 $h(\theta | \mathbf{x})$, 贝叶斯估计为 $\hat{\theta}$, 则 $(\hat{\theta} - \theta)^2$ 的后验期望

$$\text{MSE}(\hat{\theta} | \mathbf{x}) = E_{\theta | \mathbf{x}}(\hat{\theta} - \theta)^2$$

称为 $\hat{\theta}$ 的后验均方差, 其平方根称为后验标准误差, $\hat{\theta}$ 的后验均方差越小, 贝叶斯估计的误差就越小, 当 $\hat{\theta}$ 为 θ 的后验期望 $\hat{\theta}_B = E(\theta | \mathbf{x})$ 时,

$$\text{MSE}(\hat{\theta}_B | \mathbf{x}) = E_{\theta | \mathbf{x}}(\hat{\theta}_B - \theta)^2 = D(\theta | \mathbf{x})$$

称为后验方差, 其平方根称为后验标准差。后验均方差与后验方差, 有如下关系:

$$\begin{aligned} \text{MSE}(\hat{\theta} | \mathbf{x}) &= E_{\theta | \mathbf{x}}(\hat{\theta} - \theta)^2 \\ &= E_{\theta | \mathbf{x}} \left[(\hat{\theta} - \hat{\theta}_B) + (\hat{\theta}_B - \theta) \right]^2 \\ &\stackrel{\text{数据科学与工程数学基础}}{=} E_{\theta | \mathbf{x}} (\hat{\theta}_B - \hat{\theta})^2 + D(\theta | \mathbf{x}) \\ &= (\hat{\theta}_B - \hat{\theta})^2 + D(\theta | \mathbf{x}) \end{aligned}$$

上面的关系式表明, 当 $\hat{\theta}$ 取后验期望时, 可使后验均方差达到最小, 所以取后验期望作为 θ 的贝叶斯估计是合理的。

贝叶斯估计的求解 求贝叶斯估计的一般步骤为:

1. 根据总体 X 的分布, 求得条件概率 $q(\mathbf{x} | \theta)$
2. 在已知 θ 的先验分布 $\pi(\theta)$ 下, 求得 \mathbf{X} 与 θ 的联合分布密度 $g(\mathbf{x}, \theta) = \pi(\theta)q(\mathbf{x} | \theta)$
3. 求得 X 的边缘分布 $m(\mathbf{x})$
4. 计算 $h(\theta | \mathbf{x}) = \pi(\theta)q(\mathbf{x} | \theta) / m(\mathbf{x})$
5. 求数学期望 $\hat{\theta} = \int_{\Theta} \theta \cdot h(\theta | \mathbf{x}) d\theta$
6. 求得贝叶斯风险 (如果需要的话)

$$R_B(d) = \int_{\Theta} \int_{\mathcal{X}} L(\theta, d(\mathbf{x})) q(\mathbf{x} | \theta) \pi(\theta) d\mathbf{x} d\theta$$

例 9.2.13. X_1, X_2, \dots, X_n 来自正态分布 $N(\theta, \sigma_0^2)$ 的一个样本, 其中 σ_0^2 已知, θ 未知, 假设 θ 的先验分布为正态分布 $N(\mu, \tau^2)$, 其中先验均值 μ 和先验方差 τ^2 均已知, 试求 θ 的贝叶斯估计。

解. 样本 \mathbf{X} 的联合分布和 θ 的先验分布分别为

$$q(\mathbf{x} | \theta) = (2\pi\sigma_0^2)^{-\frac{n}{2}} \exp \left\{ -\frac{1}{2\sigma_0^2} \sum_{i=1}^n (x_i - \theta)^2 \right\}$$

$$\pi(\theta) = (2\pi\tau^2)^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2\tau^2} (\theta - \mu)^2 \right\}$$

由此可以写出 \mathbf{x} 与 μ 的联合分布

$$f(\mathbf{x}, \theta) = k_1 \cdot \exp \left\{ -\frac{1}{2} \left[\sigma_0^{-2} \left(n\theta^2 - 2n\theta\bar{x} + \sum_{i=1}^n x_i^2 \right) + \frac{\theta^2 - 2\theta\mu + \mu^2}{\tau^2} \right] \right\}$$

其中 $k_1 = (2\pi)^{-(n+1)/2} \tau^{-1} \sigma_0^{-n}$ 。若记 $A = \frac{n}{\sigma_0^2} + \frac{1}{\tau^2}$, $B = \frac{n\bar{x}}{\sigma_0^2} + \frac{\mu}{\tau^2}$, $C = \sigma_0^{-2} \sum_{i=1}^n x_i^2 + \frac{\mu^2}{\tau^2}$ 则有

$$f(\mathbf{x}, \theta) = k_1 \exp \left\{ -\frac{1}{2} [A\theta^2 + B\theta + C] \right\}$$

$$= k_1 \exp \left\{ -\frac{(\theta - B/A)^2}{2/A} - \frac{1}{2} (C - B^2/A) \right\}$$

注意到 A, B, C 均与 θ 无关, 样本的边际密度函数

$$m(\mathbf{x}) = \int_{-\infty}^{\infty} f(\mathbf{x}, \theta) d\theta = k_2 \exp \left\{ -\frac{1}{2} (C - B^2/A) \right\} \cdot \sqrt{\frac{2\pi}{A}}$$

应用贝叶斯公式即可得到后验分布

$$h(\theta | \mathbf{x}) = \frac{f(\mathbf{x}, \theta)}{m(\mathbf{x})} \sqrt{\frac{A}{2\pi}} \exp \left\{ -\frac{1}{2/A} (\theta - B/A)^2 \right\}$$

这说明在样本给定后, θ 的后验分布为 $N(B/A, 1/A)$, 即 $\theta | \mathbf{x} \sim N(B/A, 1/A)$ 记作 $\theta | \mathbf{x} \sim N(\mu_1, \sigma_1^2)$, 其中

$$\mu_1 = \frac{B}{A} = \frac{n\sigma_0^{-2}\bar{x} + \tau^{-2}\mu}{n\sigma_0^{-2} + \tau^{-2}}, \sigma_1^2 = \frac{1}{A} = \frac{\sigma_0^2\tau^2}{\sigma_0^2 + n\tau^2}$$

后验均值即为其贝叶斯估计:

$$\hat{\theta} = \frac{n\tau^2}{n\tau^2 + \sigma_0^2} \bar{x} + \frac{\sigma_0^2}{n\tau^2 + \sigma_0^2} \mu$$

它是样本均值 \bar{x} 与先验均值 μ 的加权平均。

基于特定损失函数的贝叶斯估计 实际上, 除了平方损失函数以外, 还有线性损失函数、加权平方损失函数、绝对值损失函数等等。我们依然可以求出对应的贝叶斯估计值。

定理 9.2.4. 设 θ 的先验分布为 $\pi(\theta)$, 取损失函数为加权平方损失函数

$$L(\theta, d) = \lambda(\theta)(d - \theta)^2$$

则 θ 的贝叶斯估计为 $d^*(\mathbf{x}) = \frac{E[\lambda(\theta)\theta | \mathbf{x}]}{E[\lambda(\theta) | \mathbf{x}]}$

定理 9.2.5. 设 $\theta = (\theta_1, \theta_2, \dots, \theta_p)^T$ 的先验分布为 $\pi(\theta)$ ，损失函数为

$$L(\theta, d) = (d - \theta)^T Q (d - \theta),$$

Q 正定。则 θ 的贝叶斯估计为 $d^*(\mathbf{x}) = E(\theta | \mathbf{x}) = \begin{bmatrix} E(\theta_1 | \mathbf{x}) \\ \vdots \\ E(\theta_p | \mathbf{x}) \end{bmatrix}$

定理 9.2.6. 设 θ 的先验分布为 $\pi(\theta)$ ，在线性损失函数

$$L(\theta, d) = \begin{cases} k_0(\theta - d), & d \leq \theta \\ k_1(d - \theta), & d > \theta \end{cases}$$

下，则 θ 的贝叶斯估计 $d^*(\mathbf{x})$ 为后验分布 $h(\theta | \mathbf{x})$ 的 $k_1 / (k_0 + k_1)$ 上侧分位数。

定理 9.2.7. 设的先验分布为 $\pi(\theta)$ ，损失函数为绝对值损失 $L(\theta, d) = |d - \theta|$ ，则 θ 的贝叶斯估计 $d^*(\mathbf{x})$ 为后验分布 $h(\theta | \mathbf{x})$ 的中位数。

最大后验估计 实际上，还可以取得后验分布的概率密度函数最大化的参数，这就是最大后验估计。

定义 9.2.14. 设 θ 的后验密度函数为 $h(\theta | \mathbf{x})$ ，若 $\hat{\theta} = \hat{\theta}(x_1, x_2, \dots, x_n)$ 使得

$$h(\hat{\theta} | \mathbf{x}) = \max_{\theta \in \Theta} h(\theta | \mathbf{x})$$

则称 $\hat{\theta}$ 为 θ 最大后验估计。

例 9.2.14. 设总体 $X \sim E(\theta)$, X_1, X_2, \dots, X_n 为来自总体 X 的样本, θ 的先验分布为指数分布 $E(\lambda)$ (λ 已知), 求 θ 的最大后验估计。

解. 因为先验概率密度函数为:

$$\pi(\theta) = \begin{cases} \lambda e^{-\lambda\theta} & \theta > 0 \\ 0 & \theta \leq 0 \end{cases}$$

样本 (X_1, X_2, \dots, X_n) 的联合概率密度为:

$$q(\mathbf{x} | \theta) = \prod_{n=2}^n f(x_i | \theta) = \begin{cases} \theta^n e^{-\theta \sum_{i=1}^n x_i} & x_1, x_2, \dots, x_n > 0 \\ 0 & \text{其它} \end{cases}$$

所以 θ 的后验分布密度

$$h(\theta | \mathbf{x}) \propto \theta^n e^{-(\lambda + \sum_{i=1}^n x_i)\theta}$$

$$\ln h(\theta | \mathbf{x}) = n \ln \theta - \left(\lambda + \sum_{i=1}^n x_i \right) \theta + \ln c(\mathbf{x})$$

$$\frac{\partial \ln h(\theta | \mathbf{x})}{\partial \theta} = \frac{n}{\theta} - \left(\lambda + \sum_{i=1}^n x_i \right) = 0$$

求得 θ 的最大后验估计为 $\hat{\theta} = \frac{1}{\bar{x} + \lambda/n}$ 。当 $n \rightarrow \infty$ 时, $\hat{\theta} \rightarrow \frac{1}{\bar{x}}$, 与传统意义上的极大似然估计是一致的。

基于后验分布 $h(\theta | \mathbf{x})$ 的贝叶斯估计, 常用有如下三种: 用后验分布的密度函数最大值作为 θ 的点估计, 称为最大后验估计; 用后验分布的中位数作为 θ 的点估计, 称为后验中位数估计; 用后验分布的均值作为 θ 的点估计, 称为后验期望估计。最为常见的是后验期望估计, 简称为贝叶斯估计, 记为 $\hat{\theta}_B$ 。

最小最大估计

在统计决策理论中, 风险函数提供了一个衡量决策函数好坏的尺度。贝叶斯决策是根据贝叶斯风险最小的原则而取的最优决策, 如果将“最优性”的准则改变, 就可以得到另一种“最优”决策。接下来我们介绍基于最大风险的最小最大决策规则。与基于贝叶斯风险的贝叶斯决策相比, 最小最大决策不依赖于参数的先验信息。

定义 9.2.15. 对于给定的统计决策问题, 设 $d(\mathbf{x})$ 为该统计问题的决策函数, 又设 $d(\mathbf{x})$ 的风险函数为 $R(\theta, d)(\theta \in \Theta)$, 称

$$M(d) = \sup_{\theta} R(\theta, d),$$

为决策函数 $d(\mathbf{x})$ 的最大风险。

最大最小规则 使最大风险最小的决策称为最小最大规则, 相应的决策函数称为最小最大决策。

定义 9.2.16. 对于给定的统计决策问题, 若在决策函数类 \mathcal{A} 中存在一个决策函数 $d^*(\mathbf{x})$, 使得

$$\sup_{\theta \in \Theta} R(\theta, d^*) = \inf_{d \in \mathcal{A}} \sup_{\theta \in \Theta} R(\theta, d),$$

则称 d^* 为最小最大 (Minmax) 决策, 或称 d^* 为该统计问题的最小最大规则或最小最大解。

当问题为估计或检验时, 称 d^* 为最小最大估计或最小最大检验。后面我们主要关注最小最大估计。

最小最大规则从风险函数的整体性质来确定决策风险的优良性。使决策函数的最大风险达到最小是考虑到最不利的情况, 要求最不利的情况尽可能地好。也就是人们常说的从最坏处着想, 争取最好的结果。因此最小最大规则是比较保守的规则。

通常, 如果对参数 θ 的先验信息有所了解, 则利用贝叶斯为好; 若对参数 θ 的信息毫无了解, 则可使用最小最大准则。寻求最小最大决策函数的一般步骤是:

- (1) 对 \mathcal{A} 中所有决策函数求最大风险 $\max_{\Theta}(R(\theta, d)), \forall d \in D$
- (2) 在所有最大风险值中选取最小值 $\min_d (\max_{\Theta}(R(\theta, d)))$

此最小值所对应的决策函数就是最小最大决策函数。

例 9.2.15. 设总体 $X \sim B(1, p)$, 即

$$P(X = x) = p^x(1 - p)^{1-x} (x = 0, 1)$$

表 9.2: 损失函数 $L(p, a)$ 取值表

$L(p, a) \setminus a$		$a_1 = \frac{1}{4}$		$a_2 = \frac{1}{2}$
		$p_1 = \frac{1}{4}$	1	4
p	$p_2 = \frac{1}{2}$	3	2	

其中 $p \in \Theta = \{\frac{1}{4}, \frac{1}{2}\}$, 决策空间为 $\mathcal{A} = \{\frac{1}{4}, \frac{1}{2}\}$, 设损失函数 $L(p, a)$ 为表 9.2 所示。试求参数 p 的最小最大估计量。

解. 如果我们选取容量为 1 的样本为 X_1 , 由于 X_1 仅取两个可能值及 \mathcal{A} 中只有两个元素, 因而决策函数的集合 D 是由 4 个元素所组成, 其分别记为 d_1, d_2, d_3, d_4 , 即有

表 9.3: 决策函数表

D		d_1	d_2	d_3	d_4
		d_1	d_2	d_3	d_4
X	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{2}$
	1	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{4}$

由上表计算可得

$$R(p_1, d_1) = L(p_1, a_1) P(X = 0) + L(p_1, a_1) P(X = 1) = 1 \times \frac{3}{4} + 1 \times \frac{1}{4} = 1$$

$$R(p_1, d_2) = L(p_1, a_2) P(X = 0) + L(p_1, a_2) P(X = 1) = 4 \times \frac{3}{4} + 4 \times \frac{1}{4} = 4$$

$$R(p_1, d_3) = L(p_1, a_1) P(X = 0) + L(p_1, a_2) P(X = 1) = 1 \times \frac{3}{4} + 4 \times \frac{1}{4} = \frac{7}{4}$$

$$R(p_1, d_4) = L(p_1, a_2) P(X = 0) + L(p_1, a_1) P(X = 1) = 4 \times \frac{3}{4} + 1 \times \frac{1}{4} = \frac{13}{4}$$

同理计算可得 $R(p_2, d_1) = 3$, $R(p_2, d_2) = 2$, $R(p_2, d_3) = \frac{5}{2}$, $R(p_2, d_4) = \frac{5}{2}$.

表 9.4: 风险函数值与最大值

$d_1(x_1)$	d_1	d_2	d_3	d_4
$R(p_1, d_i)$	1	4	$\frac{7}{4}$	$\frac{13}{4}$
$R(p_2, d_i)$	3	2	$\frac{5}{2}$	$\frac{5}{2}$
$\max_{\theta \in \Theta} R(p, d_i)$	3	4	$\frac{5}{2}$	$\frac{13}{4}$

于是 p 的最小最大值估计为：

$$\hat{p}(X_1) = d_3 = \begin{cases} \frac{1}{4} & X_1 = 0 \\ \frac{1}{2} & X_1 = 1 \end{cases}$$

最大最小决策函数 寻找最小最大决策函数通常是较困难的, 然而贝叶斯决策函数与最小最大决策函数有一定的联系。以下定理可以作为验证某一决策 d 为最小最大决策函数的方法。

定理 9.2.8. 设 $d^*(\mathbf{x})$ 为某一先验分布 $\pi(\theta)$ 下的贝叶斯决策函数, 且对任意的 $\theta \in \Theta$, $d^*(\mathbf{x})$ 的风险函数 $R(\theta, d^*) = c$ 为常数, 则 $d^*(\mathbf{x})$ 为该统计决策问题的最小最大决策函数。

证明. 用反证法。若 $d^*(\mathbf{x})$ 不是最小最大决策函数, 则存在决策函数 $d(\mathbf{x})$, 使得 $M(d) < M(d^*) = \sup_{\theta \in \Theta} R(\theta, d^*) = c$, 此时有

$$R_\pi(d) = \int_{\Theta} R(\theta, d) \pi(\theta) d\theta \leq \int_{\Theta} M(d) \pi(\theta) d\theta < c = R_\pi(d^*)$$

这与 $d^*(\mathbf{x})$ 为先验分布 $\pi(\theta)$ 下的贝叶斯决策函数相矛盾。因此, $d^*(\mathbf{x})$ 必定是一个最小最大决策函数。 \square

定理 9.2.9. 设给定一个贝叶斯决策问题, 在先验分布 $\pi_k(\theta)$ 下的贝叶斯决策函数为 d_k , 而 d_k 的贝叶斯风险为 $B_{\pi_k}(d_k)$ ($k = 1, 2, \dots$), 若

$$\lim_{k \rightarrow \infty} B_{\pi_k}(d_k) = \rho < +\infty$$

且 d^* 为一决策函数, 满足

$$\sup_{\theta \in \Theta} R(\theta, d^*) \leq \rho$$

则 d^* 为该统计决策问题的最小最大决策函数。

证明. 用反证法。若 d^* 不是最小最大决策函数, 则存在决策函数 d , 使得

$$\sup_{\theta \in \Theta} R(\theta, d) < \sup_{\theta \in \Theta} R(\theta, d^*) \leq \rho$$

此时, 存在 $\varepsilon > 0$, 使得 $R(\theta, d) \leq \rho - \varepsilon, \forall \theta \in \Theta$, 因此, 对一切 k , 有

$$B_{\pi_k}(d) = \int_{\Theta} R(\theta, d) \pi_k(\theta) d\theta \leq \rho - \varepsilon$$

由于 d_k 为在先验分布 $\pi_k(\theta)$ 下的贝叶斯决策函数, 所以有

$$B_{\pi_k}(d) \geq B_{\pi_k}(d_k) > \rho - \varepsilon$$

矛盾, 故 d^* 为最小最大决策函数。 \square

例 9.2.16. 设总体 X 服从正态分布 $N(\theta, 1)$, X_1, X_2, \dots, X_n 为来自总体 X 的样本, 损失函数为 $L(\theta, d) = (\theta - d)^2$, 求 θ 的最小最大估计。

解. 选取一列先验分布 $\{\pi_k\}$, $\pi_k(\theta) \sim N(0, k^2)$, 在 π_k 下, θ 的贝叶斯估计为

$$d_k = E(\theta | \mathbf{x}) = \frac{n\bar{x}}{n + \frac{1}{k^2}} = \frac{k^2 n \bar{x}}{k^2 n + 1}$$

由于

$$\begin{aligned} R(\theta, d_k) &= E_\theta L(\theta, d_k) = E_\theta \left[\frac{k^2 n \bar{X}}{k^2 n + 1} - \theta \right]^2 \\ &= \frac{E_\theta [k^2 n (\bar{X} - \theta) - \theta]^2}{(k^2 n + 1)^2} = \frac{k^4 n + \theta^2}{(k^2 n + 1)^2} \end{aligned}$$

所以 d_k 的贝叶斯风险为

$$B_{\pi_k}(d_k) = E_{\pi_k} \left[\frac{k^4 n + \theta^2}{(k^2 n + 1)^2} \right] = \frac{k^2}{k^2 n + 1}$$

因为 $\lim_{k \rightarrow \infty} B_{\pi_k}(d_k) = \frac{1}{n}$, 而取决策函数 $d^* = \bar{x}$, 则 d^* 的风险函数

$$R(\theta, d^*) = E_\theta L(\theta, d^*) = E_\theta [\bar{X} - \theta]^2 = \frac{1}{n}$$

从而 $\sup_{\theta \in \Theta} R(\theta, d^*) = \frac{1}{n}$, 于是 θ 的最小最大估计为 $d^* = \bar{x}$ 。

MLE、贝叶斯估计、最大后验估计、最小最大估计间的联系

极大似然估计、贝叶斯估计 极大似然估计: 是把参数 θ 看成为确定的未知参数。然后求似然函数 $L(\theta)$ 为最大的 $\hat{\theta}$ 作为极大似然估计量

$$\hat{\theta} = \arg \max_{\theta} L(\theta) = \arg \max_{\theta} \prod_{i=1}^n f(x_i | \theta)$$

贝叶斯估计: 是把参数 θ 看成为随机的未知参数, 一般 θ 具有先验分布 $\pi(\theta)$, 然后通过似然函数 $q(\mathbf{x} | \theta)$ 和贝叶斯公式将 θ 的先验分布 $\pi(\theta)$ 转化为后验分布 $h(\theta | \mathbf{x})$ 。利用公式

$$h(\theta | \mathbf{x}) = \frac{q(\mathbf{x} | \theta) \pi(\theta)}{\int q(\mathbf{x} | \theta) \pi(\theta) d\theta}, \quad \hat{\theta}_B = E(\theta | \mathbf{x}) = \int_{\Theta} \theta h(\theta | \mathbf{x}) d\theta$$

求出贝叶斯估计量 $\hat{\theta}_B$ 。

贝叶斯估计可用于贝叶斯学习: 是利用 θ 的先验分布及样本提供的信息求出 θ 的后验分布 $h(\theta | \mathbf{x})$, 然后直接求总体分布。

极大似然估计与最大后验估计的关系 当样本数趋于无穷时, 最大后验概率估计一般趋向于极大似然估计。极大似然估计也可看作参数的先验概率密度函数服从均匀分布(相当于没有先验知识)的最大后验概率估计。当参数的先验概率密度函数比较准确时, 最大后验概率估计的小样本性质大大优于极大似然估计。

极大似然法和贝叶斯方法 在方法的计算复杂度方面：此标准下选择极大似然法，因为 MLE 仅涉及一些微分运算或梯度搜索技术，而 Bayesian 要计算非常复杂的多重积分。在可理解性方面：MLE 比 Bayesian 更易理解和掌握，因为 MLE 结果是基于设计者所提供的训练样本的一个最佳答案，而 Bayesian 得到的结果则是许多可行解的加权平均，反映出对各种可行解的不确定程度。在对初始先验知识的信任程度方面，贝叶斯方法要求更高。

总之，通过使用全部 $h(\theta|\mathbf{x})$ 中的信息，Bayesian 方法比 MLE 法能够利用更多有用的信息。如果这些信息可靠，有理由认为 Bayesian 比 MLE 能够得到更准确的结果。在没有特别先验知识（如均匀分布）情况下，二种方法比较相似。若有非常多的训练样本，使 $h(\theta|\mathbf{x})$ 形成一个非常显著的尖峰，而先验概率 $\pi(\theta)$ 又是均匀分布，从本质上来说，MLE 和 Bayesian 相同。若 $h(\theta|\mathbf{x})$ 波形比较宽，或者在 $\hat{\theta}$ 附近是不对称的（此不对称由问题本身决定），MLE 和 Bayesian 产生的结果就不相同。非常明显的不对称性显然表示了分布本身的某些特点。Bayesian 能够利用这些特点，而 MLE 却忽略这些特点。

极大似然估计、贝叶斯估计和最小最大估计 在绝大多数大样本参数模型中，MLE 近似最小最大估计。常值风险函数的贝叶斯估计就是最小最大估计。除此之外，概率模型有时既含有观测变量 (observable variable)，又含有隐变量或潜在变量 (latent variable)。如果概率模型的变量都是观测变量，那么给定数据，可以直接用极大似然估计法或贝叶斯估计法估计模型参数。但是，当模型还有隐变量时，就不能简单地使用这些估计方法。可以通过使用 EM 算法 (期望极大法)，也就是含有隐变量的概率模型参数的极大似然估计法或极大后验概率估计法，来进行估计。

9.2.5 非参数估计

参数估计要求总体的密度函数的形式已知，但这种假定有时并不成立；常见的一些函数形式很难拟合实际的概率密度，实际中样本维数较高，且关于高维密度函数可以表示成一些低维密度函数乘积的假设通常也不成立；经典的密度函数都是单峰的，而在许多实际情况中却是多峰的，即有多个局部极大值。但是为了设计贝叶斯分类器，仍然需要总体分布的知识，于是提出一些直接用样本来估计总体分布的方法，称之为：估计分布的非参数方法。非参数估计可以描述为：密度函数的形式未知，也不作假设，利用训练数据（样本）直接对任意的概率密度进行估计。又称作模型无关方法。

先来回顾一下之前提到的概率密度估计问题：给定 i.i.d. 样本集： $X = \{X_1, X_2, \dots, X_n\}$ ，估计概率分布： $p(X)$ 。

非参数概率密度估计方法主要有三种：直方图密度估计、核密度估计以及 k 近邻估计。下面将对这些方法进行展开介绍。

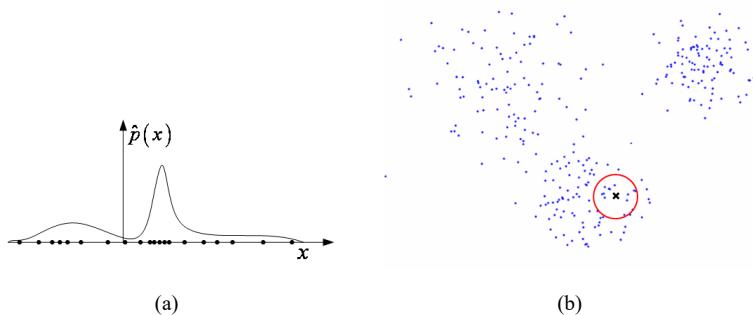


图 9.4: 概率密度估计

直方图密度估计

在经典的统计学中, 直方图主要用于描述数据的频率。本节将介绍如何使用直方图估计一个随机变量的密度。直方图密度估计与用直方图估计频率的差别在于: 在直方图密度估计中, 我们需要对频率估计进行归一化, 使其成为一个密度函数的估计。直方图估计是非参数概率密度估计最简单的方法。

一元函数的直方图密度估计 先讨论一元函数的直方图密度估计。假定有数据 $x_1, x_2, \dots, x_n \in [a, b]$ 。对区间 $[a, b]$ 做如下划分, 即 $a = a_0 < a_1 < a_2 < \dots < a_k = b, I_i = [a_{i-1}, a_i), i = 1, 2, \dots, k$ 。我们有 $\bigcup_{i=1}^k I_i = [a, b], I_i \cap I_j = \emptyset, i \neq j$ 。令 $n_i = \#\{x_i \in I_i\}$ 为落在 I_i 中数据的个数; 定义直方图密度估计为:

$$\hat{p}(x) = \begin{cases} \frac{n_i}{n(a_i - a_{i-1})}, & \text{当 } x \in I_i; \\ 0, & \text{当 } x \notin [a, b], \end{cases}$$

在实际操作中, 经常取相同的区间, 即 $I_i (i = 1, 2, \dots, k)$ 的宽度均为 h , 在此情况下, 有

$$\hat{p}(x) = \begin{cases} \frac{n_i}{nh}, & \text{当 } x \in I_i; \\ 0, & \text{当 } x \notin [a, b]. \end{cases}$$

上式中 h 既是归一化参数, 又表示每一组的组距, 称为带宽或窗宽。

另外, 我们可以看到

$$\int_a^b \hat{p}(x) dx = \sum_{i=1}^k \int_{I_i} n_i / (nh) dx = \sum_{i=1}^k n_i / n = 1.$$

由于位于同一组内所有点的直方图密度估计均相等, 因而直方图所对应的分布函数 $\hat{F}_h(x)$ 是单调增的阶梯函数。这与经验分布函数形状类似。实际上, 当分组间隔 h 缩小到每组中最多只有一个数据时, 直方图的分布函数就是经验分布函数, 即 $h \rightarrow 0$, 有 $\hat{F}_h(x) \rightarrow \hat{F}_n(x)$ 。

定理 9.2.10. 固定 x 和 h , 令估计的密度是 $\hat{p}(x)$, 如果 $x \in I_j, p_j = \int_{I_j} \hat{p}(x) dx$, 有

$$E\hat{p}(x) = p_j/h, \quad \text{var } \hat{p}(x) = \frac{p_j(1-p_j)}{nh^2}$$

证明提示: 注意到 $E\hat{p}_j = n_j/n = \int_{I_j} \hat{p}(x) dx, \text{var } \hat{p}_j = p_j(1-p_j)/n$.

例 9.2.17. 下面使用了鸢尾花数据集中的山鸢尾 (Setosa) 和维吉尼亚鸢尾 (Virginical) 两种花花萼长度的观测数据, 共计 150 条. 在下图中, 我们从左到右, 分别采用逐渐增加的带宽间隔: $h_l = 0.40, h_m = 0.19, h_r = 0.09$ 制作了 3 个直方图. 可以发现当带宽很小的时候, 个体特征比较明显, 从图中可以看到多个峰值; 而带宽过大的最左边的图上, 很多峰都不明显了. 中间的图比较合适, 它有两个主要的峰, 提供了最为重要的特征信息. 实际上, 参与直方图运算的是山鸢尾和维吉尼亚鸢尾两种花花萼长度的混合数据, 经验表明, 大部分山鸢尾的花萼长度与维吉尼亚鸢尾的花萼长度有一定的差别, 因而两个峰是合适的.

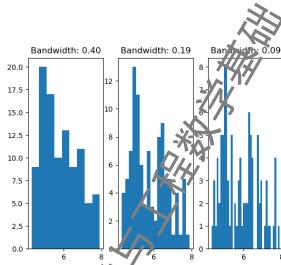


图 9.5: 山鸢尾和维吉尼亚鸢尾花萼长度

理论性质和最优带宽 由于带宽的不同, 会得到不同的估计结果. 因此选择合适的带宽, 对于得到好的密度估计非常重要. 在计算最优带宽前, 我们先定义 \hat{p} 的平方损失风险:

$$R(\hat{p}, p) = \int (\hat{p}(x) - p(x))^2 dx.$$

定理 9.2.11. 假设 $\int p'(x) dx < +\infty$, 则在平方损失风险下, 有

$$R(\hat{p}, p) \approx \frac{h^2}{12} \int (p'(u))^2 du + \frac{1}{nh}.$$

极小化上式, 得到理想带宽为

$$h^* = \frac{1}{n^{1/3}} \left(\frac{6}{\int p'(x)^2 dx} \right)^{1/3}$$

于是理想的带宽为 $h = Cn^{-1/3}$.

在大多数情况下, 我们不知道密度 $p(x)$, 因此也不知道 $p'(x)$. 对于理想带宽

$$h^* = \frac{1}{n^{1/3}} \left(\frac{6}{\int p'(x)^2 dx} \right)^{1/3}$$

也无法计算, 在实际操作中, 经常假设 $p(x)$ 为一个标准正态分布, 并进而得到一个带宽 $h_0 \approx 3.5n^{-1/3}$. 直方图密度估计的优势在于简单易懂, 在计算过程中也不涉及复杂的模型计算, 只需要计算 I_j 中样本点的个数. 另一方面, 直方图密度估计只能给出一个阶梯函数, 该估计不够光滑. 另外一个问题就是直方图密度估计的收敛速度比较慢, 也就是说, $\hat{p}(x) \rightarrow p(x)$ 比较慢.

多维直方图 下面我们扩展一维直方图的密度定义公式到任意维空间. 设有 n 个观测点 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$, 将空间分成若干小区域 R , V 是区域 R 所包含的体积. 如果有 k 个点落入 R , 则可以得到如下密度公式: $p(x)$ 的估计为

$$p(\mathbf{x}) \approx \frac{k/n}{V}$$

如果这个体积和所有的样本体积相比很小, 就会得到一个很不稳定的估计, 这时, 密度值局部变化很大, 呈现多峰不稳定的特点; 反之, 如果这个体积太大, 则会圈进大量样本, 从而使估计过于平滑.

核密度估计

如何平衡不稳定与过度光滑产生两种可能的解决方法: 核估计和 k 近邻估计. 核估计法的总体思想: 固定体积 V 不变, 它与样本总数呈反比关系即可. 注意到, 在直方图密度估计中, 每一点的密度估计只与它是否属于某个 I_i 有关, 而 I_i 是预先给定的与该点无关的区域. 不仅如此, 区域 I_i 中每个点共有相等的密度, 这相当于待估点的密度取邻域 R 的平均密度. 现在以待估点为中心, 作体积为 V 的邻域, 令该点的密度估计与纳入该邻域中的样本点的多少呈正比, 如果纳入的点多, 则取密度大, 反之亦然. 这一点还可以进一步扩展开去, 将密度估计不再局限于 R 内的带内, 而是将体积 V 合理拆分到所有样本点对待估计点贡献的加权平均, 同时保证距离远的点取较小的权, 距离近的点取较大的权, 这样就形成了核函数密度估计法的基本思想. 后面我们将看到, 这些方法都可能获得较为稳健而适度光滑的估计.

k 近邻估计的总体思想: 固定 k 值不变, 它与样本总数呈一定关系即可. 根据数据之间的疏密情况调整 V , 这样就导致了另外一种密度估计方法— k 近邻法. 下面介绍核估计和 k 近邻估计两种非参数方法, 本小节主要介绍核估计.

一维情形 直方图是不连续的. 核密度估计较光滑且比直方图估计较快地收敛到真正的密度. 先考虑一维的情况.

定义 9.2.17. 假设数据 x_1, x_2, \dots, x_n 取自连续分布 $p(x)$, 在任意点 x 处的一种核密度估计定义为

$$\hat{p}(x) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h} \omega_i = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (9.7)$$

其中 $h > 0$, 称作为带宽; $K(\cdot)$ 称为核函数 (*kernel function*) 并且满足

$$K(x) \geq 0, \quad \int K(x)dx = 1.$$

定义中关于核函数 K 的分布密度的约束可以保证 $\hat{p}(x)$ 作为概率密度函数的合理性, 也即其值非负并且积分结果为 1。实际上, 容易验证有

$$\begin{aligned} \int \hat{p}(x)dx &= \int \frac{1}{n} \sum_{i=1}^n \frac{1}{h} K\left(\frac{x-x_i}{h}\right) dx = \frac{1}{n} \sum_{i=1}^n \int \frac{1}{h} K\left(\frac{x-x_i}{h}\right) dx \\ &= \frac{1}{n} \sum_{i=1}^n \int K(u)du = \frac{1}{n} \cdot n = 1 \quad \left(\text{其中 } u = \frac{x-x_i}{h} \right). \end{aligned}$$

因此上述定义的 $\hat{p}(x)$ 是一个合理的密度估计函数。

常用的一维核函数 核密度估计中, 一个重要的部分就是核函数。以一维为例, 常用的核函数如表所示。

核函数名称	核函数 $K(u)$
Parzen 窗 (Uniform)	$\frac{1}{2}I(u \leq 1)$
三角 (Triangle)	$1 - u I(u \leq 1)$
Epanechnikov	$\frac{3}{4}(1 - u^2)I(u \leq 1)$
四次 (Quartic)	$\frac{15}{16}(1 - u^2)^2I(u \leq 1)$
三权 (Triweight)	$\frac{35}{32}(1 - u^2)^3I(u \leq 1)$
高斯 (Gauss)	$\frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}u^2)$
余弦 (Cosinus)	$\frac{\pi}{4}\cos(\frac{\pi}{2}u)I(u \leq 1)$
指数 (Exponent)	$\exp\{ u \}$

下图给出了各种带宽之下根据正态核函数做出的密度估计曲线。由图可知, 带宽 $h = 0.40$ 是最平滑的 (左边), 相反带宽 $h = 0.09$ 噪声很多, 它在密度中引入了很多虚假的波形。从图中比较, 带宽 $h = 0.19$ 是较为理想的, 它在不稳定和过于平滑之间作了较好的折中。

一维情形最优带宽 为了构造一个核密度估计, 需要选择一个核函数 K 和一个带宽 h 。理论和经验表明 K 的选择不是关键的, 但是带宽 h 的选择非常重要。带宽对模型光滑程度的影响作用较大。如果 h 非常大, 将有更多的点对 x 处的密度产生影响。由于分布是归一化的, 即

$$\int \omega_i (x - x_i) dx = \int \frac{1}{h} K\left(\frac{x-x_i}{h}\right) dx = \int K(u)du = 1,$$

因而距离 x_i 较远的点也分担了对 x 的部分权重, 从而较近的点的权重 ω_i 减弱, 距离远和距离近的点的权重相差不大。在这种情况下, $\hat{p}(x)$ 是 n 个变化幅度不大的函数的叠加, 因此 $\hat{p}(x)$ 非常平

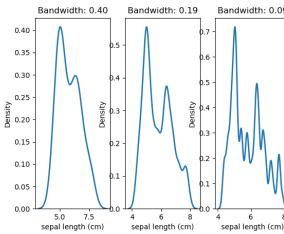


图 9.6: 山鸢尾和维吉尼亚鸢尾的花萼长度密度核估计

滑。反之, 如果 h 很小, 则各点之间的权重由于距离的影响而出现大的落差, 因而 $\hat{p}(x)$ 是 n 个以样本点为中心的尖脉冲的叠加, 就好像是一个充满噪声的估计。

如何选择合适的带宽, 是核函数密度估计能够成功应用的关键. 通过分析密度估计与真实密度之间的均方误差, 有如下定理:

定理 9.2.12. 假设 $\hat{p}(x)$ 定义如式(9.7), 是 $p(x)$ 的核估计, 令 $\text{supp}(p) = \{x : p(x) > 0\}$ 是密度 p 的支撑. 设 $x \in \text{supp}(p) \subset \mathbb{R}$ 为 $\text{supp}(p)$ 的内点 (非边界点), 当 $n \rightarrow +\infty$ 时, $h \rightarrow 0, nh \rightarrow +\infty$, 核估计有如下性质:

$$\begin{aligned} \text{Bias}(x) &= \frac{h^2}{2} \mu_2(K) p^{(2)}(x) + O(h^2) \\ V(x) &= (nh)^{-1} p(x) R(K) + O((nh)^{-1}) + O(n^{-1}) \end{aligned}$$

若 $\sqrt{(nh)}h^2 \rightarrow 0$, 则

$$\sqrt{(nh)}(\hat{p}_{n,h}(x) - p(x)) \rightarrow N(0, p(x)R(K))$$

其中 $R(K) = \int K(x)^2 dx$.

从均方误差的偏差和方差分解来看, 带宽 h 越小, 核估计的偏差越小, 但核估计的方差越大; 反之, 带宽 h 增大, 则核估计的方差变小, 但核估计偏差却增大。所以, 带宽 h 的变化不可能一方面使核估计的偏差减小, 同时又使核估计的方差减小。因而, 最佳带宽选择的标准必须在核估计的偏差和方差之间作一个权衡, 使积分均方误差达最小。

实际上, 由上述定理, 我们可以得到渐近积分均方误差 (AMISE):

$$\frac{h^4}{4} \mu_2^2 \int p^{(2)}(x)^2 dx + n^{-1} h^{-1} \int K(x)^2 dx,$$

由此可知, 最优带宽为

$$h_{\text{opt}} = \mu_2(K)^{-4/5} \left\{ \int K(x)^2 dx \right\}^{1/5} \left\{ \int p^{(2)}(x)^2 dx \right\}^{-1/5} n^{-1/5}.$$

对于上式中的最优带宽, 核函数 $K(u)$ 是已知的, 但是密度函数 $p(x)$ 是未知的. 在实际操作中, 我们经常把 $p(x)$ 看成正态分布去求解, 即 $\int p^{(2)}(x)^2 dx = \frac{3}{8}\pi^{-1/2}\sigma^{-5}$, 这样, 对于不同的核函

数, 我们可以得到相应的最优带宽. 例如当核函数是高斯时, 可以得到 $\mu_2 = 1, \int K(u)^2 du = \int \frac{1}{2\pi} \exp(-u^2) du = \pi^{-1/2}$, 这样, 最优带宽就是 $h_{\text{opt}} = 1.06\sigma n^{-1/5}$.

除了上述的方法, 从实际计算的角度, Rudemo(1982) 和 Bowman(1984) 提出用交叉验证法确定最终带宽的递推方法. 具体来说, 考虑积分平方误差:

$$\text{ISE}(h) = \int (\hat{p}(x) - p(x))^2 dx = \int \hat{p}^2 dx + \int p^2 dx - 2 \int \hat{p}p dx$$

达到最小, 将右边展开, 因此这等价于最小化式:

$$\text{ISE}(h)_{\text{opt}} = \int \hat{p}^2 dx - 2 \int \hat{p}p dx.$$

注意到等式的第二项为 $\int \hat{p}p dx = E(\hat{p})$, 因此, 可以用 $\int \hat{p}p dx$ 的一个无偏估计 $n^{-1} \sum_{i=1}^n \hat{p}_{-i}(X_i)$ 来估计, 其中 \hat{p}_{-i} 是将第 i 个观测点剔除后的概率密度估计. 下面只要估计第一项即可.

将核估计定义式代入第一项, 不难验证:

$$\begin{aligned} \int \hat{p}^2 dx &= n^{-2} h^{-2} \sum_{i=1}^n \sum_{j=1}^n \int_x K\left(\frac{X_i - x}{h}\right) K\left(\frac{X_j - x}{h}\right) dx \\ &= n^{-2} h^{-1} \sum_{i=1}^n \sum_{j=1}^n \int_t K\left(\frac{X_i - X_j}{h} - t\right) K(t) dt \end{aligned}$$

于是, $\int \hat{p}^2 dx$ 可用 $n^{-2} h^{-1} \sum_{i=1}^n \sum_{j=1}^n K \cdot K\left(\frac{X_i - X_j}{h}\right)$ 估计, 其中 $K \cdot K(u) = \int_t K(u-t) K(t) dt$ 是卷积.

所以, Rudemo 和 Bowman 提出的交叉验证法 (cross validation) 实际上是选择 h 使下一步

$$\text{ISE}(h)_1 = n^{-2} h^{-1} \sum_{i=1}^n \sum_{j=1}^n K \cdot K\left(\frac{X_i - X_j}{h}\right) - 2n^{-1} \sum_{i=1}^n \hat{p}_{-i}(X_i)$$

达到最小. 当 K 是标准正态密度函数时, $K \cdot K$ 是 $N(0, 2)$ 密度函数, 有

$$\begin{aligned} \text{ISE}(h)_1 &= \frac{1}{2\sqrt{\pi}n^2h} \sum_i \sum_j \exp\left[-\frac{1}{4}\left(\frac{X_i - X_j}{h}\right)^2\right] \\ &\quad - \frac{2}{\sqrt{2\pi}n(n-1)h} \sum_i \sum_{j \neq i} \exp\left[-\frac{1}{2}\left(\frac{X_i - X_j}{h}\right)^2\right]. \end{aligned}$$

多维情形 前面考虑的是一维情况下的核密度估计, 下面考虑多维情形.

定义 9.2.18. 假设数据 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ 是 d 维向量, 并取自一个连续分布 $p(\mathbf{x})$, 在任意点 \mathbf{x} 处的一种核密度估计定义为

$$\hat{p}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$

其中 h 是带宽, K 是定义在 d 维空间上的核函数, 即 $K : \mathbb{R}^d \rightarrow \mathbb{R}$, 并满足如下条件:

$$K(\mathbf{x}) \geq 0, \quad \int K(\mathbf{x}) d\mathbf{u} = 1.$$

类似于一维情况, 可以证明 $\int_{\mathbb{R}^d} \hat{p}(\mathbf{x}) d\mathbf{x} = 1$, 即 $\hat{p}(\mathbf{x})$ 是一个密度估计.

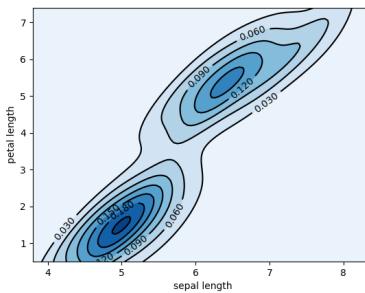


图 9.7: 鸢尾花数据集花萼长度和花瓣长度

常用的多维核函数 对于核函数的选择, 我们经常选取对称的多维密度函数来作为核函数. 例如可以选取多维标准正态密度函数来作为核函数, $K_n(\mathbf{x}) = (2\pi)^{-d/2} \exp(-\mathbf{x}^T \mathbf{x}/2)$. 其他常用的核函数还有

- $K_2(\mathbf{x}) = 3\pi^{-1} (1 - \mathbf{x}^T \mathbf{x})^2 I(\mathbf{x}^T \mathbf{x} < 1)$
- $K_3(\mathbf{x}) = 4\pi^{-1} (1 - \mathbf{x}^T \mathbf{x})^3 I(\mathbf{x}^T \mathbf{x} < 1)$
- $K_e(\mathbf{x}) = \frac{1}{2} c_d^{-1} (d+2) (1 - \mathbf{x}^T \mathbf{x}) I(\mathbf{x}^T \mathbf{x} < 1)$.

K_e 被称为多维 Epanechnikov 核函数, 其中 c_d 是一个和维度有关的常数, $c_1 = 2$, $c_2 = \pi$, $c_3 = 4\pi/3$.

最优带宽 上述的多维核密度估计中, 我们只使用了一个带宽参数 h , 这意味着在不同方向上, 我们取的带宽是一样的. 事实上, 我们可以对不同方向取不同的带宽参数, 即

$$\hat{p}(\mathbf{x}) = \frac{1}{nh_1 \cdots h_d} \sum_{i=1}^n K\left(\frac{\mathbf{x} - \mathbf{x}_i}{\mathbf{h}}\right)$$

其中, $\mathbf{h} = (h_1, h_2, \dots, h_d)$ 是一个 d 维向量. 在实际数据中, 有时候一个维度上的数据比另外一个维度上的数据分散得多, 这个时候上述的核函数就有用了. 比如说数据在一个维度上分布在 $(0, 100)$ 区间上, 而在另一个维度上仅仅分布在区间 $(0, 1)$ 上, 这时候采用不同带宽的多维核函数就比较合理了.

例 9.2.18. 下例是鸢尾花数据集中的数据, 它包含 150 对数据, 分别为鸢尾花数据集花萼长度和花瓣长度. 我们以此数据估计花萼长度和花瓣长度的联合密度函数.

关于最优带宽的选择, 我们也有类似一维情况下的结论. 对于多维核密度估计, 利用多维泰

勒展开, 有

$$\text{Bias}(\hat{p}(\mathbf{x})) \approx \frac{1}{2}h^2\alpha\nabla^2p(\mathbf{x}),$$

$$V(\hat{p}(\mathbf{x})) \approx n^{-1}h^{-d}\beta p(\mathbf{x}).$$

其中, $\alpha = \int \mathbf{x}^2 K(\mathbf{x}) d\mathbf{x}$, $\beta = \int K(\mathbf{x})^2 d\mathbf{x}$. 因此我们可以得到渐进积分均方误差

$$\text{AMISE} = \frac{1}{4}h^4\alpha^2 \int \nabla^2 p(\mathbf{x}) d\mathbf{x} + n^{-1}h^{-d}\beta.$$

由此可得最优带宽为

$$h_{\text{opt}} = \left\{ d\beta\alpha^{-2} \left(\int \nabla^2 p(\mathbf{x}) d\mathbf{x} \right) \right\}^{1/(d+4)} n^{-1/(d+4)}.$$

在上述的最优带宽中, 真实密度 $p(\mathbf{x})$ 是未知的, 因此我们可以采用多维正态密度 $\phi(\mathbf{x})$ 来代替, 进而得到

$$h_{\text{opt}} = A(K)n^{-1/(d+4)}$$

其中 $A(K) = \{d\beta\alpha^{-2} (\int \nabla^2 \phi(\mathbf{x}) d\mathbf{x})\}^{1/(d+4)}$. 对于 $A(K)$, 在知道估计中的核函数类型后, 可以计算出来, 并进而得到最优带宽 h_{opt} . 以下是不同核函数的 $A(K)$:

Kernel	Dimensionality	$A(K)$
K_n	2	1
K_n	d	$\{4/(d+2)\}^{1/(d+4)}$
K_e	2	2.40
K_e	3	2.49
K_e		$\{8c_d^{-1}(d+4)(2\sqrt{\pi})\}^{1/(d+4)}$
K_2	2	2.78
K_3	2	3.12

贝叶斯决策和非参数估计 在机器学习领域, 分类是一个基本的任务。在统计学中, 分类被看成一个决策。分类决策是对一个概念的归属作决定的过程。一个分类框架一般由 4 项基本元素构成:

1. 参数集: 概念所有可能的不同自然状态. 在分类问题中, 自然参数是可数个, 用 $\theta = \{\theta_0, \theta_1, \dots\}$ 表示.
2. 决策集: 所有可能的决策结果 $\mathcal{A} = \{a\}$. 比如: 买或卖、是否癌症、是否为垃圾邮件, 在分类问题中, 决策结果就是决策类别的归属, 所以决策集与参数集往往是一致的.
3. 决策函数集: $\Delta = \{\delta\}$, 函数 $\delta: \theta \rightarrow \mathcal{A}$.
4. 损失函数: 联系于参数和决策之间的一个损失函数. 如果概念和参数都是有限可数的, 那么所有的概念和相应的决策所对应的损失就构成了一个矩阵.

例 9.2.19. 两类问题中, 真实的参数集为 θ_1 和 θ_0 (分别简记为 1 或 0), 可能的决策集由 4 个可能的决策构成 $\Delta = \{\delta_{1,1}, \delta_{0,0}, \delta_{0,1}, \delta_{1,0}\}$. 其中, $\delta_{i,j}$ 表示把 i 判为 j , $i, j = 0, 1$, 相应的损失矩阵可能为

$$L = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

这表示判对没有损失, 判错有损失. 真实的情况为 1 判为 0, 或真实的情况为 0 判为 1, 则发生损失 1, 称为 “0-1” 损失.

从分布的角度来看, 分类问题本质上是概念属性分布的辨识问题, 于是可能通过密度估计回答概念归属的问题. 以两类问题为例: 真实的参数集为 θ_1 和 θ_0 , 在没有观测之前, 对 θ_1 和 θ_0 的决策函数可以应用先验 $p(\theta_1)$ 和 $p(\theta_0)$ 确定, 即定义决策函数

$$\delta = \begin{cases} \theta_1, & p(\theta_1) > p(\theta_0), \\ \theta_0, & p(\theta_1) < p(\theta_0). \end{cases}$$

很多情况下, 我们对概念能够收集到更多的观测数据, 于是可以建立类条件概率密度 $p(x | \theta_1)$, $p(x | \theta_0)$. 显然, 两个不同的概念在一些关键属性上一定存在差异, 这表现为两个类别在某些属性上面分布呈现差异. 综合先验信息, 可以对类别的归属通过贝叶斯公式重新组织, 即

$$p(\theta_1 | x) = \frac{p(x | \theta_1) p(\theta_1)}{p(x)}, \quad p(\theta_0 | x) = \frac{p(x | \theta_0) p(\theta_0)}{p(x)}.$$

根据贝叶斯公式, 我们可以通过后验分布制定决策:

$$\delta = \begin{cases} \theta_1, & p(\theta_1 | x) > p(\theta_0 | x), \\ \theta_0, & p(\theta_1 | x) < p(\theta_0 | x). \end{cases}$$

注意到后验概率比较中, 本质的部分是分子, 所以上式等价于

$$\delta = \begin{cases} \theta_1, & p(x | \theta_1) p(\theta_1) > p(x | \theta_0) p(\theta_0), \\ \theta_0, & p(x | \theta_1) p(\theta_1) < p(x | \theta_0) p(\theta_0). \end{cases}$$

定理 9.2.13. 后验概率最大化分类决策是 “0-1” 损失下的最优风险.

证明. 注意到条件风险

$$R(\theta_1 | x) = p(\theta_0 | x) L(\theta_0, \theta_1) + p(\theta_1 | x) L(\theta_1, \theta_1) = 1 - p(\theta_1 | x).$$

□

上述定理很容易扩展到 $k, k \geq 3$ 个不同的分类. 于是给出如下的非参数核密度估计分类计算步骤 (后验分布构造贝叶斯分类) :

1. $\forall i = 1, 2, \dots, k, \theta_i$ 下观测 $x_{i1}, x_{i2}, \dots, x_{in} \sim p(x | \theta_i)$;
2. 估计 $p(\theta_i), i = 1, 2, \dots, k$;

3. 估计 $p(x | \theta_i), i = 1, 2, \dots, k$;
4. 对新待分类点 x , 计算 $p(x | \theta_i) p(\theta_i)$;
5. 计算 $\theta^* = \operatorname{argmax} \{p(x | \theta_i) p(\theta_i)\}$.

例 9.2.20. 根据核密度估计贝叶斯分类对前面例题中提及的两类鸢尾花进行分类。

解. 假设 θ_0 表示山鸢尾, θ_1 表示维吉尼亚鸢尾, 记两类花的先验分布为

$$\text{山鸢尾: } \hat{p}(\theta_0) \leftrightarrow \text{维吉尼亚鸢尾: } \hat{p}(\theta_1).$$

用两类分别占用全部数据的频率估计先验概率。在本例中, 由于山鸢尾和维吉尼亚鸢尾各为一半, 两类的先验概率分别估计为 $\hat{p}(\theta_0) = \hat{p}(\theta_1) = 0.5$ 。然后, 对每一类考虑用核概率密度估计类条件概率: 山鸢尾: $\hat{p}(x | \theta_0) \leftrightarrow$ 维吉尼亚鸢尾: $\hat{p}(x | \theta_1)$.

最后, 根据最大后验概率进行分类:

$$\forall x, \quad \delta_x \in \begin{cases} \theta_0, & \text{当 } p(\theta_0 | x) > p(\theta_1 | x), \\ \theta_1, & \text{当 } p(\theta_1 | x) > p(\theta_0 | x). \end{cases}$$

下面我们针对一组数据点, 得到如表所示的分类结果:

表 9.5: 核密度估计贝叶斯分类结果

数值	$p^*(\theta_0 x)$	$p^*(\theta_1 x)$	真实的类别	判断的类别
5	0.9916	0.0358	0	0
7.1	0.0000	0.3140	1	1
4.4	0.3535	0.0018	0	0
4.9	0.9489	0.0400	1	0
6.5	0.0003	0.6855	1	1
5.1	0.9554	0.0267	0	0
7.2	0.0000	0.28563	1	1
5.0	0.9916	0.0358	0	0

注: p^* 表示没有归一化的分布密度

上述的概率密度估计和分类的例子已经较好地说明了非参数密度估计的优点。如果能采集足够多的训练样本, 无论实际采取哪一种核函数形式, 从理论上最终可以得到一个可靠的收敛于密度的估计结果。概率密度估计和分类例子的主要缺点是为了获得满意的密度估计, 实际需要的样本量却是非常惊人的。非参数估计要求的样本量远超过在已知分布参数形式下估计所需要的样本量。这种方法对时间和内存空间的消耗都是巨大的, 人们也正在努力寻找有效降低估计样本量的方法。

然而, 非参数密度估计最严重的问题是高维应用问题. 一般在高维空间上, 会考虑定义一个 d 维核函数为一维核函数的乘积, 每个核函数有自己的带宽, 记为 h_1, h_2, \dots, h_d , 参数数量与空间维数呈线性关系. 然而在高维空间中, 任何一个点的邻域里没有数据点是很正常的, 因而出现了体积很小的邻域中的任意两个点之间的距离却很远, 比如 10 维空间上位于一个体积为 0.001 的小邻域内的两个点的距离可以允许高到 0.5, 这样基于体积概念定义的核函数没有样本点估计. 这种现象被称为“维数灾难”问题 (curse of dimensionality). 为了使核估计能够应用, 则需要更多的样本作为代价. 因此这也严重限制了非参数密度估计在高维空间上的应用.

k 近邻估计

Parzen 窗估计一个潜在的问题是每个点都选用固定的体积. 如果 h_n 定的过大, 则那些分布较密的点由于受到过多点的支持, 使得本应突出的尖峰变得扁平; 而对于另一些相对稀疏的位置或离群点, 则可能因为体积设定过小, 而没有样本点纳入邻域, 从而使密度估计为零. 虽然可能选择像正态密度等一些连续核函数, 能够在一定程度上弱化该问题, 但很多情况下并不具有实质性的突破, 仍然没有一个标准指明应该按照哪些数据的分布情况制定带宽.

一种可行的解决方法就是让体积成为样本的函数, 不硬性规定窗函数为全体样本个数的某个函数, 而是固定贡献的样本点数, 以点 \mathbf{x} 为中心, 令体积扩张, 直到包含进 k_n 个样本为止, 其中的 k_n 是关于 n 的某一个特定函数. 被吸收到邻域中的样本就称为点 \mathbf{x} 的 k_n 个最近邻. 用停止时的体积定义估计点的密度如下:

$$\hat{p}_n(\mathbf{x}) = \frac{k_n/n}{V_n}.$$

如果在点 \mathbf{x} 附近有很多样本点, 那么这个体积就相对较小, 得到很大的概率密度; 而如果在点 \mathbf{x} 附近很稀疏, 那么这个体积就会变大, 直到进入某个概率密度很高的区域, 这个体积就会停止生长, 从而概率密度比较小.

如果样本点增多, 则 k_n 也相应增大, 以防止 V_n 快速增大导致密度趋于无穷.

另一方面, 我们还希望 k_n 的增加能够足够慢, 使得为了包含进 k_n 个样本的体积能够逐渐地趋于零. 在选择 k_n 方面, Fukunaga 和 Hosteler(1973) 给出了一个计算 k_n 的公式, 对于正态分布而言:

$$k = k_0 n^{4/(d+4)}$$

式中, k_0 是常数, 与样本量 n 和空间维数 d 无关.

如果取 $k_n = \sqrt{n}$, 并且假设 $\hat{p}_n(\mathbf{x})$ 是 $p(\mathbf{x})$ 的一个较准确的估计, 那么根据上式, 有

$$V_n \approx 1/(\sqrt{np}(\mathbf{x})).$$

这与核函数中的情况是一样的. 但是这里的初始体积是根据样本数据的具体情况确定的, 而不是事先选定的. 而且不连续梯度的点常常并不出现在样本点处, 见下图.

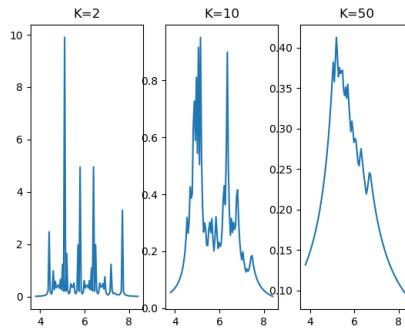


图 9.8: 鸢尾花数据集花萼长度 k_n 近邻估计图

与核函数一样, k_n 近邻估计也同样存在维度问题. 除此之外, 虽然 $\tilde{p}_n(\mathbf{x})$ 是连续的, 但 k 近邻密度估计的梯度却不一定连续. k_n 近邻估计需要的计算量相当大, 同时还要防止 k_n 增加过慢导致密度估计扩散到无穷. 这些缺点使得用 k_n 近邻法产生密度并不多见, k_n 近邻法更常用于分类问题.

9.3 概率模型与图表示

机器学习中很多模型会涉及多元随机向量的概率分布. 如果采用单个函数来描述整个随机变量的联合分布是非常低效的 (无论是计算上还是统计上), 因为这些随机变量中涉及到的直接相互作用通常只介于非常少的变量之间的. 利用随机变量之间的条件独立性关系, 可以将随机变量的联合分布分解为一些因式的乘积, 得到简洁的概率表示. 我们可以采用图论中的“图”的概率来表示这种分解, 得到概率图模型: 图中的节点表示随机变量, 边表示随机变量之间的直接作用. 有向图和无向图均可以用于表示条件独立性, 两者的主要差异是从图中读出独立性的规则不同.

下面我们首先介绍概率模型的有向图表示, 其中典型的模型有朴素贝叶斯模型和隐马尔可夫模型.

9.3.1 概率模型的有向图表示

有向图与条件独立性

定义 9.3.1. 一个有向图 G 是由节点集 V 及连接一对有序节点的边集 E 组成的。

图9.9 给出了一个有向图的例子。若 $(Y, X) \in E$, 则存在一条有向边从 Y 指向 X 。通常一

个被赋予某种概率分布的有向图常被称为贝叶斯网络，每个节点对应一个随机变量，每条边展现随机变量间的关联关系。

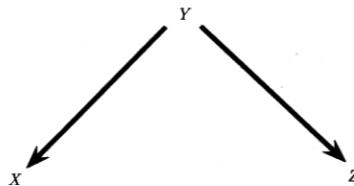


图 9.9: 节点集为 $V = \{X, Y, Z\}$ 且边集为 $E = \{\langle Y, X \rangle, \langle Y, Z \rangle\}$

图在表示变量间的独立性关系方面是很有用处的，还可以用来代替反事实去表示因果关系。在进行关于有向非循环图的讨论之前，需要先讨论一下条件独立性。

定义 9.3.2. 令 X, Y 和 Z 为随机变量。在给定 Z 的条件下，如果下式对于所有的 x, y 和 z 均成立，

$$f_{X,Y|Z}(x, y | z) = f_{X|Z}(x | z) f_{Y|Z}(y | z)$$

则 X 和 Y 称为条件独立的，记作 $X \perp\!\!\!\perp Y | Z$ 。

直观地理解，上述定义表明知道了 Z ， Y 并没有提供关于 X 的额外信息。一个等价的定义为

$$f(x, y, z) = f(x | z)$$

条件独立性具有一些基本的性质

定理 9.3.1. 下列各蕴含关系成立：

$$X \perp\!\!\!\perp Y | Z \Rightarrow Y \perp\!\!\!\perp X | Z$$

$$X \perp\!\!\!\perp Y | Z \text{ 且 } U = h(X) \Rightarrow U \perp\!\!\!\perp Y | Z$$

$$X \perp\!\!\!\perp Y | Z \text{ 且 } U = h(X) \Rightarrow X \perp\!\!\!\perp Y | (Z, U)$$

$$X \perp\!\!\!\perp Y | Z \text{ 且 } X \perp\!\!\!\perp W | (Y, Z) \Rightarrow X \perp\!\!\!\perp (W, Y) | Z$$

$$X \perp\!\!\!\perp Y | Z \text{ 且 } X \perp\!\!\!\perp Z | Y \Rightarrow X \perp\!\!\!\perp (Y, Z).$$

有向非循环图与马尔科夫条件

若一条有向边连接两个随机变量 X 和 Y (取任意一个方向)，就称 X 和 Y 是邻接的。若一条有向边从 X 指向 Y ，则称 X 是 Y 的母节点，而 Y 是 X 的子节点。 X 的所有母节点的集合记作 π_X 或 $\pi(X)$ 。两变量间的一条有向路是由一系列的同方向的有向边构成的，如下所示：

$$X \longrightarrow \dots \longrightarrow Y$$

一个从 X 开始至 Y 结束的邻接节点的序列，但是忽略其有向边的方向性，就称该序列为一个无向路。若存在一条有向路从 X 指向 Y （或 $X = Y$ ），则称 X 是 Y 的祖节点。也可以说 Y 是 X 的后裔节点。

定义 9.3.3. 如下形式的结构：

$$X \rightarrow Y \leftarrow Z$$

称作在 Y 处相遇。不具有该种形式的结构称作不相遇。

例如，

$$X \rightarrow Y \rightarrow Z$$

不相遇。相遇的性质是依赖于路的。

定义 9.3.4. 一条开始和结束都在同一个变量处的有向路是一个圈。若一个有向图没有圈，则它是非循环的。在这种情况下，称这种图为一个有向非循环图或 **DAG**。

令 \mathcal{G} 为一个具有节点集 $V = (X_1, \dots, X_k)$ 的 DAG。

定义 9.3.5. 若 P 为 V 的分布，它的概率函数为 f ，若下式成立：

$$f(v) = \prod_{i=1}^k f(x_i \mid \pi_i)$$

就说 P 是关于 \mathcal{G} 是马尔可夫的，或称 \mathcal{G} 表示 P ，其中， π_i 为 X_i 的母节点。由 \mathcal{G} 表示的分布集记为 $M(\mathcal{G})$ 。

例 9.3.1. 对于图 9.10 中的 DAG 来说， $\mathbb{P} \in M(\mathcal{G})$ 当且仅当其概率函数 f 具有以下形式：

$$f(x, y, z, w) = f(x)f(y)f(z \mid x, y)f(w \mid z)$$

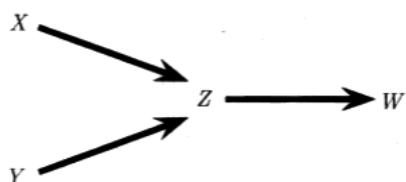


图 9.10: 另一个 DAG

下述定理表明 $\mathbb{P} \in M(\mathcal{G})$ 当且仅当马尔可夫条件成立。

定理 9.3.2. 一个分布 $\mathbb{P} \in M(\mathcal{G})$ 当且仅当下面的马尔可夫条件成立：对于每个变量 W ,

$$W \perp \widetilde{W} \mid \pi_W$$

其中， \widetilde{W} 表示除了 W 的母节点和后裔节点以外的所有其他变量。

粗略地讲，马尔可夫条件意味着每个变量 W 在给定其母节点的情况下与“过去”是独立的。

例 9.3.2. 考虑图 9.11 中的 DAG。在这种情况下，概率函数分解如下：

$$f(a, b, c, d, e) = f(a)f(b \mid a)f(c \mid a)f(d \mid b, c)f(e \mid d)$$

马尔可夫条件意味着下面的独立性关系：

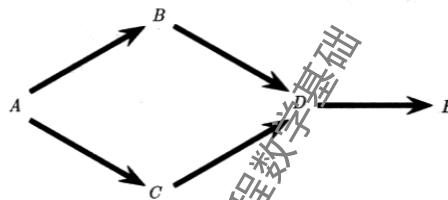


图 9.11: 另一个 DAG

$$D \perp\!\!\!\perp A \mid \{B, C\}, \quad \widetilde{W} \perp\!\!\!\perp \{A, B, C\} \mid D \text{ 且 } B \perp\!\!\!\perp C \mid A.$$

在 DAGs 中有两个首先要考虑的估计问题：第一，给定一个 DAG 为 \mathcal{G} 和来自与 \mathcal{G} 相符的分布为 f 的数据 V_1, \dots, V_n ，如何去估计 f ？第二，给定数据 V_1, \dots, V_n ，又如何去估计 \mathcal{G} ？第一个问题是一个纯粹的估计问题，而第二个问题则涉及到模型的选择。这些都是非常复杂的问题。这里仅简要介绍其主要思想，我们将在具体的模型中体会这一点。

通常，对于每个条件密度，人们常选择用某个参数模型 $f(x \mid \pi_x; \theta_x)$ ，则其似然函数为

$$\mathcal{L}(\theta) = \prod_{i=1}^n f(V_i; \theta) = \prod_{i=1}^n \prod_{j=1}^m f(X_{ij} \mid \pi_j; \theta_j)$$

其中， X_{ij} 是对于第 i 个数据点的 X_j 的值， θ_j 是第 j 个条件密度的参数。这样就可以通过极大似然方法来估计参数。

为了估计 DAG 自身的结构，几乎可以通过极大似然方法来估计每个可能的 DAG，且用 AIC (或其他的方法) 来选择一个 DAG。然而存在很多可能的 DAGs，所以需要很多数据来确保该方法是可靠的。而且从所有可能的 DAGs 中搜索是一个相当大的计算上的挑战。对于一个 DAG 结构产生一个有效的精确的置信集可能需要天文数字般的样本容量。若知道关于 DAG 的结构的部分先验信息，计算和统计上的问题至少可以部分地改善。

朴素贝叶斯模型

例 9.3.3. 设输入空间 $\mathcal{X} \subseteq R^n$ 为 n 维向量的集合, 输出空间为类标记集合 $\mathcal{Y} = \{c_1, c_2, \dots, c_K\}$ 。输入为特征向量 $x \in \mathcal{X}$, 输出为类标记 $y \in \mathcal{Y}$ 。 X 是定义在输入空间 \mathcal{X} 上的随机向量, Y 是定义在输出空间 \mathcal{Y} 上的随机变量。考虑训练数据集为

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

的分类任务, 该训练数据集由 $P(X, Y)$ 独立同分布产生。朴素贝叶斯法通过训练数据集学习联合概率分布 $P(X, Y)$ 。具体地学习以下先验概率分布和条件概率分布:

$P(Y = c_k)$, $P(X = x|Y = c_k) = P(X^{(1)} = x^{(1)}, \dots, X^{(n)} = x^{(n)}|Y = c_k)$, $k = 1, 2, \dots, K$, 再根据贝叶斯定理求得后验概率分布 $P(Y|X)$, 其中 x 为 n 维输入特征向量 $\{x^{(1)}, x^{(2)}, \dots, x^{(n)}\}$, $x \in \mathcal{X}$, c_k 为类标记。

因为条件概率分布 $P(X = x|Y = c_k)$ 有指数级数量的参数, 其估计实际是不可行的。事实上, 假设 $x^{(j)}$ 可取值有 S_j 个, $k = 1, 2, \dots, n$, Y 可取值有 K 个, 那么参数个数为 $K \prod_{j=1}^n S_j$ 。因此, 朴素贝叶斯法需要对类条件概率进行独立性假设。即:

$$P(X|Y = c_k) = \prod_j P(X^{(j)} = x^{(j)}|Y = c_k).$$

上述独立性假设, 恰好相当于假设随机变量 X 与 Y 满足如下 DAG:

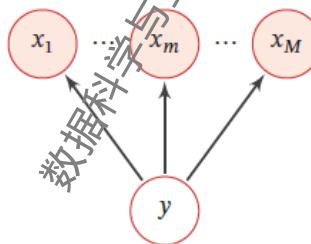


图 9.12

朴素贝叶斯法实际上学习到生成数据的机制, 所以属于生成模型。条件独立假设等于是说用于分类的特征在类确定的条件下都是条件独立的。这一假设使朴素贝叶斯法变得简单, 但有时会牺牲一定的分类准确率。

朴素贝叶斯法分类时, 对给定的输入 x , 通过学习到的模型计算后验概率分布 $P(Y = c_k|X = x)$, 将后验概率最大的类作为 x 的类输出。后验概率计算根据贝叶斯定理进行:

$$P(Y = c_k | X = x) = \frac{P(X = x | Y = c_k) P(Y = c_k)}{\sum_k P(X = x | Y = c_k) P(Y = c_k)}$$

这是朴素贝叶斯法分类的基本公式。于是, 朴素贝叶斯分类器可表示为

$$y = f(x) = \arg \max_{c_k} \frac{P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} | Y = c_k)}{\sum_k P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} | Y = c_k)}$$

注意到，在上式中分母对所有 c_k 都是相同的，所以，

$$y = \arg \max_{c_k} P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)$$

朴素贝叶斯法将实例分到后验概率最大的类中。这等价于期望风险最小化。假设选择 0-1 损失函数：

$$L(Y, f(X)) = \begin{cases} 1, & Y \neq f(X) \\ 0, & Y = f(X) \end{cases}$$

式中 $f(X)$ 是分类决策函数。这时，期望风险函数为

$$R_{\text{exp}}(f) = E[L(Y, f(X))]$$

期望是对联合分布 $P(X, Y)$ 取的。由此取条件期望

$$R_{\text{exp}}(f) = E_X \sum_{k=1}^K [L(c_k, f(X)) P(c_k \mid X)]$$

为了使期望风险最小化，只需对 $X = x$ 逐个极小化，由此得到

$$\begin{aligned} f(x) &= \arg \min_{y \in \mathcal{Y}} \sum_{k=1}^K L(c_k, y) P(c_k \mid X = x) \\ &= \arg \min_{y \in \mathcal{Y}} \sum_{k=1}^K P(y \neq c_k \mid X = x) \\ &= \arg \min_{y \in \mathcal{Y}} (1 - P(y = c_k \mid X = x)) \\ &= \arg \max_{y \in \mathcal{Y}} P(y = c_k \mid X = x) \end{aligned}$$

这样一来，根据期望风险最小化准则就得到了后验概率最大化准则：

$$f(x) = \arg \max_{c_k} P(c_k \mid X = x)$$

即朴素贝叶斯法所采用的原理。这个优化问题可采用极大似然估计或贝叶斯估计进行求解。

隐马尔可夫模型

例 9.3.4. 隐马尔可夫模型 (Hidden Markov Model, HMM) 是用来表示一种含有隐变量的马尔可夫过程，如下图所示。其中 $X_{1:T}$ 为可观测变量， $Y_{1:T}$ 为隐变量。每个可观测标量 X_t 依赖当前时刻的隐变量 Y_t ，隐变量构成一个马尔可夫链。

从定义可知，隐马尔可夫模型作了两个基本假设：一个是齐次马尔可夫性假设，即假设隐藏的马尔可夫链在任意时刻 t 的状态只依赖于其前一时刻的状态，与其他时刻的状态及观测无关，也与时刻 t 无关；另一个是观测独立性假设，即假设任意时刻的观测只依赖于该时刻的马尔可夫链的状态，与其他观测及状态无关。

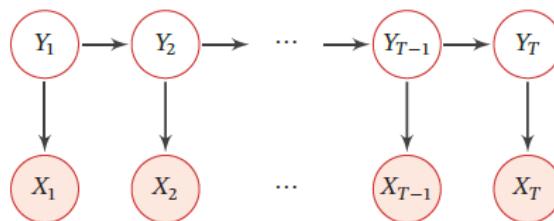


图 9.13

隐马尔可夫模型是生成模型，根据假设知，其联合概率可以分解为

$$p(\mathbf{x}, \mathbf{y}; \theta) = \prod_{t=1}^T p(y_t | y_{t-1}, \theta_s) p(x_t | y_t, \theta_t)$$

除了上述结构信息，要确定一个隐马尔可夫模型还需要以下三组参数：状态转移概率、输出观测概率和初始状态概率。在实际应用中，人们常关注隐马尔可夫模型的三个基本问题：概率计算问题、学习问题和预测问题。

对于学习问题，设所有观测数据写成 $(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{x}^{(2)}, \mathbf{y}^{(2)}), \dots, (\mathbf{x}^{(N)}, \mathbf{y}^{(N)})$ ，确定并极大化完全数据的对数似然函数，得优化问题：

$$\max_{\theta} \sum_{i=1}^N \ln p(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}; \theta)$$

若未观测到隐变量，则构建可观测变量的联合概率函数。

9.3.2 概率模型的无向图表示

机器学习中概率模型，有些可以用有向图来表示，还有一些可以用无向图来表示，一般称为概率无向图模型。概率无向图模型 (probabilistic undirected graphical models)，又称为马尔可夫随机场，是一个可以由无向图表示的联合概率分布。虽然无向图模型与有向图表达条件独立性规则不同，但是它仍能将联合概率表示分解成一组函数的乘积。它主要是借助于“团”的概念及其势函数，建立随机向量的联合概率。

接下来我们首先回顾无向图的定义，然后定义无向图表示的随机变量之间存在的成对马尔可夫性和全局马尔可夫性，最后引出团和势函数的概念以及概率无向图模型的因子分解定理。

无向图

定义 9.3.6. 一个无向图 $G = (V, E)$ 由一个有限节点集 V 和由每对节点组成的边或 (弧) 集 E 所构成。节点对应着随机变量 X, Y, Z, \dots ，而边被记作一些无序对。

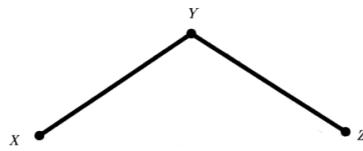


图 9.14: 节点集为 $V = \{X, Y, Z\}$ 的一个图。其边集为 $E = \{(Y, X), (Y, Z)\}$

例如, $(X, Y) \in E$ 表示 X 和 Y 通过一条边连接起来。图9.14 给出了一个无向图的例子。

定义 9.3.7. 若两个节点之间存在一条边, 则称这两个节点是邻接的, 记作 $X \sim Y$. 在图9.14 中, X 和 Y 是邻接的但是 X 和 Z 不是邻接的。若对每个 i 都有 $X_{i-1} \sim X_i$, 则序列 X_0, \dots, X_n 称为一条路。在图9.14 中, X, Y, Z 是一条路。若一个图中任意两个节点之间都存在一条边, 则称这个图是完全的 (完全图)。一个子节点集 $U \in V$ 连同其边被称作一个子图。

定义 9.3.8. 设 A, B 和 C 是 V 的不同子集, 若从 A 中的一个变量到 B 中的一个变量的路都相交于 C 中的一个变量, 就说 C 分离 A 和 B 。

例如, 在图9.15 中, $\{Y, W\}$ 和 $\{Z\}$ 被 $\{X\}$ 分离。同时, $\{W\}$ 和 $\{Z\}$ 被 $\{X, Y\}$ 分离。

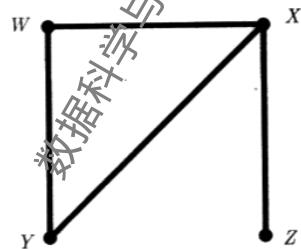


图 9.15: $\{Y, W\}$ 和 $\{Z\}$ 被 $\{X\}$ 分离。而且, $\{W\}$ 和 $\{Z\}$ 被 $\{X, Y\}$ 分离

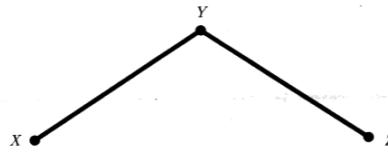
概率无向图模型

定义 9.3.9. 令 V 为具有分布 \mathbb{P} 的随机变量集。构造一个图, 其每个节点对应 V 中的每个变量。略去一对变量之间的边, 若它们在给定其余变量的条件下是独立的。即

$$X \text{ 和 } Y \text{ 之间没有边} \Leftrightarrow X \perp\!\!\!\perp Y \mid \text{其余变量},$$

其中, “其余变量” 表示除了 X 和 Y 之外的所有其他变量, 这样的图称作成对马尔可夫图。

如图9.16 所示:

图 9.16: $X \perp\!\!\!\perp Z \mid Y$ 图 9.17: 若满足成对马尔可夫性，可以得到 $X \perp\!\!\!\perp Z \mid Y$ 吗？

图中暗含着一系列的成对条件独立性关系。这些关系可以推出其他的条件独立性关系。如何从图中直接读出其他的条件独立性关系呢？事实上，有如下结论成立：

定理 9.3.3. 令 $\mathcal{G} = (V, E)$ 是一个分布为 \mathbb{P} 的成对马尔可夫图。令 A, B 和 C 为 V 的不相同的子集使得 C 分离 A 和 B ，则 $A \perp\!\!\!\perp B \mid C$ 。

若 A 和 B 不是连通的（也就是不存在一条从 A 到 B 的路），则可以把 A 和 B 看作被空集分离，则由定理 9.3.3 可知 $A \perp\!\!\!\perp B$ 。

定理 9.3.3 中的独立性条件被称作全局马尔可夫性质。将看到成对和全局马尔可夫性质是等价的。更确切的，可以描述为：给定一个图 \mathcal{G} ，令 $M_{pair}(\mathcal{G})$ 表示满足成对马尔可夫性质的分布集，因此 $P \in M_{pair}(\mathcal{G})$ ，在分布 \mathbb{P} 下，若 $X \perp\!\!\!\perp Y \mid$ 其余变量当且仅当 X 和 Y 之间不存在边；令 $M_{global}(\mathcal{G})$ 为满足全局马尔可夫性质的分布集：则 $P \in M_{global}(\mathcal{G})$ ，在分布 \mathbb{P} 下，若 $A \perp\!\!\!\perp B \mid C$ 当且仅当 C 分离 A 和 B 。

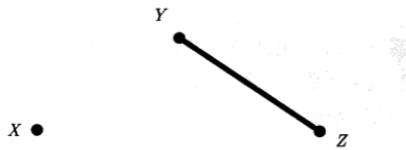
前面已知成对马尔可夫性隐含着全局马尔可夫性，反过来成立吗？实际上，成对和全局马尔可夫性质是等价的：

定理 9.3.4. 令 \mathcal{G} 为一个图，则 $M_{pair}(\mathcal{G}) = M_{global}(\mathcal{G})$ 。

上述定理保证了可以使用简单的成对性质来构建图，这就使得可以用全局马尔可夫性来推导其他独立关系。

例 9.3.5. 由图 9.18 可知 $X \perp\!\!\!\perp Y, X \perp\!\!\!\perp Z$ 和 $X \perp\!\!\!\perp (Y, Z)$ 。

定义 9.3.10. 设有联合概率分布 $P(Y)$ ，由无向图 $G = (V, E)$ 表示，在图 G 中，结点表示随机变量，边表示随机变量之间的依赖关系。如果联合概率分布 $P(Y)$ 满足成对或全局马尔可夫性，就称此联合概率分布为概率无向图模型 (*probabilistic undirected graphical model*)，或马尔可夫随机场 (*Markov random field*)。

图 9.18: $X \perp\!\!\!\perp Y$

在实际中, 我们关心的是如何求其联合概率分布。为便于模型的学习与计算, 对于给定的概率无向图模型, 我们希望将整体的联合概率写成若干子联合概率的乘积的形式, 也就是将联合概率进行因子分解。事实上, 概率无向图模型的最大特点就是易于因子分解。

概率无向图模型的因子分解

首先我们给出无向图中的团与极大团的定义。

定义 9.3.11. 若一个图的变量集中的任意两个对应的节点都是邻接的, 则称该集为一个团。若一个团任意增加一个节点后就不能成为团, 则称之为一个极大团。

例 9.3.6. 图 9.19 中的极大团为 $\{X_1, X_2\}$, $\{X_1, X_3\}$, $\{X_2, X_4\}$, $\{X_3, X_5\}$, $\{X_2, X_5, X_6\}$

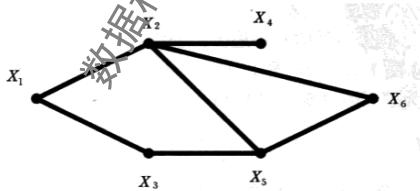


图 9.19: 极大团示例

再给出势的定义。一个势就是任意一个正函数。在特定的条件下, 可以证明分布 \mathbb{P} 关于无向图 G 是马尔可夫的当且仅当其概率函数 f 可以写作图中所有极大团 \mathcal{C} 上的函数 $\psi_C(x_C)$ 的乘积形式, 即

$$f(x) = \frac{\prod_{C \in \mathcal{C}} \psi_C(x_C)}{Z}$$

其中, \mathcal{C} 是一个极大团集, ψ_C 是一个势, 且

$$Z = \sum_x \prod_{C \in \mathcal{C}} \psi_C(x_C),$$

称为规范化因子。规范化因子保证 f 构成一个概率分布。函数 $\psi_C(x_C)$ 称为势函数，一般要求为严格正函数，通常定义为指数函数：

$$\psi_C(x_C) = \exp(-E(x_C))$$

将概率无向图模型的联合概率分布表示为其极大团上的随机变量的函数的乘积形式的操作，称为概率无向图模型的因子分解（factorization）。

定理 9.3.5. (Hammersley-Clifford 定理) 概率无向图模型的联合概率分布 $P(Y)$ 可以表示为如下形式：

$$P(Y) = \frac{1}{Z} \prod_C \Psi_C(Y_C)$$

$$Z = \sum_Y \prod_C \Psi_C(Y_C)$$

其中， C 是无向图的最大团， Y_C 是 C 的结点对应的随机变量， $\Psi_C(Y_C)$ 是 C 上定义的严格正函数，乘积是在无向图所有的最大团上进行的。

例 9.3.7. 前面已知图 9.19 中的极大团是

$$\{X_1, X_2\}, \quad \{X_1, X_3\}, \quad \{X_2, X_4\}, \quad \{X_3, X_5\}, \quad \{X_2, X_5, X_6\}$$

因此，可以把概率函数写为

$$f(x_1, x_2, x_3, x_4, x_5, x_6) \propto \psi_{12}(x_1, x_2) \psi_{13}(x_1, x_3) \psi_{24}(x_2, x_4) \times \psi_{35}(x_3, x_5) \psi_{256}(x_2, x_5, x_6)$$

例 9.3.8. 图 9.20 中对应的概率分布可以分解为

$$f(a, b, c, d, e) \propto \psi_1(a, b, c) \psi_2(b, d) \psi_3(c, e)$$

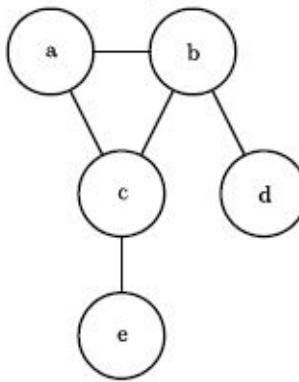


图 9.20

条件随机场模型

在实际问题中, 我们经常需要建立条件概率模型 $P(Y | X)$ 。条件随机场是给定一组输入随机变量条件下另一组输出随机变量的条件概率分布模型, 其特点是假设输出随机变量构成马尔可夫随机场。也即条件随机场是给定随机变量 X 条件下, 随机变量 Y 的马尔可夫随机场。条件随机场可以用于不同的预测问题。

定义 9.3.12. 设 X 与 Y 是随机变量, $P(Y | X)$ 是在给定 X 的条件下 Y 的条件概率分布。若随机变量 Y 构成一个由无向图 $G = (V, E)$ 表示的马尔可夫随机场, 即

$$P(Y_v | X, Y_w, w \neq v) = P(Y_v | X, Y_w, w \sim v)$$

对任意节点 v 成立, 则称条件概率分布 $P(Y | X)$ 为条件随机场。式中 $w \sim v$ 表示在图 $G = (V, E)$ 中与节点 v 有边连接的所有节点 w , $w \neq v$ 表示节点 v 以外的所有节点, Y_v 与 Y_w 为节点 v 与 w 对应的随机变量。

在条件随机场的定义中, 并未对无向图的结构进行任何假定, 也没有要求 X 和 Y 具有相同的结构。现实中, 一般假定 X 和 Y 有相同的图结构。这里, 我们考虑无向图具有下图所示的线性链结构, 即

$$\mathcal{G} = (V = \{1, 2, \dots, n\}, E = \{(i, i+1)\}), \quad i = 1, 2, \dots, n-1$$

在此情况下, $X = (X_1, X_2, \dots, X_n)$, $Y = (Y_1, Y_2, \dots, Y_n)$, 最大团是相邻两个节点的集合。

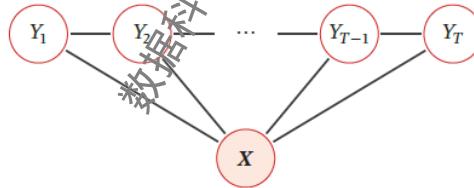


图 9.21

这种具有线性链结构的特殊的条件随机场, 称为线性链条件随机场模型。

定义 9.3.13. 设 $X = (X_1, X_2, \dots, X_n)$ 与 $Y = (Y_1, Y_2, \dots, Y_n)$ 均为线性链表示的随机变量序列, 若在给定随机变量序列 X 的条件, 随机变量序列 Y 的条件概率分布 $P(Y | X)$ 构成条件随机场, 即满足马尔可夫性

$$P(Y_i | X, Y_1, \dots, Y_{i-1}, Y_{i+1}, \dots, Y_n) = P(Y_i | X, Y_{i-1}, Y_{i+1})$$

$$i = 1, 2, \dots, n \quad (\text{在 } i = 1 \text{ 和 } n \text{ 时只考虑单边})$$

则称 $P(Y | X)$ 为线性链条件随机场。在标注问题中, X 表示输入观测序列, Y 表示对应的输出标记序列或状态序列。

根据概率无向图模型的因式分解定理, 可以给出线性链条件随机场 $P(Y|X)$ 的因子分解式, 各因子是定义在相邻两个节点 (最大团) 上的势函数。

定理 9.3.6. (线性链条件随机场的参数化形式) 设 $P(Y|X)$ 为线性链条件随机场, 则在随机变量 X 取值为 x 的条件下, 随机变量 Y 取值为 y 的条件概率具有如下形式:

$$P(y|x) = \frac{1}{Z(x)} \exp \left(\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_i, x, i) \right)$$

其中, $Z(x) = \sum_y \exp \left(\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_i, x, i) \right)$ 式中, t_k 和 s_l 是特征函数, λ_k 和 μ_l 是对应的权值。 $Z(x)$ 是规范化因子, 求和是在所有可能的输出序列上进行的。

定理中的条件概率表达式是线性链条件随机场模型的基本形式, 表示给定输入序列 x , 对输出序列 y 预测的条件概率。式中, t_k 是定义在边上的特征函数, 称为转移特征, 依赖于当前和前一个位置; s_l 是定义在结点上的特征函数, 称为状态特征, 依赖于当前位置。 t_k 和 s_l 都依赖于位置, 是局部特征函数。通常, 特征函数 t_k 和 s_l 取值为 1 或 0; 当满足特征条件时取值为 1, 否则为 0。条件随机场完全由特征函数 t_k , s_l 和对应的权值 λ_k , μ_l 确定。很显然线性链条件随机场模型是建立在条件概率分布 $P(Y|X)$ 之下, 因此是一个判别模型。

有了线性链条件随机场的定义和参数表示, 一般接下来会考虑 3 个基本的问题: 概率计算问题、学习问题和预测问题。线性链条件随机场可以用于机器学习中的标注等问题。这时, 在条件概率模型 $P(Y|X)$ 中, Y 是输出变量, 表示标记序列, X 是输入变量, 表示需要标注的观测序列。也把标记序列称为状态序列。学习时, 利用训练数据集通过极大似然估计或正则化的极大似然估计得到条件概率模型 $\hat{P}(Y|X)$ 。预测时, 对于给定的输入序列 X , 求出条件概率 $\hat{P}(y|x)$ 最大的输出序列 \hat{y} 。

已知训练数据集, 由此可知经验概率分布 $\tilde{P}(X, Y)$ 。则训练数据的对数似然函数为

$$L(\boldsymbol{\lambda}, \boldsymbol{\mu}) = L_{\tilde{P}}(P_{\boldsymbol{\lambda}, \boldsymbol{\mu}}) = \log \prod_{x,y} P_{\boldsymbol{\lambda}, \boldsymbol{\mu}}(y|x)^{\tilde{P}(x,y)} = \sum_{x,y} \tilde{P}(x,y) \log P_{\boldsymbol{\lambda}, \boldsymbol{\mu}}(y|x)$$

将线性链条件随机场的参数化形式代入上式, 可得对数似然函数为

$$\begin{aligned} L(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= \sum_{x,y} \tilde{P}(x,y) \log P_{\boldsymbol{\lambda}, \boldsymbol{\mu}}(y|x) \\ &= \sum_{j=1}^N \left(\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_i, x, i) \right) - \sum_{j=1}^N \log Z_{\boldsymbol{\lambda}, \boldsymbol{\mu}}(x_j) \end{aligned}$$

因此, 可得优化问题:

$$\max_{\boldsymbol{\lambda}, \boldsymbol{\mu}} \sum_{j=1}^N \left(\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_i, x, i) \right) - \sum_{j=1}^N \log Z_{\boldsymbol{\lambda}, \boldsymbol{\mu}}(x_j).$$

9.4 机器学习中的概率模型

9.4.1 机器学习的概率思路

在统计机器学习中, 通常假设输入与输出的随机变量 X 和 Y 遵循联合概率分布 $P(X, Y)$, $P(X, Y)$ 表示分布函数或分布密度函数. 在学习过程中, 假定这些联合概率分布存在, 但对学习系统来说, 联合概率分布的具体定义是未知的. 此外, 训练数据与测试数据被看作是依联合概率分布 $P(X, Y)$ 独立同分布产生的.

在概率模型方面, 监督学习的概率模型通常由条件概率分布 $P(Y|X)$ 来表示. 对具体的输入进行相应的输出预测时, 写作 $P(y|x)$. 无监督学习的概率模型取条件概率分布形式 $P(z|x)$ 或者 $P(x|z)$, $x \in \mathcal{X}$ 是输入, $z \in \mathcal{Z}$ 是输出, \mathcal{X} 是输入空间, \mathcal{Z} 是输出空间.

对于监督学习来说, 监督学习方法可以分为生成方法和判别方法, 所学到的模型分别称为生成模型和判别模型.

生成模型

生成方法由数据学习联合概率分布 $P(X, Y)$ 然后求出条件概率分布 $P(Y|X)$ 作为预测的模型, 即生成模型:

$$P(Y|X) = \frac{P(X, Y)}{P(X)}$$

这样的方法之所以称为生成方法, 是因为模型表示了给定输入 X 产生输出 Y 的生成关系. 典型的概率型生成模型有朴素贝叶斯法和隐马尔可夫模型等等.

判别模型

判别方法由数据直接学习决策函数 $f(X)$ 或者条件概率分布 $P(Y|X)$ 作为预测的模型, 即判别模型. 判别方法关心的是对给定的输入 X , 应该预测什么样的 Y . 典型的概率型判别模型有决策树、逻辑斯蒂回归模型、最大熵模型和条件随机场等等.

除上述内容外, 我们还需确定模型的假设空间, 假设空间的确定意味着学习的范围的确定. 概率模型的假设空间可以定义为条件概率的集合:

$$\mathcal{F} = \{P|P(Y|X)\}, \quad (9.8)$$

其中, X 和 Y 是定义在输入空间 \mathcal{X} 和输出空间 \mathcal{Y} 上的随机变量. 这时 \mathcal{F} 通常是由一个参数向量决定的条件概率分布族:

$$\mathcal{F} = \{P|P_\theta(Y|X), \theta \in R^n\}, \quad (9.9)$$

参数向量 θ 取值于 n 维欧式空间 R^n , 也称为参数空间.

在损失函数的选择上, 常用的损失函数为对数损失函数或对数似然损失函数:

$$L(Y, P(Y|X)) = -\log P(Y|X). \quad (9.10)$$

已知上述前提信息后，即可开始对模型进行训练，同样也有监督学习和无监督学习两种方法。

监督学习分为学习和预测两个过程，由学习系统和预测系统完成。在学习过程中，学习系统利用给定的训练数据集，通过学习（或训练）得到一个模型，表示为条件概率分布 $\hat{P}(Y|X)$ ，描述了输入与输出随机变量之间的映射关系。在预测过程中，预测系统对于给定的测试样本集中的输入 x_{N+1} ，由模型 $y_{N+1} = \arg \max_y \hat{P}(y|x_{N+1})$ 给出相应的输出 y_{N+1} 。

无监督学习也由学习系统和预测系统完成。在学习过程中，学习系统从训练数据集学习，得到一个最优模型，表示条件概率分布 $\hat{P}(z|x)$ 条件概率分布 $\hat{P}(x|z)$ 。在预测过程中，预测系统对于给定的输入 x_{N+1} ，由模型 $z_{N+1} = \arg \max_z \hat{P}(z|x_{N+1})$ 给出相应的输出 z_{N+1} ，进行聚类或降维，或者由模型 $\hat{P}(x|z)$ 给出输入的概率 $\hat{P}(x_{N+1}|z_{N+1})$ ，进行概率估计。

监督学习的本质是学习输入到输出的映射的统计规律。对于监督概率模型的学习，本质与概率模型的参数估计相关。无监督学习的本质是学习数据中的统计规律或潜在结构。对于无监督概率模型的学习，本质与概率模型的非参数估计相关。

在概率模型的学习和推理中还经常使用贝叶斯技巧，其主要想法是：利用贝叶斯定理，计算在给定数据条件下模型的条件概率，即后验概率，并应用这个原理进行模型的估计，以及对数据的预测。将模型、未观测要素及其参数用变量表示，使用模型的先验分布是贝叶斯学习的特点。估计的方法包括最大后验估计等各种贝叶斯估计方法。

9.4.2 机器学习中的概率模型

决策树模型

决策树是一类常见的机器学习方法。顾名思义，它是基于树结构来进行决策（比如分类或回归），这恰是人类在面临决策问题时一种很自然的处理机制。

定义 9.4.1. 决策树模型是一种对实例进行某种决策（比如分类或回归）的树形结构，其由一个根结点、若干个内部结点和若干个叶结点以及有向边组成；其中根结点包含样本全集，内部结点对应一个特征或属性，叶结点对应于决策结果（比如在分类中，对应某个具体的类）；每个结点包含的样本集合根据特征选择或属性测试的结果被划分到子结点中去。

决策树学习的目的是为了生成一颗泛化能力强，即处理未见实例能力强的决策树。其基本流程遵循“分而治之”的策略：以用决策树分类为例，从根结点开始对实例的某一个特征进行测试，根据测试结果将实例分配到其子结点；这时，每一个子结点对应着该特征的一个取值。如此递归地对实例进行测试并分配，直到达到叶结点，最后将实例分到叶结点的类中。所以在分类问题中，决策树模型表示基于特征对实例进行分类的过程。

决策树模型可以通过两种方式来表示：看成是 if-then 规则的集合或看成是定义在特征空间与类空间上的条件概率分布。下面我们将主要介绍基于条件概率分布的决策树表示与学习。

决策树可看成在给定特征条件下类的条件概率分布。这一条件概率分布定义在特征空间的一个划分上。将特征空间划分为互不相交的单元或区域，并在每个单元定义一个类的概率分布就构成了一个条件概率分布。决策树的一条路径对应于划分中的一个单元。决策树所表示的条件概率分布由各个单元给定条件下类的条件概率分布组成。

假设 X 为表示特征的随机向量， Y 表示类的随机向量，那么这个条件概率分布就可以表示为 $P(Y|X)$ 。 X 取值于给定划分下单元的集合， Y 取值于类的集合。各叶结点（单元）上的条件概率往往偏向于某一个类，即属于某一类的概率较大。决策树分类时将该结点的实例强行分到条件概率大的那一类去。

决策树学习通常包括三个步骤：特征选择、决策树的生成、决策树的修剪。由于决策树表示一个条件概率分布，所以深浅不同的决策树对应着不同复杂度的概率模型。决策树的生成对应于模型的局部选择，决策树的剪枝对应于模型的全局选择，决策树的生成只考虑局部最优，相对的决策树的剪枝则考虑全局最优。

决策树学习的关键是如何选择最优划分属性或最优的特征来划分特征空间。一般而言，随着划分过程不断进行，我们希望决策树的分支结点所包含的类别尽可能属于同一类别，即结点的“纯度”越来越高。目前主要有三类属性划分或特征选择的准则：信息增益、增益比、基尼系数。

而在决策树生成上，基于上述准则又具有以下几类方法：

1. 基于信息增益的决策树生成

现假设训练数据集为 D ， $|D|$ 表示其样本容量。设有 K 个类 C_k ， $k = 1, 2, \dots, K$ ， $|C_k|$ 为属于类 C_k 的样本个数。显然 $\sum_{k=1}^K |C_k| = |D|$ 。设某特征 A 取值为 $\{a_1, a_2, \dots, a_n\}$ ，根据特征 A 的取值将 D 划分为 n 个子集 D_1, D_2, \dots, D_n ， $|D_i|$ 为 D_i 的样本个数， $\sum_{i=1}^n |D_i| = |D|$ 。记子集 D_i 中属于类 C_k 的样本的集合为 D_{ik} ，即 $D_{ik} = D_i \cap C_k$ ， $|D_{ik}|$ 为 D_{ik} 的样本个数。因此，可以给出信息增益定义如下：

定义 9.4.2.（信息增益）特征 A 对训练数据集 D 的信息增益 $g(D, A)$ ，定义为集合 D 的经验熵 $H(D)$ 与特征 A 给定条件下 D 的经验条件熵 $H(D | A)$ 之差，即

$$g(D, A) = H(D) - H(D | A)$$

其中

$$H(D) = - \sum_{k=1}^K \frac{|C_k|}{|D|} \log_2 \frac{|C_k|}{|D|},$$

$$H(D | A) = \sum_{i=1}^n \frac{|D_i|}{|D|} H(D_i) = - \sum_{i=1}^n \frac{|D_i|}{|D|} \sum_{k=1}^K \frac{|D_{ik}|}{|D_i|} \log_2 \frac{|D_{ik}|}{|D_i|}.$$

注意这里的经验熵与经验条件熵即为前面章节所介绍的信息熵和条件熵，这里只不过是针对经验分布而言的。可以看出，决策树学习中的信息增益等价于训练数据集中类与特征的互信

息。一般而言, 信息增益越大, 则意味着使用特征 A 来进行划分所获得的“纯度提升”越大。因此, 我们可用信息增益来进行决策树的特征选择或划分属性选择, 也即求解如下最优化问题为:

$$\max_{A_i} g(D, A_i) = H(D) - H(D | A_i),$$

其中 A_i 表示样本空间的第 i 个特征。

2. 基于信息增益比的决策树生成

以信息增益准则作为划分训练数据集的特征, 存在偏向于选择取值较多的特征的问题, 为了减少这种偏好可能带来的不利影响, 可以使用信息增益比 (information gain ratio) 来选择最优特征。

定义 9.4.3. (信息增益比) 特征 A 对训练数据集 D 的信息增益比 $g_R(D, A)$, 定义为其信息增益 $g(D, A)$ 与训练数据集 D 关于特征 A 的值的熵 $H_A(D)$ 之比, 即

$$g_R(D, A) = \frac{g(D, A)}{H_A(D)}$$

其中,

$$H_A(D) = - \sum_{i=1}^n \frac{|D_i|}{|D|} \log_2 \frac{|D_i|}{|D|},$$

n 是特征 A 取值的个数。

3. 基于基尼系数的决策树生成

还可使用基尼指数对分类决策树进行最优特征选择, 基尼指数也可衡量分布或数据的不确定性, 其定义如下:

定义 9.4.4. 分类问题中, 假设有 K 个类, 样本点属于第 k 类的概率为 p_k , 则概率分布的基尼指数定义为

$$\text{Gini}(p) = \sum_{k=1}^K p_k (1 - p_k) = 1 - \sum_{k=1}^K p_k^2$$

对于给定的样本集合 D , 其基尼指数为

$$\text{Gini}(D) = 1 - \sum_{k=1}^K \left(\frac{|C_k|}{|D|} \right)^2$$

这里, C_k 是 D 中属于第 k 类的样本子集, K 是类的个数。

一般地, 基尼指数值越大, 样本集合的不确定性也就越大, 这一点与熵相似。

在决策树生成过程中, 我们需要考虑某一特征划分下数据集的基尼指数。如果样本集合 D 根据特征 A 是否取某一可能值 a 被分割成 D_1 和 D_2 两部分, 即

$$D_1 = \{(x, y) \in D \mid A(x) = a\}, \quad D_2 = D - D_1$$

则在特征 A 的条件下, 集合 D 的基尼指数定义为

$$\text{Gini}(D, A) = \frac{|D_1|}{|D|} \text{Gini}(D_1) + \frac{|D_2|}{|D|} \text{Gini}(D_2).$$

显然, 基尼指数 $\text{Gini}(D, A)$ 表示经 $A = a$ 分割后集合 D 的不确定性。

因此, 在决策树生成过程中, 我们需要在所有可能的特征以及它所有可能的切分点 a 中, 选择基尼指数最小的特征及其对应的切分点, 作为最优特征与最优切分点。此时, 最优特征和对应切分点选择的优化问题可表示为:

$$\min_{A_i, a_{ij}} \text{Gini}(D, A_i = a_{ij}),$$

其中 A_i 表示样本空间的第 i 个特征, a_{ij} 表示特征 A_i 的第 j 个可能的取值。

在优化方面, 决策树生成算法递归地产生决策树, 直到不能继续下去为止, 这样产生的决策树由于在学习时过多地考虑如何提高对训练数据的正确分类, 从而构建出分支过多、过于复杂的决策树, 因而出现过拟合的现象。剪枝是决策树学习算法对付过拟合的主要手段。

决策树的剪枝一般通过最小化决策树整体的损失函数或代价函数来实现, 也即:

$$\min_T \sum_{t=1}^{|T|} N_t H_t(T) + \alpha |T|$$

其中 t 是树 T 的叶结点, $|T|$ 代表树 T 的叶结点个数, N_t 表示具体某个叶结点的样本数, $\alpha \geq 0$ 为参数, $H_t(T)$ 表示叶节点经验熵为

$$H_t(T) = - \sum_{k=1}^K \frac{N_{tk}}{N_t} \log \frac{N_{tk}}{N_t}$$

其中, N_{tk} 表示 k 类样本点的个数, $k = 1, 2, \dots, K$ 。

由于上述优化问题中目标函数第一项实际上就是负对数似然函数:

$$\begin{aligned} \sum_{t=1}^{|T|} N_t H_t(T) &= - \sum_{t=1}^{|T|} \sum_{k=1}^K N_{tk} \log \frac{N_{tk}}{N_t} \\ &= - \log \prod_{t=1}^{|T|} \prod_{k=1}^K \left(\frac{N_{tk}}{N_t} \right)^{N_{tk}} \end{aligned}$$

因此, 上述优化问题等价于极大对数似然函数, 即优化问题:

$$\max \log \prod_{t=1}^{|T|} \prod_{k=1}^K \left(\frac{N_{tk}}{N_t} \right)^{N_{tk}}$$

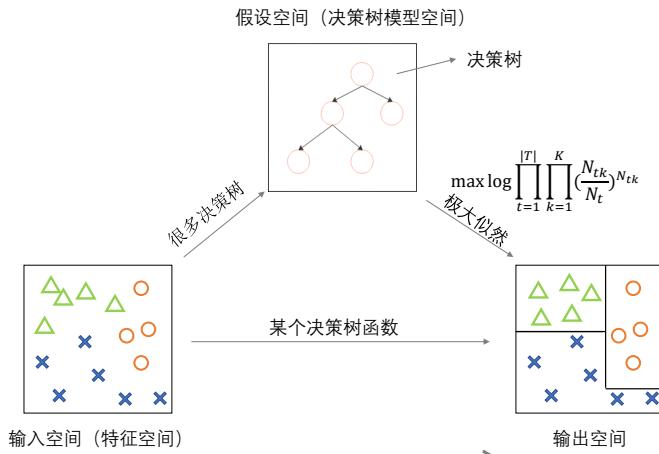


图 9.22: 从空间角度理解决策树模型

逻辑斯谛回归模型

逻辑斯谛回归是统计学习中的经典分类方法，它依赖于逻辑斯谛分布。

定义 9.4.5. (逻辑斯谛分布) 设 X 是连续随机变量, X 服从逻辑分布是指 X 具有下列分布函数和密度函数:

$$F(x) = P(X \leq x) = \frac{1}{1 + e^{-(x-\mu)/\gamma}} \quad (9.11)$$

$$f(x) = F'(x) = \frac{e^{-(x-\mu)/\gamma}}{\gamma(1 + e^{-(x-\mu)/\gamma})^2} \quad (9.12)$$

式中, μ 为位置参数, $\gamma > 0$ 为形状参数。

逻辑斯谛分布的分布函数属于逻辑函数, 其图形是一条关于点 $(\mu, \frac{1}{2})$ 为中心对称的 sigmoid 曲线。该曲线在中心附近增长速度较快, 在两端增长速度较慢。

二项逻辑斯谛回归模型是一种分类模型, 由条件概率分布 $P(Y|X)$ 表示, 形式为参数化的逻辑斯谛分布。这里, 随机变量 X 取值为实数, 随机变量 Y 的取值为 1 和 0。

定义 9.4.6. 二项逻辑斯谛回归模型是如下的条件概率分布：

$$P(Y = 1|x) = \frac{\exp(w \cdot x + b)}{1 + \exp(w \cdot x + b)} \quad (9.13)$$

$$P(Y = 0|x) = \frac{1}{1 + \exp(w \cdot x + b)} \quad (9.14)$$

其中， $x \in R^n$ 是输入， $Y \in \{0, 1\}$ 是输出， $w \in R^n$ 和 $b \in R$ 是参数， w 称为权值向量， b 称为偏置。

为了更简洁地表示，有时会将权值向量和输入向量加以扩充，记作 $w = (w^{(1)}, w^{(2)}, \dots, w^{(n)}, b)^T$ ， $x = (x^{(1)}, x^{(2)}, \dots, x^{(n)}, 1)^T$ 。此时逻辑回归模型如下：

$$P(Y = 1|x) = \frac{\exp(w \cdot x)}{1 + \exp(w \cdot x)} \quad (9.15)$$

$$P(Y = 0|x) = \frac{1}{1 + \exp(w \cdot x)} \quad (9.16)$$

在逻辑斯谛回归模型中，输出 $Y = 1$ 的对数概率是输入 x 的线性函数。线性函数的值越接近正无穷，概率值就越接近 1；线性函数的值越接近负无穷，概率值就越接近 0。给定的输入实例 x ，按照式(9.13)和(9.14)可以求得 $P(Y = 1|x)$ 和 $P(Y = 0|x)$ 。逻辑斯谛回归比较两个条件概率值的大小，将实例 x 分到概率值较大的那一类。

在模型求解上，对于给定的训练数据集 $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ，其中， $x_i \in R^n$ ， $y_i \in \{0, 1\}$ ，可以应用极大似然估计法估计模型参数。设

$$P(Y = 1|x) = \pi(x), \quad P(Y = 0|x) = 1 - \pi(x)$$

则似然函数为

$$\prod_{i=1}^N [\pi(x_i)]^{y_i} [1 - \pi(x_i)]^{1-y_i}$$

对数似然函数为

$$\begin{aligned} L(w) &= \sum_{i=1}^N [y_i \log \pi(x_i) + (1 - y_i) \log (1 - \pi(x_i))] \\ &= \sum_{i=1}^N \left[y_i \log \frac{\pi(x_i)}{1 - \pi(x_i)} + \log (1 - \pi(x_i)) \right] \\ &= \sum_{i=1}^N [y_i (w \cdot x_i) - \log (1 + \exp(w \cdot x_i))] \end{aligned}$$

因此，得到优化问题：

$$\max_w : \sum_{i=1}^N [y_i (w \cdot x_i) - \log (1 + \exp(w \cdot x_i))]$$

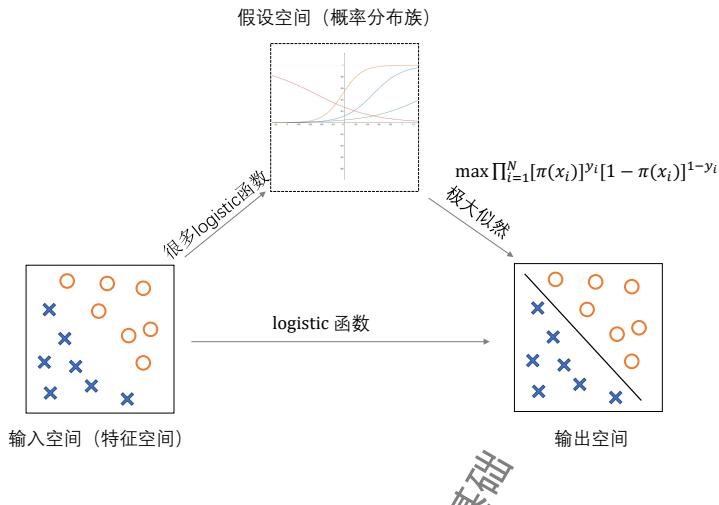


图 9.23: 从空间角度理解 Logistic 模型

最大熵模型

最大熵模型 (Maximum Entropy Model) 由最大熵原理推导实现。最大熵原理是概率模型学习的一个准则，它是指在学习概率模型时，在所有可能的概率模型 (分布) 中，熵最大的模型是最好的模型。通常用约束条件来确定概率模型的集合，因此最大熵原理也可以被表述为在满足约束条件的模型集合中选取熵最大的模型。

定义 9.4.7. (最大熵) 假设离散随机变量 X 的概率分布是 $P(X)$ ，则其熵是

$$H(P) = - \sum_x P(x) \log P(x) \quad (9.17)$$

熵满足下列不等式：

$$0 \leq H(P) \leq \log |X| \quad (9.18)$$

式中， $|X|$ 是 X 的取值个数，当且仅当 X 的分布是等概分布时右边等号成立，即当 X 服从均匀分布时熵最大。

将最大熵原理应用到分类得到最大熵分类模型，它是一个判别模型。假设分类模型是一个条件概率分布 $P(Y | X)$ ， $X \in \mathcal{X} \subseteq \mathbb{R}^n$ 表示输入， $Y \in \mathcal{Y}$ 表示输出， \mathcal{X} 和 \mathcal{Y} 分别是输入和输出的集合。这个模型表示的是对于给定的输入 X ，以条件概率 $P(Y | X)$ 输出 Y 。

通常模型还需要满足一定的约束条件。给定一个训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ，可以确定联合分布 $P(X, Y)$ 的经验分布 $\tilde{P}(x, y)$ 和边缘分布 $P(X)$ 的经验分布 $\tilde{P}(x)$ ，这里分别

用训练数据中样本出现的频率和输入数据的频率来表示。用特征函数 $f(x, y)$ 描述输入 x 和输出 y 之间的某一个事实。如果模型能够获取训练数据中的信息，那么就可以假设特征函数关于经验分布 $\tilde{P}(X, Y)$ 的期望值 $E_{\tilde{P}}(f)$ 与特征函数关于模型 $P(Y|X)$ 与经验分布 $\tilde{P}(X)$ 的期望值 $E_P(f)$ 相等，即

$$\sum_{x,y} \tilde{P}(x, y) f(x, y) = \sum_{x,y} \tilde{P}(x) P(y|x) f(x, y), \quad (9.19)$$

以此作为模型学习的约束条件。假如有 n 个特征函数 $f_i(x, y), i = 1, 2, \dots, n$ ，那么就有 n 个约束条件。

定义 9.4.8. 假设满足所有约束条件的模型集合为

$$\mathcal{C} \equiv \{P \in \mathcal{P} \mid E_P(f_i) = E_{\tilde{P}}(f_i), \quad i = 1, 2, \dots, n\} \quad (9.20)$$

定义在条件概率分布 $P(Y|X)$ 上的条件熵为

$$H(P) = - \sum_{x,y} \tilde{P}(x) P(y|x) \log P(y|x) \quad (9.21)$$

则模型集合 \mathcal{C} 中条件熵 $H(P)$ 最大的模型称为最大熵模型，式中的对数为自然对数。

最大熵模型的学习可以形式化为约束优化问题。

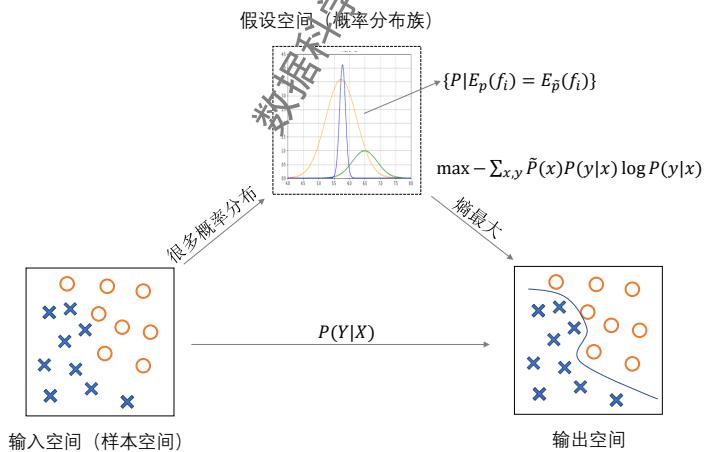


图 9.24: 从空间角度理解最大熵模型

对于给定的训练数据集 $T = (x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ 以及特征函数 $f_i(x, y), i = 1, 2, \dots, n$ ，

最大熵模型的学习等价于约束最优化问题：

$$\max_{P \in C} H(P) = - \sum_{x,y} \tilde{P}(x)P(y|x) \log P(y|x) \quad (9.22)$$

$$s.t. E_p(f_i) = E_{\tilde{P}}(f_i), i = 1, 2, \dots, n \quad (9.23)$$

$$\sum_y P(y|x) = 1 \quad (9.24)$$

通常将求最大值问题改写为等价的求最小值问题：

$$\min_{P \in C} -H(P) = \sum_{x,y} \tilde{P}(x)P(y|x) \log P(y|x) \quad (9.25)$$

$$s.t. E_p(f_i) - E_{\tilde{P}}(f_i) = 0, i = 1, 2, \dots, n \quad (9.26)$$

$$\sum_y P(y|x) = 1 \quad (9.27)$$

所得解即为最大熵模型学习的解。

9.4.3 深度学习中的概率模型

受限玻尔兹曼机

受限玻尔兹曼机是一种借助隐变量来描述复杂数据分布的概率图模型，在对复杂数据分布进行建模时，可以有效挖掘和学习出可观测变量之间复杂的依赖关系。

定义 9.4.9. 受限玻尔兹曼机 (Restricted Boltzmann Machine, RBM) 是一个二分图结构的无向图模型，如图所示。分别用可观测层和隐藏层来表示这两组变量。同一层中的节点之间没有连接，而不同层一个层中的节点与另一层中的所有节点连接，这和两层的全连接神经网络的结构相同。

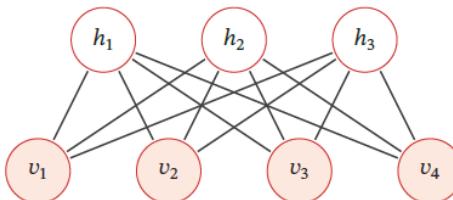


图 9.25: RBM 无向图模型

受限玻尔兹曼机模型是一个生成模型，用于生成联合分布 $p(\mathbf{v}, \mathbf{h})$ ，这里的 \mathbf{v} 和 \mathbf{h} 分别表示可观测的随机向量和隐藏的随机向量。若一个受限玻尔兹曼机由 K_v 个可观测变量和 K_h 个隐

变量组成, 权重矩阵为 $\mathbf{W} \in \mathbb{R}^{K_u \times K_h}$, 其中每个元素 w_{ij} 为可观测变量 v_i 和隐变量 h_j 之间边的权重. 偏置为 $\mathbf{a} \in \mathbb{R}^{K_v}$ 和 $\mathbf{b} \in \mathbb{R}^{K_h}$, 其中 a_i 为每个可观测的变量 v_i 的偏置, b_j 为每个隐变量 h_j 的偏置. 因此, 受限玻尔兹曼机的能量函数定义为

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{a}^\top \mathbf{v} - \mathbf{b}^\top \mathbf{h} - \mathbf{v}^\top \mathbf{W} \mathbf{h}$$

对应的联合概率分布 $p(\mathbf{v}, \mathbf{h})$ 定义为

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})),$$

其中 $Z = \sum \exp(-E(\mathbf{v}, \mathbf{h}))$ 为配分函数.

在给定受限玻尔兹曼机的联合分布 $p(\mathbf{v}, \mathbf{h})$ 后, 通常可以使用吉布斯采样方法生成服从该分布的样本。

给出了模型的表示之后, 作为概率图模型, 受限玻尔兹曼机主要涉及推断和学习两类问题。其中, 对于参数学习, 受限玻尔兹曼机是通过最大化似然函数来找到最优的参数 $\mathbf{W}, \mathbf{a}, \mathbf{b}$. 给定一组训练样本 $\mathcal{D} = \{\hat{\mathbf{v}}^{(1)}, \hat{\mathbf{v}}^{(2)}, \dots, \hat{\mathbf{v}}^{(N)}\}$, 其对数似然函数为

$$\mathcal{L}(\mathcal{D}; \mathbf{W}, \mathbf{a}, \mathbf{b}) = \frac{1}{N} \sum_{n=1}^N \log p(\hat{\mathbf{v}}^{(n)}; \mathbf{W}, \mathbf{a}, \mathbf{b}).$$

因此, 得到优化问题:

$$\max \frac{1}{N} \sum_{n=1}^N \log p(\hat{\mathbf{v}}^{(n)}; \mathbf{W}, \mathbf{a}, \mathbf{b}).$$

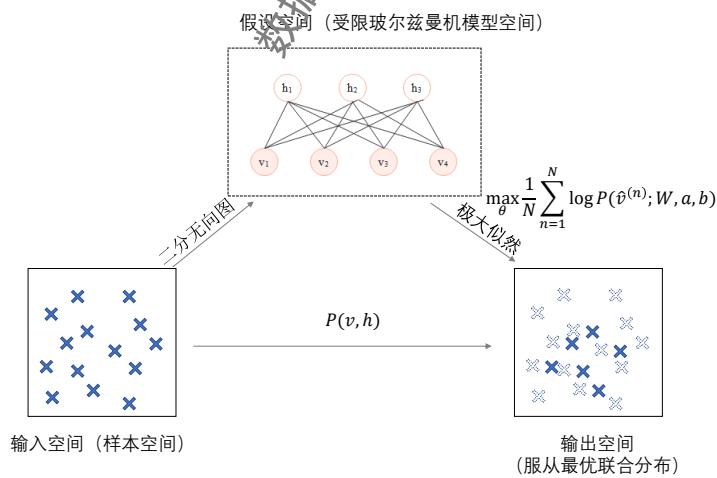


图 9.26: 从空间角度理解受限玻尔兹曼机

对于生成模型, 一般可以借助吉布斯采样的方法, 生成服从对应联合分布的样本。吉布斯采样 (Gibbs Sampling) 是一种有效地对高维空间中的分布进行采样的方法。吉布斯采样使用全条件概率作为提议分布来依次对每个维度进行采样, 并设置接受率为 1。对于一个 M 维的随机向量 $\mathbf{X} = [X_1, X_2, \dots, X_M]^\top$, 其第 m 个变量 X_m 的全条件概率为

$$p(x_m | \mathbf{x}_{\setminus m}),$$

其中 $\mathbf{x}_{\setminus m} = [x_1, x_2, \dots, x_{m-1}, x_{m+1}, \dots, x_M]^\top$ 表示除 X_m 外其他变量的取值。吉布斯采样可以按照任意的顺序根据全条件分布依次对每个变量进行采样。

吉布斯采样的每单步采样也构成一个马尔可夫链。假设每个单步 (采样维度为第 m 维) 的状态转移概率 $q(\mathbf{x} | \mathbf{x}')$ 为

$$q(\mathbf{x} | \mathbf{x}') = \begin{cases} \frac{p(\mathbf{x})}{p(\mathbf{x}'_{\setminus m})} & \text{if } \mathbf{x}_{\setminus m} = \mathbf{x}'_{\setminus m} \\ 0 & \text{otherwise,} \end{cases}$$

其中边际分布 $p(\mathbf{x}'_{\setminus m}) = \sum_{x_m'} p(\mathbf{x}')$ 。因此有 $p(x'_{\setminus m}) \equiv p(x_{\setminus m})$, 并可以得到

$$p(\mathbf{x}') q(\mathbf{x} | \mathbf{x}') = p(\mathbf{x}') \frac{p(\mathbf{x})}{p(\mathbf{x}'_{\setminus m})} = p(\mathbf{x}) \frac{p(\mathbf{x}')}{p(\mathbf{x}_{\setminus m})} = p(\mathbf{x}) q(\mathbf{x}' | \mathbf{x}).$$

根据第七章细致平衡条件可知, 该采样构成的马尔可夫链的平稳分布为 $p(\mathbf{x})$ 。

深度信念网络

定义 9.4.10. 深度信念网络 (Deep Belief Network, DBN) 是一种深层的概率有向图模型, 其图结构由多层的节点构成。每层节点的内部没有连接, 相邻两层的节点之间为全连接。网络的最底层为可观测变量, 其他层节点都为隐变量。最顶部的两层间的连接是无向的, 其他层之间的连接是有向的。

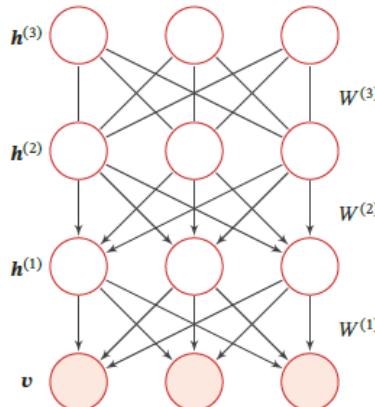


图 9.27: 深度信念网络模型

深度信念网络也是一种生成模型，它所有变量的联合概率可以分解为

$$\begin{aligned} p(\mathbf{v}, \mathbf{h}^{(1)}, \dots, \mathbf{h}^{(L)}) &= p(\mathbf{v} | \mathbf{h}^{(1)}) \left(\prod_{l=1}^{L-2} p(\mathbf{h}^{(l)} | \mathbf{h}^{(l+1)}) \right) p(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)}) \\ &= \left(\prod_{l=0}^{L-2} p(\mathbf{h}^{(l)} | \mathbf{h}^{(l+1)}) \right) p(\mathbf{h}^{(L-1)}, \mathbf{h}^{(L)}) \end{aligned}$$

其中 $\mathbf{h}^{(0)} = \mathbf{v}$, $p(\mathbf{h}^{(l)} | \mathbf{h}^{(l+1)})$ 为 Sigmoid 型条件概率分布, 定义为

$$p(\mathbf{h}^{(l)} | \mathbf{h}^{(l+1)}) = \sigma(\mathbf{a}^{(l)} + \mathbf{W}^{(l+1)} \mathbf{h}^{(l+1)}),$$

其中 $\sigma(\cdot)$ 为按位计算的 Logistic 函数, $\mathbf{a}^{(l)}$ 为偏置参数, $\mathbf{W}^{(l+1)}$ 为权重参数. 这样, 每一个层都可以看作一个 Sigmoid 信念网络.

深度信念网络也是通过最大化似然函数来找到最优的参数 $\mathbf{W}^{(1)}, \dots, \mathbf{W}^{(L)}, \mathbf{a}^{(1)}, \dots, \mathbf{a}^{(L)}$. 给定一组训练样本 $\mathcal{D} = \{\hat{\mathbf{v}}^{(1)}, \hat{\mathbf{v}}^{(2)}, \dots, \hat{\mathbf{v}}^{(N)}\}$, 其对数似然函数为

$$\mathcal{L}(\mathcal{D}; \mathbf{W}^{(1)}, \dots, \mathbf{W}^{(L)}, \mathbf{a}^{(1)}, \dots, \mathbf{a}^{(L)}) = \frac{1}{N} \sum_{n=1}^N \log p(\hat{\mathbf{v}}^{(n)}; \mathbf{W}^{(1)}, \dots, \mathbf{W}^{(L)}, \mathbf{a}^{(1)}, \dots, \mathbf{a}^{(L)}).$$

因此, 得到优化问题:

$$\max \quad \frac{1}{N} \sum_{n=1}^N \log p(\hat{\mathbf{v}}^{(n)}; \mathbf{W}^{(1)}, \dots, \mathbf{W}^{(L)}, \mathbf{a}^{(1)}, \dots, \mathbf{a}^{(L)}).$$

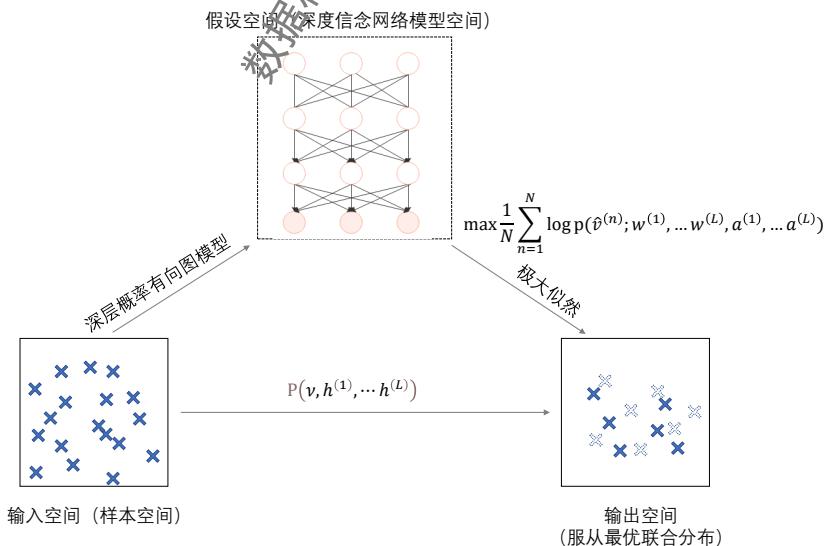


图 9.28: 从空间的角度理解深度信念网络模型

变分自编码器

定义 9.4.11. 变分自编码器 (*Variational AutoEncoder, VAE*) 是一种深度生成模型, 其思想是利用神经网络来分别建模两个复杂的条件概率密度函数。

变分自编码器其模型结构可以分为两个部分:

1、推断网络: 用神经网络来产生变分分布 $q(z|\phi)$, 也记为 $q(z|x;\phi)$ (用简单的分布 q 去近似复杂的分 $p(z|x;\theta)$)

2、生成网络: 用神经网络来产生概率分布, 估计更好的分布 $p(x|z;\theta)$

将推断网络和生成网络合并就得到了变分自编码器的整个网络结构:

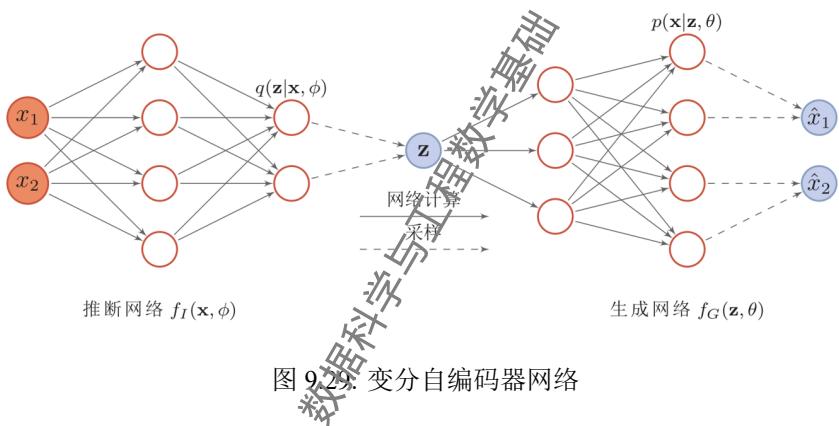


图 9.29. 变分自编码器网络

推断网络的目标是使得 $q(z|x:\phi)$ 能接近真实的后验 $p(z|x:\theta)$, 需要找到一组网络参数 ϕ^* 来最小化两个分布的 KL 散度, 即:

$$\phi^* = \arg \min_{\phi} \text{KL}(q(z|x;\phi), p(z|x;\theta)).$$

这实际上, 等价于

$$\arg \max_{\phi} \text{ELBO}(q, x; \theta, \phi),$$

其中

$$\text{ELBO}(q, x; \theta, \phi) = \mathbb{E}_{z \sim q(z;\phi)} \left[\log \frac{p(x, z; \theta)}{q(z;\phi)} \right]$$

为证据下界。

上述等价性是因为, 对数似然函数 $\log p(\mathbf{x}; \theta)$ 可以分解为:

$$\begin{aligned}\log p(\mathbf{x}; \theta) &= \sum_{\mathbf{z}} q(\mathbf{z}; \phi) \log p(\mathbf{x}; \theta) \\ &= \sum_{\mathbf{z}} q(\mathbf{z}; \phi) (\log p(\mathbf{x}, \mathbf{z}; \theta) - \log p(\mathbf{z} | \mathbf{x}; \theta)) \\ &= \sum_{\mathbf{z}} q(\mathbf{z}; \phi) \log \frac{p(\mathbf{x}, \mathbf{z}; \theta)}{q(\mathbf{z}; \phi)} - \sum_{\mathbf{z}} q(\mathbf{z}; \phi) \log \frac{p(\mathbf{z} | \mathbf{x}; \theta)}{q(\mathbf{z}; \phi)} \\ &= \text{ELBO}(q, \mathbf{x}; \theta, \phi) + \text{KL}(q(\mathbf{z}; \phi) || p(\mathbf{z} | \mathbf{x}; \theta)).\end{aligned}$$

因此, 推断网络的目标函数可以转换为

$$\begin{aligned}\phi^* &= \arg \min_{\phi} \text{KL}(q(\mathbf{z} | \mathbf{x}; \phi), p(\mathbf{z} | \mathbf{x}; \theta)) = \arg \min_{\phi} \log p(\mathbf{x}; \theta) - \text{ELBO}(q, \mathbf{x}; \theta, \phi) \\ &= \arg \max_{\phi} \text{ELBO}(q, \mathbf{x}; \theta, \phi),\end{aligned}$$

生成网络的目标:生成网络 $f_G(z; \theta)$ 的目标是找到一~~生~~网络参数 θ^* 来最大化证据下界 $\text{ELBO}(q, \mathbf{x}; \theta, \phi)$, 从而最大化对数似然, 即:

$$\theta^* = \arg \max_{\theta} \text{ELBO}(q, \mathbf{x}; \theta, \phi)$$

结合上述公式, 推断网络和生成网络的目标都为最大化证据下界 $\text{ELBO}(q, \mathbf{x}; \theta, \phi)$ 因此, 变分自编码器的优化问题:

$$\max_{\theta, \phi} \text{ELBO}(q, \mathbf{x}; \theta, \phi) = \max_{\theta, \phi} \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}; \phi)} \left[\log \frac{p(\mathbf{x} | \mathbf{z}; \theta) p(\mathbf{z}; \theta)}{q(\mathbf{z}; \phi)} \right]$$

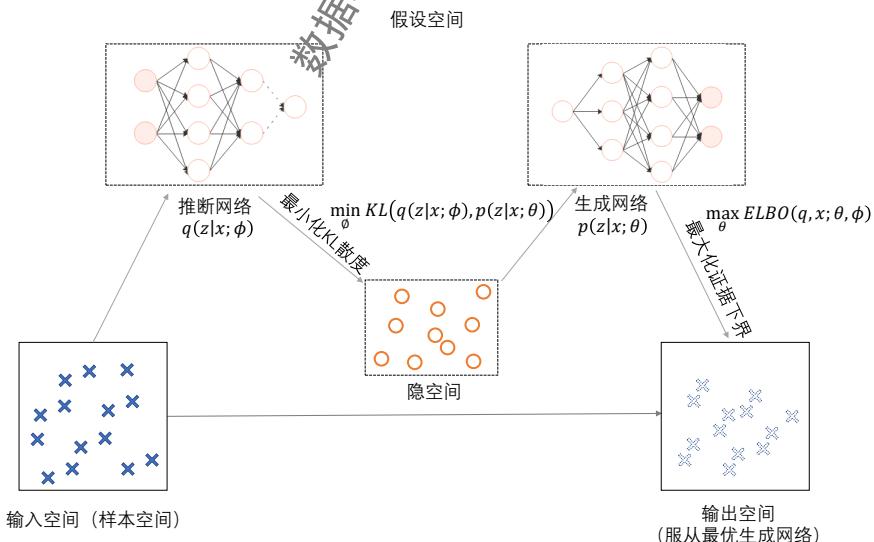


图 9.30: 从空间的角度理解变分自编码器模型

生成对抗网络

生成对抗网络 (Generative Adversarial Networks, GAN) 是通过对抗训练的方式来使得生成网络产生的样本服从真实数据分布。在生成对抗网络中, 有两个网络进行对抗训练。一个是判别网络, 目标是尽量准确地判断一个样本是来自于真实数据还是由生成网络产生; 另一个是生成网络, 目标是尽量生成判别网络无法区分来源的样本。

判别网络 (Discriminator Network) $D(x; \phi)$ 的目标是区分出一个样本 x 是来自于真实分布 $p_r(x)$ 是来自于生成模型 $p_\theta(x)$, 因此判别网络实际上是一个二分类的分类器。用标签 $y = 1$ 来表示样本来自真实分布, $y = 0$ 表示样本来自生成模型, 判别网络 $D(x; \phi)$ 的输出为 x 属于真实数据分布的概率。因此, 判别网络的目标函数可以建模为最小化交叉熵, 即

$$\min_{\phi} -(\mathbb{E}_x[y \log p(y = 1 | x) + (1 - y) \log p(y = 0 | x)])$$

生成网络 (Generator Network) $G(z; \theta)$ 的目标刚好和判别网络相反, 即让判别网络将自己生成的样本判别为真实样本。因此,

$$\max_{\theta} (\mathbb{E}_{z \sim p(z)}[\log D(G(z; \theta); \phi)]) = \min_{\theta} (\mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z; \theta); \phi))]).$$

上面的这两个目标函数是等价的。但是在实际训练时, 一般使用前者, 因为其梯度性质更好。

把判别网络和生成网络合并为一个整体, 将整个生成对抗网络的目标函数看作最小最大优化问题:

$$\begin{aligned} & \min_{\theta} \max_{\phi} (\mathbb{E}_{x \sim p_r(x)}[\log D(x; \phi)] + \mathbb{E}_{x \sim p_\theta(x)}[\log(1 - D(x; \phi))]) \\ &= \min_{\theta} \max_{\phi} (\mathbb{E}_{x \sim p_r(x)}[\log D(x; \phi)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z; \theta); \phi))]). \end{aligned}$$

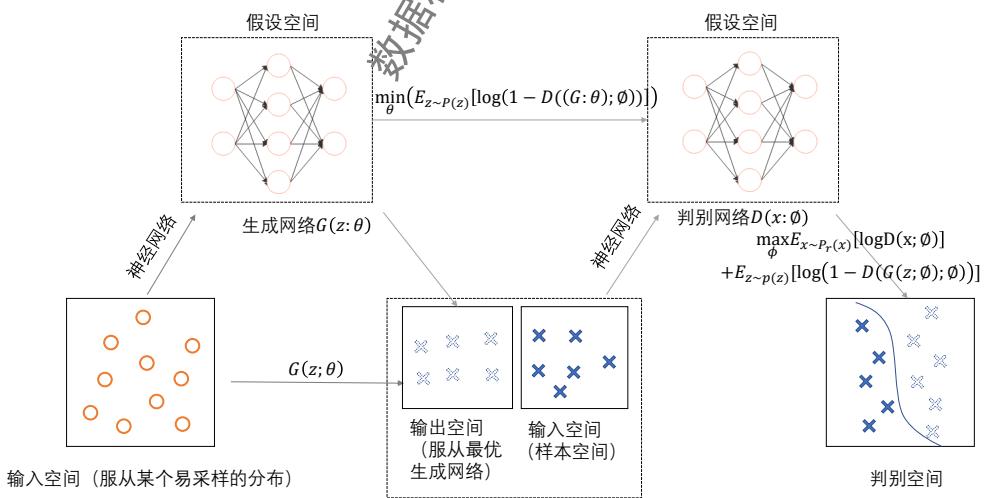


图 9.31: 从空间角理解生成对抗网络

自回归生成模型

许多数据是以序列的形式存在, 如声音、语言、视频、DNA 序列或其他的时序数据等。序列数据有两个特点: (1) 样本是变长的; (2) 样本空间非常大。

定义 9.4.12. 给定一个序列样本 $x_{1:T} = x_1, x_2, \dots, x_T$, 其概率为 $p(x_{1:T})$, 若在序列建模中, 每一步都需要将前面的输出作为当前步的输入, 也即是一种自回归的方式, 称这样的序列概率模型为自回归生成模型。

根据概率乘法公式, 序列 $x_{1:T}$ 的概率可以写为

$$p(x_{1:T}) = p(x_1)p(x_2|x_1)p(x_3|x_{1:2}) \cdots p(x_T|x_{1:(T-1)}) = \prod_{t=1}^T p(x_t|x_{1:(t-1)}) \quad (9.28)$$

其中 $x_t \in \mathbb{V}, t \in (1, \dots, T)$ 为词表 \mathbb{V} 中的一个词, $p(x_1|x_0) = p(x_1)$ 。序列数据的概率密度估计问题可变为单变量条件概率估计问题, 即给定 $x_{1:(t-1)}$ 时 x_t 的条件概率 $p(x_t|x_{1:(t-1)})$ 。

给定 N 个序列数据 $\{x_{1:T_n}^{(n)}\}_{n=1}^N$, 序列概率模型需要学习一个模型 $p_\theta(x|x_{1:(t-1)})$ 来最大化整个数据集的对数似然函数, 即为如下优化问题

$$\max_{\theta} \sum_{n=1}^N \log p_\theta(x_{1:T_n}^{(n)}) = \max_{\theta} \sum_{n=1}^N \sum_{t=1}^{T_n} \log p_\theta(x_t^{(n)}|x_{1:(t-1)}^{(n)}) \quad (9.29)$$

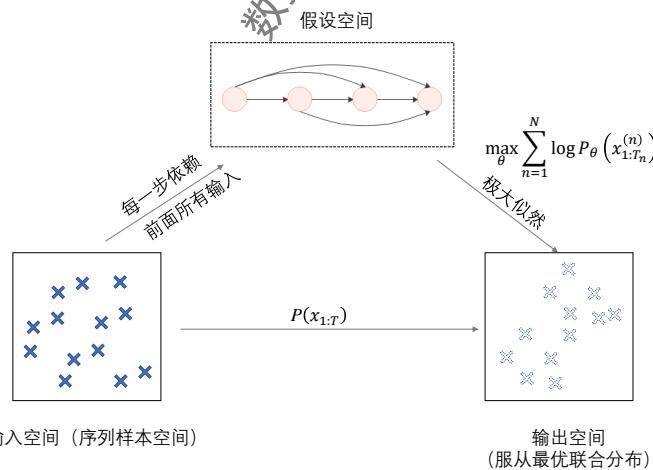


图 9.32: 从空间的角度理解自回归模型

9.4.4 强化学习中的概率模型

强化学习既不是监督学习，也不是无监督学习。一般会借助于“智能体”和“环境”两个概念对其进行表述，即智能体通过与环境的交互，根据获得的奖励信息进行学习。在交互的过程中，智能体通常能观察到环境的信息，这里简记为状态 s_t 。然后，它会根据当前的策略执行动作 a_t 。环境根据其内在的规律，达到一个新的状态 s_{t+1} ，并且给出奖励 r_{t+1} 。整个系统以这样的过程不断地持续进行，智能体也在不断地优化其执行策略。

与监督学习不同的是，在强化学习中，并非学习条件概率分布 $P(Y | X)$ 或者联合概率分布 $P(X, Y)$ 。因此，这并非大家所探讨的一般意义上的概率模型。若假定状态空间为 $\mathcal{S} = \{1, 2, \dots, S\}$ 和动作空间 $\mathcal{A} = \{1, 2, \dots, A\}$ 。现用 $\Delta_{\mathcal{A}}$ 表示在集合 \mathcal{A} 上的概率分布构成的集合，则强化学习的目标是学习出的概率模型（策略）为 $\pi: \mathcal{S} \rightarrow \Delta_{\mathcal{A}}$ 。

通常在强化学习里面称之为收益或回报，即累积的奖励。若假定智能体与环境交互的一条轨迹为：

$$\tau = \{s_0, a_0, r_1, s_1, a_1, r_2, s_2, \dots\}.$$

则累计奖励（带折扣因子 γ ）可表示为：

$$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t.$$

由于在同一策略下的轨迹并不是一成不变的，它是一个随机时序序列。因此，回报函数为一个期望值： $E_{\tau \sim \pi}[R(\tau)]$ 。

因此，强化学习即为求解如下优化问题：

$$\max_{\pi} E_{\tau \sim \pi}[R(\tau)].$$

9.5 阅读材料

概率学习方法利用（并且要求）关于不同假设的先验概率以及在给定假设时观察到不同数据的概率的知识。贝叶斯方法则提供了概率学习方法的基础。贝叶斯方法还可基于这些先验和数据观察假定，赋予每个候选假设一个后验概率。贝叶斯方法可用于确定在给定数据时最可能的假设—极大后验概率（MAP）假设。它比其他的假设更可能成为最优假设。

贝叶斯最优分类器将所有假设的预测结合起来，并用后验概率加权，以计算对新实例的最可能分类。

朴素贝叶斯分类器是在许多实际应用问题中很有效的一种贝叶斯学习方法。它之所以被称为朴素的（naive）是因为它的简化假定：属性值在给定实例的分类时条件独立。

当该假定成立时，朴素贝叶斯分类器可输出 MAP 分类。即使此假定不成立，在学习分类文本的情况下，朴素贝叶斯分类通常也是很有效的。贝叶斯网络为属性的子集上的一组条件独立性假定提供了更强的表达能力。

贝叶斯推理框架可对其他不直接应用贝叶斯公式的学习方法的分析提供理论基础。例如, 在特定条件下学习一个对应于极大似然假设的实值目标函数时, 它可使误差平方最小化。

最小描述长度准则建议选取这样的假设, 它使假设的描述长度和给定假设下数据的描述长度的和最小化。贝叶斯公式和信息论中的基本结论可提供此准则的根据。

在许多实际的学习问题中, 某些相关的实例变量是不可观察到的。EM 算法提供了一个很通用的方法, 当存在隐藏变量时进行学习。

该算法开始于一个任意的初始假设。然后迭代地计算隐藏变量的期望值 (假定当前假设是正确的), 再重新计算极大似然假设 (假定隐藏变量等于第 1 步中得到的期望值)。这一过程收敛到一个局部的极大似然假设以及隐藏变量的估计值。

在概率和统计方面有许多很好的介绍性文章, 如 Casella & Berger: (1990)。几本快速参考类书籍 (如 Maisel 1971, Speigel 1991) 也对机器学习相关的概率和统计理论提供了很好的阐述。

对贝叶斯分类器和最小平方误差分类器的基本介绍由 Duda & Hart (1973) 给出, Domigos

& Pazzani(1996) 分析了在什么条件下朴素贝叶斯方法可输出最优的分类, 即使其独立性假定不成立时 (关键在于在什么条件下即使相关联的后验概率估计不正确也可输出最优分类)。

Cmnik (1990) 讨论了使用 m-估计来估计概率。

将不同贝叶斯方法与决策树等其他算法进行比较的实验结果可在 Michie et al.(1994) 中找到。Chauvin & Rumelhart (1995) 提供了基于反向传播算法的神经网络的贝叶斯分析。

对最小描述长度准则的讨论可参考 Rissanen(1983, 1989)。Quinlan & Rivest(1989) 描述了其使用以避免决策树的过度拟合。

统计推断内容在很多书中都有涉及, 初等的参考书包括 (DeGroot and Schervish, 2000; Larsen and Marx, 1986), 中水平的参考书推荐读者参考 (Casella and Berger, 2002; Bickel and Doksum, 2000; Rice, 1995), 高级教程包括 (Cox and Hinkley, 2000; Lehmann and Casella, 1998; Lehmann, 1986; van der Vaart, 1998)。

Bootstrap 方法是 Efron(1979) 发明的。到目前为止, 已经有一些书是关于这个论题的, 包括 (Efron and Tibshirani, 1993; Davision and Hinkley, 1997; Hall, 1992; Shao and Tu, 1995)。同时, 见 3.6 节 (van der Vaart and Wellner, 1996)。

贝叶斯推断的参考书包括 (Carlin and Louis, 1996; Gelman et al., 1995; Lee, 1997; Robert, 1994; Schervish, 1995)。对于非参贝叶斯推断的技巧, 见 (Cox, 1993; Diaconis and FVeedman, 1999; Barron et al., 1999; Ghosal et al., 2000; Shen and Wasserman, 2001; Zhao, 2000)。Robins-Ritov 例子在 (Robins-Ritov, 1997) 中详细讨论, 那里它更确切地被作为非参问题讨论。例 11.10 来自 Edward George(个人通讯)。关于贝叶斯检验参考 (Berger and Delampady, 1987; Kass and Raftery, 1995)。对于无信息先验, 见 (Kass and Wasserman, 1996)。

决策理论的讨论可以见文献 (Casella and Berger, 2002; Berger, 1985; Ferguson, 1967; Lehmann and Casella, 1998)。

有关线性回归的著作见文献 (Weisberg, 1985)。从数据挖掘角度写的有关回归的书见文献

(Hastie et al., 2002).Akaike 信息准则 (AIC) 见 Akaike(1973) 的著作. 贝叶斯信息准则 (BIC) 见文献 (Schwarz, 1978).Logistic 回归的参考文献 (Agresti, 1990) 和 (Dobson, 2001).

有很多关于 DAGs 的文献包括 Edwsrds (1995) 和 Jordan(2004). 第一个用 DAGs 来表示因果关系的是 Wright(1934) . 一些现代的论述包含在文献 (Spirtes etal ,2000) 和 (Pearl,2000) 中.Robins 等 (2003) 讨论了从数据中来估计因果结构的问题.

最大似然估计和极大后验估计这两种方法都有很长的发展历史了。最初把贝叶斯方法引入模式识别领域是 David, 它指出当类条件概率密度函数未知的情况下, 正确地使用训练样本的途径是计算 $P(\omega_i|x, D)$ 。贝叶斯自己也非常看重无信息先验的作用。一个详尽的对不同的先验概率的研究请参见 Harold Jeffreys 和 Dennis Victor Lindley。Jose M. Bernardo 详细地列举了这方面的文献资料。Manfred Opper and David Haussler 描述了 Gibbs 算法, 而 David Haussler, Michael Kearns, and Robert Schapire.Bounds 则对此进行了深入的分析。

主成分分析是一种经典的多元统计分析方法, 在广泛的工程领域中都得到了重要应用。Geoffrey J.McLachlan 详细而深入地描述了最初由 Fisher 所提出的线性可分性方法, 文献 Herman Chernoff,Pierre A, Devijver ,Keinosuke Fukunaga. 也进行了这方面的论述。

期望最大化算法是由 Dempster 等人提出的。Geoffrey J.McLachlan 对这一方法和其发展历史进行了详细论述。Michael I.Jordan ,D.Michael Tiitcrington 描述了期望最大化算法的在线版本。而专门讨论在丢失数据情况下的处理方法, 则可以参考 Donald B.Rubin 的研究, 当然, 这方面的进一步深入论述这超出了本书的范围。

马尔可夫在分析俄国文学家普希金的名著《叶甫盖尼 · 奥涅金》的文字的过程中, 提出了后来被称为马尔可夫框架的思想。而 Baum 及其同事则提出了隐马尔可夫模型, 这一思想后来在语音识别领域 (awrence Rabiner) 得到了异常成功的应用。同时, 隐马尔可夫模型在 “统计语言学习” (Eugene Charniak, Frederick Jelinek) 以及 “序列符号识别” (比如 DNA 序列) (Pierre Baldi,Anders Krogh) 等领域也得到了应用。人们还把隐马尔可夫模型扩展到二维领域, 用于光学字符识别 (Gary E.Kopec)。而其中的解码算法则是由 Viterbi 和他的同事们 (G.David Forney,J Andrew J.Viterbi) 发展起来的。Padhraic Smyth 探讨了隐马尔可夫模型和图论模型 (比如贝叶斯置信网) 之间的联系。

Knuth 的经典著作是最初研究计算复杂度的著作, 他完成了这个领域的大部分工作。而该领域的标准教科书 (Thomas H.Cormen) 对于在计算机领域没有非常强的背景的读者是一本更好的入门性读物 (也为我们的几道课后习题提供了来源)。最后, Christopher M.Bishop,Padhraic Smyth 的著作都是模式识别方面的很好的教材, 虽然采用了与本书有所不同的方式, 但也都值得推荐。

习题

习题 9.1. 随机地取 8 只活塞环, 测得它们的直径为 (以 mm 计)

74.001	74.005	74.003	74.001
74,000	73.998	74.006	74.002

试求总体均值 μ 及方差 σ^2 的矩估计值, 并求样本方差 s^2 .

习题 9.2. 设某种电子器件的寿命 (以 h 计) T 服从双参数的指数分布, 其概率密度为

$$f(t) = \begin{cases} \frac{1}{\theta} e^{-(t-c)/\theta} & t \geq c \\ 0 & \text{其他} \end{cases}$$

其中 $c, \theta (c, \theta > 0)$ 为未知参数. 自一批这种器件中随机地取 n 件进行寿命试验. 设它们的失效时间依次为 $x_1 \leq x_2 \leq \cdots \leq x_n$.

(1) 求 θ 与 c 的最大似然估计值.

(2) 求 θ 与 c 的矩估计量

习题 9.3. 设 X_1, X_2, \dots, X_n 是来自概率密度

$$f(x; \theta) = \begin{cases} \theta x^{\theta-1} & 0 < x < 1 \\ 0 & \text{其他} \end{cases}$$

的总体的样本, θ 未知, 求 $U = e^{-1/\theta}$ 的最大似然估计值.

习题 9.4. 设 x_1, x_2, \dots, x_n 是来自总体 $b(m, \theta)$ 的样本值, 又 $\theta = \frac{1}{3}(1 + \beta)$, 求 β 的最大似然估计值.

习题 9.5. 设总体 X 的概率密度为

$$f(x; \theta) = \begin{cases} \frac{1}{\theta} x^{(1-\theta)/\theta} & 0 < x < 1 \\ 0 & \text{其他} \end{cases} \quad 0 < \theta < +\infty$$

X_1, X_2, \dots, X_n 是来自总体 X 的样本.

(1) 验证 θ 的最大似然估计量是 $\hat{\theta} = \frac{-1}{n} \sum_{i=1}^n \ln X_i$

(2) 证明 $\hat{\theta}$ 是 θ 的无偏估计量.

习题 9.6. (1) 设 $\hat{\theta}$ 是参数 θ 的无偏估计, 且有 $D(\hat{\theta}) > 0$, 试证

$\hat{\theta}^2 = (\hat{\theta})^2$ 不是 θ^2 的无偏估计.

(2) 试证明均匀分布

$$f(x) = \begin{cases} \frac{1}{\theta} & 0 < x \leq \theta \\ 0 & \text{其他} \end{cases}$$

中未知参数 θ 的最大似然估计量不是无偏的.

习题 9.7. 考虑高斯随机变量 $x \sim \mathcal{N}(x|\mu_x, \Sigma_x)$, 其中 $x \in \mathbb{R}^D$ 。进一步, 我们有

$$y = Ax + b + w$$

其中 $y \in \mathbb{R}^E$, $A \in \mathbb{R}^{E \times D}$, $b \in \mathbb{R}^E$, 并且 $w \sim \mathcal{N}(w|\mathbf{0}, Q)$ 是独立高斯噪声。

- (1) 写出似然函数 $p(y|x)$
- (2) 证明 $p(y) = \int p(y|x)p(x)dx$ 是高斯分布。计算 μ_y 和协方差 Σ_y
- (3) 对随机变量 y 做变换

$$z = Cy + v$$

写出 $p(z|y)$, 计算 $p(z)$, 即均值 μ_z 和协方差 Σ_z

- (4) 计算后验概率分布 $p(x|\hat{y})$

参考文献

- [1] AGRESTI,A.(1990).Categorical Data Analysis.Wiley.
- [2] AKAIKE, H.(1973).Information theory and an extension of the maximum likelihood principle.Second International Symposium on Information Theory 267-281.
- [3] BARRON,A., SCHERVISH, M.J.and WASSERMAN, L.(1999).The consistency of posterior distributions in nonparametric problems.The Annals of Statistics 27 536-561.
- [4] BERGER, J.O.(1985).Statistical Decision Theory and Bayesian Analysis (Second Edition).Springer-Verlag.
- [5] BERGER, J.O.and DELAMPADY, M.(1987).Testing precise hypotheses (P335-352).Statistical Science 2 317-335.
- [6] CARLIN, B.P.and LOUIS, T.A.(1996).Bayes and Empirical Bayes Methods or Data Analysis.Chapman Hall.
- [7] COX, D.D.(1993).An analysis of Bayesian inference for nonparametric regression.The Annals of Statistics 21 903-923.
- [8] DIAGONIS, P.and FREEDMAN, D.(1986).On inconsistent Bayes estimates of location.The Annals of Statistics 14 68-87.
- [9] EDWARDS, D.(1995).Introduction to graphical modelling.Springer-verlag.
- [10] FERGUSON, T.(1967).Mathematical Statistics: a Decision Theoretic Approach.Academic
- [11] GELMAN, A., CARLIN, J.B., STERN, H.S.and RUBIN, D.B,(1995).Bayesian Data Analysis, Chapman & Hall.
- [12] GHOSAL, SM GHOSH, J.K.and VAN DER VAART, A.W.(2000).Convergence rates of posterior distributions.The Annals of Statistics 28 500-531.
- [13] JORDAN, M.(2004).Graphical models.In Preparation.

- [14] KASS, R.E.and WASSERMAN, L.(1996).The selection of prior distributions by formal rules (corn 1998 v93 P 412).Journal of American Statistical Association 01 1343-1370.
- [15] LEE, P.M.(1997).Bayesian Statistics: An Introduction.Eldward Arnold.
- [16] LEHMANN, E.L.and CASELLA, G.(1998).Theory of Point Estimation.Springer-Verlag.
- [17] PEARL, J.(2000).Causalityi modela, reasoning, and inference.Cambridge University Press.
- [18] ROBINS, J.t SCHEINES, R., SPIRITES, P.and WASSERMAN, L.(2003).Uniform convergence in causal inference.Biometrika to appear.
- [19] LARSEN, R.J.and MARX, M.L.(1986).An Introduction to Mathematical Statistics and Its Applicationa(Second Edition).Prentice Hall.
- [20] DEGROOT, M.and Schervish, M.(2002).Probability and Statistics (Third Edition).AddisonWesley.
- [21] CASELLA, G.and BERGER, R.L.(2002).Statistical Inference.Duxbury Press.
- [22] BICKEL, P.J.and DOKSUM, K.A.(2000).Mathematical Statisticst Basic Ideas and Selected Topics, Vol.I(Second Edition).Prentice Hall.
- [23] RICE, J.A.(1995).Mathematical Statistics and Data Analysis (Second Edition).Duxbury Press.
- [24] COX, D.R.and HINKELEY, D.V.(2000).Theoretical statistics.Chapman& Hall.
- [25] LEHMANN, E.L.(1986).Testing Statistical Hypotheses Second Edition, Wiley.
- [26] VAN DER VAART, A.W.(1998).Asymptotic Statistics.Cambridge University Press.
- [27] EFRON, B.(1979).Bootstrap methods*Another look at the jackknife.The Annals of Statistics 71-26.
- [28] EFRON, B., TISBSHIRANI, R.J.(1993).An Introduction to the Bootstrap.Chapman & Hall.
- [29] DAVISON, A.C.and Hinkley, E.V.(1997).Bootstrap Methods and Their Application.Cambridge University Press.
- [30] HALL, P.(1992).The Bootstrap and Edgeworth Expansion, Springer-Verlag.
- [31] Russell G.Almond.Graphicat Belief Modelling Chapman & Hall.New York,1995.
- [32] Pierre Baldi,Sorcn Brunak,Yves Chauvin,Jacob Engelbrecht, and Aadcrs Krogh.Hidden Markov models for human genes.In Stephen J.Hanson, Jack D.Cowan, and C.Lee Giles, editors.Advances in Neural Information Processing Systems, volume 6, pages 761-768, Morgan Kaufmann, San Mateo, CA 1994.
- [33] Leonard E.Baum and Ted Petrie.Statistical inference for probabilistic functions of finite state Markov chains Annals of Mathematical Statistics, 37:1554—1563, 1966.
- [34] Leonard E.Baum, Ted Petrie, George Soules, and Norman A maximizaioit technique occurring in the statistical analysis of probabilistic functions of Markov chains.Annals of Mathematical Statistics, 41(1):164-171,1970.
- [35] Jose M. Bernardo and Adrian F.M.Smith.Bayesian Theory.Wiley, New York, 1996.

- [36] Christopher M. Bishop. Neural Networks for Pattern Recognition. Oxford University Press, Oxford, UK, 1995.
- [37] David Braverman. Learning filters for optimum pattern recognition. IRE Transactions on Information Theory, IT8:280-285, 1962.
- [38] Eugene Charniak. Statistical Language Learning. MIT Press, Cambridge, MA, 1993.
- [39] Herman Chernoff and Lincoln E. Moses. Elementary Decision Theory. Wiley, New York, 1959.
- [40] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. Introduction to Algorithms. MIT Press, Cambridge, MA, 1990.
- [41] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm (with discussion). Journal of the Royal Statistical Society. Series B 39:1-38, 1977.
- [42] Pierre A. Devijver and Josef Kittler. Pattern Recognition: A Statistical Approach. Prentice-Hall, London, 1982.
- [43] Ronald A. Fisher. The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7 Part II: 179-188, 1936.
- [44] G. David Forney, Jr. The Viterbi algorithm. Proceedings of the IEEE, 61:268-278, 1973.
- [45] Keinosuke Fukunaga. Introduction to Statistical Pattern Recognition. Academic Press, New York, second edition, 1990.
- [46] David Haussler, Michael Kearns, and Robert Schapire. Bounds on the sample complexity of Bayesian learning using information theory and the VC dimension. Machine Learning 14:84-114, 1994.
- [47] Harold Jeffreys. Theory of Probability. Oxford University Press. Oxford, UK, 1961 reprint edition, 1939.
- [48] Frederick Jelinek. Statistical Methods for Speech Recognition. MIT Press, Cambridge, MA, 1997.
- [49] Ian T. Jolliffe. Principal Component Analysis. Springer-Verlag, New York, 1986.
- [50] Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. Neural Computation, 6(2):181-214, 1994.
- [51] Donald E. Knuth. The Art of Computer Programming volume 1. Addison-Wesley, Reading, MA, first edition, 1973.
- [52] Gary E. Kopec and Phil A. Chou. Document image decoding using Markov source models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(6):602-617, 1994.
- [53] Anders Krogh, Michael Brown, I. Saira Mian, Kimmen Sjolander, and David Haussler. Hidden Markov models in computational biology: Applications to protein modelling. Journal of Molecular Biology, 235:1501-1531, 1994.

[54] Dennis Victor Lindley. The use of prior probability distributions in statistical inference and decision. In Jerzy Neyman and Elizabeth L. Scott, editors. *Proceedings Fourth Berkeley Symposium on Mathematical Statistics and Probability*, pages 453-468, University of California Press, Berkeley, CA, 1961.

数据科学与工程数学基础

第十章 优化基础

根据第1章提及的统计学习理论中的经验风险最小化准则，我们知道数据科学、人工智能和机器学习的很多问题都归结为一个优化问题。对优化问题的求解已然成为大部分数据分析和机器学习算法的核心组成部分。而且机器学习算法都是在计算机上操作的，其数学公式就表示为数值优化算法。因为来源于实际应用的优化问题是如此的多样和复杂，所以在我们介绍各种具体的数值优化算法之前，我们将安排两章内容，也即本章和下一章，来理清我们所面对的各种优化问题以及其可解的条件，然后在第12章，我们会详细介绍各种具体的数值优化求解算法。本章主要介绍优化的基础理论。我们在第5章已经看到，普通的最小二乘问题可以用标准线性代数工具求解。在这种情况下，最小化问题的解可以被有效找到并且是整体最优解，也即，除了最小二乘最优解外没有其他更优的解。这些令人满意的特性实际上可扩展到一类更广泛的优化问题，而实现优化求解的关键特性就是所谓的“凸性”性质。因此在本章中，将主要介绍：优化问题的定义、优化问题的分类、数据科学中常见的优化问题、凸集和凸函数的定义和判别方法以及保凸运算、凸优化问题的定义和标准形式，并介绍数据科学中常见的典型凸优化问题。

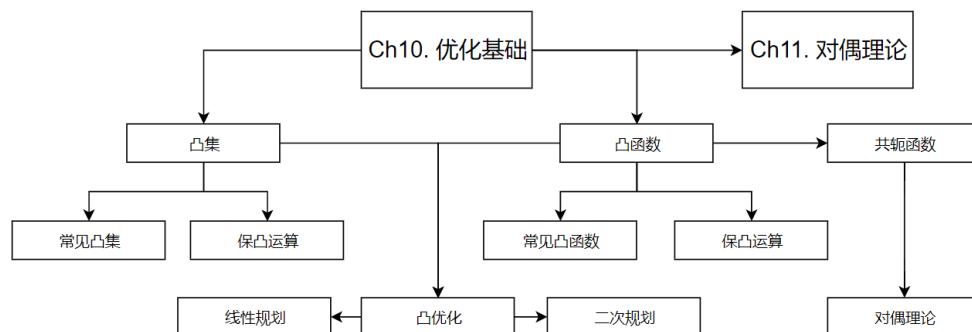


图 10.1: 本章导图

10.1 优化简介

在标准算法理论中，设计一个有效的算法来解决手头的问题是算法设计者的主要责任。自计算机科学引入的几十年以来，人们为各种任务设计了很多优美的算法，这些任务包括查找图中的最短路径、计算网络中的最佳流、压缩包含由数码相机拍摄的图像文件以及替换文本文档中的字符串等。

这些设计方法虽然对许多任务都很有用，但并没有解决更复杂的问题，例如在位图格式的图像中识别特定的人，或者将文本从英语翻译成中文。对于上述任务，可能有一个很好的算法，但是算法设计方案可能是不容易扩展的。

正如图灵在他的论文中所提倡的那样，我们要教计算机学习如何解决一个任务，而不是教给它特定任务的解决方案。实际上，这就是我们在学校中所做的，教会大家如何学习。我们希望教会计算机如何学习。这就是人工智能的思想，其核心是机器学习，并且主要就是从数据中来进行学习。比如，我们考虑一个图像数据，将图像分为两类的问题：包含汽车的图像和包含椅子的图像（假设世界上只有两种类型的图像）。在机器学习中，我们训练（教导）一台机器以实现所需的功能，同一台机器可以潜在地解决任何算法任务，并且不同于一个任务到另一个任务只能由一组参数来决定机器的功能。

机器学习中通常将机器训练过程看作一个优化问题。如果我们把 $\theta \in \mathbb{R}^d$ 作为机器的参数（也即模型，确定了参数就确定了模型），它被限制在某个集合 $\mathcal{K} \subseteq \mathbb{R}^d$ 中，如果函数 f 成功地度量了将实例映射到它们的标签与正确标签间的某种损失，那么这个训练过程可以用如下数学优化问题来描述：

$$\min_{\theta \in \mathcal{K}} f(\theta) \quad (10.1)$$

这是本书优化部分关注的主要问题，并且将特别强调机器学习中出现的具有特殊结构的函数，以便设计有效的算法。

事实上，根据度量的准则不同和参数模型的不一样，机器学习中会有很多各种具有特殊结构的优化问题。例如，我们在第 1 章中提到，在确定了训练集 \mathcal{D} 、假设空间 \mathcal{F} 以及学习准则后，如何找到最优的模型 $f(\mathbf{x}, \theta)$ 就成了一个最优化（Optimization）问题。其中 \mathbf{x} 是输入实例，它对应的输出记为 y ，它们一起形成训练数据集

$$T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}.$$

根据模型是否含有概率以及学习准则的不同，我们有如下四大类优化问题有待求解。

首先是经验风险最小化问题，求最优模型就是求解最优化问题：

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N L(y_i, f(\mathbf{x}_i)), \quad (10.2)$$

其中， \mathcal{F} 是假设空间， L 是损失函数，如平方损失函数等。

有时, 为了避免过拟合, 需引入结构风险最小化。结构风险最小化的策略认为结构风险最小的模型是最优的模型, 也就是要求解最优化问题:

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N L(y_i, f(\mathbf{x}_i)) + \lambda J(f), \quad (10.3)$$

其中 $J(f)$ 为模型的复杂度, 是定义在假设空间 \mathcal{F} 上的泛函。

上面两类优化主要针对函数类模型, 当我们使用概率分布来为实际问题建模, 我们会求解最大似然估计, 也即最小化如下负对数似然问题:

$$\min_{\theta} \mathcal{L}(\theta), \quad (10.4)$$

其中 $\mathcal{L}(\theta) = -\log p(\mathbf{y}|\mathbf{X}, \theta) = -\sum_{n=1}^N \log p(y_n|\mathbf{x}_n, \theta)$ 。

如果我们有关于参数 θ 的分布的先验知识, 则我们会求解一个最大后验估计, 也即最小化如下负对数后验问题:

$$\min_{\theta} -\log p(\theta|\mathbf{x}) \quad (10.5)$$

其中 $p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})} \propto p(\mathbf{x}|\theta)p(\theta)$ 。

机器学习的训练过程其实就是最优化问题的求解过程。在机器学习中, 优化又可以分为参数优化和超参数优化。模型 $f(\mathbf{x}; \theta)$ 中的 θ 称为模型的参数, 可以通过优化算法进行学习, 上面提及的四类优化问题都属于参数优化问题。除了可学习的参数 θ 之外, 还有一类参数是用来定义模型结构或优化策略的, 这类参数叫做超参数 (Hyper-Parameter)。在贝叶斯方法中, 超参数可以理解为参数的参数, 即控制模型参数分布的参数。常见的超参数包括: 聚类算法中的类别个数、梯度下降法的步长、正则项的系数、神经网络的层数、支持向量机中的核函数等。超参数的选取一般都是组合优化问题, 很难通过优化算法来自动学习。因此, 超参数优化是机器学习中一个经验性很强的技术, 通常是按照人的经验设定, 或者通过搜索的方法对一组超参数组合进行不断试错调整。

本书我们主要以参数优化为主。下面我们介绍机器学习中一些常见的优化问题。

10.1.1 数据科学与机器学习中最优化问题的例子

线性分类与垃圾邮件处理

我们从第 1 章已经知道, 监督学习中最基本的优化问题之一是用模型拟合数据或样本, 也称为基于经验风险最小化的优化问题。线性分类的监督学习范式就是这样的一个例子。在这个模型中, 学习者面对的是一些已标记的积极和消极的样本。每个样本用向量 \mathbf{a}_i 表示其在欧几里德空间中对应的 d 维特征向量。例如, 垃圾邮件分类问题中电子邮件的常见表示是欧几里德空间中的二进制向量, 其中空间的维数是语料中的单词数。第 i 封电子邮件是一个向量 \mathbf{a}_i , 其中邮件中出现过的单词在向量 \mathbf{a}_i 中对应的位置为 1, 否则为 0。此外, 每个样本都有一个标签 $b_i \in \{-1, +1\}$, 对应于电子邮件是否被标记为垃圾邮件/非垃圾邮件。

我们的目标是找到一个超平面来分离两类向量：带正标签的向量和带负标签的向量。如果不存在这样一个根据标签完全分离训练集的超平面，则目标是找到一个以最小错误数实现训练集分离的超平面。

从数学上讲，给定一组 m 个样本来训练，我们寻找 $\mathbf{x} \in \mathbb{R}^d$ ，它最小化了错误分类的样本的数量，即

$$\min_{\mathbf{x} \in \mathbb{R}^d} \frac{1}{m} \sum_{i \in [m]} \delta(\text{sign}(\mathbf{x}^T \mathbf{a}_i) \neq b_i)$$

其中， $\text{sign}(x) \in \{-1, +1\}$ 是符号函数，而 $\delta(z) \in \{0, 1\}$ 是指示函数，如果条件 z 满足，则取值 1，否则为 0。

上述线性分类的数学公式是数学优化问题(10.1)的特例，其中

$$f(\mathbf{x}) = \frac{1}{m} \sum_{i \in [m]} \delta(\text{sign}(\mathbf{x}^T \mathbf{a}_i) \neq b_i) = \mathbf{E}_{i \sim [m]} [l_i(\mathbf{x})]$$

上式中为了简单，我们使用了期望算子，其中 $l_i(\mathbf{x}) = \delta(\text{sign}(\mathbf{x}^T \mathbf{a}_i) \neq b_i)$ 。由于上面的优化问题是非凸的、非光滑的，所以通常采用凸松弛并用凸损失函数代替 $l_i(\mathbf{x})$ 。典型的选择包括均方误差函数和铰链损失。特别铰链损失函数

$$l_{\mathbf{a}_i, b_i}(\mathbf{x}) = \max\{0, 1 - b_i \cdot \mathbf{x}^T \mathbf{a}_i\}$$

在二分类的背景下，导致了著名的软间隔支持向量机问题。

另一个重要的优化问题是训练用于二分类的深层神经网络。例如，考虑一个有两个类别标记的图像数据集，这里用 $\{\mathbf{a}_i \in \mathbb{R}^d | i \in [n]\}$ 表示它，即含有 d 个像素的 m 个图像。我们想找到一个从图像到汽车和椅子这两类 $\{b_i \in \{0, 1\}\}$ 的映射 $f_{\mathbf{w}}(\mathbf{a}_i)$ 。该映射由机器学习模型的一组参数 \mathbf{w} 确定，比如神经网络中的权重。因此，我们试图找出把 \mathbf{a}_i 匹配到 b_i 的最佳参数，也即求解如下数学优化问题

$$\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) = \mathbf{E}_{\mathbf{a}_i, b_i} [l(f_{\mathbf{w}}(\mathbf{a}_i), b_i)].$$

矩阵补全和推荐系统

随着互联网的出现和在线媒体商店的兴起，媒体推荐已经发生了重大变化。收集到的大量数据能够有效地聚类和准确预测用户对各种媒体的偏好。一个众所周知的例子是所谓的“Netflix 挑战”——一个从用户的电影偏好的大数据集中进行推荐的自动化工具的竞赛。正如 Netflix 挑战中所证明的，自动化推荐系统最成功的方法之一是矩阵补全，它的最简单问题形式可以描述如下。

我们把整个用户-媒体偏好数据集看成是一个部分观测矩阵。矩阵中的每一行表示每个人，每一列表示一个媒体项（一部电影）。为了简单起见，让我们把观察结果看作是二元的，也即一个人喜欢或不喜欢某部电影。因此，我们有一个矩阵 $\mathbf{M} \in \{0, 1, *\}^{n \times m}$ ，其中 n 是考虑的总人

数, m 是的电影数目, 0、1 和 * 分别表示“不喜欢”、“喜欢”和“未知”:

$$M_{i,j} = \begin{cases} 0, & \text{第 } i \text{ 个人不喜欢第 } j \text{ 个电影} \\ 1, & \text{第 } i \text{ 个人喜欢第 } j \text{ 个电影} \\ *, & \text{偏好未知} \end{cases}$$

因为有很多用户和很多电影, 这个矩阵通常非常大, 只有部分位置有数值。一个自然的目标是补全矩阵, 即正确地将 0 或 1 分配给未知项。到目前为止, 这个问题是不适当的, 因为任何补全都是一样好 (或坏) 的, 而且对补全没有任何限制。

对补全的常见限制是“真”矩阵具有低秩。回想一下, 如果矩阵 $\mathbf{X} \in \mathbb{R}^{n \times m}$ 的秩 $k \ll p = \min\{n, m\}$, 那么它可以写成

$$\mathbf{X} = \mathbf{U}\mathbf{V}, \quad \mathbf{U} \in \mathbb{R}^{n \times k}, \mathbf{V} \in \mathbb{R}^{k \times m}$$

这个性质的直观解释是 \mathbf{M} 中的每个条目只能用 k 个数字来解释。在矩阵补全中, 这意味着, 直觉上, 只有 k 个因素决定一个人对电影的偏好, 比如类型、导演、演员等等。

在这样的约束下, 简单的矩阵补全 (推荐系统) 问题可以很好地表述为下述的数学优化。用 $\|\cdot\|_{OB}$ 表示仅在 \mathbf{M} 的观测 (非星号) 项上的欧几里得范数, 则矩阵补全的数学优化问题可以描述如下:

$$\begin{aligned} \min_{\mathbf{X} \in \mathbb{R}^{n \times m}} &= \frac{1}{2} \|\mathbf{X} - \mathbf{M}\|_{OB}^2 \\ \text{s.t.} & \quad \text{rank}(\mathbf{X}) \leq k \end{aligned}$$

10.1.2 其他常见的优化问题举例

最小二乘问题相关优化问题

- 最小二乘问题

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \tag{10.6}$$

- 加权最小二乘

$$\min_{\mathbf{x}} \|\mathbf{A}_w \mathbf{x} - \mathbf{y}_w\|_2^2$$

- 约束最小二乘

$$\begin{aligned} \min_{\mathbf{x}} & \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \\ \text{s.t.} & \quad \mathbf{B}\mathbf{x} = \mathbf{f} \end{aligned}$$

- 总体最小二乘

$$\begin{aligned} \min_{\Delta\mathbf{A}, \Delta\mathbf{b}, \mathbf{x}} & \|\Delta\mathbf{A}\|_F^2 + \|\Delta\mathbf{b}\|_2^2 \\ \text{s.t.} & \quad (\mathbf{A} + \Delta\mathbf{A})\mathbf{x} = \mathbf{b} + \Delta\mathbf{b} \end{aligned}$$

自然语言处理下的优化问题

- 词向量模型:

$$\max_{\mathbf{w}, b} \prod_{i=1}^m (h_{\mathbf{w}, b}(\mathbf{x}_i)^{y_i} * (1 - h_{\mathbf{w}, b}(\mathbf{x}_i)^{1-y_i}))$$

或

$$\min_{\mathbf{w}, b} - \sum_{i=1}^m (y_i \log h_{\mathbf{w}, b}(\mathbf{x}_i)) + (1 - y_i) \log(1 - h_{\mathbf{w}, b}(\mathbf{x}_i))$$

- 连续词袋模型

$$\min_{\mathbf{u}, \mathbf{v}} -\mathbf{u}_c^T \hat{\mathbf{v}} + \log \sum_{j=1}^{|V|} \exp(\mathbf{u}_j^T \hat{\mathbf{v}})$$

- 跳格模型

$$\min_{\mathbf{u}, \mathbf{v}} - \sum_{j=0, j \neq m}^{2m} \mathbf{u}_{c-m+j}^T \mathbf{v}_c + 2m \log \sum_{k=1}^{|V|} \exp(\mathbf{u}_k^T \mathbf{v}_c)$$

推荐系统中的优化问题

- 推荐系统的优化问题可以转为如下低秩矩阵恢复的优化问题:

$$\begin{aligned} \min_{\mathbf{X}} \quad & \text{rank}(\mathbf{X}) \\ \text{s.t.} \quad & \mathbf{X}_{ij} = \mathbf{M}_{ij} \quad \forall i, j \in \mathbb{E} \end{aligned}$$

或者转化为限定在秩为 r 的条件下, 低秩矩阵使得观测到的评分与预测的评分最接近:

$$\begin{aligned} \min_{\mathbf{X}} \quad & \sum_{ij} (X_{ij} - M_{ij})^2 \quad \forall i, j \in \mathbb{E} \\ \text{s.t.} \quad & \text{rank}(\mathbf{X}) = r \end{aligned}$$

低秩矩阵相关优化问题

- 鲁棒 PCA

$$\min_{\mathbf{A}, \mathbf{E}} \|\mathbf{A}\|_* + \lambda \|\mathbf{E}\|_1 \quad \text{s.t. } \mathbf{X} = \mathbf{A} + \mathbf{E}$$

- 低秩矩阵补全

$$\min_{\mathbf{A}} \|\mathbf{A}\|_* \quad \text{s.t. } P_{\Omega}(\mathbf{A}) = P_{\Omega}(\mathbf{D})$$

- 低秩矩阵表示

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_* \quad \text{s.t. } \mathbf{D} = \mathbf{BZ}$$

以及

$$\min_{\mathbf{Z}, \mathbf{E}} \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1} \quad \text{s.t. } \mathbf{D} = \mathbf{DZ} + \mathbf{E}$$

目标分类或预测中的优化问题

- 线性回归模型

$$\min_{\mathbf{w}, b} \sum_{i=1}^m (\mathbf{w}^T \mathbf{x}_i + b - y_i)^2$$

- 逻辑回归:

$$\min_{\mathbf{w}} \sum_{i=1}^N [y_i(\mathbf{w}^T \mathbf{x}_i) - \log(1 + \exp(\mathbf{w}^T \mathbf{x}_i))]$$

- 感知机

$$\min_{\mathbf{w}, b} - \sum_{\mathbf{x}_i \in M} y_i(\mathbf{w}^T \mathbf{x}_i + b)$$

- 支持向量机

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N L_{0/1}(y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1),$$

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \max(0, 1 - y_i(\mathbf{w}^T \mathbf{x}_i + b))$$

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i$$

$$s.t. \quad y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, N$$

$$\xi_i \geq 0, \quad i = 1, 2, \dots, N$$

- 非线性支持向量机

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^N \alpha_i$$

$$s.t. \quad \sum_{i=1}^N \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N$$

无监督学习的相关模型

- PCA

$$\min_{\mathbf{W}} \text{Tr}(-\mathbf{W}^T \mathbf{X} \mathbf{X}^T \mathbf{W})$$

$$s.t. \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}$$

- k 均值聚类

$$\min_C \sum_{l=1}^k \sum_{C(i)=l} \|\mathbf{x}_i - \bar{\mathbf{x}}_l\|^2$$

- 谱聚类

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{x}^T \mathbf{L} \mathbf{x} \\ & \text{s.t. } \mathbf{x}^T \mathbf{1} = 0 \end{aligned}$$

概率模型

- 最大熵模型

$$\begin{aligned} \min_{P \in C} -H(P) &= \sum_{x,y} \tilde{P}(x)P(y|x) \log P(y|x) \\ \text{s.t. } E_p(f_i) - E_{\tilde{P}}(f_i) &= 0, i = 1, 2, \dots, n \\ \sum_y P(y|x) &= 1 \end{aligned}$$

- 线性链条件随机场

$$\max_{\lambda, \mu} \sum_{j=1}^N \left(\sum_{i,k} \lambda_k t_k(y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l(y_i, x, i) \right) - \sum_{j=1}^N \log Z_{\lambda, \mu}(x_j).$$

- 深度信念网络

$$\max \quad \frac{1}{N} \sum_{n=1}^N \log p\left(\hat{\mathbf{v}}^{(n)}; \mathbf{W}^{(1)}, \dots, \mathbf{W}^{(L)}, \mathbf{a}^{(1)}, \dots, \mathbf{a}^{(L)}\right).$$

- 变分自编码器

$$\max_{\theta, \phi} \text{ELBO}(q, \mathbf{x}; \theta, \phi) = \max_{\theta, \phi} \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}; \phi)} \left[\log \frac{p(\mathbf{x} | \mathbf{z}; \theta) p(\mathbf{z}; \theta)}{q(\mathbf{z}; \phi)} \right]$$

神经网络模型

- 多层感知机模型（全连接神经网络）：

$$\min_{A_i, b_i, i=1, \dots, L} \|\sigma(A_L(\cdots(\sigma(A_1 \mathbf{x} + \mathbf{b}_1)) \cdots) + \mathbf{b}_L) - \mathbf{y}\|,$$

其中 \mathbf{x}, \mathbf{y} 分别表示输入特征和对应的标签， L 代表神经网络的层数， A_i, b_i 分别表示连接参数和偏置项， $\sigma(\cdot)$ 代表激活函数，例如：sigmoid 函数、tanh 函数、ReLU 函数等。

- 卷积神经网络：

$$\min_{K_i, B_i, i=1, \dots, L} \|\sigma(K_L * (\cdots(\sigma(K_1 * X + B_1)) \cdots) + B_L) - y\|,$$

其中 K_i, B_i 分别表示卷积核和对应的偏置项，这里我们用 $*$ 表示卷积运算，其他变量及符号同上。注意，在数学上，卷积神经网络可以看作是全连接网络的连接剪枝和参数共享的网络。另外，在实际计算中，可能还需要添加一些池化层或全连接层，这里为了便于描述进行了简化。

10.1.3 优化问题的一般形式

下面我们给出数学优化问题的一般形式以及相关的概念。

一般形式

最优化问题或者说优化问题的一般形式表示为：

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p \end{aligned} \tag{10.7}$$

其中，向量 $\mathbf{x} = (x_1, \dots, x_n)^T$ 称为问题的优化变量，函数 $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ 称为目标函数，在机器学习中常为损失函数。函数 $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ ，被称为不等式约束函数， $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$ 称为不等式约束，函数 $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ ，被称为等式约束函数， $h_j(\mathbf{x}) = 0, j = 1, \dots, p$ 称为等式约束。

这一标准形式总是可以得到的。按照惯例，不等式和等式约束的右端非零时，可以通过对任何非零右端进行移项得到。类似地，我们将 $f_i(\mathbf{x}) \geq 0$ 表示为 $-f_i(\mathbf{x}) \leq 0$ 。另外，对于如下极大化问题

$$\begin{aligned} & \max f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p \end{aligned} \tag{10.8}$$

可以通过在同样的约束下极小化 $-f_0$ 得到求解。

优化问题的形式转换

优化问题(10.7)的形式是非常灵活的，并允许许多变换，这就有利于我们将一个给定的问题转变为一个易于处理的问题。例如，优化问题

$$\min_{\mathbf{x}} \sqrt{(x_1 + 1)^2 + (x_2 - 2)^2} \quad \text{s.t. } x_1 \geq 0$$

与

$$\min_{\mathbf{x}} (x_1 + 1)^2 + (x_2 - 2)^2 \quad \text{s.t. } x_1 \geq 0$$

是等价的，而第二个优化问题的目标函数是可微的。有些情况下，还可以使用变量替换。例如，给定一个优化问题

$$\max_{\mathbf{x}} x_1 x_2^3 x_3 \quad \text{s.t. } x_i \geq 0, i = 1, 2, 3, x_1 x_2 \leq 2, x_2^3 x_3 \leq 1$$

令新变量 $z_i = \log x_i, i = 1, 2, 3$ ，在对目标函数取对数之后，该问题可以等价地写为

$$\max_{\mathbf{z}} z_1 + 3z_2 + z_3 \quad z_1 + z_2 \leq \log 2, 2z_2 + z_3 \leq 0.$$

优点是替换后的目标函数和约束函数都是线性的。

最优解的相关概念

定义 10.1.1. 目标函数和约束函数所有有定义点的集合:

$$\mathcal{D} = \bigcap_{i=0}^m \mathbf{dom} f_i \cap \bigcap_{j=1}^p \mathbf{dom} h_j$$

称满足所有约束条件的向量 $\mathbf{x} \in \mathcal{D}$ 为可行解或可行点, 全体可行点的集合称为可行集, 记为 \mathcal{F} , 其表示为:

$$\mathcal{F} = \{\mathbf{x} \in \mathcal{D} \mid f_i(\mathbf{x}) \leq 0, i = 1, \dots, m, h_j(\mathbf{x}) = 0, j = 1, \dots, p\}$$

若 $f_i(\mathbf{x})$ 和 $h_j(\mathbf{x})$ 是连续函数, 则 \mathcal{F} 是闭集。在可行集中找一点 \mathbf{x}^* , 使目标函数 $f_0(\mathbf{x})$ 在该点取最小值, 则称 \mathbf{x}^* 为问题的最优点或最优解, $f_0(\mathbf{x}^*)$ 称为最优值, 记为 p^* :

$$p^* = \inf\{f_0(\mathbf{x}) \mid f_i(\mathbf{x}) \leq 0, i = 1, \dots, m, h_j(\mathbf{x}) = 0, j = 1, \dots, p\}$$

- $p^* = \infty$, 如果问题不可行 (没有 \mathbf{x} 满足约束)
- $p^* = -\infty$, 问题无下界

优化问题(10.7)可以看成在向量空间 \mathbb{R}^n 的备选解集中选择最好的解。用 \mathbf{x} 表示备选解, $f_i(\mathbf{x}) \leq b_i$ 和 $h_j(\mathbf{x}) = 0$ 表示 \mathbf{x} 必须满足的条件, 目标函数 $f_0(\mathbf{x})$ 表示选择 \mathbf{x} 的成本 (同理也可以认为 $-f_0(\mathbf{x})$ 表示选择 \mathbf{x} 的效益或者效用)。优化问题(10.7)的解即为满足约束条件的所有备选解中成本最小 (或者效用最大) 的解。

在数据拟合中, 人们需要在一簇候选模型中选择最符合观测数据与先验知识的模型。此时, 变量为模型中的参数, 约束可以是先验知识以及参数限制 (比如说非负性)。目标函数可能是与真实模型的偏差或者是观测数据与估计模型的预测值之间的偏差, 也有可能是参数值的似然度和置信度的统计估计。此时, 寻找优化问题(10.7)的最优解即为寻找合适的模型参数值, 使之符合先验知识, 且与真实模型之间的偏差或者预测值与观测值之间的偏差最小 (或者在统计意义上更加相似)。

在最优解的相关概念中, 常被人们所探讨的是全局最优解和局部最优解, 如下定义所述。

定义 10.1.2. 整体 (全局) 最优解: 若 $\mathbf{x}^* \in \mathcal{F}$, 对于一切 $\mathbf{x} \in \mathcal{F}$, 恒有 $f_0(\mathbf{x}^*) \leq f_0(\mathbf{x})$, 则称 \mathbf{x}^* 是最优化问题(10.7)的整体最优解。

定义 10.1.3. 局部最优解: 若 $\mathbf{x}^* \in \mathcal{F}$, 存在某个领域 $N_\varepsilon(\mathbf{x}^*)$, 使得对于一切 $\mathbf{x} \in N_\varepsilon(\mathbf{x}^*) \cap \mathcal{F}$, 恒有 $f_0(\mathbf{x}^*) \leq f_0(\mathbf{x})$, 则称 \mathbf{x}^* 是最优化问题(10.7)的局部最优解。其中 $N_\varepsilon(\mathbf{x}^*) = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}^*\| < \varepsilon, \varepsilon > 0\}$

当 $\mathbf{x} \neq \mathbf{x}^*$, 有 $f_0(\mathbf{x}^*) < f_0(\mathbf{x})$ 则称 \mathbf{x}^* 为优化问题(10.7)的严格最优解。反之, 若一个点是局部最优解, 但不是严格最优解, 则称之为非严格最优解。

例 10.1.1. 下面给出一些无约束优化问题的最优值以及最优解或局部最优解示例。

- $f_0(x) = \frac{1}{x}$, $\text{dom } f_0 = R^{++} : p^* = 0$, 无最优解
- $f_0(x) = -\log x$, $\text{dom } f_0 = R^{++} : p^* = -\infty$, 无下界
- $f_0(x) = x \log x$, $\text{dom } f_0 = R^{++} : p^* = -\frac{1}{e}, x = \frac{1}{e}$ 是最优解
- $f_0(x) = x^3 - 3x : p^* = -\infty, x = 1$ 是局部最优解

由上述定义可知, 局部最优解 x^* 使 f_0 最小, 但仅对可行集上的邻近点。此时目标函数的值不一定是问题的(全局)最优值。局部最优解可能对用户没有实际意义。因此, 在实际的优化问题中局部最优解的存在是一个挑战, 因为大多数算法往往被困在局部极小, 如果存在的话, 从而不能产生期望的全局最优解。

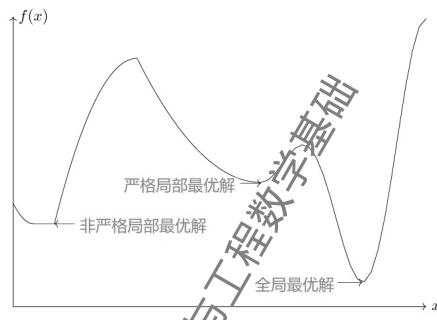


图 10.2: 函数的全局与局部最优解、严格和非严格最优解

优化算法

根据优化问题的不同形式, 其求解的困难程度可能会有很大差别。对于一个比较简单优化问题, 如果我们能用代数表达式给出其最优解, 那么这个解称为显式解。然而, 对于实际问题往往较为复杂, 是没有办法求显式求解的, 因此常采用迭代算法。主要思想是寻找一个点列, 通过迭代, 不断地逼近精确解。我们将在第 12 章详细讨论。

10.1.4 优化问题的分类

优化问题种类繁多, 因而分类的方法也有许多。可以按变量的性质分类, 按有无约束条件分类, 按目标函数的个数分类等等。概括地, 按照变量的性质分类, 可以分为连续和离散优化问题。按照约束函数是否存在分类, 可以分为无约束和约束优化问题。按照目标函数的性质分类, 可以分为凸和非凸优化问题。按照目标函数和约束函数的形式分类, 可以分为随机和确定性优化问题, 也可以分为线性和非线性规划问题。

连续优化与离散优化

根据输入变量 \mathbf{x} 的值域是否连续, 数学优化问题可以分为离散优化问题和连续优化问题。离散优化 (Discrete Optimization) 问题是指决策变量能够在离散集合上取值, 比如离散点集或整数集等。离散优化问题主要有三个分支:

1. 整数规划 (Integer Programming): 输入变量 $\mathbf{x} \in \mathbb{Z}^d$ 为整数向量。常见的整数规划问题通常为整数线性规划 (Integer Linear Programming, ILP)。
2. 混合整数规划 (Mixed Integer Programming, MIP), 即自变量既包含整数也有连续变量。
3. 组合优化 (Combinatorial Optimization): 其目标是从一个有限集合中找出使得目标函数最优的元素。很多机器学习问题都是组合优化问题, 比如特征选择、聚类问题、超参数优化问题以及结构化学习 (Structured Learning) 中标签预测问题等。

从这个意义上讲, 组合优化是整数规划的子集。的确, 绝大多数组合优化问题都可以被建模成 (混合) 整数规划模型来求解。离散优化问题的求解一般都比较困难, 优化算法的复杂度都比较高。连续优化 (Continuous Optimization) 问题是指决策变量所在的可行集合是连续的。

- 在连续优化问题中, 基于决策变量取值空间以及约束和目标函数的连续性, 可根据某点领域内的取值信息来判断该点是否最优。
- 离散优化问题不具备该性质。因此通常将离散优化问题转化为一系列连续优化问题来求解。
- 连续优化问题的求解在最优化理论与算法中处于重要地位。一般认为, 在深度学习或机器学习中, 模型中要学习的参数是连续变量。因此本书后续内容也将主要围绕讲解连续优化问题展开。

无约束优化和约束优化

根据是否有变量的约束条件, 可以将优化问题分为无约束优化问题和约束优化问题。

1. 无约束优化问题 (Unconstrained Optimization) 的决策变量没有约束条件限制, 即可行域为整个实数域 $D = \mathbb{R}^d$ 。在优化问题(10.7)中, 当我们把不等式约束 $f_i(\mathbf{x}) \leq b_i$ 和等式约束 $h_j(\mathbf{x}) = 0$ 去掉时, 即退化为无约束优化问题。
2. 约束优化问题 (Constrained Optimization) 是指带有约束条件的问题, 即变量 \mathbf{x} 需要满足一些等式或不等式的约束。在优化问题(10.7)中, 当不等式约束 $f_i(\mathbf{x}) \leq b_i$ 和等式约束 $h_j(\mathbf{x}) = 0$ 只要有一个成立, 其即被称为约束优化问题。

随机优化和确定性优化

根据是目标或约束函数中是否涉及随机变量，可以将优化问题分为随机优化问题和确定性优化问题。

1. **随机优化问题** (Stochastic Optimization) 是指目标或约束函数中涉及随机变量而带有不确定性的问题。在实际问题中，只能知道参数的某些估计。随机优化在机器学习、深度学习和强化学习中有着重要应用。
2. **确定性优化问题** (Deterministic Optimization) 是指目标和约束函数都是确定的优化问题。

许多确定性优化算法都有相应的随机版本，使得在特定问题上具有更低的计算复杂度和更好的收敛性质。

线性规划和非线性规划

根据函数的线性性质，可以将优化问题分为线性规划（线性优化）和非线性规划（非线性优化）。

1. 在优化问题(10.7)中，当目标函数和所有的约束函数都为线性函数，则该问题为**线性规划问题** (Linear Programming)。线性规划问题在约束优化问题中具有较为简单的形式，目前求解线性规划问题最流行的两类方法为单纯形法和内点法。
2. 在优化问题(10.7)中，如果目标函数或任何一个约束函数为非线性函数，则该问题为**非线性规划问题** (Nonlinear Programming)。

本课程将要介绍的优化问题主要为非线性优化问题。

凸优化与非凸优化

更进一步，根据目标函数和可行域的凸性，我们还可以把优化问题分为凸优化 (Convex Programming) 和非凸优化。

1. **凸优化问题**是一种特殊的约束优化问题，需满足目标函数为凸函数，并且等式约束函数为线性函数，不等式约束函数为凸函数。
2. **非凸优化问题**对应于标准形式(10.7)中的一个或多个目标函数或约束函数不具有凸性的问题。

在凸优化问题中，任意局部最优解都是全局最优解，因此算法设计和理论分析上比非凸优化问题简单很多。

参数优化与超参数优化

在机器学习中，优化问题又可以分为参数优化和超参数优化。

- 模型 $f(\mathbf{x}; \boldsymbol{\theta})$ 中的 $\boldsymbol{\theta}$ 称为模型的参数，可以通过优化算法进行学习，除了可学习的参数 $\boldsymbol{\theta}$ 之外，还有一类参数是用来定义模型结构或优化策略的，这类参数叫做超参数（Hyper-Parameter）。
- 常见的超参数包括：聚类算法中的类别个数、梯度下降法的步长、正则项的系数、神经网络的层数、支持向量机中的核函数等。
- 超参数的选取一般都是组合优化问题，很难通过优化算法来自动学习。
- 因此，超参数优化是机器学习的一个经验性很强的技术，通常通过搜索的方法对一组超参数组合进行不断试错调整。

除了上述分类，还有按目标函数的个数分类：单目标最优化问题，多目标最优化问题；以及按约束条件和目标函数是否是时间的函数分类：静态最优化问题和动态最优化问题（动态规划）。

10.2 凸集

凸优化是一类很重要的优化问题，它的理论基础涉及到凸集和凸函数。本节我们将首先给出凸集的定义并介绍一些相关的例子，然后给出保持凸集的一些基本运算，最后给出在机器学习中常用的一个性质，分离超平面定理。

10.2.1 凸集

在给出凸集的定义之前，我们首先回顾线段和仿射集合的定义。

直线与线段

定义 10.2.1. 对于 R^n 中的两个点 $\mathbf{x}_1 \neq \mathbf{x}_2$ ，形如

$$\mathbf{y} = \theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2, \theta \in \mathbb{R}$$

的点形成了过点 \mathbf{x}_1 和 \mathbf{x}_2 的直线。当 $0 \leq \theta \leq 1$ 时，这样的点构成了连接点 \mathbf{x}_1 和 \mathbf{x}_2 的线段。

\mathbf{y} 的表示形式 $\mathbf{y} = \mathbf{x}_2 + \theta(\mathbf{x}_1 - \mathbf{x}_2)$ 给出了另一种解释：直线上的点 \mathbf{y} 是基点 \mathbf{x}_2 （对应 $\theta = 0$ ）和方向 $\mathbf{x}_1 - \mathbf{x}_2$ （由 \mathbf{x}_2 指向 \mathbf{x}_1 ）乘以参数 θ 的和。当 θ 由 0 增加到 1，点 \mathbf{y} 相应地由 \mathbf{x}_2 移动到 \mathbf{x}_1 。如果 $\theta > 1$ ，点 \mathbf{y} 在超越了 \mathbf{x}_1 的直线上。

仿射集

定义 10.2.2. 如果通过集合 $C \subseteq \mathbb{R}^n$ 中任意两点的直线仍然在集合 C 中, 则称 C 为仿射集。即:

$$\mathbf{x}_1, \mathbf{x}_2 \in C \Rightarrow \theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2 \in C, \forall \theta \in \mathbb{R}. \quad (10.9)$$

可以归纳得出: 一个仿射集包含其中任意点的仿射组合。如果 C 是一个仿射集并且 $\mathbf{x}_0 \in C$, 则集合

$$\mathbb{V} = C - \mathbf{x}_0 = \{\mathbf{x} - \mathbf{x}_0 | \mathbf{x} \in C\}$$

是一个子空间, 即关于加法和数乘是封闭的。因此, 仿射集 C 可以表示为

$$C = \mathbb{V} + \mathbf{x}_0 = \{\mathbf{v} + \mathbf{x}_0 | \mathbf{v} \in \mathbb{V}\}$$

即一个子空间加上一个偏移。与仿射集 C 相关联的子空间 \mathbb{V} 与 \mathbf{x}_0 的选取无关, 所以 \mathbf{x}_0 可以是 C 中的任意一点。我们定义仿射集 C 的维数为子空间 $\mathbb{V} = C - \mathbf{x}_0$ 的维数, 其中 \mathbf{x}_0 是 C 中的任意元素。

例 10.2.1. 线性方程组的解集。线性方程组的解集 $C = \{\mathbf{x} | A\mathbf{x} = \mathbf{b}\}$ 是一个仿射集合, 其中 $A \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$ 。为说明这点, 任取 $\mathbf{x}_1, \mathbf{x}_2 \in C$, 则有 $A\mathbf{x}_1 = \mathbf{b}$, $A\mathbf{x}_2 = \mathbf{b}$ 。对于任意 θ , 我们有

$$A(\theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2) = \theta A\mathbf{x}_1 + (1 - \theta) A\mathbf{x}_2 = \theta \mathbf{b} + (1 - \theta) \mathbf{b} = \mathbf{b}$$

这表明, 任意的仿射组合 $\theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2$ 也在仿射集合 C 中。

我们称由集合 $C \subseteq \mathbb{R}^n$ 中的点的所有仿射组合组成的集合为 C 的仿射包, 记为 $\text{aff } C$:

$$\text{aff } C = \{\theta_1 \mathbf{x}_1 + \cdots + \theta_k \mathbf{x}_k | \mathbf{x}_1, \dots, \mathbf{x}_k \in C, \theta_1 + \cdots + \theta_k = 1\}$$

仿射包是包含 C 的最小的仿射集合, 也就是说: 如果 S 是满足 $C \subseteq S$ 的仿射集合, 那么 $\text{aff } C \subseteq S$ 。下图 10.3 展示了 \mathbb{R}^3 中圆盘 S 的仿射包, 为一个平面。

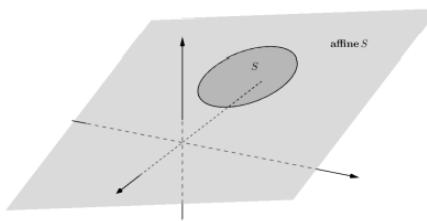


图 10.3: 仿射包

凸集

下面我们给出凸集的定义。

定义 10.2.3. 如果连接集合 C 中任意两点的线段都在 C 内，则称 C 为凸集，即

$$x_1, x_2 \in C \Rightarrow \theta x_1 + (1 - \theta) x_2 \in C, \forall 0 \leq \theta \leq 1.$$

从仿射集的定义中可以看出仿射集是凸集。

例 10.2.2. 下图显示了 \mathbb{R}^2 空间中一些简单的凸和非凸集合。

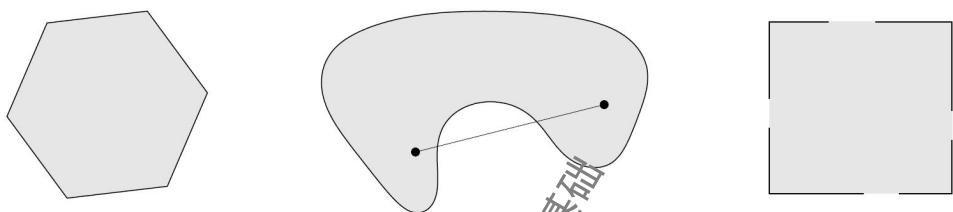


图 10.4: 一些简单的凸和非凸集合。(左) 包含其边界的六边形是凸的; (中) 肾形集合不是凸的, 因为图中所示集合中两点间的线段不为集合所包含。(右) 仅包含部分边界的正方形不是凸的。

从凸集可以引出凸组合和凸包等概念。

定义 10.2.4. 形如

$$x = \theta_1 x_1 + \cdots + \theta_k x_k, 1 \leq \theta_1 + \theta_2 + \cdots + \theta_k, \theta_i \geq 0, i = 1, 2, \dots, k$$

的点称为 x_1, \dots, x_k 的凸组合。集合 C 中点所有可能的凸组合构成的集合称作 C 的凸包, 记作 $\text{conv } C$ 。

点的凸组合可以看做是这些点的混合或加权平均, θ_i 代表混合时 x_i 所占的份量。凸包是包含 C 的最小的凸集。即 $\text{conv } C \subseteq B$ 。下图显示了凸包的定义。

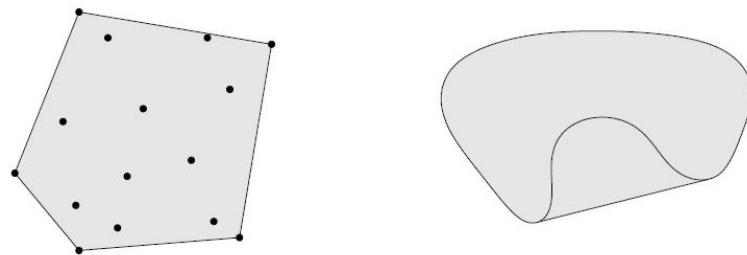


图 10.5: \mathbb{R}^2 上两个集合的凸包。(左) 十五个点的集合的凸包是一个五边形(阴影所示); (右) 肾形集合的凸包是阴影所示的集合。

锥

在集合中，有一类特殊的集合，称为锥，我们可以把凸集的定义推广到锥集合上，形成凸锥。

定义 10.2.5. 如果对于任意 $x \in \mathbb{C}$ 和 $\theta \geq 0$ 都有 $\theta x \in \mathbb{C}$ ，我们称集合 \mathbb{C} 是锥。形如 $x = \theta_1 x_1 + \theta_2 x_2 \in \mathbb{C}, \theta_1 \geq 0, \theta_2 \geq 0$ 的点称为点 x_1, x_2 的锥组合。若集合 \mathbb{C} 中任意点的锥组合都在 \mathbb{C} 中，则称 \mathbb{C} 为凸锥。

在几何上，具有此类形式的点构成了二维的扇形，这个扇形以 0 为定点，边通过 x_1 和 x_2 ，如图 10.6 所示：

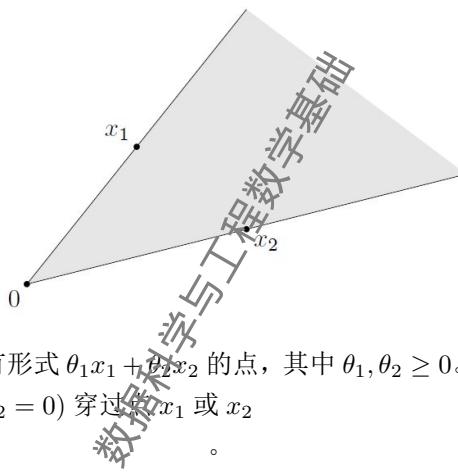


图 10.6: 扇形显示了所有具有形式 $\theta_1 x_1 + \theta_2 x_2$ 的点，其中 $\theta_1, \theta_2 \geq 0$ 。扇形的顶点 ($\theta_1 = \theta_2 = 0$) 在 O 处，其边界 ($\theta_1 = 0$ 或 $\theta_2 = 0$) 穿过 x_1 或 x_2 。

我们称集合 C 中所有元素的锥组合的集合为其锥包，即：

$$\{\theta_1 x_1 + \cdots + \theta_k x_k \mid x_i \in C, \theta_i \geq 0, i = 1\},$$

它是包含 C 的最小的凸锥。

10.2.2 重要的凸集例子

本节将描述一些重要的凸集，这些凸集在本书的后续部分将会多次遇见。

简单的例子

- 空集，任意一个点（即单点集） $\{x_0\}$ 、全空间 \mathbb{R}^n 都是 \mathbb{R}^n 的仿射（自然也是凸的）子集。
- 任意直线是仿射的。如果直线通过零点，则是子空间，因此，也是凸锥。
- 一条线段是凸的，但不是仿射的（除非退化为一个点）。

- 一条射线，即具有形式 $\{x_0 + \theta v | \theta \geq 0\}$, $v \neq \mathbf{0}$ 的集合，是凸的，但不是仿射的。如果射线的基点 x_0 是 $\mathbf{0}$ ，则它是凸锥。
- 任意子空间是仿射的，也是凸的。
- 凸锥（自然是凸的）。

超平面与半空间

任取非零向量 a ，形如 $\{x | a^T x = b\}$ 的集合称为超平面，形如 $\{x | a^T x \leq b\}$ 的集合称为半空间。其中 $a \in \mathbb{R}^n$, $a \neq \mathbf{0}$ 是对应的超平面和半空间的法向量，且 $b \in \mathbb{R}$ 。

解析地，超平面是关于 x 的非平凡线性方程的解空间（因此是一个仿射集合）。一个超平面将 \mathbb{R}^n 分成两个半空间。超平面是仿射集和凸集，半空间是凸集但不是仿射集。超平面的几何解释如图 10.7。 \mathbb{R}^2 中由法向量 a 和超平面上一点 x_0 确定的超平面。对于超平面上任意一点 x , $x - x_0$ （如深色箭头所示）都垂直于 a 。半空间的几何解释如图 10.8。 \mathbb{R}^2 上由 $a^T x = b$ 定义的超

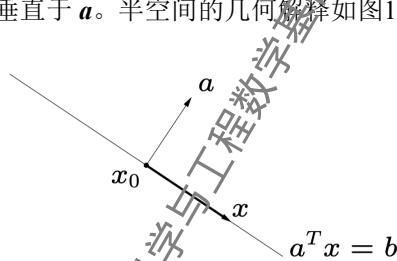


图 10.7: 超平面几何解释

平面决定了两个半空间，由 $a^T x \geq b$ 决定的半空间（无阴影）是向 a 扩展的半空间，由 $a^T x \leq b$ 确定的半空间（阴影所示）向 $-a$ 方向扩展。向量 a 是这个半空间向外的法向量。

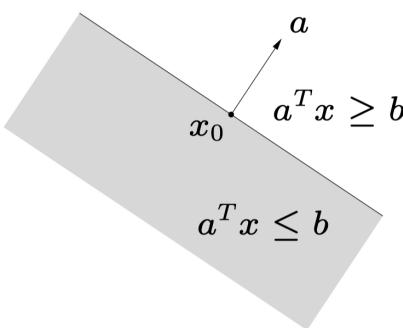


图 10.8: 半平面几何解释

Euclid 球与范数球

球是空间中到某个点距离（或两者差的范数）小于某个常数的点的集合，并将

$$B(\mathbf{x}_c, r) = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_c\|_2 \leq r\} = \{\mathbf{x}_c + r\mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\}$$

称为中心为 \mathbf{x}_c ，半径为 r 的 Euclid 球。Euclid 球是凸集，即如果 $\|\mathbf{x}_1 - \mathbf{x}_c\|_2 \leq r, \|\mathbf{x}_2 - \mathbf{x}_c\|_2 \leq r$ ，并且 $0 \leq \theta \leq 1$ ，那么

$$\begin{aligned} \|\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2 - \mathbf{x}_c\|_2 &= \|\theta(\mathbf{x}_1 - \mathbf{x}_c) + (1 - \theta)(\mathbf{x}_2 - \mathbf{x}_c)\|_2 \\ &\leq \theta\|\mathbf{x}_1 - \mathbf{x}_c\|_2 + (1 - \theta)\|\mathbf{x}_2 - \mathbf{x}_c\|_2 \\ &\leq r \end{aligned}$$

椭球

形如

$$\{\mathbf{x} \mid (\mathbf{x} - \mathbf{x}_c)^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}_c) \leq 1\}$$

的集合称为椭球，其中 $\mathbf{P} \in \mathcal{S}_{++}^n$ (即 \mathbf{P} 对称正定)。椭球的另一种表示为

$$\{\mathbf{x}_c + \mathbf{A}\mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\},$$

其中 \mathbf{A} 为非奇异的方阵。

范数球与范数维

设 $\|\cdot\|$ 是 \mathbb{R}^n 中的范数。称

$$\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_c\| \leq r\}$$

为以 r 为半径， \mathbf{x}_c 为球心的范数球。根据范数的三角不等式性质可知，范数球是一个凸集。

关于范数 $\|\cdot\|$ 的范数锥是集合

$$C = \{(\mathbf{x}, t) \mid \|\mathbf{x}\| \leq t\} \subseteq \mathbb{R}^{n+1}$$

顾名思义，它是一个凸锥。

多面体

我们将满足线性等式和不等式组的点的集合称为多面体，即

$$\{\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{C}\mathbf{x} = \mathbf{d}\},$$

其中 $\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{C} \in \mathbb{R}^{p \times n}, \mathbf{x} \leq \mathbf{y}$ 表示向量 \mathbf{x} 的每个分量均小于等于 \mathbf{y} 的对应分量。

因此，多面体是有限个半空间和超平面的交集。仿射集合（例如子空间、超平面、直线）、射线、线段和半空间都是多面体。显而易见，多面体是凸集。有界的多面体有时也称为多胞形，但也有一些作者反过来使用这两个概念（即用多胞形表示具有上面形式的集合，而当其有界时称为多面体）。图 10.9 显示了由五个半空间的交集组成的多面体。

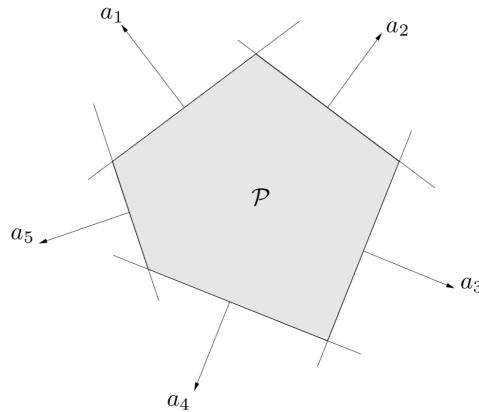


图 10.9: 多面体 \mathcal{P} (阴影所示) 是外法向量为 a_1, \dots, a_5 的五个半空间的交集。

例 10.2.3. 非负象限是具有非负分量的点的集合, 即

$$\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x_i \geq 0, i = 1, \dots, n\} = \{x \in \mathbb{R}^n \mid x \succeq 0\}.$$

(此处 \mathbb{R}_+ 表示非负实数的集合, 即 $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$ 。) 非负象限既是多面体也是锥 (因此称为多面体锥)。

单纯形

单纯形是一类重要的多面体。设 $k+1$ 个点 $\mathbf{v}_0, \dots, \mathbf{v}_k \in \mathbb{R}^n$ 仿射独立, 则 $\mathbf{v}_1 - \mathbf{v}_0, \dots, \mathbf{v}_k - \mathbf{v}_0$ 线性独立。那么, 这些点决定了一个单纯形, 如下所示:

$$C = \text{conv}\{\mathbf{v}_0, \dots, \mathbf{v}_k\} = \{\theta_0 \mathbf{v}_0 + \dots + \theta_k \mathbf{v}_k \mid \theta \geq 0, \mathbf{1}^T \theta = 1\}$$

其中 $\mathbf{1}$ 表示所有分量均为 1 的向量。

例 10.2.4. 一些常见的单纯形。1 维单纯形是一条线段; 2 维单纯形是一个三角形 (包含其内部); 3 维单纯形是一个四面体。

单位单纯形是由零向量和单位向量 $\mathbf{0}, \mathbf{e}_1, \dots, \mathbf{e}_n \in \mathbb{R}^n$ 决定的 n 维单纯形。它可以表示为满足下列条件的向量的集合,

$$x \succeq 0, \quad \mathbf{1}^T x \leq 1$$

概率单纯形是有单位向量 $\mathbf{e}_1, \dots, \mathbf{e}_n \in \mathbb{R}^n$ 决定的 $n-1$ 维单纯形。它是满足下列条件的向量的集合,

$$x \succeq 0, \quad \mathbf{1}^T x = 1$$

概率单纯形中的向量对应于含有 n 个元素的集合的概率分布, x_i 可理解为第 i 个元素的概率。

半正定锥

我们用 S^n 表示对称 $n \times n$ 矩阵的集合, 即

$$S^n = \{X \in \mathbb{R}^{n \times n} | X = X^T\}$$

这是一个维数为 $n(n+1)/2$ 的向量空间。我们用 S_+^n 表示对称半正定矩阵的集合:

$$S_+^n = \{X \in S^n | X \succeq 0\}$$

用 S_{++}^n 表示对称正定矩阵集合:

$$S_{++}^n = \{X \in S_+^n | X \succ 0\}$$

集合 S_+^n 是一个凸锥: 如果 $\theta_1, \theta_2 \geq 0$ 并且 $A, B \in S_+^n$, 那么 $\theta_1 A + \theta_2 B \in S_+^n$ 。从半正定矩阵的定义可以直接得到: 对于任意 $x \in \mathbb{R}^n$, 如果 $A \succeq 0, B \succeq 0$, 那么, 就有

$$x^T(\theta_1 A + \theta_2 B)x = \theta_1 x^T A x + \theta_2 x^T B x \geq 0$$

10.2.3 保持凸集的运算

判定一个集合为凸集的方式, 可以使用定义的方式, 即判别如下结论是否成立:

$$x_1, x_2 \in C, 0 \leq \theta \leq 1 \Rightarrow \theta x_1 + (1 - \theta) x_2 \in C$$

然而, 在实际中使用定义判别是比较困难的。事实上, 常见的集合可以由简单的凸集经过一些运算的方式得到, 而这些运算具有保凸性。这为我们判断凸集提供了极大的便利。本节将介绍一些常见的保凸运算: 取交集、仿射变换、线性分式及透视函数。

集合运算

交集 交集运算是保凸的: 如果 S_1 和 S_2 是凸集, 那么 $S_1 \cap S_2$ 也是凸集。这个性质可以扩展到无穷个集合的交: 如果对于任意 $\alpha \in \mathcal{A}$ 都有 S_α 是凸集, 那么, $\cap_{\alpha \in \mathcal{A}} S_\alpha$ 也是凸集。(子空间和仿射集合对于任意交运算也是封闭的。) 作为一个简单的例子, 多面体是半空间和超平面(它们都是凸集)的交集, 因而是凸的。

例 10.2.5. 半正定锥 S_+^n 可以表示为,

$$\cap_{z \neq 0} \{X \in S^n | z^T X z \geq 0\}.$$

对于任意 $z \neq 0$, $z^T X z$ 是关于 X 的(不恒等于零的)线性函数, 因此集合

$$\{X \in S^n | z^T X z \geq 0\}$$

实际上就是 S^n 的半空间。由此可见, 半正定锥是无穷个半空间的交集, 因此是凸的。

例 10.2.6. 考虑集合

$$S = \{x \in \mathbb{R}^m | |p(t)| \leq 1 \text{ 对于 } |t| \leq \pi/3\}$$

其中 $p(t) = \sum_{k=1}^m x_k \cos kt$ 。集合 S 可以表示为无穷个平板的交集: $S = \cap_{|t| \leq \pi/3} S_t$, 其中

$$S_t = \{x | -1 \leq (\cos t, \dots, \cos mt)^T x \leq 1\}$$

因此, S 是凸的。对于 $m = 2$ 的情况, 它的定义和集合可见图 10.10 和图 10.11。

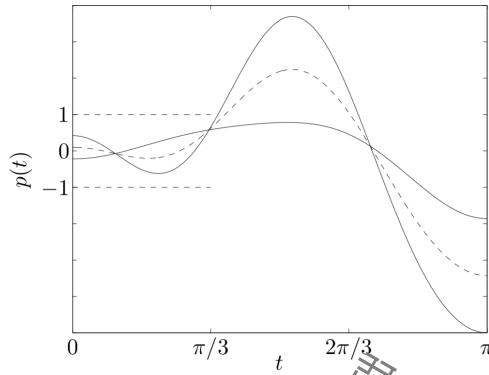


图 10.10: 对应于 $m = 2$ 中的点的三角多项式. 虚线所示的三角多项式是另外两个的平均.

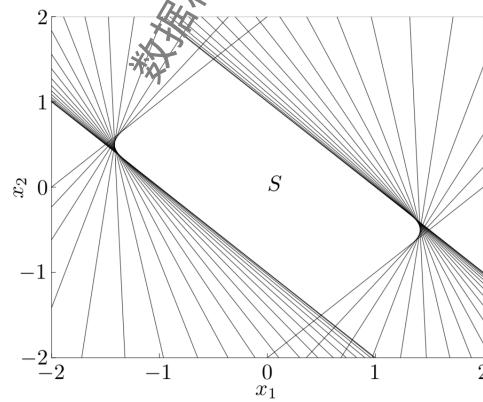


图 10.11: 图中央的白色区域显示了 $m = 2$ 情况下例定义的集合 s . 这个集合是无限多个 (图中显示了其中 20 个) 平板的交集, 所以是凸的.

在上述例子中, 通过将集合表示为(可能无穷多个)半空间的交集来证明集合的凸性。反过来, 我们也可以看到: 每个闭凸集 S 是(通常为无限多个)半空间的交集。事实上, 一个闭凸集 S 是包含它的所有半空间的交集:

$$S = \bigcap \{\mathcal{H} \mid \mathcal{H} \text{是半空间}, S \subseteq \mathcal{H}\}.$$

和运算 两个集合的和可以定义为:

$$S_1 + S_2 = \{\mathbf{x} + \mathbf{y} \mid \mathbf{x} \in S_1, \mathbf{y} \in S_2\}$$

如果 S_1 和 S_2 是凸集, 那么, $S_1 + S_2$ 是凸的。

我们也可以考虑 $S_1, S_2 \in \mathbb{R}^n \times \mathbb{R}^m$ 的部分和, 定义为

$$S = \{(\mathbf{x}, \mathbf{y}_1 + \mathbf{y}_2) \mid (\mathbf{x}, \mathbf{y}_1) \in S_1, (\mathbf{x}, \mathbf{y}_2) \in S_2\}$$

其中 $\mathbf{x} \in \mathbb{R}^n, \mathbf{y}_i \in \mathbb{R}^m$ 。 $m = 0$ 时, 部分和给出了 S_1 和 S_2 的交集; $n = 0$, 部分和等于集合之和。凸集的部分和仍然是凸集。

Cartesian 乘积 可以看出, 如果 S_1 和 S_2 是凸的, 那么其直积或 Cartesian 乘积

$$S_1 \times S_2 = \{(\mathbf{x}_1, \mathbf{x}_2) \mid \mathbf{x}_1 \in S_1, \mathbf{x}_2 \in S_2\}$$

也是凸集。

集合投影 一个凸集向它的某几个坐标的投影是凸的, 即: 如果 $S \subseteq \mathbb{R}^m \times \mathbb{R}^n$ 是凸集, 那么

$$T = \{\mathbf{x}_1 \in \mathbb{R}^m \mid (\mathbf{x}_1, \mathbf{x}_2) \in S, \text{对于某些 } \mathbf{x}_2 \in \mathbb{R}^n\}$$

是凸集。

函数映射

仿射函数 设 $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ 是仿射变换, 即 $f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$, 则

(1) 凸集在 f 下的像是凸集:

$$S \subseteq \mathbb{R}^n \text{ 为凸集} \Rightarrow f(S) \stackrel{\text{def}}{=} \{f(\mathbf{x}) \mid \mathbf{x} \in S\} \text{ 为凸集};$$

(2) 凸集在 f 下的原像是凸集:

$$C \subseteq \mathbb{R}^m \text{ 为凸集} \Rightarrow f^{-1}(C) \stackrel{\text{def}}{=} \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \in C\} \text{ 为凸集}.$$

例 10.2.7. 伸缩和平移: 如果 $S \subseteq \mathbb{R}^n$ 是凸集, $\alpha \in \mathbb{R}$ 并且 $\mathbf{a} \in \mathbb{R}^n$, 那么, 集合 αS 和 $S + \mathbf{a}$ 是凸的, 其中

$$\alpha S = \{\alpha \mathbf{x} \mid \mathbf{x} \in S\}, \quad S + \mathbf{a} = \{\mathbf{x} + \mathbf{a} \mid \mathbf{x} \in S\}.$$

例 10.2.8. 集合投影: 一个凸集向它的某几个坐标的投影是凸的, 即: 如果 $S \subseteq \mathbb{R}^m \times \mathbb{R}^n$ 是凸集, 那么

$$T = \{\mathbf{x}_1 \in \mathbb{R}^m \mid (\mathbf{x}_1, \mathbf{x}_2) \in S \text{ 对于某些 } \mathbf{x}_2 \in \mathbb{R}^n\}$$

是凸集。

例 10.2.9. 利用仿射变换保凸的性质, 可以证明 线性矩阵不等式的解集:

$$\{\mathbf{x} \mid x_1 \mathbf{A}_1 + x_2 \mathbf{A}_2 + \cdots + x_m \mathbf{A}_m \preceq \mathbf{B}\}$$

是凸集, 其中 $\mathbf{A}_i, \mathbf{B} \in S^n$ 。因为, 它可以看作是一个仿射变换的原像

例 10.2.10. 双曲锥:

$$\{\mathbf{x} \mid \mathbf{x}^T \mathbf{P} \mathbf{x} \leq (\mathbf{c}^T \mathbf{x})^2, \mathbf{c}^T \mathbf{x} \geq 0\}$$

是凸集, 其中 $\mathbf{P} \in S_+^n$ 。因为, 它可以看作是 $\mathbf{x} \rightarrow (\mathbf{P}^{\frac{1}{2}} \mathbf{x}, \mathbf{c}^T \mathbf{x})$ 变换下的原像, 而值域是凸锥。

透视函数

定义 10.2.6. 我们定义 $P : K \rightarrow \mathbb{R}^n$,

$$P(\mathbf{z}, t) = \mathbf{z}/t$$

为透视函数, 其定义域为 $\text{dom } f = K = \mathbb{R}^n \times \mathbb{R}_{++}$ 。

透视函数对向量进行伸缩, 或称为规范化, 使得最后一维分量为 1 并舍弃之。如果 $C \subseteq K$ 是凸集, 那么它的像

$$P(C) = \{P(\mathbf{x}) \mid \mathbf{x} \in C\}$$

也是凸集。这个结论很直观: 通过小孔观察一个凸的物体, 可以得到凸的像。为解释这个事实, 下面我们将说明在透视函数作用下, 线段将被映射成线段。

假设 $\mathbf{x} = (\tilde{\mathbf{x}}, x_{n+1}), \mathbf{y} = (\tilde{\mathbf{y}}, y_{n+1}) \in \mathbb{R}^{n+1}$ 并且 $x_{n+1} > 0, y_{n+1} > 0$ 。那么, 对于 $0 \leq \theta \leq 1$,

$$P(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) = \frac{\theta \tilde{\mathbf{x}} + (1 - \theta) \tilde{\mathbf{y}}}{\theta x_{n+1} + (1 - \theta) y_{n+1}} = \mu P(\mathbf{x}) + (1 - \mu) P(\mathbf{y})$$

其中,

$$\mu = \frac{\theta x_{n+1}}{\theta x_{n+1} + (1 - \theta) y_{n+1}} \in [0, 1]$$

θ 和 μ 之间的关系是单调的: 当 θ 在 $[0, 1]$ 间变化时 (形成线段 $[\mathbf{x}, \mathbf{y}]$), μ 也在 $[0, 1]$ 间变化 (形成线段 $[P(\mathbf{x}), P(\mathbf{y})]$)。这说明 $P([\mathbf{x}, \mathbf{y}]) = [P(\mathbf{x}), P(\mathbf{y})]$ 。

现在假设 C 是凸的，并且有 $C \subseteq K$ ，即对于所有 $\mathbf{x} \in C, x_{n+1} > 0$ 及 $\mathbf{x}, \mathbf{y} \in C$ 。为显示 $P(C)$ 的凸性，我们需要说明线段 $[P(\mathbf{x}), P(\mathbf{y})]$ 在 $P(C)$ 中。这条线段是线段 $[\mathbf{x}, \mathbf{y}]$ 在 P 的象，因而属于 $P(C)$ 。

一个凸集在透视函数下的原象也是凸的：如果 $C \subseteq \mathbb{R}^n$ 为凸集，那么

$$P^{-1}(C) = \{(\mathbf{x}, t) \in \mathbb{R}^{n+1} | \mathbf{x}/t \in C, t > 0\}$$

是凸集。

为证明这点，假设 $(\mathbf{x}, t) \in P^{-1}(C), (\mathbf{y}, s) \in P^{-1}(C), 0 \leq \theta \leq 1$ 。我们需要说明

$$\theta(\mathbf{x}, t) + (1 - \theta)(\mathbf{y}, s) \in P^{-1}(C)$$

即

$$\frac{\theta \mathbf{x} + (1 - \theta) \mathbf{y}}{\theta t + (1 - \theta)s} \in C$$

显然地， $\theta t + (1 - \theta)s > 0$ 。这可从下式看出，

$$\frac{\theta x + (1 - \theta)y}{\theta t + (1 - \theta)s} = \mu(\mathbf{x}/t) + (1 - \mu)(\mathbf{y}/s)$$

其中，

$$\mu = \frac{\theta t}{\theta t + (1 - \theta)s} \in [0, 1]$$

线性分式函数

线性分式函数由透视函数和仿射函数复合而成。设 $g: \mathbb{R}^n \rightarrow \mathbb{R}^{m+1}$ 是仿射的，即

$$g(\mathbf{x}) = \begin{bmatrix} \mathbf{A} \\ \mathbf{c}^T \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{b} \\ d \end{bmatrix}$$

其中 $\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m, \mathbf{c} \in \mathbb{R}^n$ 并且 $d \in \mathbb{R}$ 。则由 $f = P \circ g$ 给出的函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$

$$f(\mathbf{x}) = (\mathbf{A}\mathbf{x} + \mathbf{b})/(\mathbf{c}^T \mathbf{x} + d), \text{dom } f = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0\}$$

称为线性分式(或投射)函数。如果 $\mathbf{c} = \mathbf{0}, d > 0$ 。则 f 的定义域为 \mathbb{R}^n ，并且 f 是仿射函数。因此，我们可以将仿射和透视函数视为特殊的线性分式函数。

类似于透视函数，线性分式函数也是保凸的。如果 C 是凸集并且在 f 的定义域中(即任意 $\mathbf{x} \in C$ 满足 $\mathbf{c}^T \mathbf{x} + d > 0$)，那么 C 的象 $f(C)$ 也是凸集。根据前述的结果可以直接得到这个结论： C 在仿射映射下的象是凸的，并且在透视函数 P 下的映射(即 $f(C)$)是凸的。类似地，如果 $C \subseteq \mathbb{R}^m$ 是凸集，那么其原象 $f^{-1}(C)$ 也是凸的。

例 10.2.11. 条件概率。设 u 和 v 是分别在 $\{1, \dots, n\}$ 和 $\{1, \dots, m\}$ 中取值的随机变量，并且 p_{ij} 表示概率 $\text{prob}(u = i, v = j)$ 。那么条件概率 $f_{ij} = \text{prob}(u = i | v = j)$ 由下式给出

$$f_{ij} = \frac{p_{ij}}{\sum_{k=1}^n p_{kj}}$$

因此， f 可以通过一个线性分式映射从 p 得到。可以知道，如果 C 是一个关于 (u, v) 的联合密度的凸集，那么相应的 u 的条件密度(给定 v)的集合也是凸集。

10.2.4 分离与支撑超平面

分离超平面

本节中我们将阐述一个在之后非常重要的想法: 用超平面或仿射函数将两个不相交的凸集分离开来。

定义 10.2.7. 假设 C 和 D 是两个不相交的凸集, 即 $C \cap D = \emptyset$ 。如果仿射函数 $\mathbf{a}^T \mathbf{x} - b$ 在 C 中非正, 而在 D 中非负, 那么, 超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ 被称为集合 C 和 D 的分离超平面, 或者说超平面分离了集合 C 和 D 。如果对于任意 $\mathbf{x} \in C$ 有 $\mathbf{a}^T \mathbf{x} < b$, 并且对于任意 $\mathbf{x} \in D$ 有 $\mathbf{a}^T \mathbf{x} > b$, 则称其为集合 C 和 D 的严格分离。

图 10.12 给出了严格分离超平面示例。超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ 分离了两个不相交的凸集 C 和 D 。仿射函数 $\mathbf{a}^T \mathbf{x} - b$ 在 C 上非正而在 D 上非负。

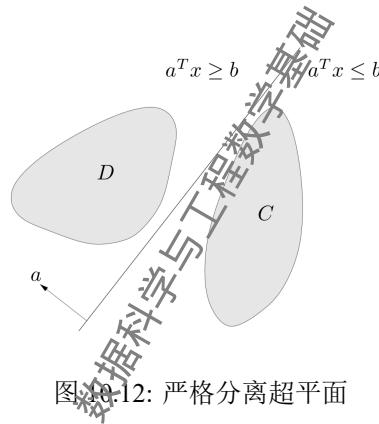


图 10.12: 严格分离超平面

实际上, 对于任意两个不相交的凸集, 必存在这样的分离超平面, 即如下超平面分离定理。

定理 10.2.1. 超平面分离定理: 假设 C 和 D 是两个不相交的凸集, 则存在非零向量 \mathbf{a} 和常数 b , 使得

$$\mathbf{a}^T \mathbf{x} \leq b, \forall \mathbf{x} \in C \text{ 且 } \mathbf{a}^T \mathbf{x} \geq b, \forall \mathbf{x} \in D,$$

即超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ 分离了 C 和 D 。

证明. 这里我们对一个特殊的情况给予证明。我们假设 C 和 D 的 (Euclid) 距离为正, 这里的距离定义为

$$\text{dist}(C, D) = \inf\{\|\mathbf{u} - \mathbf{v}\|_2 | \mathbf{u} \in C, \mathbf{v} \in D\}$$

并且存在 $\mathbf{c} \in C$ 和 $\mathbf{d} \in D$ 达到这个最小距离, 即 $\|\mathbf{c} - \mathbf{d}\|_2 = \text{dist}(C, D)$

定义

$$\mathbf{a} = \mathbf{d} - \mathbf{c}, \mathbf{b} = \frac{\|\mathbf{d}\|_2^2 - \|\mathbf{c}\|_2^2}{2}$$

我们将显示仿射函数

$$f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} - b = (\mathbf{d} - \mathbf{c})^T (\mathbf{x} - (1/2)(\mathbf{d} + \mathbf{c}))$$

在 C 中非正而在 D 中非负, 即超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ 分离了 C 和 D 。这个超平面与连接 \mathbf{c} 和 \mathbf{d} 之间的线段相垂直并且穿过其中点。

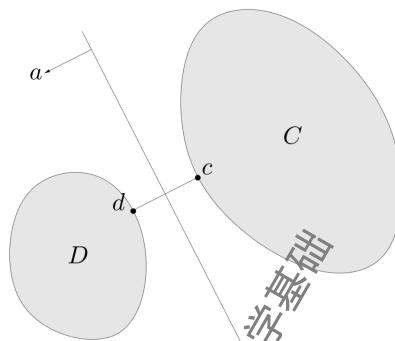


图 10.13: 两个凸集分离超平面的构造。 $c \in C$ 和 $d \in D$ 是两个集合中最靠近彼此的一个点对。分离超平面垂直并等分 c 和 d 直接的线段。

我们首先证明 f 在 D 中非负。关于 f 在 C 中非正的证明是相似的 (只需将 C 和 D 交换并考虑 $-f$ 即可)。假设存在一个点 $\mathbf{u} \in D$, 并且

$$f(\mathbf{u}) = (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - (1/2)(\mathbf{d} + \mathbf{c})) < 0$$

我们可以将 $f(\mathbf{u})$ 表示为

$$f(\mathbf{u}) = (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d} + (1/2)(\mathbf{d} - \mathbf{c})) = (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d}) + (1/2) \|\mathbf{d} - \mathbf{c}\|_2^2$$

这意味着 $(\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d}) \leq 0$ 。于是, 我们观察到

$$\frac{d}{dt} \|\mathbf{d} + t(\mathbf{u} - \mathbf{d}) - \mathbf{c}\|_2^2 \Big|_{t=0} = 2(\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d}) \leq 0$$

因此, 对于足够小的 $t > 0$ 及 $t \leq 1$ 我们有

$$\|\mathbf{d} + t(\mathbf{u} - \mathbf{d}) - \mathbf{c}\|_2 \leq \|\mathbf{d} - \mathbf{c}\|_2$$

即点 $\mathbf{d} + t(\mathbf{u} - \mathbf{d})$ 比 \mathbf{d} 更靠近 \mathbf{c} 。因为 D 是包含 \mathbf{d} 和 \mathbf{u} 的凸集, 我们有 $\mathbf{d} + t(\mathbf{u} - \mathbf{d}) \in D$ 。但这是不可能的, 因为根据假设, \mathbf{d} 应当是 D 中离 C 最近的点。 \square

一般地, 严格分离 (即上式成立严格不等号) 需要更强的假设。例如当 C 是闭凸集, D 是单点集时, 我们有如下严格分离定理:

定理 10.2.2. 严格分离定理: 设 C 是闭凸集, 点 $\mathbf{x}_0 \notin C$, 则存在非零向量 \mathbf{a} 和常数 b , 使得 $\mathbf{a}^T \mathbf{x} < b, \forall \mathbf{x} \in C$ 且 $\mathbf{a}^T \mathbf{x}_0 > b$.

支撑超平面

当点 \mathbf{x}_0 恰好在凸集 C 的边界上时, 可以构造支撑超平面。

定义 10.2.8. 给定集合 C 及其边界上一点 \mathbf{x}_0 , 如果 $\mathbf{a} \neq \mathbf{0}$ 满足 $\mathbf{a}^T \mathbf{x} \leq \mathbf{a}^T \mathbf{x}_0, \forall \mathbf{x} \in C$, 那么称集合

$$\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$$

为 C 在边界点 \mathbf{x}_0 处的支撑超平面。这等于说点 \mathbf{x}_0 与集合 C 被超平面所分离。

从几何上来说, 超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$ 与集合 C 在点 \mathbf{x}_0 处相切并且半空间 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} \leq \mathbf{a}^T \mathbf{x}_0\}$ 包含 C 。

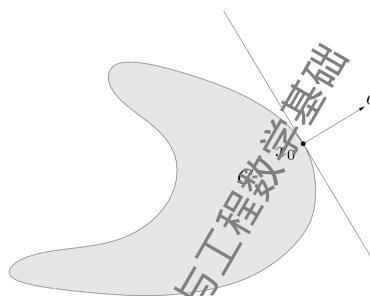


图 10.14: 超平面 $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$ 在 \mathbf{x}_0 处支撑 C

根据凸集的分离超平面定理, 我们有如下一个基本的结论, 称为支撑超平面定理。

定理 10.2.3. 如果 C 是凸集, 则在 C 的任意边界点处都存在支撑超平面。

表明对于任意非空的凸集 C 和任意 $\mathbf{x}_0 \in bdC$. 在 \mathbf{x}_0 处存在 C 的支撑超平面。支撑超平面定理从几何上看, 可以理解为: 给定一个平面后, 可以把凸集边界上的任意一点当成支撑点将凸集放置在该平面上。支撑超平面定理从超平面分离定理很容易得到证明。需要区分两种情况. 如果 C 的内部非空, 对于 $\{\mathbf{x}_0\}$ 和 $\text{int } C$ 应用超平面分离定理可以直接得到所需的结论。如果 C 的内部是空集, 则 C 必处于小于 n 维的一个仿射集合中, 并且任意包含这个仿射集合的超平面一定包含 C 和 \mathbf{x}_0 , 这是一个(平凡的)支撑超平面。

10.3 凸函数

本节我们将首先给出凸函数的定义和相关的例子, 然后给出凸函数的判定条件, 最后给出保持凸函数的一些基本运算。

10.3.1 凸函数的定义和基本性质

凸函数定义

在给出凸函数的定义前, 我们需要对广义实值函数和适当函数进行界定:

定义 10.3.1. (广义实值函数) 令 $\bar{\mathbb{R}} \stackrel{\text{def}}{=} \mathbb{R} \cup \{\pm\infty\}$ 为广义实数空间, 则映射 $f: \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ 称为广义实值函数.

适当函数是一类很重要的广义实值函数, 很多最优化理论都是建立在适当函数之上的.

定义 10.3.2. (适当函数) 给定广义实值函数 f 和非空集合 \mathcal{X} . 如果存在 $\mathbf{x} \in \mathcal{X}$ 使得 $f(\mathbf{x}) < +\infty$, 并且对任意的 $\mathbf{x} \in \mathcal{X}$, 都有 $f(\mathbf{x}) > -\infty$, 那么称函数 f 关于集合 \mathcal{X} 是适当的.

概括来说, 适当函数 f 的特点是“至少有一处取值不为正无穷”, 以及“处处取值不为负无穷”. 对最优化问题 $\min_{\mathbf{x}} f(\mathbf{x})$, 适当函数可以帮助去掉一些我们不感兴趣的函数, 从而在一个比较合理的函数类中考虑最优化问题.

定义 10.3.3. 设函数 $f: \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ 为适当函数, 如果 $\mathbf{dom} f$ 是凸集, 且

$$f(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) \leq \theta f(\mathbf{x}) + (1 - \theta) f(\mathbf{y}) \quad (10.10)$$

对所有 $\mathbf{x}, \mathbf{y} \in \mathbf{dom} f$, $0 \leq \theta \leq 1$ 都成立, 则称 f 是凸函数.

如果对所有的 $\mathbf{x}, \mathbf{y} \in \mathbf{dom} f$, $\mathbf{x} \neq \mathbf{y}$, $0 < \theta < 1$, 上述不等式严格成立, 则称函数 f 是严格凸函数. 从几何意义上讲, 上述不等式意味着连接凸函数图像上任意两点的线段都在函数图像上方. (如图 10.15 所示).

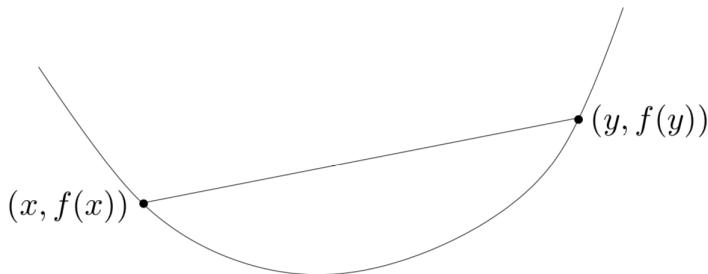


图 10.15: 凸函数示意图。图上任意两点之间的弦 (即线段) 都在函数图像之上。

如果函数 $-f$ 是凸函数, 则称函数 f 是凹函数. 一个函数是仿射函数等价于它是既凸又凹的.

基本性质-Jessen 不等式

如图 10.15 所示, 对于任意 $\mathbf{x}, \mathbf{y} \in C$ 和任意 $0 \leq \theta \leq 1$, 凸函数 f 满足不等式

$$f(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) \leq \theta f(\mathbf{x}) + (1 - \theta) f(\mathbf{y})$$

这个不等式被为 Jessen 不等式。此不等式可以很方便地扩展至更多点的凸组合: 如果函数 f 是凸函数, $\mathbf{x}_1, \dots, \mathbf{x}_k \in C$, $\theta_1, \dots, \theta_k \geq 0$ 且 $\theta_1 + \dots + \theta_k = 1$, 则下式成立

$$f(\theta_1 \mathbf{x}_1 + \dots + \theta_k \mathbf{x}_k) \leq \theta_1 f(\mathbf{x}_1) + \dots + \theta_k f(\mathbf{x}_k)$$

考虑凸集时, 此不等式可以扩展至无穷项和、积分以及期望。例如, 我们可以采用其支撑属于 C 的任意概率测度。如果 \mathbf{x} 是随机变量, 事件 $\mathbf{x} \in C$ 发生的概率为 1, 函数 f 是凸函数, 当相应的期望存在时, 我们有

$$f(\mathbf{E}\mathbf{x}) \leq \mathbf{E}f(\mathbf{x})$$

设随机变量 \mathbf{x} 的可能取值为 $\{\mathbf{x}_1, \mathbf{x}_2\}$, 相应的取值概率为 $\text{prob}(\mathbf{x} = \mathbf{x}_1) = \theta, \text{prob}(\mathbf{x} = \mathbf{x}_2) = 1 - \theta$, 则由一般形式 $f(\mathbf{E}\mathbf{x}) \leq \mathbf{E}f(\mathbf{x})$ 可以得到 $f(\theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2) \leq \theta f(\mathbf{x}_1) + (1 - \theta) f(\mathbf{x}_2)$ 。所以 $f(\mathbf{E}\mathbf{x}) \leq \mathbf{E}f(\mathbf{x})$ 可以刻画凸性: 如果函数 f 不是凸函数, 那么存在随机变量 $\mathbf{x}, \mathbf{x} \in C$ 以概率 1 发生, 使 $f(\mathbf{E}\mathbf{x}) > \mathbf{E}f(\mathbf{x})$ 。上述所有不等式均被称为 Jessen 不等式。

凸函数定义的扩展

为了叙述方便, 下面给出凸函数的一个扩展定义。通常通过定义凸函数在定义域 C 外的值为 ∞ , 从而将这个凸函数延伸至全空间 \mathbb{R}^n 。

定义 10.3.4. 如果 f 是定义在 C 上的凸函数, 按照如下方式定义它的扩展函数 $\tilde{f} : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$

$$\tilde{f} = \begin{cases} f(\mathbf{x}) & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$$

扩展函数 \tilde{f} 是定义在全空间 \mathbb{R}^n 上的, 取值集合为 $\mathbb{R} \cup \{\infty\}$ 。我们也可以从扩展函数 \tilde{f} 的定义中确定原函数 f 的定义域, 即 $C = \{\mathbf{x} | \tilde{f} \leq \infty\}$ 。类似地, 可以通过定义凹函数在定义域外的取值为 $-\infty$ 对其进行延伸。

10.3.2 凸函数举例

前文已经提到所有的线性函数和仿射函数均为凸函数 (同时也是凹函数)。事实上, 通过定义, 可以很容易判断出许多凸函数和凹函数的例子。下面列出一些常见的凸函数和凹函数的例子。

实数集 \mathbb{R} 上常见的凸函数

首先考虑 \mathbb{R} 上的一些函数, 其自变量为 x 。用 \mathbb{R}_{++} 表示正实数, 用 \mathbb{R}_+ 表示非负实数, 以后同。

- 指数函数。对任意 $a \in \mathbb{R}$, 函数 e^{ax} 在 \mathbb{R} 上是凸的。
- 幂函数。当 $a \geq 1$ 或 $a \leq 0$ 时, x^a 是在 \mathbb{R}_{++} 上的凸函数, 当 $0 \leq a \leq 1$ 时 x^a 是在 \mathbb{R}_{++} 上的凹函数。
- 绝对值幂函数。当 $p \geq 1$ 时, 函数 $|x|^p$ 在 \mathbb{R} 上是凸函数。
- 对数函数。函数 $\log x$ 在 \mathbb{R}_{++} 上的凹函数。
- 负熵。函数 $x \log x$ 在其定义域上是凸函数。

空间 \mathbb{R}^n 上常见的凸函数

下面我们给出 \mathbb{R}^n 上的一些例子。

- 仿射函数: $\mathbf{a}^T \mathbf{x} + b$, 其中 $\mathbf{a}, \mathbf{x} \in \mathbb{R}^n$ 是向量。它是 \mathbb{R}^n 上的凸(凹)函数。
- 几何平均。几何平均是其定义域上的凹函数。
- 范数。所有范数都是凸函数(向量和矩阵版本), 这是由于范数满足三角不等式。

机器学习中常见的凸函数

- 负熵。函数 $x \log x$ 在其定义域上是凸函数。
- 线性函数。 $f(\mathbf{x}) = \langle \mathbf{c}, \mathbf{x} \rangle$, 其中 $\mathbf{c} \in \mathbb{R}^n$
- 二次函数。 $f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x}$, 其中 \mathbf{A} 是半正定矩阵, 向量 $\mathbf{b} \in \mathbb{R}^n$ 。利用后续的判定条件将可以证明这一点。

10.3.3 凸函数的性质

连续性

凸函数不一定是连续函数, 但下面这个定理说明凸函数在定义域中内点处是连续的。

定理 10.3.1. 设 $f: \mathbb{R}^n \rightarrow (-\infty, +\infty]$ 为凸函数。对任意点 $x_0 \in \text{int dom } f$, 有 f 在点 x_0 处连续。这里 $\text{int dom } f$ 表示定义域 $\text{dom } f$ 的内点。

定理 10.3.1 表明凸函数“差不多”是连续的, 它的一个直接推论为:

推论 10.3.1. 设 $f(x)$ 是凸函数, 且 $\text{dom } f$ 是开集, 则 $f(x)$ 在 $\text{dom } f$ 上是连续的。

证明。由于开集中所有的点都为内点, 利用定理 10.3.1 可直接得到结论。 \square

凸函数在定义域的边界上可能不连续. 一个例子为:

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x = 0 \end{cases}$$

其中 $\text{dom } f = (-\infty, 0]$. 容易证明 $f(x)$ 是凸函数, 但其在点 $x = 0$ 处不连续.

凸下水平集

下水平集是描述实值函数取值情况的一个重要概念. 为此有如下定义:

定义 10.3.5. (α -下水平集) 对于广义实值函数 $f : \mathbb{R}^n \rightarrow \mathbb{R}$,

$$C_\alpha = \{x \mid f(x) \leq \alpha\}$$

称为 f 的 α -下水平集.

凸函数的所有下水平集都为凸集, 即有如下结果:

定理 10.3.2. 设 $f(x)$ 是凸函数, 则 $f(x)$ 所有的 α -下水平集 C_α 为凸集.

证明. 任取 $x_1, x_2 \in C_\alpha$, 对任意的 $\theta \in (0, 1)$, 根据 $f(x)$ 的凸性我们有

$$\begin{aligned} f(\theta x_1 + (1 - \theta)x_2) &\leq \theta f(x_1) + (1 - \theta)f(x_2) \\ &\leq \theta\alpha + (1 - \theta)\alpha = \alpha \end{aligned}$$

这说明 C_α 是凸集. □

10.3.4 凸函数的判定条件

根据凸函数的几何意义, 可以观察出: 若函数是凸的, 当且仅当其在与其定义域相交的任何直线上都是凸的. 换言之, 函数 f 是凸的, 当且仅当对于任意 $\mathbf{x} \in C$ 和任意向量 \mathbf{v} , 函数 $g(t) = f(\mathbf{x} + t\mathbf{v})$ 是凸的 (其定义域为 $\{t|\mathbf{x} + t\mathbf{v} \in C\}$). 这为我们提供了一个较为简单的判定方法, 即如下判定定理:

定理 10.3.3. $f(\mathbf{x})$ 是凸函数当且仅当对任意的 $\mathbf{x} \in \text{dom } f, \mathbf{v} \in \mathbb{R}^n, g : \mathbb{R} \rightarrow \mathbb{R}$,

$$g(t) = f(\mathbf{x} + t\mathbf{v}), \text{dom } g = \{t|\mathbf{x} + t\mathbf{v} \in \text{dom } f\}$$

是凸函数.

这个结论非常有用, 因为它容许我们通过将函数限制在直线上来判断其是否是凸函数. 下面的例子说明如何利用上述定理判断函数的凸性.

例 10.3.1. $f(\mathbf{X}) = -\ln \det \mathbf{X}$ 是凸函数, 其中 $\text{dom } f = \mathcal{S}_{++}^n$. 任取 $\mathbf{X} > 0$ 以及方向 $\mathbf{V} \in \mathbb{S}^n$, 将 f 限制在直线 $\mathbf{X} + t\mathbf{V}$ (t 满足 $\mathbf{X} + t\mathbf{V} \succ 0$) 上, 考虑函数 $g(t) = -\ln \det(\mathbf{X} + t\mathbf{V})$. 那么

$$\begin{aligned} g(t) &= -\ln \det \mathbf{X} - \ln \det(\mathbf{I} + t\mathbf{X}^{-1/2}\mathbf{V}\mathbf{X}^{-1/2}) \\ &= -\ln \det \mathbf{X} - \sum_{i=1}^n \ln(1 + t\lambda_i), \end{aligned}$$

其中 λ_i 是 $\mathbf{X}^{-1/2}\mathbf{V}\mathbf{X}^{-1/2}$ 的第 i 个特征值. 对每个 $\mathbf{X} > 0$ 以及方向 \mathbf{V} , 易知 g 关于 t 是凸的, 因此 f 是凸的.

如若知道 f 的可微性, 则我们还能更为方便地判断函数的凸性. 注意这时需要对 f 附加平滑条件 (分别为可微性或二次可微性).

一阶条件: f 可微

定理 10.3.4. 对于定义在凸集上的可微函数 f , f 为凸函数当且仅当

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}), \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f. \quad (10.11)$$

该定理说明可微凸函数 f 的图形始终在其任一点处切线 (切平面) 的上方, 图 10.16 描述了上述不等式的几何含义.

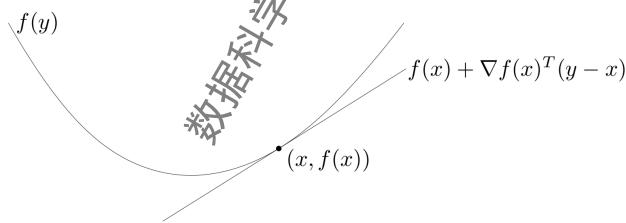


图 10.16: 如果函数 f 是凸的且可微, 那么对于任意 $x, y \in \mathbb{K}$ 有 $f(x) + \nabla f(x)^T(y - x) \leq f(y)$

证明. 假设 f 是凸函数, $\mathbf{x}, \mathbf{y} \in K$. 对任意 $\lambda \in (0, 1)$ 有:

$$(1 - \lambda)f(\mathbf{x}) + \lambda f(\mathbf{y}) \geq f((1 - \lambda)\mathbf{x} + \lambda\mathbf{y}) = f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})).$$

整理得

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \frac{f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})) - f(\mathbf{x})}{\lambda}$$

令 $\lambda \rightarrow 0$, 则 $\frac{f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})) - f(\mathbf{x})}{\lambda} \rightarrow \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$, 则

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle.$$

反之, 假设函数 f 满足式(10.11)。固定 $\mathbf{x}, \mathbf{y} \in K$ 并且 $\lambda \in [0, 1]$ 。令 $\mathbf{z} := \lambda\mathbf{x} + (1 - \lambda)\mathbf{y}$ 是凸集中的某个点, 则有不等式:

$$f(\mathbf{x}) \geq f(\mathbf{z}) + \langle \nabla f(\mathbf{z}), \mathbf{x} - \mathbf{z} \rangle, \quad (10.12)$$

$$f(\mathbf{y}) \geq f(\mathbf{z}) + \langle \nabla f(\mathbf{z}), \mathbf{y} - \mathbf{z} \rangle, \quad (10.13)$$

对(10.12)乘以 λ , 对(10.13)乘以 $1 - \lambda$, 并将两个不等式相加, 有

$$\begin{aligned} (1 - \lambda)f(\mathbf{x}) + \lambda f(\mathbf{y}) &\geq f(\mathbf{z}) + \langle \nabla f(\mathbf{z}), \lambda\mathbf{x} + (1 - \lambda)\mathbf{y} - \mathbf{z} \rangle \\ &= f(\mathbf{z}) + \langle \nabla f(\mathbf{z}), \mathbf{0} \rangle \\ &= f((1 - \lambda)\mathbf{x} + \lambda\mathbf{y}) \end{aligned} \quad (10.14)$$

□

在 \mathbb{R} 上的凸函数, 若具有一阶微分, 则它的一阶导数是单调递增的。实际上, 对于高维空间也有类似的结论。

定理 10.3.5. (梯度单调性) 设 f 为可微函数, 则 f 为凸函数当且仅当 $\text{dom } f$ 为凸集且 ∇f 为单调映射, 即

$$(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) \geq 0, \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f. \quad (10.15)$$

证明. 令 f 是凸函数。根据定理10.3.4, 我们有

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle, \text{ 且 } f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$$

将两不等式相加即得到式(10.15)。

下面假设式(10.15)对任意 \mathbf{x}, \mathbf{y} 成立。取 $\lambda \in [0, 1]$, 令 $\mathbf{x}_\lambda := \mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})$ 。由于 ∇f 是连续的, 我们有

$$\begin{aligned} f(\mathbf{y}) &= f(\mathbf{x}) + \int_0^1 \langle \nabla f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})), \mathbf{y} - \mathbf{x} \rangle d\lambda \\ &= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \langle \nabla f(\mathbf{x}_\lambda) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle d\lambda \\ &= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \frac{1}{\lambda} \langle \nabla f(\mathbf{x}_\lambda) - \nabla f(\mathbf{x}), \mathbf{x}_\lambda - \mathbf{x} \rangle d\lambda \\ &\geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \end{aligned}$$

□

推论 10.3.2. f 是严格凸函数当且仅当

$$(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) > 0, \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f.$$

f 是凹函数当且仅当

$$(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) \leq 0, \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f.$$

二阶条件

进一步地, 如果函数二阶连续可微, 我们也可以得到一些判定的条件。

对于二阶可微的一元函数, 我们可以较容易地得到函数 f 是凸函数当且仅当 $f''(x) \geq 0$ 。对于高维情形, 可以得到如下定理:

定理 10.3.6. 假设定义在开集 K 内的函数 f 二阶可微, 则函数 $f: K \rightarrow \mathbb{R}$ 是凸函数的充要条件是: 其 Hessian 矩阵是半正定阵, 即对于所有的 $\mathbf{x} \in K$, 都有

$$\nabla^2 f(\mathbf{x}) \succeq 0.$$

证明. 假设 $f: K \rightarrow \mathbb{R}$ 是二阶可微的凸函数。对任意 $\mathbf{x} \in K$, 并且对任意 $\mathbf{s} \in \mathbb{R}^n$, 由于 K 是开集, 存在 $\tau > 0$ 使得 $\mathbf{x}_\tau := \mathbf{x} + \tau \mathbf{s} \in K$ 。根据梯度单调性, 有

$$\begin{aligned} 0 &\leq \frac{1}{\tau^2} \langle \nabla f(\mathbf{x}_\tau) - \nabla f(\mathbf{x}), \mathbf{x}_\tau - \mathbf{x} \rangle \\ &= \frac{1}{\tau} \langle \nabla f(\mathbf{x}_\tau) - \nabla f(\mathbf{x}), \mathbf{s} \rangle \\ &= \frac{1}{\tau} \int_0^\tau \langle \nabla^2 f(\mathbf{x} + \lambda \mathbf{s}) \mathbf{s}, \mathbf{s} \rangle d\lambda, \end{aligned}$$

最后令 $\tau \rightarrow 0$ 即得到结论。

反之, 假设 $\forall \mathbf{x} \in K$, $\nabla^2 f(\mathbf{x}) \succeq 0$ 。对任意 $\mathbf{x}, \mathbf{y} \in K$, 有

$$\begin{aligned} f(\mathbf{y}) &= f(\mathbf{x}) + \int_0^1 \langle \nabla f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})), \mathbf{y} - \mathbf{x} \rangle d\lambda \\ &= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \langle \nabla f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle d\lambda \\ &= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \int_0^1 (\mathbf{y} - \mathbf{x})^T \nabla^2 f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) (\mathbf{y} - \mathbf{x}) d\tau d\lambda \\ &\geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \end{aligned}$$

最后一步利用了 $\nabla^2 f(\mathbf{x})$ 是半正定的。

□

例 10.3.2. 常见二次函数的凸性:

(1) 考虑二次函数 $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r$ ($\mathbf{P} \in S^n$), 容易计算出其梯度与海瑟矩阵分别为

$$\nabla f(\mathbf{x}) = \mathbf{P} \mathbf{x} + \mathbf{q}, \nabla^2 f(\mathbf{x}) = \mathbf{P}.$$

那么, f 是凸函数当且仅当 $\mathbf{P} \succeq 0$ 。

(2) 考虑最小二乘函数 $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{A} \mathbf{x} - \mathbf{b}\|_2^2$, 其梯度与海瑟矩阵分别为

$$\nabla f(\mathbf{x}) = \mathbf{A}^T (\mathbf{A} \mathbf{x} - \mathbf{b}), \nabla^2 f(\mathbf{x}) = \mathbf{A}^T \mathbf{A}.$$

注意到 $\mathbf{A}^T \mathbf{A}$ 恒为半正定矩阵, 因此对任意 \mathbf{A} , f 都是凸函数。

对于严格凸函数, 它的条件可能相对苛刻一些。一般地, 可微函数 f 是严格凸函数的充要条件是 C 是凸集, 且对于任意 $\mathbf{x} \neq \mathbf{y} \in C$, 都有

$$f(\mathbf{y}) > f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \quad (10.16)$$

成立。若函数二次可微且

$$\nabla^2 f(\mathbf{x}) \succ 0, \quad \forall \mathbf{x} \in C,$$

那么 f 是严格凸的。

例 10.3.3. 仍然考虑二次函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$, 其定义域为 $\mathbb{K} = \mathbb{R}^n$, 其表达式为

$$f(\mathbf{x}) = (1/2)\mathbf{x}^T P \mathbf{x} + \mathbf{q}^T \mathbf{x} + r$$

其中 $P \in S^n, \mathbf{q} \in \mathbb{R}^n, r \in \mathbb{R}$ 。对于任意 \mathbf{x} , $\nabla^2 f(\mathbf{x}) = P$ 。因此, 函数 f 是严格凸的, 当且仅当 $P \succ 0$ (函数是严格凹的当且仅当 $P \prec 0$)。

类似地, 函数 f 是凹函数的充要条件是, \mathbb{K} 是凸集且对于任意 $\mathbf{x} \in \mathbb{K}$, $\nabla^2 f(\mathbf{x}) \leq 0$ 。严格凸的条件可以部分由二阶条件刻画。如果对于任意的 $\mathbf{x} \in \mathbb{K}$ 有 $\nabla^2 f(\mathbf{x}) > 0$, 则函数 f 是严格凸。反过来则不一定成立: 例如, 函数 $f: \mathbb{R} \rightarrow \mathbb{R}$ 其表达式为 $f(x) = x^4$, 它是严格凸的, 但是在 $x = 0$ 处, 二阶导数为零。

为了方便对严格凸性定量描述, 人们常引入强凸性的概念。

定义 10.3.6. (强凸函数) 若存在常数 $\sigma > 0$, 使得

$$g(\mathbf{x}) = f(\mathbf{x}) - \frac{\sigma}{2} \|\mathbf{x}\|^2$$

为凸函数, 则称 $f(x)$ 为强凸函数, 其中 σ 为强凸参数。为了方便我们也称 $f(x)$ 为 σ -强凸函数。

通过直接对 $g(x) = f(x) - \frac{\sigma}{2} \|\mathbf{x}\|^2$ 应用凸函数的定义, 我们可得到另一个常用的强凸函数定义。

定义 10.3.7. (强凸函数的等价定义) 若存在常数 $\sigma > 0$, 使得对任意 $x, y \in \text{dom } f$ 以及 $\theta \in (0, 1)$, 有

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y) - \frac{\sigma}{2} \theta(1 - \theta) \|\mathbf{x} - \mathbf{y}\|^2,$$

则称 $f(x)$ 为强凸函数, 其中 σ 为强凸参数。

强凸函数具有二次下界的性质

定理 10.3.7. (二次下界) 定义在凸集 K 上的可微函数 $f: K \rightarrow \mathbb{R}$ 被称为关于范数 $\|\cdot\|$ 是 σ -强凸的, 则如下不等式成立

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\sigma}{2} \|\mathbf{y} - \mathbf{x}\|^2.$$

证明. 由强凸函数的定义, $g(\mathbf{x}) = f(\mathbf{x}) - \frac{\sigma}{2} \|\mathbf{x}\|^2$ 是凸函数, 根据凸函数的一阶条件可知

$$g(\mathbf{y}) \geq g(\mathbf{x}) + \nabla g(\mathbf{x})^T (\mathbf{y} - \mathbf{x}),$$

即

$$f(\mathbf{y}) \geq f(\mathbf{x}) - \frac{\sigma}{2} \|\mathbf{x}\|^2 + \frac{\sigma}{2} \|\mathbf{y}\|^2 + (\nabla f(\mathbf{x}) - \sigma \mathbf{x})^T (\mathbf{y} - \mathbf{x}) \quad (10.17)$$

$$= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\sigma}{2} \|\mathbf{y} - \mathbf{x}\|^2. \quad (10.18)$$

□

注意, 强凸函数也是严格凸的, 但反之不一定成立。如果 f 是二阶连续可微的, 并且 $\|\cdot\| = \|\cdot\|_2$ 是 l_2 -范数, 则强凸性蕴含

$$\nabla^2 f(\mathbf{x}) \geq \sigma \mathbf{I}$$

对任意 $\mathbf{x} \in K$ 。强凸性表示, 函数

$$f(\mathbf{y}) = f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \quad (10.19)$$

存在一个二次函数 $\frac{\sigma}{2} \|\mathbf{y} - \mathbf{x}\|^2$ 作为下界。式(10.19)通常被称为 Bregman 散度。

定义 10.3.8. (Bregman 散度) 函数 $f: K \rightarrow \mathbb{R}$ 在 $u, w \in K$ 两点间的 Bregman 散度定义为:

$$D_f(u, w) := f(w) - (f(u) + \langle \nabla f(u), w - u \rangle).$$

一般说来, Bregman 散度关于 u 和 w 不是对称的, 即 $D_f(u, w)$ 一般和 $D_f(w, u)$ 不相等。从定义可以看出, 强凸函数减去一个正定二次函数仍然是凸的; 强凸函数一定是严格凸函数, 当 $m = 0$ 时退化成凸函数。无论从哪个定义出发, 容易看出和凸函数相比, 强凸函数有更好的性质。

定理 10.3.8. 设 f 为强凸函数且存在最小值, 则 f 的最小值点唯一。

证明. 采用反证法. 设 $x \neq y$ 均为 f 的最小值点, 根据强凸函数的等价定义, 取 $\theta \in (0, 1)$, 则有

$$\begin{aligned} f(\theta x + (1 - \theta)y) &\leq \theta f(x) + (1 - \theta)f(y) - \frac{\sigma}{2}\theta(1 - \theta)\|x - y\|^2 \\ &= f(x) - \frac{\sigma}{2}\theta(1 - \theta)\|x - y\|^2 \\ &< f(x), \end{aligned}$$

其中严格不等号成立是因为 $x \neq y$. 这显然和 $f(x)$ 为最小值矛盾, 得证. □

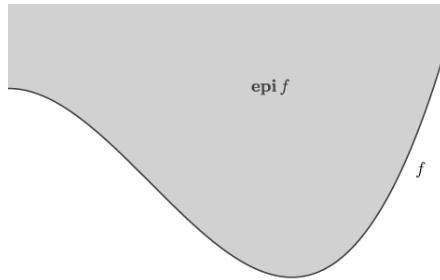
上镜图与凸函数判定

上镜图是从集合的角度来描述一个函数的具体性质. 我们有上镜图定义如下:

定义 10.3.9. (上镜图) 对于广义实值函数 $f: \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$,

$$\text{epi } f = \{(x, t) \in \mathbb{R}^{n+1} \mid f(x) \leq t\}$$

称为 f 的上镜图。

图 10.17: 函数 f 和其上镜图 $\text{epi } f$

上镜图将函数和集合建立了联系, f 的很多性质都可以在 $\text{epi } f$ 上得到体现. 实际上, 可以使用上方图 $\text{epi } f$ 来判断 f 的凸性. 我们有如下定理:

定理 10.3.9. 函数 $f(x)$ 为凸函数当且仅当其上方图 $\text{epi } f$ 是凸集.

10.3.5 保凸运算

前面已经提及判定函数凸性的方法, 包括从定义出发; 多维转化为一维函数判定; 利用一阶或二阶导数的信息来判断凸性等等. 事实上, 函数的凸性可以由简单的凸函数通过一些保凸的运算得到. 了解这些运算, 为复杂函数凸性的判定提供了极大的便利, 包括: 非负加权求和、复合仿射映射、逐点最大和逐点上(下)确界、最小化、透视函数等.

非负加权求和与复合仿射映射

显而易见, 如果函数 f 是凸函数且 $\alpha \geq 0$, 则函数 αf 也是凸函数, 如果函数 f_1 和 f_2 都是凸函数, 则他们的和 $f_1 + f_2$ 也是凸函数. 将非负伸缩以及求和运算结合起来, 可以看出, 凸函数的非负加权求和仍然是凸函数, 即如果 $f_i, i = 1, \dots, m$ 是凸函数, $w_i \geq 0, i = 1, \dots, m$, 那么, 函数

$$f = w_1 f_1 + \dots + w_m f_m$$

也是凸函数. 类似地, 凹函数的非负加权求和仍然是凹函数. 严格凸(凹)函数的非负、非零加权求和是严格凸(凹)函数.

这个性质可以扩展至无限项的求和以及积分的情形. 例如, 如果固定任意 $y \in \mathcal{A}$, 函数 $f(x, y)$ 关于 x 是凸函数, 且对任意 $y \in \mathcal{A}$, 有 $w(y) \geq 0$, 则函数

$$g(x) = \int_{\mathcal{A}} w(y) f(x, y) dy$$

关于 x 是凸函数(若此积分存在).

另外, 若 f 是凸函数, 则复合放射映射函数 $f(\mathbf{A}\mathbf{x} + \mathbf{b})$ 也是凸函数。实际上, 对于复合函数的保凸性, 在下面将会给出一般性的讨论。

例 10.3.4. 利用与仿射函数的复合函数保凸, 可以证明:

- 线性不等式的对数障碍函数:

$$f(\mathbf{x}) = -\sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \quad \text{dom } f = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} \leq b_i, i = 1, \dots, m\}$$

是凸函数。

- 仿射函数的(任意)范数: $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} + \mathbf{b}\|$ 都是凸函数。

逐点最大和逐点确界函数

逐点最大函数 如果函数 f_1 和 f_2 均为凸函数, 则二者的逐点最大函数

$$f(\mathbf{x}) = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$$

仍然是凸函数, 定义域为 $\mathbb{K} = \mathbb{K}_1 \cap \mathbb{K}_2$ 。这个性质可以很容易验证: 任取 $0 \leq \theta \leq 1$ 以及 $\mathbf{x}, \mathbf{y} \in \mathbb{K}$, 有

$$\begin{aligned} f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) &= \max\{f_1(\theta\mathbf{x} + (1 - \theta)\mathbf{y}), f_2(\theta\mathbf{x} + (1 - \theta)\mathbf{y})\} \\ &\leq \max\{\theta f_1(\mathbf{x}) + (1 - \theta)f_1(\mathbf{y}), \theta f_2(\mathbf{x}) + (1 - \theta)f_2(\mathbf{y})\} \\ &\leq \theta \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\} + (1 - \theta) \max\{f_1(\mathbf{y}), f_2(\mathbf{y})\} \\ &= \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}) \end{aligned}$$

从而说明了函数 f 的凸性。同样地, 如果函数 f_1, \dots, f_m 为凸函数, 则它们的逐点最大函数

$$f(\mathbf{x}) = \max\{f_1(\mathbf{x}), \dots, f_m(\mathbf{x})\}$$

仍然是凸函数。

例 10.3.5. 分片线性函数. 函数

$$f(\mathbf{x}) = \max\{\mathbf{a}_1^T \mathbf{x} + b_1, \dots, \mathbf{a}_L^T \mathbf{x} + b_L\}$$

定义了一个分片线性(实际上是仿射)函数(具有 L 个或者更少的子区域)。因为它是一系列仿射函数的逐点最大函数, 所以它是凸函数。

反之亦成立: 任意具有 L 个或者更少子区域的分片线性凸函数都可以表述成上述形式。

例 10.3.6. $\mathbf{x} \in \mathbb{R}^n$ 最大的 r 个分量之和:

$$f(\mathbf{x}) = x_{[1]} + x_{[2]} + \dots + x_{[r]}$$

是凸函数($x_{[i]}$ 是 \mathbf{x} 的从大到小排列的第 i 个分量)。可通过改造函数形式为

$$f(\mathbf{x}) = \max\{x_{i_1} + x_{i_2} + \dots + x_{i_r} | 1 \leq i_1 < i_2 < \dots < i_r \leq n\}$$

去证明原函数的凸性。

逐点上确界函数 逐点最大的性质可以扩展至无限个凸函数的逐点上确界。如果对于任意 $y \in \mathcal{A}$, 函数 $f(x, y)$ 关于 x 都是凸函数, 则

$$g(x) = \sup_{y \in \mathcal{A}} f(x, y)$$

关于 x 亦是凸函数。此时, 函数 g 的定义域为

$$\mathbb{K} = \{x | (x, y) \in \mathbb{K}, \forall y \in \mathcal{A}, \sup_{y \in \mathcal{A}} f(x, y) < \infty\}$$

类似地, 一系列凹函数的逐点下确界仍然是凹函数。

例 10.3.7. 集合的支撑函数。令集合 $C \subseteq \mathbb{R}^n$, 且 $C \neq \emptyset$, 定义集合 C 的支撑函数 S_C 为

$$S_C(x) = \sup\{x^T y | y \in C\}$$

其定义域为 $\mathbb{K} = \{x | \sup_{y \in C} x^T y < \infty\}$ 。对于任意 $y \in C$, $x^T y$ 是 x 的线性函数, 所以 S_C 是一系列线性函数的逐点上确界函数, 因此是凸函数。

例 10.3.8. 矩阵范数。考虑函数 $f(X) = \|X\|_2$, 定义域为 $\mathbb{K} = \mathbb{R}^{p \times q}$, 其中, $\|\cdot\|_2$ 表示谱函数或者最大奇异值。函数 f 则可以重新写为

$$f(X) = \sup\{\mathbf{u}^T X \mathbf{v} | \|\mathbf{u}\|_2 = 1, \|\mathbf{v}\|_2 = 1\}$$

由于它是 X 的一族线性函数的逐点上确界, 所以是凸函数。

上述例子表明, 一个建立函数凸性的好方法是将其表示为一族仿射函数的逐点上确界。几乎所有的凸函数都可以表示成一族仿射函数的逐点上确界。

定理 10.3.10. 如果函数 $f : K \rightarrow \mathbb{R}$ 是凸函数, 其定义域为 $K = \mathbb{R}^n$, 我们有

$$f(x) = \sup\{g(x) | g \text{ 是仿射函数}, g(z) \leq f(z), \forall z\}$$

换言之, 函数 f 是它所有的仿射全局下估计的逐点上确界。

证明. 设函数 f 是凸函数, 定义域为 $K = \mathbb{R}^n$, 显然下面的不等式成立

$$f(x) \geq \sup\{g(x) | g \text{ 是仿射函数}, g(z) \leq f(z), \forall z\}$$

因为函数 g 是函数 f 的任意仿射下估计, 我们有 $g(x) \leq f(x)$ 。为了建立等式, 我们说明, 对任意 $x \in \mathbb{R}^n$, 存在仿射函数 g 是函数 f 的全局下估计, 并且满足 $g(x) = f(x)$ 。

毫无疑问, 函数 f 的上境图是凸集, 因此我们在点 $(x, f(x))$ 处可以找到此凸集的支撑超平面, 即存在 $a \in \mathbb{R}^n, b \in \mathbb{R}$ 且 $(a, b) \neq 0$, 使得对任意 $(z, t) \in \text{epi } f$, 有

$$\begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} x - z \\ f(x) - t \end{bmatrix} \leq 0$$

由于 $(z, t) \in \text{epi } f$ 等价于存在 $s \geq 0$, 使得 $t = f(z) + s$ 。因此, 对任意 $z \in \mathbb{K} = \mathbb{R}^n$ 以及所有 $s \geq 0$, 都有

$$a^T(x - z) + b(f(x) - f(z) - s) \leq 0 \tag{10.20}$$

为了保证不等式(10.20)对所有的 $s \geq 0$ 均成立, 必须要 $b \geq 0$ 。如果 $b = 0$, 对所有的 $z \in \mathbb{R}^n$, 不等式(10.20)可以简化为 $a^T(x - z) \leq 0$ 。这意味着 $a = \mathbf{0}$, 于是和假设 $(a, b) \neq \mathbf{0}$ 矛盾。因此, $b > 0$, 即支撑超平面不是竖直的。

在 $b > 0$ 的情况下, 对任意 z , 令 $s = 0$, 式(10.20)可以重新表述为

$$g(z) = f(x) + (a/b)^T(x - z) \leq f(z)$$

由此说明函数 g 是函数 f 的一个仿射下估计, 并且满足 $g(x) = f(x)$ 。 \square

逐点下确界函数 一些特殊形式的最小化同样可以得到凸函数。如果函数 f 关于 (x, y) 是凸函数, 集合 C 是非空凸集, 定义函数

$$g(x) = \inf_{y \in C} f(x, y) \quad (10.21)$$

如果对任意的 x , 都有 $g(x) > -\infty$, 那么, 函数 g 关于 x 是凸函数。此时, 函数 g 的定义域是 C 在 x 方向上的投影, 即

$$\mathbf{dom} g = \{x \mid \exists y \in C, s.t., (x, y) \in \mathbb{K}\}$$

可以利用 Jensen 不等式来证明函数 g 的凸性。

证明. 任取 $x_1, x_2 \in \mathbf{dom} g$, 令 $\varepsilon > 0$, 则存在 $y_1, y_2 \in C$, 使 $f(x_i, y_i) \leq g(x_i) + \varepsilon (i = 1, 2)$ 。设 $\theta \in [0, 1]$ 。我们有

$$\begin{aligned} g(\theta x_1 + (1 - \theta)x_2) &= \inf_{y \in C} f(\theta x_1 + (1 - \theta)x_2, y) \\ &\leq f(\theta x_1 + (1 - \theta)x_2, \theta y_1 + (1 - \theta)y_2) \\ &\stackrel{\text{数理基础}}{\leq} \theta f(x_1, y_1) + (1 - \theta)f(x_2, y_2) \\ &\leq \theta g(x_1) + (1 - \theta)g(x_2) + \varepsilon \end{aligned}$$

因为上式对任意 $\varepsilon > 0$ 均成立, 所以不等式

$$g(\theta x_1 + (1 - \theta)x_2) \leq \theta g(x_1) + (1 - \theta)g(x_2) \quad (10.22)$$

成立。结论得证。 \square

例 10.3.9. 点到某一集合的距离。某点 x 到集合 $S \subseteq \mathbb{R}^n$ 的距离定义为

$$\text{dist}(x, S) = \inf_{y \in S} \|x - y\|$$

函数 $\|x - y\|$ 关于 (x, y) 是凸的。所以, 若集合 S 是凸集, 则 $\text{dist}(x, S)$ 是关于 x 的凸函数。

复合函数

本节给定函数 $h: \mathbb{R}^k \rightarrow \mathbb{R}$ 以及 $g: \mathbb{R}^n \rightarrow \mathbb{R}^k$, 定义复合函数 $f = h \circ g: \mathbb{R}^n \rightarrow \mathbb{R}$ 为

$$f(x) = h(g(x)), \quad \mathbf{dom} f = \{x \in \mathbf{dom} g \mid g(x) \in \mathbf{dom} h\}$$

我们考虑当函数 f 保凸或者保凹时, 函数 h 和 g 必须满足的条件。

标量复合

考虑 $k = 1$ 的情况, 即 $h: \mathbb{R} \rightarrow \mathbb{R}, g: \mathbb{R}^n \rightarrow \mathbb{R}$ 。

当 $n = 1$ 时, (事实上, 将函数限定在与其定义域相交的任意直线上得到的函数决定了原函数的凸性) 为了找出复合规律, 假设函数 h 和 g 是二次可微的, 且 $\text{dom } g = \text{dom } h = \mathbb{R}$ 。根据二阶判定条件, 在上述假设下, 函数 f 是凸的等价于 $f'' \geq 0$ (即对所有的 $x \in \mathbb{R}, f''(x) \geq 0$)。计算可得复合函数 $f = h \circ g$ 的二阶导数为

$$f''(x) = h''(g(x))g'(x)^2 + h'(g(x))g''(x). \quad (10.23)$$

易知, 若函数 g 是凸函数 ($g'' \geq 0$), 函数 h 是凸函数且非减(即 $h'' \geq 0$ 且 $h' \geq 0$), 从式(10.23)可以得出 $f'' \geq 0$, 即函数 f 是凸函数。类似地, 由式(10.23)可以得出如下结论

$$\begin{aligned} &\text{如果 } h \text{ 是凸函数且非减, } g \text{ 是凸函数, 则 } f \text{ 是凸函数} \\ &\text{如果 } h \text{ 是凸函数且非增, } g \text{ 是凹函数, 则 } f \text{ 是凸函数} \\ &\text{如果 } h \text{ 是凹函数且非减, } g \text{ 是凹函数, 则 } f \text{ 是凹函数} \\ &\text{如果 } h \text{ 是凹函数且非增, } g \text{ 是凸函数, 则 } f \text{ 是凹函数} \end{aligned} \quad (10.24)$$

当 $n > 1$ 时, 即 $\text{dom } g = \mathbb{R}^n, \text{dom } h = \mathbb{R}$, 此时, 不再假设函数 h 和 g 可微, 相似的复合规则仍然成立:

$$\begin{aligned} &\text{如果 } h \text{ 是凸函数且 } \tilde{h} \text{ 非减, } g \text{ 是凸函数, 则 } f \text{ 是凸函数} \\ &\text{如果 } h \text{ 是凸函数且 } \tilde{h} \text{ 非增, } g \text{ 是凹函数, 则 } f \text{ 是凸函数} \\ &\text{如果 } h \text{ 是凹函数且 } \tilde{h} \text{ 非减, } g \text{ 是凹函数, 则 } f \text{ 是凹函数} \\ &\text{如果 } h \text{ 是凹函数且 } \tilde{h} \text{ 非增, } g \text{ 是凸函数, 则 } f \text{ 是凹函数} \end{aligned} \quad (10.25)$$

其中 \tilde{h} 是 h 的扩展函数。这些结论和(10.24)不同之处在于要求扩展函数 \tilde{h} 在整个 \mathbb{R} 上非增或者非减。

下面开始证明其中一种情形: 如果 h 是凸函数且 \tilde{h} 非减, g 是凸函数, 则 $f = h \circ g$ 是凸函数。(10.25)中的其它结论可以类似得到证明。

证明. 假设 $\mathbf{x}, \mathbf{y} \in \mathbb{K}, 0 \leq \theta \leq 1$ 。由于 $\mathbf{x}, \mathbf{y} \in \mathbb{K}$, 我们有 $\mathbf{x}, \mathbf{y} \in \text{dom } g$, 且 $g(\mathbf{x}), g(\mathbf{y}) \in \text{dom } h$ 。因为 $\text{dom } g$ 是凸集, 则有 $\theta\mathbf{x} + (1 - \theta)\mathbf{y} \in \text{dom } g$ 。由函数 g 的凸性可得

$$g(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \leq \theta g(\mathbf{x}) + (1 - \theta)g(\mathbf{y}) \quad (10.26)$$

由 $g(\mathbf{x}), g(\mathbf{y}) \in \text{dom } h$ 可得 $\theta g(\mathbf{x}) + (1 - \theta)g(\mathbf{y}) \in \text{dom } h$ 。即式(10.26)的右端在 $\text{dom } h$ 内。根据假设 \tilde{h} 是非减的, 可以理解为其定义域在负方向上无限延伸。式(10.26)的右端在 $\text{dom } h$ 内, 我们知道其左侧仍在定义域内, 即 $g(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \in \text{dom } h$, 因此 \mathbb{K} 是凸集。

根据前提条件, \tilde{h} 非减, 利用不等式(10.26), 我们有

$$h(g(\theta\mathbf{x} + (1 - \theta)\mathbf{y})) \leq h(\theta g(\mathbf{x}) + (1 - \theta)g(\mathbf{y})) \quad (10.27)$$

由函数 h 的凸性, 可得

$$h(\theta g(\mathbf{x}) + (1 - \theta)g(\mathbf{y})) \leq \theta h(g(\mathbf{x})) + (1 - \theta)h(g(\mathbf{y})) \quad (10.28)$$

综合式(10.27)和式(10.28), 可得

$$h(g(\theta \mathbf{x} + (1 - \theta)\mathbf{y})) \leq \theta h(g(\mathbf{x})) + (1 - \theta)h(g(\mathbf{y}))$$

结论得证。 \square

例 10.3.10. 下面列出几个标量函数复合的例子:

- 若 g 是凸函数, 则 $\exp g(\mathbf{x})$ 是凸函数;
- 若 g 是正值凹函数, 则 $1/g(\mathbf{x})$ 是凸函数;
- 若 g_i 是正值凹函数, 则 $\sum_{i=1}^m \ln(g_i(\mathbf{x}))$ 是凹函数;

向量复合

考虑 $k > 1$ 的情况, $f(\mathbf{x}) = h(g(\mathbf{x})) = h(g_1(\mathbf{x}), \dots, g_k(\mathbf{x}))$, 其中, $h: \mathbb{R}^k \rightarrow \mathbb{R}$, $g_i: \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, k$ 。

和上节一样, 首先假设 $n = 1$ 。和 $k = 1$ 的情形类似, 为了得到复合规则, 假设函数二次可微, 且 $\text{dom } g_i = \mathbb{R}$, $\text{dom } h = \mathbb{R}^k$ 。对函数 f 进行二次微分, 可得

$$f''(\mathbf{x}) = \mathbf{g}'(\mathbf{x})^T \nabla^2 h(\mathbf{g}(\mathbf{x})) \mathbf{g}'(\mathbf{x}) + \nabla h(\mathbf{g}(\mathbf{x}))^T \mathbf{g}''(\mathbf{x}) \quad (10.29)$$

上式可以看成是式(10.23)对应的向量形式。此时, 需要判断在什么条件下对所有 \mathbf{x} , 有 $f''(\mathbf{x}) \geq 0$ (或者对所有 \mathbf{x} , 有 $f''(\mathbf{x}) \leq 0$, 则 f 是凹函数)。利用式(10.29), 得到如下复合规则:

如果 h 是凸函数且在每维分量上 h 非减, g_i 是凸函数, 则 f 是凸函数;

如果 h 是凸函数且在每维分量上 h 非增, g_i 是凹函数, 则 f 是凸函数;

如果 h 是凹函数且在每维分量上 h 非减, g_i 是凹函数, 则 f 是凹函数。

如果 h 是凹函数且在每维分量上 h 非增, g_i 是凸函数, 则 f 是凹函数。

然后考虑 $n > 1$ 的情况, 类似的复合结论仍然成立。

例 10.3.11. 函数 $h(\mathbf{z}) = \log(\sum_{i=1}^k e^{z_i})$ 是凸函数且在每一维分量上非减, 因此只要 g_i 是凸函数, $\log(\sum_{i=1}^k e^{g_i})$ 就是凸函数。

例 10.3.12. 点 \mathbf{x} 到凸集 S 的距离: $\text{dist}(\mathbf{x}, S) = \inf_{\mathbf{y} \in S} \|\mathbf{x} - \mathbf{y}\|$ 是凸函数。

透视函数

给定函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$, 则 f 的透视函数 $g: \mathbb{R}^{n+1} \rightarrow \mathbb{R}$ 定义为

$$g(\mathbf{x}, t) = t f(\mathbf{x}/t),$$

其定义域为

$$\mathbf{dom} g = \{(\mathbf{x}, t) | \mathbf{x}/t \in \mathbb{K}, t > 0\}$$

透视运算是保凸运算: 如果函数 f 是凸函数, 则其透视函数 g 也是凸函数。类似地, 若 f 是凹函数, 则 g 亦是凹函数。

可以从多个角度来证明此结论, 从上境图的角度, 当 $t > 0$, 我们有

$$\begin{aligned} (\mathbf{x}, t, s) \in \mathbf{epi} g &\iff t f(\mathbf{x}/t) \leq s \\ &\iff f(\mathbf{x}/t) \leq s/t \\ &\iff (\mathbf{x}/t, s/t) \in \mathbf{epi} f \end{aligned}$$

因此, $\mathbf{epi} g$ 是透视映射下 $\mathbf{epi} f$ 的原像, 此透视映射将 (u, v, w) 映射为 $(u, w)/v$ 。因此, $\mathbf{epi} g$ 是凸集, g 是凸函数。

例 10.3.13. Euclid 范数平方。 \mathbb{R}^n 上的凸函数 $f(\mathbf{x}) = \mathbf{x}^T \mathbf{x}$ 的透视函数定义为

$$g(\mathbf{x}, t) = t(\mathbf{x}/t)^T (\mathbf{x}/t) = \frac{\mathbf{x}^T \mathbf{x}}{t}$$

当 $t > 0$ 时, 它关于 (\mathbf{x}, t) 是凸函数。

例 10.3.14. 负对数。 考虑 \mathbb{R}_{++} 上的凸函数 $f(x) = -\log x$, 其透视函数为

$$g(x, t) = -t \log(x/t) = t \log(t/x) = t \log t - t \log x$$

在 \mathbb{R}_{++}^2 上它是凸函数。函数 g 被称为关于 t 和 x 的相对熵。当 $\mathbf{x} = 1$ 时, g 为负熵函数。根据函数 g 的凸性, 可以推导出其它函数的凸性。例如: 定义两个向量 $u, v \in \mathbb{R}_{++}^n$ 的相对熵

$$\sum_{i=1}^n u_i \log(u_i/v_i)$$

因为它可以转化为 (u, v) 的相对熵和线性函数的求和, 所以是凸函数。

例 10.3.15. 设 $f: \mathbb{R}^m \rightarrow \mathbb{R}$ 是凸函数, $A \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m, \mathbf{c} \in \mathbb{R}^n, d \in \mathbb{R}$ 。定义

$$g(\mathbf{x}) = (\mathbf{c}^T \mathbf{x} + d) f((A\mathbf{x} + \mathbf{b})/(\mathbf{c}^T \mathbf{x} + d))$$

其定义域为

$$\mathbf{dom} g = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0, (A\mathbf{x} + \mathbf{b})/(\mathbf{c}^T \mathbf{x} + d) \in \mathbb{K}\}$$

则 g 是凸函数。

10.3.6 共轭函数

本节介绍一种函数运算称作共轭函数，它将在后续优化问题的对偶理论中发挥重要的作用。

定义

定义 10.3.10. 任一适当函数 f 的共轭函数定义为：

$$f^*(y) = \sup_{x \in \text{dom } f} (y^T x - f(x)).$$

图 10.18 描述了共轭函数的几何意义。与 f 的函数性质无关， f^* 始终是凸函数。这是因为

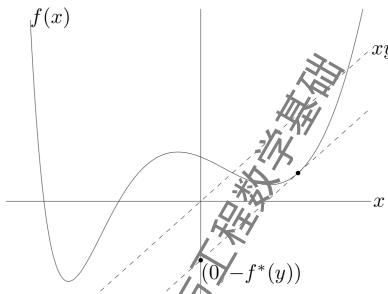


图 10.18：函数 $f: \mathbb{R} \rightarrow \mathbb{R}$ 以及某一 $y \in \mathbb{R}$ ，共轭函数 $f^*(y)$ 是线性函数 yx 和 $f(x)$ 之间的最大差值，见图中虚线所示。如果 f 可微，在满足 $f'(x) = y$ 的点 x 处差值最大。

若 f^* 是凸函数，它是一系列关于 y 的凸函数（实质上是仿射函数）的逐点上确界。

\mathbb{R} 上一些常见凸函数的共轭函数

下面我们从一些简单的例子开始，通过求解常见函数的共轭函数，获得共轭函数的一些直观理解。

- 仿射函数

$f(x) = ax + b$ 。作为 x 的函数，当且仅当 $y = a$ ，即为常数时， $yx - ax - b$ 有界。因此，共轭函数 f^* 的定义域为单点集 a ，且 $f^*(a) = -b$ 。

- 负对数函数

$f(x) = -\log x$ ，定义域为 $\mathbb{K} = \mathbb{R}_{++}$ 。当 $y \geq 0$ 时，函数 $xy + \log x$ 无上界，当 $y < 0$ 时，在 $x = -\frac{1}{y}$ 处函数达到最大。因此，定义域为 $\mathbb{K}^* = \{y | y < 0\} = -\mathbb{R}_{++}$ ，共轭函数为 $f^*(y) = -\log(-y) - 1 (y < 0)$ 。

- 指数函数

$f(x) = e^x$ 当 $y < 0$ 时, 函数 $xy - e^x$ 无界。当 $y > 0$ 时, 函数 $xy - e^x$ 在 $x = \log y$ 处达到最大值。因此, $f^*(y) = y \log y - y$ 。当 $y = 0$ 时, $f^*(y) = \sup_x -e^x = 0$ 。综合起来, $\mathbb{K}^* = \mathbb{R}_+$, $f^*(y) = y \log y - y$ (规定 $0 \log 0 = 0$)。

- 负熵函数

$f(x) = x \log x$, 定义域为 $\mathbb{K} = \mathbb{R}_+$ 。对所有 y , 函数 $xy - x \log x$ 关于 x 在 \mathbb{R}_+ 上有界, 因此 $\mathbb{K}^* = \mathbb{R}$ 。在 $x = e^{y-1}$ 处, 函数达到最大值。因此 $f^*(y) = e^{y-1}$ 。

- 反函数

$f(x) = \frac{1}{x}, x \in \mathbb{R}_{++}$ 。当 $y > 0$ 时, $yx - 1/x$ 无上界。当 $y = 0$ 时, 函数有上确界 0; 当 $y < 0$ 时, 在 $x = (-y)^{-1/2}$ 处达到上确界。因此, $f^*(y) = -2(-y)^{1/2}$ 且 $\mathbb{K}^* = -\mathbb{R}_+$ 。

\mathbb{R}^n 上一些常见凸函数的共轭函数

- 严格凸的二次函数

考虑函数 $f(x) = \frac{1}{2}x^T Qx$, $Q \in S_{++}^n$ 。对于所有的 y , x 的函数 $y^T x - \frac{1}{2}x^T Qx$ 都有上界并在 $x = Q^{-1}y$ 处达到上确界, 因此, $f^*(y) = y^T Q^{-1}y$ 。

- 指示函数

设 I_S 是某个集合 $S \subseteq \mathbb{R}^n$ (不一定是凸集) 的示性函数, 即当 x 在 $\text{dom } I_S = S$ 内时, $I_S(x) = 0$ 。示性函数的共轭函数为 $I_S^*(y) = \sup_{x \in S} y^T x$, 它是集合 S 的支撑函数。

- 范数平方

考虑函数 $f(x) = (1/2)\|x\|^2$, 其中 $\|\cdot\|$ 是范数, 对偶范数为 $\|\cdot\|_*$ 。此函数的共轭函数为 $f^*(y) = (1/2)\|y\|_*^2$ 。一方面, 由 $y^T x \leq \|y\|_* \|x\|$ 可知, 对任意 x 下式成立

$$y^T x - 1/2\|x\|^2 \leq \|y\|_* \|x\| - 1/2\|x\|^2$$

上式右端是关于 $\|x\|$ 的二次函数, 其最大值为 $1/2\|y\|_*^2$ 。因此, 对任意 x , 我们有

$$y^T x - (1/2)\|x\|^2 \leq (1/2)\|y\|_*^2$$

即 $f^*(y) \leq (1/2)\|y\|_*^2$ 。另一方面, 任取满足 $y^T x = \|y\|_* \|x\|$ 的向量 x , 对其进行伸缩, 使得 $\|x\| = \|y\|_*$ 。此时,

$$y^T x - (1/2)\|x\|^2 \leq (1/2)\|y\|_*^2$$

因此, $f^*(y) \geq (1/2)\|y\|_*^2$ 。

基本性质

- Fenchel 不等式

从共轭函数的定义可以知道, 对任意 x 和 y , 不等式

$$f(x) + f^*(y) \geq x^T y$$

成立, 这就是 Fenchel 不等式(当 f 可微时, 亦称为 Young 不等式)。

- 可微函数

可微函数 f 的共轭函数亦称为 f 的 Legendre 变换。设函数 f 是凸函数且可微，其定义域为 $\mathbb{K} = \mathbb{R}^n$ ，对于 $z \in \mathbb{R}^n$ ，若

$$y = \nabla f(z)$$

则 $f^*(y) = z^T \nabla f(z) - f(z)$ 。

- 伸缩变换和复合仿射变换

伸缩变换和复合仿射变换在后续对偶理论中有着重要的作用。若 $a > 0$ 以及 $b \in \mathbb{R}$ ， $g(x) = af(x) + b$ 的共轭函数为 $g^*(y) = af^*(y/a) - b$ 。同理，对于复合仿射变换，设 $A \in \mathbb{R}^{n \times n}$ 非奇异， $b \in \mathbb{R}^n$ ，则函数 $g(x) = f(Ax + b)$ 的共轭函数为

$$g^*(y) = f^*(A^{-T}y) - b^T A^{-T}y$$

其定义域为 $\text{dom } g^* = A^T \text{dom } f^*$ 。

- 独立函数的和：如果函数 $f(u, v) = f_1(u) + f_2(v)$ ，其中 f_1 和 f_2 是凸函数，且共轭函数分别为 f_1^* 和 f_2^* ，则

$$f^*(w, z) = f_1^*(w) + f_2^*(z).$$

换言之，独立函数的和的共轭函数是各个凸函数的共轭函数的和。（“独立”的含义是各个函数具有不同的变量）。

机器学习中的共轭函数

机器学习中常涉及优化问题。在优化问题的求解时，常需要将优化问题转化为对偶问题。这便涉及到对偶函数的计算。实际上对偶函数的计算可转化为共轭函数的计算，下面列出其中一些常见的例子。

例 10.3.16. (二次函数) 考虑二次函数 $f(x) = \frac{1}{2}x^T Ax + b^T x + c$ 。

(1) 强凸情形 ($A \succ 0$) :

$$f^*(y) = \frac{1}{2}(y - b)^T A^{-1}(y - b) - c;$$

(2) 一般凸情形 ($A \succeq 0$) :

$$f^*(y) = \frac{1}{2}(y - b)^T A^\dagger(y - b) - c, \text{dom } f^* = \mathcal{R}(A) + b$$

这里 $\mathcal{R}(A)$ 为 A 的像空间。

例 10.3.17. (任意范数) 给定任意范数，则 $f = \|x\|$ 的共轭函数为

$$f_0^*(y) = \begin{cases} 0 & \|y\|_* \leq 1 \\ \infty & \text{其他情况} \end{cases}$$

可以看出此函数是对偶范数单位球的示性函数。

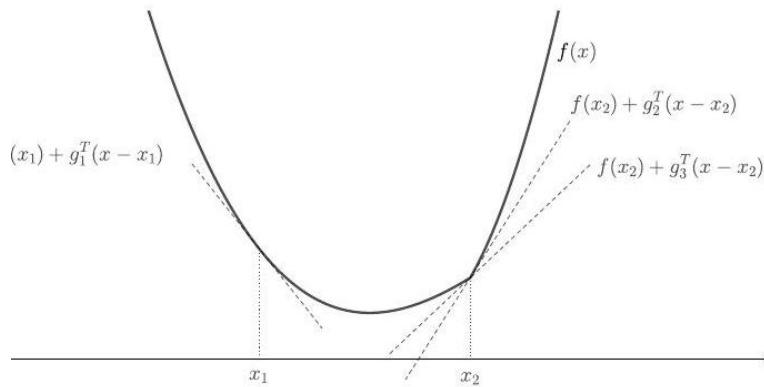


图 10.19: 虚线表示次梯度对应的切线 (一维)

例 10.3.18. (负熵函数) 负熵函数 $f_0(\mathbf{x}) = \sum_{i=1}^n \log x_i$ 的共轭函数为

$$f_0^*(\mathbf{y}) = \sum_{i=1}^n y_i - 1$$

这可以由标量负熵函数的共轭函数和共轭函数的性质得出。

共轭函数是凸分析和优化的基本对象。我们将在后续课程中看到，拉格朗日对偶 $g(\lambda, \mu)$ 有时可以用原目标函数的共轭来表示。

10.3.7 次梯度

次梯度的定义

前面介绍了可微函数的梯度。但是对于一般的函数，之前定义的梯度不一定存在。对于凸函数，类比梯度的一阶性质，我们可以引人次梯度的概念，其在凸优化算法设计与理论分析中扮演着重要角色。

定义 10.3.11. (次梯度) 设 f 为适当凸函数， x 为定义域 $\text{dom } f$ 中的一点。若向量 $g \in \mathbb{R}^n$ 满足

$$f(y) \geq f(x) + g^T(y - x), \quad \forall y \in \text{dom } f,$$

则称 g 为函数 f 在点 x 处的一个次梯度。进一步地，称集合

$$\partial f(x) = \{g \mid g \in \mathbb{R}^n, f(y) \geq f(x) + g^T(y - x), \forall y \in \text{dom } f\}$$

为 f 在点 x 处的次微分。

我们可以通过下图 10.19 可以看出梯度与次梯度之间在几何意义上的区别与联系：

次梯度的存在性

一个问题自然就是：次梯度在什么条件下是存在的？实际上对一般凸函数 f 而言， f 未必在所有的点处都存在次梯度。但对于定义域中的内点， f 在其上的次梯度总是存在的。

定理 10.3.11. (次梯度存在性) 设 f 为凸函数， $\text{dom } f$ 为其定义域。如果 $x \in \text{int dom } f$ ，则 $\partial f(x)$ 是非空的，其中 $\text{int dom } f$ 的含义是集合 $\text{dom } f$ 的所有内点。

证明。考虑 $f(x)$ 的上方图 $\text{epi } f$ 。由于 $(x, f(x))$ 是 $\text{epi } f$ 边界上的点，且 $\text{epi } f$ 为凸集，根据支撑超平面定理，存在 $a \in \mathbb{R}^n, b \in \mathbb{R}$ 使得：

$$\begin{bmatrix} a \\ b \end{bmatrix}^T \left(\begin{bmatrix} y \\ t \end{bmatrix} - \begin{bmatrix} x \\ f(x) \end{bmatrix} \right) \leq 0, \quad \forall (y, t) \in \text{epi } f.$$

即

$$a^T(y - x) \leq b(f(x) - f(y)). \quad (10.30)$$

我们断言 $b < 0$ 。这是因为根据 t 的任意性，在式(10.30)中令 $t \rightarrow +\infty$ ，可以得知式(10.30)成立的必要条件是 $b \leq 0$ ；

同时由于 x 是内点，因此当取 $y = x + \varepsilon a \in \text{dom } f, \varepsilon > 0$ 时， $b = 0$ 不能使得式(10.30)成立。于是令 $g = -\frac{a}{b}$ ，则对任意 $y \in \text{dom } f$ ，我们有

$$g^T(y - x) = \frac{a^T(y - x)}{b} \leq -(f(x) - f(y)),$$

整理得

$$f(y) \geq f(x) + g^T(y - x).$$

这说明 g 是 f 在点 x 处的次梯度。 \square

根据定义可以计算一些简单函数的次微分，在这里我们给出一个例子。

例 10.3.19. (ℓ_2 范数的次微分) 设 $f(x) = \|x\|_2$ ，则 $f(x)$ 在点 $x = 0$ 处不可微，我们求其在该点处的次梯度。注意到对任意的 g 且 $\|g\|_2 \leq 1$ ，根据柯西不等式，

$$g^T(x - 0) \leq \|g\|_2 \|x\|_2 \leq \|x\|_2 - 0$$

因此

$$\{g \mid \|g\|_2 \leq 1\} \subseteq \partial f(0).$$

接下来说明若 $\|g\|_2 > 1$ ，则 $g \notin \partial f(0)$ 。取 $x = g$ ，若 g 为次梯度，则

$$\|g\|_2 - 0 \geq g^T(g - 0) = \|g\|_2^2 > \|g\|_2$$

这显然是矛盾的。综上，我们有

$$\partial f(0) = \{g \mid \|g\|_2 \leq 1\}$$

次梯度的性质

凸函数 $f(x)$ 的次梯度和次微分有许多有用的性质. 下面的定理说明次微分 $\partial f(x)$ 在一定条件下分别为闭凸集和非空有界集.

定理 10.3.12. 设 f 是凸函数, 则 $\partial f(x)$ 有如下性质:

1. 对任何 $x \in \text{dom } f, \partial f(x)$ 是一个闭凸集 (可能为空集);
2. 如果 $x \in \text{int dom } f$, 则 $\partial f(x)$ 非空有界集.

证明. 设 $g_1, g_2 \in \partial f(x)$, 并设 $\lambda \in (0, 1)$, 由次梯度的定义我们有

$$f(y) \geq f(x) + g_1^T(y - x), \quad \forall y \in \text{dom } f$$

$$f(y) \geq f(x) + g_2^T(y - x), \quad \forall y \in \text{dom } f$$

由上面第一式的 λ 倍加上第二式的 $(1 - \lambda)$ 倍, 我们得到 $\lambda g_1 + (1 - \lambda)g_2 \in \partial f(x)$, 从而 $\partial f(x)$ 是凸集. 此外令 $g_k \in \partial f(x)$ 为次梯度且 $g_k \rightarrow g$, 则

$$f(y) \geq f(x) + g_k^T(y - x), \quad \forall y \in \text{dom } f,$$

在上述不等式中取极限, 并注意到极限的保号性, 最终我们有

$$f(y) \geq f(x) + g^T(y - x), \quad \forall y \in \text{dom } f.$$

这说明 $\partial f(x)$ 为闭集.

下设 $x \in \text{int dom } f$, 我们来证明 $\partial f(x)$ 是非空有界的. 首先, $\partial f(x)$ 非空是上一定理的直接结果, 因此我们只需要证明有界性. 对 $i = 1, 2, \dots, n$, 定义 $e_i = (0, \dots, 1, \dots, 0)$ (第 i 个分量为 1, 其余分量均为 0), 易知 $\{e_i\}_{i=1}^n$ 为 \mathbb{R}^n 的一组标准正交基. 取定充分小的正数 r , 使得

$$B = \{x \pm re_i \mid i = 1, 2, \dots, n\} \subset \text{dom } f$$

对任意 $g \in \partial f(x)$, 不妨设 g 不为 0. 存在 $y \in B$ 使得

$$f(y) \geq f(x) + g^T(y - x) = f(x) + r\|g\|_\infty.$$

由此得到

$$\|g\|_\infty \leq \frac{\max_{y \in B} f(y) - f(x)}{r} < +\infty$$

即 $\partial f(x)$ 有界. □

定理 10.3.13. 设 $f(x)$ 在 $x_0 \in \text{int dom } f$ 处可微, 则 $\partial f(x_0) = \{\nabla f(x_0)\}$

证明. 根据可微凸函数的一阶条件可知梯度 $\nabla f(x_0)$ 为次梯度. 下证 $f(x)$ 在点 x_0 处不可能有其他次梯度. 设 $g \in \partial f(x_0)$, 根据次梯度的定义, 对任意的非零 $v \in \mathbb{R}^n$ 且 $x_0 + tv \in \text{dom } f, t > 0$ 有

$$f(x_0 + tv) \geq f(x_0) + tg^T v.$$

若 $g \neq \nabla f(x_0)$, 取 $v = g - \nabla f(x_0) \neq 0$, 上式变形为

$$\frac{f(x_0 + tv) - f(x_0) - t\nabla f(x_0)^T v}{t\|v\|} \geq \frac{(g - \nabla f(x_0))^T v}{\|v\|} = \|v\|.$$

不等式两边令 $t \rightarrow 0$, 根据 Fréchet 可微的定义, 左边趋于 0, 而右边是非零正数, 可得到矛盾. □

和梯度类似, 凸函数的次梯度也具有某种单调性. 这一性质在很多和次梯度有关的算法的收敛性分析中起到了关键的作用.

定理 10.3.14. (次梯度的单调性) 设 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 为凸函数, $x, y \in \text{dom } f$, 则

$$(u - v)^T(x - y) \geq 0$$

其中 $u \in \partial f(x), v \in \partial f(y)$.

证明. 由次梯度的定义,

$$f(y) \geq f(x) + u^T(y - x)$$

$$f(x) \geq f(y) + v^T(x - y).$$

将以上两个不等式相加即得结论. \square

对于闭凸函数 (即凸下半连续函数), 次梯度还具有某种连续性.

定理 10.3.15. 设 $f(x)$ 是闭凸函数且 ∂f 在点 \bar{x} 附近存在且非空. 若序列 $x^k \rightarrow \bar{x}, g^k \in \partial f(x^k)$ 为 $f(x)$ 在点 x^k 处的次梯度, 且 $g^k \rightarrow \bar{g}$, 则 $\bar{g} \in \partial f(\bar{x})$.

证明. 对任意 $y \in \text{dom } f$, 根据次梯度的定义,

$$f(y) \geq f(x^k) + \langle g^k, y - x^k \rangle.$$

对上述不等式两边取下极限, 我们有

$$\begin{aligned} f(y) &\geq \liminf_{k \rightarrow \infty} [f(x^k) + \langle g^k, y - x^k \rangle] \\ &\geq f(\bar{x}) + \langle \bar{g}, y - \bar{x} \rangle \end{aligned}$$

其中第二个不等式利用了 $f(x)$ 的下半连续性以及 $g^k \rightarrow \bar{g}$, 由此可推出 $\bar{g} \in \partial f(\bar{x})$. \square

凸函数的方向导数

在微积分中我们接触过方向导数的概念. 设 f 为适当函数, 给定点 x_0 以及方向 $d \in \mathbb{R}^n$, 方向导数 (若存在) 定义为

$$\lim_{t \downarrow 0} \phi(t) = \lim_{t \downarrow 0} \frac{f(x_0 + t\mathbf{d}) - f(x_0)}{t},$$

其中 $t \downarrow 0$ 表示 t 单调下降趋于 0. 对于凸函数 $f(\mathbf{x})$, 易知 $\phi(t)$ 在 $(0, +\infty)$ 上是单调不减的, 上式中的极限号 \lim 可以替换为下确界 \inf . 上述此时极限总是存在 (可以为无穷), 进而凸函数总是可以定义方向导数.

定义 10.3.12. (方向导数) 对于凸函数 f , 给定点 $x_0 \in \text{dom } f$ 以及方向 $d \in \mathbb{R}^n$, 其方向导数定义为

$$\partial f(x_0; d) = \inf_{t > 0} \frac{f(x_0 + t\mathbf{d}) - f(x_0)}{t}.$$

方向导数可能是正负无穷, 但在定义域的内点处方向导数 $\partial f(x_0; d)$ 是有限的.

定理 10.3.16. 设 $f(\mathbf{x})$ 为凸函数, $\mathbf{x}_0 \in \text{int dom } f$, 则对任意 $\mathbf{d} \in \mathbb{R}^n$, $\partial f(\mathbf{x}_0; \mathbf{d})$ 有限.

证明. 首先 $\partial f(\mathbf{x}_0; \mathbf{d})$ 不为正无穷是显然的. 由于 $\mathbf{x}_0 \in \text{int dom } f$, 根据存在定理可知 $f(\mathbf{x})$ 在点 \mathbf{x}_0 处存在次梯度 \mathbf{g} . 根据方向导数的定义, 我们有

$$\begin{aligned}\partial f(\mathbf{x}_0; \mathbf{d}) &= \inf_{t>0} \frac{f(\mathbf{x}_0 + t\mathbf{d}) - f(\mathbf{x}_0)}{t} \\ &\geq \inf_{t>0} \frac{\mathbf{g}^T \mathbf{d}}{t} = \mathbf{g}^T \mathbf{d}.\end{aligned}$$

其中的不等式利用了次梯度的定义. 这说明 $\partial f(\mathbf{x}_0; \mathbf{d})$ 不为负无穷. \square

凸函数的方向导数和次梯度之间有很强的联系. 以下结果表明, 凸函数 $f(\mathbf{x})$ 关于 \mathbf{d} 的方向导数 $\partial f(\mathbf{x}; \mathbf{d})$ 正是 f 在点 \mathbf{x} 处的所有次梯度与 \mathbf{d} 的内积的最大值.

定理 10.3.17. 设 $f: \mathbb{R}^n \rightarrow (-\infty, +\infty]$ 为凸函数, 点 $\mathbf{x}_0 \in \text{int dom } f$, \mathbf{d} 为 \mathbb{R}^n 中任一方向, 则

$$\partial f(\mathbf{x}_0; \mathbf{d}) = \max_{\mathbf{g} \in \partial f(\mathbf{x}_0)} \mathbf{g}^T \mathbf{d}.$$

证明. 为了方便, 对任意 $\mathbf{v} \in \mathbb{R}^m$, 我们定义 $q(\mathbf{v}) = \partial f(\mathbf{x}_0; \mathbf{v})$. 根据上述命题的证明过程可直接得出对任意 $\mathbf{g} \in \partial f(\mathbf{x}_0)$,

$$q(\mathbf{d}) = \partial f(\mathbf{x}_0; \mathbf{d}) \geq \mathbf{g}^T \mathbf{d}.$$

这说明 $\partial f(\mathbf{x}_0; \mathbf{d})$ 是 $\mathbf{g}^T \mathbf{d}$ 的一个上界, 接下来说明该上界为上确界. 构造函数

$$h(\mathbf{v}, t) = t \left(f\left(\mathbf{x}_0 + \frac{\mathbf{v}}{t}\right) - f(\mathbf{x}_0) \right),$$

可知 $h(\mathbf{v}, t)$ 为 $\tilde{f}(\mathbf{v}) = f(\mathbf{x}_0 + \mathbf{v}) - f(\mathbf{x}_0)$ 的透视函数, 并且

$$q(\mathbf{v}) = \inf_{t'>0} \frac{f(\mathbf{x}_0 + t' \mathbf{v}) - f(\mathbf{x}_0)}{t'} \stackrel{t=1/t'}{=} \inf_{t>0} h(\mathbf{v}, t).$$

根据定理透视函数保凸性知 $h(\mathbf{v}, t)$ 为凸函数, 又根据取下确界仍为凸函数, 因此 $q(\mathbf{v})$ 关于 \mathbf{v} 是凸函数. 由上述命题直接可以得出 $\text{dom } q = \mathbb{R}^n$, 因此 $q(\mathbf{v})$ 在全空间任意一点次梯度存在. 对方向 \mathbf{d} , 设 $\hat{\mathbf{g}} \in \partial q(\mathbf{d})$, 则对任意 $\mathbf{v} \in \mathbb{R}^n$ 以及 $\lambda \geq 0$, 我们有

$$\lambda q(\mathbf{v}) = q(\lambda \mathbf{v}) \geq q(\mathbf{d}) + \hat{\mathbf{g}}^T (\lambda \mathbf{v} - \mathbf{d}).$$

令 $\lambda = 0$, 我们有 $q(\mathbf{d}) \leq \hat{\mathbf{g}}^T \mathbf{d}$; 令 $\lambda \rightarrow +\infty$, 我们有

$$q(\mathbf{v}) \geq \hat{\mathbf{g}}^T \mathbf{v},$$

进而推出

$$f(\mathbf{x} + \mathbf{v}) \geq f(\mathbf{x}) + q(\mathbf{v}) \geq f(\mathbf{x}) + \hat{\mathbf{g}}^T \mathbf{v}.$$

这说明 $\hat{\mathbf{g}} \in \partial f(\mathbf{x})$ 且 $\hat{\mathbf{g}}^T \mathbf{d} \geq q(\mathbf{d})$. 即 $q(\mathbf{d})$ 为 $\mathbf{g}^T \mathbf{d}$ 的上确界, 且当 $\mathbf{g} = \hat{\mathbf{g}}$ 时上确界达到. \square

上述定理可对一般的 $\mathbf{x} \in \text{dom } f$ 作如下推广:

定理 10.3.18. 设 f 为适当凸函数, 且在 \mathbf{x}_0 处次微分不为空集, 则对任意 $\mathbf{d} \in \mathbb{R}^n$ 有

$$\partial f(\mathbf{x}_0; \mathbf{d}) = \sup_{\mathbf{g} \in \partial f(\mathbf{x}_0)} \mathbf{g}^T \mathbf{d},$$

且当 $\partial f(\mathbf{x}_0; \mathbf{d})$ 不为无穷时, 上确界可以取到.

次梯度的计算规则

如何计算一个不可微凸函数的次梯度在优化算法设计中是很重要的问题. 根据定义来计算次梯度一般来说比较繁琐, 我们来介绍一些次梯度的计算规则. 本小节讨论的计算规则都默认 $x \in \text{int dom } f$.

基本规则 我们首先不加证明地给出一些计算次梯度 (次微分) 的基本规则.

1. 可微凸函数: 设 f 为凸函数, 若 f 在点 x 处可微, 则 $\partial f(x) = \{\nabla f(x)\}$.
2. 凸函数的非负线性组合: 设 f_1, f_2 为凸函数, 且满足

$$\text{int dom } f_1 \cap \text{dom } f_2 \neq \emptyset,$$

而 $x \in \text{dom } f_1 \cap \text{dom } f_2$. 若

$$f(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x), \quad \alpha_1, \alpha_2 \geq 0,$$

则 $f(x)$ 的次微分

$$\partial f(x) = \alpha_1 \partial f_1(x) + \alpha_2 \partial f_2(x).$$

3. 线性变量替换: 设 h 为适当凸函数, 并且函数 h 满足

$$f(x) = h(Ax + b), \quad \forall x \in \mathbb{R}^m,$$

其中 $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$. 若存在 $x^\sharp \in \mathbb{R}^m$ 使得 $Ax^\sharp + b \in \text{int dom } h$, 则

$$\partial f(x) = A^\top \partial h(Ax + b), \quad \forall x \in \text{int dom } f.$$

两个函数之和的次梯度计算规则 以 $\text{int dom } f_1 \cap \text{dom } f_2 \neq \emptyset$ 为 Moreau-Rockafellar 定理给出两个凸函数之和的次微分的计算方法.

定理 10.3.19. 设 $f_1, f_2 : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ 是两个凸函数, 则对任意的 $x_0 \in \mathbb{R}^n$,

$$\partial f_1(x_0) + \partial f_2(x_0) \subseteq \partial(f_1 + f_2)(x_0).$$

进一步地, 若 $\text{int dom } f_1 \cap \text{dom } f_2 \neq \emptyset$, 则对任意的 $x_0 \in \mathbb{R}^n$,

$$\partial(f_1 + f_2)(x_0) = \partial f_1(x_0) + \partial f_2(x_0).$$

函数族的上确界函数的次梯度计算规则 前面我们已经提介绍过一族凸函数的上确界函数仍是凸函数. 我们对这样得到的凸函数的次梯度有如下重要结果:

定理 10.3.20. 设 $f_1, f_2, \dots, f_m : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ 均为凸函数, 令

$$f(x) = \max \{f_1(x), f_2(x), \dots, f_m(x)\}, \quad \forall x \in \mathbb{R}^n.$$

对 $x_0 \in \bigcap_{i=1}^m \text{int dom } f_i$, 定义 $I(x_0) = \{i \mid f_i(x_0) = f(x_0)\}$, 则

$$\partial f(x_0) = \text{conv} \bigcup_{i \in I(x_0)} \partial f_i(x_0).$$

固定分量的函数极小值函数次梯度计算规则 设 $h: \mathbb{R}^n \times \mathbb{R}^m \rightarrow (-\infty, +\infty]$ 是关于 (x, y) 的凸函数, 则 $f(x) \stackrel{\text{def}}{=} \inf_y h(x, y)$ 是关于 $x \in \mathbb{R}^n$ 的凸函数. 以下结果可以用于求解 f 在点 x 处的一个次梯度.

定理 10.3.21. 考虑函数

$$f(x) = \inf_y h(x, y)$$

其中

$$h: \mathbb{R}^n \times \mathbb{R}^m \rightarrow (-\infty, +\infty]$$

是关于 (x, y) 的凸函数. 对 $\hat{x} \in \mathbb{R}^n$, 设 $\hat{y} \in \mathbb{R}^m$ 满足 $h(\hat{x}, \hat{y}) = f(\hat{x})$, 且存在 $g \in \mathbb{R}^n$ 使得 $(g, 0) \in \partial h(\hat{x}, \hat{y})$, 则 $g \in \partial f(\hat{x})$.

在机器学习中, 存在一些非常常见的不可微的函数, 但是存在次梯度. 例如分段线性函数以及 ℓ_1 正则项.

例 10.3.20. (分段线性函数) 令

$$f(x) = \max_{i=1, 2, \dots, m} \{a_i^T x + b_i\}$$

其中 $x, a_i \in \mathbb{R}^n, b_i \in \mathbb{R}, i = 1, 2, \dots, m$, 则

$$\partial f(x) = \text{conv}\{a_i \mid i \in I(x)\}$$

其中

$$I(x) = \{i \mid a_i^T x + b_i = f(x)\}$$

例 10.3.21. (ℓ_1 范数) 定义 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 为 ℓ_1 范数, 则对 $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, 有

$$f(x) = \|x\|_1 = \max_{s \in \{-1, 1\}^n} s^T x.$$

于是

$$\partial f(x) = J_1 \times J_2 \times \dots \times J_n, \quad J_k = \begin{cases} [-1, 1], & x_k = 0 \\ \{1\}, & x_k > 0 \\ \{-1\}, & x_k < 0 \end{cases}$$

例 10.3.22. 设 C 是 \mathbb{R}^n 中一闭凸集, 令

$$f(\mathbf{x}) = \inf_{\mathbf{y} \in C} \|\mathbf{x} - \mathbf{y}\|_2.$$

令 $\hat{\mathbf{x}} \in \mathbb{R}^n$, 我们来求 f 在 $\hat{\mathbf{x}}$ 处的一个次梯度.

(1) 若 $f(\hat{\mathbf{x}}) = 0$, 则容易验证 $\mathbf{g} = 0 \in \partial f(\hat{\mathbf{x}})$;

(2) 若 $f(\hat{\mathbf{x}}) > 0$, 由 C 是闭凸集, 可取 $\hat{\mathbf{y}}$ 为 $\hat{\mathbf{x}}$ 在 C 上的投影, 即

$$\hat{\mathbf{y}} = \mathcal{P}_c(\hat{\mathbf{x}}) \stackrel{\text{def}}{=} \underset{\mathbf{y} \in C}{\operatorname{argmin}} \|\hat{\mathbf{x}} - \mathbf{y}\|_2.$$

利用 $\hat{\mathbf{y}}$ 的定义可以验证

$$\mathbf{g} = \frac{1}{\|\hat{\mathbf{x}} - \hat{\mathbf{y}}\|_2} (\hat{\mathbf{x}} - \hat{\mathbf{y}}) = \frac{1}{\|\hat{\mathbf{x}} - \mathcal{P}_c(\hat{\mathbf{x}})\|_2} (\hat{\mathbf{x}} - \mathcal{P}_c(\hat{\mathbf{x}})),$$

满足固定分量的函数极小值情形的条件. 故 $\mathbf{g} \in \partial f(\hat{\mathbf{x}})$.

10.4 凸优化

10.4.1 凸优化问题

凸优化问题的标准形式

定义 10.4.1. 形如

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \mathbf{a}_j^T \mathbf{x} = b_j, \quad j = 1, \dots, p \end{aligned} \tag{10.31}$$

的优化问题, 若 f_0, \dots, f_m 为凸函数, 则称为凸优化问题。特别地, 当 $m = p = 0$ 时, 式(10.31)被称为无约束凸优化问题。

标准形式的凸优化问题也经常等价地表达为:

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

有时原凸优化问题并非上述形式, 但可以进行转化成标准形式, 如下例题所述。

例 10.4.1.

$$\begin{aligned} & \min f_0(\mathbf{x}) = x_1^2 + x_2^2 \\ \text{s.t. } & f_1(\mathbf{x}) = x_1/(1 + x_2^2) \leq 0 \\ & h_1(\mathbf{x}) = (x_1 + x_2)^2 = 0 \end{aligned}$$

可转化成:

$$\begin{aligned} & \min f_0(\mathbf{x}) = x_1^2 + x_2^2 \\ \text{s.t. } & f_1(\mathbf{x}) = x_1 \leq 0 \\ & h_1(\mathbf{x}) = x_1 + x_2 = 0 \end{aligned}$$

对比凸优化问题(10.31)和一般优化问题的标准形式问题(??), 可以看出, 凸优化问题有三个附加要求:

- 目标函数必须是凸的,
- 不等式约束函数必须是凸的,
- 等式约束函数 $h_i(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} - b_i$ 必须是仿射的。

由凸函数的性质可知: 凸优化问题的可行集是凸的。因此, 凸优化问题本质上是在一个凸集上极小化一个凸的目标函数。

凸优化问题的等价问题

如果从一个问题的解，很容易得到另一个问题的解，并且反之亦然，我们称两个问题是等价的。例如，考虑问题

$$\begin{aligned} \min \quad & \tilde{f}(\mathbf{x}) = \alpha_0 f_0(\mathbf{x}) \\ \text{s.t.} \quad & \tilde{f}_i(\mathbf{x}) = \alpha_i f_i(\mathbf{x}) \leq 0, i = 1, \dots, m \\ & \beta_j a_j^T \mathbf{x} = \beta_j b_j, j = 1, \dots, p \end{aligned} \quad (10.32)$$

其中 $\alpha_i > 0, i = 0, \dots, m$ 且 $\beta_i \neq 0, i = 1, \dots, p$ 。这个问题是通过将优化问题(10.31)的目标函数和不等式约束函数乘以正的常数，并将等式约束函数乘以非零常数得到的。因此，问题(10.32)的可行集与原问题(10.31)是相同的。显然， \mathbf{x} 是原问题(10.31)的最优解，当且仅当它也是问题(10.32)的最优解。综上所述，这两个问题是等价的。事实上，这两个问题(10.32)和(10.31)是不同的（除非 α_i 和 β_i 都是 1），因为目标函数和约束函数都不同。下面介绍几种产生等价问题的一些典型方法。

消除等式约束 一个凸问题的等式约束必须是线性的，即具有 $A\mathbf{x} = b$ 的形式。在这种情况下，可以通过寻找 $A\mathbf{x} = b$ 的一个特解 \mathbf{x}_0 和域为 A 的零空间的矩阵 F 来消除这些等式约束，从而得到关于 \mathbf{z} 的优化问题。

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & A\mathbf{x} = b \end{aligned}$$

等价于

$$\begin{aligned} \min \quad & f_0(\mathbf{Fz} + \mathbf{x}_0) \\ \text{s.t.} \quad & f_i(\mathbf{Fz} + \mathbf{x}_0) \leq 0, \end{aligned}$$

其中 \mathbf{F} 和 \mathbf{x}_0 满足：

$$A\mathbf{x} = b \iff \mathbf{x} = \mathbf{Fz} + \mathbf{x}_0$$

因为凸函数和仿射函数的复合依然是凸的，消除等式约束可以保持问题的凸性。而且消除等式约束的过程（以及从变换后问题的解重构出原问题的解）只需利用标准的线性代数运算。

至少在理论上，这意味着我们可以集中精力于不含等式约束的凸优化问题。但是，在很多情况下，由于消除等式约束会使得问题更难理解和求解，甚至使得求解它的算法失效。因此，有时候还是会保留等式约束。例如，当变量 \mathbf{x} 维数很高时，消除等式约束确实有可能破坏问题的稀疏性或其它结构信息。

引入等式约束 在凸优化问题中引入新的变量和等式约束, 前提是等式约束是线性的, 所得的优化问题仍然是凸的。例如, 如果目标函数或约束函数具有 $f_i(\mathbf{A}_i \mathbf{x} + \mathbf{b}_i)$ 的形式, 其中 $\mathbf{A}_i \in \mathbb{R}^{k_i \times n}$, 那么, 可以引入新的变量 $\mathbf{y}_i \in \mathbb{R}^{k_i}$, 用 $f_i(\mathbf{y}_i)$ 替换 $f_i(\mathbf{A}_i \mathbf{x} + \mathbf{b}_i)$ 并添加线性等式约束 $\mathbf{y}_i = \mathbf{A}_i \mathbf{x} + \mathbf{b}_i$:

$$\begin{aligned} \min \quad & f_0(\mathbf{A}_0 \mathbf{x} + \mathbf{b}_0) \\ \text{s.t.} \quad & f_i(\mathbf{A}_i \mathbf{x} + \mathbf{b}_i) \leq 0, \quad i = 1, \dots, m \end{aligned}$$

等价于

$$\begin{aligned} \min \quad & f_0(\mathbf{y}_0) \\ \text{s.t.} \quad & f_i(\mathbf{y}_i) \leq 0, \quad i = 1, \dots, m \\ & \mathbf{y}_i = \mathbf{A}_i \mathbf{x} + \mathbf{b}_i, \quad i = 0, \dots, m \end{aligned}$$

添加松弛变量 通过在不等式约束中引入松弛变量 $s_i, i = 1, \dots, m$, 可以得到新的等式约束 $f_i(\mathbf{x}) + s_i = 0$ 。因为凸优化问题中的等式约束必须是仿射的, 所以 f_i 必须是仿射函数。换言之, 可以在线性不等式约束中引入松弛变量, 从而保持原问题的凸性。即

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i = 1, \dots, m \end{aligned}$$

等价于

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{a}_i^T \mathbf{x} + s_i = b_i, \quad i = 1, \dots, m \\ & s_i \geq 0, \quad i = 1, \dots, m \end{aligned}$$

上境图问题形式 凸优化问题(10.31)的上境图形式为

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & f_0(\mathbf{x}) - t \leq 0 \\ & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

目标函数是线性的 (因而也是凸的) 并且新的约束函数 $f_0(\mathbf{x}) - t$ 也是 (\mathbf{x}, t) 上的凸函数, 所以上境图问题也是凸的。

有时也称线性目标函数对凸优化问题是普适的, 因为任何凸优化问题都可以轻易地转换为具有线性目标函数的问题。凸优化问题的上境图形式具有一些实际的用途。通过假设凸优化问题的目标函数为线性, 我们可以简化理论分析。也可以用于简化算法, 因为通过上述变换, 每个解决线性目标的凸优化问题的算法都可以用来解决任意的凸优化问题 (前提是它可以处理约束 $f_0(\mathbf{x}) - t \leq 0$)。

优化部分变量 已知等式

$$\inf_{x,y} f(x,y) = \inf_x \tilde{f}(x)$$

成立, 其中 $\tilde{f}(x) = \inf_y f(x,y)$ 。换言之, 可以通过先优化一部分变量再优化另一部分变量来达到函数优化的目的。这个简单而普适的原则可以用来将原问题转换为其等价形式。对于一般形式, 其描述冗长而不直观, 因此, 我们这里仅用一个例子来进行说明。

设变量 $\mathbf{x} \in \mathbb{R}^n$ 被分为 $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$, 其中 $\mathbf{x}_1 \in \mathbb{R}^{n_1}, \mathbf{x}_2 \in \mathbb{R}^{n_2}$, 并且 $n_1 + n_2 = n$ 。考虑问题

$$\begin{aligned} \min \quad & f_0(\mathbf{x}_1, \mathbf{x}_2) \\ \text{s.t.} \quad & f_i(\mathbf{x}_1) \leq 0, \quad i = 1, \dots, m_1 \\ & \tilde{f}_j(\mathbf{x}_2) \leq 0, \quad j = 1, \dots, m_2 \end{aligned} \quad (10.33)$$

其约束相互独立, 也就是说每个约束函数只与 \mathbf{x}_1 或 \mathbf{x}_2 有关。首先优化 \mathbf{x}_2 , 定义函数 \tilde{f}_0 为

$$\tilde{f}_0(\mathbf{x}_1) = \inf_z \{f_0(\mathbf{x}_1, z) \mid \tilde{f}_j(z) \leq 0, \quad j = 1, \dots, m_2\}$$

则原问题(10.33)等价于

$$\begin{aligned} \min \quad & \tilde{f}_0(\mathbf{x}_1) \\ \text{s.t.} \quad & f_i(\mathbf{x}_1) \leq 0, \quad i = 1, \dots, m_1. \end{aligned} \quad (10.34)$$

例 10.4.2. 考虑严格凸的二次目标问题

$$\begin{aligned} \min \quad & \mathbf{x}_1^T \mathbf{P}_{11} \mathbf{x}_1 + 2\mathbf{x}_1^T \mathbf{P}_{12} \mathbf{x}_2 + \mathbf{x}_2^T \mathbf{P}_{22} \mathbf{x}_2 \\ \text{s.t.} \quad & f_i(\mathbf{x}_1) \leq 0, \quad i = 1, \dots, m \end{aligned}$$

其中, \mathbf{P}_{11} 和 \mathbf{P}_{22} 均为对称矩阵, 并且变量 \mathbf{x}_2 不受约束。这里, 我们可以解析地优化 \mathbf{x}_2 , 令

$$\frac{\partial(\mathbf{x}_1^T \mathbf{P}_{11} \mathbf{x}_1 + 2\mathbf{x}_1^T \mathbf{P}_{12} \mathbf{x}_2 + \mathbf{x}_2^T \mathbf{P}_{22} \mathbf{x}_2)}{\partial \mathbf{x}_2} = \mathbf{0},$$

得 $\mathbf{x}_2 = -\mathbf{P}_{22}^{-1} \mathbf{P}_{12} \mathbf{x}_1$, 于是

$$\inf_{\mathbf{x}_2} (\mathbf{x}_1^T \mathbf{P}_{11} \mathbf{x}_1 + 2\mathbf{x}_1^T \mathbf{P}_{12} \mathbf{x}_2 + \mathbf{x}_2^T \mathbf{P}_{22} \mathbf{x}_2) = \mathbf{x}_1^T (\mathbf{P}_{11} - \mathbf{P}_{12} \mathbf{P}_{22}^{-1} \mathbf{P}_{12}^T) \mathbf{x}_1$$

因此, 原问题等价于

$$\begin{aligned} \min \quad & \mathbf{x}_1^T (\mathbf{P}_{11} - \mathbf{P}_{12} \mathbf{P}_{22}^{-1} \mathbf{P}_{12}^T) \mathbf{x}_1 \\ \text{s.t.} \quad & f_i(\mathbf{x}_1) \leq 0, \quad i = 1, \dots, m \end{aligned} \quad (10.35)$$

注意, 最优化凸函数的部分变量将保持凸性不变。如果问题(10.33)中的目标函数 f_0 是关于变量 \mathbf{x}_1 和 \mathbf{x}_2 的联合凸函数, 并且 $f_i, i = 1, \dots, m_1$ 和 $\tilde{f}_j, j = 1, \dots, m_2$ 都是凸函数, 那么, 其等价问题(10.34)也是凸的。

凸优化的全局最优化

定理 10.4.1. 凸优化问题中, 局部最优点就是(全局)最优点。

证明. 设 \mathbf{x} 是凸优化问题的局部最优解, 即存在 $R > 0$, 对任意可行的 \mathbf{z} 且 $\|\mathbf{z} - \mathbf{x}\|_2 \leq R$, 则 $f_0(\mathbf{z}) \geq f_0(\mathbf{x})$ 。设 \mathbf{y} 是最优点使得 $f_0(\mathbf{y}) < f_0(\mathbf{x})$, 且 $\|\mathbf{y} - \mathbf{x}\|_2 > R$ 。

考虑 $\mathbf{z} = (1 - \theta)\mathbf{x} + \theta\mathbf{y}$, 其中

$$\theta = \frac{R}{2\|\mathbf{y} - \mathbf{x}\|_2}.$$

则易证明 $\|\mathbf{z} - \mathbf{x}\|_2 = R/2 < R$ 。又 \mathbf{z} 是两个可行点的凸组合, 因此也是可行的。根据凸函数的性质可知

$$f_0(\mathbf{z}) \leq (1 - \theta)f_0(\mathbf{x}) + \theta f_0(\mathbf{y}) < f_0(\mathbf{x}).$$

这与 \mathbf{x} 是局部最优解矛盾。 \square

10.4.2 典型凸优化及其在数据科学中应用示例

线性规划

例 10.4.3. 当目标函数和约束函数都是仿射时, 问题称作线性规划 (LP)。一般的线性规划具有以下形式

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} + d \\ \text{s.t.} \quad & \mathbf{Gx} \leq \mathbf{h} \\ & \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{10.36}$$

其中, $\mathbf{G} \in \mathbb{R}^{m \times n}, \mathbf{A} \in \mathbb{R}^{p \times n}$ 。线性规划问题都是凸优化问题。

有时会将目标函数中的常数 d 省略, 因为它不影响最优解 (以及可行解) 集合。考虑到极大化目标函数 $\mathbf{c}^T \mathbf{x} + d$ 等价于极小化 $-\mathbf{c}^T \mathbf{x} - d$ (仍然是凸的), 因此, 具有仿射目标函数和约束函数的最大化问题也被称为线性规划问题。

线性规划的几何解释可以见图 10.20: 线性规划(10.36)的可行集是多面体 P ; 这一问题是在 P 上极小化仿射函数 $\mathbf{c}^T \mathbf{x} + d$ (或者极小化线性函数 $\mathbf{c}^T \mathbf{x}$)。

事实上, 线性规划(10.36)已经被广泛深入地研究, 人们在研究其求解算法会常常使用: 标准形和不等式形。标准形的线性规划

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{10.37}$$

中仅有的不等式都是关于变量的非负性约束。如果线性规划没有等式约束, 则称为不等式形的线性规划, 常写作

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} \leq \mathbf{b} \end{aligned} \tag{10.38}$$

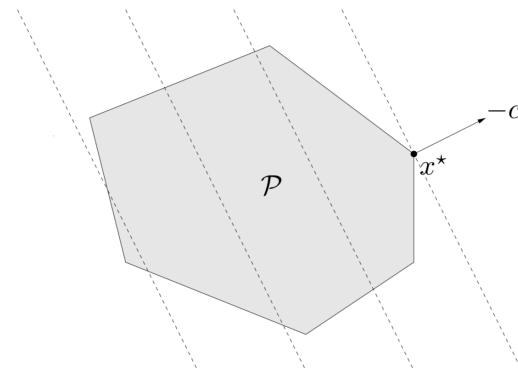


图 10.20: 线性规划的几何解释。可行集 \mathcal{P} 是多面体, 如阴影所示。目标 $\mathbf{c}^T \mathbf{x}$ 是线性的, 所以其等位曲线是与 \mathbf{c} 正交的超平面 (如虚线所示)。点 \mathbf{x}^* 是最优的, 它是 \mathcal{P} 中在方向 $-\mathbf{c}$ 上最远的点。

至于线性规划各种形式间的转换, 以及更多有关线性规划的介绍, 读者参考相关的专著。线性规划出现在非常多的领域和应用中。这里我们给出一些典型的例子。

例 10.4.4. 分片线性极小化

$$\min_{\mathbf{x}} \max_{i=1, \dots, m} (\mathbf{a}_i^T \mathbf{x} + \mathbf{b}_i)$$

等价于如下线性规划问题:

$$\begin{aligned} \min & \quad t \\ \text{s.t.} & \quad \mathbf{a}_i^T \mathbf{x} + \mathbf{b}_i \leq t, \quad i = 1, \dots, m \end{aligned}$$

例 10.4.5. 马尔科夫决策过程 在马尔科夫决策过程中, 考虑终止时间 $T = \infty$ 的情形。假设奖励有界, 为求出最优动作以及最优期望奖励, 将 Bellman 方程转化为如下线性规划问题:

$$\begin{aligned} \max_{V \in \mathbb{R}^{|S|}} & \quad \sum_i V(i) \\ \text{s.t.} & \quad V(i) \geq \sum_j P_a(i, j)(r(i, a) + \gamma V(j)), \forall i \in S, \forall a \in A, \end{aligned}$$

其中 $V(i)$ 是向量 V 的第 i 个分量, 表示从状态 i 出发得到的累积奖励, $P_a(i, j)$ 是转移概率, $r(i, a)$ 是单步奖励, γ 为折现因子。

例 10.4.6. 多面体的 Chebyshev 中心 多面体 $\mathcal{P} = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i = 1, \dots, m\}$ 的 Chebyshev 中心是最大的内切球球心

$$\mathcal{B} = \{\mathbf{x}_c + \mathbf{u} | \|\mathbf{u}\|_2 \leq r\}$$

- 对于所有的 $\mathbf{x} \in B$, 有 $\mathbf{a}_i^T \mathbf{x} \leq b_i$ 。这等价于

$$\sup\{\mathbf{a}_i^T(\mathbf{x}_c + \mathbf{u}) \mid \|\mathbf{u}\|_2 \leq r\} = \mathbf{a}_i^T \mathbf{x}_c + r \|\mathbf{a}_i\|_2 \leq b_i$$

- 因此 \mathbf{x}_c, r 可以通过解决一个 LP 问题被确定下来

$$\max r$$

$$\text{s.t. } \mathbf{a}_i^T \mathbf{x}_c + r \|\mathbf{a}_i\|_2 \leq b_i, \quad i = 1, \dots, m$$

例 10.4.7. 压缩感知中的基追踪问题 基追踪问题是压缩感知中的一个基本问题, 可以写为

$$\min \|\mathbf{x}\|_1,$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}.$$

对每个 $|x_i|$ 引入一个新的变量 z_i , 可以将问题 (II) 转化为

$$\min \sum_{i=1}^n z_i$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b},$$

$$-z_i \leq x_i \leq z_i, \quad i = 1, 2, \dots, n,$$

这是一个线性规划问题。

例 10.4.8. 分式线性问题

$$\min f_0(\mathbf{x})$$

$$\text{s.t. } \mathbf{G}\mathbf{x} \leq \mathbf{h}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

其中

$$f_0(\mathbf{x}) = \frac{\mathbf{c}^T \mathbf{x} + d}{\mathbf{e}^T \mathbf{x} + f}, \quad \text{dom } f_0 = \{\mathbf{x} \mid \mathbf{e}^T \mathbf{x} + f \geq 0\}$$

可以转换为等价的线性规划

$$\min \mathbf{c}^T \mathbf{y} + dz$$

$$\text{s.t. } \mathbf{G}\mathbf{y} - \mathbf{h}z \leq \mathbf{0}$$

$$\mathbf{A}\mathbf{y} - \mathbf{b}z = \mathbf{0}$$

$$\mathbf{e}^T \mathbf{y} + fz = 1$$

$$z \geq 0$$

二次规划

当凸优化问题(10.31)的目标函数是(凸)二次型, 并且约束函数为仿射函数时, 该问题称为二次规划(QP), 具体表示为

$$\begin{aligned} \min \quad & \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r \\ \text{s.t.} \quad & \mathbf{G}\mathbf{x} \leq \mathbf{h} \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned} \tag{10.39}$$

其中, $\mathbf{P} \in S_+^n$, $\mathbf{G} \in \mathbb{R}^{m \times n}$, $\mathbf{A} \in \mathbb{R}^{p \times n}$ 。可以看出线性规划是二次规划的特例, 通过在(10.39)中取 $\mathbf{P} = 0$ 可得。实际上求解二次规划问题, 相当于在多面体上极小化一个凸二次函数, 如图10.21所示:

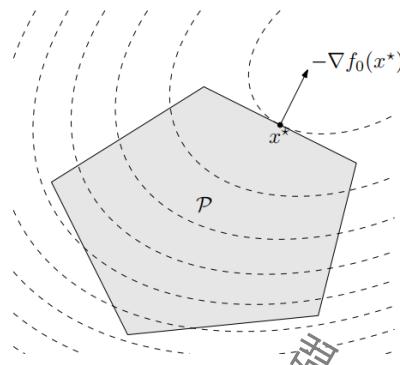


图 10.21: QP 的几何解释。多面体为可行集 \mathcal{P} , 虚线为凸二次目标函数的等值线。最优点为 \mathbf{x}^* 。

例 10.4.9. 最小二乘及回归 最小化凸二次函数

$$\|\mathbf{Ax} - \mathbf{b}\|_2^2 = \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{b}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{b}$$

的问题是一个 (无约束的) 二次规划。在很多领域中, 都会遇到这个问题, 并有很多的名字, 例如回归分析或最小二乘逼近。这个问题很简单, 有著名的解析解 $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}$, 其中, \mathbf{A}^\dagger 是 \mathbf{A} 的伪逆。

例 10.4.10. 随机线性规划问题

$$\min \quad \bar{\mathbf{c}}^T \mathbf{x} + \gamma \mathbf{x}^T \Sigma \mathbf{x} = \mathbf{E}(\mathbf{c}^T \mathbf{x}) + \gamma \text{var}(\mathbf{c}^T \mathbf{x})$$

$$\text{s.t.} \quad \mathbf{Gx} \leq \mathbf{h}$$

$$\mathbf{Ax} = \mathbf{b}$$

其中 \mathbf{c} 是随机向量, 均值为 $\bar{\mathbf{c}}$, 协方差 Σ 。因此, $\mathbf{c}^T \mathbf{x}$ 是随机变量, 均值 $\bar{\mathbf{c}}^T \mathbf{x}$, 方差 $\mathbf{x}^T \Sigma \mathbf{x}$ 。 $\gamma > 0$ 为风险厌恶参数; 权衡期望损失和方差 (风险)。

二次约束二次规划 (QCQP)

若目标函数与不等式约束函数均为二次函数, 则可得到如下二次约束二次规划:

$$\min \quad (1/2) \mathbf{x}^T \mathbf{P}_0 \mathbf{x} + \mathbf{q}_0^T \mathbf{x} + r_0$$

$$\text{s.t.} \quad (1/2) \mathbf{x}^T \mathbf{P}_i \mathbf{x} + \mathbf{q}_i^T \mathbf{x} + r_i \leq 0, \quad i = 1, \dots, m$$

$$\mathbf{Ax} = \mathbf{b}$$

$\mathbf{P}_i \in S_+^n$, $i = 0, \dots, m$, 目标和限制函数都是凸二次型。如果 $\mathbf{P}_i \in S_{++}^n$, 可行域是 m 个椭球和一个仿射集合的交集。显然, 二次规划是二次约束二次规划的特例, 通过在二次规划中令 $\mathbf{P}_i = 0, i = 1, \dots, m$ 可得。

二阶锥规划 (SOCP)

一个与二次规划紧密相关的问题是二阶锥规划 (SOCP):

$$\begin{aligned} \min \quad & \mathbf{f}^T \mathbf{x} \\ \text{s.t.} \quad & \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T \mathbf{x} + d_i, \quad i = 1, \dots, m \\ & \mathbf{F} \mathbf{x} = \mathbf{g} \end{aligned} \quad (10.40)$$

其中, $\mathbf{x} \in \mathbb{R}^n$ 为优化变量, $\mathbf{A}_i \in \mathbb{R}^{n_i \times n}$ 以及 $\mathbf{F} \in \mathbb{R}^{p \times n}$ 。我们称这种形式的约束

$$\|\mathbf{A} \mathbf{x} + \mathbf{b}\|_2 \leq \mathbf{c}^T \mathbf{x} + d$$

为二阶锥约束, 其中 $\mathbf{A} \in \mathbb{R}^{k \times n}$ 。这是因为它等同于要求仿射函数 $(\mathbf{A} \mathbf{x} + \mathbf{b}, \mathbf{c}^T \mathbf{x} + d)$ 在 \mathbb{R}^{k+1} 的二阶锥中。当 $c_i = 0, i = 1, \dots, m$ 时, SOCP 等同于QCQO。类似地, 如果 $\mathbf{A}_i = 0, i = 1, \dots, m$, SOCP 退化为线性规划。

例 10.4.11. 鲁棒线性规划: 二阶锥规划的特例

优化问题中的参数经常是不确定的, 例如, 在线性规划中

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i = 1, \dots, m \end{aligned}$$

其中的参数 $\mathbf{c}, \mathbf{a}_i, b_i$ 含有一些不确定性或变化。两种通用方式处理不确定性, 为简洁起见, 假设 \mathbf{c} 和 b_i 是固定的, 且只考虑 \mathbf{a}_i 。

一种为确定性方法: 必须满足约束所有的 $\mathbf{a}_i \in \mathcal{E}_i$, 即已知 \mathbf{a}_i 在给定的椭球中:

$$\mathbf{a}_i \in \mathcal{E}_i = \{\bar{\mathbf{a}}_i + \mathbf{P}_i \mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\}$$

其中, $\bar{\mathbf{a}}_i \in \mathbb{R}^n$ 是椭球中心, 半轴 $\mathbf{P}_i \in \mathbb{R}^{n \times n}$ 的奇异向量决定。因此, 可以得到相应的鲁棒线性规划, 具体表示为

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad \forall \mathbf{a}_i \in \mathcal{E}_i, \quad i = 1, \dots, m \end{aligned} \quad (10.41)$$

对于所有 $\mathbf{a}_i \in \mathcal{E}_i$, 都有 $\mathbf{a}_i^T \mathbf{x} \leq b_i$ 成立, 这个约束条件等价于

$$\sup\{\mathbf{a}_i^T \mathbf{x} \mid \mathbf{a}_i \in \mathcal{E}_i\} \leq b_i$$

其左端可以重新写作

$$\begin{aligned} \sup\{\mathbf{a}_i^T \mathbf{x} \mid \mathbf{a}_i \in \mathcal{E}_i\} &= \bar{\mathbf{a}}_i^T \mathbf{x} + \sup\{\mathbf{u}^T \mathbf{P}_i^T \mathbf{x} \mid \|\mathbf{u}\|_2 \leq 1\} \\ &= \bar{\mathbf{a}}_i^T \mathbf{x} + \|\mathbf{P}_i^T \mathbf{x}\|_2 \end{aligned}$$

形式。因此，鲁棒线性约束可以简化为

$$\bar{\mathbf{a}}_i^T \mathbf{x} + \|\mathbf{P}_i^T \mathbf{x}\|_2 \leq b_i$$

这是一个二阶锥约束。因此，鲁棒线性规划(10.41)也可以看成是一个 *SOCPr*。

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \bar{\mathbf{a}}_i^T \mathbf{x} + \|\mathbf{P}_i^T \mathbf{x}\|_2 \leq b_i, \quad i = 1, \dots, m \end{aligned}$$

另一种为随机性方法： \mathbf{a}_i 是随机变量；以概率 η 满足约束

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{prob}(\mathbf{a}_i^T \mathbf{x} \leq b_i) \geq \eta, \quad i = 1, \dots, m \end{aligned}$$

假设 \mathbf{a}_i 是服从高斯分布，均值为 $\bar{\mathbf{a}}_i$ ，协方差阵 Σ_i ，即 $\mathbf{a}_i \sim \mathcal{N}(\bar{\mathbf{a}}_i, \Sigma_i)$ 。 $\mathbf{a}_i^T \mathbf{x}$ 服从高斯分布，均值为 $\bar{\mathbf{a}}_i^T \mathbf{x}$ ，方差为 $\mathbf{x}^T \Sigma_i \mathbf{x}$ 。因此

$$\mathbf{prob}(\mathbf{a}_i^T \mathbf{x} \leq b_i) = \phi\left(\frac{b_i - \bar{\mathbf{a}}_i^T \mathbf{x}}{\|\Sigma_i^{1/2} \mathbf{x}\|_2}\right)$$

其中 $\phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u e^{-t^2/2} dt$ 。这样便可以得到鲁棒线性规划

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{prob}(\mathbf{a}_i^T \mathbf{x} \leq b_i) \geq \eta, \quad i = 1, \dots, m, \end{aligned}$$

其中 $\eta \geq 1/2$ ，等价于 *SOCPr*

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \bar{\mathbf{a}}_i^T \mathbf{x} + \phi^{-1}(\eta) \|\Sigma_i^{1/2} \mathbf{x}\|_2 \leq b_i, \quad i = 1, \dots, m, \end{aligned}$$

半定规划 (SDP)

半定规划 (semidefinite programming, SDP) 是线性规划在矩阵空间中的一种推广，它与线性规划不同的地方是其自变量取值于半正定矩阵空间。并具有如下一般形式

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & x_1 \mathbf{F}_1 + \dots + x_n \mathbf{F}_n + \mathbf{G} \preceq 0 \\ & \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned}$$

其中 $\mathbf{G}, \mathbf{F}_1, \dots, \mathbf{F}_n$ 都是对称矩阵。如果这些矩阵为对角阵，那么上式中的线性矩阵不等式 (LMI) 等价于 n 个线性不等式，此时，SDP 便退化为线性规划。

仿照线性规划的分析，SDP 同样具有标准形式和不等式形式的半定规划。标准形式的 SDP 具有对变量 $X \in S^n$ 的线性等式约束和 (矩阵) 非负定约束：

$$\begin{aligned} \min \quad & \text{Tr}(\mathbf{C} \mathbf{X}) \\ \text{s.t.} \quad & \text{Tr}(\mathbf{A}_j \mathbf{X}) = \mathbf{b}_j, \quad j = 1, \dots, p \\ & \mathbf{X} \succeq 0 \end{aligned}$$

其中 $\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_P \in S^n$, $\text{Tr}(\cdot)$ 是迹函数。将这一形式与标准形式的线性规划进行比较, 在线性规划 (LP) 和 SDP 的标准形式中, 我们在变量的 p 个线性等式约束和变量非负约束下极小化变量的线性函数。

如同不等式形式的 LP, 不等式形式的 SDP 不含有等式的约束, 但是具有一个 LMI:

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n \preceq \mathbf{B} \end{aligned}$$

其优化变量为 $\mathbf{x} \in \mathbb{R}^n$, 参数为 $\mathbf{B}, \mathbf{A}_1, \dots, \mathbf{A}_n \in S^k$, $\mathbf{c} \in \mathbb{R}^n$ 。

例 10.4.12. 最大割问题的半定规划松弛 令 $G = \langle V, E \rangle$ 是一个无向图, 其中 V 是含有 n 个顶点的顶点集, E 表示边的集合。假定对于边 $(i, j) \in E$ 的权重为 w_{ij} 。最大割问题是找到节点集合 V 的一个子集 U , 使得 U 与它的补集 \bar{U} 之间相连边的权重之和最大化。若令 $x_i = 1, i \in U$ 和 $x_i = -1, i \in \bar{U}$, 则可得如下整数规划

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} (1 - x_i x_j) w_{ij} \\ \text{s.t.} \quad & x_i \in \{-1, 1\}, i = 1, 2, \dots, n. \end{aligned} \tag{10.42}$$

显然, 只有 x_i 与 x_j 不相等时, 即分别在集合 U 和 \bar{U} 中, 目标函数中 w_{ij} 的系数非零。该问题很难在多项式时间内找到它的最优解。接下来探讨如何将其松弛成一个半定规划问题。

令 \mathbf{W} 表示无向图的邻接矩阵, \mathbf{D} 表示该图的度矩阵, 并定义 $\mathbf{A} = -\frac{1}{4}(\mathbf{D} - \mathbf{W})$ 为图的拉普拉斯矩阵的 $-\frac{1}{4}$ 倍, 则问题(10.42)可以等价地写为

$$\begin{aligned} \min \quad & \mathbf{x}^T \mathbf{A} \mathbf{x} \\ \text{s.t.} \quad & x_i^2 = 1, i = 1, 2, \dots, n. \end{aligned} \tag{10.43}$$

现在令 $\mathbf{X} = \mathbf{x} \mathbf{x}^T$, 注意到约束条件 $x_i^2 = 1$ 。利用矩阵形式, 我们可将最大割问题化为

$$\begin{aligned} \min \quad & \langle \mathbf{A}, \mathbf{X} \rangle \\ \text{s.t.} \quad & X_{ii} = 1, i = 1, 2, \dots, n, \\ & \mathbf{X} \succeq 0, \\ & \text{rank}(\mathbf{X}) = 1. \end{aligned} \tag{10.44}$$

容易验证问题(10.43)与(10.44)是等价的。现在将问题(10.44)的约束 $\text{rank}(\mathbf{X}) = 1$ 去掉, 那么便得到最大割的半定规划松弛形式

$$\begin{aligned} \min \quad & \langle \mathbf{A}, \mathbf{X} \rangle \\ \text{s.t.} \quad & X_{ii} = 1, i = 1, 2, \dots, n, \\ & \mathbf{X} \succeq 0. \end{aligned} \tag{10.45}$$

需要声明的是问题(10.45)与原问题并不等价, 但确实能得到一个较好的近似解。

例 10.4.13. QCQP 问题的半定规划松弛 考虑二次约束二次规划问题

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & \mathbf{x}^T \mathbf{A}_0 \mathbf{x} + 2\mathbf{b}_0^T \mathbf{x} + c_0, \\ \text{s.t.} \quad & \mathbf{x}^T \mathbf{A}_i \mathbf{x} + 2\mathbf{b}_i^T \mathbf{x} + c_i \leq 0, \quad i = 1, 2, \dots, m, \end{aligned} \quad (10.46)$$

其中 \mathbf{A}_i 为 $n \times n$ 对称矩阵. 当部分 \mathbf{A}_i 为对称不定矩阵时, 问题 (10.46) 是 NP 难的非凸优化问题. 现在我们写出问题 (10.46) 的半定规划松弛问题. 对任意 $\mathbf{x} \in \mathbb{R}^n$ 以及 $\mathbf{A} \in \mathcal{S}^n$, 有恒等式

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \text{Tr}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = \text{Tr}(\mathbf{A} \mathbf{x} \mathbf{x}^T) = \langle \mathbf{A}, \mathbf{x} \mathbf{x}^T \rangle,$$

因此该优化问题中所有的二次项均可用下面的方式进行等价刻画:

$$\mathbf{x}^T \mathbf{A}_i \mathbf{x} + 2\mathbf{b}_i^T \mathbf{x} + c_i = \langle \mathbf{A}_i, \mathbf{x} \mathbf{x}^T \rangle + 2\mathbf{b}_i^T \mathbf{x} + c_i.$$

所以, 原始问题等价于

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & \langle \mathbf{A}_0, \mathbf{x} \mathbf{x}^T \rangle + 2\mathbf{b}_0^T \mathbf{x} + c_0 \\ \text{s.t.} \quad & \langle \mathbf{A}_i, \mathbf{x} \mathbf{x}^T \rangle + 2\mathbf{b}_i^T \mathbf{x} + c_i \leq 0, \quad i = 1, 2, \dots, m, \\ & \mathbf{X} = \mathbf{x} \mathbf{x}^T. \end{aligned}$$

进一步地,

$$\begin{aligned} \mathbf{x}^T \mathbf{A}_i \mathbf{x} + 2\mathbf{b}_i^T \mathbf{x} + c_i &= \left\langle \begin{pmatrix} \mathbf{A}_i & \mathbf{b}_i \\ \mathbf{b}_i^T & c_i \end{pmatrix}, \begin{pmatrix} \mathbf{X} & \mathbf{x} \\ \mathbf{x}^T & 1 \end{pmatrix} \right\rangle \\ &\stackrel{\text{def}}{=} \langle \bar{\mathbf{A}}_i, \bar{\mathbf{X}} \rangle, \quad i = 0, 1, \dots, m. \end{aligned}$$

接下来将等价问题 (10.46) 松弛为半定规划问题. 在问题 (10.46) 中, 唯一的非线性部分是约束 $\mathbf{X} = \mathbf{x} \mathbf{x}^T$, 我们将其松弛成半正定约束 $\bar{\mathbf{X}} \succeq \mathbf{x} \mathbf{x}^T$. 可以证明 $\bar{\mathbf{X}} \succeq 0$ 与 $\mathbf{X} \succeq \mathbf{x} \mathbf{x}^T$ 是等价的. 因此这个问题的半定规划松弛可以写成

$$\begin{aligned} \min \quad & \langle \bar{\mathbf{A}}_0, \bar{\mathbf{X}} \rangle \\ \text{s.t.} \quad & \langle \bar{\mathbf{A}}_i, \bar{\mathbf{X}} \rangle \leq 0, \quad i = 1, 2, \dots, m \\ & \bar{\mathbf{X}} \succeq 0 \\ & \bar{\mathbf{X}}_{n+1, n+1} = 1 \end{aligned}$$

其中“松弛”来源于我们将 $\mathbf{X} = \mathbf{x} \mathbf{x}^T$ 替换成了 $\bar{\mathbf{X}} \succeq \mathbf{x} \mathbf{x}^T$.

10.5 阅读材料

Minkowski 被认为第一个对凸集进行了系统的分析并且引入了很多基础性的概念, 例如支撑超平面, 支撑超平面定理, Minkowski 距离函数 (参见习题 3.34), 凸集的极限点等. 一些早期的综述可见 Bonnoscn 和 Fenchel 的, Eggleston 的, Klee 的, 以及 Valentine 的. 最近的一些书籍专注于凸集的几何特征, 例如 Lay 以及 Webster, Klee, Fenchel, Tikhomorov 以及 Berger 给出了非常有趣的综述, 讲述了凸性的历史及其在数学方面的应用. 与线性规划问题相关联, 对线性不

等式及多项式集合已经有了充分的讨论。关于线性不等式和线性规划的一些里程碑式的文献有：Motzkin, vonNeumann 和 Morgenstern, Kantorovich, Koopmans, 以及 Dantzig。Dantzig 中讨论了线性不等式，包括直至 1963 年前后的历史调研。

20 世纪 60 年代中在非线性规划的研究中提出了广义不等式（参见 Luenberger 的以及 Isii 的），并且被扩展到锥优化问题中。Bonnans 和 Fan 的是关于广义线性不等式集合（关于半正定锥）的一篇早期文献。对于超平面分离定理证明的推广，我们推荐读者参看 Rockafellar 的，以及 Hiriart-Urruty 和 Lemarechal 的。Dantzig 的包含了 vonNeumann 和 Morgenstern 在给出的择一定理，关于择一定理的更多文献，参见第 5 章。一些术语（包括 Pareto 最优性，有效制造，价格 A 的解释）在 Luenberger 的中进行了详尽的讨论。

凸的几何性质在经典的力矩理论（Krdn 和 Nudelman, Karlin 和 Studden 的）中有显著的作用。一个著名的例子是非负多项式与力矩锥的对偶性，参见习题 2.37。凸分析的标准参考文献是 Rockafellar。其他关于凸函数的书有 Stoer 和 Witzgall, Roberts 和 Varberg, VanTiel, Hiriart-Urruty 和 Lemarechal, Ekeland 和 Temam, Borwein 和 Lewis, Florenzano 和 LeVan, Barvinok, 以及 Bertsckas, Nedic 和 Ozdaglar。很多非线性规划的教材也有一些章节涉及凸函数（例如，Mangasarian,

Bazaraa, Sherali 和 Shetty, Bertsekas, Polyak, 以及 Peressini, Sullivan 和 Uhl），很多文献中提到了 Jensen 不等式。在不等式的一般性研究中，Jensen 不等式起到了重要的作用，Hardy, Littlewood 和 Polya 以及 Beckenbach 和 Bellman 中都涉及了不等式的研究。透视函数的概念来源于 Hiriart-Urruty 和 Lemarechal。一些定义（相对熵和 Kullback-Leibler 散度），以及相应的习题，可以参考 Cover 和 Thomas。早期的一些关于拟凸函数（以及凸性的其他扩展）的重要参考文献有 Nikaido, Mangasarian,

Arrow 和 Enthoven, Ponstein, 以及 Luenberger。更全面的此类参考文献可以参照 Bazaraa, Sherali 和 Shetty Prekopa 对对数-凹函数做了一个综述。在 Barndorff-Nielsen 中提到了 Laplace 变换的对数-凸性。关于对数-凹函数的积分的结论证明可以参看 Prekopa。广义不等式在最近的关于锥优化的参考文献中广泛使用，如 Nesterov 和 Nemirovski; Ben-Tal 和 Nemirovski 以及第 4 章最后所列的参考文献。关于广义不等式的凸性在 Luenberger 和 Isii 中亦有涉及。矩阵单调性和矩阵凸性由 Lowner 提出，Davis, Roberts 和 Varberg 以及 Marshall 和 Olkin 对其进行了详细讨论。例 3.48 中提到的函数 X^p 的凸性或者凹性的相关结论可以参看 Bondar。另一个简单的例子，函数 e^X 不是矩阵凸的，可以参看 Marshall 和 Olkin。

自 20 世纪 40 年代以来，线性规划已被广泛地研究，并且是很多极好的书的主题，包括 Dantzig 的，Luenberger 的，Schrijver 的，Papadimitriou 和 Steiglitz 的，Bertrand 和 Tsitsiklis 的，Vanderbei 的以及 Roos、Terlaky 和 Vial 的。Dantzig 和 Schrijver 也给出了线性规划的详细讨论。最近的综述参见 Todd 的。Schaible 给出了分式规划的概述，其中包含了线性分式问题及其扩展，例如凸-凹分式问题。例 4.7 中的经济增长模型出现在 vonNeumann 的文献中。关于二次规划问题的研究开始于 20 世纪 50 年代（例如，Frank 和 Wolfe 的，Markowitz 的，Hildreth 的）。其研究

的动机是第 148 页讨论的投资组合优化问题 (Markowitz 的) 和第 147 页讨论的随机损失的线性规划问题 (参见 Freund)。对于二阶锥规划的兴趣要晚一些, 是从 Nesterov 和 Nemirovski 的才开始的。关于 SOCPs 理论和应用的综述由 Alizadch 和 Goldfarb 的, Ben-Tal 和 Nemirovski 的 (在那里, 问题称为锥二次规划), 以及 Lobo、Vandenberghe、Boyd 和 Lebret 的给出。鲁棒线性规划和广义的鲁棒凸规划, 由 Ben-Tal 和 Nemirovski 的以及 ElGhaoui 和 Lebret 的提出。Goldfarb 和 Iyengar 的讨论了鲁棒 QCQPs 及其在投资优化中的应用。ElGhaoui、Oustry 和 Lebret 的则关注于鲁棒半定规划。几何规划问题自 20 世纪 60 年代起为人所知. 其在工程设计领域的应用首先由 Duffin、Peterson 和 Zener 的以及 Zener 的提出. Peterson 的以及 Ecker 的描述了七十年代取得的进展. 这些文章和书籍包括了应用在工程, 特别是在化学和土木工程中的例子。Fishburn 和 Dunlop 的, Sapatnekar、Rao、

Vaidya 和 Kang 的以及 Hershenson、Boyd 和 Lee 的将几何规划应用于集成电路的设计问题. 关于悬臂梁设计的例子 (第 156 页) 来源于 Vanderplaats. 关于 Perron-Frobenius 特征值的不同性质, Berman 和 Plemmons 在中给出了证明。Nesterov 和 Nemirovski 的引入了锥形式问题作为非线性凸优化的标准问题形式。随后 Ben-Tal 和 Nemirovski 的发展了锥规划方法, 并给出了许多应用。Alizadch 以及

Nesterov 和 Nemirovski 的首次对半定规划进行了系统的研究, 并且指出了其在凸优化领域的广泛应用。20 世纪 90 年代半定规划的持续研究受到多方面应用的激励, 如组合优化 (Goemana 和 Williamson 的), 控制 (Boyd、ElGhaoui、Feron 和 Balakrishnan 的. Scherer、Gahinet 和 Chilali 的, Dullcrud 和 Paganini 的), 通信与信号处理 (Luo 的, Davidson、Luo、Wong 和 Ma 的) 以及其他工程领域. 由 Wolkowicz、Saigal 和 Vandenberghe、Boyd 编著的书以及 Todd 的, Lewis 和 Overton 的, Vandenberghe 和 Boyd 的等文章提供了综述和扩展的文献, 关于 SDP 和矩问题的联系, 我们在第 163 页给出了一个简单的例子,

而 Bertsimas 和 Sethuraman 的, Nesterov 的及 Lasserre 的对其进行了细致的研究。最速混合 Markov 链问题来自于 Boyd、Diaconis 和 Xiao 的. 多准则问题和 Pareto 最优性是经济学的基础工具, 参见 Pareto 的, Debreu 的及 Luenberger 的. 例 4.9 的结论被称为 Gauss-Markov 定理而为人所知 (Kailath、Sayed 和 Hassibi 的)。

10.6 习题

习题 10.1. 下面的集合哪些是凸集?

- (a) 平板, 即形如 $\{x \in \mathbf{R}^n \mid \alpha \leq a^T x \leq \beta\}$ 的集合.
- (b) 矩形, 即形如 $\{x \in \mathbf{R}^n \mid \alpha_i \leq x_i \leq \beta_i, i = 1, \dots, n\}$ 的集合。当 $n > 2$ 时, 矩形有时也称为超矩形.
- (c) 楔形, 即 $\{x \in \mathbf{R}^n \mid a_1^T x \leq b_1, a_2^T x \leq b_2\}$.

(d) 距离给定点比距离给定集合近的点构成的集合, 即

$$\{x \mid \|x - x_0\|_2 \leq \|x - y\|_2, \forall y \in S\}$$

其中 $S \subseteq \mathbf{R}^n$ 。

习题 10.2. 令 $C \subseteq \mathbf{R}^n$ 为下列二次不等式的解集,

$$C = \{x \in \mathbf{R}^n \mid x^T Ax + b^T x + c \leq 0\}$$

其中 $A \in \mathbf{S}^n, b \in \mathbf{R}^n, c \in \mathbf{R}$ 。

(a) 证明: 如果 $A \succeq 0$, 那么 C 是凸集。

(b) 证明: 如果对某些 $\lambda \in \mathbf{R}$ 有 $A + \lambda gg^T \succeq 0$, 那么 C 和由 $g^T x + h = 0$ (这里 $g \neq 0$) 定义的超平面的交集是凸集。

以上命题的逆命题是否成立?

习题 10.3. 证明如果 S_1 和 S_2 是 $\mathbf{R}^{m \times n}$ 中的凸集, 那么他们的部分和

$$S = \{(x, y_1 + y_2) \mid x \in \mathbf{R}^m, y_1, y_2 \in \mathbf{R}^n, (x, y_1) \in S_1, (x, y_2) \in S_2\}$$

也是凸的。

习题 10.4. 支撑超平面

(a) 将闭凸集 $\{x \in \mathbf{R}_+^2 \mid x_1 x_2 \geq 1\}$ 表示为半空间的交集。

(b) 令 $C = \{x \in \mathbf{R}^n \mid \|x\|_\infty \leq 1\}$ 表示 \mathbf{R}^n 空间中的单位 l_∞ 范数球, 并令 \hat{x} 为 C 的边界上的点。显示地写出集合 C 在 \hat{x} 处的支撑超平面。

习题 10.5. 设 $f : \mathbf{R} \rightarrow \mathbf{R}$ 递增, 在其定义域 (a, b) 是凸函数, 令 g 表示其反函数, 即具有定义域 $(f(a), f(b))$, 且对所有 $a < x < b$ 满足 $g(f(x)) = x$ 。函数 g 是凸函数还是反函数? 为什么?

习题 10.6. 证明 $x^* = (1, 1/2, -1)$ 是如下优化问题的最优解

$$\begin{aligned} & \text{minimize} && (1/2)x^T Px + q^T x + r \\ & \text{subject to} && -1 \leq x_i \leq 1, \quad i = 1, 2, 3 \end{aligned}$$

其中

$$P = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, \quad q = \begin{bmatrix} -22.0 \\ -14.5 \\ 13.0 \end{bmatrix}, \quad r = 1$$

习题 10.7. 考虑极小化二次函数

$$f_0(\mathbf{x}) = (1/2)\mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r$$

其中, $\mathbf{P} \in \mathbf{S}_+^n$ (n 阶半正定矩阵)。给出 \mathbf{x} 为 f_0 最小解的重要条件, 并说明 \mathbf{x} 何时无解, 有唯一解, 有多个解。

习题 10.8. 计算 $f(x)$ 的共轭函数, 以及共轭函数的定义域。

- $f(x) = -\log x$
- $f(x) = e^x$

习题 10.9. 证明: *Gauss* 概率密度函数的累积分布函数

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-u^2/2} du$$

是对数-凹函数. 即 $\log(\Phi(x))$ 是凹函数。

习题 10.10. 考虑优化问题

$$\begin{aligned} & \text{minimize} && f_0(x_1, x_2) \\ & \text{subject to} && 2x_1 + x_2 \geq 1 \\ & && x_1 + 3x_2 \geq 1 \\ & && x_1 \geq 0, x_2 \geq 0 \end{aligned}$$

给出以下函数最优集和最优值

- (a) $f_0(x_1, x_2) = x_1 + x_2$
- (b) $f_0(x_1, x_2) = -x_1 - x_2$
- (c) $f_0(x_1, x_2) = x_1$
- (d) $f_0(x_1, x_2) = \max\{x_1, x_2\}$
- (e) $f_0(x_1, x_2) = x_1^2 + 9x_2^2$

10.7 参考文献

K.J.Arrow and A.C. Enthoven. Quasi-concave programming. *Econometrica*, 29(4):779-800, 1961.

F.Alizadeh and D.Goldfarb. Second-order cone programming. *Mathematical Programming Series B*, 95:3-51, 2003.

E.F.Beckenbach and R.Bellman. *Inequalities*. Springer, second edition, 1965.

S.Boyd, P.Diaconis, and L.Xiao. Fastest mixing Markov chain on a graph. *SIAM Review*, 46(4):667-689, 2004.

S.Boyd, L.KI Ghaoui, E.Feron, and V.Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics, 1994.

M.Berger. Convexity. *The American Mathematical Monthly*, 97(8):650-678, 1990.

D.P.Bertsekas. *Nonlinear Programming*. Athena Scientific, second edition, 1999.

D.P.Bortsckas. *Convex Analysis and Optimization*. Athena Scientific, 2003. With A.Nedic and A.E.Ozdaglar.

T.Bonnesen and W.Fenchel.Theorie der konvexen Korper.Chelsea Publishing Company, 1948. First published in 1934.

R.Bellman and K.Fan.On systems of linear inequalities in Hermitian matrix variables.In V.L.Klee, editor, Convexity, volume VII of Proceedings of the Symposia in Pure Mathematics pages 1-11.American Mathematical Society, 1963.

A.Ben-Israel.Linear equations and inequalities on finite dimensional, real or complex vector spaces: A unified theory.Journal of Mathematical Analysis and Applications 27:367-389, 1969.

J.M.Borwein and A.S.Lewis.Convex Analysis and Nonlinear Optimization.Springer, 2000.

O.Barndorff-Nielsen.Information and Exponential Families in Statistical Theory.John Wiley & Sons, 1978.

J.V.Bondar.Comments on and complements to Inequalities: Theory of Majorization and Its Applications.Linear Algebra and Its Applicationsy 199:115-129, 1994.

A.Berman and R.J.Plenunons.Nonnegative Matrices in the Mathematical Sciences.Society for Industrial and Applied Mathematics, 1994.First published in 1979 by Academic Press.

M.S.Bazaraa, H.D.Sherali, and C.M.Shetty.Nonlineair Programming.Theory and Algorithms.John Wiley & Sons, second edition, 1993.

D.Bertsimas and J.N. Tsitsiklis. Introduction to Linear Optimization, Athena Scientific, 1997.

A.Ben-Tal and A.Nemirovski.Robust convex optimization.Mathematics of Operations Research, 23(4):769-805, 1998.

A.Ben-Tal and A.Nemirovski.Robust solutions of uncertain linear programs.Operations Research Letters, 25(1):1-13, 1999.

A.Ben-Tal and A.Nemirovski.Lectures on Modem Convex Optimization.Analysis, Algorithms, and Engineering Applications.Society for Industrial and Applied Mathematics, 2001.

T.M.Cover and J.A.Thomas.Elements of Information Theory.John Wiley & Sons, 1991.

G.B.Dantzig.Linear Programming and Extensions.Princeton University Press, 1963.

C.Davis.Notions generalizing convexity for functions defined on spaces of matrices.In V.L.Klee, editor, Convexity, volume VII of Proceedings of the Symposia in Pure Mathematics. pages 187-201.American Mathematical Society, 1963.

G.Debreu.Theory of Value: An Axiomatic Analysis of Economic Equilibrium. Yale University Press, 1959.

T.N.Davidson, Z-Q.Luo, and K.M.Wong.Design of orthogonal pulse shapes for coinmuications via semidefinite programming.IEEE Transactions on Signal Processing, 48(5):1433-1445, 2000.

G.E.Dullcrud and F.Paganini.A Course in Robust Control Theory.A Convex Approach.Springer, 2000.

- R.J.Duffin, E.L.Peterson, and C.Zener. Geometric Programming. Theory and Applications. John Wiley & Sons, 1967.
- L.El Ghaoui and H.Lebret. Robust solutions to least-squares problems with uncertain data. SIAM Journal of Matrix Analysis and Applications, 18(4):1035- 1064, 1997.
- J.G. Ecker. Geometric programming: Methods, computations and applications. SIAM Review, 22(3):338-362, 1980.
- H.G.Eggleston. Convexity. Cambridge University Press, 1958.
- I.Ekeland and R.Temam. Convex Analysis and Variational Inequalities. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1999. Originally published in 1976.
- J.P.Fishburn and A.E.Dunlop. TILOS: A posynomial programming approach to transistor sizing. In IEEE International Conference on Computer-Aided Design: ICCAD-85. Digest of Technical Papers, pages 326 328. TEEE Computer Society Press, 1985.
- W.Fenchel. Convexity through the ages. In P.M.Gruber and J.M.Wills, editors, Convexity and Its Applications, pages 120-130. Birkhauser Verlag, 1983.
- M.Florenzano and C.Le Van. Finite Dimensional Convexity and Optimization. Number 13 in Studies in Economic Theory. Springer, 2001.
- M.Frank and P.Wolfe. An algorithm for quadratic programming. Naval Research Logistics Quarterly, 3:95-110, 1956.
- R.J.PYeund. The introduction of risk into a programming model. Econometrica,24(3):253-263, 1956.
- D.Goldfarb and G.Iyengar. Robust convex quadratically constrained programs. Mathematical Programming Series B, 97:495-515, 2003.
- D.Goldfarb and G.Iyengar. Robust portfolio selection problems. Mathematics of Operations Research, 28(1):1-38, 2003.
- M.X.Goemans and D.P.Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the Association for Computing Machinery, 42(6):1115-1145, 1995.
- M.del Mar Hershenson, S.P.Boyd, and T.H.Lee. Optimal design of a CMOS opamp via geometric programming. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 20(1):1-21, 2001.
- C.Hildreth. A quadratic programming procedure. Naval Research Logistics Quarterly, 4:79 85, 1957.
- J.-B.Hiriart-Urruty and C.Lemarechal. Convex Analysis and Minimization Algorithms. Springer, 1993. Two volumes.
- K.Isii. Inequalities of the types of ChebyKhev and Cramer-Rao and mathematical programming. Annals of The Institute of Statistical Mathematics, 16:277-293, 1964.

J.L.W. V. Jensen. Sur les fonctions convexes et les inegalites entre les valeurs moyennes. *Acta Mathematica*, 30:175-193, 1906.

L.V.Kantorovich. Mathematical methods of organizing and planning production. *Management Science*, 6(4):366-422, 1960. Translated from Russian. First published in 1939.

V.L.Klee, editor. *Convexity*, volume 7 of *Proceedings of Symposia in Pure Mathematics*. American Mathematical Society, 1963.

V.Klee. What is a convex set? *The American Mathematical Monthly*, 78(6):616- 631, 1971.

T.C.Koopmans, editor. *Activity Analysis of Production and Allocation*, volume 13 of *Cowles Commission far Research in Economics Monographs*. John Wiley & Sons, 1951.

S-Kaxlin and W.J.Studden. *Tchebycheff Systems: With Applications in Analysis and Statistics*. John Wiley & Sons, 1966.

T.Kailath, A.H.Sayed.and B.Hassibi. *Linear Estimation*. Prentice-Hall, 2000.

J.B.Lasserre. Bounds on measures satisfying moment conditions. *The Annals of Applied Probability*, 12(3):1114-1137, 2002.

S.R.Lay. *Convex Sets and Their Applications*. John Wiley & Sons, 1982.

A.S.Lewis and M.L.Overton. *Eigenvalue optimization*. *Acta Numerica*, 5:149-190, 1996.

K.Lowner. *Über monotone Matrixfunktionen*. *Mathematische Zeitschrift*, 38:177-216, 1934.

D.G.Luenberger. *Microeconomic Theory*. McGraw-Hill, 1995.

D.G.Luenberger. *Quasi-convex programming*. *SIAM Journal on Applied Mathematics*, 16(5), 1968.

D.G.Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, 1969.

D.G.Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, second edition, 1984.

Z.-Q.Luo. Applications of convex optimization in signal processing and digital communication. *Mathematical Programming Series B*, 97:177-207f 2003.

M.S.Lobo, L.Vandenberghe, S.Boyd, and H.Lebret. Applications of second-order cone progrsuning. *Linear Algebra and Its Applications*, 284:193-228, 1998.

O.Mangasarian. *Nonlinear Programming*. Society for Industrial and Applied Mathematics, 1994. First published in 1969 by McGraw-Hill.

H.Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77-91, 1952.

H.Markowitz. The optimization of a quadratic function subject to linear constraints. *Naval Research Logistics Quarterly*, 3:111-133, 1956.

W.-K.Ma, T.N.Davidson, K, M.Wong, Z.-Q.Luo, and P.-C.Ching. Quasimaximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA. *IEEE Transactions on Signal Processing*, 50:912- 922, 2002.

A.W.Marshall and I.Olkin. *Inequalities: Theory of Majorization and Its Applications*. Academic Press, 1979.

- T.Motzkin. Beitrage zur Theorie der linearen Ungleichungen. PhD thesis, University of Basel, 1933.
- Y.Nesterov. Squared functional systems and optimization problems. In J.Prenk, C.Roos, T.Terlaky, and S.Zhang, editors, High Performance Optimization Techniques, pages 405-440. Kluwer, 2000.
- H.Nikaido. On von Neumann's minimax theorem. Pacific Journal of Mathematics, 1954.
- Y.Nesterov and A.Nemirovskii. Interior-Point Polynomial Methods in Convex Programming. Society for Industrial and Applied Mathematics, 1994.
- V.Pareto. Manual of Political Economy. A.M.Kelley Publishers, 1971. Translated from the French edition. First published in Italian in 1906.
- E.L.Peterson. Geometric programming. SIAM Remew, 18(1):1-51, 1976.
- B.T.Polyak. Introduction to Optimization. Optimization Software, 1987. Translated from Russian.
- J.Ponstein. Seven kinds of convexity. SIAM Review, 9(1):115-119, 1967.
- A.Prekopa. Logarithmic concave measures with application to stochastic programming. Acta Scientiarum Mathematicarum, 32:301-315, 1971.
- A.Prekopa. On logarithmic concave measures and functions. Acta Scientiarum Mathematicarum, 34:335-343, 1973.
- C.H.Papadimitriou and K.Steiglitz. Combinatorial Optimization. Algorithms and Complexity. Dover Publications, 1998. First published in 1982 by Prentice-Hall.
- A.L.Peressini, F.E.Sullivan, and J.J.Uhl. The Mathematics of Nonlinear Programming. Undergraduate Texts in Mathematics. Springer, 1988.
- R.T.Rockafellar. Convex Analysis. Princeton University Press, 1970.
- C.Roos, T.Terlaky, and J-Ph.Vial. Theory and Algorithms for Linear Optimization. An Interior Point Approach. John Wiley & Sons, 1997.
- A.W.Roberts and D.E.Varberg. Convex Functions. Academic Press, 1973.
- S.Schaible. Bibliography in fractional programming. Zeitschrift fur Operations Research, 26:211-241, 1982.
- S.Schaible. Fractional programming. Zeitschrift fur Operations Research, 27:39-54, 1983.
- A.Schrijver. Theory of Linear and Integer Programming. John Wiley Sons, 1986.
- C.Scherer, P.Gahinet, and M.Chilali. Multiobjective output-feedback control via LMI optimization. IEEE Transactions on Automatic Control, 42(7):896-906, 1997.
- S.S.Sapatnekar, V.B.Rao, P.M.Vaidya, and S.-M.Kang. An exact solution to the transistor sizing problem for CMOS circuits using convex optimization. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 12(11):1621-1634, 1993.
- J.Stoer and C.Witzgall. Convexity and Optimization in Finite Dimensions I. Springer-Verlag, 1970.
- V.M.Tikhomorov. Convex analysis. In R.V.Gamkrelidze, editor, Analysis II: Convex Analysis and Approximation Theory, volume 14, pages 1-92. Springer, 1990.

M.J.Todd.The many facets of linear programming.Mathematical Programming Series B,91:417-436,2002.

F.A.Valentine.Convex Sets.McGraw-Hill,1964.

G.N.Vanderplaats.Numerical Optimization Techniques for Engineering Design.McGraw-Hill,1984.

R.J.Vanderbei.Linear Programming: Foundations and Extensions.Khiwer,1996.

J.von Neumann.A model of general economic equilibrium.Review of Economic Studies, 13(1):1-9, 1945-46.

J.von Neumann.Discussion of a maximum problem.In A.H.Taub, editor, John von Neumann. Collected Works, volume VI, pages 89-95.Pergamon Press,1963.Unpublished working paper from 1947.

J.von Neumann and O.Morgenstern.Theory of Games and Economic Behavior.Princeton University Press,third edition,1953.First published in 1944.

J.van Tiel.Convex Analysis.An Introductory Text John Wiley & Sons,1984.

H.Wolkowicz, R.Saigal, and L.Vandenberghe, editors.Handbook of Semidefinite Programming.Kluwer Academic Publishers, 2000.

R.Webster.Convexity.Oxford University Press,1994.

C.Zener.Engineering Design by Geometric Programming.John Wiley & Sons, 1971.

第十一章 最优性条件和对偶理论

本章介绍拉格朗日对偶函数和拉格朗日对偶问题，把标准形式（可能是非凸）的优化问题转化为对偶问题进行求解；介绍凸优化的最优性条件；介绍数据科学中各种常见的优化问题的对偶性问题。

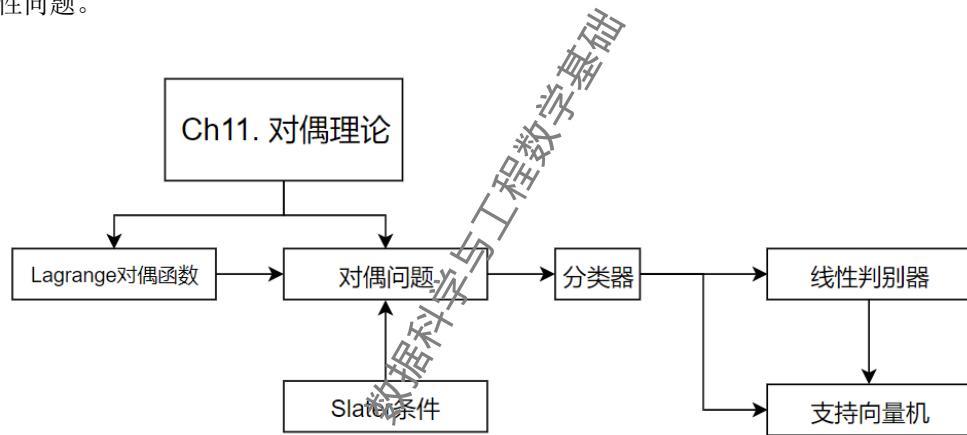


图 11.1: 本章导图

我觉得这张导图里面缺少 KKT 条件，在对偶问题向右箭头到 KKT 条件，然后再箭头指向分类器，这样会比较好

11.1 无约束优化的最优性条件

求解优化问题需要清楚最优解应当满足何种条件，即最优性条件。通过最优性条件，将可获得解析解求解的表达式，亦或者是数值解的迭代求解方法。先考虑如下无约束优化问题：

$$\min f(\mathbf{x}) \quad (11.1)$$

无约束可微问题的最优化条件

根据多元函数微积分的知识, 可知最优解处的一阶必要条件为:

定理 11.1.1. 假设 f 在全空间 \mathbb{R}^n 可微。若 \mathbf{x}^* 是一个局部极小解, 那么

$$\nabla f(\mathbf{x}^*) = 0 \quad (11.2)$$

证明. 任取 $\mathbf{v} \in \mathbb{R}^n$, 考虑 f 在点 $\mathbf{x} = \mathbf{x}^*$ 处的泰勒展开

$$f(\mathbf{x}^* + t\mathbf{v}) = f(\mathbf{x}^*) + t\mathbf{v}^T \nabla f(\mathbf{x}^*) + o(t),$$

整理得

$$\frac{f(\mathbf{x}^* + t\mathbf{v}) - f(\mathbf{x}^*)}{t} = \mathbf{v}^T \nabla f(\mathbf{x}^*) + o(1).$$

根据 \mathbf{x}^* 的最优化, 在上式中分别对 t 取点 0 处的左、右极限可知

$$\lim_{t \rightarrow 0^+} \frac{f(\mathbf{x}^* + t\mathbf{v}) - f(\mathbf{x}^*)}{t} = \mathbf{v}^T \nabla f(\mathbf{x}^*) \geq 0,$$

$$\lim_{t \rightarrow 0^-} \frac{f(\mathbf{x}^* + t\mathbf{v}) - f(\mathbf{x}^*)}{t} = \mathbf{v}^T \nabla f(\mathbf{x}^*) \leq 0,$$

即对任意的 \mathbf{v} 有 $\mathbf{v}^T \nabla f(\mathbf{x}^*) = 0$, 由 \mathbf{v} 的任意性知 $\nabla f(\mathbf{x}^*) = 0$.

□

注意, 上面的条件仅仅是必要的. 对于 $f(x) = x^3$, $x \in \mathbb{R}$, 满足 $f'(x) = 0$ 的点为 $x^* = 0$, 但其不是一个局部最优解. 实际上, 我们称满足 $\nabla f(x) = 0$ 的点 x 为 f 的稳定点 (有时也称为驻点或临界点). 可以看出, 除了一阶必要条件, 还需要对函数加一些额外的限制条件, 才能保证最优解的充分性. 我们会在后面的小节中继续讨论.

如果一阶必要条件满足, 我们仍然不能确定当前点是否是一个局部极小点. 这里考虑使用二阶信息来进一步判断给定点的最优化. 实际上, 若函数具有二阶连续可微的性质, 则根据多元函数的泰勒展开可得如下定理:

定理 11.1.2. 假设 f 在点 \mathbf{x}^* 的一个开邻域内是二阶连续可微的, 则以下最优化条件成立:

- 二阶必要条件: 如果 \mathbf{x}^* 是 f 的一个局部极小点, 那么

$$\nabla f(\mathbf{x}^*) = 0, \quad \nabla^2 f(\mathbf{x}^*) \succeq 0.$$

- 二阶充分条件: 如果在点 \mathbf{x}^* 处, 有

$$\nabla f(\mathbf{x}^*) = 0, \quad \nabla^2 f(\mathbf{x}^*) \succ 0$$

成立, 那么 \mathbf{x}^* 是 f 的一个局部极小点。

证明. 考虑 $f(\mathbf{x})$ 在点 \mathbf{x}^* 处的二阶泰勒展开,

$$f(\mathbf{x} + \mathbf{d}) = f(\mathbf{x}) + \frac{1}{2} \mathbf{d}^T \nabla^2 f(\mathbf{x}) \mathbf{d} + o(\|\mathbf{d}\|^2)$$

这里因为一阶必要条件成立, 所以 $\nabla f(\mathbf{x}^*) = \mathbf{0}$. 反设 $\nabla^2 f(\mathbf{x}^*) \succeq \mathbf{0}$ 不成立, 即 $\nabla^2 f(\mathbf{x}^*)$ 有负的特征值. 取 \mathbf{d} 为其负特征值 λ_- 对应的特征向量, 通过对上式变形得到

$$\frac{f(\mathbf{x}^* + \mathbf{d}) - f(\mathbf{x}^*)}{\|\mathbf{d}\|^2} = \frac{1}{2} \frac{\mathbf{d}^T}{\|\mathbf{d}\|} \nabla^2 f(\mathbf{x}^*) \frac{\mathbf{d}}{\|\mathbf{d}\|} + o(1).$$

这里注意 $\frac{\mathbf{d}}{\|\mathbf{d}\|}$ 是 \mathbf{d} 的单位化, 因此

$$\frac{f(\mathbf{x}^* + \mathbf{d}) - f(\mathbf{x}^*)}{\|\mathbf{d}\|^2} = \frac{1}{2} \lambda_- + o(1).$$

当 $\|\mathbf{d}\|$ 充分小时, $f(\mathbf{x}^* + \mathbf{d}) < f(\mathbf{x}^*)$, 这和点 \mathbf{x}^* 的最优化矛盾. 因此二阶必要条件成立. 当 $\nabla^2 f(\mathbf{x}) \succ \mathbf{0}$ 时, 对任意的 $\mathbf{d} \neq \mathbf{0}$ 有 $\mathbf{d}^T \nabla^2 f(\mathbf{x}^*) \mathbf{d} \geq \lambda_{\min} \|\mathbf{d}\|^2 > 0$, 这里 $\lambda_{\min} > 0$ 是 $\nabla^2 f(\mathbf{x}^*)$ 的最小特征值. 因此我们有

$$\frac{f(\mathbf{x}^* + \mathbf{d}) - f(\mathbf{x}^*)}{\|\mathbf{d}\|^2} \geq \frac{1}{2} \lambda_{\min} + o(1).$$

当 $\|\mathbf{d}\|$ 充分小时有 $f(\mathbf{x}^* + \mathbf{d}) \geq f(\mathbf{x}^*)$, 即二阶充分条件成立. \square

我们以线性最小二乘问题为例来说明其最优化条件的具体形式.

例 11.1.1. 线性最小二乘问题可以表示为

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{2} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2,$$

其中 $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$ 分别是给定的矩阵和向量. 易知 $f(\mathbf{x})$ 是可微且凸的, 因此, \mathbf{x}^* 为一个全局最优解当且仅当

$$\nabla f(\mathbf{x}^*) = \mathbf{A}^T (\mathbf{A}\mathbf{x}^* - \mathbf{b}) = \mathbf{0}.$$

因此, 线性最小二乘问题本质上等于求解线性方程组, 可以利用数值代数知识对其有效求解.

无约束不可微优化的最优化条件

本节仍考虑问题:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}),$$

但其中 $f(\mathbf{x})$ 为不可微函数. 很多实际问题的目标函数不是光滑的, 例如 $f(\mathbf{x}) = \|\mathbf{x}\|_1$. 对于此类问题, 由于目标函数可能不存在梯度和海瑟矩阵, 因此上一小节中的一阶和二阶条件不适用. 此时我们必须使用其他最优化条件来判断不可微问题的最优点.

优化问题一阶充要条件 对于目标函数是凸函数的情形, 我们已经引入了次梯度的概念并给出了其计算法则. 一个自然的问题是: 可以利用次梯度代替梯度来构造最优化条件吗? 实际上有如下定理:

定理 11.1.3. 假设 f 是适当且凸的函数, 则 \mathbf{x}^* 为无约束优化问题的一个全局极小点当且仅当

$$0 \in \partial f(\mathbf{x}^*)$$

证明. 先证必要性. 因为 \mathbf{x}^* 为全局极小点, 所以

$$f(\mathbf{y}) \geq f(\mathbf{x}^*) = f(\mathbf{x}^*) + \mathbf{0}^T(\mathbf{y} - \mathbf{x}^*), \quad \forall \mathbf{y} \in \mathbb{R}^n.$$

因此, $\mathbf{0} \in \partial f(\mathbf{x}^*)$. 再证充分性. 如果 $\mathbf{0} \in \partial f(\mathbf{x}^*)$, 那么根据次梯度的定义

$$f(\mathbf{y}) \geq f(\mathbf{x}^*) + \mathbf{0}^T(\mathbf{y} - \mathbf{x}^*) = f(\mathbf{x}^*), \quad \forall \mathbf{y} \in \mathbb{R}^n.$$

因而 \mathbf{x}^* 为一个全局极小点. \square

这说明条件 $\mathbf{0} \in \partial f(\mathbf{x}^*)$ 是 \mathbf{x}^* 为全局最优解的充要条件. 这个结论比前面的一阶条件要强, 其原因是凸问题有非常好的性质.

复合优化问题的一阶必要条件 在实际问题中, 目标函数不一定是凸函数, 但它可以写成一个光滑函数与一个非光滑凸函数的和. 在压缩感知中, 我们使用 ℓ_1 范数来获得信号的稀疏性; 再比如在机器学习中使用 ℓ_1 正则化; 还有经典的 LASSO 回归问题. 这时我们需要考虑复合优化问题

$$\min_{\mathbf{x} \in \mathbb{R}^n} \psi(\mathbf{x}) \stackrel{\text{def}}{=} f(\mathbf{x}) + h(\mathbf{x}), \quad (11.3)$$

其中 f 为光滑函数 (可能非凸), h 为凸函数 (可能非光滑). 对于其任何局部最优解, 我们给出如下一阶必要条件:

定理 11.1.4. 令 \mathbf{x}^* 为问题 (11.3) 的一个局部极小点, 那么

$$-\nabla f(\mathbf{x}^*) \in \partial h(\mathbf{x}^*),$$

其中 $\partial h(\mathbf{x}^*)$ 为凸函数 h 在点 \mathbf{x}^* 处的次梯度集合.

证明. 因为 \mathbf{x}^* 为一个局部极小点, 所以对任意单位向量 $\mathbf{d} \in \mathbb{R}^n$ 和足够小的 $t > 0$,

$$f(\mathbf{x}^* + t\mathbf{d}) + h(\mathbf{x}^* + t\mathbf{d}) \geq f(\mathbf{x}^*) + h(\mathbf{x}^*).$$

给定任一方向 $\mathbf{d} \in \mathbb{R}^n$, 其中 $\|\mathbf{d}\| = 1$. 因为对光滑函数和凸函数都可以考虑方向导数, 根据方向导数的定义,

$$\begin{aligned} \psi'(\mathbf{x}^*; \mathbf{d}) &= \lim_{t \rightarrow 0^+} \frac{\psi(\mathbf{x}^* + t\mathbf{d}) - \psi(\mathbf{x}^*)}{t} \\ &= \nabla f(\mathbf{x}^*)^T \mathbf{d} + \partial h(\mathbf{x}^*; \mathbf{d}) \\ &= \nabla f(\mathbf{x}^*)^T \mathbf{d} + \sup_{\theta \in \partial h(\mathbf{x}^*)} \theta^T \mathbf{d}, \end{aligned}$$

其中 $\partial h(\mathbf{x}^*; \mathbf{d})$ 表示凸函数 $h(\mathbf{x})$ 在点 \mathbf{x}^* 处的方向导数, 最后一个等式利用了凸函数方向导数和次梯度的关系. 现在用反证法证明我们所需要的结论. 反设 $-\nabla f(\mathbf{x}^*) \notin \partial h(\mathbf{x}^*)$, 根据次梯度的性质可知 $\partial h(\mathbf{x}^*)$ 是有界闭凸集, 又根据严格分离定理, 存在 $\mathbf{d} \in \mathbb{R}^n$ 以及常数 b 使得

$$\theta^T \mathbf{d} < b < -\nabla f(\mathbf{x}^*)^T \mathbf{d}, \quad \forall \theta \in \partial h(\mathbf{x}^*).$$

根据 $\partial h(\mathbf{x}^*)$ 是有界闭集可知对此方向 \mathbf{d} ,

$$\psi'(\mathbf{x}^*; \mathbf{d}) = \nabla f(\mathbf{x}^*)^T \mathbf{d} + \sup_{\theta \in \partial h(\mathbf{x}^*)} \theta^T \mathbf{d} < 0.$$

这说明对充分小的非负实数 t ,

$$\psi(\mathbf{x}^* + t\mathbf{d}) < \psi(\mathbf{x}^*).$$

这与 \mathbf{x}^* 的局部极小性矛盾. 因此 $-\nabla f(\mathbf{x}^*) \in \partial h(\mathbf{x}^*)$. \square

该定理给出了当目标函数一部分是非光滑凸函数时的一阶必要条件. 在这里注意, 由于目标函数可能是整体非凸的, 因此一般没有一阶充分条件.

例 11.1.2. 我们以 ℓ_1 范数正则化的优化问题为例, 给出其最优解的最优化条件. 前面我们已经介绍其一般形式可以写成

$$\min_{\mathbf{x} \in \mathbb{R}^n} \psi(\mathbf{x}) \stackrel{\text{def}}{=} f(\mathbf{x}) + \mu \|\mathbf{x}\|_1,$$

其中 $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ 为光滑函数, 正则系数 $\mu > 0$ 用来调节解的稀疏度.

尽管 $\|\mathbf{x}\|_1$ 不是可微的, 但我们可以计算其次微分, 在次梯度计算的例子中, 我们已经计算出

$$\partial_i \|\mathbf{x}\|_1 = \begin{cases} \{1\}, & x_i > 0, \\ [-1, 1], & x_i = 0, \\ \{-1\}, & x_i < 0. \end{cases}$$

因此, 如果 \mathbf{x}^* 是优化问题的一个局部最优解, 那么其满足

$$-\nabla f(\mathbf{x}^*) \in \mu \partial \|\mathbf{x}^*\|_1,$$

即

$$\nabla_i f(\mathbf{x}^*) = \begin{cases} -\mu, & x_i^* > 0, \\ a \in [-\mu, \mu], & x_i^* = 0, \\ \mu, & x_i^* < 0. \end{cases}$$

进一步地, 如果 $f(\mathbf{x})$ 是凸的 (比如在 LASSO 回归中 $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$), 那么满足上式的 \mathbf{x}^* 就是全局最优解.

11.2 Lagrange 对偶函数

11.2.1 Lagrange 函数与对偶函数

Lagrange 函数

现考虑标准形式的约束优化问题:

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p \end{aligned} \tag{11.4}$$

其中自变量 $\mathbf{x} \in \mathbb{R}^n$, 假设定义域 $\mathcal{D} = \bigcap_{i=0}^m \mathbf{dom} f_i \cap \bigcap_{j=1}^p \mathbf{dom} h_j$ 是非空集合。我们亦称该问题为原问题。注意, 这里并没有假设问题(11.4)是凸优化问题。约束优化问题在实际问题中或机器学习领域中更为常见。例如:

例 11.2.1. 最大割问题:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{x}^T \mathbf{W} \mathbf{x} \\ \text{s.t.} \quad & x_i^2 = 1, \quad i = 1, \dots, n \end{aligned}$$

其中, $\mathbf{W} \in \mathbb{S}^n$ 。

例 11.2.2. 支持向量机:

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, N \end{aligned}$$

约束优化问题的最优性理论条件相比于无约束优化问题更为复杂。通常, 我们将其约束条件追加到目标函数中, 从而将约束优化转化为无约束优化的方式进行分析。这也即下面将要介绍的 Lagrange 对偶函数和对偶问题。引入对偶问题不仅利于分析约束优化问题的最优性条件, 而且在后面我们还可以发现这将便于将非凸的优化问题转化为凸优化问题进行求解。

定义 11.2.1. 定义问题(11.4)的 **Lagrange** 函数 $L : \mathcal{D} \times \mathbb{R}_+^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ 为

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^p \nu_j h_j(\mathbf{x})$$

其中定义域为 $\mathbf{dom} L = \mathcal{D} \times \mathbb{R}_+^m \times \mathbb{R}^p$ 。 λ_i 是第 i 个不等式约束 $f_i(\mathbf{x}) \leq 0$ 的 **Lagrange** 乘子; 类似地, ν_j 是第 j 个等式约束 $h_j(\mathbf{x}) = 0$ 对应的 **Lagrange** 乘子。向量 $\boldsymbol{\lambda}$ 和 $\boldsymbol{\nu}$ 称为问题(11.4)的对偶变量或者 **Lagrange** 乘子向量。可以看出拉格朗日函数, 即为原问题的目标函数添加约束条件的加权和, 得到增广的目标函数。

Lagrange 对偶函数

定义 11.2.2. 定义 **Lagrange** 对偶函数(或对偶函数) $g : \mathbb{R}_+^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ 是拉格朗日函数关于 \mathbf{x} 取得的下确界: 即对 $\boldsymbol{\lambda} \in \mathbb{R}_+^m, \boldsymbol{\nu} \in \mathbb{R}^p$, 有

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} \left(f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^p \nu_j h_j(\mathbf{x}) \right)$$

如果 Lagrange 函数关于 \mathbf{x} 无下界, 则对偶函数取值为 $-\infty$ 。因为对偶函数是一族关于 $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ 的仿射函数的逐点下确界, 所以即使原问题(11.4)不是凸的, 对偶函数也是凹函数。

最优值的下界

定理 11.2.1. 对偶函数构成了原问题(11.4)最优值 p^* 的下界: 即对任意 $\lambda \geq \mathbf{0}$ 和 ν , 下式成立

$$g(\lambda, \nu) \leq p^* \quad (11.5)$$

证明. 设 $\tilde{\mathbf{x}}$ 是原问题 ((11.4)) 的一个可行点, 即 $f_i(\tilde{\mathbf{x}}) \leq 0$ 且 $h_i(\tilde{\mathbf{x}}) = 0$ 。根据假设, $\lambda \geq \mathbf{0}$, 有

$$\sum_{i=1}^m \lambda_i f_i(\tilde{\mathbf{x}}) + \sum_{j=1}^p \nu_j h_j(\tilde{\mathbf{x}}) \leq 0$$

左边第一项非正, 第二项为零。

根据上述不等式, 有

$$L(\tilde{\mathbf{x}}, \lambda, \nu) = f_0(\tilde{\mathbf{x}}) + \sum_{i=1}^m \lambda_i f_i(\tilde{\mathbf{x}}) + \sum_{j=1}^p \nu_j h_j(\tilde{\mathbf{x}}) \leq f_0(\tilde{\mathbf{x}}).$$

因此

$$g(\lambda, \nu) = \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \nu) \leq L(\tilde{\mathbf{x}}, \lambda, \nu) \leq f_0(\tilde{\mathbf{x}})$$

由于每一个可行点 $\tilde{\mathbf{x}}$ 都满足 $g(\lambda, \nu) \leq f_0(\tilde{\mathbf{x}})$, 因此 $g(\lambda, \nu) \leq p^*$ 成立。 \square

虽然不等式(11.5)成立, 但是当 $g(\lambda, \nu) = -\infty$ 时, 其意义不大。只有当 $\lambda \geq \mathbf{0}$, 且 $(\lambda, \nu) \in \text{dom } g$, 即 $g(\lambda, \nu) > -\infty$ 时, 对偶函数才能给出 p^* 的一个非平凡下界。称满足条件 $\lambda \geq \mathbf{0}$ 和 $(\lambda, \nu) \in \text{dom } g$ 的 (λ, ν) 是对偶可行的。

11.2.2 常见优化问题目标函数的对偶函数

线性方程组的最小二乘解

考虑问题

$$\begin{aligned} \min \quad & \mathbf{x}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned} \quad (11.6)$$

其中 $\mathbf{A} \in \mathbb{R}^{p \times n}$ 。这个问题没有不等式约束, 只有 p 个等式约束。它的 Lagrange 函数表示为

$$L(\mathbf{x}, \nu) = \mathbf{x}^T \mathbf{x} + \nu^T (\mathbf{A}\mathbf{x} - \mathbf{b}),$$

其定义域为 $\mathbb{R}^n \times \mathbb{R}^p$ 。它的对偶函数是 $g(\nu) = \inf_{\mathbf{x}} L(\mathbf{x}, \nu)$ 。因为 $L(\mathbf{x}, \nu)$ 是关于 \mathbf{x} 的二次凸函数, 可以通过求解如下最优化条件得到函数的最小值,

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \nu) = 2\mathbf{x} + \mathbf{A}^T \nu = \mathbf{0} \implies \mathbf{x} = -(1/2)\mathbf{A}^T \nu$$

在点 $\mathbf{x} = -(1/2)\mathbf{A}^T \nu$ 处, Lagrange 函数达到最小值。此时, 对偶函数为

$$g(\nu) = L((-(1/2)\mathbf{A}^T \nu), \nu) = -(1/4)\nu^T \mathbf{A} \mathbf{A}^T \nu - \mathbf{b}^T \nu$$

它是一个二次凹函数, 定义域为 \mathbb{R}^p 。根据对偶函数是原问题最优值的下界这一性质可知, 对任意 $\nu \in \mathbb{R}^p$, 都有

$$p^* \geq -(1/4)\nu^T \mathbf{A} \mathbf{A}^T \nu - \mathbf{b}^T \nu$$

标准形式的线性规划

考虑标准形式的线性规划问题

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{11.7}$$

为了推导 Lagrange 函数, 对 n 个不等式约束引入 Lagrange 乘子 λ_i , 对等式约束引入 Lagrange 乘子 ν_i , 则有

$$\begin{aligned} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) &= \mathbf{c}^T \mathbf{x} - \sum_{i=1}^n \lambda_i x_i + \boldsymbol{\nu}^T (\mathbf{A} \mathbf{x} - \mathbf{b}) \\ &= -\mathbf{b}^T \boldsymbol{\nu} + (\mathbf{c} + \mathbf{A}^T \boldsymbol{\nu} - \boldsymbol{\lambda})^T \mathbf{x} \end{aligned}$$

对偶函数为

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\nu} + \inf_{\mathbf{x}} (\mathbf{c} + \mathbf{A}^T \boldsymbol{\nu} - \boldsymbol{\lambda})^T \mathbf{x}$$

可以很容易确定对偶函数的解析表达式, 因为线性函数只有恒为零时才有下界。因此 $\mathbf{c} + \mathbf{A}^T \boldsymbol{\nu} - \boldsymbol{\lambda} = \mathbf{0}$ 时, $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\nu}$, 其余情况下 $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = -\infty$, 即

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \begin{cases} -\mathbf{b}^T \boldsymbol{\nu} & \mathbf{A}^T \boldsymbol{\nu} - \boldsymbol{\lambda} + \mathbf{c} = \mathbf{0} \\ -\infty & \text{其他情况} \end{cases}$$

注意到对偶函数 g 只有在 $\mathbb{R}^m \times \mathbb{R}^p$ 上的一个正常仿射子集上才是有限值。后面我们将会看到这是一种常见的情况。

只有当 $\boldsymbol{\lambda}, \boldsymbol{\nu}$ 满足 $\boldsymbol{\lambda} \geq \mathbf{0}$ 和 $\boldsymbol{\nu} - \mathbf{A}^T \mathbf{c} = \mathbf{0}$ 时, 下界性质(11.5)才是非平凡的, 在此情形下, $-\mathbf{b}^T \boldsymbol{\nu}$ 给出了线性规划问题(11.7)最优值的一个下界。

双向划分问题

$$\begin{aligned} \min \quad & \mathbf{x}^T \mathbf{W} \mathbf{x} \\ \text{s.t.} \quad & x_i^2 = 1, \quad i = 1, \dots, n \end{aligned} \tag{11.8}$$

其中, $\mathbf{W} \in S^n$ 。约束条件要求 x_i 的值为 1 或者 -1, 所以原问题等价于寻找这样的向量, 其分量 ± 1 , 并使 $\mathbf{x}^T \mathbf{W} \mathbf{x}$ 最小。可行集是有限的, 包含 2^n 个离散点, 所以此问题本质上可以通过遍历所有可行点来求得最小值。然而, 可行点的数量是指数增长的。所以, 只有当问题规模较小(比如 $n \leq 30$)时, 遍历法才是可行的。一般而言, 问题(11.8)很难求解。

可以将问题(11.8)看成 n 个元素的集合 $\{1, \dots, n\}$ 上的双向划分问题, 对任意可行点 \mathbf{x} , 其对应的划分为

$$\{1, \dots, n\} = \{i \mid x_i = -1\} \cup \{i \mid x_i = 1\}$$

矩阵系数 W_{ij} 是将 i, j 置于同一分区内的成本; $-W_{ij}$ 可以看成分量 i 和 j 在不同分区内的成本。问题(11.8)中的目标函数是考虑分量间所有配对的成本, 因此问题(11.8)也即寻找使得总成本最小的划分。

下面来推导此问题的对偶函数。Lagrange 函数为

$$\begin{aligned} L(\mathbf{x}, \boldsymbol{\nu}) &= \mathbf{x}^T \mathbf{W} \mathbf{x} + \sum_{i=1}^n \nu_i (x_i^2 - 1) \\ &= \mathbf{x}^T (\mathbf{W} + \text{diag}(\boldsymbol{\nu})) \mathbf{x} - \mathbf{1}^T \boldsymbol{\nu} \end{aligned}$$

对 \mathbf{x} 求极小得到 Lagrange 对偶函数

$$\begin{aligned} g(\boldsymbol{\nu}) &= \inf_{\mathbf{x}} \mathbf{x}^T (\mathbf{W} + \text{diag}(\boldsymbol{\nu})) \mathbf{x} - \mathbf{1}^T \boldsymbol{\nu} \\ &= \begin{cases} -\mathbf{1}^T \boldsymbol{\nu} & \mathbf{W} + \text{diag}(\boldsymbol{\nu}) \succeq \mathbf{0} \\ -\infty & \text{其他情况} \end{cases} \end{aligned}$$

对偶函数构成了问题(11.8)的最优值的一个下界。例如, 令对偶变量取值为

$$\boldsymbol{\nu} = -\lambda_{\min}(\mathbf{W}) \mathbf{I}$$

上述取值是对偶可行的, 这是因为

$$\mathbf{W} + \text{diag}(\boldsymbol{\nu}) = \mathbf{W} - \lambda_{\min}(\mathbf{W}) \mathbf{I} \succeq \mathbf{0}$$

由此得到了最优值 p^* 的一个下界

$$p^* \geq -\mathbf{1}^T \boldsymbol{\nu} = n \lambda_{\min}(\mathbf{W}) \quad (11.9)$$

11.2.3 Lagrange 对偶函数与共轭函数的联系

Lagrange 对偶函数与共轭函数的联系

上一章已经介绍函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 的共轭函数 f^* 定义为

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } f} (\mathbf{y}^T \mathbf{x} - f(\mathbf{x}))$$

从表达式中可以看出 Lagrange 对偶函数和共轭函数具有形式上的相似性。事实上, Lagrange 对偶函数和共轭函数紧密相关。下面简单地说明一下它们之间的联系, 考虑问题

$$\min f_0(\mathbf{x})$$

$$\text{s.t. } \mathbf{x} = \mathbf{0}$$

上述问题的 Lagrange 函数为 $L(\mathbf{x}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \boldsymbol{\nu}^T \mathbf{x}$, 其对偶函数为

$$g(\boldsymbol{\nu}) = \inf_{\mathbf{x}} (f_0(\mathbf{x}) + \boldsymbol{\nu}^T \mathbf{x}) = -\sup_{\mathbf{x}} ((-\boldsymbol{\nu})^T \mathbf{x} - f_0(\mathbf{x})) = -f_0^*(-\boldsymbol{\nu})$$

更一般地, 考虑一个优化问题, 其具有线性不等式以及等式约束,

$$\min f_0(\mathbf{x})$$

$$\text{s.t. } \mathbf{A} \mathbf{x} \leq \mathbf{b}$$

$$(11.10)$$

$$\mathbf{C} \mathbf{x} = \mathbf{d}$$

利用函数 f_0 的共轭函数, 我们可以将问题(11.10) 的对偶函数表述为

$$\begin{aligned} g(\lambda, \nu) &= \inf_x (f_0(x) + \lambda^T(Ax - b) + \nu^T(Cx - d)) \\ &= -b^T\lambda - d^T\nu + \inf_x (f_0(x) + (A^T\lambda + C^T\nu)^T x) \\ &= -b^T\lambda - d^T\nu - f_0^*(-A^T\lambda - C^T\nu) \end{aligned} \quad (11.11)$$

函数 g 的定义域也可以由函数 f_0^* 的定义域得到,

$$\mathbf{dom} g = \{(\lambda, \nu) \mid -A^T\lambda - C^T\nu \in \mathbf{dom} f_0^*\}$$

因此, Lagrange 对偶函数可以用共轭函数来表示。

利用共轭函数计算 Lagrange 对偶函数

在第 10 章的内容中, 已经计算过许多函数的共轭函数。因此, 根据这里对偶函数与共轭函数的关系, 可以利用前面共轭函数的结论, 直接求得对偶函数。

例 11.2.3. 考虑问题

$$\begin{aligned} \min \quad & f_0(x) = \|x\| \\ \text{s.t.} \quad & Ax = b \end{aligned} \quad (11.12)$$

其中, $\|\cdot\|$ 是任意范数。函数 $f_0 = \|\cdot\|$ 的共轭函数为

$$f_0^*(y) = \begin{cases} 0 & \|y\|_* \leq 1 \\ \infty & \text{其他情况} \end{cases} \quad (11.13)$$

可以看出此函数是对偶范数单位球的亲性函数。

利用上面 Lagrange 对偶函数与共轭函数的联系(11.11), 可以得到问题(11.12)的对偶函数

$$g(\nu) = -b^T\nu - f_0^*(-A^T\nu) = \begin{cases} -b^T\nu & \|A^T\nu\|_* \leq 1 \\ -\infty & \text{其他情况} \end{cases}$$

例 11.2.4. 考虑熵的最大化问题

$$\begin{aligned} \min \quad & f_0(x) = \sum_{i=1}^n x_i \log x_i \\ \text{s.t.} \quad & Ax \leq b \\ & \mathbf{1}^T x = 1 \end{aligned} \quad (11.14)$$

其中, $\mathbf{dom} f_0 = \mathbb{R}_{++}^n$ 。关于实变量 x 的负熵函数 $x \log x$ 的共轭函数是 e^{y-1} 。由于函数 f_0 是不同变量的负熵函数的和, 其共轭函数为

$$f_0^*(y) = \sum_{i=1}^n e^{y_i - 1}$$

其定义域为 $\text{dom } f_0^* = \mathbb{R}^n$ 。根据结论(11.11), 问题(11.14)的对偶函数为

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\lambda} - \nu - \sum_{i=1}^n e^{-\mathbf{a}_i^T \boldsymbol{\lambda} - \nu - 1} = -\mathbf{b}^T \boldsymbol{\lambda} - \nu - e^{-\nu - 1} \sum_{i=1}^n e^{-\mathbf{a}_i^T \boldsymbol{\lambda}}$$

其中 \mathbf{a}_i 是矩阵 \mathbf{A} 的第 i 列向量。

11.3 Lagrange 对偶问题

11.3.1 Lagrange 对偶问题

对于任意一组 $(\boldsymbol{\lambda}, \boldsymbol{\nu})$, 其中 $\boldsymbol{\lambda} \geq \mathbf{0}$, Lagrange 对偶函数给出了优化问题(11.4)的最优值 p^* 的一个下界。因此, 我们可以得到和参数 $\boldsymbol{\lambda}$ 、 $\boldsymbol{\nu}$ 相关的一个下界。一个自然的问题是: 从 Lagrange 函数能够得到的最好下界是什么? 为了研究这个问题, 本节引入了如下优化问题:

定义 11.3.1. 定义问题(11.4)的 **Lagrange 对偶问题**:

$$\begin{aligned} \max \quad & g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned} \tag{11.15}$$

在本书中, 原始问题(11.4)有时被称为原问题。前面提到的对偶可行的概念, 即描述满足 $\boldsymbol{\lambda} \geq \mathbf{0}$ 和 $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) > -\infty$ 的一组 $(\boldsymbol{\lambda}, \boldsymbol{\nu})$, 此时具有意义。它意味着, 这样的一组 $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ 是对偶问题(11.15)的一个可行解。称解 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 是对偶最优解或者是最优 Lagrange 乘子, 如果它是对偶问题(11.15)的最优解。

Lagrange 对偶问题(11.15)是一个凸优化问题, 这是因为目标函数是凹函数, 且约束集合是凸集, 因此, 对偶问题的凸性和原问题(11.4)是否是凸优化问题无关。

在将原问题转化为对偶问题时, 有时可通过显示表达对偶约束来进行。对偶函数的定义域

$$\text{dom } g = \{(\boldsymbol{\lambda}, \boldsymbol{\nu}) \mid g(\boldsymbol{\lambda}, \boldsymbol{\nu}) > -\infty\}$$

的维数一般都小于 $m + p$ 。事实上, 很多情况下, 我们可以求出 $\text{dom } g$ 的仿射包并将其表示为一系列线性等式约束, 也就是说, 我们可以识别出对偶问题(11.15)的目标函数 g 所“隐含”的等式约束。这样处理之后就可以得到一个等价问题, 在等价问题中, 这些等式约束都被显式地表达为优化问题的约束条件。接下来, 通过以下两个例子来具体说明如何用显示表达对偶约束。

例 11.3.1. 标准形式线性规划

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{11.16}$$

的 Lagrange 对偶函数为

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \begin{cases} -\mathbf{b}^T \boldsymbol{\nu} & \mathbf{A}^T \boldsymbol{\nu} - \boldsymbol{\lambda} + \mathbf{c} = \mathbf{0} \\ -\infty & \text{其他情况} \end{cases}$$

它的对偶问题是在满足约束 $\lambda \geq 0$ 的条件下, 极大化对偶函数 g , 即

$$\max g(\lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{其他情况} \end{cases} \quad (11.17)$$

$$\text{s.t. } \lambda \geq 0$$

当且仅当 $A^T \nu - \lambda + c = 0$ 时, 对偶函数 g 有界。因此, 可以通过将此“隐含”的等式约束“显式”化, 从而得到其等价问题

$$\begin{aligned} \max & -b^T \nu \\ \text{s.t. } & A^T \nu - \lambda + c = 0 \\ & \lambda \geq 0 \end{aligned} \quad (11.18)$$

进一步地, 这个问题可以表述为

$$\begin{aligned} \max & -b^T \nu \\ \text{s.t. } & A^T \nu + c \geq 0 \end{aligned} \quad (11.19)$$

这是一个不等式形式的线性规划。

注意到这三个问题之间细微的差别。标准形式线性规划(11.16)的 Lagrange 对偶问题是优化问题(11.17), 而这个优化问题等价于问题(11.18)和(11.19)(但形式不同)。称问题(11.18)和(11.19)都是标准形式线性规划(11.16)的 Lagrange 对偶问题。

例 11.3.2. 不等式形式的线性规划问题

$$\begin{aligned} \min & c^T x \\ \text{s.t. } & Ax \leq b \end{aligned} \quad (11.20)$$

的 Lagrange 函数为

$$L(x, \lambda) = c^T x + \lambda^T (Ax - b) = -b^T \lambda + (A^T \lambda + c)^T x$$

所以, 对偶函数为

$$g(\lambda) = \inf_x L(x, \lambda) = -b^T \lambda + \inf_x (A^T \lambda + c)^T x \quad (11.21)$$

若线性函数的系数不等于 0, 则线性函数的下确界是 $-\infty$ 。因此, 对偶函数可重新表示为

$$g(\lambda) = \begin{cases} -b^T \lambda & A^T \lambda + c = 0 \\ -\infty & \text{其他情况} \end{cases}$$

如果 $\lambda \geq 0$ 且 $A^T \lambda + c = 0$, 那么, 对偶变量 λ 是对偶可行的。和前面一样, 我们可以显式表达对偶可行的条件并作为约束来重新描述对偶问题

$$\begin{aligned} \max & -b^T \lambda \\ \text{s.t. } & A^T \lambda + c = 0 \\ & \lambda \geq 0 \end{aligned} \quad (11.22)$$

该对偶问题是一个标准形式的线性规划。

通过以上两个例子, 可以发现一个非常有趣的现象, 标准形式线性规划问题和不等式形式线性规划问题与它们的对偶问题之间都存在对称性: 标准形式线性规划的对偶问题是只含有不等式约束的线性规划问题, 反之亦然。此外, 问题(11.24)的 Lagrange 对偶问题就是(等价于)原问题(11.20)。

例 11.3.3. 二次规划问题

$$\begin{aligned} \min \quad & \mathbf{x}^T \mathbf{P} \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} \leq \mathbf{b} \end{aligned} \quad (11.23)$$

其中 $\mathbf{P} \in S_+^n$ 。类似地, 可计算出它的对偶函数为

$$g(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} (\mathbf{x}^T \mathbf{P} \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{A} \mathbf{x} - \mathbf{b})) = -\frac{1}{4} \boldsymbol{\lambda}^T \mathbf{A} \mathbf{P}^{-1} \mathbf{A}^T \boldsymbol{\lambda} - \mathbf{b}^T \boldsymbol{\lambda}$$

同样地, 将对偶可行的条件作为约束可得对偶问题

$$\begin{aligned} \max \quad & -\frac{1}{4} \boldsymbol{\lambda}^T \mathbf{A} \mathbf{P}^{-1} \mathbf{A}^T \boldsymbol{\lambda} - \mathbf{b}^T \boldsymbol{\lambda} \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned} \quad (11.24)$$

该对偶问题仍然是一个二次规划问题。

11.3.2 对偶性质

弱对偶性

用 p^* 标记原问题的最优值, d^* 标记 Lagrange 对偶问题的最优值。下述定理将告诉我们, d^* 是通过 Lagrange 函数得到的原问题最优值 p^* 的最好下界。

定理 11.3.1. 不等式

$$d^* \leq p^* \quad (11.25)$$

成立。即使原问题不是凸优化, 上述不等式亦成立。这个性质称为弱对偶性。

根据定理 11.2.1, 可直接推导出弱对偶性成立。即使当 d^* 和 p^* 都趋于无穷时, 弱对偶性不等式(11.25)也成立。例如, 如果原问题无下界, 即 $p^* = -\infty$, 为了保证弱对偶性, 必须有 $d^* = -\infty$, 即 Lagrange 对偶问题不可行。反过来, 若对偶问题无上界, 即 $d^* = \infty$, 为了保证弱对偶性成立, 必须有 $p^* = \infty$, 即原问题不可行。

定义 11.3.2. 差值 $p^* - d^*$ 是原问题的最优值与其通过 Lagrange 对偶函数得到的最好(最大)下界之间的差值。因此, 称 $p^* - d^*$ 是原问题的最优对偶间隙。最优对偶间隙总是非负的。

当原问题很难求解时, 弱对偶不等式(11.25)给出了原问题最优值的一个下界, 这是因为对偶问题总是凸问题, 而且在很多情况下都可以进行有效的求解, 得到 d^* 。考虑双向划分问题(11.8), 其对偶问题是一个半定规划问题

$$\begin{aligned} \max \quad & -\mathbf{1}^T \boldsymbol{\nu} \\ \text{s.t.} \quad & \mathbf{W} + \text{diag}(\boldsymbol{\nu}) \succeq \mathbf{0} \end{aligned}$$

其中, $\nu \in \mathbb{R}^n$ 。即使当 n 取相对较大的值 (例如 $n = 100$ 时), 该对偶问题都可以进行有效求解, 其最优值给出了双向划分问题最优值的一个下界, 而这个下界至少和由 $\lambda_{\min}(\mathbf{W})$ 推导出的下界(11.9)一样好。

强对偶性和 Slater 约束准则

定义 11.3.3. 如果原问题和对偶问题的最优值相等, 即等式

$$p^* = d^* \quad (11.26)$$

成立, 最优对偶间隙为零, 那么, 它们满足强对偶性。

对于一般情况, 强对偶性不成立。但是, 如果原问题(11.4)是凸问题, 即表述为如下形式

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m, \\ & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned} \quad (11.27)$$

其中, 函数 f_0, \dots, f_m 是凸函数, 强对偶性通常 (但不总是) 成立。有很多研究成果给出了强对偶性成立的条件 (除了凸性条件以外), 例如, Slater 条件。这些条件称为约束准则。

定义 11.3.4. Slater 条件: 至少存在一点 $\mathbf{x} \in \text{relint } \mathcal{D}$ ¹ 使得下式成立

$$f_i(\mathbf{x}) < 0, \quad i = 1, \dots, m, \quad \mathbf{A}\mathbf{x} = \mathbf{b}. \quad (11.28)$$

因为不等式约束严格成立, 所以, 满足上述条件的点是严格可行的。

下述定理将告诉我们, 当 Slater 条件成立, 且原问题是凸问题时, 强对偶性成立。即强对偶性定理:

定理 11.3.2. (强对偶性定理) 假设函数 f_0, f_1, \dots, f_m 以及 h_1, \dots, h_p 均为凸函数, 而且满足 Slater 条件, 那么

$$\sup_{\lambda \geq 0, \nu} g(\lambda, \nu) = \inf_{\mathbf{x} \in K} f(\mathbf{x}).$$

即对偶间隙为零。

该定理的证明将放在后面介绍。另外, Slater 条件可以进一步改进, 当不等式约束函数 f_i 中有一些是仿射函数时, 并不要求严格不等号成立。

¹给定集合 \mathcal{D} , 记其仿射包 $\text{aff } \mathcal{D}$ 。则:

$$\text{relint } \mathcal{D} = \{\mathbf{x} \in \mathcal{D} \mid \exists r > 0, \text{使得 } B(\mathbf{x}, r) \cap \text{aff } \mathcal{D} \subset \mathcal{D}\}$$

定义 11.3.5. 改进的 **Slater** 条件: 已知前面的 k 个约束函数 f_1, \dots, f_k 是仿射函数, 存在一点 $\mathbf{x} \in \text{relint } \mathcal{D}$, 使得不等式

$$f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, k, \quad f_i(\mathbf{x}) < 0, \quad i = k + 1, \dots, m, \quad \mathbf{A}\mathbf{x} = \mathbf{b}. \quad (11.29)$$

成立。换言之, 仿射不等式不需要严格成立。

注意, 当所有约束条件都是线性等式或不等式且 $\text{dom } f_0$ 是开集时, 改进的 Slater 条件(11.29)就是该优化问题的可行性条件。若 Slater 条件 (或是改进的 Slater 条件) 满足, 则当 $d^* > -\infty$ 时, 对偶问题能够取得最优值, 即存在一组对偶可行解 (λ^*, ν^*) 使得 $g(\lambda^*, \nu^*) = d^* = p^*$ 。

不满足强对偶性的示例

考虑函数 $f: D \rightarrow \mathbb{R}$ 定义为 $f(x, y) := e^{-x}$, 其中

$$D = \{(x, y) \mid y > 0\} \subseteq \mathbb{R}^2.$$

凸优化问题为

$$\begin{aligned} \min_{(x,y) \in D} \quad & e^{-x} \\ \text{s.t.} \quad & \frac{x^2}{y} \leq 0. \end{aligned}$$

由于 e^{-x} 和 $\frac{x^2}{y}$ 都是 D 上的凸函数, 所以这的确是一个凸优化。我们可以看出实际上约束条件等价于 $x = 0$, 变量 y 是冗余的。显然, 优化问题的最优值为 1。现在我们考虑它的 Lagrange 对偶。可以写出 Lagrange 函数为

$$L(x, y, \lambda) = e^{-x} + \lambda \frac{x^2}{y}.$$

便可推出

$$g(\lambda) = \inf_{(x,y) \in D} L(x, y, \lambda) = 0,$$

对所有的 $\lambda \geq 0$ 。因此, 强对偶性在这个例子中不成立。

11.3.3 常见优化问题的对偶问题及强对偶性

线性方程组的最小二乘解 考虑问题(11.6)

$$\min \quad \mathbf{x}^T \mathbf{x}$$

$$\text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b}$$

其对偶问题为

$$\max \quad -(1/4) \nu^T \mathbf{A} \mathbf{A}^T \nu - \mathbf{b}^T \nu$$

它是一个凹二次函数的无约束极大化问题。

此时, Slater 条件就是原问题的可行性条件。所以, 如果 $\mathbf{b} \in \mathcal{R}(\mathbf{A})$, 即 $p^* < \infty$, 就有 $p^* = d^*$, 则强对偶性成立, 即使 $p^* = \infty$ 亦如此。并且当 $p^* = \infty$ 时, $\mathbf{b} \notin \mathcal{R}(\mathbf{A})$, 故存在 \mathbf{z} 使得 $\mathbf{A}^T \mathbf{z} = 0$, $\mathbf{b}^T \mathbf{z} \neq 0$ 。因此, 对偶函数在直线 $\{t\mathbf{z} \mid t \in \mathbb{R}\}$ 上无界, 也就是说, 对偶问题最优值无界, $d^* = \infty$ 。

二次约束二次规划

考虑约束和目标函数都是二次函数的优化问题 (QCQP)

$$\begin{aligned} \min \quad & (1/2)\mathbf{x}^T \mathbf{P}_0 \mathbf{x} + \mathbf{q}_0^T \mathbf{x} + r_0 \\ \text{s.t.} \quad & (1/2)\mathbf{x}^T \mathbf{P}_i \mathbf{x} + \mathbf{q}_i^T \mathbf{x} + r_i \leq 0, \quad i = 1, \dots, m \end{aligned} \quad (11.30)$$

其中, $\mathbf{P}_0 \in S_{++}^n$, $\mathbf{P}_i \in S_{++}^n$, $i = 1, \dots, m$ 。其 Lagrange 函数为

$$L(\mathbf{x}, \boldsymbol{\lambda}) = (1/2)\mathbf{x}^T \mathbf{P}(\boldsymbol{\lambda}) \mathbf{x} + \mathbf{q}(\boldsymbol{\lambda})^T \mathbf{x} + r(\boldsymbol{\lambda})$$

其中

$$\mathbf{P}(\boldsymbol{\lambda}) = \mathbf{P}_0 + \sum_{i=1}^m \lambda_i \mathbf{P}_i, \quad \mathbf{q}(\boldsymbol{\lambda}) = \mathbf{q}_0 + \sum_{i=1}^m \lambda_i \mathbf{q}_i, \quad r(\boldsymbol{\lambda}) = r_0 + \sum_{i=1}^m \lambda_i r_i$$

如果 $\boldsymbol{\lambda} \geq 0$, 有 $\mathbf{P}(\boldsymbol{\lambda}) \succ \mathbf{0}$ 及

$$g(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) = -(1/2)\mathbf{q}(\boldsymbol{\lambda})^T \mathbf{P}(\boldsymbol{\lambda})^{-1} \mathbf{q}(\boldsymbol{\lambda}) + r(\boldsymbol{\lambda})$$

因此, 对偶问题可以表述为

$$\begin{aligned} \max \quad & - (1/2)\mathbf{q}(\boldsymbol{\lambda})^T \mathbf{P}(\boldsymbol{\lambda})^{-1} \mathbf{q}(\boldsymbol{\lambda}) + r(\boldsymbol{\lambda}) \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq 0 \end{aligned} \quad (11.31)$$

原问题满足 Slater 条件, 也就是二次不等式约束严格成立, 即存在一点 \mathbf{x} , 使得

$$(1/2)\mathbf{x}^T \mathbf{P}_i \mathbf{x} + \mathbf{q}_i^T \mathbf{x} + r_i < 0, \quad i = 1, \dots, m$$

根据强对偶定理可知, 优化问题(11.31)和(11.30)之间强对偶性成立。

11.3.4 强对偶性定理的证明

对偶间隙的几何解释

为了直观理解对偶间隙, 考虑如下带有一个不等式约束的优化问题:

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_1(\mathbf{x}) \leq 0 \end{aligned}$$

自变量 \mathbf{x} 的自然定义域为 \mathcal{D} , 优化问题的最优值 p^* 。

定义集合

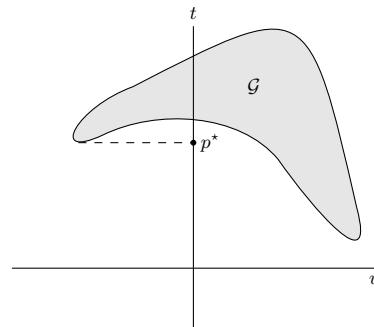
$$\mathcal{G} = \{(f_1(\mathbf{x}), f_0(\mathbf{x})) \in \mathbb{R} \times \mathbb{R} \mid \mathbf{x} \in \mathcal{D}\} \quad (11.32)$$

下面结合该集合, 给出最优值、对偶函数及对偶间隙的一些几何解释。

原问题: 利用集合 \mathcal{G} , 表达最优值 p^*

$$p^* = \inf\{t \mid (u, t) \in \mathcal{G}, u \leq 0\}$$

这对应于在图11.2中, 取集合 \mathcal{G} 在左半平面的最低点。

图 11.2: p^* 的几何解释

对偶问题: 在 $(u, t) \in \mathcal{G}$ 上, 定义仿射函数

$$(\lambda, 1)^T(u, t) = \lambda u + t$$

得到

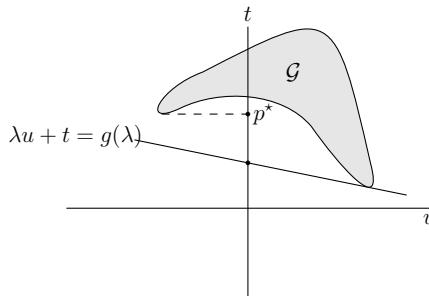
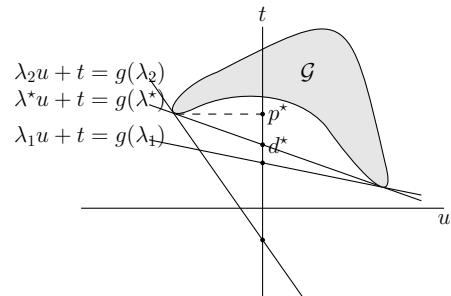
$$g(\lambda) = \inf\{(\lambda, 1)^T(u, t) \mid (u, t) \in \mathcal{G}\}$$

如果下确界有限, 则不等式

$$(\lambda, 1)^T(u, t) \geq g(\lambda)$$

是集合 \mathcal{G} 的一个(非竖直)支撑超平面。显然, 对偶问题的最优值

$$d^* = \max_{\lambda \geq 0} g(\lambda)$$

图 11.3: $g(\lambda)$ 的几何解释: 支撑超平面与坐标轴 $u = 0$ 的交点即为 $g(\lambda)$ 。图 11.4: d^* 以及最优对偶间隙 $p^* - d^*$ 的几何解释。可知此时强对偶性不成立。

从图11.3可以看出支撑超平面 $\lambda u + t = g(\lambda)$ 与纵坐标轴的交点即为 $g(\lambda)$ 。当我们对 λ 取上确界, 便可得到图11.4所示的 d^* 。从图中可以看出 p^* 是在 $g(\lambda)$ 之上的。我们仍然可以从数学上

证明这一几何直观。假设 $\lambda \geq 0$, 如果 $u \leq 0$, 则 $t \geq (\lambda, 1)^T(u, t)$ 成立, 有

$$\begin{aligned} p^* &= \inf\{t \mid (u, t) \in \mathcal{G}, u \leq 0\} \\ &\geq \inf\{(\lambda, 1)^T(u, t) \mid (u, t) \in \mathcal{G}, u \leq 0\} \\ &\geq \inf\{(\lambda, 1)^T(u, t) \mid (u, t) \in \mathcal{G}\} \\ &= g(\lambda) \end{aligned}$$

因此, $p^* \geq d^*$, 即弱对偶性成立。在上图所示中可以看出是存在对偶间隙, 即强对偶性不成立。如果强对偶性成立, 那么对偶间隙为零, 此时 p^* 与 d^* 重合, 则有如下几何图11.6所示:

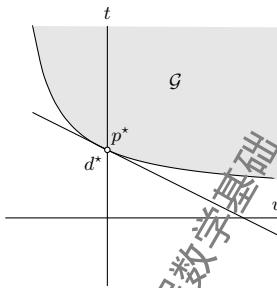


图 11.5: 强对偶性成立的图示。

强对偶性定理的证明

考虑问题

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & A\mathbf{x} = \mathbf{b} \end{aligned}$$

其中函数 f_0, \dots, f_m 均为凸函数。假设 Slater 条件满足: 即存在一点 $\tilde{\mathbf{x}} \in \text{relint } \mathcal{D}$ 使得 $f_i(\tilde{\mathbf{x}}) < 0, i = 1, \dots, m$ 且 $A\tilde{\mathbf{x}} = \mathbf{b}$ 。现需证明强对偶性成立。

为简化证明, 附加两个假设条件:

- \mathcal{D} 的内点集不为空集 (即 $\text{relint } \mathcal{D} = \text{int } \mathcal{D}$)
- $\text{rank } A = p$ 。假设最优值 p^* 有限。

证明. 定义集合 \mathcal{A}

$$\mathcal{A} = \{(\mathbf{u}, \mathbf{v}, t) \mid \exists \mathbf{x} \in \mathcal{D}, f_i(\mathbf{x}) \leq u_i, i = 1, \dots, m, h_j(\mathbf{x}) = v_j, j = 1, \dots, p, f_0(\mathbf{x}) \leq t\}$$

显然, 它是凸集。定义另一凸集 \mathcal{B} 为

$$\mathcal{B} = \{(0, 0, s) \in R^m \times R^p \times R \mid s < p^*\}$$

集合 \mathcal{A} 和集合 \mathcal{B} 不相交。设存在 $(\mathbf{u}, \mathbf{v}, t) \in \mathcal{A} \cap \mathcal{B}$ 。因为 $(\mathbf{u}, \mathbf{v}, t) \in \mathcal{B}$, 有 $\mathbf{u} = \mathbf{0}$, $\mathbf{v} = \mathbf{0}$, 以及 $f_0(\mathbf{x}) \leq t < p^*$, 而这与 p^* 是原问题的最优值矛盾。

因此, 根据超平面定理, 存在 $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}}, \mu) \neq 0$ 和 α 使得

$$(\mathbf{u}, \mathbf{v}, t) \in \mathcal{A} \Rightarrow \tilde{\boldsymbol{\lambda}}^T \mathbf{u} + \tilde{\boldsymbol{\nu}}^T \mathbf{v} + \mu t \geq \alpha \quad (11.33)$$

和

$$(\mathbf{u}, \mathbf{v}, t) \in \mathcal{B} \Rightarrow \tilde{\boldsymbol{\lambda}}^T \mathbf{u} + \tilde{\boldsymbol{\nu}}^T \mathbf{v} + \mu t \leq \alpha \quad (11.34)$$

根据式 (11.33), 有 $\tilde{\boldsymbol{\lambda}} \geq \mathbf{0}$ 和 $\mu \geq 0$ 。

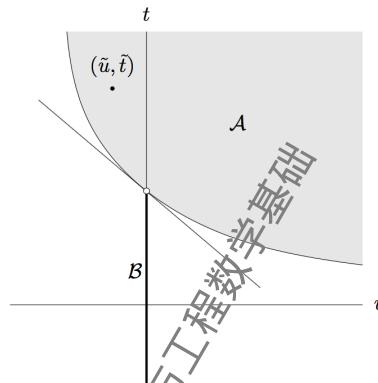


图 11.6: 两个集合都是凸集, 不相交, 因此存在分离超平面。

式 (11.34) 意味着 $\mu t \leq \alpha$ 对所有 $t \leq p^*$ 成立, 因此 $\mu p^* \leq \alpha$ 。结合式 (11.33), 对任意 $\mathbf{x} \in \mathcal{D}$, 下式成立

$$\sum_{i=1}^m \tilde{\lambda}_i f_i(\mathbf{x}) + \tilde{\boldsymbol{\nu}}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) + \mu f_0(\mathbf{x}) \geq \alpha \geq \mu p^* \quad (11.35)$$

设 $\mu > 0$, 式 (11.35) 两端除以 μ , 可得任意 $\mathbf{x} \in \mathcal{D}$, 下式成立

$$L(\mathbf{x}, \tilde{\boldsymbol{\lambda}}/\mu, \tilde{\boldsymbol{\nu}}/\mu) \geq p^*$$

定义

$$\boldsymbol{\lambda} = \tilde{\boldsymbol{\lambda}}/\mu, \quad \boldsymbol{\nu} = \tilde{\boldsymbol{\nu}}/\mu$$

对 \mathbf{x} 求极小可以得到 $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \geq p^*$ 。根据强对偶性, 有 $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq p^*$, 因此 $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = p^*$ 。说明当 $\mu > 0$ 时强对偶性成立, 且对偶问题能达到最优值。

考虑当 $\mu = 0$ 时的情形。根据式 (11.35), 对任意 $\mathbf{x} \in \mathcal{D}$, 有

$$\sum_{i=1}^m \tilde{\lambda}_i f_i(\mathbf{x}) + \tilde{\boldsymbol{\nu}}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) \geq 0 \quad (11.36)$$

满足 Slater 条件的点 $\tilde{\mathbf{x}}$ 同样满足式 (11.36), 因此有 $\sum_{i=1}^m \tilde{\lambda}_i f_i(\mathbf{x}) \geq 0$

- $f_i(\tilde{\mathbf{x}}) < 0$ 且 $\tilde{\lambda}_i \geq 0$, 有 $\tilde{\lambda} = \mathbf{0}$
- $(\tilde{\lambda}, \tilde{\nu}, \mu) \neq \mathbf{0}$ 且 $\tilde{\lambda} = \mathbf{0}$, $\mu = 0$, 所以 $\tilde{\nu} \neq \mathbf{0}$
- 式 (11.36) 表明对任意 $\mathbf{x} \in \mathcal{D}$ 有 $\tilde{\nu}^T(\mathbf{A}\mathbf{x} - \mathbf{b}) \geq 0$ 。
- 又因为 $\tilde{\mathbf{x}}$ 满足 $\tilde{\nu}^T(\mathbf{A}\tilde{\mathbf{x}} - \mathbf{b}) = 0$, 且 $\tilde{\mathbf{x}} \in \text{int } \mathcal{D}$, 因此除了 $\mathbf{A}^T\tilde{\nu} = \mathbf{0}$ 的情况, 总存在 \mathcal{D} 中的点使得 $\tilde{\nu}^T(\mathbf{A}\mathbf{x} - \mathbf{b}) < 0$ 。而 $\mathbf{A}^T\tilde{\nu} = \mathbf{0}$ 显然与假设 $\text{rank } \mathbf{A} = p$ 矛盾。

因此 $\mu = 0$ 情形不存在。故证毕。 \square

11.3.5 强弱对偶性的极大极小描述

下面将原、对偶优化问题以一种更为对称的方式进行表达。这将更有助于对原问题和对偶问题的理解。实际上, 原问题的最优值写成如下形式

$$p^* = \inf_{\mathbf{x}} \sup_{\lambda \geq \mathbf{0}, \nu} L(\mathbf{x}, \lambda, \nu)$$

这是因为

$$\begin{aligned} \sup_{\lambda \geq \mathbf{0}, \nu} L(\mathbf{x}, \lambda, \nu) &= \sup_{\lambda \geq \mathbf{0}, \nu} \left(f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^p \nu_j h_j(\mathbf{x}) \right) \\ &= \begin{cases} f_0(\mathbf{x}) & f_i(\mathbf{x}) \leq 0, h_j(\mathbf{x}) = 0, \quad i = 1, \dots, m, j = 1, \dots, p \\ \infty & \text{其他情况} \end{cases} \end{aligned}$$

根据对偶函数的定义, 有

$$d^* = \sup_{\lambda \geq \mathbf{0}, \nu} \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \nu)$$

因此弱对偶性可以表述为下述不等式 (显然成立)

$$\sup_{\lambda \geq \mathbf{0}, \nu} \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \nu) \leq \inf_{\mathbf{x}} \sup_{\lambda \geq \mathbf{0}, \nu} L(\mathbf{x}, \lambda, \nu) \quad (11.37)$$

强对偶性可以表示为下面的不等式

$$\sup_{\lambda \geq \mathbf{0}, \nu} \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \nu) = \inf_{\mathbf{x}} \sup_{\lambda \geq \mathbf{0}, \nu} L(\mathbf{x}, \lambda, \nu)$$

强对偶性意味着对 \mathbf{x} 求极小和对 $\lambda \geq \mathbf{0}, \nu$ 求极大可以互换而不影响结果。

11.4 最优性条件

11.4.1 互补松弛条件

假设原问题和对偶问题的最优值都可以达到且相等 (即强对偶性成立)。令 \mathbf{x}^* 是原问题的最优解, $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 是对偶问题的最优解, 这表明

$$\begin{aligned} f_0(\mathbf{x}^*) &= g(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) \\ &= \inf_{\mathbf{x}} (f_0(\mathbf{x}) + \sum_{i=1}^m \boldsymbol{\lambda}_i^* f_i(\mathbf{x}) + \sum_{j=1}^p \boldsymbol{\nu}_j^* h_j(\mathbf{x})) \\ &\leq f_0(\mathbf{x}^*) + \sum_{i=1}^m \boldsymbol{\lambda}_i^* f_i(\mathbf{x}^*) + \sum_{j=1}^p \boldsymbol{\nu}_j^* h_j(\mathbf{x}^*) \\ &\leq f_0(\mathbf{x}^*) \end{aligned}$$

第一个等式说明最优对偶间隙为零, 第二个等式是对偶函数的定义, 第三个不等式成立是因为 Lagrange 函数关于 \mathbf{x} 的下确界小于等于其在 $\mathbf{x} = \mathbf{x}^*$ 处的值, 最后一个不等式成立则是因为 $\boldsymbol{\lambda}_i^* \geq 0, f_i(\mathbf{x}^*) \leq 0, i = 1, \dots, m$, 以及 $h_j(\mathbf{x}^*) = 0, j = 1, \dots, p$ 。因此, 在上面的式子链中, 最后两个不等式取等号。

可以由此得出一些有意义的结论。一方面, 由于第三个不等式变为等式, 因此, $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 关于 \mathbf{x} 求极小值是在 \mathbf{x}^* 处取得的。其中 Lagrange 函数 $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 也可以有其他最优点; \mathbf{x}^* 只是其中一个最优点。

另一方面的结论是

$$\sum_{i=1}^m \boldsymbol{\lambda}_i^* f_i(\mathbf{x}^*) = 0$$

事实上, 求和项的每一项都非正, 因此有

$$\boldsymbol{\lambda}_i^* f_i(\mathbf{x}^*) = 0, \quad i = 1, \dots, m. \quad (11.38)$$

上述条件称为互补松弛性; 它对任意原问题最优解 \mathbf{x}^* 以及对偶问题最优解 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 都成立 (当强对偶性成立时)。我们可以将互补松弛条件写成

$$\boldsymbol{\lambda}_i^* > 0 \implies f_i(\mathbf{x}^*) = 0$$

或者

$$f_i(\mathbf{x}^*) < 0 \implies \boldsymbol{\lambda}_i^* = 0$$

这表明, 在最优点处, 除非第 i 个约束起作用, 否则第 i 个最优 Lagrange 乘子取值为零。

11.4.2 KKT 最优性条件

假设函数 $f_0, \dots, f_m, h_1, \dots, h_p$ 可微 (因此定义域是开集), 此时并没有假设这些函数是凸函数。

优化问题的 KKT 条件

定义 11.4.1. 令 \mathbf{x}^* 和 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 分别是原问题和对偶问题的最优解, 其对偶间隙为零。因此, 就有

- 原始约束:

$$f_i(\mathbf{x}^*) \leq 0, \quad i = 1, \dots, m$$

$$h_j(\mathbf{x}^*) = 0, \quad j = 1, \dots, p$$

- 对偶约束: $\lambda_i^* \geq 0, \quad i = 1, \dots, m$
- 互补松弛: $\lambda_i^* f_i(\mathbf{x}^*) = 0, \quad i = 1, \dots, m$
- 稳定性条件:

$$\nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{j=1}^p \nu_j^* \nabla h_j(\mathbf{x}^*) = 0$$

这些公式被称为 **Karush-Kuhn-Tucker (KKT)** 条件。

总之, 对于目标函数和约束函数可微的任意优化问题, 如果强对偶性成立, 那么, 原问题和对偶问题的任意一对最优解都必须满足 KKT 条件。事实上, 它们两者是等价的, 即如下定理所述。

定理 11.4.1. 对于凸优化问题 (11.27), 如果 Slater 条件成立, 那么 $\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*$ 分别是原始、对偶全局最优解当且仅当它们满足 KKT 条件。

证明. 由本节的起始部分, 易知必要性显然成立。下面考虑充分性: 为了说明这一点, 注意到前面两个条件说明了 \mathbf{x}^* 是原问题的可行解。因为 $\lambda_i^* \geq 0$, $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 是 \mathbf{x} 的凸函数; 最后一个 KKT 条件说明在 $\mathbf{x} = \mathbf{x}^*$ 处, Lagrange 函数的导数为零。因此, $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 关于 \mathbf{x} 求极小在 \mathbf{x}^* 处取得最小值。我们得出结论

$$\begin{aligned} g(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) &= L(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) \\ &= f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* f_i(\mathbf{x}^*) + \sum_{j=1}^p \nu_j^* h_j(\mathbf{x}^*) \\ &= f_0(\mathbf{x}^*) \end{aligned}$$

最后一行成立是因为 $h_j(\mathbf{x}^*) = 0$ 以及 $\lambda_i^* f_i(\mathbf{x}^*) = 0$ 。这说明原问题的解 \mathbf{x}^* 和对偶问题的解 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 之间的对偶间隙为零, 因此分别是原、对偶问题最优解。总之, 对目标函数和约束函数可微的任意凸优化问题, 任意满足 KKT 条件的点分别是原、对偶最优解, 对偶间隙为零。 \square

KKT 条件在优化领域有着重要的作用。在一些特殊情形下, 是可以解析求解 KKT 条件的 (因此也可以求解优化问题)。更一般地, 很多求解凸优化问题的方法可以认为或理解为求解 KKT 条件的方法。

例 11.4.1. 考虑问题

$$\begin{aligned} \min \quad & (1/2)\mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned} \quad (11.39)$$

其中, $\mathbf{P} \in S_+^n$ 。此问题的 KKT 条件为

$$\mathbf{A} \mathbf{x}^* = \mathbf{b}, \quad \mathbf{P} \mathbf{x}^* + \mathbf{q} + \mathbf{A}^T \mathbf{v}^* = \mathbf{0}$$

我们可以将其写成

$$\mathbf{H}_x = \begin{bmatrix} \mathbf{P} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}^* \\ \mathbf{v}^* \end{bmatrix} = \begin{bmatrix} -\mathbf{q} \\ \mathbf{b} \end{bmatrix}$$

求解变量 \mathbf{x}^* , \mathbf{v}^* 的 $m+n$ 个方程, 其中变量的维数为 $m+n$, 可以得到优化问题 (11.39) 的最优原变量和对偶变量。

11.4.3 通过解对偶问题求解原问题

前面提到, 如果强对偶性成立, 且存在一个对偶最优解 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$, 那么任意原问题最优点也是 $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 的最优解。这个性质可以让我们从对偶最优方程去求解原问题最优解。

更具体地, 假设强对偶性成立, 对偶最优解 $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 已知。假设 $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 的最小值点唯一, 即下列问题的解

$$\min \quad f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i^* f_i(\mathbf{x}) + \sum_{j=1}^p \nu_j^* h_j(\mathbf{x}) \quad (11.40)$$

唯一。对于优化凸问题而言, 这是必然会发生 (比如说, $L(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ 是关于 \mathbf{x} 的严格凸函数)。如果问题(11.40)的解是原问题的可行解, 那么, 它就是原问题的最优解; 反之, 如果它不是原问题的可行解, 那么, 原问题不存在最优点, 即原问题的最优解无法达到。当对偶问题比原问题更容易求解时, 比如说对偶问题可以解析求解或者有某些特殊的结构更易分析, 上述方法很有意义。

例 11.4.2. 考虑熵的最大化问题

$$\min \quad f_0(\mathbf{x}) = \sum_{i=1}^n x_i \log x_i$$

$$\text{s.t.} \quad \mathbf{A} \mathbf{x} \leq \mathbf{b}$$

$$\mathbf{1}^T \mathbf{x} = 1$$

其中定义域为 \mathbb{R}_{++}^n , 其对偶问题为

$$\begin{aligned} \max \quad & -\mathbf{b}^T \boldsymbol{\lambda} - \nu - e^{-\nu-1} \sum_{i=1}^n e^{-a_i^T \boldsymbol{\lambda}} \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq \mathbf{0} \end{aligned}$$

假设改进的 *Slater* 条件成立, 即存在 $\mathbf{x} > 0$ 使得 $\mathbf{A} \mathbf{x} \leq \mathbf{b}$ 以及 $\mathbf{1}^T \mathbf{x} = 1$, 因此强对偶性成立, 存在一个对偶最优解 $(\boldsymbol{\lambda}^*, \nu^*)$ 。

设对偶问题已经解出。 (λ^*, ν^*) 处的拉格朗日函数为

$$L(\mathbf{x}, \lambda^*, \nu^*) = \sum_{i=1}^n x_i \log x_i + \lambda^{*T}(\mathbf{A}\mathbf{x} - \mathbf{b}) + \nu^*(\mathbf{1}^T \mathbf{x} - 1)$$

它在 D 上严格凸且有下界, 因此有一个唯一解 \mathbf{x}^* ,

$$x_i^* = 1 / \exp(\mathbf{a}_i^T \lambda^* + \nu^* + 1), \quad i = 1, \dots, n$$

其中 \mathbf{a}_i 是矩阵 \mathbf{A} 的列向量。如果 \mathbf{x}^* 是原问题的可行解, 那么, 它必然是原问题(11.14)的最优解; 反之, 如果 \mathbf{x}^* 不是原问题的可行解, 那么, 就说原问题的最优解不能达到。

例 11.4.3. 在等式约束下极小化可分函数

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) = \sum_{i=1}^n f_i(x_i) \\ \text{s.t.} \quad & \mathbf{a}^T \mathbf{x} = b \end{aligned}$$

其中 $\mathbf{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$, 函数 $f_i : \mathbb{R} \rightarrow \mathbb{R}$ 是可微函数, 也是严格凸函数。目标函数是可分的, 因为它可以表示为关于一系列单变量 x_1, \dots, x_n 的函数求和的形式。假设函数 f_0 的定义域与约束集有交集, 即存在一点 $\mathbf{x}_0 \in \text{dom } f_0$, 使得 $\mathbf{a}^T \mathbf{x}_0 = b$ 。由此可知, 该问题存在唯一最优解 \mathbf{x}^* 。

该问题的 Lagrange 函数为

$$L(\mathbf{x}, \nu) = \sum_{i=1}^n f_i(x_i) + \nu(\mathbf{a}^T \mathbf{x} - b) = -b\nu + \sum_{i=1}^n (f_i(x_i) + \nu a_i x_i)$$

同样是可分函数, 因此, 对偶函数为

$$\begin{aligned} g(\nu) &= -b\nu + \inf_{\mathbf{x}} \left(\sum_{i=1}^n (f_i(x_i) + \nu a_i x_i) \right) \\ &= -b\nu + \sum_{i=1}^n \inf_{x_i} (f_i(x_i) + \nu a_i x_i) \\ &= -b\nu - \sum_{i=1}^n f_i^*(-\nu a_i) \end{aligned}$$

故对偶问题可表示为

$$\max \quad -b\nu - \sum_i f_i^*(-\nu a_i)$$

其中, $\nu \in \mathbb{R}$ 是实变量。

现在假设找到了一个对偶最优解 ν^* 。事实上, 有很多简单的方法来求解一个实变量的凸问题, 比如说二分法。因为每个函数 f_i 都是严格凸的, 所以, 函数 $L(\mathbf{x}, \nu^*)$ 关于 \mathbf{x} 是严格凸的, 故具有唯一的最小点 $\tilde{\mathbf{x}}$ 。然而, 已知 \mathbf{x}^* 是 $L(\mathbf{x}, \nu^*)$ 的最小点, 因此, 就有 $\tilde{\mathbf{x}} = \mathbf{x}^*$ 。这可以通过求解 $\nabla_{\mathbf{x}} L(\mathbf{x}, \nu^*) = 0$ 得到 \mathbf{x}^* , 即求解方程组 $f_i'(\mathbf{x}^*) = -\nu^* a_i$, $i = 1, \dots, n$ 。

11.5 数据科学中常见模型的对偶问题

在模式识别问题和分类问题中, 给定数据集

$$D = (\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N), y_i \in \{-1, +1\},$$

我们希望 (从给定的函数族中) 找到一个函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$, 使得 $f(\mathbf{x}_i)$ 与 y_i 的符号一致。如果函数对这些训练集都成立, 我们称 f 能对数据集分离、分类或判别。我们有时也考虑弱分离, 在这种情况下, 允许一定的容忍度, 只需要弱不等式成立。

11.5.1 线性可分支持向量机

间隔与支持向量

给定上述训练样本集 D , 分类学习最基本的想法就是基于训练集 D 在样本空间中找到一个划分超平面, 将不同类别的样本分开。但能将训练样本分开的划分超平面可能有很多, 如图 11.7 所示, 我们应该努力去找到哪一个呢?

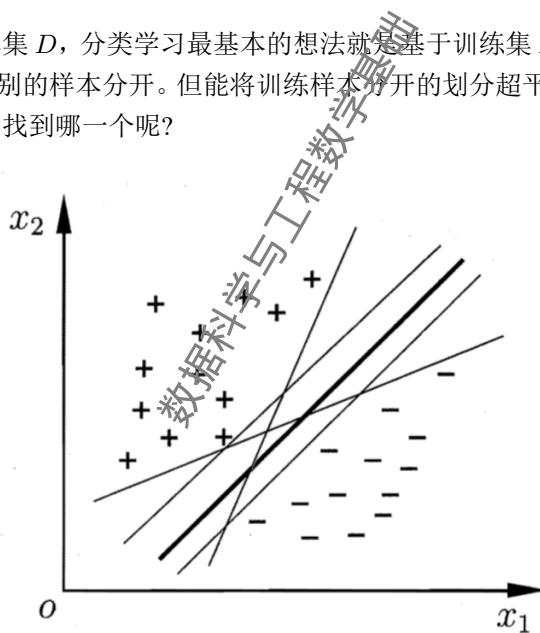


图 11.7: 存在多个划分超平面将两类训练样本分开

直观上看, 应该去找位于两类训练样本“正中间”的划分超平面, 即图 11.7 中最粗的那条线, 因为该划分超平面对于训练样本局部扰动的“容忍”性最好。例如, 由于训练集的局限性或噪声等因素, 训练集外的样本可能比图 11.7 中的训练样本更接近两个类的分隔界, 这将使图中许多划分超平面出现错误, 而“正中间”的超平面受影响最小。换言之, 这个划分超平面所产生的分类结果是最鲁棒的, 对未见示例的泛化能力最强。

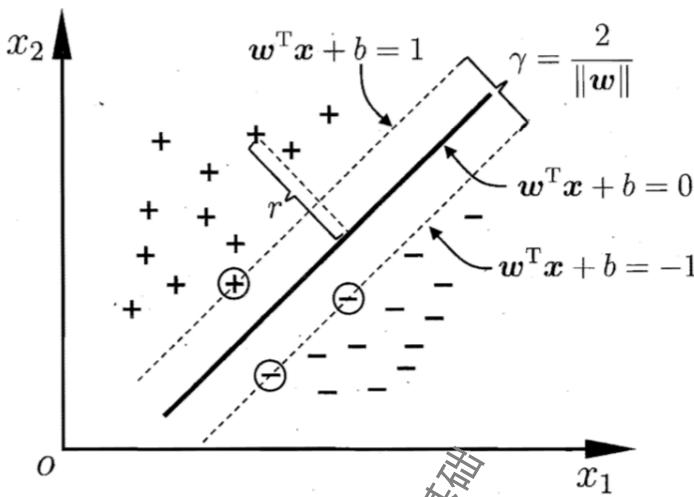


图 11.8: 支持向量与间隔

在样本空间中, 划分超平面可通过如下线性方程来描述:

$$w^T x + b = 0 \quad (11.41)$$

其中, $w = (w_1, w_2, \dots, w_d)^T$ 为法向量, 决定了超平面的方向; b 为偏置项, 决定了超平面与原点之间的距离。显然, 划分超平面可被法向量 w 和偏置 b 确定, 下面我们将其记为 (w, b) 。样本空间中任一点 x 到超平面 (w, b) 的距离可写为

$$r = \frac{|w^T x + b|}{\|w\|} \quad (11.42)$$

假设超平面 (w, b) 能将训练样本正确分类, 即对于 $(x_i, y_i) \in D$, 若 $y_i = +1$, 则有 $w^T x_i + b > 0$; 若 $y_i = -1$, 则有 $w^T x_i + b < 0$ 。令

$$\begin{cases} w^T x_i + b \geq +1, & y_i = +1; \\ w^T x_i + b \leq -1, & y_i = -1. \end{cases} \quad (11.43)$$

如图 11.8 所示, 距离超平面最近的几个训练样本点使式(11.43)的等号成立, 它们被称为“支持向量”(support vector), 两个异类支持向量到超平面的距离之和为

$$\gamma = \frac{2}{\|w\|} \quad (11.44)$$

它被称为“间隔”(margin)。想要找到具有“最大间隔”(maximum margin) 的划分超平面, 也就是要找到能满足式中约束的参数 w 和 b , 使得 γ 最大, 即

$$\begin{aligned} \max_{w, b} \quad & \frac{2}{\|w\|} \\ \text{s.t.} \quad & y_i(w^T x_i + b) \geq 1, \quad i = 1, 2, \dots, N \end{aligned} \quad (11.45)$$

显然, 为了最大化间隔, 仅需最大化 $\|\mathbf{w}\|^{-1}$, 这等价于最小化 $\|\mathbf{w}\|^2$ 。于是, 式(11.45)可重写为

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad i = 1, 2, \dots, N \end{aligned} \quad (11.46)$$

这就是支持向量机 (Support Vector Machine, 简称 SVM) 的基本型。

对偶问题

我们希望求解式(11.46)来得到大间隔划分超平面所对应的模型

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \quad (11.47)$$

其中, \mathbf{w} 和 b 是模型参数。注意到式(11.46)本身是一个凸二次规划问题, 能直接用现成的优化计算包求解, 但我们可以有更高效的办法。

对式(11.46)使用拉格朗日乘子法可得到其“对偶问题”。具体来说, 对式(11.46)的每条约束添加拉格朗日乘子 $\alpha \geq 0$, 则该问题的拉格朗日函数可写为

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^N \alpha_i (1 - y_i(\mathbf{w}^T \mathbf{x}_i + b)) \quad (11.48)$$

其中 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)^T$ 。令 $L(\mathbf{w}, b, \alpha)$ 对 \mathbf{w} 和 b 的偏导为零可得

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i, \quad (11.49)$$

$$0 = \sum_{i=1}^N \alpha_i y_i \quad (11.50)$$

将式(11.49)代入(11.48), 即可将 $L(\mathbf{w}, b, \alpha)$ 中的 \mathbf{w} 和 b 消去, 再考虑式(11.50)的约束, 就得到式(11.46)的对偶问题

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s.t.} \quad & \sum_{i=1}^N \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, N \end{aligned} \quad (11.51)$$

解出 α 后, 求出 \mathbf{w} 与 b 即可得到模型

$$\begin{aligned} f(\mathbf{x}) &= \text{sign}(\mathbf{w}^T \mathbf{x} + b) \\ &= \text{sign} \left(\sum_{i=1}^N \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b \right) \end{aligned} \quad (11.52)$$

从对偶问题(11.51)得到的解是公式(11.48)中的拉格朗日乘子, 它恰对应着训练样本 (\mathbf{x}_i, y_i) 。注意到式(11.46)中有不等式约束, 因此上述过程需满足 KKT (Karush-Kuhn-Tucker) 条件, 即要

求

$$\begin{cases} \alpha_i \geq 0 \\ y_i f(\mathbf{x}_i) - 1 \geq 0 \\ \alpha_i(y_i f(\mathbf{x}_i) - 1) = 0 \end{cases} \quad (11.53)$$

于是, 对任意训练样本 (\mathbf{x}_i, y_i) , 总有 $\alpha_i = 0$ 或 $y_i(\mathbf{x}_i) = 1$ 。若 $\alpha_i = 0$, 则该样本不会在式(11.60)的求和式中出现, 也就不会对 $f(\mathbf{x})$ 产生影响; 若 $\alpha_i > 0$, 则必有 $y_i f(\mathbf{x}_i) = 1$, 所对应的样本点位于最大间隔边界上, 是一个支持向量。这也表明了支持向量机的一个重要性质: 训练完成后, 大部分的训练样本都不需保留, 最终模型仅与支持向量有关, 支持向量机也因此而得名。

11.5.2 线性支持向量机

软间隔

现实中的训练样本集, 通常由于数据本身具有噪音或其他原因, 导致数据集是线性不可分的, 或者是近似线性可分。对于线性近似可分数据集, 在第六章已经介绍, 通过引入“软间隔”的方式, 即松弛变量的方式使其“可分”。因此得到如下软间隔优化问题:

$$\begin{aligned} & \min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \\ & \text{s.t. } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, i = 1, 2, \dots, N \\ & \quad \xi_i \geq 0, i = 1, 2, \dots, N \end{aligned} \quad (11.54)$$

对偶问题

同理, 对式(11.54)使用拉格朗日乘子法可得到其“对偶问题”。具体来说, 对式(11.54)的每条约束添加拉格朗日乘子 $\alpha \geq \mathbf{0}$, $\mu \geq \mathbf{0}$, 则该问题的拉格朗日函数可写为

$$L(\mathbf{w}, b, \xi, \alpha, \mu) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i + \sum_{i=1}^N \alpha_i(1 - \xi_i - y_i(\mathbf{w}^T \mathbf{x}_i + b)) - \sum_{i=1}^N (\mu_i \xi_i) \quad (11.55)$$

其中 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)^T$ 。令 $L(\mathbf{w}, b, \alpha)$ 对 \mathbf{w} 、 b 和 ξ 的偏导为零可得

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i, \quad (11.56)$$

$$0 = \sum_{i=1}^N \alpha_i y_i \quad (11.57)$$

$$\alpha + \mu = C \mathbf{1} \quad (11.58)$$

将式(11.56)和式(11.58)代入(11.55), 即可将 $L(\mathbf{w}, b, \alpha)$ 中的 \mathbf{w} 和 b 消去, 再考虑式(11.57)的约束, 就得到式(11.54)的对偶问题

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{s.t.} \quad & \sum_{i=1}^N \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N \end{aligned} \quad (11.59)$$

解出 α 后, 求出 \mathbf{w} 与 b 即可得到模型

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{sign}(\mathbf{w}^T \mathbf{x} + b) \\ &= \mathbf{sign} \left(\sum_{i=1}^N \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b \right) \end{aligned} \quad (11.60)$$

在线性不可分的情况下, 对应于 $\alpha_i > 0$ 的样本点的实例 \mathbf{x}_i 称为支持向量。根据需满足的 KKT (Karush-Kuhn-Tucker) 条件易知, 若 $\alpha_i < C$, 则 $\xi_i = 0$, 支持向量恰好落在间隔边界上; 若 $\alpha_i = C$, 当 $0 < \xi_i < 1$ 时, 分类正确, 支持向量落在间隔边界与分离超平面之间, 当 $\xi_i = 1$ 时, 则支持向量在分离超平面上, 当 $\xi_i > 1$ 时, 则位于分离超平面误分类的一侧。

11.6 阅读材料

详细介绍 Lagrange 对偶理论的文献很多, 如 Luenberger, Rockafellar, Whittle, Hiriart-Urruty 和 Lemarechal 以及 Bertsekas, Nedic 和 Ozdaglar. Lagrange 对偶这个名字来源于利用 Lagrange 乘子法求解具有等式约束的优化问题, 参见 Courant 和 Hilbert。

5.2.5 中矩阵对策的极大极小结论的提出事实上是早于线性规划对偶理论的, von Neuman 和 Morgenstern 通过一个择一定理证明了这个结论。第 219 页提到的关于线性规划的强对偶性的结论是基于 von Neumann 以及 Gale, Kuhn 和 Tucker 的。非凸二次规划问题 (5.32) 的强对偶性是采用信赖域方法求解非线性优化的文献中的一个基本结论 (Nocedal 和 Wright)。这和控制理论中的 S 过程也有关联, 见附录 §B.1 中的讨论, 将 §5.3.2 中强对偶性的证明扩展至改进的 Slater 条件可以参看文献 Rockafellar。

鞍点性质成立的条件 (5.47) 可以参看文献 Rockafellar 以及 Bertsekas, Nedic 和 Ozdaglar;

KKT 条件得名于 Karush(他在 1939 年未发表的硕士论文中提到了这个结论, 文献 Kuhn 对其进行了整理) 以及 Kuhn 和 Tucker. John 也推导了类似的最优性条件。例 5.2 中的注水算法在信息理论以及通信领域得到了应用 (Cover 和 Thomas)。

Farkas 引理由 Farkas 提出。这个引理也是关于线性不等式和等式系统的择一性理论的最为知名的定理, 事实上, 关于这个定理还有很多不同的变化形式; 参见 Mangasarian。Farkas 引理在资产定价 (例 5.10) 中的应用在文献 Bertsimas 和 Tsitsiklis 以及 Ross 中都有涉及。

参考文献 Isii, Luenberger, Berman, 以及 Rockafellar 中都提到了 Lagrange 对偶理论在广义不等式问题中的扩展。在文献 Nesterov 和 Nemirovski 以及 Ben-Tal 和 Nemirovski 中, 这种扩展在锥规划问题中予以讨论。广义不等式的强择一定理在参考文献 Ben-Israel, Berman 和 Ben-Israel 以及 Craven 和 Kohila 中被提及。文献 Bellman 和 Fan, Wolkowicz, 以及 Lasscrre 给出了 Farkas 引理在线性矩阵不等式中的扩展。

11.7 习题

习题 11.1. 推导共轭函数: $f(x) = \max_{i=1, \dots, n} x_i$, 定义在 \mathbf{R}^n 上。

习题 11.2. 考虑优化问题

$$\text{minimize } e^{-x}$$

$$\text{subject to } x^2/y \leq 1$$

优化变量为 x 和 y , 定义域为 $\mathcal{D} = \{(x, y) | y > 0\}$ 。

- (a) 证明这是一个凸优化问题, 求解最优值。
- (b) 给出 Lagrange 对偶问题, 求解对偶问题的最优解 λ^* 和最优值 d^* 。给最优对偶间隙。
- (c) Slater 条件对此问题是否成立?

习题 11.3. 给定函数 $f(X) = \text{tr}(X^{-1})$, 定义域 $\text{dom } f = \mathbf{S}_{++}^n$ 。证明 $f(X)$ 的共轭函数为:

$$f^*(Y) = -2 \ln(-Y)^{1/2}, \quad \text{dom } f^* = -\mathbf{S}_+^n$$

习题 11.4. 考虑问题

$$\text{minimize } c^T x$$

$$\text{subject to } f(x) \leq 0$$

其中 $c \neq 0$ 。利用共轭 f^* 表达对偶问题。我们不假设函数 f 是凸的, 证明对偶问题是凸的。

习题 11.5. 求解线性规划

$$\text{minimize } e^T x$$

$$\text{subject to } Gx \leq h$$

$$Ax = b$$

的对偶函数, 给出对偶问题。

习题 11.6. 证明弱极大极小不等式

$$\sup_{z \in Z} \inf_{w \in W} f(w, z) \leq \inf_{w \in W} \sup_{z \in Z} f(w, z)$$

总是成立。函数 $f: \mathbf{R}^n \times \mathbf{R}^m \rightarrow \mathbf{R}$, $W \subseteq \mathbf{R}^n$, $Z \subseteq \mathbf{R}^m$ 任意。

习题 11.7. 写出下述非线性规划的 KKT 条件并求解

$$(1) \quad \text{maximize} \quad f(x) = (x - 3)^2$$

$$\text{subject to} \quad 1 \leq x \leq 5$$

$$(2) \quad \text{minimize} \quad f(x) = (x - 3)^2$$

$$\text{subject to} \quad 1 \leq x \leq 5$$

习题 11.8. 考虑等式约束的最小二乘问题

$$\text{minimize} \quad \|\mathbf{Ax} - \mathbf{b}\|_2^2$$

$$\text{subject to} \quad \mathbf{Gx} = \mathbf{h}$$

其中 $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\text{rank}(\mathbf{A}) = n$, $\mathbf{G} \in \mathbb{R}^{p \times n}$, $\text{rank}(\mathbf{G}) = p$. 给出 KKT 条件, 推导原问题最优解 x^* 以及对偶问题最优解 v^* 的表达式.

习题 11.9. 用 Lagrange 乘子法证明: 矩阵 $\mathbf{A} \in \mathbb{R}^{m \times n}$ 的 l_2 范数

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2 = 1, \mathbf{x} \in \mathbb{R}^n} \|\mathbf{Ax}\|_2$$

的平方是 $\mathbf{A}^T \mathbf{A}$ 的最大特征值。

习题 11.10. 用 Lagrange 乘子法求欠定方程 $\mathbf{Ax} = \mathbf{b}$ 的最小二范数解, 其中 $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \leq n$, $\text{rank}(\mathbf{A}) = m$

11.8 参考文献

A. Berman and A. Ben-Israel. More on linear inequalities with applications to matrix theory. *Journal of Mathematical Analysis and Applications*, 33:482-496, 1971.

R.Bellman and K.Fan.On systems of linear inequalities in Hermitian matrix variables.In V.L.Klee, editor, *Convexity*, volume VII of *Proceedings of the Symposia in Pure Mathematics* pages 1-11.American Mathematical Society, 1963.

A.Ben-Israel.Linear equations and inequalities on finite dimensional, real or complex vector spaces: A unified theory.*Journal of Mathematical Analysis and Applications* 27:367-389, 1969.

D.Bertsimas and J.N. Tsitsiklis. *Introduction to Linear Optimization*, Athena Scientific, 1997.

A.Ben-Tal and A.Nemirovski.*Lectures on Modern Convex Optimization*.Analysis, Algorithms, and Engineering Applications.Society for Industrial and Applied Mathematics, 2001.

D.P.Bortsckas.*Convex Analysis and Optimization*.Athena Scientific, 2003. With A.Nedic and A.E.Ozdaglar.

A.Berman.*Cones, Matrices and Mathematical Programming*. Springer, 1973.

- R.Courant and D.Hilbert.Method of Mathematical Physics. Volume 1. Interscience Publishers, 1953. Translated and revised from the 1937 German original.
- B.D.Craven and J.J.Koliha.Generalizations of Farkas' theorem.SIAM Journal on Numerical Analysis, 8(6), 1977.
- T.M.Cover and J.A.Thomas.Elements of Information Theory.John Wiley & Sons, 1991.
- J.Farkas.Theorie der einfachen Ungleichungen.Journal fur die Reine und Angewandte Mathematik, 124:1-27, 1902.
- D.Gale, H.W.Kuhn, and A.W.Tucker.Linear programming and the theory of games.In T.C.Koopmans, editor, Activity Analysis of Production and Allocation, volume 13 of Cowles Commission for Research in Economics Monographs, pages 317-335.John Wiley & Sons, 1951.
- J.-B.Hiriart-Urruty and C.Lemarechal. Convex Analysis and Minimization Algorithms.Springer, 1993.Two volumes.
- K.Isii.Inequalities of the types of ChebyKhev and Cramer-Rao and mathematical programming. Annals of The Institute of Statistical Mathematics, 16:277-293, 1964.
- F.John.Extremum problems with inequalities as subsidiary conditions.In J.Moser, editor, Pritz John, Collected Papers, pages 543-560.Birkhauser Verlag, 1985.First published in 1948.
- H.W.Kuhn and A.W.Tucker.Nonlinear programming.In J.Neyman, editor, Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability pages 481-492.University of California Press, 1951.
- H.W.Kuhn.Nonlinear programming, A historical view.In R.W.Cottle and C.E.Lemke, editors. Nonlinear Programming, volume 9 of SIAM-AMS Proceedings, pages 1-26.American Mathematical Society, 1976.
- J.B.Lasserre.A new Farkas lemma for positive semidefinite matrices.IEEE Transactions on Automatic Control 40(6):1131-1133.1995.
- D.G.Luenberger.Optimization by Vector Space Methods.John Wiley & Sons,1969.
- O.Mangasarian.Nonlinear Programming.Society for Industrial and Applied Mathematics, 1994.First published in 1969 by McGraw-Hill.
- Y.Nesterov and A.Nemirovskii.Interior-Point Polynomial Methods in Convex Programming.Society for Industrial and Applied Mathematics, 1994.
- J.Nocedal and S.J.Wright.Numerical Optimization Springer, 1999.
- R.T.Rockafellar.Convex Analysis.Princeton University Press, 1970.
- R.T.Rockafellar.Conjugate Duality and Optimization.Society for Industrial and Applied Mathematics, 1989.First published in 1974.
- S.M.Ross.An Introduction to Mathematical Finance: Options and Other Topics.Cambridge University Press, 1999.
- P.Whittle.Optimization under Constraints.John Wiley & Sons, 1971.

H.Wolkowicz. Some applications of optimization in matrix theory. *Linear Algebra and Its Applications*, 40:101-118, 1981.

J.von Neumann. Discussion of a maximum problem. In A.H.Taub, editor, John von Neumann. *Collected Works*, volume VI, pages 89-95. Pergamon Press, 1963. Unpublished working paper from 1947.

J.von Neumann and O.Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, third edition, 1953. First published in 1944.

数据科学与工程数学基础

第十二章 优化算法

对于特定的优化问题，可以找到问题的解析解。比如最小二乘问题。然而，很多问题并没有解析解，或者虽然有解析解，但是利用解析式求解最优值的方式需要极高的运算量。采用迭代的方法逐渐逼近一个最优解是一种可行的方式。优化算法可分为无约束优化算法和约束优化算法，其中无约束优化算法可分为零阶方法（一维搜索），一阶方法和二阶方法；约束优化算法可分为可行方向法和制约函数法。

本章主要介绍无约束优化和约束优化算法的性质和求解方法，除此之外，本章还介绍了深度学习中常用的优化算法，以便读者在实践中使用。

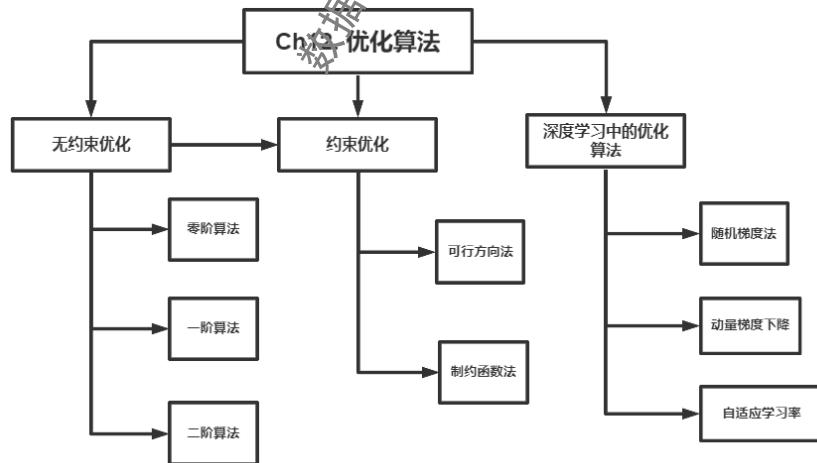


图 12.1: 本章导图

12.1 无约束优化

本节讨论下述无约束优化问题的求解方法

$$\min f(\mathbf{x}) \quad (12.1)$$

其中 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 的可微函数。

根据前一章对无约束优化问题最优化条件的讨论, 可知使得目标函数 $f(\mathbf{x})$ 的梯度等于零, 求得平稳点; 然后用充分条件进行判别, 便可求出所要的最优解。然而, 对于一般的 n 元函数 $f(\mathbf{x})$ 来说, 通常 $\nabla f(\mathbf{x}) = 0$ 是一个非线性方程组, 求它的解析解相当困难。对于不可微函数, 更无法使用这样的方法。为此, 常使用迭代法进行求解。

迭代法

迭代法的基本思想是: 为了求函数 $f(\mathbf{x})$ 的最优解, 首先给定一个初始估计 $\mathbf{x}^{(0)}$, 然后按某种规则 (即算法) 找出比 $\mathbf{x}^{(0)}$ 更好的解 $\mathbf{x}^{(1)}$ (对极小化问题, $f(\mathbf{x}^{(1)}) < f(\mathbf{x}^{(0)})$; 对极大化问题, $f(\mathbf{x}^{(1)}) > f(\mathbf{x}^{(0)})$), 再按此种规则找出比 $\mathbf{x}^{(1)}$ 更好的解 $\mathbf{x}^{(2)}, \dots$ 。如此即可得到一个解的序列 $\{\mathbf{x}^{(k)}\}$ 。若这个解序列有极限 \mathbf{x}^* , 即

$$\lim_{k \rightarrow \infty} \|\mathbf{x}^{(k)} - \mathbf{x}^*\| = 0$$

则称它收敛于 \mathbf{x}^* 。

若这算法是有效的, 那么它所产生的解的序列将收敛于该问题的最优解。除此之外, 算法的渐进收敛速度是衡量算法性能的一个重要指标。这里重点介绍 **Q-收敛速度** (**Q** 的含义是 quotient), 以及 **R-收敛速度** (**R** 的含义是 root)。

- **Q-收敛速度:** 设 \mathbf{x}^k 为算法产生的迭代点列且收敛于 \mathbf{x}^* , 对充分大的 k ,

若满足有

$$\frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} \leq a, a \in (0, 1), \quad (12.2)$$

则称算法是 **Q-线性收敛的**;

若满足

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} = 0, \quad (12.3)$$

则称算法是 **Q-超线性收敛的**;

若满足

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} = 1, \quad (12.4)$$

则称算法是 **Q-次线性收敛的**;

若满足

$$\frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|^2} \leq a, a > 0, \quad (12.5)$$

则称算法是 **Q**-二次收敛的。

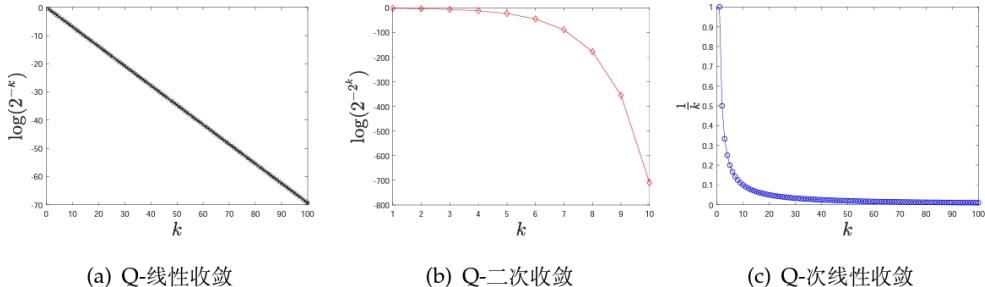


图 12.2: 不同 Q-收敛速度比较

从图12.2可以看出不同 Q 收敛速度的效果图。一般来说，具有 Q-超线性收敛速度和 Q-二次收敛速度的算法是收敛较快的。

- **R-收敛速度**: 设 \mathbf{x}^k 为算法产生的迭代点列且收敛于 \mathbf{x}^* ，若存在 Q-线性收敛于 0 的非负序列 t_k 并且对任意 k 成立

$$\|\mathbf{x}^k - \mathbf{x}^*\| \leq t_k, \quad (12.6)$$

则称算法是 **R**-线性收敛的。

类似地，可以定义 **R**-超线性收敛和 **R**-二次收敛等收敛速度。

不过，由于计算机只能进行有限次迭代，一般说很难得到准确解，而只能得到近似解。当满足所要求的精度时，即可停止迭代。又因为真正的最优解事先未知，通常根据相继两次迭代的结果，决定什么时候停止计算。常用的终止计算准则有以下几种。

- 根据相继两次迭代的绝对误差

$$\begin{aligned} \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| &< \varepsilon_1 \\ |f(\mathbf{x}^{(k+1)}) - f(\mathbf{x}^{(k)})| &< \varepsilon_2 \end{aligned}$$

- 根据相继两次迭代的相对误差

$$\begin{aligned} \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|}{\|\mathbf{x}^{(k)}\|} &< \varepsilon_3 \\ \frac{|f(\mathbf{x}^{(k+1)}) - f(\mathbf{x}^{(k)})|}{|f(\mathbf{x}^{(k)})|} &< \varepsilon_4 \end{aligned}$$

- 根据目标函数梯度的模足够小

$$\|\nabla f(\mathbf{x}^{(k)})\| < \varepsilon_5$$

其中， $\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4, \varepsilon_5$ 为事先给定的足够小的正数。

最优化问题的迭代方法

若由某算法所产生的解的序列 $\{\mathbf{x}^{(k)}\}$ 使目标函数值 $f(\mathbf{x}^{(k)})$ 逐步减少, 就称这算法为 **下降算法**。“下降”的要求比较容易实现, 它包含了很多种具体算法。显然, 求解极小化问题应采用下降算法。

现假定已迭代到点 $\mathbf{x}^{(k)}$, 若从 $\mathbf{x}^{(k)}$ 出发沿任何方向移动都不能使目标函数值下降, 则 $\mathbf{x}^{(k)}$ 是一局部极小点, 迭代停止。若从 $\mathbf{x}^{(k)}$ 出发至少存在一个方向可使目标函数值有所下降, 则可选定能使目标函数值下降的某方向 $\mathbf{p}^{(k)}$, 沿这个方向迈进适当一步, 得到下一个迭代点 $\mathbf{x}^{(k+1)}$, 并使 $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ 。这相当于在射线 $\mathbf{x} = \mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}$ 上选定新点 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}$ (见图 12.3), 其中, $\mathbf{p}^{(k)}$ 称为 **搜索方向**; λ_k 称为 **步长**或**步长因子**。

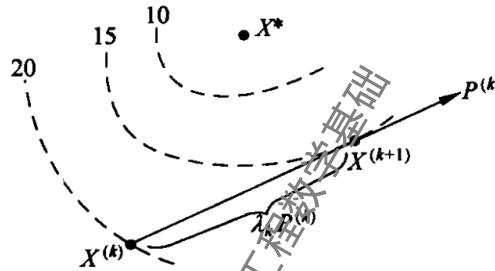


图 12.3: 下降算法迭代过程示意图

本文主要考虑的下降算法的步骤, 可总结如下:

算法 12.1 下降算法框架

- 1: 选定某一初始点 $\mathbf{x}^{(0)}$, 并令 $k := 0$;
- 2: 依据一定规则, 确定搜索方向 $\mathbf{p}^{(k)}$;
- 3: 从 $\mathbf{x}^{(k)}$ 出发, 沿方向 $\mathbf{p}^{(k)}$ 求步长 (步长因子) λ_k , 以产生下一个迭代点 $\mathbf{x}^{(k+1)}$;
- 4: 检查得到的新点 $\mathbf{x}^{(k+1)}$ 是否为极小点或近似极小点。若是, 则停止迭代。否则, 令 $k := k + 1$, 转回第二步继续进行迭代。

在以上步骤中, 存在两个**关键问题**: 一方面, 如何确定搜索方向 $\mathbf{p}^{(k)}$; 另一方面, 如何确定步长 λ_k 。确定步长 λ_k 的过程, 实际上是解决在确定搜索方向之后, 在该方向走多远的问题? 本质上, 它是一个一元函数的优化问题, 故称之为**线搜索** (一维搜索)。一般分为两类:

- **精确线搜索**, 即沿射线 $\mathbf{x} = \mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}$ 求目标函数 $f(\mathbf{x})$ 的极小, 换言之

$$\lambda_k = \arg \min_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})$$

这样确定的步长为**最佳步长**。

- **非精确线搜索**, 不要求 λ_k 是上述优化问题的极小点, 只要步长 λ_k 能使目标函数值下降充分即可;

根据利用目标函数的信息不同, 确定搜索方向的方法也有差异。我们将其分为如下两类:

- **一阶方法**: 该方法利用目标函数的梯度信息进行优化, 均为梯度类算法。适用于不需要很高精度的大数据优化问题, 例如: 机器学习、深度学习;
- **二阶方法**: 该方法利用目标函数的 Hessian 矩阵进行优化, 例如牛顿法、拟牛顿法。适用于需要高精度的优化问题, 例如: 科学计算。

下面我们将针对这些方面进行展开论述。

12.1.1 线搜索

精确线搜索

前面已提及线搜索分为精确和非精确线搜索, 我们先介绍精确线搜索。线搜索本质上是求如下优化问题:

$$\lambda_k = \arg \min_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}).$$

求解这一问题也称之为精确线搜索。这样得到的最优解具有很好的性质: 某种程度上, 下一个迭代点的函数值在该方向上已经不能够再下降(达到最优)。即在搜索方向上所得最优点处目标函数的梯度和该搜索方向正交。如下定理所述。

定理 12.1.1. 设目标函数 $f(\mathbf{x})$ 具有一阶连续偏导数, $\mathbf{x}^{(k+1)}$ 按照下述规则产生

$$\begin{cases} \lambda_k = \min_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)} \end{cases}$$

则有

$$\nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(k)} = 0 \quad (12.7)$$

证明. 构造函数 $\varphi(\lambda) = f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})$, 则得

$$\begin{cases} \varphi(\lambda_k) = \min_{\lambda} \varphi(\lambda) \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)} \end{cases}$$

即 λ_k 为 $\varphi(\lambda)$ 的极小点。此外

$$\varphi'(\lambda) = \nabla f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})^T \mathbf{p}^{(k)}.$$

由 $\varphi'(\lambda)|_{\lambda=\lambda_k} = 0$, 可得

$$\nabla f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})^T \mathbf{p}^{(k)} = \nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(k)} = 0$$

□

式(12.7)的几何意义见图12.4。图中 $p^{(k)}$ 就是前一迭代点 $x^{(k)}$ 的下降方向，当使用精确线搜索，达到 $k+1$ 个迭代点 $x^{(k+1)}$ 时，必然与该点的梯度方向垂直。若不然，梯度方向与搜索方向为钝角，则意味着函数值还可以继续下降，这与最小值矛盾。

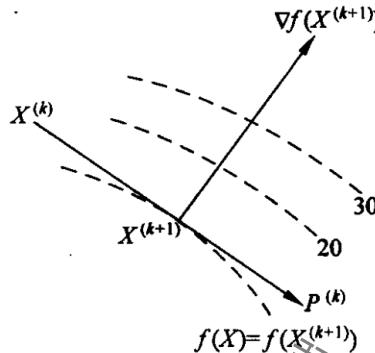


图 12.4

限于篇幅，下面仅介绍斐波那契法和0.618法。实际上，由于线搜索本质上是单变量函数的最优化问题。因此，一维优化方法均可用于此，包括：

- 试探法（“成功-失败”法，斐波那契法，0.618法等）；
- 插值法（抛物线插值法，三次插值法等）；
- 微积分中的求根法（切线法，二分法等）。

斐波那契法

设 $y = f(t)$ 是区间 $[a, b]$ 上的下单峰函数（图12.5），在此区间内它有唯一极小点 t^* 。若在此区间内任取两点 a_1 和 b_1 ， $a_1 < b_1$ ，并计算函数值 $f(a_1)$ 和 $f(b_1)$ ，可能出现以下两种情况：

- $f(a_1) < f(b_1)$ （图12.5(a)），这时极小点 t^* 必在区间 $[a, b_1]$ 内；
- $f(a_1) \geq f(b_1)$ （图12.5(b)），这时极小点 t^* 必在区间 $[a_1, b]$ 内。

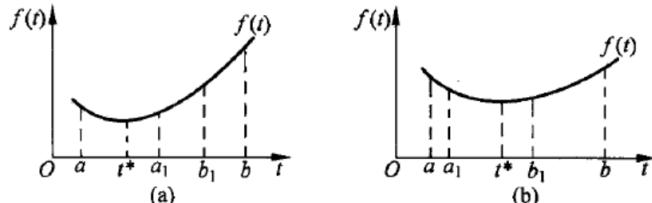


图 12.5

这说明, 只要在区间 $[a, b]$ 内取两个不同点, 并算出它们的函数值加以比较, 就可以把搜索区间 $[a, b]$ 缩小成 $[a, b_1]$ 或 $[a_1, b]$ (缩小后的区间仍需包含极小点)。现在, 如果要继续缩小搜索区间 $[a, b_1]$ (或 $[a_1, b]$), 就只需在上述区间内再取一点算出其函数值, 并与 $f(a_1)$ (或 $f(b_1)$) 加以比较即可。只要缩小后的区间包含极小点 t^* , 则区间缩小得越小, 就越接近于函数的极小点, 但计算函数值的次数也就越多。这就说明区间的缩短率和函数值的计算次数有关。现在要问, 计算函数值 n 次, 能把包含有极小点的区间缩小到什么程度呢? 或者换一种说法, 计算函数值 n 次能把原来多大的区间缩小成长度为一个单位的区间呢?

如果用 F_n 表示计算 n 个函数值能缩短为单位区间的最大原区间长度, 下面分三种情形并逐步深入地进行分析。

- 先考虑 $n \leq 1$ 的情形:

显然

$$F_0 = F_1 = 1 \quad (12.8)$$

其原因是, 只有当原区间长度本来就是一个单位长度时才不必计算函数值; 此外, 只计算一次函数值无法将区间缩短, 故只有区间长度本来就是单位区间才行。

- 考虑 $n = 2$ 的情形: 为方便描述, 把计算函数值的点称作试算点或试点。

在区间 $[a, b]$ 内取两个不同点 a_1 和 b_1 (图12.6(a)), 计算其函数值以缩短区间, 缩短后的区间为 $[a, b_1]$ 或 $[a_1, b]$ 。显然, 这两个区间长度之和必大于 $[a, b]$ 的长度, 也就是说, 计算两次函数值一般无法把长度大于两个单位的区间缩成单位区间。但是, 对于长度为两个单位的区间, 可以如图12.6(b) 那样选取试点 a_1 和 b_1 , 图中 ε 为任意小的正数, 缩短后的区间长度为 $1 + \varepsilon$ 。由于 ε 可任意选取, 故缩短后的区间长度接近于一个单位长度。由此可得 $F_2 = 2$ 。



图 12.6

- 考虑 $n > 2$ 的情形:

根据同样的分析 (见图12.7) 可得

$$F_3 = 3, F_4 = 5, F_5 = 8, \dots$$

序列 $\{F_n\}$ 可写成一个递推公式

$$F_n = F_{n-1} + F_{n-2}, \quad n \geq 2 \quad (12.9)$$

利用式(12.9), 可依次算出各 F_n 的值, 这些 F_n 就是通常所说的斐波那契数。

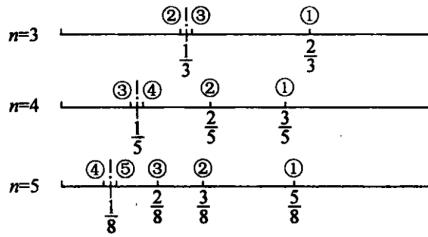


图 12.7

在总结斐波那契法具体步骤之前, 我们先通过简单的例子, 获得关于这一结果的一些直观认识。由以上讨论可知, 计算 n 次函数值所能获得的最大缩短率 (缩短后的区间长度与原区间长度之比) 为 $1/F_n$ 。例如 $F_{20} = 10946$, 所以计算 20 个函数值即可把原长度为 L 的区间缩短为

$$\frac{L}{10946} = 0.000091$$

的区间。现在, 要想计算 n 个函数值, 而把区间 $[a_0, b_0]$ 的长度缩短为原来长度的 δ 倍, 即缩短后的区间长度为

$$b_{n-1} - a_{n-1} \leq (b_0 - a_0)\delta$$

则只要 n 足够大, 能使下式成立即可

$$\frac{L}{\delta^n} \geq \frac{1}{\delta} \quad (12.10)$$

其中, δ 为一个正小数, 称为区间缩短的相对精度。有时给出区间缩短的绝对精度 η , 即要求

$$b_{n-1} - a_{n-1} \leq \eta$$

显然, 上述相对精度和绝对精度之间有如下关系

$$\eta = (b_0 - a_0)\delta$$

现在我们将斐波那契法缩短区间的具体步骤总结如下:

1. 根据需要的精度 δ , 确定试点的个数 n 。
2. 选取前两个试点的位置; 由式(12.9)可知第一次缩短时的两个试点位置分别是 (见图12.8):

$$\begin{cases} t_1 = a_0 + \frac{F_{n-2}}{F_n}(b_0 - a_0) \\ \quad = b_0 + \frac{F_{n-1}}{F_n}(a_0 - b_0) \\ t'_1 = a_0 + \frac{F_{n-1}}{F_n}(b_0 - a_0) \end{cases} \quad (12.11)$$

它们在区间内的位置是对称的;

3. 计算函数值 $f(t_1)$ 和 $f(t'_1)$, 并比较它们的大小; 若 $f(t_1) < f(t'_1)$, 则取

$$a_1 = a_0 \quad b_1 = t'_1 \quad t'_2 = t_1$$

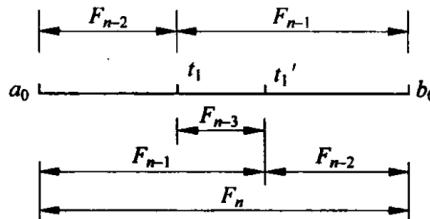


图 12.8

并令

$$t_2 = b_1 + \frac{F_{n-2}}{F_{n-1}}(a_1 - b_1)$$

否则, 取

$$a_1 = t_1 \quad b_1 = b_0 \quad t_2' = t_1'$$

并令

$$t_2' = a_1 + \frac{F_{n-2}}{F_{n-1}}(b_1 - a_1)$$

4. 计算 $f(t_2)$ 或 $f(t_2')$ (其中的一个已经算出), 如第 3 步那样一步步迭代。计算试点的一般公式为

$$\begin{cases} t_k = b_{k-1} + \frac{F_{n-k}}{F_{n-k+1}}(a_{k-1} - b_{k-1}) \\ t_k' = a_1 + \frac{F_{n-k}}{F_{n-k+1}}(b_{k-1} - a_{k-1}) \end{cases} \quad (12.12)$$

其中, $k = 1, 2, \dots, n-1$;

5. 当进行至 $k = n-1$ 时

$$t_{n-1} = t'_{n-1} = \frac{1}{2}(a_{n-2} + b_{n-2})$$

这就无法借比较函数值 $f(t_{n-1})$ 和 $f(t'_{n-1})$ 的大小以确定最终区间, 为此, 取

$$\begin{cases} t_{n-1} = \frac{1}{2}(a_{n-2} + b_{n-2}) \\ t'_{n-1} = a_{n-2} + \left(\frac{1}{2} + \varepsilon\right)(b_{n-2} - a_{n-2}) \end{cases} \quad (12.13)$$

其中 ε 为任意小的数。在 t_{n-1} 和 t'_{n-1} 这两点中, 以函数值较小者为近似极小点, 相应的函数值为近似极小值, 并得最终区间 $[a_{n-2}, t'_{n-1}]$ 或 $[t_{n-1}, b_{n-2}]$ 。

由上述分析可知, 斐波那契法使用对称搜索的方法, 逐步缩短所考察的区间, 它能以尽量少的函数求值次数, 达到预定的某一缩短率。

例 12.1.1. 试用斐波那契法求函数 $f(t) = t^2 - t + 2$ 的近似极小点和极小值, 要求缩短后的区间长度不大于区间 $[-1, 3]$ 的 0.08 倍。

解. 容易验证, 在此区间上函数 $f(t) = t^2 - t + 2$ 为严格凸函数。为了进行比较, 我们给出其精确解是: $t^* = 0.5$, $f(t^*) = 1.75$ 。

已知 $\delta = 0.08$, $F_n \geq 1/\delta = 1/0.08 = 12.5$, 由斐波那契序列得, $n = 6$ 。又 $a_0 = -1$, $b_0 = 3$, 故取

$$t_1 = b_0 + \frac{F_5}{F_6}(a_0 - b_0) = 3 + \frac{8}{13}(-1 - 3) = 0.538$$

$$t_1' = a_0 + \frac{F_5}{F_6}(b_0 - a_0) = -1 + \frac{8}{13}(3 - (-1)) = 1.462$$

$$f(t_1) = 0.538^2 - 0.538 + 2 = 1.751$$

$$f(t_1') = 1.462^2 - 1.462 + 2 = 2.675$$

由于 $f(t_1) < f(t_1')$, 故取 $a_1 = -1$, $b_1 = 1.462$, $t_2' = 0.538$

$$t_2 = b_1 + \frac{F_4}{F_5}(a_1 - b_1) = 1.462 + \frac{5}{8}(-1 - 1.462) = -0.077$$

$$f(t_2) = (-0.077)^2 - (-0.077) + 2 = 2.083$$

由于 $f(t_2) > f(t_2') = 1.751$, 故取 $a_2 = -0.077$, $b_2 = 1.462$, $t_3' = 0.538$

$$t_3' = a_2 + \frac{F_3}{F_4}(b_2 - a_2) = -0.077 + \frac{3}{5}(1.462 + 0.077) = 0.846$$

$$f(t_3') = 0.846^2 - 0.846 + 2 = 1.870$$

由于 $f(t_3') > f(t_3) = 1.751$, 故取 $a_3 = -0.077$, $b_3 = 0.846$, $t_4' = 0.538$

$$t_4 = b_3 + \frac{F_2}{F_3}(a_3 - b_3) = 0.846 + \frac{2}{3}(-0.077 - 0.846) = 0.231$$

$$f(t_4) = 0.231^2 - 0.231 + 2 = 1.822$$

由于 $f(t_4) > f(t_4') = 1.751$, 故取 $a_4 = 0.231$, $b_4 = 0.846$, $t_5' = 0.538$ 。现令 $\varepsilon = 0.01$, 则

$$t_5' = a_4 + \left(\frac{1}{2} + \varepsilon\right)(b_4 - a_4)$$

$$= 0.231 + (0.5 + 0.01)(0.846 - 0.231) = 0.545$$

$$f(t_5') = 0.545^2 - 0.545 + 2 = 1.752 > f(t_5) = 1.751$$

故取 $a_5 = 0.231$, $b_5 = 0.545$ 。由于 $f(t_5) = 1.751 < f(t_5') = 1.752$, 所以 t_5 为近似极小点, 近似极小值为 1.751。

缩短后的区间长度为 $0.545 - 0.231 = 0.314$, 显然 $0.314/4 = 0.0785 < 0.08$ 。

0.618 法

由上节可知, 当用斐波那契法以 n 个试点来缩短某一区间时, 区间长度的第一次缩短率为 F_{n-1}/F_n , 其后各次分别为

$$\frac{F_{n-2}}{F_{n-1}}, \quad \frac{F_{n-3}}{F_{n-2}}, \quad \dots, \quad \frac{F_1}{F_2}$$

现将以上数列分为奇数项 F_{2k-1}/F_{2k} 和偶数项 F_{2k}/F_{2k+1} ，可以证明，这两个数列收敛于同一个极限。

设当 $k \rightarrow \infty$ 时

$$\frac{F_{2k-1}}{F_{2k}} \rightarrow \lambda \quad \frac{F_{2k}}{F_{2k+1}} \rightarrow \mu$$

由于

$$\frac{F_{2k-1}}{F_{2k}} = \frac{F_{2k-1}}{F_{2k-1} + F_{2k-2}} = \frac{1}{1 + \frac{F_{2k-2}}{F_{2k-1}}}$$

故当 $k \rightarrow \infty$ 时

$$\lim_{k \rightarrow \infty} \frac{F_{2k-1}}{F_{2k}} = \frac{1}{1 + \mu} = \lambda \quad (12.14)$$

同理可证

$$\mu = \frac{1}{1 + \lambda} \quad (12.15)$$

将式(12.14)带入式(12.15)得

$$\mu = \frac{1 + \mu}{2 + \mu}$$

即

$$\mu^2 + \mu - 1 = 0$$

从而可得

$$\frac{\sqrt{5} - 1}{2}$$

若把式(12.15)带入式(12.14)，则得

$$\lambda^2 + \lambda - 1 = 0$$

故有

$$\lambda = \mu = \frac{\sqrt{5} - 1}{2} = 0.6180339887418948 \quad (12.16)$$

这促使人们考虑用不变的区间缩短率 0.618，代替斐波那契法每次不同的缩短率，就得到了 0.618 法。每次的试点均取在区间长度的 0.618 倍和 0.382 倍处。显然，0.618 法是一种等速对称进行试探的方法。

当用 0.618 法时，计算 n 个试点的函数值可以把原区间 $[a_0, b_0]$ 连续缩短 $n - 1$ 次，因为每次的缩短率均为 μ ，故最后的区间长度为

$$(b_0 - a_0)\mu^{n-1}$$

这就是说，当已知缩短的相对精度为 δ 时，可用下式计算试点个数 n

$$\mu^{n-1} \leq \delta \quad (12.17)$$

这个方法可以看成是斐波那契法的近似，实现起来比较容易，效果也相当好，因而易于为人们所接受。

非精确线搜索

需要指出的是, 尽管使用精确线搜索算法时我们可以在多数情况下得到优化问题的解, 但这样选取的步长通常需要很大计算量, 在实际应用中较少使用。另一个想法: 不要求步长是最小值点, 而是仅仅要求它是满足某些不等式性质的近似解, 这种线搜索方法被称为非精确线搜索算法。由于非精确线搜索算法结构简单, 在实际应用中较为常见。

在非精确线搜索算法中, 若选取不合适的线搜索准则将会导致算法无法收敛。为便于理解这一点, 我们给出一个例子。

例 12.1.2. 考虑一维无约束优化问题

$$\min_x f(x) = x^2,$$

迭代初始点 $x^0 = 1$ 。由于问题是一维的, 下降方向只有 $\{-1, +1\}$ 两种。我们选取 $d^k = -\text{sign}(x^k)$, 且只要求选取的步长满足迭代点处函数值单调下降, 即 $f(x^k + \lambda_k d^k) < f(x^k)$ 。考虑选取如下两种步长:

$$\lambda_{k,1} = \frac{1}{3^{k+1}}, \quad \lambda_{k,2} = 1 + \frac{2}{3^{k+1}},$$

通过简单计算可以得到

$$x_1^k = \frac{1}{2} \left(1 + \frac{1}{3^k} \right), \quad x_2^k = \frac{(-1)^k}{2} \left(1 + \frac{1}{3^k} \right).$$

显然, 序列 $\{f(x_1^k)\}$ 和序列 $\{f(x_2^k)\}$ 均单调下降, 但序列 $\{x_1^k\}$ 收敛的点不是极小值点, 序列 $\{x_2^k\}$ 则在原点左右振荡, 不存在极限。

出现上述情况的原因是在迭代过程中函数值 $f(x^k)$ 的下降量不够充分, 以至于算法无法收敛到极小值点。为了避免这种情况发生, 必须引入一些更合理的线搜索准则来确保迭代的收敛性。

Armijo 准则 首先引入 Armijo 准则, 它是一个常用的线搜索准则。引入 Armijo 准则的目的是保证每一步迭代充分下降。

定义 12.1.1. (Armijo 准则) 设 d^k 是点 x^k 处的下降方向, 若

$$f(x^k + \lambda d^k) \leq f(x^k) + c_1 \lambda \nabla f(x^k)^T d^k,$$

则称步长 λ 满足 Armijo 准则, 其中 $c_1 \in (0, 1)$ 是一个常数。

Armijo 准则有非常直观的几何含义, 它指的是点 $(\lambda, \phi(\lambda))$ 必须在直线

$$l(\lambda) = \phi(0) + c_1 \lambda \nabla \phi(0)^T d^k$$

的下方。如图 12.9 所示, 区间 $[0, \lambda_1]$ 中的点均满足 Armijo 准则。我们注意到 d^k 为下降方向, 这说明 $l(\lambda)$ 的斜率为负, 选取符合 Armijo 准则的 λ 确实会使得函数值下降。在实际应用中, 参数 c_1 通常选为一个很小的正数, 例如 $c_1 = 10^{-3}$, 这使得 Armijo 准则非常容易得到满足。但是仅仅使

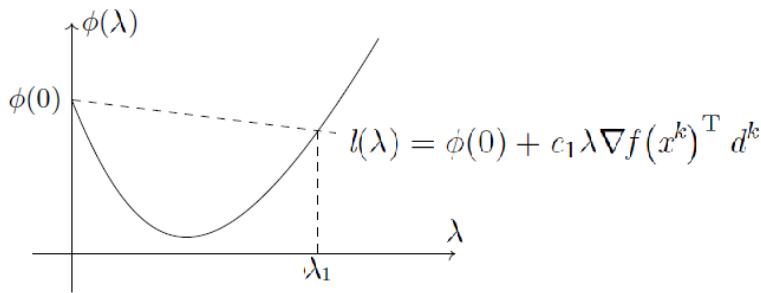


图 12.9: Armijo 准则

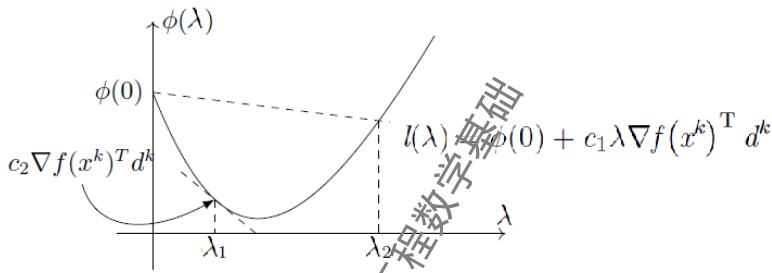


图 12.10: Wolfe 准则

用 Armijo 准则并不能保证迭代的收敛性. 这是因为 $\lambda = 0$ 显然满足条件, 而这意味着迭代序列中的点固定不变, 研究这样的步长是没有意义的. 为此, Armijo 准则需要配合其他准则共同使用.

Wolfe 准则 为了克服 Armijo 准则的缺陷, 我们需要引入其他准则来保证每一步的步长不会太小. 为此我们引入 Armijo-Wolfe 准则, 简称 Wolfe 准则.

定义 12.1.2. (Wolfe 准则) 设 \mathbf{d}^k 是点 \mathbf{x}^k 处的下降方向, 若

$$f(\mathbf{x}^k + \lambda \mathbf{d}^k) \leq f(\mathbf{x}^k) + c_1 \lambda \nabla f(\mathbf{x}^k)^T \mathbf{d}^k,$$

$$\nabla f(\mathbf{x}^k + \lambda \mathbf{d}^k)^T \mathbf{d}^k \geq c_2 \nabla f(\mathbf{x}^k)^T \mathbf{d}^k,$$

则称步长 λ 满足 Wolfe 准则, 其中 $c_1, c_2 \in (0, 1)$ 为给定的常数且 $c_1 < c_2$.

在 Wolfe 准则中, 第一个不等式即是 Armijo 准则, 而第二个不等式则是 Wolfe 准则的本质要求. 注意到 $\nabla f(\mathbf{x}^k + \lambda \mathbf{d}^k)^T \mathbf{d}^k$ 恰好就是 $\phi(\lambda)$ 的导数, Wolfe 准则实际要求 $\phi(\lambda)$ 在点 λ 处切线的斜率不能小于 $\phi'(0)$ 的 c_2 倍. 如图 12.10 所示, 在区间 $[\lambda_1, \lambda_2]$ 中的点均满足 Wolfe 准则. 注意到在 $\phi(\lambda)$ 的极小值点 λ^* 处有 $\phi'(\lambda^*) = \nabla f(\mathbf{x}^k + \lambda^* \mathbf{d}^k)^T \mathbf{d}^k = 0$, 因此 λ^* 永远满足第二个不等式. 而选择较小的 c_1 可使得 λ^* 同时满足第一个不等式条件, 即 Wolfe 准则在绝大多数情况下会包含线搜索子问题的精确解. 在实际应用中, 参数 c_2 通常取为 0.9.

非精确线搜索算法 在优化算法的实现中, 寻找一个满足 Armijo 准则的步长是比较容易的, 一个最常用的算法是回退法. 给定初值 $\hat{\lambda}$, 回退法通过不断以指数方式缩小试探步长, 找到第一个满足 Armijo 准则的点. 回退法的基本过程如下所示:

算法 12.2 回退法

- 1: 选择初始步长 $\hat{\lambda}$, 参数 $\gamma, c \in (0, 1)$. 初始化 $\lambda \leftarrow \hat{\lambda}$
- 2: **while** $f(\mathbf{x}^k + \lambda \mathbf{d}^k) > f(\mathbf{x}^k) + c \lambda \nabla f(\mathbf{x}^k)^T \mathbf{d}^k$ **do**
- 3: $\lambda \leftarrow \gamma \lambda$.
- 4: **end while**
- 5: 输出 $\lambda_k = \lambda$.

具体来说, 回退法选取

$$\lambda_k = \gamma^{j_0} \hat{\lambda},$$

其中

$$j_0 = \min \left\{ j = 0, 1, \dots \mid f(\mathbf{x}^k + \gamma^j \hat{\lambda} \mathbf{d}^k) \leq f(\mathbf{x}^k) + c_1 \gamma^j \hat{\lambda} \nabla f(\mathbf{x}^k)^T \mathbf{d}^k \right\}$$

参数 $\gamma \in (0, 1)$ 为一个给定的实数. 该算法被称为回退法是因为 λ 的试验值是由大至小的, 它可以确保输出的 λ_k 能尽量地大. 此外该算法不会无限进行下去, 因为 \mathbf{d}^k 是一个下降方向, 当 λ 充分小时, Armijo 准则总是成立的.

回退法的实现简单、原理直观, 所以它是最常用的线搜索算法之一. 然而, 回退法的缺点也很明显: 第一, 回退法以指数的方式缩小步长, 因此对初值 $\hat{\lambda}$ 和参数 γ 的选取比较敏感, 当 γ 过大时每一步试探步长改变量很小, 此时回退法效率比较低, 当 γ 过小时回退法过于激进, 导致最终找到的步长太小, 错过了选取大步长的机会. 为了提高回退法的效率, 还有其他类型基于多项式插值的线搜索算法. 第二, 它无法保证找到满足 Wolfe 准则的步长, 但对一些优化算法而言, 找到满足 Wolfe 准则的步长是十分必要的. 为此, Fletcher 提出了一个用于寻找满足 Wolfe 准则的算法. 这个算法比较复杂, 有较多细节, 这里不展开阐述.

12.1.2 一阶方法

上一小节, 我们讨论了线搜索, 即确定步长的相关方法. 本小节开始考虑确定搜索方向的问题. 我们期望的搜索方向应当能够保证函数值在局部范围内下降, 而且在局部范围内尽可能“最优”. 根据对目标函数的近似程度不同, 这些方法可分为一阶方法(梯度类方法)和二阶方法(牛顿类方法). 本小节, 我们将一起探讨被机器学习领域广泛应用的一阶方法(梯度类方法).

梯度下降法

在求解无约束优化问题中, 梯度法是最为古老但又十分基本的一种数字方法, 却依然在深度学习领域中发挥着作用。它的迭代过程简单, 使用方便, 而且是理解某些其它最优化方法的基础, 所以我们先来说明这一方法。

假定无约束优化问题中的目标函数 $f(\mathbf{x})$ 有一阶连续偏导数, 具有极小点 \mathbf{x}^* 。设 $\mathbf{x}^{(k)}$ 表示极小点的第 k 次近似, 为了探讨当前的搜索方向应满足的性质, 不妨设此时方向为 \mathbf{p}^k 。那么第 $k+1$ 次近似点 $\mathbf{x}^{(k+1)}$ 可以表示为

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)} \quad (\lambda \geq 0)$$

其中 λ 为步长。我们期望函数值下降 $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$, 即

$$f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) < f(\mathbf{x}^{(k)}).$$

在小范围内, $f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)})$ 可以由 $\mathbf{x}^{(k)}$ 点处的泰勒级数很好地近似。利用泰勒展开有

$$f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) = f(\mathbf{x}^{(k)}) + \lambda \nabla f(\mathbf{x}^{(k)})^T \mathbf{p}^{(k)} + o(\lambda)$$

其中

$$\lim_{\lambda \rightarrow 0^+} \frac{o(\lambda)}{\lambda} = 0$$

因此, 对于充分小的 λ , 只要

$$\nabla f(\mathbf{x}^{(k)})^T \mathbf{p}^{(k)} < 0 \quad (12.18)$$

即可保证 $f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) < f(\mathbf{x}^{(k)})$ 。这时若取

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}$$

就能使目标函数值得到改善。易知, 即便假定 $\mathbf{p}^{(k)}$ 的模一定 (且不为零), 并设 $\nabla f(\mathbf{x}^{(k)}) \neq 0$ (否则, $\mathbf{x}^{(k)}$ 是平稳点), 能使式(12.18)成立的 $\mathbf{p}^{(k)}$ 也有无限多个, 那么我们应该选择哪个方向呢?

为了使目标函数值能得到尽量大的改善, 必须寻求使 $\nabla f(\mathbf{x}^{(k)})^T \mathbf{p}^{(k)}$ 取最小值的 $\mathbf{p}^{(k)}$ 。由线性代数学知道

$$\nabla f(\mathbf{x}^{(k)})^T \mathbf{p}^{(k)} = \|\nabla f(\mathbf{x}^{(k)})\| \cdot \|\mathbf{p}^{(k)}\| \cos \theta \quad (12.19)$$

式中 θ 为向量 $\nabla f(\mathbf{x}^{(k)})$ 与 $\mathbf{p}^{(k)}$ 的夹角。当 $\mathbf{p}^{(k)}$ 与 $\nabla f(\mathbf{x}^{(k)})$ 反向时, $\theta = 180^\circ$, $\cos \theta = -1$ 。这时式(12.18)成立, 而且其左端取最小值。我们称方向

$$\mathbf{p}^{(k)} = -\nabla f(\mathbf{x}^{(k)})$$

为负梯度方向, 它是使函数值下降最快的方向 (在 $\mathbf{x}^{(k)}$ 的某一小范围内)。将该方向作为搜索方向, 便得到了梯度下降法算法 12.3。

再次强调, 这里仅是讨论了搜索方向, 从算法中可以看出还需要结合前一小节的方式确定步长。事实上, 在实际计算中人们可能会采取更为简单的方式确定步长。有时可以采取合适的固定步长的方式。有时也可采用可接受点算法, 就是取某一 λ 进行试算, 看是否满足不等式

$$f(\mathbf{x}^{(k)} - \lambda \nabla f(\mathbf{x}^{(k)})) < f(\mathbf{x}^{(k)}) \quad (12.20)$$

算法 12.3 梯度下降法

- 1: 给定初始近似点 $\mathbf{x}^{(0)}$ 及精度 $\varepsilon > 0$, 计算 $\nabla f(\mathbf{x}^{(0)})$ 。
- 2: 若 $\|\nabla f(\mathbf{x}^{(0)})\|^2 \leq \varepsilon$, 则 $f(\mathbf{x}^{(0)})$ 即为近似极小点;
- 3: 若 $\|\nabla f(\mathbf{x}^{(0)})\|^2 > \varepsilon$, 利用线搜索确定步长 λ_0 , 并计算

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \lambda_0 \nabla f(\mathbf{x}^{(0)}).$$

- 4: 一般地, 设已迭代到点 $\mathbf{x}^{(k)}$, 计算 $\nabla f(\mathbf{x}^{(k)})$ 。若 $\|\nabla f(\mathbf{x}^{(k)})\|^2 \leq \varepsilon$, 则 $\mathbf{x}^{(k)}$ 即为所求的近似解; 若 $\|\nabla f(\mathbf{x}^{(k)})\|^2 > \varepsilon$, 则求步长 λ_k , 并确定下一个近似点

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \lambda_k \nabla f(\mathbf{x}^{(k)})$$

如此继续, 直至达到要求的精度为止。

若上述不等式成立, 就可以迭代下去。否则, 缩小 λ 使满足不等式式(12.20)。由于采用负梯度方向, 满足式(12.20)的 λ 总是存在的。如果采用的是精确线搜索, 即通过在负梯度方向的一维搜索, 来确定使 $f(\mathbf{x})$ 最小的 λ_k , 这种梯度下降法就是所谓的最速下降法。

例 12.1.3. 试用梯度法求

$$f(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 1)^2$$

的极小点, 已知 $\varepsilon = 0.1$ 。

解. 取初始点 $\mathbf{x}^{(0)} = (0, 0)^T$

$$\begin{aligned}\nabla f(\mathbf{x}^{(0)}) &= [2(x_1 - 1), 2(x_2 - 1)]^T \\ \nabla f(\mathbf{x}^{(0)}) &= (-2, -2)^T\end{aligned}$$

$$\|\nabla f(\mathbf{x}^{(0)})\|^2 = (-2)^2 + (-2)^2 = 8 > \varepsilon$$

令 $\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \lambda_0 \nabla f(\mathbf{x}^{(0)}) = \begin{pmatrix} 2\lambda_0 \\ 2\lambda_0 \end{pmatrix}$, 代入 $f(x)$, 可得:

$$f(\mathbf{x}^{(1)}) = (2\lambda_0 - 1)^2 + (2\lambda_0 - 1)^2$$

要使得上式最小, 令 $df(\mathbf{x}^{(1)})/d\lambda_0 = 0$, 可得

$$\lambda_0 = \frac{1}{2}$$

因此,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \lambda_0 \nabla f(\mathbf{x}^{(0)}) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\nabla f(\mathbf{x}^{(1)}) = [2(1 - 1), 2(1 - 1)]^T = (0, 0)^T$$

故 $\mathbf{x}^{(1)}$ 即为极小点。

由这个例子可知, 对于目标函数的等值线为圆的问题来说, 不管初始点位置取在哪里, 负梯度方向总是直指圆心, 而圆心即为极值点。这样, 只要一次迭代即可达到最优解。前面提到有时还可采用固定步长求解, 甚至可以得到如下收敛定理。

定理 12.1.2. 设函数 $f(x)$ 为凸的梯度 L -利普希茨连续函数, $f^* = f(x^*) = \inf_x f(x)$ 存在且可达。如果步长 λ_k 取为常数 λ 且满足 $0 < \lambda \leq \frac{1}{L}$, 那么由梯度下降法得到的点列 $\{x^{(k)}\}$ 的函数值收敛到最优值, 且在函数值的意义下收敛速度为 $\mathcal{O}(\frac{1}{k})$ 。

证明. 因为函数 f 是利普希茨可微函数, 对任意的 x , 根据梯度 L -利普希茨连续的性质:

$$f(x - \lambda \nabla f(x)) \leq f(x) - \lambda \left(1 - \frac{L\lambda}{2}\right) \|\nabla f(x)\|^2.$$

现在记 $\tilde{x} = x - \lambda \nabla f(x)$, 我们有

$$\begin{aligned} f(\tilde{x}) &\leq f(x) - \frac{\lambda}{2} \|\nabla f(x)\|^2 \leq f^* + \nabla f(x)^T (x - x^*) - \frac{\lambda}{2} \|\nabla f(x)\|^2 \text{ (凸性)} \\ &= f^* + \frac{1}{2\lambda} \left(\|x - x^*\|^2 - \|x - x^* - \lambda \nabla f(x)\|^2 \right) \\ &= f^* + \frac{1}{2\lambda} \left(\|x - x^*\|^2 - \|\tilde{x} - x^*\|^2 \right). \end{aligned}$$

在上式中取 $x = x^{(i-1)}$, $\tilde{x} = x^{(i)}$ 并将不等式对 $i = 1, 2, \dots, k$ 求和得到

$$\begin{aligned} \sum_{i=1}^k \left(f(x^{(i)}) - f^* \right) &\leq \frac{1}{2\lambda} \sum_{i=1}^k \left(\|x^{(i-1)} - x^*\|^2 - \|x^{(i)} - x^*\|^2 \right) \\ &= \frac{1}{2\lambda} \left(\|x^{(0)} - x^*\|^2 - \|x^{(k)} - x^*\|^2 \right) \\ &\leq \frac{1}{2\lambda} \|x^{(0)} - x^*\|^2. \end{aligned}$$

易知 $f(x^{(i)})$ 是非增的, 所以

$$f(x^{(k)}) - f^* \leq \frac{1}{k} \sum_{i=1}^k \left(f(x^{(i)}) - f^* \right) \leq \frac{1}{2k\lambda} \|x^{(0)} - x^*\|^2.$$

□

例 12.1.4. 试求 $f(x) = x_1^2 + 25x_2^2$ 的极小点。

解. 取初始点 $x^{(0)} = (2, 2)^T$, 固定步长 $\lambda = 0.01$, 其迭代过程如表 12.1 所示。

通过这个例子, 我们可以观察到: 通过迭代在开头几步, 目标函数值下降较快, 但接近极小点 x^* 时, 收敛速度就不理想了。特别是当目标函数的等值线椭圆比较扁平时, 收敛速度就更慢了。因此, 在实用中, 常将梯度法和其它方法(后面介绍的二阶方法)联合起来应用。在前期使用梯度法, 而在接近极小点时, 则使用收敛较快的其它方法。

步骤	点	x_1	x_2	$\frac{\partial f(\mathbf{x}^{(k)})}{\partial x_1}$	$\frac{\partial f(\mathbf{x}^{(k)})}{\partial x_2}$	$\ \nabla f(\mathbf{x}^{(k)})\ $
0	$\mathbf{x}^{(0)}$	2	2	4	100	100.08
1	$\mathbf{x}^{(1)}$	1.96	1.00	3.92	50	50.15
2	$\mathbf{x}^{(2)}$	1.92	0.50	3.84	25	25.29
3	$\mathbf{x}^{(3)}$	1.88	0.25	3.76	12.5	13.06
...
200	$\mathbf{x}^{(200)}$	3.45×10^{-2}	6.22×10^{-61}	6.89×10^{-2}	3.11×10^{-59}	0.07

表 12.1

共轭梯度法

由于负梯度方向的最速下降性，很容易使人们认为负梯度方向是理想的搜索方向，最速下降法是一种理想的极小化方法。必须指出， \mathbf{x} 点处的负梯度方向 $-\nabla f(\mathbf{x})$ ，仅在 \mathbf{x} 点附近才具有这种“最速下降”的性质，而对于整个极小化过程来说，那就是另外一回事了。由例 12.1.3 可知，若目标函数的等值线为一族同心圆（或同心球面），则从任意初始点出发，沿最速下降方向一步即可达到极小点。但通常的情况并不是这样，例如 12.1.4，一般二元二次凸函数的等值线为一族同心椭圆，当用最速下降法趋近极小点时，其搜索路径呈直角锯齿状（图 12.11）。实际上，对于

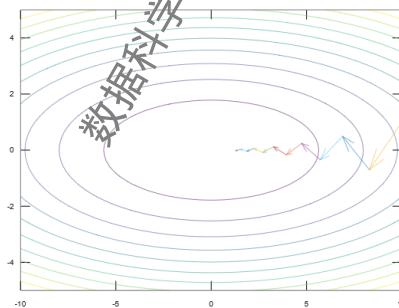


图 12.11：搜索路径呈直角锯齿状

正定二次函数，我们还可以寻找到更好的下降方向，即接下来要介绍的共轭梯度法。

(1) 共轭方向

设 \mathbf{x} 和 \mathbf{y} 是 n 维向量，若有

$$\mathbf{x}^T \mathbf{y} = 0$$

就称 \mathbf{x} 和 \mathbf{y} 正交。再设 A 为 $n \times n$ 对称正定阵，如果 \mathbf{x} 和 $A\mathbf{y}$ 正交，即有

$$\mathbf{x}^T A \mathbf{y} = 0 \tag{12.21}$$

则称 \mathbf{x} 和 \mathbf{y} 关于 \mathbf{A} 共轭, 或 \mathbf{x} 和 \mathbf{y} 为 \mathbf{A} 共轭 (\mathbf{A} 正交)。

一般地, 设 \mathbf{A} 为 $n \times n$ 对称正定阵, 若非零向量组 $\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(n)}$ 满足条件

$$(\mathbf{p}^{(i)})^T \mathbf{A} \mathbf{p}^{(j)} = 0 \quad (i \neq j; \quad i, j = 1, 2, \dots, n) \quad (12.22)$$

则称该向量组为 \mathbf{A} 共轭。特别地, 如果 $\mathbf{A} = \mathbf{I}$ (单位阵), 则上述条件即为通常的正交条件。因此, \mathbf{A} 共轭概念实际上是通常正交概念的推广。

定理 12.1.3. 设 \mathbf{A} 为 $n \times n$ 对称正定阵, $\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(n)}$ 为 \mathbf{A} 共轭的非零向量, 则这一组向量线性无关。

证明. 设向量 $\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(n)}$ 之间存在如下线性关系

$$\alpha_1 \mathbf{p}^{(1)} + \alpha_2 \mathbf{p}^{(2)} + \dots + \alpha_n \mathbf{p}^{(n)} = 0$$

对 $i = 1, 2, \dots, n$, 用 $(\mathbf{p}^{(i)})^T \mathbf{A}$ 左乘上式得

$$\alpha_i (\mathbf{p}^{(i)})^T \mathbf{A} \mathbf{p}^{(i)} = 0$$

但 $\mathbf{p}^{(i)} \neq 0, \mathbf{A}$ 为正定, 即

$$(\mathbf{p}^{(i)})^T \mathbf{A} \mathbf{p}^{(i)} > 0$$

故必有

$$\alpha_i = 0, \quad i = 1, 2, \dots, n$$

从而 $\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \dots, \mathbf{p}^{(n)}$ 线性无关。 □

(2) 正定二次函数的共轭梯度法

现在利用共轭的关系, 求解一类特殊的无约束优化问题

$$\min f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c \quad (12.23)$$

式中 \mathbf{A} 为 $n \times n$ 对称正定阵; \mathbf{x}, \mathbf{b} 为 n 维向量; c 为常数。问题式(12.23)称为正定二次函数极小问题, 它在最优化问题中起到极其重要的作用。下述定理将告诉我们, 求解正定二次函数的优化问题, 只需要找到一组 (n 个) 共轭方向, 而不是负梯度方向。在理想的情况下, 二次正定函数的优化问题只要迭代 n 步就会终止, 并找到最优解 \mathbf{x}^* 。

定理 12.1.4. 设向量 $\mathbf{p}^{(i)}, i = 0, 1, 2, \dots, n-1$, 为 \mathbf{A} 共轭, 则从任一点 $\mathbf{x}^{(0)}$ 出发, 相继以 $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \dots, \mathbf{p}^{(n-1)}$ 为搜索方向的下述算法

$$\begin{cases} \min_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) = f(\mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}) \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)} \end{cases}$$

经 n 次一维搜索收敛于问题式(12.23)的极小点 \mathbf{x}^* 。

证明. 由式(12.23)

$$\nabla f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$$

设相继各次搜索得到的近似解分别为 $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}$, 则

$$\nabla f(\mathbf{x}^{(k)}) = \mathbf{A}\mathbf{x}^{(k)} + \mathbf{b}$$

以及

$$\begin{aligned}\nabla f(\mathbf{x}^{(k+1)}) &= \mathbf{A}\mathbf{x}^{(k+1)} + \mathbf{b} \\ &= \mathbf{A}(\mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}) + \mathbf{b} \\ &= \nabla f(\mathbf{x}^{(k)}) + \lambda_k \mathbf{A}\mathbf{p}^{(k)}\end{aligned}$$

假定 $\nabla f(\mathbf{x}^{(k)}) \neq 0, k = 0, 1, 2, \dots, n-1$, 否则已经找到了最优解。这时有

$$\begin{aligned}\nabla f(\mathbf{x}^{(n)}) &= \nabla f(\mathbf{x}^{(n-1)}) + \lambda_{n-1} \mathbf{A}\mathbf{p}^{n-1} \\ &= \dots \\ &= \nabla f(\mathbf{x}^{(k+1)}) + \lambda_{k+1} \mathbf{A}\mathbf{p}^{k+1} + \lambda_{k+2} \mathbf{A}\mathbf{p}^{(k+2)} + \dots + \lambda_{n-1} \mathbf{A}\mathbf{p}^{(n-1)}\end{aligned}$$

由于在进行一维搜索时, 为确定最佳步长 λ_k , 令

$$\frac{df(\mathbf{x}^{(k+1)})}{d\lambda} = \frac{df[\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}]}{d\lambda} = \nabla f(\mathbf{x}^{(k+1)})^\top \mathbf{p}^{(k)} = 0 \quad (12.24)$$

故对 $k = 0, 1, 2, \dots, n-1$ 有

$$(\mathbf{p}^{(k)})^\top \nabla f(\mathbf{x}^{(n)}) = (\mathbf{p}^{(k)})^\top \nabla f(\mathbf{x}^{(k+1)}) + \lambda_{k+1} (\mathbf{p}^{(k)})^\top \mathbf{A}\mathbf{p}^{(k+1)} + \dots + \lambda_{n-1} (\mathbf{p}^{(k)})^\top \mathbf{A}\mathbf{p}^{(n-1)} = 0$$

这就是 $\nabla f(\mathbf{x}^{(n)})$ 和 n 个线性无关的向量 $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \dots, \mathbf{p}^{(n-1)}$ (它们为 \mathbf{A} 共轭) 正交, 从而必有

$$\nabla f(\mathbf{x}^{(n)}) = 0$$

即 \mathbf{x}^n 为 $f(\mathbf{x})$ 的极小点 \mathbf{x}^* 。 □

下面我们就二维正定二次函数的情况加以说明, 以便对上述定理有更加直观的认识。在前面使用梯度法的图12.11中, 可以看到搜索路径呈直角锯齿状。通常的迭代过程是不可能在有限步之内结束的。那么共轭方向是如何做到的呢? 如图12.12所示, 图中的一族共心椭圆来代表二维正定二次函数的等值线。而椭圆族的中心表示该二次函数的极小点 \mathbf{x}^* 。其中, $\mathbf{p}^{(0)}$ 与 $\mathbf{p}^{(1)}$ 即为二维空间的一组共轭方向。首先在 $\mathbf{x}^{(0)}$ 沿着方向 $\mathbf{p}^{(0)}$ 迭代至 $\mathbf{x}^{(2)}$ 。然后, 沿着共轭方向 $\mathbf{p}^{(1)}$ 方向进行搜索。因为 $\mathbf{p}^{(1)}$ 方向经过极小点。所以, 在第二次迭代时, 利用精确线搜索必将找到最优点 \mathbf{x}^* 。

那么如何寻找到这些共轭方向, 自然是求解正定二次优化问题的关键。在我们正式给出共轭梯度法之前, 这里仍然以二维的情形进行思考。如图12.13所示, 我们希望理清共轭方向与负梯度方向之间可能存在的关系。可以看出第二次的共轭方向 \mathbf{p}_1 不再是负梯度方向 $-\mathbf{g}_1$, 而是负梯度方向 $-\mathbf{g}_1$ 与前一次共轭方向 \mathbf{p}_0 的线性组合。

这样, 我们可以假定:

$$\mathbf{p}_1 = -\mathbf{g}_1 + \beta_0 \mathbf{p}_0.$$

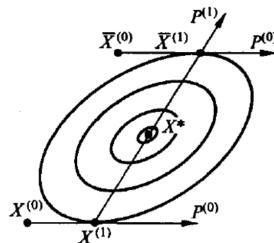


图 12.12

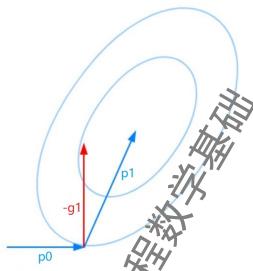


图 12.13

其中初始共轭方向 p_0 取为负梯度方向 $-g_0$ 。由 p_0 和 p_1 共轭，便可推出

$$\beta_0 = \frac{\mathbf{g}_1^T \mathbf{A} \mathbf{p}_0}{\mathbf{p}_0^T \mathbf{A} \mathbf{p}_0} = \frac{\mathbf{g}_1^T (\mathbf{g}_1 - \mathbf{g}_0)}{\mathbf{p}_0^T (\mathbf{g}_1 - \mathbf{g}_0)} = \frac{\mathbf{g}_1^T \mathbf{g}_1}{\mathbf{g}_0^T \mathbf{g}_0}.$$

其中第三个等式是根据精确线搜索的性质 $\mathbf{g}_1^T \mathbf{p}_0 = 0$ 得到的。我们暂且猜想这对 n 维空间的正定二次函数也是成立的。如果猜想成立，则很容易总结出共轭梯度法，如算法 12.4 所示。

接下来需要证明我们的猜想，即由上述共轭梯度法得到的搜索方向 $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \dots, \mathbf{p}^{(n-1)}$ 确实为 \mathbf{A} 共轭。为证明这一点，可以考虑使用归纳的方式去证明。因此，只需要考虑前 k 次迭代成立的条件下（ $k \leq 1$ 时容易验证成立），第 $k+1$ 次迭代这一结论依然成立。下面先证明几个子定理，然后再证明搜索方向的共轭性。

定理 12.1.5. 由共轭梯度法得到的第 $k+1$ 个迭代点的梯度方向与前面的搜索方向正交，即：
 $\nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(j)} = 0, j = 0, \dots, k.$

算法 12.4 共轭梯度法

- 1: 选择初始近似 $\mathbf{x}^{(0)}$, 给出允许误差 $\varepsilon > 0$;
- 2: 计算

$$\mathbf{p}^{(0)} = -\nabla f(\mathbf{x}^{(0)})$$

- 3: 一般地, 假定已得出 $\mathbf{x}^{(k)}$ 和 $\mathbf{p}^{(k)}$, 则利用精确线搜索可计算其第 $k+1$ 次近似 $\mathbf{x}^{(k+1)}$
- 4: 若 $\|\nabla f(\mathbf{x}^{(k+1)})\|^2 \leq \varepsilon$, 停止计算, $\mathbf{x}^{(k+1)}$ 即为要求的近似解。否则, 若 $k < n-1$, 则

$$\mathbf{p}^{(k+1)} = -\nabla f(\mathbf{x}^{(k+1)}) + \beta_k \mathbf{p}^{(k)} \quad (12.25)$$

$$\beta_k = \frac{\nabla f(\mathbf{x}^{(k+1)})^T \nabla f(\mathbf{x}^{(k+1)})}{\nabla f(\mathbf{x}^{(k)})^T \nabla f(\mathbf{x}^{(k)})} \quad (12.26)$$

计算 β_k 和 $\mathbf{p}^{(k+1)}$, 并转向第 3 步。

证明. 当 $j = k$ 时, 由精确线搜索性质, 显然成立。现在考虑 $j < k$ 时,

$$\begin{aligned} \nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(j)} &= \nabla f(\mathbf{x}^{(j+1)})^T \mathbf{p}^{(j)} + \sum_{i=j+1}^k (\nabla f(\mathbf{x}^{(i+1)})^T \mathbf{p}^{(j)} - \nabla f(\mathbf{x}^{(i)})^T \mathbf{p}^{(j)}) \\ &= \sum_{i=j+1}^k (\nabla f(\mathbf{x}^{(i+1)})^T \nabla f(\mathbf{x}^{(i)})^T) \mathbf{p}^{(j)} \quad (\text{精确线搜索}) \\ &= \sum_{i=j+1}^k (A(\mathbf{x}^{(i+1)} - \mathbf{x}^{(i)}))^T \mathbf{p}^{(j)} \\ &= \sum_{i=j+1}^k \lambda_i (\mathbf{p}^{(i)})^T A \mathbf{p}^{(j)} = 0 \quad (\text{相互共轭}) \end{aligned} \quad (12.27)$$

□

定理 12.1.6. 由共轭梯度法得到的第 $k+1$ 个迭代点的梯度方向与前面的梯度方向正交, 即:

$$\nabla f(\mathbf{x}^{(k+1)})^T \nabla f(\mathbf{x}^{(j)}) = 0, \quad j = 0, \dots, k.$$

证明. 根据共轭梯度法中搜索方向的迭代公式:

$$\begin{aligned} \nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(j)} &= \nabla f(\mathbf{x}^{(k+1)})^T (-\nabla f(\mathbf{x}^{(j)}) + \beta_{j-1} \mathbf{p}^{(j-1)}) \\ &= -\nabla f(\mathbf{x}^{(k+1)})^T \nabla f(\mathbf{x}^{(j)}) + \beta_{j-1} \nabla f(\mathbf{x}^{(k+1)})^T \mathbf{p}^{(j-1)} \\ &= -\nabla f(\mathbf{x}^{(k+1)})^T \nabla f(\mathbf{x}^{(j)}) \quad (\text{上一定理结论}) \end{aligned} \quad (12.28)$$

根据上一定理的结论可知, 整个式子为 0。故由最后一个等式知结论成立。□

定理 12.1.7. 共轭梯度法产生的搜索方向 $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \dots, \mathbf{p}^{(n-1)}$ 为 A 共轭。

证明. 仍然考虑第 $k+1$ 次迭代的情形 (为简化描述, 下用 \mathbf{g}_k 表示 $\nabla f(\mathbf{x}^{(k)})$):

先证当 $j = k$ 时, $(\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{p}^{(k+1)} = 0$. 因为 $\mathbf{p}^{(k+1)} = -\mathbf{g}_{k+1} + \beta_k \mathbf{p}^{(k)}$, 所以

$$\begin{aligned} (\mathbf{p}^{(k)})^T \mathbf{A} \mathbf{p}^{(k+1)} &= -(\mathbf{p}^{(k)})^T \mathbf{A} \mathbf{g}_{k+1} + \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} (\mathbf{p}^{(k)})^T \mathbf{A} \mathbf{p}^{(k)} \\ &= -\frac{(\mathbf{g}_{k+1} - \mathbf{g}_k)^T \mathbf{g}_{k+1}}{\lambda_k} + \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} (\mathbf{p}^{(k)})^T \mathbf{A} \mathbf{p}^{(k)} \\ &= -\frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\lambda_k} + \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} (\mathbf{p}^{(k)})^T \mathbf{A} \mathbf{p}^{(k)} \end{aligned} \quad (12.29)$$

根据精确先搜索知 $\lambda_k = -\frac{(\mathbf{p}^{(k)})^T \mathbf{g}_k}{(\mathbf{p}^{(k)})^T \mathbf{A}(\mathbf{p}^{(k)})} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{(\mathbf{p}^{(k)})^T \mathbf{A}(\mathbf{p}^{(k)})}$. 因此, 上式为 0.

现考虑 $j < k$ 时, $(\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{p}^{(k+1)} = 0$. 同理,

$$\begin{aligned} (\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{p}^{(k+1)} &= -(\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{g}_{k+1} + \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} (\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{p}^{(k)} \\ &= -\frac{(\mathbf{g}_{j+1} - \mathbf{g}_j)^T \mathbf{g}_{k+1}}{\lambda_j} + \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} (\mathbf{p}^{(j)})^T \mathbf{A} \mathbf{p}^{(k)} \end{aligned} \quad (12.30)$$

根据定理 12.1.6 知等式右边第一部分为 0, 又由于 $\mathbf{p}^{(j)}$, $j = 0, \dots, k$ 为 \mathbf{A} 共轭, 所以第二部分为 0. 故得证. \square

式(12.26)最先由弗莱彻 (Fletcher) 和瑞夫斯 (Reeves) 提出, 故此法亦称为 FR 共轭梯度法. 实际上, 共轭梯度法还有很多其他等价形式. 应当指出, 从理论上说, 对于正定二次函数的情形, 进行 n 次迭代即可达到极小点. 但是, 在实际计算中, 由于数据的四舍五入以及计算误差的积累, 往往做不到这一点. 此外, 由于 n 维问题的共轭方向最多只有 n 个, 在 n 步以后继续如上进行是没有意义的. 因此, 在实际应用时, 如迭代到 n 步还不收敛, 就将 $\mathbf{x}^{(n)}$ 作为新的初始近似, 重新开始迭代. 根据实际经验, 采用这种再开始的办法, 一般都可得到较好的效果.

例 12.1.5. 试用共轭梯度法求下述二次函数的极小点

$$f(\mathbf{x}) = \frac{3}{2}x_1^2 + \frac{1}{2}x_2^2 - x_1x_2 - 2x_1$$

解. 将 $f(\mathbf{x})$ 化成式(12.23)的形式, 得

$$\mathbf{A} = \begin{pmatrix} 3 & -1 \\ -1 & 1 \end{pmatrix}$$

现从 $\mathbf{x}^{(0)} = (-2, 4)^T$ 开始, 由于

$$\nabla f(\mathbf{x}) = [(3x_1 - x_2 - 2), (x_2 - x_1)]^T$$

故

$$\nabla f(\mathbf{x}^{(0)}) = (-12, 6)^T$$

$$\mathbf{p}^{(0)} = -\nabla f(\mathbf{x}^{(0)}) = (12, -6)^T$$

$$\lambda_0 = -\frac{\nabla f(\mathbf{x}^{(0)})^T \mathbf{p}^{(0)}}{(\mathbf{p}^{(0)})^T \mathbf{A} \mathbf{p}^{(0)}} = -\frac{(-12, 6) \begin{pmatrix} 12 \\ -6 \end{pmatrix}}{(12, -6) \begin{pmatrix} 3 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 12 \\ -6 \end{pmatrix}} = \frac{180}{612} = \frac{5}{17}$$

于是

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)} = \begin{pmatrix} -2 \\ 4 \end{pmatrix} + \frac{5}{17} \begin{pmatrix} 12 \\ -6 \end{pmatrix} = \left(\frac{26}{17}, \frac{38}{17} \right)^T$$

$$\nabla f(\mathbf{x}^{(1)}) = \left(\frac{6}{17}, \frac{12}{17} \right)^T$$

$$\beta_0 = \frac{\nabla f(\mathbf{x}^{(1)})^T \nabla f(\mathbf{x}^{(1)})}{\nabla f(\mathbf{x}^{(0)})^T \nabla f(\mathbf{x}^{(0)})} = \frac{\left(\frac{6}{17}, \frac{12}{17} \right) \begin{pmatrix} \frac{6}{17} \\ \frac{12}{17} \end{pmatrix}}{\begin{pmatrix} -12, 6 \end{pmatrix} \begin{pmatrix} 6 \\ 12 \end{pmatrix}} = \frac{1}{289}$$

$$\mathbf{p}^{(1)} = -\nabla f(\mathbf{x}^{(1)}) + \beta_0 \mathbf{p}^{(0)} = \begin{pmatrix} \frac{6}{17} \\ \frac{12}{17} \end{pmatrix} + \frac{1}{289} \begin{pmatrix} 12 \\ -6 \end{pmatrix} = \left(-\frac{90}{289}, -\frac{210}{289} \right)^T$$

搜索方向 $\mathbf{p}^{(1)}$ 如图 12.14 所示：

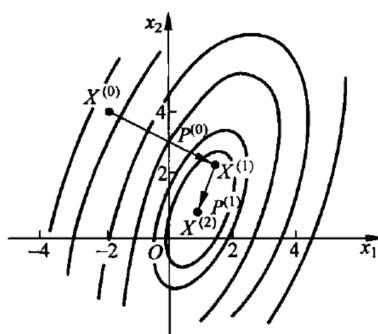


图 12.14

再计算

$$\begin{aligned}\lambda_1 &= -\frac{\nabla f(\mathbf{x}^{(1)})^T \mathbf{p}^{(1)}}{(\mathbf{p}^{(1)})^T A \mathbf{p}^{(1)}} \\ &= -\frac{\left(\frac{6}{17}, \frac{12}{17}\right) \left(-\frac{90}{289}, -\frac{210}{289}\right)^T}{\left(-\frac{90}{289}, -\frac{210}{289}\right) \begin{pmatrix} 3 & -1 \\ -1 & 1 \end{pmatrix} \left(-\frac{90}{289}, -\frac{210}{289}\right)^T} \\ &= \frac{6 \times 17 \times 90 + 12 \times 17 \times 210}{(-60, -120)(-90, -210)^T} = \frac{17(6 \times 90 + 12 \times 210)}{60 \times 90 + 120 \times 210} = \frac{17}{10}\end{aligned}$$

故

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \lambda_1 \mathbf{p}^{(1)} = \begin{pmatrix} \frac{26}{17} \\ \frac{38}{17} \end{pmatrix} + \frac{17}{10} \begin{pmatrix} -\frac{90}{289} \\ -\frac{210}{289} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

这就是 $f(\mathbf{x})$ 的极小点。

次梯度算法

在实际应用中经常会遇到不可微的函数, 对于这类函数我们无法在每个点处求出梯度, 但往往它们的最优值都是在不可微点处取到的。为了能处理这种情形, 这一节介绍次梯度算法。现在我们在问题(12.1)中假设 $f(\mathbf{x})$ 为凸函数, 但不一定可微。对凸函数可以在定义域的内点处定义次梯度 $\mathbf{g} \in \partial f(\mathbf{x})$ 。类比梯度法的构造, 我们有如下次梯度算法的迭代格式:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \lambda_k \mathbf{g}^k, \quad \mathbf{g}^k \in \partial f(\mathbf{x}^k), \quad (12.31)$$

其中 $\lambda_k > 0$ 为步长。它通常有如下四种选择:

1. 固定步长 $\lambda_k = \lambda$;
2. 固定 $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|$, 即 $\lambda_k \|\mathbf{g}^k\|$ 为常数;
3. 消失步长 $\lambda_k \rightarrow 0$ 且 $\sum_{k=0}^{\infty} \lambda_k = +\infty$;
4. 选取 λ_k 使其满足某种线搜索准则。

次梯度算法(12.31)的构造虽然是受梯度法的启发, 但在很多方面次梯度算法有其独特性质。首先, 我们知道次微分 $\partial f(\mathbf{x})$ 是一个集合, 在次梯度算法的构造中只要求从这个集合中选出一个次梯度即可, 但在实际中不同的次梯度取法可能会产生截然不同的效果; 其次, 对于梯度法, 判断一阶最优性条件只需要验证 $\|\nabla f(\mathbf{x}^*)\|$ 是否充分小即可, 但对于次梯度算法, 此时有 $0 \in \partial f(\mathbf{x}^*)$, 而这个条件在实际应用中往往是不易直接验证的, 这导致我们不能使用它作为次梯度算法的停机条件; 此外, 步长选取在次梯度法中的影响非常大, 因此, 此梯度算法的收敛性分析, 相比于梯度法较为复杂一些。为便于读者抓住主要内容, 这里不再对次梯度算法的收敛性进行展开叙述。

应用实例：梯度法求解 LASSO 回归

本小节介绍用梯度法来求解 LASSO 回归。LASSO 回归问题可以看作是对压缩感知基追踪的一个近似解法，这将在后续的制约函数法理解这一点。LASSO 问题的形式为

$$\min_{\mathbf{x}} f(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2 + \mu \|\mathbf{x}\|_1$$

LASSO 问题的目标函数 $f(\mathbf{x})$ 不光滑，在某些点处无法求出梯度，因此不能直接对原始问题使用梯度法求解。

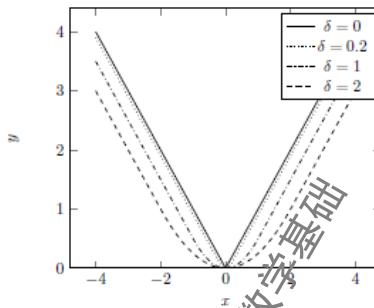


图 12.15: 绝对值函数的光滑近似

考虑到目标函数的不光滑项为 $\|\mathbf{x}\|_1$ ，它实际上是 \mathbf{x} 各个分量绝对值的和。因此，在实际应用中，人们常考虑利用如下一维光滑函数近似：

$$l_\delta(x) = \begin{cases} \frac{1}{2\delta}x^2, & |x| < \delta, \\ |x| - \frac{\delta}{2}, & \text{其他.} \end{cases}$$

实际上它是 Huber 损失函数的一种变形。图 12.15 展示了当 δ 取不同值时 $l_\delta(x)$ 对绝对值函数的逼近程度。易知，当 $\delta \rightarrow 0$ 时，光滑函数 $l_\delta(x)$ 和绝对值函数 $|x|$ 会越来越接近。

这样便可构造光滑化的 LASSO 问题为

$$\min_{\mathbf{x}} f_\delta(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2 + \mu L_\delta(\mathbf{x}),$$

其中 δ 为给定的光滑化参数，以及

$$L_\delta(\mathbf{x}) = \sum_{i=1}^n l_\delta(x_i).$$

这时容易计算出 $f_\delta(\mathbf{x})$ 的梯度为

$$\nabla f_\delta(\mathbf{x}) = \mathbf{A}^T(\mathbf{Ax} - \mathbf{b}) + \mu \nabla L_\delta(\mathbf{x}),$$

其中 $\nabla L_\delta(\mathbf{x})$ 是逐个分量定义的：

$$(\nabla L_\delta(\mathbf{x}))_i = \begin{cases} \text{sign}(x_i), & |x_i| > \delta, \\ \frac{x_i}{\delta}, & |x_i| \leq \delta. \end{cases}$$

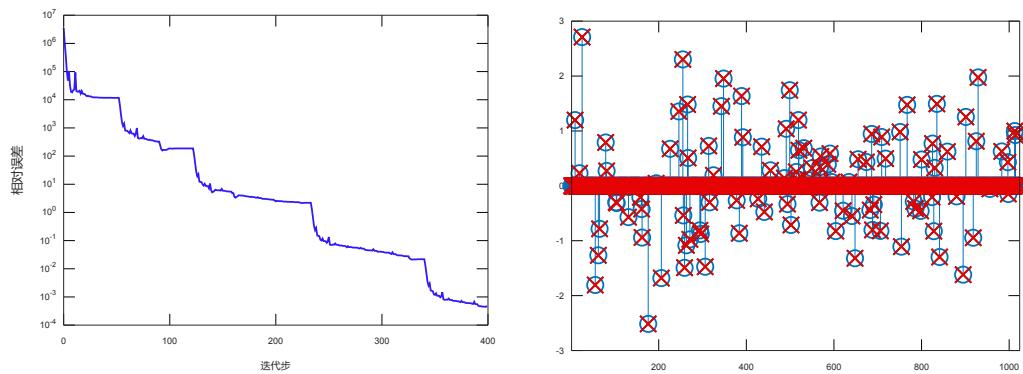


图 12.16: 光滑化 LASSO 问题求解结果

现在我们谈论步长的问题, 显然 $f_\delta(\mathbf{x})$ 的梯度是 L -利普希茨连续的, 且相应常数为 $L = \|\mathbf{A}^\top \mathbf{A}\|_2 + \frac{\mu}{\delta}$. 根据定理12.1.2, 若采用固定步长则需满足 $0 < \lambda \leq \frac{1}{L}$ 才能保证算法收敛。如果 δ 过小, 那么我们需要选取充分小的步长 λ 使得梯度法收敛。

图 12.16展示了光滑化 LASSO 问题的求解结果。其中真解 \mathbf{x}^* 是一个稀疏度 (非零元个数与总元素个数的比值) 为 0.1 的 1024 维的向量, 通过 512×1024 维的随机矩阵 \mathbf{A} 对其进行“观测”, 得到向量 \mathbf{b} 。然后利用光滑化的 LASSO 问题作为求解模型, 利用梯度法进行求解。这里的正则化参数我们设置为 $\mu = 10^{-3}$ 。只要 $|f_\delta(\mathbf{x}^k) - f_\delta(\mathbf{x}^{k-1})| < 10^{-8}$, 或者 $\|\nabla f_\delta(\mathbf{x})\| < 10^{-6}$ 或者最大迭代步数达到 3000, 则算法停止。另外, 为了加快算法的收敛速度, 可以采用连续化策略来从较大的正则化参数 μ_0 逐渐减小到 μ 。我们将在制约函数法内容中给出连续化策略合理性的解释。

应用实例：次梯度法求解正定矩阵补全问题

正定矩阵补全问题是一种特殊的矩阵恢复问题, 它的具体形式为

$$\begin{aligned} & \text{find } \mathbf{X} \in \mathcal{S}^n, \\ & \text{s.t. } X_{ij} = M_{ij}, \quad (i, j) \in \Omega, \\ & \quad \mathbf{X} \succeq 0. \end{aligned}$$

其中 Ω 是已经观测的分量位置集合。问题本质上是一个目标函数为常数的半定规划问题, 但由于其特殊性我们可以使用次梯度算法求解。考虑两个集合

$$\begin{aligned} C_1 &= \{\mathbf{X} \mid X_{ij} = M_{ij}, (i, j) \in \Omega\}, \\ C_2 &= \{\mathbf{X} \mid \mathbf{X} \succeq 0\}, \end{aligned}$$

因此, 求解正定矩阵补全问题等价于寻找闭凸集 C_1 和 C_2 的交集。定义欧几里得距离函数

$$d_j(\mathbf{X}) = \inf_{\mathbf{Y} \in C_j} \|\mathbf{X} - \mathbf{Y}\|_F,$$

则可将这个问题转化为无约束非光滑优化问题

$$\min f(\mathbf{X}) = \max \{d_1(\mathbf{X}), d_2(\mathbf{X})\}$$

由次梯度计算规则可知

$$\partial f(\mathbf{X}) = \begin{cases} \partial d_1(\mathbf{X}), & d_1(\mathbf{X}) > d_2(\mathbf{X}) \\ \partial d_2(\mathbf{X}), & d_1(\mathbf{X}) < d_2(\mathbf{X}) \\ \text{conv}(\partial d_1(\mathbf{X}) \cup \partial d_2(\mathbf{X})), & d_1(\mathbf{X}) = d_2(\mathbf{X}) \end{cases}$$

而又根据固定分量的函数极小值求次梯度的例子, 我们可以求得距离函数的一个次梯度为

$$G_j = \begin{cases} 0, & \mathbf{X} \in C_j, \\ \frac{1}{d_j(\mathbf{X})} (\mathbf{X} - \mathcal{P}_{C_j}(\mathbf{X})), & \mathbf{X} \notin C_j, \end{cases}$$

其中 $\mathcal{P}_{C_j}(\mathbf{X}) = \arg \min_{Y \in C_j} \|\mathbf{Y} - \mathbf{X}\|_F$ 为 \mathbf{X} 到 C_j 的投影. 对于集合 C_1 , \mathbf{X} 在它上面的投影为

$$(\mathcal{P}_{C_1}(\mathbf{X}))_{ij} = \begin{cases} \mathbf{M}_{ij}, & (i, j) \in \Omega \\ \mathbf{X}_{ij}, & (i, j) \notin \Omega \end{cases}$$

对于集合 C_2 , \mathbf{X} 在它上面的投影为

$$\mathcal{P}_{C_2}(\mathbf{X}) = \sum_{i=1}^n \max(0, \lambda_i) \mathbf{q}_i \mathbf{q}_i^T$$

其中 λ_i, \mathbf{q}_i 分别是 \mathbf{X} 的第 i 个特征值和特征向量. 在这里注意, 为了比较 $d_1(\mathbf{X})$ 和 $d_2(\mathbf{X})$ 的大小关系, 我们在计算次梯度时还是要将 \mathbf{X} 到两个集合的投影分别求出, 之后再选取距离较大的一个计算出次梯度. 因此, 完整的次梯度计算过程为:

1. 给定点 \mathbf{X} , 根据上式计算出 \mathbf{X} 到 C_1 和 C_2 的投影, 分别记为 \mathbf{P}_1 和 \mathbf{P}_2 ;
2. 比较 $d_j(\mathbf{X}) = \|\mathbf{X} - \mathbf{P}_j\|_F, j = 1, 2$, 较大者记为 \hat{j} ;
3. 计算次梯度 $G = \frac{\mathbf{X} - \mathbf{P}_{\hat{j}}}{d_{\hat{j}}(\mathbf{X})}$.

12.1.3 二阶方法

牛顿法

从上一讲中可以看出, 梯度法仅使用了目标函数的一阶信息. 如果函数足够光滑, 那么就可以使用更多的信息, 例如二阶信息. 直观上, 可以期望得到更好的优化算法. 这就是本讲我们将探究的牛顿类方法.

设二次函数 $f(\mathbf{x})$ 在 \mathbf{x} 具有二阶连续偏导函数, 因此, 可以在该点处进行 Taylor 展开. 不妨设 $f(\mathbf{x})$ 在 \mathbf{x} 处的二阶 Taylor 近似 (或模型) \hat{f} 为

$$\hat{f}(\mathbf{x} + \mathbf{v}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{v} + \frac{1}{2} \mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v} \quad (12.32)$$

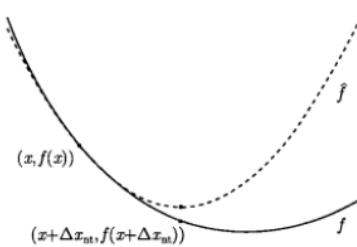


图 12.17

这是 \mathbf{v} 的二次凸函数, 当 Hessian 矩阵为半正定时, 显然在 $\mathbf{v} = \Delta \mathbf{x}_{nt}$ 处达到最小值。如下图所示:

若 Hessian 矩阵 $\nabla^2 f(\mathbf{x})$ 正定, 对二阶近似求极小, 得牛顿方程 $\nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} = -\nabla f(\mathbf{x})$, 并解得:

$$\Delta \mathbf{x}_{nt} = -\nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x}).$$

它被称之为 f 在 \mathbf{x} 处的 **Newton 步径**。由正定性可知, 除非 $\nabla f(\mathbf{x}) = \mathbf{0}$, 否则就有

$$\nabla f(\mathbf{x})^T \Delta \mathbf{x}_{nt} = -\nabla f(\mathbf{x})^T \nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x}) < 0$$

因此, Newton 步径是下降方向 (除非 \mathbf{x} 是最优点)。将 \mathbf{x} 加上 Newton 步径 $\Delta \mathbf{x}_{nt}$ 能够极小化 f 在 \mathbf{x} 处的二阶近似。特别地, 当 $f(\mathbf{x})$ 就是正定二次函数时, 只需要一步便能求出最优解。

通过简单计算, 可以得到牛顿法的单步下降量约为 $f(\mathbf{x}) - \hat{f}(\mathbf{x} + \Delta \mathbf{x}_{nt})$, 即:

$$f(\mathbf{x}) - \hat{f}(\mathbf{x} + \Delta \mathbf{x}_{nt}) \triangleq \frac{1}{2} \delta(\mathbf{x})^2,$$

其中 \hat{f} 仍是 f 在 \mathbf{x} 处的二阶近似。这一量可作为牛顿法的终止判定准则。因此, 我们可以得到牛顿法具体求解算法如下。

算法 12.5 牛顿法

1: 给定初始点 $\mathbf{x} \in \text{dom } f$, 误差阈值 $\epsilon > 0$

2: 计算 Newton 步径和减量。

$$\Delta \mathbf{x}_{nt} := -\nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x}); \quad \delta^2 := \nabla f(\mathbf{x})^T \nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x})$$

3: 停止准则。如果 $\delta^2/2 \leq \epsilon$, 退出。

4: 直线搜索。通过回溯直线搜索确定步长 λ 。

5: 改进。 $\mathbf{x} := \mathbf{x} + \lambda \Delta \mathbf{x}_{nt}$

6: 重复上述步骤, 直至退出。

经典牛顿法有很好的局部收敛性质。实际上我们有如下定理:

定理 12.1.8. (经典牛顿法的收敛性) 假设目标函数 f 是二阶连续可微的函数, 且海瑟矩阵在最优值点 \mathbf{x}^* 的一个邻域 $N_\delta(\mathbf{x}^*)$ 内是利普希茨连续的, 即存在常数 $L > 0$ 使得

$$\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in N_\delta(\mathbf{x}^*)$$

如果函数 $f(\mathbf{x})$ 在点 \mathbf{x}^* 处满足 $\nabla f(\mathbf{x}^*) = 0, \nabla^2 f(\mathbf{x}^*) \succ 0$, 则对于步长为 1 时的迭代有如下结论:

- (1) 如果初始点离 \mathbf{x}^* 足够近, 则牛顿法产生的迭代点列 $\{\mathbf{x}^k\}$ 收敛到 \mathbf{x}^* ;
- (2) $\{\mathbf{x}^k\}$ 收敛到 \mathbf{x}^* 的速度是 Q-二次的;
- (3) $\{\|\nabla f(\mathbf{x}^k)\|\}$ Q-二次收敛到 0.

证明. 从牛顿法的定义和最优值点 \mathbf{x}^* 的性质 $\nabla f(\mathbf{x}^*) = 0$ 可得

$$\begin{aligned} \mathbf{x}^{k+1} - \mathbf{x}^* &= \mathbf{x}^k - \nabla^2 f(\mathbf{x}^k)^{-1} \nabla f(\mathbf{x}^k) - \mathbf{x}^* \\ &= \nabla^2 f(\mathbf{x}^k)^{-1} [\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^*))] \end{aligned} \quad (12.33)$$

根据泰勒公式, 可得

$$\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^*) = \int_0^1 \nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) (\mathbf{x}^k - \mathbf{x}^*) dt$$

因此我们有估计

$$\begin{aligned} &\|\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^*))\| \\ &= \left\| \int_0^1 [\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)] (\mathbf{x}^k - \mathbf{x}^*) dt \right\| \\ &\leq \int_0^1 \|\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)\| \|\mathbf{x}^k - \mathbf{x}^*\| dt \\ &\leq \|\mathbf{x}^k - \mathbf{x}^*\|^2 \int_0^1 L t dt \\ &= \frac{L}{2} \|\mathbf{x}^k - \mathbf{x}^*\|^2 \end{aligned} \quad (12.34)$$

其中第二个不等式是由于海瑟矩阵的局部利普希茨连续性. 又因为 $\nabla^2 f(\mathbf{x}^*)$ 是非奇异的且 f 二阶连续可微, 因此存在 r , 使得对任意满足 $\|\mathbf{x} - \mathbf{x}^*\| \leq r$ 的点 \mathbf{x} 均有 $\|\nabla^2 f(\mathbf{x})^{-1}\| \leq 2 \|\nabla^2 f(\mathbf{x}^*)^{-1}\|$. 结合 (12.33) 式与 (12.34) 式可得:

$$\begin{aligned} &\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \\ &\leq \|\nabla^2 f(\mathbf{x}^k)^{-1}\| \|\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^*) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^*))\| \\ &\leq L \|\nabla^2 f(\mathbf{x}^*)^{-1}\| \|\mathbf{x}^k - \mathbf{x}^*\|^2 \end{aligned}$$

因此, 当初始点 \mathbf{x}^0 满足

$$\|\mathbf{x}^0 - \mathbf{x}^*\| \leq \min \left\{ \delta, r, \frac{1}{2L \|\nabla^2 f(\mathbf{x}^*)^{-1}\|} \right\} \stackrel{\text{def}}{=} \hat{\delta}$$

时, 可保证迭代点列一直处于邻域 $N_{\hat{\delta}}(\mathbf{x}^*)$ 中, 因此 $\{\mathbf{x}^k\}$ Q-二次收敛到 \mathbf{x}^* . 由牛顿方程可知

$$\begin{aligned}\|\nabla f(\mathbf{x}^{k+1})\| &= \|\nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k) - \nabla^2 f(\mathbf{x}^k) \Delta \mathbf{x}_{nt}^k\| \\ &= \left\| \int_0^1 \nabla^2 f(\mathbf{x}^k + td^k) d^k dt - \nabla^2 f(\mathbf{x}^k) \Delta \mathbf{x}_{nt}^k \right\| \\ &\leq \int_0^1 \|\nabla^2 f(\mathbf{x}^k + td^k) - \nabla^2 f(\mathbf{x}^k)\| \|\Delta \mathbf{x}_{nt}^k\| dt \\ &\leq \frac{L}{2} \|\Delta \mathbf{x}_{nt}^k\|^2 \leq \frac{1}{2} L \left\| \nabla^2 f(\mathbf{x}^k)^{-1} \right\|^2 \|\nabla f(\mathbf{x}^k)\|^2 \\ &\leq 2L \left\| \nabla^2 f(\mathbf{x}^*)^{-1} \right\|^2 \|\nabla f(\mathbf{x}^k)\|^2.\end{aligned}$$

这证明了梯度的范数 Q-二次收敛到 0. \square

牛顿法的收敛速度快, 但也存在着缺陷。函数的 Hessian 矩阵本身计算代价大, 难以存储。Hessian 矩阵还可能面临着不正定的问题, 应该如何修正? 在高维问题中, 求解 Hessian 矩阵的逆 (或者是解大规模线性方程组) 的计算量更大。能否以较小的代价找到 Hessian 矩阵的一个较好的近似? 这就是接下来要介绍的修正牛顿法和拟牛顿法 (变尺度法)。

修正牛顿法

尽管前面讨论了牛顿法, 在实际应用中此格式几乎是不能使用的. 前面也已提及经典牛顿法有如下缺陷:

- 每一步迭代需要求解一个 n 维线性方程组, 这导致在高维问题中计算量很大. 海瑟矩阵 $\nabla^2 f(\mathbf{x})$ 既不容易计算又不容易储存.
- 当 $\nabla^2 f(\mathbf{x})$ 不正定时, 由牛顿方程给出的牛顿步径 $\Delta \mathbf{x}_{nt}$ 的性质通常比较差. 例如可以验证当海瑟矩阵正定时, $\Delta \mathbf{x}_{nt}$ 是一个下降方向, 而在其他情况下 $\Delta \mathbf{x}_{nt}$ 不一定为下降方向.
- 当迭代点距最优值较远时, 直接选取步长 $\alpha = 1$ 会使得迭代极其不稳定, 在有些情况下迭代点列会发散.

为了克服这些缺陷, 我们必须对经典牛顿法做出某种修正或变形, 使其成为真正可以使用的算法. 这里介绍带线搜索的修正牛顿法, 其基本思想是对牛顿方程中的海瑟矩阵 $\nabla^2 f(\mathbf{x})$ 进行修正, 使其变成正定矩阵; 同时引入线搜索以改善算法稳定性. 它的一般框架见算法 12.6. 该算法的关键在于修正矩阵 \mathbf{E}^k 如何选取.

一个最直接的取法是取 $\mathbf{E}^k = \tau_k \mathbf{I}$, 即取 \mathbf{E}^k 为单位矩阵的常数倍. 根据矩阵理论可以知道, 当 τ_k 充分大时, 总可以保证 \mathbf{B}^k 是正定矩阵. 然而 τ_k 不宜取得过大, 这是因为当 τ_k 趋于无穷时, \mathbf{d}^k 的方向会接近负梯度方向. 比较合适的取法是先估计 $\nabla^2 f(\mathbf{x}^k)$ 的最小特征值, 再适当选择 τ_k .

另一种 \mathbf{E}^k 的选取是隐式的, 它是通过修正 Cholesky 分解的方式来求解牛顿方程. 我们知道当海瑟矩阵正定时, 方程组可以用 Cholesky 分解快速求解. 当海瑟矩阵不定或条件数较大时,

算法 12.6 修正牛顿法

- 1: 给定初始点 \mathbf{x}^0 .
- 2: **for** $k = 0, 1, 2, \dots$ **do**
- 3: 确定矩阵 \mathbf{E}^k 使得矩阵 $\mathbf{B}^k \stackrel{\text{def}}{=} \nabla^2 f(\mathbf{x}^k) + \mathbf{E}^k$ 正定且条件数较小.
- 4: 求解修正的牛顿方程 $\mathbf{B}^k \mathbf{d}^k = -\nabla f(\mathbf{x}^k)$ 得方向 \mathbf{d}^k .
- 5: 使用任意一种线搜索准则确定步长 α_k .
- 6: 更新 $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{d}^k$.
- 7: **end for**

Cholesky 分解会失败. 而修正 Cholesky 分解算法对基本 Cholesky 分解算法进行修正, 且修正后的分解和原矩阵相差不大. 我们首先回顾 Cholesky 分解的定义. 对任意对称正定矩阵 $\mathbf{A} = (a_{ij})$, 它的 Cholesky 分解可写作

$$\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^T$$

其中 $\mathbf{L} = (l_{ij})$ 是对角线元素均为 1 的下三角矩阵, $\mathbf{D} = \text{Diag}(d_1, d_2, \dots, d_n)$ 是对角矩阵且对角线元素均为正. 根据 Cholesky 分解的形式, 如果 \mathbf{A} 正定且条件数较小, 矩阵 \mathbf{D} 的对角线元素不应该太小. 如果计算过程中发现 d_j 过小就应该及时修正. 同时我们需要保证该修正有界, 因此对修正后的矩阵元素也需要有上界约束. 具体来说, 我们选取两个正参数 δ, β 使得

$$d_j \geq \delta, \quad l_{ij} \sqrt{d_j} \leq \beta, \quad i = j + 1, j + 2, \dots, n.$$

因此, 我们只需要修改 Cholesky 分解时 d_j 的更新即可保证上述条件成立. 具体更新方式为

$$d_j = \max \left\{ |c_{ij}|, \left(\frac{\theta_j}{\beta} \right)^2, \delta \right\}, \quad \theta_j = \max_{i > j} |c_{ij}|.$$

可以证明, 修正的 Cholesky 分解算法实际上是计算修正矩阵 $\nabla^2 f(\mathbf{x}^k) + \mathbf{E}^k$ 的 Cholesky 分解, 其中 \mathbf{E}^k 是对角矩阵且对角线元素非负. 当 $\nabla^2 f(\mathbf{x}^k)$ 正定且条件数足够小时有 $\mathbf{E}^k = 0$.

拟牛顿法

拟牛顿法 (变尺度法) 是近 40 多年来发展起来的, 它是求解无约束优化问题的一种有效方法. 由于它既避免了计算二阶导数矩阵及其求逆过程, 又比梯度法的收敛速度快, 特别是对高维问题具有显著的优越性, 因而使拟牛顿法获得了很高的声誉, 至今仍被公认为求解无约束优化问题最有效的算法之一. 下面我们就来简要地介绍拟牛顿法的基本原理及其计算过程.

(1) 割线方程

若想获得 Hessian 矩阵或它的逆的一个近似, 就应当寻找到它需要满足的条件. 以便构造近似矩阵, 并期望其也满足同样的条件. 显然, 这样的一个近似应当满足 Taylor 展开. 根据 Taylor 展开, 梯度函数 $\nabla f(\mathbf{x})$ 在点 $\mathbf{x}^{(k+1)}$ 处的近似为

$$\nabla f(\mathbf{x}) \approx \nabla f(\mathbf{x}^{(k+1)}) + \nabla^2 f(\mathbf{x}^{(k+1)})(\mathbf{x} - \mathbf{x}^{(k+1)})$$

令 $\mathbf{x} = \mathbf{x}^{(k)}$, 即有

$$\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}) = \nabla^2 f(\mathbf{x}^{(k+1)})(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) \quad (12.35)$$

或

$$\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} = \nabla^2 f(\mathbf{x}^{(k+1)})^{-1}[\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)})] \quad (12.36)$$

这两个式子并称为割线方程, 即拟牛顿条件。

为表述方便, 令

$$\begin{cases} \Delta \mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}) \\ \Delta \mathbf{x}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \end{cases} \quad (12.37)$$

则式(12.35)变为

$$\Delta \mathbf{g}^{(k)} = \nabla^2 f(\mathbf{x}^{(k+1)}) \Delta \mathbf{x}^{(k)}$$

式(12.36)变为

$$\Delta \mathbf{x}^{(k)} = \nabla^2 f(\mathbf{x}^{(k+1)})^{-1} \Delta \mathbf{g}^{(k)}$$

如果得到满足割线方程的 **Hessian** 矩阵的近似 (下面均用 \mathbf{H} 表示), 或者 **Hessian** 矩阵的逆的近似 (下面均用 $\bar{\mathbf{H}}$ 表示), 则可以得到拟牛顿方法的一般求解算法框架 12.7。

算法 12.7 拟牛顿法计算框架

- 1: 给定 $\mathbf{x}^0 \in \mathbb{R}^n$, 初始矩阵 $\mathbf{H}^0 \in \mathbb{R}^{n \times n}$ (或 $\bar{\mathbf{H}}^0$), 令 $k = 0$.
- 2: **while** $k = 0, 1, 2, \dots$ **do**
- 3: 计算方向 $\mathbf{d}^k = -(\mathbf{H}^k)^{-1} \nabla f(\mathbf{x}^k)$ 或 $\mathbf{d}^k = -\bar{\mathbf{H}}^k \nabla f(\mathbf{x}^k)$.
- 4: 通过线搜索找到合适的步长 $\lambda_k > 0$, 令 $\mathbf{x}^{k+1} = \mathbf{x}^k + \lambda_k \mathbf{d}^k$.
- 5: 更新海瑟矩阵的近似矩阵 \mathbf{H}^{k+1} 或其逆矩阵的近似 $\bar{\mathbf{H}}^{k+1}$.
- 6: $k \leftarrow k + 1$.
- 7: **end while**

下面, 我们将讨论如何借助割线方程 (拟牛顿条件), 具体的构造 Hessian 矩阵或其逆的近似。为了能较为清晰地阐述拟牛顿法, 这里补充几点说明。基于(12.35)式得到 Hessian 矩阵的近似, 具有较好的理论性质, 迭代序列较为稳定, 但仍然可能在大规模问题上是非常耗时的。基于(12.36)式得到 Hessian 矩阵的逆的近似, 更加实用。由于上述两种方式之间具有很好的形式对称性, 下面仅基于 Hessian 矩阵逆的近似进行探究。

(2) 秩一更新

现在考虑 Hessian 矩阵逆的近似构造。设 $\bar{\mathbf{H}}^{(k)}$ 是第 k 步 Hessian 矩阵的逆的近似, 现需构造出 $\bar{\mathbf{H}}^{(k+1)}$, 则直观的想法是对 $\bar{\mathbf{H}}^{(k)}$ 做尽可能少的改动便得到 $\bar{\mathbf{H}}^{(k+1)}$, 即对它进行秩一修正。考虑到对称性, 则可设

$$\bar{\mathbf{H}}^{(k+1)} = \bar{\mathbf{H}}^{(k)} + a \mathbf{u} \mathbf{u}^T \quad (12.38)$$

其中 $\mathbf{u} \in \mathbb{R}^n$, $a \in \mathbb{R}$ 待定。

显然, 我们需要构造出的 Hessian 逆矩阵依然满足相应的拟牛顿条件, 即割线方程。因此, 利用割线方程, 有

$$\begin{aligned}\Delta \mathbf{x}^{(k)} &= \bar{\mathbf{H}}^{(k+1)} \Delta \mathbf{g}^{(k)} \\ &= (\bar{\mathbf{H}}^{(k)} + a \mathbf{u} \mathbf{u}^T) \Delta \mathbf{g}^{(k)}\end{aligned}\quad (12.39)$$

整理得

$$a(\mathbf{u}^T \Delta \mathbf{g}^{(k)}) \mathbf{u} = \Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)}.$$

因此 \mathbf{u} 与 $\Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)}$ 共线。不妨令 $\mathbf{u} = \Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)}$, 则代入可得

$$a = \frac{1}{(\Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)})^T \Delta \mathbf{g}^{(k)}}.$$

从而, 得到秩一更新公式:

$$\bar{\mathbf{H}}^{(k+1)} = \bar{\mathbf{H}}^{(k)} + \frac{(\Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)}) (\Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)})^T}{(\Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)})^T \Delta \mathbf{g}^{(k)}}.$$

上述矩阵也称之为尺度矩阵。

如果考虑的是 Hessian 矩阵本身的近似, 则按照上述过程, 同理可得对应的秩一更新公式如下:

$$\mathbf{H}^{(k+1)} = \mathbf{H}^{(k)} + \frac{(\Delta \mathbf{g}^{(k)} - \mathbf{H}^{(k)} \Delta \mathbf{x}^{(k)}) (\Delta \mathbf{g}^{(k)} - \mathbf{H}^{(k)} \Delta \mathbf{x}^{(k)})^T}{(\Delta \mathbf{g}^{(k)} - \mathbf{H}^{(k)} \Delta \mathbf{x}^{(k)})^T \Delta \mathbf{x}^{(k)}}.$$

通过对比发现, 实际上二者之间具有非常好的形式对称性, 可看做是做了如下形式上的替换:

$$\bar{\mathbf{H}} \rightarrow \mathbf{H}, \quad \Delta \mathbf{x}^{(k)} \leftrightarrow \Delta \mathbf{g}^{(k)}.$$

这样的形式对称性对我们了解拟牛顿法大有裨益。

秩一更新虽然结构简单, 易计算, 但是秩一更新存在着重大缺陷。它不能保证在迭代过程中保持正定。因此, 需要寻求更好的近似。

(3) DFP 和 BFGS

为克服秩一修正的缺陷, 直观的改进方式是对它进行秩二修正。同样地考虑对称性, 则可设

$$\bar{\mathbf{H}}^{(k+1)} = \bar{\mathbf{H}}^{(k)} + a \mathbf{u} \mathbf{u}^T + b \mathbf{v} \mathbf{v}^T \quad (12.40)$$

其中 $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, $a, b \in \mathbb{R}$ 待定。

同上所述, 仍然利用割线方程, 有

$$\begin{aligned}\Delta \mathbf{x}^{(k)} &= \bar{\mathbf{H}}^{(k+1)} \Delta \mathbf{g}^{(k)} \\ &= (\bar{\mathbf{H}}^{(k)} + a \mathbf{u} \mathbf{u}^T + b \mathbf{v} \mathbf{v}^T) \Delta \mathbf{g}^{(k)}\end{aligned}\quad (12.41)$$

整理得

$$a(\mathbf{u}^T \Delta \mathbf{g}^{(k)}) \mathbf{u} + b(\mathbf{v}^T \Delta \mathbf{g}^{(k)}) \mathbf{v} = \Delta \mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta \mathbf{g}^{(k)}.$$

因此, \mathbf{u}, \mathbf{v} 的线性组合等于 $\Delta\mathbf{x}^{(k)} - \bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)}$ 。同样地, 不妨令 $\mathbf{u} = \Delta\mathbf{x}^{(k)}$, $\mathbf{v} = \bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)}$, 则代入可得

$$a = \frac{1}{(\Delta\mathbf{x}^{(k)})^T \Delta\mathbf{g}^{(k)}},$$

$$b = -\frac{1}{(\Delta\mathbf{g}^{(k)})^T \bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)}}.$$

从而, 得到更新公式:

$$\bar{\mathbf{H}}^{(k+1)} = \bar{\mathbf{H}}^{(k)} + \frac{\Delta\mathbf{x}^{(k)} \Delta\mathbf{x}^{(k)T}}{(\Delta\mathbf{x}^{(k)})^T \Delta\mathbf{g}^{(k)}} - \frac{\bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)} (\bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)})^T}{(\Delta\mathbf{g}^{(k)})^T \bar{\mathbf{H}}^{(k)} \Delta\mathbf{g}^{(k)}}. \quad (12.42)$$

这种迭代公式由 Davidon 发现, 并由 Fletcher 以及 Powell 进一步发展。因此被称为 **DFP 公式**。

利用前面提及的形式对称性, 可得如下更新公式:

$$\mathbf{H}^{(k+1)} = \mathbf{H}^{(k)} + \frac{\Delta\mathbf{g}^{(k)} \Delta\mathbf{g}^{(k)T}}{(\Delta\mathbf{g}^{(k)})^T \Delta\mathbf{x}^{(k)}} - \frac{\mathbf{H}^{(k)} \Delta\mathbf{x}^{(k)} (\mathbf{H}^{(k)} \Delta\mathbf{x}^{(k)})^T}{(\Delta\mathbf{x}^{(k)})^T \mathbf{H}^{(k)} \Delta\mathbf{x}^{(k)}}.$$

这种迭代格式就是著名的 **BFGS 公式**。尽管 DFP 格式和 BFGS 格式存在这种对偶关系, 但实际上, BFGS 格式效果更好些。因此, 在实际中 BFGS 格式被使用的更多。

综上, 可将拟牛顿法 (以 DFP 为例) 的计算方法总结在算法 12.8。与共轭梯度法相类似, 如果迭代 n 次仍不收敛, 则以 $\mathbf{x}^{(n)}$ 为新的 $\mathbf{x}^{(0)}$, 以这时的 $\mathbf{x}^{(0)}$ 为起点重新开始一轮新的迭代。

对于拟牛顿法, 我们也可以得到其基本的收敛性以及收敛速度。

定理 12.1.9. (BFGS 全局收敛性) 假设初始矩阵 \mathbf{H}^0 是对称正定矩阵, 目标函数 $f(\mathbf{x})$ 是二阶连续可微函数, 且下水平集

$$\mathcal{L} = \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \leq f(\mathbf{x}^0)\}$$

是凸的, 并且存在正数 m 以及 M 使得对于任意的 $\mathbf{z} \in \mathbb{R}^n$ 以及任意的 $\mathbf{x} \in \mathcal{L}$ 有

$$m\|\mathbf{z}\|^2 \leq \mathbf{z}^T \nabla^2 f(\mathbf{x}) \mathbf{z} \leq M\|\mathbf{z}\|^2,$$

则采用 BFGS 格式并结合 Wolfe 线搜索的拟牛顿算法全局收敛到 $f(\mathbf{x})$ 的极小值点 \mathbf{x}^* 。

该定理叙述了 BFGS 格式的全局收敛性, 但没有说明以什么速度收敛。下面这个定理介绍了在一定条件下 BFGS 格式会达到 Q -超线性收敛速度。这里仍然只给出定理结果, 感兴趣的读者可以查阅相关文献, 了解详细的证明过程。

定理 12.1.10. (BFGS 收敛速度) 设 $f(\mathbf{x})$ 二阶连续可微, 在最优点 \mathbf{x}^* 的一个邻域内海瑟矩阵利普希茨连续, 且使用 BFGS 迭代格式收敛到 f 的最优值点 \mathbf{x}^* 。若迭代点列 $\{\mathbf{x}^k\}$ 满足

$$\sum_{k=1}^{\infty} \|\mathbf{x}^k - \mathbf{x}^*\| < +\infty,$$

则 $\{\mathbf{x}^k\}$ 以 Q -超线性收敛到 \mathbf{x}^* 。

算法 12.8 拟牛顿法 (DFP)

1: 给定初始点 $\mathbf{x}^{(0)}$ 及梯度允许误差 $\varepsilon > 0$;

2: 若

$$\|\nabla f(\mathbf{x}^{(0)})\|^2 \neq \varepsilon$$

则 $\mathbf{x}^{(0)}$ 即为近似极小点, 停止迭代。否则, 转向下一步;

3: 令

$$\bar{\mathbf{H}}^{(0)} = \mathbf{I} \text{(单位阵)}$$

$$\mathbf{p}^{(0)} = -\bar{\mathbf{H}}^{(0)} \nabla f(\mathbf{x}^{(0)})$$

在 $\mathbf{p}^{(0)}$ 方向进行一维搜索, 确定最佳步长 λ_0

$$\min_{\lambda} f(\mathbf{x}^{(0)} + \lambda \mathbf{p}^{(0)}) = f(\mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)})$$

如此可得下一个近似点

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)}$$

4: 一般地, 设已得到近似点 $\mathbf{x}^{(k)}$, 算出 $\nabla f(\mathbf{x}^{(k)})$ 若

$$\|\nabla f(\mathbf{x}^{(k)})\|^2 \leq \varepsilon$$

则 $\mathbf{x}^{(k)}$ 即为所求的近似解, 停止迭代; 否则, 按式(12.42)计算 $\bar{\mathbf{H}}^{(k)}$, 并令

$$\mathbf{p}^{(k)} = -\bar{\mathbf{H}}^{(k)} \nabla f(\mathbf{x}^{(k)})$$

在 $\mathbf{p}^{(k)}$ 方向进行一维搜索, 确定最佳步长 λ_k

$$\min_{\lambda} f(\mathbf{x}^{(k)} + \lambda \mathbf{p}^{(k)}) = f(\mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)})$$

其下一个近似点为

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{p}^{(k)}$$

5: 若 $\mathbf{x}^{(k+1)}$ 点满足精度要求, 则 $\mathbf{x}^{(k+1)}$ 即为所求的近似解。否则, 转回第 (4) 步, 直到求出某点满足精度要求为止。

例 12.1.6. 试用 DFP 法重新计算下述二次函数的极小点

$$f(\mathbf{x}) = \frac{3}{2}x_1^2 + \frac{1}{2}x_2^2 - x_1x_2 - 2x_1$$

解. 和例 12.1.5 一样, 仍从 $\mathbf{x}^{(0)} = (-2, 4)^T$ 开始, 并取

$$\bar{\mathbf{H}}^{(0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\nabla f(\mathbf{x}) = [(3x_1 - x_2 - 2), (x_2 - x_1)]^T$$

$$\nabla f(\mathbf{x}^{(0)}) = (-12, 6)^T$$

$$\mathbf{p}^{(0)} = -\bar{\mathbf{H}}^{(0)} \nabla f(\mathbf{x}^{(0)}) = -\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -12 \\ 6 \end{pmatrix} = \begin{pmatrix} 12 \\ -6 \end{pmatrix}$$

利用一维搜索, 即 $\min_{\lambda} f(\mathbf{x}^{(0)} + \lambda \mathbf{p}^{(0)})$, 可算得

$$\lambda_0 = \frac{5}{17}$$

从而得到下一迭代点

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)} = \begin{pmatrix} -2 \\ 4 \end{pmatrix} + \frac{5}{17} \begin{pmatrix} 12 \\ -6 \end{pmatrix} = \left(\frac{26}{17}, \frac{38}{17} \right)^T$$

$$\nabla f(\mathbf{x}^{(1)}) = \left(\frac{6}{17}, \frac{12}{17} \right)^T$$

$$\Delta \mathbf{x}^{(0)} = \mathbf{x}^{(1)} - \mathbf{x}^{(0)} = \left(\frac{26}{17}, \frac{38}{17} \right)^T - \left(-2, 4 \right)^T = \left(\frac{60}{17}, -\frac{30}{17} \right)^T$$

$$\begin{aligned} \Delta \mathbf{g}^{(0)} &= \nabla f(\mathbf{x}^{(1)}) - \nabla f(\mathbf{x}^{(0)}) \\ &= \left(\frac{6}{17}, \frac{12}{17} \right)^T - (-12, 6)^T = \left(\frac{210}{17}, -\frac{90}{17} \right)^T \end{aligned}$$

这时更新 Hessian 逆矩阵为

$$\begin{aligned} \bar{\mathbf{H}}^{(1)} &= \bar{\mathbf{H}}^{(0)} + \frac{\Delta \mathbf{x}^{(0)} (\Delta \mathbf{x}^{(0)})^T}{(\Delta \mathbf{g}^{(0)})^T \Delta \mathbf{x}^{(0)}} - \frac{\bar{\mathbf{H}}^{(0)} \Delta \mathbf{g}^{(0)} (\Delta \mathbf{g}^{(0)})^T \bar{\mathbf{H}}^{(0)}}{\Delta \mathbf{g}^{(0)} (\Delta \mathbf{g}^{(0)})^T \bar{\mathbf{H}}^{(0)} \Delta \mathbf{g}^{(0)}} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{\left(\frac{60}{17}, -\frac{30}{17} \right)^T \left(\frac{60}{17}, -\frac{30}{17} \right)}{\left(\frac{210}{17}, -\frac{90}{17} \right) \left(\frac{60}{17}, -\frac{30}{17} \right)^T} - \frac{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \left(\frac{210}{17}, -\frac{90}{17} \right)^T \left(\frac{210}{17}, -\frac{90}{17} \right) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}{\left(\frac{210}{17}, -\frac{90}{17} \right) \left(\frac{1}{17}, \frac{1}{17} \right) \left(\frac{210}{17}, -\frac{90}{17} \right)^T} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{1}{17} \begin{pmatrix} 4 & -2 \\ -2 & 1 \end{pmatrix} - \frac{1}{58} \begin{pmatrix} 49 & -21 \\ -21 & 9 \end{pmatrix} = \frac{1}{986} \begin{pmatrix} 385 & 241 \\ 241 & 891 \end{pmatrix} \\ \mathbf{p}^{(1)} &= -\bar{\mathbf{H}}^{(1)} \nabla f(\mathbf{x}^{(1)}) = -\frac{1}{986} \begin{pmatrix} 385 & 241 \\ 241 & 891 \end{pmatrix} \begin{pmatrix} \frac{6}{17} \\ \frac{12}{17} \end{pmatrix} = -\begin{pmatrix} \frac{9}{29} \\ \frac{21}{29} \end{pmatrix} \end{aligned}$$

再由一维搜索 $\min_{\lambda} f(\mathbf{x}^{(1)} + \lambda \mathbf{p}^{(1)})$, 得

$$\lambda_1 = \frac{29}{17}$$

从而

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \lambda_1 \mathbf{p}^{(1)} = \begin{pmatrix} \frac{26}{17} \\ \frac{38}{17} \end{pmatrix} + \frac{29}{17} \begin{pmatrix} -\frac{9}{29} \\ -\frac{21}{29} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\nabla f(\mathbf{x}^{(2)}) = (0, 0)^T$$

可知 $\mathbf{x}^{(2)} = (1, 1)^T$ 为极小点。

在以上讨论中, 我们取第一个尺度矩阵 $\bar{\mathbf{H}}^{(0)}$ 为对称正定阵, 以后的尺度矩阵由式(12.42)逐步形成。可以证明, 这样构成的尺度矩阵均为对称正定阵。由此可知其搜索方向 $\mathbf{p}^{(k)} = -\bar{\mathbf{H}}^{(k)}\nabla f(\mathbf{x}^{(k)})$ 为下降方向, 这就可以保证每次迭代均能使目标函数值有所改善。

当把 DFP 拟牛顿法用于正定二次函数时, 产生的搜索方向为共轭方向, 因而也具有有限步收敛的性质。若将初始尺度矩阵也取为单位矩阵, 对这种函数来说, DFP 法就与共轭梯度法一样了。

还要指出, 可以采用不同的方法来构造尺度矩阵 $\bar{\mathbf{H}}^{(k)}$, 从而就有不同的拟牛顿法。DFP 法属于拟牛顿法的一种。开始时取 $\bar{\mathbf{H}}^{(0)} = \mathbf{I}$, 这相当于第一步采用最速下降法。以后的 $\bar{\mathbf{H}}^{(k)}$ 接近于 $\mathbf{H}(\mathbf{x}^{(k)})^{-1}$, 当达到极小点时, 从理论上讲, 这时的尺度矩阵应等于该点处 Hessian 矩阵的逆阵。

例 12.1.7. 试用 DFP 法求

$$\min f(\mathbf{x}) = 4(x_1 - 5)^2 + (x_2 - 6)^2$$

解. 取

$$\bar{\mathbf{H}}^{(0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{x}^{(0)} = \begin{pmatrix} 8 \\ 9 \end{pmatrix}$$

由于

$$\nabla f(\mathbf{x}) = [8(x_1 - 5), 2(x_2 - 6)]^T$$

$$\nabla f(\mathbf{x}^{(0)}) = (24, 6)^T$$

故

$$\begin{aligned} \mathbf{x}^{(1)} &= \mathbf{x}^{(0)} + \lambda_0 \mathbf{p}^{(0)} = \mathbf{x}^{(0)} + \lambda_0 [-\bar{\mathbf{H}}^{(0)} \nabla f(\mathbf{x}^{(0)})] \\ &= \begin{pmatrix} 8 \\ 9 \end{pmatrix} - \lambda_0 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 24 \\ 6 \end{pmatrix} = \begin{pmatrix} 8 \\ 9 \end{pmatrix} - \lambda_0 \begin{pmatrix} 24 \\ 6 \end{pmatrix} \\ &= \begin{pmatrix} 8 - 24\lambda_0 \\ 9 - 6\lambda_0 \end{pmatrix} \end{aligned}$$

$$f(\mathbf{x}^{(1)}) = 4[(8 - 24\lambda_0) - 5]^2 + [(9 - 6\lambda_0) - 6]^2$$

令

$$\frac{df(\mathbf{x}^{(1)})}{d\lambda_0} = 0$$

可得

$$\lambda_0 = \frac{17}{130}$$

这便得到下一个迭代点

$$\begin{aligned}\mathbf{x}^{(1)} &= [(8 - 24\lambda_0), (9 - 6\lambda_0)]^T = (4.862, 8.215)^T \\ \nabla f(\mathbf{x}^{(1)}) &= (-1.108, 4.431)^T \\ \Delta \mathbf{x}^{(0)} &= \mathbf{x}^{(1)} - \mathbf{x}^{(0)} = (-3.138, -0.785)^T \\ \Delta \mathbf{g}^{(0)} &= \nabla f(\mathbf{x}^{(1)}) - \nabla f(\mathbf{x}^{(0)}) = (-25.108, -1.569)^T\end{aligned}$$

由此可得

$$\begin{aligned}\bar{\mathbf{H}}^{(1)} &= \bar{\mathbf{H}}^{(0)} + \frac{\Delta \mathbf{x}^{(0)}(\Delta \mathbf{x}^{(0)})^T}{(\Delta \mathbf{g}^{(0)})^T \Delta \mathbf{x}^{(0)}} - \frac{\bar{\mathbf{H}}^{(0)} \Delta \mathbf{g}^{(0)} (\Delta \mathbf{g}^{(0)})^T \bar{\mathbf{H}}^{(0)}}{(\Delta \mathbf{g}^{(0)})^T \bar{\mathbf{H}}^{(0)} \Delta \mathbf{g}^{(0)}} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{(-3.138, -0.785)^T (-3.138, -0.785)}{(-25.108, -1.569)(-3.138, -0.785)^T} - \\ &\quad \frac{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} (-25.108, -1.569)^T (-25.108, -1.569) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}{(-25.108, -1.569) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} (-25.108, -1.569)^T} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0.1231 & 0.0308 \\ 0.0308 & 0.0077 \end{pmatrix} - \begin{pmatrix} 0.9961 & 0.0622 \\ 0.0622 & 0.0039 \end{pmatrix} = \begin{pmatrix} 0.1270 & -0.0315 \\ -0.0315 & 1.0038 \end{pmatrix}\end{aligned}$$

故

$$\begin{aligned}\mathbf{x}^{(2)} &= \mathbf{x}^{(1)} - \lambda_1 \bar{\mathbf{H}}^{(1)} \nabla f(\mathbf{x}^{(1)}) \\ &= \begin{pmatrix} 4.862 \\ 8.215 \end{pmatrix} - \lambda_1 \begin{pmatrix} 0.1270 & -0.0315 \\ -0.0315 & 1.0038 \end{pmatrix} \begin{pmatrix} -1.108 \\ 4.431 \end{pmatrix}\end{aligned}$$

如上求最佳步长, 可得

$$\lambda_1 = 0.4942$$

代入上式得

$$\mathbf{x}^{(2)} = (5, 6)^T$$

这就是极小点。

实际上, 对上述例题做进一步的计算, 可以发现尺度矩阵 $\bar{\mathbf{H}}$ 对 Hessian 矩阵的逆有着较好的近似。若计算该例题的目标函数 $f(\mathbf{x})$ 的 Hessian 矩阵, 则

$$\mathbf{A} = \begin{pmatrix} 8 & 0 \\ 0 & 2 \end{pmatrix}$$

故

$$\mathbf{A}^{-1} = \begin{pmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$$

现计算出该问题的 $\bar{\mathbf{H}}^{(2)}$, 有

$$\bar{\mathbf{H}}^{(2)} = \begin{pmatrix} 1.25 \times 10^{-1} & -8.882 \times 10^{-16} \\ -8.882 \times 10^{-16} & 5.00 \times 10^{-1} \end{pmatrix}$$

可知二者几乎相等。

在以上几小节中, 我们介绍了求解无约束优化问题的解析法, 这些方法只是众多算法中的一部分。一般认为, 从迭代次数上考虑, 拟牛顿法所需迭代次数较少, 共轭梯度法次之, 最速下降法所需迭代次数最多。但从每次迭代所需的计算工作量来看, 却正好相反, 最速下降法最简单, 拟牛顿法比它们都繁琐。

应用实例: 牛顿法求解 Logistic 回归

在前面我们已经介绍了二分类的 Logistic 回归模型:

$$\min_{\mathbf{x}} L(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^m \ln (1 + \exp(-b_i \mathbf{a}_i^T \mathbf{x})) + \lambda \|\mathbf{x}\|_2^2.$$

接下来推导求解该问题的牛顿法, 这转化为计算目标函数 $L(\mathbf{x})$ 的梯度和 Hessian 矩阵的问题。

根据第六章介绍的向量值函数求导法, 容易算出梯度为

$$\begin{aligned} \nabla L(\mathbf{x}) &= \frac{1}{m} \sum_{i=1}^m \frac{1}{1 + \exp(-b_i \mathbf{a}_i^T \mathbf{x})} \cdot \exp(-b_i \mathbf{a}_i^T \mathbf{x}) \cdot (-b_i \mathbf{a}_i) + 2\lambda \mathbf{x} \\ &= -\frac{1}{m} \sum_{i=1}^m (1 - p_i(\mathbf{x})) b_i \mathbf{a}_i + 2\lambda \mathbf{x} \end{aligned}$$

其中 $p_i(\mathbf{x}) = \frac{1}{1 + \exp(-b_i \mathbf{a}_i^T \mathbf{x})}$ 。引入矩阵 $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m]^T \in \mathbb{R}^{m \times n}$, 向量 $\mathbf{b} = (b_1, b_2, \dots, b_m)^T$, 以及

$$\mathbf{p}(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \dots, p_m(\mathbf{x}))^T.$$

此时梯度可简写为:

$$\nabla L(\mathbf{x}) = -\frac{1}{m} \mathbf{A}^T (\mathbf{b} - \mathbf{b} \odot \mathbf{p}(\mathbf{x})) + 2\lambda \mathbf{x}.$$

再对梯度求导, 并写成更为紧凑的矩阵形式, 可得到 Hessian 矩阵

$$\nabla^2 L(\mathbf{x}) = \frac{1}{m} \mathbf{A}^T \mathbf{P}(\mathbf{x}) \mathbf{A} + 2\lambda \mathbf{I}$$

其中 $\mathbf{P}(\mathbf{x})$ 为由 $\{p_i(\mathbf{x}) (1 - p_i(\mathbf{x}))\}_{i=1}^m$ 生成的对角矩阵。因此, 牛顿法可以写作

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \left(\frac{1}{m} (\mathbf{A}^T \mathbf{P}(\mathbf{x}^k) \mathbf{A} + 2\lambda \mathbf{I}) \right)^{-1} \left(\frac{1}{m} \mathbf{A}^T (\mathbf{b} - \mathbf{b} \odot \mathbf{p}(\mathbf{x}^k)) - 2\lambda \mathbf{x}^k \right).$$

在实际中, λ 经常取为 $\frac{1}{100m}$ 。另外, 当变量规模不是很大时, 可以利用正定矩阵的 Cholesky 分解来求解牛顿方程; 当变量规模较大时, 可以使用共轭梯度法进行不精确求解。这里采用 LIBSVM 网站的数据集, 包括: a9a、ijcnn1 和 CINA 数据集。然后使用牛顿法进行求解, 其求解结果参见图 12.18。从中可以看出, 在精确解附近梯度范数具有 Q-超线性收敛性。

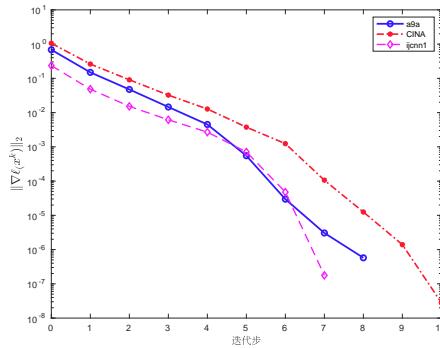


图 12.18: 牛顿法求解 Logistic 回归模型

应用实例：拟牛顿法求解压缩感知问题

考虑压缩感知问题：

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{x}\|_1, \quad \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}, \quad (12.43)$$

其中 $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$ 为给定的矩阵和向量. 这是一个约束优化问题, 如何将其转化为一个无约束优化问题呢? 自然地, 我们可以考虑其对偶问题. 由于问题 (12.43) 的对偶问题的无约束优化形式不是可微的, 即无法计算梯度 (读者可以自行验证), 我们考虑如下正则化问题:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{x}\|_1 + \frac{\alpha}{2} \|\mathbf{x}\|_2^2, \quad \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b}, \quad (12.44)$$

这里 $\alpha > 0$ 为正则化参数. 显然, 当 α 趋于无穷大时, 问题 (12.44) 的解会逼近 (12.43) 的解. 由于问题 (12.44) 的目标函数是强凸的, 其对偶问题的无约束优化形式的目标函数是可微的. 具体地, 问题 (12.44) 的对偶问题为

$$\min_{\mathbf{y} \in \mathbb{R}^m} f(\mathbf{y}) = -\mathbf{b}^T \mathbf{y} + \frac{\alpha}{2} \|\mathbf{A}^T \mathbf{y} - \mathcal{P}_{[-1,1]^n}(\mathbf{A}^T \mathbf{y})\|_2^2, \quad (12.45)$$

其中 $\mathcal{P}_{[-1,1]^n}(\mathbf{x})$ 为 \mathbf{x} 到集合 $[-1, 1]^n$ 的投影. 通过简单计算, 可知

$$\nabla f(\mathbf{y}) = -\mathbf{b} + \alpha \mathbf{A} (\mathbf{A}^T \mathbf{y} - \mathcal{P}_{[-1,1]^n}(\mathbf{A}^T \mathbf{y}))$$

那么, 我们可以利用 L-BFGS 方法来求解问题 (12.45). 在得到该问题的解 \mathbf{y}^* 之后, 问题 (12.44) 的解 \mathbf{x}^* 可通过下式近似得到:

$$\mathbf{x}^* \approx \alpha (\mathbf{A}^T \mathbf{y}^* - \mathcal{P}_{[-1,1]^n}(\mathbf{A}^T \mathbf{y}^*))$$

进一步地, 当 α 充分大时, 问题 (12.44) 的解就是原问题 (12.43) 的解. 因此, 我们可以通过选取合适的 α , 通过求解问题 (12.45) 来得到问题 (12.43) 的解. 我们用第 LASSO 问题中的 \mathbf{A} 和 \mathbf{b} , 分别选取 $\alpha = 5, 10$, 调用 BFGS (在实际中, 我们通常使用的时有限内存 BFGS 算法, 感兴趣的读者可以查阅阅读材料中的相关文献) 方法求解问题 (12.45), 其中内存长度取为 5. 迭代收敛过程见图 12.19. 从图中我们可以看到, 当靠近最优解时, BFGS 方法的迭代点列呈 Q-线性收敛.

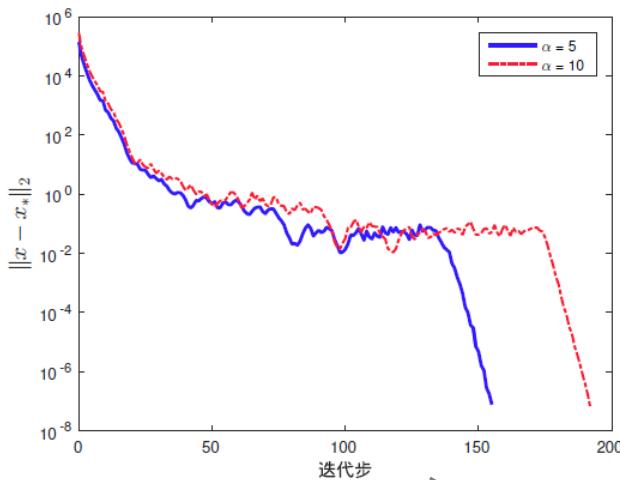


图 12.19: 压缩感知问题

12.2 约束优化

实际工作中遇到的大多数优化问题，其变量的取值多受到一定限制，这种限制由约束条件来体现。带有约束条件的优化问题称为约束优化问题，其一般形式为

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, m \\ & h_j(\mathbf{x}) = 0, \quad j = 1, 2, \dots, p \end{aligned} \tag{12.46}$$

另外，一些特定方法可能会针对于求解仅含有不等式约束优化问题：

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, m \end{aligned} \tag{12.47}$$

或者是仅含有等式约束优化问题：

$$\begin{aligned} & \min f_0(\mathbf{x}) \\ \text{s.t. } & h_j(\mathbf{x}) = 0, \quad j = 1, 2, \dots, p \end{aligned} \tag{12.48}$$

求解约束优化问题要比求解无约束优化问题困难得多。对有约束的极小化问题来说，除了要使目标函数在每次迭代有所下降之外，还要时刻注意解的可行性问题（某些算法的中间步骤除外），这就给寻优工作带来了很大困难。为了实际求解和（或）简化其优化工作，通常可采用以下方法：将无约束优化问题的求解方法直接推广至约束优化问题；将约束问题化为逐序列的无约束问题；以及将复杂问题变换为较简单问题的其它方法。本讲主要介绍：可行方向法、外点法（二次罚函数法、增广拉格朗日法）和内点法（倒数障碍函数法、对数障碍函数法）。

12.2.1 可行方向法

现在介绍求解不等式约束优化问题(12.47)的可行方向法。下面对约束优化问题相关的概念进行界定。

定义 12.2.1. 考虑约束优化的某一可行点 $\mathbf{x}^{(0)}$, 对该点的任一方向 \mathbf{d} 来说, 若存在实数 $\lambda'_0 > 0$, 使对任意 $\lambda \in [0, \lambda'_0]$ 均有

$$f_0(\mathbf{x}^{(0)} + \lambda \mathbf{d}) < f_0(\mathbf{x}^{(0)})$$

就称方向 \mathbf{d} 为 $\mathbf{x}^{(0)}$ 点的一个下降方向。

由于约束优化还涉及到一个关键的问题是解的可行性问题。如果一个下降方向却不是可行的, 那么对优化目标函数依然没有价值。这里我们定义可行方向为:

定义 12.2.2. 假定 $\mathbf{x}^{(0)}$ 是待求解的约束优化问题的一个可行点, 现考虑此点的某一方向 \mathbf{d} , 若存在实数 $\lambda_0 > 0$, 使对于任意 $\lambda \in [0, \lambda_0]$ 均有

$$\mathbf{x}^{(0)} + \lambda \mathbf{d} \text{ 满足约束条件} \quad (12.49)$$

满足约束条件, 则称方向 \mathbf{d} 是 $\mathbf{x}^{(0)}$ 点的一个可行方向。

如果方向 \mathbf{d} 既是 $\mathbf{x}^{(0)}$ 点的可行方向, 又是这个点的下降方向, 就称它是该点的可行下降方向。对于含有不等式约束优化的一个可行解 $\mathbf{x}^{(0)}$, 在不等式约束条件 $f_i(\mathbf{x}) \leq 0$ 下可行时存在两种可能。其一为 $f_i(\mathbf{x}^{(0)}) < 0$ 。这时点 $\mathbf{x}^{(0)}$ 不是处于由这一约束条件形成的可行域边界上, 因而这一约束对 $\mathbf{x}^{(0)}$ 点的微小摄动不起限制作用。从而称这个约束条件是 $\mathbf{x}^{(0)}$ 点的不起作用约束(或无效约束)。其二为 $f_i(\mathbf{x}^{(0)}) = 0$ 。这时 $\mathbf{x}^{(0)}$ 点处于该约束条件形成的可行域边界上, 它对 $\mathbf{x}^{(0)}$ 的摄动起到了某种限制作用。故称这个约束是 $\mathbf{x}^{(0)}$ 点的起作用约束(有效约束)。显而易见, 对于含有等式约束优化的一个可行解 $\mathbf{x}^{(0)}$, 等式约束条件对所有可行点来说都是起作用约束。

直接判定某可行点处的下降方向或可行方向是不易的, 通常需要对函数进行光滑性假设。若假定 $f_0(\mathbf{x})$ 和 $f_i(\mathbf{x})$ 具有一阶连续偏导数, 可以对可行下降方向的理解进一步深化, 以便设计有效的求解方法。在无约束优化中, 我们知道将目标函数 $f_0(\mathbf{x})$ 在点 $\mathbf{x}^{(0)}$ 处作一阶泰勒展开, 可得满足条件

$$\nabla f_0(\mathbf{x}^{(0)})^T \mathbf{d} < 0 \quad (12.50)$$

的方向 \mathbf{d} 必为 $\mathbf{x}^{(0)}$ 点的下降方向。

在光滑性假设下, 可行方向应该满足什么条件? 若 \mathbf{d} 是可行点 $\mathbf{x}^{(0)}$ 处的任一可行方向(图12.20), 则对该点的所有有效约束均有

$$-\nabla f_i(\mathbf{x}^{(0)})^T \mathbf{d} \geq 0, \quad i \in J \quad (12.51)$$

其中 J 为 $\mathbf{x}^{(0)}$ 这个点所有起作用约束下标的集合。这是因为它需要保持有效约束函数不会上升, 从而保证下一个迭代点仍然可行, 即小于或等于 0。因此, 同时满足式(12.50)和(12.51)的方向一

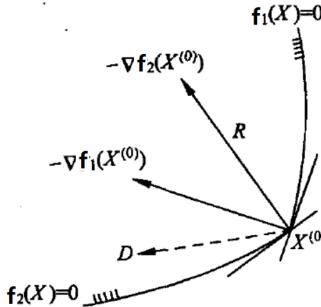


图 12.20

定是可行下降方向。如果 $\mathbf{x}^{(0)}$ 点不是极小点, 继续寻优时的搜索方向就应从该点的可行下降方向中去找。显然, 若某点存在可行下降方向, 它就不会是极小点。另外, 若某点为极小点, 则在该点不存在可行下降方向, 即如下定理所述。

定理 12.2.1. 设 \mathbf{x}^* 是不等式约束优化问题(12.4)的一个局部极小点, 目标函数 $f_0(\mathbf{x})$ 在 \mathbf{x}^* 处可微, 而且

$f_i(\mathbf{x})$ 在 \mathbf{x}^* 处可微, 当 $i \in J$

$f_i(\mathbf{x})$ 在 \mathbf{x}^* 处连续, 当 $i \notin J$

则在 \mathbf{x}^* 点不存在可行下降方向, 从而不存在向量 \mathbf{d} 同时满足:

$$\begin{cases} \nabla f_0(\mathbf{x}^*)^T \mathbf{d} < 0 \\ -\nabla f_i(\mathbf{x}^*)^T \mathbf{d} > 0, \quad i \in J \end{cases} \quad (12.52)$$

证明. 事实上, 通过反证法即可证明。若存在满足式(12.52)的方向 \mathbf{d} , 则通过泰勒展开可以发现, 在该点的一个小的邻域内, 沿该方向搜索一定可找到更好的可行点。从而, 与 \mathbf{x}^* 为局部极小点的假设矛盾。□

式(12.52)的几何意义: 满足该条件的方向 \mathbf{d} , 与点 \mathbf{x}^* 处目标函数负梯度方向和有效约束函数负梯度方向的夹角均为锐角。式(12.52)就是下降方向(12.50)和可行方向(12.51)的交集。

因此, 为了设计出求解方法, 需要解决第 k 个迭代可行点 $\mathbf{x}^{(k)}$ (非局部极小点) 处搜索方向的问题。显然, 此处的搜索方向应当从式(12.52)可行下降方向中寻找, 即从下述不等式组中确定向量 \mathbf{d} :

$$\begin{cases} \nabla f_0(\mathbf{x}^{(k)})^T \mathbf{d} < 0 \\ -\nabla f_i(\mathbf{x}^{(k)})^T \mathbf{d} > 0, \quad i \in J \end{cases} \quad (12.53)$$

显然, 在 $\mathbf{x}^{(k)}$ 处满足可行下降的方向可能有多个, 这时就需要思考如下问题: 我们应该选择哪一个方向? 借鉴无约束优化中的梯度下降法, 确实能带给我们一些启发。在梯度下降法中, 我

们期望选择的搜索方向使得目标函数值最快地下降，因此得到了搜索方向为负梯度方向。同理，在此同样希望目标函数值尽快地下降，但是这里无法给出一个解析方向。需要转换为如下优化问题：

$$\begin{aligned} \min \quad & \eta \\ \text{s.t.} \quad & \nabla f_0(\mathbf{x}^{(k)})^T \mathbf{d} \leq \eta \\ & \nabla f_i(\mathbf{x}^{(k)})^T \mathbf{d} \leq \eta, \quad i \in J(\mathbf{x}^{(k)}) \\ & \eta \leq 0 \end{aligned} \quad (12.54)$$

幸运地，这是一个易于求解的线性规划问题。上述优化问题还存在一个不足，就是对搜索方向 \mathbf{d} 做缩放仍然是满足约束，从而导致无限的最优解。实际上，我们只关注的是这个方向（分量的相对大小）。所以还需加一些限制：

$$\begin{aligned} \min \quad & \eta \\ \text{s.t.} \quad & \nabla f_0(\mathbf{x}^{(k)})^T \mathbf{d} \leq \eta \\ & \nabla f_i(\mathbf{x}^{(k)})^T \mathbf{d} \leq \eta, \quad i \in J(\mathbf{x}^{(k)}) \\ & -1 \leq d_i \leq 1, \quad i = 1, 2, \dots, n \\ & \eta \leq 0 \end{aligned} \quad (12.55)$$

其中 $d_i (i = 1, 2, \dots, n)$ 为向量 \mathbf{d} 的分量。将线性规划式(12.55)的最优解记为 $(\mathbf{d}^{(k)}, \eta_k)$ ，如果求出的 $\eta_k = 0$ ，说明在 $\mathbf{x}^{(k)}$ 点不存在可行下降方向，在 $\nabla f_i(\mathbf{x}^{(k)})$ (此处 $i \in J(\mathbf{x}^{(k)})$) 线性无关的条件下， $\mathbf{x}^{(k)}$ 满足 KKT 条件。若解出的 $\eta_k < 0$ ，则得到可行下降方向 $\mathbf{d}^{(k)}$ ，这就是我们所要的搜索方向。

现考虑约束优化式(12.47)，设 $\mathbf{x}^{(k)}$ 是它的一个可行解，但不是要求的极小点。为了求它的极小点或近似极小点，根据式(12.55)确定在 $\mathbf{x}^{(k)}$ 点的某一可行下降方向 $\mathbf{d}^{(k)}$ 。然后，根据线搜索确定步长 λ_k 。从而得到下一迭代点 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{d}^{(k)}$ 。若满足精度要求，迭代停止， $\mathbf{x}^{(k+1)}$ 就是所要的点。否则，从 $\mathbf{x}^{(k+1)}$ 出发继续进行迭代，直到满足要求为止。上述这种方法称为可行方向法，我们这里一般指的是 Zoutendijk 在 1960 年提出的算法及其变形。下面就将其具体迭代步骤总结为算法 12.9。由于机器学习中较少使用可行方向法，因此，这里仅介绍了不等式约束情形下的可行方向法。对于更为一般的情形（包括等式约束），限于篇幅，这里不做拓展。实际上，通过简单地思考过程，可以很容易地将该方法推广至线性等式约束的情形。

例 12.2.1. 用可行方向法解下述约束优化问题

$$\begin{aligned} \max \quad & 4x_1 + 4x_2 - x_1^2 - x_2^2 \\ \text{s.t.} \quad & x_1 + 2x_2 \leq 4 \end{aligned}$$

解. 先将该约束优化问题写成

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) = -(4x_1 + 4x_2 - x_1^2 - x_2^2) \\ \text{s.t.} \quad & f_1(\mathbf{x}) = x_1 + 2x_2 - 4 \leq 0 \end{aligned}$$

算法 12.9 可行方向法

- 1: 确定允许误差 $\varepsilon_1 > 0$ 和 $\varepsilon_2 > 0$, 选初始近似点 $\mathbf{x}^{(0)}$ 满足约束条件, 并令 $k := 0$;
 - 2: 确定起作用约束指标集
- $$J(\mathbf{x}^{(k)}) = \{i \mid f_i(\mathbf{x}^{(k)}) = 0, 1 \leq i \leq m\}$$
- (1) 若 $J(\mathbf{x}^{(k)}) = \emptyset$ (\emptyset 为空集), 而且 $\|\nabla f_0(\mathbf{x}^{(k)})\| \leq \varepsilon_1$, 停止迭代, 得点 $\mathbf{x}^{(k)}$;
 - (2) 若 $J(\mathbf{x}^{(k)}) = \emptyset$, 但 $\|\nabla f_0(\mathbf{x}^{(k)})\| > \varepsilon_1$, 则取 $\mathbf{d}^{(k)} = -\nabla f_0(\mathbf{x}^{(k)})$, 然后转向第 5 步;
 - (3) 若 $J(\mathbf{x}^{(k)}) \neq \emptyset$, 转下一步;
- 3: 求解线性规划

$$\left\{ \begin{array}{l} \min \eta \\ \nabla f_0(\mathbf{x}^{(k)})^T \mathbf{d} \leq \eta \\ \nabla f_i((\mathbf{x}^{(k)})^T \mathbf{d}) \leq \eta, \quad i \in J(\mathbf{x}^{(k)}) \\ -1 \leq d_i \leq 1, \quad i = 1, 2, \dots, n \\ \eta \leq 0 \end{array} \right.$$

设它的最优解是 $(\mathbf{d}^{(k)}, \eta_k)$;

- 4: 检验是否满足

$$|\eta_k| \leq \varepsilon_2$$

若满足则停止迭代, 得到点 $\mathbf{x}^{(k)}$; 否则, 以 $\mathbf{d}^{(k)}$ 为搜索方向, 并转下一步;

- 5: 解下述一维优化问题

$$\lambda_k \in \min_{0 \leq \lambda \leq \bar{\lambda}} f_0(\mathbf{x}^{(k)} + \lambda \mathbf{d}^{(k)})$$

此处

$$\bar{\lambda} = \max\{\lambda \mid f_i(\mathbf{x}^{(k)} + \lambda \mathbf{d}^{(k)}) \leq 0, \quad i = 1, 2, \dots, m\}$$

- 6: 令

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \mathbf{d}^{(k)}$$

$$k := k + 1$$

转回第 2 步。

取初始可行点 $\mathbf{x}^{(0)} = (0, 0)^T, f_0(\mathbf{x}^{(0)}) = 0$

$$\nabla f_0(\mathbf{x}) = \begin{pmatrix} 2x_1 - 4 \\ 2x_2 - 4 \end{pmatrix}, \quad \nabla f_0(\mathbf{x}^{(0)}) = \begin{pmatrix} -4 \\ -4 \end{pmatrix}$$

$$\nabla f_1(\mathbf{x}) = (1, 2)^T$$

$f_1(\mathbf{x}^{(0)}) = -4 < 0$, 从而 $J(\mathbf{x}^{(0)}) = \emptyset$ (空集)。由于

$$\|\nabla f_0(\mathbf{x}^{(0)})\|^2 = (-4)^2 + (-4)^2 = 32$$

所以 $\mathbf{x}^{(0)}$ 不是 (近似) 极小点。现取搜索方向

$$\mathbf{d}^{(0)} = -\nabla f_0(\mathbf{x}^{(0)}) = (4, 4)^T$$

从而

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \lambda \mathbf{d}^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \lambda \begin{pmatrix} 4 \\ 4 \end{pmatrix} = \begin{pmatrix} 4\lambda \\ 4\lambda \end{pmatrix}$$

将其代入约束条件, 并令 $f_1(\mathbf{x}^{(1)}) = 0$, 解得 $\bar{\lambda} = 1/3$ 。

$$f_0(\mathbf{x}^{(1)}) = -16\lambda - 16\lambda + 16\lambda^2 + 16\lambda^2 = 32\lambda^2 - 32\lambda$$

令 $f_0(\mathbf{x}^{(1)})$ 对 λ 的导数等于零, 解得 $\lambda = 1/2$ 。因 λ 大于 $\bar{\lambda}(\bar{\lambda} = 1/3)$, 故取 $\lambda_0 = \bar{\lambda} = 1/3$ 。

$$\mathbf{x}^{(1)} = \left(\frac{4}{3}, \frac{4}{3} \right)^T, \quad f_0(\mathbf{x}^{(1)}) = -\frac{64}{9}$$

$$\nabla f_0(\mathbf{x}^{(1)}) = \left(-\frac{4}{3}, -\frac{4}{3} \right)^T, \quad f_1(\mathbf{x}^{(1)}) = 0$$

现构成下述线性规划问题

$$\begin{aligned} & \min \eta \\ \text{s.t. } & -\frac{4}{3}d_1 - \frac{4}{3}d_2 \leq \eta \\ & d_1 + 2d_2 \leq \eta \\ & -1 \leq d_1 \leq 1, \quad -1 \leq d_2 \leq 1 \\ & \eta \leq \end{aligned}$$

从而得到, $\eta = -4/10$, 搜索方向

$$\mathbf{d}^{(1)} = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = \begin{pmatrix} 1.0 \\ -0.7 \end{pmatrix}$$

由此

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \lambda \mathbf{d}^{(1)} = \begin{pmatrix} 4/3 + \lambda \\ 4/3 - 0.7\lambda \end{pmatrix}$$

$$f_0(\mathbf{x}^{(2)}) = 1.49\lambda^2 - 0.4\lambda - 7.111$$

令 $\frac{df(\mathbf{x}^{(2)})}{d\lambda} = 0$, 得到 $\lambda = 0.134$ 。现暂用该步长, 算出

$$\mathbf{x}^{(2)} = \begin{pmatrix} 4/3 + 0.134 \\ 4/3 - 0.7 \times 0.134 \end{pmatrix} = \begin{pmatrix} 1.467 \\ 1.239 \end{pmatrix}$$

因 $f_1(\mathbf{x}^{(2)}) = -0.055 < 0$, 上面算出的 $\mathbf{x}^{(2)}$ 为可行点, 说明选取 $\lambda_1 = 0.134$ 正确。继续迭代下去, 可得最优解为

$$\mathbf{x}^* = (1.6, 1.2)^T, \quad f_0(\mathbf{x}^*) = -7.2.$$

原问题的最优解不变, 其原问题目标函数值为 $-f_0(\mathbf{x}^*) = 7.2$ 。

12.2.2 外点法

本节介绍求解约束优化问题的制约函数法。使用这种方法，可将约束优化问题的求解，转化为求解一系列无约束优化问题，因而也称这种方法为无约束极小化技术，简记为 SUMT(sequential unconstrained minimization technique)。常用的制约函数可分为两类：罚函数 (penalty function) 和障碍函数 (barrier function)。对应于这两种函数的 SUMT 可分为：外点法和内点法。这一节我们主要是了解外点法。

约束优化问题相比于无约束优化问题的难点，在于不能简单地将负梯度方向作为搜索方向。一种直观地想法，通过将约束项转化为一种惩罚项。违背约束时，进行惩罚；反之，不惩罚。将惩罚项添加到目标函数中，从而将约束优化问题转化为无约束优化问题进行求解。当惩罚的力度足够大时，约束自然需要得到满足，不然无法极小化目标函数。这就是外点法的主要思想。

下面先考虑等式约束的转化，构造一个函数 $\phi(t)$

$$\phi(t) = \begin{cases} 0, & \text{当 } t = 0 \\ \infty, & \text{当 } t \neq 0 \end{cases} \quad (12.56)$$

现把 $h_j(\mathbf{x})$ 视为 t ，显然当 \mathbf{x} 满足约束条件时， $\phi(h_j(\mathbf{x})) = 0$ ， $j = 1, 2, \dots, p$ ；当 \mathbf{x} 不满足约束条件时， $\phi(h_j(\mathbf{x})) = \infty$ 。

现在考虑不等式约束的转化，构造一个函数 $\psi(t)$

$$\psi(t) = \begin{cases} 0, & \text{当 } t \geq 0 \\ \infty, & \text{当 } t < 0 \end{cases} \quad (12.57)$$

现把 $-f_i(\mathbf{x})$ 视为 t ，显然当 \mathbf{x} 满足约束条件时， $\psi(-f_i(\mathbf{x})) = 0$ ， $i = 1, 2, \dots, m$ ；当 \mathbf{x} 不满足约束条件时， $\psi(-f_i(\mathbf{x})) = \infty$ 。

再构造罚函数

$$P(\mathbf{x}) = f_0(\mathbf{x}) + \sum_{j=1}^p \phi(h_j(\mathbf{x})) + \sum_{i=1}^m \psi(-f_i(\mathbf{x})) \quad (12.58)$$

就转换成求解无约束问题

$$\min P(\mathbf{x}) \quad (12.59)$$

若该问题有解，假定其解为 \mathbf{x}^* ，则由式(12.56)和式(12.57)知应有 $\phi(h_j(\mathbf{x}^*)) = 0$ 和 $\psi(-f_i(\mathbf{x}^*)) = 0$ 。这就是说点 \mathbf{x}^* 满足约束条件。因而， \mathbf{x}^* 不仅是问题式(12.59)的极小解，它也是原问题式(12.47)的极小解。这样一来，就把有约束问题式(12.47)的求解化成了求解无约束问题式(12.59)。

二次罚函数法

显然，用上述方法构造的函数存在不足， $\phi(t)$ 和 $\psi(t)$ 在 $t = 0$ 处不连续，更没有导数。为此，将它们修改为

$$\phi(t) = \begin{cases} 0, & \text{当 } t = 0 \\ t^2, & \text{当 } t \neq 0 \end{cases} \quad \text{以及} \quad \psi(t) = \begin{cases} 0, & \text{当 } t \geq 0 \\ t^2, & \text{当 } t < 0 \end{cases} \quad (12.60)$$

修改后的函数 $\phi(t)$, $\psi(t)$, 当 $t = 0$ 时导数等于零, 而且 $\phi(t)$, $\psi(t)$ 和 $\phi'(t)$, $\psi'(t)$ 对任意 t 都连续。当 \mathbf{x} 满足约束条件时仍有

$$\sum_{j=1}^p \phi(h_j(\mathbf{x})) = 0, \quad \sum_{i=1}^m \psi(-f_i(\mathbf{x})) = 0$$

但是, 当 \mathbf{x} 不满足约束条件时

$$0 < \sum_{j=1}^p \phi(h_j(\mathbf{x})) < \infty$$

$$0 < \sum_{i=1}^m \psi(-f_i(\mathbf{x})) < \infty$$

因此, 可能存在因为惩罚的力度不够, 导致无约束优化问题的解不满足原约束条件的情形发生。故我们需要对无约束优化问题(12.59)的目标函数 $P(\mathbf{x})$ 进行适当的改变。

定义 12.2.3. 对一般的约束优化问题(12.46), 定义如下二次罚函数:

$$P(\mathbf{x}, M) = f_0(\mathbf{x}) + M \left[\sum_{j=1}^p [h_j(\mathbf{x})]^2 + \sum_{i=1}^m [\min(0, -f_i(\mathbf{x}))]^2 \right] \quad (12.61)$$

其中等式第二项为惩罚项, $M > 0$ 为罚因子。

若求得的无约束优化问题的最优解 $\mathbf{x}(M)$ 满足约束条件, 则它必定是原问题的极小解。事实上, 对于所有满足约束条件的 \mathbf{x}

$$\begin{aligned} f_0(\mathbf{x}) + M \left[\sum_{j=1}^p \phi(h_j(\mathbf{x})) + \sum_{i=1}^m \psi(-f_i(\mathbf{x})) \right] &= P(\mathbf{x}, M) \\ &\geq P(\mathbf{x}(M), M) \\ &= f_0(\mathbf{x}(M)) \end{aligned}$$

即当 \mathbf{x} 满足约束条件时, 有 $f_0(\mathbf{x}) \geq f_0(\mathbf{x}(M))$ 。

虽然有罚因子 M , 但仍然可能出现最优解不满足约束条件。所以需要不断地增大罚因子 M 的值, 迫使最优解满足所有约束条件。这样便得到二次罚函数法的迭代过程如算法12.10。对于为何应不断地增大罚因子, 下面也给出了图解。图12.21示出了这种惩罚项的例子, 图中左半部表示约束条件 $f_1(\mathbf{x}) = a - x \leq 0$ 的情形, 右半部则表示 $f_2(x) = x - b \leq 0$ 的情形。若对于某一个罚因子 M , 例如 M_1 , $\mathbf{x}(M_1)$ 不满足约束条件, 就加大罚因子的值。随着 M 值得增加, 惩罚函数中的惩罚项所起的作用随之增大。 $\min P(\mathbf{x}, M)$ 的解 $\mathbf{x}(M)$ 与约束集的“距离”就越来越近, 当

$$0 < M_1 < M_2 < \cdots < M_k < \cdots$$

趋于无穷大时, 点列 $\{\mathbf{x}(M_k)\}$ 就从可行域的外部趋于原问题式(12.46)的极小点 \mathbf{x}_{\min} 。

算法 12.10 二次罚函数法

- 1: 给定初值 $\mathbf{x}^{(0)}$ 。取 $M_1 > 0$ (例如说取 $M_1 = 1$)，允许误差 $\varepsilon > 0$ ，并令 $k := 0$ ；
- 2: 以 $\mathbf{x}^{(k)}$ 为初始点，求无约束优化问题的最优解：

$$\mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x}} P(\mathbf{x}, M_k)$$

式中

$$P(\mathbf{x}, M_k) = f_0(\mathbf{x}) + M_k \left[\sum_{j=1}^p [h_j(\mathbf{x})]^2 + \sum_{i=1}^m [\min(0, -f_i(\mathbf{x}))]^2 \right]$$

- 3: 若对某一个 $j (1 \leq j \leq p)$ 或 $i (1 \leq i \leq m)$ 有

$$|h_j(\mathbf{x}^{(k)})| \geq \varepsilon, f_i(\mathbf{x}^{(k)}) \geq \varepsilon$$

则取 $M_{k+1} > M_k$ (例如, $M_{k+1} = cM_k$, ($c > 1$)), 令 $k := k + 1$, 并转向第 2 步。否则, 停止迭代, 得

$$x_{\min} \approx \mathbf{x}^{(k)}$$

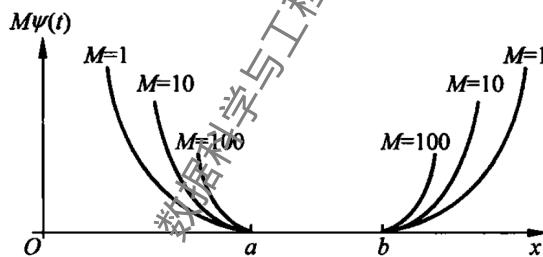


图 12.21: 罚因子对罚函数的影响

借此可对外点法作如下经济解释：把目标函数看成“价格”，约束条件看成某种“规定”，采购人可在规定范围内购置最便宜的东西。此外对违反规定制定了一种“罚款”政策，若符合规定，罚款为零；否则，要收罚款。采购人付出的总代价应是价格和罚款的总和。采购者的目标是使总代价最小，这就是上述的无约束问题。当罚款规定不够苛刻时，采购人可能存在一些投机行为，可能违反部分规定，使得总代价最小。当罚款规定得很苛刻时，违反规定支付的罚款很高，这就迫使采购人符合规定。在数学上表现为罚因子 M_k 足够大时，上述无约束问题的最优解应满足约束条件，而成为约束条件的最优解。

例 12.2.2. 求解约束优化问题

$$\begin{aligned} \min \quad & f_0(\mathbf{x}) = x_1 + x_2 \\ \text{s.t.} \quad & f_1(\mathbf{x}) = x_1^2 - x_2 \leq 0 \\ & f_2(\mathbf{x}) = -x_1 \leq 0 \end{aligned}$$

解. 构造罚函数

$$\begin{aligned} P(\mathbf{x}, M) &= x_1 + x_2 + M \{ [\min(0, -(x_1^2 - x_2))]^2 + [\min(0, x_1)]^2 \} \\ \frac{\partial P}{\partial x_1} &= 1 + 2M[\min(0, (x_1^2 - x_2)(2x_1))] + 2M[\min(0, x_1)] \\ \frac{\partial P}{\partial x_2} &= 1 + 2M[\min(0, (-x_1^2 + x_2))] \end{aligned}$$

对于不满足约束条件的点 $\mathbf{x} = (x_1, x_2)^\top$, 有

$$x_1^2 - x_2 > 0, \quad x_1 < 0$$

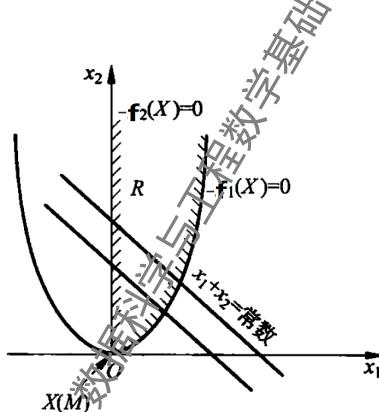


图 12.22

令

$$\frac{\partial P}{\partial x_1} = \frac{\partial P}{\partial x_2} = 0$$

得 $\min P(\mathbf{x}, M)$ 的解为

$$\mathbf{x}(M) = \left(-\frac{1}{2(1+M)}, \left(\frac{1}{4(1+M)^2} - \frac{1}{2M} \right) \right)^\top$$

取 $M = 1, 2, 3, 4$, 可得出以下结果:

$$M = 1: \quad \mathbf{x} = (-1/4, -7/16)^\top$$

$$M = 2: \quad \mathbf{x} = (-1/6, -2/9)^\top$$

$$M = 3: \quad \mathbf{x} = (-1/8, -29/192)^\top$$

$$M = 4: \quad \mathbf{x} = (-1/10, -23/200)^\top$$

可知 $\mathbf{x}(M)$ 从约束条件外面逐步逼近约束条件的边界, 当 $M \rightarrow \infty$ 时, $\mathbf{x}(M)$ 趋于原问题的极小解 $\mathbf{x}_{\min} = (0, 0)^T$ (见图 12.22)。

二次罚函数法虽然计算简便, 但是二次罚函数法也存在不足。例如, 为了保证最优解可行, 罚因子必须趋于无穷大。实际上, 罚因子趋于无穷大时, 子问题变得病态而难以求解。是否能对二次罚函数进行某种修正, 使得对于有限的罚因子得到的最优解也是可行的? 这就是接下来要介绍的增广拉格朗日函数法。

增广拉格朗日函数法

为方便理解, 这里仅介绍等式约束优化问题的增广拉格朗日函数法。该方法是对二次罚函数的一个修正。它是在拉格朗日函数的基础之上, 增加约束条件的二次罚函数。定义如下:

定义 12.2.4. 对等式约束优化问题(12.48), 定义如下增广拉格朗日函数:

$$L_M(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \sum_{j=1}^p [\lambda_j h_j(\mathbf{x})] + M \sum_{j=1}^p [h_j(\mathbf{x})]^2 \quad (12.62)$$

其中 $\boldsymbol{\lambda}$ 为拉格朗日乘子, $M > 0$ 仍为罚因子。

因此, 希望每次迭代时求解如下无约束优化问题去近似原优化问题:

$$\min_{\mathbf{x}} L_{M_k}(\mathbf{x}, \boldsymbol{\lambda}^{(k)}).$$

在得到该优化问题的具体迭代步骤前, 我们需要理清这些问题: 一方面, 增广拉格朗日函数优化问题的最优解能趋于原等式约束优化问题的最优解? 罚因子在每次迭代采用逐步放大时, 拉格朗日乘子 $\boldsymbol{\lambda}$ 在每次迭代时需要如何变化? 下面通过对比原问题与增广拉格朗日函数优化问题的最优性条件, 来寻找答案。

原等式约束优化问题的最优解 \mathbf{x}^* 与相应的拉格朗日乘子 $\boldsymbol{\lambda}^*$, 应满足

$$\nabla f_0(\mathbf{x}^*) + \sum_{j=1}^p [\lambda_j^* \nabla h_j(\mathbf{x}^*)] = 0 \quad (12.63)$$

而增广拉格朗日函数在第 k 次迭代时的最优解 $\mathbf{x}^{(k+1)}$, 应满足

$$\nabla f_0(\mathbf{x}^{(k+1)}) + \sum_{j=1}^p [\lambda_j^{(k)} + 2M_k h_j(\mathbf{x}^{(k+1)})] \nabla h_j(\mathbf{x}^{(k+1)}) = 0 \quad (12.64)$$

通过对比发现, 为使得迭代点列 $\mathbf{x}^{(k)}$ 收敛到 \mathbf{x}^* , 则需要保证上述式(12.63)与式(12.64)的一致性。因此

$$\lambda_j^* \approx \lambda_j^{(k)} + 2M_k h_j(\mathbf{x}^{(k+1)}), \quad j = 1, \dots, p. \quad (12.65)$$

从而得到乘子的下一步迭代格式

$$\lambda_j^{(k+1)} = \lambda_j^{(k)} + 2M_k h_j(\mathbf{x}^{(k+1)}), \quad j = 1, \dots, p. \quad (12.66)$$

直观分析：若 $\mathbf{x}^{(k)}$, $\boldsymbol{\lambda}^{(k)}$ 分别收敛到 \mathbf{x}^* , $\boldsymbol{\lambda}^*$, 则由式(12.65)知

$$h_j(\mathbf{x}^{(k+1)}) \approx \frac{1}{2M_k}(\lambda_j^* - \lambda_j^{(k)}).$$

因此 $h_j(\mathbf{x}^{(k)})$ 也将趋于 0。当 λ_j^* 足够接近 $\lambda_j^{(k)}$ 时，允许罚因子不需要足够大（无需趋于无穷大），也能保证约束条件满足。事实上，我们可以得到如下收敛定理：

定理 12.2.2. 假设 \mathbf{x}^* , $\boldsymbol{\lambda}^*$ 分别是原等式约束优化问题(12.48)的严格局部最小解和相应的拉格朗日乘子，那么，存在足够大的常数 $\bar{M} > 0$ 和足够小的常数 $\delta > 0$ ，如果对某个 k ，有

$$\frac{1}{M_k} \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| < \delta, \quad M_k \geq \bar{M},$$

则

$$\boldsymbol{\lambda}^{(k)} \rightarrow \boldsymbol{\lambda}^*, \quad \mathbf{x}^{(k)} \rightarrow \mathbf{x}^*.$$

该定理表明增广拉格朗日函数法的收敛性并不要求 M_k 趋于正无穷，只需要大于某个 \bar{M} 。最后，将增广拉格朗日函数法的迭代步骤总结如下：

算法 12.11 增广拉格朗日函数法

1: 给定初值 $\mathbf{x}^{(0)}$ 和乘子 $\boldsymbol{\lambda}^{(0)}$ 。取 $M_0 > 0$ ，允许精度要求 $\eta_k > 0$ 和约束条件违反误差 $\varepsilon > 0$ ，并令 $k := 0$ ；

2: 以 $\mathbf{x}^{(k)}$ 为初始点，求无约束优化问题的最优解：

$$\mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x}} L_{M_k}(\mathbf{x}, \boldsymbol{\lambda}^{(k)})$$

使得精度满足

$$\|\nabla_{\mathbf{x}} L_{M_k}(\mathbf{x}^{(k+1)}, \boldsymbol{\lambda}^{(k)})\| \leq \eta_k.$$

3: 若对所有 $j (1 \leq j \leq p)$ 有 $|h_j(\mathbf{x}^{(k)})| \leq \varepsilon$ ，则停止迭代，返回近似解 $\mathbf{x}^{(k+1)}$, $\boldsymbol{\lambda}^{(k)}$ ；否则转到下一步。

4: 更新： $\lambda_j^{(k+1)} = \lambda_j^{(k)} + 2M_k h_j(\mathbf{x}^{(k+1)})$, $M_{k+1} = cM_k$ ($c > 1$)，令 $k := k + 1$ ，并转向第 2 步。

总体上，可以看出外点法具有一定的优点，例如函数 $P(\mathbf{x}, M)$ 是在 \mathbb{R}^n 上进行优化，初始点可任意选择，这给计算带来了很大方便。而且外点法也可用于非凸函数的最优化。外点法同时适用于含有等式和不等式约束条件的优化问题。然而，如果可行域函数的性质比较复杂，甚至没有定义，这时就无法使用外点法。

应用实例：二次罚函数法求解低秩矩阵恢复

在前面的章节中，我们介绍了低秩矩阵恢复问题（又称矩阵补全问题），并引入了该问题的形式如下：

$$\begin{aligned} \min \quad & \|X\|_* \\ \text{s.t.} \quad & X_{ij} = M_{ij}, \quad (i, j) \in \Omega, \end{aligned}$$

我们对其中的等式约束引入二次罚函数可以得到

$$\min \quad \|X\|_* + \frac{\sigma}{2} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2.$$

当罚因子 $\sigma = \frac{1}{\mu}$ 时，上述优化问题转化为如下优化问题

$$\min \quad \mu \|X\|_* + \frac{1}{2} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2. \quad (12.67)$$

因此我们可以使用罚函数法的策略求解低秩矩阵恢复问题，具体见算法 12.12。

算法 12.12 低秩矩阵恢复的罚函数法

```

1: 给定初值  $X^0$ , 最终参数  $\mu$ , 初始参数  $\mu_0$ , 因子  $\gamma \in (0, 1)$ ,  $k \leftarrow 0$ .
2: while  $\mu_k \geq \mu$  do
3:   以  $X^k$  为初值,  $\mu = \mu_k$  为正则化参数求解问题 (12.67), 得  $X^{k+1}$ .
4:   if  $\mu_k = \mu$  then
5:     停止迭代, 输出  $X^{k+1}$ .
6:   else
7:     更新罚因子  $\mu_{k+1} = \max \{\mu, \gamma \mu_k\}$ .
8:      $k \leftarrow k + 1$ .
9:   end if
10:   $k \leftarrow k + 1$ 
11: end while

```

由于我们还没有学习如何处理带核范数的优化问题，这里先跳过子问题求解的叙述。实际上求解子问题 (12.67) 可使用之后讲到的近似点梯度法。

应用实例：增广拉格朗日函数法求解半定规划问题

考虑半定规划问题：

$$\begin{aligned} \min \quad & \text{Tr}(\mathbf{C}X) \\ \text{s.t.} \quad & \text{Tr}(\mathbf{A}_j X) = b_j, j = 1, \dots, p \\ & X \succeq 0 \end{aligned} \quad (12.68)$$

其中 $\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p \in S^n$, $\text{Tr}(\cdot)$ 是迹函数。其对偶问题为:

$$\begin{aligned} & \min_{\mathbf{y} \in \mathbb{R}^p} -\mathbf{b}^T \mathbf{y}, \\ & \text{s.t. } \sum_{j=1}^p y_j \mathbf{A}_j \preceq \mathbf{C}. \end{aligned} \quad (12.69)$$

我们可以利用增广拉格朗日函数法求解。对于原始问题, 引入乘子 $\boldsymbol{\lambda} \in \mathbb{R}^p$, 罚因子 σ , 并记 $\mathcal{A}(\mathbf{X}) = (\text{Tr}(\mathbf{A}_1 \mathbf{X}), \text{Tr}(\mathbf{A}_2 \mathbf{X}), \dots, \text{Tr}(\mathbf{A}_p \mathbf{X}))^T$, 则增广拉格朗日函数为

$$L_\sigma(\mathbf{X}, \boldsymbol{\lambda}) = \langle \mathbf{C}, \mathbf{X} \rangle - \boldsymbol{\lambda}^T (\mathcal{A}(\mathbf{X}) - \mathbf{b}) + \frac{\sigma}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2, \quad \mathbf{X} \succeq 0.$$

那么, 增广拉格朗日函数法为

$$\begin{cases} \mathbf{X}^{k+1} \approx \arg \min_{\mathbf{X} \in S_+^n} L_{\sigma_k}(\mathbf{X}, \boldsymbol{\lambda}^k), \\ \boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - \sigma_k (\mathcal{A}(\mathbf{X}^{k+1}) - \mathbf{b}), \\ \sigma_{k+1} = \min \{\rho \sigma_k, \bar{\sigma}\}. \end{cases}$$

这里, 当迭代收敛时, \mathbf{X}^k 和 $\boldsymbol{\lambda}^k$ 分别收敛到问题 (12.68) 和 (12.69) 的解。

12.2.3 内点法

前面已经提及, 如果 $f(\mathbf{x})$ 在可行域外的性质比较复杂, 甚至没有定义, 这时就无法使用外点法。这促使我们去思考能否设计一种类似于外点法的算法, 通过逐序列的无约束优化问题的解, 不断地逼近原始约束优化问题的解。同时要求这种方法满足, 每次迭代过程始终在可行域内部进行。人们也把取在可行域内部 (即既不在可行域外, 也不在可行域边界上) 的可行点称为内点或严格内点。

一种较为直观的想法, 仍然使用约束条件改造原目标函数, 使得它在原可行域的边界上设置一道“障碍”, 使迭代点靠近可行域的边界时, 给出的新目标函数值迅速增大, 从而使迭代点始终留在可行域内部。这种方法实际上就是接下来要介绍的内点法。这时, 改造后的目标函数通常被称为障碍函数。满足这种要求的障碍函数, 其极小解自然不会在可行域的边界上达到。这是因为虽然可行域是一个闭集, 但因极小点不在闭集的边界上, 只能在可行域内部取得, 因而实际上是具有无约束性质的优化问题, 可借助于无约束优化的方法进行计算。由于要保持在可行域内部, 因此, 内点法主要用于不等式约束问题(12.47)。

倒数障碍函数法及对数障碍函数法

根据上述分析, 需要将约束优化式(12.47)转化为下述一系列无约束性质的极小化问题:

$$\min_{\mathbf{x} \in R_0} \bar{P}(\mathbf{x}, r_k) \quad (12.70)$$

其中 $\bar{P}(\mathbf{x}, r_k)$ 是障碍函数, $R_0 = \{\mathbf{x} \mid -f_i(\mathbf{x}) > 0, \quad i = 1, 2, \dots, m\}$ 。只要能在可行域边界上建立起“障碍”, 我们便可以设计出不同的障碍函数。根据障碍函数的不同, 便可得到不同的障碍函数法。通常, 我们使用倒数障碍函数和对数障碍函数。

定义 12.2.5. 对一般的约束优化问题(12.47), 定义如下倒数障碍函数:

$$\bar{P}(\mathbf{x}, r_k) = f_0(\mathbf{x}) + r_k \sum_{i=1}^m \frac{1}{-f_i(\mathbf{x})}, \quad (r_k > 0) \quad (12.71)$$

其中等式第二项为障碍项, $r_k > 0$ 为罚因子。

可以计算出, 在接近可行域的边界上 (即至少有一个 $-f_i(\mathbf{x}) \rightarrow 0^+$), $\frac{1}{-f_i(\mathbf{x})}$ 趋于正无穷大, 则 $\bar{P}(\mathbf{x}, r_k)$ 趋于正无穷大。

定义 12.2.6. 对一般的约束优化问题(12.47), 定义如下对数障碍函数:

$$\bar{P}(\mathbf{x}, r_k) = f_0(\mathbf{x}) - r_k \sum_{i=1}^m \log(-f_i(\mathbf{x})), \quad (r_k > 0) \quad (12.72)$$

其中等式第二项为障碍项, $r_k > 0$ 为罚因子。

同样地, 此时在接近可行域的边界上 (即至少有一个 $-f_i(\mathbf{x}) \rightarrow 0^+$), $\log(-f_i(\mathbf{x}))$ 趋于负无穷大, 则 $\bar{P}(\mathbf{x}, r_k)$ 趋于正无穷大。

如果从可行域内部的某一点 $\mathbf{x}^{(0)}$ 出发, 按无约束极小化方法对式(12.70)进行迭代 (在进行一维搜索时要使用控制步长, 以免迭代点跑到 R_0 以外), 则随着障碍因子 r_k 的逐步减小, 即

$$r_1 > r_2 > \cdots > r_k > \cdots > 0$$

障碍项所起的作用也越来越小, 因而, 求出的 $\min \bar{P}(\mathbf{x}, r_k)$ 的解 $\mathbf{x}(r_k)$ 也逐步逼近原问题式(12.47)的极小解 \mathbf{x}_{\min} 。现在, 可将一般的内点法的迭代步骤总结为算法 12.13。

值得指出的是, 收敛准则也可采用不同的形式, 例如:

$$\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\| < \varepsilon$$

或

$$\|f_0(\mathbf{x}^{(k)}) - f_0(\mathbf{x}^{(k-1)})\| < \varepsilon.$$

例 12.2.3. 试用内点法求解

$$\min f_0(\mathbf{x}) = \frac{1}{3}(x_1 + 1)^3 + x_2$$

$$s.t. f_1(\mathbf{x}) = 1 - x_1 \leq 0$$

$$f_2(\mathbf{x}) = -x_2 \leq 0$$

解. 构造倒数障碍函数

$$\bar{P}(\mathbf{x}, r) = \frac{1}{3}(x_1 + 1)^3 + x_2 + \frac{r}{x_1 - 1} + \frac{r}{x_2}$$

$$\frac{\partial \bar{P}}{\partial x_1} = (x_1 + 1)^2 - \frac{r}{(x_1 - 1)^2} = 0$$

$$\frac{\partial \bar{P}}{\partial x_2} = 1 - \frac{r}{x_2^2} = 0$$

算法 12.13 内点法

- 1: 取 $r_1 > 0$ (例如取 $r_1 = 1$), 允许误差 $\varepsilon > 0$;
- 2: 找出一可行点 $\mathbf{x}^{(0)} \in R_0$, 并令 $k = 0$;
- 3: 构造障碍函数, 障碍项可采用倒数障碍函数 (式(12.71)), 也可采用对数障碍函数 (例如式(12.72));
- 4: 以 $\mathbf{x}^{(k)} \in R_0$ 为初始点, 对障碍函数进行无约束极小化 (在 R_0 内):

$$\mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x} \in R_0} \bar{P}(\mathbf{x}, r_k) \quad (12.73)$$

式中 $\bar{P}(\mathbf{x}, r_k)$ 见式(12.71)或(12.72);

- 5: 检验是否满足收敛准则

$$r_k \sum_{i=1}^m \frac{1}{-f_i(\mathbf{x}^{(k)})} \leq \varepsilon \quad (\text{倒数障碍函数})$$

或

$$\left| r_k \sum_{i=1}^m \log(-f_i(\mathbf{x}^{(k)})) \right| \leq \varepsilon \quad (\text{对数障碍函数})$$

如满足上述准则, 则以 $\mathbf{x}^{(k)}$ 为原问题的近似极小解 \mathbf{x}_{\min} ; 否则, 取 $r_{k+1} < r_k$ (例如取 $r_{k+1} = r_k/10$ 或 $r_k/5$), 令 $k := k + 1$, 转向第 3 步继续进行迭代。

联立解上述两个方程, 得

$$x_1(r) = \sqrt{1 + \sqrt{r}}, \quad x_2(r) = \sqrt{r}$$

如此得最优解:

$$\mathbf{x}_{\min} = \lim_{r \rightarrow 0} \left(\sqrt{1 + \sqrt{r}}, \sqrt{r} \right)^T = (1, 0)^T$$

由于此例可解析求解, 故可如上进行。但很多实际问题不便用解析法, 仍需用迭代法求解。

例 12.2.4. 使用内点法解

$$\min f_0(\mathbf{x}) = x_1 + x_2$$

$$\text{s.t. } f_1(\mathbf{x}) = x_1^2 - x_2 \leq 0$$

$$f_2(\mathbf{x}) = -x_1 \leq 0$$

解. 构造对数障碍函数如下:

$$\bar{P}(\mathbf{x}, r) = x_1 + x_2 - r \log(-x_1^2 + x_2) - r \log x_1$$

各次迭代结果示于表 12.2 和图 12.23。

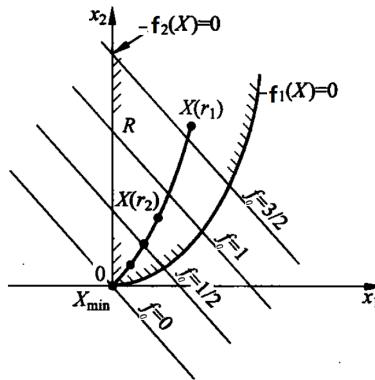


图 12.23

障碍因子	r	$x_1(r)$	$x_2(r)$
r_1	1.000	0.500	1.250
r_2	0.500	0.309	0.595
r_3	0.250	0.183	0.283
r_4	0.100	0.085	0.107
r_5	0.0001	0.000	0.000

表 12.2

初始内点

实际上, 上述内点法还存在一个未解决的问题: 我们知道, 内点法的迭代过程必须由某个内点开始。这在高维空间并不是一件简单的事情。在处理实际问题时, 如果不能找出某个内点作为初始点, 内点法的迭代就无法展开。那么, 怎样得到初始的内点呢?

我们可以尝试先任找一点 $\mathbf{x}^{(0)}$ 为初始点, 则约束条件只会存在于如下两类:

$$S_0 = \{i \mid -f_i(\mathbf{x}^{(0)}) \leq 0, \quad 1 \leq i \leq m\}$$

$$T_0 = \{i \mid -f_i(\mathbf{x}^{(0)}) > 0, \quad 1 \leq i \leq m\}$$

我们期望在寻找的过程中 T_0 内的条件始终满足, 而 S_0 中的约束条件趋向满足 (即函数值增大)。这恰好等价于, 求解将 T_0 作为约束条件, 而 S_0 内约束条件的函数之和的相反数作为目标的优化问题。这时将 $\mathbf{x}^{(0)}$ 作为初始点, 用内点法求解该优化问题得到下一迭代点重复上述步骤, 至所有约束条件得到满足, 即得到原问题的初始内点。

初始内点的具体迭代步骤如下:

1. 任取一点 $\mathbf{x}^{(0)}, r_0 > 0$ (例如 $r_0 = 1$), 令 $k = 0$;

2. 定出指标集 S_k 及 T_k

$$S_k = \{i \mid -f_i(\mathbf{x}^{(k)}) \leq 0, \quad 1 \leq i \leq m\}$$

$$T_k = \{i \mid -f_i(\mathbf{x}^{(k)}) > 0, \quad 1 \leq i \leq m\}$$

3. 检查集合 S_k 是否为空集, 若为空集, 则 $\mathbf{x}^{(k)}$ 在 R_0 内, 初始内点找到, 迭代停止, 否则转向第 4 步;

4. 构造函数

$$\tilde{P}(\mathbf{x}, r_k) = \sum_{i \in S_k} f_i(\mathbf{x}) + r_k \sum_{i \in T_k} \frac{1}{-f_i(\mathbf{x})}, \quad (r_k > 0)$$

以 $\mathbf{x}^{(k)}$ 为初始点, 在保持对集合

$$\tilde{R}_k = \{\mathbf{x} \mid -f_i(\mathbf{x}) > 0, \quad i \in T_k\}$$

可行的情况下, 极小化 $\tilde{P}(\mathbf{x}, r_k)$, 即

$$\min \tilde{P}(\mathbf{x}, r_k), \quad \mathbf{x} \in \tilde{R}_k$$

得 $\mathbf{x}^{(k+1)}, \mathbf{x}^{(k+1)} \in \tilde{R}_k$, 转向第 5 步;

5. 令 $0 < r_{k+1} < r_k$ (比如说 $r_{k+1} = r_k/10, k := k+1$), 转向第 2 步。

应用实例：对数障碍函数求解网络比率优化问题

我们考虑最优网络流问题的一种变形, 有时被称为网络比率优化问题。我们用 L 个弧或边组成的有向图描述网络。货物或信息包通过边在网络上移动。网络支持 n 个流, 它们的(非负的)比率 x_1, \dots, x_n 是优化变量。每个流沿着网络上一个固定的, 或预先设定的道路(或线路)从源结点向目标结点移动。每条边可以支持多个流通过。每条边上的总交通量等于通过它的所有流量的比率之和。每条边有一个正的容量, 这是在它上面能够通过的总交通量的最大值。

我们可以用下面定义的流-边关联矩阵 $\mathbf{A} \in \mathbf{R}^{L \times n}$ 描述这些边上的容量限制,

$$A_{ij} = \begin{cases} 1 & \text{流 } j \text{ 通过边 } i \\ 0 & \text{其他情况。} \end{cases}$$

这样就可以将边 i 上的总交通量写成 $(\mathbf{Ax})_i$, 于是边容量约束可以用 $\mathbf{Ax} \leq \mathbf{c}$ 表示, 其中 c_i 是边 i 上的容量。通常每个道路只通过所有边中的一小部分, 因此 \mathbf{A} 是稀疏矩阵。

在网络比率问题中道路是固定的(并作为问题的参数记录在矩阵 \mathbf{A} 中); 变量是流的比率 x_i 。目标是选择流的比率使下面的可分效用函数 U 达到最大,

$$U(\mathbf{x}) = U_1(x_1) + \dots + U_n(x_n)$$

我们假定每个 U_i (从而 U) 是凹和非减的。可以把 $U_i(x_i)$ 视为通过以比率 x_i 支持第 i 个流产生的收入; 于是 $U(\mathbf{x})$ 是相应的流产生的总收入。网络比率优化问题可写成

$$\max U(\mathbf{x})$$

$$\text{s.t. } \mathbf{Ax} \leq \mathbf{c}, \quad \mathbf{x} \geq 0$$

这是一个凸优化问题。我们用对数障碍方法构造对应的无约束优化问题, 其目标函数为

$$-tU(\mathbf{x}) - \sum_{i=1}^L \log(\mathbf{c} - \mathbf{Ax})_i - \sum_{j=1}^n \log x_j.$$

每步迭代我们需要用 Newton 方法求解对应的无约束优化, 确定 Newton 步径 $\Delta\mathbf{x}_{\text{nt}}$ 需要求解线性方程组

$$(\mathbf{D}_0 + \mathbf{A}^T \mathbf{D}_1 \mathbf{A} + \mathbf{D}_2) \Delta\mathbf{x}_{\text{nt}} = -\mathbf{g},$$

其中

$$\begin{aligned}\mathbf{D}_0 &= -t \operatorname{diag}(U_1''(\mathbf{x}), \dots, U_n''(\mathbf{x})) \\ \mathbf{D}_1 &= \operatorname{diag}(1/(\mathbf{c} - \mathbf{Ax})_1^2, \dots, 1/(\mathbf{c} - \mathbf{Ax})_L^2) \\ \mathbf{D}_2 &= \operatorname{diag}(1/x_1^2, \dots, 1/x_n^2)\end{aligned}$$

是对角矩阵, 而 $\mathbf{g} \in \mathbf{R}^n$ 。我们可以精确描述这个 $n \times n$ 系数矩阵的稀疏结构: 当且仅当流 i 和流 j 共享一条边时才成立

$$(\mathbf{D}_0 + \mathbf{A}^T \mathbf{D}_1 \mathbf{A} + \mathbf{D}_2)_{ij} \neq 0.$$

如果道路都比较短, 而每条边只有较少的道路通过, 那么这个矩阵就是稀疏的, 因此其稀疏的 Cholesky 因式分解可以被利用。当 Newton 系统的某些 (不是很多) 行和列稠密时, 我们也可以进行有效的求解。这种情况发生于仅有很少数的流和大量的流相交的时候, 如果比较长的流很少就可能出现这种情况。

12.3 复合优化算法

本节主要考虑如下复合优化问题:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \psi(\mathbf{x}) \stackrel{\text{def}}{=} f(\mathbf{x}) + h(\mathbf{x}), \quad (12.74)$$

其中 $f(\mathbf{x})$ 为可微函数 (可能非凸), $h(\mathbf{x})$ 可能为不可微函数。问题 (12.74) 出现在很多应用领域中, 例如压缩感知、图像处理、机器学习等, 如何高效求解该问题是近年来的热门课题。在梯度法求解 LASSO 问题中, 我们曾利用光滑化的思想处理不可微项 $h(\mathbf{x})$, 但这种做法没有充分利用 $h(\mathbf{x})$ 的性质, 在实际应用中有一定的局限性。在本节内容, 我们将介绍若干适用于求解问题 (12.74) 的优化算法。我们首先引入针对该问题的近似点梯度法和 Nesterov 加速算法, 之后介绍求解特殊结构复合优化问题的分块坐标下降法及交替方向乘子法。需要注意的是, 许多实际问题并不直接具有本节介绍的算法所能处理的形式, 我们需要利用拆分、引入辅助变量等技巧将其进行等价变形, 最终化为本节所介绍的优化问题形式。

12.3.1 近似点梯度法

在机器学习、图像处理领域中,许多模型包含两部分:一部分是误差项,一般为光滑函数;另外一部分是正则项,可能为非光滑函数,用来保证求解问题的特殊结构.例如最常见的 LASSO 问题就是用 ℓ_1 范数构造正则项保证求解的参数是稀疏的,从而起到筛选变量的作用.由于有非光滑部分的存在,此类问题属于非光滑的优化问题,我们可以考虑使用次梯度算法进行求解.然而次梯度算法并不能充分利用光滑部分的信息,也很难在迭代中保证非光滑项对应的解的结构信息,这使得次梯度算法在求解这类问题时往往收敛较慢.

本小节将介绍求解这类问题非常有效的一种算法—近似点梯度法.它能克服次梯度算法的缺点,充分利用光滑部分的信息,并在迭代过程中显式地保证解的结构,从而能够达到和求解光滑问题的梯度算法相近的收敛速度.在后面的内容中,我们首先引入邻近算子,它是近似点梯度算法中处理非光滑部分的关键;接着介绍近似点梯度算法的迭代格式,并给出一些实际的例子.为了讨论简便,方便读者理解,我们主要介绍凸函数的情形.

邻近算子

邻近算子是处理非光滑问题的一个非常有效的工具,也与许多算法的设计密切相关,比如我们即将介绍的近似点梯度法.当然该算子并不局限于非光滑函数,也可以用来处理光滑函数.本小节将介绍邻近算子的相关内容,为引入近似点梯度算法做准备.首先给出邻近算子的定义.

定义 12.3.1. (邻近算子) 对于一个凸函数 h , 定义它的邻近算子为

$$\text{prox}_h(\mathbf{x}) = \arg \min_{\mathbf{u} \in \text{dom } h} \left\{ h(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \right\}.$$

可以看到,邻近算子的目的是求解一个距 \mathbf{x} 不算太远的点,并使函数值 $h(\mathbf{x})$ 也相对较小.一个很自然的问题是,上面给出的邻近算子的定义是不是有意义的,即定义中的优化问题的解是不是存在唯一的.若答案是肯定的,我们就可使用邻近算子去构建迭代格式.下面的定理将给出定义中优化问题解的存在唯一性.

定理 12.3.1. (存在唯一性) 如果 h 是适当的闭凸函数,则对任意的 $\mathbf{x} \in \mathbb{R}^n$, $\text{prox}_h(\mathbf{x})$ 的值存在且唯一.

证明.为了简化证明,我们假设 h 至少在定义域内的一点处存在次梯度,保证次梯度存在的一个充分条件是 $\text{dom } h$ 内点集非空.我们定义辅助函数

$$m(\mathbf{u}) = h(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2,$$

下面说明 $m(\mathbf{u})$ 最小值点的存在性.因为 $h(\mathbf{u})$ 是凸函数,且至少在一点处存在次梯度,所以 $h(\mathbf{u})$ 有全局下界:

$$h(\mathbf{u}) \geq h(\mathbf{v}) + \theta^T(\mathbf{u} - \mathbf{v}),$$

这里 $\mathbf{v} \in \text{dom } h$, $\boldsymbol{\theta} \in \partial h(\mathbf{v})$. 进而得到

$$\begin{aligned} m(\mathbf{u}) &= h(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \\ &\geq h(\mathbf{v}) + \boldsymbol{\theta}^T(\mathbf{u} - \mathbf{v}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \end{aligned}$$

这表明 $m(\mathbf{u})$ 具有二次下界. 容易验证 $m(\mathbf{u})$ 为适当闭函数且具有强制性 (当 $\|\mathbf{u}\| \rightarrow +\infty$ 时, $m(\mathbf{u}) \rightarrow +\infty$).

因为 $m(\mathbf{u})$ 是适当的, 故存在 \mathbf{u}_0 使得 $m(\mathbf{u}_0) < +\infty$. 记 $\bar{\gamma} = m(\mathbf{u}_0)$, 并定义下水平集 $C_{\bar{\gamma}} \stackrel{\text{def}}{=} \{\mathbf{u} : m(\mathbf{u}) \leq \bar{\gamma}\}$. 因为 m 是强制的, 则 $C_{\bar{\gamma}}$ 是非空有界的 (假设无界, 则存在点列 $\{\mathbf{u}^k\} \subset C_{\bar{\gamma}}$ 满足 $\lim_{k \rightarrow \infty} \|\mathbf{u}^k\| = +\infty$, 由强制性有 $\lim_{k \rightarrow \infty} m(\mathbf{u}^k) = +\infty$, 这与 $m(\mathbf{u}) \leq \bar{\gamma}$ 矛盾).

我们先证下确界 $t \stackrel{\text{def}}{=} \inf_{\mathbf{u}} m(\mathbf{u}) > -\infty$. 采用反证法. 假设 $t = -\infty$, 则存在点列 $\{\mathbf{u}^k\}_{k=1}^{\infty} \subset C_{\bar{\gamma}}$, 使得 $\lim_{k \rightarrow \infty} m(\mathbf{u}^k) = t = -\infty$. 因为 $C_{\bar{\gamma}}$ 的有界性, 点列 $\{\mathbf{u}^k\}$ 一定存在聚点, 记为 \mathbf{u}^* . 根据上方图的闭性, 我们知道 $(\mathbf{u}^*, t) \in \text{epi } m$, 即有 $m(\mathbf{u}^*) \leq t = -\infty$. 这与函数的适当性矛盾, 故 $t > -\infty$. 利用上面的论述, 我们知道 $f(\mathbf{u}^*) \leq t$. 因为 t 是下确界, 故必有 $f(\mathbf{u}^*) = t$. 这就证明了下确界是可取得的.

接下来证明唯一性. 注意到 $m(\mathbf{u})$ 是强凸函数, 根据强凸函数的性质可直接得出 $m(\mathbf{u})$ 的最小值唯一. 综上 $\text{prox}_h(\mathbf{x})$ 是良定义的. \square

另外, 根据最优性条件可以得到如下等价结论:

定理 12.3.2. (邻近算子与次梯度的关系) 如果 h 是适当的闭凸函数, 则

$$\mathbf{u} = \text{prox}_h(\mathbf{x}) \iff \mathbf{x} - \mathbf{u} \in \partial h(\mathbf{u})$$

证明. 若 $\mathbf{u} = \text{prox}_h(\mathbf{x})$, 则由最优性条件得 $0 \in \partial h(\mathbf{u}) + (\mathbf{x} - \mathbf{u})$, 因此有 $\mathbf{x} - \mathbf{u} \in \partial h(\mathbf{u})$. 反之, 若 $\mathbf{x} - \mathbf{u} \in \partial h(\mathbf{u})$ 则由次梯度的定义可得到

$$h(\mathbf{v}) \geq h(\mathbf{u}) + (\mathbf{x} - \mathbf{u})^T(\mathbf{v} - \mathbf{u}), \quad \forall \mathbf{v} \in \text{dom } h.$$

两边同时加 $\frac{1}{2} \|\mathbf{v} - \mathbf{x}\|^2$, 即有

$$\begin{aligned} h(\mathbf{v}) + \frac{1}{2} \|\mathbf{v} - \mathbf{x}\|^2 &\geq h(\mathbf{u}) + (\mathbf{x} - \mathbf{u})^T(\mathbf{v} - \mathbf{u}) + \frac{1}{2} \|\mathbf{v} - \mathbf{x}\|^2 \\ &\geq h(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2, \quad \forall \mathbf{v} \in \text{dom } h \end{aligned}$$

因此我们得到 $\mathbf{u} = \text{prox}_h(\mathbf{x})$. \square

用 th 代替 h , 上面的等价结论形式上可以写成

$$\mathbf{u} = \text{prox}_{th}(\mathbf{x}) \iff \mathbf{u} \in \mathbf{x} - t\partial h(\mathbf{u}).$$

邻近算子的计算可以看成是次梯度算法的隐式格式 (后向迭代). 对于非光滑情形, 由于次梯度不唯一, 显式格式的迭代并不唯一, 而隐式格式却能得到唯一解. 此外在步长的选择上面, 隐式格式也要优于显式格式. 下面给出一些常见的 ℓ_1 范数和 ℓ_2 范数对应的例子. 计算邻近算子的过程实际上是在求解一个优化问题, 下面给出具体计算过程.

例 12.3.1. (邻近算子的例子) 在下面所有例子中, 常数 $t > 0$ 为正实数.

(1) ℓ_1 范数:

$$h(\mathbf{x}) = \|\mathbf{x}\|_1, \quad \text{prox}_{th}(\mathbf{x}) = \text{sign}(\mathbf{x}) \max\{|\mathbf{x}| - t, 0\}.$$

(2) ℓ_2 范数:

$$h(\mathbf{x}) = \|\mathbf{x}\|_2, \quad \text{prox}_{th}(\mathbf{x}) = \begin{cases} \left(1 - \frac{t}{\|\mathbf{x}\|_2}\right) \mathbf{x}, & \|\mathbf{x}\|_2 \geq t, \\ 0, & \text{其他.} \end{cases}$$

证明. (1) 因为求解 ℓ_1 范数的邻近算子, 所对应的优化问题是可拆分的. 因此, 我们只需要考虑一维的情形. 易知, 邻近算子 $u = \text{prox}_{th}(x)$ 的最优性条件为

$$x - u \in t\partial|u| = \begin{cases} \{t\}, & u > 0, \\ [-t, t], & u = 0, \\ \{-t\}, & u < 0, \end{cases}$$

因此, 当 $x > t$ 时, $u = x - t$; 当 $x < -t$ 时, $u = x + t$; 当 $x \in [-t, t]$ 时, $u = 0$, 即有 $u = \text{sign}(x) \max\{|x| - t, 0\}$.

(2) 邻近算子 $\mathbf{u} = \text{prox}_{th}(\mathbf{x})$ 的最优性条件为

$$x - \mathbf{u} \in t\partial\|\mathbf{u}\|_2 = \begin{cases} \left\{ \frac{tu}{\|\mathbf{u}\|_2} \right\}, & \mathbf{u} \neq 0, \\ \{\mathbf{w} : \|\mathbf{w}\|_2 \leq t\}, & \mathbf{u} = 0, \end{cases}$$

因此, 当 $\|\mathbf{x}\|_2 > t$ 时, $\mathbf{u} = \mathbf{x} - \frac{tx}{\|\mathbf{x}\|_2}$; 当 $\|\mathbf{x}\|_2 \leq t$ 时, $\mathbf{u} = 0$.

□

另外一种比较常用的邻近算子是关于示性函数的邻近算子. 集合 C 的示性函数定义为

$$I_C(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in C, \\ +\infty, & \text{其他,} \end{cases}$$

它可以用来把约束变成目标函数的一部分.

例 12.3.2. (闭凸集上的投影) 设 C 为 \mathbb{R}^n 上的闭凸集, 则示性函数 I_C 的邻近算子为点 \mathbf{x} 到集合 C 的投影, 即

$$\begin{aligned} \text{prox}_{I_C}(\mathbf{x}) &= \arg \min_u \left\{ I_C(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \right\} \\ &= \arg \min_{\mathbf{u} \in C} \|\mathbf{u} - \mathbf{x}\|^2 = \mathcal{P}_C(\mathbf{x}). \end{aligned}$$

此外, 应用定理 12.3.2 可进一步得到

$$\mathbf{u} = \mathcal{P}_C(\mathbf{x}) \Leftrightarrow \mathbf{x} - \mathbf{u} \in \partial I_C(\mathbf{u})$$

$$\Leftrightarrow (\mathbf{x} - \mathbf{u})^\top (\mathbf{z} - \mathbf{u}) \leq I_C(\mathbf{z}) - I_C(\mathbf{u}) = 0, \quad \forall \mathbf{z} \in C$$

此结论有较强的几何意义: 若点 \mathbf{x} 位于 C 外部, 则从投影点 \mathbf{u} 指向 \mathbf{x} 的向量与任意起点为 \mathbf{u} 且指向 C 内部的向量的夹角为直角或钝角.

近似点梯度法

下面将引入本节的重点：近似点梯度算法。我们将考虑如下复合优化问题：

$$\min \psi(\mathbf{x}) = f(\mathbf{x}) + h(\mathbf{x}), \quad (12.75)$$

其中函数 f 为可微函数，其定义域 $\text{dom } f = \mathbb{R}^n$ ，函数 h 为凸函数，可以是非光滑的，并且一般计算此项的邻近算子并不复杂。比如 LASSO 问题，两项分别为 $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2$, $h(\mathbf{x}) = \mu \|\mathbf{x}\|_1$ 。一般的带凸集约束的优化问题也可以用 (12.75) 式表示，即对问题

$$\min_{\mathbf{x} \in C} \phi(\mathbf{x}),$$

复合优化问题中的两项可以写作 $f(\mathbf{x}) = \phi(\mathbf{x})$, $h(\mathbf{x}) = I_C(\mathbf{x})$ ，其中 $I_C(\mathbf{x})$ 为示性函数。近似点梯度法的思想非常简单：注意到 $\psi(\mathbf{x})$ 有两部分，对于光滑部分 f 做梯度下降，对于非光滑部分 h 使用邻近算子，则近似点梯度法的迭代公式为

$$\mathbf{x}^{k+1} = \text{prox}_{t_k h}(\mathbf{x}^k - t_k \nabla f(\mathbf{x}^k)),$$

其中 $t_k > 0$ 为每次迭代的步长，它可以是一个常数或者由线搜索得出。近似点梯度法跟众多算法都有很强的联系，在一些特定条件下，近似点梯度法还可以转化为其他算法：当 $h(\mathbf{x}) = 0$ 时，迭代公式变为梯度下降法

$$\mathbf{x}^{k+1} = \mathbf{x}^k - t_k \nabla f(\mathbf{x}^k)$$

当 $h(\mathbf{x}) = I_C(\mathbf{x})$ 时，迭代公式变为投影梯度法

$$\mathbf{x}^{k+1} = \mathcal{P}_C(\mathbf{x}^k - t_k \nabla f(\mathbf{x}^k)).$$

近似点梯度法可以总结为算法 12.14。

算法 12.14 近似点梯度法

Require: 函数 $f(\mathbf{x}), h(\mathbf{x})$, 初始点 \mathbf{x}^0 . 初始化 $k = 0$.

- 1: **while** 未达到停止准则 **do**
- 2: $\mathbf{x}^{k+1} = \text{prox}_{t_k h}(\mathbf{x}^k - t_k \nabla f(\mathbf{x}^k)).$
- 3: $k \leftarrow k + 1.$
- 4: **end while**

如何理解近似点梯度法？根据邻近算子的定义，把迭代公式展开：

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg \min_{\mathbf{u}} \left\{ h(\mathbf{u}) + \frac{1}{2t_k} \|\mathbf{u} - \mathbf{x}^k + t_k \nabla f(\mathbf{x}^k)\|^2 \right\} \\ &= \arg \min_{\mathbf{u}} \left\{ h(\mathbf{u}) + f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T (\mathbf{u} - \mathbf{x}^k) + \frac{1}{2t_k} \|\mathbf{u} - \mathbf{x}^k\|^2 \right\}, \end{aligned}$$

可以发现，近似点梯度法实质上就是将问题的光滑部分线性展开，再加上二次项并保留非光滑部分，然后求极小来作为每一步的估计。此外，根据定理 12.3.2，近似点梯度算法可以形式上写成

$$\mathbf{x}^{k+1} = \mathbf{x}^k - t_k \nabla f(\mathbf{x}^k) - t_k \mathbf{g}^k, \quad \mathbf{g}^k \in \partial h(\mathbf{x}^{k+1}).$$

其本质上是对光滑部分做显式的梯度下降, 关于非光滑部分做隐式的梯度下降. 算法 12.14 中步长 t_k 的选取较为关键. 当 f 为梯度 L -利普希茨连续函数时, 可取固定步长 $t_k = t \leq \frac{1}{L}$. 当 L 未知时可使用线搜索准则

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T (\mathbf{x}^{k+1} - \mathbf{x}^k) + \frac{1}{2t_k} \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2.$$

Nesterov 加速

上一小节介绍了近似点梯度算法, 如果光滑部分的梯度是利普希茨连续的, 则它的收敛速度可以达到 $\mathcal{O}(\frac{1}{k})$. 一个自然的问题是如果仅用梯度信息, 我们能不能取得更快的收敛速度. Nesterov 分别在 1983 年、1988 年和 2005 年提出了三种改进的一阶算法, 收敛速度能达到 $\mathcal{O}(\frac{1}{k^2})$. 实际上, 这三种算法都可以应用到近似点梯度算法上. 在 Nesterov 加速算法刚提出的时候, 由于牛顿算法有更快的收敛速度, Nesterov 加速算法在当时并没有引起太多的关注. 但近年来, 随着数据量的增大, 牛顿型方法由于其过大的计算复杂度, 不便于有效地应用到实际中, Nesterov 加速算法作为一种快速的一阶算法重新被挖掘出来并迅速流行起来. Beck 和 Teboulle 就在 2008 年给出了 Nesterov 在 1983 年提出的算法的近似点梯度版 FISTA. 本节将对这些加速方法做一定的介绍和总结, 为了便于读者的理解, 我们仍将主要讨论凸函数的加速算法.

考虑如下复合优化问题:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \psi(\mathbf{x}) = f(\mathbf{x}) + h(\mathbf{x}),$$

其中 $f(\mathbf{x})$ 是连续可微的凸函数且梯度是利普希茨连续的 (利普希茨常数是 L), $h(\mathbf{x})$ 是适当的闭凸函数. 我们希望能够利用 Nesterov 加速近似点梯度算法, 这就是本小节要介绍的 FISTA 算法.

FISTA 算法由两步组成: 第一步沿着前两步的计算方向计算一个新点, 第二步在该新点处做一步近似点梯度迭代, 即

$$\begin{aligned} \mathbf{y}^k &= \mathbf{x}^{k-1} + \frac{k-2}{k+1} (\mathbf{x}^{k-1} - \mathbf{x}^{k-2}), \\ \mathbf{x}^k &= \text{prox}_{t_k h}(\mathbf{y}^k - t_k \nabla f(\mathbf{y}^k)). \end{aligned}$$

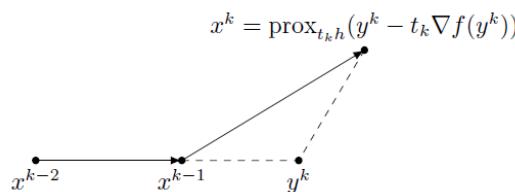


图 12.24: FISTA 算法迭代图示

从图 12.24 中可以直观地看出 FISTA 算法的迭代情况. 可以得到这一做法对每一步迭代的计算量几乎没有影响, 而带来的效果是显著的. 如果选取 t_k 为固定的步长并小于或等于 $\frac{1}{L}$, 其收敛速度达到了 $\mathcal{O}(\frac{1}{k^2})$. 感兴趣的读者, 可以在相关参考文献中查阅这一收敛性分析的推导过程. 完整的 FISTA 算法见算法 12.15.

算法 12.15 FISTA 算法

Require: $\mathbf{x}^0 = \mathbf{x}^{-1} \in \mathbb{R}^n, k \leftarrow 1$.

- 1: **while** 未达到停止准则 **do**
- 2: 计算 $\mathbf{y}^k = \mathbf{x}^{k-1} + \frac{k-2}{k+1} (\mathbf{x}^{k-1} - \mathbf{x}^{k-2})$,
- 3: 选取 $t_k = t \in (0, \frac{1}{L}]$, 计算 $\mathbf{x}^k = \text{prox}_{t_k h}(\mathbf{y}^k - t_k \nabla f(\mathbf{y}^k))$.
- 4: $k \leftarrow k + 1$
- 5: **end while**

为了对算法做更好的推广, 可以给出 FISTA 算法的一个等价变形, 只是把原来算法中的第一步拆成两步迭代, 相应算法见算法 12.16. 当 $\gamma_k = \frac{2k}{k+1}$ 时, 并且取固定步长时, 两个算法是等价的. 但是当 γ_k 采用别的取法时, 算法 12.16 将给出另一个版本的加速算法.

算法 12.16 FISTA 等价变形

Require: $\mathbf{v}_0 = \mathbf{x}_0 \in \mathbb{R}^n, k \leftarrow 1$.

- 1: **while** 未达到停止准则 **do**
- 2: 计算 $\mathbf{y}^k = (1 - \gamma_k) \mathbf{x}^{k-1} + \gamma_k \mathbf{v}^{k-1}$.
- 3: 选取 t_k , 计算 $\mathbf{x}^k = \text{prox}_{t_k h}(\mathbf{y}^k - t_k \nabla f(\mathbf{y}^k))$.
- 4: 计算 $\mathbf{v}^k = \mathbf{x}^{k-1} + \frac{1}{\gamma_k} (\mathbf{x}^k - \mathbf{x}^{k-1})$.
- 5: $k \leftarrow k + 1$.
- 6: **end while**

应用实例: 低秩矩阵恢复

考虑将等式约束转化为二次罚函数项的低秩矩阵恢复模型:

$$\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \mu \|\mathbf{X}\|_* + \frac{1}{2} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2,$$

其中 \mathbf{M} 是想要恢复的低秩矩阵, 但是只知道其在下标集 Ω 上的值. 令

$$f(\mathbf{X}) = \frac{1}{2} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2, \quad h(\mathbf{X}) = \mu \|\mathbf{X}\|_*$$

定义矩阵 $\mathbf{P} \in \mathbb{R}^{m \times n}$:

$$P_{ij} = \begin{cases} 1, & (i, j) \in \Omega, \\ 0, & \text{其他,} \end{cases}$$

则

$$f(\mathbf{X}) = \frac{1}{2} \|\mathbf{P} \odot (\mathbf{X} - \mathbf{M})\|_F^2,$$

$$\nabla f(\mathbf{X}) = \mathbf{P} \odot (\mathbf{X} - \mathbf{M}),$$

$$\text{prox}_{t_k h}(\mathbf{X}) = \mathbf{U} \text{Diag}(\max\{|\mathbf{d}| - t_k \mu, 0\}) \mathbf{V}^T,$$

其中 $\mathbf{X} = \mathbf{U} \text{Diag}(\mathbf{d}) \mathbf{V}^T$ 为矩阵 \mathbf{X} 的约化的奇异值分解. 则近似点梯度法的迭代格式为

$$\mathbf{Y}^k = \mathbf{X}^k - t_k \mathbf{P} \odot (\mathbf{X}^k - \mathbf{M}),$$

$$\mathbf{X}^{k+1} = \text{prox}_{t_k h}(\mathbf{Y}^k)$$

对应的加速算法 FISTA 的迭代格式为

$$\mathbf{Y}^k = \mathbf{X}^{k-1} + \frac{k-2}{k+1} (\mathbf{X}^{k-1} - \mathbf{X}^{k-2}) \mathbf{Z}^k = \mathbf{Y}^k - t_k \mathbf{P} \odot (\mathbf{Y}^k - \mathbf{M}),$$

$$\mathbf{X}^{k+1} = \text{prox}_{t_k h}(\mathbf{Z}^k)$$

12.3.2 分块坐标下降法

在许多实际的优化问题中, 人们所考虑的目标函数虽然有成千上万的自变量, 对这些变量联合求解目标函数的极小值通常很困难, 但这些自变量具有某种“可分离”的形式: 当固定其中若干变量时, 函数的结构会得到极大的简化. 这种特殊的形式使得人们可以将原问题拆分成数个只有少数自变量的子问题. 分块坐标下降法 (block coordinate descent, BCD) 正是利用了这样的思想来求解这种具有特殊结构的优化问题, 在多数实际问题中有良好的数值表现. 本小节介绍分块坐标下降法的基本迭代格式, 同时给出它在 K -均值聚类和非负矩阵分解问题中的应用.

问题引入

考虑具有如下形式的问题:

$$\min_{\mathbf{x} \in \mathcal{X}} F(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_s) = f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_s) + \sum_{i=1}^s r_i(\mathbf{x}_i), \quad (12.76)$$

其中 \mathcal{X} 是函数的可行域, 这里将自变量 \mathbf{x} 拆分成 s 个变量块 $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_s$, 每个变量块 $\mathbf{x}_i \in \mathbb{R}^{n_i}$. 函数 f 是关于 \mathbf{x} 的可微函数, 每个 $r_i(\mathbf{x}_i)$ 关于 \mathbf{x}_i 是适当的闭凸函数, 但不一定可微.

在问题 (12.76) 中, 目标函数 F 的性质体现在 f , 每个 r_i 以及自变量的分块上. 通常情况下, f 对于所有变量块 \mathbf{x}_i 不可分, 但单独考虑每一块自变量时, f 有简单结构; r_i 只和第 i 个自变量块有关, 因此 r_i 在目标函数中是一个可分项. 求解问题 (12.76) 的难点在于如何利用分块结构处理不可分的函数 f . 需要注意, 在给出问题 (12.76) 时, 唯一引入凸性的部分是 r_i . 其余部分没有引入凸性, 可行域 \mathcal{X} 不是一定是凸集, f 也不一定是凸函数.

需要指出的是, 并非所有问题都适合按照问题 (12.76) 进行处理. 下面给出两个例子, 并将在应用实例中介绍如何使用分块坐标下降法求解它们.

例 12.3.3. (聚类问题) 前面我们介绍了 K -均值聚类问题的等价形式:

$$\begin{aligned} \min_{\Phi, H} \quad & \|A - \Phi H\|_{F'}^2 \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n \times k}, \text{每一行只有一个元素为1, 其余为0,} \\ & H \in \mathbb{R}^{k \times p}. \end{aligned}$$

这是一个矩阵分解问题, 自变量总共有两块. 注意到变量 Φ 取值在离散空间上, 因此聚类问题不是凸问题.

例 12.3.4. (非负矩阵分解) 设 M 是已知矩阵, 考虑求解如下极小化问题:

$$\min_{X, Y \geq 0} \frac{1}{2} \|XY - M\|_F^2 + \alpha r_1(X) + \beta r_2(Y).$$

在这个例子中自变量总共有两块, 且均有非负的约束.

上述的所有例子中, 函数 f 关于变量全体一般是非凸的, 这使得求解问题 (12.76) 变得很有挑战性. 首先, 应用在非凸问题上的算法的收敛性不易分析, 很多针对凸问题设计的算法通常会失效; 其次, 目标函数的整体结构十分复杂, 这使得变量的更新需要很大计算量. 对于这类问题, 我们最终的目标是要设计一种算法, 它具有简单的变量更新格式, 同时具有一定的(全局)收敛性. 而分块坐标下降法则是处理这类问题较为有效的算法.

算法结构

考虑问题 (12.76), 我们所感兴趣的分块坐标下降法具有如下更新方式: 按照 x_1, x_2, \dots, x_s 的次序依次固定其他 $(s-1)$ 块变量极小化, 完成一块变量的极小化后, 它的值便立即被更新到变量空间中, 更新下一块变量时将使用每个变量最新的值. 根据这种更新方式定义辅助函数

$$f_i^k(x_i) = f(x_1^k, \dots, x_{i-1}^k, x_i, x_{i+1}^{k-1}, \dots, x_s^{k-1}),$$

其中 x_j^k 表示在第 k 次迭代中第 j 块自变量的值, x_i 是函数的自变量. 函数 f_i^k 表示在第 k 次迭代更新第 i 块变量时所需要考虑的目标函数的光滑部分. 考虑第 i 块变量时前 $(i-1)$ 块变量已经完成更新, 因此上标为 k . 而后面下标从 $(i+1)$ 起的变量仍为旧的值, 因此上标为 $(k-1)$. 在每一步更新中, 通常使用以下三种更新格式之一:

$$x_i^k = \arg \min_{x_i \in \mathcal{X}_i^k} \{f_i^k(x_i) + r_i(x_i)\}, \quad (12.77)$$

$$x_i^k = \arg \min_{x_i \in \mathcal{X}_i^k} \left\{ f_i^k(x_i) + \frac{L_i^{k-1}}{2} \|x_i - x_i^{k-1}\|_2^2 + r_i(x_i) \right\} \quad (12.78)$$

$$x_i^k = \arg \min_{x_i \in \mathcal{X}_i^k} \left\{ \langle \hat{g}_i^k, x_i - \hat{x}_i^{k-1} \rangle + \frac{L_i^{k-1}}{2} \|x_i - \hat{x}_i^{k-1}\|_2^2 + r_i(x_i) \right\}, \quad (12.79)$$

其中 $L_i^k > 0$ 为常数,

$$\mathcal{X}_i^k = \{x \in \mathbb{R}^{n_i} \mid (x_1^k, \dots, x_{i-1}^k, x, x_{i+1}^{k-1}, \dots, x_s^{k-1}) \in \mathcal{X}\}.$$

在更新格式 (12.79) 中, $\hat{\mathbf{x}}_i^{k-1}$ 采用外推定义:

$$\hat{\mathbf{x}}_i^{k-1} = \mathbf{x}_i^{k-1} + \omega_i^{k-1} (\mathbf{x}_i^{k-1} - \mathbf{x}_i^{k-2}),$$

其中 $\omega_i^k \geq 0$ 为外推的权重, $\hat{\mathbf{g}}_i^k \stackrel{\text{def}}{=} \nabla f_i^k(\hat{\mathbf{x}}_i^{k-1})$ 为外推点处的梯度. 在外推式中取权重 $\omega_i^k = 0$ 即可得到不带外推的更新格式. 此时计算 (12.79) 等价于进行一次近似点梯度法的更新. 在 (12.79) 式使用外推是为了加快分块坐标下降法的收敛速度. 我们可以通过如下的方式理解这三种格式: 格式 (12.77) 是最直接的, 即固定其他分量然后对单一变量求极小; 格式 (12.78) 则是增加了一个近似点项 $\frac{L_i^{k-1}}{2} \|\mathbf{x}_i - \mathbf{x}_i^{k-1}\|_2^2$ 来限制下一步迭代不应该与当前位置相距过远, 增加近似点项的作用是使得算法能够收敛; 格式 (12.79) 首先对 $f_i^k(\mathbf{x})$ 进行线性化以简化子问题的求解, 在此基础上引入了 Nesterov 加速算法的技巧加快收敛.

为了直观地说明分块坐标下降法的迭代过程, 我们给出一个简单的例子.

例 12.3.5. 考虑二元二次函数的优化问题

$$\min f(x, y) = x^2 - 2xy + 10y^2 - 4x - 20y,$$

现在对变量 x, y 使用分块坐标下降法求解. 当固定 y 时, 可知当 $x = 2 + y$ 时函数取极小值; 当固定 x 时, 可知当 $y = 1 + \frac{x}{10}$ 时函数取极小值. 故采用格式 (12.77) 的分块坐标下降法为

$$\begin{aligned} x^{k+1} &= 2 + y^k \\ y^{k+1} &= 1 + \frac{x^{k+1}}{10} \end{aligned}$$

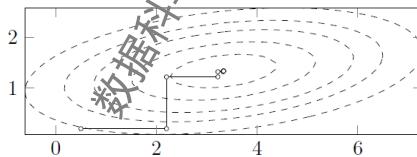


图 12.25

图 12.25 描绘了当初始点为 $(x, y) = (0.5, 0.2)$ 时的迭代点轨迹, 可以看到在进行了 7 次迭代后迭代点与最优解已充分接近. 回忆一下我们曾经对一个类似的问题使用过梯度法, 而梯度法的收敛相当缓慢. 一个直观的解释是: 对于比较病态的问题, 由于分块坐标下降法是对逐个分量处理, 它能较好地捕捉目标函数的各向异性, 而梯度法则会受到很大影响.

结合上述更新格式(12.77)-(12.79)可以得到分块坐标下降法的基本框架, 详见算法 12.17.

算法 12.17 的子问题可采用三种不同的更新格式, 一般来说这三种格式会产生不同的迭代序列, 可能会收敛到不同的解, 坐标下降算法的数值表现也不相同. 格式 (12.77) 是最直接的更新方式, 它严格保证了整个迭代过程的目标函数值是下降的. 然而由于 f 的形式复杂, 子问题求解难度较大. 在收敛性方面, 格式 (12.77) 在强凸问题上可保证目标函数收敛到极小值, 但在非凸问题

算法 12.17 分块坐标下降法

Require: 初始化: 选择两组初始点 $(\mathbf{x}_1^{-1}, \mathbf{x}_2^{-1}, \dots, \mathbf{x}_s^{-1}) = (\mathbf{x}_1^0, \mathbf{x}_2^0, \dots, \mathbf{x}_s^0)$.

```

1: for  $k = 1, 2, \dots$  do
2:   for  $i = 1, 2, \dots$  do
3:     使用格式 (12.77) 或 (12.78) 或 (12.79) 更新  $\mathbf{x}_i^k$ .
4:   end for
5:   if 满足停机条件 then
6:     返回  $(\mathbf{x}_1^k, \mathbf{x}_2^k, \dots, \mathbf{x}_s^k)$ , 算法终止.
7:   end if
8: end for

```

上不一定收敛. 格式 (12.78) (12.79) 则是对格式 (12.77) 的修正, 不保证迭代过程目标函数的单调性, 但可以改善收敛性结果. 使用格式 (12.78) 可使得算法收敛性在函数 F 为非严格凸时有所改善. 格式 (12.79) 实质上为目标函数的一阶泰勒展开近似, 在一些测试问题上有更好的表现, 可能的原因是使用一阶近似可以避开一些局部极小值点. 此外, 格式 (12.79) 的计算量很小, 比较容易实现.

应用实例: K 均值聚类

下面对聚类问题使用分块坐标下降法进行求解. 其目标函数为

$$\begin{aligned} \min_{\Phi, \mathbf{H}} \quad & \|\mathbf{A} - \Phi \mathbf{H}\|_F^2 \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n \times s} \text{ 每一行只有一个元素为1, 其余为0,} \\ & \mathbf{H} \in \mathbb{R}^{k \times p}. \end{aligned}$$

接下来分别讨论在固定 Φ 和 \mathbf{H} 的条件下如何极小化另一块变量. 当固定 \mathbf{H} 时, 设 Φ 的每一行为 ϕ_i^T , 那么根据矩阵分块乘法,

$$\mathbf{A} - \Phi \mathbf{H} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_n^T \end{bmatrix} - \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_n^T \end{bmatrix} \mathbf{H} = \begin{bmatrix} \mathbf{a}_1^T - \phi_1^T \mathbf{H} \\ \mathbf{a}_2^T - \phi_2^T \mathbf{H} \\ \vdots \\ \mathbf{a}_n^T - \phi_n^T \mathbf{H} \end{bmatrix}.$$

注意到 ϕ_i 只有一个分量为 1, 其余分量为 0, 不妨设其第 j 个分量为 1, 此时 $\phi_i^T \mathbf{H}$ 相当于将 \mathbf{H} 的第 j 行取出, 因此 $\|\mathbf{a}_i^T - \phi_i^T \mathbf{H}\|$ 为 \mathbf{a}_i^T 与 \mathbf{H} 的第 j 个行向量的距离. 我们的最终目的是极小化 $\|\mathbf{A} - \Phi \mathbf{H}\|_F^2$, 所以 j 应该选矩阵 \mathbf{H} 中距离 \mathbf{a}_i^T 最近的那一行, 即

$$\Phi_{ij} = \begin{cases} 1, & j = \arg \min_l \|\mathbf{a}_i - \mathbf{h}_l\|, \\ 0, & \text{其他.} \end{cases}$$

其中 \mathbf{h}_l^T 表示矩阵 \mathbf{H} 的第 l 行. 当固定 Φ 时, 此时考虑 \mathbf{H} 的每一行 \mathbf{h}_j^T , 根据目标函数的等价性有

$$\|\mathbf{A} - \Phi \mathbf{H}\|_F^2 = \sum_{j=1}^k \sum_{\mathbf{a} \in S_j} \|\mathbf{a} - \mathbf{h}_j\|^2,$$

因此只需要对每个 \mathbf{h}_j 求最小即可. 设 $\bar{\mathbf{a}}_j$ 是目前第 j 类所有点的均值, 则

$$\begin{aligned} \sum_{\mathbf{a} \in S_j} \|\mathbf{a} - \mathbf{h}_j\|^2 &= \sum_{\mathbf{a} \in S_j} \|\mathbf{a} - \bar{\mathbf{a}}_j + \bar{\mathbf{a}}_j - \mathbf{h}_j\|^2 \\ &= \sum_{\mathbf{a} \in S_j} \left(\|\mathbf{a} - \bar{\mathbf{a}}_j\|^2 + \|\bar{\mathbf{a}}_j - \mathbf{h}_j\|^2 + 2 \langle \mathbf{a} - \bar{\mathbf{a}}_j, \bar{\mathbf{a}}_j - \mathbf{h}_j \rangle \right) \\ &= \sum_{\mathbf{a} \in S_j} \left(\|\mathbf{a} - \bar{\mathbf{a}}_j\|^2 + \|\bar{\mathbf{a}}_j - \mathbf{h}_j\|^2 \right), \end{aligned}$$

这里利用了交叉项 $\sum_{\mathbf{a} \in S_j} \langle \mathbf{a} - \bar{\mathbf{a}}_j, \bar{\mathbf{a}}_j - \mathbf{h}_j \rangle = 0$ 的事实. 因此容易看出, 此时 \mathbf{h}_j 直接取为 $\bar{\mathbf{a}}_j$ 即可达到最小值. 综上, 我们得到了针对聚类问题的分块坐标下降法, 它每一次迭代分为两步: (1) 固定参考点 \mathbf{H} , 将每个样本点分到和其最接近的参考点代表的类中; (2) 固定聚类方式 Φ , 重新计算每个类所有点的均值并将其作为新的参考点. 这个过程恰好就是经典的 K-均值聚类算法, 因此可以得到结论: K-均值聚类算法本质上是一个分块坐标下降法.

应用实例: 非负矩阵分解

非负矩阵分解问题也可以使用分块坐标下降法求解. 现在考虑最基本的非负矩阵分解问题

$$\min_{\mathbf{X}, \mathbf{Y} \geq 0} \frac{1}{2} \|\mathbf{XY} - \mathbf{M}\|_F^2$$

它的一个等价形式为

$$\min_{\mathbf{X}, \mathbf{Y}} \frac{1}{2} \|\mathbf{XY} - \mathbf{M}\|_F^2 + I_{\geq 0}(\mathbf{X}) + I_{\geq 0}(\mathbf{Y}),$$

其中 $I_{\geq 0}(\cdot)$ 为集合 $\{\mathbf{X} \mid \mathbf{X} \geq 0\}$ 的示性函数. 不难验证该问题具有形式 (12.76). 以下考虑求解方法. 注意到 \mathbf{X} 和 \mathbf{Y} 耦合在一起, 在固定 \mathbf{Y} 的条件下, 我们无法直接按照格式 (12.77) 或格式 (12.78) 的形式给出子问题的显式解. 若要采用这两种格式需要额外设计算法求解子问题, 最终会产生较大计算量. 但我们总能使用格式 (12.79) 来对子问题进行线性化, 从而获得比较简单的更新格式. 今 $f(\mathbf{X}, \mathbf{Y}) = \frac{1}{2} \|\mathbf{XY} - \mathbf{M}\|_F^2$, 则

$$\frac{\partial f}{\partial \mathbf{X}} = (\mathbf{XY} - \mathbf{M}) \mathbf{Y}^T, \quad \frac{\partial f}{\partial \mathbf{Y}} = \mathbf{X}^T (\mathbf{XY} - \mathbf{M})$$

注意到在格式 (12.79) 中, 当 $r_i(\mathbf{X})$ 为凸集示性函数时即是求解到该集合的投影, 因此得到分块坐标下降法如下:

$$\begin{aligned} \mathbf{X}^{k+1} &= \max \left\{ \mathbf{X}^k - t_k^x (\mathbf{X}^k \mathbf{Y}^k - \mathbf{M}) (\mathbf{Y}^k)^T, 0 \right\} \\ \mathbf{Y}^{k+1} &= \max \left\{ \mathbf{Y}^k - t_k^y (\mathbf{X}^k)^T (\mathbf{X}^k \mathbf{Y}^k - \mathbf{M}), 0 \right\} \end{aligned}$$

其中 t_k^x, t_k^y 是步长, 分别对应格式 (12.79) 中的 $\frac{1}{L_i^k}, i = 1, 2$.

12.3.3 交替方向乘子法

统计学、机器学习和科学计算中出现了很多结构复杂且可能非凸、非光滑的优化问题。交替方向乘子法很自然地提供了一个适用范围广泛、容易理解和实现、可靠性不错的解决方案。本节首先介绍交替方向乘子法的基本算法；然后给出交替方向乘子法在矩阵分离实际问题中的应用。

问题引入

本节考虑如下凸问题：

$$\begin{aligned} \min_{\mathbf{x}_1, \mathbf{x}_2} \quad & f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) \\ \text{s.t.} \quad & \mathbf{A}_1 \mathbf{x}_1 + \mathbf{A}_2 \mathbf{x}_2 = \mathbf{b}, \end{aligned} \quad (12.80)$$

其中 f_1, f_2 是适当的闭凸函数，但不要求是光滑的， $\mathbf{x}_1 \in \mathbb{R}^n, \mathbf{x}_2 \in \mathbb{R}^m, \mathbf{A}_1 \in \mathbb{R}^{p \times n}, \mathbf{A}_2 \in \mathbb{R}^{p \times m}, \mathbf{b} \in \mathbb{R}^p$ 。这个问题的特点是目标函数可以分成彼此分离的两块，但是变量被线性约束结合在一起。常见的一些无约束和带约束的优化问题都可以表示成这一形式。下面的一些例子将展示如何把某些一般的优化问题转化为适用交替方向乘子法求解的标准形式。

例 12.3.6. 可以分成两块的无约束优化问题

$$\min_{\mathbf{x}} f_1(\mathbf{x}) + f_2(\mathbf{x}).$$

为了将此问题转化为标准形式 (12.80)，需要将目标函数改成可分的形式。我们可以通过引入一个新的变量 z 并令 $\mathbf{x} = z$ ，将问题转化为

$$\begin{aligned} \min_{\mathbf{x}, z} \quad & f_1(\mathbf{x}) + f_2(z), \\ \text{s.t.} \quad & \mathbf{x} - z = 0. \end{aligned}$$

例 12.3.7. 带线性变换的无约束优化问题

$$\min_{\mathbf{x}} f_1(\mathbf{x}) + f_2(\mathbf{A}\mathbf{x}).$$

类似地，我们可以引入一个新的变量 z ，令 $z = \mathbf{A}\mathbf{x}$ ，则问题变为

$$\begin{aligned} \min_{\mathbf{x}, z} \quad & f_1(\mathbf{x}) + f_2(z), \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} - z = 0 \end{aligned}$$

对比问题 (12.80) 可知 $\mathbf{A}_1 = \mathbf{A}$ 和 $\mathbf{A}_2 = -\mathbf{I}$ 。

算法结构

下面给出交替方向乘子法 (alternating direction method of multipliers, ADMM) 的迭代格式，首先写出问题 (12.80) 的增广拉格朗日函数

$$\begin{aligned} L_\rho(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}) = & f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) + \mathbf{y}^\top (\mathbf{A}_1 \mathbf{x}_1 + \mathbf{A}_2 \mathbf{x}_2 - \mathbf{b}) \\ & + \frac{\rho}{2} \|\mathbf{A}_1 \mathbf{x}_1 + \mathbf{A}_2 \mathbf{x}_2 - \mathbf{b}\|_2^2 \end{aligned}$$

其中 $\rho > 0$ 是二次罚项的系数. 常见的求解带约束问题的增广拉格朗日函数法为如下更新:

$$\begin{aligned} (\mathbf{x}_1^{k+1}, \mathbf{x}_2^{k+1}) &= \arg \min_{\mathbf{x}_1, \mathbf{x}_2} L_\rho(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}^k) \\ \mathbf{y}^{k+1} &= \mathbf{y}^k + \tau \rho (\mathbf{A}_1 \mathbf{x}_1^{k+1} + \mathbf{A}_2 \mathbf{x}_2^{k+1} - \mathbf{b}), \end{aligned}$$

其中 τ 为步长. 在实际求解中, 第一步迭代同时对 \mathbf{x}_1 和 \mathbf{x}_2 进行优化有时候比较困难, 而固定一个变量求解关于另一个变量的极小问题可能比较简单, 因此我们可以考虑对 \mathbf{x}_1 和 \mathbf{x}_2 交替求极小, 这就是交替方向乘子法的基本思路. 其迭代格式可以总结如下:

$$\begin{aligned} \mathbf{x}_1^{k+1} &= \arg \min_{\mathbf{x}_1} L_\rho(\mathbf{x}_1, \mathbf{x}_2^k, \mathbf{y}^k), \\ \mathbf{x}_2^{k+1} &= \arg \min_{\mathbf{x}_2} L_\rho(\mathbf{x}_1^{k+1}, \mathbf{x}_2, \mathbf{y}^k), \\ \mathbf{y}^{k+1} &= \mathbf{y}^k + \tau \rho (\mathbf{A}_1 \mathbf{x}_1^{k+1} + \mathbf{A}_2 \mathbf{x}_2^{k+1} - \mathbf{b}), \end{aligned}$$

其中 τ 为步长.

观察交替方向乘子法的迭代格式, 第一步固定 \mathbf{x}_2, \mathbf{y} 对 \mathbf{x}_1 求极小; 第二步固定 \mathbf{x}_1, \mathbf{y} 对 \mathbf{x}_2 求极小; 第三步更新拉格朗日乘子 \mathbf{y} . 与无约束优化问题不同, 交替方向乘子法针对的问题 (12.80) 是带约束的优化问题, 因此算法的收敛准则应当借助约束优化问题的最优性条件 (KKT 条件). 因为 f_1, f_2 均为闭凸函数, 约束为线性约束, 所以当 Slater 条件成立时, 可以使用凸优化问题的 KKT 条件来作为交替方向乘子法的收敛准则. 问题 (12.80) 的拉格朗日函数为

$$L(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}) = f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) + \mathbf{y}^T (\mathbf{A}_1 \mathbf{x}_1 + \mathbf{A}_2 \mathbf{x}_2 - \mathbf{b})$$

根据 KKT 条件, 若 $\mathbf{x}_1^*, \mathbf{x}_2^*$ 为问题 (12.80) 的最优解, \mathbf{y}^* 为对应的拉格朗日乘子, 则以下条件满足:

$$0 \in \partial_{\mathbf{x}_1} L(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{y}^*) = \partial f_1(\mathbf{x}_1^*) + \mathbf{A}_1^T \mathbf{y}^* \quad (12.81)$$

$$0 \in \partial_{\mathbf{x}_2} L(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{y}^*) = \partial f_2(\mathbf{x}_2^*) + \mathbf{A}_2^T \mathbf{y}^* \quad (12.82)$$

$$\mathbf{A}_1 \mathbf{x}_1^* + \mathbf{A}_2 \mathbf{x}_2^* = \mathbf{b} \quad (12.83)$$

在这里条件 (12.83) 又称为原始可行性条件, 条件 (12.81) 和条件 (12.82) 又称为对偶可行性条件. 由于问题中只含等式约束, KKT 条件中的互补松弛条件可以不加考虑. 在 ADMM 迭代中, 我们得到的迭代点实际为 $(\mathbf{x}_1^k, \mathbf{x}_2^k, \mathbf{y}^k)$, 因此收敛准则应当针对 $(\mathbf{x}_1^k, \mathbf{x}_2^k, \mathbf{y}^k)$ 检测条件 (12.81)-(12.83). 接下来讨论如何具体计算这些收敛准则.

一般来说, 原始可行性条件 (12.83) 在迭代中是不满足的, 为了检测这个条件, 需要计算原始可行性残差

$$r^k = \mathbf{A}_1 \mathbf{x}_1^k + \mathbf{A}_2 \mathbf{x}_2^k - \mathbf{b}$$

的模长, 这一计算是比较容易的.

现在来看两个对偶可行性条件. 考虑 ADMM 迭代更新 \mathbf{x}_2 的步骤

$$\mathbf{x}_2^k = \arg \min_{\mathbf{x}} \left\{ f_2(\mathbf{x}) + \frac{\rho}{2} \left\| \mathbf{A}_1 \mathbf{x}_1^k + \mathbf{A}_2 \mathbf{x} - \mathbf{b} + \frac{\mathbf{y}^{k-1}}{\rho} \right\|^2 \right\},$$

假设这一子问题有显式解或能够精确求解, 根据最优性条件不难推出

$$0 \in \partial f_2(\mathbf{x}_2^k) + \mathbf{A}_2^T [\mathbf{y}^{k-1} + \rho (\mathbf{A}_1 \mathbf{x}_1^k + \mathbf{A}_2 \mathbf{x}_2^k - \mathbf{b})]$$

注意到当 ADMM 步长 $\tau = 1$ 时, 根据迭代格式可知上式方括号中的表达式就是 \mathbf{y}^k , 最终我们有

$$0 \in \partial f_2(\mathbf{x}_2^k) + \mathbf{A}_2^T \mathbf{y}^k$$

这恰好就是条件 (12.82). 上面的分析说明在 ADMM 迭代过程中, 若 \mathbf{x}_2 的更新能取到精确解且步长 $\tau = 1$, 对偶可行性条件 (12.82) 是自然成立的, 因此无需针对条件 (12.82) 单独验证最优性条件.

然而, 在迭代过程中条件 (12.81) 却不能自然满足. 实际上, 由 \mathbf{x}_1 的更新公式

$$\mathbf{x}_1^k = \arg \min_{\mathbf{x}} \left\{ f_1(\mathbf{x}) + \frac{\rho}{2} \left\| \mathbf{A}_1 \mathbf{x} + \mathbf{A}_2 \mathbf{x}_2^{k-1} - \mathbf{b} + \frac{\mathbf{y}^{k-1}}{\rho} \right\|^2 \right\},$$

假设子问题能精确求解, 根据最优性条件

$$0 \in \partial f_1(\mathbf{x}_1^k) + \mathbf{A}_1^T [\rho (\mathbf{A}_1 \mathbf{x}_1^k + \mathbf{A}_2 \mathbf{x}_2^{k-1} - \mathbf{b}) + \mathbf{y}^{k-1}].$$

注意, 这里 \mathbf{x}_2 上标是 $k-1$, 因此根据 ADMM 的迭代第三式, 同样取 $\tau = 1$, 我们有

$$0 \in \partial f_1(\mathbf{x}_1^k) + \mathbf{A}_1^T (\mathbf{y}^k + \mathbf{A}_2 (\mathbf{x}_2^{k-1} - \mathbf{x}_2^k))$$

对比条件 (12.81) 可知多出来的项为 $\mathbf{A}_1^T \mathbf{A}_2 (\mathbf{x}_2^{k-1} - \mathbf{x}_2^k)$, 因此要检测对偶可行性只需要检测残差

$$s^k = \mathbf{A}_1^T \mathbf{A}_2 (\mathbf{x}_2^{k-1} - \mathbf{x}_2^k)$$

的模长是否充分小, 这一检测同样也是比较容易的. 综上, 当 \mathbf{x}_2 更新取到精确解且 $\tau = 1$ 时, 判断 ADMM 是否收敛只需要检测前述两个残差 r^k, s^k 是否充分小:

$$0 \approx \|r^k\| = \|\mathbf{A}_1 \mathbf{x}_1^k + \mathbf{A}_2 \mathbf{x}_2^k - \mathbf{b}\| \quad (\text{原始可行性}),$$

$$0 \approx \|s^k\| = \|\mathbf{A}_1^T \mathbf{A}_2 (\mathbf{x}_2^{k-1} - \mathbf{x}_2^k)\| \quad (\text{对偶可行性}).$$

应用实例: 矩阵分离问题

考虑矩阵分离问题:

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{S}} \quad & \|\mathbf{X}\|_* + \mu \|\mathbf{S}\|_1, \\ \text{s.t.} \quad & \mathbf{X} + \mathbf{S} = \mathbf{M}, \end{aligned}$$

其中 $\|\cdot\|_1$ 与 $\|\cdot\|_*$ 分别表示矩阵 ℓ_1 范数与核范数. 引入乘子 \mathbf{Y} 作用在约束 $\mathbf{X} + \mathbf{S} = \mathbf{M}$ 上, 我们可以得到此问题的增广拉格朗日函数

$$L_\rho(\mathbf{X}, \mathbf{S}, \mathbf{Y}) = \|\mathbf{X}\|_* + \mu \|\mathbf{S}\|_1 + \langle \mathbf{Y}, \mathbf{X} + \mathbf{S} - \mathbf{M} \rangle + \frac{\rho}{2} \|\mathbf{X} + \mathbf{S} - \mathbf{M}\|_F^2.$$

在第 $(k+1)$ 步, 交替方向乘子法分别求解关于 \mathbf{X} 和 \mathbf{S} 的子问题来更新得到 \mathbf{X}^{k+1} 和 \mathbf{S}^{k+1} . 对于 \mathbf{X} 子问题,

$$\begin{aligned}\mathbf{X}^{k+1} &= \arg \min_{\mathbf{X}} L_{\rho}(\mathbf{X}, \mathbf{S}^k, \mathbf{Y}^k) \\ &= \arg \min_{\mathbf{X}} \left\{ \|\mathbf{X}\|_* + \frac{\rho}{2} \left\| \mathbf{X} + \mathbf{S}^k - \mathbf{M} + \frac{\mathbf{Y}^k}{\rho} \right\|_F^2 \right\}, \\ &= \arg \min_{\mathbf{X}} \left\{ \frac{1}{\rho} \|\mathbf{X}\|_* + \frac{1}{2} \left\| \mathbf{X} + \mathbf{S}^k - \mathbf{M} + \frac{\mathbf{Y}^k}{\rho} \right\|_F^2 \right\}, \\ &= \mathbf{U} \text{Diag} \left(\text{prox}_{(1/\rho)\|\cdot\|_1}(\sigma(\mathbf{A})) \right) \mathbf{V}^T,\end{aligned}$$

其中 $\mathbf{A} = \mathbf{M} - \mathbf{S}^k - \frac{\mathbf{Y}^k}{\rho}$, $\sigma(\mathbf{A})$ 为 \mathbf{A} 的所有非零奇异值构成的向量并且 $\mathbf{U} \text{Diag}(\sigma(\mathbf{A})) \mathbf{V}^T$ 为 \mathbf{A} 的约化奇异值分解. 对于 \mathbf{S} 子问题,

$$\begin{aligned}\mathbf{S}^{k+1} &= \arg \min_{\mathbf{S}} L_{\rho}(\mathbf{X}^{k+1}, \mathbf{S}, \mathbf{Y}^k) \\ &= \arg \min_{\mathbf{S}} \left\{ \mu \|\mathbf{S}\|_1 + \frac{\rho}{2} \left\| \mathbf{X}^{k+1} + \mathbf{S} - \mathbf{M} + \frac{\mathbf{Y}^k}{\rho} \right\|_F^2 \right\} \\ &= \text{prox}_{(\mu/\rho)\|\cdot\|_1} \left(\mathbf{M} - \mathbf{X}^{k+1} - \frac{\mathbf{Y}^k}{\rho} \right).\end{aligned}$$

对于乘子 \mathbf{Y} , 依然使用常规更新, 即

$$\mathbf{Y}^{k+1} = \mathbf{Y}^k + \tau \rho \left(\mathbf{X}^{k+1} + \mathbf{S}^{k+1} - \mathbf{M} \right).$$

那么, 交替方向乘子法的迭代格式为

$$\begin{aligned}\mathbf{X}^{k+1} &= \mathbf{U} \text{Diag} \left(\text{prox}_{(1/\rho)\|\cdot\|_1}(\sigma(\mathbf{A})) \right) \mathbf{V}^T, \\ \mathbf{S}^{k+1} &= \text{prox}_{(\mu/\rho)\|\cdot\|_1} \left(\mathbf{M} - \mathbf{X}^{k+1} - \frac{\mathbf{Y}^k}{\rho} \right), \\ \mathbf{Y}^{k+1} &= \mathbf{Y}^k + \tau \rho \left(\mathbf{X}^{k+1} + \mathbf{S}^{k+1} - \mathbf{M} \right).\end{aligned}$$

值得说明的是, 在实际中, 大多数问题并不直接具有问题 (12.80) 的形式. 我们需要通过一系列拆分技巧将问题化成 ADMM 的标准形式, 同时要求每一个子问题尽量容易求解. 需要指出的是, 对同一个问题可能有多种拆分方式, 不同方式导出的最终算法可能差异巨大, 读者应当选择最容易求解的拆分方式.

12.4 深度学习常用优化算法

深度学习算法在许多情况下都涉及到优化算法, 但是用于深度模型训练的优化算法与传统的优化算法在几个方面有所不同. 在大多数机器学习问题中, 我们关注的点是在测试集上的不

可解的性能度量 P 。因此，我们只是间接地优化 P 。我们希望通过降低代价函数 $J(\theta)$ 来提高 P ，这一点不同于纯优化最小化 J 本身。

通常，在机器学习或深度学习中，通常利用经验风险最小化的原则，寻找最优的模型。即极小化如下代价函数

$$J(\theta) = \mathbb{E}_{(x,y) \sim \hat{P}_{data}} L(f(x; \theta), y) \quad (12.84)$$

其中， L 是每个样本的损失函数， $f(x; \theta)$ 是输入为 x 时所预测的输出， \hat{P}_{data} 是经验分布，监督学习中， y 是目标输出。在本节中，我们不讨论结构风险最小化的问题。

12.4.1 随机梯度下降

结合前一章的内容可知，在利用优化算法求解极小化代价函数时，最常用的目标函数的性质是梯度：

$$\nabla_{\theta} J(\theta) = \mathbb{E}_{(x,y) \sim \hat{P}_{data}} \nabla_{\theta} L(f(x; \theta), y) \quad (12.85)$$

在一阶优化或二阶优化算法时，均需要用到梯度信息。然而，由于机器学习，特别是深度学习中的数据集非常大。准确计算这个梯度的计算量非常大，因为需要考虑到整个数据集上的每个样本。每次迭代都需要很大的运算量，那么多次迭代所需的运算量更是让人无法接受。该如何解决这个问题呢？

随机梯度法

考虑到，梯度的计算实际上可以看作是求期望。根据期望的性质可知，任何一个样本都可看作是期望值的一个无偏估计。因此，考虑如下方式计算梯度：

$$\nabla_{\theta} J(\theta) = \nabla_{\theta} L(f(x^{(i)}; \theta), y^{(i)}) \quad (12.86)$$

这也被称为随机梯度，对应的优化算法被称为随机梯度下降法。

仔细审视随机梯度，可发现理论上确实是可行的。这是因为，首先， n 个样本均值的标准差是 σ/\sqrt{n} ，其中 σ 是样本值真实的标准差，分母 \sqrt{n} 。这表明使用更多样本来估计梯度的方法是低于线性的。换言之，一个基于 100 个样本，另一个基于 10,000 个样本，后者用于梯度计算的计算量是前者的 100 倍，但却只降低了 10 倍的均值标准差。如果能够快速计算出梯度估计值，而不是缓慢计算准确值，那么大多数优化算法会收敛地更快（就总的计算量而言，而不是指更新次数）。其次，从小数目样本中获得梯度的统计估计的动机是训练集的冗余性。在最坏情况下，训练集中所有 m 个样本可以是彼此相同的拷贝。基于采样的梯度估计可以使用单个样本计算出正确的梯度，而比原来的做法少花了 m 倍时间。实践中，尽管不太可能真的遇见这种最坏情况，但我们可能会发现大量样本确实都对梯度做出了非常相似的贡献。

在进一步介绍随机梯度法之前，有必要对涉及到的相关概念做一些区分。通常，使用整个训练集的梯度优化算法被称为 **batch** 或确定性梯度算法，简称梯度算法。每次从一个固定大小

的训练集中随机抽取单个样本的梯度优化算法被称为**随机梯度算法**。每次只使用单个样本的梯度优化算法有时被称为**随机梯度**或者**在线算法**。其中“**在线**”通常是指从连续产生的数据流中抽取样本的情况，而不是从一个固定大小的训练集中遍历多次采样的情况。大多数用于深度学习的算法介于梯度法和上面提到的随机梯度法两者之间，使用一个以上，而又不是全部的训练样本。传统上，这些会被称为**minibatch**随机梯度方法，通常也简单地称为随机梯度方法，这也是现如今被广泛运用的随机梯度法。

随机方法的典型示例是随机梯度下降(SGD)，其算法概括如下：

算法 12.18 随机梯度法

Require: 学习速率 ϵ_k ，初始参数 θ

- 1: **repeat**
- 2: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch，对应目标为 $\mathbf{y}^{(i)}$ ；
- 3: 计算梯度估计： $\hat{\mathbf{g}} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)})$
- 4: 应用更新： $\theta \leftarrow \theta - \epsilon_k \hat{\mathbf{g}}$
- 5: **until** 达到停止准则

学习速率

SGD 算法中的一个关键参数是学习速率(即前面提到的步长，在机器学习领域，人们常称之为学习速率)。在实践中，随着时间的推移有必要逐渐降低学习速率。逐步降低学习速率的原因是 SGD 在梯度估计引入的噪音(m 个训练样本的随机采样)并不会在极小值处消失。相比之下，当我们使用 batch 梯度下降到达极小值时，整个代价函数的真实梯度会变得很小，甚至为 0，因此 batch 梯度下降可以使用固定的学习速率。若假定在第 k 次迭代得学习速率我们记为 ϵ_k ，则保证 SGD 收敛的一个充分条件是

$$\sum_{k=1}^{\infty} \epsilon_k = \infty \quad (12.87)$$

且

$$\sum_{k=1}^{\infty} \epsilon_k^2 = \infty \quad (12.88)$$

实践中，一般会线性衰减学习速率到第 τ 次迭代：

$$\epsilon_k = (1 - \alpha) \epsilon_0 + \alpha \tau_r \quad (12.89)$$

其中， $\alpha = \frac{k}{\tau}$ 。在 τ 步迭代之后，一般使 ϵ 保持常数。

学习速率可通过试验和误差来选取，通常最好的选择方法是画出目标函数值随时间变化的学习曲线。使用线性时间表时，参数选择为 $\epsilon_0, \epsilon_{\tau}, \tau$ 。通常 τ 被设为需要反复遍历训练样本几百次的迭代次数，且设为大于 1% 的 ϵ_0 。所以主要问题是如何设置 ϵ_0 。若 ϵ_0 太大，学习曲线将会

剧烈振荡，代价函数值通常会明显增加。温和的振荡是良好的，特别是训练于随即代价函数上，例如由信号丢失引起的代价函数。如果学习速率太慢，那么学习进程会缓慢。如果初始学习速率太低，那么学习可能会卡在一个相当高的损失值。通常，就总训练时间和最终损失值而言，最优初始学习速率会高于大约迭代 100 步后输出最好效果的学习速率。因此，通常最好是检测最早的一次迭代，使用一个高于此时效果最佳学习速率的学习速率，但又不能太高以致严重的不稳定性。

SGD 和相关的 minibatch 或在线基于梯度的优化的最重要性质是每一步更新的计算时间不会随着训练样本数目而增加。对于大数据集，SGD 初始快速更新只需非常少量样本计算梯度的能力远远超过了其缓慢的渐进收敛。对于足够大的数据集，SGD 可能在处理整个训练集之前就收敛到最终测试集误差的某个固定容差范围内。SGD 应用于凸问题时， k 步迭代后的额外误差量级是 $O(\frac{1}{\sqrt{k}})$ ，在强凸情况下是 $O(\frac{1}{k})$ 。除非假定额外的条件，否则这些界限不能进一步改进。

batch 梯度下降在理论上比随机梯度下降有更好的收敛率，然而，Carmér 界限 (Carmér, 1946; Rao, 1945) 指出，泛化误差的下降速度不会快于 $O(\frac{1}{k})$ 。Bottou 和 Bousquet(2008) 由此认为对于机器学习任务，不值得探寻收敛快于 $O(\frac{1}{k})$ 的优化算法——更快的收敛可能对应着过拟合。此外，渐进分析掩盖了随机梯度下降在少量更新步之后的很多优点。我们也可以权衡 batch 梯度下降和随机梯度下降两者的优势，在学习过程中逐渐增大 minibatch 的大小。

12.4.2 动量梯度下降

虽然随机梯度下降仍然是非常受欢迎的优化方法，但学习速率有时会很慢。可能出现如图 12.26 所示“震荡”的现象。我们试图去分析出现“震荡”现象的最直接原因。在历史梯度的

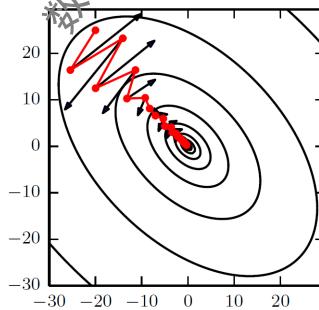


图 12.26

某些分量方向上，出现反复迂回的现象；而在一致前进的分量方向又没有得到加速，所以产生“震荡”。例如：假设第 $k-1$ 次迭代的梯度 $\nabla_{\theta} L(\theta_{k-1}) = [0.5 \ 0.2]^T$ ，而在第 k 次迭代的梯度 $\nabla_{\theta} L(\theta_k) = [-0.4 \ 0.1]^T$ 。显然，这两次迭代在第一维分量上产生震荡，第二维的分量方向行进一致。这促使我们思考：如何在震荡的分量上抵消震荡，在行进一致的分量上进行加速？

动量法

一个简单的方法：将历史数据与当前的梯度进行某种加法，那么自然会抵消震荡部分的分量，并且加速行进一致的分量。这就是动量法的思想，它积累了之前梯度指数级衰减的移动平均，并且继续沿该方向移动。这便得到动量法的更新规则如下：

$$\begin{cases} \mathbf{v} \leftarrow \alpha \mathbf{v} - \epsilon \nabla_{\theta} \left(\frac{1}{m} \sum_{i=1}^m L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)}) \right) \\ \theta \leftarrow \theta + \mathbf{v} \end{cases} \quad (12.90)$$

其中第一个式子的第一项可以看作是历史梯度信息，第一个式子的第二项为当前的负梯度。然后将它们的加权和作为当前下降的方向。动量法本质上解决了两个问题：Hessian 矩阵的不良条件数和随机梯度的方差。总体上，避免了将计算浪费在来回的震荡之中。下面将带动量的 SGD 算法总结如下：

算法 12.19 动量法

Require: 学习速率 ϵ ，动量参数 α ，初始参数 θ ，初始速度 \mathbf{v}

- 1: **repeat**
- 2: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch，对应目标为 $\mathbf{y}^{(i)}$ ；
- 3: 计算梯度估计： $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)})$
- 4: 计算速度更新： $\mathbf{v} \leftarrow \alpha \mathbf{v} - \epsilon \mathbf{g}$
- 5: 应用更新： $\theta \leftarrow \theta + \mathbf{v}$
- 6: **until** 达到停止准则

动量法有着物理学上的直观含义。从形式上看，动量算法引入了变量 \mathbf{v} 充当速度的角色——它代表参数在参数空间移动的方向和速度。名称动量 (momentum) 来自物理类比，根据牛顿运动定律，负梯度是移动参数空间中粒子的力。动量在物理学上是质量乘以速度。在动量学习算法中，我们假设是单位质量，因此速度向量 \mathbf{v} 也可以看作是粒子的动量。实际上动量（速度）被设为历史负梯度信息的指数衰减平均，其中超参数 $\alpha \in [0, 1)$ 决定了之前梯度的贡献衰减得有多快。这可以通过简单的展开得到：

$$\mathbf{v}_k = \alpha \mathbf{v}_{k-1} - \epsilon \mathbf{g}_k = \alpha^2 \mathbf{v}_{k-2} - \alpha \epsilon \mathbf{g}_{k-1} - \epsilon \mathbf{g}_k = \dots$$

对动量法做一个最理想的假设：如果动量动量算法总是观测到梯度 \mathbf{g} ，那么它会在方向 $-\mathbf{g}$ 上不停加速，直到达到最后速度的步长为

$$\frac{\epsilon \|\mathbf{g}\|}{1 - \alpha} \quad (12.91)$$

因此，将动量的超参数视为 $\frac{1}{1-\alpha}$ 。

Nesterov 动量法

动量法中使用历史的梯度数据来改进搜索方向, 那能不能使用未来的梯度数据来改进当前搜索方向呢? 实际上, 这就是 Nesterov 加速梯度算法。假设我们试探性的向前迈进了一步, 然后计算下一步的梯度, 再利用它改善当前的前进方向。

受 Nesterov 加速梯度算法启发, Sutskever(2013) 提出了动量算法的一个变种。这种情况的更新规则如下:

$$\begin{cases} \mathbf{v} \leftarrow \alpha \mathbf{v} - \epsilon \nabla_{\boldsymbol{\theta}} \left[\frac{1}{m} \sum_{i=1}^m L \left(\mathbf{f}(\mathbf{x}^{(i)}; \boldsymbol{\theta} + \alpha \mathbf{v}), \mathbf{y}^{(i)} \right) \right] \\ \boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mathbf{v} \end{cases} \quad (12.92)$$

其中参数 α 和 ϵ 发挥了和标准动量方法中类似的作用。Nesterov 动量和标准动量之间的区别体现在梯度计算上。Nesterov 动量中, 梯度计算在施加当前速度之后。 $\boldsymbol{\theta} + \alpha \mathbf{v}$ 即表示试探性的下一步, 那么第一个式子的第二项即为下一步的梯度。Nesterov 动量的随机梯度下降算法如下算法 12.20。

算法 12.20 Nesterov 动量法

Require: 学习速率 ϵ , 动量参数 α , 初始参数 $\boldsymbol{\theta}$, 初始速度 \mathbf{v}

- 1: **repeat**
- 2: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch, 对应目标为 $\mathbf{y}^{(i)}$;
- 3: 应用临时更新: $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta} + \alpha \mathbf{v}$
- 4: 计算梯度 (在临时点): $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\tilde{\boldsymbol{\theta}}} \sum_{i=1}^m L \left(\mathbf{f}(\mathbf{x}^{(i)}; \tilde{\boldsymbol{\theta}}), \mathbf{y}^{(i)} \right)$
- 5: 计算速度更新: $\mathbf{v} \leftarrow \alpha \mathbf{v} - \epsilon \mathbf{g}$
- 6: 应用更新: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mathbf{v}$
- 7: **until** 达到停止准则

在凸 batch 梯度的情况下, Nesterov 动量将额外误差收敛率从 $O(1/k)$ (k 步后) 改进到 $O(1/k^2)$ 。可惜, 在随机梯度的情况下, Nesterov 动量没有改进收敛率。

12.4.3 自适应学习速率

前面讨论的是如何优化搜索方向, 实际上还有一个关键的问题是优化学习速率。神经网络研究员早就意识到学习速率肯定是以设置的超参数之一, 因为它对模型的性能有显著的影响。通常, 损失函数高度敏感于参数空间中的某些方向, 动量算法虽然可以在一定程度缓解这些问题, 但这样做的代价是引入了另一个超参数。在这种情况下, 自然会问有没有其他方法。如果我们相信方向敏感度在某种程度是轴对齐的, 那么每个参数设置不同的学习速率, 在整个学习过程中自动适应这些学习速率是有道理的。

事实上, **Delta-bar-delta** 算法 (Jacobs, 1988) 是一个早期的在训练时适应模型参数各自学习速率的启发式方法。该方法基于一个很简单的想法: 如果损失对于某个给定模型参数的偏导保持相同的符号, 那么学习速率应该增加; 如果对于该参数的偏导变化了符号, 那么学习速率应减小。当然, 这种方法只能应用于全 batch 优化中。而对于基于 minibatch 的算法是否有相应的启发式学习速率? 最近, 研究者们提出了一些增量 (或者基于 minibatch) 的算法来自适应模型参数的学习速率。这节将简要回顾其中一些算法。

AdaGrad

我们再次回到下图12.27所示“震荡”的现象。这次不从搜索方向的角度思考, 而从学习速

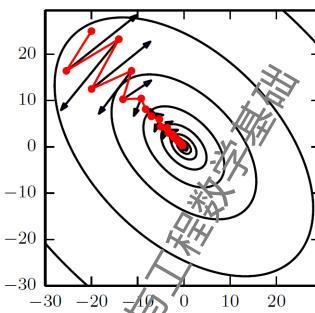


图 12.27

率的角度思考, 直观地发现: 在左下至右上的维度前进的步伐较大, 导致了反复迂回; 在左上至右下的维度前进的步伐较小, 导致了下降不够。实际上, 我们可以从梯度的分量值上获得这一直观认识。当梯度的分量较大时, 意味着该方向函数值变化快, 应减小学习率; 当梯度的分量较小时, 函数值变化平缓, 应增大步长。因此, 可以根据梯度历史值自适应地更新学习速率。这就是 **AdaGrad** 算法: 按照每个参数的梯度历史值的平方和的平方根成反比缩放每个参数 (Duchi, 2011), 独立地适应所有模型参数的学习速率。这样具有最大偏导的参数, 自适应地降低了学习速率, 而具有小偏导的参数在学习速率上应相对变大。AdaGrad 算法如算法12.21所示。

在凸优化背景中, AdaGrad 算法具有一些令人满意的理论性质。然而, 经验上已经发现, 对于训练深度神经网络模型而言, 从训练开始时积累梯度平方会导致有效学习速率过早和过量的减小。AdaGrad 在某些深度学习模型上效果不错, 但不是全部。

如果在 AdaGrad 中使用真实梯度 $\nabla f(x^k)$, 那么 AdaGrad 也可以看成是一种介于一阶和二阶之间的优化算法。考虑 $f(x)$ 在点 x^k 处的二阶泰勒展开:

$$f(x) \approx f(x^k) + \nabla f(x^k)^T (x - x^k) + \frac{1}{2} (x - x^k)^T B^k (x - x^k)$$

若使用常数倍单位矩阵近似 B^k 时可得到梯度法; 若利用海瑟矩阵作为 B^k 时可得到牛顿法。而

算法 12.21 AdaGrad

Require: 全局学习速率 ϵ , 初始参数 θ , 小常数 δ (为了数值稳定大约设为 10^{-7})

- 1: 初始化梯度累计变量 $r = \mathbf{0}$;
- 2: **repeat**
- 3: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch, 对应目标为 $\mathbf{y}^{(i)}$;
- 4: 计算梯度: $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)})$
- 5: 累积平方梯度: $r \leftarrow r + \mathbf{g} \odot \mathbf{g}$ (\odot 表示逐元素相乘)
- 6: 计算更新: $\Delta\theta \leftarrow -\frac{\epsilon}{\delta + \sqrt{r}} \odot \mathbf{g}$ (逐元素地应用除和求平方根)
- 7: 应用更新: $\theta \leftarrow \theta + \Delta\theta$
- 8: **until** 达到停止准则

AdaGrad 则是使用一个对角矩阵来作为 B^k 。具体地, 取

$$B^k = \frac{1}{\epsilon} \text{Diag}(\sqrt{r} + \delta)$$

时导出的算法就是 AdaGrad。

RMSProp

在了解动量法之后, 我们会发现 AdaGrad 算法存在明显不足: 将所有的历史梯度值的平方直接相加, 一方面可能使得学习速率在达到这样的凸结构前就变得太小了。另一方面, 对于遥远过去的历史数据没有衰减。借助动量法的思想, **RMSProp** 算法 (Hinton, 2012) 修改 AdaGrad 算法, 改变梯度积累为指数加权的移动平均:

$$r \leftarrow \rho r + (1 - \rho) \mathbf{g} \odot \mathbf{g}.$$

AdaGrad 旨在应用于凸问题时快速收敛。当应用于非凸函数训练神经网络时, 学习轨迹可能穿过了很多不同的结构, 最终到达一个局部是凸的碗状的区域。RMSProp 使用指数衰减平均以丢弃遥远过去的历史, 使其能够在找到碗状凸结构后快速收敛, 它就像一个初始化于该碗状结构的 AdaGrad 算法实例。

RMSProp 的标准形式如算法 12.22, 相比于 AdaGrad, 使用移动平均引入了一个新的超参数 ρ , 用来控制移动平均的长度范围。经验上, RMSProp 已被证明是一种有效且实用的深度神经网络优化算法。目前它是深度学习从业者经常采用的优化方法之一。

Adam

在前面我们探讨了搜索方向上的优化, 得到了动量法; 其次, 我们通过学习速率上的优化, 获得了 AdaGrad 算法、RMSProp 算法。显然, 可以将二者结合起来, 即接下来要介绍的 Adam 算法。它可看作是结合了 RMSProp 和动量法的变种。

算法 12.22 RMSProp

Require: 全局学习速率 ϵ , 衰减速率 ρ , 初始参数 θ , 小常数 δ (为了数值稳定大约设为 10^{-6})

- 1: 初始化梯度累计变量 $r = 0$;
- 2: **repeat**
- 3: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch, 对应目标为 $\mathbf{y}^{(i)}$;
- 4: 计算梯度: $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)})$
- 5: 累积平方梯度: $r \leftarrow \rho r + (1 - \rho) \mathbf{g} \odot \mathbf{g}$
- 6: 计算更新: $\Delta\theta \leftarrow -\frac{\epsilon}{\delta + \sqrt{r}} \odot \mathbf{g}$
- 7: 应用更新: $\theta \leftarrow \theta + \Delta\theta$
- 8: **until** 达到停止准则

Adam(kingma and Ba, 2014) 是另一种学习速率自适应的优化算法。“Adam”这个名字派生自短语“adaptive moments”。首先, 在 Adam 中, 动量直接并入了梯度一阶矩(指数加权)的估计。将动量加入 RMSProp 最直观的方法是将动量应用于缩放后的梯度:

$$\mathbf{s} \leftarrow \rho_1 \mathbf{s} + (1 - \rho_1) \mathbf{g},$$

结合缩放的动量使用没有明确的理论动机。其次, 与 RMSProp 类似, 也会记录迭代过程中梯度的二阶矩:

$$\mathbf{r} \leftarrow \rho_2 \mathbf{r} + (1 - \rho_2) \mathbf{g} \odot \mathbf{g}.$$

再次, Adam 包括偏置修正, 修正从原点初始化的一阶矩和(非中心的)二阶矩的估计:

$$\hat{\mathbf{s}} \leftarrow \frac{\mathbf{s}}{1 - \rho_1^t}, \hat{\mathbf{r}} \leftarrow \frac{\mathbf{r}}{1 - \rho_2^t}.$$

Adam 算法具体步骤如算法 12.23 所示。

从算法中可以看出, RMSProp 也采用了(非中心的)二阶矩估计, 然而缺失了修正因子。因此, 不像 Adam, RMSProp 二阶矩估计可能在训练初期有很高的偏置。Adam 通常被认为对超参数的选择相当鲁棒, 尽管学习速率有时需要改为与建议的默认值不同的值。Adam 也是深度学习从业者经常采用的优化方法之一。它已经被实现在许多主流的深度学习框架之中, 包括 Pytorch 和 Tensorflow。

12.4.4 应用实例: 多层感知机

多层感知机也叫全连接神经网络, 是一种基本的网络结构, 在前面已经简要地介绍了这一模型。考虑有 L 个隐藏层的多层感知机, 给定输入 $\mathbf{x} \in \mathbb{R}^p$, 则多层感知机的输出可用如下迭代过程表示:

$$\mathbf{f}^{(l)} = \sigma \left(\mathbf{A}^{(l)} \mathbf{f}^{(l-1)} + \mathbf{b}^{(l)} \right), \quad l = 1, 2, \dots, L + 1$$

算法 12.23 Adam

Require: 步长 ϵ (建议默认为 0.001), 矩估计的指数衰减速率 ρ_1, ρ_2 (建议分别默认为 0.9 和 0.999), 用于数值稳定的小常数 δ (建议默认为 10^{-8}), 初始参数 θ

- 1: 初始化一阶和二阶矩变量 $s = \mathbf{0}, r = \mathbf{0}, t = 0$
- 2: **repeat**
- 3: 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的 minibatch, 对应目标为 $\mathbf{y}^{(i)}$;
- 4: 计算梯度: $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), \mathbf{y}^{(i)})$
- 5: $t \leftarrow t + 1$
- 6: 更新有偏一阶矩估计: $s \leftarrow \rho_1 s + (1 - \rho_1) \mathbf{g}$
- 7: 更新有偏二阶矩估计: $r \leftarrow \rho_2 r + (1 - \rho_2) \mathbf{g} \odot \mathbf{g}$
- 8: 修正一阶矩和二阶矩的偏差: $\hat{s} \leftarrow \frac{s}{1 - \rho_1^t}, \hat{r} \leftarrow \frac{r}{1 - \rho_2^t}$
- 9: 计算更新: $\Delta\theta \leftarrow -\epsilon \frac{\hat{s}}{\sqrt{\hat{r}} + \delta}$
- 10: 应用更新: $\theta \leftarrow \theta + \Delta\theta$
- 11: **until** 达到停止准则

其中 $\mathbf{A}^{(l)} \in \mathbb{R}^{m_{l-1} \times m_l}$ 为系数矩阵, $\mathbf{b}^{(l)} \in \mathbb{R}^{m_l}$ 为非齐次项, $\sigma(\cdot)$ 为非线性激活函数. 该感知机的输出为 $\mathbf{f}^{(L+1)}$.

现在用非线性函数 $h(\mathbf{x}; \mathbf{A})$ 来表示该多层感知机, 其中

$$\mathbf{A} = \left(\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(L)}, \mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \dots, \mathbf{b}^{(L)} \right)$$

表示所有网络参数的集合, 则学习问题可以表示成经验损失函数求极小问题:

$$\min \frac{1}{N} \sum_{i=1}^N L(h(\mathbf{x}_i; \mathbf{A}), \mathbf{y}_i).$$

同样地, 由于目标函数表示成了样本平均的形式, 我们可以用随机梯度算法:

$$\mathbf{A}^{k+1} = \mathbf{A}^k - \tau_k \nabla_{\mathbf{A}} L\left(h\left(\mathbf{x}_{s_k}; \mathbf{A}^k\right), \mathbf{y}_{s_k}\right),$$

其中 s_k 为从 $\{1, 2, \dots, N\}$ 中随机抽取的一个样本. 算法最核心的部分为求梯度, 由于函数具有复合结构, 因此可以采用后传算法. 假定已经得到关于第 l 隐藏层的导数 $\frac{\partial L}{\partial \mathbf{f}^{(l)}}$, 然后可以通过下面递推公式得到关于第 l 隐藏层参数的导数以及关于前一个隐藏层的导数:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{b}^{(l)}} &= \frac{\partial L}{\partial \mathbf{f}^{(l)}} \odot \frac{\partial \sigma}{\partial \mathbf{z}} \\ \frac{\partial L}{\partial \mathbf{A}^{(l)}} &= \left(\frac{\partial L}{\partial \mathbf{f}^{(l)}} \odot \frac{\partial \sigma}{\partial \mathbf{z}} \right) \left(\mathbf{f}^{(l-1)} \right)^T, \\ \frac{\partial L}{\partial \mathbf{f}^{(l-1)}} &= \left(\mathbf{A}^{(l)} \right)^T \left(\frac{\partial L}{\partial \mathbf{f}^{(l)}} \odot \frac{\partial \sigma}{\partial \mathbf{z}} \right). \end{aligned}$$

其中 \odot 为逐元素相乘. 完整的后传算法见算法 12.24.

算法 12.24 后传算法

```
1:  $\mathbf{g} \leftarrow \nabla_{\hat{\mathbf{f}}} L(\hat{\mathbf{f}}, \mathbf{y}_{s_k}).$ 
2: for  $l = L + 1, L, \dots, 1$  do
3:    $\mathbf{g} \leftarrow \mathbf{g} \odot \frac{\partial \sigma}{\partial z}.$ 
4:    $\frac{\partial L}{\partial \mathbf{b}^{(l)}} = \mathbf{g}.$ 
5:    $\frac{\partial L}{\partial \mathbf{A}^{(l)}} = \mathbf{g} \left( \mathbf{f}^{(l-1)} \right)^T.$ 
6:    $\mathbf{g} \leftarrow \left( \mathbf{A}^{(l)} \right)^T \mathbf{g}.$ 
7: end for
```

12.5 在线凸优化算法简介

将优化视为一个过程的观点已经在各种领域中都变得非常突出，并在建模和系统方面取得了惊人的成功，现在已经成为我们日常生活的一部分。随着机器学习的发展，在线机器学习也成为一个热门的研究方向。而其背后正是近年来许多学者所提出的在线凸优化的框架。本节我们将对这一前沿领域做简要介绍，包括在线凸优化模型的建立、实际应用（在线投资组合选择）以及求解算法。

12.5.1 在线凸优化模型

在在线凸优化（Online Convex Optimization, OCO）问题中，一个在线参与者（玩家）迭代式的做出决策。在做出每一个决策时，与他的选择相关的结果对参与者来说是未知的。在做出决策后，决策者会付出一个代价。每一个可能的决策都会付出一个（可能是不同的）代价，这些代价是决策者无法提前预知的，可以由对手选择，甚至取决于决策者自身采取的行动。

为使得这个架构有意义，定义一些约束是非常有必要的：

- 对手给出的代价不允许是无界的。否则对手可以在每一步中不断降低代价的值，使得算法永远不能从第一次支付代价后恢复。因此代价被假定为局限在某一个有界范围内。
- 尽管决策集中元素的个数不必是有限的，但它必须是有界的，且/或是有结构的。为理解这一规定的必要性，可以考虑在一个无穷可能决策集上的决策问题。对手可以在参与者选择的所有策略上不固定地附加较高的代价，并令其他策略的代价为零。这就使得任何有意义的性能指标无法使用。

令人惊讶的是，在不超过这两个约束的情况下，可以导出一些有趣的结论和算法。在在线凸优化架构模型中，决策集被模型化为欧式空间中的一个凸集，记为 $\mathcal{K} \subset \mathbb{R}^n$ 。代价被模型化为 \mathcal{K} 上的有界凸函数。OCO 架构可以看作一个有结构的、不断重复的博弈过程。

这一学习架构的规则为：在第 t 次迭代时，在线参与者选择 $x_t \in \mathcal{K}$ 。在参与者做出这一选择后，给出一个凸函数 $f_t \in \mathcal{F} : \mathcal{K} \mapsto \mathbb{R}$ 。此处 \mathcal{F} 为对手可以使用的有界代价函数族。在线参与者付出的代价为 $f_t(x_t)$ ，即选择 x_t 时代价函数的值。令 T 表示博弈进行的总迭代次数。

一个自然地问题，是什么使得一个算法成为好的 OCO 算法呢？由于该架构为一个博弈过程，它天然具有对抗性，其合理的性能评估指标也将来自于博弈论：决策者的遗憾（regret）。它定义为在事后看来，决策者做出决策所付出的总代价与固定的最好决策总代价之间的差。在 OCO 中，通常对一个算法在最坏的情况下做出决策的遗憾上界感兴趣。

令 \mathcal{A} 为一个 OCO 算法，它将某特定博弈中的历史决策映射到决策集中。经 T 次的迭代后， \mathcal{A} 的遗憾的形式化定义为：

$$\text{遗憾}_T(\mathcal{A}) = \sup_{f_1, \dots, f_T \subseteq \mathcal{F}} \left\{ \sum_{t=1}^T f_t(x_t) - \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x}) \right\} \quad (12.93)$$

直观地讲，如果一个算法的遗憾是 T 的次线性函数，即 $\text{遗憾}_T(\mathcal{A}) = o(T)$ ，则该算法的性能较好，因为这意味着算法的平均性能在事后看来与最好的固定策略是一样的。

在一个 T 迭代重复博弈中，在线优化算法的执行时间定义为在迭代 $t \in [T]$ ¹ 时，最坏情形下得到 x_t 所需的时间。通常，执行时间会依赖于 n （决策集 \mathcal{K} 的维数）、 T （博弈迭代的总次数）、代价函数的参数及基本凸集。

OCO 建模实例：在线投资组合

OCO 在最近几年成为在线学习的主要架构的原因得益于它极强的建模能力，使得它被广泛地应用于诸多领域，例如：在线路由，广告选择，垃圾邮件过滤和投资组合选择等。

这里详细介绍最近被广泛研究的投资组合选择问题，考虑一个对股票市场不做任何统计性假设（用以区别传统的几何布朗运动股票价格模型）的投资组合选择模型，并将其称为“通用投资组合选择”（universal portfolio selection）模型。

在每一迭代 $t \in [T]$ 中，决策者在 n 种资产上选择其财富的一个分布 $\mathbf{x}_t \in \Delta_n$ ，此处 $\Delta_n = \{\mathbf{x} \in \mathbb{R}_+^n, \sum_i = 1\}$ 为 n 维单纯形。对手独立地选择资产的回报，即一个所有元素都严格为正的向量 $\mathbf{r}_t \in \mathbb{R}^n$ ，其每一个分量 $r_t(i)$ 表示资产在迭代 t 和 $t+1$ 之间的价格的比值。例如，若第 i 个分量为一个 Google 股票持有者用 GOOG 标记的 NASDAQ 交易量，则：

$$r_t(i) = \frac{\text{在 } t+1 \text{ 时刻 GOOG 的价格}}{\text{在 } t \text{ 时刻 GOOG 的价格}}$$

令 W_t 为在迭代 t 时的总财富，则在忽略交易成本的情况下，有

$$W_{t+1} = W_t \cdot \mathbf{r}_t^T \mathbf{x}_t$$

投资者在迭代 $t+1$ 和 t 时的财富的比值为 $\mathbf{r}_t^T \mathbf{x}_t$ 。这样经过 T 次迭代后，总资产财富可用下式给出：

$$W_T = W_1 \cdot \prod_{t=1}^T \mathbf{r}_t^T \mathbf{x}_t$$

¹在此以后，符号 $[n]$ 表示整数集合 $\{1, \dots, n\}$ 。

决策者的目标是最大化整体的财富收益 W_T/W_1 ，它可以通过最大化下面的对数值更为方便的求得：

$$\log \frac{W_T}{W_1} = \sum_{t=1}^T \log \mathbf{r}_t^T \mathbf{x}_t$$

尽管它被标示为收益最大化而不是代价最小化，但这并没有本质区别。这一设定下的收益就定义为该财富比例变化的对数，即

$$f_t(\mathbf{x}) = \log(\mathbf{r}_t^T \mathbf{x}).$$

注意到由于 \mathbf{x}_t 为投资者财富的分布，即便有 $\mathbf{x}_{t+1} = \mathbf{x}_t$ ，由于价格的变化，投资者仍然需要通过交易来调整资产。

在这种情况下，遗憾被定义为：

$$\text{遗憾}_T = \max_{\mathbf{x}^* \in \Delta_n} \sum_{t=1}^T \log(\mathbf{r}_t^T \mathbf{x}^*) - \sum_{t=1}^T \log(\mathbf{r}_t^T \mathbf{x}_t)$$

即

$$\text{遗憾}_T = \max_{\mathbf{x}^* \in \Delta_n} \sum_{t=1}^T f_t(\mathbf{x}^*) - \sum_{t=1}^T f_t(\mathbf{x}_t)$$

这是非常直观的。公式中的第一项是财富的对数，它是由尽可能好的事后分布 \mathbf{x}^* 所积累的。由于这一分布是固定的，它对于每一个交易期后重新平衡头寸的策略，因此，它被称为持续的再平衡投资组合（constant rebalanced portfolio）。第二项就是在在线决策者积累财富的对数。因此最小化遗憾就对应于最大化投资者的财富在一个投资策略集中表现最佳基准财富的比值。

在这一设定下，通用投资者组合选择非常符合 OCO 架构。一般地，通用投资者组合选择算法被定义为求得的遗憾收敛于零的算法。尽管这一算法需要使用指数次的计算时间，但它最先是由 Cover 提出的。

12.5.2 一阶方法

本小节将描述并分析在线凸优化中的一个最简单也最基本的算法，它在实践中也非常有效。本小节中介绍的算法的目标都是最小化遗憾（regret），而不是优化误差（在在线假设下，它是病态的）。

回顾等式 (12.93) 中给出的 OCO 设定下有关遗憾的定义，将其中在上标、下标和上确界中有关函数类的记号去掉后，就得到比较清晰的形式：

$$\text{遗憾} = \sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x})$$

表 12.3 中详细给出了不同类型凸函数遗憾的上界和下界，它们是依赖于预测迭代数量的。

表 12.3: 可以达到的渐进遗憾界

	α 强凸	β 光滑
上界	$\frac{1}{\alpha} \log T$	\sqrt{T}
下界	$\frac{1}{\alpha} \log T$	\sqrt{T}
平均遗憾	$\frac{\log T}{\alpha T}$	$\frac{1}{\sqrt{T}}$

为将遗憾与优化误差进行对比, 考虑平均遗憾(即遗憾/T)是非常有用的。令 $\bar{\mathbf{x}}_T = \frac{1}{T} \sum_{t=1}^T \mathbf{x}_t$ 为决策向量的平均, 若函数 f_t 都是同样的一个函数 $f : \mathcal{K} \mapsto \mathbb{R}$, 则 Jensen 不等式意味着 $f(\bar{\mathbf{x}}_T)$ 收敛于 $f(\mathbf{x}^\star)$ 的速率最多为平均遗憾, 因为

$$f(\bar{\mathbf{x}}_T) - f(\mathbf{x}^\star) \leq \frac{1}{T} \sum_{t=1}^T [f(\mathbf{x}_t) - f(\mathbf{x}^\star)] = \frac{\text{遗憾}}{T}$$

下面将介绍实现上述 OCO 结果的算法和下界。

在线梯度下降法

也许应用于多数一般设定的在线凸优化问题的最简单算法是在线梯度下降法。这一算法是基于离线优化中最基本的标准梯度下降法的, 它首次由 zinkevich 引入到在线形式中。

该算法的伪代码在算法 12.25 中给出。在每一次迭代中, 算法首先在上一次迭代的基础上、沿上一次代价函数的负梯度方向移动一步步长。这一步可能得到一个不在基本凸集中的点。此时, 算法将该点投影回凸集, 即求凸集中与这一个点最接近的点。尽管下一个代价函数的值与当前得到的代价函数的值可能完全不同, 但该算法得到的遗憾是次线性的。这一结论可形式化整理为下面的定理。

算法 12.25 在线梯度下降法 (OGD)

- 1: 输入: 凸集 $\mathcal{K}, T, \mathbf{x}_1 \in \mathcal{K}$, 步长序列 $\{\eta_t\}$
- 2: **for** $t = 1$ 到 T **do**
- 3: 执行 \mathbf{x}_t 并考查代价函数 $f_t(\mathbf{x}_t)$.
- 4: 更新及投影:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)$$

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{K}}(\mathbf{y}_{t+1})$$

- 5: **end for**

定理 12.5.1. 步长大小为 $\{\eta_t = \frac{D}{G\sqrt{t}}, t \in [T]\}$ 的在线梯度下降算法保证对所有的 $T \geq 1$ 有如下结论：

$$\text{遗憾}_T = \sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x}^* \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x}^*) \leq \frac{3}{2} GD\sqrt{T}$$

这里 D 表示凸集 \mathcal{K} 的上界， G 表示函数 f_t 在凸集 \mathcal{K} 上的次梯度的范数上界。

证明. 令 $\mathbf{x}^* \in \arg \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x})$ 。记 $\nabla_t \triangleq \nabla f_t(\mathbf{x}_t)$ 。由凸性可得

$$f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*) \leq \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) \quad (12.94)$$

首先，利用 \mathbf{x}_{t+1} 的更新规则，可给出 $\nabla_t^T (\mathbf{x}_t - \mathbf{x}^*)$ 的上界：

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 = \|\Pi_{\mathcal{K}}(\mathbf{x}_t - \eta_t \nabla_t) - \mathbf{x}^*\|^2 \leq \|\mathbf{x}_t - \eta_t \nabla_t - \mathbf{x}^*\|^2 \quad (12.95)$$

于是，

$$\begin{aligned} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 &\leq \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \eta_t^2 \|\nabla_t\|^2 - 2\eta_t \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) \\ 2\nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) &\leq \frac{\|\mathbf{x}_t - \mathbf{x}^*\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2}{\eta_t} + \eta_t G^2 \end{aligned} \quad (12.96)$$

将式 (12.94) 和式 (12.96) 对 $t = 1$ 到 T 求和，并令 $\eta_0 = \frac{D}{G\sqrt{t}}$ (其中 $\frac{1}{\eta_0} \triangleq 0$)：

$$\begin{aligned} 2\left(\sum_{t=1}^T f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)\right) &\leq 2 \sum_{t=1}^T \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) \\ &\leq \sum_{t=1}^T \frac{\|\mathbf{x}_t - \mathbf{x}^*\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2}{\eta_t} + G^2 \sum_{t=1}^T \eta_t \\ &\leq \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}^*\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^T \eta_t \\ &\leq D^2 \sum_{t=1}^T \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^T \eta_t \\ &\leq D^2 \frac{1}{\eta_T} + G^2 \sum_{t=1}^T \eta_t \\ &\leq 3DG\sqrt{T} \end{aligned} \quad (12.97)$$

得到最后一个不等式的原因是 $\eta_t = \frac{D}{G\sqrt{t}}$ 和 $\sum_{t=1}^T \frac{1}{\sqrt{t}} \leq 2\sqrt{T}$ 。 \square

在线梯度下降算法是很容易实现的，在更新过程中得到的梯度只需使用线性时间。但是，投影步有可能需要非常长的时间。

下界

前面章节中引入并分析了一个非常简单且自然的在线凸优化问题。在继续研究之前一个值得考虑的问题是前述的界是否可以改进？度量 OCO 算法的性能可以同时使用遗憾和计算效率。因此，需要自问是否存在更简单的算法达到更紧的遗憾界。

在线梯度下降法的计算效率看起来改进的余地不大，在不考虑每一步迭代中使用的投影算法时，它的每次迭代都是线性时间复杂度。那么是否能得到更好的遗憾值呢？

也许令人惊讶，这一问题的答案是否定的：在最坏情形下，在线梯度下降法在相差一个小常数因子的意义下达到了紧遗憾界！这一结果可形式化地在下面定理中给出。

定理 12.5.2. 在最坏的情形下，任何在线凸优化算法的遗憾都是 $\Omega(DG\sqrt{T})$ 。即使其代价函数是由固定平稳分布得到的，这一结果仍然成立。

12.5.3 二阶方法

到目前为止，我们仅考虑了最小化遗憾的一阶方法。本节考虑一个拟牛顿结果，即一个在线凸优化算法，该算法估计了二阶导数，或在超过一维时，估计了其黑塞矩阵。但严格地讲，此处分析的算法仍然是一阶算法，因为它仅使用了梯度的信息。

此处引入并分析的算法称为在线牛顿步 (online Newton step) 算法，其细节参见算法 12.26。在每一次迭代时，这一算法选择一个向量，该向量是前面各步迭代中使用的向量与一个附加向量之和的投影向量。对在线梯度下降算法，这一附加的向量是前一个代价函数的梯度向量，而对在线牛顿步算法，这一向量则是不同的：它保持了在使用前面的代价函数时，使用离线 Newton-Raphson 方法能得到的方向。Newton-Raphson 算法会沿着黑塞矩阵的逆与梯度向量乘积的方向移动。在线牛顿步算法中，这一方向为 $A_t^{-1}\nabla_t$ ，其中矩阵 A_t 是与黑塞矩阵相关的，在后面将对其进行分析。

由于在当前的向量中增加了一个牛顿向量 $A_t^{-1}\nabla_t$ 的倍数，最终得到的点可能会在凸集之外，因此需要一个附加的投影步来得到 \mathbf{y}_t ，即在时刻 t 的决策向量。这一投影与了前面在线梯度下降算法中使用的标准欧氏投影是不同的。它是在用 A_t 定义的范数的基础上得到的，而不是在欧氏范数意义下的。

算法 12.26 在线牛顿步算法

- 1: 输入：凸集 \mathcal{K} , $T, \mathbf{x}_1 \in \mathcal{K} \subseteq \mathbb{R}^n$, 参数 $\gamma, \epsilon > 0, A_0 = \epsilon I_n$
- 2: **for** $t = 1$ 到 T **do**
- 3: 执行 \mathbf{x}_t 并考代价函数 $f_t(\mathbf{x}_t)$
- 4: 秩 1 更新： $A_t = A_{t-1} + \nabla_t \nabla_t^T$
- 5: 牛顿步及投影：

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \frac{1}{\gamma} A_t^{-1} \nabla_t$$

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{K}}^{A_t}(\mathbf{y}_{t+1})$$

- 6: **end for**

在线牛顿步算法的优点是，它对 exp 凹函数存在对数遗憾，正如下面的定理所示，给出了

在线牛顿步算法遗憾的界。

定理 12.5.3. 参数为 $\gamma = \frac{1}{2} \min \frac{1}{4GD}, \alpha, \epsilon - \frac{1}{\gamma^2 D^2}$ 及 $T > 4$ 的算法 12.26 保证了

$$\text{遗憾}_T \leq 5\left(\frac{1}{\alpha} + GD\right)n \log T$$

第一步, 首先证明下面的引理。

引理 12.5.1. 在线牛顿步算法的遗憾界为

$$\text{遗憾}_T(ONS) \leq 4\left(\frac{1}{\alpha} + GD\right)\left(\sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + 1\right)$$

证明. 令 $\mathbf{x}^* \in \arg \min_{\mathbf{x} \in \mathbb{I}} \sum_{t=1}^T f_t(\mathbf{x})$ 为事后最好的决策。我们容易得到 (这一结论留作课后习题), 对 $\gamma = \frac{1}{2} \min\{\frac{1}{4GD}, \alpha\}$,

$$f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*) \leq \text{基础}$$

其中定义

$$R_t \triangleq \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) - \frac{\gamma}{2} (\mathbf{x}^* - \mathbf{x}_t)^T \nabla_t \nabla_t^T (\mathbf{x}^* - \mathbf{x}_t)$$

由算法更新的公式 $\mathbf{x}_{t+1} = \Pi_{\mathcal{K}}^{A_t}(\mathbf{y}_{t+1})$, 现由 \mathbf{y}_{t+1} 定义:

$$\mathbf{y}_{t+1} - \mathbf{x}^* = \mathbf{x}_t - \mathbf{x}^* - \frac{1}{\gamma} A_t^{-1} \nabla_t \quad (12.98)$$

及

$$A_t(\mathbf{y}_{t+1} - \mathbf{x}^*) = A_t(\mathbf{x}_t - \mathbf{x}^*) - \frac{1}{\gamma} \nabla_t \quad (12.99)$$

将式 (12.98) 的转置乘以式 (12.99) 可得

$$\begin{aligned} & (\mathbf{y}_{t+1} - \mathbf{x}^*)^T A_t (\mathbf{y}_{t+1} - \mathbf{x}^*) \\ &= (\mathbf{x}_t - \mathbf{x}^*)^T A_t (\mathbf{x}_t - \mathbf{x}^*) - \frac{2}{\gamma} \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) + \frac{1}{\gamma^2} \nabla_t^T A_t^{-1} \nabla_t \end{aligned} \quad (12.100)$$

由于 \mathbf{x}_{t+1} 为 \mathbf{y}_{t+1} 在 A_t 诱导范数意义下的投影, 因此,

$$\begin{aligned} (\mathbf{y}_{t+1} - \mathbf{x}^*)^T A_t (\mathbf{y}_{t+1} - \mathbf{x}^*) &= \|\mathbf{y}_{t+1} - \mathbf{x}^*\|_{A_t}^2 \\ &\geq \|\mathbf{x}_{t+1} - \mathbf{x}^*\|_{A_t}^2 \\ &= (\mathbf{x}_{t+1} - \mathbf{x}^*)^T A_t (\mathbf{x}_{t+1} - \mathbf{x}^*) \end{aligned}$$

这一不等式就是在在线梯度下降算法中使用广义投影而不是使用标准投影的原因。将这一事实结合式 (12.100) 就得到

$$\begin{aligned} \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) &\leq \frac{1}{2\gamma} \nabla_t^T A_t^{-1} \nabla_t + \frac{\gamma}{2} (\mathbf{x}_t - \mathbf{x}^*)^T A_t (\mathbf{x}_t - \mathbf{x}^*) \\ &\quad - \frac{\gamma}{2} (\mathbf{x}_{t+1} - \mathbf{x}^*)^T A_t (\mathbf{x}_{t+1} - \mathbf{x}^*) \end{aligned} \quad (12.101)$$

将上式对 $t = 1$ 到 T 求和, 可得

$$\begin{aligned} \sum_{t=1}^T \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) &\leq \frac{1}{2\gamma} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + \frac{\gamma}{2} (\mathbf{x}_1 - \mathbf{x}^*)^T A_1 (\mathbf{x}_1 - \mathbf{x}^*) \\ &\quad + \frac{\gamma}{2} \sum_{t=2}^T (\mathbf{x}_t - \mathbf{x}^*)^T (A_t - A_{t-1}) (\mathbf{x}_t - \mathbf{x}^*) \\ &\quad - \frac{\gamma}{2} (\mathbf{x}_{T+1} - \mathbf{x}^*)^T A_T (\mathbf{x}_{T+1} - \mathbf{x}^*) \\ &\leq \frac{1}{2\gamma} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + \frac{\gamma}{2} \sum_{t=1}^T (\mathbf{x}_t - \mathbf{x}^*)^T \nabla_t \nabla_t^T (\mathbf{x}_t - \mathbf{x}^*) \\ &\quad - \frac{\gamma}{2} (\mathbf{x}_1 - \mathbf{x}^*)^T (A_1 - \nabla_1 \nabla_1^T) (\mathbf{x}_1 - \mathbf{x}^*) \end{aligned}$$

在最后一个不等式中使用了 $A_t - A_{t-1} = \nabla_t \nabla_t^T$, 及矩阵 A_T 为半正定的事实, 因此不等式的最后一项是负的。故

$$\sum_{t=1}^T R_t \leq \frac{1}{2\gamma} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + \frac{\gamma}{2} (\mathbf{x}_1 - \mathbf{x}^*)^T (A_1 - \nabla_1 \nabla_1^T) (\mathbf{x}_1 - \mathbf{x}^*)$$

利用算法参数 $A_1 - \nabla_1 \nabla_1^T = \epsilon I_n$, $\epsilon = \frac{1}{\gamma^2 D^2}$, 及直径的记号 $\|\mathbf{x}_1 - \mathbf{x}^*\| \leq D^2$, 有

$$\begin{aligned} \text{遗憾}_T(ONS) &\leq \sum_{t=1}^T R_t \leq \frac{1}{2\gamma} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + \frac{\gamma}{2} D^2 \epsilon \\ &\leq \frac{1}{2\gamma} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t + \frac{1}{2\gamma} \end{aligned}$$

由于 $\gamma = \frac{1}{2} \min\{\frac{1}{4GD}, \alpha\}$, 可得 $\frac{1}{\gamma} \leq 8(\frac{1}{\alpha} + GD)$, 这就证明了引理。 \square

现在可以证明定理 12.5.3。

定理 12.5.3 的证明. 首先证明 $\sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t$ 的上界式被等比级数的和限定的。注意到

$$\nabla_t^T A_t^{-1} \nabla_t = A_t^{-1} \bullet \nabla_t \nabla_t^T = A_t^{-1} \bullet (A_t - A_{t-1})$$

对其中的矩阵 $A, B \in \mathbb{R}^{n \times n}$, 记 $A \bullet B = \sum_{i=1}^n \sum_{j=1}^n A_{ij} B_{ij} = \text{Tr}(AB^T)$, 它等价于将这些矩阵看作 \mathbb{R}^{n^2} 中的向量时的内积。

对实数 $a, b \in \mathbb{R}_+$, 在点 a 处 b 的对数的一阶 Taylor 展开意味着 $a^{-1}(a - b) \leq \log \frac{a}{b}$ 。对半正定矩阵也有一个类似的结果, 即 $A^{-1} \bullet (A - B) \leq \log \frac{|A|}{|B|}$, 其中 $|A|$ 表示矩阵 A 的行列式 (这一结论留作练习中证明)。利用这一事实, 有

$$\begin{aligned} \sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t &= \sum_{t=1}^T A_t^{-1} \bullet \nabla_t \nabla_t^T \\ &= \sum_{t=1}^T A_t^{-1} \bullet (A_t - A_{t-1}) \\ &\leq \sum_{t=1}^T \log \frac{|A_t|}{|A_{t-1}|} = \log \frac{|A_T|}{|A_0|} \end{aligned}$$

由于 $A_T = \sum_{t=1}^T \nabla_t \nabla_t^T + \epsilon I_n$ 且 $\|\nabla_t\| \leq G$, A_T 最大的特征值是 $TG^2 + \epsilon$ 。因此 A_T 的行列式满足 $|A_T| \leq (TG^2 + \epsilon)^n$ 。回顾 $\epsilon = \frac{1}{\gamma^2 D^2}$ 以及对 $T > 4$ 有 $\gamma = \frac{1}{2} \min\{\frac{1}{4GD}, \alpha\}$, 于是

$$\sum_{t=1}^T \nabla_t^T A_t^{-1} \nabla_t \leq \log\left(\frac{TG^2 + \epsilon}{\epsilon}\right)^n \leq n \log(TG^2 \gamma^2 D^2 + 1) \leq n \log T$$

带入引理12.5.1的结论可得

$$\text{遗憾}_T(ONS) \leq 4\left(\frac{1}{\alpha} + GD\right)(n \log T + 1)$$

故定理在 $n > 1, T \geq 8$ 时成立。 \square

实现与运行时间

在线牛顿步算法需要 $O(n^2)$ 的存储空间来存储矩阵 A_t 。每一次迭代需要经计算矩阵 A_t^{-1} 、当前梯度、一个矩阵于向量的乘积, 以及可能需要的向量基本凸集 \mathcal{K} 上的投影。

一种初等的实现方法需要在每一次迭代时计算矩阵 A_t 的逆。但是, 当 A_t 可逆时, 根据前面章节中介绍矩阵求逆的定理可得, 对可逆矩阵 A 和向量 \mathbf{x} , 有

$$(A + \mathbf{x}\mathbf{x}^T) = A^{-1} - \frac{\mathbf{x}\mathbf{x}^T A^{-1}}{1 + \mathbf{x}^T A^{-1} \mathbf{x}}$$

因此, 给定 A_{t-1}^{-1} 和 ∇_t , 可以仅使用矩阵和向量的乘法和向量于向量的乘法, 用 $O(n^2)$ 时间给出 A_t^{-1} 。

在线牛顿步算法也需要在 \mathcal{K} 上投影, 但与在线梯度下降算法和其他在线凸优化算法的情形有所不同。此处需要的投影 (记为 $\Pi_{\mathcal{K}}^{A_t}$) 为向量在矩阵 A_t 诱导范数下的投影, 即 $\|\mathbf{x}\|_{A_t} = \sqrt{\mathbf{x}^T A_t \mathbf{x}}$ 。它等价于求向量 $\mathbf{x} \in \mathcal{K}$, 使其最小化 $(\mathbf{x} - \mathbf{y})^T A_t (\mathbf{x} - \mathbf{y})$, 其中 \mathbf{y} 为被投影的点。这是一个凸优化, 它可以使用多项式时间得到任意精度的解。

在相差常数倍的前提下, 广义投影算法、在线牛顿步算法可以使用 $O(n^2)$ 的时间和空间复杂度实现。此外, 它们仅需在每一步中给出梯度的信息。

12.6 阅读材料

关于优化算法的系统研究始于 20 世纪 40 年代后期, 1951 年 Kuhn 和 Tucker 提出了著名的 KKT 条件。此后, 无论在基本理论还是在使用算法的研究方面都发展很快。目前, 优化算法已成为数学规划中内容十分丰富的一个分支。限于篇幅, 本篇仅叙述了优化算法中最基本的一些概念和算法, 力图便于读者掌握一些重要方法, 并为进一步深入学习打好基础。

在求解无约束优化问题的方法中, 拟牛顿法 (变尺度法)、共轭梯度法占有十分重要的地位。如欲进一步研究, 除本篇末列出的参考文献外, 还可参阅: 邓乃杨等著, 无约束最优化计算方法. 北京: 科学出版社, 1982 年。

对约束优化问题, 除本篇提到者外, 梯度投影法、简约梯度法、约束变尺度法、乘子罚函数法、序列二次规划法和起作用约束集法 (active set method) 等都是很重要的方法。其中简约梯

度法和起作用约束集法对处理线性约束非线性目标函数十分有效。处理一般非线性约束非线性目标函数的有效方法，有待进一步研究。

在最后一节中，我们讨论了一系列算法，通过自适应每个模型参数的学习速率以解决优化深度模型中的难题。此时，一个自然的问题是：该选择哪种算法呢？

遗憾的是，目前在这一点上没有达成共识。Schaul(2014)展示了许多优化算法在大量学习任务上极具价值的比较。虽然结果表明，具有自适应学习速率（以 RMSProp 和 AdaDelta 为代表）的算法族表现得相当鲁棒，不分伯仲，但没有哪个算法能脱颖而出。

目前，最流行并且使用很高的优化算法包括 SGD、具动量的 SGD、RMSProp、具动量的 RMSProp、AdaDelta 和 Adam。此时，选择哪一个算法似乎主要取决于使用者对算法的熟悉程度（以便调节超参数）。

12.7 习题基础

习题 12.1. 试用斐波那契法求函数

$$f(x) = x^2 - 6x + 2$$

在区间 $[0, 10]$ 上的极小点，要求缩短后的区间长度不大于原区间长度的 8%。

习题 12.2. 试用 0.618 法重做习题 1.1，并将计算结果与斐波那契法所得计算结果进行比较。

习题 12.3. 试用最速下降法求解

$$\min f(\mathbf{x}) = x_1^2 + x_2^2 + x_3^2$$

选初始点 $\mathbf{x}^{(0)} = (2, -2, 1)^T$ ，要求做三次迭代，并验证相邻两步的搜索方向正交。

习题 12.4. 试用最速下降法求函数

$$f(\mathbf{x}) = -(x_1 - 2)^2 - 2x_2^2$$

的极大点。先以 $\mathbf{x}^{(0)} = (0, 0)^T$ 为初始点进行计算，求出极大点；再以 $\mathbf{x}^{(0)} = (0, 1)^T$ 为初始点进行两次迭代。最后比较从上述两个不同初始点出发的寻优过程。

习题 12.5. 试用牛顿法重新解习题 1.4。

习题 12.6. 试用牛顿法求解

$$\max f(\mathbf{x}) = \frac{1}{x_1^2 + x_2^2 + 2}$$

取初始点 $\mathbf{x}^{(0)} = (4, 0)^T$ ，用最佳步长进行。然后采用固定步长 $\lambda = 1$ ，观察迭代情况，并加以分析说明。

习题 12.7. 试用共轭梯度法求二次函数

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x}$$

的极小点, 此处

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

习题 12.8. 令 $\mathbf{x}^{(i)} (i = 1, 2, \dots, n)$ 为一组 \mathbf{A} 共轭向量 (假定为列向量), \mathbf{A} 为 $n \times n$ 对称正定阵, 试证

$$\mathbf{A}^{-1} = \sum_{i=1}^n \frac{\mathbf{x}^{(i)} (\mathbf{x}^{(i)})^T}{(\mathbf{x}^{(i)})^T \mathbf{A} \mathbf{x}^{(i)}}$$

习题 12.9. 试用拟牛顿法求解

$$\min f(\mathbf{x}) = (x_1 - 2)^2 + (x_2 - 2x_2)^2$$

取初始点 $\mathbf{x}^{(0)} = (0.00, 3.00)^T$, 要求近似极小点处梯度的模不大于 0.5。

习题 12.10. 试以 $\mathbf{x}^{(0)} = (0, 0)^T$ 为初始点, 使用

- (1) 最速下降法 (迭代 4 次);
- (2) 牛顿法;
- (3) 拟牛顿法。求解无约束优化问题

$$\min f(\mathbf{x}) = 2x_1^2 + x_2^2 + 2x_1x_2 + x_1 - x_2$$

并绘图表示使用上述各方法的寻优过程。

习题 12.11. 分析约束优化问题

$$\begin{cases} \min f(\mathbf{x}) = (x_1 - 2)^2 + (x_2 - 3)^2 \\ x_1^2 + (x_2 - 2) \geq 4 \\ x_2 \leq 2 \end{cases}$$

在以下各点的可行下降方向 (使用式(12.51)和式(??)):

(1) $\mathbf{x}^{(1)} = (0, 0)^T$; (2) $\mathbf{x}^{(2)} = (2, 2)^T$; (3) $\mathbf{x}^{(3)} = (3, 2)^T$ 。并绘图表示各点可行下降方向的范围。

习题 12.12. 试用可行方向法求解

$$\begin{cases} \min f(\mathbf{x}) = 2x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1 - 6x_2 \\ x_1 + x_2 \leq 2 \\ x_1 + 5x_2 \leq 5 \\ x_1, x_2 \geq 0 \end{cases}$$

习题 12.13. 试用 SUMT 外点法求解

$$\begin{cases} \min f(\mathbf{x}) = x_1^2 + x_2^2 \\ x_2 = 1 \end{cases}$$

并求出罚因子等于 1 和 10 时的近似解。

习题 12.14. 试用 SUMT 外点法求解

$$\begin{cases} \max f(\mathbf{x}) = x_1 \\ (x_2 - 2) + (x_1 - 1)^3 \leq 0 \\ (x_1 - 1)^3 - (x_2 - 2) \leq 0 \\ x_1, x_2 \geq 0 \end{cases}$$

习题 12.15. 试用 SUMT 内点法求解

$$\begin{cases} \min f(\mathbf{x}) = (x + 1)^2 \\ x \geq 0 \end{cases}$$

习题 12.16. 试用 SUMT 内点法求解

$$\begin{cases} \min f(\mathbf{x}) = x \\ -1 \leq x \leq 1 \end{cases}$$

12.8 参考文献

- [1] 南京大学数学系计算数学专业编. 最优化方法. 北京: 科学出版社, 1978
- [2] 王德人编. 非线性方程组解法与最优化方法. 北京: 人民教育出版社, 1978
- [3] 中国科学院数学研究所运筹室编. 最优化方法. 北京: 科学出版社, 1980
- [4] 马仲蕃、魏权龄、赖炎连编. 数学规划讲义. 北京: 中国人民大学出版社, 1981
- [5] 薛嘉庆编. 最优化原理与方法. 北京: 冶金工业出版社, 1983
- [6] 席少霖、赵凤治编著. 最优化计算方法. 上海: 上海科学技术出版社, 1983
- [7] 郭耀煌等编著. 运筹学与工程系统分析. 北京: 中国建筑工业出版社, 1986
- [8] 徐光辉主编, 刘彦佩、程侃副主编. 运筹学基础手册. 北京: 科学出版社, 1999
- [9] M. 啊佛里耳著. 李元熹等译. 非线性规划——分析与方法. 上海: 上海科学技术出版社, 1979
- [10] D.M. 希梅尔布劳著, 张义燊等译. 实用非线性规划. 北京: 科学出版社, 1981
- [11] D.G. 鲁恩伯杰著, 夏尊铨等译. 线性与非线性规划引论. 北京: 科学出版社, 1980
- [12] David A. Wismer, Chattergy R. Introduction To Nonlinear Optimization: A Problem Solving Approach, North-Holland Publishing Company, 1978

- [13] Mokhtar S Bazaraa, Shetty C M. Nonlinear Programming: Theory and Algorithms, John Wiley & Sons, 1979
- [14] Philip E Gill, Walter Murray and Margaret H. Wright, Practical Optimization, Academic Press, 1981
- [15] Fletcher R. Practical Methods of Optimization, Vol. 2, John Wiley & Sons, 1981
- [16] Edited by A. Bachem, M. Grötschel and B. korte, Mathematical Programming: The State of the Art, Bonn 1982, Springer-Verlag, 1983
- [17] Bottou L. and Bousquet, O. The tradeoffs of large scale learning. In NIPS' 2008, 2008
- [18] Bottou L. Online algorithms and stochastic approximations. In D. Saad, editor, Online Learning in Neural Networks. Cambridge University Press, Cambridge, UK, 1998
- [19] Polyak B. T. Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics, 4(5), 1–17, 1964
- [20] Sutskever I., Vinyals O., and Le Q. V. Sequence to sequence learning with neural networks. In NIPS' 2014, arXiv:1409.3215, 2014
- [21] Jacobs R. A. Increased rates of convergence through learning rate adaptation. Neural networks, 1(4), 295–307, 1988
- [22] Duchi J., Hazan E., and Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 2011
- [23] Hinton G. E. Tutorial on deep learning. IPAM Graduate Summer School: Deep Learning, Feature Learning, 2012
- [24] Kingma D. and Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014
- [25] Schaul T., Antonoglou I., and Silver D. Unit tests for stochastic optimization. In International Conference on Learning Representations, 2014