

MATRIX ANALYSIS AND APPLICATIONS  
矩阵分析与应用

Second Edition  
(第2版)

张贤达 著  
Zhang Xianda



清华大学出版社



张贤达 1969年毕业于原西安军事电信工程学院，1982年获哈尔滨工业大学工学硕士学位，1987年获日本东北大学工学博士学位。曾任原航空工业部304研究所高级工程师、研究员，1992年9月起任清华大学自动化系教授，1993年被批准为博士生导师，从事信号与信息处理教学与科研。1993年起，享受国务院政府特殊津贴；1997年被教育部和国家人事部评为“全国优秀留学回国人员”，1999年评为教育部首批“长江学者”，在西安电子科技大学任特聘教授三年。发表SCI收录学术论文80余篇，出版学术著作6部。论著被SCI他引1100余次，Google学术搜索他引6700余次。



0151.21  
20-2

• 014006459

MATRIX ANALYSIS AND APPLICATIONS  
**矩阵分析与应用** Second Edition  
(第2版)

张贤达 著  
Zhang Xianda



0151.21

20-2



北航 C1693498

清华大学出版社  
北京



## 内 容 简 介

本书系统、全面地介绍矩阵分析的主要理论、具有代表性的方法及一些典型应用。全书共 10 章，内容包括矩阵代数基础、特殊矩阵、矩阵微分、梯度分析与最优化、奇异值分析、矩阵方程求解、特征分析、子空间分析与跟踪、投影分析、张量分析。前 3 章为全书的基础，组成矩阵代数；后 7 章介绍矩阵分析的主体内容及典型应用。为了方便读者对数学理论的理解以及培养应用矩阵分析进行创新应用的能力，本书始终贯穿一条主线——物理问题“数学化”，数学结果“物理化”。与第 1 版相比，本书的篇幅有明显的删改和压缩，大量补充了近几年发展迅速的矩阵分析新理论、新方法及新应用。

本书为北京市高等教育精品教材重点立项项目，适合于需要矩阵知识比较多的理科和工科尤其是信息科学与技术（电子、通信、自动控制、计算机、系统工程、模式识别、信号处理、生物医学、生物信息）等各学科有关教师、研究生和科技人员教学、自学或进修之用。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

### 图书在版编目(CIP)数据

矩阵分析与应用/张贤达著.—2 版.—北京：清华大学出版社，2013

ISBN 978-7-302-33859-8

I. ①矩… II. ①张… III. ①矩阵分析 IV. ①O151. 21

中国版本图书馆 CIP 数据核字(2013)第 215838 号

责任编辑：王一玲

封面设计：傅瑞学

责任校对：白 蕾

责任印制：杨 艳

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载：<http://www.tup.com.cn>, 010-62795954

印 装 者：三河市春园印刷有限公司

经 销：全国新华书店

开 本：185mm×260mm 印 张：42.5 字 数：1008 千字

版 次：2004 年 10 月第 1 版 2013 年 11 月第 2 版 印 次：2013 年 11 月第 1 次印刷

印 数：1~2000

定 价：89.00 元

---

产品编号：055920-01

## 第 2 版序言

《矩阵分析与应用》于 2004 年 10 月出版以来, 已先后印刷发行 14300 册, 2008 年获清华大学优秀教材一等奖, 2011 年获北京市高等教育精品教材重点项目资助; 截至 2013 年 8 月, 已被 SCI 他引 220 余次, Google 学术搜索他引 740 余次, CNKI 中国引文数据库他引 1400 余次。

最近几年, 矩阵理论经历了巨大的变化: 矩阵分析的理论和方法在物理、力学、信号处理、图像处理、无线通信、计算机视觉、机器学习、生物信息学、医学图像处理、自动控制、系统工程、航空航天等学科中获得了广泛的应用, 有力地推动了这些学科的创新研究。同时, 这些学科的新应用又催生了矩阵分析的一批新理论和新方法。

为了适应矩阵分析与应用的新发展, 根据从 2004 年在清华大学开设的研究生学位课程“矩阵分析与应用”的课堂教学实践, 笔者对《矩阵分析与应用》一书进行了重大修改。修改的主要宗旨是: 以工学和工程应用为主要背景, 论述矩阵分析的典型理论、方法和应用; 同时重点介绍最近几年涌现出来的矩阵分析的新理论、新方法与新应用。为了方便读者对数学理论的理解以及培养应用矩阵分析进行创新应用的能力, 本书的修改始终贯穿一条主线——物理问题“数学化”, 数学结果“物理化”: 从物理问题的数学建模出发, 引出矩阵问题; 对得到的矩阵分析结果尽可能给予物理解释, 赋予其物理含义。

新版仍由 10 章组成, 内容可以分为以下两部分。

第 1 部分为“矩阵代数”: 包括矩阵代数基础(第 1 章)、特殊矩阵(第 2 章)和矩阵微分(第 3 章), 共 3 章。

第 2 部分为“矩阵分析与应用”: 包括梯度分析与最优化(第 4 章)、奇异值分析(第 5 章)、矩阵方程求解(第 6 章)、特征分析(第 7 章)、子空间分析与跟踪(第 8 章)、投影分析(第 9 章)和张量分析(第 10 章), 共 7 章。

与第 1 版相比, 第 2 版的主要修订内容如下:

(1) 章的变动: 删去了第 1 版的“Toeplitz 矩阵”(第 3 章)和“矩阵的变换与分解”(第 4 章)两章, 增设了“矩阵微分”(第 3 章)和“张量分析”(第 10 章)两章; 另将第 1 版的“总体最小二乘方法”(第 7 章)加以大量修改和扩充, 更名为“矩阵方程求解”(第 2 版第 6 章)。

(2) 删除的主要内容:

- ① 比较容易和比较难的数学证明, 前者变作习题, 后者改为参阅有关参考文献;
- ② 工学和工程中应用比较窄的一些矩阵分析理论和方法;
- ③ 专业性比较强的应用举例。

(3) 新增矩阵分析与应用的主要内容:

- ① 稀疏表示与压缩感知(1.12 节);
- ② 矩阵微分与梯度矩阵辨识、Hessian 矩阵辨识(第 3 章);

③ 凸优化理论 (4.3 节)、平滑凸优化的一阶算法 (4.4 节)、非平滑凸优化的次梯度法 (4.5 节)、非平滑凸函数的平滑凸优化 (4.6 节) 以及原始-对偶内点法 (4.9 节);

④ 矩阵完备 (5.6 节);

⑤ Tikhonov 正则化与正则 Gauss-Seidel 法 (6.2 节);

⑥ 非负矩阵分解 (6.6 和 6.7 节);

⑦ 稀疏矩阵方程求解 (6.8 和 6.9 节);

⑧ 张量分析及非负张量分解 (第 10 章)。

它们多数是近几年发展迅速的矩阵分析新理论、新方法及新应用。

虽然本书增加了大量新内容,但是由于删除、修改了更多的内容,所以全书的篇幅反而有比较明显的缩减。

在“矩阵分析与应用”研究生学位课程的教学实践和本书的修订中,韩芳明、李细林、李剑、苏泳涛、丁子哲、高秋彬、王锟、常冬霞、王曦元、陈忠、栾天祥等博士和邹红星教授提供了一些很好的建议;王锟、陈忠和郑亮为本书绘制了部分插图。符玺、毛洪亮、石群、周游、金成等博士研究生和杨哲硕士研究生认真校对了本书初稿。在此一并向他们表示谢意!

本书的修订得到了国家自然科学基金委重大研究项目和多个基金项目、教育部博士点专项基金、清华信息科学与技术国家实验室、国防重点实验室基金、航天支撑技术基金以及 Intel 公司等的课题资助。

全书由笔者使用 L<sup>A</sup>T<sub>E</sub>X 撰写及排版,多数插图也由笔者用 L<sup>A</sup>T<sub>E</sub>X 绘制。

张 贤 达

2013 年 8 月于清华大学

## 首版前言

矩阵不仅是各数学学科，而且也是许多理工学科的重要数学工具。就其本身的研究而言，矩阵理论和线性代数也是极富创造性的领域。它们的创造性又极大地推动和丰富了其他众多学科的发展：许多新的理论、方法和技术的诞生与发展就是矩阵理论和线性代数的创造性应用与推广的结果。可以毫不夸张地说，矩阵理论和线性代数在物理、力学、信号与信息处理、通信、电子、系统、控制、模式识别、土木、电机、航空和航天等众多学科中是最富创造性和灵活性，并起着不可替代作用的数学工具。

作者在从事信号处理、神经计算、通信和模式识别的长期科学的研究中，深刻感受到矩阵分析在科学的研究中所起的重要作用，并体现在作者和合作者在国际权威和著名杂志发表的一系列论文中。另一方面，在十余年的研究生教学中，笔者对工科尤其是信息科学与技术各学科的研究生在矩阵理论与线性代数方面知识的不足与欠缺颇有体会。矩阵分析理论与方法的重要性，以及作者的教学和研究体会，催发了作者著作本书的意愿。虽然作者的《信号处理中的线性代数》一书曾由科学出版社于 1997 年出版，但本书无论是在体系结构上，还是在内容的组织与安排上，与《信号处理中的线性代数》大不相同。

国内外出版了不少深受读者喜爱的矩阵理论和线性代数的书，而本书试图从一个新的角度，提出从矩阵的梯度分析、奇异值分析、特征分析、子空间分析、投影分析出发，构筑论述矩阵分析的一个新体系。此外，在国内外的有关书中，涉及矩阵理论和线性代数的应用时，一般侧重于某一、二个特定的学科，本书则介绍矩阵分析在数理统计、数值计算、信号处理、电子、通信、模式识别、神经计算、系统科学等多学科中的大生动应用。鉴于本书介绍的理论与应用的广泛性，故取名《矩阵分析与应用》。

全书共分 10 章，其主要内容可概括如下：

- (1) 矩阵分析的基础知识 (第 1 ~ 4 章)：矩阵与线性方程组、特殊矩阵、Toeplitz 矩阵、矩阵的变换与分解。
- (2) 梯度分析 (第 5 章)：包括一阶梯度和二阶梯度的计算，以及实现最优化的梯度算法及其重要改进 (递推最小二乘算法、共轭梯度算法、仿射投影算法和自然梯度算法)。
- (3) 矩阵的奇异值分析 (第 6 ~ 7 章)：第 6 章介绍奇异值分解及其各种推广 (乘积奇异值分解、广义奇异值分解、约束奇异值分解、结构奇异值)。第 7 章是奇异值分解在线性代数中的应用，介绍总体最小二乘方法、约束总体最小二乘、结构总体最小二乘。
- (4) 矩阵的特征分析 (第 8 章)：包含矩阵的特征值分解以及各种推广 (广义特征值分解、Rayleigh 商、广义 Rayleigh 商、二次特征值问题、矩阵的联合对角化)。
- (5) 子空间分析 (第 9 章)：子空间的构造、特征子空间分析方法、子空间的跟踪。
- (6) 投影分析 (第 10 章)：包含沿着矩阵的基本空间 (列空间或者行空间)，到另一基本空间的正交投影和斜投影。

本书试图在以下方面形成特点：

- (1) 加大选材的广度和深度，充分体现内容的新颖性和先进性。为了与矩阵理论的国

际新发展“接轨”，书中系统地介绍了矩阵分析的一些新领域、新理论和新方法，如总体最小二乘方法及其推广，二次特征值问题，矩阵的联合对角化，斜投影，子空间方法，仿射投影算法和自然梯度算法等。

(2) 突出矩阵分析理论与科学技术应用的密切结合。本书在介绍每一种重要理论与方法的同时，都会选择介绍相应的应用。而在应用例子的选择上，则尽可能包括比较多的学科。事实上，本书的应用举例不仅涉及数理统计和数值计算等数学领域，更包括了信号处理、电子、通信、模式识别、神经计算、雷达、图像处理、系统辨识等信息科学与技术的不同学科与领域。

(3) 强调创新能力的培养。书中介绍大量应用例子时，侧重于讲述应用的基本机理，其出发点是让读者体会矩阵分析的灵活性与创新性，学会如何使用矩阵分析的工具，进行创新研究。

为便于读者理解重要的概念和方法，书中穿插了大量的例题。为了方便读者检验学习效果，全书在参考全国硕士研究生招生部分数学试题和其他有关文献的基础上，选编了 340 余道习题。此外，本书不仅汇总了矩阵分析有关的大量数学性质和公式，而且汇编了 820 余条索引，可供读者作为一本矩阵手册使用。

本书是从一个工科研究和教学人员的视角进行材料的选择和内容论述的。作者在著作本书的过程中，参考了大量的国外有关矩阵分析与线性代数的论文和著作，其中以 SIAM 的多种杂志为主要参考文献源；而应用的举例则主要参考 IEEE 的几家汇刊。虽然作者竭力而为，但囿于理解水平和能力，书中未能如愿乃至不妥，甚至错误之处可能不乏其例。在此，诚恳希望诸位专家、同仁和广大读者不吝赐教。

作者原本打算对《信号处理中的线性代数》一书作较大修改，最终变成了重写，始自本人在西安电子科技大学任特聘教授之际，完成于回到清华大学任教二年之后，历时四载有余。然而，本书系作者积十余年教学和二十多年科学研究之体会与成果而成，借此机会感谢教育部“长江学者奖励计划”、国家自然科学基金委重大研究项目和多个基金项目、教育部博士点专项基金、国防重点实验室基金、航天支撑技术基金以及 Intel 公司等的课题资助。

全书由笔者使用 LATEX 撰写及排版。

张贤达  
2004 年 6 月谨识于清华大学

# 目 录

<b>第1章 矩阵代数基础 . . . . .</b>	<b>1</b>
<b>1.1 矩阵的基本运算 . . . . .</b>	<b>1</b>
1.1.1 矩阵与向量 . . . . .	1
1.1.2 矩阵的基本运算 . . . . .	4
1.1.3 向量的线性无关性与非奇异矩阵 . . . . .	8
<b>1.2 矩阵的初等变换 . . . . .</b>	<b>9</b>
1.2.1 初等行变换与阶梯型矩阵 . . . . .	9
1.2.2 初等行变换的两个应用 . . . . .	11
1.2.3 初等列变换 . . . . .	14
<b>1.3 向量空间、线性映射与 Hilbert 空间 . . . . .</b>	<b>15</b>
1.3.1 集合的基本概念 . . . . .	16
1.3.2 向量空间 . . . . .	17
1.3.3 线性映射 . . . . .	20
1.3.4 内积空间、赋范空间与 Hilbert 空间 . . . . .	23
<b>1.4 内积与范数 . . . . .</b>	<b>26</b>
1.4.1 向量的内积与范数 . . . . .	26
1.4.2 向量的相似比较 . . . . .	30
1.4.3 矩阵的内积与范数 . . . . .	32
<b>1.5 随机向量 . . . . .</b>	<b>36</b>
1.5.1 概率密度函数 . . . . .	36
1.5.2 随机向量的统计描述 . . . . .	38
1.5.3 高斯随机向量 . . . . .	41
<b>1.6 矩阵的性能指标 . . . . .</b>	<b>43</b>
1.6.1 矩阵的二次型 . . . . .	44
1.6.2 行列式 . . . . .	45
1.6.3 矩阵的特征值 . . . . .	47
1.6.4 矩阵的迹 . . . . .	49
1.6.5 矩阵的秩 . . . . .	51
<b>1.7 逆矩阵与伪逆矩阵 . . . . .</b>	<b>54</b>
1.7.1 逆矩阵的定义与性质 . . . . .	54
1.7.2 矩阵求逆引理 . . . . .	56
1.7.3 左逆矩阵与右逆矩阵 . . . . .	59
<b>1.8 Moore-Penrose 逆矩阵 . . . . .</b>	<b>61</b>
1.8.1 Moore-Penrose 逆矩阵的定义与性质 . . . . .	61

---

1.8.2 Moore-Penrose 逆矩阵的计算 . . . . .	64
1.8.3 非一致方程的最小范数最小二乘解 . . . . .	67
1.9 矩阵的直和与 Hadamard 积 . . . . .	67
1.9.1 矩阵的直和 . . . . .	67
1.9.2 Hadamard 积 . . . . .	68
1.10 Kronecker 积与 Khatri-Rao 积 . . . . .	71
1.10.1 Kronecker 积及其性质 . . . . .	71
1.10.2 广义 Kronecker 积 . . . . .	73
1.10.3 Khatri-Rao 积 . . . . .	74
1.11 向量化与矩阵化 . . . . .	74
1.11.1 矩阵的向量化与向量的矩阵化 . . . . .	74
1.11.2 向量化算子的性质 . . . . .	77
1.12 稀疏表示与压缩感知 . . . . .	78
1.12.1 稀疏向量与稀疏表示 . . . . .	78
1.12.2 人脸识别的稀疏表示 . . . . .	80
1.12.3 稀疏编码 . . . . .	81
1.12.4 压缩感知的稀疏表示 . . . . .	82
本章小结 . . . . .	86
习题 . . . . .	86
<b>第 2 章 特殊矩阵 . . . . .</b>	<b>101</b>
2.1 Hermitian 矩阵 . . . . .	101
2.2 置换矩阵、互换矩阵与选择矩阵 . . . . .	103
2.2.1 置换矩阵与互换矩阵 . . . . .	103
2.2.2 广义置换矩阵与选择矩阵 . . . . .	106
2.3 正交矩阵与酉矩阵 . . . . .	109
2.4 带型矩阵与三角矩阵 . . . . .	112
2.4.1 带型矩阵 . . . . .	112
2.4.2 三角矩阵 . . . . .	113
2.5 求和向量与中心化矩阵 . . . . .	115
2.5.1 求和向量 . . . . .	115
2.5.2 中心化矩阵 . . . . .	116
2.6 相似矩阵与相合矩阵 . . . . .	117
2.6.1 相似矩阵 . . . . .	117
2.6.2 相合矩阵 . . . . .	119
2.7 Vandermonde 矩阵 . . . . .	120
2.8 Fourier 矩阵 . . . . .	123
2.8.1 Fourier 矩阵的定义与性质 . . . . .	123

---

2.8.2 适定方程计算的初等行变换方法 . . . . .	124
2.8.3 FFT 算法的推导 . . . . .	126
2.9 Hadamard 矩阵 . . . . .	129
2.10 Toeplitz 矩阵 . . . . .	132
2.10.1 对称 Toeplitz 矩阵 . . . . .	132
2.10.2 Toeplitz 矩阵的离散余弦变换 . . . . .	134
2.11 Hankel 矩阵 . . . . .	136
本章小结 . . . . .	138
习题 . . . . .	138
<b>第3章 矩阵微分 . . . . .</b>	<b>143</b>
3.1 Jacobian 矩阵与梯度矩阵 . . . . .	143
3.1.1 Jacobian 矩阵 . . . . .	144
3.1.2 梯度矩阵 . . . . .	145
3.1.3 偏导和梯度计算 . . . . .	147
3.2 一阶实矩阵微分与 Jacobian 矩阵辨识 . . . . .	152
3.2.1 一阶实矩阵微分 . . . . .	152
3.2.2 标量函数的 Jacobian 矩阵辨识 . . . . .	153
3.2.3 实值矩阵函数的 Jacobian 矩阵辨识 . . . . .	161
3.3 二阶实矩阵微分与 Hessian 矩阵辨识 . . . . .	164
3.3.1 Hessian 矩阵 . . . . .	164
3.3.2 Hessian 矩阵的辨识原理 . . . . .	165
3.3.3 Hessian 矩阵的辨识方法 . . . . .	168
3.4 共轭梯度与复 Hessian 矩阵 . . . . .	170
3.4.1 全纯函数与复变函数的偏导 . . . . .	170
3.4.2 复矩阵微分 . . . . .	174
3.4.3 复 Hessian 矩阵 . . . . .	179
3.5 复梯度矩阵与复 Hessian 矩阵的辨识 . . . . .	182
3.5.1 实标量函数的复梯度矩阵辨识 . . . . .	182
3.5.2 矩阵函数的复梯度矩阵辨识 . . . . .	184
3.5.3 复 Hessian 矩阵辨识 . . . . .	187
本章小结 . . . . .	189
习题 . . . . .	189
<b>第4章 梯度分析与最优化 . . . . .</b>	<b>193</b>
4.1 实变函数无约束优化的梯度分析 . . . . .	193
4.1.1 单变量函数 $f(x)$ 的平稳点与极值点 . . . . .	194
4.1.2 多变量函数 $f(x)$ 的平稳点与极值点 . . . . .	196

---

4.1.3 多变量函数 $f(\mathbf{X})$ 的平稳点与极值点 . . . . .	198
4.1.4 实变函数的梯度分析 . . . . .	200
4.2 复变函数无约束优化的梯度分析 . . . . .	202
4.2.1 多变量复变函数 $f(\mathbf{z}, \mathbf{z}^*)$ 的平稳点与极值点 . . . . .	202
4.2.2 多变量复变函数 $f(\mathbf{Z}, \mathbf{Z}^*)$ 的平稳点与极值点 . . . . .	204
4.2.3 无约束最小化问题的梯度分析 . . . . .	206
4.3 凸优化理论 . . . . .	209
4.3.1 标准约束优化问题 . . . . .	209
4.3.2 凸集与凸函数 . . . . .	211
4.3.3 凸函数辨识的充分必要条件 . . . . .	214
4.3.4 凸优化方法及其梯度分析 . . . . .	216
4.4 平滑凸优化的一阶算法 . . . . .	222
4.4.1 梯度法与梯度投影法 . . . . .	222
4.4.2 共轭梯度算法 . . . . .	227
4.4.3 收敛速率 . . . . .	231
4.4.4 Nesterov 最优梯度法 . . . . .	232
4.5 非平滑凸优化的次梯度法 . . . . .	240
4.5.1 次梯度与次微分 . . . . .	240
4.5.2 近似函数 . . . . .	243
4.5.3 共轭函数 . . . . .	244
4.5.4 原始-对偶次梯度算法 . . . . .	246
4.5.5 投影次梯度法 . . . . .	248
4.6 非平滑凸函数的平滑凸优化 . . . . .	249
4.6.1 非平滑函数的平滑逼近 . . . . .	249
4.6.2 近似梯度法 . . . . .	252
4.7 约束优化算法 . . . . .	256
4.7.1 Lagrangian 乘子法与对偶上升法 . . . . .	256
4.7.2 罚函数法 . . . . .	257
4.7.3 增广 Lagrangian 乘子法 . . . . .	261
4.7.4 交替方向乘子法 . . . . .	263
4.8 Newton 法 . . . . .	266
4.8.1 无约束优化的 Newton 法 . . . . .	266
4.8.2 无约束优化的复 Newton 法 . . . . .	268
4.8.3 等式约束优化的 Newton 法 . . . . .	269
4.8.4 等式约束优化的复 Newton 法 . . . . .	272
4.9 原始-对偶内点法 . . . . .	274
4.9.1 非线性优化的原始-对偶问题 . . . . .	274

---

4.9.2 一阶原始-对偶内点法 . . . . .	275
4.9.3 二阶原始-对偶内点法 . . . . .	277
本章小结 . . . . .	280
习题 . . . . .	280
<b>第5章 奇异值分析 . . . . .</b>	<b>285</b>
5.1 数值稳定性与条件数 . . . . .	285
5.2 奇异值分解 . . . . .	288
5.2.1 奇异值分解及其解释 . . . . .	288
5.2.2 奇异值的性质 . . . . .	292
5.2.3 秩亏缺最小二乘解 . . . . .	296
5.3 乘积奇异值分解 . . . . .	298
5.3.1 乘积奇异值分解问题 . . . . .	298
5.3.2 乘积奇异值分解的精确计算 . . . . .	299
5.4 奇异值分解的应用 . . . . .	301
5.4.1 静态系统的奇异值分解 . . . . .	301
5.4.2 图像压缩 . . . . .	304
5.5 广义奇异值分解 . . . . .	304
5.5.1 广义奇异值分解的定义与性质 . . . . .	304
5.5.2 广义奇异值分解的实际算法 . . . . .	307
5.5.3 高阶广义奇异值分解 . . . . .	310
5.5.4 应用 . . . . .	312
5.6 矩阵完备 . . . . .	313
5.6.1 矩阵恢复与矩阵分解 . . . . .	313
5.6.2 矩阵完备及其可辨识性 . . . . .	315
5.6.3 矩阵完备的奇异值阈值化法 . . . . .	319
本章小结 . . . . .	323
习题 . . . . .	323
<b>第6章 矩阵方程求解 . . . . .</b>	<b>325</b>
6.1 最小二乘方法 . . . . .	325
6.1.1 普通最小二乘 . . . . .	325
6.1.2 Gauss-Markov 定理 . . . . .	327
6.1.3 普通最小二乘解与最大似然解的等价性 . . . . .	329
6.1.4 数据最小二乘 . . . . .	329
6.2 Tikhonov 正则化与正则 Gauss-Seidel 法 . . . . .	330
6.2.1 Tikhonov 正则化 . . . . .	330
6.2.2 正则 Gauss-Seidel 法 . . . . .	332

---

6.3 总体最小二乘 . . . . .	336
6.3.1 总体最小二乘问题 . . . . .	336
6.3.2 总体最小二乘解 . . . . .	337
6.3.3 总体最小二乘解的性能 . . . . .	341
6.3.4 总体最小二乘拟合 . . . . .	344
6.4 约束总体最小二乘 . . . . .	348
6.4.1 约束总体最小二乘方法 . . . . .	348
6.4.2 超分辨谐波恢复 . . . . .	350
6.4.3 正则化约束总体最小二乘图像恢复 . . . . .	351
6.5 盲矩阵方程求解的子空间方法 . . . . .	353
6.6 非负矩阵分解的优化理论 . . . . .	355
6.6.1 非负性约束与稀疏性约束 . . . . .	355
6.6.2 非负矩阵分解的数学模型及解释 . . . . .	356
6.6.3 散度与变形对数 . . . . .	360
6.7 非负矩阵分解算法 . . . . .	364
6.7.1 非负矩阵分解的乘法算法 . . . . .	364
6.7.2 投影梯度法和 Nesterov 最优梯度法 . . . . .	369
6.7.3 交替非负最小二乘算法 . . . . .	371
6.7.4 拟牛顿法与多层分解法 . . . . .	373
6.7.5 稀疏非负矩阵分解 . . . . .	374
6.8 稀疏矩阵方程求解: 优化理论 . . . . .	377
6.8.1 $L_1$ 范数最小化 . . . . .	377
6.8.2 RIP 条件 . . . . .	379
6.8.3 与 Tikhonov 正则化最小二乘的关系 . . . . .	381
6.8.4 $L_1$ 范数最小化的梯度分析 . . . . .	382
6.9 稀疏矩阵方程求解: 优化算法 . . . . .	384
6.9.1 正交匹配追踪法 . . . . .	384
6.9.2 LASSO 算法与 LARS 算法 . . . . .	386
6.9.3 同伦算法 . . . . .	389
6.9.4 Bregman 迭代算法 . . . . .	390
本章小结 . . . . .	395
习题 . . . . .	396
<b>第7章 特征分析 . . . . .</b>	<b>399</b>
7.1 特特征值问题与特征方程 . . . . .	399
7.1.1 特特征值问题 . . . . .	399
7.1.2 特特征多项式 . . . . .	401

---

7.2 特特征值与特征向量 . . . . .	402
7.2.1 特特征值 . . . . .	402
7.2.2 特特征向量 . . . . .	403
7.2.3 与其他矩阵函数的关系 . . . . .	405
7.2.4 特特征值和特征向量的性质 . . . . .	408
7.2.5 矩阵的可对角化定理 . . . . .	413
7.3 Cayley-Hamilton 定理及其应用 . . . . .	415
7.3.1 Cayley-Hamilton 定理 . . . . .	415
7.3.2 逆矩阵和广义逆矩阵的计算 . . . . .	417
7.3.3 矩阵幂的计算 . . . . .	418
7.3.4 矩阵指数函数的计算 . . . . .	420
7.4 特特征值分解的几种典型应用 . . . . .	423
7.4.1 标准正交变换与迷向圆变换 . . . . .	423
7.4.2 Pisarenko 谐波分解 . . . . .	426
7.4.3 离散 Karhunen-Loeve 变换 . . . . .	428
7.4.4 主分量分析 . . . . .	430
7.5 广义特征值分解 . . . . .	432
7.5.1 广义特征值分解及其性质 . . . . .	433
7.5.2 广义特征值分解算法 . . . . .	435
7.5.3 广义特征值分解的总体最小二乘方法 . . . . .	436
7.5.4 应用举例——ESPRIT 方法 . . . . .	437
7.5.5 相似变换在广义特征值分解中的应用 . . . . .	440
7.6 Rayleigh 商 . . . . .	442
7.6.1 Rayleigh 商的定义及性质 . . . . .	443
7.6.2 Rayleigh 商迭代 . . . . .	444
7.6.3 Rayleigh 商问题求解的共轭梯度算法 . . . . .	445
7.7 广义 Rayleigh 商 . . . . .	447
7.7.1 广义 Rayleigh 商的定义及性质 . . . . .	447
7.7.2 应用举例 1: 类鉴别有效性的评估 . . . . .	449
7.7.3 应用举例 2: 干扰抑制的鲁棒波束形成 . . . . .	450
7.8 二次特征值问题 . . . . .	452
7.8.1 二次特征值问题的描述 . . . . .	452
7.8.2 二次特征值问题求解 . . . . .	454
7.8.3 应用举例 . . . . .	458
7.9 联合对角化 . . . . .	462
7.9.1 联合对角化问题 . . . . .	462
7.9.2 正交近似联合对角化 . . . . .	464

---

7.9.3 非正交近似联合对角化 . . . . .	466
7.10 Fourier 分析与特征分析 . . . . .	467
7.10.1 周期函数的 Fourier 分析 . . . . .	467
7.10.2 非周期函数的特征分析 . . . . .	469
本章小结 . . . . .	474
习题 . . . . .	474
<b>第8章 子空间分析与跟踪 . . . . .</b>	<b>483</b>
8.1 子空间的一般理论 . . . . .	483
8.1.1 子空间的基 . . . . .	483
8.1.2 无交连、正交与正交补 . . . . .	485
8.1.3 子空间的正交投影与夹角 . . . . .	488
8.1.4 主角与补角 . . . . .	490
8.1.5 子空间的旋转 . . . . .	491
8.2 列空间、行空间与零空间 . . . . .	492
8.2.1 矩阵的列空间、行空间与零空间 . . . . .	492
8.2.2 子空间的基构造: 初等变换法 . . . . .	495
8.2.3 基本空间的标准正交基构造: 奇异值分解法 . . . . .	498
8.2.4 构造两个零空间交的标准正交基 . . . . .	501
8.3 子空间方法 . . . . .	502
8.3.1 信号子空间与噪声子空间 . . . . .	503
8.3.2 子空间方法应用 1: 多重信号分类 (MUSIC) . . . . .	505
8.3.3 子空间方法应用 2: 子空间白化 . . . . .	507
8.4 Grassmann 流形与 Stiefel 流形 . . . . .	508
8.4.1 不变子空间 . . . . .	508
8.4.2 Grassmann 流形 . . . . .	509
8.4.3 Stiefel 流形 . . . . .	510
8.5 投影逼近子空间跟踪 . . . . .	513
8.5.1 投影逼近子空间跟踪的基本理论 . . . . .	513
8.5.2 投影逼近子空间跟踪算法 . . . . .	516
8.6 快速子空间分解 . . . . .	517
8.6.1 Rayleigh-Ritz 逼近 . . . . .	518
8.6.2 快速子空间分解算法 . . . . .	519
本章小结 . . . . .	522
习题 . . . . .	522

---

<b>第 9 章 投影分析 . . . . .</b>	<b>527</b>
9.1 投影与正交投影 . . . . .	527
9.1.1 投影定理 . . . . .	528
9.1.2 均方估计 . . . . .	529
9.2 投影矩阵与正交投影矩阵 . . . . .	531
9.2.1 幂等矩阵 . . . . .	531
9.2.2 投影算子与正交投影算子 . . . . .	533
9.2.3 到列空间的投影矩阵与正交投影矩阵 . . . . .	535
9.2.4 投影矩阵的导数 . . . . .	537
9.3 投影矩阵与正交投影矩阵的应用举例 . . . . .	538
9.3.1 投影梯度 . . . . .	538
9.3.2 预测滤波器的表示 . . . . .	540
9.4 投影矩阵和正交投影矩阵的更新 . . . . .	544
9.5 满列秩矩阵的斜投影算子 . . . . .	545
9.5.1 斜投影算子的定义及性质 . . . . .	546
9.5.2 斜投影算子的几何解释 . . . . .	550
9.5.3 斜投影算子的递推 . . . . .	552
9.6 满行秩矩阵的斜投影算子 . . . . .	553
9.6.1 满行秩矩阵的斜投影算子定义 . . . . .	553
9.6.2 斜投影的计算 . . . . .	555
9.6.3 斜投影算子的应用 . . . . .	557
本章小结 . . . . .	558
习题 . . . . .	558
<b>第 10 章 张量分析 . . . . .</b>	<b>563</b>
10.1 张量及其表示 . . . . .	563
10.2 张量的矩阵化与向量化 . . . . .	569
10.2.1 张量的水平展开与向量化 . . . . .	569
10.2.2 张量的纵向展开 . . . . .	573
10.3 张量的基本代数运算 . . . . .	577
10.3.1 张量的内积、范数与外积 . . . . .	577
10.3.2 张量的 $n$ -模式积 . . . . .	579
10.3.3 张量的秩 . . . . .	583
10.4 张量的 Tucker 分解 . . . . .	585
10.4.1 Tucker 分解 (高阶奇异值分解) . . . . .	585
10.4.2 三阶奇异值分解 . . . . .	588
10.4.3 高阶奇异值分解的交替最小二乘算法 . . . . .	592

10.5 张量的平行因子分解 . . . . .	596
10.5.1 双线性模型 . . . . .	596
10.5.2 平行因子分析 . . . . .	598
10.5.3 CP 分解的唯一性条件 . . . . .	604
10.5.4 CP 分解的交替最小二乘算法 . . . . .	606
10.6 多路数据分析的预处理与后处理 . . . . .	610
10.6.1 多路数据的中心化与比例化 . . . . .	610
10.6.2 正则化与数据阵列的压缩 . . . . .	611
10.7 非负张量分解 . . . . .	613
10.7.1 非负张量分解的乘法算法 . . . . .	614
10.7.2 非负张量分解的交替最小二乘算法 . . . . .	617
本章小结 . . . . .	619
习题 . . . . .	619
参考文献 . . . . .	621
索引 . . . . .	648

# 第1章 矩阵代数基础

在科学与工程中，经常会遇到求解线性方程组的问题。矩阵是描述和求解线性方程组最基本和最有用的数学工具。矩阵不仅有很多基本的数学运算（如转置、内积、外积、逆矩阵、广义逆矩阵等），而且还有多种重要的标量函数（如范数、二次型、行列式、特征值、秩和迹），更包含多种特殊运算（如直和、直积、Hadamard 积、Kronecker 积、向量化）。本章将介绍矩阵代数的这些基本知识。

## 1.1 矩阵的基本运算

首先引出矩阵和向量的概念，给出本书中经常使用的基本符号。

### 1.1.1 矩阵与向量

在科学和工程中，经常会遇到  $m \times n$  线性方程组

$$\left. \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \end{array} \right\} \quad (1.1.1)$$

它使用  $m$  个方程描述  $n$  个未知量之间的线性关系。这一线性方程组很容易用矩阵-向量形式简记为

$$\mathbf{Ax} = \mathbf{b} \quad (1.1.2)$$

式中

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \quad (1.1.3)$$

称为  $m \times n$  矩阵，是一个按照长方阵列排列的复数或实数集合；而

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} \quad (1.1.4)$$

分别为  $n \times 1$  向量和  $m \times 1$  向量，是按照列方式排列的复数或实数集合，统称列向量。

类似地，按照行方式排列的复数或实数集合称为行向量。例如， $1 \times n$  行向量为

$$\mathbf{a} = [a_1, \dots, a_n] \quad (1.1.5)$$

为了区分实数或复数矩阵, 常令  $\mathbb{R}$  和  $\mathbb{C}$  分别表示实数和复数的集合,  $\mathbb{R}^{m \times n}$  和  $\mathbb{C}^{m \times n}$  分别表示所有  $m \times n$  实数和复数矩阵的向量空间。于是, 有矩阵的下列符号表示

$$\mathbf{A} \in \mathbb{R}^{m \times n} \iff \mathbf{A} = [a_{ij}] = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}, \quad a_{ij} \in \mathbb{R} \quad (1.1.6)$$

$$\mathbf{A} \in \mathbb{C}^{m \times n} \iff \mathbf{A} = [a_{ij}] = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}, \quad a_{ij} \in \mathbb{C} \quad (1.1.7)$$

当  $m = n$  时, 称矩阵  $\mathbf{A}$  为正方矩阵 (square matrix); 若  $m < n$ , 则称矩阵  $\mathbf{A}$  为宽矩阵 (broad matrix); 当  $m > n$  时, 便称矩阵  $\mathbf{A}$  为高矩阵 (tall matrix)。

在物理问题的建模中, 矩阵  $\mathbf{A}$  往往是物理系统 (如线性系统、滤波器、无线信道等) 的符号表示; 而科学和工程中遇到的向量可分为以下三种<sup>[255]</sup>:

- (1) 物理向量 泛指既有幅值, 又有方向的物理量, 如速度、加速度、位移等。
- (2) 几何向量 为了将物理向量可视化, 常用带方向的 (简称“有向”) 线段表示之。这种有向线段称为几何向量。例如,  $\mathbf{v} = \overrightarrow{AB}$  表示的有向线段, 其起点为  $A$ , 终点为  $B$ 。
- (3) 代数向量 几何向量可以用代数形式表示。例如, 若平面上的几何向量  $\mathbf{v} = \overrightarrow{AB}$  的起点坐标  $A = (a_1, a_2)$ , 终点坐标  $B = (b_1, b_2)$ , 则该几何向量可以表示为代数形式  $\mathbf{v} = \begin{bmatrix} b_1 - a_1 \\ b_2 - a_2 \end{bmatrix}$ 。这种用代数形式表示的几何向量称为代数向量。

图 1.1.1 归纳了向量的分类。

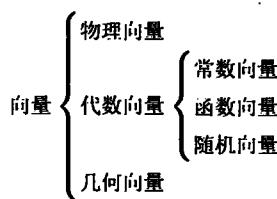


图 1.1.1 向量的分类

根据元素取值种类的不同, 代数向量又可分为以下三种:

- (1) 常数向量 向量的元素全部为实常数或者复常数, 如  $\mathbf{a} = [1, 5, 4]^T$  等。
- (2) 函数向量 向量的元素包含了函数值, 如  $\mathbf{x} = [1, x^2, \dots, x^n]^T$  等。
- (3) 随机向量 向量的元素为随机变量或随机过程, 如  $\mathbf{x}(n) = [x_1(n), \dots, x_m(n)]^T$ , 其中  $x_1(n), \dots, x_m(n)$  是  $m$  个随机过程或随机信号。

实际应用中遇到的往往是物理向量, 而几何向量是物理向量的可视化, 代数向量则可看作是物理向量的运算化工具。

若令

$$\mathbf{a}_1 = \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} a_{12} \\ \vdots \\ a_{m2} \end{bmatrix}, \quad \dots, \quad \mathbf{a}_n = \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix} \quad (1.1.8)$$

则矩阵  $\mathbf{A}$  可以用列向量记作

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \quad (1.1.9)$$

一个  $n \times n$  正方矩阵  $\mathbf{A}$  的主对角线是指从左上角到右下角沿  $i = j, j = 1, \dots, n$  相连接的线段。位于主对角线上的元素称为  $\mathbf{A}$  的对角元素，它们是  $a_{ii}, i = 1, \dots, n$ 。

矩阵  $\mathbf{A}$  从右上角到左下角沿

$$(i, n - i + 1), \quad i = 1, 2, \dots, n$$

相连接的线段称为矩阵  $\mathbf{A}$  的交叉对角线 (也称次对角线)。

主对角线以外元素全部为零的  $n \times n$  矩阵称为对角矩阵，记作

$$\mathbf{D} = \text{diag}(d_{11}, \dots, d_{nn}) \quad (1.1.10)$$

若对角矩阵主对角线元素全部等于 1，则称其为单位矩阵，用符号  $\mathbf{I}_{n \times n}$  示之。所有元素为零的  $m \times n$  矩阵称为零矩阵，记为  $\mathbf{O}_{m \times n}$ 。

一个全部元素为零的向量称为零向量。当维数已经明了或者不紧要时，常省去单位矩阵、零矩阵和零向量表示维数的下标，将它们分别简记为  $\mathbf{I}, \mathbf{O}$  和  $\mathbf{0}$ 。

只有一个元素为 1，其他元素皆等于 0 的列向量称为基本向量，即

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \dots, \quad \mathbf{e}_n = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \quad (1.1.11)$$

显然， $n \times n$  单位矩阵  $\mathbf{I}$  可以用  $n$  个基本向量表示为  $\mathbf{I} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n]$ 。

在本书中，我们经常会用到以下矩阵符号：

$\mathbf{A}(i, :)$ :  $\mathbf{A}$  的第  $i$  行；

$\mathbf{A}(:, j)$ :  $\mathbf{A}$  的第  $j$  列；

$\mathbf{A}(p : q, r : s)$ : 由  $\mathbf{A}$  的第  $p$  行到第  $q$  行，第  $r$  列到第  $s$  列组成的  $(q - p + 1) \times (s - r + 1)$  子矩阵。例如

$$\mathbf{A}(3 : 6, 2 : 4) = \begin{bmatrix} a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \\ a_{52} & a_{53} & a_{54} \\ a_{62} & a_{63} & a_{64} \end{bmatrix}$$

分块矩阵是一个以矩阵作元素的矩阵

$$\mathbf{A} = [\mathbf{A}_{ij}] = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1n} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{m1} & \mathbf{A}_{m2} & \cdots & \mathbf{A}_{mn} \end{bmatrix}$$

### 1.1.2 矩阵的基本运算

矩阵的基本运算包括矩阵的转置、共轭、共轭转置、加法和乘法。

**定义 1.1.1** 若  $A = [a_{ij}]$  是一个  $m \times n$  矩阵，则  $A$  的转置记作  $A^T$ ，是一个  $n \times m$  矩阵，其元素定义为  $[A^T]_{ij} = a_{ji}$ ；矩阵  $A$  的复数共轭  $A^*$  仍然是一个  $m \times n$  矩阵，其元素定义为  $[A^*]_{ij} = a_{ij}^*$ ；而矩阵  $A$  的(复)共轭转置记作  $A^H$ ，它是一个  $n \times m$  矩阵，定义为

$$A^H = \begin{bmatrix} a_{11}^* & a_{21}^* & \cdots & a_{m1}^* \\ a_{12}^* & a_{22}^* & \cdots & a_{m2}^* \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n}^* & a_{2n}^* & \cdots & a_{mn}^* \end{bmatrix} \quad (1.1.12)$$

共轭转置又叫 Hermitian 伴随、Hermitian 转置或 Hermitian 共轭。

满足  $A^T = A$  的正方实矩阵和  $A^H = A$  的正方复矩阵分别称为对称矩阵和 Hermitian 矩阵(复共轭对称矩阵)。

共轭转置与转置之间存在下列关系：

$$A^H = (A^*)^T = (A^T)^* \quad (1.1.13)$$

一个  $m \times n$  分块矩阵  $A$  的共轭转置是一个由  $A$  的每个分块矩阵的共轭转置组成的  $n \times m$  分块矩阵

$$A^H = \begin{bmatrix} A_{11}^H & A_{21}^H & \cdots & A_{m1}^H \\ A_{12}^H & A_{22}^H & \cdots & A_{m2}^H \\ \vdots & \vdots & \ddots & \vdots \\ A_{1n}^H & A_{2n}^H & \cdots & A_{mn}^H \end{bmatrix}$$

列向量的转置结果为行向量，行向量的转置结果为列向量。由于书中遇到的大多数向量为列向量，为节省书写的空间，本书采用转置符号 T 将  $m \times 1$  列向量记作  $x = [x_1, \dots, x_m]^T$ 。

矩阵最简单的代数运算是两个矩阵的加法、矩阵与一个标量的乘法。

**定义 1.1.2** 两个  $m \times n$  矩阵  $A = [a_{ij}]$  和  $B = [b_{ij}]$  之和记作  $A + B$ ，定义为  $[A + B]_{ij} = a_{ij} + b_{ij}$ 。

**定义 1.1.3** 令  $A = [a_{ij}]$  是一个  $m \times n$  矩阵，且  $\alpha$  是一个标量。乘积  $\alpha A$  是一个  $m \times n$  矩阵，定义为  $[\alpha A]_{ij} = \alpha a_{ij}$ 。

定义 1.1.3 可以推广为矩阵与向量的乘积、矩阵与矩阵的乘积。

**定义 1.1.4**  $m \times n$  矩阵  $A = [a_{ij}]$  与  $r \times 1$  向量  $x = [x_1, \dots, x_r]^T$  的乘积  $Ax$  只有当  $n = r$  时才存在，它是一个  $m \times 1$  向量，定义为

$$[Ax]_i = \sum_{j=1}^n a_{ij}x_j, \quad i = 1, \dots, m$$

**定义 1.1.5**  $m \times n$  矩阵  $A = [a_{ij}]$  与  $r \times s$  矩阵  $B = [b_{ij}]$  的乘积  $AB$  只有当  $n = r$  时才存在, 它是一个  $m \times s$  矩阵, 定义为

$$[AB]_{ij} = \sum_{k=1}^n a_{ik} b_{kj}, \quad i = 1, \dots, m; \quad j = 1, \dots, s$$

根据定义, 容易验证矩阵的加法服从下面的运算法则:

(1) 加法交换律 (commutative law of addition)  $A + B = B + A$

(2) 加法结合律 (associative law of addition)  $(A + B) + C = A + (B + C)$

**定理 1.1.1** 矩阵的乘积服从下面的运算法则:

(1) 乘法结合律 (associative law of multiplication) 若  $A \in \mathbb{C}^{m \times n}$ ,  $B \in \mathbb{C}^{n \times p}$ ,  $C \in \mathbb{C}^{p \times q}$ , 则  $A(BC) = (AB)C$ .

(2) 乘法左分配律 (left distributive law of multiplication) 若  $A$  和  $B$  是两个  $m \times n$  矩阵, 且  $C$  是一个  $n \times p$  矩阵, 则  $(A + B)C = AC + BC$ .

(3) 乘法右分配律 (right distributive law of multiplication) 若  $A$  是一个  $m \times n$  矩阵, 并且  $B$  和  $C$  是两个  $n \times p$  矩阵, 则  $A(B + C) = AB + AC$ .

(4) 若  $\alpha$  是一个标量, 并且  $A$  和  $B$  是两个  $m \times n$  矩阵, 则  $\alpha(A + B) = \alpha A + \alpha B$ .

**证明** 这里只证明 (1) 和 (2), 其他部分的证明留给读者作练习。

(1) 令  $A_{m \times n} = [a_{ij}]$ ,  $B_{n \times p} = [b_{ij}]$ ,  $C_{p \times q} = [c_{ij}]$ , 则

$$\begin{aligned} [A(BC)]_{ij} &= \sum_{k=1}^n a_{ik} (BC)_{kj} = \sum_{k=1}^n a_{ik} \left[ \sum_{l=1}^p b_{kl} c_{lj} \right] \\ &= \sum_{l=1}^p \sum_{k=1}^n (a_{ik} b_{kl}) c_{lj} = \sum_{l=1}^p [AB]_{il} c_{lj} = [(AB)C]_{ij} \end{aligned}$$

即有  $A(BC) = (AB)C$ 。

(2) 由矩阵的乘法知

$$[AC]_{ij} = \sum_{k=1}^n a_{ik} c_{kj}, \quad [BC]_{ij} = \sum_{k=1}^n b_{ik} c_{kj}$$

再由矩阵的加法, 得

$$[AC + BC]_{ij} = [AC]_{ij} + [BC]_{ij} = \sum_{k=1}^n (a_{ik} + b_{ik}) c_{kj} = [(A + B)C]_{ij}$$

故有  $(A + B)C = AC + BC$ . ■

一般说来, 矩阵的乘法不满足交换律, 即  $AB \neq BA$ .

令向量  $x = [x_1, x_2, \dots, x_n]^T$  和  $y = [y_1, y_2, \dots, y_n]^T$ , 矩阵与向量的乘积  $Ax = y$  可视为向量  $x$  的线性变换。此时,  $n \times n$  矩阵  $A$  称为线性变换矩阵。若向量  $y$  到  $x$  的线性逆变换  $A^{-1}$  存在, 则

$$x = A^{-1}y \tag{1.1.14}$$

这一方程可视为在原线性变换  $\mathbf{A}\mathbf{x} = \mathbf{y}$  两边左乘  $\mathbf{A}^{-1}$  之后得到的结果  $\mathbf{A}^{-1}\mathbf{A}\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$ 。因此，线性逆变换  $\mathbf{A}^{-1}$  应该满足  $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$  之关系。另一方面， $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$  也应该是可逆的，即两边左乘  $\mathbf{A}$  后得到的  $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{A}^{-1}\mathbf{y}$  应该与原线性变换  $\mathbf{A}\mathbf{x} = \mathbf{y}$  一致，故  $\mathbf{A}^{-1}$  还应该满足  $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$ 。

综合以上讨论，可以得到逆矩阵的定义如下。

**定义 1.1.6** 令  $\mathbf{A}$  是一个  $n \times n$  矩阵。称矩阵  $\mathbf{A}$  可逆，若可以找到一个  $n \times n$  矩阵  $\mathbf{A}^{-1}$  满足  $\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$ ，并称  $\mathbf{A}^{-1}$  是矩阵  $\mathbf{A}$  的逆矩阵。

下面是共轭、转置、共轭转置和逆矩阵的性质。

(1) 矩阵的共轭、转置和共轭转置满足分配律

$$(\mathbf{A} + \mathbf{B})^* = \mathbf{A}^* + \mathbf{B}^*, \quad (\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T, \quad (\mathbf{A} + \mathbf{B})^H = \mathbf{A}^H + \mathbf{B}^H$$

(2) 矩阵乘积的转置、共轭转置和逆矩阵满足关系式

$$\begin{aligned} (\mathbf{AB})^T &= \mathbf{B}^T \mathbf{A}^T, & (\mathbf{AB})^H &= \mathbf{B}^H \mathbf{A}^H \\ (\mathbf{AB})^{-1} &= \mathbf{B}^{-1} \mathbf{A}^{-1} & (\mathbf{A}, \mathbf{B} \text{ 为可逆的正方矩阵}) \end{aligned}$$

(3) 共轭、转置和共轭转置等符号均可与求逆符号交换，即有

$$(\mathbf{A}^*)^{-1} = (\mathbf{A}^{-1})^*, \quad (\mathbf{A}^T)^{-1} = (\mathbf{A}^{-1})^T, \quad (\mathbf{A}^H)^{-1} = (\mathbf{A}^{-1})^H$$

因此，常常分别采用紧凑的数学符号  $\mathbf{A}^{-*}$ ,  $\mathbf{A}^{-T}$  和  $\mathbf{A}^{-H}$ 。

(4) 对于任意矩阵  $\mathbf{A}$ ，矩阵  $\mathbf{B} = \mathbf{A}^H \mathbf{A}$  都是 Hermitian 矩阵。若  $\mathbf{A}$  可逆，则对于 Hermitian 矩阵  $\mathbf{B} = \mathbf{A}^H \mathbf{A}$ ，有  $\mathbf{A}^{-H} \mathbf{B} \mathbf{A}^{-1} = \mathbf{A}^{-H} \mathbf{A}^H \mathbf{A} \mathbf{A}^{-1} = \mathbf{I}$ 。

在一些应用中，常常涉及一个  $n \times n$  矩阵  $\mathbf{A}$  与它自身的乘积，从中可以引出两个重要的概念。

**定义 1.1.7** 矩阵  $\mathbf{A}_{n \times n}$  称为幂等矩阵 (idempotent matrix)，若  $\mathbf{A}^2 = \mathbf{AA} = \mathbf{A}$ 。

幂等矩阵  $\mathbf{A}$  具有以下性质 [403]：

(1)  $\mathbf{A}^n = \mathbf{A}$  对于  $n = 1, 2, 3, \dots$  成立。

(2)  $\mathbf{I} - \mathbf{A}$  为幂等矩阵 (注意： $\mathbf{A} - \mathbf{I}$  不一定是幂等矩阵)。

(3)  $\mathbf{A}^H$  为幂等矩阵。

(4)  $\mathbf{I} - \mathbf{A}^H$  为幂等矩阵。

(5) 若  $\mathbf{B}$  也为幂等矩阵，并且  $\mathbf{AB} = \mathbf{BA}$ ，则  $\mathbf{AB}$  为幂等矩阵。

(6)  $\mathbf{A}(\mathbf{I} - \mathbf{A}) = \mathbf{O}$  (零矩阵)。

(7)  $(\mathbf{I} - \mathbf{A})\mathbf{A} = \mathbf{O}$  (零矩阵)。

(8) 函数  $f(s\mathbf{I} + t\mathbf{A}) = (\mathbf{I} - \mathbf{A})f(s) + \mathbf{A}f(s + t)$ 。

**定义 1.1.8** 矩阵  $\mathbf{A}_{n \times n}$  称为对合矩阵 (involutory matrix) 或幂单矩阵 (unipotent matrix)，若  $\mathbf{A}^2 = \mathbf{AA} = \mathbf{I}$ 。

若  $\mathbf{A}$  为对合或者幂单矩阵，则函数  $f(\cdot)$  具有以下性质<sup>[403]</sup>

$$f(s\mathbf{I} + t\mathbf{A}) = \frac{1}{2}[(\mathbf{I} + \mathbf{A})f(s+t) + (\mathbf{I} - \mathbf{A})f(s-t)] \quad (1.1.15)$$

幂等矩阵与对合矩阵的关系：矩阵  $\mathbf{A}$  是对合矩阵，当且仅当  $\frac{1}{2}(\mathbf{A} + \mathbf{I})$  为幂等矩阵。

$n \times n$  矩阵  $\mathbf{A}$  称为幂零矩阵 (nilpotent matrix)，若  $\mathbf{A}^2 = \mathbf{AA} = \mathbf{O}$  (零矩阵)。

若  $\mathbf{A}$  为幂零矩阵，则函数  $f(\cdot)$  具有以下性质<sup>[403]</sup>

$$f(s\mathbf{I} + t\mathbf{A}) = \mathbf{I}f(s) + t\mathbf{A}f'(s) \quad (1.1.16)$$

式中  $f'(s)$  表示函数  $f(s)$  的一阶导数。

除了上述矩阵的基本运算外，还可定义矩阵函数：三角函数<sup>[403]</sup>

$$\sin(\mathbf{A}) = \sum_{n=0}^{\infty} \frac{(-1)^n \mathbf{A}^{2n+1}}{(2n+1)!} = \mathbf{A} - \frac{1}{3!} \mathbf{A}^3 + \frac{1}{5!} \mathbf{A}^5 - \dots \quad (1.1.17)$$

$$\cos(\mathbf{A}) = \sum_{n=0}^{\infty} \frac{(-1)^n \mathbf{A}^{2n}}{(2n)!} = \mathbf{I} - \frac{1}{2!} \mathbf{A}^2 + \frac{1}{4!} \mathbf{A}^4 - \dots \quad (1.1.18)$$

以及矩阵的指数函数和对数函数<sup>[328, 198]</sup>

$$e^{\mathbf{A}} = \sum_{n=0}^{\infty} \frac{1}{n!} \mathbf{A}^n = \mathbf{I} + \mathbf{A} + \frac{1}{2} \mathbf{A}^2 + \frac{1}{3!} \mathbf{A}^3 + \dots \quad (1.1.19)$$

$$e^{-\mathbf{A}} = \sum_{n=0}^{\infty} \frac{1}{n!} (-1)^n \mathbf{A}^n = \mathbf{I} - \mathbf{A} + \frac{1}{2} \mathbf{A}^2 - \frac{1}{3!} \mathbf{A}^3 + \dots \quad (1.1.20)$$

$$e^{\mathbf{At}} = \mathbf{I} + \mathbf{At} + \frac{1}{2} \mathbf{A}^2 t^2 + \frac{1}{3!} \mathbf{A}^3 t^3 + \dots \quad (1.1.21)$$

$$\ln(\mathbf{I} + \mathbf{A}) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n} \mathbf{A}^n = \mathbf{A} - \frac{1}{2} \mathbf{A}^2 + \frac{1}{3} \mathbf{A}^3 - \dots \quad (1.1.22)$$

如果矩阵  $\mathbf{A}$  的元素  $a_{ij}$  都是参数  $t$  的函数，则矩阵的导数定义为

$$\frac{d\mathbf{A}}{dt} = \dot{\mathbf{A}} = \begin{bmatrix} \frac{da_{11}}{dt} & \frac{da_{12}}{dt} & \dots & \frac{da_{1n}}{dt} \\ \frac{da_{21}}{dt} & \frac{da_{22}}{dt} & \dots & \frac{da_{2n}}{dt} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{da_{m1}}{dt} & \frac{da_{m2}}{dt} & \dots & \frac{da_{mn}}{dt} \end{bmatrix} \quad (1.1.23)$$

同样可定义矩阵的高阶导数。

矩阵的积分定义为

$$\int \mathbf{A} dt = \begin{bmatrix} \int a_{11} dt & \int a_{12} dt & \dots & \int a_{1n} dt \\ \int a_{21} dt & \int a_{22} dt & \dots & \int a_{2n} dt \\ \vdots & \vdots & \ddots & \vdots \\ \int a_{m1} dt & \int a_{m2} dt & \dots & \int a_{mn} dt \end{bmatrix} \quad (1.1.24)$$

同样也可定义矩阵的多重积分。

指数矩阵函数的导数定义为

$$\frac{d e^{At}}{dt} = A e^{At} = e^{At} A \quad (1.1.25)$$

矩阵乘积的导数定义为

$$\frac{d}{dt}(AB) = \frac{dA}{dt}B + A\frac{dB}{dt} \quad (1.1.26)$$

其中,  $A$  和  $B$  都是变量  $t$  的矩阵函数。

### 1.1.3 向量的线性无关性与非奇异矩阵

考查式 (1.1.1) 描述的  $m \times n$  线性方程组, 它可写成矩阵方程  $\mathbf{Ax} = \mathbf{b}$ 。若记  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$ , 则式 (1.1.1) 的  $m$  个方程可以合并写成标量与向量乘积之和

$$\mathbf{a}_1x_1 + \dots + \mathbf{a}_nx_n = \mathbf{b}$$

并称为列向量  $\mathbf{a}_1, \dots, \mathbf{a}_n$  的线性组合。

**定义 1.1.9** 一组  $m$  维向量  $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$  称为线性无关, 若方程

$$c_1\mathbf{u}_1 + \dots + c_n\mathbf{u}_n = \mathbf{0}$$

只有零解  $c_1 = \dots = c_n = 0$ 。若能够找到一组不全部为零的系数  $c_1, \dots, c_n$  使得上述方程成立, 则称  $m$  维向量组  $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$  线性相关。

向量的线性无关可以准确地描述什么样的  $n \times n$  线性方程组  $\mathbf{Ax} = \mathbf{b}$  具有唯一的非零解  $\mathbf{x}$ 。

一个  $n \times n$  矩阵  $\mathbf{A}$  是非奇异的, 当且仅当矩阵方程  $\mathbf{Ax} = \mathbf{0}$  只有零解  $\mathbf{x} = \mathbf{0}$ 。若  $\mathbf{Ax} = \mathbf{0}$  存在非零解  $\mathbf{x} \neq \mathbf{0}$ , 则矩阵  $\mathbf{A}$  是奇异的。

由于线性方程组  $\mathbf{Ax} = \mathbf{0}$  等价为

$$\mathbf{a}_1x_1 + \dots + \mathbf{a}_nx_n = \mathbf{0}$$

式中,  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$ 。由定义 1.1.9 立即可以得出结论: 当且仅当矩阵  $\mathbf{A}$  的列向量  $\mathbf{a}_1, \dots, \mathbf{a}_n$  线性无关时, 矩阵方程  $\mathbf{Ax} = \mathbf{0}$  只有零解  $\mathbf{x} = \mathbf{0}$ , 即矩阵  $\mathbf{A}$  是非奇异的。由于这一结果的重要性, 现用定理形式叙述之。

**定理 1.1.2**  $n \times n$  矩阵  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  是非奇异的, 当且仅当它的  $n$  个列向量  $\mathbf{a}_1, \dots, \mathbf{a}_n$  线性无关。

综上所述,  $n \times n$  矩阵  $\mathbf{A}$  的非奇异性可以根据其列向量的线性无关性或矩阵方程  $\mathbf{Ax} = \mathbf{b}$  存在唯一非零解或矩阵方程  $\mathbf{Ax} = \mathbf{0}$  只有零解加以判断。

## 1.2 矩阵的初等变换

涉及矩阵行与行之间的简单运算称为初等行运算。事实上，矩阵的初等行运算往往可以解决一些重要问题。例如，只使用初等行运算，就可以解决矩阵方程求解、矩阵求逆和矩阵基本空间的基向量构造等复杂问题。

### 1.2.1 初等行变换与阶梯型矩阵

**定义 1.2.1** 令矩阵  $A \in \mathbb{C}^{m \times n}$  的  $m$  个行向量分别为  $r_1, \dots, r_m$ 。下列运算称为矩阵  $A$  的初等行运算 (elementary row operation) 或初等行变换 (elementary row transformation):

- (1) 互换矩阵的任意两行，如  $r_p \leftrightarrow r_q$ ，称为 I 型初等行变换。
- (2) 一行元素同乘一个非零常数  $\alpha$ ，如  $\alpha r_p \rightarrow r_p$ ，称为 II 型初等行变换。
- (3) 将第  $p$  行元素同乘一个非零常数  $\beta$  后，加给第  $q$  行，即  $\beta r_p + r_q \rightarrow r_q$ ，称为 III 型初等行变换。

若矩阵  $A_{m \times n}$  经过一系列初等行运算，变换成为矩阵  $B_{m \times n}$ ，则称矩阵  $A$  和  $B$  为行等价矩阵 (row equivalent matrix)。

一个非零行最左边的非零元素称为该行的首项元素 (leading entry)。如果首项元素等于 1，便称为首一元素 (leading 1 entry)。

从矩阵方程的求解以及基本空间的基向量构造等应用出发，常常希望将一个矩阵经过初等行运算之后，变换为阶梯型矩阵。

**定义 1.2.2** 一个  $m \times n$  矩阵称为阶梯型 (echelon form) 矩阵，若：

- (1) 全部由零组成的所有行都位于矩阵的底部。
- (2) 每一个非零行的首项元素总是出现在上一个非零行的首项元素的右边。
- (3) 首项元素下面的同列元素全部为零。

例如，下面是阶梯型矩阵的几个例子

$$A = \begin{bmatrix} 2 & * & * \\ 0 & 5 & * \\ 0 & 0 & 3 \\ 0 & 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & * & * \\ 0 & 3 & * \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 3 & * & * \\ 0 & 0 & 5 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & * \\ 0 & 0 & 9 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

式中，\* 表示该元素可以为任意值。

**定义 1.2.3**<sup>[255]</sup> 一个阶梯型矩阵  $A$  称为行简约阶梯型 (row reduced echelon form, RREF)，若  $A$  的每一非零行的首项元素等于 1 (即为首一元素)，并且每一个首一元素也是它所在列唯一的非零元素。

行简约阶梯型也称行阶梯标准型或 Hermite 标准型。

给定一个  $m \times n$  矩阵  $B$ ，下面的算法通过初等行变换将  $B$  化成行简约阶梯型矩阵。

**算法 1.2.1** 将  $m \times n$  矩阵化成行简约阶梯型 <sup>[255]</sup>

步骤1 将含有一个非零元素的列设定为最左边的第1列。

步骤2 如果需要，将第1行与其他行互换，以使第1个非零列在第1行有一个非零元素。

步骤3 如果第1行的首项元素为 $a$ ，则将该行的所有元素乘以 $1/a$ ，以使该行的首项元素等于1，成为首一元素。

步骤4 通过初等行变换，将其他行位于第1行首一元素下面的全部元素变成0。

步骤5 对第 $i=2, 3, \dots, m$ 行依次重复以上步骤，以使每一行的首一元素出现在上一行的首一元素的右边，并使与第 $i$ 行首一元素同列的其他各行元素都变为0。

**定理 1.2.1** 任何一个矩阵 $A_{m \times n}$ 都与一个并且唯一的一个行简约阶梯型矩阵是行等价的。

**证明** 参见文献[306, Appendix A]。

当矩阵的初等行变换产生一个行阶梯型矩阵时，若将行阶梯型矩阵进一步简化为行简约阶梯型，则相应的初等行变换将不会改变行阶梯型矩阵各非零行首项元素的位置。也就是说，任何一个矩阵的行阶梯型的首项元素与行简约阶梯型的首一元素总是处于相同的位置。由此可以引出下面的定义。

**定义 1.2.4**<sup>[306, p.15]</sup> 矩阵 $A_{m \times n}$ 的主元位置(pivot position)就是矩阵 $A$ 中与其阶梯型的首项元素相对应的位置。矩阵 $A$ 中包含主元位置的每一列都称为 $A$ 的主元列(pivot column)。

下面的例子说明如何通过初等行运算，将一个矩阵变换为行阶梯型和行简约阶梯型，以及如何判断原矩阵的主元列。

**例 1.2.1** 已知 $3 \times 5$ 矩阵

$$A = \begin{bmatrix} -3 & 6 & -1 & 1 & -7 \\ 1 & -2 & 2 & 3 & -1 \\ 2 & -4 & 5 & 8 & -4 \end{bmatrix}$$

第2行乘-2，加到第3行；并且第2行乘3，加到第1行，则

$$A \sim \begin{bmatrix} 0 & 0 & 5 & 10 & -10 \\ 1 & -2 & 2 & 3 & -1 \\ 0 & 0 & 1 & 2 & -2 \end{bmatrix}$$

第1行乘 $-2/5$ ，加到第2行；同时第1行乘 $-1/5$ ，加到第3行，得

$$A \sim \begin{bmatrix} 0 & 0 & 5 & 10 & -10 \\ 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

交换第1行和第2行，又得到

$$A \sim \begin{bmatrix} 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 5 & 10 & -10 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{阶梯型})$$

下面画横杠的元素所在的位置称为主元位置。因此，矩阵  $A$  的主元列为第 1 列和第 3 列，即有

$$\left[ \begin{array}{c} -3 \\ 1 \\ 2 \end{array} \right], \quad \left[ \begin{array}{c} -1 \\ 2 \\ 5 \end{array} \right] \quad (A \text{ 的主元列})$$

进一步地，阶梯型矩阵的第 2 行乘以  $1/5$ ，行阶梯型简化为

$$A \sim \left[ \begin{array}{ccccc} 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 1 & 2 & -2 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right] \quad (\text{行简约阶梯型})$$

### 1.2.2 初等行变换的两个应用

下面介绍初等行变换的两个重要应用：矩阵方程求解和矩阵求逆。

#### 1. 矩阵方程求解

考查  $n \times n$  矩阵方程  $Ax = b$  的求解，其中矩阵  $A$  存在逆矩阵  $A^{-1}$ 。现在，希望通过初等行变换，得到方程的解  $x = A^{-1}b$ 。

$n \times n$  矩阵方程  $Ax = b$  中的  $A$  和  $b$  分别称作数据矩阵和数据向量，而  $x$  称为未知数（或未知参数）向量。为了方便讨论矩阵方程的求解，常将数据矩阵和数据向量组合成一个  $n \times (n+1)$  维的新矩阵  $B = [A, b]$ ，并称为矩阵方程  $Ax = b$  的增广矩阵。

注意到矩阵方程的解  $x = A^{-1}b$  也可以写成矩阵方程的形式  $Ix = A^{-1}b$ ，其对应的增广矩阵为  $[I, A^{-1}b]$ 。于是，我们可以将矩阵方程的这一求解过程与它们对应的增广矩阵形式分别书写为

$$\begin{array}{ll} \text{方程求解} & Ax = b \xrightarrow{\text{初等行变换}} x = A^{-1}b \\ \text{增广矩阵} & [A, b] \xrightarrow{\text{初等行变换}} [I, A^{-1}b] \end{array}$$

这表明，若对增广矩阵  $[A, b]$  使用初等行变换，使得左边变成一个  $n \times n$  维单位矩阵，则变换后的增广矩阵的第  $n+1$  列给出原矩阵方程的解  $x = A^{-1}b$ 。这样一种求解矩阵方程的初等行变换方法称为高斯消去（Gauss elimination）法或 Gauss-Jordan 消去法。

#### 例 1.2.2 用高斯消去法求解线性方程组

$$x_1 + x_2 + 2x_3 = 6$$

$$3x_1 + 4x_2 - x_3 = 5$$

$$-x_1 + x_2 + x_3 = 2$$

对其增广矩阵进行初等行变换

$$\begin{array}{c}
 \left[ \begin{array}{cccc} 1 & 1 & 2 & 6 \\ 3 & 4 & -1 & 5 \\ -1 & 1 & 1 & 2 \end{array} \right] \xrightarrow{\text{第2行减去第1行的3倍}} \left[ \begin{array}{cccc} 1 & 1 & 2 & 6 \\ 0 & 1 & -7 & -13 \\ -1 & 1 & 1 & 2 \end{array} \right] \xrightarrow{\text{第1行加到第3行}} \left[ \begin{array}{cccc} 1 & 1 & 2 & 6 \\ 0 & 1 & -7 & -13 \\ 0 & 2 & 3 & 8 \end{array} \right] \\
 \left[ \begin{array}{cccc} 1 & 1 & 2 & 6 \\ 0 & 1 & -7 & -13 \\ 0 & 2 & 3 & 8 \end{array} \right] \xrightarrow{\text{第1行减去第2行}} \left[ \begin{array}{cccc} 1 & 0 & 9 & 19 \\ 0 & 1 & -7 & -13 \\ 0 & 2 & 3 & 8 \end{array} \right] \xrightarrow{\text{第3行减去第2行的2倍}} \left[ \begin{array}{cccc} 1 & 0 & 9 & 19 \\ 0 & 1 & -7 & -13 \\ 0 & 0 & 1 & 2 \end{array} \right] \\
 \left[ \begin{array}{cccc} 1 & 0 & 9 & 19 \\ 0 & 1 & -7 & -13 \\ 0 & 0 & 17 & 34 \end{array} \right] \xrightarrow{\text{第3行乘以 } 1/17} \left[ \begin{array}{cccc} 1 & 0 & 9 & 19 \\ 0 & 1 & -7 & -13 \\ 0 & 0 & 1 & 2 \end{array} \right] \xrightarrow{\text{第1行减去第3行的9倍}} \left[ \begin{array}{cccc} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array} \right]
 \end{array}$$

即通过高斯消去法得到方程组的解为  $x_1 = 1$ ,  $x_2 = 1$  和  $x_3 = 2$ 。

初等行变换方法也适用于  $m \times n$  矩阵方程  $\mathbf{Ax} = \mathbf{b}$  的求解。此时, 需要将增广矩阵化成行简约阶梯型矩阵。具体算法如下。

### 算法 1.2.2 $m \times n$ 矩阵方程 $\mathbf{Ax} = \mathbf{b}$ 的求解 [255]

步骤 1 构造增广矩阵  $\mathbf{B} = [\mathbf{A}, \mathbf{b}]$ 。

步骤 2 使用算法 1.2.1 将增广矩阵  $\mathbf{B}$  变成行简约阶梯型, 它与原增广矩阵等价。

步骤 3 从简化的矩阵得到对应的线性方程组, 它与原线性方程组等价。

步骤 4 得到新的线性方程组的通解 (general solution)。

### 例 1.2.3 考查线性方程组及其增广矩阵

$$\begin{aligned}
 2x_1 + 2x_2 - x_3 &= 1 \\
 -2x_1 - 2x_2 + 4x_3 &= 1 \\
 2x_1 + 2x_2 + 5x_3 &= 5 \\
 -2x_1 - 2x_2 - 2x_3 &= -3
 \end{aligned}
 \quad \text{和} \quad \mathbf{B} = \left[ \begin{array}{cccc} 2 & 2 & -1 & 1 \\ -2 & -2 & 4 & 1 \\ 2 & 2 & 5 & 5 \\ -2 & -2 & -2 & -3 \end{array} \right]$$

第 1 行元素乘以  $1/2$ , 使第 1 个元素为 1

$$\left[ \begin{array}{cccc} 2 & 2 & -1 & 1 \\ -2 & -2 & 4 & 1 \\ 2 & 2 & 5 & 5 \\ -2 & -2 & -2 & -3 \end{array} \right] \rightarrow \left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ -2 & -2 & 4 & 1 \\ 2 & 2 & 5 & 5 \\ -2 & -2 & -2 & -3 \end{array} \right]$$

利用初等行变换, 使第 2~4 行的第 1 个元素都变成 0

$$\left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ -2 & -2 & 4 & 1 \\ 2 & 2 & 5 & 5 \\ -2 & -2 & -2 & -3 \end{array} \right] \rightarrow \left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 3 & 2 \\ 0 & 0 & 6 & 4 \\ 0 & 0 & -3 & -2 \end{array} \right]$$

第 2 行元素乘以  $1/3$ , 使得其第 3 列元素等于 1, 即有

$$\left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 3 & 2 \\ 0 & 0 & 6 & 4 \\ 0 & 0 & -3 & -2 \end{array} \right] \rightarrow \left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 & \frac{2}{3} \\ 0 & 0 & 6 & 4 \\ 0 & 0 & -3 & -2 \end{array} \right]$$

利用初等行变换, 使第 2 行首项元素 1 的上边和下边的元素全部变为 0, 得到

$$\left[ \begin{array}{cccc} 1 & 1 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 & \frac{2}{3} \\ 0 & 0 & 6 & 4 \\ 0 & 0 & -3 & -2 \end{array} \right] \rightarrow \left[ \begin{array}{cccc} 1 & 1 & 0 & \frac{5}{6} \\ 0 & 0 & 1 & \frac{2}{3} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right]$$

对应的线性方程组为  $x_1 + x_2 = \frac{5}{6}$  和  $x_3 = \frac{2}{3}$ 。该方程组有无穷多组解, 其通解为  $x_1 = \frac{5}{6} - x_2, x_3 = \frac{2}{3}$ 。若  $x_2 = 1$ , 则得一特解 (particular solution) 为  $x_1 = -\frac{1}{6}, x_2 = 1$  和  $x_3 = \frac{2}{3}$ 。

考察齐次线性方程组 (homogeneous linear system of equations)

$$\left. \begin{array}{l} a_{11}x_1 + \cdots + a_{1n}x_n = 0 \\ \vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n = 0 \end{array} \right\} \quad (1.2.1)$$

显然,  $\mathbf{x} = [0, 0, \dots, 0]^T$  是任何齐次线性方程组的一个解。零向量解称为平凡解 (trivial solution)。平凡解以外的任何其他解称为非平凡解 (nontrivial solution)。

任何一个复矩阵方程  $A_{m \times n}x_{n \times 1} = b_{m \times 1}$  都可以写为

$$(A_r + j A_i)(x_r + j x_i) = b_r + j b_i \quad (1.2.2)$$

式中,  $A_r, x_r, b_r$  和  $A_i, x_i, b_i$  分别代表  $A, x, b$  的实部和虚部。展开上式, 得

$$A_r x_r - A_i x_i = b_r \quad (1.2.3)$$

$$A_i x_r + A_r x_i = b_i \quad (1.2.4)$$

利用矩阵分块形式, 上式可合并为

$$\left[ \begin{array}{cc} A_r & -A_i \\ A_i & A_r \end{array} \right] \left[ \begin{array}{c} x_r \\ x_i \end{array} \right] = \left[ \begin{array}{c} b_r \\ b_i \end{array} \right] \quad (1.2.5)$$

于是, 含  $n$  个复未知数的  $m$  个复方程转变为含  $2n$  个实未知数的  $2m$  个实方程。

特别地, 若  $m = n$ , 则有

$$\begin{array}{ll} \text{复矩阵方程求解} & Ax = b \xrightarrow{\text{初等行变换}} x = A^{-1}b \\ \text{实增广矩阵} & \left[ \begin{array}{ccc} A_r & -A_i & b_r \\ A_i & A_r & b_i \end{array} \right] \xrightarrow{\text{初等行变换}} \left[ \begin{array}{ccc} I_n & O_n & x_r \\ O_n & I_n & x_i \end{array} \right] \end{array}$$

这表明, 若将复矩阵  $A \in \mathbb{C}^{n \times n}$  和复向量  $b \in \mathbb{C}^n$  排成  $2n \times (2n+1)$  增广矩阵, 并且利用初等行变换将增广矩阵的左边变成  $2n \times 2n$  单位矩阵, 则最右边的  $2n \times 1$  列向量的上、下一半分别给出复矩阵方程的解向量  $x$  的实部和虚部。

## 2. 矩阵求逆的高斯消去法

考虑  $n \times n$  非奇异矩阵  $A$  的求逆。这个问题也可以建模成一个矩阵方程  $AX = I$ , 因为该方程的解  $X = A^{-1}$  就是矩阵  $A$  的逆矩阵。易知, 矩阵方程  $AX = I$  的增广矩阵为  $[A, I]$ , 而其解  $X = A^{-1}$  或解方程  $IX = A^{-1}$  的增广矩阵为  $[I, A^{-1}]$ 。于是, 我们有下面的初等行变换关系

$$\begin{array}{ll} \text{方程求解} & AX = I \xrightarrow{\text{初等行变换}} X = A^{-1} \\ \text{增广矩阵} & [A, I] \xrightarrow{\text{初等行变换}} [I, A^{-1}] \end{array}$$

这意味着, 我们只要对  $n \times 2n$  维增广矩阵  $[A, I]$  进行初等行变换, 使得其左边一半变成  $n \times n$  维单位矩阵, 则其右边另外一半即给出  $n \times n$  维矩阵  $A$  的逆矩阵  $A^{-1}$ 。这一初等行变换方法就是矩阵求逆的高斯消去法。

若复矩阵  $A \in \mathbb{C}^{n \times n}$  非奇异, 则求其逆矩阵的问题可以建模成复矩阵方程  $(A_r + jA_i)(X_r + jX_i) = I$ 。这一复矩阵方程又可以改写为以下形式

$$\begin{bmatrix} A_r & -A_i \\ A_i & A_r \end{bmatrix} \begin{bmatrix} X_r \\ X_i \end{bmatrix} = \begin{bmatrix} I_n \\ O_n \end{bmatrix} \quad (1.2.6)$$

由此立即得初等行变换关系

$$\begin{array}{ll} \text{复矩阵方程求解} & AX = I \xrightarrow{\text{初等行变换}} X = A^{-1} \\ \text{实增广矩阵} & \begin{bmatrix} A_r & -A_i & I_n \\ A_i & A_r & O_n \end{bmatrix} \xrightarrow{\text{初等行变换}} \begin{bmatrix} I_n & O_n & X_r \\ O_n & I_n & X_i \end{bmatrix} \end{array}$$

也就是说, 只要对  $2n \times 3n$  维增广矩阵进行初等行变换, 使得其左边变成  $2n \times 2n$  维单位矩阵, 则其右边  $2n \times n$  矩阵的上、下一半即分别给出  $n \times n$  复矩阵  $A$  的逆矩阵  $A^{-1}$  的实部和虚部矩阵。

### 1.2.3 初等列变换

**定义 1.2.5** 令矩阵  $A \in \mathbb{C}^{m \times n}$  的  $n$  个列向量分别为  $a_1, \dots, a_n$ 。下列运算称为矩阵  $A$  的初等列运算 (elementary column operation) 或初等列变换 (elementary column transformation):

- (1) 互换矩阵的任意两列, 如  $a_p \leftrightarrow a_q$ , 称为 I 型初等列变换。
- (2) 一列元素同乘一个非零常数  $\alpha$ , 如  $\alpha a_p \rightarrow a_p$ , 称为 II 型初等列变换。

注意, 初等列变换不包括第  $p$  列乘以一个非零常数后, 加到第  $q$  列, 因为这一运算将改变解向量中第  $p$  个元素的结构。

若  $m \times n$  矩阵  $A$  经过一系列初等列运算, 变换成为矩阵  $B$ , 则称矩阵  $A$  和  $B$  为列等价矩阵 (column equivalent matrix)。

值得注意的是, 求解矩阵方程  $Ax = b$  时, 通常对增广矩阵  $[A, b]$  进行初等行变换。这一变换对方程的解  $x$  没有任何影响。然而, 初等列变换只适用于数据矩阵  $A$ , 并且初等列变换将改变方程的解  $x$  的元素的排列顺序和大小。例如, 对于矩阵方程

$$A_{m \times n} x_{n \times 1} = [a_1, \dots, a_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \sum_{i=1}^n a_i x_i = b_{m \times 1} \quad (1.2.7)$$

交换矩阵  $A$  的两列, 例如  $a_p$  和  $a_q$  互换时, 解向量  $x$  的元素  $x_p$  和  $x_q$  也必须互换位置:  $A$  的第  $p$  列乘以某个常数  $\alpha \neq 0$ , 则  $x$  的第  $p$  个元素为  $x_p / \alpha$ 。这两种现象分别称为解向量元素的排序和幅值的不确定性或模糊性。不过, 这两种不确定性在盲信号分离中又是允许的: 因为从观测数据  $y = Ax$  中将混合的信号分离开, 是主要目的, 而对这些信号的排序并不特别关心。一个分离的信号相差某个固定的复值因子, 从信号的保真角度讲, 也是允许的, 因为固定的相位差通过信号处理的方法进行补偿之后, 一个波形被放大或者缩小某个尺度, 并不影响波形的保真。

**定义 1.2.6** 一个  $m \times n$  矩阵称为列阶梯型 (column echelon form) 矩阵, 若:

- (1) 全部由零组成的所有列都位于矩阵的最右边。
- (2) 每一个非零列的首项元素总是出现在左边一个非零列的首项元素的右边。
- (3) 首项元素右面的同行元素全部为零。

例如, 下面是列阶梯型矩阵的两个例子

$$A = \begin{bmatrix} 2 & 0 & 0 \\ * & 5 & 0 \\ * & * & 0 \\ * & * & 0 \end{bmatrix}, \quad A = \begin{bmatrix} 5 & 0 & 0 & 0 \\ * & 2 & 0 & 0 \\ * & * & 0 & 0 \\ * & * & 5 & 0 \\ * & * & * & 7 \end{bmatrix}$$

### 1.3 向量空间、线性映射与 Hilbert 空间

虽然许多工程问题也可以不使用线性空间进行研究, 但是线性空间的使用却可以给问题的描述带来诸多的方便。客观地讲, 线性空间在本质上是某一类事物在矩阵代数里的一个抽象的集合表示, 线性映射或线性变换则反映线性空间中元素间最基本的线性联系, 它们为线性函数的研究提供了极大的方便。可以说, 线性代数就是研究线性空间和线性变换理论的数学分支。例如, 一个  $2 \times 1$  向量  $[x_0, y_0]^T$  可以想象成用笛卡儿坐标  $x, y$  表示的平面上的某个点。类似地, 一个  $3 \times 1$  向量  $[x_0, y_0, z_0]^T$  可认为是三维空间的一个点。我们生活的实际世界就是一个典型的三维空间。依此类推, 一个  $n \times 1$  向量可视为  $n$  维空间的一个点。因此,  $n$  维空间很自然地是所有  $n \times 1$  向量的集合。显然,  $n (> 3)$  维空间是一维空间 (即直线)、二维空间 (即平面) 和三维空间的推广。

### 1.3.1 集合的基本概念

在引出向量空间和子空间的定义之前，先介绍集合的有关概念。顾名思义，集合就是某些元素的集体表示。

集合通常用花括号表示为  $S = \{\cdot\}$ ，花括号内为集合  $S$  的元素。如果集合的元素只有几个，通常便在花括号内罗列出所有的元素，例如  $S = \{a, b, c, d\}$ 。若  $S$  是满足某种性质  $P(x)$  的元素  $x$  的集合，则记为  $S = \{x|P(x)\}$ 。只有一个元素  $\alpha$  的集合称为单元素集 (singleton)，记作  $\{\alpha\}$ 。

下面是与集合运算有关的几个常用数学符号：

$\forall$  表示“对所有 …”；

$x \in A$  读作“ $x$  属于集合  $A$ ”，意即  $x$  是集合  $A$  的一个元素；

$x \notin A$  表示  $x$  不是集合  $A$  的元素；

$\exists$  代表“使得”；

$\exists$  意即“存在”；

$A \Rightarrow B$  表示“若有条件  $A$ ，则有结果  $B$ ”或“ $A$  意味着  $B$ ”。

例如，“在集合  $V$  中存在一个零元素  $\theta$ ，使得加法  $x + \theta = x = \theta + x$  对于  $V$  中的所有元素  $x$  均成立”这一冗长的叙述，便可用上述符号简洁地表示为

$$\exists \theta \in V \quad \exists \quad x + \theta = x = \theta + x, \quad \forall x \in V$$

令  $A$  和  $B$  为集合，则集合有以下基本运算。

符号  $A \subseteq B$  读作“集合  $A$  包含于集合  $B$ ”或“ $A$  是  $B$  的一个子集”，意指  $A$  的每一个元素都是  $B$  的元素，即  $x \in A \Rightarrow x \in B$ 。

若  $A \subset B$ ，则称  $A$  是  $B$  的一个真子集。符号  $B \supset A$  读作“ $B$  包含  $A$ ”或“ $B$  是  $A$  的超集 (superset)”。没有任何元素的集合记作  $\emptyset$ ，称为空集。

符号  $A = B$  读作“集合  $A$  等于集合  $B$ ”，意即  $A \subseteq B$  且  $B \subseteq A$ ，或  $x \in A \Leftrightarrow x \in B$  (集合  $A$  的元素一定是集合  $B$  的元素，反之亦然)。 $A = B$  的否定写作  $A \neq B$ ，意即  $A$  不属于  $B$ ，反过来  $B$  也不属于  $A$ 。

$A$  和  $B$  的并集 (union) 记作  $A \cup B$ ，定义为

$$X = A \cup B = \{x \in X | x \in A \text{ 或 } x \in B\} \quad (1.3.1)$$

它表示并集  $X$  的元素由属于集合  $A$  或  $B$  的元素组合而成。

集合  $A$  和  $B$  的交集 (intersection) 用符号  $A \cap B$  表示，定义为

$$X = A \cap B = \{x \in X | x \in A \text{ 和 } x \in B\} \quad (1.3.2)$$

即交集的元素由  $A$  和  $B$  共有的元素构成。

符号  $Z = A + B$  表示集合  $A$  和  $B$  的和集, 定义为

$$Z = A + B = \{z = x + y \in Z \mid x \in A, y \in B\} \quad (1.3.3)$$

即和集的元素由  $A$  的元素与  $B$  的元素之和组成。

集合差 (set-theoretic difference) “ $A$  减  $B$ ” 定义为

$$X = A - B = \{x \in X \mid x \in A, \text{ 但 } x \notin B\} \quad (1.3.4)$$

也称差集。差集  $A - B$  的元素由  $A$  中所有不属于  $B$  的元素组成。差集也常用符号  $X = A \setminus B$  表示。

子集合  $A$  在集合  $X$  中的补集 (complement) 定义为

$$A^c = X - A = \{x \in X \mid x \notin A\} \quad (1.3.5)$$

若  $X$  和  $Y$  为集合, 且  $x \in X$  和  $y \in Y$ , 则所有有序对 (ordered pair)  $(x, y)$  的集合记为  $X \times Y$ , 称作集合  $X$  和  $Y$  的笛卡儿积, 即

$$X \times Y = \{(x, y) \mid x \in X, y \in Y\} \quad (1.3.6)$$

类似地,  $X_1 \times X_2 \times \cdots \times X_n$  表示  $n$  个集合  $X_1, X_2, \dots, X_n$  的笛卡儿积, 其元素为有序  $n$  元组 (ordered  $n$ -uples)  $(x_1, x_2, \dots, x_n)$ 。

上述集合的有关概念及符号, 在本书中将经常用到。例如, 一个以矩阵  $\mathbf{X} \in \mathbb{R}^{n \times n}$  和  $\mathbf{Y} \in \mathbb{R}^{n \times n}$  为变元的标量函数  $f(\mathbf{X}, \mathbf{Y})$  即可用笛卡儿积记作  $f : \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times n} \rightarrow \mathbb{R}$ 。

### 1.3.2 向量空间

**定义 1.3.1 (向量空间)** [22, 40, 255] 以向量为元素的集合  $V$  称为向量空间, 若加法运算定义为两个向量之间的加法, 乘法运算定义为向量与标量域  $S$  中的标量之间的乘法, 并且对于向量集合  $V$  中的向量  $\mathbf{x}, \mathbf{y}, \mathbf{w}$  和标量域  $S$  中的标量  $a_1, a_2$ , 以下两个闭合性和关于加法及乘法的 8 个公理 (axiom) (也称公设 (postulate) 或定律 (law)) 均满足:

闭合性 (closure properties)

(c1) 若  $\mathbf{x} \in V$  和  $\mathbf{y} \in V$ , 则  $\mathbf{x} + \mathbf{y} \in V$ , 即  $V$  在加法下是闭合的, 简称加法的闭合性 (closure for addition);

(c2) 若  $a_1$  是一个标量,  $\mathbf{y} \in V$ , 则  $a_1 \mathbf{y} \in V$ , 即  $V$  在标量乘法下是闭合的, 简称标量乘法的闭合性 (closure for scalar multiplication)。

加法的公理

(a1)  $\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}, \forall \mathbf{x}, \mathbf{y} \in V$ , 称为加法交换律 (commutative law for addition);

(a2)  $\mathbf{x} + (\mathbf{y} + \mathbf{w}) = (\mathbf{x} + \mathbf{y}) + \mathbf{w}, \forall \mathbf{x}, \mathbf{y}, \mathbf{w} \in V$ , 称为加法结合律 (associative law for addition);

(a3) 在  $V$  中存在一个零向量  $\mathbf{0}$ , 使得对于任意向量  $\mathbf{y} \in V$ , 恒有  $\mathbf{y} + \mathbf{0} = \mathbf{y}$  (零向量的存在性);

(a4) 给定一个向量  $\mathbf{y} \in V$ , 存在另一个向量  $-\mathbf{y} \in V$  使得  $\mathbf{y} + (-\mathbf{y}) = \mathbf{0} = (-\mathbf{y}) + \mathbf{y}$  (负向量的存在性)。

标量乘法的公理

(s1)  $a(b\mathbf{y}) = (ab)\mathbf{y}$  对所有向量  $\mathbf{y}$  和所有标量  $a, b$  成立, 称为标量乘法结合律 (associative law for scalar multiplication);

(s2)  $a(\mathbf{x} + \mathbf{y}) = a\mathbf{x} + a\mathbf{y}$  对所有向量  $\mathbf{x}, \mathbf{y} \in V$  和标量  $a$  成立, 称为标量乘法分配律 (distributive law for scalar multiplication);

(s3)  $(a + b)\mathbf{y} = a\mathbf{y} + b\mathbf{y}$  对所有向量  $\mathbf{y}$  和所有标量  $a, b$  成立 (标量乘法分配律);

(s4)  $1\mathbf{y} = \mathbf{y}$  对所有  $\mathbf{y} \in V$  成立, 称为标量乘法单位律 (unity law for scalar multiplication)。

由于向量空间服从向量加法的交换律、结合律以及标量乘法的结合律、分配律, 所以定义 1.3.1 给出的向量空间为线性空间。

如果  $V$  中的向量为实向量, 并且标量域为实数域, 则称  $V$  是实向量空间。若  $V$  中的向量为复向量, 且标量域为复数域, 便称  $V$  为复向量空间。

如下面的定理所归纳的那样, 向量空间还有其他一些有用的性质。

**定理 1.3.1** 如果  $V$  是一个向量空间, 则:

- (1) 零向量  $\mathbf{0}$  是唯一的。
- (2) 对每一个向量  $\mathbf{y}$ , 加法的逆运算  $-\mathbf{y}$  是唯一的。
- (3) 对每一个向量  $\mathbf{y}$ , 恒有  $0\mathbf{y} = \mathbf{0}$ 。
- (4) 对每一个标量  $a$ , 恒有  $a\mathbf{0} = \mathbf{0}$ 。
- (5) 若  $a\mathbf{y} = \mathbf{0}$ , 则  $a = 0$  或者  $\mathbf{y} = \mathbf{0}$ 。
- (6)  $(-1)\mathbf{y} = -\mathbf{y}$ 。

**证明** 参见文献 [255, pp.365~366]。

$\mathbb{R}^n$  和  $\mathbb{C}^n$  是向量空间最重要的两个例子。

对于一个正整数  $n$ , 实数的所有有序  $n$  元组  $[x_1, \dots, x_n]$  的集合记为  $\mathbb{R}^n$ , 它的每一个元素称为向量 (均为  $n \times 1$  向量)。特别地, 若  $n = 1$ , 则  $\mathbb{R}$  的元素称为标量。如果对集合  $\mathbb{R}^n$  定义两个向量的加法和一个标量与一个向量的乘法, 则称  $\mathbb{R}^n$  为  $n$  阶实向量空间。

类似地, 若在复数的所有有序  $n$  元组的集合  $\mathbb{C}^n$  内定义向量加法和标量乘法, 则称  $\mathbb{C}^n$  为  $n$  阶复向量空间。

在很多场合, 我们并不对  $n$  阶向量空间  $\mathbb{R}^n$  或者  $\mathbb{C}^n$  中所有的向量组合感兴趣, 而只是关心向量空间中某个特定的向量子集  $W$ 。以  $\mathbb{R}^3$  的子集为例

$$W = \{\mathbf{x} | \mathbf{x} = [x_1, x_2, 0]^T, x_1 \text{ 和 } x_2 \text{ 为实数}\}$$

从几何的观点看,  $\mathbb{R}^3$  是一个三维空间, 而  $W$  是二维  $x-y$  平面, 可以用  $\mathbb{R}^2$  表示。

**定义 1.3.2** 令  $V$  和  $W$  是两个向量空间, 若  $W$  是  $V$  中一个非空的子集合, 则称子集合  $W$  是  $V$  的一个子空间。

显然,一个  $n$  维零向量是  $n$  阶向量空间的一个子空间。在本书中,假定子空间  $W$  是非空的,零向量只是子集合  $W$  中的一个元素,而非唯一的元素。

下面的定理提供了确定  $\mathbb{R}^n$  的子集合  $W$  是否为  $\mathbb{R}^n$  的子空间的一种简便方法。

**定理 1.3.2**  $\mathbb{R}^n$  的子集合  $W$  是  $\mathbb{R}^n$  的一个子空间,当且仅当以下三个条件均满足:

(1) 当向量  $\mathbf{x}, \mathbf{y}$  属于  $W$ ,则  $\mathbf{x} + \mathbf{y}$  也属于  $W$ ,即满足加法的闭合性:  $\mathbf{x}, \mathbf{y} \in W \Rightarrow (\mathbf{x} + \mathbf{y}) \in W$ 。

(2) 当  $\mathbf{x} \in W$ ,且  $a$  为标量,则  $a\mathbf{x}$  也属于  $W$ ,即满足与标量乘积的闭合性。

(3) 零向量  $\mathbf{0}$  是  $W$  的元素。

**证明** [255, p.169] 充分性证明。假定  $W$  是满足上述条件(1)和(2)的  $\mathbb{R}^n$  的子集合,为了证明  $W$  是  $\mathbb{R}^n$  的子空间,必须证明  $W$  满足向量空间的 10 个基本性质(定义 1.3.1)。条件(1)和(2)意味着子集合  $W$  满足定义 1.3.1 中的性质(c1)和(c2)。注意到  $W$  是  $V$  的子集合,所以  $W$  也满足定义 1.3.1 中的性质(a1), (a2), (s1), (s2), (s3) 和 (s4)。由于非空的子集合包含零向量,故(a3)满足。容易看出  $-\mathbf{x} = (-1)\mathbf{x}$ 。因此,如果  $\mathbf{x} \in W$ ,则由条件(2)知  $-\mathbf{x} \in W$ ,即定义 1.3.1 中的基本性质(a4)也满足。由于  $W$  满足定义 1.3.1 中的 10 个基本性质,故  $W$  是  $\mathbb{R}^n$  的子空间。

必要性证明。令  $W$  是  $\mathbb{R}^n$  的子空间,则性质(a3)意味着零向量是子空间  $W$  的一个元素。另外,子空间满足性质(c1)和(c2),意味着条件(1)和(2)在  $W$  中成立。■

**定义 1.3.3** 若  $A$  和  $B$  是向量空间  $V$  的两个向量子空间,则

$$A + B = \{\mathbf{x} + \mathbf{y} \mid \mathbf{x} \in A, \mathbf{y} \in B\} \quad (1.3.7)$$

称为子空间  $A$  和  $B$  的和,而

$$A \cap B = \{\mathbf{x} \in V \mid \mathbf{x} \in A \text{ 及 } \mathbf{x} \in B\} \quad (1.3.8)$$

称为子空间  $A$  和  $B$  的交。

**定义 1.3.4** 若  $A$  和  $B$  是向量空间  $V$  的两个子空间,并满足  $V = A + B$  和  $A \cap B = \{\mathbf{0}\}$ ,则称  $V$  是子空间  $A$  和  $B$  的直接求和,简称直和(direct sum),记作  $V = A \oplus B$ 。

**定理 1.3.3** 若  $A$  和  $B$  是向量空间  $V$  的向量子空间,则  $A + B$  和  $A \cap B$  也是  $V$  的向量子空间。

**证明** 由于向量子空间  $A$  和  $B$  都包含零向量,故  $\mathbf{0} = \mathbf{0} + \mathbf{0}$  表明  $A + B$  也包含零向量。若  $\mathbf{x}, \mathbf{x}' \in A$  和  $\mathbf{y}, \mathbf{y}' \in B$ ,且  $c$  为标量,则  $(\mathbf{x} + \mathbf{y}) + (\mathbf{x}' + \mathbf{y}') = (\mathbf{x} + \mathbf{x}') + (\mathbf{y} + \mathbf{y}') \in A + B$ (因为子空间  $A$  和  $B$  分别满足加法的闭合性),且  $c(\mathbf{x} + \mathbf{y}) = c\mathbf{x} + c\mathbf{y} \in A + B$ (因为子空间  $A$  和  $B$  均具有与标量乘法的闭合性)。这说明  $A + B$  也具有加法的闭合性和与标量乘法的闭合性,即  $A + B$  满足定理 1.3.2 的 3 个条件。因此,  $A + B$  是向量子空间。类似地,可以证明  $A \cap B$  也是向量子空间。■

**推论 1.3.1** 若  $A$  和  $B$  是向量空间  $V$  的子空间,则  $A + B$  是  $V$  中包含向量子空间  $A$  和  $B$  的最小向量子空间。

**推论 1.3.2** 若  $A$  和  $B$  是向量空间  $V$  的子空间，则子空间的交  $A \cap B$  是  $V$  中既属于  $A$ ，又属于  $B$  的最大向量子空间。

以上两个推论的证明参见文献 [40]。

### 1.3.3 线性映射

以上讨论了向量空间内向量的有关简单运算：向量加法、向量与标量的乘法，但尚未涉及两个向量空间之间的转换关系。然而，在自然科学、社会科学和数学的一些分支中，不同向量空间内向量之间的线性变换起着重要的作用。因此，为了研究两个向量空间之间的关系，有必要考虑能够实现从一个向量空间到另一个向量空间的转换关系的函数。事实上，在我们的日常生活中，也经常遇到这种转换。当我们欲将一幅图像变换为另一幅图像时，通常会移动它的位置，或者旋转它。例如，函数  $T(x, y) = (\alpha x, \beta y)$  就能够将图像的  $x$  坐标和  $y$  坐标改变尺度。根据  $\alpha$  和  $\beta$  大于 1 还是小于 1，图像就能够被放大或者缩小。

下面从线性映射的角度，对向量空间的结构做一番讨论。

映射本身就是一类函数，因此常使用一般函数通用的符号来表示映射。若令  $V$  是 Euclidean  $m$  空间  $\mathbb{R}^m$  内的子空间， $W$  是另一个不同维 Euclidean  $n$  空间  $\mathbb{R}^n$  内的子空间，则

$$T : V \mapsto W \quad (1.3.9)$$

称为子空间  $V$  到子空间  $W$  的映射（或函数、变换），它表示将子空间  $V$  的每一个向量变成子空间  $W$  的一个相对应向量的一种规则。于是，若  $v \in V$  和  $w \in W$ ，则向量  $w$  是  $v$  的映射或变换，即有

$$w = T(v) \quad (1.3.10)$$

并称子空间  $V$  是映射  $T$  的始集 (initial set) 或定义域 (domain)，称  $W$  是映射的终集 (final set) 或上域 (codomain)。

若  $v$  是向量空间  $V$  的某个向量，则  $T(v)$  称为向量  $v$  在映射  $T$  下的像 (image)，或映射  $T$  在点  $v$  的值 (value)，而  $v$  称为  $T(v)$  的原像。对于向量空间  $V$  的子空间  $A$ ，映射  $T(A)$  表示子空间  $A$  的元素 (即向量) 在映射  $T$  下的值的集合，写作

$$T(A) = \{T(v) | v \in A\} \quad (1.3.11)$$

特别地， $T(V)$  代表对  $V$  内所有向量的变换输出的集合，称为映射  $T$  的值域 (range)，其符号为

$$T(V) = \text{Im}(T) = \{T(v) | v \in V\} \quad (1.3.12)$$

一般地，映射  $T : V \mapsto W$  的值域  $\text{Im}(T)$  是  $W$  的一个子集合。如果  $\text{Im}(T) = W$ ，即映射的值域等于向量空间  $W$ ，则称  $T : V \mapsto W$  为满射 (surjective)。映射  $T : V \mapsto W$  称为单

(值映) 射 (injective), 若它将  $V$  的不同向量映射为  $W$  的不同向量, 即

$$\mathbf{v}_1, \mathbf{v}_2 \in V, \mathbf{v}_1 \neq \mathbf{v}_2 \Rightarrow T(\mathbf{v}_1) \neq T(\mathbf{v}_2)$$

或者

$$T(\mathbf{v}_1) = T(\mathbf{v}_2) \Rightarrow \mathbf{v}_1 = \mathbf{v}_2$$

特别地, 若映射  $T : V \mapsto W$  既是单射, 又是满射, 则称为一对一映射 (bijective)。一个一对一映射  $T : V \mapsto W$  存在逆映射  $T^{-1} : W \mapsto V$ 。逆映射  $T^{-1}$  的任务就是将映射  $T$  所做过的每一件事情恢复原状。因此, 若  $T(\mathbf{v}) = \mathbf{w}$ , 则  $T^{-1}(\mathbf{w}) = \mathbf{v}$ 。其结果是,  $T^{-1}(T(\mathbf{v})) = \mathbf{v}, \forall \mathbf{v} \in V$  和  $T(T^{-1}(\mathbf{w})) = \mathbf{w}, \forall \mathbf{w} \in W$ 。

矩阵与向量的乘法  $A_{m \times n} \mathbf{x}_{n \times 1}$  也可视为将  $\mathbb{C}^n$  的向量  $\mathbf{x}$  变换为  $\mathbb{C}^m$  的某个向量  $\mathbf{y} = A\mathbf{x}$  的映射  $T : \mathbf{x} \mapsto A\mathbf{x}$ , 故矩阵与一向量的乘法常称为该向量的矩阵变换 (matrix transformation)。

考察线性变换  $\mathbf{y} = T(\mathbf{x}) = r\mathbf{x}$ 。当  $0 < r < 1$  时, 称线性变换  $T(\mathbf{x}) = r\mathbf{x}$  为压缩映射 (contracting mapping), 因为  $T$  在  $\mathbf{x}$  的像点  $r\mathbf{x}$  的向量长度小于  $\mathbf{x}$  的长度。相反, 如果  $r > 1$ , 则称  $T(\mathbf{x}) = r\mathbf{x}$  为膨胀映射 (dilation mapping), 因为变换  $r\mathbf{x}$  的作用是将向量  $\mathbf{x}$  的长度拉伸。

**定义 1.3.5** 令  $V$  和  $W$  分别是  $\mathbb{R}^m$  和  $\mathbb{R}^n$  的子空间, 并且  $T : V \mapsto W$  是一映射。称  $T$  为线性映射 (linear mapping) 或线性变换 (linear transformation), 若对于  $\mathbf{v} \in V, \mathbf{w} \in W$  和所有标量  $c$ , 映射  $T$  满足叠加性

$$T(\mathbf{v} + \mathbf{w}) = T(\mathbf{v}) + T(\mathbf{w}) \quad (1.3.13)$$

和齐次性

$$T(c\mathbf{v}) = cT(\mathbf{v}) \quad (1.3.14)$$

定义中的两个关系式也可合并写作线性关系式

$$T(c_1\mathbf{v} + c_2\mathbf{w}) = c_1T(\mathbf{v}) + c_2T(\mathbf{w}) \quad (1.3.15)$$

即线性是叠加性和齐次性的合称。更一般地, 若  $\mathbf{u}_1, \dots, \mathbf{u}_p$  均为线性变换  $T$  的域, 反复使用式 (1.3.15), 则可以得到

$$T(c_1\mathbf{u}_1 + \dots + c_p\mathbf{u}_p) = c_1T(\mathbf{u}_1) + \dots + c_pT(\mathbf{u}_p) \quad (1.3.16)$$

这一公式在工程和物理中被称为叠加原理。如果  $\mathbf{u}_1, \dots, \mathbf{u}_p$  分别为某个系统或过程的输入信号向量, 则  $T(\mathbf{u}_1), \dots, T(\mathbf{u}_p)$  可分别视为该系统或过程的输出信号向量。识别一个系统是否为线性系统的判据是: 如果系统的输入为线性表达式  $\mathbf{y} = c_1\mathbf{u}_1 + \dots + c_p\mathbf{u}_p$ , 则当系统的输出也满足相同的线性关系  $T(\mathbf{y}) = T(c_1\mathbf{u}_1 + \dots + c_p\mathbf{u}_p) = c_1T(\mathbf{u}_1) + \dots + c_pT(\mathbf{u}_p)$  时, 该系统为线性系统。否则, 为非线性系统。

例 1.3.1 考查变换  $T : \mathbb{R}^3 \mapsto \mathbb{R}^2$

$$T_1(\mathbf{x}) = \begin{bmatrix} x_1 + x_2 \\ x_1^2 - x_2^2 \end{bmatrix}, \quad \text{其中, } \mathbf{x} = [x_1, x_2, x_3]^T$$

$$T_2(\mathbf{x}) = \begin{bmatrix} x_1 - x_2 \\ x_2 + x_3 \end{bmatrix}, \quad \text{其中, } \mathbf{x} = [x_1, x_2, x_3]^T$$

容易看出, 变换  $T_1 : \mathbb{R}^3 \mapsto \mathbb{R}^2$  不满足线性关系式, 故不是线性变换; 而变换  $T_2 : \mathbb{R}^3 \mapsto \mathbb{R}^2$  满足线性关系式, 为线性变换。

例 1.3.2 考虑线性算子  $T : \mathbb{R}^2 \mapsto \mathbb{R}^2$ , 它将平面上的向量  $\mathbf{x}$  映射为  $y$  轴上的正交投影  $\mathbf{w}$ , 参见图 1.3.1。这一线性算子称为正交投影算子。

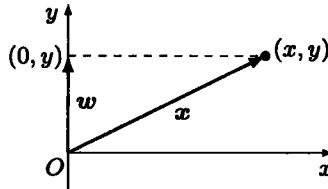


图 1.3.1 正交投影

由图 1.3.1, 可以写出与  $\mathbf{w} = T(\mathbf{x})$  的分量相关的方程为

$$w_1 = 0 = 0x + 0y$$

$$w_2 = y = 0x + 1y$$

或写成矩阵形式

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

由于是线性方程, 所以正交投影算子  $T(\mathbf{x})$  为线性变换, 相应的标准矩阵为  $A = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$ 。

线性子空间和线性映射之间存在下列内在联系。

**定理 1.3.4** [40,p.29] 令  $V$  和  $W$  是两个向量空间,  $T : V \mapsto W$  为一线性变换。

(1) 若  $M$  是  $V$  的线性子空间, 则  $T(M)$  是  $W$  的线性子空间;

(2) 若  $N$  是  $W$  的线性子空间, 则线性反变换  $T^{-1}(N)$  是  $V$  的线性子空间。

线性映射具有以下基本性质: 若  $T : V \mapsto W$  是一线性映射, 则

$$T(\mathbf{0}) = \mathbf{0} \quad \text{和} \quad T(-\mathbf{x}) = -T(\mathbf{x}) \tag{1.3.17}$$

特别地, 对于线性变换  $\mathbf{y} = A\mathbf{x}$ , 若已知变换矩阵  $A$ , 由输入向量  $\mathbf{x}$  求输出向量  $\mathbf{y}$ , 则称  $A\mathbf{x} = \mathbf{y}$  为正向问题 (forward problem); 反之, 若已知变换矩阵  $A$ , 由输出向量  $\mathbf{y}$  求输入向量  $\mathbf{x}$ , 则称  $A\mathbf{x} = \mathbf{y}$  为逆问题 (inverse problem)。正向问题的实质是矩阵-向量计算, 而逆问题的本质则是矩阵方程的求解。

两个具有相同结构的向量空间  $E$  和  $F$  称为同构 (isomorphic)，记作  $E \cong F$ 。两个实 (或复) 内积空间  $E$  和  $F$  同构，若存在一个一对一线性映射  $T : E \mapsto F$  能保持向量的内积不变，即  $\langle Tx, Ty \rangle = \langle x, y \rangle$  对所有向量  $x, y \in E$  成立。这样一种映射  $T$  称为向量空间的同构映射 (isomorphism)。

#### 1.3.4 内积空间、赋范空间与 Hilbert 空间

向量空间只定义了向量的加法以及标量与向量的乘法，并且向量空间的和、交与直和等也只涉及两个向量空间的元素 (即向量) 之间比较简单的关系。显然，向量之间的乘法也是必须考虑的一种基本运算。

令  $\mathbb{K}$  表示一标量域 (field of scalars)，它既可以是实数域  $\mathbb{R}$ ，也可以是复数域  $\mathbb{C}$ ，而  $V$  为一  $n$  维向量空间  $\mathbb{R}^n$  或  $\mathbb{C}^n$ 。

**定义 1.3.6** (内积与内积向量空间)<sup>[420,p.18]</sup> 若对所有  $x, y, z \in V$  和  $\alpha, \beta \in \mathbb{K}$ ，映射函数  $\langle \cdot, \cdot \rangle : V \times V \mapsto \mathbb{K}$  满足以下三条公理：

- (1) 共轭对称性  $\langle x, y \rangle = \langle y, x \rangle^*$ ,
- (2) 第一变元的线性性  $\langle \alpha x + \beta y, z \rangle = \alpha \langle x, z \rangle + \beta \langle y, z \rangle$ ,
- (3) 非负性  $\langle x, x \rangle \geq 0$ ，并且  $\langle x, x \rangle = 0 \Leftrightarrow x = 0$  (严格正性)，

则称  $\langle x, y \rangle$  为向量  $x$  与  $y$  的内积 (inner product)， $V$  为内积向量空间 (inner vector space)。

两个向量的内积可以度量它们之间的夹角

$$\cos \theta = \frac{\langle x, y \rangle}{\sqrt{\langle x, x \rangle} \sqrt{\langle y, y \rangle}} \quad (1.3.18)$$

满足内积三个公理的实向量空间和复向量空间分别称为实内积向量空间和复内积向量空间。

**注释 1** 对于实内积向量空间，共轭对称性退化为实对称性，因为  $\langle x, y \rangle = \langle y, x \rangle^* = \langle y, x \rangle$ 。

**注释 2** 第一变元的线性性包含了齐次性  $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$  和可加性  $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$ 。

**注释 3** 共轭对称性和第一变元的线性性意味着

$$\langle x, \alpha y \rangle = \langle \alpha y, x \rangle^* = \alpha^* \langle y, x \rangle^* = \alpha^* \langle x, y \rangle \quad (1.3.19)$$

$$\langle x, y + z \rangle = \langle y + z, x \rangle^* = \langle y, x \rangle^* + \langle z, x \rangle^* = \langle x, y \rangle + \langle x, z \rangle \quad (1.3.20)$$

内积向量空间具有向量的加法、标量与向量的乘法以及两个向量的乘法 (内积)，可以度量两个向量之间的夹角。如果还能够增加关于向量的长度 (size 或 length)、距离 (distance) 和邻域 (neighborhood) 等测度的话，那么向量空间无疑将更加实用和完美；而向量的范数能够担负这一重任。

**定义 1.3.7** (范数和赋范向量空间) 令  $V$  是一 (实或复) 向量空间。向量  $x$  的范数是一实函数  $p(x) : V \rightarrow \mathbb{R}$ ，若对所有向量  $x, y \in V$  和任意一个标量  $c \in \mathbb{K}$  (其中  $\mathbb{K}$  表示

$\mathbb{R}$  或者  $\mathbb{C}$ ), 下面的公理全部成立:

- (1) 非负性:  $p(\mathbf{x}) \geq 0$ , 并且  $p(\mathbf{x}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{0}$ ;
- (2) 齐次性:  $p(c\mathbf{x}) = |c| \cdot p(\mathbf{x})$  对所有复常数  $c$  成立;
- (3) 三角不等式:  $p(\mathbf{x} + \mathbf{y}) \leq p(\mathbf{x}) + p(\mathbf{y})$ 。

并称  $V$  为赋范向量空间 (normed vector space)。

最常用的向量范数为 Euclidean 范数或者  $L_2$  范数, 记作  $\|\cdot\|_2$ , 定义为

$$\|\mathbf{x}\|_E = \|\mathbf{x}\|_2 = \sqrt{x_1^2 + \cdots + x_m^2} \quad (1.3.21)$$

$L_2$  范数可以直接度量一个向量  $\mathbf{x}$  的长度  $\text{size}(\mathbf{x}) = \|\mathbf{x}\|_2$ , 两个向量之间的距离

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 \quad (1.3.22)$$

以及一个向量的  $\epsilon$  邻域 (其中  $\epsilon > 0$ )

$$N_\epsilon(\mathbf{x}) = \{\mathbf{y} \mid \|\mathbf{y} - \mathbf{x}\|_2 \leq \epsilon\} \quad (1.3.23)$$

另外, 还有两种不完全满足范数三个公理的向量范数。

**定义 1.3.8** <sup>[476]</sup> 向量  $\mathbf{x} \in V$  的半范数 (seminorm) 又叫伪范数 (pseudo-norm), 定义为: 若对所有向量  $\mathbf{x}, \mathbf{y} \in V$  和任意一个标量  $c$ , 满足条件

- (1)  $p(\mathbf{x}) \geq 0$ ;
- (2)  $p(c\mathbf{x}) = |c| \cdot p(\mathbf{x})$ ;
- (3)  $p(\mathbf{x} + \mathbf{y}) \leq p(\mathbf{x}) + p(\mathbf{y})$ .

注意, 半范数与范数的唯一区别是: 半范数不完全满足范数的第 1 个公理, 有可能  $\mathbf{x} \neq \mathbf{0}$  时  $p(\mathbf{x}) = 0$ 。例如, 容易验证  $p(\mathbf{x}) = x_1 + \cdots + x_n$  是零均值向量  $\mathbf{x}$  的半范数, 但半范数  $p(\mathbf{x}) = 0$  并不意味着  $\mathbf{x} = \mathbf{0}$ 。

所有的范数都是半范数, 但半范数不一定是范数。

**定义 1.3.9** <sup>[476]</sup> 向量  $\mathbf{x} \in V$  的拟范数 (quasi-norm) 定义为: 若对所有向量  $\mathbf{x}, \mathbf{y} \in V$  和任意一个标量  $c$ , 满足:

- (1)  $p(\mathbf{x}) \geq 0$ , 且  $p(\mathbf{x}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{0}$ ;
- (2)  $p(c\mathbf{x}) = |c| \cdot p(\mathbf{x})$ ;
- (3)  $p(\mathbf{x} + \mathbf{y}) \leq C(p(\mathbf{x}) + p(\mathbf{y}))$ , 其中  $C \neq 1$  为某个正实数。

可见, 拟范数不严格满足范数公理中的三角不等式, 只满足  $C$  不等式  $p(\mathbf{x} + \mathbf{y}) \leq C(p(\mathbf{x}) + p(\mathbf{y}))$ 。同一种定义公式有时给出拟范数, 有时则给出范数, 取决于参数的不同。例如, 容易验证

$$\|\mathbf{x}\|_p = \left( \sum_{i=1}^m x_i^p \right)^{1/p} \quad (1.3.24)$$

是拟范数 (若  $0 < p < 1$ ) 或范数 (若  $p \geq 1$ )。

本书将采用符号  $\|\cdot\|$  统一表示向量和矩阵的各种范数。

**定义 1.3.10 (完备性)** 一个向量空间  $V$  称为完备向量空间, 若对于  $V$  中的每一个 Cauchy 序列  $\{v_n\}_{n=1}^{\infty} \subset V$ , 在向量空间  $V$  内存在一个元素  $v$ , 使得  $\lim_{n \rightarrow \infty} v_n \rightarrow v$ , 即  $V$  内的每一个 Cauchy 序列都收敛在向量空间  $V$  内。特别地, 一个向量空间  $V$  称为相对于范数完备的向量空间, 若对于每一个 Cauchy 序列  $\{v_n\}_{n=1}^{\infty} \subset V$ , 在向量空间  $V$  内存在一个元素  $v$ , 使得依范数收敛  $\lim_{n \rightarrow \infty} \|v_n\| \rightarrow \|v\|$  满足。

向量空间元素的任何一个 Cauchy 序列依范数收敛为空间内的一个元素  $\lim_{n \rightarrow \infty} \|v_n\| \rightarrow \|v\|$  也可等价叙述为: 二者之差的范数趋于零, 即  $\lim_{n \rightarrow \infty} \|v_n - v\| \rightarrow 0$ 。

**定义 1.3.11 (Banach 空间)**<sup>[340]</sup> 一个赋范向量空间  $V$  称为 Banach 空间, 若对每一个 Cauchy 序列  $\{v_n\}_{n=1}^{\infty} \subset V$ , 在  $V$  内存在一个元素  $v$ , 使得  $\lim_{n \rightarrow \infty} v_n \rightarrow v$ 。

一个有限维的赋范线性向量空间一定是 Banach 空间, 因为它会自动满足 Cauchy 序列的收敛条件。

**定义 1.3.12 (Hilbert 空间)** 一个相对于范数完备即满足范数收敛  $\lim_{n \rightarrow \infty} \|v_n\| \rightarrow \|v\|$  的赋范向量空间  $V$  称为 Hilbert 空间。

显然, 一个 Hilbert 空间一定是 Banach 空间, 但一个 Banach 空间不一定是 Hilbert 空间。这是因为, 范数收敛  $\lim_{n \rightarrow \infty} \|v_n\| \rightarrow \|v\|$  一定满足极限收敛  $\lim_{n \rightarrow \infty} v_n \rightarrow v$ , 但极限收敛  $\lim_{n \rightarrow \infty} v_n \rightarrow v$  不一定意味着范数收敛。

表 1.3.1 汇总了几种向量空间的比较。

表 1.3.1 几种向量空间的比较

向量空间	定义了向量的加法和向量的数乘, 以向量为元素的集合 $\mathbb{R}^n$ 或 $\mathbb{C}^n$
内积向量空间	定义了内积 $\langle x, y \rangle$ (向量的乘法) 的向量空间
赋范向量空间	定义了范数 $\ x\ $ 的向量空间, 可度量向量的长度、距离与邻域
Banach 空间	满足 $\lim_{n \rightarrow \infty} v_n \rightarrow v, \forall v_n, v \in \mathbb{C}^n$ 的完备赋范向量空间
Hilbert 空间	满足 $\lim_{n \rightarrow \infty} \ v_n\  \rightarrow \ v\ , \forall v_n, v \in \mathbb{C}^n$ 的完备赋范向量空间
Euclidean 空间	具有 Euclidean 范数 $\ x\ _2$ 的赋范向量空间

**定义 1.3.13 (伴随算子)** 令  $T$  是 Hilbert 空间  $H$  内的有界线性算子。若  $\langle Tx, y \rangle = \langle x, T^*y \rangle$  对所有向量  $x, y \in H$  成立, 则称  $T^*$  是  $T$  的伴随算子 (adjoint operator)。

表 1.3.2 列出了几种常用的有界线性算子  $T$  及其伴随算子  $T^*$ 。

在矩阵代数中, 伴随算子与 Hermitian 算子即复共轭转置算子常被等同对待。若  $T = T^*$ , 则称  $T$  是自伴随的 (self-adjoint)。当讲到自伴随算子时, 总是指有界的线性自伴随算子。

线性代数研究有限维向量空间之间的线性映射关系, 而无限维向量空间之间的映射关系的研究称为线性函数分析 (linear functional analysis) 或线性算子理论 (linear operator theory)<sup>[282]</sup>。

表 1.3.2 几种常用算子及伴随算子

算子 $T$	伴随算子 $T^*$
矩阵乘法 $\mathbf{AB}$	矩阵复共轭转置乘法 $\mathbf{A}^H \mathbf{B}$
卷积 $\mathbf{x}(n) * \mathbf{y}(n) = \sum_{i=1}^{\infty} x_i(n)y^*(n-i)$	互相关 $E\{\mathbf{x}(n)\mathbf{y}^H(n)\}$
补零 (zero padding)	截尾 (truncation)
衍射建模 (diffraction modeling)	偏移成像 (imaging by migration)

## 1.4 内积与范数

1.3 节分别介绍了向量的内积和范数必须满足的公理，并引出了内积向量空间、赋范向量空间、Banach 空间和 Hilbert 空间的定义。本节将分别具体讨论向量和矩阵的内积及范数的定义及有关性质。

### 1.4.1 向量的内积与范数

$n$  阶复向量  $\mathbf{x} = [x_1, \dots, x_n]^T, \mathbf{y} = [y_1, \dots, y_n]^T$  之间的内积

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^H \mathbf{y} = \sum_{i=1}^n x_i^* y_i \quad (1.4.1)$$

称为典范内积 (canonical inner product)。采用典范内积的有限维向量空间  $\mathbb{R}^n$  或者  $\mathbb{C}^n$  习惯称为  $n$  阶 Euclidean 空间或者 Euclidean  $n$  空间。

注意，在一些文献中，也常用以下典范内积形式

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}^* = \sum_{i=1}^n x_i y_i^* \quad (1.4.1)$$

另外，还经常使用加权内积

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^H \mathbf{G} \mathbf{y} \quad (1.4.2)$$

其中，加权矩阵  $\mathbf{G}$  为正定的 Hermitian 矩阵，即满足条件  $\mathbf{x}^H \mathbf{G} \mathbf{x} > 0, \forall \mathbf{x} \in \mathbb{C}^n$ 。

令  $x(t), y(t)$  是复数域  $\mathbb{C}$  的两个连续函数，并且  $t$  的定义域为  $[a, b]$ ，则  $x(t)$  和  $y(t)$  之间的内积定义为

$$\langle x(t), y(t) \rangle \stackrel{\text{def}}{=} \int_a^b x^*(t) y(t) dt \quad (1.4.3)$$

可以验证，该内积满足内积的三个公理，所以实数域  $\mathbb{R}$  是一维内积空间，但是它不是 Euclidean 空间，因为实数域不是有限维的。

**例 1.4.1** 序列  $\{e^{j2\pi f n}\}_{n=0}^{N-1}$  是一个以单位时间间隔被采样的频率为  $f$  的正弦波。复正弦波向量  $\mathbf{e}_n(f)$  定义为  $(n+1) \times 1$  向量，即  $\mathbf{e}_n(f) = \left[1, e^{j(\frac{2\pi}{n+1})f}, \dots, e^{j(\frac{2\pi}{n+1})nf}\right]^T$ 。这样

一来,  $N$  个数据样本  $x(n), n = 0, 1, \dots, N - 1$  的离散 Fourier 变换 (DFT) 就可以用向量的典范内积表示为

$$X(f) = \sum_{n=0}^{N-1} x(n)e^{-j(\frac{2\pi}{N})nf} = e_{N-1}^H \mathbf{x} = \langle e_{N-1}, \mathbf{x} \rangle$$

其中,  $\mathbf{x} = [x(0), x(1), \dots, x(N-1)]^T$  常称为数据向量。

下面是内积空间的范数具备的一般性质。

**定理 1.4.1** <sup>[40]</sup> 在实或复内积空间里, 范数具有以下性质:

- (1)  $\|\mathbf{0}\| = 0$ , 并且  $\|\mathbf{x}\| > 0, \forall \mathbf{x} \neq \mathbf{0}$ ;
- (2)  $\|c\mathbf{x}\| = |c| \cdot \|\mathbf{x}\|$  对所有向量  $\mathbf{x}$  和标量  $c$  成立;
- (3) 范数服从极化恒等式 (polarization identity)

$$\langle \mathbf{x}, \mathbf{y} \rangle = \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2), \quad \forall \mathbf{x}, \mathbf{y} \quad (\text{实内积空间}) \quad (1.4.4)$$

$$\begin{aligned} \langle \mathbf{x}, \mathbf{y} \rangle &= \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2 - j\|\mathbf{x} + j\mathbf{y}\|^2 + j\|\mathbf{x} - j\mathbf{y}\|^2) \\ &\quad \forall \mathbf{x}, \mathbf{y} \quad (\text{复内积空间}) \end{aligned} \quad (1.4.5)$$

- (4) 范数满足平行四边形法则 (parallelogram law)

$$\|\mathbf{x} + \mathbf{y}\|^2 + \|\mathbf{x} - \mathbf{y}\|^2 = 2(\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2), \quad \forall \mathbf{x}, \mathbf{y} \quad (1.4.6)$$

- (5) 范数满足三角不等式  $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|, \forall \mathbf{x}, \mathbf{y}$ ;

- (6) 范数服从 Cauchy-Schwartz 不等式

$$|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \cdot \|\mathbf{y}\| \quad (1.4.7)$$

等号  $|\langle \mathbf{x}, \mathbf{y} \rangle| = \|\mathbf{x}\| \cdot \|\mathbf{y}\|$  成立, 当且仅当  $\mathbf{y} = c\mathbf{x}$ , 其中,  $c$  为某个非零常数。

下面分别介绍常数向量、函数向量和随机向量的内积与范数。

### 1. 常数向量的典范内积与范数

常数向量的内积通常采用典范内积, 而常用的向量范数有以下几种。

- (1)  $L_0$  范数 (也称 0 范数)

$$\|\mathbf{x}\|_0 \stackrel{\text{def}}{=} \text{非零元素的个数} \quad (1.4.8)$$

- (2)  $L_1$  范数 (也称和范数或 1 范数)

$$\|\mathbf{x}\|_1 \stackrel{\text{def}}{=} \sum_{i=1}^m |x_i| = |x_1| + \dots + |x_m| \quad (1.4.9)$$

- (3)  $L_2$  范数 (常称 Euclidean 范数, 有时也称 Frobenius 范数)

$$\|\mathbf{x}\|_2 = \left( |x_1|^2 + \dots + |x_m|^2 \right)^{1/2} \quad (1.4.10)$$

(4)  $L_\infty$  范数 (也称无穷范数或极大范数)

$$\|x\|_\infty = \max\{|x_1|, \dots, |x_m|\} \quad (1.4.11)$$

(5)  $L_p$  范数 (也称 Hölder 范数 [294])

$$\|x\|_p = \left( \sum_{i=1}^m |x_i|^p \right)^{1/p}, \quad p \geq 1 \quad (1.4.12)$$

注释 1  $L_0$  范数不满足范数公理中的齐次性  $\|cx\|_0 = |c| \|x\|_0$ , 它只是一种虚拟的范数。然而,  $L_0$  范数在稀疏向量与稀疏表示中却起着关键的作用, 详见 1.12 节。

注释 2 当  $p = 2$  时,  $L_p$  范数与 Euclidean 范数完全等价。另外, 无穷范数是  $L_p$  范数的极限形式, 即有

$$\|x\|_\infty = \lim_{p \rightarrow \infty} \left( \sum_{i=1}^m |x_i|^p \right)^{1/p} \quad (1.4.13)$$

范数  $\|x\|$  称为酉不变的, 若  $\|Ux\| = \|x\|$  对所有向量  $x \in \mathbb{C}^m$  和所有酉矩阵  $U \in \mathbb{C}^{m \times m}$  恒成立。

**命题 1.4.1** [238] Euclidean 范数  $\|\cdot\|_2$  是酉不变的。

假定向量  $x$  和  $y$  有共同的起点 (即原点  $O$ ), 它们的端点分别为  $x$  和  $y$ , 则  $\|x - y\|_2$  度量两个向量  $x, y$  两端点  $x, y$  之间的标准 Euclidean 距离。特别地, 非负的标量  $\langle x, x \rangle^{1/2}$  称为向量  $x$  的 Euclidean 长度。Euclidean 长度为 1 的向量叫做归一化 (或标准化) 向量。对于任何不为零的向量  $x \in \mathbb{C}^m$ , 向量  $x/\langle x, x \rangle^{1/2}$  都是归一化的, 并且它与  $x$  同方向。

Euclidean 范数是应用最为广泛的向量范数定义。在本书后面的讨论中, 如无特别声明, 向量范数均指 Euclidean 范数。

利用向量的典范内积和 Euclidean 范数可以定义两个向量之间的夹角。

**定义 1.4.1** 两个向量之间的夹角定义为

$$\cos \theta \stackrel{\text{def}}{=} \frac{\langle x, y \rangle}{\sqrt{\langle x, x \rangle} \sqrt{\langle y, y \rangle}} = \frac{x^H y}{\|x\| \cdot \|y\|} \quad (1.4.14)$$

显然, 若  $x^H y = 0$ , 则  $\theta = \pi/2$ , 此时, 称常数向量  $x$  和  $y$  正交。因此, 两个常数向量正交的数学定义如下。

**定义 1.4.2** 两个常数向量  $x$  和  $y$  称为正交, 并记作  $x \perp y$ , 若它们的内积等于零, 即  $\langle x, y \rangle = x^H y = 0$ 。

由定义知, 零向量  $\mathbf{0}$  与同一空间的任何向量都正交。

## 2. 函数向量的内积与范数

若  $x(t)$  和  $y(t)$  分别是变量  $t$  的函数向量, 则它们的内积定义为

$$\langle x(t), y(t) \rangle \stackrel{\text{def}}{=} \int_a^b x^H(t) y(t) dt \quad (1.4.15)$$

其中, 变量  $t$  在区间  $[a, b]$  内取值, 且  $a < b$ 。变量  $t$  可以是时间变量、频率变量或者空间变量。

两个函数向量的夹角定义为

$$\cos \theta \stackrel{\text{def}}{=} \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle}} = \frac{\int_a^b \mathbf{x}^H(t) \mathbf{y}(t) dt}{\|\mathbf{x}(t)\| \cdot \|\mathbf{y}(t)\|} \quad (1.4.16)$$

式中,  $\|\mathbf{x}(t)\|$  是函数向量  $\mathbf{x}(t)$  的范数, 定义为

$$\|\mathbf{x}(t)\| \stackrel{\text{def}}{=} \left( \int_a^b \mathbf{x}^H(t) \mathbf{x}(t) dt \right)^{1/2} \quad (1.4.17)$$

显然, 若两个函数向量的内积等于零, 即

$$\int_{-\infty}^{\infty} \mathbf{x}^H(t) \mathbf{y}(t) dt = 0$$

则  $\theta = \pi/2$ 。此时, 称两个函数向量正交, 并记作  $\mathbf{x}(t) \perp \mathbf{y}(t)$ 。

### 3. 随机向量的内积与范数

若  $\mathbf{x}(\xi)$  和  $\mathbf{y}(\xi)$  分别是样本变量  $\xi$  的随机向量, 则它们的内积定义为

$$\langle \mathbf{x}(\xi), \mathbf{y}(\xi) \rangle \stackrel{\text{def}}{=} E\{\mathbf{x}^H(\xi) \mathbf{y}(\xi)\} \quad (1.4.18)$$

其中, 样本变量  $\xi$  可以是时间  $t$ 、圆频率  $f$ 、角频率  $\omega$  或空间变量  $s$  等。

随机向量  $\mathbf{x}(\xi)$  的范数  $\|\mathbf{x}(\xi)\|$  的平方定义为

$$\|\mathbf{x}(\xi)\|^2 \stackrel{\text{def}}{=} E\{\mathbf{x}^H(\xi) \mathbf{x}(\xi)\} \quad (1.4.19)$$

与常数向量和函数向量的情况不同,  $m \times 1$  随机向量  $\mathbf{x}(\xi)$  和  $n \times 1$  随机向量  $\mathbf{y}(\xi)$  称为正交, 若  $\mathbf{x}(\xi)$  的任意元素与  $\mathbf{y}(\xi)$  的任意元素正交。这意味着, 两个向量的互相关矩阵为零矩阵  $\mathbf{O}_{m \times n}$ , 即

$$E\{\mathbf{x}(\xi) \mathbf{y}^H(\xi)\} = \mathbf{O}_{m \times n} \quad (1.4.20)$$

并记作  $\mathbf{x}(\xi) \perp \mathbf{y}(\xi)$ 。

下面的命题表明, 任意两个正交向量之和的范数平方等于各个向量范数平方之和。

**命题 1.4.2** 若  $\mathbf{x} \perp \mathbf{y}$ , 则  $\|\mathbf{x} + \mathbf{y}\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2$ 。

**证明** 由范数公理知

$$\|\mathbf{x} + \mathbf{y}\|^2 = \langle \mathbf{x} + \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{x} \rangle + \langle \mathbf{x}, \mathbf{y} \rangle + \langle \mathbf{y}, \mathbf{x} \rangle + \langle \mathbf{y}, \mathbf{y} \rangle \quad (1.4.21)$$

由于  $\mathbf{x}$  和  $\mathbf{y}$  正交, 所以  $\langle \mathbf{x}, \mathbf{y} \rangle = E\{\mathbf{x}^T \mathbf{y}\} = 0$ 。又由内积公理知  $\langle \mathbf{y}, \mathbf{x} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle = 0$ 。将这一结果代入式 (1.4.21) 立即得  $\|\mathbf{x} + \mathbf{y}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle + \langle \mathbf{y}, \mathbf{y} \rangle = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2$ 。本命题得证。■

这一命题也称 Pythagorean 定理。

下面从数学定义、几何解释和物理意义三个方面，对常数向量、函数向量和随机向量的正交作一归纳与总结。

(1) 数学定义：两个向量  $\mathbf{x}$  和  $\mathbf{y}$  正交，若它们的内积等于零，即  $\langle \mathbf{x}, \mathbf{y} \rangle = 0$  (对常数向量和函数向量)，或者它们的外积的数学期望等于零矩阵，即  $E[\mathbf{x}\mathbf{y}^H] = \mathbf{O}$  (对随机向量)。

(2) 几何解释：若两个向量正交，则这两个向量之间的夹角为  $90^\circ$ ，并且一个向量到另一个向量的投影等于零。

(3) 物理意义：当两个向量正交时，一个向量将不含另一个向量的任何成分，即这两个向量之间不存在任何相互作用或干扰。

记住这些要点，将有助于在实际中灵活使用向量的正交。

#### 1.4.2 向量的相似比较

聚类 (clustering) 和分类 (classification) 是统计数据分析的重要技术。所谓聚类，就是将一给定的大数据集聚为几个小的子数据集，并且每个子集 (目标类) 的数据都具有共同或者相似的特征。分类则是将一个或者多个未知类属的数据或特征向量划分到具有最接近特征的某个已知目标类别中。

实现聚类和分类的主要数学工具为距离测度。

两个概率密度之间的距离称为测度 (metric)，若下列条件均成立：

- (1)  $D(\mathbf{p}\|\mathbf{g}) \geq 0$ ，等号成立当且仅当  $\mathbf{p} = \mathbf{g}$  (非负性和正定性)；
- (2)  $D(\mathbf{p}\|\mathbf{g}) = D(\mathbf{g}\|\mathbf{p})$  (对称性)；
- (3)  $D(\mathbf{p}\|\mathbf{z}) \leq D(\mathbf{p}\|\mathbf{g}) + D(\mathbf{g}\|\mathbf{z})$  (三角不等式)。

显然，平方 Euclidean 距离为测度。

在模式识别中，原始数据向量需要通过某种变换或处理方法，变成一个低维的向量。由于这种低维向量抽取了原始数据向量的特征，直接用于模式的聚类和分类，故称为“模式向量” (mode vector) 或“特征向量”。例如，云的颜色和语调的参数即分别构成天气和语音分类的模式或特征向量。

聚类或分类的基本准则是：用距离测度，度量两个未知特征向量的相似度或一个未知特征向量与某个已知特征向量之间的相似度。顾名思义，相似度就是两个向量之间的相似程度的度量。

考虑模式分类问题。为简单计，假定有  $M$  个类型的模式向量  $\mathbf{s}_1, \dots, \mathbf{s}_M$ 。现在的问题是：给定一任意的未知模式向量  $\mathbf{x}$ ，希望判断它归属于哪一类模式。为此，需要将未知模式向量  $\mathbf{x}$  同  $M$  个已知模式向量进行比对，看  $\mathbf{x}$  与其中哪一个样本模式向量最相似，并据此作出模式或信号分类的判断。

向量之间的相似度常采用相异度 (dissimilarity) 进行反向度量：相异度越小的两个向量之间越相似。

令  $D(\mathbf{x}, \mathbf{s}_1), \dots, D(\mathbf{x}, \mathbf{s}_M)$  分别表示未知模式向量  $\mathbf{x}$  和已知模式向量  $\mathbf{s}_1, \dots, \mathbf{s}_M$  之间的相异度的符号。以  $\mathbf{x}$  与  $\mathbf{s}_1, \mathbf{s}_2$  的相异度为例, 若

$$D(\mathbf{x}, \mathbf{s}_1) \leq D(\mathbf{x}, \mathbf{s}_2) \quad (1.4.22)$$

则称未知模式向量  $\mathbf{x}$  与样本模式向量  $\mathbf{s}_1$  更相似。

最简单和最直观的相异度是两个向量之间的 Euclidean 距离。未知模式向量  $\mathbf{x}$  与第  $i$  个已知模式向量  $\mathbf{s}_i$  之间的 Euclidean 距离记作  $D_E(\mathbf{x}, \mathbf{s}_i)$ , 定义为

$$D_E(\mathbf{x}, \mathbf{s}_i) = \|\mathbf{x} - \mathbf{s}_i\|_2 = \sqrt{(\mathbf{x} - \mathbf{s}_i)^T (\mathbf{x} - \mathbf{s}_i)} \quad (1.4.23)$$

除了满足测度的非负性、对称性和三角不等式之外, Euclidean 距离  $D_E(\mathbf{x}, \mathbf{y})$  还具有一个基本性质: Euclidean 距离等于零的两个向量完全相似, 即

$$D_E(\mathbf{x}, \mathbf{y}) = 0 \iff \mathbf{x} = \mathbf{y}$$

若

$$D_E(\mathbf{x}, \mathbf{s}_i) = \min_k D_E(\mathbf{x}, \mathbf{s}_k), \quad k = 1, \dots, M \quad (1.4.24)$$

则称  $\mathbf{s}_i \in \{\mathbf{s}_1, \dots, \mathbf{s}_M\}$  是到  $\mathbf{x}$  的近邻 (即最近的邻居)。

作为一种广泛使用的分类法, 近邻分类 (nearest neighbor classification) 法将未知类型的模式向量  $\mathbf{x}$  归并为它的近邻所属的模式类型。

另一个常用的距离函数是 Mahalanobis 距离, 由 Mahalanobis 于 1936 年在统计中作为距离测度提出的<sup>[329]</sup>。向量  $\mathbf{x}$  到其均值向量  $\mu$  的 Mahalanobis 距离为

$$D_M(\mathbf{x}, \mu) = \sqrt{(\mathbf{x} - \mu)^T \mathbf{C}_x^{-1} (\mathbf{x} - \mu)} \quad (1.4.25)$$

式中  $\mathbf{C}_x = \text{Cov}(\mathbf{x}, \mathbf{x}) = E\{(\mathbf{x} - \mu)(\mathbf{x} - \mu)^T\}$  是向量  $\mathbf{x}$  的自协方差矩阵。

两个向量  $\mathbf{x} \in \mathbb{R}^n$  和  $\mathbf{y} \in \mathbb{R}^n$  之间的 Mahalanobis 距离记作  $D_M(\mathbf{x}, \mathbf{y})$ , 定义为<sup>[329]</sup>

$$D_M(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{y})} \quad (1.4.26)$$

其中  $\mathbf{C} = E\{(\mathbf{x} - \mu_x)(\mathbf{y} - \mu_y)^T\}$  是两个向量  $\mathbf{x}$  与  $\mathbf{y}$  之间的互协方差矩阵, 而  $\mu_x$  和  $\mu_y$  分别是向量  $\mathbf{x}$  和  $\mathbf{y}$  的均值向量。

若协方差矩阵为单位矩阵, 即  $\mathbf{C} = \mathbf{I}$ , 则 Mahalanobis 距离退化为 Euclidean 距离。如果协方差矩阵取对角矩阵, 则相应的 Mahalanobis 距离称为归一化 Euclidean 距离

$$D_M(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n \frac{(x_i - y_i)^2}{\sigma_i^2}} \quad (1.4.27)$$

式中,  $\sigma_i$  是  $x_i$  和  $y_i$  在整个样本集合的标准差。

令

$$\mu = \frac{1}{M} \sum_{i=1}^M \mathbf{s}_i, \quad \mathbf{C} = \sum_{i=1}^M \sum_{j=1}^M (\mathbf{s}_i - \mu)(\mathbf{s}_j - \mu)^T \quad (1.4.28)$$

分别为  $M$  个已知模式向量  $\mathbf{s}_i$  的样本均值向量和样本互协方差矩阵。于是，未知模式向量  $\mathbf{x}$  到已知模式向量  $\mathbf{s}_i$  之间的 Mahalanobis 距离定义为

$$D_M(\mathbf{x}, \mathbf{s}_i) = \sqrt{(\mathbf{x} - \mathbf{s}_i)^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{s}_i)} \quad (1.4.29)$$

根据近邻分类法，若

$$D_M(\mathbf{x}, \mathbf{s}_i) = \min_k D_M(\mathbf{x}, \mathbf{s}_k), \quad k = 1, \dots, M \quad (1.4.30)$$

则将未知模式向量  $\mathbf{x}$  归为  $\mathbf{s}_i$  所属的模式类型。

向量之间的相异度的测度不一定局限于距离函数。两个向量所夹锐角的余弦函数

$$D(\mathbf{x}, \mathbf{s}_i) = \cos(\theta_i) = \frac{\mathbf{x}^T \mathbf{s}_i}{\|\mathbf{x}\|_2 \|\mathbf{s}_i\|_2} \quad (1.4.31)$$

也是相异度的一种有效测度。若  $\cos(\theta_i) < \cos(\theta_j), \forall j \neq i$  成立，则认为未知模式向量  $\mathbf{x}$  与样本模式向量  $\mathbf{s}_i$  最相似。式 (1.4.31) 的变型

$$D(\mathbf{x}, \mathbf{s}_i) = \frac{\mathbf{x}^T \mathbf{s}_i}{\mathbf{x}^T \mathbf{x} + \mathbf{s}_i^T \mathbf{s}_i + \mathbf{x}^T \mathbf{s}_i} \quad (1.4.32)$$

称为 Tanimoto 测度<sup>[477]</sup>，它广泛应用于信息恢复、疾病分类、动物和植物分类等。

待分类的信号称为目标信号，分类通常是根据某种物理或几何概念进行的。令  $X$  为目标信号， $A_i$  代表第  $i$  类目标的分类概念。于是，采用目标-概念距离 (object-concept distance)  $D(X, A_i)$  描述与目标之间的相异度<sup>[457]</sup>，从而有类似于式 (1.4.22) 的关系

$$D(X, A_i) \leq D(X, A_j), \quad \forall i, j \quad (1.4.33)$$

因此，将目标信号  $X$  归为目标-概念距离  $D(X, A_i)$  最小的第  $i$  类目标  $C_i$ 。

以上介绍了五种相异度：Euclidean 距离、Mahalanobis 距离、夹角余弦、Tanimoto 测度以及目标-概念距离。

### 1.4.3 矩阵的内积与范数

将向量的内积与范数加以推广，即可引出矩阵的内积与范数。

令  $m \times n$  复矩阵  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  和  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$ ，将这两个矩阵分别“拉长”为  $mn \times 1$  向量

$$\mathbf{a} = \text{vec}(\mathbf{A}) = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_n \end{bmatrix}, \quad \mathbf{b} = \text{vec}(\mathbf{B}) = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_n \end{bmatrix}$$

$\text{vec}(\mathbf{A})$  称为矩阵  $\mathbf{A}$  的(列)向量化。矩阵的向量化将在 1.11 节中作详细介绍。

矩阵的内积记作  $\langle \mathbf{A}, \mathbf{B} \rangle : \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \rightarrow \mathbb{C}$ ，定义为两个“拉长向量”  $\mathbf{a}$  和  $\mathbf{b}$  之间的内积

$$\langle \mathbf{A}, \mathbf{B} \rangle = \langle \text{vec}(\mathbf{A}), \text{vec}(\mathbf{B}) \rangle = \sum_{i=1}^n \mathbf{a}_i^H \mathbf{b}_i = \sum_{i=1}^n \langle \mathbf{a}_i, \mathbf{b}_i \rangle \quad (1.4.34)$$

或等价写作

$$\langle \mathbf{A}, \mathbf{B} \rangle = \text{vec}(\mathbf{A})^H \text{vec}(\mathbf{B}) = \text{tr}(\mathbf{A}^H \mathbf{B}) \quad (1.4.35)$$

式中  $\text{tr}(\mathbf{C})$  表示正方矩阵  $\mathbf{C}$  的迹函数，定义为该矩阵对角元素之和。

令  $\mathbb{K}$  表示一实数域或复数域， $\mathbb{K}^{m \times n}$  表示  $m \times n$  实数或复数矩阵的集合。

矩阵  $\mathbf{A} \in \mathbb{K}^{m \times n}$  的范数记作  $\|\mathbf{A}\|$ ，它是矩阵  $\mathbf{A}$  的实值函数，必须具有以下性质：

(1) 正值性：对于任何非零矩阵  $\mathbf{A} \neq \mathbf{O}$ ，其范数大于零，即  $\|\mathbf{A}\| > 0$  若  $\mathbf{A} \neq \mathbf{O}$  (零矩阵)；并且  $\|\mathbf{A}\| = 0$  当且仅当  $\mathbf{A} = \mathbf{O}$ 。

(2) 正比例性：对于任意  $c \in \mathbb{K}$ ，有  $\|c\mathbf{A}\| = |c| \cdot \|\mathbf{A}\|$ 。

(3) 三角不等式： $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$ 。

(4) 两个矩阵乘积的范数小于或等于两个矩阵范数的乘积，即  $\|\mathbf{AB}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$ 。

**例 1.4.2** 考查  $n \times n$  矩阵  $\mathbf{A}$  的实值函数  $f(\mathbf{A}) = \sum_{i=1}^n \sum_{j=1}^n |a_{ij}|$ 。容易验证：

(1)  $f(\mathbf{A}) \geq 0$ ，并且当  $\mathbf{A} = \mathbf{0}$  即  $a_{ij} \equiv 0$  时  $f(\mathbf{A}) = 0$ 。

(2)  $f(c\mathbf{A}) = \sum_{i=1}^n \sum_{j=1}^n |ca_{ij}| = |c| \sum_{i=1}^n \sum_{j=1}^n |a_{ij}| = |c|f(\mathbf{A})$ 。

(3)  $f(\mathbf{A} + \mathbf{B}) = \sum_{i=1}^n \sum_{j=1}^n (|a_{ij} + b_{ij}|) \leq \sum_{i=1}^n \sum_{j=1}^n (|a_{ij}| + |b_{ij}|) = f(\mathbf{A}) + f(\mathbf{B})$ 。

(4) 对于两个矩阵的乘积，有

$$\begin{aligned} f(\mathbf{AB}) &= \sum_{i=1}^n \sum_{j=1}^n \left| \sum_{k=1}^n a_{ik} b_{kj} \right| \leq \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n |a_{ik}| |b_{kj}| \\ &\leq \sum_{i=1}^n \sum_{j=1}^n \left( \sum_{k=1}^n |a_{ik}| \sum_{l=1}^n |b_{kl}| \right) = f(\mathbf{A})f(\mathbf{B}) \end{aligned}$$

因此，实函数  $f(\mathbf{A}) = \sum_{i=1}^n \sum_{j=1}^n |a_{ij}|$  是一种矩阵范数。

矩阵的范数有三种主要类型：诱导范数、元素形式范数和 Schatten 范数。

### 1. 诱导范数 (induced norm)

诱导范数又称  $m \times n$  矩阵空间上的算子范数 (operator norm)，定义为

$$\|\mathbf{A}\| = \max\{\|\mathbf{Ax}\| : \mathbf{x} \in \mathbb{K}^n, \|\mathbf{x}\| = 1\} \quad (1.4.36)$$

$$= \max \left\{ \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|} : \mathbf{x} \in \mathbb{K}^n, \mathbf{x} \neq \mathbf{0} \right\} \quad (1.4.37)$$

常用的诱导范数为  $p$ -范数

$$\|\mathbf{A}\|_p \stackrel{\text{def}}{=} \max_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{Ax}\|_p}{\|\mathbf{x}\|_p} \quad (1.4.38)$$

$p$  范数也称 Minkowski  $p$  范数或者  $L_p$  范数。特别地， $p = 1, 2, \infty$  时，对应的诱导范数分

别为

$$\|\mathbf{A}\|_1 = \max_{1 \leq i \leq n} \sum_{j=1}^m |a_{ij}| \quad (1.4.39)$$

$$\|\mathbf{A}\|_{\text{spec}} = \|\mathbf{A}\|_2 \quad (1.4.40)$$

$$\|\mathbf{A}\|_\infty = \max_{1 \leq j \leq m} \sum_{i=1}^n |a_{ij}| \quad (1.4.41)$$

也就是说, 诱导  $L_1$  和  $L_\infty$  范数分别直接是该矩阵的各列元素绝对值之和的最大值 (最大绝对列和) 及最大绝对行和; 而诱导  $L_2$  范数则是矩阵  $\mathbf{A}$  的最大奇异值。

诱导  $L_1$  范数  $\|\mathbf{A}\|_1$  和诱导  $L_\infty$  范数  $\|\mathbf{A}\|_\infty$  也分别称为绝对列和范数 (column-sum norm) 及绝对行和范数 (row-sum norm)。诱导  $L_2$  范数习惯称为谱范数 (spectrum norm)。

例如, 矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & -2 & 3 \\ -4 & 5 & -6 \\ 7 & -8 & -9 \\ -10 & 11 & 12 \end{bmatrix}$$

的绝对列和范数与绝对行和范数分别为

$$\|\mathbf{A}\|_1 = \max\{22, 26, 30\} = 30, \quad \|\mathbf{A}\|_\infty = \max\{6, 15, 24, 33\} = 33$$

## 2. “元素形式”范数 (“entrywise” norm)

将  $m \times n$  矩阵先按照列堆栈的形式, 排列成一个  $mn \times 1$  向量, 然后采用向量的范数定义, 即得到矩阵的范数。由于这类范数是使用矩阵的元素表示的, 故称为元素形式范数。元素形式范数是下面的  $p$  矩阵范数

$$\|\mathbf{A}\|_p \stackrel{\text{def}}{=} \left( \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^p \right)^{1/p} \quad (1.4.42)$$

以下是三种典型的元素形式  $p$  范数:

(1)  $L_1$  范数 (和范数) ( $p = 1$ )

$$\|\mathbf{A}\|_1 \stackrel{\text{def}}{=} \sum_{i=1}^m \sum_{j=1}^n |a_{ij}| \quad (1.4.43)$$

(2) Frobenius 范数 ( $p = 2$ )

$$\|\mathbf{A}\|_F \stackrel{\text{def}}{=} \left( \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2} \quad (1.4.44)$$

(3) 最大范数 (max norm) 即  $p = \infty$  的  $p$  范数, 定义为

$$\|\mathbf{A}\|_\infty = \max_{i=1, \dots, m; j=1, \dots, n} \{|a_{ij}|\} \quad (1.4.45)$$

Frobenius 范数可以视为向量的 Euclidean 范数对按照矩阵各列依次排列的“拉长向量” $\mathbf{x} = [a_{11}, \dots, a_{m1}, a_{12}, \dots, a_{m2}, \dots, a_{1n}, \dots, a_{mn}]^T$  的推广。矩阵的 Frobenius 范数有时也称 Euclidean 范数、Schur 范数、Hilbert-Schmidt 范数或者  $L_2$  范数。

Frobenius 范数又可写作迹函数的形式

$$\|\mathbf{A}\|_F \stackrel{\text{def}}{=} \langle \mathbf{A}, \mathbf{A} \rangle^{1/2} = \sqrt{\text{tr}(\mathbf{A}^H \mathbf{A})} \quad (1.4.46)$$

由正定的矩阵  $\Omega$  进行加权的 Frobenius 范数

$$\|\mathbf{A}\|_\Omega = \sqrt{\text{tr}(\mathbf{A}^H \Omega \mathbf{A})} \quad (1.4.47)$$

称为 Mahalanobis 范数。

Schatten 范数就是用矩阵的奇异值定义的范数，将在第 5 章（奇异值分析）中介绍。

注意，向量  $\mathbf{x}$  的  $L_p$  范数  $\|\mathbf{x}\|_p$  相当于该向量的长度。当矩阵  $\mathbf{A}$  作用于长度为  $\|\mathbf{x}\|_p$  的向量  $\mathbf{x}$  时，得到线性变换结果为向量  $\mathbf{Ax}$ ，其长度为  $\|\mathbf{Ax}\|_p$ 。线性变换矩阵  $\mathbf{A}$  可视为一线性放大器算子。因此，比率  $\|\mathbf{Ax}\|_p / \|\mathbf{x}\|_p$  提供了线性变换  $\mathbf{Ax}$  相对于  $\mathbf{x}$  的放大倍数，而矩阵  $\mathbf{A}$  的  $p$  范数  $\|\mathbf{A}\|_p$  是由  $\mathbf{A}$  产生的最大放大倍数。类似地，放大器算子  $\mathbf{A}$  的最小放大倍数由

$$\min |\mathbf{A}|_p \stackrel{\text{def}}{=} \min_{\mathbf{x} \neq 0} \frac{\|\mathbf{Ax}\|_p}{\|\mathbf{x}\|_p} \quad (1.4.48)$$

给出。比率  $\|\mathbf{A}\|_p / \min |\mathbf{A}|_p$  描述放大器算子  $\mathbf{A}$  的“动态范围”。

若  $\mathbf{A}, \mathbf{B}$  是  $m \times n$  矩阵，则矩阵的范数具有以下性质

$$\|\mathbf{A} + \mathbf{B}\| + \|\mathbf{A} - \mathbf{B}\| = 2(\|\mathbf{A}\|^2 + \|\mathbf{B}\|^2) \quad (\text{平行四边形法则}) \quad (1.4.49)$$

$$\|\mathbf{A} + \mathbf{B}\| \cdot \|\mathbf{A} - \mathbf{B}\| \leq \|\mathbf{A}\|^2 + \|\mathbf{B}\|^2 \quad (1.4.50)$$

以下是矩阵的内积与范数之间的关系<sup>[238]</sup>。

(1) Cauchy-Schwartz 不等式

$$|\langle \mathbf{A}, \mathbf{B} \rangle|^2 \leq \|\mathbf{A}\|^2 \|\mathbf{B}\|^2 \quad (1.4.51)$$

等号成立，当且仅当  $\mathbf{A} = c\mathbf{B}$ ，其中， $c$  是某个复常数。

(2) Pythagoras 定理:  $\langle \mathbf{A}, \mathbf{B} \rangle = 0 \Rightarrow \|\mathbf{A} + \mathbf{B}\|^2 = \|\mathbf{A}\|^2 + \|\mathbf{B}\|^2$

(3) 极化恒等式

$$\text{Re}(\langle \mathbf{A}, \mathbf{B} \rangle) = \frac{1}{4} (\|\mathbf{A} + \mathbf{B}\|^2 - \|\mathbf{A} - \mathbf{B}\|^2) \quad (1.4.52)$$

$$\text{Re}(\langle \mathbf{A}, \mathbf{B} \rangle) = \frac{1}{2} (\|\mathbf{A} + \mathbf{B}\|^2 - \|\mathbf{A}\|^2 - \|\mathbf{B}\|^2) \quad (1.4.53)$$

式中  $\text{Re}(\langle \mathbf{A}, \mathbf{B} \rangle)$  表示  $\mathbf{A}^H \mathbf{B}$  的实部。

## 1.5 随机向量

在概率论中, 常称  $\omega$  为基本事件或样本,  $\Omega$  为样本空间,  $A(\in \mathcal{F})$  为事件,  $\mathcal{F}$  是事件的全体, 而  $P(A)$  称为事件的概率。三元组  $(\Omega, \mathcal{F}, P)$  构成概率空间。用  $L_p = L_p(\Omega, \mathcal{F}, P)$  表示随机变量  $\xi = \xi(\omega)$  的空间, 其中  $E\{|\xi|^p\} = \int_{\Omega} |\xi|^p dP < \infty$ 。

称  $L_p (p > 1)$  为 Banach 空间。在 Banach 空间中, 起重要作用的是空间  $L_2 = L_2(\Omega, \mathcal{F}, P)$ , 即具有有限二阶矩  $E\{\xi^2\} < \infty$  的随机变量的 Hilbert 空间, 简称  $L_2$  空间。只研究向量空间中一阶和二阶统计性质的理论称为  $L^2$  理论。

另外, 在信号处理、自动控制、通信、电子工程、神经网络等应用中, 观测数据和加性噪声通常取随机变量。由随机变量组成的向量称为随机向量。本节介绍随机向量的  $L^2$  理论, 先讨论实随机向量, 然后再推广到复随机向量。

### 1.5.1 概率密度函数

描述随机向量的统计函数有累积分布函数、概率密度函数、均值函数和协方差函数等, 先讨论随机向量的累积分布函数和概率密度函数。

#### 1. 实随机向量的概率密度函数

一个含有  $m$  个随机变量的实值向量

$$\mathbf{x}(\xi) = [x_1(\xi), \dots, x_m(\xi)]^T \quad (1.5.1)$$

称为  $m \times 1$  实随机向量, 或简称随机向量(当维数无关紧要时)。其中,  $\xi$  表示样本点, 例如它可以是时间  $t$ , 圆频率  $f$ , 角频率  $\omega$  或位置  $s$  等。

一个随机向量所有元素的联合累积分布函数常用符号  $F_{\mathbf{x}}(x_1, \dots, x_m)$  表示, 联合概率密度函数常用  $f_{\mathbf{x}}(x_1, \dots, x_m)$  作符号。为了简化, 令  $F(\mathbf{x}) = F_{\mathbf{x}}(x_1, \dots, x_m)$  和  $f(\mathbf{x}) = f_{\mathbf{x}}(x_1, \dots, x_m)$ 。一个随机向量由它的联合累积分布函数或联合概率密度函数完全描述。一组概率的集合函数

$$F(\mathbf{x}) \stackrel{\text{def}}{=} P\{\xi : x_1(\xi) \leq x_1, \dots, x_m(\xi) \leq x_m\} \quad (1.5.2)$$

定义为向量  $\mathbf{x}(\xi)$  的联合累积分布函数, 简称分布函数。式中,  $x_i$  为实数。

随机向量  $\mathbf{x}(\xi)$  的(联合)概率密度函数定义为

$$\begin{aligned} f(\mathbf{x}) &\stackrel{\text{def}}{=} \lim_{\Delta x_1 \rightarrow 0, \dots, \Delta x_m \rightarrow 0} \frac{P\{\xi : x_1 < x_1(\xi) \leq x_1 + \Delta x_1, \dots, x_m < x_m(\xi) \leq x_m + \Delta x_m\}}{\Delta x_1 \cdots \Delta x_m} \\ &= \frac{\partial^m}{\partial x_1 \cdots \partial x_m} F_{\mathbf{x}}(x_1, \dots, x_m) \end{aligned} \quad (1.5.3)$$

联合概率密度函数的  $m - 1$  重积分函数

$$f(x_i) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{\mathbf{x}}(x_1, \dots, x_m) dx_1 \cdots dx_{i-1} dx_{i+1} \cdots dx_m \quad (1.5.4)$$

称为随机变量  $x_i$  的边缘概率密度函数。

由式(1.5.2)和式(1.5.4)易知

$$F(\mathbf{x}) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_m} f_{\mathbf{v}}(v_1, \dots, v_m) dv_1 \cdots dv_m \quad (1.5.5)$$

就是说, 随机向量  $\mathbf{x}(\xi)$  的联合分布函数等于其联合概率密度函数的积分。

**定义 1.5.1** [392] 随机变量  $x_1(\xi), \dots, x_m(\xi)$  称为(联合)独立, 若对于  $m$  个事件  $\{x_1(\xi) \leq x_1\}, \dots, \{x_m(\xi) \leq x_m\}$ , 有概率关系

$$P\{x_1(\xi) \leq x_1, \dots, x_m(\xi) \leq x_m\} = P\{x_1(\xi) \leq x_1\} \cdots P\{x_m(\xi) \leq x_m\} \quad (1.5.6)$$

成立。这意味着

$$F(\mathbf{x}) = F_{\mathbf{x}}(x_1, \dots, x_m) = F_{x_1}(x_1) \cdots F_{x_m}(x_m) \quad (1.5.7)$$

$$f(\mathbf{x}) = f_{\mathbf{x}}(x_1, \dots, x_m) = f_{x_1}(x_1) \cdots f_{x_m}(x_m) \quad (1.5.8)$$

用文字表述, 即有: 若  $m$  个随机变量的联合分布函数(或联合概率密度函数)等于各个随机变量的边缘分布函数(或边缘概率密度函数)之积, 则这  $m$  个随机变量是联合独立的。为了与线性独立(即线性无关)相区别, 随机变量之间的独立被称为统计独立。

## 2. 复随机向量的概率密度函数

一个复随机变量定义为  $x(\xi) = x_R(\xi) + j x_I(\xi)$ , 其中,  $x_R(\xi)$  和  $x_I(\xi)$  分别为实值随机变量。

复随机向量可以表示成

$$\mathbf{x}(\xi) = \mathbf{x}_R(\xi) + j \mathbf{x}_I(\xi) = \begin{bmatrix} x_{R1}(\xi) \\ \vdots \\ x_{Rm}(\xi) \end{bmatrix} + j \begin{bmatrix} x_{I1}(\xi) \\ \vdots \\ x_{Im}(\xi) \end{bmatrix} \quad (1.5.9)$$

复随机向量的累积分布函数定义为

$$F(\mathbf{x}) \stackrel{\text{def}}{=} P\{\mathbf{x}(\xi) \leq \mathbf{x}\} \stackrel{\text{def}}{=} P\{x_R(\xi) \leq x_R, x_I(\xi) \leq x_I\} \quad (1.5.10)$$

概率密度函数定义为

$$f(\mathbf{x}) \stackrel{\text{def}}{=} \frac{\partial^{2m} F(\mathbf{x})}{\partial x_{R1} \partial x_{I1} \cdots \partial x_{Rm} \partial x_{Im}} \quad (1.5.11)$$

由式(1.5.10)和式(1.5.11)知, 累积分布函数是概率密度函数关于所有实部和虚部的  $2m$  重积分, 即

$$\begin{aligned} F(\mathbf{x}) &= F_{\mathbf{x}}(x_1, x_2, \dots, x_m) \\ &= \int_{-\infty}^{x_{R1}} \int_{-\infty}^{x_{I1}} \cdots \int_{-\infty}^{x_{Rm}} \int_{-\infty}^{x_{Im}} f(v_1, \dots, v_m) dv_{R1} dv_{I1} \cdots dv_{Rm} dv_{Im} \\ &= \int_{-\infty}^{\mathbf{x}} f(\mathbf{v}) d\mathbf{v} \end{aligned} \quad (1.5.12)$$

特别地

$$\int_{-\infty}^{\infty} f(\mathbf{x}) d\mathbf{x} = 1 \quad (1.5.13)$$

### 1.5.2 随机向量的统计描述

分布函数和概率密度函数常常是未知的，因此它们在很多实际问题中的应用并不方便。与之不同，随机向量的一阶和二阶统计量却使用方便。

#### 1. 均值向量

随机向量的最重要的统计运算为数学期望。考查  $m \times 1$  随机向量  $\mathbf{x}(\xi) = [x_1(\xi), \dots, x_m(\xi)]^T$ 。令随机变量  $x_i(\xi)$  的均值  $E\{x_i(\xi)\} = \mu_i$ ，则随机向量的数学期望称为均值向量，记作  $\boldsymbol{\mu}_x$ ，定义为

$$\boldsymbol{\mu}_x = E\{\mathbf{x}(\xi)\} = \begin{bmatrix} E\{x_1(\xi)\} \\ \vdots \\ E\{x_m(\xi)\} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_m \end{bmatrix} \quad (1.5.14)$$

式中，数学期望定义为

$$E\{\mathbf{x}(\xi)\} \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \mathbf{x} f(x) dx \quad (1.5.15)$$

$$E\{\mathbf{x}(\xi)\} \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \mathbf{x} f(x) dx \quad (1.5.16)$$

式 (1.5.14) 表明，均值向量的元素是随机向量各个元素的均值。

#### 2. 相关矩阵与协方差矩阵

均值向量是随机向量的一阶矩，它描述随机向量的元素围绕其均值的散布情况。与均值向量不同，随机向量的二阶矩为矩阵，它描述随机向量分布的散布情况。

相关矩阵与协方差矩阵与两个向量的外积密切相关。两个向量  $\mathbf{x} \in \mathbb{C}^{m \times 1}$  与  $\mathbf{y} \in \mathbb{C}^{n \times 1}$  的外积 (outer product) 给出一  $m \times n$  复矩阵，记作  $\mathbf{x} \circ \mathbf{y}$ ，定义为

$$\mathbf{x} \circ \mathbf{y} = \mathbf{x} \mathbf{y}^H \quad (1.5.17)$$

随机向量的自相关矩阵定义为该向量与自身的外积的数学期望

$$\mathbf{R}_x \stackrel{\text{def}}{=} E\{\mathbf{x}(\xi) \mathbf{x}^H(\xi)\} = \begin{bmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{bmatrix} \quad (1.5.18)$$

式中， $r_{ii}, i = 1, \dots, m$  表示随机变量  $x_i(\xi)$  的自相关函数，定义为

$$r_{ii} \stackrel{\text{def}}{=} E\{|x_i(\xi)|^2\}, \quad i = 1, \dots, m \quad (1.5.19)$$

而  $r_{ij}$  表示随机变量  $x_i(\xi)$  和  $x_j(\xi)$  之间的互相关函数，定义为

$$r_{ij} \stackrel{\text{def}}{=} E\{x_i(\xi) x_j^*(\xi)\}, \quad i, j = 1, \dots, m, \quad i \neq j \quad (1.5.20)$$

显然，自相关矩阵是复共轭对称的，即为 Hermitian 矩阵。

随机向量  $\mathbf{x}(\xi)$  的自协方差矩阵记为  $\mathbf{C}_x$ , 定义为

$$\mathbf{C}_x = \text{Cov}(\mathbf{x}, \mathbf{x}) \stackrel{\text{def}}{=} E\{[\mathbf{x}(\xi) - \boldsymbol{\mu}_x][\mathbf{x}(\xi) - \boldsymbol{\mu}_x]^H\} = \begin{bmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mm} \end{bmatrix} \quad (1.5.21)$$

式中, 主对角线的元素

$$c_{ii} \stackrel{\text{def}}{=} E\{|x_i(\xi) - \mu_i|^2\}, \quad i = 1, \dots, m \quad (1.5.22)$$

表示随机变量  $x_i(\xi)$  的方差  $\sigma_i^2$ , 即  $c_{ii} = \sigma_i^2$ , 而非主对角线元素

$$c_{ij} \stackrel{\text{def}}{=} E\{[x_i(\xi) - \mu_i][x_j(\xi) - \mu_j]^*\} = E\{x_i(\xi)x_j^*(\xi)\} - \mu_i\mu_j^* = c_{ji}^* \quad (1.5.23)$$

表示随机变量  $x_i(\xi)$  和  $x_j(\xi)$  之间的协方差。自协方差矩阵也是 Hermitian 矩阵。

自协方差矩阵有时也称为方差矩阵, 用符号  $\text{Var}(\mathbf{x})$  表示, 即有  $\text{Var}(\mathbf{x}) = E\{[\mathbf{x}(\xi) - \boldsymbol{\mu}_x][\mathbf{x}(\xi) - \boldsymbol{\mu}_x]^H\}$ 。显然,  $\text{Var}(\mathbf{x}) = \text{Cov}(\mathbf{x}, \mathbf{x})$ 。

自相关矩阵和自协方差矩阵之间存在下列关系

$$\mathbf{C}_x = \mathbf{R}_x - \boldsymbol{\mu}_x \boldsymbol{\mu}_x^H \quad (1.5.24)$$

推广自相关矩阵和自协方差矩阵的概念, 则有随机向量  $\mathbf{x}(\xi)$  和  $\mathbf{y}(\xi)$  的互相关矩阵

$$\mathbf{R}_{xy} \stackrel{\text{def}}{=} E\{\mathbf{x}(\xi)\mathbf{y}^H(\xi)\} = \begin{bmatrix} r_{x_1, y_1} & \cdots & r_{x_1, y_m} \\ \vdots & \ddots & \vdots \\ r_{x_m, y_1} & \cdots & r_{x_m, y_m} \end{bmatrix} \quad (1.5.25)$$

和互协方差矩阵

$$\mathbf{C}_{xy} \stackrel{\text{def}}{=} E\{[\mathbf{x}(\xi) - \boldsymbol{\mu}_x][\mathbf{y}(\xi) - \boldsymbol{\mu}_y]^H\} = \begin{bmatrix} c_{x_1, y_1} & \cdots & c_{x_1, y_m} \\ \vdots & \ddots & \vdots \\ c_{x_m, y_1} & \cdots & c_{x_m, y_m} \end{bmatrix} \quad (1.5.26)$$

式中,  $r_{x_i, y_j} \stackrel{\text{def}}{=} E\{x_i(\xi)y_j^*(\xi)\}$  是随机变量  $x_i(\xi)$  和  $y_j(\xi)$  之间的互相关,  $c_{x_i, y_j} \stackrel{\text{def}}{=} E\{[x_i(\xi) - \mu_{x_i}][y_j(\xi) - \mu_{y_j}]^*\}$  是随机变量  $x_i(\xi)$  和  $y_j(\xi)$  之间的互协方差。

易知, 互协方差矩阵与互相关矩阵之间存在下列关系

$$\mathbf{C}_{xy} = \mathbf{R}_{xy} - \boldsymbol{\mu}_x \boldsymbol{\mu}_y^H \quad (1.5.27)$$

一个随机向量的自相关矩阵和自协方差矩阵均为正方的复共轭对称矩阵, 而两个维数不同的随机向量的互相关矩阵和互协方差矩阵是非正方的矩阵。即使随机向量  $\mathbf{x}(\xi)$  和  $\mathbf{y}(\xi)$  维数相同, 互相关矩阵和互协方差矩阵为正方矩阵, 它们也不是复共轭对称的。

利用定义公式, 很容易验证自协方差矩阵与互协方差矩阵的以下性质:

(1) 自协方差矩阵是复共轭转置对称的, 即有  $[\text{Var}(\mathbf{x})]^H = \text{Var}(\mathbf{x})$ 。

- (2) 线性组合向量  $\mathbf{Ax} + \mathbf{b}$  的自协方差矩阵  $\text{Var}(\mathbf{Ax} + \mathbf{b}) = \text{Var}(\mathbf{Ax}) = \mathbf{A}\text{Var}(\mathbf{x})\mathbf{A}^H$ 。
- (3) 互协方差矩阵不是复共轭转置对称的，但满足  $\text{Cov}(\mathbf{x}, \mathbf{y}) = [\text{Cov}(\mathbf{y}, \mathbf{x})]^H$ 。
- (4)  $\text{Cov}(\mathbf{x}_1 + \mathbf{x}_2, \mathbf{y}) = \text{Cov}(\mathbf{x}_1, \mathbf{y}) + \text{Cov}(\mathbf{x}_2, \mathbf{y})$ 。
- (5) 若  $\mathbf{x}$  和  $\mathbf{y}$  具有相同的维数，则

$$\text{Var}(\mathbf{x} + \mathbf{y}) = \text{Var}(\mathbf{x}) + \text{Cov}(\mathbf{x}, \mathbf{y}) + \text{Cov}(\mathbf{y}, \mathbf{x}) + \text{Var}(\mathbf{y})$$

- (6)  $\text{Cov}(\mathbf{Ax}, \mathbf{By}) = \mathbf{A}\text{Cov}(\mathbf{x}, \mathbf{y})\mathbf{B}^H$ 。

### 3. 两个随机向量的统计不相关与正交

互协方差函数描述两个随机信号  $x_i(\xi)$  和  $x_j(\xi)$  之间的相关(联)程度。一般说来，互协方差函数越大，则这两个随机信号的相关程度越强；反之，相关程度越弱。但是，这种使用互协方差的绝对大小度量两个随机向量的相关程度并不方便。

两个随机变量  $x(\xi)$  和  $y(\xi)$  之间的相关系数定义为

$$\rho_{xy} \stackrel{\text{def}}{=} \frac{c_{xy}}{\sqrt{\text{E}\{|x(\xi)|^2\}\text{E}\{|y(\xi)|^2\}}} = \frac{c_{xy}}{\sigma_x \sigma_y} \quad (1.5.28)$$

式中， $c_{xy}$  是随机变量  $x(\xi)$  和  $y(\xi)$  之间的互协方差，而  $\sigma_x^2$  和  $\sigma_y^2$  分别是  $x(\xi)$  和  $y(\xi)$  的方差。对相关系数的定义公式使用 Cauchy-Schwartz 不等式，易知

$$0 \leq |\rho_{xy}| \leq 1 \quad (1.5.29)$$

相关系数  $\rho_{xy}$  给出了两个随机变量  $x(\xi)$  和  $y(\xi)$  之间的相似程度的度量： $\rho_{xy}$  越接近于零，随机变量  $x(\xi)$  和  $y(\xi)$  的相似度越弱；反之，若  $\rho_{xy}$  越接近于 1，则  $x(\xi)$  和  $y(\xi)$  的相似度越大。特别地，相关系数的两个极端值 0 和 1 有着重要的意义。

$\rho_{xy} = 0$  意味着互协方差  $c_{xy} = 0$ ，这表明随机变量  $x(\xi)$  和  $y(\xi)$  之间不存在任何相关部分。因此，若  $\rho_{xy} = 0$ ，则称随机变量  $x(\xi)$  和  $y(\xi)$  不相关。鉴于这种不相关是在统计意义上定义的，所以常称之为统计不相关。

容易验证，若  $x(\xi) = cy(\xi)$ ，其中， $c$  为一复常数，则  $|\rho_{xy}| = 1$ 。满足条件  $x(\xi) = cy(\xi) = |c|e^{j\Phi(c)}y(\xi)$  的随机变量  $x(\xi)$  和  $y(\xi)$  只是相差一个固定的幅值比例因子和一个固定的相位  $\Phi(c)$ 。这样的两个随机变量称为完全相关(或相干)。

将两个随机变量之间的不相关条件  $c_{xy} = 0, i \neq j$  加以推广，立即得到  $m \times 1$  随机向量  $\mathbf{x}(\xi)$  和  $n \times 1$  随机向量  $\mathbf{y}(\xi)$  统计不相关定义如下。

**定义 1.5.2**  $m \times 1$  随机向量  $\mathbf{x}(\xi)$  与  $n \times 1$  随机向量  $\mathbf{y}(\xi)$  统计不相关，若它们的互协方差矩阵等于零矩阵，即  $C_{xy} = \mathbf{O}_{m \times n}$ 。

两个随机变量  $x(\xi)$  和  $y(\xi)$  称为正交，若它们的互相关等于零，即

$$r_{xy} = \text{E}\{x(\xi)y^*(\xi)\} = 0 \quad (1.5.30)$$

类似地，两个随机向量  $\mathbf{x}(\xi) = [x_1(\xi), \dots, x_m(\xi)]^T$  和  $\mathbf{y}(\xi) = [y_1(\xi), \dots, y_n(\xi)]^T$  称为正交，若  $\mathbf{x}(\xi)$  的任一元素  $x_i(\xi)$  与随机向量  $\mathbf{y}(\xi)$  的任意元素  $y_j(\xi)$  正交，即  $r_{x_i, y_j} =$

$E\{x_i(\xi)y_j(\xi)\} = 0, i = 1, \dots, m; j = 1, \dots, n$ 。显然, 这意味着这两个随机向量的互相关矩阵等于零矩阵, 即有  $R_{xy} = O_{m \times n}$ 。

**定义 1.5.3** 称  $m$  维随机向量  $\mathbf{x}(\xi)$  与  $n$  维随机向量  $\mathbf{y}(\xi)$  正交, 若它们的互相关矩阵等于零矩阵, 即  $R_{xy} = O_{m \times n}$ 。

比较互协方差矩阵和互相关矩阵的定义知, 若随机向量  $\mathbf{x}(\xi)$  和  $\mathbf{y}(\xi)$  均具有零均值向量, 则  $C_{xy} = R_{xy}$ 。因此, 对于分别具有零均值向量的两个随机向量而言, 它们之间的统计不相关与正交是等价的。

### 1.5.3 高斯随机向量

若随机向量  $\mathbf{x} = [x_1(\xi), \dots, x_m(\xi)]^T$  的各个分量  $x_i(\xi)$  是高斯或正态随机变量, 则称  $\mathbf{x}(\xi)$  为高斯或正态随机向量。实高斯随机向量和复高斯随机向量的概率密度函数表示稍有不同。

一个均值向量为  $\bar{\mathbf{x}} = [\bar{x}_1, \dots, \bar{x}_m]^T$  和协方差矩阵为  $\Gamma_x = E\{(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T\}$  的实高斯随机向量记作  $\mathbf{x} \sim N(\bar{\mathbf{x}}, \Gamma_x)$ 。若高斯随机向量的各元素为独立同分布 (independent identically distributed, iid), 则协方差矩阵  $\Gamma_x = E\{(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T\} = \text{diag}(\sigma_1^2, \dots, \sigma_m^2)$ , 其中  $\sigma_i^2 = E\{(x_i - \bar{x}_i)^2\}$  是高斯随机变量  $x_i$  的方差。

在各元素相互统计独立的条件下, 高斯随机向量的概率密度函数是向量的  $m$  个随机变量的联合概率密度函数

$$\begin{aligned} f(\mathbf{x}) &= f(x_1, \dots, x_m) = f(x_1) \cdots f(x_m) \\ &= \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{(x_1 - \bar{x}_1)^2}{2\sigma_1^2}\right) \cdots \frac{1}{\sqrt{2\pi\sigma_m^2}} \exp\left(-\frac{(x_m - \bar{x}_m)^2}{2\sigma_m^2}\right) \\ &= \frac{1}{(2\pi)^{m/2}\sigma_1 \cdots \sigma_m} \exp\left(-\frac{(x_1 - \bar{x}_1)^2}{2\sigma_1^2} - \cdots - \frac{(x_m - \bar{x}_m)^2}{2\sigma_m^2}\right) \\ &= \frac{1}{(2\pi)^{m/2}|\Gamma_x|^{1/2}} \exp\left(-\frac{1}{2}[\mathbf{x} - \bar{\mathbf{x}}]^T \begin{bmatrix} \sigma_1^{-2} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_m^{-2} \end{bmatrix} [\mathbf{x} - \bar{\mathbf{x}}]\right) \end{aligned}$$

整理后, 即可得到各元素统计独立的高斯随机向量  $\mathbf{x} \sim N(\bar{\mathbf{x}}, \Gamma_x)$  的概率密度函数为

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{m/2}|\Gamma_x|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}})^T \Gamma_x^{-1} (\mathbf{x} - \bar{\mathbf{x}})\right) \quad (1.5.31)$$

若元素之间不相互统计独立, 则高斯随机向量  $\mathbf{x} \sim N(\bar{\mathbf{x}}, \Gamma_x)$  的概率密度函数仍然由式 (1.5.31) 给出, 但指数项为 [392, 413]

$$(\mathbf{x} - \bar{\mathbf{x}})^T \Gamma_x^{-1} (\mathbf{x} - \bar{\mathbf{x}}) = \sum_{i=1}^m \sum_{j=1}^m [\Gamma_x^{-1}]_{i,j} (x_i - \mu_i)(x_j - \mu_j) \quad (1.5.32)$$

式中,  $[\Gamma_x^{-1}]_{i,j}$  表示逆矩阵  $\Gamma_x^{-1}$  的  $(i, j)$  元素,  $\mu_i = E\{x_i\}$  是随机变量  $x_i$  的均值。

实高斯随机向量的特征函数为

$$\Phi_x(\omega) = \exp\left(j\omega^T \mu_x - \frac{1}{2}\omega^T \Gamma_x \omega\right) \quad (1.5.33)$$

式中,  $\omega = [\omega_1, \dots, \omega_m]^T$ 。

令  $x = [x_1, \dots, x_m]^T$ , 其每个元素服从复正态分布, 即  $x_i \sim CN(\mu_i, \sigma_i^2)$ , 则  $x$  称为复高斯随机向量, 记作  $x \sim CN(\mu_x, \Gamma_x)$ , 其中,  $\mu_x = [\mu_1, \dots, \mu_m]^T$  和  $\Gamma$  分别为随机向量  $x$  的均值向量和协方差矩阵。若  $x_i = u_i + jv_i$ , 并且实随机向量  $[u_1, v_1]^T, \dots, [u_m, v_m]^T$  统计独立, 则复随机正态向量  $x$  的概率密度函数为 [413, p.35-5]

$$f(x) = \prod_{i=1}^m f(x_i) = \left(\pi^m \prod_{i=1}^m \sigma_i^2\right)^{-1} \exp\left(-\sum_{i=1}^m \frac{1}{\sigma_i^2} |x_i - \mu_i|^2\right) \quad (1.5.34)$$

$$= \frac{1}{\pi^m |\Gamma_x|} \exp[-(x - \mu_x)^H \Gamma_x^{-1} (x - \mu_x)] \quad (1.5.35)$$

式中,  $\Gamma_x = \text{diag}(\sigma_1^2, \dots, \sigma_m^2)$ 。复高斯随机向量的特征函数由下式给出

$$\Phi_x(\omega) = \exp\left[j \operatorname{Re}(\omega^H \mu_x) - \frac{1}{4} \omega^H \Gamma_x \omega\right] \quad (1.5.36)$$

高斯随机向量具有以下重要性质。

- (1) 概率密度函数由均值向量和协方差矩阵完全描述。
- (2) 若高斯随机向量的各个分量相互统计不相关, 则它们也是统计独立的。
- (3) 均值向量  $\mu_x$  和协方差矩阵  $\Gamma_x$  的高斯随机向量  $x$  的线性变换  $y(\xi) = Ax(\xi)$  仍然为高斯随机向量, 其概率密度函数为

$$f(y) = \frac{1}{(2\pi)^{m/2} |\Gamma_y|^{1/2}} \exp\left[-\frac{1}{2}(y - \mu_y)^T \Gamma_y^{-1} (y - \mu_y)\right] \quad (\text{实高斯随机向量}) \quad (1.5.37)$$

$$f(y) = \frac{1}{\pi^m |\Gamma_y|} \exp[-(y - \mu_y)^H \Gamma_y^{-1} (y - \mu_y)] \quad (\text{复正态随机向量}) \quad (1.5.38)$$

在阵列处理、无线通信和多信道信号处理中, 常常使用多个传感器或者阵元接收多路信号。在大多数情况下, 可以假定每个传感器上的加性噪声都是高斯白噪声, 并且这些传感器上的加性高斯白噪声是彼此统计不相关的。

**例 1.5.1** 零均值的实高斯白噪声向量  $x(t) = [x_1(t), \dots, x_m(t)]^T$  的各个元素为相互统计不相关的实高斯白噪声过程。若这些高斯白噪声具有相同的方差  $\sigma^2$ , 则有

$$c_{x_i, x_j} = r_{x_i, x_j} = \begin{cases} \sigma^2, & i = j \\ 0, & i \neq j \end{cases} \quad (1.5.39)$$

于是, 实高斯白噪声向量的自协方差矩阵

$$C_x = R_x = E\{x(t)x^T(t)\} = \begin{bmatrix} r_{x_1, x_1} & \cdots & r_{x_1, x_m} \\ \vdots & \ddots & \vdots \\ r_{x_m, x_1} & \cdots & r_{x_m, x_m} \end{bmatrix} = \sigma^2 I$$

因此, 实高斯白噪声向量的统计表示为

$$\mathbb{E}\{\boldsymbol{x}(t)\} = \mathbf{0} \quad \text{和} \quad \mathbb{E}\{\boldsymbol{x}(t)\boldsymbol{x}^T(t)\} = \sigma^2 \mathbf{I} \quad (1.5.40)$$

**例 1.5.2** 复高斯随机向量  $\boldsymbol{x}(t) = [x_1(t), \dots, x_m(t)]^T$  的各个元素为复高斯白噪声, 它们彼此统计不相关。若它们都具有零均值和相同的方差  $\sigma^2$ , 则意味着每一个复高斯白噪声过程的实部  $x_{Rk}(t)$  和虚部  $x_{Ik}(t)$  是两个相互统计独立的实高斯白噪声过程, 它们具有相同的方差。因此,  $x_k(t)$  为零均值和方差  $\sigma^2$  的高斯白噪声过程意味着

$$\begin{aligned} \mathbb{E}\{x_{Rk}(t)\} &= 0, \quad \mathbb{E}\{x_{Ik}(t)\} = 0 \\ \mathbb{E}\{x_{Rk}^2(t)\} &= \mathbb{E}\{x_{Ik}^2(t)\} = \frac{1}{2}\sigma^2 \\ \mathbb{E}\{x_{Rk}(t)x_{Ik}(t)\} &= 0 \\ \mathbb{E}\{x_k(t)x_k^*(t)\} &= \mathbb{E}\{x_{Rk}^2(t)\} + \mathbb{E}\{x_{Ik}^2(t)\} = \sigma^2 \end{aligned}$$

由上述条件知

$$\begin{aligned} \mathbb{E}\{x_k^2(t)\} &= \mathbb{E}\{[x_{Rk}(t) + jx_{Ik}(t)]^2\} \\ &= \mathbb{E}\{x_{Rk}^2(t)\} - \mathbb{E}\{x_{Ik}^2(t)\} + j2\mathbb{E}\{x_{Rk}(t)x_{Ik}(t)\} \\ &= \frac{1}{2}\sigma^2 - \frac{1}{2}\sigma^2 + 0 = 0 \end{aligned}$$

由于  $x_1(t), \dots, x_m(t)$  是  $m$  个彼此不相关的高斯白噪声过程, 故

$$\mathbb{E}\{x_i(t)x_j(t)\} = 0, \quad \mathbb{E}\{x_i(t)x_j^*(t)\} = 0, \quad i \neq j$$

综合以上条件, 即可得到复高斯白噪声向量  $\boldsymbol{x}(t)$  的统计表示为

$$\mathbb{E}\{\boldsymbol{x}(t)\} = \mathbf{0} \quad (1.5.41)$$

$$\mathbb{E}\{\boldsymbol{x}(t)\boldsymbol{x}^H(t)\} = \sigma^2 \mathbf{I} \quad (1.5.42)$$

$$\mathbb{E}\{\boldsymbol{x}(t)\boldsymbol{x}^T(t)\} = \mathbf{O} \quad (1.5.43)$$

注意复高斯白噪声向量和实高斯白噪声向量的统计表示的这一区别。

各个元素具有零均值和相同方差  $\sigma^2$  的实和复高斯白噪声向量  $\boldsymbol{x}(t)$  常用符号  $\boldsymbol{x}(t) \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$  和  $\boldsymbol{x}(t) \sim CN(\mathbf{0}, \sigma^2 \mathbf{I})$  分别表示。

## 1.6 矩阵的性能指标

一个  $m \times n$  维矩阵是一种含有  $m \times n$  个元素的多变量表示。在数学中, 经常希望使用一个数或标量来概括多变量表示。其中, 矩阵的性能指标就是这类典型的例子。前面介绍的矩阵的内积与范数的定义形式尽管有多种, 但它们都是矩阵的一种标量函数。本节将介绍概括矩阵性质的其他几个重要的标量指标, 它们分别是矩阵的二次型、行列式、特征值、迹和秩。

### 1.6.1 矩阵的二次型

任意一个正方矩阵  $A$  的二次型定义为  $\mathbf{x}^H A \mathbf{x}$ , 其中  $\mathbf{x}$  可以是任意的非零复向量。

以实矩阵为例, 考查二次型

$$\begin{aligned}\mathbf{x}^T A \mathbf{x} &= [x_1, x_2, x_3] \begin{bmatrix} 1 & 4 & 2 \\ -1 & 7 & 5 \\ -1 & 6 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\ &= x_1^2 + 7x_2^2 + 3x_3^2 + 3x_1x_2 + x_1x_3 + 11x_2x_3\end{aligned}$$

这是变元  $x$  的二次型函数, 故称  $\mathbf{x}^T A \mathbf{x}$  为矩阵  $A$  的二次型。

推而广之, 若  $\mathbf{x} = [x_1, \dots, x_n]^T$ , 且  $n \times n$  矩阵  $A$  的元素为  $a_{ij}$ , 则二次型

$$\begin{aligned}\mathbf{x}^T A \mathbf{x} &= \sum_{i=1}^n \sum_{j=1}^n x_i x_j a_{ij} = \sum_{i=1}^n a_{ii} x_i^2 + \sum_{i=1, i \neq j}^n \sum_{j=1}^n a_{ij} x_i x_j \\ &= \sum_{i=1}^n a_{ii} x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^n (a_{ij} + a_{ji}) x_i x_j\end{aligned}$$

根据这一公式, 显然

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix} 1 & 4 & 2 \\ -1 & 7 & 5 \\ -1 & 6 & 3 \end{bmatrix}, \quad \mathbf{B} = \mathbf{A}^T = \begin{bmatrix} 1 & -1 & -1 \\ 4 & 7 & 6 \\ 2 & 5 & 3 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1.0 & 1.5 & 0.5 \\ 1.5 & 7.0 & 5.5 \\ 0.5 & 5.5 & 3.0 \end{bmatrix}, \\ \mathbf{D} &= \begin{bmatrix} 1 & 114 & 52 \\ -111 & 7 & 2 \\ -51 & 9 & 3 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} 1 & 114 & 52 \\ -111 & 7 & 4 \\ -51 & 7 & 3 \end{bmatrix}, \quad \dots\end{aligned}$$

具有相同的二次型, 即

$$\begin{aligned}\mathbf{x}^T A \mathbf{x} &= \mathbf{x}^T B \mathbf{x} = \mathbf{x}^T C \mathbf{x} = \mathbf{x}^T D \mathbf{x} = \mathbf{x}^T F \mathbf{x} \\ &= x_1^2 + 7x_2^2 + 3x_3^2 + 3x_1x_2 + x_1x_3 + 11x_2x_3\end{aligned}$$

这就是说, 对于任何一个二次型函数

$$f(x_1, \dots, x_n) = \sum_{i=1}^n a_{ii} x_i^2 + \sum_{i=1, i \neq j}^n \sum_{j=1}^n a_{ij} x_i x_j$$

而言, 存许多矩阵  $A$ , 它们的二次型  $\mathbf{x}^T A \mathbf{x} = f(x_1, \dots, x_n)$  相同。但是, 只有一个唯一的对称矩阵  $A$  满足  $\mathbf{x}^T A \mathbf{x} = f(x_1, \dots, x_n)$ , 其元素为  $a_{ij} = a_{ji} = \frac{1}{2}(a_{ij} + a_{ji})$ , 其中,  $i = 1, \dots, n, j = 1, \dots, n$ 。因此, 为了保证定义的唯一性, 在讨论矩阵  $A$  的二次型时, 有必要假定  $A$  为实对称矩阵或复共轭对称(即 Hermitian)矩阵。这一假定还能够保证二次型函数一定是实值函数, 因为  $(\mathbf{x}^H A \mathbf{x})^* = (\mathbf{x}^H A \mathbf{x})^H = \mathbf{x}^H A^H \mathbf{x} = \mathbf{x}^H A \mathbf{x}$  对任意复共轭对称矩阵  $A$  和非零复向量  $\mathbf{x}$  均成立。实值函数的基本优点之一是适合于同零值比较大小。

如果将大于零的二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x}$  称为正定的二次型，则与之对应的 Hermitian 矩阵称为正定矩阵。类似地，还可以定义 Hermitian 矩阵的半正定性、负定性和半负定性。

**定义 1.6.1** 一个复共轭对称矩阵  $\mathbf{A}$  称为：

- (1) 正定矩阵，记作  $\mathbf{A} > 0$ ，若 二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x} > 0, \forall \mathbf{x} \neq \mathbf{0}$ ；
- (2) 半正定矩阵，记作  $\mathbf{A} \succeq 0$ ，若 二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x} \geq 0, \forall \mathbf{x} \neq \mathbf{0}$  (也称非负定的)；
- (3) 负定矩阵，记作  $\mathbf{A} < 0$ ，若 二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x} < 0, \forall \mathbf{x} \neq \mathbf{0}$ ；
- (4) 半负定矩阵，记作  $\mathbf{A} \preceq 0$ ，若 二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x} \leq 0, \forall \mathbf{x} \neq \mathbf{0}$  (也称非正定的)；
- (5) 不定矩阵，若二次型  $\mathbf{x}^H \mathbf{A} \mathbf{x}$  既可能取正值，也可能取负值。

**例 1.6.1** 实对称矩阵

$$\mathbf{R} = \begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$

是正定的，因为二次型  $\mathbf{x}^H \mathbf{R} \mathbf{x} = 2x_1^2 + x_2^2 + 2x_3^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 > 0$ ，除非  $x_1 = x_2 = x_3 = 0$ 。

一句话小结：作为一个性能指标，矩阵的二次型刻画矩阵的正定性。

## 1.6.2 行列式

一个  $n \times n$  正方矩阵  $\mathbf{A}$  的行列式记作  $\det(\mathbf{A})$  或  $|\mathbf{A}|$ ，定义为

$$\det(\mathbf{A}) = |\mathbf{A}| = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} \quad (1.6.1)$$

若  $\mathbf{A} = \{a\} \in \mathbb{C}^{1 \times 1}$ ，则其行列式由  $\det(\mathbf{A}) = a$  给出。

矩阵  $\mathbf{A}$  去掉第  $i$  行和第  $j$  列之后得到的剩余行列式记作  $A_{ij}$ ，称为元素  $a_{ij}$  的余子式 (cofactor)。特别地，当  $j = i$  时， $A_i = A_{ii}$  称为  $\mathbf{A}$  的主子式。若令  $\mathbf{A}_{ij}$  是  $n \times n$  矩阵  $\mathbf{A}$  删去第  $i$  行和第  $j$  列之后得到的  $(n-1) \times (n-1)$  子矩阵，则

$$A_{ij} = (-1)^{i+j} \det(\mathbf{A}_{ij}) \quad (1.6.2)$$

一个  $n \times n$  矩阵的行列式等于其任意行 (或列) 的元素与相对应的余子式乘积之和，即有

$$\det(\mathbf{A}) = a_{i1} A_{i1} + \cdots + a_{in} A_{in} = \sum_{j=1}^n a_{ij} (-1)^{i+j} \det(\mathbf{A}_{ij}) \quad (1.6.3)$$

或者

$$\det(\mathbf{A}) = a_{1j} A_{1j} + \cdots + a_{nj} A_{nj} = \sum_{i=1}^n a_{ij} (-1)^{i+j} \det(\mathbf{A}_{ij}) \quad (1.6.4)$$

因此，行列式可以递推计算： $n$  阶行列式由  $(n-1)$  阶行列式计算， $(n-1)$  阶行列式则由  $(n-2)$  行列式计算等。

特别地, 对于  $3 \times 3$  矩阵  $\mathbf{A}$ , 其行列式可以通过

$$\begin{aligned}\det(\mathbf{A}) &= \det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = a_{11}A_{11} + a_{12}A_{12} + a_{13}A_{13} \\ &= a_{11}(-1)^{1+1} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} + a_{12}(-1)^{1+2} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}(-1)^{1+3} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{33} \end{vmatrix} \\ &= a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31}) + a_{13}(a_{21}a_{33} - a_{22}a_{31})\end{aligned}$$

递推计算。这一方法称为三阶行列式计算的对角线法。

**定义 1.6.2** 行列式不等于零的矩阵称为非奇异矩阵。

### 1. 关于行列式的等式关系 [324]

- (1) 如果矩阵的两行 (或列) 互换位置, 则行列式数值保持不变, 但符号改变。
- (2) 若矩阵的某行 (或列) 是其他行 (或列) 的线性组合, 则  $\det(\mathbf{A}) = 0$ 。特别地, 若某行 (或列) 与另一行 (或列) 成正比或相等, 或者某行 (或列) 的元素均等于零, 则  $\det(\mathbf{A}) = 0$ 。
- (3) 单位矩阵的行列式等于 1, 即  $\det(\mathbf{I}) = 1$ 。
- (4) 任何一个正方矩阵  $\mathbf{A}$  和它的转置矩阵  $\mathbf{A}^T$  具有相同的行列式, 即  $\det(\mathbf{A}) = \det(\mathbf{A}^T)$ , 但  $\det(\mathbf{A}^H) = [\det(\mathbf{A}^T)]^*$ 。
- (5) 一个 Hermitian 矩阵的行列式为实数, 因为

$$\det(\mathbf{A}) = \det(\mathbf{A}^H) = \det(\mathbf{A}^T) \Rightarrow \det(\mathbf{A}) = \det(\mathbf{A}^*) = [\det(\mathbf{A})]^* \quad (1.6.5)$$

- (6) 两个矩阵乘积的行列式等于它们的行列式的乘积, 即

$$\det(\mathbf{AB}) = \det(\mathbf{A}) \det(\mathbf{B}), \quad \mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n} \quad (1.6.6)$$

- (7) 给定一个任意的常数 (可以是复数)  $c$ , 则  $\det(c\mathbf{A}) = c^n \det(\mathbf{A})$ 。
- (8) 若  $\mathbf{A}$  非奇异, 则  $\det(\mathbf{A}^{-1}) = 1/\det(\mathbf{A})$ 。
- (9) 三角 (上三角或下三角) 矩阵  $\mathbf{A}$  的行列式等于其主对角线所有元素的乘积

$$\det(\mathbf{A}) = \prod_{i=1}^n a_{ii}$$

一个对角矩阵  $\mathbf{A} = \text{diag}(a_{11}, \dots, a_{nn})$  的行列式也等于其对角元素的乘积。

- (10) 对于矩阵  $\mathbf{A}_{m \times m}, \mathbf{B}_{m \times n}, \mathbf{C}_{n \times m}, \mathbf{D}_{n \times n}$ , 分块矩阵的行列式满足

$$\mathbf{A} \text{ 非奇异} \iff \det \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \det(\mathbf{A}) \det(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}) \quad (1.6.7)$$

$$\mathbf{D} \text{ 非奇异} \iff \det \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \det(\mathbf{D}) \det(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}) \quad (1.6.8)$$

下面证明式 (1.6.7)

$$\begin{aligned}\det \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} &= \det \left( \begin{bmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{C} & \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A}^{-1}\mathbf{B} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \right) \\ &= \det(\mathbf{A}) \cdot \det(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})\end{aligned}$$

类似地，可证明式 (1.6.8)。

## 2. 关于行列式的不等式关系<sup>[324]</sup>

(1) Cauchy-Schwartz 不等式：若  $\mathbf{A}, \mathbf{B}$  都是  $m \times n$  矩阵，则

$$|\det(\mathbf{A}^H \mathbf{B})|^2 \leq \det(\mathbf{A}^H \mathbf{A}) \det(\mathbf{B}^H \mathbf{B})$$

(2) Hadamard 不等式：对于  $m \times m$  矩阵  $\mathbf{A}$ ，有

$$\det(\mathbf{A}) \leq \prod_{i=1}^m \left( \sum_{j=1}^m |a_{ij}|^2 \right)^{1/2}$$

(3) Fischer 不等式：若  $\mathbf{A}_{m \times m}, \mathbf{B}_{m \times m}, \mathbf{C}_{n \times n}$ ，则

$$\det \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^H & \mathbf{C} \end{pmatrix} \leq \det(\mathbf{A}) \det(\mathbf{C})$$

(4) Minkowski 不等式：若  $\mathbf{A}_{m \times m} \neq \mathbf{O}_{m \times m}, \mathbf{B}_{m \times m} \neq \mathbf{O}_{m \times m}$  半正定，则

$$\sqrt[m]{\det(\mathbf{A} + \mathbf{B})} \geq \sqrt[m]{\det(\mathbf{A})} + \sqrt[m]{\det(\mathbf{B})}$$

(5) 正定矩阵  $\mathbf{A}$  的行列式大于 0，即  $\det(\mathbf{A}) > 0$ 。

(6) 半正定矩阵  $\mathbf{A}$  的行列式大于或者等于 0，即  $\det(\mathbf{A}) \geq 0$ 。

(7) 若  $m \times m$  矩阵  $\mathbf{A}$  半正定，则  $(\det(\mathbf{A}))^{1/m} \leq \frac{1}{m} \det(\mathbf{A})$ 。

(8) 若矩阵  $\mathbf{A}_{m \times m}, \mathbf{B}_{m \times m}$  均半正定，则  $\det(\mathbf{A} + \mathbf{B}) \geq \det(\mathbf{A}) + \det(\mathbf{B})$ 。

(9) 若  $\mathbf{A}_{m \times m}$  正定， $\mathbf{B}_{m \times m}$  半正定，则  $\det(\mathbf{A} + \mathbf{B}) \geq \det(\mathbf{A})$ 。

(10) 若  $\mathbf{A}_{m \times m}$  正定， $\mathbf{B}_{m \times m}$  半负定，则  $\det(\mathbf{A} + \mathbf{B}) \leq \det(\mathbf{A})$ 。

一句话小结：作为一个性能指标，矩阵的行列式主要刻画矩阵的奇异性。

### 1.6.3 矩阵的特征值

若  $n \times 1$  非零向量  $\mathbf{u}$  作为线性变换  $\mathcal{L}$  的输入时，所产生的输出与输入只相差一个比例因子  $\lambda$ ，即

$$\mathcal{L}\mathbf{u} = \lambda\mathbf{u}, \quad \mathbf{u} \neq 0 \tag{1.6.9}$$

则称标量  $\lambda$  和向量  $\mathbf{u}$  分别为线性变换  $\mathcal{L}$  的特征值和特征向量。由于  $\mathcal{L}\mathbf{u} = \lambda\mathbf{u}$  意味着输入向量在线性变换下能够保持方向不变，所以  $\mathbf{u}$  刻画了线性变换或系统固有的向量特

征。这就是特征向量的物理含义所在，而特征值  $\lambda$  则可视为线性变换或系统对特定的特征向量  $\mathbf{u}$  所固有的增益。

当线性变换  $\mathcal{L}$  为  $n \times n$  矩阵  $\mathbf{A}$  时，上述定义便引申为矩阵的特征值和特征向量的定义：若线性代数方程

$$\mathbf{A}\mathbf{u} = \lambda\mathbf{u} \quad (1.6.10)$$

具有  $n \times 1$  非零解（向量） $\mathbf{u}$ ，则标量  $\lambda$  称为矩阵  $\mathbf{A}$  的一个特征值，而  $\mathbf{u}$  称为  $\mathbf{A}$  的对应于  $\lambda$  的特征向量。式 (1.6.10) 可视为特征值的第一定义公式。

线性方程式 (1.6.10) 可以等价写作

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{u} = \mathbf{0} \quad (1.6.11)$$

由于上式对非零向量  $\mathbf{u}$  成立，故线性代数方程式 (1.6.10) 存在非零解  $\mathbf{u} \neq \mathbf{0}$  的唯一条件是矩阵  $\mathbf{A} - \lambda\mathbf{I}$  的行列式等于零，即

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0 \quad (1.6.12)$$

这是特征值的第二定义公式。注意，一个  $n \times n$  矩阵只有  $n$  个特征值，但其中有些特征值有可能取相同值。

特征值的第二定义公式 (1.6.12) 反映了以下事实：

(1) 若式 (1.6.12) 对  $\lambda = 0$  成立，则直接有  $\det(\mathbf{A}) = 0$ 。这意味着，只要矩阵  $\mathbf{A}$  有一个特征值为零，则该矩阵一定是奇异矩阵。

(2) 只有零矩阵的全部特征值为零，任何奇异的非零矩阵  $\mathbf{A}$  一定存在非零的特征值。奇异矩阵的非零特征值意味着，原矩阵的所有对角元素同时减去该特征值后，所得矩阵仍然是奇异矩阵。注意，奇异矩阵的每个对角元素减去不是特征值的同一标量后，所得矩阵的行列式一定不等于零，即所得矩阵是非奇异的。

(3) 若矩阵  $\mathbf{A}$  的所有特征值都不等于零，则原矩阵的行列式一定不等于零，因而它一定是非奇异矩阵。然而，非奇异矩阵的所有对角元素同时减去它的任何一个非零特征值后，所得矩阵一定是奇异的，因为它的行列式等于零。

事实 (1) 说明，零特征值反映矩阵的奇异性；而事实 (2) 和事实 (3) 则表明，特征值可以刻画矩阵所有对角元素的结构。在一个矩阵的一些重要性质的描述中，对角元素往往起着主导性作用。

矩阵  $\mathbf{A}$  的特征值常用符号  $\text{eig}(\mathbf{A})$  表示。下面是特征值的一些基本性质：

- (1)  $\text{eig}(\mathbf{AB}) = \text{eig}(\mathbf{BA})$ 。
- (2)  $m \times n$  矩阵  $\mathbf{A}$  最多有  $\min\{m, n\}$  个不同特征值。
- (3) 若  $\text{rank}(\mathbf{A}) = r$ ，则矩阵  $\mathbf{A}$  最多有  $r$  个非零特征值。
- (4) 逆矩阵的特征值  $\text{eig}(\mathbf{A}^{-1}) = 1/\text{eig}(\mathbf{A})$ 。

(5) 令  $I$  为单位矩阵, 则

$$\text{eig}(I + cA) = 1 + c \text{ eig}(A) \quad (1.6.13)$$

$$\text{eig}(A - cI) = \text{eig}(A) - c \quad (1.6.14)$$

一个复共轭对称矩阵的正定性与其特征值有着密切的关系。

**引理 1.6.1** 正定矩阵的所有特征值都是正实数。

**证明** 设  $A$  为正定矩阵, 则其二次型  $x^H A x > 0$  对任意非零向量  $x$  成立。若  $\lambda$  是正定矩阵  $A$  的任意一个特征值, 即  $Au = \lambda u$ , 则  $u^H A u = u^H \lambda u$ , 从而有  $\lambda = u^H A u / u^H u$  一定是正实数, 因为它是两个正实数之比。 ■

注意到矩阵  $A$  对应于特征值  $\lambda$  的特征向量  $u$  的内积  $u^H u$  是一个恒大于零的实数, 对上述引理加以推广和运用, 易知矩阵的正定性和半正定性等都可以用特征值描述:

- (1) 正定矩阵: 所有特征值取正实数的矩阵。
- (2) 半正定矩阵: 各个特征值取非负实数的矩阵。
- (3) 负定矩阵: 全部特征值为负实数的矩阵。
- (4) 半负定矩阵: 每个特征值取非正实数的矩阵。
- (5) 不定矩阵: 特征值有些取正实数, 另一些取负实数的矩阵。

若  $A$  是一个正定或者半正定矩阵, 则

$$\det(A) \leq \prod_i A_{ii} \quad (1.6.15)$$

这一不等式称为 Hadamard 不等式 [238, p.477]。

一句话小结: 作为一个性能指标, 矩阵的特征值既刻画原矩阵的奇异性, 又反映原矩阵所有对角元素的结构, 还刻画矩阵的正定性。

之所以称为矩阵的特征值, 正是因为它反映了矩阵的奇异性、正定性以及对角元素的特殊结构等重要特征。

#### 1.6.4 矩阵的迹

**定义 1.6.3**  $n \times n$  矩阵  $A$  的对角元素之和称为  $A$  的迹 (trace), 记作  $\text{tr}(A)$ , 即有

$$\text{tr}(A) = a_{11} + \cdots + a_{nn} = \sum_{i=1}^n a_{ii} \quad (1.6.16)$$

非正方矩阵无迹的定义。下面是矩阵的迹满足的等式、不等式关系与其他一些性质。

##### 1. 关于迹的等式 [324]

- (1) 若  $A$  和  $B$  均为  $n \times n$  矩阵, 则  $\text{tr}(A \pm B) = \text{tr}(A) \pm \text{tr}(B)$ 。

(2) 若  $\mathbf{A}$  和  $\mathbf{B}$  均为  $n \times n$  矩阵, 并且  $c_1$  和  $c_2$  为常数, 则  $\text{tr}(c_1\mathbf{A} \pm c_2\mathbf{B}) = c_1\text{tr}(\mathbf{A}) \pm c_2\text{tr}(\mathbf{B})$ 。特别地, 若  $\mathbf{B} = \mathbf{O}$ , 则  $\text{tr}(c\mathbf{A}) = c\text{tr}(\mathbf{A})$ 。

(3) 矩阵  $\mathbf{A}$  的转置、复数共轭和复共轭转置的迹分别为  $\text{tr}(\mathbf{A}^T) = \text{tr}(\mathbf{A})$ ,  $\text{tr}(\mathbf{A}^*) = [\text{tr}(\mathbf{A})]^*$  和  $\text{tr}(\mathbf{A}^H) = [\text{tr}(\mathbf{A})]^*$ 。

(4) 若  $\mathbf{A} \in \mathbb{C}^{m \times n}$ ,  $\mathbf{B} \in \mathbb{C}^{n \times m}$ , 则  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ 。

(5) 若  $\mathbf{A}$  是一个  $m \times n$  矩阵, 则  $\text{tr}(\mathbf{A}^H \mathbf{A}) = 0 \iff \mathbf{A} = \mathbf{O}_{m \times n}$  (零矩阵)。

(6)  $\mathbf{x}^H \mathbf{Ax} = \text{tr}(\mathbf{Axx}^H)$  和  $\mathbf{y}^H \mathbf{x} = \text{tr}(\mathbf{xy}^H)$ 。

(7) 迹等于特征值之和, 即  $\text{tr}(\mathbf{A}) = \lambda_1 + \dots + \lambda_n$ 。

(8) 分块矩阵的迹满足

$$\text{tr} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{D})$$

式中,  $\mathbf{A} \in \mathbb{C}^{m \times m}$ ,  $\mathbf{B} \in \mathbb{C}^{m \times n}$ ,  $\mathbf{C} \in \mathbb{C}^{n \times m}$ ,  $\mathbf{D} \in \mathbb{C}^{n \times n}$ 。

(9) 对于任何正整数  $k$ , 有

$$\text{tr}(\mathbf{A}^k) = \sum_{i=1}^n \lambda_i^k \quad (1.6.17)$$

灵活运用迹的等式  $\text{tr}(\mathbf{UV}) = \text{tr}(\mathbf{VU})$ , 可以得到一些常用的重要结果。例如, 矩阵  $\mathbf{A}^H \mathbf{A}$  和  $\mathbf{AA}^H$  的迹相等, 且有

$$\text{tr}(\mathbf{A}^H \mathbf{A}) = \text{tr}(\mathbf{AA}^H) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} a_{ij}^* = \sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2 \quad (1.6.18)$$

又如, 在迹的等式  $\text{tr}(\mathbf{UV}) = \text{tr}(\mathbf{VU})$  中, 若分别令  $\mathbf{U} = \mathbf{A}$ ,  $\mathbf{V} = \mathbf{BC}$  和  $\mathbf{U} = \mathbf{AB}$ ,  $\mathbf{V} = \mathbf{C}$ , 则有

$$\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{BCA}) = \text{tr}(\mathbf{CAB}) \quad (1.6.19)$$

类似地, 若分别令  $\mathbf{U} = \mathbf{A}$ ,  $\mathbf{V} = \mathbf{BCD}$ ;  $\mathbf{U} = \mathbf{AB}$ ,  $\mathbf{V} = \mathbf{CD}$  及  $\mathbf{U} = \mathbf{ABC}$ ,  $\mathbf{V} = \mathbf{D}$ , 又有

$$\text{tr}(\mathbf{ABCD}) = \text{tr}(\mathbf{BCDA}) = \text{tr}(\mathbf{CDAB}) = \text{tr}(\mathbf{DABC}) \quad (1.6.20)$$

利用等式 (1.6.19) 还易知, 若矩阵  $\mathbf{A}$  与  $\mathbf{B}$  均为  $m \times m$  矩阵, 且  $\mathbf{B}$  非奇异, 则

$$\text{tr}(\mathbf{BAB}^{-1}) = \text{tr}(\mathbf{B}^{-1}\mathbf{AB}) = \text{tr}(\mathbf{ABB}^{-1}) = \text{tr}(\mathbf{A}) \quad (1.6.21)$$

## 2. 关于迹的不等式<sup>[324]</sup>

(1) 对一个复矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$ , 有  $\text{tr}(\mathbf{A}^H \mathbf{A}) = \text{tr}(\mathbf{AA}^H) \geq 0$ 。

(2) Schur 不等式  $\text{tr}(\mathbf{A}^2) \leq \text{tr}(\mathbf{A}^T \mathbf{A})$ 。

(3) 若  $A, B$  均为  $m \times n$  矩阵, 则

$$\text{tr}[(A^T B)^2] \leq \text{tr}(A^T A) \text{tr}(B^T B) \quad (\text{Cauchy-Schwartz 不等式})$$

$$\text{tr}[(A^T B)^2] \leq \text{tr}(A^T A B^T B)$$

$$\text{tr}[(A^T B)^2] \leq \text{tr}(A A^T B B^T)$$

$$(4) \text{tr}[(A + B)(A + B)^T] \leq 2[\text{tr}(A A^T) + \text{tr}(B B^T)]。$$

$$(5) \text{若 } A \text{ 和 } B \text{ 为 } m \times m \text{ 对称矩阵, 则 } \text{tr}(AB) \leq \frac{1}{2}\text{tr}(A^2 + B^2)。$$

类似于向量的 Euclidean 范数  $\|x\|_2 = (x^H x)^{1/2}$ , 一个  $m \times n$  复矩阵  $A$  的 Frobenius 范数也可利用  $m \times m$  矩阵  $A^H A$  或者  $n \times n$  矩阵  $AA^H$  的迹定义为 [328, p.10]

$$\|A\|_F = \sqrt{\text{tr}(A^H A)} = \sqrt{\text{tr}(AA^H)} \quad (1.6.22)$$

一句话小结: 作为一个性能指标, 矩阵的迹反映所有特征值之和。

### 1.6.5 矩阵的秩

仅当  $n \times n$  矩阵  $A$  存在逆矩阵  $A^{-1}$  时, 矩阵方程  $Ax = b$  有解  $x = A^{-1}b$ 。逆矩阵  $A^{-1}$  存在, 仅当行列式  $|A| \neq 0$ 。因此, 在求矩阵方程  $Ax = b$  的解  $x = A^{-1}b$  时, 需要事先确定行列式  $|A|$  是否等于零。一个  $n \times n$  矩阵的行列式不等于零, 当且仅当该矩阵的行或者列彼此线性无关。

上述讨论也可以推广到  $m \times n$  矩阵  $A$  的情况: 矩阵方程  $Ax = b$  是否有解, 取决于矩阵  $A$  的行或者列是否线性无关。此外, 在讨论矩阵  $A_{m \times n}$  的一些重要性质时, 线性无关的行和列也常常起着重要的作用。一个自然会问的问题是: “一个给定的矩阵  $A_{m \times n}$  究竟有多少个线性无关的行向量和列向量?”在回答这个问题之前, 首先让我们来考虑一个与之有关的问题“一组  $p$  维向量中最多能够有几个线性无关的向量?”

**定理 1.6.1**<sup>[444]</sup> 在  $p$  维(行或列)向量的集合之中, 最多存在  $p$  个线性无关的(行或列)向量。

有了上述定理, 即可回答“矩阵  $A_{m \times n}$  有多少个线性无关的行向量和列向量”这一重要问题。

**定理 1.6.2**<sup>[444]</sup> 矩阵  $A_{m \times n}$  的线性无关行数与线性无关列数相同。

从定理 1.6.2 出发, 可以引出矩阵的秩的定义。

**定义 1.6.4** 矩阵  $A_{m \times n}$  的秩定义为该矩阵中线性无关的行或列的数目。

需要指出, 矩阵的秩只是强调该矩阵的线性无关的行数和线性无关的列数, 并没有给出这些线性无关的行和列所在位置的任何信息。

矩阵方程  $A_{m \times n}x_{n \times 1} = b_{m \times 1}$  称为一致方程 (consistent equation), 若它至少有一个(精确)解。无任何精确解存在的矩阵方程称为非一致方程 (inconsistent equation)。根据矩阵  $A$  的秩的大小, 矩阵方程又可分为以下三种类型。

(1) 适定方程: 若  $m = n$ , 并且  $\text{rank}(\mathbf{A}) = n$ , 即矩阵  $\mathbf{A}$  非奇异, 则称矩阵方程  $\mathbf{Ax} = \mathbf{b}$  为适定 (well-determined) 方程。

(2) 欠定方程: 若独立的方程个数小于独立的未知参数个数, 则称矩阵方程  $\mathbf{Ax} = \mathbf{b}$  为欠定 (under-determined) 方程。

(3) 超定方程: 若独立的方程个数大于独立的未知参数个数, 则称矩阵方程  $\mathbf{Ax} = \mathbf{b}$  为超定 (over-determined) 方程。

下面是术语“适定”、“欠定”和“超定”的含义。

**适定的双层含义** 方程组的解是唯一的; 独立的方程个数与独立未知参数的个数相同, 正好可以唯一地确定该方程组的解。适定方程  $\mathbf{Ax} = \mathbf{b}$  的唯一解由  $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$  给出。适定方程为一致方程。

**欠定的含义** 独立的方程个数比独立的未知参数的个数少, 意味着方程个数不足于确定方程组的唯一解。事实上, 这样的方程组存在无穷多组解  $\mathbf{x}$ 。欠定方程为一致方程。

**超定的含义** 独立的方程个数超过独立的未知参数的个数, 对于确定方程组的唯一解显得方程过剩。因此, 超定方程  $\mathbf{Ax} = \mathbf{b}$  没有使得方程组严格满足的精确解  $\mathbf{x}$ 。超定方程为非一致方程。

根据定义 1.6.4, 秩  $\text{rank}(\mathbf{A}) = r_A$  的矩阵  $\mathbf{A}$  有  $r_A$  个线性无关的列向量。这  $r_A$  个线性无关的列向量的所有线性组合, 便形成了一个向量空间, 称为矩阵  $\mathbf{A}$  的列空间、 $\mathbf{A}$  的值域 (range) 或  $\mathbf{A}$  的流形, 常记为  $\mathcal{R}(\mathbf{A})$ 。列空间  $\mathcal{R}(\mathbf{A})$  具有维数  $r_A$ 。因此, 矩阵的秩也可以利用矩阵的列空间的维数定义。

**定义 1.6.5** 矩阵  $\mathbf{A}_{m \times n}$  的列空间  $\mathcal{R}(\mathbf{A})$  或  $\text{Col}(\mathbf{A})$  的维数定义为该矩阵的秩, 即有

$$r_A = \dim[\mathcal{R}(\mathbf{A})] = \dim[\text{Col}(\mathbf{A})] \quad (1.6.23)$$

由此定义知, 若矩阵  $\mathbf{A}$  的秩为  $r_A$ , 则该矩阵的列空间  $\mathcal{R}(\mathbf{A})$  是一个  $r_A$  维子空间。关于矩阵  $\mathbf{A}$  的秩的下列叙述等价, 每一叙述在不同的场合作用。

- (1)  $\text{rank}(\mathbf{A}) = k$ ;
- (2) 存在  $\mathbf{A}$  的  $k$  列且不多于  $k$  列组成一线性无关组;
- (3) 存在  $\mathbf{A}$  的  $k$  行且不多于  $k$  行组成一线性无关组;
- (4) 存在  $\mathbf{A}$  的一个  $k \times k$  子矩阵具有非零行列式, 而且  $\mathbf{A}$  的所有  $(k+1) \times (k+1)$  子矩阵都具有零行列式;

(5) 列空间  $\mathcal{R}(\mathbf{A})$  的维数等于  $k$ ;

(6)  $k = n - \dim[\text{Null}(\mathbf{A})]$ , 其中,  $\text{Null}(\mathbf{A})$  表示矩阵  $\mathbf{A}$  的零空间。

下面讨论矩阵秩的性质。由于这些性质与两个矩阵乘积的秩密切相关, 有必要先讨论乘积矩阵的秩。

**定理 1.6.3**<sup>[444]</sup> 乘积矩阵  $\mathbf{AB}$  的秩  $\text{rank}(\mathbf{AB})$  满足不等式

$$\text{rank}(\mathbf{AB}) \leq \min\{\text{rank}(\mathbf{A}), \text{rank}(\mathbf{B})\} \quad (1.6.24)$$

**引理 1.6.2**  $m \times n$  矩阵  $A$  左乘  $m \times m$  非奇异矩阵  $P$  或者右乘  $n \times n$  非奇异矩阵  $Q$ , 将不改变  $A$  的秩。

**证明** 由于  $m \times m$  矩阵  $P$  非奇异, 即  $\text{rank}(P) = m$ , 故  $\text{rank}(A) \leq \text{rank}(PA)$ 。令  $M = PA$ , 则根据定理 1.6.3 知  $\text{rank}(M) \leq \text{rank}(A)$ 。另外, 由  $A = P^{-1}M$  及定理 1.6.3 又有  $\text{rank}(A) \leq \text{rank}(M)$ 。于是,  $\text{rank}(A) = \text{rank}(M) = \text{rank}(PA)$ 。类似地, 可以证明  $\text{rank}(A) = \text{rank}(AQ)$ 。 ■

**引理 1.6.3**  $\text{rank}[A, B] \leq \text{rank}(A) + \text{rank}(B)$ 。

**引理 1.6.4**  $\text{rank}(A + B) \leq \text{rank}[A, B] \leq \text{rank}(A) + \text{rank}(B)$ 。

**引理 1.6.5** 对于  $m \times n$  矩阵  $A$  和  $n \times q$  矩阵  $B$ , 秩不等式  $\text{rank}(AB) \geq \text{rank}(A) + \text{rank}(B) - n$  成立。

矩阵的秩具有以下性质、等式关系和不等式关系。

### 1. 秩的性质

- (1) 秩是一个正整数。
- (2) 秩等于或小于矩阵的行数或列数。
- (3) 当  $n \times n$  矩阵  $A$  的秩等于  $n$  时, 则  $A$  是非奇异矩阵, 或称  $A$  满秩 (full rank)。
- (4) 如果  $\text{rank}(A_{m \times n}) < \min\{m, n\}$ , 则称  $A$  是秩亏缺的 (rank deficient)。
- (5) 若  $\text{rank}(A_{m \times n}) = m (< n)$ , 则称矩阵  $A$  具有满行秩 (full row rank)。
- (6) 若  $\text{rank}(A_{m \times n}) = n (< m)$ , 则称矩阵  $A$  具有满列秩 (full column rank)。
- (7) 任何矩阵  $A$  左乘满列秩矩阵或者右乘满行秩矩阵后, 矩阵  $A$  的秩保持不变。

### 2. 关于秩的等式

- (1) 若  $A \in \mathbb{C}^{m \times n}$ , 则  $\text{rank}(A^H) = \text{rank}(A^T) = \text{rank}(A^*) = \text{rank}(A)$ 。
- (2) 若  $A \in \mathbb{C}^{m \times n}$  和  $c \neq 0$ , 则  $\text{rank}(cA) = \text{rank}(A)$ 。
- (3) 若  $A \in \mathbb{C}^{m \times m}$  和  $C \in \mathbb{C}^{n \times n}$  非奇异, 则  $\text{rank}(AB) = \text{rank}(B) = \text{rank}(BC) = \text{rank}(ABC)$ , 即矩阵  $B$  左乘与 (或) 右乘一个非奇异矩阵后, 其秩保持不变。
- (4) 如果  $A, B \in \mathbb{C}^{m \times n}$ , 则  $\text{rank}(A) = \text{rank}(B)$ , 当且仅当存在非奇异矩阵  $X \in \mathbb{C}^{m \times m}$  和  $Y \in \mathbb{C}^{n \times n}$  使得  $B = XAY$ 。
- (5)  $\text{rank}(AA^T) = \text{rank}(A^TA) = \text{rank}(A)$  和  $\text{rank}(AA^H) = \text{rank}(A^HA) = \text{rank}(A)$ 。
- (6) 若  $A \in \mathbb{C}^{m \times m}$ , 则  $\text{rank}(A) = m \Leftrightarrow \det(A) \neq 0 \Leftrightarrow A$  非奇异。

### 3. 关于秩的不等式

- (1) 对于任意  $m \times n$  矩阵  $A$  均有  $\text{rank}(A) \leq \min\{m, n\}$ 。

- (2) 若  $A, B \in \mathbb{C}^{m \times n}$ , 则  $\text{rank}(A + B) \leq \text{rank}(A) + \text{rank}(B)$ 。  
(3) 若  $A \in \mathbb{C}^{m \times k}$  和  $B \in \mathbb{C}^{k \times n}$ , 则

$$\text{rank}(A) + \text{rank}(B) - k \leq \text{rank}(AB) \leq \min\{\text{rank}(A), \text{rank}(B)\}$$

特别地, 对于幂等矩阵  $A^2 = A$ , 有<sup>[403]</sup>  $\text{rank}(A) = \text{tr}(A)$ 。

一句话小结: 作为一个性能指标, 矩阵的秩刻画矩阵行与行之间或者列与列之间的线性无关性, 从而反映矩阵的满秩性和秩亏缺性。

以上分别介绍了矩阵的几种重要性能指标: 二次型、行列式、特征值、迹和秩。表 1.6.1 总结了矩阵的这些标量性能指标以及它们所描述的矩阵性能。

表 1.6.1 矩阵的性能指标

性能指标	描述的矩阵性能
二次型	矩阵的正定性与负定性
行列式	矩阵的奇异性
特征值	矩阵的奇异性、正定性和对角元素的结构
迹	矩阵对角元素之和、特征值之和
秩	行(或列)之间的线性无关性; 矩阵方程的适定性

## 1.7 逆矩阵与伪逆矩阵

矩阵求逆是一种经常遇到的重要运算。特别地, 矩阵求逆引理在信号处理、系统科学、神经网络、自动控制等学科中经常用到。本节介绍正方满秩矩阵的逆矩阵和非正方满(行或列)秩矩阵的伪逆矩阵。至于一个非正方的秩亏缺矩阵的逆矩阵, 将在 1.8 节专题讨论。

### 1.7.1 逆矩阵的定义与性质

一个  $n \times n$  矩阵称为非奇异矩阵, 若它具有  $n$  个线性无关的列向量和  $n$  个线性无关的行向量。非奇异矩阵也可以从线性系统的观点出发定义: 一线性变换或正方矩阵  $A$  称为非奇异的, 若它只对零输入产生零输出。否则, 它是奇异的。如果一个矩阵非奇异, 那么它必定存在逆矩阵。反之, 一奇异矩阵肯定不存在逆矩阵。一个  $n \times n$  的正方矩阵  $B$  满足  $BA = AB = I$  时, 就称矩阵  $B$  是矩阵  $A$  的逆矩阵, 记为  $A^{-1}$ 。

若矩阵  $A \in \mathbb{C}^{n \times n}$  的逆矩阵存在, 则称矩阵  $A$  是非奇异的或可逆的。关于矩阵的非奇异性或可逆性, 下列叙述等价<sup>[238]</sup>:

- (1)  $A$  非奇异;

- (2)  $A^{-1}$  存在;
- (3)  $\text{rank}(A) = n$ ;
- (4)  $A$  的行线性无关;
- (5)  $A$  的列线性无关;
- (6)  $\det(A) \neq 0$ ;
- (7)  $A$  的值域的维数是  $n$ ;
- (8)  $A$  的零空间的维数是 0;
- (9)  $Ax = b$  对每一个  $b \in \mathbb{C}^n$  都是一致方程;
- (10)  $Ax = b$  对每一个  $b$  有唯一的解;
- (11)  $Ax = 0$  只有平凡解  $x = 0$ 。

$n \times n$  矩阵  $A$  的逆矩阵  $A^{-1}$  具有以下性质 [32, 238]:

- (1)  $A^{-1}A = AA^{-1} = I$ 。
- (2)  $A^{-1}$  是唯一的。
- (3) 逆矩阵的行列式等于原矩阵行列式的倒数, 即  $|A^{-1}| = \frac{1}{|A|}$ 。
- (4) 逆矩阵是非奇异的。
- (5)  $(A^{-1})^{-1} = A$ 。
- (6) 复共轭转置矩阵的逆矩阵  $(A^H)^{-1} = (A^{-1})^H = A^{-H}$ 。
- (7) 若  $A^H = A$ , 则  $(A^{-1})^H = A^{-1}$ 。
- (8)  $(A^*)^{-1} = (A^{-1})^*$ 。
- (9) 若  $A$  和  $B$  均可逆, 则  $(AB)^{-1} = B^{-1}A^{-1}$ 。
- (10) 若  $A = \text{diag}(a_1, \dots, a_m)$  为对角矩阵, 则其逆矩阵

$$A^{-1} = \text{diag}(a_1^{-1}, \dots, a_m^{-1})$$

- (11) 若  $A$  非奇异, 则有  $A$  为正交矩阵  $\Leftrightarrow A^{-1} = A^T$  和  $A$  为酉矩阵  $\Leftrightarrow A^{-1} = A^H$ 。

下面证明性质 (1),(2),(9), 其他性质的证明留给读者作为练习。

**证明** 性质 (1) 的证明: 假定  $A^{-1}A = I$ , 并且存在另外一个矩阵  $P$  满足  $AP = I$ 。

于是, 左乘逆矩阵  $A^{-1}$  后, 得  $A^{-1}AP = A^{-1}$ 。由于  $A^{-1}A = I$ , 故有  $P = A^{-1}$ 。因此有  $A^{-1}A = AA^{-1} = I$ 。

性质 (2) 的证明: 令  $P$  是矩阵  $A$  的另一个逆矩阵。在 (1) 的证明中, 已经证明满足  $AP = I$  的矩阵  $P = A^{-1}$ 。下面证明满足  $PA = I$  的矩阵为  $P = A^{-1}$ 。在  $PA = I$  两边右乘  $A^{-1}$ , 得  $PA A^{-1} = A^{-1}$ 。由于  $AA^{-1} = I$ , 故立即有  $P = A^{-1}$ 。因此, 同时满足  $PA = AP = I$  的矩阵  $P = A^{-1}$ , 即  $A$  的逆矩阵  $A^{-1}$  是唯一的。

性质 (9) 的证明: 假定矩阵  $A$  和  $B$  是两个可逆的正方矩阵。易知

$$B^{-1}A^{-1}AB = B^{-1}B = I \quad \text{和} \quad ABB^{-1}A^{-1} = AA^{-1} = I$$

因此,  $B^{-1}A^{-1}$  是矩阵  $AB$  的逆矩阵, 即  $(AB)^{-1} = B^{-1}A^{-1}$ 。 ■

### 1.7.2 矩阵求逆引理

**引理 1.7.1 (Sherman-Morrison 公式)** 令  $A$  是一个  $n \times n$  的可逆矩阵，并且  $x$  和  $y$  是两个  $n \times 1$  向量，使得  $(A + xy^H)$  可逆，则

$$(A + xy^H)^{-1} = A^{-1} - \frac{A^{-1}xy^H A^{-1}}{1 + y^H A^{-1}x} \quad (1.7.1)$$

证明 由于

$$A + xy^H = A(I + A^{-1}xy^H)$$

故有

$$(A + xy^H)^{-1} = (I + A^{-1}xy^H)^{-1}A^{-1} \quad (1)$$

若  $(I + B)$  可逆，并且  $B \neq I$ ，则  $(I + B)^{-1} = I - B + B^2 - B^3 + \dots$ 。将这一公式代入式 (1) 中的  $(I + A^{-1}xy^H)^{-1}$ ，立即有

$$\begin{aligned} (I + A^{-1}xy^H)^{-1} &= I - A^{-1}xy^H + (A^{-1}xy^H)^2 - (A^{-1}xy^H)^3 + \dots \\ &= I - A^{-1}xy^H + A^{-1}xy^H A^{-1}xy^H - \dots \end{aligned} \quad (2)$$

将式 (2) 代入式 (1)，易知

$$\begin{aligned} (A + xy^H)^{-1} &= A^{-1} - A^{-1}xy^H A^{-1} + A^{-1}x(y^H A^{-1}x)y^H A^{-1} - \dots \\ &= A^{-1} - A^{-1}xy^H A^{-1}[1 - (y^H A^{-1}x) + (y^H A^{-1}x)^2 - \dots] \end{aligned} \quad (3)$$

由矩阵  $(I + A^{-1}xy^H)$  的可逆性知，标量  $y^H A^{-1}x \neq -1$ ，从而有

$$1 - (y^H A^{-1}x) + (y^H A^{-1}x)^2 - \dots = \frac{1}{1 + y^H A^{-1}x}$$

将上式代入到式 (3) 的中括号项，立即得式 (1.7.1)。 ■

引理 1.7.1 称为矩阵求逆引理，是 Sherman 与 Morrison<sup>[450, 451]</sup> 于 1949 年和 1950 年得到的。

矩阵求逆引理可以推广为矩阵之和的求逆公式

$$\begin{aligned} (A + UBV)^{-1} &= A^{-1} - A^{-1}UB(B + BVA^{-1}UB)^{-1}BVA^{-1} \\ &= A^{-1} - A^{-1}U(I + BVA^{-1}U)^{-1}BVA^{-1} \end{aligned} \quad (1.7.2)$$

或者

$$(A - UV)^{-1} = A^{-1} + A^{-1}U(I - VA^{-1}U)^{-1}VA^{-1} \quad (1.7.3)$$

这一公式是 Woodbury 于 1950 年得到的<sup>[517]</sup>，也称 Woodbury 公式。矩阵  $I - VA^{-1}U$  有时称为容量矩阵 (capacitance matrix)。

当  $\mathbf{U} = \mathbf{u}$ ,  $\mathbf{B} = b$  和  $\mathbf{V} = \mathbf{v}^H$  时, Woodbury 公式给出结果

$$(\mathbf{A} + b\mathbf{u}\mathbf{v}^H)^{-1} = \mathbf{A}^{-1} - \frac{b}{1 + b\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}}\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1} \quad (1.7.4)$$

特别地, 若  $b = 1$ , 则式 (1.7.4) 简化为 Sherman 与 Morrison 的矩阵求逆引理公式 (1.7.1)。

事实上, 在 Woodbury 得到求逆公式 (1.7.2) 之前, Duncan<sup>[150]</sup> 和 Guttman<sup>[211]</sup> 就已经分别于 1944 年和 1946 年得到了下面的求逆公式

$$(\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1} = \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{U}(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} \quad (1.7.5)$$

这一公式也被称为 Duncan-Guttman 求逆公式<sup>[406, 407]</sup>。

除了 Woodbury 公式之外, 矩阵之和的逆矩阵还有下面的形式<sup>[225]</sup>

$$(\mathbf{A} + \mathbf{UBV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}(\mathbf{I} + \mathbf{UBV}\mathbf{A}^{-1})^{-1}\mathbf{UBV}\mathbf{A}^{-1} \quad (1.7.6)$$

$$= \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{UB}(\mathbf{I} + \mathbf{VA}^{-1}\mathbf{UB})^{-1}\mathbf{VA}^{-1} \quad (1.7.7)$$

$$= \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{UBV}(\mathbf{I} + \mathbf{A}^{-1}\mathbf{UBV})^{-1}\mathbf{A}^{-1} \quad (1.7.8)$$

$$= \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{UBV}\mathbf{A}^{-1}(\mathbf{I} + \mathbf{UBV}\mathbf{A}^{-1})^{-1} \quad (1.7.9)$$

下面是分块矩阵的几种求逆公式。

(1) 矩阵  $\mathbf{A}$  可逆时<sup>[28]</sup>

$$\begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{V} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{U}(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} & -\mathbf{A}^{-1}\mathbf{U}(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \\ -(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} & (\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \end{bmatrix} \quad (1.7.10)$$

(2) 矩阵  $\mathbf{A}$  和  $\mathbf{D}$  可逆时<sup>[241, 242]</sup>

$$\begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{V} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1} & -\mathbf{A}^{-1}\mathbf{U}(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \\ -\mathbf{D}^{-1}\mathbf{V}(\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1} & (\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \end{bmatrix} \quad (1.7.11)$$

(3) 矩阵  $\mathbf{A}$  和  $\mathbf{D}$  可逆时<sup>[150]</sup>

$$\begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{V} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1} & -(\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1}\mathbf{UD}^{-1} \\ -(\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} & (\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \end{bmatrix} \quad (1.7.12)$$

或者<sup>[14, p.138]</sup>

$$\begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{V} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A} - \mathbf{UD}^{-1}\mathbf{V})^{-1} & -(\mathbf{V} - \mathbf{DU}^{-1}\mathbf{A})^{-1} \\ ((\mathbf{U} - \mathbf{AV}^{-1}\mathbf{D})^{-1} & (\mathbf{D} - \mathbf{VA}^{-1}\mathbf{U})^{-1} \end{bmatrix} \quad (1.7.13)$$

利用逆矩阵的定义, 不难分别验证增广矩阵求逆和分块矩阵求逆的正确性, 留给读者作练习。矩阵求逆在信号处理、神经网络、自动控制和系统理论等中具有广泛的应用。

下面介绍 Woodbury 公式的两个典型应用。

令  $\mathbf{J}_n$  是一个  $n \times n$  矩阵, 其元素全部为 1, 则由于  $n \times n$  矩阵 (其中,  $a \neq b$ )

$$\mathbf{V} = \begin{bmatrix} a & b & \cdots & b \\ b & a & \cdots & b \\ \vdots & \vdots & \ddots & \vdots \\ b & b & \cdots & a \end{bmatrix} = [(a - b)\mathbf{I}_n + b\mathbf{J}_n] = (a - b) \left( \mathbf{I}_n + \frac{b}{a - b}\mathbf{J}_n \right) \quad (1.7.14)$$

故由  $\mathbf{J}_n = \mathbf{1}\mathbf{1}^T$  (其中  $\mathbf{1}$  是全部元素为 1 的向量), 可求得逆矩阵

$$\mathbf{V}^{-1} = \frac{1}{a-b} \left( \mathbf{I}_n + \frac{b}{a-b} \mathbf{J}_n \right)^{-1} = \frac{1}{a-b} \left[ \mathbf{I}_n - \frac{b}{a+(n-1)b} \mathbf{J}_n \right] \quad (1.7.15)$$

假定  $\mathbf{A}, \mathbf{U}, \mathbf{V}$  均为  $n \times n$  矩阵, 则利用式 (1.7.3), 可以得到求解矩阵方程  $(\mathbf{A}-\mathbf{U}\mathbf{V})\mathbf{x} = \mathbf{b}$  的方法如下<sup>[212]</sup>:

- (1) 求解矩阵方程  $\mathbf{A}\mathbf{y} = \mathbf{b}$  得到  $\mathbf{y}$ 。
- (2) 通过求解矩阵方程  $\mathbf{A}\mathbf{w}_i = \mathbf{u}_i$  得到  $\mathbf{w}_i$ , 其中  $\mathbf{u}_i$  是矩阵  $\mathbf{U}$  的第  $i$  列; 然后构造矩阵  $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_n]$ , 此即  $\mathbf{W} = \mathbf{A}^{-1}\mathbf{U}$  的结果。
- (3) 构造矩阵  $\mathbf{C} = \mathbf{I} - \mathbf{V}\mathbf{W}$  和向量  $\mathbf{V}\mathbf{y}$ , 并求解线性方程  $\mathbf{C}\mathbf{z} = \mathbf{V}\mathbf{y}$ , 得到  $\mathbf{z}$ 。
- (4) 矩阵方程  $(\mathbf{A}-\mathbf{U}\mathbf{V})\mathbf{x} = \mathbf{b}$  的解由  $\mathbf{x} = \mathbf{y} + \mathbf{W}\mathbf{z}$  给出。

顺便指出, 上述方法的所有四个步骤都只需要矩阵的初等变换和基本运算, 并不需要直接计算逆矩阵。

最后介绍 Hermitian 矩阵的求逆引理。令 Hermitian 矩阵的分块形式为

$$\mathbf{R}_{m+1} = \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m \\ \mathbf{r}_m^H & \rho_m \end{bmatrix} \quad (1.7.16)$$

下面考虑使用  $\mathbf{R}_m^{-1}$  递推  $\mathbf{R}_{m+1}^{-1}$ 。为此, 令

$$\mathbf{Q}_{m+1} = \begin{bmatrix} \mathbf{Q}_m & \mathbf{q}_m \\ \mathbf{q}_m^H & \alpha_m \end{bmatrix} \quad (1.7.17)$$

于是

$$\mathbf{R}_{m+1}\mathbf{Q}_{m+1} = \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m \\ \mathbf{r}_m^H & \rho_m \end{bmatrix} \begin{bmatrix} \mathbf{Q}_m & \mathbf{q}_m \\ \mathbf{q}_m^H & \alpha_m \end{bmatrix} = \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_m \\ \mathbf{0}_m^H & 1 \end{bmatrix} \quad (1.7.18)$$

由此可以导出下面四个方程式

$$\mathbf{R}_m\mathbf{Q}_m + \mathbf{r}_m\mathbf{q}_m^H = \mathbf{I}_m \quad (1.7.19)$$

$$\mathbf{r}_m^H\mathbf{Q}_m + \rho_m\mathbf{q}_m^H = \mathbf{0}_m^H \quad (1.7.20)$$

$$\mathbf{R}_m\mathbf{q}_m + \mathbf{r}_m\alpha_m = \mathbf{0}_m \quad (1.7.21)$$

$$\mathbf{r}_m^H\mathbf{q}_m + \rho_m\alpha_m = 1 \quad (1.7.22)$$

若  $\mathbf{R}_m$  可逆, 则由式 (1.7.21) 有

$$\mathbf{q}_m = -\alpha_m \mathbf{R}_m^{-1} \mathbf{r}_m \quad (1.7.23)$$

此结果代入式 (1.7.22) 后, 即有

$$\alpha_m = \frac{1}{\rho_m - \mathbf{r}_m^H \mathbf{R}_m^{-1} \mathbf{r}_m} \quad (1.7.24)$$

将式 (1.7.24) 代入式 (1.7.23), 又可以求得

$$\mathbf{q}_m = \frac{-\mathbf{R}_m^{-1} \mathbf{r}_m}{\rho_m - \mathbf{r}_m^H \mathbf{R}_m^{-1} \mathbf{r}_m} \quad (1.7.25)$$

若将式 (1.7.25) 代入式 (1.7.19), 则

$$\mathbf{Q}_m = \mathbf{R}_m^{-1} - \mathbf{R}_m^{-1} \mathbf{r}_m \mathbf{q}_m^H = \mathbf{R}_m^{-1} + \frac{\mathbf{R}_m^{-1} \mathbf{r}_m (\mathbf{R}_m^{-1} \mathbf{r}_m)^H}{\rho_m - \mathbf{r}_m^H \mathbf{R}_m^{-1} \mathbf{r}_m} \quad (1.7.26)$$

为了简化式 (1.7.24) ~ 式 (1.7.26), 不妨令

$$\mathbf{b}_m \stackrel{\text{def}}{=} [b_0^{(m)}, b_1^{(m)}, \dots, b_{m-1}^{(m)}]^T = -\mathbf{R}_m^{-1} \mathbf{r}_m \quad (1.7.27)$$

$$\beta_m \stackrel{\text{def}}{=} \rho_m - \mathbf{r}_m^H \mathbf{R}_m^{-1} \mathbf{r}_m = \rho_m + \mathbf{r}_m^H \mathbf{b}_m \quad (1.7.28)$$

这样一来, 式 (1.7.24) ~ 式 (1.7.26) 即可依次简化为

$$\begin{aligned}\alpha_m &= \frac{1}{\beta_m} \\ \mathbf{q}_m &= \frac{1}{\beta_m} \mathbf{b}_m \\ \mathbf{Q}_m &= \mathbf{R}_m^{-1} + \frac{1}{\beta_m} \mathbf{b}_m \mathbf{b}_m^H\end{aligned}$$

将它们代入式 (1.7.18), 即得

$$\mathbf{R}_{m+1}^{-1} = \mathbf{Q}_{m+1} = \begin{bmatrix} \mathbf{R}_m^{-1} & \mathbf{0}_m \\ \mathbf{0}_m^H & 0 \end{bmatrix} + \frac{1}{\beta_m} \begin{bmatrix} \mathbf{b}_m \mathbf{b}_m^H & \mathbf{b}_m \\ \mathbf{b}_m^H & 1 \end{bmatrix} \quad (1.7.29)$$

这一由  $\mathbf{R}_m^{-1}$  求  $\mathbf{R}_{m+1}^{-1}$  的秩 1 修正公式称为 Hermitian 矩阵的分块求逆引理<sup>[371]</sup>。

### 1.7.3 左逆矩阵与右逆矩阵

从广义的角度讲, 任何一个矩阵  $\mathbf{G}$  都可以称为矩阵  $\mathbf{A}$  的逆矩阵, 若它与矩阵  $\mathbf{A}$  的乘积等于单位矩阵  $\mathbf{I}$ , 即  $\mathbf{GA} = \mathbf{I}$ 。根据矩阵  $\mathbf{A}$  本身的特点, 满足这一定义的矩阵  $\mathbf{G}$  存在以下三种可能的答案:

- (1) 在某些情况下,  $\mathbf{G}$  存在, 并且唯一;
- (2) 在另一些情况下,  $\mathbf{G}$  存在, 但不唯一;
- (3) 在有些情况下,  $\mathbf{G}$  不存在。

**例 1.7.1** 考虑以下三个矩阵

$$\mathbf{A}_1 = \begin{bmatrix} 2 & -2 & -1 \\ 1 & 1 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 4 & 8 \\ 5 & -7 \\ -2 & 3 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} 1 & 3 & 1 \\ 2 & 5 & 1 \end{bmatrix}$$

对矩阵  $\mathbf{A}_1$ , 存在唯一矩阵

$$\mathbf{G} = \begin{bmatrix} -1 & -2 & 5 \\ -1 & -1 & 3 \\ -1 & -2 & 4 \end{bmatrix}$$

不仅使得  $\mathbf{GA}_1 = \mathbf{I}_{3 \times 3}$ , 而且使得  $\mathbf{A}_1 \mathbf{G} = \mathbf{I}_{3 \times 3}$ 。此时, 矩阵  $\mathbf{G}$  实际就是矩阵  $\mathbf{A}_1$  的逆矩阵, 即  $\mathbf{G} = \mathbf{A}_1^{-1}$ 。

在矩阵  $A_2$  的情况下, 存在多个  $2 \times 3$  矩阵  $L$  使得  $LA_2 = I_{2 \times 2}$ , 如

$$L = \begin{bmatrix} \frac{7}{68} & \frac{2}{17} & 0 \\ 0 & 2 & 5 \end{bmatrix}, \quad L = \begin{bmatrix} 0 & 3 & 7 \\ 0 & 2 & 5 \end{bmatrix}, \dots$$

对矩阵  $A_3$ , 没有任何  $3 \times 2$  矩阵使得  $G_3 A_3 = I_{3 \times 3}$ , 但存在多个  $3 \times 2$  矩阵  $R$ , 使得  $A_3 R = I_{2 \times 2}$ , 例如

$$R = \begin{bmatrix} 1 & 1 \\ -1 & 0 \\ 3 & -1 \end{bmatrix}, \quad R = \begin{bmatrix} -1 & 1 \\ 0 & 0 \\ 2 & -1 \end{bmatrix}, \dots$$

总结以上讨论知, 除了满足  $AA^{-1} = A^{-1}A = I$  的逆矩阵  $A^{-1}$  外, 还存在两种其他形式的逆矩阵, 它们只满足  $LA = I$  或  $AR = I$ 。

**定义 1.7.1** [444] 满足  $LA = I$ , 但不满足  $AL = I$  的矩阵  $L$  称为矩阵  $A$  的左逆矩阵 (left inverse)。类似地, 满足  $AR = I$ , 但不满足  $RA = I$  的矩阵称为矩阵  $A$  的右逆矩阵 (right inverse)。

- (1) 仅当  $m \geq n$  时, 矩阵  $A \in \mathbb{C}^{m \times n}$  可能有左逆矩阵。
- (2) 仅当  $m \leq n$  时, 矩阵  $A \in \mathbb{C}^{m \times n}$  可能有右逆矩阵。

如例 1.7.1 所示, 对于给定的  $m \times n$  矩阵  $A$ , 当  $m > n$  时, 可能存在多个  $n \times m$  矩阵  $L$  使得  $LA = I_n$ ; 而当  $m < n$  时, 则可能有多个  $n \times m$  矩阵  $R$  满足  $AR = I_m$ , 即一个矩阵  $A$  的左逆矩阵或者右逆矩阵往往非唯一。下面考虑左和右逆矩阵的唯一解。

考察  $m > n$  并且  $A$  具有满列秩 ( $\text{rank } A = n$ ) 的情况。此时,  $n \times n$  矩阵  $A^H A$  是可逆的。容易验证

$$L = (A^H A)^{-1} A^H \tag{1.7.30}$$

满足左逆矩阵的定义  $LA = I$ 。这种左逆矩阵是唯一确定的, 常称为左伪逆矩阵 (left pseudo inverse)。

再考察  $m < n$  并且  $A$  具有满行秩 ( $\text{rank } A = m$ ) 的情况。此时,  $m \times m$  矩阵  $A A^H$  是可逆的。定义

$$R = A^H (A A^H)^{-1} \tag{1.7.31}$$

不难验证, 它满足右逆矩阵的定义  $AR = I$ 。这一特殊的右逆矩阵也是唯一确定的, 常称为右伪逆矩阵 (right pseudo inverse)。

左伪逆矩阵与超定方程的最小二乘解密切相关, 而右伪逆矩阵则与欠定方程的最小二乘最小范数解密切联系在一起。

下面是左伪逆矩阵与右伪逆矩阵的阶数递推 [544]。

考虑  $n \times m$  矩阵  $F_m$  (其中  $n > m$ ), 并设  $F_m^\dagger = (F_m^H F_m)^{-1} F_m^H$  是  $F_m$  的左伪逆矩阵。令  $F_m = [F_{m-1}, f_m]$ , 其中  $f_m$  是矩阵  $F_m$  的第  $m$  列, 且  $\text{rank}(F_m) = m$ , 则计算  $F_m^\dagger$  的递推公式由

$$F_m^\dagger = \begin{bmatrix} F_{m-1}^\dagger - F_{m-1}^\dagger f_m e_m^H \Delta_m^{-1} \\ e_m^H \Delta_m^{-1} \end{bmatrix} \tag{1.7.32}$$

给出, 式中  $e_m = [I_n - F_{m-1}F_{m-1}^\dagger]f_m$  及  $\Delta_m^{-1} = [f_m^H e_m]^{-1}$ ; 且初始值为  $F_1^\dagger = f_1^H / (f_1^H f_1)$ 。

对于矩阵  $F_m \in \mathbb{C}^{n \times m}$ , 其中  $n < m$ , 若记  $F_m = [F_{m-1}, f_m]$ , 则右伪逆矩阵  $F_m^\dagger = F_m^H (F_m F_m^H)^{-1}$  具有以下递推公式

$$F_m^\dagger = \begin{bmatrix} F_{m-1}^\dagger - \Delta_m F_{m-1}^\dagger f_m c_m \\ \Delta_m c_m^H \end{bmatrix} \quad (1.7.33)$$

式中,  $c_m^H = f_m^H (I_n - F_{m-1}F_{m-1}^\dagger)$ ,  $\Delta_m = c_m^H f_m$ 。递推的初始值为  $F_1^\dagger = f_1^H / (f_1^H f_1)$ 。

## 1.8 Moore-Penrose 逆矩阵

在 1.7 节, 我们分别讨论了非奇异的正方矩阵  $A$  的逆矩阵  $A^{-1}$ , 以及  $m \times n (m \neq n)$  长方形满列秩矩阵的左伪逆矩阵  $(A^H A)^{-1} A^H$  和满行秩矩阵的右伪逆矩阵  $A^H (A A^H)^{-1}$ 。那么, 一个秩亏缺的矩阵是否存在逆矩阵? 这个逆矩阵又应该满足什么样的条件?

### 1.8.1 Moore-Penrose 逆矩阵的定义与性质

考虑一个  $m \times n$  维的秩亏缺矩阵  $A$ , 其中  $m$  和  $n$  之间的大小不论, 但秩  $\text{rank}(A) = k < \min\{m, n\}$ 。 $m \times n$  秩亏缺矩阵的逆矩阵称为广义逆矩阵, 它是一个  $n \times m$  矩阵。令  $A^-$  表示  $A$  的广义逆矩阵。

由矩阵秩的性质  $\text{rank}(AB) \leq \min\{\text{rank}(A), \text{rank}(B)\}$  知, 无论  $AA^- = I_{m \times m}$  还是  $A^-A = I_{n \times n}$  都不可能成立, 因为  $m \times m$  矩阵  $AA^-$  和  $n \times n$  矩阵  $A^-A$  都是秩亏缺矩阵, 它们的秩的最大值为  $\text{rank}(A) = k$ , 小于  $\min\{m, n\}$ 。

既然两个矩阵的乘积  $AA^- \neq I_{m \times m}$  和  $A^-A \neq I_{n \times n}$ , 那么就有必要使用三个矩阵的乘积定义一个秩亏缺矩阵  $A$  的逆矩阵。

考虑线性矩阵方程  $Ax = y$  的求解。矩阵方程两边左乘  $AA^-$ , 则有  $AA^-Ax = AA^-y$ 。若  $A^-$  是矩阵  $A$  的广义逆矩阵, 则  $Ax = y \Rightarrow x = A^-y$ 。将  $x = A^-y$  代入  $AA^-Ax = AA^-y$ , 立即得  $AA^-Ax = Ax$ 。由于此式对任意非零向量  $x$  均应该成立, 因此要求下列约束条件必须满足

$$AA^-A = A \quad (1.8.1)$$

满足条件  $AA^-A = A$  的矩阵  $A^-$  称为  $A$  的广义逆矩阵, 其维数为  $n \times m$ 。遗憾的是, 这样定义的广义逆矩阵不是唯一的, 存在明显的缺陷。

定义  $AA^-A = A$  只能保证  $A^-$  是矩阵  $A$  的广义逆矩阵, 但并不能反过来保证  $A$  也是  $A^-$  的广义逆矩阵, 而矩阵  $A$  和  $A^-$  本来应该是互为逆矩阵。这就是定义公式  $AA^-A = A$  非唯一确定和存在明显缺陷的主要原因之一。

为了保证广义逆矩阵的唯一定义, 至少必须增加  $A$  也是  $A^-$  的广义逆矩阵的约束条件。不妨用符号  $A^\dagger$  表示矩阵  $A$  可能存在的唯一定义的广义逆矩阵。

考虑原矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的解方程  $\mathbf{x} = \mathbf{A}^\dagger \mathbf{y}$  的求解：已知广义逆矩阵  $\mathbf{A}^\dagger$  和向量  $\mathbf{x}$ ，求  $\mathbf{y}$ 。解方程  $\mathbf{x} = \mathbf{A}^\dagger \mathbf{y}$  两边左乘  $\mathbf{A}^\dagger \mathbf{A}$ ，得  $\mathbf{A}^\dagger \mathbf{A}\mathbf{x} = \mathbf{A}^\dagger \mathbf{A}\mathbf{A}^\dagger \mathbf{y}$ 。由于矩阵  $\mathbf{A}$  是  $\mathbf{A}^\dagger$  的广义逆矩阵，故  $\mathbf{x} = \mathbf{A}^\dagger \mathbf{y} \Rightarrow \mathbf{A}\mathbf{x} = \mathbf{y}$ 。将  $\mathbf{A}\mathbf{x} = \mathbf{y}$  代入  $\mathbf{A}^\dagger \mathbf{A}\mathbf{x} = \mathbf{A}^\dagger \mathbf{A}\mathbf{A}^\dagger \mathbf{y}$  中，立即知  $\mathbf{A}^\dagger \mathbf{y} = \mathbf{A}^\dagger \mathbf{A}\mathbf{A}^\dagger \mathbf{y}$  对任意非零向量  $\mathbf{y}$  都应该成立，故

$$\mathbf{A}^\dagger \mathbf{A}\mathbf{A}^\dagger = \mathbf{A}^\dagger \quad (1.8.2)$$

也必须满足。

可见，若  $\mathbf{A}^\dagger$  是秩亏缺矩阵  $\mathbf{A}$  的广义逆矩阵，则式 (1.8.1) 和式 (1.8.2) 所示的两个条件必须同时满足。然而，仅有  $m \times n$  矩阵  $\mathbf{A}$  与  $n \times m$  广义逆矩阵  $\mathbf{A}^\dagger$  之间的三矩阵乘积之间的约束仍然是不够的，还必须考虑对这两个矩阵的乘积作出相应的约束。

如果  $m \times n$  矩阵  $\mathbf{A}$  是满列秩或者满行秩时，我们当然希望广义逆矩阵  $\mathbf{A}^\dagger$  能够包括左和右伪逆矩阵作为特例在内。虽然  $m \times n$  满列秩矩阵  $\mathbf{A}$  的左伪逆矩阵  $\mathbf{L} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  满足  $\mathbf{L}\mathbf{A} = \mathbf{I}_{n \times n}$ ，不存在  $\mathbf{AL} = \mathbf{I}_{m \times m}$ ，但是  $\mathbf{AL} = \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H = (\mathbf{AL})^H$  是一个复共轭对称矩阵。无独有偶，右伪逆矩阵  $\mathbf{R} = \mathbf{A}^H (\mathbf{AA}^H)^{-1}$  虽然只满足  $\mathbf{RA} = \mathbf{I}_{m \times m}$ ，但乘积矩阵  $\mathbf{RA} = \mathbf{A}^H (\mathbf{AA}^H)^{-1} \mathbf{A} = (\mathbf{RA})^H$  也是一个复共轭对称矩阵。因此， $m \times n$  秩亏缺矩阵  $\mathbf{A}$  的  $n \times m$  广义逆矩阵  $\mathbf{A}^\dagger$  之间的乘积  $\mathbf{AA}^\dagger$  和  $\mathbf{A}^\dagger \mathbf{A}$  虽然不可能分别等于单位矩阵  $\mathbf{I}_{m \times m}$  和  $\mathbf{I}_{n \times n}$ ，但应该满足下述两个复共轭对称条件

$$\mathbf{AA}^\dagger = (\mathbf{AA}^\dagger)^H, \quad \mathbf{A}^\dagger \mathbf{A} = (\mathbf{A}^\dagger \mathbf{A})^H \quad (1.8.3)$$

综合式 (1.8.1) ~ 式 (1.8.3) 所示四个条件，可以引出下面的定义。

**定义 1.8.1**<sup>[402]</sup> 令  $\mathbf{A}$  是任意  $m \times n$  矩阵，称矩阵  $\mathbf{A}^\dagger$  是  $\mathbf{A}$  的广义逆矩阵，若  $\mathbf{A}^\dagger$  满足以下四个条件（常称 Moore-Penrose 条件）：

- (1)  $\mathbf{AA}^\dagger \mathbf{A} = \mathbf{A}$ ;
- (2)  $\mathbf{A}^\dagger \mathbf{AA}^\dagger = \mathbf{A}^\dagger$ ;
- (3)  $\mathbf{AA}^\dagger$  为 Hermitian 矩阵，即  $\mathbf{AA}^\dagger = (\mathbf{AA}^\dagger)^H$ ;
- (4)  $\mathbf{A}^\dagger \mathbf{A}$  为 Hermitian 矩阵，即  $\mathbf{A}^\dagger \mathbf{A} = (\mathbf{A}^\dagger \mathbf{A})^H$ 。

**注释 1** Moore<sup>[351]</sup> 于 1935 年从投影角度出发，证明了  $m \times n$  矩阵  $\mathbf{A}$  的广义逆矩阵  $\mathbf{A}^\dagger$  必须满足两个条件，但这两个条件不方便使用。20 年后，Penrose 于 1955 年提出了定义广义逆矩阵的以上四个条件<sup>[402]</sup>。1956 年，Rado<sup>[422]</sup> 证明了 Penrose 的四条件与 Moore 的两个条件等价。于是，人们后来便将广义逆矩阵需要满足的四个条件习惯称为 Moore-Penrose 条件，并将这种广义逆矩阵称为 Moore-Penrose 逆矩阵。

**注释 2** 特别地，Moore-Penrose 条件 (1) 是  $\mathbf{A}$  的广义逆矩阵  $\mathbf{A}^\dagger$  必须满足的条件；而条件 (2) 则是  $\mathbf{A}^\dagger$  的广义逆矩阵  $\mathbf{A}$  必须满足的条件。

根据满足的 Moore-Penrose 四个条件的多少，可以对广义逆矩阵进行分类<sup>[189]</sup>：

- ① 满足全部四个条件的矩阵  $\mathbf{A}^\dagger$  称为  $\mathbf{A}$  的 Moore-Penrose 逆矩阵。

② 只满足条件(1)和(2)的矩阵  $\mathbf{G} = \mathbf{A}^\dagger$  称为  $\mathbf{A}$  的自反广义逆矩阵。

③ 满足条件(1),(2)和(3)的矩阵  $\mathbf{A}^\dagger$  称为  $\mathbf{A}$  的正规化广义逆矩阵。

④ 满足条件(1),(2)和(4)的矩阵  $\mathbf{A}^\dagger$  称为  $\mathbf{A}$  的弱广义逆矩阵。

容易验证, 逆矩阵和上节介绍的各种广义逆矩阵都是 Moore-Penrose 逆矩阵的特例:

(1)  $n \times n$  正方非奇异矩阵  $\mathbf{A}_{n \times n}$  的逆矩阵  $\mathbf{A}^{-1}$  满足 Moore-Penrose 逆矩阵的所有四个条件。

(2)  $m \times n$  矩阵  $\mathbf{A}_{m \times n}$  ( $m > n$ ) 的左伪逆矩阵  $(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  满足 Moore-Penrose 逆矩阵的全部四个条件。

(3)  $m \times n$  矩阵  $\mathbf{A}_{m \times n}$  ( $m < n$ ) 的右伪逆矩阵  $\mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1}$  也满足 Moore-Penrose 逆矩阵的所有四个条件。

(4) 满足  $\mathbf{L} \mathbf{A}_{m \times n} = \mathbf{I}_n$  的一般左逆矩阵  $\mathbf{L}_{n \times m}$  是满足 Moore-Penrose 条件(1),(2), (4) 的弱广义逆矩阵。

(5) 满足  $\mathbf{A} \mathbf{R} = \mathbf{I}_m$  的一般右逆矩阵是满足 Moore-Penrose 条件(1),(2),(3) 的正规化广义逆矩阵。

与逆矩阵  $\mathbf{A}^{-1}$ 、左伪逆矩阵  $(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$  和右伪逆矩阵  $\mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1}$  均是唯一确定的一样, Moore-Penrose 逆矩阵也是唯一定义的。

任意一个  $m \times n$  矩阵  $\mathbf{A}$  的 Moore-Penrose 逆矩阵都可以由<sup>[52]</sup>

$$\mathbf{A}^\dagger = (\mathbf{A}^H \mathbf{A})^\dagger \mathbf{A}^H \quad (\text{若 } m \geq n) \quad (1.8.4)$$

或者<sup>[202]</sup>

$$\mathbf{A}^\dagger = \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^\dagger \quad (\text{若 } m \leq n) \quad (1.8.5)$$

确定。将以上公式代入定义 1.8.1, 可以验证  $\mathbf{A}^\dagger = (\mathbf{A}^H \mathbf{A})^\dagger \mathbf{A}^H$  和  $\mathbf{A}^\dagger = \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^\dagger$  分别满足 Moore-Penrose 逆矩阵的四个条件。

综合以上讨论以及文献 [324, 328, 417, 419, 424], 可以将 Moore-Penrose 逆矩阵  $\mathbf{A}^\dagger$  具有的性质汇总如下。

(1) Moore-Penrose 逆矩阵  $\mathbf{A}^\dagger$  是唯一的。

(2) 矩阵共轭转置的 Moore-Penrose 逆矩阵  $(\mathbf{A}^H)^\dagger = (\mathbf{A}^\dagger)^H = \mathbf{A}^{\dagger H} = \mathbf{A}^{H\dagger}$ 。

(3) Moore-Penrose 逆矩阵的广义逆矩阵等于原矩阵, 即  $(\mathbf{A}^\dagger)^\dagger = \mathbf{A}$ 。

(4) 若  $c \neq 0$ , 则  $(c\mathbf{A})^\dagger = \frac{1}{c}\mathbf{A}^\dagger$ 。

(5) 若  $\mathbf{D} = \text{diag}(d_{11}, \dots, d_{nn})$  为  $n \times n$  对角矩阵, 则  $\mathbf{D}^\dagger = \text{diag}(d_{11}^\dagger, \dots, d_{nn}^\dagger)$ , 其中,  $d_{ii}^\dagger = d_{ii}^{-1}$  (若  $d_{ii} \neq 0$ ) 或者  $d_{ii}^\dagger = 0$  (若  $d_{ii} = 0$ )。

(6) 零矩阵  $\mathbf{O}_{m \times n}$  的广义逆矩阵为  $n \times m$  零矩阵, 即有  $\mathbf{O}_{m \times n}^\dagger = \mathbf{O}_{n \times m}$ 。

(7) 向量  $\mathbf{x}$  的 Moore-Penrose 逆矩阵为  $\mathbf{x}^\dagger = (\mathbf{x}^H \mathbf{x})^{-1} \mathbf{x}^H$ 。

(8) 任意矩阵  $A_{m \times n}$  的 Moore-Penrose 逆矩阵都可以由  $A^\dagger = (A^H A)^\dagger A^H$  或  $A^\dagger = A^H (A A^H)^\dagger$  确定。特别地, 满秩矩阵的 Moore-Penrose 逆矩阵如下:

① 若  $A$  满列秩, 则  $A^\dagger = (A^H A)^{-1} A^H$ , 即满列秩矩阵  $A$  的 Moore-Penrose 逆矩阵退化为  $A$  的左伪逆矩阵。

② 若  $A$  满行秩, 则  $A^\dagger = A^H (A A^H)^{-1}$ , 即满行秩矩阵  $A$  的 Moore-Penrose 逆矩阵退化为  $A$  的右伪逆矩阵。

③ 若  $A$  为非奇异的正方矩阵, 则  $A^\dagger = A^{-1}$ , 即非奇异矩阵  $A$  的 Moore-Penrose 逆矩阵退化为  $A$  的逆矩阵。

(9) 对矩阵  $A_{m \times n}$ , 虽然  $A A^\dagger \neq I_m$ ,  $A^\dagger A \neq I_n$ ,  $A^H (A^H)^\dagger \neq I_n$  和  $(A^H)^\dagger A^H \neq I_m$ , 但下列结果为真:

$$\textcircled{1} \quad A^\dagger A A^H = A^H \text{ 和 } A^H A A^\dagger = A^H$$

$$\textcircled{2} \quad A A^\dagger (A^\dagger)^H = (A^\dagger)^H \text{ 和 } (A^H)^\dagger A^\dagger A = (A^\dagger)^H$$

$$\textcircled{3} \quad (A^H)^\dagger A A = A \text{ 和 } A A^H (A^H)^\dagger = A$$

$$\textcircled{4} \quad A^H (A^\dagger)^H A^\dagger = A^\dagger \text{ 和 } A^\dagger (A^\dagger)^H A^H = A^\dagger$$

(10) 若  $A = BC$ , 并且  $B$  满列秩,  $C$  满行秩, 则

$$A^\dagger = C^\dagger B^\dagger = C^H (C C^H)^{-1} (B^H B)^{-1} B^H$$

(11) 若  $A^H = A$ , 并且  $A^2 = A$ , 则  $A^\dagger = A$ 。

(12)  $(A A^H)^\dagger = (A^\dagger)^H A^\dagger$  和  $(A A^H)^\dagger (A A^H) = A A^\dagger$ 。

(13) 若矩阵  $A_i$  相互正交, 即  $A_i^H A_j = O$ ,  $i \neq j$ , 则  $(A_1 + \dots + A_m)^\dagger = A_1^\dagger + \dots + A_m^\dagger$ 。

(14) 关于广义逆矩阵的秩, 有  $\text{rank}(A^\dagger) = \text{rank}(A) = \text{rank}(A^H) = \text{rank}(A^\dagger A) = \text{rank}(A A^\dagger) = \text{rank}(A A^\dagger A) = \text{rank}(A^\dagger A A^\dagger)$ 。

### 1.8.2 Moore-Penrose 逆矩阵的计算

假定  $m \times n$  矩阵  $A$  的秩为  $r$ , 其中,  $r \leq \min(m, n)$ 。下面介绍求 Moore-Penrose 逆矩阵  $A^\dagger$  的四种方法。

#### 1. 方程求解法

Penrose<sup>[402]</sup> 在定义广义逆矩阵  $A^\dagger$  时, 提出了计算  $A^\dagger$  的两步法如下。

第一步: 求解矩阵方程  $A A^H X^H = A$  和  $A^H A Y = A^H$ , 分别得到  $X^H$  和  $Y$ 。

第二步: 计算广义逆矩阵  $A^\dagger = X A Y$ 。

以下是计算 Moore-Penrose 逆矩阵的两种方程求解法<sup>[202]</sup>。

#### 算法 1.8.1 方程求解法 1

步骤 1 计算矩阵  $B = A A^H$ 。

步骤 2 求解矩阵方程  $B^2 X^H = B$  得到矩阵  $X^H$ 。

步骤 3 计算  $B$  的 Moore-Penrose 逆矩阵  $B^\dagger = (AA^H)^\dagger = XBX^H$ 。

步骤 4 计算矩阵  $A$  的 Moore-Penrose 逆矩阵  $A^\dagger = A^H(AA^H)^\dagger = A^H B^\dagger$ 。

### 算法 1.8.2 方程求解法 2

步骤 1 计算矩阵  $B = A^H A$ 。

步骤 2 求解矩阵方程  $B^2 X^H = B$  得到矩阵  $X^H$ 。

步骤 3 计算  $B$  的 Moore-Penrose 逆矩阵  $B^\dagger = (A^H A)^\dagger = XBX^H$ 。

步骤 4 计算矩阵  $A$  的 Moore-Penrose 逆矩阵  $A^\dagger = (A^H A)^\dagger A^H = B^\dagger A^H$ 。

若矩阵  $A_{m \times n}$  的列数大于行数，则矩阵乘积  $AA^H$  的维数比  $A^H A$  的维数小，故选择算法 1.8.1 可花费较少的计算量。反之，若  $A$  的行数大于列数，则选择算法 1.8.2。

## 2. 满秩分解法

令秩亏矩阵  $A_{m \times n}$  具有秩  $r < \min\{m, n\}$ 。若  $A = FG$ ，其中， $F_{m \times r}$  的秩为  $r$ （满列秩矩阵），且  $G_{r \times n}$  的秩也为  $r$ （满行秩矩阵），则称  $A = FG$  为矩阵  $A$  的满秩分解（full-rank decomposition）。

问题是，任意一个矩阵都存在满秩分解吗？下面的命题给出了这个问题的肯定答案。

**命题 1.8.1** [444] 一个秩为  $r$  的  $m \times n$  矩阵  $A$  可以分解为

$$A = F_{m \times r} G_{r \times n} \quad (1.8.6)$$

式中， $F$  和  $G$  分别具有满列秩和满行秩。

若  $A = FG$  是矩阵  $A_{m \times n}$  的满秩分解，则

$$A^\dagger = G^\dagger F^\dagger = G^H (GG^H)^{-1} (F^H F)^{-1} F^H \quad (1.8.7)$$

满足定义 1.8.1 中的四个条件，故  $n \times m$  矩阵  $A^\dagger$  是  $A_{m \times n}$  的 Moore-Penrose 逆矩阵。

初等行变换很容易求出一个秩亏矩阵  $A \in \mathbb{C}^{m \times n}$  的满秩分解<sup>[410]</sup>

- (1) 使用初等行变换将矩阵  $A$  变成行简约阶梯型。
- (2) 按照  $A$  的主元列的顺序组成满列秩矩阵  $F$  的列向量。
- (3) 按照行简约阶梯型的非零行的顺序组成满行秩矩阵  $G$  的行向量。最后，满秩分解为  $A = FG$ 。

在例 1.2.1 中，我们通过初等行变换，得到了  $3 \times 5$  矩阵

$$A = \begin{bmatrix} -3 & 6 & -1 & 1 & -7 \\ 1 & -2 & 2 & 3 & -1 \\ 2 & -4 & 5 & 8 & -4 \end{bmatrix}$$

的简约阶梯型

$$\begin{bmatrix} 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 1 & 2 & -2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

矩阵  $\mathbf{A}$  的主元列为第 1 列和第 3 列, 故

$$\mathbf{F} = \begin{bmatrix} -3 & -1 \\ 1 & 2 \\ 2 & 5 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 1 & 2 & -2 \end{bmatrix}$$

即有满秩分解  $\mathbf{A} = \mathbf{FG}$

$$\begin{bmatrix} -3 & 6 & -1 & 1 & -7 \\ 1 & -2 & 2 & 3 & -1 \\ 2 & -4 & 5 & 8 & -4 \end{bmatrix} = \begin{bmatrix} -3 & -1 \\ 1 & 2 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 & -2 & 0 & -1 & 3 \\ 0 & 0 & 1 & 2 & -2 \end{bmatrix}$$

### 3. 递推法

对矩阵  $\mathbf{A}_{m \times n}$  的前  $k$  列进行分块  $\mathbf{A}_k = [\mathbf{A}_{k-1}, \mathbf{a}_k]$ , 其中,  $\mathbf{a}_k$  是矩阵  $\mathbf{A}$  的第  $k$  列。于是, 分块矩阵  $\mathbf{A}_k$  的 Moore-Penrose 逆矩阵  $\mathbf{A}_k^\dagger$  可以由  $\mathbf{A}_{k-1}^\dagger$  递推计算。当递推到  $k=n$  时, 即获得矩阵  $\mathbf{A}$  的 Moore-Penrose 逆矩阵  $\mathbf{A}^\dagger$ 。这样一种列递推的算法是 Greville 于 1960 年提出的<sup>[205]</sup>。

#### 算法 1.8.3 求 Moore-Penrose 逆矩阵的列递推算法

初始值  $\mathbf{A}_1^\dagger = \mathbf{a}_1^\dagger = (\mathbf{a}_1^H \mathbf{a}_1)^{-1} \mathbf{a}_1^H$ 。

递推 令  $k = 2, 3, \dots, n$ , 进行以下计算

$$\begin{aligned} \mathbf{d}_k &= \mathbf{A}_{k-1}^\dagger \mathbf{a}_k \\ \mathbf{b}_k &= \begin{cases} (1 + \mathbf{d}_k^H \mathbf{d}_k)^{-1} \mathbf{d}_k^H \mathbf{A}_{k-1}^\dagger, & \mathbf{a}_k - \mathbf{A}_{k-1} \mathbf{d}_k = \mathbf{0} \\ (\mathbf{a}_k - \mathbf{A}_{k-1} \mathbf{d}_k)^\dagger, & \mathbf{a}_k - \mathbf{A}_{k-1} \mathbf{d}_k \neq \mathbf{0} \end{cases} \\ \mathbf{A}_k^\dagger &= \begin{bmatrix} \mathbf{A}_{k-1}^\dagger - \mathbf{d}_k \mathbf{b}_k \\ \mathbf{b}_k \end{bmatrix} \end{aligned}$$

上述列递推算法原则上适用于所有矩阵, 但是当矩阵  $\mathbf{A}$  的行比列少的时候, 为了减少递推次数, 宜先使用列递推算法求出  $\mathbf{A}^H$  的 Moore-Penrose 逆矩阵  $(\mathbf{A}^H)^\dagger = \mathbf{A}^{H\dagger}$ , 再利用  $\mathbf{A}^\dagger = (\mathbf{A}^{H\dagger})^H$  之关系得到  $\mathbf{A}^\dagger$ 。

### 4. 迹方法

已知矩阵  $\mathbf{A}_{m \times n}$  的秩为  $r$ 。

#### 算法 1.8.4 求 Moore-Penrose 逆矩阵的迹方法<sup>[413]</sup>

步骤 1 计算  $\mathbf{B} = \mathbf{A}^T \mathbf{A}$ 。

步骤 2 令  $\mathbf{C}_1 = \mathbf{I}$ 。

步骤 3 计算

$$\mathbf{C}_{i+1} = \frac{1}{i} \text{tr}(\mathbf{C}_i \mathbf{B}) \mathbf{I} - \mathbf{C}_i \mathbf{B}, \quad i = 1, 2, \dots, r-1$$

步骤 4 计算

$$\mathbf{A}^\dagger = \frac{\mathbf{r}}{\text{tr}(\mathbf{C}_i \mathbf{B})} \mathbf{C}_i \mathbf{A}^T$$

注意,  $\mathbf{C}_{i+1} \mathbf{B} = \mathbf{O}$ ,  $\text{tr}(\mathbf{C}_i \mathbf{B}) \neq 0$ 。

### 1.8.3 非一致方程的最小范数最小二乘解

前面讨论过一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小范数解  $\mathbf{x} = \mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1}\mathbf{b}$  和非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小二乘解  $\mathbf{x} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{b}$ 。注意，当矩阵  $\mathbf{A}$  秩亏缺时，非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小二乘解不是唯一的。此时，往往希望适当选择一个广义逆矩阵，以便在最小二乘解中获得一个具有最小范数的解。这样一种解称为非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小范数最小二乘解 (minimum norm least squares solution)。

**定义 1.8.2** 对于非一致方程  $\mathbf{A}_{m \times n}\mathbf{x}_{n \times 1} = \mathbf{y}_{m \times 1}$ ，解  $\mathbf{G}\mathbf{y}$  称为  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小范数最小二乘解，若

$$\|\mathbf{G}\mathbf{y}\|_n \leq \|\hat{\mathbf{x}}\|_n \quad \forall \hat{\mathbf{x}} \in \{\hat{\mathbf{x}} : \|\mathbf{A}\hat{\mathbf{x}} - \mathbf{y}\|_m \leq \|\mathbf{A}\mathbf{z} - \mathbf{y}\|_m \quad \forall \mathbf{y} \in \mathbb{R}^m, \mathbf{z} \in \mathbb{R}^n\} \quad (1.8.8)$$

式中， $\|\cdot\|_n$  和  $\|\cdot\|_m$  分别是在  $\mathbb{R}^n$  和  $\mathbb{R}^m$  空间的范数；花括号 {} 表示  $\hat{\mathbf{x}}$  是非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小二乘解，而  $\|\mathbf{G}\mathbf{y}\|_n \leq \|\hat{\mathbf{x}}\|_n$  表示  $\mathbf{G}\mathbf{y}$  是在所有的最小二乘解中具有最小范数的那个解。

**定理 1.8.1**<sup>[424]</sup> 广义逆矩阵  $\mathbf{G}$  使得  $\mathbf{G}\mathbf{y}$  是非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小范数最小二乘解，当且仅当  $\mathbf{G}$  满足条件

$$\mathbf{A}\mathbf{G}\mathbf{A} = \mathbf{A}, \quad (\mathbf{A}\mathbf{G})^\# = \mathbf{A}\mathbf{G}, \quad \mathbf{G}\mathbf{A}\mathbf{G} = \mathbf{G}, \quad (\mathbf{G}\mathbf{A})^\# = \mathbf{G}\mathbf{A} \quad (1.8.9)$$

式中， $\mathbf{A}^\#$  是  $\mathbf{A}$  的伴随矩阵。

利用伴随矩阵的性质  $\mathbf{B}^\# = \mathbf{B}^H$  易知，定理 1.8.1 中的第二个条件  $(\mathbf{A}\mathbf{G})^\# = \mathbf{A}\mathbf{G}$  即  $(\mathbf{A}\mathbf{G})^H = \mathbf{A}\mathbf{G}$ ，第四个条件  $(\mathbf{G}\mathbf{A})^\# = \mathbf{G}\mathbf{A}$  即  $(\mathbf{G}\mathbf{A})^H = \mathbf{G}\mathbf{A}$ 。因此，定理 1.8.1 也可以等价表述为：矩阵  $\mathbf{G}$  使得  $\mathbf{G}\mathbf{y}$  是非一致方程  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的最小范数最小二乘解，当且仅当  $\mathbf{G}$  是  $\mathbf{A}$  的 Moore-Penrose 逆矩阵。

## 1.9 矩阵的直和与 Hadamard 积

本节将讨论两个矩阵之间的特殊求和与乘积。

### 1.9.1 矩阵的直和

**定义 1.9.1**<sup>[203]</sup>  $m \times m$  矩阵  $\mathbf{A}$  与  $n \times n$  矩阵  $\mathbf{B}$  的直和 (direct sum) 记作  $\mathbf{A} \oplus \mathbf{B}$ ，它是一个  $(m+n) \times (m+n)$  矩阵，定义为

$$\mathbf{A} \oplus \mathbf{B} = \begin{bmatrix} \mathbf{A} & \mathbf{O}_{m \times n} \\ \mathbf{O}_{n \times m} & \mathbf{B} \end{bmatrix} \quad (1.9.1)$$

类似地，可以定义多个矩阵的直和。

根据定义，容易证明矩阵的直和具有以下性质 [238, 405]：

- (1) 若  $c$  为常数，则  $c(\mathbf{A} \oplus \mathbf{B}) = c\mathbf{A} \oplus c\mathbf{B}$ 。

(2) 直和通常不满足交换性质, 即  $\mathbf{A} \oplus \mathbf{B} \neq \mathbf{B} \oplus \mathbf{A}$  除非  $\mathbf{A} = \mathbf{B}$ 。

(3) 若  $\mathbf{A}, \mathbf{B}$  为  $m \times m$  矩阵, 且  $\mathbf{C}, \mathbf{D}$  为  $n \times n$  矩阵, 则

$$(\mathbf{A} \pm \mathbf{B}) \oplus (\mathbf{C} \pm \mathbf{D}) = (\mathbf{A} \oplus \mathbf{C}) \pm (\mathbf{B} \oplus \mathbf{D})$$

$$(\mathbf{A} \oplus \mathbf{C})(\mathbf{B} \oplus \mathbf{D}) = \mathbf{AB} \oplus \mathbf{CD}$$

(4) 若  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  分别是  $m \times m, n \times n, p \times p$  矩阵, 则

$$\mathbf{A} \oplus (\mathbf{B} \oplus \mathbf{C}) = (\mathbf{A} \oplus \mathbf{B}) \oplus \mathbf{C} = \mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C}$$

(5) 若  $\mathbf{A}_{m \times m}$  和  $\mathbf{B}_{n \times n}$  均为正交矩阵, 则  $\mathbf{A} \oplus \mathbf{B}$  是  $(m+n) \times (m+n)$  正交矩阵。

(6) 矩阵直和的复共轭、转置、复共轭转置与逆矩阵

$$(\mathbf{A} \oplus \mathbf{B})^* = \mathbf{A}^* \oplus \mathbf{B}^*$$

$$(\mathbf{A} \oplus \mathbf{B})^T = \mathbf{A}^T \oplus \mathbf{B}^T$$

$$(\mathbf{A} \oplus \mathbf{B})^H = \mathbf{A}^H \oplus \mathbf{B}^H$$

$$(\mathbf{A} \oplus \mathbf{B})^{-1} = \mathbf{A}^{-1} \oplus \mathbf{B}^{-1} \quad (\text{若 } \mathbf{A}, \mathbf{B} \text{ 可逆})$$

(7) 矩阵直和的迹、秩、行列式

$$\begin{aligned}\operatorname{tr}\left(\bigoplus_{i=0}^{N-1} \mathbf{A}_i\right) &= \sum_{i=0}^{N-1} \operatorname{tr}(\mathbf{A}_i) \\ \operatorname{rank}\left(\bigoplus_{i=0}^{N-1} \mathbf{A}_i\right) &= \sum_{i=0}^{N-1} \operatorname{rank}(\mathbf{A}_i) \\ \det\left(\bigoplus_{i=0}^{N-1} \mathbf{A}_i\right) &= \prod_{i=0}^{N-1} \det(\mathbf{A}_i)\end{aligned}$$

### 1.9.2 Hadamard 积

**定义 1.9.2**  $m \times n$  矩阵  $\mathbf{A} = [a_{ij}]$  与  $m \times n$  矩阵  $\mathbf{B} = [b_{ij}]$  的 Hadamard 积记作  $\mathbf{A} * \mathbf{B}$ , 它仍然是一个  $m \times n$  矩阵, 其元素定义为两个矩阵对应元素的乘积

$$(\mathbf{A} * \mathbf{B})_{ij} = a_{ij} b_{ij} \tag{1.9.2}$$

即 Hadamard 积是一映射  $\mathbb{R}^{m \times n} \times \mathbb{R}^{m \times n} \mapsto \mathbb{R}^{m \times n}$ 。

Hadamard 积也称 Schur 积或者对应元素乘积 (elementwise product)。

下面的定理描述了矩阵 Hadamard 积的正定性, 常称为 Hadamard 积定理 [238]。

**定理 1.9.1** 若  $m \times m$  矩阵  $\mathbf{A}, \mathbf{B}$  是正定 (或半正定) 的, 则它们的 Hadamard 积  $\mathbf{A} * \mathbf{B}$  也是正定 (或半正定) 的。

**推论 1.9.1** (Fejer 定理)<sup>[238]</sup>  $m \times m$  矩阵  $\mathbf{A}$  是半正定矩阵, 当且仅当

$$\sum_{i=1}^m \sum_{j=1}^m a_{ij} b_{ij} \geq 0$$

对所有  $m \times m$  半正定矩阵  $\mathbf{B}$  成立。

下面的两个定理描述了矩阵的 Hadamard 积与迹之间的关系。

**定理 1.9.2**<sup>[328, p.46]</sup> 令  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  为  $m \times n$  矩阵, 并且  $\mathbf{1} = [1, 1, \dots, 1]^T$  为  $n \times 1$  求和向量,  $\mathbf{D} = \text{diag}(d_1, d_2, \dots, d_m)$ , 其中,  $d_i = \sum_{j=1}^n a_{ij}$ , 则

$$\text{tr}(\mathbf{A}^T (\mathbf{B} * \mathbf{C})) = \text{tr}((\mathbf{A}^T * \mathbf{B}^T) \mathbf{C}) \quad (1.9.3)$$

$$\mathbf{1}^T \mathbf{A}^T (\mathbf{B} * \mathbf{C}) \mathbf{1} = \text{tr}(\mathbf{B}^T \mathbf{D} \mathbf{C}) \quad (1.9.4)$$

**定理 1.9.3**<sup>[328, p.46]</sup> 令  $\mathbf{A}, \mathbf{B}$  为  $n \times n$  正方矩阵, 并且  $\mathbf{1} = [1, 1, \dots, 1]^T$  为  $n \times 1$  求和向量。假定  $\mathbf{M}$  是一个  $n \times n$  对角矩阵  $\mathbf{M} = \text{diag}(\mu_1, \mu_2, \dots, \mu_n)$ , 而  $\mathbf{m} = \mathbf{M}\mathbf{1}$  为  $n \times 1$  向量, 则有

$$\text{tr}(\mathbf{A} \mathbf{M} \mathbf{B}^T \mathbf{M}) = \mathbf{m}^T (\mathbf{A} * \mathbf{B}) \mathbf{m} \quad (1.9.5)$$

$$\text{tr}(\mathbf{A} \mathbf{B}^T) = \mathbf{1}^T (\mathbf{A} * \mathbf{B}) \mathbf{1} \quad (1.9.6)$$

$$\mathbf{M} \mathbf{A} * \mathbf{B}^T \mathbf{M} = \mathbf{M} (\mathbf{A} * \mathbf{B}^T) \mathbf{M} \quad (1.9.7)$$

由定义易知, Hadamard 积满足交换律、结合律以及加法的分配律

$$\mathbf{A} * \mathbf{B} = \mathbf{B} * \mathbf{A} \quad (1.9.8)$$

$$\mathbf{A} * (\mathbf{B} * \mathbf{C}) = (\mathbf{A} * \mathbf{B}) * \mathbf{C} \quad (1.9.9)$$

$$\mathbf{A} * (\mathbf{B} \pm \mathbf{C}) = \mathbf{A} * \mathbf{B} \pm \mathbf{A} * \mathbf{C} \quad (1.9.10)$$

下面汇总了 Hadamard 积的性质<sup>[328]</sup>。

(1) 若  $\mathbf{A}, \mathbf{B}$  均为  $m \times n$  矩阵, 则

$$(\mathbf{A} * \mathbf{B})^T = \mathbf{A}^T * \mathbf{B}^T, \quad (\mathbf{A} * \mathbf{B})^H = \mathbf{A}^H * \mathbf{B}^H, \quad (\mathbf{A} * \mathbf{B})^* = \mathbf{A}^* * \mathbf{B}^*$$

(2) 矩阵  $\mathbf{A}_{m \times n}$  与零矩阵  $\mathbf{O}_{m \times n}$  的 Hadamard 积  $\mathbf{A} * \mathbf{O}_{m \times n} = \mathbf{O}_{m \times n} * \mathbf{A} = \mathbf{O}_{m \times n}$ 。

(3) 若  $c$  为常数, 则  $c(\mathbf{A} * \mathbf{B}) = (c\mathbf{A}) * \mathbf{B} = \mathbf{A} * (c\mathbf{B})$ 。

(4) 正定 (或半正定) 矩阵  $\mathbf{A}, \mathbf{B}$  的 Hadamard 积  $\mathbf{A} * \mathbf{B}$  也是正定 (或半正定) 的。

(5) 矩阵  $\mathbf{A}_{m \times m} = [a_{ij}]$  与单位矩阵  $\mathbf{I}_m$  的 Hadamard 积为  $m \times m$  对角矩阵, 即

$$\mathbf{A} * \mathbf{I}_m = \mathbf{I}_m * \mathbf{A} = \text{diag}(\mathbf{A}) = \text{diag}(a_{11}, a_{22}, \dots, a_{mm})$$

(6) 若  $\mathbf{A}, \mathbf{B}, \mathbf{D}$  为  $m \times m$  矩阵, 且  $\mathbf{D}$  为对角矩阵, 则

$$(\mathbf{D}\mathbf{A}) * (\mathbf{B}\mathbf{D}) = \mathbf{D}(\mathbf{A} * \mathbf{B})\mathbf{D}$$

(7) 若  $\mathbf{A}, \mathbf{C}$  为  $m \times m$  矩阵, 并且  $\mathbf{B}, \mathbf{D}$  为  $n \times n$  矩阵, 则

$$(\mathbf{A} \oplus \mathbf{B}) * (\mathbf{C} \oplus \mathbf{D}) = (\mathbf{A} * \mathbf{C}) \oplus (\mathbf{B} * \mathbf{D})$$

(8) 若  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$  均为  $m \times n$  矩阵, 则

$$(\mathbf{A} + \mathbf{B}) * (\mathbf{C} + \mathbf{D}) = \mathbf{A} * \mathbf{C} + \mathbf{A} * \mathbf{D} + \mathbf{B} * \mathbf{C} + \mathbf{B} * \mathbf{D}$$

(9) 若  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  为  $m \times n$  矩阵, 则

$$\text{tr}(\mathbf{A}^T(\mathbf{B} * \mathbf{C})) = \text{tr}((\mathbf{A}^T * \mathbf{B}^T)\mathbf{C})$$

Hadamard 积服从以下不等式:

(1) Oppenheim 不等式<sup>[31,p.144]</sup> 令  $\mathbf{A}$  与  $\mathbf{B}$  是  $n \times n$  半正定矩阵, 则

$$|\mathbf{A} * \mathbf{B}| \geq a_{11} \cdots a_{nn} |\mathbf{B}| \quad (1.9.11)$$

(2) 令  $\mathbf{A}$  与  $\mathbf{B}$  是  $n \times n$  半正定矩阵, 则<sup>[345]</sup>

$$|\mathbf{A} * \mathbf{B}| \geq |\mathbf{AB}| \quad (1.9.12)$$

(3) 特征值不等式<sup>[31,p.144]</sup> 令  $\mathbf{A}$  与  $\mathbf{B}$  是  $n \times n$  半正定矩阵,  $\lambda_1, \dots, \lambda_n$  是 Hadamard 积  $\mathbf{A} * \mathbf{B}$  的特征值, 而  $\hat{\lambda}_1, \dots, \hat{\lambda}_n$  是矩阵乘积  $\mathbf{AB}$  的特征值, 则

$$\prod_{i=k}^n \lambda_i \geq \prod_{i=k}^n \hat{\lambda}_i, \quad k = 1, \dots, n \quad (1.9.13)$$

(4) Hadamard 积的秩不等式<sup>[345]</sup> 令  $\mathbf{A}$  和  $\mathbf{B}$  是  $n \times n$  矩阵, 则

$$\text{rank}(\mathbf{A} * \mathbf{B}) \leq \text{rank}(\mathbf{A})\text{rank}(\mathbf{B}) \quad (1.9.14)$$

作为 Oppenheim 不等式的特例, 若  $\mathbf{B} = \mathbf{I}_n$ , 且  $\mathbf{A}$  为  $n \times n$  半正定矩阵, 则得 Hadamard 不等式

$$|\mathbf{A}| \leq a_{11} \cdots a_{nn} \quad (1.9.15)$$

这是因为  $|\mathbf{A}| = b_{11} \cdots b_{nn} |\mathbf{A}| \leq |\mathbf{I}_n * \mathbf{A}|$ , 而  $\mathbf{I}_n * \mathbf{A} = \text{diag}(a_{11}, \dots, a_{nn})$ , 故立即有  $|\mathbf{A}| \leq a_{11} \cdots a_{nn}$ 。

矩阵的 Hadamard 积在有损压缩算法(例如 JPEG)中被使用。

在编程语言(例如 MATLAB 和 Mathematica)中, 两个矩阵的 Hadamard 积通常是针对它们各自的阵列型数据 array() 采用符号 \* 运行的。这一阵列型数据的 Hadamard 积需要再转换成矩阵型数据, 才是矩阵的 Hadamard 积。矩阵数据的阵列型数据化称为矩阵的向量化, 而阵列型数据的矩阵型数据化则称为向量的矩阵化。向量化和矩阵化将在稍后专题讨论。

## 1.10 Kronecker 积与 Khatri-Rao 积

1.9 节介绍的 Hadamard 积是关于两个矩阵元素的乘积。本节讨论两个矩阵之间的另外两种特殊乘积——Kronecker 积和 Khatri-Rao 积。

### 1.10.1 Kronecker 积及其性质

两个矩阵的 Kronecker 积分为右 Kronecker 积和左 Kronecker 积。

**定义 1.10.1 (右 Kronecker 积)** [36]  $m \times n$  矩阵  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  和  $p \times q$  矩阵  $\mathbf{B}$  的右 Kronecker 积记作  $\mathbf{A} \otimes \mathbf{B}$ , 是一个  $mp \times nq$  矩阵, 定义为

$$\mathbf{A} \otimes \mathbf{B} = [\mathbf{a}_1 \mathbf{B}, \dots, \mathbf{a}_n \mathbf{B}] = [\mathbf{a}_{ij} \mathbf{B}]_{i=1, j=1}^{m, n} = \begin{bmatrix} \mathbf{a}_{11} \mathbf{B} & \mathbf{a}_{12} \mathbf{B} & \cdots & \mathbf{a}_{1n} \mathbf{B} \\ \mathbf{a}_{21} \mathbf{B} & \mathbf{a}_{22} \mathbf{B} & \cdots & \mathbf{a}_{2n} \mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}_{m1} \mathbf{B} & \mathbf{a}_{m2} \mathbf{B} & \cdots & \mathbf{a}_{mn} \mathbf{B} \end{bmatrix} \quad (1.10.1)$$

**定义 1.10.2 (左 Kronecker 积)** [203, 426]  $m \times n$  矩阵  $\mathbf{A}$  和  $p \times q$  矩阵  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_q]$  的(左) Kronecker 积  $\mathbf{A} \otimes \mathbf{B}$  是一个  $mp \times nq$  矩阵, 定义为

$$[\mathbf{A} \otimes \mathbf{B}]_{\text{left}} = [\mathbf{A} \mathbf{b}_1, \dots, \mathbf{A} \mathbf{b}_q] = [\mathbf{b}_{ij} \mathbf{A}]_{i=1, j=1}^{p, q} = \begin{bmatrix} \mathbf{A} \mathbf{b}_{11} & \mathbf{A} \mathbf{b}_{12} & \cdots & \mathbf{A} \mathbf{b}_{1q} \\ \mathbf{A} \mathbf{b}_{21} & \mathbf{A} \mathbf{b}_{22} & \cdots & \mathbf{A} \mathbf{b}_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A} \mathbf{b}_{p1} & \mathbf{A} \mathbf{b}_{p2} & \cdots & \mathbf{A} \mathbf{b}_{pq} \end{bmatrix} \quad (1.10.2)$$

显然, 无论左或右 Kronecker 积都是一映射:  $\mathbb{R}^{m \times n} \times \mathbb{R}^{p \times q} \mapsto \mathbb{R}^{mp \times nq}$ 。

容易看出, 如果用右 Kronecker 积的形式书写, 则左 Kronecker 积可写成  $[\mathbf{A} \otimes \mathbf{B}]_{\text{left}} = \mathbf{B} \otimes \mathbf{A}$ 。鉴于通常多采用右 Kronecker 积, 为了避免混淆, 本书后面将对 Kronecker 积采用右 Kronecker 积的定义, 除非另有申明。

特别地, 当  $n = 1$  和  $q = 1$  时, 两个矩阵的 Kronecker 积给出两个向量  $\mathbf{a} \in \mathbb{R}^m$  和  $\mathbf{b} \in \mathbb{R}^p$  的 Kronecker 积

$$\mathbf{a} \otimes \mathbf{b} = [\mathbf{a}_i \mathbf{b}]_{i=1}^m = \begin{bmatrix} \mathbf{a}_1 \mathbf{b} \\ \vdots \\ \mathbf{a}_m \mathbf{b} \end{bmatrix} \quad (1.10.3)$$

其结果为  $mp \times 1$  列向量。显然, 两个向量的外积  $\mathbf{x} \circ \mathbf{y} = \mathbf{x} \mathbf{y}^T$  也可以用 Kronecker 积表示为

$$\mathbf{x} \circ \mathbf{y} = \mathbf{x} \otimes \mathbf{y}^T$$

Kronecker 积也称直积 (direct product) 或者张量积 (tensor product) [324]。

汇总文献 [36, 61] 和其他文献, Kronecker 积具有以下性质。

- (1) 对于矩阵  $\mathbf{A}_{m \times n}$  和  $\mathbf{B}_{p \times q}$ , 一般有  $\mathbf{A} \otimes \mathbf{B} \neq \mathbf{B} \otimes \mathbf{A}$ 。
- (2) 任意矩阵与零矩阵的 Kronecker 积等于零矩阵, 即  $\mathbf{A} \otimes \mathbf{O} = \mathbf{O} \otimes \mathbf{A} = \mathbf{O}$ 。
- (3) 若  $\alpha$  和  $\beta$  为常数, 则  $\alpha \mathbf{A} \otimes \beta \mathbf{B} = \alpha \beta (\mathbf{A} \otimes \mathbf{B})$ 。
- (4)  $m$  维与  $n$  维两个单位矩阵的 Kronecker 积为  $mn$  维单位矩阵, 即  $\mathbf{I}_m \otimes \mathbf{I}_n = \mathbf{I}_{mn}$ 。

(5) 对于矩阵  $A_{m \times n}, B_{n \times k}, C_{l \times p}, D_{p \times q}$ , 有

$$(AB) \otimes (CD) = (A \otimes C)(B \otimes D) \quad (1.10.4)$$

(6) 对于矩阵  $A_{m \times n}, B_{p \times q}, C_{p \times q}$ , 有

$$A \otimes (B \pm C) = A \otimes B \pm A \otimes C \quad (1.10.5)$$

$$(B \pm C) \otimes A = B \otimes A \pm C \otimes A \quad (1.10.6)$$

(7) Kronecker 积的转置与复共轭转置

$$(A \otimes B)^T = A^T \otimes B^T, \quad (A \otimes B)^H = A^H \otimes B^H \quad (1.10.7)$$

(8) Kronecker 积的逆矩阵和广义逆矩阵

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \quad (A \otimes B)^\dagger = A^\dagger \otimes B^\dagger \quad (1.10.8)$$

(9) Kronecker 积的秩

$$\text{rank}(A \otimes B) = \text{rank}(A)\text{rank}(B) \quad (1.10.9)$$

(10) Kronecker 积的行列式

$$\det(A_{n \times n} \otimes B_{m \times m}) = (\det A)^m (\det B)^n \quad (1.10.10)$$

(11) Kronecker 积的迹

$$\text{tr}(A \otimes B) = \text{tr}(A)\text{tr}(B) \quad (1.10.11)$$

(12) 对于矩阵  $A_{m \times n}, B_{m \times n}, C_{p \times q}, D_{p \times q}$ , 有

$$(A + B) \otimes (C + D) = A \otimes C + A \otimes D + B \otimes C + B \otimes D \quad (1.10.12)$$

(13) 对于矩阵  $A_{m \times n}, B_{p \times q}, C_{k \times l}$ , 有

$$(A \otimes B) \otimes C = A \otimes (B \otimes C) \quad (1.10.13)$$

(14) 对于矩阵  $A_{m \times n}, B_{k \times l}, C_{p \times q}, D_{r \times s}$ , 有

$$(A \otimes B) \otimes (C \otimes D) = A \otimes B \otimes C \otimes D \quad (1.10.14)$$

(15) 对于矩阵  $A_{m \times n}, B_{p \times q}, C_{n \times r}, D_{q \times s}, E_{r \times k}, F_{s \times l}$ , 有

$$(A \otimes B)(C \otimes D)(E \otimes F) = (ACE) \otimes (BDF) \quad (1.10.15)$$

(16) 作为式 (1.10.15) 的特例, 有 [36, 203]

$$A \otimes D = (AI_p) \otimes (I_q D) = (A \otimes I_q)(I_p \otimes D) \quad (1.10.16)$$

式中  $I_p \otimes D$  为块对角矩阵 (对右 Kronecker 积) 或稀疏矩阵 (对左 Kronecker 积), 而  $A \otimes I_q$  为稀疏矩阵 (右 Kronecker 积) 或块对角矩阵 (左 Kronecker 积)。

(17) 对于矩阵  $A_{m \times n}, B_{p \times q}$ , 有  $\exp(A \otimes B) = \exp(A) \otimes \exp(B)$ 。

(18) 令  $A \in \mathbb{C}^{m \times n}, B \in \mathbb{C}^{p \times q}, b \in \mathbb{C}^p$ , 则 [328,p.47]

$$K_{pm}(A \otimes B) = (B \otimes A)K_{qn} \quad (1.10.17)$$

$$K_{pm}(A \otimes B)K_{nq} = B \otimes A \quad (1.10.18)$$

$$K_{pm}(A \otimes b) = b \otimes A \quad (1.10.19)$$

$$K_{mp}(b \otimes A) = A \otimes b \quad (1.10.20)$$

### 1.10.2 广义 Kronecker 积

与两个矩阵的 Kronecker 积不同, 广义 Kronecker 积是多个矩阵组成的矩阵组与另一个矩阵的 Kronecker 积。

**定义 1.10.3 (广义 Kronecker 积)** [405] 给定  $N$  个  $m \times r$  矩阵  $\mathbf{A}_i, i = 1, \dots, N$  组成矩阵组  $\{\mathbf{A}\}_N$ 。该矩阵组与  $N \times l$  矩阵  $\mathbf{B}$  的 Kronecker 积称为广义 Kronecker 积, 定义为

$$\{\mathbf{A}\}_N \otimes \mathbf{B} = \begin{bmatrix} \mathbf{A}_1 \otimes \mathbf{b}_1 \\ \vdots \\ \mathbf{A}_N \otimes \mathbf{b}_N \end{bmatrix} \quad (1.10.21)$$

式中,  $\mathbf{b}_i$  是矩阵  $\mathbf{B}$  的第  $i$  个行向量。

显然, 若每一个矩阵  $\mathbf{A}_i$  相同, 则广义 Kronecker 积简化为一般的左 Kronecker 积。

**例 1.10.1** 令

$$\{\mathbf{A}\}_2 = \left\{ \begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 2 & -j \\ 1 & j \end{bmatrix} \right\}, \quad \mathbf{B} = \begin{bmatrix} 1 & 2 \\ 1 & -1 \end{bmatrix}$$

则广义 Kronecker 积为

$$\{\mathbf{A}\}_2 \otimes \mathbf{B} = \left[ \begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 2 & -j \\ 1 & j \end{bmatrix} \otimes [1, 2] \quad \begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 2 & -j \\ 1 & j \end{bmatrix} \otimes [1, -1] \right] = \begin{bmatrix} 1 & 1 & 2 & 2 \\ 2 & -1 & 4 & -2 \\ 2 & -j & -2 & j \\ 1 & j & -1 & -j \end{bmatrix}$$

需要注意的是, 两个矩阵组  $\{\mathbf{A}\}$  和  $\{\mathbf{B}\}$  的广义 Kronecker 积得到的仍然是矩阵组, 而不是单个矩阵。

**例 1.10.2** 令

$$\{\mathbf{A}\}_2 = \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \end{bmatrix} \right\}, \quad \{\mathbf{B}\} = \left\{ \begin{bmatrix} 1, 1 \\ 1, -1 \end{bmatrix} \right\}$$

则广义 Kronecker 积为

$$\{\mathbf{A}\}_2 \otimes \{\mathbf{B}\} = \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \end{bmatrix} \otimes [1, 1] \quad \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \end{bmatrix} \otimes [1, -1] \right\} = \left\{ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \\ 1 & j & -1 & -j \end{bmatrix} \right\}$$

广义 Kronecker 积在滤波器组的分析、Haar 变换和 Hadamard 变换的快速算法的推导中有着重要的应用 [405]。基于广义 Kronecker 积, 可以推导快速 Fourier 变换 (FFT) 算法, 详见第 2 章 2.8.3 节。

### 1.10.3 Khatri-Rao 积

**定义 1.10.4 (Khatri-Rao 积)** 两个具有相同列数的矩阵  $\mathbf{G} \in \mathbb{R}^{p \times n}$  和  $\mathbf{F} \in \mathbb{R}^{q \times n}$  的 Khatri-Rao 积记为  $\mathbf{F} \odot \mathbf{G}$ , 并定义为<sup>[266, 423]</sup>

$$\mathbf{F} \odot \mathbf{G} = [\mathbf{f}_1 \otimes \mathbf{g}_1, \mathbf{f}_2 \otimes \mathbf{g}_2, \dots, \mathbf{f}_n \otimes \mathbf{g}_n] \in \mathbb{R}^{pq \times n} \quad (1.10.22)$$

它由两个矩阵的对应列向量的 Kronecker 积排列而成。因此, Khatri-Rao 积又叫对应列 Kronecker 积 (columnwise Kronecker product)。

更一般地, 若  $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_u]$  和  $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_u]$  是两个分块矩阵, 且分块矩阵  $\mathbf{A}_i$  和  $\mathbf{B}_i$  具有相同的列数, 则

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{A}_1 \otimes \mathbf{B}_1, \mathbf{A}_2 \otimes \mathbf{B}_2, \dots, \mathbf{A}_u \otimes \mathbf{B}_u] \quad (1.10.23)$$

Khatri-Rao 积具有以下性质<sup>[36, 319]</sup>:

(1) Khatri-Rao 积本身的基本性质

$$\text{分配律 } (\mathbf{A} + \mathbf{B}) \odot \mathbf{D} = \mathbf{A} \odot \mathbf{D} + \mathbf{B} \odot \mathbf{D}$$

$$\text{结合律 } \mathbf{A} \odot \mathbf{B} \odot \mathbf{C} = (\mathbf{A} \odot \mathbf{B}) \odot \mathbf{C} = \mathbf{A} \odot (\mathbf{B} \odot \mathbf{C})$$

$$\text{交换律 } \mathbf{A} \odot \mathbf{B} = \mathbf{K}_{nn}(\mathbf{B} \odot \mathbf{A}) \text{ (其中, 矩阵 } \mathbf{K}_{nn} \text{ 为交换矩阵)}$$

(2) Khatri-Rao 积与 Hadamard 积的关系

$$(\mathbf{A} \odot \mathbf{B}) * (\mathbf{C} \odot \mathbf{D}) = (\mathbf{A} * \mathbf{C}) \odot (\mathbf{B} * \mathbf{D}) \quad (1.10.24)$$

$$(\mathbf{A} \odot \mathbf{B})^T (\mathbf{A} \odot \mathbf{B}) = (\mathbf{A}^T \mathbf{A}) * (\mathbf{B}^T \mathbf{B}) \quad (1.10.25)$$

$$(\mathbf{A} \odot \mathbf{B})^\dagger = [(\mathbf{A}^T \mathbf{A}) * (\mathbf{B}^T \mathbf{B})]^\dagger (\mathbf{A} \odot \mathbf{B})^T \quad (1.10.26)$$

推而广之, 有  $(\mathbf{A} \odot \mathbf{B} \odot \mathbf{C})^T (\mathbf{A} \odot \mathbf{B} \odot \mathbf{C}) = (\mathbf{A}^T \mathbf{A}) * (\mathbf{B}^T \mathbf{B}) * (\mathbf{C}^T \mathbf{C})$  和  $(\mathbf{A} \odot \mathbf{B} \odot \mathbf{C})^\dagger = [(\mathbf{A}^T \mathbf{A}) * (\mathbf{B}^T \mathbf{B}) * (\mathbf{C}^T \mathbf{C})]^\dagger (\mathbf{A} \odot \mathbf{B} \odot \mathbf{C})^T$ 。

(3) Khatri-Rao 积与 Kronecker 积的关系

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{F} \odot \mathbf{G}) = \mathbf{A}\mathbf{F} \odot \mathbf{B}\mathbf{G} \quad (1.10.27)$$

## 1.11 向量化与矩阵化

矩阵与向量之间存在相互转换的函数或算子, 它们是向量化算子和矩阵化算子。

### 1.11.1 矩阵的向量化与向量的矩阵化

矩阵  $\mathbf{A} \in \mathbb{R}^{m \times n}$  的向量化 (vectorization)  $\text{vec}(\mathbf{A})$  是一线性变换, 它将矩阵  $\mathbf{A} = [a_{ij}]$  的元素按列堆栈 (column stacking), 排列成一个  $mn \times 1$  向量

$$\text{vec}(\mathbf{A}) = [a_{11}, \dots, a_{m1}, \dots, a_{1n}, \dots, a_{mn}]^T \quad (1.11.1)$$

矩阵也可以按行堆栈 (stack the rows) 为行向量, 称为矩阵的行向量化, 用符号  $\text{rvec}(\mathbf{A})$  表示, 定义为

$$\text{rvec}(\mathbf{A}) = [a_{11}, \dots, a_{1n}, \dots, a_{m1}, \dots, a_{mn}] \quad (1.11.2)$$

例如

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad \text{vec}(\mathbf{A}) = [a_{11}, a_{21}, a_{12}, a_{22}]^T, \quad \text{rvec}(\mathbf{A}) = [a_{11}, a_{12}, a_{21}, a_{22}]$$

注意, 矩阵的向量化结果为列向量, 行向量化结果为行向量。显然, 矩阵的向量化和行向量化之间存在下列关系

$$\text{rvec}(\mathbf{A}) = (\text{vec}(\mathbf{A}^T))^T, \quad \text{vec}(\mathbf{A}^T) = (\text{rvec}(\mathbf{A}))^T \quad (1.11.3)$$

对一幅图像进行采样, 采样数据组成一矩阵。为了传送图像信号, 通常先按行扫描, 然后将各行数据串接起来。因此, 这是一种典型的行向量化。

显然, 对于一个  $m \times n$  矩阵  $\mathbf{A}$ , 向量  $\text{vec}(\mathbf{A})$  和  $\text{vec}(\mathbf{A}^T)$  含有相同的元素, 但排列次序不同。因此, 存在一个唯一的  $mn \times mn$  置换矩阵, 可以将一个矩阵的向量化  $\text{vec}(\mathbf{A})$  变换为其转置矩阵的向量化  $\text{vec}(\mathbf{A}^T)$ 。这一置换矩阵称为交换矩阵 (commutation matrix), 记作  $\mathbf{K}_{mn}$ , 定义为

$$\mathbf{K}_{mn} \text{vec}(\mathbf{A}) = \text{vec}(\mathbf{A}^T) \quad (1.11.4)$$

类似地, 可以将转置矩阵的向量化  $\text{vec}(\mathbf{A}^T)$  变换为原矩阵的向量化  $\text{vec}(\mathbf{A})$  的交换矩阵是一个  $nm \times nm$  置换矩阵, 记作  $\mathbf{K}_{nm}$ , 定义为

$$\mathbf{K}_{nm} \text{vec}(\mathbf{A}^T) = \text{vec}(\mathbf{A}) \quad (1.11.5)$$

由式 (1.11.4) 和式 (1.11.5) 易知  $\mathbf{K}_{nm} \mathbf{K}_{mn} \text{vec}(\mathbf{A}) = \mathbf{K}_{nm} \text{vec}(\mathbf{A}^T) = \text{vec}(\mathbf{A})$ 。由于此式对任意  $m \times n$  矩阵  $\mathbf{A}$  均成立, 故  $\mathbf{K}_{nm} \mathbf{K}_{mn} = \mathbf{I}_{mn}$ , 即有  $\mathbf{K}_{mn}^{-1} = \mathbf{K}_{nm}$ 。

$mn \times mn$  交换矩阵  $\mathbf{K}_{mn}$  具有以下常用性质<sup>[327]</sup>:

(1)  $\mathbf{K}_{mn} \text{vec}(\mathbf{A}) = \text{vec}(\mathbf{A}^T)$  和  $\mathbf{K}_{nm} \text{vec}(\mathbf{A}^T) = \text{vec}(\mathbf{A})$ , 其中  $\mathbf{A}$  为  $m \times n$  矩阵。

(2)  $\mathbf{K}_{mn}^T \mathbf{K}_{mn} = \mathbf{K}_{mn} \mathbf{K}_{mn}^T = \mathbf{I}_{mn}$  或  $\mathbf{K}_{mn}^{-1} = \mathbf{K}_{nm}$ 。

(3)  $\mathbf{K}_{mn}^T = \mathbf{K}_{nm}$ 。

(4)  $\mathbf{K}_{mn}$  可以表示为基本向量的 Kronecker 积

$$\mathbf{K}_{mn} = \sum_{j=1}^n (\mathbf{e}_j^T \otimes \mathbf{I}_m \otimes \mathbf{e}_j)$$

(5)  $\mathbf{K}_{1n} = \mathbf{K}_{n1} = \mathbf{I}_n$ 。

(6) 交换矩阵的秩  $\text{rank}(\mathbf{K}_{mn}) = 1 + d(m-1, n-1)$ , 其中  $d(m, n)$  是  $m$  和  $n$  之间的最大公约数 ( $d(n, 0) = d(0, n) = n$ )。

(7) 交换矩阵  $K_{nn}$  的特征值取 1 和 -1, 它们的多重度分别为  $\frac{1}{2}n(n+1)$  和  $\frac{1}{2}n(n-1)$ 。

(8)  $K_{mn}(A \otimes B)K_{pq} = B \otimes A$ , 或等价写作  $K_{mn}(A \otimes B) = (B \otimes A)K_{qp}$ , 其中  $A$  是  $n \times p$  矩阵,  $B$  为  $m \times q$  矩阵。特别地,  $K_{mn}(A_{n \times n} \otimes B_{m \times m}) = (B \otimes A)K_{mn}$ 。

(9)  $\text{tr}(K_{mn}(A_{m \times n} \otimes B_{m \times n})) = \text{tr}(A^T B) = (\text{vec } A^T)^T K_{mn}(\text{vec } A)$ 。

$mn \times mn$  交换矩阵  $K_{mn}$  的构造方法如下: 每一行只赋一个元素 1, 其他元素全部为 0。首先, 第 1 行第 1 个元素为 1, 然后这个 1 元素右移  $m$  位, 变成第 2 行该位置的 1 元素。第 2 行该位置的 1 元素再右移  $m$  位, 又变成第 3 行该位置的 1 元素。依此类推, 找到下一行 1 元素的位置。但是, 如果向右移位时超过第  $mn$  列, 则应该转到下一行继续移位, 并且多移 1 位, 再在此位置赋 1。例如

$$K_{24} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad K_{42} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

因此, 交换矩阵  $K_{mn}$  和  $K_{nm}$  是唯一确定的。以矩阵  $A_{4 \times 2}$  为例, 显然有

$$K_{42} \text{vec}(A) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{41} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{42} \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ a_{31} \\ a_{32} \\ a_{41} \\ a_{42} \end{bmatrix} = \text{vec}(A^T)$$

一个  $mn \times 1$  向量  $a = [a_1, \dots, a_{mn}]^T$  转换为一个  $m \times n$  矩阵  $A$  的运算称为矩阵化 (matrixing, maxicization), 用符号  $\text{unvec}_{m,n}(a)$  表示, 定义为

$$A_{m \times n} = \text{unvec}_{m,n}(a) = \begin{bmatrix} a_1 & a_{m+1} & \cdots & a_{m(n-1)+1} \\ a_2 & a_{m+2} & \cdots & a_{m(n-1)+2} \\ \vdots & \vdots & \ddots & \vdots \\ a_m & a_{2m} & \cdots & a_{mn} \end{bmatrix} \quad (1.11.6)$$

显然, 矩阵  $A$  第  $(i, j)$  元素  $A_{ij}$  与向量  $a$  的第  $k$  个元素  $a_k$  之间存在下列转换公式

$$A_{ij} = a_{i+(j-1)m}, \quad i = 1, \dots, m; j = 1, \dots, n \quad (1.11.7)$$

类似地, 一个  $1 \times mn$  的行向量  $b = [b_1, \dots, b_{mn}]$  直接转换为一个  $m \times n$  矩阵  $B$  的运算

称为行向量的矩阵化, 记作  $\text{unrvec}_{m,n}(\mathbf{b})$ , 定义为

$$\mathbf{B}_{m \times n} = \text{unrvec}_{m,n}(\mathbf{b}) = \begin{bmatrix} b_1 & b_2 & \cdots & b_n \\ b_{n+1} & b_{n+2} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{(m-1)n+1} & b_{(m-1)n+2} & \cdots & b_{mn} \end{bmatrix} \quad (1.11.8)$$

观察易知, 矩阵  $\mathbf{B}$  的元素  $B_{ij}$  与行向量  $\mathbf{b}$  的元素  $b_k$  之间存在下列关系

$$B_{ij} = b_{j+(i-1)n}, \quad i = 1, \dots, m; j = 1, \dots, n \quad (1.11.9)$$

按照定义, 矩阵化和向量化之间存在以下关系

$$\begin{array}{c} \left[ \begin{array}{ccc} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mn} \end{array} \right] \xrightarrow{\substack{\text{列向量化} \\ \text{矩阵化}}} [A_{11}, \dots, A_{m1}, \dots, A_{1n}, \dots, A_{mn}]^T \\ \left[ \begin{array}{ccc} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mn} \end{array} \right] \xrightarrow{\substack{\text{行向量化} \\ \text{矩阵化}}} [A_{11}, \dots, A_{1n}, \dots, A_{m1}, \dots, A_{mn}] \end{array}$$

或表示为

$$\text{unvec}_{m,n}(\mathbf{a}) = \mathbf{A}_{m \times n} \iff \text{vec}(\mathbf{A}_{m \times n}) = \mathbf{a}_{mn \times 1} \quad (1.11.10)$$

$$\text{unrvec}_{m,n}(\mathbf{b}) = \mathbf{B}_{m \times n} \iff \text{rvec}(\mathbf{B}_{m \times n}) = \mathbf{b}_{1 \times mn} \quad (1.11.11)$$

### 1.11.2 向量化算子的性质

向量化算子  $\text{vec}$  具有以下性质 [69, 328]。

- (1) 转置矩阵的向量化  $\text{vec}(\mathbf{A}^T) = \mathbf{K}_{mn} \text{vec}(\mathbf{A})$ , 其中  $\mathbf{A} \in \mathbb{C}^{m \times n}$ 。
- (2) 矩阵之和的向量化  $\text{vec}(\mathbf{A} + \mathbf{B}) = \text{vec}(\mathbf{A}) + \text{vec}(\mathbf{B})$ 。
- (3) 矩阵乘积的迹

$$\text{tr}(\mathbf{A}^T \mathbf{B}) = (\text{vec}(\mathbf{A}))^T \text{vec}(\mathbf{B}) \quad (1.11.12)$$

$$\text{tr}(\mathbf{A}^H \mathbf{B}) = (\text{vec}(\mathbf{A}))^H \text{vec}(\mathbf{B}) \quad (1.11.13)$$

$$\text{tr}(\mathbf{ABC}) = (\text{vec}(\mathbf{A}))^T (\mathbf{I}_p \otimes \mathbf{B}) \text{vec}(\mathbf{C}) \quad (1.11.14)$$

而四个矩阵乘积的迹为 [328, p.31]

$$\text{tr}(\mathbf{ABCD}) = (\text{vec}(\mathbf{D}^T))^T (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B}) = (\text{vec}(\mathbf{D}))^T (\mathbf{A} \otimes \mathbf{C}^T) \text{vec}(\mathbf{B}^T)$$

- (4)  $m \times n$  矩阵  $\mathbf{A}$  和  $\mathbf{B}$  的 Hadamard 积的向量化函数

$$\text{vec}(\mathbf{A} * \mathbf{B}) = \text{vec}(\mathbf{A}) * \text{vec}(\mathbf{B}) = \text{diag}(\text{vec}(\mathbf{A})) \text{vec}(\mathbf{B}) \quad (1.11.15)$$

式中  $\text{diag}(\text{vec}(\mathbf{A}))$  表示向量化函数  $\text{vec}(\mathbf{A})$  各元素为对角元素的对角矩阵。

- (5) 两个向量的 Kronecker 积可以表示成向量外积的向量化

$$\mathbf{a} \otimes \mathbf{b} = \text{vec}(\mathbf{ba}^T) = \text{vec}(\mathbf{b} \circ \mathbf{a}) \quad (1.11.16)$$

(6) 向量化函数与 Khatri-Rao 积的关系<sup>[61]</sup>

$$\text{vec}(\mathbf{U}_{m \times p} \mathbf{V}_{p \times p} \mathbf{W}_{p \times n}) = (\mathbf{W}^T \odot \mathbf{U}) \text{d}(\mathbf{V}) \quad (1.11.17)$$

式中,  $\text{d}(\mathbf{V}) = [v_{11}, \dots, v_{pp}]^T$  是由矩阵  $\mathbf{V}$  的对角元素组成的列向量。

(7) 矩阵  $\mathbf{A}_{m \times p} \mathbf{B}_{p \times q} \mathbf{C}_{q \times n}$  乘积的向量化与 Kronecker 积的关系<sup>[440,p.263]</sup>

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B}) \quad (1.11.18)$$

$$\text{vec}(\mathbf{ABC}) = (\mathbf{I}_q \otimes \mathbf{AB}) \text{vec}(\mathbf{C}) = (\mathbf{C}^T \mathbf{B}^T \otimes \mathbf{I}_m) \text{vec}(\mathbf{A}) \quad (1.11.19)$$

$$\text{vec}(\mathbf{AC}) = (\mathbf{I}_p \otimes \mathbf{A}) \text{vec}(\mathbf{C}) = (\mathbf{C}^T \otimes \mathbf{I}_m) \text{vec}(\mathbf{A}) \quad (1.11.20)$$

(8) Kronecker 积的向量化: 令  $\mathbf{X} \in \mathbb{R}^{p \times m}$  和  $\mathbf{Y} \in \mathbb{R}^{n \times q}$ , 则<sup>[328,p.184]</sup>

$$\text{vec}(\mathbf{X} \otimes \mathbf{Y}) = (\mathbf{I}_m \otimes \mathbf{K}_{qp} \otimes \mathbf{I}_n) (\text{vec} \mathbf{X} \otimes \text{vec} \mathbf{Y}) \quad (1.11.21)$$

**例 1.11.1** 矩阵方程  $\mathbf{AXB} = \mathbf{C}$  中  $\mathbf{A}$  和  $\mathbf{X}$  分别是  $m \times n$  和  $n \times p$  矩阵, 而  $\mathbf{B}$  和  $\mathbf{C}$  的维数分别是  $p \times q$  和  $m \times q$ 。利用向量化函数的性质  $\text{vec}(\mathbf{AXB}) = (\mathbf{B}^T \otimes \mathbf{A}) \text{vec}(\mathbf{X})$ , 原矩阵方程的向量化  $\text{vec}(\mathbf{AXB}) = \text{vec}(\mathbf{C})$  可以用 Kronecker 积改写为<sup>[424]</sup>  $(\mathbf{B}^T \otimes \mathbf{A}) \text{vec}(\mathbf{X}) = \text{vec}(\mathbf{C})$ , 由此得  $\text{vec}(\mathbf{X}) = (\mathbf{B}^T \otimes \mathbf{A})^\dagger \text{vec}(\mathbf{C})$ 。然后, 将  $\text{vec}(\mathbf{X})$  矩阵化, 即可获得原矩阵方程  $\mathbf{AXB} = \mathbf{C}$  的解矩阵  $\mathbf{X}$ 。

**例 1.11.2** 在系统理论中, 经常需要求解矩阵方程  $\mathbf{AX} + \mathbf{XB} = \mathbf{Y}$ , 其中, 所有矩阵的维数均为  $n \times n$ 。利用向量化算子的性质  $\text{vec}(\mathbf{ADB}) = (\mathbf{B}^T \otimes \mathbf{A}) \text{vec}(\mathbf{D})$ , 立即有

$$(\mathbf{I}_n \otimes \mathbf{A} + \mathbf{B}^T \otimes \mathbf{I}_n) \text{vec}(\mathbf{X}) = \text{vec}(\mathbf{Y})$$

由此得

$$\text{vec}(\mathbf{X}) = (\mathbf{I}_n \otimes \mathbf{A} + \mathbf{B}^T \otimes \mathbf{I}_n)^\dagger \text{vec}(\mathbf{Y})$$

然后, 将  $\text{vec}(\mathbf{X})$  矩阵化, 即可得到矩阵方程  $\mathbf{AX} + \mathbf{XB} = \mathbf{Y}$  的解  $\mathbf{X}$ 。

## 1.12 稀疏表示与压缩感知

使用少量基本信号的线性组合表示一目标信号, 称为信号的稀疏表示。压缩感知又称压缩采样, 是一种与数据采集的传统 Nyquist 方法不同的新采样技术。压缩感知理论认为, 某些信号和图像可以从比传统方法少得多的样本中恢复或重构。压缩感知与稀疏表示密切相关。

### 1.12.1 稀疏向量与稀疏表示

一个含有大多数零元素的向量或者矩阵称为稀疏向量 (sparse vector) 或者稀疏矩阵 (sparse matrix)。

信号向量  $\mathbf{y} \in \mathbb{R}^m$  最多可分解为  $m$  个正交基 (向量)  $\mathbf{g}_k \in \mathbb{R}^m, k = 1, \dots, m$ , 这些正交基的集合称为完备正交基 (complete orthogonal basis)。此时, 信号分解

$$\mathbf{y} = \mathbf{G}\mathbf{c} = \sum_{i=1}^m c_i \mathbf{g}_i \quad (1.12.1)$$

中的系数向量  $\mathbf{c}$  一定是非稀疏的。

若将信号向量  $\mathbf{y} \in \mathbb{R}^m$  分解为  $n$  个  $m$  维向量  $\mathbf{a}_i \in \mathbb{R}^m, i = 1, \dots, n$  (其中  $n > m$ ) 的线性组合

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \sum_{i=1}^n x_i \mathbf{a}_i \quad (n > m) \quad (1.12.2)$$

则  $n (> m)$  个向量  $\mathbf{a}_i \in \mathbb{R}^m, i = 1, \dots, n$  不可能是正交基的集合。为了与基区别, 这些列向量通常被称为原子 (atom) 或框架。由于原子的个数  $n$  大于向量空间  $\mathbb{R}^m$  的维数, 所以称这些原子的集合是过完备的 (overcomplete)。过完备的原子组成的矩阵  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n] \in \mathbb{R}^{m \times n} (n > m)$  称为字典或库 (dictionary)。

对字典 (矩阵)  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , 通常作如下假设:

- (1)  $\mathbf{A}$  的行数  $m$  小于列数  $n$ 。
- (2)  $\mathbf{A}$  具有满行秩, 即  $\text{rank}(\mathbf{A}) = m$ 。
- (3)  $\mathbf{A}$  的列具有单位 Euclidean 范数  $\|\mathbf{a}_j\|_2 = 1, j = 1, \dots, n$ 。

信号过完备分解式 (1.12.2) 为欠定方程, 存在无穷多组解向量  $\mathbf{x}$ 。求解这种欠定方程有两种常用方法。

### 1. 经典方法 (求最小 $L_2$ 范数解)

$$\min \|\mathbf{x}\|_2 \quad \text{subject to } \mathbf{A}\mathbf{x} = \mathbf{y} \quad (1.12.3)$$

这种方法的优点是: 解是唯一的, 其物理解释为最小能量解。然而, 由于这种解的每个元素通常取非零值, 故不符合许多实际应用的稀疏表示要求。

### 2. 现代方法 (求最小 $L_0$ 范数解)

$$\min \|\mathbf{x}\|_0 \quad \text{subject to } \mathbf{A}\mathbf{x} = \mathbf{y} \quad (1.12.4)$$

式中  $L_0$  范数  $\|\mathbf{x}\|_0$  是向量  $\mathbf{x}$  的非零元素的个数。

这种方法的优点是: 针对许多实际应用情况, 只选择一个稀疏的解向量, 因为稀疏的系数向量  $\mathbf{x}$  是许多应用中令人感兴趣的解。这一方法的缺点是计算比较难于处理。

在存在观测数据误差或背景噪声的情况下, 最小  $L_0$  范数解为

$$\min \|\mathbf{x}\|_0 \quad \text{subject to } \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \varepsilon \quad (1.12.5)$$

式中  $\varepsilon$  为一很小的误差或扰动。

当系数向量  $\mathbf{x}$  是稀疏向量时, 信号分解  $\mathbf{y} = \mathbf{A}\mathbf{x}$  称为 (信号的) 稀疏分解 (sparse decomposition)。其中, 字典矩阵  $\mathbf{A}$  的列常称为解释变量 (explanatory variables); 向量  $\mathbf{y}$

称为响应变量 (response variable) 或目标信号;  $\mathbf{A}\mathbf{x}$  称为响应的线性预测; 而  $\mathbf{x}$  则可视为目标信号  $\mathbf{y}$  相对于字典  $\mathbf{A}$  的一种表示。

因此, 称式 (1.12.4) 是目标信号  $\mathbf{y}$  相对于字典  $\mathbf{A}$  的稀疏表示 (sparse representation), 而式 (1.12.5) 则称为目标信号的稀疏逼近 (sparse approximation)。

给定一个正整数  $K$ , 若向量  $\mathbf{x}$  的  $L_0$  范数  $\|\mathbf{x}\|_0 \leq K$ , 则称  $\mathbf{x}$  是  $K$  稀疏的。当给定一信号向量  $\mathbf{y}$  和一字典  $\mathbf{A}$  时, 满足  $\mathbf{A}\mathbf{x} = \mathbf{y}$  的系数向量  $\mathbf{x}$  若具有最小  $L_0$  范数, 则称  $\mathbf{x}$  是目标信号  $\mathbf{y}$  相对于字典  $\mathbf{A}$  的最稀疏表示 (sparsest representation)。

稀疏表示属于线性求逆问题 (linear inverse problem)。在通信和信息论中, 矩阵  $\mathbf{A} \in \mathbb{R}^{m \times N}$  和向量  $\mathbf{x} \in \mathbb{R}^N$  分别代表编码矩阵和待发送的明码文本 (plaintext), 观测向量  $\mathbf{y} \in \mathbb{R}^m$  则为密码文本 (ciphertext)。线性求逆问题便成了解码问题: 即如何从密码文本  $\mathbf{y}$  恢复原始明码文本  $\mathbf{x}$ 。

稀疏表示是信号处理、通信和信息论、计算机视觉、机器学习和模式识别等领域近几年的一大研究和应用热点。

### 1.12.2 人脸识别的稀疏表示

作为一个典型应用, 具体考虑人脸识别问题。假定共有  $c$  类目标, 每一类目标的脸部的每一幅训练图像的矩阵表示结果已经向量化, 表示成  $m \times 1$  向量 (其中  $m = R_1 \times R_2$  为一幅图像的采样样本数目, 例如  $m = 512 \times 512$ ), 并且每一列都归一化为单位 Euclidean 范数。于是, 第  $i$  类目标的脸部在不同照度下拍摄的  $N_i$  个训练图像即可表示成  $m \times N_i$  维数据矩阵  $\mathbf{D}_i = [\mathbf{d}_{i,1}, \mathbf{d}_{i,2}, \dots, \mathbf{d}_{i,N_i}] \in \mathbb{R}^{m \times N_i}$ 。给定一足够丰富的训练集  $\mathbf{D}_i$ , 则第  $i$  个实验对象在另一照度下拍摄的新图像  $\mathbf{y}$  即可以表示成已知训练图像的一线性组合  $\mathbf{y} \approx \mathbf{D}_i \boldsymbol{\alpha}_i$ , 其中  $\boldsymbol{\alpha}_i \in \mathbb{R}^m$  为系数向量。问题是: 在实际应用中, 往往不知道新的实验样本的具体目标属性, 而需要进行人脸识别: 判断该样本究竟属于哪一个目标类。

如果我们大致知道或者猜测到新的测试样本是  $c$  类目标中的某类目标的信号, 就可以将这  $c$  类目标的训练样本构造的字典合写成一个训练数据矩阵

$$\mathbf{D} = [\mathbf{D}_1, \dots, \mathbf{D}_c] = [\mathbf{d}_{1,1}, \dots, \mathbf{d}_{1,N_1}, \dots, \mathbf{d}_{c,1}, \dots, \mathbf{d}_{c,N_c}] \in \mathbb{R}^{m \times N} \quad (1.12.6)$$

其中  $N = \sum_{i=1}^c N_i$  表示所有  $c$  类目标的训练图像的总个数。于是, 待识别的人脸图像  $\mathbf{y}$  可以表示成线性组合

$$\mathbf{y} = \mathbf{D}\boldsymbol{\alpha}_0 = [\mathbf{d}_{1,1}, \dots, \mathbf{d}_{1,N_1}, \dots, \mathbf{d}_{c,1}, \dots, \mathbf{d}_{c,N_c}] \begin{bmatrix} \mathbf{0}_{N_1} \\ \vdots \\ \mathbf{0}_{N_{i-1}} \\ \boldsymbol{\alpha}_i \\ \mathbf{0}_{N_{i+1}} \\ \vdots \\ \mathbf{0}_{N_c} \end{bmatrix} \quad (1.12.7)$$

其中  $\mathbf{0}_{N_k}, k = 1, \dots, i-1, i+1, \dots, c$  为  $N_k$  维零向量。

现在，人脸识别便变成一个矩阵方程的求解问题或者线性求逆问题：已知数据向量  $\mathbf{y}$  和数据矩阵  $\mathbf{D}$ ，求矩阵方程  $\mathbf{y} = \mathbf{D}\boldsymbol{\alpha}_0$  的解向量  $\boldsymbol{\alpha}_0$ 。

需要注意的是：通常  $m < N$ ，故矩阵方程  $\mathbf{y} = \mathbf{D}\boldsymbol{\alpha}_0$  欠定，具有无穷多个解。其中，最稀疏的解才是我们感兴趣的解。

鉴于解向量必须是稀疏向量，故人脸识别问题可以描述成一个优化问题

$$\min \|\boldsymbol{\alpha}_0\|_0 \quad \text{subject to } \mathbf{y} = \mathbf{D}\boldsymbol{\alpha}_0 \quad (1.12.8)$$

这是一个典型的  $L_0$  范数最小化问题。这一问题的求解将在第 6 章中详细讨论。

### 1.12.3 稀疏编码

稀疏编码可以给出刺激 (stimuli) 的简洁表示。只给定未标识的输入数据，稀疏编码对可以捕捉数据中的高级特征的基函数进行机器学习。当应用于自然图像时，稀疏编码学会的基函数类似于视觉皮层的感受域<sup>[379]</sup>。当应用于其他自然刺激 (例如语音和视频) 时，稀疏编码可以产生听觉和视觉定位的基 (localized bases)<sup>[313, 378]</sup>。

稀疏编码的大多数模型基于线性生成模型<sup>[336]</sup>。在这一模型中，符号以一种线性的方式组合，并用于对输入进行逼近。稀疏编码问题的提法是<sup>[309]</sup>：给定一个  $m$  维实值输入向量  $\mathbf{x} \in \mathbb{R}^m$ ，确定  $n$  个  $m$  维基向量  $\mathbf{a}_1, \dots, \mathbf{a}_n \in \mathbb{R}^m$  以及一个稀疏的  $n$  维权向量或者系数向量  $\mathbf{s} \in \mathbb{R}^n$ ，使得部分基向量的加权线性组合可以充分逼近输入向量，即  $\mathbf{x} \approx \mathbf{As}$ ，其中  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n] \in \mathbb{R}^{m \times n}$ 。

如果给定的是  $m$  维实值输入向量的一组集合  $(\mathbf{x}_1, \dots, \mathbf{x}_k)$ ，则稀疏编码的目的就是确定基矩阵  $\mathbf{A}$  和系数矩阵  $\mathbf{S}$ ，使得  $\mathbf{X} \approx \mathbf{AS}$ ，其中  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_k] \in \mathbb{R}^{m \times k}$  和  $\mathbf{S} = [s_1, \dots, s_k] \in \mathbb{R}^{n \times k}$  分别表示输入矩阵和系数矩阵，并且系数矩阵的每一个列向量都是稀疏向量，即稀疏向量  $s_i$  是与输入向量  $\mathbf{x}_i$  对应的系数向量。

稀疏编码的主要特点是系数向量只有少数元素不等于零，大多数元素为零。因此，对应于某个输入向量，基矩阵中只有少数基向量被激活，这与新陈代谢的观点不谋而合：越少的神经元被激励时，所用的能量也就越少<sup>[336]</sup>。

稀疏编码的另一个主要特点取决于它本身是临界完备的，还是过完备的。如果基向量的个数  $n$  等于输入向量的维数  $m$ ，则编码就称临界完备的。临界完备编码 (critically complete coding) 的目标是求一可逆的加权矩阵  $\mathbf{A}$ ，利用它对输入进行变换，以满足对输出作用的某种优化准则，如解相关、稀疏性等。临界完备编码的典型例子有：JPEG 图像压缩中使用的离散余弦变换和正交小波变换等。在临界完备编码的情况下，输入向量的小变化都有可能导致系数的突然变化，从而使得编码对输入向量中的误差或者噪声敏感。为了克服这一缺点，需要使基向量的个数  $n$  大于输入向量的维数  $m$ ，即采用过完备编码 (overcomplete coding)。

采用具有过完备基集合的稀疏编码的主要理由有以下几个方面<sup>[380]</sup>：

- (1) 稀疏化 (sparsification) 会淘汰那些对描述给定的图像结构无用的大量基向量, 因为这些基向量与稀疏向量的零元素相乘, 而被完全淘汰。
- (2) 过完备码对噪声和其他形式的退化具有更大的数值稳定性。
- (3) 在使用生成模型对输入结构进行匹配时, 过完备编码比临界完备编码具有更大的灵活性。

#### 1.12.4 压缩感知的稀疏表示

由 Nyquist 采样定理知, 只有采样速率大于或者等于信号带宽的 2 倍时, 才能精确地重建或重构原始信号。然而, 对于超宽带通信和信号处理、计算机视觉、生物医学成像、遥感成像、传感器网络等众多应用, 信号的带宽越来越大, 从而对信号的采样速率、传输速度和存储空间的要求也越来越高。为了应对和缓解这些变化带来的挑战与压力, 通常的做法是先使用 Nyquist 速率采样, 再进行采样数据的压缩。问题是, 对于超宽带信号, Nyquist 速率采样成本太高, 而且大量被压缩掉的数据对信号而言是不重要或者冗余的信息。

稀疏信号是指在大多数采样时刻的取值等于零或者近似等于零, 只有少数采样时刻的取值明显不等于零的信号。许多自然信号在时域并不是稀疏信号, 但是在某个变换域是稀疏的。这些稀疏变换工具包括 Fourier 变换、短时 Fourier 变换、小波变换和 Gabor 变换等。例如, 窄带信号通过 Fourier 变换, 其频谱是稀疏的。又如, 语音信号在时域不是稀疏的, 但经过短时 Fourier 变换后, 在频域为稀疏的。在时域不是稀疏的, 在某个变换域为稀疏的信号常称为可压缩信号。

对于稀疏的或可压缩的信号, 既然传统方法采样得到的多数数据会被舍掉, 何不避免获取全部数据, 而直接采样需要保留的数据呢? 压缩 + 低速率采样构成了与 Nyquist 采样理论不同的一种采样新理论——压缩感知 (compressed sensing, CS)。

令  $x(t)$  是一连续时间信号, 理想情况下, 我们希望使用 Nyquist 速率采样, 得到  $n$  个离散时间信号的向量  $\mathbf{x} = [x(1), \dots, x(n)]^T \in \mathbb{R}^n$ 。然而, 实际上, 我们使用远低于 Nyquist 速率采样, 只得到一低维测量数据向量  $\mathbf{y} = [y(1), \dots, y(m)]^T \in \mathbb{R}^m$ , 其中

$$y(k) = \langle \phi_k, \mathbf{x} \rangle, \quad k \in M \quad (1.12.9)$$

式中  $M \subset \{1, \dots, n\}$  是一个基数 (cardinality)  $m \ll n$  的子集, 而  $\phi_k$  表示感知基  $\Phi \in \mathbb{R}^{n \times n}$  的第  $k$  列。上式又可写成向量形式

$$\mathbf{y} = \mathbf{A}\mathbf{x} \quad (1.12.10)$$

式中, 感知矩阵  $\mathbf{A} \in \mathbb{R}^{m \times n}$  的  $m$  个行向量由感知基 (sensing basis)  $\Phi$  的  $m$  个列向量的转置排列组成, 即  $\mathbf{A} = [\phi_1, \dots, \phi_m]^T$ , 其中  $1, \dots, m \in M$ 。

感知波形  $\mathbf{y}$  可以是时域或空域的采样向量; 若感知波形为像素的指标函数, 则  $\mathbf{y}$  是由数字摄像机的传感器采集的图像数据向量; 若感知波形为正弦波, 则  $\mathbf{y}$  是 Fourier 系数向量, 这正是核磁共振成像 (magnetic resonance imaging, MRI) 的感知模式。

然而, 即使感知波形  $\mathbf{y}$  和感知矩阵  $\mathbf{A}$  已知, 我们也无法通过求解矩阵方程  $\mathbf{y} = \mathbf{Ax}$ , 恢复或者重构高维信号向量  $\mathbf{x}$ , 这主要是因为  $m \ll n$ , 使得求解欠定的矩阵方程的  $n$  维解向量  $\mathbf{x}$  往往不实际, 何况使用 Nyquist 速率对超宽带信号采样的成本也很高。例如, 对某个 1G 带宽的超宽带视频信号, 至少得用 2G Hz 的 Nyquist 速率采样, 则  $n = 2 \times 10^9$ , 即整个解向量  $\mathbf{x}$  至少含有  $2 \times 10^9$  个元素。

实际的信号或者图像常常是用时域表示的, 并且在某个变换域(例如频域)是可压缩的, 而可压缩的信号或图像往往可以使用稀疏向量充分逼近。于是, 可以使用某个表示矩阵 (representation matrix)  $\Psi \in \mathbb{R}^{n \times n}$ , 将  $\mathbf{x}$  从时域变换到频域或者其他某个可压缩的变换域, 得到  $\mathbf{x}$  的稀疏表示

$$\mathbf{x} = \Psi \alpha \quad (1.12.11)$$

其中, 系数向量  $\alpha \in \mathbb{R}^n$  是  $\mathbf{x}$  的  $K$ -稀疏表示, 即  $\alpha$  只含  $K$  个非零元素。

综合式 (1.12.10) 和式 (1.12.11) 立即得

$$\mathbf{y} = \mathbf{A}\Psi\alpha = \mathbf{V}\alpha \quad (1.12.12)$$

式中  $\mathbf{V} = \mathbf{A}\Psi$  称为全息字典或库 (holographic dictionary), 因为它包含了感知和表示的全面信息。

现在的问题是: 在给定感知基  $\Phi \in \mathbb{R}^{n \times n}$  和表示基  $\Psi \in \mathbb{R}^{n \times n}$  的情况下, 能否通过求解欠定的矩阵方程式 (1.12.12), 由低维的感知波形  $\mathbf{y} \in \mathbb{R}^m$  精确地或高概率地重构出 Nyquist 速率采样的高维数据向量  $\mathbf{x} \in \mathbb{R}^n$ 。

用低维的采样数据向量恢复或重构 Nyquist 速率采样的高维数据向量, 称为压缩感知。压缩感知是一种采样新理论<sup>[140]</sup>, 又称压缩采样 (compressive sampling)。

图 1.12.1 画出了压缩感知的方框图。图中, 虚线所示的部分为虚拟部分, 是实际中不执行的操作。

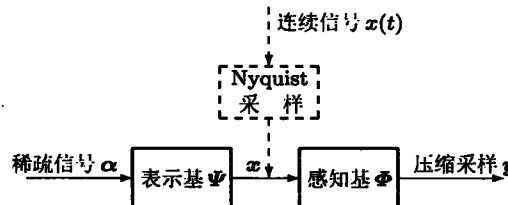


图 1.12.1 信号的压缩感知

表 1.12.1 比较了传统感知与压缩感知之间的不同及联系<sup>[529]</sup>。

压缩感知依赖于两个基本性质: 稀疏性(与感兴趣的信号有关)和非相干性(与传感或感知的方式有关)<sup>[85]</sup>:

(1) 稀疏性 (sparsity) 表达的思想是: 当使用合适的表示基  $\Psi$  作为信号表示时, 许多自然的信号和图像都是稀疏的或可压缩的。

表 1.12.1 传统感知与压缩感知的比较

方法	传统感知	压缩感知
采样速率	Nyquist 速率或更高速率	低速率
感知方式	感知后压缩 (数字式)	感知期间压缩 (物理方式)
感知量	大, 后期压缩丢弃很多数据	小, 因而可快速感知, 传感器少且低廉
压缩比	较好的压缩比, 压缩是自适应的	压缩是非自适应的
感知数据计算	简单	复杂, 涉及稀疏优化

(2) 非相干性 (incoherence) 的基本思想是: 当感知基  $\Phi$  与表示基  $\Psi$  不相干时, 与感兴趣的自然信号和图像不同, 采样或者感知的波形具有极为稠密的表示式 (1.12.12), 其中  $\alpha$  是一个  $K$ -稀疏的系数向量。

考虑  $m \times n$  测量矩阵  $A = [a_1, \dots, a_n]$ , 其列向量已经全部归一化, 即  $\|a_i\|_2 = 1, i = 1, \dots, n$ 。衡量一个矩阵质量的经典测度是矩阵列向量之间的相干 (coherence) [139, 207, 478], 定义为两个不同列向量之间的互相关的最大绝对值

$$\mu(A) = \max_{i \neq j} |\langle a_i, a_j \rangle| = \max_{i \neq j} |a_i^H a_j| \quad (1.12.13)$$

粗略地讲, 相干参数  $\mu$  可以度量两个列向量之间是如何相类似, 若相干参数大, 则至少有两个列向量彼此相类似。反之, 若相干参数  $\mu$  小, 则测量矩阵  $A$  的各个列是几乎相互正交的。

两个  $n \times n$  矩阵  $A$  和  $B$  之间的互相干参数 (mutual coherence parameter) 定义为 [138, 85]

$$\mu(A, B) = \sqrt{n} \max_{1 \leq j, k \leq n} |\langle a_j, b_k \rangle| \quad (1.12.14)$$

通俗地讲, 互相干度量  $A$  的列向量  $a_j$  和  $B$  的列向量  $b_k$  之间的最大相关。如果  $A$  和  $B$  含有相关的任何两个列向量, 则矩阵  $A$  和  $B$  之间的相干参数就大; 反之, 若两个矩阵之间的互相干很小, 则一个矩阵的所有列向量都与另一矩阵的各个列向量几乎相互正交。

一个  $m \times n$  感知基矩阵  $\Phi$  称为非相干的 (incoherent) [478, 479], 若

$$\max_{j \neq k} |\langle \phi_j, \phi_k \rangle| \leq \frac{1}{\sqrt{m}} \quad (1.12.15)$$

一个  $m \times n$  ( $m < n$ ) 宽矩阵  $\Phi$  称为紧致框架 (tight frame), 若

$$\Phi \Phi^T = \frac{n}{m} I_{m \times m} \quad (1.12.16)$$

满足  $M M^T = c^2 I_{m \times m}$  的所有  $m \times n$  ( $m < n$ ) 宽矩阵  $M$  的集合称为共形矩阵 (conformal matrices)。在所有的共形矩阵中, 紧致框架具有最小的谱范数。

由文献 [85] 知, 感知基矩阵  $\Phi \in \mathbb{R}^{n \times n}$  和表示基矩阵  $\Psi \in \mathbb{R}^{n \times n}$  之间的互相干  $\mu(\Phi, \Psi) \in [1, \sqrt{n}]$ 。因此, 称感知基矩阵  $\Phi \in \mathbb{R}^{n \times n}$  和表示基矩阵  $\Psi \in \mathbb{R}^{n \times n}$  非相干, 若

$$\max_{j \neq k} |\langle \phi_j, \psi_k \rangle| \leq 1 \quad (1.12.17)$$

**定理 1.12.1**<sup>[83]</sup> 令  $x \in \mathbb{R}^n$ , 感知矩阵  $A \in \mathbb{R}^{m \times n}$  由感知基  $\Phi$  的  $m$  个列向量转置组成; 并且  $x$  用表示基  $\Psi$  表示的系数向量  $\alpha$  是  $K$ -稀疏的。若

$$m \geq C\mu^2(\Phi, \Psi)K \log(n/\delta) \quad (1.12.18)$$

对某个正常数  $C$  成立, 则  $L_1$  范数最小化问题

$$\min \|\alpha\|_1 \quad \text{subject to } y = A\Psi\alpha \quad (1.12.19)$$

的解  $\alpha$  可以以  $1 - \delta$  的概率精确求出, 从而高维离散时间向量  $x$  可以从低维采样向量  $y$  以  $1 - \delta$  的概率重构。

从定理 1.12.1 可以得出以下结果:

- (1) 感知基  $\Phi$  和表示基  $\Psi$  之间的相干性越小, 所需要的测量样本数  $m$  就越少。
- (2) 虽然低速率只采样了  $m$  个数据, 它比用 Nyquist 速率采样的信号长度  $n$  少得多, 但是并不会造成任何信息的丢失, 因为信号可以高概率地精确恢复或者重构。如果  $\mu(\Phi, \Psi)$  等于或者接近 1, 则只要用  $K \log n$  数量级的  $m$  个测量数据即可。

综上所述, 压缩感知包含了以下两个关键步骤:

- (1) 通过非二次型凸优化问题

$$\min \|\alpha\|_0 \quad \text{subject to } y = V\alpha \quad (1.12.20)$$

的求解, 估计稀疏的系数向量  $\alpha$ 。

- (2) Nyquist 速率的采样数据向量通过  $x = A\alpha$  重构。

压缩感知的采样速率不再取决于信号的带宽, 而主要取决于稀疏性和非相干性(也称等距约束性)。压缩感知问题主要包括了以下三个问题:

- (1) 具有稀疏表示能力的过完备字典  $\Psi \in \mathbb{R}^{n \times n}$  的设计。
- (2) 满足与过完备字典  $\Psi$  非相干或等距约束性准则的感知矩阵  $A \in \mathbb{R}^{m \times n}$  或者感知基  $\Phi \in \mathbb{R}^{n \times n}$  的设计。
- (3) 非二次型凸优化问题式 (1.12.20) 的求解。

注意, 由于  $\alpha \in \mathbb{R}^n$  是一个  $K$ -稀疏向量, 不需要对其  $n - K$  个“0”元素进行存储, 也不对它们进行任何运算, 从而大大节省内存空间和计算时间。因此, 稀疏向量计算的复杂性和代价仅仅取决于稀疏向量的非零元素的个数。

## 本章小结

本章从线性方程组出发, 引出了向量和矩阵的概念, 并介绍了矩阵代数的基本知识:

- (1) 向量的范数、内积、线性相关性、正交性和相似度;
- (2) 矩阵的标量性能指标: 范数、二次型、行列式、特征值、秩和迹;
- (3) 向量子空间的基本概念;
- (4) 矩阵的逆矩阵、Moore-Penrose 逆矩阵以及线性方程组的求解;
- (5) 矩阵的特殊求和与乘积: 直和、直积、Hadamard 积与 Kronecker 积;
- (6) 向量化与矩阵化。

特别地, 还重点介绍了求逆矩阵、广义逆矩阵和 Moore-Penrose 逆矩阵的几种具体算法。

围绕向量和矩阵的一些重要概念和定义, 本章还重点介绍了向量的相似度在模式识别的应用(分类)以及信号的稀疏表示、稀疏编码与压缩感知。

## 习 题

1.1 令  $\mathbf{x} = [x_1, \dots, x_m]^T, \mathbf{y} = [y_1, \dots, y_n]^T, \mathbf{z} = [z_1, \dots, z_k]^T$  为复向量, 分别用矩阵形式表示外积  $\mathbf{x} \circ \mathbf{y}$  和  $\mathbf{x} \circ \mathbf{y} \circ \mathbf{z}$ 。

1.2 证明矩阵加法的结合律  $(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$  和矩阵乘法的右分配律  $(\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC}$ 。

1.3 令

$$\mathbf{X} = \begin{bmatrix} 6 & 0 & 0 \\ -3 & 4 & 0 \\ 0 & 5 & 1 \end{bmatrix} \quad \text{和} \quad \mathbf{Y} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -2 & 1 \\ 7 & 0 & -1 \end{bmatrix}$$

求  $\mathbf{X}^2, \mathbf{Y}^2, \mathbf{XY}, \mathbf{YX}$ , 并证明

$$(\mathbf{X} - \mathbf{Y})^2 = \mathbf{X}^2 + \mathbf{Y}^2 - \mathbf{XY} - \mathbf{YX} = \begin{bmatrix} 52 & -37 & -19 \\ -26 & 37 & 1 \\ -64 & 54 & 20 \end{bmatrix}$$

1.4 假定  $\mathbf{A}$  和  $\mathbf{B}$  具有相同的维数, 证明

$$(\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{B})^T = (\mathbf{A} + \mathbf{B})(\mathbf{A}^T + \mathbf{B}^T) = \mathbf{AA}^T + \mathbf{BB}^T + \mathbf{AB}^T + \mathbf{BA}^T$$

1.5 令  $\mathbf{A} = \begin{bmatrix} 0.4 & 0.6 \\ 0.2 & 0.8 \end{bmatrix}$ , 计算  $\mathbf{A}^2, \mathbf{A}^4$  和  $\mathbf{A}^5$ 。

1.6 已知线性方程组

$$\left\{ \begin{array}{l} 2y_1 - y_2 = x_1 \\ y_1 + 2y_2 = x_2, \\ -2y_1 + 3y_2 = x_3 \end{array} \right. \quad \left\{ \begin{array}{l} 3z_1 - z_2 = y_1 \\ 5z_1 + 2z_2 = y_2 \end{array} \right.$$

用  $z_1, z_2$  表示  $x_1, x_2, x_3$ 。

1.7 利用初等行变换, 将下列矩阵化简为简约阶梯型矩阵

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 & 8 & 4 \\ 0 & 0 & 0 & 1 & 4 & 9 & 7 \\ 0 & 3 & -11 & -3 & -8 & -15 & -32 \\ 0 & -2 & -8 & 1 & 6 & 13 & 21 \end{bmatrix}$$

1.8 利用初等行变换求解线性方程组

$$2x_1 - 4x_2 + 3x_3 - 4x_4 - 11x_5 = 28$$

$$-x_1 + 2x_2 - x_3 + 2x_4 + 5x_5 = -13$$

$$-3x_3 + 2x_4 + 5x_5 = -10$$

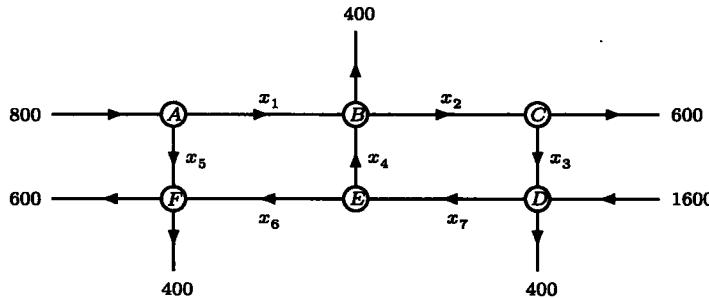
$$3x_1 - 5x_2 + 10x_3 - 7x_4 + 12x_5 = 31$$

1.9 假定

$$1^3 + 2^3 + \cdots + n^3 = a_1n + a_2n^2 + a_3n^3 + a_4n^4$$

试求常数  $a_1, a_2, a_3, a_4$ 。 (提示: 分别令  $n = 1, 2, 3, 4$ , 得到线性方程组。)

1.10 题图 1.10 画出了某城市 6 个交通枢纽的交通网络图<sup>[255]</sup>。其中, 节点表示交通枢纽的编号, 数字表示在交通高峰期每小时驶入和驶出某个交通枢纽的车辆数。

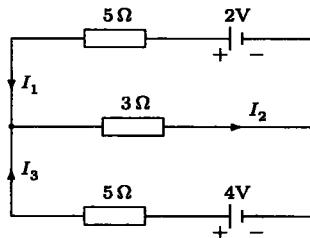


题图 1.10 交通网络图

(1) 写出表示交通网络图各个交通枢纽的交通流量的线性方程组, 并求解该方程组。

(2) 若  $x_6 = 300$  辆/小时,  $x_7 = 1300$  辆/小时, 求交通流量  $x_1 \sim x_5$ 。

1.11 题图 1.11 画出了一电路, 求各个支路的电流。



题图 1.11 电路图

1.12 证明自协方差矩阵和互协方差的下列性质:

- (1)  $\text{Var}(\mathbf{A}\mathbf{x} + \mathbf{b}) = \mathbf{A}\text{Var}(\mathbf{x})\mathbf{A}^H$ , 其中  $\mathbf{A}$  和  $\mathbf{b}$  分别为常数矩阵和常数向量。
- (2)  $\text{Cov}(\mathbf{x}, \mathbf{y}) = [\text{Cov}(\mathbf{y}, \mathbf{x})]^H$ 。
- (3)  $\text{Cov}(\mathbf{A}\mathbf{x}, \mathbf{B}\mathbf{y}) = \mathbf{A}\text{Cov}(\mathbf{x}, \mathbf{y})\mathbf{B}^H$ 。

1.13 令  $F: \mathbb{R}^3 \mapsto \mathbb{R}^2$  是一变换, 定义为

$$F(\mathbf{x}) = \begin{bmatrix} 2x_1 - x_2 \\ x_2 + 5x_3 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

试确定  $F$  是否为线性变换?

1.14 令  $\mathbf{A}$  是一个  $3 \times 4$  矩阵, 证明  $\mathbf{A}$  的列线性相关。

1.15 令  $\mathbf{A}$  是一个  $4 \times 3$  矩阵, 证明  $\mathbf{A}$  的行线性相关。

1.16 令  $V$  是一  $n$  维子空间, 且  $\mathbf{b}$  为任意  $n \times 1$  向量。证明: 存在  $n \times 1$  向量  $\mathbf{v}_0 \in V$ , 得

$$\|\mathbf{b} - \mathbf{v}_0\| \leq \|\mathbf{b} - \mathbf{v}\| \quad \forall \mathbf{v} \in V$$

(提示: 令  $\mathbf{e}_1, \dots, \mathbf{e}_n$  为子空间  $V$  的正交基, 其中,  $\mathbf{e}_i, i = 1, \dots, n$  是仅第  $i$  个元素为 1, 其他元素等于 0 的  $n \times 1$  向量。)

1.17 [306] 矩阵的秩在工程控制系统的设计中起着重要的作用。一个离散时间的控制系统的状态空间模型包括了差分方程

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \quad k = 0, 1, \dots$$

式中,  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times m}$ , 并且  $\mathbf{x}_k \in \mathbb{R}^n$  为描述系统在  $k$  时刻状态的向量, 简称状态向量; 而  $\mathbf{u}_k$  为系统在  $k$  时刻的输入或控制向量。矩阵对  $(\mathbf{A}, \mathbf{B})$  称为可控的, 若

$$\text{rank}([\mathbf{B}, \mathbf{AB}, \mathbf{A}^2\mathbf{B}, \dots, \mathbf{A}^{n-1}\mathbf{B}]) = n$$

若  $(\mathbf{A}, \mathbf{B})$  是可控的, 则最多用  $n$  步即可将系统控制到任意一个指定的状态  $\mathbf{x}$ 。试确定以下矩阵对是否可控:

$$(1) \mathbf{A} = \begin{bmatrix} 0.9 & 1 & 0 \\ 0 & -0.9 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

$$(2) \mathbf{A} = \begin{bmatrix} 0.8 & -0.3 & 0 \\ 0.2 & 0.5 & 1 \\ 0 & 0 & -0.5 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

1.18 令  $U$  和  $V$  是 Euclidean  $n$  空间  $\mathbb{R}^n$  的两个子空间, 并假定  $V$  是  $U$  的子集。证明:  $\dim(V) \leq \dim(U)$ 。若  $\dim(V) = \dim(U)$ , 证明  $U$  包含于  $V$ , 因此  $V = U$ 。

1.19 令正方矩阵  $\mathbf{A}$  和  $\mathbf{B}$  具有相同的维数, 证明  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ 。

1.20 证明  $\mathbf{x}^T \mathbf{A} \mathbf{x} = \text{tr}(\mathbf{x}^T \mathbf{A} \mathbf{x})$  和  $\mathbf{x}^T \mathbf{A} \mathbf{x} = \text{tr}(\mathbf{A} \mathbf{x} \mathbf{x}^T)$ 。

1.21 令  $A \in \mathbb{R}^{n \times n}$ , 证明 Schur 不等式  $\text{tr}(A^2) \leq \text{tr}(A^T A)$ , 其中等式成立, 当且仅当  $A$  是对称矩阵。

1.22 令  $A$  为  $n \times n$  矩阵, 证明

$$\frac{\partial |A - \lambda I|}{\partial \lambda} = - \sum_{k=1}^n |A_k - \lambda I|$$

式中,  $A_k$  是从矩阵  $A$  中删去第  $k$  行和第  $k$  列剩下的  $(n-1) \times (n-1)$  子矩阵。

1.23 满足  $|A - \lambda I| = 0$  的根称为矩阵  $A$  的特征根。证明: 若  $\lambda$  是矩阵  $A$  的一个单特征根, 则至少有一个行列式  $|A_k - \lambda I| = 0$ 。

1.24 直线方程可以表示为  $ax + by = -1$ 。证明一条通过点  $(x_1, y_1)$  和  $(x_2, y_2)$  的直线方程为

$$\begin{vmatrix} 1 & x & y \\ 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \end{vmatrix} = 0$$

1.25 平面方程可以表示为  $ax + by + cz = -1$ 。证明: 通过三点  $(x_i, y_i, z_i)$ ,  $i = 1, 2, 3$  的平面方程由下式决定

$$\begin{vmatrix} 1 & x & y & z \\ 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \end{vmatrix} = 0$$

1.26 在不展开行列式的情况下, 证明下列结果

$$2 \begin{vmatrix} a & b & c \\ d & e & f \\ x & y & z \end{vmatrix} = \begin{vmatrix} a+b & b+c & c+a \\ d+e & e+f & f+d \\ x+y & y+z & z+x \end{vmatrix}$$

和

$$2 \begin{vmatrix} 0 & a & b \\ a & 0 & c \\ b & c & 0 \end{vmatrix} = \begin{vmatrix} b+a & c & c \\ b & a+c & b \\ a & a & c+b \end{vmatrix}$$

1.27 令  $A_{n \times n}$  正定, 并且  $B_{n \times n}$  半正定, 证明  $\det(A + B) \geq \det(A)$ 。

1.28 令  $A_{n \times n}$  和  $B_{n \times n}$  都是半正定矩阵, 证明  $\det(A + B) \geq \det(A) + \det(B)$ 。

1.29 令  $A_{12 \times 12}$  满足  $A^5 = 3A$ 。试求  $|A|$  的所有可能的数值。

1.30 已知  $X = [A, B]$  是一个分块矩阵, 证明

$$|X|^2 = |AA^T + BB^T| = \begin{vmatrix} A^T A & A^T B \\ B^T A & B^T B \end{vmatrix}$$

1.31<sup>[444, p.361]</sup> 已知  $n \times n$  矩阵  $M = I - X(X^T X)^{-1} X^T$ 。若矩阵  $X$  的秩为  $r_X$ , 且  $y$  是一个正态分布的随机向量, 即  $y \sim N(Xb, \sigma^2 I)$ , 证明

(1)  $E\{y^T M y\} = (n - r_X)\sigma^2$ 。

(2)  $y^T M y$  和  $y^T (I - M) y$  是统计独立的随机变量。

(3)  $y^T M y / \sigma^2$  服从自由度为  $(n - r_X)$  的  $\chi^2$  分布, 即  $y^T M y / \sigma^2 \sim \chi^2_{n-r_X}$ 。

(4) 当  $\mathbf{X}\mathbf{b} = \mathbf{0}$  时,  $\mathbf{y}^T(\mathbf{I} - \mathbf{M})\mathbf{y}/\sigma^2 \sim \chi_{r_X}^2$ 。

**1.32** 令  $\mathbf{A}^2 = \mathbf{A}$ , 用下面两种方法证明  $\text{rank}(\mathbf{I} - \mathbf{A}) = n - \text{rank}(\mathbf{A})$ : (1) 利用矩阵  $\mathbf{A}$  的迹与秩相等的性质; (2) 考虑线性方程组  $(\mathbf{I} - \mathbf{A})\mathbf{x} = \mathbf{0}$  的线性无关解。

**1.33** 已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 2 \\ -1 & 0 & 2 & -3 \\ 2 & 4 & 8 & 5 \end{bmatrix}$$

求矩阵  $\mathbf{A}$  的秩和零维。(提示: 将矩阵  $\mathbf{A}$  化为阶梯型。)

**1.34** 令  $\mathbf{A}$  是一个  $m \times n$  矩阵, 证明  $\text{rank}(\mathbf{A}) \leq m$  和  $\text{rank}(\mathbf{A}) \leq n$ 。

**1.35** 考虑线性方程组

$$x_1 + 3x_2 - x_3 = a_1$$

$$x_1 + 2x_2 = a_2$$

$$3x_1 + 7x_2 - x_3 = a_3$$

(1) 确定线性方程组为一致方程的充分必要条件。

(2) 假定三种情况:

①  $a_1 = 2, a_2 = 2, a_3 = 6$ ;

②  $a_1 = 1, a_2 = 0, a_3 = -2$ ;

③  $a_1 = 0, a_2 = 1, a_3 = 2$ .

判断线性方程组是否为一致方程。若是一致方程, 则给出相对应的解。

**1.36** 令  $\mathbf{A}$  是一个  $3 \times 4$  矩阵, 其零维等于 1。证明对  $3 \times 1$  实向量  $\mathbf{b}$  的每一种选择,  $3 \times 4$  线性方程  $\mathbf{Ax} = \mathbf{b}$  均是一致方程。

**1.37** 当  $\alpha$  取何值时, 线性方程组

$$(\alpha + 3)x_1 + x_2 + 2x_3 = \alpha$$

$$3(\alpha + 1)x_1 + \alpha x_2 + (\alpha + 3)x_3 = 3$$

$$\alpha x_1 + (\alpha - 1)x_2 + x_3 = \alpha$$

有唯一解、无解和无穷多解。当方程组有无穷多解时, 求出它的通解。

**1.38** 当  $\alpha$  和  $\beta$  取何值时, 线性方程组

$$x_1 + 3x_2 + 6x_3 + x_4 = 3$$

$$x_1 + x_2 + 2x_3 + 3x_4 = 1$$

$$x_1 - 5x_2 - 10x_3 + 12x_4 = \alpha$$

$$3x_1 - x_2 - \beta x_3 + 15x_4 = 3$$

有唯一解、无解和无穷多解。当方程组有无穷多解时, 求出它的通解。

1.39 [444] 已知矩阵方程  $Ax = b$  为

$$x_1 + 2x_2 + 3x_3 = 26$$

$$3x_1 + 7x_2 + 10x_3 = 87$$

$$2x_1 + 11x_2 + 7x_3 = 73$$

(1) 利用高斯消去法求解方程。

(2) 将矩阵  $A$  的第  $j$  列用  $b$  代替，并记所得矩阵为  $A_j$ 。证明 (1) 中求出的方程的解可以表示为

$$x_j = |A_j|/|A|, \quad j = 1, 2, 3$$

这一方法称为求解线性方程的 Cramer 法则。

(3) 证明对  $A_{n \times n}x_{n \times 1} = b_{n \times 1}$  的一般情况，若  $|A| \neq 0$ ，则由 Cramer 法则求出的解  $x = [x_1, x_2, \dots, x_n]^T$  确实满足线性方程  $Ax = b$ 。

1.40 假定  $A$  和  $B$  都是  $n \times n$  矩阵，并且  $A$  非奇异。从线性方程组的角度证明：若  $AB = O$  (零矩阵)，则  $B = O$ 。

1.41 设向量组

$$\mathbf{a}_1 = [1, 1, 1, 3]^T, \quad \mathbf{a}_2 = [-1, -3, 5, 1]^T$$

$$\mathbf{a}_3 = [3, 2, -1, p+2]^T, \quad \mathbf{a}_4 = [-2, -6, 10, p]^T$$

(1)  $p$  为何值时，此向量组线性无关？用  $\mathbf{a}_1 \sim \mathbf{a}_4$  的线性组合表示  $\mathbf{a} = [4, 1, 6, 10]^T$ 。

(2)  $p$  取何值时，该向量组线性相关？求出此时矩阵  $[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4]$  的秩和一个极大线性无关的向量组。

1.42 已知向量组

$$\mathbf{a}_1 = \begin{bmatrix} a \\ 2 \\ 10 \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} -2 \\ 1 \\ 5 \end{bmatrix}, \quad \mathbf{a}_3 = \begin{bmatrix} -1 \\ 1 \\ 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ b \\ c \end{bmatrix}$$

试分别求出满足以下条件的  $a, b, c$  值：

(1)  $\mathbf{b}$  可由  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  线性表示，且唯一。

(2)  $\mathbf{b}$  不能由  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  线性表示。

(3)  $\mathbf{b}$  可由  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  线性表示，但表示不唯一。并求出一般表达式。

1.43 令矩阵  $A_{m \times n}$  的秩  $r = \text{rank}(A)$ 。证明：若  $A$  分块为

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

式中， $r \times r$  矩阵  $A_{11}$  是一个秩为  $r$  的非奇异矩阵，则  $A_{22} = A_{21}A_{11}^{-1}A_{12}$ 。（提示：注意  $[A_{21}, A_{22}]$  的行是  $[A_{11}, A_{12}]$  的  $r$  行的线性组合，即有  $[A_{21}, A_{22}] = F[A_{11}, A_{12}]$ 。类似地，矩阵  $A$  的右边  $n-r$  列是左边  $r$  列的线性组合。）

1.44 令两个向量相互正交, 证明它们线性无关。

1.45 矩阵  $A^2 = A$  和  $B^2 = B$ , 并且  $B$  的列是  $A$  的列的线性组合。证明  $AB = B$ 。

1.46 证明

$$A \text{ 非奇异} \Leftrightarrow \det \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \det(A) \det(D - CA^{-1}B)$$

1.47 证明  $\text{rank}(A + B) \leq \text{rank}[A, B] \leq \text{rank}(A) + \text{rank}(B)$ 。

1.48 证明  $\text{rank}[A, B] \leq \text{rank}(A) + \text{rank}(B)$ 。

1.49 验证向量组

$$A = \left\{ \begin{bmatrix} 1 \\ 0 \\ 1 \\ 2 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ -2 \\ 1 \\ 0 \end{bmatrix} \right\}$$

是一组正交向量。

1.50 令  $B = \{v_1, v_2, v_3\}$  是 Euclidean 空间  $\mathbb{R}^3$  的一组正交基。给定向量  $u \in \mathbb{R}^3$ , 试确定常数  $a_1, a_2, a_3$  使得  $u = a_1v_1 + a_2v_2 + a_3v_3$ 。

1.51 证明满足  $(I - A)(I + A) = O$  的矩阵  $A$  为对合矩阵。

1.52 设  $A_{n \times n}$  为对合矩阵, 证明  $B = \frac{1}{2}(I + A)$  为幂等矩阵。

1.53 令  $A$  是一个幂等矩阵, 证明: 其所有特征值取 1 或者 0。

1.54 若  $A$  是一个幂等矩阵, 证明:  $A^H, IA$  和  $I - A^H$  均为幂等矩阵。

1.55 使用 Gram-Schmidt 正交化方法构造子空间  $V = \text{Span}\{v_1, v_2, v_3\}$  的正交基和标准正交基, 其中

$$v_1 = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 1 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 0 \\ 3 \\ 1 \\ 1 \end{bmatrix}, \quad v_3 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

1.56 已知  $3 \times 5$  矩阵

$$A = \begin{bmatrix} 1 & 3 & 2 & 5 & 7 \\ 2 & 1 & 0 & 6 & 1 \\ 1 & 1 & 2 & 5 & 4 \end{bmatrix}$$

用 MATLAB 函数 `orth(A)` 和 `null(A)` 分别求矩阵  $A$  的列空间  $\text{Span}(A)$  的正交基和零空间  $\text{Null}(A)$ 。

1.57 令  $x$  和  $y$  是 Euclidean  $n$  空间  $\mathbb{R}^n$  的任意两个向量, 证明 Cauchy-Schwartz 不等式  $|x^T y| \leq \|x\| \|y\|$ 。 (提示: 观察  $\|x - cy\|^2 \geq 0$  对所有标量  $c$  成立。)

1.58 令  $x$  和  $y$  是 Euclidean  $n$  空间  $\mathbb{R}^n$  的任意两个向量, 证明三角不等式  $\|x + y\| \leq \|x\| + \|y\|$ 。 (提示: 展开  $\|x + y\|^2$ , 并利用 Cauchy-Schwartz 不等式。)

1.59 令  $B = \{v_1, v_2, \dots, v_n\}$  是子空间  $W$  的标准正交基, 且  $u$  是子空间  $W$  内的向量。证明: 若  $u = a_1v_1 + a_2v_2 + \dots + a_nv_n$ , 则

$$\|u\|^2 = |a_1|^2 + |a_2|^2 + \dots + |a_n|^2$$

**1.60** 证明任意一组由  $\mathbb{R}^3$  中的 4 个或更多个向量的集合不可能组成  $\mathbb{R}^3$  的正交基。

**1.61** 定义变换  $H : \mathbb{R}^2 \mapsto \mathbb{R}^2$  为

$$H(\mathbf{x}) = \begin{bmatrix} x_1 + x_2 - 1 \\ 3x_1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

判断  $H$  是否为线性变换。

**1.62** 令

$$\mathbf{P} = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.8 & 0.3 \\ 0.2 & 0 & 0.4 \end{bmatrix}, \quad \mathbf{x}_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

假定一系统的状态向量可以用 Markov 链  $\mathbf{x}_{k+1} = \mathbf{P}\mathbf{x}_k$ ,  $k = 0, 1, \dots$  描述。试计算状态向量  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{15}$ , 分析系统随时间的变化。

**1.63** 已知矩阵函数

$$\mathbf{A}(x) = \begin{bmatrix} 2x & -1 & x & 2 \\ 4 & x & 1 & -1 \\ 3 & 2 & x & 5 \\ 1 & -2 & 3 & x \end{bmatrix}$$

求  $\frac{d^3|\mathbf{A}(x)|}{dx^3}$ 。(提示: 按任意行或列展开行列式  $|\mathbf{A}(x)|$ , 并且只需要关心  $x^4$  和  $x^3$  项。)

**1.64** 证明: 若  $\mathbf{A}_1$  非奇异, 则

$$\begin{vmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{vmatrix} = |\mathbf{A}_1| |\mathbf{A}_4 - \mathbf{A}_3 \mathbf{A}_1^{-1} \mathbf{A}_2|$$

**1.65** 求矩阵  $\mathbf{A}^T = \mathbf{A}$  和  $\mathbf{B}^T \neq \mathbf{B}$ , 使得下面的每一个二次型分别可以写成  $\mathbf{x}^T \mathbf{A} \mathbf{x}$  和  $\mathbf{x}^T \mathbf{B} \mathbf{x}$ :

- (1)  $7x_1^2 + 14x_1x_2 + 5x_2^2$ 。
- (2)  $(x_1 - 2x_2)^2 + (3x_2 - x_3)^2 + (6x_1 - 4x_3)^2$ 。
- (3)  $2(x_1^2 + x_2^2 + x_3^2) - 2(x_1x_2 + x_1x_3 + x_2x_3)$ 。
- (4)  $a_1x_1^2 + a_2x_2^2 + a_3x_3^2 + b_1x_1x_2 + b_2x_1x_3 + b_3x_2x_3$ 。

**1.66** 判断下列二次型的正定性:

- (1)  $f = -2x_1^2 - 8x_2^2 - 6x_3^2 + 2x_1x_2 + 2x_1x_3$ 。
- (2)  $f = x_1^2 + 4x_2^2 + 9x_3^2 + 15x_4^2 - 2x_1x_2 + 4x_1x_3 + 2x_1x_4 - 6x_2x_4 - 12x_3x_4$ 。

**1.67** 证明:

- (1) 若  $\mathbf{B}$  为实的非奇异矩阵, 则  $\mathbf{A} = \mathbf{B}\mathbf{B}^T$  正定。
- (2) 若  $|\mathbf{C}| \neq 0$ , 则  $\mathbf{A} = \mathbf{C}\mathbf{C}^H$  正定。

**1.68** 令  $\mathbf{A}$  和  $\mathbf{B}$  均为  $n \times n$  实数矩阵, 证明迹函数的 Cauchy-Schwartz 不等式:

(1)  $(\text{tr}(\mathbf{A}^T \mathbf{B}))^2 \leq \text{tr}(\mathbf{A}^T \mathbf{A}) \text{tr}(\mathbf{B}^T \mathbf{B})$ , 其中等号成立, 当且仅当  $\mathbf{A}$  和  $\mathbf{B}$  中之一是另一个的倍数;

(2)  $(\text{tr}(\mathbf{A}^T \mathbf{B}))^2 \leq \text{tr}(\mathbf{A}^T \mathbf{A} \mathbf{B}^T \mathbf{B})$ , 其中等号成立, 当且仅当  $\mathbf{AB}^T$  为对称矩阵;

(3)  $(\text{tr}(\mathbf{A}^T \mathbf{B}))^2 \leq \text{tr}(\mathbf{A} \mathbf{A}^T \mathbf{B} \mathbf{B}^T)$ , 其中等号成立, 当且仅当  $\mathbf{A}^T \mathbf{B}$  为对称矩阵。

1.69 证明  $\det(\mathbf{I} + \mathbf{uv}^T) = 1 + \mathbf{u}^T \mathbf{v}$ 。

1.70 证明  $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{BCA}) = \text{tr}(\mathbf{CAB})$ 。

1.71 令矩阵  $\mathbf{A}$  的特征值为  $\lambda_i$ , 证明  $\text{eig}(\mathbf{I} + c\mathbf{A}) = 1 + c\lambda_i$  和  $\text{eig}(\mathbf{A} - c\mathbf{I}) = \lambda_i - c$ 。

1.72 设  $n \times n$  ( $n \geq 3$ ) 矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & a & \cdots & a \\ a & 1 & \cdots & a \\ \vdots & \vdots & \ddots & \vdots \\ a & a & \cdots & 1 \end{bmatrix}$$

当  $a$  取何值时, 矩阵  $\mathbf{A}$  的秩为  $n - 1$ 。

1.73 已知  $\mathbf{AB} = \mathbf{BA} = \mathbf{O}$  (零矩阵) 和  $\text{rank}(\mathbf{A}^2) = \text{rank}(\mathbf{A})$ , 证明

(1)  $\text{rank}(\mathbf{A} + \mathbf{B}) = \text{rank}(\mathbf{A}) + \text{rank}(\mathbf{B})$ 。

(2)  $\text{rank}(\mathbf{A}^k + \mathbf{B}^k) = \text{rank}(\mathbf{A}^k) + \text{rank}(\mathbf{B}^k)$ , 其中  $k$  是某个整数。

1.74 令  $\mathbf{C}_{n \times n}$  是一任意对称矩阵, 证明: 存在满足  $\mathbf{AB} = \mathbf{O}$  的两个唯一非负定矩阵  $\mathbf{A}$  和  $\mathbf{B}$ , 使得  $\mathbf{C} = \mathbf{A} - \mathbf{B}$ 。

1.75 已知向量组  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p\}$  中  $\mathbf{x}_p \neq \mathbf{0}$ 。令  $a_1, a_2, \dots, a_{p-1}$  为任意常数, 并且

$$\mathbf{y}_i = \mathbf{x}_i + a_i \mathbf{x}_p, \quad i = 1, 2, \dots, p-1$$

证明向量组  $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{p-1}\}$  线性无关的充分条件是向量组  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p\}$  线性无关。

1.76 证明: 若  $\mathbf{A}^T \mathbf{A} = \mathbf{A}$ , 则  $\mathbf{A} = \mathbf{A}^T = \mathbf{A}^2$ 。

1.77 [444] 设

$$\mathbf{K} = \mathbf{K}^T = \mathbf{K}^3, \quad \mathbf{K}\mathbf{1} = \mathbf{0}, \quad \mathbf{K} \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix}$$

式中,  $\mathbf{1}$  是一个元素全部为 1 的向量。计算下列值, 并且说明为什么在无须计算  $\mathbf{K}$  的情况下, 可以得到下列值的原因:

(1)  $\mathbf{K}$  的阶数。

(2)  $\mathbf{K}$  的秩。

(3)  $\mathbf{K}$  的迹和行列式。

(4)  $\mathbf{K}^{26}$  的迹和行列式。

(5) 矩阵  $6\mathbf{K}^{60} - 7\mathbf{K}^{37} + 3\mathbf{I}$  的迹与行列式。

1.78 利用迹的性质, 证明不等式  $|\langle \mathbf{A}, \mathbf{B} \rangle|^2 \leq \|\mathbf{A}\|^2 \|\mathbf{B}\|^2$  的等号成立, 当且仅当  $\mathbf{A} = c\mathbf{B}$ , 其中  $c$  是一复常数。

1.79 假定下面提到的每个逆矩阵都存在, 证明以下结果:

- (1)  $(A^{-1} + I)^{-1} = A(A + I)^{-1}$ 。
- (2)  $(A^{-1} + B^{-1})^{-1} = A(A + B)^{-1}B = B(A + B)^{-1}A$ 。
- (3)  $(I + AB)^{-1}A = A(I + BA)^{-1}$ 。
- (4)  $(A + B)^{-1} = A^{-1}B^{-1}$  意味着  $A + ABA^{-1} = B + B^{-1}AB$ 。
- (5)  $A - A(A + B)^{-1}A = B - B(A + B)^{-1}B$ 。

**1.80** 验证分块矩阵求逆公式:

- (1) 矩阵  $A$  可逆时, 为

$$\begin{bmatrix} A & U \\ V & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}U(D - VA^{-1}U)^{-1}VA^{-1} & -A^{-1}U(D - VA^{-1}U)^{-1} \\ -(D - VA^{-1}U)^{-1}VA^{-1} & (D - VA^{-1}U)^{-1} \end{bmatrix}$$

- (2) 矩阵  $A$  和  $D$  可逆时, 为

$$\begin{bmatrix} A & U \\ V & D \end{bmatrix}^{-1} = \begin{bmatrix} (A - UD^{-1}V)^{-1} & -A^{-1}U(D - VA^{-1}U)^{-1} \\ -D^{-1}V(A - UD^{-1}V)^{-1} & (D - VA^{-1}U)^{-1} \end{bmatrix}$$

- (3) 矩阵  $A$  和  $D$  可逆时, 为

$$\begin{bmatrix} A & U \\ V & D \end{bmatrix}^{-1} = \begin{bmatrix} (A - UD^{-1}V)^{-1} & -(A - UD^{-1}V)^{-1}UD^{-1} \\ -(D - VA^{-1}U)^{-1}VA^{-1} & (D - VA^{-1}U)^{-1} \end{bmatrix}$$

**1.81** 证明逆矩阵的以下性质:

- (1) 逆矩阵的行列式等于原矩阵行列式的倒数, 即  $|A^{-1}| = \frac{1}{|A|}$ 。
- (2) 逆矩阵是非奇异的。
- (3)  $(A^{-1})^{-1} = A$ 。
- (4) 复共轭转置矩阵的逆矩阵  $(A^H)^{-1} = (A^{-1})^H = A^{-H}$ 。
- (5) 若  $A^H = A$ , 则  $(A^{-1})^H = A^{-1}$ 。
- (6)  $(A^*)^{-1} = (A^{-1})^*$ 。

**1.82** 证明: 逆矩阵的特征值等于原矩阵特征值的倒数, 即  $\text{eig}(A^{-1}) = 1/\text{eig}(A)$ 。

**1.83** 证明: 一个正方矩阵  $A$  可逆, 当且仅当  $AB = I$  对某个正方矩阵  $B$  成立。

**1.84** 用矩阵求逆公式  $A^{-1} = \frac{1}{\det(A)} \text{adj}(A)$  分别计算矩阵

$$A = \begin{bmatrix} 1 & 3 & 2 \\ 5 & 2 & 0 \\ 2 & -1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 4 & 1 & 3 \\ 0 & 2 & 0 \\ -4 & 1 & -4 \end{bmatrix}$$

的逆矩阵。(注:  $\text{adj}(A)$  表示矩阵  $A$  的伴随矩阵。)

**1.85** 若  $Y = (AX + B)(CX + D)^{-1}$ , 试用  $Y$  表示  $X$ 。

**1.86** 只满足条件  $AGA = A$  的矩阵  $G$  称为矩阵  $A$  的广义逆矩阵, 记作  $G = A^-$ 。设  $A, G$  和  $H$  分别是  $m \times n, n \times m$  和  $n \times p$  矩阵。证明: 若  $\text{rank}(A) = \text{rank}(AH)$ , 则  $GAH = H \implies G = A^-$ 。

1.87 令  $I$  为  $n \times n$  单位矩阵,  $J_{n \times n}$  是全部元素等于 1 的矩阵。若  $a + (n - 1)b = 0$ , 证明  $(a - b)^{-1}I$  是矩阵  $(a - b)I + bJ$  的满足定义  $AGA = A$  的广义逆矩阵。

1.88 一个对角矩阵  $H$  称为 Hermitian 标准型, 若它的对角线元素仅由 0 和 1 组成。对于任意一个正方矩阵  $A$ , 总是存在非奇异矩阵  $C$  使得  $CA = H$  为 Hermitian 标准型。证明  $C = A^-$  是矩阵  $A$  的广义逆矩阵。

1.89 令  $A_{m \times n}$  和  $R_{p \times n}$  是两个复矩阵, 并且  $N_{n \times q}$  是一满足  $RN = O$  的任意矩阵, 其秩为  $n - \text{rank}(R)$ , 记

$$\begin{aligned} D &= N(N^H A^H A N)^{-1} N^H A^H \\ E &= A^H - A^H A D \end{aligned}$$

证明:

(1) 方程  $\begin{bmatrix} A^H A & R^H \\ R & O \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} A^H \\ O \end{bmatrix}$  是一致方程。

(2) 若  $\begin{bmatrix} A^H A & R^H \\ R & O \end{bmatrix}^{-1} = \begin{bmatrix} C_1 & C_2^H \\ C_2 & C_3 \end{bmatrix}$ , 则

①  $C_2^H$  是  $R$  的广义逆矩阵。

②  $RC_1R^H = O$ 。

③  $AC_1A^H$  为幂等矩阵, 即  $(AC_1A^H)^2 = AC_1A^H$ 。

④  $C_1$  是矩阵  $A^H A$  的广义逆矩阵, 并且  $C_1A^H$  是矩阵  $A$  的广义逆矩阵<sup>[265]</sup>。

1.90 令

$$T = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

证明  $-(a_1^2 + a_2^2 + a_3^2)^{-1}T$  是  $T$  的广义逆矩阵  $T^-$ 。

1.91 利用矩阵的满秩分解, 求矩阵

$$A = \begin{bmatrix} 1 & 2 & 4 & 3 \\ 3 & -1 & 2 & -2 \\ 5 & -4 & 0 & -7 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 2 & -1 \\ 3 & 2 & 1 \\ -1 & -2 & -1 \\ 3 & 5 & 4 \end{bmatrix}$$

的广义逆矩阵  $A^-$  和  $B^-$ 。

1.92 设  $KA = O$  和  $K^2 = K$ , 且  $K$  非奇异, 证明矩阵  $A$  的广义逆矩阵  $A^- = (A - K)^{-1}$ 。

1.93 已知  $K^2 = K$ , 且  $Z^-$  是矩阵  $Z = KAK$  的广义逆矩阵。证明:  $KZ^-K$  也是  $Z$  的一个广义逆矩阵。

1.94 设  $G$  是  $A$  的一个广义逆矩阵。证明: 它也是矩阵  $AG$  的一个广义逆矩阵, 当且仅当  $G^2$  是  $A$  的一个广义逆矩阵。

1.95 满足 Moore-Penrose 逆矩阵两个条件  $AGA = A$  和  $GAG = G$  的矩阵  $G$  称为矩阵  $A$  的自反广义逆矩阵 (reflexive generalized inverse)。证明:  $\text{rank}(G) = \text{rank}(A)$  对  $AGA = A$  成立, 当且仅当  $G$  是  $A$  的一个自反广义逆矩阵。

1.96 令

$$G = Q \begin{bmatrix} I_r & U \\ V & VU \end{bmatrix} P$$

其中,  $I_r$  为单位矩阵,  $P$  和  $Q$  非奇异, 并且

$$PAQ = \begin{bmatrix} I_r & O \\ O & O \end{bmatrix}, \quad O \text{ 为零矩阵}$$

证明  $G$  是  $A$  的自反广义逆矩阵。

1.97 验证  $A^\dagger = (A^H A)^\dagger A^H$  和  $A^\dagger = A^H (A A^H)^\dagger$  分别满足 Moore-Penrose 逆矩阵的四个条件。

1.98 证明右伪逆矩阵  $F_m^\dagger = F_m^H (F_m F_m^H)^{-1}$  的递推公式

$$F_m^\dagger = \begin{bmatrix} F_{m-1}^\dagger - \Delta_m F_{m-1}^\dagger f_m c_m \\ \Delta_m c_m^H \end{bmatrix}$$

式中,  $c_m^H = f_m^H (I_n - F_{m-1} F_{m-1}^\dagger)$ ,  $\Delta_m = c_m^H f_m$ 。递推的初始值为  $F_1^\dagger = f_1^H / (f_1^H f_1)$ 。

1.99 证明:

(1) 所有左和右逆矩阵都是自反广义逆矩阵, 它们分别满足 Moore-Penrose 对称条件  $AGA = A$  和  $GAG = G$  之中的一个条件。

(2) 一个满行 (列) 秩矩阵  $A$  的所有广义逆矩阵  $A^\dagger$  都是右 (左) 逆矩阵。

1.100 求  $3 \times 1$  向量  $a = [1, 5, 7]^T$  的 Moore-Penrose 逆矩阵。

1.101 证明  $A(A^T A)^{-2} A^T$  是  $AA^T$  的 Moore-Penrose 逆矩阵。

1.102 证明关于 Moore-Penrose 逆矩阵的下列定义条件 (1) 和条件 (2) 等价:

(1)  $AGA = A$ ,  $GAG = G$ ,  $(AG)^\# = AG$ ,  $(GA)^\# = GA$ 。

(2)  $A^\# AG = A^\#$ ,  $G^\# GA = G^\#$ 。

1.103 设  $A$  是一对称矩阵, 并且  $M$  是  $A$  的 Moore-Penrose 逆矩阵。证明: 矩阵  $M^2$  是  $A^2$  的 Moore-Penrose 逆矩阵。

1.104 令  $A$  是一个幂等矩阵, 证明  $A = A^\dagger$ 。

1.105 已知矩阵

$$A = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 0 & 2 & 2 & 2 \\ -1 & 4 & 5 & 3 \end{bmatrix}$$

利用矩阵的满秩分解法, 求 Moore-Penrose 逆矩阵  $A^\dagger$ 。

1.106 分别利用递推法 (算法 1.8.3) 和迹方法 (算法 1.8.4) 求矩阵

$$X = \begin{bmatrix} 1 & 0 & -2 \\ 0 & 1 & -1 \\ -1 & 1 & 1 \\ 2 & -1 & 2 \end{bmatrix}$$

的 Moore-Penrose 逆矩阵  $\mathbf{X}^\dagger$ 。

1.107 考虑映射  $\mathbf{U}\mathbf{V} = \mathbf{W}$ , 其中  $\mathbf{U} \in \mathbb{C}^{m \times n}, \mathbf{V} \in \mathbb{C}^{n \times p}, \mathbf{W} \in \mathbb{C}^{m \times p}$ , 并且  $\mathbf{U}$  是一个秩亏缺矩阵。证明  $\mathbf{V} = \mathbf{U}^\dagger \mathbf{W}$ , 其中  $\mathbf{U}^\dagger \in \mathbb{C}^{n \times m}$  是  $\mathbf{U}$  的 Moore-Penrose 逆矩阵。

1.108 证明: 若  $\mathbf{A}\mathbf{x} = \mathbf{b}$  为一致方程, 则其通解为  $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} + (\mathbf{I} - \mathbf{A}^\dagger \mathbf{A})\mathbf{z}$ , 其中  $\mathbf{A}^\dagger$  是  $\mathbf{A}$  的 Moore-Penrose 逆矩阵, 并且  $\mathbf{z}$  为任意向量。

1.109 令矩阵  $\mathbf{A}$  和  $\mathbf{B}$  是使得矩阵乘积  $\mathbf{AB}$  存在, 并且  $\mathbf{B}_1 = \mathbf{A}^\dagger \mathbf{AB}$  和  $\mathbf{A}_1 = \mathbf{AB}_1 \mathbf{B}_1^\dagger$ , 证明 Cline<sup>[119]</sup> 建立的下列结果

$$\mathbf{AB} = \mathbf{A}_1 \mathbf{B}_1 \quad \text{和} \quad (\mathbf{AB})^\dagger = (\mathbf{A}_1 \mathbf{B}_1)^\dagger = \mathbf{B}_1^\dagger \mathbf{A}_1^\dagger$$

1.110 证明线性映射  $T : V \mapsto W$  的下列性质

$$T(\mathbf{0}) = \mathbf{0} \quad \text{和} \quad T(-\mathbf{x}) = -T(\mathbf{x})$$

1.111 令  $U$  是  $P_3$  空间的子空间, 定义为

$$U = \{p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 : a_3 = -2a_0 + 3a_1 + a_2\}$$

证明  $U$  与  $P_3$  同构。

1.112 已知  $P_{\text{ex}}(n) = \text{tr}(\mathbf{R}\Phi(n-1))$ , 其中  $\mathbf{R} = \mathbb{E}\{\mathbf{x}(n)\mathbf{x}^H(n)\}$  是  $M \times 1$  随机数据向量  $\mathbf{x}(n)$  的自相关矩阵, 而

$$\Phi(n) \approx \lambda^2 \Phi(n-1) + \sigma^2 \mathbb{E}\left\{\hat{\mathbf{R}}^{-1}(n)\mathbf{x}(n)\mathbf{x}^H(n)\hat{\mathbf{R}}^{-1}(n)\right\}$$

式中  $0 < \lambda < 1$ , 并且  $\hat{\mathbf{R}}(n) = \sum_{i=0}^n \lambda^{n-i} \mathbf{x}(i)\mathbf{x}^H(i)$  是真实自相关矩阵  $\mathbf{R}$  的样本估计。证明

$$P_{\text{ex}}(\infty) = \text{tr}(\mathbf{R}\Phi(\infty)) \approx \frac{1-\lambda}{1+\lambda} M \sigma^2$$

(提示: 求逆矩阵的数学期望  $\mathbb{E}\{\hat{\mathbf{R}}^{-1}\}$  的近似。)

1.113 证明: 对于任何  $m \times n$  矩阵  $\mathbf{A}$ , 其向量化函数

$$\text{vec}(\mathbf{A}) = (\mathbf{I}_n \otimes \mathbf{A})\text{vec}(\mathbf{I}_n) = (\mathbf{A}^T \otimes \mathbf{I}_n)\text{vec}(\mathbf{I}_m)$$

1.114 证明  $\mathbf{A} \otimes \mathbf{B}$  非奇异的充要条件是  $\mathbf{A}, \mathbf{B}$  非奇异。证明  $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$ 。

1.115 证明 Kronecker 积的 Moore-Penrose 逆矩阵的关系  $(\mathbf{A} \otimes \mathbf{B})^\dagger = \mathbf{A}^\dagger \otimes \mathbf{B}^\dagger$ 。

1.116 若  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  为相同维数的正方矩阵, 并且  $\mathbf{C}^T = \mathbf{C}$ , 证明

$$(\text{vec}(\mathbf{C}))^T (\mathbf{A} \otimes \mathbf{B}) \text{vec}(\mathbf{C}) = (\text{vec}(\mathbf{C}))^T (\mathbf{B} \otimes \mathbf{A}) \text{vec}(\mathbf{C})$$

1.117 令  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  和  $\mathbf{D}$  具有适当的维数, 使得  $\mathbf{ABCD}$  满足矩阵乘积定义。证明

$$\begin{aligned} \text{tr}(\mathbf{ABCD}) &= (\text{vec}(\mathbf{D}^T))^T (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B}) \\ &= (\text{vec}(\mathbf{D}))^T (\mathbf{A} \otimes \mathbf{C}^T) \text{vec}(\mathbf{B}^T) \end{aligned}$$

**1.118** 令  $A$  为  $m \times m$  对称矩阵,  $B$  为  $m \times n$  矩阵,  $C = AB$ , 且  $D = I_m - CC^\dagger$ 。证明:

$$(AC)^\dagger = C^\dagger A^\dagger [I_m - (DA^\dagger)^\dagger DA^\dagger]$$

**1.119** 令  $A, B \in \mathbb{R}^{m \times n}$ , 证明:

$$\text{tr}(A^T B) = (\text{vec}(A))^T \text{vec}(B) = \sum \text{vec}(A * B)$$

式中  $\sum \text{vec}(\cdot)$  表示对列向量函数的所有元素求和。

**1.120** 令  $x_i$  和  $x_j$  是矩阵  $X$  的列向量, 它们的协方差矩阵为  $\text{Cov}(x_i, x_j^T) = M_{ij}$ 。于是, 向量化函数  $\text{vec}(X)$  的方差-协方差矩阵  $\text{Var}(\text{vec}(X))$  是一分块矩阵, 其子矩阵为  $M_{ij}$ , 即有  $\text{Var}(\text{vec}(X)) = \{M_{ij}\}$ 。对于  $M_{ij} = m_{ij}V$  的特殊情况, 若  $m_{ij}$  是矩阵  $M$  的元素, 证明:

$$(1) \text{Var}(\text{vec}(X)) = M \otimes V.$$

$$(2) \text{Var}(\text{vec}(TX)) = M \otimes TVT^T.$$

$$(3) \text{Var}(\text{vec}(X^T)) = V \otimes M.$$

**1.121** 给定  $n \times n$  矩阵  $A$  和  $B$ 。(1) 令  $d = [d_1, \dots, d_n]^T$ , 并且  $D = \text{diag}(d_1, \dots, d_n)$ , 证明  $d^T(A * B)d = \text{tr}(ADB^T D)$ 。(2) 若  $A$  和  $B$  均为正定矩阵, 证明 Hadamard 积  $A * B$  是正定矩阵。这一性质称为 Hadamard 积的正定性。

**1.122** 证明: (1)  $\text{vec}(xy^T) = y \otimes x$ ; (2)  $\text{vec}(A \otimes b) = \text{vec}(A) \otimes b$ ; (3)  $\text{vec}(a_{p \times 1}^T \otimes B_{m \times n}) = (I_{pn} \otimes I_m)(a \otimes \text{vec}(B))$ 。

**1.123** 证明:  $a \otimes b = \text{vec}(ba^T)$ 。

**1.124** 证明:  $\text{vec}(PQ) = (Q^T \otimes P)\text{vec}(I) = (Q^T \otimes I)\text{vec}(P) = (I \otimes P)\text{vec}(Q)$ 。

**1.125** 验证:  $\text{tr}(XA) = (\text{vec}(A^T))^T \text{vec}(X) = (\text{vec}(X^T))^T \text{vec}(A)$ 。

**1.126** 令  $A$  是一  $m \times n$  矩阵, 而  $B$  是一  $p \times q$  矩阵, 证明:  $(A \otimes B)K_{nq} = K_{mp}(B \otimes A)$  和  $K_{pm}(A \otimes B)K_{nq} = B \otimes A$ 。

**1.127** 对任意  $m \times n$  矩阵  $A$  和  $p \times 1$  向量  $b$ , 证明:

$$(1) K_{pm}(A \otimes b) = b \otimes A$$

$$(2) K_{mp}(b \otimes A) = A \otimes b$$

$$(3) (A \otimes b)K_{np} = b^T \otimes A$$

$$(4) (b^T \otimes A)K_{pn} = A \otimes b$$

**1.128** 证明:  $m$  维单位矩阵与  $n$  维单位矩阵的 Kronecker 积给出  $mn$  维单位矩阵, 即有  $I_m \otimes I_n = I_{mn}$ 。

**1.129** 令  $X \in \mathbb{R}^{I \times JK}, G \in \mathbb{R}^{P \times QR}, A \in \mathbb{R}^{I \times P}, B \in \mathbb{R}^{J \times Q}, C \in \mathbb{R}^{K \times R}$ , 并且  $A^T A = I_P, B^T B = I_Q, C^T C = I_R$ , 证明: 若  $X = AG(C \otimes B)^T$ , 且  $X$  和  $A, B, C$  给定, 则矩阵  $G$  可以由  $G = A^T X (C \otimes B)$  恢复或重构。

1.130 利用向量化与 Kronecker 积的关系  $\text{vec}(UVW) = (W^T \otimes U)\text{vec}(V)$  求  $X = AG(C \otimes B)^T$  的向量化表示。

1.131 证明 Khatri-Rao 积与 Hadamard 积的下列关系式

$$(A \odot B) * (C \odot D) = (A * C) \odot (B * D)$$

$$(A \odot B \odot C)^T (A \odot B \odot C) = (A^T A) * (B^T B) * (C^T C)$$

$$(A \odot B)^\dagger = [(A^T A) * (B^T B)]^\dagger (A \odot B)^T$$

1.132 证明向量的内积与外积之间的下列关系

$$\langle x_1 \circ y_1, x_2 \circ y_2 \rangle = (x_1^T x_2)(y_1^T y_2)$$

其中  $x \circ y$  表示向量  $x$  与  $y$  的外积。

1.133 试将向量的  $L_0$  范数推广为  $m \times n$  矩阵的  $L_0$  范数。如何定义一个矩阵是  $K$ -稀疏的？

## 第2章 特殊矩阵

在实际应用中，经常会遇到元素之间存在某种特殊结构关系的矩阵，统称为特殊矩阵。了解这些矩阵的内部特殊结构，有助于灵活地使用这些矩阵，简化很多问题的表示和求解。本章将重点介绍一些比较常见的特殊矩阵。为了方便读者更深入地了解和应用这些特殊矩阵，将结合一些实际问题，对其中一些特殊矩阵加以解说。

### 2.1 Hermitian 矩阵

一个正方的复值矩阵  $A = [a_{ij}] \in \mathbb{C}^{n \times n}$  称为 Hermitian 矩阵，若  $A = A^H$ ，即其元素  $a_{ij} = a_{ji}^*$ 。换言之，Hermitian 矩阵是一种复共轭对称矩阵。

对一个实值矩阵，Hermitian 矩阵与对称矩阵等价。

Hermitian 矩阵又有以下几种特殊形式：

- (1) 矩阵  $A$  称为反 Hermitian 矩阵，若  $A = -A^H$ 。
- (2) 中央 Hermitian 矩阵  $R$  是一个元素满足对称性  $r_{ij} = r_{n-j+1, n-i+1}^*$  的  $n \times n$  正方矩阵。
- (3) 中央 Hermitian 矩阵的一个特殊子类是双重对称的矩阵，它既是关于主对角线对称的 Hermitian 矩阵，又是关于交叉对角线对称的交叉对称矩阵，例如

$$R = \begin{bmatrix} r_{11} & r_{21}^* & r_{31}^* & r_{41}^* \\ r_{21} & r_{22} & r_{32}^* & r_{31}^* \\ r_{31} & r_{32} & r_{22} & r_{21}^* \\ r_{41} & r_{31} & r_{21} & r_{11} \end{bmatrix}$$

在这种特殊情况下， $r_{ij} = r_{ji}^* = r_{n-j+1, n-i+1} = r_{n-i+1, n-j+1}^*$ 。

Hermitian 矩阵具有以下性质：

- (1)  $A$  是 Hermitian 矩阵，当且仅当  $x^H A x$  对所有复值向量  $x$  均是实数。
- (2) 对所有  $A \in \mathbb{C}^{n \times n}$ ，矩阵  $A + A^H$ ， $AA^H$  和  $A^H A$  均是 Hermitian 矩阵。
- (3) 若  $A$  是 Hermitian 矩阵，则  $A^k$  对所有  $k = 1, 2, 3, \dots$  都是 Hermitian 矩阵。若  $A$  还是非奇异的，则  $A^{-1}$  是 Hermitian 矩阵。
- (4) 若  $A$  和  $B$  是 Hermitian 矩阵，则  $\alpha A + \beta B$  对所有实数  $\alpha$  和  $\beta$  均是 Hermitian 矩阵。
- (5) 若  $A$  和  $B$  是反 Hermitian 矩阵，则  $\alpha A + \beta B$  对所有实数  $\alpha$  和  $\beta$  均是反 Hermitian 矩阵。

- (6) 对所有  $A \in \mathbb{C}^{n \times n}$ ,  $A - A^H$  为反 Hermitian 矩阵。
- (7) 若  $A$  是 Hermitian 矩阵, 则  $jA$  ( $j = \sqrt{-1}$ ) 是反 Hermitian 矩阵。
- (8) 若  $A$  是反 Hermitian 矩阵, 则  $jA$  是 Hermitian 矩阵。
- (9) 任何一个复值矩阵  $A$  都可以作唯一的分解  $A = B + jC$ , 其中  $B = \frac{1}{2}(A + A^H)$  和  $C = \frac{1}{2j}(A - A^H)$ 。
- (10) 若  $A$  和  $B$  均为 Hermitian 矩阵, 则  $AB + BA$  和  $j(AB - BA)$  也都是 Hermitian 矩阵。

Hermitian 矩阵的正定性判据: 一个  $n \times n$  Hermitian 矩阵  $A$  是正定的, 当且仅当它满足以下任何一个条件:

- (1) 二次型函数  $x^H Ax > 0$ ,  $\forall x \neq 0$ 。
  - (2) 矩阵  $A$  的所有特征值都大于零。
  - (3) 所有主子矩阵  $A_k$ ,  $1 \leq k \leq n$  都具有正的行列式, 其中,  $A_k = A(1:k, 1:k)$  由矩阵  $A$  的第  $1 \sim k$  行和第  $1 \sim k$  列组成。
  - (4) 存在一个非奇异的  $n \times n$  矩阵  $R$ , 使得  $A = R^H R$ 。
  - (5) 存在一个非奇异的  $n \times n$  矩阵  $P$ , 使得共轭对称矩阵  $P^H AP$  是正定的。
- 令  $z$  为一高斯随机向量。为方便计, 假定它具有零均值向量。此时, 高斯随机向量的协方差矩阵  $C_{zz}$  定义为随机向量  $z$  与它自身的外积的期望值, 即  $C_{zz} = E\{zz^H\}$ 。协方差矩阵总是正定的, 其证明如下: 首先, 以协方差矩阵为核的二次型可以写作  $x^H C_{zz} x = E\{|x^H z|^2\}$ 。由于向量  $x$  和  $z$  分别为常数向量和随机向量, 它们不可能正交, 即内积  $x^H z \neq 0$ 。于是, 二次型  $x^H C_{zz} x > 0$ , 即协方差矩阵  $C_{zz}$  是正定矩阵。

正定矩阵和半正定矩阵服从下面一些重要的不等式 [238, Sec.8.7]:

- (1) Hadamard 不等式 若  $m \times m$  矩阵  $A = [a_{ij}]$  正定, 则

$$\det(A) \leq \prod_{i=1}^m a_{ii}$$

当且仅当  $A$  是对角矩阵, 等式成立。

- (2) Fischer 不等式 令分块矩阵  $P = \begin{bmatrix} A & B \\ B^H & C \end{bmatrix}$  为正定矩阵, 其中, 分块矩阵  $A$  和  $C$  是正方的非零矩阵, 则

$$\det(P) \leq \det(A) \det(C)$$

- (3) Oppenheim 不等式 如果  $m \times m$  矩阵  $A$  和  $B$  是半正定矩阵, 则

$$\det(A) \prod_{i=1}^m b_{ii} \leq \det(A \odot B)$$

式中,  $A \odot B$  是矩阵  $A$  和  $B$  的 Hadamard 积。

(4) Minkowski 不等式 若  $m \times m$  矩阵  $\mathbf{A}, \mathbf{B}$  是正定矩阵, 则

$$\sqrt[m]{\det(\mathbf{A} + \mathbf{B})} \geq \sqrt[m]{\det(\mathbf{A})} + \sqrt[m]{\det(\mathbf{B})}$$

(5) Ostrowski-Taussky 定理 若  $H(\mathbf{A}_{m \times m}) = \frac{1}{2}(\mathbf{A} + \mathbf{A}^H)$  为正定矩阵, 则

$$\det H(\mathbf{A}) \leq |\det(\mathbf{A})|$$

等号成立, 当且仅当  $\mathbf{A}$  为自伴随矩阵, 即  $\mathbf{A}^\# = \mathbf{A}$ 。

## 2.2 置换矩阵、互换矩阵与选择矩阵

与单位矩阵密切相关的是置换矩阵、交换矩阵、互换矩阵和移位矩阵。这四种矩阵都只由 0 和 1 组成, 并且每行和每列都只有一个非零元素 1, 但非零元素 1 所处的位置不同。可以说, 单位矩阵、置换矩阵、交换矩阵、互换矩阵和移位矩阵都是由基本向量的不同排列而生成的。

### 2.2.1 置换矩阵与互换矩阵

**定义 2.2.1** 一个正方矩阵称为置换矩阵 (permutation matrix), 若它的每一行和每一列有一个且仅有一个非零元素 1。

置换矩阵  $\mathbf{P}$  有下列性质 [61]:

- (1)  $(\mathbf{P}_{m \times n})^T = \mathbf{P}_{n \times m}$ 。
- (2)  $\mathbf{P}^T \mathbf{P} = \mathbf{P} \mathbf{P}^T = \mathbf{I}$ , 这说明置换矩阵是正交矩阵。
- (3)  $\mathbf{P}^T = \mathbf{P}^{-1}$ 。
- (4)  $\mathbf{P}^T \mathbf{A} \mathbf{P}$  与  $\mathbf{A}$  具有相同的对角线元素, 但排列顺序可能不同。

**例 2.2.1** 给定一个  $5 \times 4$  矩阵

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{51} & a_{52} & a_{53} & a_{54} \end{bmatrix}$$

若令置换矩阵

$$\mathbf{P}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{P}_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

则有

$$\mathbf{P}_5 \mathbf{A} = \begin{bmatrix} a_{51} & a_{52} & a_{53} & a_{54} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{11} & a_{12} & a_{13} & a_{14} \end{bmatrix}, \quad \mathbf{A} \mathbf{P}_4 = \begin{bmatrix} a_{13} & a_{12} & a_{14} & a_{11} \\ a_{23} & a_{22} & a_{24} & a_{21} \\ a_{33} & a_{32} & a_{34} & a_{31} \\ a_{43} & a_{42} & a_{44} & a_{41} \\ a_{53} & a_{52} & a_{54} & a_{51} \end{bmatrix}$$

也就是说, 用置换矩阵左乘矩阵  $\mathbf{A}$ , 相当于将  $\mathbf{A}$  的行进行重新排列; 而用置换矩阵右乘  $\mathbf{A}$ , 相当于对  $\mathbf{A}$  的列进行重新排列。行或者列新的排列顺序由置换矩阵的结构所决定。

$p \times q$  置换矩阵可以是  $q$  个  $p \times 1$  基本向量  $e_1, e_2, \dots, e_q$  的随意排列。如果有规则的排列, 置换矩阵便演变为几种特殊形式的置换矩阵, 它们在矩阵分析与应用中经常被使用。

显然, 单位矩阵就是一个特殊的置换矩阵。置换矩阵还有另外三个特殊形式: 交换矩阵、互换矩阵与移位矩阵。

### 1. 交换矩阵

如第 1 章所述, 交换矩阵  $\mathbf{K}_{mn}$  定义为满足  $\mathbf{K}_{mn} \text{vec}(\mathbf{A}_{m \times n}) = \text{vec}(\mathbf{A}^T)$  的特殊置换矩阵。这种矩阵的作用是交换  $mn \times 1$  向量的元素位置, 以使得变换后的向量  $\mathbf{K}_{mn} \text{vec}(\mathbf{A})$  与  $\text{vec}(\mathbf{A}^T)$  相等, 故称交换矩阵 (commutation matrix)。

### 2. 互换矩阵

互换矩阵 (exchange matrix) 常用符号  $\mathbf{J}$  表示, 定义为

$$\mathbf{J} = \begin{bmatrix} 0 & & & 1 \\ & \ddots & & \\ & & 1 & \\ 1 & & & 0 \end{bmatrix} \quad (2.2.1)$$

它仅在交叉对角线上具有元素 1, 而所有其他元素全等于零。互换矩阵又称反射矩阵 (reflection matrix) 或后向单位矩阵 (backward identity matrix), 因为互换矩阵可以看作基本向量的反向排列  $[e_n, e_{n-1}, \dots, e_1]$ 。

通过左乘和右乘, 互换矩阵  $\mathbf{J}$  可以将一矩阵的行或列的顺序反转 (互换)。这就是术语“互换矩阵”的含义。具体说来, 用  $m \times m$  互换矩阵  $\mathbf{J}_m$  左乘  $m \times n$  矩阵  $\mathbf{A}$ , 将使  $\mathbf{A}$  的行的顺序反转 (相对于中心水平轴互换行的位置)

$$\mathbf{J}_m \mathbf{A} = \begin{bmatrix} a_{m1} & a_{m2} & \cdots & a_{mn} \\ \vdots & \vdots & \vdots & \vdots \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{11} & a_{12} & \cdots & a_{1n} \end{bmatrix} \quad (2.2.2)$$

用  $\mathbf{J}_n$  右乘  $m \times n$  矩阵  $\mathbf{A}$ , 则使  $\mathbf{A}$  的列序反转 (相对于中心垂直轴相互交换列的位置),

即有

$$\mathbf{A} \mathbf{J}_n = \begin{bmatrix} a_{1n} & \cdots & a_{12} & a_{11} \\ a_{2n} & \cdots & a_{22} & a_{21} \\ \vdots & \vdots & \vdots & \vdots \\ a_{mn} & \cdots & a_{m2} & a_{m1} \end{bmatrix} \quad (2.2.3)$$

容易验证

$$\mathbf{J}^2 = \mathbf{J}\mathbf{J} = \mathbf{I}, \quad \mathbf{J}^T = \mathbf{J} \quad (2.2.4)$$

前一个性质称为互换矩阵的对合性 (involuntary property)，后一个性质是互换矩阵的对称性。也就是说，互换矩阵是对合矩阵和对称矩阵。

第1章已介绍过，交换矩阵具有性质  $\mathbf{K}_{mn}^T = \mathbf{K}_{mn}^{-1} = \mathbf{K}_{nm}$ 。显然，若  $m = n$ ，则有  $\mathbf{K}_{nn}^T = \mathbf{K}_{nn}^{-1} = \mathbf{K}_{nn}$ 。这意味着

$$\mathbf{K}_{nn}^2 = \mathbf{K}_{nn} \mathbf{K}_{nn} = \mathbf{I}_{nn} \quad (2.2.5)$$

$$\mathbf{K}_{nn}^T = \mathbf{K}_{nn} \quad (2.2.6)$$

即交换矩阵和互换矩阵一样，也同时是对合矩阵和对称矩阵。

MATLAB 函数 `flipud(A)` 和 `fliplr(A)` 分别将矩阵  $\mathbf{A}$  的行和列的顺序翻转，即  $\mathbf{JA} = \text{flipud}(\mathbf{A})$  和  $\mathbf{AJ} = \text{fliplr}(\mathbf{A})$ 。

特别地，当用  $m \times m$  互换矩阵  $\mathbf{J}_m$  左乘矩阵  $m \times n$  矩阵  $\mathbf{A}$ ，然后再用  $n \times n$  互换矩阵  $\mathbf{J}_n$  右乘  $m \times n$  矩阵  $\mathbf{J}_m \mathbf{A}$  时，则有

$$\mathbf{J}_m \mathbf{A} \mathbf{J}_n = \begin{bmatrix} a_{m,n} & a_{m,n-1} & \cdots & a_{m,1} \\ a_{m-1,n} & a_{m-1,n-1} & \cdots & a_{m-1,1} \\ \vdots & \vdots & \vdots & \vdots \\ a_{1,n} & a_{1,n-1} & \cdots & a_{1,1} \end{bmatrix} \quad (2.2.7)$$

在维数清楚时，将省去  $\mathbf{J}$  矩阵的维数下标。与矩阵相类似， $\mathbf{Jc}$  将使列向量  $\mathbf{c}$  的元素顺序反转，而  $\mathbf{c}^T \mathbf{J}$  使行向量  $\mathbf{c}^T$  的元素顺序反转。

容易证明，若  $\mathbf{R}$  是交叉对称矩阵，则  $\mathbf{R}^T = \mathbf{JRJ}$  和  $\mathbf{R} = \mathbf{JR}^T \mathbf{J}$ 。

另外，中央对称矩阵  $\mathbf{R} = \mathbf{JRJ}$ ，而中央 Hermitian 矩阵  $\mathbf{R} = \mathbf{JR}^* \mathbf{J}$ 。

当中央对称矩阵的维数是偶数 ( $n = 2r$ ) 时，它可以分块为下列形式

$$\mathbf{R}_{\text{even}} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{JB}^* \mathbf{J} & \mathbf{JA}^* \mathbf{J} \end{bmatrix}$$

其中， $\mathbf{A}$  和  $\mathbf{B}$  是无特殊结构的一般  $r \times r$  矩阵。类似地，当维数是奇数 ( $n = 2r + 1$ ) 时，中央对称矩阵可以分块为

$$\mathbf{R}_{\text{odd}} = \begin{bmatrix} \mathbf{A} & \mathbf{x} & \mathbf{B} \\ \mathbf{x}^H & \alpha & \mathbf{x}^H \mathbf{J} \\ \mathbf{JB}^* \mathbf{J} & \mathbf{Jx} & \mathbf{JA}^* \mathbf{J} \end{bmatrix}$$

式中， $\mathbf{A}$  和  $\mathbf{B}$  是一般的  $r \times r$  矩阵， $\mathbf{J}$  是  $r \times r$  反射矩阵， $\mathbf{x}$  是  $r \times 1$  向量，而  $\alpha$  为标量。

### 3. 移位矩阵

$n \times n$  移位矩阵 (shift matrix) 定义为

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (2.2.8)$$

换言之, 移位矩阵的元素  $p_{i,i+1} = 1 (1 \leq i \leq n-1)$ ,  $p_{n1} = 1$ , 其余皆为零。显然, 移位矩阵可以用基本向量表示为  $\mathbf{P} = [\mathbf{e}_n, \mathbf{e}_1, \dots, \mathbf{e}_{n-1}]$ 。

移位矩阵乃是因其能够使别的矩阵的首行或者最后一列移动位置而得名。例如, 对一个  $m \times n$  矩阵  $\mathbf{A}$ , 若左乘  $m \times m$  移位矩阵  $\mathbf{P}_m$ , 则

$$\mathbf{P}_m \mathbf{A} = \begin{bmatrix} a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \\ a_{11} & a_{12} & \cdots & a_{1n} \end{bmatrix}$$

相当于将矩阵  $\mathbf{A}$  的第 1 行移位到第  $m$  行下面。类似地, 若右乘  $n \times n$  移位矩阵  $\mathbf{P}_n$ , 则

$$\mathbf{A} \mathbf{P}_n = \begin{bmatrix} a_{1n} & a_{11} & \cdots & a_{1,n-1} \\ a_{2n} & a_{21} & \cdots & a_{2,n-1} \\ \vdots & \vdots & \vdots & \vdots \\ a_{mn} & a_{m1} & \cdots & a_{m,n-1} \end{bmatrix}$$

相当于将矩阵  $\mathbf{A}$  的第  $n$  列移位到第 1 列前面。

容易看出, 与正方的互换矩阵  $\mathbf{J}_n$  和交换矩阵  $\mathbf{K}_{nn}$  不同, 移位矩阵既不具有对合性, 也不是对称矩阵。

### 2.2.2 广义置换矩阵与选择矩阵

考虑下面的观测数据模型

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) = \sum_{i=1}^n a_i s_i(t) \quad (2.2.9)$$

式中,  $\mathbf{s}(t) = [s_1(t), \dots, s_n(t)]^T$  表示源信号向量;  $\mathbf{A}$  是一个  $m \times n$  常系数矩阵 ( $m \geq n$ ), 表示信号的混合过程, 称为混合矩阵。混合矩阵是满列秩的。现在的问题是: 如何仅根据  $m$  维观测数据向量  $\mathbf{x}(t)$  恢复  $n$  维源信号向量  $\mathbf{s}(t)$ 。这个问题称为盲信号分离。这里, 术语“盲”具有两层含义: 源信号  $s_1(t), \dots, s_n(t)$  不可观测, 信号如何混合未知 (即混合矩阵  $\mathbf{A}$  未知)。

盲信号分离问题的核心是混合矩阵  $\mathbf{A}$  的广义逆矩阵  $\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$  的辨识, 因为源信号向量很容易利用  $\mathbf{s}(t) = \mathbf{A}^\dagger \mathbf{x}(t)$  进行恢复。然而, 混合矩阵的辨识存在两种不确定性或模糊性。

(1) 观察知, 若源信号向量中第  $i$  个和第  $j$  个信号交换顺序, 并且混合矩阵  $\mathbf{A}$  的第  $i$  列和第  $j$  列也交换位置的话, 则观测数据向量不变。这说明, 仅根据观测数据向量, 是不可能辨识源信号的排列顺序的。这种模糊性称为分离信号的排序不确定性。

(2) 由观测数据模型易知

$$\mathbf{x}(t) = \sum_{i=1}^n \frac{\alpha_i}{\alpha_i} \alpha_i s_i(t)$$

这表明, 仅根据观测数据向量, 也不可能辨识源信号  $s_i(t)$  的精确幅值。这种模糊性称为分离信号的幅值不确定性。

虽然存在分离信号的排序不确定性和幅值不确定性, 但是从信号分离的角度看问题, 这两种不确定性是完全允许的, 因为原来混合的信号已被分离开, 而且一个固定的尺度因子的误差最多只影响信号的初始相位, 并不影响信号的波形。信号的波形通常保留了信号的有用信息。

由于盲信号分离存在分离信号的排序不确定性和幅值不确定性, 所以辨识出来的混合矩阵会相应存在两种不确定性: 各列排序的不确定性和每列元素可能相差一个固定的常数倍。这两种不确定性可以通过广义置换矩阵一并描述。

**定义 2.2.2** 一个正方矩阵称为广义置换矩阵 (generalized permutation matrix), 简称  $g$  矩阵, 若其每行和每列有一个并且仅有一个非零元素。

容易证明, 一个正方矩阵是  $g$  矩阵, 当且仅当它可以分解为一个置换矩阵和一个非奇异的对角矩阵之积, 即有

$$\mathbf{G} = \mathbf{P}\mathbf{D} \quad (2.2.10)$$

式中,  $\mathbf{D}$  为非奇异的对角矩阵。例如

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 & 0 & \alpha \\ 0 & 0 & \beta & 0 & 0 \\ 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda & 0 \\ \rho & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \rho & & & & 0 \\ & \gamma & & & \\ & & \beta & & \\ & & & \lambda & \\ 0 & & & & \alpha \end{bmatrix}$$

根据定义知, 如果用广义置换矩阵左乘 (或右乘), 则不仅使矩阵  $\mathbf{A}$  的行 (或列) 进行重新排列, 而且每行 (或列) 的元素还同乘一个比例因子。例如

$$\begin{bmatrix} 0 & 0 & 0 & 0 & \alpha \\ 0 & 0 & \beta & 0 & 0 \\ 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda & 0 \\ \rho & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{51} & a_{52} & a_{53} & a_{54} \end{bmatrix} = \begin{bmatrix} \alpha a_{51} & \alpha a_{52} & \alpha a_{53} & \alpha a_{54} \\ \beta a_{31} & \beta a_{32} & \beta a_{33} & \beta a_{34} \\ \gamma a_{21} & \gamma a_{22} & \gamma a_{23} & \gamma a_{24} \\ \lambda a_{41} & \lambda a_{42} & \lambda a_{43} & \lambda a_{44} \\ \rho a_{11} & \rho a_{12} & \rho a_{13} & \rho a_{14} \end{bmatrix}$$

回到刚才的盲信号分离问题, 易知其核心问题即是辨识  $\mathbf{P}\mathbf{D}\mathbf{A}^\dagger$ , 然后利用  $\mathbf{s}(t) = \mathbf{P}\mathbf{D}\mathbf{A}^\dagger \mathbf{x}(t) = \mathbf{G}\mathbf{A}^\dagger \mathbf{x}(t)$  得到分离的信号, 其中,  $\mathbf{P}$  和  $\mathbf{D}$  分别是置换矩阵和对角矩阵, 而  $\mathbf{G} = \mathbf{P}\mathbf{D}$  为广义置换矩阵。

顾名思义，选择矩阵 (selective matrix) 是一种可以对某个给定矩阵的某些行或者某些列进行选择的矩阵。以  $m \times N$  矩阵

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1(1) & \mathbf{x}_1(2) & \cdots & \mathbf{x}_1(N) \\ \mathbf{x}_2(1) & \mathbf{x}_2(2) & \cdots & \mathbf{x}_2(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_m(1) & \mathbf{x}_m(2) & \cdots & \mathbf{x}_m(N) \end{bmatrix}$$

为例。令

$$\mathbf{J}_1 = [\mathbf{I}_{m-1}, \mathbf{0}_{m-1}], \quad \mathbf{J}_2 = [\mathbf{0}_{m-1}, \mathbf{I}_{m-1}]$$

是两个  $(m-1) \times m$  矩阵，式中， $\mathbf{I}_{m-1}$  和  $\mathbf{0}_{m-1}$  分别是  $(m-1) \times (m-1)$  单位矩阵和  $(m-1) \times 1$  零向量。

直接计算得

$$\mathbf{J}_1 \mathbf{X} = \begin{bmatrix} \mathbf{x}_1(1) & \mathbf{x}_1(2) & \cdots & \mathbf{x}_1(N) \\ \mathbf{x}_2(1) & \mathbf{x}_2(2) & \cdots & \mathbf{x}_2(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_{m-1}(1) & \mathbf{x}_{m-1}(2) & \cdots & \mathbf{x}_{m-1}(N) \end{bmatrix}$$

$$\mathbf{J}_2 \mathbf{X} = \begin{bmatrix} \mathbf{x}_2(1) & \mathbf{x}_2(2) & \cdots & \mathbf{x}_2(N) \\ \mathbf{x}_3(1) & \mathbf{x}_3(2) & \cdots & \mathbf{x}_3(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_m(1) & \mathbf{x}_m(2) & \cdots & \mathbf{x}_m(N) \end{bmatrix}$$

即是说，矩阵  $\mathbf{J}_1 \mathbf{X}$  选择的是原矩阵  $\mathbf{X}$  的前  $m-1$  行，而矩阵  $\mathbf{J}_2 \mathbf{X}$  选择出原矩阵  $\mathbf{X}$  的后  $m-1$  行。

类似地，若令

$$\mathbf{J}_1 = \begin{bmatrix} \mathbf{I}_{N-1} \\ \mathbf{0}_{N-1} \end{bmatrix}, \quad \mathbf{J}_2 = \begin{bmatrix} \mathbf{0}_{N-1} \\ \mathbf{I}_{N-1} \end{bmatrix}$$

是两个  $N \times (N-1)$  矩阵，则

$$\mathbf{X} \mathbf{J}_1 = \begin{bmatrix} \mathbf{x}_1(1) & \mathbf{x}_1(2) & \cdots & \mathbf{x}_1(N-1) \\ \mathbf{x}_2(1) & \mathbf{x}_2(2) & \cdots & \mathbf{x}_2(N-1) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_m(1) & \mathbf{x}_m(2) & \cdots & \mathbf{x}_m(N-1) \end{bmatrix}$$

$$\mathbf{X} \mathbf{J}_2 = \begin{bmatrix} \mathbf{x}_1(2) & \mathbf{x}_1(3) & \cdots & \mathbf{x}_1(N) \\ \mathbf{x}_2(2) & \mathbf{x}_2(3) & \cdots & \mathbf{x}_2(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_m(2) & \mathbf{x}_m(3) & \cdots & \mathbf{x}_m(N) \end{bmatrix}$$

换言之，矩阵  $\mathbf{X} \mathbf{J}_1$  选择的是原矩阵  $\mathbf{X}$  的前  $N-1$  列，而矩阵  $\mathbf{X} \mathbf{J}_2$  选择出原矩阵  $\mathbf{X}$  的后  $N-1$  列。

### 2.3 正交矩阵与酉矩阵

向量  $\mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{C}^n$  组成一正交组, 若  $\mathbf{x}_i^H \mathbf{x}_j = 0, 1 \leq i < j \leq k$ 。此外, 若向量还是归一化的, 即  $\|\mathbf{x}\|_2^2 = \mathbf{x}_i^H \mathbf{x}_i = 1, i = 1, \dots, k$ , 则该正交组称为标准正交组。

**定理 2.3.1** 一组正交的非零向量是线性无关的。

**证明** 假设  $\{\mathbf{x}_1, \dots, \mathbf{x}_k\}$  是一正交组, 并假定  $0 = \alpha_1 \mathbf{x}_1 + \dots + \alpha_k \mathbf{x}_k$ 。于是有

$$0 = \mathbf{0}^H \mathbf{0} = \sum_{i=1}^k \sum_{j=1}^k \alpha_i^* \alpha_j \mathbf{x}_i^H \mathbf{x}_j = \sum_{i=1}^k |\alpha_i|^2 \mathbf{x}_i^H \mathbf{x}_i$$

由于向量是正交的, 且  $\mathbf{x}_i^H \mathbf{x}_i > 0$ , 故  $\sum_{i=1}^k |\alpha_i|^2 \mathbf{x}_i^H \mathbf{x}_i = 0$  的条件是所有  $|\alpha_i|^2 = 0$  即所有  $\alpha_i = 0$ , 从而  $\{\mathbf{x}_1, \dots, \mathbf{x}_k\}$  是线性无关的。 ■

**定义 2.3.1** 一实的正方矩阵  $\mathbf{Q} \in \mathbb{R}^{n \times n}$  称为正交矩阵, 若

$$\mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \quad (2.3.1)$$

一复值正方矩阵  $\mathbf{U} \in \mathbb{C}^{n \times n}$  称为酉矩阵, 若

$$\mathbf{U}\mathbf{U}^H = \mathbf{U}^H\mathbf{U} = \mathbf{I} \quad (2.3.2)$$

实矩阵  $\mathbf{Q}_{m \times n}$  称为半正交矩阵 (semi-orthogonal matrix), 若它只满足  $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}_m$  或者  $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}_n$ 。类似地, 复矩阵  $\mathbf{U}_{m \times n}$  称为仿酉矩阵 (para-unitary matrix), 若它只满足  $\mathbf{U}\mathbf{U}^H = \mathbf{I}_m$  或者  $\mathbf{U}^H\mathbf{U} = \mathbf{I}_n$ 。

由于正交矩阵事实上就是实的酉矩阵, 所以下面只讨论酉矩阵。

**定理 2.3.2**<sup>[238]</sup> 若  $\mathbf{U} \in \mathbb{C}^{n \times n}$ , 则下列叙述等价:

- (1)  $\mathbf{U}$  是酉矩阵;
- (2)  $\mathbf{U}$  是非奇异的, 并且  $\mathbf{U}^H = \mathbf{U}^{-1}$ ;
- (3)  $\mathbf{U}\mathbf{U}^H = \mathbf{U}^H\mathbf{U} = \mathbf{I}$ ;
- (4)  $\mathbf{U}^H$  是酉矩阵;
- (5)  $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]$  的列组成标准正交组, 即

$$\mathbf{u}_i^H \mathbf{u}_j = \delta(i - j) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

- (6)  $\mathbf{U}$  的行组成标准正交组;

- (7) 对所有  $\mathbf{x} \in \mathbb{C}^n$  而言,  $\mathbf{y} = \mathbf{U}\mathbf{x}$  的 Euclidean 长度与  $\mathbf{x}$  的 Euclidean 长度相同, 即  $\mathbf{y}^H \mathbf{y} = \mathbf{x}^H \mathbf{x}$ 。

若线性变换矩阵  $\mathbf{A}$  为酉矩阵, 则线性变换  $\mathbf{Ax}$  称为酉变换。酉变换具有以下性质:

(1) 向量内积在酉变换下是不变的, 即

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{Ax}, \mathbf{Ay} \rangle \quad (2.3.3)$$

这是因为  $\langle \mathbf{Ax}, \mathbf{Ay} \rangle = (\mathbf{Ax})^H \mathbf{Ay} = \mathbf{x}^H \mathbf{A}^H \mathbf{Ay} = \mathbf{x}^H \mathbf{y} = \langle \mathbf{x}, \mathbf{y} \rangle$ 。

(2) 向量范数在酉变换下是不变的, 即

$$\|\mathbf{Ax}\|^2 = \|\mathbf{x}\|^2 \quad (2.3.4)$$

因为  $\|\mathbf{Ax}\|^2 = \langle \mathbf{Ax}, \mathbf{Ax} \rangle = \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2$ 。

(3) 两个向量的夹角在酉变换下也是不变的, 即

$$\cos \theta = \frac{\langle \mathbf{Ax}, \mathbf{Ay} \rangle}{\|\mathbf{Ax}\| \|\mathbf{Ay}\|} = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|} \quad (2.3.5)$$

这是酉变换前两个性质的综合应用结果。

酉矩阵的行列式满足

$$|\det(\mathbf{A})| = 1, \quad \text{若 } \mathbf{A} \text{ 为酉矩阵} \quad (2.3.6)$$

证明如下: 根据行列式性质知, 对于任何复矩阵, 有  $\det(\mathbf{A}^H \mathbf{A}) = \det(\mathbf{A}^H) \det(\mathbf{A}) = \det(\mathbf{A}) \det(\mathbf{A}) = [\det(\mathbf{A})]^2$ 。当矩阵  $\mathbf{A}$  为酉矩阵时,  $\mathbf{A}^H \mathbf{A} = \mathbf{I}$ , 而单位矩阵的行列式等于 1。因此, 上式变成  $[\det(\mathbf{A})]^2 = \det(\mathbf{I}) = 1$ , 即得  $|\det(\mathbf{A})| = 1$ 。

表 2.3.1 归纳出了实向量、实矩阵与复向量、复矩阵之间的性质比较。

表 2.3.1 实向量、实矩阵与复向量、复矩阵的性质比较

实向量、实矩阵	复向量、复矩阵
范数 $\ \mathbf{x}\  = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$	范数 $\ \mathbf{x}\  = \sqrt{ x_1 ^2 +  x_2 ^2 + \cdots +  x_n ^2}$
转置 $\mathbf{A}^T = [a_{ji}]$ , $(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T$	共轭转置 $\mathbf{A}^H = [a_{ji}^*]$ , $(\mathbf{AB})^H = \mathbf{B}^H \mathbf{A}^H$
内积 $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$	内积 $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^H \mathbf{y}$
正交性 $\mathbf{x}^T \mathbf{y} = 0$	正交性 $\mathbf{x}^H \mathbf{y} = 0$
对称矩阵 $\mathbf{A}^T = \mathbf{A}$	Hermitian 矩阵 $\mathbf{A}^H = \mathbf{A}$
正交矩阵 $\mathbf{Q}^T = \mathbf{Q}^{-1}$	酉矩阵 $\mathbf{U}^H = \mathbf{U}^{-1}$
特征值分解 $\mathbf{A} = \mathbf{Q} \Sigma \mathbf{Q}^T = \mathbf{Q} \Sigma \mathbf{Q}^{-1}$	特征值分解 $\mathbf{A} = \mathbf{U} \Sigma \mathbf{U}^H = \mathbf{U} \Sigma \mathbf{U}^{-1}$
范数的正交不变性 $\ \mathbf{Q}\mathbf{x}\  = \ \mathbf{x}\ $	范数的酉不变性 $\ \mathbf{U}\mathbf{x}\  = \ \mathbf{x}\ $
内积的正交不变性 $\langle \mathbf{Q}\mathbf{x}, \mathbf{Q}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$	内积的酉不变性 $\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$

**定义 2.3.2** 一个满足  $\mathbf{B} = \mathbf{U}^H \mathbf{A} \mathbf{U}$  的矩阵  $\mathbf{B} \in \mathbb{C}^{n \times n}$  被称为与  $\mathbf{A} \in \mathbb{C}^{n \times n}$  酉等价。如果  $\mathbf{U}$  取实数 (因而是实正交的), 则称  $\mathbf{B}$  与  $\mathbf{A}$  正交等价。

**定理 2.3.3** 若  $n \times n$  矩阵  $A = [a_{ij}]$  和  $B = [b_{ij}]$  是酉等价的，则

$$\sum_{i=1}^n \sum_{j=1}^n |b_{ij}|^2 = \sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2$$

**证明** 利用矩阵乘法知  $\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2 = \text{tr}(A^H A)$ 。因此，只要证明  $\text{tr}(B^H B) = \text{tr}(A^H A)$  即可。由  $A$  和  $B$  的酉等价性  $B = U^H A U$ ，故有  $\text{tr}(B^H B) = \text{tr}(U^H A^H U U^H A U) = \text{tr}(U^H A^H A U) = \text{tr}(A^H A)$ ，从而定理得证。 ■

**定义 2.3.3** 矩阵  $A \in \mathbb{C}^{n \times n}$  称为正规矩阵 (normal matrix)，若  $A^H A = A A^H$ 。

容易验证，Hermitian 矩阵、斜 Hermitian 矩阵和酉矩阵都属于正规矩阵。

下面汇总了酉矩阵的有用性质 [324]：

- (1)  $A_{m \times m}$  为酉矩阵  $\Leftrightarrow A$  的列是标准正交的向量。
- (2)  $A_{m \times m}$  为酉矩阵  $\Leftrightarrow A$  的行是标准正交的向量。
- (3)  $A_{m \times m}$  为实矩阵时， $A$  为酉矩阵  $\Leftrightarrow A$  为正交矩阵。
- (4)  $A_{m \times m}$  为酉矩阵  $\Leftrightarrow A A^H = A^H A = I_m$

$$\begin{aligned} &\Leftrightarrow A^T \text{ 为酉矩阵} \\ &\Leftrightarrow A^H \text{ 为酉矩阵} \\ &\Leftrightarrow A^* \text{ 为酉矩阵} \\ &\Leftrightarrow A^{-1} \text{ 为酉矩阵} \\ &\Leftrightarrow A^i \text{ 为酉矩阵, } i = 1, 2, \dots \end{aligned}$$

- (5)  $A_{m \times m}, B_{m \times m}$  为酉矩阵  $\Rightarrow AB$  为酉矩阵。

(6) 若  $A_{m \times m}$  为酉矩阵，则

- ①  $|\det(A)| = 1$ 。
- ②  $\text{rank}(A) = m$ 。
- ③  $A$  是正规矩阵，即  $A A^H = A^H A$ 。
- ④  $\lambda$  为  $A$  的特征值  $\Rightarrow |\lambda| = 1$ 。
- ⑤  $B_{m \times n} \Rightarrow \|AB\|_F = \|B\|_F$ 。
- ⑥  $B_{n \times m} \Rightarrow \|BA\|_F = \|B\|_F$ 。
- ⑦  $x_{m \times 1} \Rightarrow \|Ax\|_2 = \|x\|_2$ 。

(7) 若  $A_{m \times m}, B_{n \times n}$  为酉矩阵，则

- ①  $A \oplus B$  为酉矩阵。
- ②  $A \otimes B$  为酉矩阵。

一个对角元素只取 +1 和 -1 两种值的  $N \times N$  对角矩阵称为符号矩阵 (signature matrix)，利用符号矩阵，可以引出与正交矩阵相仿的  $J$  正交矩阵的定义。

**定义 2.3.4** 令  $J$  为  $N \times N$  符号矩阵, 满足

$$QJQ^T = J \quad (2.3.7)$$

的  $N \times N$  矩阵  $Q$  称为  $J$  正交矩阵 ( $J$ -orthogonal matrix), 或称超正规矩阵 (hypernormal matrix)。

由定义易知, 当符号矩阵取单位矩阵, 即  $J = I$  时,  $J$  正交矩阵退化为正交矩阵。因此, 更确切地说, 正交矩阵实质上是单位正交矩阵。

$J$  正交矩阵具有以下性质 (证明留作习题):

- (1)  $J$  正交矩阵  $Q$  非奇异, 其行列式的绝对值等于 1。
- (2) 任何一个  $N \times N$  维  $J$  正交矩阵  $Q$  也可以等价定义为

$$Q^T J Q = J \quad (2.3.8)$$

综合式 (2.3.7) 和式 (2.3.8), 立即得

$$Q^T J Q = Q J Q^T \quad (2.3.9)$$

这一对称性称为“双曲对称性” (hyperbolic symmetry)。

矩阵

$$Q = J - 2 \frac{vv^T}{v^T J v} \quad (2.3.10)$$

称为双曲 Householder 矩阵<sup>[71]</sup>。显然, 若  $J = I$ , 则双曲 Householder 矩阵退化为 Householder 矩阵。

## 2.4 带型矩阵与三角矩阵

三角矩阵是矩阵的分解与变换的标准形式之一, 它又是带型矩阵的一个特例。

### 2.4.1 带型矩阵

满足条件  $a_{ij} = 0, |i - j| > k$  的矩阵  $A \in \mathbb{C}^{m \times n}$  称为带型矩阵 (banded matrix)。特别地, 若  $a_{ij} = 0, \forall i > j + p$ , 就称  $A$  具有下带宽  $p$ ; 若  $a_{ij} = 0, \forall j > i + q$ , 则称矩阵  $A$  具有上带宽  $q$ 。下面是一个  $7 \times 5$  带型矩阵的例子, 它具有下带宽 1 和上带宽 2:

$$\begin{bmatrix} x & x & x & 0 & 0 \\ x & x & x & x & 0 \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

其中,  $\times$  表示任意非零元素。

带型矩阵的一种特殊形式是特别令人感兴趣的, 这就是三对角矩阵。矩阵  $A \in C^{n \times n}$  是三对角矩阵, 若每当  $|i - j| > 1$  时  $a_{ij} = 0$ 。显然, 三对角矩阵是上、下带宽各为 1 的带型正方矩阵。另外一方面, 三对角矩阵也是下述 Hessenberg 矩阵的一个特例。

$n \times n$  正方矩阵  $A$  称为上 Hessenberg 矩阵, 若它具有形式

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & a_{32} & a_{33} & \cdots & a_{3n} \\ 0 & 0 & a_{43} & \cdots & a_{4n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix}$$

矩阵  $A$  称作下 Hessenberg 矩阵, 若  $A^T$  是上 Hessenberg 矩阵。

事实上, 三对角矩阵就是一个既是上 Hessenberg, 又是下 Hessenberg 的正方矩阵。

#### 2.4.2 三角矩阵

两种特殊的常用带型矩阵为上三角矩阵和下三角矩阵。三角矩阵是矩阵分解中的典范形式之一。

满足条件  $a_{ij} = 0, i > j$  的正方矩阵  $U = [u_{ij}]$  称为上三角矩阵 (upper triangular matrix), 其一般形式为

$$U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

满足条件  $l_{ij} = 0, i < j$  的正方矩阵  $L = [l_{ij}]$  称为下三角矩阵 (lower triangular matrix), 其一般形式为

$$L = \begin{bmatrix} l_{11} & & & 0 \\ l_{12} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \rightarrow |L| = l_{11}l_{22} \cdots l_{nn}$$

将有关三角矩阵的定义加以归纳, 一个正方矩阵  $A = [a_{ij}]$  称为:

- (1) 下三角矩阵, 若  $a_{ij} = 0 (i < j)$ ;
- (2) 严格下三角矩阵, 若  $a_{ij} = 0 (i \leq j)$ ;
- (3) 单位下三角矩阵, 若  $a_{ij} = 0 (i < j), a_{ii} = 1 (\forall i)$ ;
- (4) 上三角矩阵, 若  $a_{ij} = 0 (i > j)$ ;
- (5) 严格上三角矩阵, 若  $a_{ij} = 0 (i \geq j)$ ;
- (6) 单位上三角矩阵, 若  $a_{ij} = 0 (i > j), a_{ii} = 1 (\forall i)$ 。

下面列举上三角矩阵的性质:

(1) 上三角矩阵之积为上三角矩阵, 即若  $\mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_k$  各为上三角矩阵, 则  $\mathbf{U} = \mathbf{U}_1 \mathbf{U}_2 \cdots \mathbf{U}_k$  为上三角矩阵。

(2) 上三角矩阵  $\mathbf{U} = [u_{ij}]$  的行列式等于对角线元素之积, 即

$$\det(\mathbf{U}) = u_{11}u_{22} \cdots u_{nn} = \prod_{i=1}^n u_{ii}$$

(3) 上三角矩阵的逆矩阵为上三角矩阵。

(4) 上三角矩阵  $\mathbf{U}_{n \times n}$  的  $k$  次幂  $\mathbf{U}^k$  仍为上三角矩阵, 并且其第  $i$  个对角线元素等于  $u_{ii}^k$ 。

(5) 上三角矩阵  $\mathbf{U}_{n \times n} = \mathbf{U} = [u_{ij}]$  的特征值为  $u_{11}, u_{22}, \dots, u_{nn}$ 。

(6) 正定 Hermitian 矩阵  $\mathbf{A}$  可以分解为  $\mathbf{A} = \mathbf{T}^H \mathbf{D} \mathbf{T}$ , 其中,  $\mathbf{T}$  为单位上三角复矩阵,  $\mathbf{D}$  为实对角矩阵。

下三角矩阵的性质如下:

(1) 下三角矩阵之积为下三角矩阵, 即若  $\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_k$  各为下三角矩阵, 则  $\mathbf{L} = \mathbf{L}_1 \mathbf{L}_2 \cdots \mathbf{L}_k$  为下三角矩阵。

(2) 下三角矩阵的行列式等于对角线元素之积, 即

$$\det(\mathbf{L}) = l_{11}l_{22} \cdots l_{nn} = \prod_{i=1}^n l_{ii}$$

(3) 下三角矩阵的逆矩阵为下三角矩阵。

(4) 下三角矩阵  $\mathbf{L}_{n \times n}$  的  $k$  次幂  $\mathbf{L}^k$  仍为下三角矩阵, 且第  $i$  个对角线元素等于  $l_{ii}^k$ 。

(5) 下三角矩阵  $\mathbf{L}_{n \times n}$  的特征值为  $l_{11}, l_{22}, \dots, l_{nn}$ 。

(6) 一个正定矩阵  $\mathbf{A}_{n \times n}$  能够分解为下三角矩阵  $\mathbf{L}_{n \times n}$  与其转置之积, 即  $\mathbf{A} = \mathbf{L}\mathbf{L}^T$ 。这一分解称为矩阵  $\mathbf{A}$  的 Cholesky 分解。

有时称满足  $\mathbf{A} = \mathbf{L}\mathbf{L}^T$  的下三角矩阵  $\mathbf{L}$  为矩阵  $\mathbf{A}$  的平方根。更一般地, 满足

$$\mathbf{B}^2 = \mathbf{A} \quad (2.4.1)$$

的任何矩阵  $\mathbf{B}$  称为  $\mathbf{A}$  的平方根, 记作  $\mathbf{A}^{1/2}$ 。需要注意的是, 一个正方矩阵  $\mathbf{A}$  的平方根不一定是唯一的。

若对角线或者交叉对角线上的矩阵是可逆的, 则分块三角矩阵的求逆公式为

$$\begin{bmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{B} & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} & \mathbf{O} \\ -\mathbf{C}^{-1}\mathbf{B}\mathbf{A}^{-1} & \mathbf{C}^{-1} \end{bmatrix} \quad (2.4.2)$$

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{O} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{O} & \mathbf{C}^{-1} \\ \mathbf{B}^{-1} & -\mathbf{B}^{-1}\mathbf{AC}^{-1} \end{bmatrix} \quad (2.4.3)$$

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{O} & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{BC}^{-1} \\ \mathbf{O} & \mathbf{C}^{-1} \end{bmatrix} \quad (2.4.4)$$

## 2.5 求和向量与中心化矩阵

本节介绍求和向量与中心化矩阵。

### 2.5.1 求和向量

所有元素等于 1 的向量称为求和向量 (summuining vector), 记为  $\mathbf{1} = [1, 1, \dots, 1]^T$ 。以  $n = 4$  为例, 求和向量  $\mathbf{1} = [1, 1, 1, 1]^T$ 。之所以称为求和向量, 乃是因为  $n$  个标量的求和都可以表示为求和向量与另外一个向量之间的内积。

**例 2.5.1** 若令  $\mathbf{x} = [a, b, -c, d]^T$ , 则求和  $a + b - c + d$  可以表示为

$$a + b - c + d = [1, 1, 1, 1] \begin{bmatrix} a \\ b \\ -c \\ d \end{bmatrix} = \mathbf{1}^T \mathbf{x} = \mathbf{x}^T \mathbf{1}$$

在某些运算中, 可能遇到不同维数的求和向量。此时, 为了避免混淆, 常写出求和向量的维数, 如  $\mathbf{1}_3 = [1, 1, 1]^T$ 。考虑求和向量与矩阵的乘积

$$\mathbf{1}_3^T \mathbf{X}_{3 \times 2} = [1, 1, 1] \begin{bmatrix} 4 & -1 \\ -4 & 3 \\ 1 & -1 \end{bmatrix} = [1, 1] = \mathbf{1}_2^T$$

求和向量与自己的内积是一个等于该向量维数的标量, 即有

$$\mathbf{1}_n^T \mathbf{1}_n = n \quad (2.5.1)$$

求和向量之间的外积是一个所有元素为 1 的矩阵, 例如

$$\mathbf{1}_2 \mathbf{1}_3^T = \begin{bmatrix} 1 \\ 1 \end{bmatrix} [1, 1, 1] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \mathbf{J}_{2 \times 3}$$

更一般地, 有

$$\mathbf{1}_p \mathbf{1}_q^T = \mathbf{J}_{p \times q} \quad (\text{所有元素为 1 的矩阵}) \quad (2.5.2)$$

于是, 一个所有元素为  $\alpha$  的  $p \times q$  矩阵可以表示为  $\alpha \mathbf{J}_{p \times q}$ 。

容易验证

$$\mathbf{J}_{m \times p} \mathbf{J}_{p \times n} = p \mathbf{J}_{m \times n} \quad (2.5.3)$$

$$\mathbf{J}_{p \times q} \mathbf{1}_q = q \mathbf{1}_p \quad (2.5.4)$$

$$\mathbf{1}_p^T \mathbf{J}_{p \times q} = p \mathbf{1}_q^T \quad (2.5.5)$$

特别地, 对于  $n \times n$  矩阵  $\mathbf{J}_n$ , 有

$$\mathbf{J}_n = \mathbf{1}_n \mathbf{1}_n^T, \quad \mathbf{J}_n^2 = n \mathbf{J}_n \quad (2.5.6)$$

于是, 若令

$$\bar{\mathbf{J}}_n = \frac{1}{n} \mathbf{J}_n \quad (2.5.7)$$

则有  $\bar{\mathbf{J}}_n^2 = \bar{\mathbf{J}}_n$ , 即  $\bar{\mathbf{J}}_n$  是一个幂等矩阵。

### 2.5.2 中心化矩阵

矩阵

$$\mathbf{C}_n = \mathbf{I}_n - \bar{\mathbf{J}}_n = \mathbf{I}_n - \frac{1}{n} \mathbf{J}_n \quad (2.5.8)$$

称为中心化矩阵 (centering matrix)。

容易验证, 中心化矩阵既是对称矩阵, 又是幂等矩阵, 即有

$$\mathbf{C}_n = \mathbf{C}_n^T = \mathbf{C}_n^2 \quad (2.5.9)$$

此外, 中心化矩阵还具有以下特性

$$\left. \begin{array}{l} \mathbf{C}_n \mathbf{1} = \mathbf{0} \\ \mathbf{C}_n \mathbf{J}_n = \mathbf{J}_n \mathbf{C}_n = \mathbf{0} \end{array} \right\} \quad (2.5.10)$$

求和向量  $\mathbf{1}$  与中心化矩阵  $\mathbf{J}$  在数理统计中非常有用 [444, p.67]。

首先, 一组数据  $x_1, \dots, x_n$  的均值可以用求和向量表示, 即有

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n) = \frac{1}{n} \mathbf{x}^T \mathbf{1} = \frac{1}{n} \mathbf{1}^T \mathbf{x} \quad (2.5.11)$$

式中,  $\mathbf{x} = [x_1, \dots, x_n]^T$  为数据向量。

其次, 利用中心化矩阵的定义式 (2.5.8) 及其性质公式 (2.5.10), 可以得到

$$\begin{aligned} \mathbf{C}\mathbf{x} &= \mathbf{x} - \bar{\mathbf{J}}\mathbf{x} = \mathbf{x} - \frac{1}{n} \mathbf{1}\mathbf{1}^T \mathbf{x} = \mathbf{x} - \bar{x}\mathbf{1} \\ &= [x_1 - \bar{x}, \dots, x_n - \bar{x}]^T \end{aligned} \quad (2.5.12)$$

换言之, 矩阵  $\mathbf{C}$  对数据向量  $\mathbf{x}$  的线性变换  $\mathbf{C}\mathbf{x}$  是原数据向量的各个元素减去  $n$  个数据的均值的结果。这就是中心化矩阵的数学含义所在。

此外, 如果求向量  $\mathbf{C}\mathbf{x}$  的内积, 则有

$$\begin{aligned} (\mathbf{C}\mathbf{x})^T \mathbf{C}\mathbf{x} &= [x_1 - \bar{x}, \dots, x_n - \bar{x}] [x_1 - \bar{x}, \dots, x_n - \bar{x}]^T \\ &= \sum_{i=1}^n (x_i - \bar{x})^2 \end{aligned}$$

由式 (2.5.10) 知  $\mathbf{C}^T \mathbf{C} = \mathbf{C}\mathbf{C} = \mathbf{C}$ , 上式又可简化为

$$\mathbf{x}^T \mathbf{C}\mathbf{x} = \sum_{i=1}^n (x_i - \bar{x})^2 \quad (2.5.13)$$

式右是我们熟悉的数据  $x_1, \dots, x_n$  的协方差。即是说, 一组数据的协方差可以用核矩阵为中心化矩阵的二次型  $\mathbf{x}^T \mathbf{C}\mathbf{x}$  表示。

## 2.6 相似矩阵与相合矩阵

本节讨论矩阵的两种特殊线性变换。

### 2.6.1 相似矩阵

令  $S \in \mathbb{C}^{n \times n}$  为非奇异矩阵, 考查矩阵  $A \in \mathbb{C}^{n \times n}$  的线性变换

$$B = S^{-1}AS \quad (2.6.1)$$

令线性变换  $B$  的特征值为  $\lambda$ , 对应的特征向量为  $y$ , 即

$$By = \lambda y \quad (2.6.2)$$

将式 (2.6.1) 代入式 (2.6.2), 即有  $S^{-1}ASy = \lambda y$  或  $A(Sy) = \lambda(Sy)$ 。若令  $x = Sy$  或  $y = S^{-1}x$ , 则立即有

$$Ax = \lambda x \quad (2.6.3)$$

比较式 (2.6.2) 和式 (2.6.3) 知, 矩阵  $A$  和  $B = S^{-1}AS$  具有相同的特征值, 并且矩阵  $B$  的特征向量  $y$  是矩阵  $A$  的特征向量  $x$  的线性变换, 即  $y = S^{-1}x$ 。由于矩阵  $A$  和  $B = S^{-1}AS$  的特征值相同, 特征向量存在线性变换的关系, 所以称这两个矩阵“相似”。于是, 有下面的数学定义。

**定义 2.6.1** (相似矩阵与相似变换) 矩阵  $B \in \mathbb{C}^{n \times n}$  称为矩阵  $A \in \mathbb{C}^{n \times n}$  的相似矩阵, 若存在一非奇异矩阵  $S \in \mathbb{C}^{n \times n}$  使得  $B = S^{-1}AS$ 。此时, 线性变换  $A \mapsto S^{-1}AS$  称为矩阵  $A$  的相似变换。关系“ $B$  相似于  $A$ ”常简写作  $B \sim A$ 。

相似矩阵具有以下基本性质:

(1) 自反性  $A \sim A$ , 即任一矩阵与它自己相似。

(2) 对称性 若  $A$  相似于  $B$ , 则  $B$  也相似于  $A$ 。

(3) 传递性 若  $A \sim B$  和  $B \sim C$ , 则  $A \sim C$ 。

下面是关于相似矩阵的两个重要定理。

**定理 2.6.1** 令  $A, B \in \mathbb{C}^{n \times n}$ 。若  $B$  与  $A$  相似, 则  $\det(B) = \det(A)$  和  $\text{tr}(B) = \text{tr}(A)$ , 即相似矩阵的行列式相等, 并具有相同的迹。

**证明** 对相似关系  $B = S^{-1}AS$  分别运用行列式的性质, 得

$$\begin{aligned} \det(B) &= \det(S^{-1}AS) = \det(S^{-1})\det(A)\det(S) \\ &= \det(A)\det(S^{-1})\det(S) = \det(A)\det(S^{-1}S) \\ &= \det(A)\det(I) = \det(A) \end{aligned}$$

利用迹的性质, 又有  $\text{tr}(B) = \text{tr}(S^{-1}AS) = \text{tr}(S^{-1}(AS)) = \text{tr}(ASS^{-1}) = \text{tr}(A)$ 。 ■

**定理 2.6.2** 令  $A, B \in \mathbb{C}^{n \times n}$ 。若  $B$  与  $A$  相似，则  $B$  的特征多项式  $\det(B - zI)$  与  $A$  的特征多项式  $\det(A - zI)$  相同。

**证明** 对任意  $z$ , 有

$$\begin{aligned}\det(B - zI) &= \det(S^{-1}AS - zS^{-1}S) \\ &= \det(S^{-1}(A - zI)S) \\ &= \det(S^{-1})\det(A - zI)\det(S) \\ &= (\det(S))^{-1}\det(S)\det(A - zI) \\ &= \det(A - zI)\end{aligned}$$

即定理得证。 ■

注意到一个矩阵的特征值定义为该矩阵的特征多项式的根，上述定理给出以下推论。

**推论 2.6.1** 若  $A, B \in \mathbb{C}^{n \times n}$ , 并且  $A$  和  $B$  相似，则它们具有相同的特征值（包括多重特征值在内）。

这个推论启发我们，如果想得到一个矩阵  $A$  的特征值，可以通过相似变换，使  $A$  的相似矩阵为三角矩阵。这样一来，该三角矩阵的对角元素便给出矩阵  $A$  的所有特征值（包括多重度在内）。

下面是相似矩阵的重要性质：

- (1) 相似矩阵  $B \sim A$  具有相同的行列式，即  $|B| = |A|$ 。
- (2) 若矩阵  $S^{-1}AS = T$  (上三角矩阵)，则  $T$  的对角元素给出矩阵  $A$  的特征值  $\lambda_i$ 。
- (3) 两个相似矩阵具有完全相同的特征值。
- (4) 若  $A$  的特征值各不相同，则一定可以找到一相似矩阵  $S^{-1}AS = D$  (对角矩阵)，其对角元素即是矩阵  $A$  的特征值。
- (5)  $n \times n$  矩阵  $A$  与对角矩阵相似的充分必要条件是：矩阵  $A$  的  $n$  个特征向量线性无关。
- (6) 相似矩阵  $B = S^{-1}AS$  意味着  $B^2 = S^{-1}ASS^{-1}AS = S^{-1}A^2S$ ，从而有  $B^k = S^{-1}A^kS$ 。也就是说，若  $B \sim A$ ，则  $B^k \sim A^k$ 。这一性质称为相似矩阵的幂性质。
- (7) 若矩阵  $B = S^{-1}AS$  和  $A$  均可逆，则  $B^{-1} = S^{-1}A^{-1}S$ ，即当两个矩阵相似时，它们的逆矩阵也相似。

在相似变换中最重要的酉相似变换。如果矩阵  $A$  经过酉矩阵相似变换为  $B$ ，就称  $A$  和  $B$  是酉相似的。例如，若 Hermitian 矩阵  $A$  经过酉矩阵  $U^{-1} = U^H$  相似变换为对角矩阵  $\Sigma$ ，即有  $\Sigma = U^H A U$ ，则根据推论 2.6.1 知，Hermitian 矩阵  $A$  与酉相似的对角矩阵  $\Sigma$  具有相同的特征值，这正是 Hermitian 矩阵  $A^H = A$  的特征值分解  $A = U\Sigma U^H$  的理论基础。

### 2.6.2 相合矩阵

与相似矩阵在形式上部分相同的矩阵是相合矩阵。

**定义 2.6.2** (相合矩阵与相合变换) 令  $A, B, C \in \mathbb{C}^{n \times n}$ , 并且  $C$  非奇异, 则矩阵  $B = C^H A C$  称为  $A$  的相合矩阵 (congruent matrix), 而线性变换  $A \mapsto C^H A C$  称为相合变换。

相合矩阵具有以下特性:

- (1) 自反性  $A$  相合于  $A$ , 即任一矩阵与它自己相合。
- (2) 对称性 若  $A$  相合于  $B$ , 则  $B$  也相合于  $A$ 。
- (3) 传递性 若  $A$  相合于  $B$ , 而  $B$  又相合于  $D$ , 则  $A$  相合于  $D$ 。

**证明** 对于相合而言:

- (1) 若取  $C = I$ , 则显然有  $A$  与  $C^H A C = A$  相合。特性 (1) 得证。
- (2) 若  $A$  相合于  $B$ , 即  $A = C^H B C$ , 则有

$$B = (C^H)^{-1} A C^{-1} = (C^{-1})^H A C^{-1} = T^H A T$$

式中,  $T = C^{-1}$  为非奇异矩阵。上式表明, 矩阵  $B$  相合于  $A$ 。特性 (2) 得证。

- (3) 若  $B = C_1^H A C_1$  和  $D = C_2^H B C_2$ , 则

$$A = (C_1^H)^{-1} B C_1^{-1} = (C_1^H)^{-1} [(C_2^H)^{-1} D C_2^{-1}] C_1^{-1} = [(C_1 C_2)^{-1}]^H D (C_1 C_2)^{-1}$$

即  $A$  相合于  $D$ , 因为  $C_1 C_2$  非奇异。特性 (3) 得证。 ■

对于一个  $n \times n$  维 Hermitian 矩阵  $A$ , 存在一个非奇异矩阵  $T$ , 使得  $A = T D T^H$ , 其中,  $D = \text{diag}(d_1, \dots, d_n)$  是对角矩阵, 并且对角元素  $d_i$  只取  $+1, -1$  和  $0$  这 3 种值, 它们分别与矩阵  $A$  的正特征值、负特征值和零特征值相对应。此时, 称  $T D T^H$  是矩阵  $A$  的相合规范型 (congruent canonical form), 而对角矩阵  $D$  则称为矩阵  $A$  的规范相合矩阵 (canonical congruent matrix)。

术语“相合”具有以下两层含义:

- (1) 两个相合矩阵  $A$  和  $B$  的二次型函数相吻合。考查二次型函数  $f(\mathbf{x}) = \mathbf{x}^H A \mathbf{x}$ 。若令  $\mathbf{x} = \mathbf{C} \mathbf{y}$ , 其中  $\mathbf{C}$  为非奇异矩阵, 则有

$$\mathbf{x}^H A \mathbf{x} = \mathbf{y}^H C^H A C \mathbf{y} = \mathbf{y}^H B \mathbf{y} \quad (2.6.4)$$

式中,  $B = C^H A C$ 。这表明, 两个相合矩阵具有相同的二次型函数。

- (2) Hermitian 矩阵  $A$  的规范相合矩阵  $D$  与酉相似对角化矩阵 (特征值矩阵)  $\Sigma$  的元素具有以下关系: 零元素的个数相同, 对应的非零元素具有相同的符号。

## 2.7 Vandermonde 矩阵

本节考查每行元素组成一个等比序列的两类特殊矩阵, 它们是 Vandermonde 矩阵和 Fourier 矩阵, 在信号处理中有着广泛的应用。事实上, Fourier 矩阵是 Vandermonde 矩阵的一种特例。因此, 本节先介绍 Vandermonde 矩阵。

$n \times n$  维 Vandermonde 矩阵是取以下特殊形式的矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \quad (2.7.1)$$

或

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \cdots & x_n^{n-1} \end{bmatrix} \quad (2.7.2)$$

即矩阵每行 (或列) 的元素组成一个等比序列。

Vandermonde 矩阵有一个突出的性质:  $n$  个参数  $x_1, x_2, \dots, x_n$  各异时, Vandermonde 矩阵非奇异。为此, 需要证明行列式  $\det(\mathbf{A}) \neq 0$ ,  $x_i \neq x_j, \forall i \neq j$ 。最简单的方法莫过于直接评价 Vandermonde 矩阵的行列式 [36, p.193]。显然, 可以将  $\det(\mathbf{A})$  视为  $x_1$  的  $n-1$  阶多项式。作为一个  $n-1$  阶多项式,  $\det(\mathbf{A})$  有根  $x_1 = x_2, x_1 = x_3, \dots, x_1 = x_n$ , 因为每当矩阵的两行 (或列) 相同时, 行列式等于零。于是, 可以将行列式表示为

$$\det(\mathbf{A}) = (x_2 - x_1)(x_3 - x_1) \cdots (x_n - x_1)q(x_2, x_3, \dots, x_n)$$

式中,  $q(x_2, x_3, \dots, x_n)$  是一个只与  $x_2, x_3, \dots, x_n$  有关的多项式。类似地, 行列式  $\det(\mathbf{A})$  也可以分别视为关于  $x_2, x_3, \dots, x_n$  的  $n-1$  阶多项式。于是, 可将 Vandermonde 矩阵的行列式用多项式形式表示为

$$\det(\mathbf{A}) = \prod_{1 \leq j < i \leq n} (x_i - x_j) \phi(x_1, x_2, \dots, x_n)$$

式中,  $\phi$  是关于  $x_1, x_2, \dots, x_n$  的多项式。由于行列式  $\det(\mathbf{A})$  中  $x_i$  的阶数为  $n$ , 所以  $\phi$  必定为常数项, 不可能与任何  $x_i$  有关。特别地, 当  $n=2$  时, Vandermonde 矩阵的行列式  $\begin{vmatrix} 1 & 1 \\ x_1 & x_2 \end{vmatrix} = x_2 - x_1$ 。由此知  $\phi = 1$ 。因此,  $n \times n$  维 Vandermonde 矩阵的行列式由下式给出 [36, p.193]

$$\det(\mathbf{A}) = \prod_{i,j=1, i>j}^n (x_i - x_j) \quad (2.7.3)$$

显然, 若  $x_i \neq x_j, \forall i \neq j$ , 则  $\det(\mathbf{A}) \neq 0$ , 即 Vandermonde 矩阵非奇异。

在多项式插值问题中，通常需要求最高阶次为  $n - 1$  次的多项式  $p(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_1x + a_0$ ，并要求它满足

$$\left. \begin{array}{l} p(x_1) = a_0 + a_1x_1 + a_2x_1^2 + \cdots + a_{n-1}x_1^{n-1} = y_1 \\ p(x_2) = a_0 + a_1x_2 + a_2x_2^2 + \cdots + a_{n-1}x_2^{n-1} = y_2 \\ \vdots \\ p(x_n) = a_0 + a_1x_n + a_2x_n^2 + \cdots + a_{n-1}x_n^{n-1} = y_n \end{array} \right\} \quad (2.7.4)$$

其中， $x_1, x_2, \dots, x_n$  和  $y_1, y_2, \dots, y_n$  为已知。插值条件式 (2.7.4) 是一组线性方程，共有  $n$  个方程和  $n$  个未知系数  $a_0, a_1, \dots, a_{n-1}$ 。方程组可写作  $\mathbf{A}\mathbf{a} = \mathbf{y}$ ，其中， $\mathbf{a} = [a_0, a_1, \dots, a_{n-1}]^T$ ,  $\mathbf{y} = [y_1, y_2, \dots, y_n]^T$ ，且矩阵  $\mathbf{A}$  是如式 (2.7.1) 所示的 Vandermonde 矩阵。若数据点  $x_1, x_2, \dots, x_n$  各不相同，则插值问题总有一个解，因为  $\mathbf{A}$  在这种情况下是非奇异的。

若有两个或多个元素  $x_i$  相同，则相应的多项式插值问题是欠定的。此时，可以使用汇合型 Vandermonde 矩阵 (confluent Vandermonde matrices) 表示插值问题。汇合型 Vandermonde 矩阵是由不相同的的所有元素组成的 Vandermonde 矩阵。例如，若  $x_i = x_{i+1} = \cdots = x_{i+k}$ ，且  $x_i \neq x_{i-1}$ ，则汇合型 Vandermonde 矩阵  $\mathbf{A}$  的第  $(i+k)$  行的元素为

$$A_{i+k,j} = \begin{cases} 0, & \text{若 } j \leq k \\ \frac{(j-1)!}{(j-k-1)!} x_i^{j-k-1}, & \text{若 } j > k \end{cases} \quad (2.7.5)$$

汇合型 Vandermonde 矩阵具有与 Vandermonde 矩阵相同的性质。

在信号处理中经常遇到是复 Vandermonde 矩阵。

**例 2.7.1 (扩展 Prony 方法)** 在谐波恢复的扩展 Prony 方法中，信号模型假定是一组  $p$  个指数函数的叠加，这组指数函数有任意的幅值、相位、频率和阻尼因子。于是，离散时间的数学模型

$$\hat{x}_n = \sum_{i=1}^p b_i z_i^n, \quad n = 0, 1, \dots, N-1 \quad (2.7.6)$$

被用作拟合观测数据  $x_0, x_1, \dots, x_{N-1}$  的数学模型。通常， $b_i$  和  $z_i$  假定为复数，并且

$$b_i = A_i \exp(j\theta_i), \quad z_i = \exp[(\alpha_i + j2\pi f_i)\Delta t]$$

其中， $A_i$  是幅值， $\theta_i$  是相位 (弧度)， $\alpha_i$  为阻尼因子， $f_i$  为振荡频率 (Hz)， $\Delta t$  代表采样间隔 (秒)。式 (2.7.6) 的矩阵形式是

$$\Phi \mathbf{b} = \hat{\mathbf{x}}$$

其中， $\mathbf{b} = [b_0, b_1, \dots, b_p]^T$ ,  $\hat{\mathbf{x}} = [\hat{x}_0, \hat{x}_1, \dots, \hat{x}_{N-1}]^T$ ，而  $\Phi$  是一复 Vandermonde 矩阵

$$\Phi = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ z_1 & z_2 & z_3 & \cdots & z_p \\ z_1^2 & z_2^2 & z_3^2 & \cdots & z_p^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ z_1^{N-1} & z_2^{N-1} & z_3^{N-1} & \cdots & z_p^{N-1} \end{bmatrix} \quad (2.7.7)$$

使平方误差  $\epsilon = \sum_{n=1}^{N-1} |x_n - \hat{x}_n|^2$  最小, 便得到最小二乘解

$$\mathbf{b} = [\Phi^H \Phi]^{-1} \Phi^H \mathbf{x} \quad (2.7.8)$$

容易证明, 式 (2.7.8) 中的  $\Phi^H \Phi$  的计算可以大大简化, 使得无须作 Vandermonde 矩阵的乘法运算, 就能够直接利用

$$\Phi^H \Phi = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1p} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ \gamma_{p1} & \gamma_{p2} & \cdots & \gamma_{pp} \end{bmatrix} \quad (2.7.9)$$

计算出  $\Phi^H \Phi$ , 其中

$$\gamma_{ij} = \frac{(z_i^* z_j)^N - 1}{(z_i^* z_j) - 1} \quad (2.7.10)$$

式 (2.7.7) 所示的  $N \times p$  矩阵  $\Phi$  是在信号处理中广泛应用的 Vandermonde 矩阵之一。信号处理中另外一种与式 (2.7.7) 类似的 Vandermonde 矩阵为

$$\Phi = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{\lambda_1} & e^{\lambda_2} & \cdots & e^{\lambda_d} \\ \vdots & \vdots & \vdots & \vdots \\ e^{\lambda_1(N-1)} & e^{\lambda_2(N-1)} & \cdots & e^{\lambda_d(N-1)} \end{bmatrix} \quad (2.7.11)$$

在信号重构、系统辨识和其他一些信号处理问题中, 需要对 Vandermonde 矩阵求逆。

$n \times n$  复 Vandermonde 矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ a_1 & a_2 & \cdots & a_n \\ \vdots & \vdots & \vdots & \vdots \\ a_1^{n-1} & a_2^{n-1} & \cdots & a_n^{n-1} \end{bmatrix}, \quad a_k \in \mathbb{C} \quad (2.7.12)$$

的逆矩阵由下式给出 [357]

$$\mathbf{A}^{-1} = \begin{bmatrix} \frac{\sigma_{n-1}(a_2, a_3, \dots, a_n)}{\prod_{k=2}^n (a_k - a_1)} & -\frac{\sigma_{n-2}(a_2, a_3, \dots, a_n)}{\prod_{k=2}^n (a_k - a_1)} & \dots & \frac{(-1)^{n+1}}{\prod_{k=2}^n (a_k - a_1)} \\ \frac{\sigma_{n-1}(a_1, a_3, \dots, a_n)}{(a_2 - a_1) \prod_{k=3}^n (a_k - a_2)} & \frac{\sigma_{n-2}(a_1, a_3, \dots, a_n)}{(a_2 - a_1) \prod_{k=3}^n (a_k - a_2)} & \dots & \frac{(-1)^{n+2}}{(a_2 - a_1) \prod_{k=3}^n (a_k - a_2)} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\sigma_{n-1}(a_1, a_2, \dots, a_{n-1})}{(-1)^{n+1} \prod_{k=1}^{n-1} (a_n - a_k)} & \frac{\sigma_{n-2}(a_1, a_2, \dots, a_{n-1})}{(-1)^{n+2} \prod_{k=1}^{n-1} (a_n - a_k)} & \dots & \frac{1}{\prod_{k=1}^{n-1} (a_n - a_k)} \end{bmatrix} \quad (2.7.13)$$

## 2.8 Fourier 矩阵

Fourier 矩阵是一种特殊结构的 Vandermonde 矩阵，在信号处理、图像处理、生物医学和生物信息、模式识别、自动控制等中有着广泛的应用。

### 2.8.1 Fourier 矩阵的定义与性质

离散时间信号  $x_0, x_1, \dots, x_{N-1}$  的 Fourier 变换称为信号的离散 Fourier 变换 (DFT) 或频谱，定义为

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi nk/N} = \sum_{n=0}^{N-1} x_n w^{nk}, \quad k = 0, 1, \dots, N-1 \quad (2.8.1)$$

写成矩阵形式，有

$$\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w & \cdots & w^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w^{N-1} & \cdots & w^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix} \quad (2.8.2)$$

或简记作

$$\hat{x} = Fx \quad (2.8.3)$$

式中， $x = [x_0, x_1, \dots, x_{N-1}]^T$  和  $\hat{x} = [X_0, X_1, \dots, X_{N-1}]^T$  分别是离散时间信号向量和频谱向量，而

$$F = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w & \cdots & w^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w^{N-1} & \cdots & w^{(N-1)(N-1)} \end{bmatrix}, \quad w = e^{-j2\pi/N} \quad (2.8.4)$$

称为 (原始) Fourier 矩阵，其  $(i, k)$  元素为  $F(i, k) = w^{(i-1)(k-1)}$ 。

显然，Fourier 矩阵的每一行和每一列的元素都分别组成各自的等比序列，是一种具有特殊结构的  $N \times N$  维 Vandermonde 矩阵。

另由定义易知，Fourier 矩阵为对称矩阵，即  $F^T = F$ 。

式 (2.8.3) 表明，一个离散时间信号向量的离散 Fourier 变换可以用矩阵  $F$  表示。这就是为什么称矩阵  $F$  为 Fourier 矩阵的缘故。

根据定义容易验证  $F^H F = F F^H = NI$ 。注意到 Fourier 矩阵是一个  $N \times N$  特殊 Vandermonde 矩阵，它是非奇异的。于是，由  $F^H F = NI$  知，Fourier 矩阵的逆矩阵

$$F^{-1} = \frac{1}{N} F^H = \frac{1}{N} F^* \quad (2.8.5)$$

因此，由式 (2.8.3) 立即有

$$x = F^{-1} \hat{x} = \frac{1}{N} F^* \hat{x} \quad (2.8.6)$$

或写作

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix} = \frac{1}{N} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w^* & \cdots & (w^{N-1})^* \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (w^{N-1})^* & \cdots & (w^{(N-1)(N-1)})^* \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix} \quad (2.8.7)$$

即有

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j2\pi n k / N}, \quad n = 0, 1, \dots, N-1 \quad (2.8.8)$$

这恰好就是离散 Fourier 逆变换的公式。

根据定义易知,  $n \times n$  阶 Fourier 矩阵具有以下性质 [324]:

(1) Fourier 矩阵为对称矩阵, 即  $\mathbf{F}^T = \mathbf{F}$ 。

(2) Fourier 矩阵的逆矩阵  $\mathbf{F}^{-1} = \frac{1}{N} \mathbf{F}^*$ 。

(3)  $\mathbf{F}^2 = \mathbf{P} = [\mathbf{e}_1, \mathbf{e}_n, \mathbf{e}_{n-1}, \dots, \mathbf{e}_2]$  (置换矩阵), 其中,  $\mathbf{e}_k$  是标准向量 (仅第  $k$  个元素为 1, 其他元素皆为 0 的向量)。

(4)  $\mathbf{F}^4 = \mathbf{I}$ 。

(5) 令  $\sqrt{n}\mathbf{F} = \mathbf{C} + j\mathbf{S}$ , 则  $\mathbf{C}\mathbf{S} = \mathbf{S}\mathbf{C}$  和  $\mathbf{C}^2 + \mathbf{S}^2 = \mathbf{I}$ , 且矩阵  $\mathbf{C}$  和  $\mathbf{S}$  的元素

$$C_{ij} = \cos\left(\frac{2\pi}{n}(i-1)(j-1)\right)$$

$$S_{ij} = \sin\left(\frac{2\pi}{n}(i-1)(j-1)\right)$$

式中,  $i, j = 1, 2, \dots, n$ 。

问题是, 无论利用式 (2.8.3) 计算离散 Fourier 变换, 还是使用式 (2.8.6) 计算离散 Fourier 逆变换, 都希望有快速算法。

下面考虑离散 Fourier 变换的快速算法——快速 Fourier 变换 (FFT) 算法。为此, 我们先来考虑  $2^n \times 2^n$  方程  $\mathbf{A}\mathbf{x} = \hat{\mathbf{x}}$  的计算, 其中  $\mathbf{A} \in \mathbb{C}^{2^n \times 2^n}$  为变换矩阵, 而  $\mathbf{x} \in \mathbb{C}^{2^n}$  和  $\hat{\mathbf{x}} \in \mathbb{C}^{2^n}$  分别为输入和输出向量。

## 2.8.2 适定方程计算的初等行变换方法

为方便计, 对于  $N \times N$  (其中  $N = 2^n$ ) 方程  $\mathbf{A}\mathbf{x} = \hat{\mathbf{x}}$ , 记

$$\mathbf{A} = \begin{bmatrix} \mathbf{b}_0 \\ \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{N-1} \end{bmatrix}, \quad \mathbf{x} = [x_0, x_1, \dots, x_{N-1}]^T, \quad \hat{\mathbf{x}} = [\hat{x}_0, \hat{x}_1, \dots, \hat{x}_{N-1}]^T$$

考虑对方程  $\mathbf{A}\mathbf{x} = \hat{\mathbf{x}}$  进行一种简单的初等行变换: 只对增广矩阵  $[\mathbf{A}, \hat{\mathbf{x}}]$  的行向量的位置进行重新排列。显然, 这种初等行变换将使得变换矩阵  $\mathbf{A}$  的行和输出向量  $\hat{\mathbf{x}}$  的元素之间的下标排列完全相同, 但不会改变输入向量  $\mathbf{x}$  的元素的下标排列。

为了描述这一行重排的效果, 考虑将变换矩阵  $A$  的行向量的下标用指标向量 (index vector) 表示<sup>[405]</sup>

$$\mathbf{i} = \begin{bmatrix} \langle 0 \rangle \\ \langle 1 \rangle \\ \vdots \\ \langle N-1 \rangle \end{bmatrix}, \quad N = 2^n \quad (2.8.9)$$

其中  $\langle i \rangle$  是变换矩阵  $A$  的第  $i+1$  (其中  $i = 0, 1, \dots, N-1$ ) 行向量下标的二进制表示。

二进制码的 Kronecker 积定义为二进制码的顺序排列。若  $a, b, c, d$  均为二进制表示, 则二进制 Kronecker 积定义为

$$\begin{bmatrix} a \\ b \end{bmatrix}_2 \otimes \begin{bmatrix} c \\ d \end{bmatrix}_2 = \begin{bmatrix} ac \\ ad \\ bc \\ bd \end{bmatrix}_2 \quad (2.8.10)$$

式中,  $xy$  表示两个二进制码的顺序排列, 而非乘积。

例如

$$\mathbf{i}_4 = \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} = \begin{bmatrix} 0_1 0_0 \\ 0_1 1_0 \\ 1_1 0_0 \\ 1_1 1_0 \end{bmatrix} = \begin{bmatrix} 00 \\ 01 \\ 10 \\ 11 \end{bmatrix} = \begin{bmatrix} \langle 0 \rangle \\ \langle 1 \rangle \\ \langle 2 \rangle \\ \langle 3 \rangle \end{bmatrix} \quad (2.8.11)$$

和

$$\mathbf{i}_8 = \begin{bmatrix} 0_2 \\ 1_2 \end{bmatrix} \otimes \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} = \begin{bmatrix} 0_2 \\ 0_1 \\ 1_2 \\ 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_1 0_0 \\ 0_1 1_0 \\ 1_1 0_0 \\ 1_1 1_0 \end{bmatrix} = \begin{bmatrix} 0_2 0_1 0_0 \\ 0_2 0_1 1_0 \\ 0_2 1_1 0_0 \\ 0_2 1_1 1_0 \\ 1_2 0_1 0_0 \\ 1_2 0_1 1_0 \\ 1_2 1_1 0_0 \\ 1_2 1_1 1_0 \end{bmatrix} = \begin{bmatrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{bmatrix} = \begin{bmatrix} \langle 0 \rangle \\ \langle 1 \rangle \\ \langle 2 \rangle \\ \langle 3 \rangle \\ \langle 4 \rangle \\ \langle 5 \rangle \\ \langle 6 \rangle \\ \langle 7 \rangle \end{bmatrix} \quad (2.8.12)$$

分别表示  $4 \times 4$  和  $8 \times 8$  变换矩阵  $A$  的行的 (原始) 指标向量, 同时也分别是  $4 \times 1$  和  $8 \times 1$  输入向量  $x$  的元素的 (原始) 指标向量。

更一般地, 若指标向量采用二进制的 Kronecker 积递推计算

$$\mathbf{i}_N = \begin{bmatrix} 0_{n-1} \\ 1_{n-1} \end{bmatrix} \otimes \begin{bmatrix} 0_{n-2} \\ 1_{n-2} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} = \begin{bmatrix} 0_{n-1} \cdots 0_1 0_0 \\ 0_{n-1} \cdots 0_1 1_0 \\ \vdots \\ 1_{n-1} \cdots 1_1 0_0 \\ 1_{n-1} \cdots 1_1 1_0 \end{bmatrix} = \begin{bmatrix} \langle 0 \rangle \\ \langle 1 \rangle \\ \vdots \\ \langle N-2 \rangle \\ \langle N-1 \rangle \end{bmatrix} \quad (2.8.13)$$

其中, 每个 0 或 1 的下标表示相应的二进制位置, 则指标向量  $\mathbf{i}_N$  既是  $N \times N$  变换矩阵  $A$  的行向量下标的正常次序排列, 也是  $N \times 1$  输入向量  $x = [x_0, x_1, \dots, x_{N-1}]^T$  的下标的正常顺序排列。

反之, 如果定义指标向量的二进制 Kronecker 积表示为

$$\mathbf{i}_{N,\text{rev}} = \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} \otimes \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0_{n-1} \\ 1_{n-1} \end{bmatrix} \quad (2.8.14)$$

则它是指标向量  $i_N$  的元素的反转二进制码序 (bit-reversed order)。换言之，反转指标向量  $i_{N,\text{rev}}$  表示变换矩阵  $A$  的行的下标的二进制码的反转结果。例如，1000 是 0001 的反转。同时， $i_{N,\text{rev}}$  也是输出向量  $\hat{x}$  的元素的二进制码的反转。

例如

$$i_{4,\text{rev}} = \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} \otimes \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} = \begin{bmatrix} 0_0 0_1 \\ 0_0 1_1 \\ 1_0 0_1 \\ 1_0 1_1 \end{bmatrix} = \begin{bmatrix} 00 \\ 10 \\ 01 \\ 11 \end{bmatrix} = \begin{bmatrix} \langle 0 \rangle \\ \langle 2 \rangle \\ \langle 1 \rangle \\ \langle 3 \rangle \end{bmatrix} \quad (2.8.15)$$

是指标向量  $i_4$  的元素的反转二进制码序，而

$$i_{8,\text{rev}} = \begin{bmatrix} 0_0 \\ 1_0 \end{bmatrix} \otimes \begin{bmatrix} 0_1 \\ 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_2 \\ 1_2 \end{bmatrix} = \begin{bmatrix} 0_0 0_1 \\ 0_0 1_1 \\ 1_0 0_1 \\ 1_0 1_1 \end{bmatrix} \otimes \begin{bmatrix} 0_2 \\ 1_2 \end{bmatrix} = \begin{bmatrix} 000 \\ 100 \\ 010 \\ 110 \\ 001 \\ 101 \\ 011 \\ 111 \end{bmatrix} = \begin{bmatrix} \langle 0 \rangle \\ \langle 4 \rangle \\ \langle 2 \rangle \\ \langle 6 \rangle \\ \langle 1 \rangle \\ \langle 5 \rangle \\ \langle 3 \rangle \\ \langle 7 \rangle \end{bmatrix} \quad (2.8.16)$$

则是指标向量  $i_8$  的元素的反转二进制码序。注意，二进制 Kronecker 积的中间结果应该按照二进制码的习惯顺序书写，例如  $0_0 0_1 1_2$  应该写成 100。

### 2.8.3 FFT 算法的推导

重要的是，当变换矩阵为 Fourier 矩阵时，按照反转指标向量  $i_{N,\text{rev}}$  对  $\hat{x} = Fx$  进行初等行变换，很容易得到 FFT 算法。为此，需要利用广义 Kronecker 积对  $N \times N$  原始 Fourier 矩阵  $F$  进行改写。

**例 2.8.1** 考虑原始  $4 \times 4$  Fourier 矩阵

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{-j\pi/2} & e^{-j\pi} & e^{-j3\pi/2} \\ 1 & e^{-j\pi} & e^{-j2\pi/2} & e^{-j3\pi} \\ 1 & e^{-j3\pi/2} & e^{-j3\pi} & e^{-j9\pi/2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \quad (2.8.17)$$

令

$$\{A\}_2 = \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \end{bmatrix} \right\}, \quad B = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.8.18)$$

则

$$F_{4,\text{rev}} = \{A\}_2 \otimes B = \left[ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \end{bmatrix} \otimes [1, 1] \quad \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \\ 1 & j & -1 & -j \end{bmatrix} \right] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \\ 1 & j & -1 & -j \end{bmatrix} \quad (2.8.19)$$

恰好是对原始  $(4 \times 4)$  Fourier 矩阵  $F_4$  的行向量按照反转指标向量  $i_{4,\text{rev}}$  进行初等行变换的结果。

例 2.8.2 令

$$\{\mathbf{A}\}_4 = \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \\ 1 & e^{-j\pi/4} \\ 1 & -e^{-j\pi/4} \\ 1 & e^{-j3\pi/4} \\ 1 & -e^{-j3\pi/4} \end{bmatrix} \right\} \quad (2.8.20)$$

于是, 可得  $(8 \times 8)$  Fourier 矩阵

$$\begin{aligned} \mathbf{F}_{8,\text{rev}} &= \{\mathbf{A}\}_4 \otimes (\{\mathbf{A}\}_2 \otimes \mathbf{B}) = \{\mathbf{A}\}_4 \otimes \mathbf{F}_4 \\ &= \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & -j \\ 1 & j \\ 1 & e^{-j\pi/4} \\ 1 & -e^{-j\pi/4} \\ 1 & e^{-j3\pi/4} \\ 1 & -e^{-j3\pi/4} \end{bmatrix} \right\} \otimes \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \\ 1 & j & -1 & -j \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j & 1 & -j & -1 & j \\ 1 & j & -1 & -j & 1 & j & -1 & -j \\ 1 & e^{-j\pi/4} & -j & e^{-j3\pi/4} & -1 & -e^{-j\pi/4} & j & -e^{-j3\pi/4} \\ 1 & -e^{-j\pi/4} & -j & -e^{-j3\pi/4} & -1 & e^{-j\pi/4} & j & e^{-j3\pi/4} \\ 1 & e^{-j3\pi/4} & j & e^{-j\pi/4} & -1 & -e^{-j3\pi/4} & -j & -e^{-j\pi/4} \\ 1 & -e^{-j3\pi/4} & j & -e^{-j\pi/4} & -1 & e^{-j3\pi/4} & -j & e^{-j\pi/4} \end{bmatrix} \end{aligned} \quad (2.8.21)$$

正好是  $(8 \times 8)$  原始 Fourier 矩阵  $\mathbf{F}_8$  的行向量按照反转指标向量  $i_{8,\text{rev}}$  进行初等行变换的结果。

更一般地,  $(N \times N)$  Fourier 矩阵的新形式可以递推构造

$$\mathbf{F}_{N,\text{rev}} = \{\mathbf{A}\}_{N/2} \otimes \mathbf{F}_{N/2}^{\text{new}} = \{\mathbf{A}\}_{N/2} \otimes \{\mathbf{A}\}_{N/4} \otimes \cdots \otimes \{\mathbf{A}\}_2 \otimes \mathbf{B} \quad (2.8.22)$$

其中, 矩阵组  $\{\mathbf{A}\}_2$  和  $2 \times 2$  矩阵  $\mathbf{B}$  由式 (2.8.18) 给出, 并且

$$\{\mathbf{A}\}_{N/2} = \left\{ \begin{array}{c} \{\mathbf{A}\}_{N/4} \\ \{\mathbf{R}\} \end{array} \right\}, \quad \{\mathbf{R}\} = \left\{ \begin{array}{c} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \vdots \\ \mathbf{R}_{N/2} \end{array} \right\} \quad (2.8.23)$$

其中

$$\mathbf{R}_k = \begin{bmatrix} 1 & e^{-j(2k-1)\pi/N} \\ 1 & -e^{-j(2k-1)\pi/N} \end{bmatrix}, \quad k = 1, 2, \dots, \frac{N}{2} \quad (2.8.24)$$

$\mathbf{F}_{N,\text{rev}}$  是由式(2.8.4)定义的 $(N \times N)$ 原始Fourier矩阵 $\mathbf{F}_N$ 的行向量按照反转指标向量 $i_{N,\text{rev}}$ 进行初等行变换的结果。于是，经过初等行变换后，输出向量

$$\hat{\mathbf{x}}_{\text{rev}} = \mathbf{F}_{N,\text{rev}} \mathbf{x} \quad (2.8.25)$$

的下标服从反转指标向量 $i_{N,\text{rev}}$ 的排列规则。例如，输入向量 $\mathbf{x}$ 的4位DFT为

$$\begin{bmatrix} X_0 \\ X_2 \\ X_1 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (2.8.26)$$

而8位DFT为

$$\begin{bmatrix} X_0 \\ X_4 \\ X_2 \\ X_6 \\ X_1 \\ X_5 \\ X_3 \\ X_7 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j & 1 & -j & -1 & j \\ 1 & j & -1 & -j & 1 & j & -1 & -j \\ 1 & e^{-j\pi/4} & -j & e^{-j3\pi/4} & -1 & -e^{-j\pi/4} & j & -e^{-j3\pi/4} \\ 1 & -e^{-j\pi/4} & -j & -e^{-j3\pi/4} & -1 & e^{-j\pi/4} & j & e^{-j3\pi/4} \\ 1 & e^{-j3\pi/4} & j & e^{-j\pi/4} & -1 & -e^{-j3\pi/4} & -j & -e^{-j\pi/4} \\ 1 & -e^{-j3\pi/4} & j & -e^{-j\pi/4} & -1 & e^{-j3\pi/4} & -j & e^{-j\pi/4} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} \quad (2.8.27)$$

显然，输出和输入的序号之间存在二进制码位的反转关系。这与FFT算法的结果是完全一致的。

对于Fourier逆变换 $\mathbf{x} = \mathbf{F}^{-1}\hat{\mathbf{x}}$ ，由于 $\mathbf{F}^{-1} = \frac{1}{N}\mathbf{F}^*$ ，故由式(2.8.22)知

$$\mathbf{F}_N^{-1} = \frac{1}{N}\{\bar{\mathbf{A}}\}_{N/2} \otimes \mathbf{F}_{N/2}^{-1} = \frac{1}{N}\{\bar{\mathbf{A}}\}_{N/2} \otimes \{\bar{\mathbf{A}}\}_{N/4} \otimes \cdots \otimes \{\bar{\mathbf{A}}\}_2 \otimes \mathbf{B} \quad (2.8.28)$$

式中

$$\{\bar{\mathbf{A}}\}_{N/2} = \left\{ \begin{array}{l} \{\bar{\mathbf{A}}\}_{N/4} \\ \{\bar{\mathbf{R}}\} \end{array} \right\}, \quad \{\bar{\mathbf{R}}\} = \left\{ \begin{array}{l} \bar{\mathbf{R}}_1 \\ \bar{\mathbf{R}}_2 \\ \vdots \\ \bar{\mathbf{R}}_{N/2} \end{array} \right\} \quad (2.8.29)$$

其中

$$\{\bar{\mathbf{A}}\}_2 = \{\mathbf{A}^*\}_2 = \left\{ \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & j \\ 1 & -j \end{bmatrix} \right\}, \quad \mathbf{B} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.8.30)$$

$$\bar{\mathbf{R}}_k = \mathbf{R}_k^* = \begin{bmatrix} 1 & e^{j(2k-1)\pi/N} \\ 1 & -e^{j(2k-1)\pi/N} \end{bmatrix}, \quad k = 1, 2, \dots, \frac{N}{2} \quad (2.8.31)$$

一个 $n \times n$ 矩阵

$$\mathbf{C}_n = \begin{bmatrix} c_0 & c_{-1} & \cdots & c_{1-n} \\ c_1 & c_0 & \cdots & c_{2-n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n-1} & c_{n-2} & \cdots & c_0 \end{bmatrix} \quad (2.8.32)$$

称为循环矩阵，式中， $c_{-k} = c_{n-k}$ ,  $k = 1, 2, \dots, n-1$ 。有趣的是，这类循环矩阵可以被 Fourier 矩阵对角化，即有 [99, 127]

$$\mathbf{C}_n = \mathbf{F}_n^H \mathbf{A}_n \mathbf{F}_n \quad (2.8.33)$$

式中， $n \times n$  维 Fourier 矩阵  $\mathbf{F}_n$  的元素为

$$[\mathbf{F}_n]_{ik} = \frac{1}{\sqrt{n}} e^{j2\pi ik/n}, \quad 0 \leq i, k \leq n-1 \quad (2.8.34)$$

且  $\mathbf{A}_n$  是一个  $n \times n$  对角矩阵，其对角元素为循环矩阵  $\mathbf{C}_n$  的特征值。值得指出的是，循环矩阵  $\mathbf{C}_n$  的特征值  $\lambda_1, \lambda_2, \dots, \lambda_n$  可以利用  $\mathbf{C}_n$  的第 1 列元素  $c_0, c_1, \dots, c_{n-1}$  的离散 Fourier 变换得到，即有 [99]

$$\lambda_k = \sum_{i=0}^{n-1} c_i e^{j2\pi ik/n}, \quad k = 0, 1, \dots, n-1 \quad (2.8.35)$$

并且这一运算可以利用快速 Fourier 变换 (FFT) 实现。

式 (2.8.3) 和式 (2.8.8) 一起组成非对称形式的 Fourier 变换对。

在有些文献 (例如文献 [324]) 中，Fourier 矩阵定义为

$$\mathbf{F} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & w & \cdots & w^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & w^{N-1} & \cdots & w^{(N-1)(N-1)} \end{bmatrix}, \quad w = e^{-j2\pi/N} \quad (2.8.36)$$

此时，Fourier 矩阵的逆矩阵  $\mathbf{F}^{-1} = \mathbf{F}^*$ ，并且离散 Fourier 变换对取以下对称形式

$$\hat{x}(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}, \quad k = 0, 1, \dots, N-1 \quad (2.8.37)$$

$$x(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \hat{x}(k) e^{j2\pi nk/N}, \quad n = 0, 1, \dots, N-1 \quad (2.8.38)$$

## 2.9 Hadamard 矩阵

Hadamard 矩阵是在通信、信息论和信号处理中一种重要的特殊矩阵。

**定义 2.9.1**  $\mathbf{H}_n \in \mathbb{R}^{n \times n}$  称为 Hadamard 矩阵，若它的所有元素取 +1 或者 -1，且

$$\mathbf{H}_n \mathbf{H}_n^T = \mathbf{H}_n^T \mathbf{H}_n = n \mathbf{I}_n \quad (2.9.1)$$

Hadamard 矩阵的性质如下。

(1) 观察知，用 -1 乘 Hadamard 矩阵的任意一行或者任意一列的元素，得到的结果仍然为一 Hadamard 矩阵。于是，可以得到第 1 列和第 1 行的所有元素为 +1 的 Hadamard 矩阵，并称为规范化 Hadamard 矩阵。

(2) 只有当  $n = 2$  或者  $n$  是 4 的整数倍时, Hadamard 矩阵才存在。

(3) 容易验证  $\frac{1}{\sqrt{n}} \mathbf{H}_n$  为标准正交矩阵。

(4)  $n \times n$  Hadamard 矩阵  $\mathbf{H}_n$  的行列式  $\det(\mathbf{H}_n) = n^{n/2}$ 。

对 Hadamard 矩阵进行规范化, 将大大方便高维数的 Hadamard 矩阵的构造。下面的定理给出了规范化的标准正交 Hadamard 矩阵的一种通用构造方法。

**定理 2.9.1** 令  $n = 2^k$ ,  $k = 1, 2, \dots$ , 则规范化的标准正交 Hadamard 矩阵具有通用构造公式

$$\bar{\mathbf{H}}_n = \frac{1}{\sqrt{2}} \begin{bmatrix} \bar{\mathbf{H}}_{n/2} & \bar{\mathbf{H}}_{n/2} \\ \bar{\mathbf{H}}_{n/2} & -\bar{\mathbf{H}}_{n/2} \end{bmatrix} \quad (2.9.2)$$

其中

$$\bar{\mathbf{H}}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.9.3)$$

**证明** 用数学归纳法证明。显然,  $\bar{\mathbf{H}}_2$  是规范化的正交 Hadamard 矩阵, 因为容易验证  $\bar{\mathbf{H}}_2^T \bar{\mathbf{H}}_2 = \bar{\mathbf{H}}_2 \bar{\mathbf{H}}_2^T = \mathbf{I}_2$ 。假设  $n = 2^k$  时  $\bar{\mathbf{H}}_{2^k}$  是规范化的正交 Hadamard 矩阵, 即有  $\bar{\mathbf{H}}_{2^k}^T \bar{\mathbf{H}}_{2^k} = \bar{\mathbf{H}}_{2^k} \bar{\mathbf{H}}_{2^k}^T = \mathbf{I}_{2^k \times 2^k}$ 。于是, 对于  $n = 2^{k+1}$ , 容易看出

$$\bar{\mathbf{H}}_{2^{k+1}} = \frac{1}{\sqrt{2}} \begin{bmatrix} \bar{\mathbf{H}}_{2^k} & \bar{\mathbf{H}}_{2^k} \\ \bar{\mathbf{H}}_{2^k} & -\bar{\mathbf{H}}_{2^k} \end{bmatrix}$$

满足正交条件, 即

$$\bar{\mathbf{H}}_{2^{k+1}}^T \bar{\mathbf{H}}_{2^{k+1}} = \frac{1}{2} \begin{bmatrix} \bar{\mathbf{H}}_{2^k}^T & \bar{\mathbf{H}}_{2^k}^T \\ \bar{\mathbf{H}}_{2^k}^T & -\bar{\mathbf{H}}_{2^k}^T \end{bmatrix} \begin{bmatrix} \bar{\mathbf{H}}_{2^k} & \bar{\mathbf{H}}_{2^k} \\ \bar{\mathbf{H}}_{2^k} & -\bar{\mathbf{H}}_{2^k} \end{bmatrix} = \mathbf{I}_{2^{k+1} \times 2^{k+1}}$$

类似地, 容易证明  $\bar{\mathbf{H}}_{2^{k+1}} \bar{\mathbf{H}}_{2^{k+1}}^T = \mathbf{I}_{2^{k+1} \times 2^{k+1}}$ 。另外, 由于  $\bar{\mathbf{H}}_{2^k}$  是规范化的, 所以  $\bar{\mathbf{H}}_{2^{k+1}}$  也是规范化的。因此, 定理对于  $n = 2^{k+1}$  也成立。 ■

非规范化的 Hadamard 矩阵可以利用矩阵的 Kronecker 积写成

$$\mathbf{H}_n = \mathbf{H}_{n/2} \otimes \mathbf{H}_2 = \mathbf{H}_2 \otimes \cdots \otimes \mathbf{H}_2 \quad (n = 2^k) \quad (2.9.4)$$

共  $k$  个  $2 \times 2$  非规范化的 Hadamard 矩阵  $\mathbf{H}_2$  的 Kronecker 积, 其中

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.9.5)$$

显然, 规范化和非规范化的 Hadamard 积之间存在以下关系

$$\bar{\mathbf{H}}_n = \frac{1}{\sqrt{n}} \mathbf{H}_n \quad (2.9.6)$$

例 2.9.1 当  $n = 2^3 = 8$  时, Hadamard 矩阵

$$\begin{aligned} \mathbf{H}_8 &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \end{aligned}$$

容易看出, Hadamard 矩阵的每一行都是区间  $(0, 1)$  上的分段线性函数。如果用  $\phi_0(t), \phi_1(t), \dots, \phi_7(t)$  分别表示 Hadamard 矩阵第 1~8 行的波形函数, 则如图 2.9.1 所示。

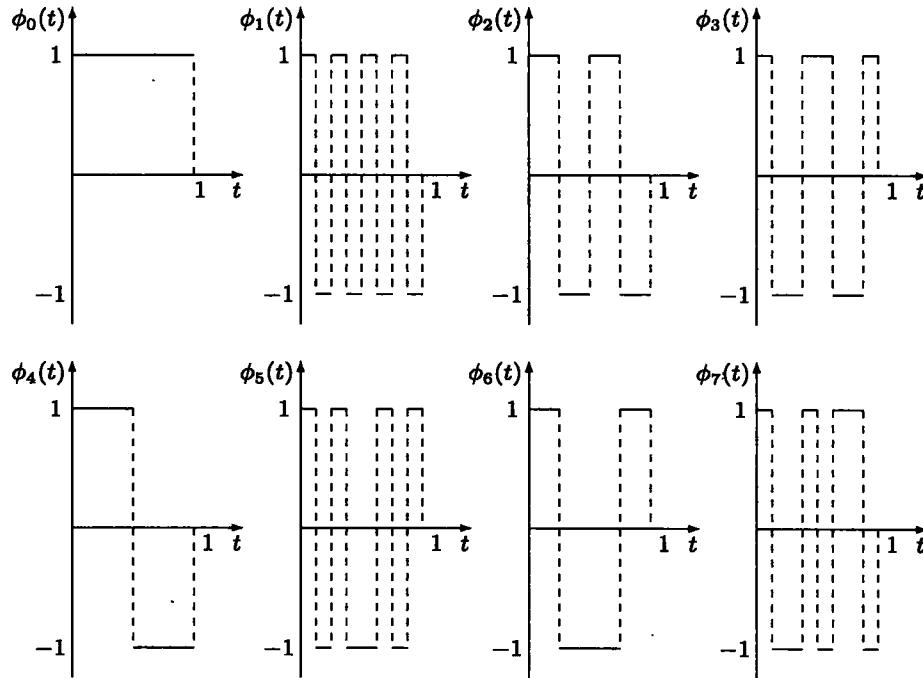


图 2.9.1 Hadamard 矩阵每行的波形

由图 2.9.1 容易看出,  $\phi_0(t), \phi_1(t), \dots, \phi_7(t)$  这八个矩形脉冲函数相互正交, 即

$$\int_0^1 \phi_i(t) \phi_j(t) dt = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (2.9.7)$$

修正 Walsh-Hadamard 变换矩阵也可以利用广义 Kronecker 积递推计算<sup>[405]</sup>

$$\mathbf{R}_N = \{\mathbf{B}\}_{N/2} \otimes \mathbf{R}_{N/2}, \quad \mathbf{R}_1 = 1 \quad (2.9.8)$$

其中, 矩阵组  $\{\mathbf{B}\}_{N/2}$  的第  $i$  个矩阵

$$\mathbf{B}_i = \begin{cases} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, & i=0 \\ \sqrt{2}\mathbf{I}_2, & \text{其他} \end{cases} \quad (2.9.9)$$

当  $\mathbf{H}$  为 Hadamard 矩阵时, 线性变换  $\mathbf{Y} = \mathbf{H}\mathbf{X}$  称为矩阵  $\mathbf{X}$  的 Hadamard 变换。由于 Hadamard 矩阵是规范化的标准正交矩阵, 并且元素只取 +1 或 -1, 故 Hadamard 矩阵是唯一只使用加法和减法的标准正交变换。Hadamard 矩阵可以用作移动通信中的编码, 得到的码称为 Hadamard 码(或称 Walsh-Hadamard 码)。另外, 由于 Hadamard 矩阵的行向量之间的正交性, 行向量可以用来仿真码分多址中各个用户的扩频波形向量。

## 2.10 Toeplitz 矩阵

20世纪初, Toeplitz 在研究与 Laurent 级数有关的双线性函数的一篇论文<sup>[475]</sup>中, 提出了一种具有特殊结构的矩阵: 其任何一条对角线的元素取相同值, 即

$$\mathbf{A} = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & \cdots & a_{-n} \\ a_1 & a_0 & a_{-1} & \cdots & a_{-n+1} \\ a_2 & a_1 & a_0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & a_{-1} \\ a_n & a_{n-1} & \cdots & a_1 & a_0 \end{bmatrix} = [a_{i-j}]_{i,j=0}^n \quad (2.10.1)$$

这种形式取  $\mathbf{A} = [a_{i-j}]_{i,j=0}^n$  的矩阵称为 Toeplitz 矩阵。显然, 一个  $(n+1) \times (n+1)$  Toeplitz 矩阵由其第一行元素  $a_0, a_{-1}, \dots, a_{-n}$  和第一列元素  $a_0, a_1, \dots, a_n$  完全确定。

### 2.10.1 对称 Toeplitz 矩阵

最常见的 Toeplitz 矩阵为对称 Toeplitz 矩阵  $\mathbf{A} = [a_{|i-j|}]_{i,j=0}^n$ , 即其元素还满足对称关系  $a_{-i} = a_i, i = 1, 2, \dots, n$ 。可见, 对称 Toeplitz 矩阵仅由其第 1 行元素就可以完全描述。因此, 常将  $(n+1) \times (n+1)$  对称 Toeplitz 矩阵  $\mathbf{A}$  简记作  $\mathbf{A} = \text{Toep}[a_0, a_1, \dots, a_n]$ 。

若一个复 Toeplitz 矩阵的元素满足复共轭对称关系  $a_{-i} = a_i^*$ , 即

$$\mathbf{A} = \begin{bmatrix} a_0 & a_1^* & a_2^* & \cdots & a_n^* \\ a_1 & a_0 & a_1^* & \cdots & a_{n-1}^* \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_2 & a_1 & a_0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & a_1^* \\ a_n & a_{n-1} & \cdots & a_1 & a_0 \end{bmatrix} \quad (2.10.2)$$

则称为 Hermitian Toeplitz 矩阵。特别地，具有特殊结构

$$\mathbf{A}_S = \begin{bmatrix} 0 & -a_1^* & -a_2^* & \cdots & -a_n^* \\ a_1 & 0 & -a_1^* & \cdots & -a_{n-1}^* \\ a_2 & a_1 & 0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -a_1^* \\ a_n & a_{n-1} & \cdots & a_1 & 0 \end{bmatrix} \quad (2.10.3)$$

的  $(n+1) \times (n+1)$  维 Toeplitz 矩阵称为斜 Hermitian Toeplitz 矩阵；而

$$\mathbf{A} = \begin{bmatrix} a_0 & -a_1^* & -a_2^* & \cdots & -a_n^* \\ a_1 & a_0 & -a_1^* & \cdots & -a_{n-1}^* \\ a_2 & a_1 & a_0 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -a_1^* \\ a_n & a_{n-1} & \cdots & a_1 & a_0 \end{bmatrix} \quad (2.10.4)$$

称为斜 Hermitian 型 Toeplitz 矩阵。

下面的定理给出了对称 Toeplitz 矩阵半正定性的一种简单检验方法，它不需要计算任何主子式。

**定理 2.10.1** [330] 令  $\mathbf{R}_p = r_{|i-j|}, i, j = 0, \dots, p$  是一个对称 Toeplitz 矩阵。若  $m$  是满足  $\mathbf{R}_{m-1}$  正定和  $D_m = 0$  条件的最小正整数，则矩阵  $\mathbf{R}_p (p \geq m)$  是半正定的，当且仅当系数  $\{r_i, i > m\}$  服从递归方程

$$r_i = - \sum_{k=1}^m a_m(k) r_{i-k}, \quad i = m+1, m+2, \dots, p \quad (2.10.5)$$

式中， $\{a_m(k)\}, 1 \leq k \leq m$  为  $m$  阶自回归 (autoregressive) 模型 AR( $m$ ) 的系数。

Toeplitz 矩阵具有以下性质 [413]：

- (1) Toeplitz 矩阵的线性组合仍然为 Toeplitz 矩阵。
- (2) 若 Toeplitz 矩阵  $\mathbf{A}$  的元素  $a_{ij} = a_{|i-j|}$ ，则  $\mathbf{A}$  为对称 Toeplitz 矩阵。
- (3) Toeplitz 矩阵  $\mathbf{A}$  的转置  $\mathbf{A}^T$  仍然为 Toeplitz 矩阵。
- (4) Toeplitz 矩阵的元素相对于交叉对角线对称。

在统计信号处理和其他相关领域，经常需要求解线性方程组  $\mathbf{Ax} = \mathbf{b}$ ，其中，系数矩阵  $\mathbf{A}$  为对称 Toeplitz 矩阵。这类方程称为 Toeplitz 线性方程组。

利用 Toeplitz 矩阵的特殊结构，可以得到求解 Toeplitz 线性方程组的一类 Levinson 递推算法。对于实的正定 Toeplitz 矩阵，其预测多项式 (Levinson 多项式) 的经典 Levinson 递推 [312] 在计算上存在很大的冗余。为了减少计算冗余，Delsarte 与 Genin 提出了一种分基 Levinson 算法 [132]。随后，他们又提出了分基 Schur 算法 [133]。后来，Krishna 和 Morgera [285] 将分基 Levinson 递推从实数 Toeplitz 线性方程组推广到复数 Toeplitz 线性方程组。

虽然这些算法是递推的，但它们的计算复杂度为  $O(n^2)$ 。除了 Levinson 递推外，也可以利用快速 Fourier 变换 (FFT) 求解 Toeplitz 线性方程组，而且这类算法只需要  $O(n \log_2 n)$  的计算复杂度，比 Levinson 递推更快速。鉴于此，多数文献称这种求解 Toeplitz 线性方程组的快速 Fourier 变换为快速算法（如 Kumar 算法 [290] 和 Davis 算法 [128] 等），个别文献 [18] 称这类算法为超快速算法。

Kumar 算法的计算复杂度为  $O(n \log^2 n)$ 。特别地，文献 [99] 中求解 Toeplitz 方程组的共轭梯度算法只需要  $O(n \log n)$  的计算复杂度，比任何现有的直接方法都快。

### 2.10.2 Toeplitz 矩阵的离散余弦变换

$N$  阶离散余弦变换可以用一个  $N \times N$  矩阵  $\mathbf{T}$  表示，其中， $\mathbf{T} = [t_{m,l}]_{m,l=0}^{N-1}$  的元素定义为

$$t_{m,l} = \tau_m \cos \left[ \frac{\pi}{2N} m(2l+1) \right], \quad m, l = 0, 1, \dots, N-1 \quad (2.10.6)$$

且

$$\tau_m = \begin{cases} \sqrt{1/N}, & m = 0 \\ \sqrt{2/N}, & m = 1, 2, \dots, N-1 \end{cases} \quad (2.10.7)$$

根据上述定义易知， $\mathbf{T}^{-1} = \mathbf{T}^T$ ，即  $\mathbf{T}$  是正交矩阵。因此，任意矩阵  $\mathbf{A}$  的离散余弦变换矩阵  $\hat{\mathbf{A}} = \mathbf{T} \mathbf{A} \mathbf{T}^T$  与原矩阵  $\mathbf{A}$  具有相同的特征值。

特别地，我们考虑一  $N \times N$  实 Toeplitz 矩阵

$$\mathbf{A} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{N-1} \\ a_{-1} & a_0 & \cdots & a_{N-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{-N+1} & a_{-N+2} & \cdots & a_0 \end{bmatrix} = [a_{l,k}]_{l,k=0}^{N-1} = [a_{k-l}]_{l,k=0}^{N-1} \quad (2.10.8)$$

的离散余弦变换  $\hat{\mathbf{A}} = \mathbf{T} \mathbf{A} \mathbf{T}^T$ 。令  $\hat{a}_{m,n} = [\hat{a}_{m,n}]$ ， $c_{m,n} = \cos \left[ \frac{\pi}{2N} m(2n+1) \right]$ ，则

$$\hat{a}_{m,n} = \tau_m \left( \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} c_{m,l} a_{l,k} c_{n,k} \right) \tau_n \quad (2.10.9)$$

下面是计算 Toeplitz 矩阵的离散余弦变换的快速算法 [374]。

#### 算法 2.10.1 Toeplitz 矩阵的快速离散余弦变换

步骤 1 给定 Toeplitz 矩阵  $\mathbf{A} = [a_{l,k}]_{l,k=0}^{N-1} = [a_{k-l}]_{l,k=0}^{N-1}$ ，计算

$$x_{m,0} = \sum_{l=0}^{N-1} w^{m(2l+1)} a_{l,0}, \quad w = \exp \left( -j \frac{\pi}{2N} \right) \quad (2.10.10)$$

其中  $m = 0, 1, \dots, N-1$ 。（计算复杂度：一次  $2N$  点 DFT）

步骤 2 计算

$$v_1(n) = \sum_{k=0}^{N-1} w^{2nk} a_{k+1}, \quad v_2(n) = \sum_{k=0}^{N-1} w^{2nk} a_{1-N+k} \quad (2.10.11)$$

其中  $n = -(N-1), \dots, (N-1)$ 。(计算复杂度: 两次  $2N$  点 DFT)

### 步骤 3 计算

$$x_{m,N} = (-1)^m x_{m,0} + w^{-m}((-1)^m v_1(n) - v_2(n)) \quad (2.10.12)$$

其中  $m = 0, 1, \dots, N-1$ 。(计算复杂度:  $N$  次乘法运算)

### 步骤 4 计算

$$u_1(n) = \sum_{k=0}^{N-1} kw^{2nk} a_{k+1}, \quad u_2(n) = \sum_{k=0}^{N-1} kw^{2nk} a_{1-N+k} \quad (2.10.13)$$

其中  $u_2(n), n = -(N-1), \dots, (N-1)$ 。(计算复杂度: 两次  $2N$  点 DFT)

### 步骤 5 由

$$y_{m,n} = \frac{1}{w^{-2n} - w^{2m}} \{ x_{m,0} w^{-2n} - x_{m,N} w^{-2n} (-1)^n + w^m [v_1(n) - (-1)^m v_2(n)] \}$$

和

$$y_{m,-m} = x_{m,0} + (N-1)(-1)^m x_{m,N} - w^{-m} [u_1(-m) - (-1)^m u_2(-m)]$$

计算  $y_{m,n}$ , 其中  $m = 0, 1, \dots, N-1; n = -(N-1), \dots, (N-1)$ 。(计算复杂度: 对每个值只需要若干次乘法运算)

### 步骤 6 计算

$$\hat{a}_{m,n} = \tau_m \tau_n \operatorname{Re} \left[ w^n \frac{y_{m,n} + y_{m,-n}^*}{2} \right] \quad (2.10.14)$$

其中  $m = 0, 1, \dots, N-1; n = -(N-1), \dots, (N-1)$ 。(计算复杂度: 对每个值为若干次乘法运算)

计算 Toeplitz 矩阵  $A$  的特征值的通常做法是: 利用变换方法(如 Givens 旋转等)将  $A$  的非对角线元素转换成零, 即对  $A$  进行对角化。如前所述, 由于  $T$  为正交矩阵,  $\hat{A} = TAT^T$  与  $A$  具有相同的特征值, 所以在求  $A$  的特征值时可以对  $A$  的离散余弦变换  $\hat{A}$  实施对角化, 而且这比直接对角化  $A$  更好, 因为大多数的化简工作已经用离散余弦变换做过了。因此, 离散余弦变换可以用作一快速特征值预置条件器。况且, 在某些情况下, 变换后的矩阵  $\hat{A}$  已是  $A$  的足够精确的特征值估计<sup>[201]</sup>。

一些例子表明<sup>[374]</sup>, Toeplitz 矩阵  $A$  的快速余弦变换  $\hat{A}$  还可用作  $A$  的逼近, 因为  $\hat{A}$  的主要分量集中在一个小得多的矩阵分块里, 它相当于  $A$  的稳定部分。从秩的判断出发,  $\hat{A}$  的秩可明显看出, 而  $A$  的秩则不容易看出。注意, 由于  $A$  和  $\hat{A}$  具有相同的特征值, 所以二者的秩相同。

## 2.11 Hankel 矩阵

正方矩阵  $A \in \mathbb{C}^{(n+1) \times (n+1)}$  称为 Hankel 矩阵, 若

$$A = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_n \\ a_1 & a_2 & a_3 & \cdots & a_{n+1} \\ a_2 & a_3 & a_4 & \cdots & a_{n+2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_n & a_{n+1} & a_{n+2} & \cdots & a_{2n} \end{bmatrix} \quad (2.11.1)$$

显然, 只要序列  $a_0, a_1, \dots, a_{2n-1}, a_{2n}$  给定, Hankel 矩阵的一般项就由  $a_{ij} = a_{i+j-2}$  规定。事实上, Hankel 矩阵是一个交叉对角线上具有相同元素的矩阵。

假定给出了一系列复数  $s_0, s_1, s_2, \dots$ , 它们定义了一个无穷阶对称矩阵

$$S = \begin{bmatrix} s_0 & s_1 & s_2 & \cdots \\ s_1 & s_2 & s_3 & \cdots \\ s_2 & s_3 & s_4 & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (2.11.2)$$

称矩阵  $S$  为无穷阶 Hankel 矩阵, 并简记作  $S = [s_{i+k}]_0^\infty$ 。

下面的定理给出了无穷阶 Hankel 矩阵具有有限秩的充分必要条件。

**定理 2.11.1** <sup>[185]</sup> 无穷阶 Hankel 矩阵  $S = [s_{i+k}]_0^\infty$  具有有限秩  $r$ , 当且仅当存在  $r$  个常数  $\alpha_1, \alpha_2, \dots, \alpha_r$ , 使得

$$s_l = \sum_{i=1}^r \alpha_i s_{l-i}, \quad l = r, r+1, \dots \quad (2.11.3)$$

成立, 其中,  $r$  是具有该性质的最小整数。

**推论 2.11.1** 如果无穷阶 Hankel 矩阵  $S$  具有有限秩  $r$ , 则  $D_r = \det[s_{i+k}]_0^{r-1} \neq 0$ 。

事实上, 从关系式 (2.11.3) 可以得出结论: 矩阵  $S$  的任意行 (或列) 都是最前面  $r$  行 (或列) 的线性组合。因此, 阶数为  $r$  的任意余子式都可以用形式  $\alpha D_r$  表示, 其中,  $\alpha$  是某个常数。由此可得不等式  $D_r \neq 0$ 。注意, 对于一个秩  $r$  的有穷阶 Hankel 矩阵,  $D_r \neq 0$  有可能不成立。例如, 元素  $s_0 = s_1 = 0$ , 但  $s_2 \neq 0$  的矩阵

$$S_2 = \begin{bmatrix} s_0 & s_1 \\ s_1 & s_2 \end{bmatrix}$$

的秩等于 1, 但是同时  $D_1 = s_0 = 0$ 。

下面讨论无穷阶 Hankel 矩阵与有理式函数之间的联系。

假定存在一本征有理式函数  $R(z) = g(z)/h(z)$ , 其中

$$h(z) = a_0 z^m + a_1 z^{m-1} + \cdots + a_{m-1} z + a_m \quad (2.11.4)$$

$$g(z) = b_0 z^m + b_1 z^{m-1} + \cdots + b_{m-1} z + b_m \quad (2.11.5)$$

现在将函数  $R(z)$  写作  $z$  的负次幂的幂级数

$$R(z) = \frac{g(z)}{h(z)} = s_0 + s_1 z^{-1} + s_2 z^{-2} + \dots$$

如果函数  $R(z)$  的所有极点 (也就是满足  $R(z) \rightarrow \infty$  的所有  $z$  值) 都位于半径为  $a$  的圆内, 即  $|z| \leq a$ , 则上述级数对于  $|z| > a$  收敛。用分母  $h(z)$  同乘上式的两边, 得到

$$\begin{aligned} & (a_0 z^m + a_1 z^{m-1} + \dots + a_{m-1} z + a_m)(s_0 + s_1 z^{-1} + s_2 z^{-2} + \dots) \\ &= b_0 z^m + b_1 z^{m-1} + \dots + b_{m-1} z + b_m \end{aligned} \quad (2.11.6)$$

比较上式两边  $z$  的同次幂项的系数, 便得到下列的一组关系

$$\left. \begin{array}{l} a_0 s_0 = b_0 \\ a_0 s_1 + a_1 s_0 = b_1 \\ \vdots \\ a_0 s_m + a_1 s_{m-1} + \dots + a_m s_0 = b_m \end{array} \right\} \quad (2.11.7)$$

$$a_0 s_l + a_1 s_{l-1} + \dots + a_m s_{l-m} = 0, \quad l = m+1, m+2, \dots \quad (2.11.8)$$

令  $\alpha_i = -a_i/a_0$ ,  $i = m+1, m+2, \dots$ , 我们就可以用式 (2.11.3) 书写关系式 (2.11.8), 其中,  $r = m$ 。因此, 根据定理 2.11.1, 具有系数  $s_0, s_1, s_2, \dots$  的无穷 Hankel 矩阵  $S = [s_{i+k}]_0^\infty$  的秩为有限大 ( $\leq m$ )。

相反, 如果矩阵  $S$  具有有限的秩, 则式 (2.11.3) 成立, 这些方程可以用式 (2.11.8) 重写, 其中,  $m = r$ 。于是, 如果利用式 (2.11.7) 定义一组数  $b_0, b_1, \dots, b_m$ , 便得到关系式

$$\frac{b_0 z^m + b_1 z^{m-1} + \dots + b_m}{a_0 z^m + a_1 z^{m-1} + \dots + a_m} = s_0 + s_1 z^{-1} + s_2 z^{-2} + \dots$$

此关系式得以成立的分母最小阶数  $m$  就是式 (2.11.3) 成立的最小数  $m$ 。根据定理 2.11.1, 这个最小的  $m$  值等于矩阵  $S$  的秩。上述结果可以用下面的定理来表述。

**定理 2.11.2** <sup>[185]</sup> 矩阵  $S = [s_{i+k}]_0^\infty$  具有有限大的秩, 当且仅当级数

$$R(z) = s_0 + s_1 z^{-1} + s_2 z^{-2} + \dots$$

是变量  $z$  的有理式函数。当这种情况发生时, 矩阵  $S$  的秩等于函数  $R(z)$  的极点个数, 其中包括极点的多重度在内。

运用定理 2.11.2, 可以得到有关自回归-移动平均 (autoregressive moving average, AR-MA) 模型的一个重要结果。

**例 2.11.1** 令一线性时不变的因果 ARMA 过程由

$$\sum_{i=0}^p a(i)x(n-i) = \sum_{j=0}^q b(j)e(n-j) \quad (2.11.9)$$

产生, 其中,  $e(n)$  是一个激励白噪声序列。不失一般性, 假定  $a(0) = 1$ , 并且 MA 阶数  $q$  小于或等于 AR 阶数  $p$ , 即  $q \leq p$ 。ARMA 模型的传递函数  $H(z)$  定义为

$$H(z) = \sum_{i=0}^{\infty} h(i)z^{-i} = \frac{b(0) + b(1)z^{-1} + \cdots + b(q)z^{-q}}{a(0) + a(1)z^{-1} + \cdots + a(p)z^{-p}} \quad (2.11.10)$$

在定理 2.11.2 中作变量代换  $m = p$ , 并令

$$\begin{aligned} a_i &= a(p-i), & i &= 0, 1, \dots, p \\ b_i &= \begin{cases} b(q-i), & i = 0, 1, \dots, q \\ 0, & i = q+1, q = 2, 3, \dots, p \end{cases} \end{aligned}$$

显然, 对 ARMA( $p, q$ ) 模型式 (2.11.9) 应用定理 2.11.2, 立即有重要结论: 由 ARMA 模型的冲激响应  $h(i)$  构造的 Hankel 矩阵  $H$  的秩等于  $p$ , 即

$$\text{rank}(H) = \text{rank} \begin{bmatrix} h(0) & h(1) & h(2) & \cdots \\ h(1) & h(2) & h(3) & \cdots \\ h(2) & h(3) & h(3) & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} = p$$

利用一个 ARMA( $p, q$ ) 模型的 Hankel 矩阵的秩等于  $p$  这一结果, 可以分析 ARMA 建模时 AR 参数的唯一可辨识性。这一可辨识性是 Gersch 分析的 [187]。

## 本章小结

本章介绍了两类具有特殊结构的矩阵: 一类特殊矩阵与矩阵的运算有关, 如互换矩阵、置换矩阵、选择矩阵、带型矩阵、相似矩阵以及相合矩阵等; 另一类特殊矩阵与有关事物的抽象表示有关, 如循环矩阵、Hermitian 矩阵、中心化矩阵、Vandermonde 矩阵、Fourier 矩阵、Hadamard 矩阵、Toeplitz 矩阵和 Hankel 矩阵等。

在讲述一些特殊矩阵时, 本章还依次介绍了置换矩阵在盲信号分离中的应用; 中心化矩阵在数理统计中的应用; Vandermonde 矩阵在谐波恢复中的应用; 广义 Kronecker 积在 Fourier 矩阵的构造和 FFT 设计中的应用。此外, 本章还介绍了 Toeplitz 矩阵的快速余弦变换。

## 习题

**2.1** 证明多个正交矩阵的乘积仍然为正交矩阵。

**2.2** 令  $A$  为实对称矩阵,  $B$  为实反对称矩阵, 且这两个矩阵是乘积可交换的, 即  $AB = BA$ 。证明: 若  $A - B$  是非奇异的, 则  $(A + B)(A - B)^{-1}$  是正交矩阵。

**2.3** 令  $E_{\alpha(p)}A$  是使矩阵  $A$  第  $p$  行乘常数  $\alpha$  的初等矩阵, 且  $E_{(p)+\alpha(q)}A$  是矩阵  $A$  的第  $q$  行乘非零常数  $\alpha$  后, 加到  $A$  的第  $p$  行的初等矩阵。证明初等矩阵的下列性质:

$$(1) \det(E_{\alpha(p)}) = \alpha.$$

$$(2) \det(E_{(p)+\alpha(q)}) = 1.$$

**2.4** 令  $A_{n \times n}$  是下三角矩阵。要求:

(1) 求  $A$  可对角化的条件。

(2) 若  $a_{11} = \dots = a_{nn}$ , 且至少有一个元素  $a_{ij} \neq 0$  ( $i > j$ ), 证明:  $A$  不可对角化。

**2.5** 令

$$T = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ 0 & t_{22} & \cdots & t_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & t_{nn} \end{bmatrix}$$

为上三角矩阵。证明:

(1) 若对某个  $1 \leq i \leq n$  有  $t_{ii} = 0$ , 则  $T$  奇异。

(2) 若  $t_{ii} \neq 0$ ,  $i = 1, 2, \dots, n$ , 则  $T$  非奇异。

**2.6** 设矩阵

$$A = \begin{bmatrix} 3 & 2 & -2 \\ -c & -1 & c \\ 4 & 2 & -3 \end{bmatrix}$$

求  $c$  值, 使得  $B = P^{-1}AP$  为对角矩阵。并求出矩阵  $P$  和  $B$ 。

**2.7** 若  $A$  为幂等矩阵和对称矩阵, 证明  $A$  是半正定的。

**2.8** 证明: 一个维数为奇数的反对称矩阵, 其行列式必等于零。

**2.9** 一个  $n$  阶 Helmert 矩阵  $H_n$  的第 1 行为  $n^{-1/2}1_n^T$ , 其他  $n-1$  行具有分块形式 [444, p.71]

$$\frac{1}{\sqrt{\lambda_i}} [1_i^T, -i, 0_{n-i-1}^T], \quad \lambda_i = i(i+1), \quad i = 1, 2, \dots, n-1$$

式中,  $1_i^T$  和  $0_i^T$  分别表示元素全部为 1 和 0 的  $i$  阶行向量。例如

$$H_4 = \begin{bmatrix} 1/\sqrt{4} & 1/\sqrt{4} & 1/\sqrt{4} & 1/\sqrt{4} \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 1/\sqrt{6} & 1/\sqrt{6} & -2/\sqrt{6} & 0 \\ 1/\sqrt{12} & 1/\sqrt{12} & 1/\sqrt{12} & -3\sqrt{12} \end{bmatrix}$$

将  $n$  阶 Helmert 矩阵分块为

$$H = \begin{bmatrix} h^T \\ K \end{bmatrix}$$

式中,  $h = n^{-1/2}1^{-1/2}$ , 而  $K$  表示  $H$  的最后  $n-1$  行。

(1) 证明:  $HH^T = I_n$ 。

(2) 对于  $n$  阶向量  $\mathbf{x}$ , 证明  $n^{-1}\bar{x}_n^2 = \mathbf{x}^T \mathbf{h}^T \mathbf{h} \mathbf{x}$ , 其中  $\bar{x}_n = \frac{1}{n} \sum_{i=1}^n x_i$ ; 并证明

$$S_n = \sum_{i=1}^n \left( x_i - \frac{1}{n} \sum_{k=1}^n x_k / n \right)^2$$

可以表示为  $S_n = \mathbf{x}^T \mathbf{K}^T \mathbf{K} \mathbf{x}$ .

(3) 推导递推公式

$$S_n = S_{n-1} + (1 - 1/n)(\bar{x}_{n-1} - x_n)$$

式中,  $\bar{x}_{n-1} = \frac{1}{n-1} \sum_{i=1}^{n-1} x_i$ .

**2.10** 令  $\mathbf{A}$  和  $\mathbf{B}$  为对称矩阵, 并且满足

$$|\mathbf{I} - \lambda \mathbf{A}| |\mathbf{I} - \mu \mathbf{B}| = |\mathbf{I} - \lambda \mathbf{A} - \mu \mathbf{B}|, \quad \forall \lambda, \mu$$

证明  $\mathbf{AB} = \mathbf{O}$  (零矩阵)。

**2.11** 证明: 若  $\mathbf{A}$  为实反对称矩阵, 则  $\mathbf{A} + \mathbf{I}$  非奇异。

**2.12** 假定  $\mathbf{A}$  是一实反对称矩阵, 证明: Cayley 变换  $\mathbf{T} = (\mathbf{I} - \mathbf{A})(\mathbf{I} + \mathbf{A})^{-1}$  为正交矩阵。

**2.13** 令  $\mathbf{A}$  是正交矩阵, 且  $\mathbf{A} + \mathbf{I}$  非奇异。证明: 矩阵  $\mathbf{A}$  可表示为 Cayley 变换

$$\mathbf{A} = (\mathbf{I} - \mathbf{S})(\mathbf{I} + \mathbf{S})^{-1}$$

式中,  $\mathbf{S}$  为实反对称矩阵。

**2.14** 证明:

$$(1) \det(E_{\alpha(p)}) = \alpha.$$

$$(2) \det(E_{(p)+\alpha(q)}) = 1.$$

**2.15** 令  $\mathbf{P}$  是一个  $n \times n$  置换矩阵, 证明: 存在一个正整数  $k$ , 使得  $\mathbf{P}^k = \mathbf{I}$ 。 (提示: 考虑矩阵序列  $\mathbf{P}, \mathbf{P}^2, \mathbf{P}^3, \dots$ )

**2.16** 假定  $\mathbf{P}$  和  $\mathbf{Q}$  是两个  $n \times n$  置换矩阵, 证明:  $\mathbf{PQ}$  也是一个  $n \times n$  置换矩阵。

**2.17** 证明: 对于每一个矩阵  $\mathbf{A}$ , 都存在一个三角矩阵  $\mathbf{T}$ , 使得  $\mathbf{TA}$  为酉矩阵。

**2.18** 令  $\mathbf{A}$  是一个给定的矩阵。证明: 可以找到一个主对角线上的元素取  $\pm 1$  的矩阵  $\mathbf{J}$ , 使得  $\mathbf{JA} + \mathbf{I}$  非奇异。

**2.19** 证明: 若  $\mathbf{H} = \mathbf{A} + j\mathbf{B}$  为 Hermitian 矩阵, 且  $\mathbf{A}$  非奇异, 则行列式的绝对值的平方

$$|\det(\mathbf{H})|^2 = |\mathbf{A}|^2 |\mathbf{I} + \mathbf{A}^{-1} \mathbf{B} \mathbf{A}^{-1} \mathbf{B}|$$

**2.20** 证明下列叙述等价:

(1)  $\mathbf{U}$  是酉矩阵;

- (2)  $\mathbf{U}$  是非奇异的，并且  $\mathbf{U}^H = \mathbf{U}^{-1}$ ；  
 (3)  $\mathbf{U}\mathbf{U}^H = \mathbf{U}^H\mathbf{U} = \mathbf{I}$ ；  
 (4)  $\mathbf{U}^H$  是酉矩阵；  
 (5)  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_n]$  的列组成标准正交组，即

$$\mathbf{u}_i^H \mathbf{u}_j = \delta(i-j) = \begin{cases} 1, & i=j \\ 0, & i \neq j \end{cases}$$

- (6)  $\mathbf{U}$  的行组成标准正交组。

**2.21 证明  $J$  正交矩阵的以下性质：**

- (1)  $J$  正交矩阵  $\mathbf{Q}$  非奇异，其行列式的绝对值等于 1。  
 (2) 任何一个  $N \times N$  维  $J$  正交矩阵  $\mathbf{Q}$  也可以等价定义为

$$\mathbf{Q}^T J \mathbf{Q} = J$$

**2.22 令  $A, S$  为  $n \times n$  矩阵，且  $S$  非奇异。要求：**

- (1) 证明  $(S^{-1}AS)^2 = S^{-1}A^2S$  和  $(S^{-1}AS)^3 = S^{-1}A^3S$ 。  
 (2) 利用数学归纳法证明  $(S^{-1}AS)^k = S^{-1}A^kS$ ，其中， $k$  为正整数。

**2.23 证明：若  $A$  是可对角化的，并且  $B$  与  $A$  相似，则  $B$  是可对角化的。(提示：假定  $S^{-1}AS = D$  和  $W^{-1}AW = B$ 。)**

**2.24 证明相似矩阵的幂性质：若  $B$  与  $A$  相似，则  $B^k$  与  $A^k$  相似。**

**2.25 假定  $B$  与  $A$  相似，证明：**

- (1)  $B + \alpha I$  与  $A + \alpha I$  相似。  
 (2)  $B^T$  与  $A^T$  相似。  
 (3) 若  $A, B$  非奇异，则  $B^{-1}$  与  $A^{-1}$  相似。

**2.26 假定  $A, B$  为  $n \times n$  矩阵，并且  $B$  非奇异，证明： $AB$  与  $BA$  相似。**

**2.27 证明：若  $n \times n$  矩阵  $A$  与  $n \times n$  单位矩阵  $I$  相似，则  $A = I$ 。**

**2.28 令  $A$  是一个  $n \times n$  实矩阵，证明： $B = (A + A^T)/2$  为对称矩阵，而  $C = (A - A^T)/2$  为反对称矩阵。**

**2.29** 给定  $n+1$  个不同的数  $x_0, x_1, \dots, x_n$  和任意  $n+1$  个数的集合  $\{y_0, y_1, \dots, y_n\}$ ，则存在一个唯一的多项式  $p(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ ，使得  $p(x_0) = y_0, p(x_1) = y_1, \dots, p(x_n) = y_n$ 。求  $a_0, a_1, \dots, a_n$  的表达式。

**2.30** 给定  $n+1$  个不同的数  $\beta_0, \beta_1, \dots, \beta_n$  和任意  $n+1$  个数的集合  $\{y_0, y_1, \dots, y_n\}$ ，证明：存在唯一的一个多项式  $y(x) = a_0 e^{\beta_0 x} + a_1 e^{\beta_1 x} + \dots + a_n e^{\beta_n x}$  满足约束条件  $y(0) = y_0, y'(0) = y_1, \dots, y^{(n)}(0) = y_n$ ，其中， $y^{(k)}$  表示  $\frac{dy(x)}{dx^k}$ 。

**2.31**<sup>[255, p.101]</sup> 第  $i$  行和第  $j$  列元素为  $1/(i+j-1)$  的  $n \times n$  矩阵称为 Hilbert 矩阵。令  $\mathbf{A}$  是一个  $6 \times 6$  维 Hilbert 矩阵，并且

$$\mathbf{b} = [1, 2, 1, 1.414, 1, 2]^T, \quad \mathbf{b} + \Delta\mathbf{b} = [1, 2, 1, 1.4142, 1, 2]^T$$

试用 MATLAB 求解矩阵方程  $\mathbf{Ax}_1 = \mathbf{b}$  和  $\mathbf{Ax}_2 = \mathbf{b} + \Delta\mathbf{b}$ ，并比较  $\mathbf{x}_1$  和  $\mathbf{x}_2$ 。为什么尽管向量  $\mathbf{A}$  的扰动很小， $\mathbf{x}_1$  和  $\mathbf{x}_2$  却相差很大？

**2.32**<sup>[36, p.68]</sup> 矩阵  $\mathbf{A} = [a_{ij}], i, j = 1, 2, 3, 4$  称为 Lorentz 矩阵，若变换  $\mathbf{x} = \mathbf{Ay}$  使得二次型  $Q(\mathbf{x}) = \mathbf{x}^T \mathbf{Ax} = x_1^2 - x_2^2 - x_3^2 - x_4^2$  不变，即  $Q(\mathbf{x}) = Q(\mathbf{y})$ 。证明：两个 Lorentz 矩阵的乘积仍然为 Lorentz 矩阵。

**2.33**<sup>[36, p.265]</sup>  $n \times n$  矩阵  $\mathbf{M}$  称为 Markov 矩阵，若其元素满足条件  $m_{ij} \geq 0, \sum_{i=1}^n m_{ij} = 1, j = 1, 2, \dots, n$ 。假定  $\mathbf{P}$  和  $\mathbf{Q}$  均为 Markov 矩阵，证明：

- (1) 对于常数  $0 \leq \lambda \leq 1$ ，矩阵  $\lambda\mathbf{P} + (1 - \lambda)\mathbf{Q}$  是 Markov 矩阵。
- (2) 矩阵乘积  $\mathbf{PQ}$  也为 Markov 矩阵。

**2.34**<sup>[36, p.265]</sup> 一个  $n \times 1$  向量  $\mathbf{x}$  称为概率向量 (probability vector)，若其元素满足与概率公式类似的条件  $x_i \geq 0$  和  $\sum_{i=1}^n x_i = 1$ 。证明：若  $\mathbf{x}$  为概率向量，则矩阵  $\mathbf{M}$  是 Markov 矩阵，当且仅当  $\mathbf{Mx}$  是概率向量。

**2.35** 令  $y_i = y_i(x_1, x_2, \dots, x_n), i = 1, 2, \dots, n$  是关于  $x_1, x_2, \dots, x_n$  的  $n$  个函数。矩阵  $\mathbf{J} = \mathbf{J}(\mathbf{y}, \mathbf{x}) = [\partial y_i / \partial x_j]$  称为函数  $y_i(x_1, x_2, \dots, x_n), i = 1, 2, \dots, n$  的 Jacobian 矩阵，其行列式称为 Jacobian 行列式。式中， $\mathbf{y} = [y_1, y_2, \dots, y_n]^T$ ， $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ 。证明： $\mathbf{J}(\mathbf{z}, \mathbf{y})\mathbf{J}(\mathbf{y}, \mathbf{x}) = \mathbf{J}(\mathbf{z}, \mathbf{x})$ 。

**2.36** 令  $n \times n$  矩阵  $\mathbf{X}$  和  $\mathbf{Y}$  分别为对称矩阵，并且  $\mathbf{Y} = \mathbf{AXA}^T$ 。证明：Jacobian 行列式  $|\mathbf{J}(\mathbf{Y}, \mathbf{X})|$  等于矩阵  $\mathbf{A}$  的行列式的  $n+1$  次方，即  $|\mathbf{J}(\mathbf{Y}, \mathbf{X})| = |\mathbf{A}|^{n+1}$ 。

**2.37** 证明：若  $\mathbf{A}$  为正规矩阵 (即满足  $\mathbf{AA}^H = \mathbf{A}^H \mathbf{A}$ )，则  $\mathbf{A} - \lambda\mathbf{I}$  为正规矩阵。

**2.38** 若  $\mathbf{B}$  为正规矩阵，并且存在一个角度  $\theta$ ，使得  $\mathbf{Ae}^{j\theta} + \mathbf{A}^H e^{-j\theta} \geq 0$ ，其中， $\mathbf{A}^2 = \mathbf{B}$ 。证明： $\mathbf{A}$  是正规矩阵。

**2.39** 满足条件  $\mathbf{AB} = \mathbf{BA}$  的矩阵  $\mathbf{A}$  和  $\mathbf{B}$  称为可交换矩阵 (commute matrix)。证明：若  $\mathbf{A}$  和  $\mathbf{B}$  可交换，则  $\mathbf{A}^H$  和  $\mathbf{B}$  可交换的条件是  $\mathbf{A}$  为正规矩阵。

**2.40**<sup>[36, p.226]</sup> 令  $\mathbf{A}$  为复矩阵，证明  $\mathbf{A}$  为正规矩阵，当且仅当下列条件之一成立：

(1)  $\mathbf{A} = \mathbf{B} + j\mathbf{C}$ ，其中， $\mathbf{B}$  和  $\mathbf{C}$  为 Hermitian 矩阵，并且可交换。

(2)  $\mathbf{A} = \mathbf{U}^H \mathbf{D} \mathbf{U}$ ，其中， $\mathbf{U}$  为酉矩阵，且  $\mathbf{D}$  为对角矩阵。

(3)  $\mathbf{A} = \mathbf{UH}$ ，其中， $\mathbf{U}$  为酉矩阵， $\mathbf{H}$  为 Hermitian 矩阵，并且  $\mathbf{U}$  和  $\mathbf{H}$  可交换。

**2.41** 证明：若  $\mathbf{A}$  是斜 Hermitian 矩阵，则  $\langle \mathbf{Ax}, \mathbf{x} \rangle = 0$  对任意向量  $\mathbf{x} \in \mathbb{C}^n$  成立。

## 第3章 矩阵微分

矩阵微分是多变量函数微分的推广。矩阵微分(包括矩阵偏导和梯度)是矩阵的重要运算工具之一，在统计学、流形计算、几何物理、微分几何、经济计量以及众多工程中有着广泛的应用。特别地，在许多工程应用(如阵列信号处理、通信系统、雷达、声呐)中，信号和系统参数往往都表示成复值向量或矩阵。本章主要介绍矩阵微分的理论、计算方法与应用，先讨论函数的变元为实值向量和实值矩阵的情况，然后再推广到函数变元为复值向量和复值矩阵的矩阵微分。

### 3.1 Jacobian 矩阵与梯度矩阵

本章的前半部分讨论实值标量函数、实值向量函数和实值矩阵函数相对于实向量变元或矩阵变元的偏导。为了方便理解，首先对变元和函数作统一的符号规定：

$\mathbf{x} = [x_1, \dots, x_m]^T \in \mathbb{R}^m$  为实向量变元；

$\mathbf{X} = [x_1, \dots, x_n] \in \mathbb{R}^{m \times n}$  为实矩阵变元；

$f(\mathbf{x}) \in \mathbb{R}$  为实值标量函数，其变元为  $m \times 1$  实值向量  $\mathbf{x}$ ，记作  $f : \mathbb{R}^m \rightarrow \mathbb{R}$ ；

$f(\mathbf{X}) \in \mathbb{R}$  为实值标量函数，其变元为  $m \times n$  实值矩阵  $\mathbf{X}$ ，记作  $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ ；

$f(\mathbf{x}) \in \mathbb{R}^p$  为  $p$  维实列向量函数，其变元为  $m \times 1$  实值向量  $\mathbf{x}$ ，记作  $f : \mathbb{R}^m \rightarrow \mathbb{R}^p$ ；

$f(\mathbf{X}) \in \mathbb{R}^p$  为  $p$  维实列向量函数，变元为  $m \times n$  实矩阵  $\mathbf{X}$ ，记作  $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ ；

$F(\mathbf{x}) \in \mathbb{R}^{p \times q}$  为  $p \times q$  实矩阵函数，变元为  $m \times 1$  实向量  $\mathbf{x}$ ，记作  $F : \mathbb{R}^m \rightarrow \mathbb{R}^{p \times q}$ ；

$F(\mathbf{X}) \in \mathbb{R}^{p \times q}$  为  $p \times q$  实矩阵函数，变元为  $m \times n$  实矩阵  $\mathbf{X}$ ，记作  $F : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$ 。

表 3.1.1 汇总了以上实值函数的分类。

表 3.1.1 实值函数的分类

函数类型	向量变元 $\mathbf{x} \in \mathbb{R}^m$	矩阵变元 $\mathbf{X} \in \mathbb{R}^{m \times n}$
标量函数 $f \in \mathbb{R}$	$f(\mathbf{x})$ $f : \mathbb{R}^m \rightarrow \mathbb{R}$	$f(\mathbf{X})$ $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$
向量函数 $f \in \mathbb{R}^p$	$f(\mathbf{x})$ $f : \mathbb{R}^m \rightarrow \mathbb{R}^p$	$f(\mathbf{X})$ $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$
矩阵函数 $F \in \mathbb{R}^{p \times q}$	$F(\mathbf{x})$ $F : \mathbb{R}^m \rightarrow \mathbb{R}^{p \times q}$	$F(\mathbf{X})$ $F : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$

本节主要讨论实值标量函数和实值矩阵函数的偏导。

### 3.1.1 Jacobian 矩阵

$1 \times m$  行向量偏导算子记为

$$\mathbf{D}_{\mathbf{x}} \stackrel{\text{def}}{=} \frac{\partial}{\partial \mathbf{x}^T} = \left[ \frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_m} \right] \quad (3.1.1)$$

于是, 实值标量函数  $f(\mathbf{x})$  在  $\mathbf{x}$  的偏导向量由  $1 \times m$  行向量

$$\mathbf{D}_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} = \left[ \frac{\partial f(\mathbf{x})}{\partial x_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_m} \right] \quad (3.1.2)$$

给出。

当实值标量函数  $f(\mathbf{X})$  的变元为实值矩阵  $\mathbf{X} \in \mathbb{R}^{m \times n}$  时, 存在两种可能的定义

$$\mathbf{D}_{\mathbf{X}} f(\mathbf{X}) = \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}^T} = \begin{bmatrix} \frac{\partial f(\mathbf{X})}{\partial x_{11}} & \cdots & \frac{\partial f(\mathbf{X})}{\partial x_{m1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{X})}{\partial x_{1n}} & \cdots & \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \end{bmatrix} \in \mathbb{R}^{n \times m} \quad (3.1.3)$$

和

$$\mathbf{D}_{\text{vec } \mathbf{X}} f(\mathbf{X}) = \frac{\partial f(\mathbf{X})}{\partial \text{vec } (\mathbf{X})} = \left[ \frac{\partial f(\mathbf{X})}{\partial x_{11}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{m1}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{1n}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \right] \quad (3.1.4)$$

其中  $\mathbf{D}_{\mathbf{X}} f(\mathbf{X})$  和  $\mathbf{D}_{\text{vec } \mathbf{X}} f(\mathbf{X})$  分别称为实值标量函数  $f(\mathbf{X})$  关于矩阵变元  $\mathbf{X}$  的 Jacobian 矩阵和行偏导向量, 两者之间的关系为

$$\mathbf{D}_{\text{vec } \mathbf{X}} f(\mathbf{X}) = \text{rvec}(\mathbf{D}_{\mathbf{X}} f(\mathbf{X})) = (\text{vec}(\mathbf{D}_{\mathbf{X}}^T f(\mathbf{X})))^T \quad (3.1.5)$$

即实值标量函数  $f(\mathbf{X})$  的行向量偏导  $\mathbf{D}_{\text{vec } \mathbf{X}} f(\mathbf{X})$  等于 Jacobian 矩阵的转置  $\mathbf{D}_{\mathbf{X}}^T f(\mathbf{X})$  的列向量化  $\text{vec}(\mathbf{D}_{\mathbf{X}}^T f(\mathbf{X}))$  的转置。这一重要关系是 Jacobian 矩阵辨识的基础。

事实上, 在实际应用中 Jacobian 矩阵比行偏导向量更有用。

现在考虑实值矩阵函数  $\mathbf{F}(\mathbf{X}) = [f_{kl}]_{k=1, l=1}^{p, q} \in \mathbb{R}^{p \times q}$  的情况, 其中, 矩阵变元  $\mathbf{X} \in \mathbb{R}^{m \times n}$ 。此时, 有多种可能的行偏导矩阵定义。例如

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}^T} = \left[ \frac{\partial f_{kl}(\mathbf{X})}{\partial \mathbf{X}^T} \right]_{k=1, l=1}^{p, q} = \begin{bmatrix} \frac{\partial f_{11}(\mathbf{X})}{\partial \mathbf{X}^T} & \cdots & \frac{\partial f_{1q}(\mathbf{X})}{\partial \mathbf{X}^T} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{p1}(\mathbf{X})}{\partial \mathbf{X}^T} & \cdots & \frac{\partial f_{pq}(\mathbf{X})}{\partial \mathbf{X}^T} \end{bmatrix} \in \mathbb{R}^{pn \times qm}$$

或者

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}^T} = \left[ \frac{\partial \mathbf{F}(\mathbf{X})}{\partial x_{ji}} \right]_{i=1, j=1}^{m, n} = \begin{bmatrix} \frac{\partial f_{11}(\mathbf{X})}{\partial x_{j1}} & \cdots & \frac{\partial f_{1q}(\mathbf{X})}{\partial x_{j1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{p1}(\mathbf{X})}{\partial x_{ji}} & \cdots & \frac{\partial f_{pq}(\mathbf{X})}{\partial x_{ji}} \end{bmatrix}_{j=1, i=1}^{n, m} \in \mathbb{R}^{pn \times qm}$$

或者

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}^T} = \left[ \frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}^T} \right]_{i=1, j=1}^{m, n} = \begin{bmatrix} \frac{\partial \mathbf{F}(\mathbf{X})}{\partial x_{11}} & \cdots & \frac{\partial \mathbf{F}(\mathbf{X})}{\partial x_{m1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{F}(\mathbf{X})}{\partial x_{1n}} & \cdots & \frac{\partial \mathbf{F}(\mathbf{X})}{\partial x_{mn}} \end{bmatrix} \in \mathbb{R}^{pn \times qm}$$

正如 Magnus 与 Neudecker<sup>[328]</sup> 指出的那样, 上述三种定义都是不好的定义, 因为它们不适合于计算比较复杂的矩阵函数的 Jacobian 矩阵, 有时甚至会给出错误的 Jacobian 矩阵。例如, 对于矩阵函数  $\mathbf{F}(\mathbf{X}) = \mathbf{X}$ , 第一个定义给出

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}^T} = \left[ \frac{\partial \mathbf{X}}{\partial \mathbf{X}^T} \right]_{k=1, l=1}^{p, q} = (\text{vec } \mathbf{I}_m)(\text{vec } \mathbf{I}_n)^T$$

这是一个秩等于 1 的矩阵, 与真实的 Jacobian 矩阵(即  $mn \times mn$  维单位矩阵)不符。

Magnus 与 Neudecker<sup>[328]</sup> 给出了关于矩阵函数的 Jacobian 矩阵的一种好的定义: 先通过列向量化, 将  $p \times q$  矩阵函数  $\mathbf{F}(\mathbf{X})$  转换成  $pq \times 1$  列向量

$$\text{vec}(\mathbf{F}(\mathbf{X})) \stackrel{\text{def}}{=} [f_{11}(\mathbf{X}), \dots, f_{p1}(\mathbf{X}), \dots, f_{1q}(\mathbf{X}), \dots, f_{pq}(\mathbf{X})]^T \in \mathbb{R}^{pq} \quad (3.1.6)$$

然后, 该列向量对矩阵变元  $\mathbf{X}$  的列向量化的转置  $(\text{vec } \mathbf{X})^T$  求偏导, 给出  $pq \times mn$  维 Jacobian 矩阵

$$\mathbf{D}_{\mathbf{X}} \mathbf{F}(\mathbf{X}) \stackrel{\text{def}}{=} \frac{\partial \text{vec}(\mathbf{F}(\mathbf{X}))}{\partial (\text{vec } \mathbf{X})^T} \in \mathbb{R}^{pq \times mn} \quad (3.1.7)$$

其具体表达式为

$$\mathbf{D}_{\mathbf{X}} \mathbf{F}(\mathbf{X}) = \begin{bmatrix} \frac{\partial f_{11}}{\partial (\text{vec } \mathbf{X})^T} \\ \vdots \\ \frac{\partial f_{p1}}{\partial (\text{vec } \mathbf{X})^T} \\ \vdots \\ \frac{\partial f_{1q}}{\partial (\text{vec } \mathbf{X})^T} \\ \vdots \\ \frac{\partial f_{pq}}{\partial (\text{vec } \mathbf{X})^T} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_{11}}{\partial x_{11}} & \cdots & \frac{\partial f_{11}}{\partial x_{m1}} & \cdots & \frac{\partial f_{11}}{\partial x_{1n}} & \cdots & \frac{\partial f_{11}}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{p1}}{\partial x_{11}} & \cdots & \frac{\partial f_{p1}}{\partial x_{m1}} & \cdots & \frac{\partial f_{p1}}{\partial x_{1n}} & \cdots & \frac{\partial f_{p1}}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{1q}}{\partial x_{11}} & \cdots & \frac{\partial f_{1q}}{\partial x_{m1}} & \cdots & \frac{\partial f_{1q}}{\partial x_{1n}} & \cdots & \frac{\partial f_{1q}}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{pq}}{\partial x_{11}} & \cdots & \frac{\partial f_{pq}}{\partial x_{m1}} & \cdots & \frac{\partial f_{pq}}{\partial x_{1n}} & \cdots & \frac{\partial f_{pq}}{\partial x_{mn}} \end{bmatrix} \quad (3.1.8)$$

### 3.1.2 梯度矩阵

采用列向量形式定义的偏导算子称为列向量偏导算子, 习惯称为梯度算子。

$m \times 1$  列向量偏导算子即梯度算子记作  $\nabla_{\mathbf{x}}$ , 定义为

$$\nabla_{\mathbf{x}} \stackrel{\text{def}}{=} \frac{\partial}{\partial \mathbf{x}} = \left[ \frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_m} \right]^T \quad (3.1.9)$$

因此, 实值标量函数  $f(\mathbf{x})$  的梯度向量  $\nabla_{\mathbf{x}} f(\mathbf{x})$  为  $m \times 1$  列向量, 定义为

$$\nabla_{\mathbf{x}} f(\mathbf{x}) \stackrel{\text{def}}{=} \left[ \frac{\partial f(\mathbf{x})}{\partial x_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_m} \right]^T = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \quad (3.1.10)$$

将矩阵变元  $\mathbf{X}$  列向量化后，即可直接定义关于矩阵变元  $\mathbf{X}$  的梯度算子为

$$\nabla_{\text{vec} \mathbf{X}} = \frac{\partial}{\partial \text{vec} \mathbf{X}} = \left[ \frac{\partial}{\partial x_{11}}, \dots, \frac{\partial}{\partial x_{m1}}, \dots, \frac{\partial}{\partial x_{1n}}, \dots, \frac{\partial}{\partial x_{mn}} \right]^T \quad (3.1.11)$$

由此得到实值标量函数  $f(\mathbf{X})$  关于矩阵变元  $\mathbf{X}$  的梯度向量

$$\nabla_{\text{vec} \mathbf{X}} f(\mathbf{X}) = \frac{\partial f(\mathbf{X})}{\partial \text{vec} \mathbf{X}} = \left[ \frac{\partial f(\mathbf{X})}{\partial x_{11}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{m1}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{1n}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \right]^T \quad (3.1.12)$$

另外，可以直接定义梯度矩阵

$$\nabla_{\mathbf{X}} f(\mathbf{X}) = \begin{bmatrix} \frac{\partial f(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f(\mathbf{X})}{\partial x_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{X})}{\partial x_{m1}} & \dots & \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \end{bmatrix} = \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} \quad (3.1.13)$$

显然，梯度矩阵  $\nabla_{\mathbf{X}} f(\mathbf{X})$  是梯度向量  $\nabla_{\text{vec} \mathbf{X}} f(\mathbf{X})$  的矩阵化

$$\nabla_{\mathbf{X}} f(\mathbf{X}) = \text{unvec}(\nabla_{\text{vec} \mathbf{X}} f(\mathbf{X})) \quad (3.1.14)$$

比较式 (3.1.13) 和式 (3.1.3)，又有

$$\nabla_{\mathbf{X}} f(\mathbf{X}) = D_{\mathbf{X}}^T f(\mathbf{X}) \quad (3.1.15)$$

即是说，实值标量函数  $f(\mathbf{X})$  的梯度矩阵等于 Jacobian 矩阵的转置。

正如 Kreutz-Delgado<sup>[281]</sup> 指出的那样，在流形计算<sup>[4]</sup>、几何物理<sup>[442],[175]</sup> 以及微分几何<sup>[459]</sup> 等中，当定义一个标量函数关于变元向量的偏导数时，行向量偏导向量和 Jacobian 矩阵是“最自然的”选择。我们在后面还将看到，在矩阵微分中，行向量偏导向量和 Jacobian 矩阵也是一种最自然的选择。然而，在最优化和许多工程问题中，采用列向量形式定义的偏导（梯度向量和梯度矩阵）却是一种比行向量偏导和 Jacobian 矩阵更加自然的选择。

显然，对于一个给定的实值标量函数  $f(\mathbf{x})$ ，其梯度向量直接等于偏导向量的转置。在此意义上，行向量形式的偏导向量是列向量形式的梯度向量的协变形式 (covariant form of the gradient vector)，故又简称为协梯度向量 (cogradient vector)。类似地，Jacobian 矩阵有时也称为梯度矩阵的协变形式或简称协 (同) 梯度矩阵。协梯度是一协变算子 (covariant operator)<sup>[175]</sup>，它本身虽然不是梯度，但却是梯度的紧密伙伴 (转置后即变为梯度)。

有鉴于此，Jacobian 算子  $\frac{\partial}{\partial \mathbf{x}^T}$  和  $\frac{\partial}{\partial \mathbf{X}^T}$  又称 (行) 偏导算子、梯度算子的协变形式或协梯度算子 (cogradient operator)。

梯度方向的负方向  $-\nabla_{\mathbf{x}} f(\mathbf{x})$  称为函数  $f$  在点  $\mathbf{x}$  的梯度流 (gradient flow)，记作

$$\dot{\mathbf{x}} = -\nabla_{\mathbf{x}} f(\mathbf{x}) \quad \text{或} \quad \dot{\mathbf{X}} = -\nabla_{\text{vec} \mathbf{X}} f(\mathbf{X}) \quad (3.1.16)$$

从梯度向量的定义式可以看出：

(1) 在梯度流方向, 函数  $f(\mathbf{x})$  以最大减小率下降。反之, 在其反方向即正的梯度方向, 函数值以最大增大率增加。

(2) 梯度向量的每个分量给出了标量函数在该分量方向上的变化率。

对于实值矩阵函数  $\mathbf{F}(\mathbf{X}) \in \mathbb{R}^{p \times q}$  (其中矩阵变元  $\mathbf{X} \in \mathbb{R}^{m \times n}$ ), 梯度矩阵定义为

$$\nabla_{\mathbf{X}} \mathbf{F}(\mathbf{X}) = \frac{\partial \text{vec}^T \mathbf{F}(\mathbf{X})}{\partial \text{vec} \mathbf{X}} = \left( \frac{\partial \text{vec} \mathbf{F}(\mathbf{X})}{\partial \text{vec}^T \mathbf{X}} \right)^T \quad (3.1.17)$$

与 Jacobian 矩阵的情况相仿, 公式

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial f_{11}}{\partial \mathbf{X}} & \cdots & \frac{\partial f_{1q}}{\partial \mathbf{X}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{p1}}{\partial \mathbf{X}} & \cdots & \frac{\partial f_{pq}}{\partial \mathbf{X}} \end{bmatrix} \in \mathbb{R}^{pm \times qn} \quad (3.1.18)$$

对于梯度矩阵是一个不好的定义, 因为正如 3.2 节将看到的那样, 当  $\mathbf{F}(\mathbf{X}) = \mathbf{X}$  时, 式 (3.1.17) 给出正确的梯度矩阵  $\nabla_{\mathbf{X}} \mathbf{F}(\mathbf{X}) = \frac{\partial \text{vec}^T \mathbf{F}(\mathbf{X})}{\partial \text{vec} \mathbf{X}} = \mathbf{I}_n \otimes \mathbf{I}_m = \mathbf{I}_{mn}$ , 而式 (3.1.18) 则给出错误的梯度矩阵  $\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} = \mathbf{I}_n (\mathbf{I}_m)^T$ , 因为其秩不应该等于 1。

显然有

$$\nabla_{\mathbf{X}} \mathbf{F}(\mathbf{X}) = (\mathbf{D}_{\mathbf{X}} \mathbf{F}(\mathbf{X}))^T \quad (3.1.19)$$

换言之, 矩阵函数的梯度矩阵是其 Jacobian 矩阵的转置。

### 3.1.3 偏导和梯度计算

实值函数相对于矩阵变元的梯度计算具有以下性质和法则 [324]:

(1) 若  $f(\mathbf{X}) = c$  为常数, 其中,  $\mathbf{X}$  为  $m \times n$  矩阵, 则梯度  $\frac{\partial c}{\partial \mathbf{X}} = \mathbf{O}_{m \times n}$ 。

(2) 线性法则 若  $f(\mathbf{X})$  和  $g(\mathbf{X})$  分别是矩阵  $\mathbf{X}$  的实值函数,  $c_1$  和  $c_2$  为实常数, 则

$$\frac{\partial [c_1 f(\mathbf{X}) + c_2 g(\mathbf{X})]}{\partial \mathbf{X}} = c_1 \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + c_2 \frac{\partial g(\mathbf{X})}{\partial \mathbf{X}} \quad (3.1.20)$$

(3) 乘积法则 若  $f(\mathbf{X})$ 、 $g(\mathbf{X})$  和  $h(\mathbf{X})$  都是矩阵  $\mathbf{X}$  的实值函数, 则

$$\frac{\partial [f(\mathbf{X})g(\mathbf{X})]}{\partial \mathbf{X}} = g(\mathbf{X}) \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + f(\mathbf{X}) \frac{\partial g(\mathbf{X})}{\partial \mathbf{X}} \quad (3.1.21)$$

和

$$\begin{aligned} \frac{\partial [f(\mathbf{X})g(\mathbf{X})h(\mathbf{X})]}{\partial \mathbf{X}} &= g(\mathbf{X})h(\mathbf{X}) \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} + f(\mathbf{X})h(\mathbf{X}) \frac{\partial g(\mathbf{X})}{\partial \mathbf{X}} + \\ &\quad f(\mathbf{X})g(\mathbf{X}) \frac{\partial h(\mathbf{X})}{\partial \mathbf{X}} \end{aligned} \quad (3.1.22)$$

(4) 商法则 若  $g(\mathbf{X}) \neq 0$ , 则

$$\frac{\partial [f(\mathbf{X})/g(\mathbf{X})]}{\partial \mathbf{X}} = \frac{1}{g^2(\mathbf{X})} \left[ g(\mathbf{X}) \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} - f(\mathbf{X}) \frac{\partial g(\mathbf{X})}{\partial \mathbf{X}} \right] \quad (3.1.23)$$

(5) 链式法则 令  $\mathbf{X}$  为  $m \times n$  矩阵, 且  $y = f(\mathbf{X})$  和  $g(y)$  分别是以矩阵  $\mathbf{X}$  和标量  $y$  为变元的实值函数, 则

$$\frac{\partial g(f(\mathbf{X}))}{\partial \mathbf{X}} = \frac{dg(y)}{dy} \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} \quad (3.1.24)$$

推而广之, 若记  $g(\mathbf{F}(\mathbf{X})) = g(\mathbf{F})$ , 其中  $\mathbf{F} = [f_{kl}] \in \mathbb{R}^{p \times q}$ ,  $\mathbf{X} = [x_{ij}] \in \mathbb{R}^{m \times n}$ , 则链式法则为<sup>[403]</sup>

$$\left[ \frac{\partial g(\mathbf{F})}{\partial \mathbf{X}} \right]_{ij} = \frac{\partial g(\mathbf{F})}{\partial x_{ij}} = \sum_{k=1}^p \sum_{l=1}^q \frac{\partial g(\mathbf{F})}{\partial f_{kl}} \frac{\partial f_{kl}}{\partial x_{ij}} \quad (3.1.25)$$

在计算一个以向量或者矩阵为变元的函数的偏导时, 有以下基本假设。

**独立性基本假设** 假定实值函数的向量变元  $\mathbf{x} = [x_i]_{i=1}^m \in \mathbb{R}^m$  或者矩阵变元  $\mathbf{X} = [x_{ij}]_{i=1,j=1}^{m,n} \in \mathbb{R}^{m \times n}$  本身无任何特殊结构, 即向量或矩阵变元的元素之间是各自独立的。

上述独立性基本假设可以用数学公式表示成

$$\frac{\partial x_i}{\partial x_j} = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & \text{其他} \end{cases} \quad (3.1.26)$$

以及

$$\frac{\partial x_{kl}}{\partial x_{ij}} = \delta_{ki} \delta_{lj} = \begin{cases} 1, & k = i \text{ 且 } l = j \\ 0, & \text{其他} \end{cases} \quad (3.1.27)$$

式 (3.1.26) 和式 (3.1.27) 分别是一个实值(标量、向量或矩阵)函数关于向量变元和矩阵变元的偏导计算的基本公式。下面举例说明。

**例 3.1.1** 求实值函数  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x}$  的 Jacobian 矩阵。由于  $\mathbf{x}^T \mathbf{A} \mathbf{x} = \sum_{k=1}^n \sum_{l=1}^n a_{kl} x_k x_l$ , 故利用式 (3.1.26) 可求出行偏导向量  $\frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}^T}$  的第  $i$  个分量为

$$\left[ \frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}^T} \right]_i = \frac{\partial}{\partial x_i} \sum_{k=1}^n \sum_{l=1}^n a_{kl} x_k x_l = \sum_{k=1}^n x_k a_{ki} + \sum_{l=1}^n x_l a_{il}$$

立即得行偏导向量  $Df(\mathbf{x}) = \mathbf{x}^T \mathbf{A} + \mathbf{A}^T \mathbf{x} = \mathbf{x}^T (\mathbf{A} + \mathbf{A}^T)$  和梯度向量  $\nabla_{\mathbf{X}} f(\mathbf{x}) = (Df(\mathbf{x}))^T = (\mathbf{A}^T + \mathbf{A}) \mathbf{x}$ 。

**例 3.1.2** 求实值标量函数  $f(\mathbf{X}) = \mathbf{a}^T \mathbf{X} \mathbf{X}^T \mathbf{b}$  的 Jacobian 矩阵, 其中  $\mathbf{X} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{n \times 1}$ 。由于

$$\mathbf{a}^T \mathbf{X} \mathbf{X}^T \mathbf{b} = \sum_{k=1}^m \sum_{l=1}^m a_k \left( \sum_{p=1}^n x_{kp} x_{lp} \right) b_l$$

再利用式 (3.1.27), 易知

$$\begin{aligned} \left[ \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}^T} \right]_{ij} &= \frac{\partial f(\mathbf{X})}{\partial x_{ji}} = \sum_{k=1}^m \sum_{l=1}^m \sum_{p=1}^n \frac{\partial a_k x_{kp} x_{lp} b_l}{\partial x_{ji}} \\ &= \sum_{k=1}^m \sum_{l=1}^m \sum_{p=1}^n \left[ a_k x_{lp} b_l \frac{\partial x_{kp}}{\partial x_{ji}} + a_k x_{kp} b_l \frac{\partial x_{lp}}{\partial x_{ji}} \right] \\ &= \sum_{i=1}^m \sum_{l=1}^m \sum_{j=1}^n a_j x_{li} b_l + \sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^n a_k x_{ki} b_j \\ &= \sum_{i=1}^m \sum_{j=1}^n [\mathbf{X}^T \mathbf{b}]_i a_j + [\mathbf{X}^T \mathbf{a}]_i b_j \end{aligned}$$

由此得 Jacobian 矩阵和梯度矩阵分别为

$$\mathbf{D}_{\mathbf{X}} f(\mathbf{X}) = \mathbf{X}^T (\mathbf{b} \mathbf{a}^T + \mathbf{a} \mathbf{b}^T) \quad \text{和} \quad \nabla_{\mathbf{X}} f(\mathbf{X}) = (\mathbf{a} \mathbf{b}^T + \mathbf{b} \mathbf{a}^T) \mathbf{X} \quad (3.1.28)$$

**例 3.1.3** 考查目标函数  $f(\mathbf{X}) = \text{tr}(\mathbf{X} \mathbf{B})$ , 其中  $\mathbf{X}$  和  $\mathbf{B}$  分别为  $m \times n$  和  $n \times m$  实矩阵。首先, 矩阵乘积的元素为  $[\mathbf{X} \mathbf{B}]_{kl} = \sum_{p=1}^n x_{kp} b_{pl}$ , 故矩阵乘积的迹  $\text{tr}(\mathbf{X} \mathbf{B}) = \sum_{p=1}^n \sum_{l=1}^n x_{lp} b_{pl}$ 。于是, 利用式 (3.1.27), 易求得

$$\left[ \frac{\partial \text{tr}(\mathbf{X} \mathbf{B})}{\partial \mathbf{X}^T} \right]_{ij} = \frac{\partial}{\partial x_{ji}} \left( \sum_{p=1}^n \sum_{l=1}^n x_{lp} b_{pl} \right) = \sum_{p=1}^n \sum_{l=1}^n \frac{\partial x_{lp}}{\partial x_{ji}} b_{pl} = b_{ij}$$

即有  $\frac{\partial \text{tr}(\mathbf{X} \mathbf{B})}{\partial \mathbf{X}^T} = \mathbf{B}$ 。又由于  $\text{tr}(\mathbf{B} \mathbf{X}) = \text{tr}(\mathbf{X} \mathbf{B})$ , 故  $n \times m$  Jacobian 矩阵和  $m \times n$  梯度矩阵分别为

$$\mathbf{D}_{\mathbf{X}} \text{tr}(\mathbf{X} \mathbf{B}) = \mathbf{D}_{\mathbf{X}} \text{tr}(\mathbf{B} \mathbf{X}) = \mathbf{B} \quad \text{和} \quad \nabla_{\mathbf{X}} \text{tr}(\mathbf{X} \mathbf{B}) = \nabla_{\mathbf{X}} \text{tr}(\mathbf{B} \mathbf{X}) = \mathbf{B}^T \quad (3.1.29)$$

下面是矩阵函数的 Jacobian 矩阵和梯度矩阵的计算举例。

**例 3.1.4** 令  $\mathbf{F}(\mathbf{X}) = \mathbf{X} \in \mathbb{R}^{m \times n}$ , 则直接计算偏导得

$$\frac{\partial f_{kl}}{\partial x_{ij}} = \frac{\partial x_{kl}}{\partial x_{ij}} = \delta_{lj} \delta_{ki}$$

于是得 Jacobian 矩阵

$$\mathbf{D}_{\mathbf{X}} \mathbf{X} = I_n \otimes I_m = I_{mn} \in \mathbb{R}^{mn \times mn} \quad (3.1.30)$$

**例 3.1.5** 令  $\mathbf{F}(\mathbf{X}) = \mathbf{A} \mathbf{X} \mathbf{B}$ , 其中  $\mathbf{A} \in \mathbb{R}^{p \times m}$ ,  $\mathbf{X} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times q}$ 。计算偏导

$$\frac{\partial f_{kl}}{\partial x_{ij}} = \frac{\partial (\mathbf{A} \mathbf{X} \mathbf{B})_{kl}}{\partial x_{ij}} = \frac{\partial \left( \sum_{u=1}^m \sum_{v=1}^n a_{ku} x_{uv} b_{vl} \right)}{\partial x_{ij}} = b_{jl} a_{ki}$$

由此得  $pq \times mn$  Jacobian 矩阵和  $mn \times pq$  梯度矩阵分别为

$$\mathbf{D}_{\mathbf{X}} (\mathbf{A} \mathbf{X} \mathbf{B}) = \mathbf{B}^T \otimes \mathbf{A} \quad \text{和} \quad \nabla_{\mathbf{X}} (\mathbf{A} \mathbf{X} \mathbf{B}) = \mathbf{B} \otimes \mathbf{A}^T \quad (3.1.31)$$

例 3.1.6 令  $F(\mathbf{X}) = \mathbf{A}\mathbf{X}^T\mathbf{B}$ , 其中  $\mathbf{A} \in \mathbb{R}^{p \times n}$ ,  $\mathbf{X} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{m \times q}$ 。计算偏导

$$\frac{\partial f_{kl}}{\partial x_{ij}} = \frac{\partial (\mathbf{A}\mathbf{X}^T\mathbf{B})_{kl}}{\partial x_{ij}} = \frac{\partial \left( \sum_{u=1}^m \sum_{v=1}^n a_{ku} x_{vu} b_{vl} \right)}{\partial x_{ij}} = b_{il} a_{kj}$$

由此得  $pq \times mn$  Jacobian 矩阵和  $mn \times pq$  梯度矩阵分别为

$$\mathbf{D}_{\mathbf{X}}(\mathbf{A}\mathbf{X}^T\mathbf{B}) = (\mathbf{B}^T \otimes \mathbf{A})\mathbf{K}_{mn} \quad \text{和} \quad \nabla_{\mathbf{X}}(\mathbf{A}\mathbf{X}^T\mathbf{B}) = \mathbf{K}_{nm}(\mathbf{B} \otimes \mathbf{A}^T) \quad (3.1.32)$$

式中  $\mathbf{K}_{mn}$  和  $\mathbf{K}_{nm}$  为交换矩阵。

例 3.1.7 令  $\mathbf{X} \in \mathbb{R}^{m \times n}$ , 则

$$\begin{aligned} \frac{\partial f_{kl}}{\partial x_{ij}} &= \frac{\partial (\mathbf{X}\mathbf{X}^T)_{kl}}{\partial x_{ij}} = \frac{\partial \left( \sum_{u=1}^n x_{ku} x_{lu} \right)}{\partial x_{ij}} = \delta_{li} x_{kj} + x_{lj} \delta_{ki} \\ \frac{\partial f_{kl}}{\partial x_{ij}} &= \frac{\partial (\mathbf{X}^T\mathbf{X})_{kl}}{\partial x_{ij}} = \frac{\partial \left( \sum_{u=1}^n x_{uk} x_{ul} \right)}{\partial x_{ij}} = x_{il} \delta_{kj} + \delta_{lj} x_{ik} \end{aligned}$$

于是得 Jacobian 矩阵

$$\begin{aligned} \mathbf{D}_{\mathbf{X}}(\mathbf{X}\mathbf{X}^T) &= (\mathbf{I}_m \otimes \mathbf{X})\mathbf{K}_{mn} + (\mathbf{X} \otimes \mathbf{I}_m) \\ &= (\mathbf{K}_{mm} + \mathbf{I}_{m^2})(\mathbf{X} \otimes \mathbf{I}_m) \in \mathbb{R}^{mm \times mn} \end{aligned} \quad (3.1.33)$$

$$\begin{aligned} \mathbf{D}_{\mathbf{X}}(\mathbf{X}^T\mathbf{X}) &= (\mathbf{X}^T \otimes \mathbf{I}_n)\mathbf{K}_{mn} + (\mathbf{I}_n \otimes \mathbf{X}^T) \\ &= (\mathbf{K}_{nn} + \mathbf{I}_{n^2})(\mathbf{I}_n \otimes \mathbf{X}^T) \in \mathbb{R}^{nn \times mn} \end{aligned} \quad (3.1.34)$$

以及梯度矩阵

$$\nabla_{\mathbf{X}}(\mathbf{X}\mathbf{X}^T) = (\mathbf{X}^T \otimes \mathbf{I}_m)(\mathbf{K}_{mm} + \mathbf{I}_{m^2}) \in \mathbb{R}^{mn \times mm} \quad (3.1.35)$$

$$\nabla_{\mathbf{X}}(\mathbf{X}^T\mathbf{X}) = (\mathbf{I}_n \otimes \mathbf{X})(\mathbf{K}_{nn} + \mathbf{I}_{n^2}) \in \mathbb{R}^{mn \times nn} \quad (3.1.36)$$

式中使用了  $(\mathbf{A}_{p \times m} \otimes \mathbf{B}_{q \times n})\mathbf{K}_{mn} = \mathbf{K}_{pq}(\mathbf{B}_{q \times n} \otimes \mathbf{A}_{p \times m})$  以及  $\mathbf{K}_{mn}^T = \mathbf{K}_{nm}$ 。

例 3.1.8 三个矩阵的乘积函数的偏导

$$\begin{aligned} \frac{\partial (\mathbf{X}^T\mathbf{B}\mathbf{X})_{kl}}{\partial x_{ij}} &= \frac{\partial \sum_p [x_{pk}(\mathbf{B}\mathbf{X})_{pl} + (\mathbf{X}^T\mathbf{B})_{kp}x_{pl}]}{\partial x_{ij}} \\ &= \sum_p \left( \delta_{pi} \delta_{kj} (\mathbf{B}\mathbf{X})_{pl} + \delta_{pi} \delta_{lj} (\mathbf{X}^T\mathbf{B})_{kp} \right) \\ &= (\mathbf{B}\mathbf{X})_{il} \delta_{kj} + \delta_{lj} (\mathbf{X}^T\mathbf{B})_{ki} \\ \frac{\partial (\mathbf{X}\mathbf{B}\mathbf{X}^T)_{kl}}{\partial x_{ij}} &= \frac{\partial \sum_p [x_{kp}(\mathbf{B}\mathbf{X}^T)_{pl} + (\mathbf{X}\mathbf{B})_{kp}x_{lp}]}{\partial x_{ij}} \\ &= \sum_p \left( \delta_{ki} \delta_{pj} (\mathbf{B}\mathbf{X}^T)_{pl} + \delta_{pj} \delta_{li} (\mathbf{X}\mathbf{B})_{kp} \right) \\ &= (\mathbf{B}\mathbf{X}^T)_{jl} \delta_{ki} + \delta_{li} (\mathbf{X}\mathbf{B})_{kj} \end{aligned}$$

于是得 Jacobian 矩阵以及梯度矩阵分别为

$$\mathbf{D}_X(\mathbf{X}^T \mathbf{B} \mathbf{X}) = ((\mathbf{B} \mathbf{X})^T \otimes \mathbf{I}_n) \mathbf{K}_{mn} + (\mathbf{I}_n \otimes (\mathbf{X}^T \mathbf{B})) \in \mathbb{R}^{nn \times mn} \quad (3.1.37)$$

$$\mathbf{D}_X(\mathbf{X} \mathbf{B} \mathbf{X}^T) = (\mathbf{X} \mathbf{B}^T) \otimes \mathbf{I}_m + (\mathbf{I}_m \otimes (\mathbf{X} \mathbf{B})) \mathbf{K}_{mn} \in \mathbb{R}^{mn \times mn} \quad (3.1.38)$$

$$\nabla_X(\mathbf{X}^T \mathbf{B} \mathbf{X}) = \mathbf{K}_{nm} ((\mathbf{B} \mathbf{X}) \otimes \mathbf{I}_n) + (\mathbf{I}_n \otimes (\mathbf{B}^T \mathbf{X})) \in \mathbb{R}^{mn \times nn} \quad (3.1.39)$$

$$\nabla_X(\mathbf{X} \mathbf{B} \mathbf{X}^T) = (\mathbf{B} \mathbf{X}^T) \otimes \mathbf{I}_m + \mathbf{K}_{nm} (\mathbf{I}_m \otimes (\mathbf{X} \mathbf{B})^T) \in \mathbb{R}^{mn \times mm} \quad (3.1.40)$$

表 3.1.2 总结了一些矩阵函数的偏导  $\partial f_{kl}/\partial x_{ij}$  与 Jacobian 矩阵、梯度矩阵的关系。

表 3.1.2 矩阵函数  $\mathbf{F}(\mathbf{X})$  的偏导与 Jacobian 矩阵、梯度矩阵的关系

矩阵函数	$\partial f_{kl}/\partial x_{ij}$	Jacobian 矩阵 $\mathbf{D}_X \mathbf{F}(\mathbf{X})$	梯度矩阵 $\nabla_X \mathbf{F}(\mathbf{X})$
$\mathbf{A} \mathbf{X} \mathbf{B}$	$b_{jl} a_{ki}$	$\mathbf{B}^T \otimes \mathbf{A}$	$\mathbf{B} \otimes \mathbf{A}^T$
$\mathbf{A}^T \mathbf{X} \mathbf{B}$	$b_{jl} a_{ik}$	$\mathbf{B}^T \otimes \mathbf{A}^T$	$\mathbf{B} \otimes \mathbf{A}$
$\mathbf{A} \mathbf{X} \mathbf{B}^T$	$b_{lj} a_{ki}$	$\mathbf{B} \otimes \mathbf{A}$	$\mathbf{B}^T \otimes \mathbf{A}^T$
$\mathbf{A}^T \mathbf{X} \mathbf{B}^T$	$b_{lj} a_{ik}$	$\mathbf{B} \otimes \mathbf{A}^T$	$\mathbf{B}^T \otimes \mathbf{A}$
$\mathbf{A} \mathbf{X}^T \mathbf{B}$	$b_{il} a_{kj}$	$(\mathbf{B}^T \otimes \mathbf{A}) \mathbf{K}_{mn}$	$\mathbf{K}_{nm} (\mathbf{B} \otimes \mathbf{A}^T)$
$\mathbf{A}^T \mathbf{X}^T \mathbf{B}$	$b_{il} a_{jk}$	$(\mathbf{B}^T \otimes \mathbf{A}^T) \mathbf{K}_{mn}$	$\mathbf{K}_{nm} (\mathbf{B} \otimes \mathbf{A})$
$\mathbf{A} \mathbf{X}^T \mathbf{B}^T$	$b_{il} a_{kj}$	$(\mathbf{B} \otimes \mathbf{A}) \mathbf{K}_{mn}$	$\mathbf{K}_{nm} (\mathbf{B}^T \otimes \mathbf{A}^T)$
$\mathbf{A}^T \mathbf{X}^T \mathbf{B}^T$	$b_{il} a_{jk}$	$(\mathbf{B} \otimes \mathbf{A}^T) \mathbf{K}_{mn}$	$\mathbf{K}_{nm} (\mathbf{B}^T \otimes \mathbf{A})$
$\mathbf{X} \mathbf{X}^T$	$\delta_{il} x_{kj} + x_{lj} \delta_{ki}$	$(\mathbf{K}_{mm} + \mathbf{I}_{m^2})(\mathbf{X} \otimes \mathbf{I}_m)$	$(\mathbf{X}^T \otimes \mathbf{I}_m)(\mathbf{K}_{mm} + \mathbf{I}_{m^2})$
$\mathbf{X}^T \mathbf{X}$	$x_{il} \delta_{kj} + \delta_{lj} x_{ik}$	$(\mathbf{K}_{nn} + \mathbf{I}_{n^2})(\mathbf{I}_n \otimes \mathbf{X}^T)$	$(\mathbf{I}_n \otimes \mathbf{X})(\mathbf{K}_{nn} + \mathbf{I}_{n^2})$
$\mathbf{X}^T \mathbf{B} \mathbf{X}$	$(\mathbf{B} \mathbf{X})_{il} \delta_{kj} + \delta_{lj} (\mathbf{X}^T \mathbf{B})_{ki}$	$((\mathbf{B} \mathbf{X})^T \otimes \mathbf{I}_n) \mathbf{K}_{mn} + (\mathbf{I}_n \otimes (\mathbf{X}^T \mathbf{B}))$	$\mathbf{K}_{nm} ((\mathbf{B} \mathbf{X}) \otimes \mathbf{I}_n) + (\mathbf{I}_n \otimes (\mathbf{B}^T \mathbf{X}))$
$\mathbf{X} \mathbf{B} \mathbf{X}^T$	$(\mathbf{B} \mathbf{X}^T)_{jl} \delta_{ki} + \delta_{li} (\mathbf{X} \mathbf{B})_{kj}$	$(\mathbf{X} \mathbf{B}^T) \otimes \mathbf{I}_m + (\mathbf{I}_m \otimes (\mathbf{X} \mathbf{B})) \mathbf{K}_{mn}$	$(\mathbf{B} \mathbf{X}^T) \otimes \mathbf{I}_m + \mathbf{K}_{nm} (\mathbf{I}_m \otimes (\mathbf{X} \mathbf{B})^T)$

若  $\mathbf{X} = \mathbf{x} \in \mathbb{R}^{m \times 1}$ , 则由表 3.1.2 知

$$\mathbf{D}_x(\mathbf{x} \mathbf{x}^T) = \mathbf{I}_{m^2} (\mathbf{x} \otimes \mathbf{I}_m) + \mathbf{K}_{mm} (\mathbf{x} \otimes \mathbf{I}_m) = (\mathbf{x} \otimes \mathbf{I}_m) + (\mathbf{I}_m \otimes \mathbf{x}) \quad (3.1.41)$$

因为  $\mathbf{K}_{mm} (\mathbf{x} \otimes \mathbf{I}_m) = (\mathbf{I}_m \otimes \mathbf{x}) \mathbf{K}_{m1}$  以及  $\mathbf{K}_{m1} = \mathbf{I}_m$ 。类似地, 可得

$$\mathbf{D}_x(\mathbf{x} \mathbf{x}^T) = (\mathbf{K}_{11} + \mathbf{I}_1) (\mathbf{I}_1 \otimes \mathbf{x}^T) = 2\mathbf{x}^T \quad (3.1.42)$$

应当指出, 虽然直接计算偏导  $\partial f_{kl}/\partial x_{ij}$  可以正确求出很多矩阵函数的 Jacobian 矩阵和梯度矩阵, 但是对于复杂的矩阵函数 (例如矩阵的逆矩阵、Moore-Penrose 逆矩阵和矩阵的指数函数等), 偏导  $\partial f_{kl}/\partial x_{ij}$  的计算就比较繁琐和困难。因此, 自然希望有一种容易记忆和掌握的数学工具, 能够有效地计算实值标量函数和实值矩阵函数的 Jacobian 矩阵或梯度矩阵。这正是 3.2 节要讨论的主题。

## 3.2 一阶实矩阵微分与 Jacobian 矩阵辨识

矩阵微分是计算标量、向量或者矩阵函数关于其向量或矩阵变元的偏导的有效数学工具。本节主要介绍一阶实矩阵微分的有关理论、计算方法及应用。

### 3.2.1 一阶实矩阵微分

矩阵微分用符号  $d\mathbf{X}$  表示, 定义为  $d\mathbf{X} = [dX_{ij}]_{i=1,j=1}^{m,n}$ 。

**例 3.2.1** 考虑标量函数  $\text{tr}(\mathbf{U})$  的微分, 得

$$d(\text{tr } \mathbf{U}) = d\left(\sum_{i=1}^n u_{ii}\right) = \sum_{i=1}^n du_{ii} = \text{tr}(d\mathbf{U})$$

即有  $d(\text{tr } \mathbf{U}) = \text{tr}(d\mathbf{U})$ 。

**例 3.2.2** 考虑矩阵乘积  $\mathbf{UV}$  的微分矩阵, 有

$$\begin{aligned} [d(\mathbf{UV})]_{ij} &= d([\mathbf{UV}]_{ij}) = d\left(\sum_k u_{ik}v_{kj}\right) = \sum_k d(u_{ik}v_{kj}) \\ &= \sum_k [(du_{ik})v_{kj} + u_{ik}dv_{kj}] = \sum_k (du_{ik})v_{kj} + \sum_k u_{ik}dv_{kj} \\ &= [(d\mathbf{U})\mathbf{V}]_{ij} + [\mathbf{U}d\mathbf{V}]_{ij} \end{aligned}$$

从而得  $d(\mathbf{UV}) = (d\mathbf{U})\mathbf{V} + \mathbf{U}d\mathbf{V}$ 。

以上举例表明, 实矩阵微分具有以下两个基本性质:

**转置** 矩阵转置的微分等于矩阵微分的转置, 即有  $d(\mathbf{X}^T) = (d\mathbf{X})^T$ 。

**线性**  $d(\alpha\mathbf{X} + \beta\mathbf{Y}) = \alpha d\mathbf{X} + \beta d\mathbf{Y}$ 。

下面汇总了矩阵微分的常用计算公式 [328, pp.148~154]。

- (1) 常数矩阵的微分矩阵为零矩阵, 即  $d\mathbf{A} = \mathbf{0}$ 。
- (2) 常数  $\alpha$  与矩阵  $\mathbf{X}$  的乘积的微分矩阵  $d(\alpha\mathbf{X}) = \alpha d\mathbf{X}$ 。
- (3) 矩阵转置的微分矩阵等于原矩阵的微分矩阵的转置, 即  $d(\mathbf{X}^T) = (d\mathbf{X})^T$ 。
- (4) 两个矩阵函数的和 (差) 的微分矩阵为  $d(\mathbf{U} \pm \mathbf{V}) = d\mathbf{U} \pm d\mathbf{V}$ 。
- (5) 常数矩阵与矩阵乘积的微分矩阵为  $d(\mathbf{AXB}) = \mathbf{A}(d\mathbf{X})\mathbf{B}$ 。
- (6) 矩阵函数  $\mathbf{U} = \mathbf{F}(\mathbf{X}), \mathbf{V} = \mathbf{G}(\mathbf{X}), \mathbf{W} = \mathbf{H}(\mathbf{X})$  乘积的微分矩阵为

$$d(\mathbf{UV}) = (d\mathbf{U})\mathbf{V} + \mathbf{U}(d\mathbf{V}) \quad (3.2.1)$$

$$d(\mathbf{UVW}) = (d\mathbf{U})\mathbf{VW} + \mathbf{U}(d\mathbf{V})\mathbf{W} + \mathbf{UV}(d\mathbf{W}) \quad (3.2.2)$$

- (7) 矩阵  $\mathbf{X}$  的迹的矩阵微分  $d(\text{tr}(\mathbf{X}))$  等于矩阵微分  $d\mathbf{X}$  的迹  $\text{tr}(d\mathbf{X})$ , 即

$$d(\text{tr}(\mathbf{X})) = \text{tr}(d\mathbf{X}) \quad (3.2.3)$$

特别地, 矩阵函数  $\mathbf{F}(\mathbf{X})$  的迹的矩阵微分为  $d(\text{tr}(\mathbf{F}(\mathbf{X}))) = \text{tr}(d(\mathbf{F}(\mathbf{X})))$ 。

(8) 行列式的微分为

$$d|\mathbf{X}| = |\mathbf{X}| \text{tr}(\mathbf{X}^{-1} d\mathbf{X}) \quad (3.2.4)$$

特别地, 矩阵函数  $\mathbf{F}(\mathbf{X})$  的行列式的微分为  $d|\mathbf{F}(\mathbf{X})| = |\mathbf{F}(\mathbf{X})| \text{tr}(\mathbf{F}^{-1}(\mathbf{X}) d(\mathbf{F}(\mathbf{X})))$ 。

(9) 矩阵函数的 Kronecker 积的微分矩阵为

$$d(\mathbf{U} \otimes \mathbf{V}) = (d\mathbf{U}) \otimes \mathbf{V} + \mathbf{U} \otimes d\mathbf{V} \quad (3.2.5)$$

(10) 矩阵函数的 Hadamard 积的微分矩阵为

$$d(\mathbf{U} * \mathbf{V}) = (d\mathbf{U}) * \mathbf{V} + \mathbf{U} * d\mathbf{V} \quad (3.2.6)$$

(11) 向量化函数  $\text{vec}(\mathbf{X})$  的微分矩阵等于  $\mathbf{X}$  的微分矩阵的向量化函数, 即

$$d(\text{vec}(\mathbf{X})) = \text{vec}(d\mathbf{X}) \quad (3.2.7)$$

(12) 矩阵对数的微分矩阵为

$$d \log \mathbf{X} = \mathbf{X}^{-1} d\mathbf{X} \quad (3.2.8)$$

特别地, 矩阵函数的对数的微分矩阵为  $d \log(\mathbf{F}(\mathbf{X})) = \mathbf{F}^{-1}(\mathbf{X}) d(\mathbf{F}(\mathbf{X}))$ 。

(13) 逆矩阵的微分矩阵为

$$d(\mathbf{X}^{-1}) = -\mathbf{X}^{-1} (d\mathbf{X}) \mathbf{X}^{-1} \quad (3.2.9)$$

(14) Moore-Penrose 逆矩阵的微分矩阵为

$$\begin{aligned} d(\mathbf{X}^\dagger) &= -\mathbf{X}^\dagger (d\mathbf{X}) \mathbf{X}^\dagger + \mathbf{X}^\dagger (\mathbf{X}^\dagger)^T (d\mathbf{X}^T) (\mathbf{I} - \mathbf{X} \mathbf{X}^\dagger) \\ &\quad + (\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}) (d\mathbf{X}^T) (\mathbf{X}^\dagger)^T \mathbf{X}^\dagger \end{aligned} \quad (3.2.10)$$

$$d(\mathbf{X}^\dagger \mathbf{X}) = \mathbf{X}^\dagger (d\mathbf{X}) (\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}) + \left( \mathbf{X}^\dagger (d\mathbf{X}) (\mathbf{I} - \mathbf{X}^\dagger \mathbf{X}) \right)^T \quad (3.2.11)$$

$$d(\mathbf{X} \mathbf{X}^\dagger) = (\mathbf{I} - \mathbf{X} \mathbf{X}^\dagger) (d\mathbf{X}) \mathbf{X}^\dagger + \left( (\mathbf{I} - \mathbf{X} \mathbf{X}^\dagger) (d\mathbf{X}) \mathbf{X}^\dagger \right)^T \quad (3.2.12)$$

### 3.2.2 标量函数的 Jacobian 矩阵辨识

在多变量函数的微积分中, 称多变量函数  $f(x_1, \dots, x_m)$  在点  $(x_1, \dots, x_m)$  可微分, 若  $f(x_1, \dots, x_m)$  的全改变量可以写作

$$\begin{aligned} \Delta f(x_1, \dots, x_m) &= f(x_1 + \Delta x_1, \dots, x_m + \Delta x_m) - f(x_1, \dots, x_m) \\ &= A_1 \Delta x_1 + \dots + A_m \Delta x_m + O(\Delta x_1, \dots, \Delta x_m) \end{aligned} \quad (3.2.13)$$

式中,  $A_1, \dots, A_m$  分别与  $\Delta x_1, \dots, \Delta x_m$  无关, 而  $O(\Delta x_1, \dots, \Delta x_m)$  表示偏改变量  $\Delta x_1, \dots, \Delta x_m$  的二阶及高阶项。这时, 函数  $f(x_1, \dots, x_m)$  的偏导数  $\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_m}$  一定存在,

并且

$$\frac{\partial f}{\partial x_1} = A_1, \dots, \frac{\partial f}{\partial x_m} = A_m$$

全改变量  $\Delta f(x_1, \dots, x_m)$  的线性主部

$$A_1 \Delta x_1 + \dots + A_m \Delta x_m = \frac{\partial f}{\partial x_1} dx_1 + \dots + \frac{\partial f}{\partial x_m} dx_m$$

称为多变量函数  $f(x_1, \dots, x_m)$  的全微分, 记为

$$df(x_1, \dots, x_m) = \frac{\partial f}{\partial x_1} dx_1 + \dots + \frac{\partial f}{\partial x_m} dx_m \quad (3.2.14)$$

多变量函数  $f(x_1, \dots, x_m)$  在点  $(x_1, \dots, x_m)$  可微分的充分条件是: 偏导数  $\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_m}$  均存在, 并且连续。

一阶实矩阵微分为 Jacobian 矩阵的辨识提供了一种有效的方法。

### 1. 标量函数 $f(\mathbf{x})$ 的 Jacobian 矩阵辨识

考虑标量函数  $f(\mathbf{x})$ , 其变元向量  $\mathbf{x} = [x_1, \dots, x_m]^T \in \mathbb{R}^m$ 。将变元向量的元素  $x_1, \dots, x_m$  视为  $m$  个变量, 利用式 (3.2.14), 可以直接引出以向量为变元的标量函数  $f(\mathbf{x})$  的全微分表达式

$$\begin{aligned} df(\mathbf{x}) &= \frac{\partial f(\mathbf{x})}{\partial x_1} dx_1 + \dots + \frac{\partial f(\mathbf{x})}{\partial x_m} dx_m \\ &= \left[ \frac{\partial f(\mathbf{x})}{\partial x_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_m} \right] \begin{bmatrix} dx_1 \\ \vdots \\ dx_m \end{bmatrix} \end{aligned} \quad (3.2.15)$$

或简记为

$$df(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} d\mathbf{x} = (\mathbf{d}\mathbf{x})^T \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \quad (3.2.16)$$

式中

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} = \left[ \frac{\partial f(\mathbf{x})}{\partial x_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_m} \right] \quad (3.2.17)$$

$$d\mathbf{x} = [dx_1, \dots, dx_m]^T \quad (3.2.18)$$

式 (3.2.16) 称为微分法则的向量形式, 它启示了一个重要的应用: 若令  $\mathbf{A} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T}$ , 则一阶微分可以写作迹函数形式

$$df(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} d\mathbf{x} = \text{tr}(\mathbf{A} d\mathbf{x}) \quad (3.2.19)$$

这表明, 标量函数  $f(\mathbf{x})$  的 Jacobian 矩阵与微分矩阵之间存在等价关系

$$df(\mathbf{x}) = \text{tr}(\mathbf{A} d\mathbf{x}) \iff D_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} = \mathbf{A} \quad (3.2.20)$$

换言之, 若函数  $f(\mathbf{x})$  的微分可以写作  $df(\mathbf{x}) = \text{tr}(\mathbf{A}d\mathbf{x})$ , 则矩阵  $\mathbf{A}$  就是函数  $f(\mathbf{x})$  关于其变元向量  $\mathbf{x}$  的 Jacobian 矩阵。

## 2. 标量函数 $f(\mathbf{X})$ 的 Jacobian 矩阵辨识

进一步考查标量函数  $f(\mathbf{X})$ , 其变元为  $m \times n$  实矩阵  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ 。记  $\mathbf{x}_j = [x_{1j}, \dots, x_{mj}]^T, j = 1, \dots, n$ , 则由标量函数  $f(\mathbf{x})$  的全微分公式 (3.2.15) 易知, 实值矩阵作变元的标量函数  $f(\mathbf{X})$  的全微分为

$$\begin{aligned} df(\mathbf{X}) &= \frac{\partial f(\mathbf{X})}{\partial \mathbf{x}_1} d\mathbf{x}_1 + \dots + \frac{\partial f(\mathbf{X})}{\partial \mathbf{x}_n} d\mathbf{x}_n \\ &= \left[ \frac{\partial f(\mathbf{X})}{\partial x_{11}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{m1}} \right] \begin{bmatrix} dx_{11} \\ \vdots \\ dx_{m1} \end{bmatrix} + \dots + \left[ \frac{\partial f(\mathbf{X})}{\partial x_{1n}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \right] \begin{bmatrix} dx_{1n} \\ \vdots \\ dx_{mn} \end{bmatrix} \\ &= \left[ \frac{\partial f(\mathbf{X})}{\partial x_{11}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{m1}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{1n}}, \dots, \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \right] \begin{bmatrix} dx_{11} \\ \vdots \\ dx_{m1} \\ \vdots \\ dx_{1n} \\ \vdots \\ dx_{mn} \end{bmatrix} \\ &= \frac{\partial f(\mathbf{X})}{\partial \text{vec}^T(\mathbf{X})} d(\text{vec} \mathbf{X}) = D_{\text{vec} \mathbf{X}} f(\mathbf{X}) d(\text{vec} \mathbf{X}) \end{aligned} \quad (3.2.21)$$

利用行向量偏导与 Jacobian 矩阵的关系  $D_{\text{vec} \mathbf{X}} f(\mathbf{X}) = (\text{vec}(D_{\mathbf{X}}^T f(\mathbf{X})))^T$ , 式 (3.2.21) 可以改写为

$$df(\mathbf{X}) = (\text{vec}(\mathbf{A}^T))^T d(\text{vec} \mathbf{X}) \quad (3.2.22)$$

式中

$$\mathbf{A} = D_{\mathbf{X}} f(\mathbf{X}) = \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}^T} = \begin{bmatrix} \frac{\partial f(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f(\mathbf{X})}{\partial x_{m1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{X})}{\partial x_{1n}} & \dots & \frac{\partial f(\mathbf{X})}{\partial x_{mn}} \end{bmatrix} \quad (3.2.23)$$

是标量函数  $f(\mathbf{X})$  的 Jacobian 矩阵。

利用向量化算子  $\text{vec}$  与迹函数之间的关系式  $\text{tr}(\mathbf{B}^T \mathbf{C}) = (\text{vec}(\mathbf{B}))^T \text{vec}(\mathbf{C})$ , 令  $\mathbf{B} = \mathbf{A}^T$  和  $\mathbf{C} = d\mathbf{X}$ , 则式 (3.2.22) 可以用迹函数表示为

$$df(\mathbf{X}) = \text{tr}(\mathbf{A} d\mathbf{X}) \quad (3.2.24)$$

综合以上讨论, 有下面的命题。

**命题 3.2.1** 若矩阵的标量函数  $f(\mathbf{X})$  在  $m \times n$  矩阵点  $\mathbf{X}$  可微分, 则 Jacobian 矩阵

可以通过下式直接辨识

$$df(\mathbf{x}) = \text{tr}(\mathbf{A}d\mathbf{x}) \iff D_{\mathbf{x}}f(\mathbf{x}) = \mathbf{A} \quad (3.2.25)$$

$$df(\mathbf{X}) = \text{tr}(\mathbf{A}d\mathbf{X}) \iff D_{\mathbf{X}}f(\mathbf{X}) = \mathbf{A} \quad (3.2.26)$$

命题 3.2.1 启示了利用矩阵微分直接辨识标量函数  $f(\mathbf{X})$  的 Jacobian 矩阵  $D_{\mathbf{X}}f(\mathbf{X})$  的有效方法：

(1) 求实值函数  $f(\mathbf{X})$  相对于变元矩阵  $\mathbf{X}$  的矩阵微分  $df(\mathbf{X})$ ，并将其表示成规范形式  $df(\mathbf{X}) = \text{tr}(\mathbf{A}d\mathbf{X})$ ；

(2) 实值函数  $f(\mathbf{X})$  相对于  $m \times n$  变元矩阵  $\mathbf{X}$  的 Jacobian 矩阵由  $\mathbf{A}$  直接给出。

业已证明<sup>[328]</sup>，Jacobian 矩阵  $\mathbf{A}$  是唯一确定的：若存在  $\mathbf{A}_1$  和  $\mathbf{A}_2$  满足  $df(\mathbf{X}) = \mathbf{A}_i d\mathbf{X}, i = 1, 2$ ，则  $\mathbf{A}_1 = \mathbf{A}_2$ 。

由于标量函数  $f(\mathbf{X})$  相对于  $m \times n$  矩阵变元  $\mathbf{X}$  的 Jacobian 矩阵和梯度矩阵之间存在转置关系，所以命题 3.2.1 也意味着

$$df(\mathbf{X}) = \text{tr}(\mathbf{A}d\mathbf{X}) \iff \nabla_{\mathbf{X}}f(\mathbf{X}) = \mathbf{A}^T \quad (3.2.27)$$

由于 Jacobian 矩阵  $\mathbf{A}$  的唯一确定性，故梯度矩阵是唯一确定的。

考察二次型函数  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x}$ ，其中， $\mathbf{A}$  是一个正方的常数矩阵。首先将标量函数写成迹函数形式，然后利用矩阵乘积的微分易得

$$\begin{aligned} df(\mathbf{x}) &= d(\text{tr}(\mathbf{x}^T \mathbf{A} \mathbf{x})) = \text{tr}[(d\mathbf{x})^T \mathbf{A} \mathbf{x} + \mathbf{x}^T \mathbf{A} d\mathbf{x}] \\ &= \text{tr}[(d\mathbf{x}^T \mathbf{A} \mathbf{x})^T + \mathbf{x}^T \mathbf{A} d\mathbf{x}] = \text{tr}(\mathbf{x}^T \mathbf{A}^T d\mathbf{x} + \mathbf{x}^T \mathbf{A} d\mathbf{x}) \\ &= \text{tr}(\mathbf{x}^T (\mathbf{A} + \mathbf{A}^T) d\mathbf{x}) \end{aligned}$$

由命题 3.2.1 直接得二次型函数  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x}$  关于变元向量  $\mathbf{x}$  的梯度向量为

$$\nabla_{\mathbf{x}}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = \frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} = [\mathbf{x}^T (\mathbf{A} + \mathbf{A}^T)]^T = (\mathbf{A}^T + \mathbf{A}) \mathbf{x} \quad (3.2.28)$$

显然，若  $\mathbf{A}$  为对称矩阵，则  $\nabla_{\mathbf{x}}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = \frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{A}\mathbf{x}$ 。

### 3. 矩阵的标量函数：迹

对于  $\text{tr}(\mathbf{X}^T \mathbf{X})$ ，注意到  $\text{tr}(\mathbf{A}^T \mathbf{B}) = \text{tr}(\mathbf{B}^T \mathbf{A})$ ，有

$$\begin{aligned} d\text{tr}(\mathbf{X}^T \mathbf{X}) &= \text{tr}(d(\mathbf{X}^T \mathbf{X})) = \text{tr}((d\mathbf{X})^T \mathbf{X} + \mathbf{X}^T d\mathbf{X}) \\ &= \text{tr}((d\mathbf{X})^T \mathbf{X}) + \text{tr}(\mathbf{X}^T d\mathbf{X}) \\ &= \text{tr}(2\mathbf{X}^T d\mathbf{X}) \end{aligned}$$

故由命题 3.2.1 直接得  $\text{tr}(\mathbf{X}^T \mathbf{X})$  关于  $\mathbf{X}$  的梯度矩阵为

$$\frac{\partial \text{tr}(\mathbf{X}^T \mathbf{X})}{\partial \mathbf{X}} = (2\mathbf{X}^T)^T = 2\mathbf{X} \quad (3.2.29)$$

考虑三个矩阵乘积的迹函数  $\text{tr}(\mathbf{X}^T \mathbf{A} \mathbf{X})$ , 其微分

$$\begin{aligned}\text{d} \text{tr}(\mathbf{X}^T \mathbf{A} \mathbf{X}) &= \text{tr}(\text{d}(\mathbf{X}^T \mathbf{A} \mathbf{X})) \\ &= \text{tr}((\text{d}\mathbf{X})^T \mathbf{A} \mathbf{X} + \mathbf{X}^T \mathbf{A} \text{d}\mathbf{X}) \\ &= \text{tr}((\text{d}\mathbf{X})^T \mathbf{A} \mathbf{X}) + \text{tr}(\mathbf{X}^T \mathbf{A} \text{d}\mathbf{X}) \\ &= \text{tr}((\mathbf{A} \mathbf{X})^T \text{d}\mathbf{X}) + \text{tr}(\mathbf{X}^T \mathbf{A} \text{d}\mathbf{X}) \\ &= \text{tr}(\mathbf{X}^T (\mathbf{A}^T + \mathbf{A}) \text{d}\mathbf{X})\end{aligned}$$

从而得梯度矩阵

$$\frac{\partial \text{tr}(\mathbf{X}^T \mathbf{A} \mathbf{X})}{\partial \mathbf{X}} = [\mathbf{X}^T (\mathbf{A}^T + \mathbf{A})]^T = (\mathbf{A} + \mathbf{A}^T) \mathbf{X} \quad (3.2.30)$$

再看一个包含了逆矩阵的迹函数  $\text{tr}(\mathbf{A} \mathbf{X}^{-1})$ 。计算得

$$\begin{aligned}\text{d} \text{tr}(\mathbf{A} \mathbf{X}^{-1}) &= \text{tr}[\text{d}(\mathbf{A} \mathbf{X}^{-1})] = \text{tr}[\mathbf{A} \text{d}\mathbf{X}^{-1}] \\ &= -\text{tr}[\mathbf{A} \mathbf{X}^{-1} (\text{d}\mathbf{X}) \mathbf{X}^{-1}] = -\text{tr}(\mathbf{X}^{-1} \mathbf{A} \mathbf{X}^{-1} \text{d}\mathbf{X})\end{aligned}$$

由此得梯度矩阵

$$\frac{\partial \text{tr}(\mathbf{A} \mathbf{X}^{-1})}{\partial \mathbf{X}} = -(\mathbf{X}^{-1} \mathbf{A} \mathbf{X}^{-1})^T \quad (3.2.31)$$

对于四个矩阵乘积的迹函数  $\text{tr}(\mathbf{X} \mathbf{A} \mathbf{X} \mathbf{B})$ , 其微分矩阵

$$\begin{aligned}\text{d} \text{tr}(\mathbf{X} \mathbf{A} \mathbf{X} \mathbf{B}) &= \text{tr}[\text{d}(\mathbf{X} \mathbf{A} \mathbf{X} \mathbf{B})] \\ &= \text{tr}[(\text{d}\mathbf{X}) \mathbf{A} \mathbf{X} \mathbf{B} + \mathbf{X} \mathbf{A} (\text{d}\mathbf{X}) \mathbf{B}] \\ &= \text{tr}[(\mathbf{A} \mathbf{X} \mathbf{B} + \mathbf{B} \mathbf{X} \mathbf{A}) \text{d}\mathbf{X}]\end{aligned}$$

由此得梯度矩阵

$$\frac{\partial \text{tr}(\mathbf{X} \mathbf{A} \mathbf{X} \mathbf{B})}{\partial \mathbf{X}} = (\mathbf{A} \mathbf{X} \mathbf{B} + \mathbf{B} \mathbf{X} \mathbf{A})^T \quad (3.2.32)$$

以上举例可以总结出应用命题 3.2.1 的要点如下:

- (1) 标量函数  $f(\mathbf{X})$  总可以写成迹函数的形式, 因为  $f(\mathbf{X}) = \text{tr}(f(\mathbf{X}))$ ;
- (2) 无论  $\text{d}\mathbf{X}$  出现在迹函数内的任何位置, 总可以通过迹函数的性质  $\text{tr}[\mathbf{A}(\text{d}\mathbf{X}) \mathbf{B}] = \text{tr}(\mathbf{B} \mathbf{A} \text{d}\mathbf{X})$ , 将  $\text{d}\mathbf{X}$  写到迹函数变量的最右端, 从而得到迹函数微分矩阵的规范形式。
- (3) 对于  $(\text{d}\mathbf{X})^T$ , 总可以通过迹函数的性质  $\text{tr}[\mathbf{A}(\text{d}\mathbf{X})^T \mathbf{B}] = \text{tr}(\mathbf{A}^T \mathbf{B}^T \text{d}\mathbf{X})$ , 写成迹函数微分矩阵的规范形式。

表 3.2.1 汇总了几种典型的迹函数的微分矩阵与梯度矩阵的对应关系。

#### 4. 矩阵的标量函数: 行列式

表 3.2.1 几种迹函数的微分矩阵与 Jacobian 矩阵<sup>[328]</sup>

迹函数 $f(\mathbf{X})$	微分矩阵 $df(\mathbf{X})$	Jacobian 矩阵 $\partial f(\mathbf{X})/\partial \mathbf{X}^T$
$\text{tr}(\mathbf{X})$	$\text{tr}(Id\mathbf{X})$	$I$
$\text{tr}(\mathbf{X}^{-1})$	$-\text{tr}(\mathbf{X}^{-2}d\mathbf{X})$	$-\mathbf{X}^{-2}$
$\text{tr}(\mathbf{AX})$	$\text{tr}(\mathbf{Ad}\mathbf{X})$	$\mathbf{A}$
$\text{tr}(\mathbf{X}^2)$	$2\text{tr}(\mathbf{Xd}\mathbf{X})$	$2\mathbf{X}$
$\text{tr}(\mathbf{X}^T\mathbf{X})$	$2\text{tr}(\mathbf{X}^T d\mathbf{X})$	$2\mathbf{X}^T$
$\text{tr}(\mathbf{X}^T\mathbf{AX})$	$\text{tr}[\mathbf{X}^T(\mathbf{A} + \mathbf{A}^T)d\mathbf{X}]$	$\mathbf{X}^T(\mathbf{A} + \mathbf{A}^T)$
$\text{tr}(\mathbf{XA}\mathbf{X}^T)$	$\text{tr}[(\mathbf{A} + \mathbf{A}^T)\mathbf{X}^T d\mathbf{X}]$	$(\mathbf{A} + \mathbf{A}^T)\mathbf{X}^T$
$\text{tr}(\mathbf{X}\mathbf{AX})$	$\text{tr}[(\mathbf{AX} + \mathbf{XA})d\mathbf{X}]$	$\mathbf{AX} + \mathbf{XA}$
$\text{tr}(\mathbf{AX}^{-1})$	$-\text{tr}(\mathbf{X}^{-1}\mathbf{AX}^{-1}d\mathbf{X})$	$-\mathbf{X}^{-1}\mathbf{AX}^{-1}$
$\text{tr}(\mathbf{AX}^{-1}\mathbf{B})$	$-\text{tr}(\mathbf{X}^{-1}\mathbf{BAX}^{-1}d\mathbf{X})$	$-\mathbf{X}^{-1}\mathbf{BAX}^{-1}$
$\text{tr}[(\mathbf{X} + \mathbf{A})^{-1}]$	$-\text{tr}[(\mathbf{X} + \mathbf{A})^{-2}d\mathbf{X}]$	$-(\mathbf{X} + \mathbf{A})^{-2}$
$\text{tr}(\mathbf{XA}\mathbf{XB})$	$\text{tr}[(\mathbf{AXB} + \mathbf{BXA})d\mathbf{X}]$	$\mathbf{AXB} + \mathbf{BXA}$
$\text{tr}(\mathbf{XA}\mathbf{X}^T\mathbf{B})$	$\text{tr}[(\mathbf{AX}^T\mathbf{B} + \mathbf{A}^T\mathbf{X}^T\mathbf{B}^T)d\mathbf{X}]$	$\mathbf{AX}^T\mathbf{B} + \mathbf{A}^T\mathbf{X}^T\mathbf{B}^T$
$\text{tr}(\mathbf{AXX}^T\mathbf{B})$	$\text{tr}[\mathbf{X}^T(\mathbf{BA} + \mathbf{A}^T\mathbf{B}^T)d\mathbf{X}]$	$\mathbf{X}^T(\mathbf{BA} + \mathbf{A}^T\mathbf{B}^T)$
$\text{tr}(\mathbf{AX}^T\mathbf{XB})$	$\text{tr}[(\mathbf{BA} + \mathbf{A}^T\mathbf{B}^T)\mathbf{X}^T d\mathbf{X}]$	$(\mathbf{BA} + \mathbf{A}^T\mathbf{B}^T)\mathbf{X}^T$

表中,  $\mathbf{A}^{-2} = \mathbf{A}^{-1}\mathbf{A}^{-1}$ 。

由矩阵微分  $d|\mathbf{X}| = |\mathbf{X}|\text{tr}(\mathbf{X}^{-1}d\mathbf{X})$  和命题 3.2.1, 立即得行列式的梯度矩阵为

$$\frac{\partial |\mathbf{X}|}{\partial \mathbf{X}} = |\mathbf{X}|(\mathbf{X}^{-1})^T = |\mathbf{X}|\mathbf{X}^{-T} \quad (3.2.33)$$

又如, 考虑行列式的对数  $\log |\mathbf{X}|$ , 其矩阵微分为

$$d \log |\mathbf{X}| = |\mathbf{X}|^{-1}d|\mathbf{X}| = |\mathbf{X}|^{-1}\text{tr}(|\mathbf{X}|\mathbf{X}^{-1}d\mathbf{X}) = \text{tr}(\mathbf{X}^{-1}d\mathbf{X}) \quad (3.2.34)$$

故行列式对数函数  $\log |\mathbf{X}|$  的梯度矩阵为

$$\frac{\partial \log |\mathbf{X}|}{\partial \mathbf{X}} = \mathbf{X}^{-T} \quad (3.2.35)$$

考虑  $\mathbf{X}^2$  的行列式。由矩阵函数  $\mathbf{U} = \mathbf{F}(\mathbf{X})$  的行列式的微分  $d|\mathbf{U}| = |\mathbf{U}|\text{tr}(\mathbf{U}^{-1}d\mathbf{X})$  知,  $d|\mathbf{X}^2| = d|\mathbf{X}|^2 = 2|\mathbf{X}|d|\mathbf{X}| = 2|\mathbf{X}|^2\text{tr}(\mathbf{X}^{-1}d\mathbf{X})$ 。应用命题 3.2.1, 立即得

$$\frac{\partial |\mathbf{X}|^2}{\partial \mathbf{X}} = 2|\mathbf{X}|^2(\mathbf{X}^{-1})^T = 2|\mathbf{X}|^2\mathbf{X}^{-T} \quad (3.2.36)$$

更一般地,  $|\mathbf{X}^k|$  的矩阵微分为

$$\begin{aligned} d|\mathbf{X}^k| &= |\mathbf{X}^k|\text{tr}(\mathbf{X}^{-k}d\mathbf{X}^k) \\ &= |\mathbf{X}^k|\text{tr}(\mathbf{X}^{-k} \cdot k\mathbf{X}^{k-1}d\mathbf{X}) \\ &= k|\mathbf{X}^k|\text{tr}(\mathbf{X}^{-1}d\mathbf{X}) \end{aligned}$$

于是有

$$\frac{\partial |\mathbf{X}^k|}{\partial \mathbf{X}} = k|\mathbf{X}^k|\mathbf{X}^{-T} \quad (3.2.37)$$

令  $\mathbf{X} \in \mathbb{R}^{m \times n}$ , 并且  $\text{rank}(\mathbf{X}) = m$  即  $\mathbf{X}\mathbf{X}^T$  可逆, 则对于矩阵乘积  $\mathbf{X}\mathbf{X}^T$  的行列式, 有

$$\begin{aligned} d|\mathbf{X}\mathbf{X}^T| &= |\mathbf{X}\mathbf{X}^T| \text{tr}((\mathbf{X}\mathbf{X}^T)^{-1}d(\mathbf{X}\mathbf{X}^T)) \\ &= |\mathbf{X}\mathbf{X}^T| [\text{tr}((\mathbf{X}\mathbf{X}^T)^{-1}(d\mathbf{X})\mathbf{X}^T) + \text{tr}((\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}(d\mathbf{X})^T)] \\ &= |\mathbf{X}\mathbf{X}^T| [\text{tr}(\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}d\mathbf{X}) + \text{tr}(\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}d\mathbf{X})] \\ &= \text{tr}(2|\mathbf{X}\mathbf{X}^T|\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}d\mathbf{X}) \end{aligned}$$

式中, 使用了迹的性质公式  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$  和  $\text{tr}(\mathbf{A}^T\mathbf{B}) = \text{tr}(\mathbf{B}^T\mathbf{A})$ 。由命题 3.2.1 立即得梯度矩阵

$$\frac{\partial |\mathbf{X}\mathbf{X}^T|}{\partial \mathbf{X}} = 2|\mathbf{X}\mathbf{X}^T|(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X} \quad (3.2.38)$$

式中, 使用了矩阵转置与求逆可以交换顺序的性质, 即

$$[(\mathbf{X}\mathbf{X}^T)^{-1}]^T = [(\mathbf{X}\mathbf{X}^T)^T]^{-1} = (\mathbf{X}\mathbf{X}^T)^{-1}$$

类似地, 令  $\mathbf{X} \in \mathbb{R}^{m \times n}$ 。若  $\text{rank}(\mathbf{X}) = n$  即  $\mathbf{X}^T\mathbf{X}$  可逆, 则有

$$d|\mathbf{X}^T\mathbf{X}| = \text{tr}(2|\mathbf{X}^T\mathbf{X}|(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T d\mathbf{X}) \quad (3.2.39)$$

由此得

$$\frac{\partial |\mathbf{X}^T\mathbf{X}|}{\partial \mathbf{X}} = 2|\mathbf{X}^T\mathbf{X}|\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} \quad (3.2.40)$$

对于对数函数  $\log|\mathbf{X}^T\mathbf{X}|$ , 矩阵微分为

$$d \log |\mathbf{X}^T\mathbf{X}| = |\mathbf{X}^T\mathbf{X}|^{-1}d|\mathbf{X}^T\mathbf{X}| = 2\text{tr}((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T d\mathbf{X}) \quad (3.2.41)$$

故有

$$\frac{\partial \log |\mathbf{X}^T\mathbf{X}|}{\partial \mathbf{X}} = 2\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} \quad (3.2.42)$$

考虑三个矩阵乘积  $\mathbf{AXB}$  的行列式, 有

$$\begin{aligned} d|\mathbf{AXB}| &= |\mathbf{AXB}| \text{tr}((\mathbf{AXB})^{-1}d(\mathbf{AXB})) \\ &= |\mathbf{AXB}| \text{tr}((\mathbf{AXB})^{-1}\mathbf{A}(d\mathbf{X})\mathbf{B}) \\ &= |\mathbf{AXB}| \text{tr}(\mathbf{B}(\mathbf{AXB})^{-1}\mathbf{A}d\mathbf{X}) \end{aligned}$$

在得到最后一个式子时, 使用了  $\text{tr}(\mathbf{CB}) = \text{tr}(\mathbf{BC})$ 。

这样一来, 由命题 3.2.1 立即得

$$\frac{\partial |\mathbf{AXB}|}{\partial \mathbf{X}} = |\mathbf{AXB}|\mathbf{A}^T(\mathbf{B}^T\mathbf{X}^T\mathbf{A}^T)^{-1}\mathbf{B}^T \quad (3.2.43)$$

令  $f(\mathbf{X}) = |\mathbf{X}\mathbf{A}\mathbf{X}^T|$ , 则其微分为

$$\begin{aligned} d|\mathbf{X}\mathbf{A}\mathbf{X}^T| &= |\mathbf{X}\mathbf{A}\mathbf{X}^T| \operatorname{tr}((\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}d(\mathbf{X}\mathbf{A}\mathbf{X}^T)) \\ &= |\mathbf{X}\mathbf{A}\mathbf{X}^T| \left[ \operatorname{tr}((\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}(d\mathbf{X})\mathbf{A}\mathbf{X}^T) + \operatorname{tr}((\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{A}(d\mathbf{X})^T) \right] \\ &= |\mathbf{X}\mathbf{A}\mathbf{X}^T| \left[ \operatorname{tr}(\mathbf{A}\mathbf{X}^T(\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}d\mathbf{X}) + \operatorname{tr}((\mathbf{X}\mathbf{A})^T(\mathbf{X}\mathbf{A}^T\mathbf{X}^T)^{-1}d\mathbf{X}) \right] \\ &= |\mathbf{X}\mathbf{A}\mathbf{X}^T| \operatorname{tr}([\mathbf{A}\mathbf{X}^T(\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1} + (\mathbf{X}\mathbf{A})^T(\mathbf{X}\mathbf{A}^T\mathbf{X}^T)^{-1}]d\mathbf{X}) \end{aligned}$$

于是, 命题 3.2.1 给出梯度

$$\begin{aligned} \frac{\partial|\mathbf{X}\mathbf{A}\mathbf{X}^T|}{\partial\mathbf{X}} &= |\mathbf{X}\mathbf{A}\mathbf{X}^T| \left[ (\mathbf{X}\mathbf{A}^T\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{A}^T + (\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{A} \right] \quad (3.2.44) \\ &= 2|\mathbf{X}\mathbf{A}\mathbf{X}^T|(\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{A}, \quad \text{若 } \mathbf{A} \text{ 为对称矩阵} \end{aligned}$$

类似地, 行列式  $|\mathbf{X}^T\mathbf{A}\mathbf{X}|$  的梯度为

$$\begin{aligned} \frac{\partial|\mathbf{X}^T\mathbf{A}\mathbf{X}|}{\partial\mathbf{X}} &= |\mathbf{X}^T\mathbf{A}\mathbf{X}|[\mathbf{A}\mathbf{X}(\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1} + \mathbf{A}^T\mathbf{X}(\mathbf{X}^T\mathbf{A}^T\mathbf{X})^{-1}] \quad (3.2.45) \\ &= 2|\mathbf{X}^T\mathbf{A}\mathbf{X}|\mathbf{A}\mathbf{X}(\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}, \quad \text{若 } \mathbf{A} \text{ 为对称矩阵} \end{aligned}$$

表 3.2.2 汇总了一些典型的行列式函数的微分矩阵与梯度矩阵的对应关系。

表 3.2.2 几种行列式函数的实微分矩阵与 Jacobian 矩阵

行列式 $f(\mathbf{X})$	实微分矩阵 $df(\mathbf{X})$	Jacobian 矩阵 $\partial f(\mathbf{X})/\partial\mathbf{X}$
$ \mathbf{X} $	$ \mathbf{X}  \operatorname{tr}(\mathbf{X}^{-1}d\mathbf{X})$	$ \mathbf{X} \mathbf{X}^{-1}$
$\log \mathbf{X} $	$\operatorname{tr}(\mathbf{X}^{-1}d\mathbf{X})$	$\mathbf{X}^{-1}$
$ \mathbf{X}^{-1} $	$- \mathbf{X}^{-1} \operatorname{tr}(\mathbf{X}^{-1}d\mathbf{X})$	$- \mathbf{X}^{-1} \mathbf{X}^{-1}$
$ \mathbf{X}^2 $	$2 \mathbf{X} ^2\operatorname{tr}(\mathbf{X}^{-1}d\mathbf{X})$	$2 \mathbf{X} ^2\mathbf{X}^{-1}$
$ \mathbf{X}^k $	$k \mathbf{X} ^k\operatorname{tr}(\mathbf{X}^{-1}d\mathbf{X})$	$k \mathbf{X} ^k\mathbf{X}^{-1}$
$ \mathbf{X}\mathbf{X}^T $	$2 \mathbf{X}\mathbf{X}^T \operatorname{tr}(\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}d\mathbf{X})$	$2 \mathbf{X}\mathbf{X}^T \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$
$ \mathbf{X}^T\mathbf{X} $	$2 \mathbf{X}^T\mathbf{X} \operatorname{tr}((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T d\mathbf{X})$	$2 \mathbf{X}^T\mathbf{X} (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$
$\log \mathbf{X}^T\mathbf{X} $	$2\operatorname{tr}((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T d\mathbf{X})$	$2(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$
$ \mathbf{A}\mathbf{X}\mathbf{B} $	$ \mathbf{A}\mathbf{X}\mathbf{B} \operatorname{tr}(\mathbf{B}(\mathbf{A}\mathbf{X}\mathbf{B})^{-1}\mathbf{A}d\mathbf{X})$	$ \mathbf{A}\mathbf{X}\mathbf{B} \mathbf{B}(\mathbf{A}\mathbf{X}\mathbf{B})^{-1}\mathbf{A}$
$ \mathbf{X}\mathbf{A}\mathbf{X}^T $	$ \mathbf{X}\mathbf{A}\mathbf{X}^T \operatorname{tr}([\mathbf{A}\mathbf{X}^T(\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1} + (\mathbf{X}\mathbf{A})^T(\mathbf{X}\mathbf{A}^T\mathbf{X}^T)^{-1}]d\mathbf{X})$	$ \mathbf{X}\mathbf{A}\mathbf{X}^T [\mathbf{A}\mathbf{X}^T(\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1} + (\mathbf{X}\mathbf{A})^T(\mathbf{X}\mathbf{A}^T\mathbf{X}^T)^{-1}]$
$ \mathbf{X}^T\mathbf{A}\mathbf{X} $	$ \mathbf{X}^T\mathbf{A}\mathbf{X} \operatorname{tr}([(X^TAX)^{-T}(\mathbf{A}\mathbf{X})^T + (X^TAX)^{-1}\mathbf{X}^T\mathbf{A}]d\mathbf{X})$	$ \mathbf{X}^T\mathbf{A}\mathbf{X} [(X^TAX)^{-T}(\mathbf{A}\mathbf{X})^T + (X^TAX)^{-1}\mathbf{X}^T\mathbf{A}]$

### 3.2.3 实值矩阵函数的 Jacobian 矩阵辨识

令  $f_{kl} = f_{kl}(\mathbf{X})$  表示实值矩阵函数  $\mathbf{F}(\mathbf{X})$  的第  $k$  行、第  $l$  列的元素，则  $d f_{kl}(\mathbf{X}) = [d\mathbf{F}(\mathbf{X})]_{kl}$  表示以  $m \times n$  实值矩阵为变元的标量函数的微分。

由式 (3.2.21) 有

$$d f_{kl}(\mathbf{X}) = \left[ \frac{\partial f_{kl}(\mathbf{X})}{\partial x_{11}}, \dots, \frac{\partial f_{kl}(\mathbf{X})}{\partial x_{m1}}, \dots, \frac{\partial f_{kl}(\mathbf{X})}{\partial x_{1n}}, \dots, \frac{\partial f_{kl}(\mathbf{X})}{\partial x_{mn}} \right] \begin{bmatrix} dx_{11} \\ \vdots \\ dx_{m1} \\ \vdots \\ dx_{1n} \\ \vdots \\ dx_{mn} \end{bmatrix} \quad (3.2.46)$$

利用这一结果易知，全微分矩阵的向量化函数  $d(\text{vec}\mathbf{F}(\mathbf{X}))$  具有以下表达式

$$d(\text{vec}\mathbf{F}(\mathbf{X})) = \mathbf{A} d(\text{vec}\mathbf{X}) \quad (3.2.47)$$

式中

$$d(\text{vec}\mathbf{F}(\mathbf{X})) = [d f_{11}(\mathbf{X}), \dots, d f_{p1}(\mathbf{X}), \dots, d f_{1q}(\mathbf{X}), \dots, d f_{pq}(\mathbf{X})]^T \quad (3.2.48)$$

$$d(\text{vec}\mathbf{X}) = [dx_{11}, \dots, dx_{m1}, \dots, dx_{1n}, \dots, dx_{mn}]^T \quad (3.2.49)$$

以及

$$\mathbf{A} = \begin{bmatrix} \frac{\partial f_{11}(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f_{11}(\mathbf{X})}{\partial x_{m1}} & \dots & \frac{\partial f_{11}(\mathbf{X})}{\partial x_{1n}} & \dots & \frac{\partial f_{11}(\mathbf{X})}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{p1}(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f_{p1}(\mathbf{X})}{\partial x_{m1}} & \dots & \frac{\partial f_{p1}(\mathbf{X})}{\partial x_{1n}} & \dots & \frac{\partial f_{p1}(\mathbf{X})}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{1q}(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f_{1q}(\mathbf{X})}{\partial x_{m1}} & \dots & \frac{\partial f_{1q}(\mathbf{X})}{\partial x_{1n}} & \dots & \frac{\partial f_{1q}(\mathbf{X})}{\partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{pq}(\mathbf{X})}{\partial x_{11}} & \dots & \frac{\partial f_{pq}(\mathbf{X})}{\partial x_{m1}} & \dots & \frac{\partial f_{pq}(\mathbf{X})}{\partial x_{1n}} & \dots & \frac{\partial f_{pq}(\mathbf{X})}{\partial x_{mn}} \end{bmatrix} = \frac{\partial \text{vec}\mathbf{F}(\mathbf{X})}{\partial (\text{vec}\mathbf{X})^T} \quad (3.2.50)$$

换言之，矩阵  $\mathbf{A}$  即是矩阵函数  $\mathbf{F}(\mathbf{X})$  的 Jacobian 矩阵  $D_{\mathbf{X}}\mathbf{F}(\mathbf{X})$ 。

对于一个包含有  $\mathbf{X}$  和  $\mathbf{X}^T$  的矩阵函数  $\mathbf{F}(\mathbf{X}) \in \mathbb{R}^{p \times q}$ ，其中  $\mathbf{X} \in \mathbb{R}^{m \times n}$ ，则一阶矩阵微分为

$$d(\text{vec}\mathbf{F}(\mathbf{X})) = \mathbf{A} \text{vec}(d\mathbf{X}) + \mathbf{B} d(\text{vec}\mathbf{X}^T)$$

利用  $d(\text{vec}\mathbf{X}^T) = \mathbf{K}_{mn} \text{vec}(d\mathbf{X})$ ，上式可以改写为

$$d(\text{vec}\mathbf{F}(\mathbf{X})) = (\mathbf{A} + \mathbf{B}\mathbf{K}_{mn}) d(\text{vec}\mathbf{X}) \quad (3.2.51)$$

上述结果可以总结为下面的命题。

**命题 3.2.2** 矩阵函数  $F(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$  的  $pq \times mn$  维 Jacobian 矩阵可以通过下式辨识

$$\begin{aligned} d(\text{vec } F(\mathbf{X})) &= \mathbf{A}d(\text{vec } \mathbf{X}) + \mathbf{B}d(\text{vec } \mathbf{X}^T) \\ \iff D_{\mathbf{X}} F(\mathbf{X}) &= \frac{\partial \text{vec } F(\mathbf{X})}{\partial (\text{vec } \mathbf{X})^T} = \mathbf{A} + \mathbf{B}\mathbf{K}_{mn} \end{aligned} \quad (3.2.52)$$

或  $mn \times pq$  维梯度矩阵可以辨识为

$$\nabla_{\mathbf{X}} F(\mathbf{X}) = (D_{\mathbf{X}} F(\mathbf{X}))^T = \mathbf{A}^T + \mathbf{K}_{nm} \mathbf{B}^T \quad (3.2.53)$$

重要的是, 由于

$$dF(\mathbf{X}) = \mathbf{A}(d\mathbf{X})\mathbf{B} \iff d(\text{vec } F(\mathbf{X})) = (\mathbf{B}^T \otimes \mathbf{A})d(\text{vec } \mathbf{X}) \quad (3.2.54)$$

$$dF(\mathbf{X}) = \mathbf{C}(d\mathbf{X}^T)\mathbf{D} \iff d(\text{vec } F(\mathbf{X})) = (\mathbf{D}^T \otimes \mathbf{C})\mathbf{K}_{mn}d(\text{vec } \mathbf{X}) \quad (3.2.55)$$

所以命题 3.2.2 的辨识形式可以进一步简化成直接对  $F(\mathbf{X})$  微分。

**定理 3.2.1** 矩阵函数  $F(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$  的  $pq \times mn$  维 Jacobian 矩阵可以通过下式辨识

$$\begin{aligned} dF(\mathbf{X}) &= \mathbf{A}(d\mathbf{X})\mathbf{B} + \mathbf{C}(d\mathbf{X}^T)\mathbf{D} \\ \iff D_{\mathbf{X}} F(\mathbf{X}) &= \frac{\partial \text{vec } F(\mathbf{X})}{\partial (\text{vec } \mathbf{X})^T} = (\mathbf{B}^T \otimes \mathbf{A}) + (\mathbf{D}^T \otimes \mathbf{C})\mathbf{K}_{mn} \end{aligned} \quad (3.2.56)$$

或  $mn \times pq$  维梯度矩阵可以辨识为

$$\nabla_{\mathbf{X}} F(\mathbf{X}) = \frac{\partial \text{vec } F(\mathbf{X})}{\partial (\text{vec } \mathbf{X})} = (\mathbf{B} \otimes \mathbf{A}^T) + \mathbf{K}_{nm}(\mathbf{D} \otimes \mathbf{C}^T) \quad (3.2.57)$$

表 3.2.3 总结了实值函数的矩阵微分与 Jacobian 矩阵之间的对应关系。

表 3.2.3 实值函数的矩阵微分与 Jacobian 矩阵的对应关系

函数类型	矩阵微分	Jacobian 矩阵
$f(x) : \mathbb{R} \rightarrow \mathbb{R}$	$df(x) = Adx$	$A \in \mathbb{R}$
$f(\mathbf{x}) : \mathbb{R}^m \rightarrow \mathbb{R}$	$df(\mathbf{x}) = Ad\mathbf{x}$	$A \in \mathbb{R}^{1 \times m}$
$f(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$	$df(\mathbf{X}) = \text{tr}(Ad\mathbf{X})$	$A \in \mathbb{R}^{n \times m}$
$f(\mathbf{x}) : \mathbb{R}^m \rightarrow \mathbb{R}^p$	$df(\mathbf{x}) = Ad\mathbf{x}$	$A \in \mathbb{R}^{p \times m}$
$f(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$	$df(\mathbf{X}) = Ad(\text{vec } \mathbf{X})$	$A \in \mathbb{R}^{p \times mn}$
$F(\mathbf{x}) : \mathbb{R}^m \rightarrow \mathbb{R}^{p \times q}$	$d(\text{vec } F(\mathbf{x})) = Ad\mathbf{x}$	$A \in \mathbb{R}^{pq \times m}$
$F(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$	$dF(\mathbf{X}) = \mathbf{A}(d\mathbf{X})\mathbf{B}$	$(\mathbf{B}^T \otimes \mathbf{A}) \in \mathbb{R}^{pq \times mn}$
$F(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{p \times q}$	$dF(\mathbf{X}) = \mathbf{C}(d\mathbf{X}^T)\mathbf{D}$	$(\mathbf{D}^T \otimes \mathbf{C})\mathbf{K}_{mn} \in \mathbb{R}^{pq \times mn}$

**例 3.2.3** 矩阵函数  $\mathbf{A}\mathbf{X}^T\mathbf{B}$  的矩阵微分为  $d(\mathbf{A}\mathbf{X}^T\mathbf{B}) = \mathbf{A}(d\mathbf{X}^T)\mathbf{B}$ , 于是得矩阵函数  $\mathbf{A}\mathbf{X}^T\mathbf{B}$  的 Jacobian 矩阵

$$\mathbf{D}_{\mathbf{X}}(\mathbf{A}\mathbf{X}^T\mathbf{B}) = (\mathbf{B}^T \otimes \mathbf{A})\mathbf{K}_{mn} \quad (3.2.58)$$

转置后, 即可获得矩阵函数的梯度矩阵。

**例 3.2.4** 矩阵函数  $\mathbf{X}^T\mathbf{B}\mathbf{X}$  的矩阵微分为  $d(\mathbf{X}^T\mathbf{B}\mathbf{X}) = \mathbf{X}^T\mathbf{B}d\mathbf{X} + d(\mathbf{X}^T)\mathbf{B}\mathbf{X}$ , 故矩阵函数  $\mathbf{X}^T\mathbf{B}\mathbf{X}$  的 Jacobian 矩阵为

$$\mathbf{D}_{\mathbf{X}}(\mathbf{X}^T\mathbf{B}\mathbf{X}) = \mathbf{I} \otimes (\mathbf{X}^T\mathbf{B}) + ((\mathbf{B}\mathbf{X})^T \otimes \mathbf{I})\mathbf{K}_{mn} \quad (3.2.59)$$

转置后, 又可得到矩阵函数的梯度矩阵。

表 3.2.4 汇总了一些典型矩阵函数的矩阵微分与 Jacobian 矩阵。

表 3.2.4 矩阵函数的矩阵微分与 Jacobian 矩阵

矩阵函数 $\mathbf{F}(\mathbf{X})$	矩阵微分 $d\mathbf{F}(\mathbf{X})$	Jacobian 矩阵
$\mathbf{X}^T\mathbf{X}$	$\mathbf{X}^T d\mathbf{X} + (d\mathbf{X}^T)\mathbf{X}$	$(\mathbf{I}_n \otimes \mathbf{X}^T) + (\mathbf{X}^T \otimes \mathbf{I}_n)\mathbf{K}_{mn}$
$\mathbf{X}\mathbf{X}^T$	$\mathbf{X}(d\mathbf{X}^T) + (d\mathbf{X})\mathbf{X}^T$	$(\mathbf{I}_m \otimes \mathbf{X})\mathbf{K}_{mn} + (\mathbf{X} \otimes \mathbf{I}_m)$
$\mathbf{A}\mathbf{X}^T\mathbf{B}\mathbf{C}$	$\mathbf{A}(d\mathbf{X}^T)\mathbf{B}\mathbf{C} + \mathbf{A}\mathbf{X}^T\mathbf{B}(d\mathbf{X})\mathbf{C}$	$((\mathbf{B}\mathbf{C})^T \otimes \mathbf{A})\mathbf{K}_{mn} + \mathbf{C}^T \otimes (\mathbf{A}\mathbf{X}^T\mathbf{B})$
$\mathbf{A}\mathbf{X}\mathbf{B}\mathbf{X}^T\mathbf{C}$	$\mathbf{A}(d\mathbf{X})\mathbf{B}\mathbf{X}^T\mathbf{C} + \mathbf{A}\mathbf{X}\mathbf{B}(d\mathbf{X}^T)\mathbf{C}$	$(\mathbf{B}\mathbf{X}^T\mathbf{C})^T \otimes \mathbf{A} + (\mathbf{C}^T \otimes (\mathbf{A}\mathbf{X}\mathbf{B}))\mathbf{K}_{mn}$
$\mathbf{X}^{-1}$	$-\mathbf{X}^{-1}(d\mathbf{X})\mathbf{X}^{-1}$	$-(\mathbf{X}^{-T} \otimes \mathbf{X}^{-1})$
$\mathbf{X}^k$	$\sum_{j=1}^k \mathbf{X}^{j-1}(d\mathbf{X})\mathbf{X}^{k-j}$	$\sum_{j=1}^k (\mathbf{X}^T)^{k-j} \otimes \mathbf{X}^{j-1}$
$\log \mathbf{X}$	$\mathbf{X}^{-1}d\mathbf{X}$	$\mathbf{I} \otimes \mathbf{X}^{-1}$
$\exp(\mathbf{X})$	$\sum_{k=0}^{\infty} \frac{1}{(k+1)!} \sum_{j=0}^k \mathbf{X}^j(d\mathbf{X})\mathbf{X}^{k-j}$	$\sum_{k=0}^{\infty} \frac{1}{(k+1)!} \sum_{j=0}^k (\mathbf{X}^T)^{k-j} \otimes \mathbf{X}^j$

不言而喻, 表 3.2.4 也适用于以向量为变元的矩阵函数  $\mathbf{F}(\mathbf{x}) : \mathbb{R}^{m \times 1} \rightarrow \mathbb{R}^{p \times q}$  和标量函数  $f(\mathbf{X})$  或  $f(\mathbf{x})$  等。例如, 对于矩阵函数  $\mathbf{F}(\mathbf{x}) = \mathbf{x}\mathbf{x}^T \in \mathbb{R}^{m \times m}$  和标量函数  $f(\mathbf{x}) = \mathbf{x}^T\mathbf{x}$ , 由表 3.2.4 直接得

$$\mathbf{D}_{\mathbf{x}}(\mathbf{x}\mathbf{x}^T) = (\mathbf{x} \otimes \mathbf{I}_m)\mathbf{K}_{m1} + (\mathbf{I}_m \otimes \mathbf{x}) = (\mathbf{x} \otimes \mathbf{I}_m) + (\mathbf{I}_m \otimes \mathbf{x})$$

$$\mathbf{D}_{\mathbf{x}}(\mathbf{x}^T\mathbf{x}) = (\mathbf{x}^T \otimes \mathbf{I}_1)\mathbf{K}_{m1} + (\mathbf{I}_1 \otimes \mathbf{x}^T) = 2\mathbf{x}^T$$

因为  $\mathbf{I}_1 = 1$  和  $\mathbf{K}_{m1} = \mathbf{I}_m$ 。

需要注意的是, 一些矩阵函数的矩阵微分可能无法表示成定理 3.2.1 所要求的规范形式, 但一定可以表示成命题 3.2.2 的规范形式。此时, 就必须使用命题 3.2.2 辨识 Jacobian 矩阵。

**例 3.2.5** 两个矩阵  $\mathbf{X} \in \mathbb{R}^{p \times m}$  和  $\mathbf{Y} \in \mathbb{R}^{n \times q}$  的 Kronecker 积  $\mathbf{F}(\mathbf{X}, \mathbf{Y}) = \mathbf{X} \otimes \mathbf{Y}$  的矩阵微分  $d\mathbf{F}(\mathbf{X}, \mathbf{Y}) = (d\mathbf{X}) \otimes \mathbf{Y} + \mathbf{X} \otimes (d\mathbf{Y})$ 。由 Kronecker 积的向量化公式  $\text{vec}(\mathbf{X} \otimes \mathbf{Y}) =$

$(I_m \otimes K_{qp} \otimes I_n)(\text{vec } X \otimes \text{vec } Y)$ , 有

$$\begin{aligned}\text{vec}(\mathbf{d}X \otimes Y) &= (I_m \otimes K_{qp} \otimes I_n)(\mathbf{d} \text{vec } X \otimes \text{vec } Y) \\ &= (I_m \otimes K_{qp} \otimes I_n)(I_{pm} \otimes \text{vec } Y)\mathbf{d} \text{vec } X\end{aligned}\quad (3.2.60)$$

$$\begin{aligned}\text{vec}(X \otimes \mathbf{d}Y) &= (I_m \otimes K_{qp} \otimes I_n)(\text{vec } X \otimes \mathbf{d} \text{vec } Y) \\ &= (I_m \otimes K_{qp} \otimes I_n)(\text{vec } X \otimes I_{nq})\mathbf{d} \text{vec } Y\end{aligned}\quad (3.2.61)$$

因此, Jacobian 矩阵分别为

$$D_X(X \otimes Y) = (I_m \otimes K_{qp} \otimes I_n)(I_{pm} \otimes \text{vec } Y) \quad (3.2.62)$$

$$D_Y(X \otimes Y) = (I_m \otimes K_{qp} \otimes I_n)(\text{vec } X \otimes I_{nq}) \quad (3.2.63)$$

本节的分析与举例充分说明, 一阶矩阵微分的确是辨识实值函数的 Jacobian 矩阵和梯度矩阵的有效数学工具, 它运算简单, 并且易于掌握。

### 3.3 二阶实矩阵微分与 Hessian 矩阵辨识

一阶实矩阵微分可用于辨识实标量函数和实矩阵函数的 Jacobian 矩阵和梯度矩阵。本节将讨论实标量函数和实矩阵函数的二阶偏导和二阶微分。实二阶矩阵微分可以很方便地辨识一个实函数的 Hessian 矩阵。

#### 3.3.1 Hessian 矩阵

实值函数  $f(\mathbf{x})$  相对于  $m \times 1$  实向量  $\mathbf{x}$  的二阶偏导称为 Hessian 矩阵, 记作  $H[f(\mathbf{x})]$ , 定义为

$$H[f(\mathbf{x})] = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} = \frac{\partial}{\partial \mathbf{x}} \left[ \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T} \right] \in \mathbb{R}^{m \times m} \quad (3.3.1)$$

或记作

$$H[f(\mathbf{x})] = \nabla_{\mathbf{x}}^2 f(\mathbf{x}) = \nabla_{\mathbf{x}}(D_{\mathbf{x}} f(\mathbf{x})) \quad (3.3.2)$$

式中  $D_{\mathbf{x}}$  为协梯度算子。于是, Hessian 矩阵的第  $(i, j)$  元素定义为

$$[Hf(\mathbf{x})]_{i,j} = \left[ \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \right]_{i,j} = \frac{\partial}{\partial x_i} \left[ \frac{\partial f(\mathbf{x})}{\partial x_j} \right] \quad (3.3.3)$$

或写作

$$H[f(\mathbf{x})] = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_m \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_m \partial x_m} \end{bmatrix} \in \mathbb{R}^{m \times m} \quad (3.3.4)$$

即实标量函数  $f(\mathbf{x})$  的 Hessian 矩阵是一个  $m \times m$  正方矩阵, 由标量函数  $f(\mathbf{x})$  关于向量变元  $\mathbf{x}$  的元素  $x_i$  的  $m^2$  个二阶偏导组成。

由定义式知, 实标量函数  $f(\mathbf{x})$  的 Hessian 矩阵是一个实对称矩阵

$$(\mathbf{H}[f(\mathbf{x})])^\top = \mathbf{H}[f(\mathbf{x})] \quad (3.3.5)$$

因为二次可导连续函数  $f(\mathbf{x})$  的二次求导与求导顺序无关, 即  $\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}$ 。

仿照实标量函数  $f(\mathbf{x})$  的 Hessian 矩阵的定义公式, 实标量函数  $f(\mathbf{X})$  的 Hessian 矩阵定义为

$$\mathbf{H}[f(\mathbf{X})] = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec } \mathbf{X} \partial (\text{vec } \mathbf{X})^\top} = \nabla_{\mathbf{X}} (\mathbf{D}_{\mathbf{X}} f(\mathbf{X})) \in \mathbb{R}^{mn \times mn} \quad (3.3.6)$$

其元素表示形式为

$$\mathbf{H}[f(\mathbf{X})] = \begin{bmatrix} \frac{\partial^2 f}{\partial x_{11} \partial x_{11}} & \dots & \frac{\partial^2 f}{\partial x_{11} \partial x_{m1}} & \dots & \frac{\partial^2 f}{\partial x_{11} \partial x_{1n}} & \dots & \frac{\partial^2 f}{\partial x_{11} \partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 f}{\partial x_{m1} \partial x_{11}} & \dots & \frac{\partial^2 f}{\partial x_{m1} \partial x_{m1}} & \dots & \frac{\partial^2 f}{\partial x_{m1} \partial x_{1n}} & \dots & \frac{\partial^2 f}{\partial x_{m1} \partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 f}{\partial x_{1n} \partial x_{11}} & \dots & \frac{\partial^2 f}{\partial x_{1n} \partial x_{m1}} & \dots & \frac{\partial^2 f}{\partial x_{1n} \partial x_{1n}} & \dots & \frac{\partial^2 f}{\partial x_{1n} \partial x_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 f}{\partial x_{mn} \partial x_{11}} & \dots & \frac{\partial^2 f}{\partial x_{mn} \partial x_{m1}} & \dots & \frac{\partial^2 f}{\partial x_{mn} \partial x_{1n}} & \dots & \frac{\partial^2 f}{\partial x_{mn} \partial x_{mn}} \end{bmatrix} \quad (3.3.7)$$

由  $\frac{\partial^2 f}{\partial x_{ij} \partial x_{kl}} = \frac{\partial^2 f}{\partial x_{kl} \partial x_{ij}}$  立即知, 实标量函数  $f(\mathbf{X})$  的 Hessian 矩阵是一个实对称矩阵

$$[\mathbf{H}f(\mathbf{X})]^\top = \mathbf{H}[f(\mathbf{X})] \quad (3.3.8)$$

### 3.3.2 Hessian 矩阵的辨识原理

下面讨论实标量函数和实矩阵函数的 Hessian 矩阵的辨识。Hessian 矩阵在最优化的全局最优点的判别和 Newton 算法中起着关键的作用。

我们分两种情况讨论 Hessian 矩阵的辨识。

#### 1. 标量函数 $f(\mathbf{x})$ 的 Hessian 矩阵辨识

很多情况下, 直接根据定义求标量函数  $f(\mathbf{x})$  或  $f(\mathbf{X})$  的 Hessian 矩阵可能比较麻烦。更简单的方法是利用实函数  $f(\mathbf{x})$  或  $f(\mathbf{X})$  的二阶实微分矩阵与 Hessian 矩阵之间的对应关系。

注意到微分  $d\mathbf{x}$  不是向量  $\mathbf{x}$  的函数, 故有

$$d^2\mathbf{x} = d(d\mathbf{x}) = 0 \quad (3.3.9)$$

记住这一点, 由式 (3.2.16) 易求得二阶微分  $d^2f(\mathbf{x}) = d(df(\mathbf{x}))$  为

$$d^2f(\mathbf{x}) = (d\mathbf{x})^\top \frac{\partial df(\mathbf{x})}{\partial \mathbf{x}} = (d\mathbf{x})^\top \frac{\partial}{\partial \mathbf{x}} \left( \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^\top} \right) d\mathbf{x} = (d\mathbf{x})^\top \frac{\partial f^2(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^\top} d\mathbf{x}$$

或写作简洁形式

$$d^2 f(\mathbf{x}) = (d\mathbf{x})^T \mathbf{H}[f(\mathbf{x})] d\mathbf{x} \quad (3.3.10)$$

称为实标量函数  $f(\mathbf{x})$  的二阶微分法则的向量形式。式中

$$\mathbf{H}[f(\mathbf{x})] = \frac{\partial f^2(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \quad (3.3.11)$$

是函数  $f(\mathbf{x})$  的 Hessian 矩阵，其第  $(i, j)$  元素

$$h_{ij} = \frac{\partial}{\partial x_i} \left( \frac{\partial f(\mathbf{x})}{\partial x_j} \right) = \frac{\partial f^2(\mathbf{x})}{\partial x_i \partial x_j} \quad (3.3.12)$$

注意到实标量函数  $f(\mathbf{x})$  的一阶微分  $df(\mathbf{x}) = \mathbf{A}d\mathbf{x}$  中的矩阵  $\mathbf{A} \in \mathbb{R}^{1 \times m}$  通常是变元  $\mathbf{x}$  的实值行向量函数，其微分仍然是实值行向量函数，故有

$$d\mathbf{A} = (d\mathbf{x})^T \mathbf{B} \in \mathbb{R}^{1 \times m}$$

其中  $\mathbf{B} \in \mathbb{R}^{m \times m}$ 。于是，实标量函数  $f(\mathbf{x})$  的二阶微分取二次型函数形式

$$d^2 f(\mathbf{x}) = d(\mathbf{A}d\mathbf{x}) = (d\mathbf{x})^T \mathbf{B} d\mathbf{x} \quad (3.3.13)$$

比较式 (3.3.13) 和式 (3.3.10) 知，实标量函数  $f(\mathbf{x})$  的 Hessian 矩阵  $\mathbf{H}_x f(\mathbf{x}) = \mathbf{B}$ 。为了确保 Hessian 矩阵为实对称矩阵，故取

$$\mathbf{H}[f(\mathbf{x})] = \frac{1}{2} (\mathbf{B}^T + \mathbf{B}) \quad (3.3.14)$$

## 2. 标量函数 $f(\mathbf{X})$ 的 Hessian 矩阵辨识

由式 (3.2.21) 得标量函数  $f(\mathbf{X})$  的二阶微分

$$\begin{aligned} d^2 f(\mathbf{X}) &= (d\text{vec}\mathbf{X})^T \frac{\partial df(\mathbf{X})}{\partial \text{vec}\mathbf{X}} \\ &= (d\text{vec}\mathbf{X})^T \frac{\partial}{\partial \text{vec}\mathbf{X}} \left( \frac{\partial f(\mathbf{X})}{\partial (\text{vec}\mathbf{X})^T} \right) d(\text{vec}\mathbf{X}) \\ &= (d\text{vec}\mathbf{X})^T \frac{\partial f^2(\mathbf{X})}{\partial \text{vec}\mathbf{X} \partial (\text{vec}\mathbf{X})^T} d(\text{vec}\mathbf{X}) \end{aligned}$$

即有

$$d^2 f(\mathbf{X}) = (d(\text{vec}\mathbf{X}))^T \mathbf{H}[f(\mathbf{X})] d(\text{vec}\mathbf{X}) \quad (3.3.15)$$

这一公式称为实标量函数  $f(\mathbf{X})$  的二阶 (矩阵) 微分法则。式中

$$\mathbf{H}[f(\mathbf{X})] = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}\mathbf{X} \partial (\text{vec}\mathbf{X})^T} \quad (3.3.16)$$

是标量函数  $f(\mathbf{X})$  的 Hessian 矩阵。

对于实标量函数  $f(\mathbf{X})$ , 其一阶微分  $\mathrm{d}f(\mathbf{X}) = \mathbf{A}\mathrm{d}(\mathrm{vec}\mathbf{X})$  中的矩阵  $\mathbf{A}$  通常是变元矩阵  $\mathbf{X}$  的实值行向量函数, 其微分仍然是实值行向量函数, 故有

$$\mathrm{d}\mathbf{A} = (\mathrm{d}(\mathrm{vec}\mathbf{X}))^T \mathbf{B} \in \mathbb{R}^{1 \times mn}$$

其中  $\mathbf{B} \in \mathbb{R}^{mn \times mn}$ 。于是, 实标量函数  $f(\mathbf{X})$  的二阶微分取二次型函数形式

$$\mathrm{d}^2 f(\mathbf{X}) = (\mathrm{d}(\mathrm{vec}\mathbf{X}))^T \mathbf{B} \mathrm{d}(\mathrm{vec}\mathbf{X}) \quad (3.3.17)$$

比较二阶矩阵微分的两个表达式 (3.3.17) 和式 (3.3.15) 知, 实值标量函数  $f(\mathbf{X})$  的 Hessian 矩阵

$$\mathbf{H}[f(\mathbf{X})] = \frac{1}{2}(\mathbf{B}^T + \mathbf{B}) \quad (3.3.18)$$

因为实 Hessian 矩阵必须是实对称矩阵。

以上结果可以归结为下面的命题。

**命题 3.3.1** 以向量  $\mathbf{x}$  或者矩阵  $\mathbf{X}$  为变元的标量函数的二阶微分与 Hessian 矩阵之间存在下面的二阶辨识关系

$$\mathrm{d}^2 f(\mathbf{x}) = (\mathrm{d}\mathbf{x})^T \mathbf{B} \mathrm{d}\mathbf{x} \iff \mathbf{H}[f(\mathbf{x})] = \frac{1}{2}(\mathbf{B}^T + \mathbf{B}) \quad (3.3.19)$$

$$\mathrm{d}^2 f(\mathbf{X}) = (\mathrm{d}(\mathrm{vec}\mathbf{X}))^T \mathbf{B} \mathrm{d}(\mathrm{vec}\mathbf{X}) \iff \mathbf{H}[f(\mathbf{X})] = \frac{1}{2}(\mathbf{B}^T + \mathbf{B}) \quad (3.3.20)$$

令  $\mathbf{x}, \mathbf{x}, \mathbf{X}$  分别代表函数的实标量变元、 $m \times 1$  实向量变元和  $m \times n$  实矩阵变元, 而  $f(\cdot), \mathbf{f}(\cdot), \mathbf{F}(\cdot)$  则分别表示实标量函数、 $p \times 1$  实向量函数和  $p \times q$  实矩阵函数。

表 3.3.1 的二阶辨识表 (second identification table) 描述了不同实函数的二阶实微分矩阵与实 Hessian 矩阵之间的基本对应关系。

表 3.3.1 二阶辨识表 [328, p.190]

实函数	二阶实微分矩阵	实 Hessian 矩阵 $\mathbf{H}$	$\mathbf{H}$ 的维数
$f(\mathbf{x})$	$\mathrm{d}^2[f(\mathbf{x})] = \beta(\mathrm{d}\mathbf{x})^2$	$\mathbf{H}[f(\mathbf{x})] = \beta$	$1 \times 1$
$\mathbf{f}(\mathbf{x})$	$\mathrm{d}^2[\mathbf{f}(\mathbf{x})] = (\mathrm{d}\mathbf{x})^T \mathbf{B} \mathrm{d}\mathbf{x}$	$\mathbf{H}[\mathbf{f}(\mathbf{x})] = \frac{1}{2}(\mathbf{B} + \mathbf{B}^T)$	$m \times m$
$\mathbf{f}(\mathbf{X})$	$\mathrm{d}^2[\mathbf{f}(\mathbf{X})] = \mathrm{d}(\mathrm{vec}(\mathbf{X}))^T \mathbf{B} \mathrm{d}(\mathrm{vec}(\mathbf{X}))$	$\mathbf{H}[\mathbf{f}(\mathbf{X})] = \frac{1}{2}(\mathbf{B} + \mathbf{B}^T)$	$mn \times mn$
$\mathbf{f}(\mathbf{x})$	$\mathrm{d}^2[\mathbf{f}(\mathbf{x})] = \mathbf{b}(\mathrm{d}\mathbf{x})^2$	$\mathbf{H}[\mathbf{f}(\mathbf{x})] = \mathbf{b}$	$p \times 1$
$\mathbf{f}(\mathbf{x})$	$\mathrm{d}^2[\mathbf{f}(\mathbf{x})] = (\mathbf{I}_m \otimes \mathrm{d}\mathbf{x})^T \mathbf{B} \mathrm{d}\mathbf{x}$	$\mathbf{H}[\mathbf{f}(\mathbf{x})] = \frac{1}{2}[\mathbf{B} + (\mathbf{B}')_v]$	$pm \times m$
$\mathbf{f}(\mathbf{X})$	$\mathrm{d}^2[\mathbf{f}(\mathbf{X})] = (\mathbf{I}_m \otimes \mathrm{d} \mathrm{vec}(\mathbf{X}))^T \mathbf{B} \mathrm{d}(\mathrm{vec}(\mathbf{X}))$	$\mathbf{H}[\mathbf{f}(\mathbf{X})] = \frac{1}{2}[\mathbf{B} + (\mathbf{B}')_v]$	$pmn \times mn$
$\mathbf{F}(\mathbf{x})$	$\mathrm{d}^2[\mathbf{F}(\mathbf{x})] = \mathbf{B}(\mathrm{d}\mathbf{x})^2$	$\mathbf{H}[\mathbf{F}(\mathbf{x})] = \mathrm{vec}(\mathbf{B})$	$pq \times 1$
$\mathbf{F}(\mathbf{x})$	$\mathrm{d}^2[\mathrm{vec}(\mathbf{F})] = (\mathbf{I}_{mp} \otimes \mathrm{d}\mathbf{x})^T \mathbf{B} \mathrm{d}\mathbf{x}$	$\mathbf{H}[\mathbf{F}(\mathbf{x})] = \frac{1}{2}[\mathbf{B} + (\mathbf{B}')_v]$	$pmq \times m$
$\mathbf{F}(\mathbf{X})$	$\mathrm{d}^2[\mathrm{vec}(\mathbf{F})] = (\mathbf{I}_{mp} \otimes \mathrm{d} \mathrm{vec}(\mathbf{X}))^T \mathbf{B} \mathrm{d}(\mathrm{vec}(\mathbf{X}))$	$\mathbf{H}[\mathbf{F}(\mathbf{x})] = \frac{1}{2}[\mathbf{B} + (\mathbf{B}')_v]$	$pmqn \times mn$

在实向量函数  $f \in \mathbb{R}^p$  的情况下, 表 3.3.1 中的  $pmn \times mn$  矩阵  $B$  和  $(B')_v$ , 分别为

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_p \end{bmatrix}, \quad (B')_v = \begin{bmatrix} B_1^T \\ B_2^T \\ \vdots \\ B_p^T \end{bmatrix} \quad (3.3.21)$$

而在实矩阵函数  $F \in \mathbb{R}^{p \times q}$  的情况下,  $pmqn \times mn$  矩阵  $B$  和  $(B')_v$ , 分别为

$$B = \begin{bmatrix} B_{11} \\ \vdots \\ B_{p1} \\ \vdots \\ B_{1q} \\ \vdots \\ B_{pq} \end{bmatrix}, \quad (B')_v = \begin{bmatrix} B_{11}^T \\ \vdots \\ B_{p1}^T \\ \vdots \\ B_{1q}^T \\ \vdots \\ B_{pq}^T \end{bmatrix} \quad (3.3.22)$$

所有分块矩阵  $B_1, \dots, B_p$  以及  $B_{11}, \dots, B_{pq}$  都是  $m \times m$  矩阵 (当  $f$  和  $F$  分别是  $m \times 1$  向量  $x$  的向量函数和矩阵函数时), 或者都是  $mn \times mn$  矩阵 (当  $f$  和  $F$  分别是  $m \times n$  矩阵  $X$  的向量函数和矩阵函数时)。

### 3.3.3 Hessian 矩阵的辨识方法

命题 3.3.1 表明, 为了辨识 Hessian 矩阵, 需要将实标量函数  $f(X)$  的二阶微分写成关于变元矩阵的向量化  $\text{vec}(X)$  的二次型规范形式  $d^2 f(X) = (\text{d}(\text{vec}X))^T B \text{d}(\text{vec}X)$ 。这种规范形式需要对二阶矩阵微分的结果进行向量化运算, 有些麻烦。能否避免向量化的运算, 而直接由二阶矩阵微分辨识 Hessian 矩阵呢? 下面就来讨论这个问题。

如前所述, 标量函数的一阶微分可写成迹函数的规范形式  $df(X) = \text{tr}(A \text{d}X)$ , 其中  $A = A(X) \in \mathbb{R}^{n \times m}$  一般是变元矩阵  $X \in \mathbb{R}^{m \times n}$  的矩阵函数。不失一般性, 假定矩阵函数  $A = A(X)$  的微分矩阵为

$$\text{d}A = B(\text{d}X)C, \quad B, C \in \mathbb{R}^{n \times m} \quad (3.3.23)$$

或者

$$\text{d}A = U(\text{d}X)^T V, \quad U \in \mathbb{R}^{n \times n}, V \in \mathbb{R}^{m \times m} \quad (3.3.24)$$

注意, 这两种微分矩阵分别包括了  $(\text{d}X)C$ ,  $B \text{d}X$  和  $(\text{d}X)^T V$ ,  $U(\text{d}X)^T$  等特例在内。

将式 (3.3.23) 和式 (3.3.24) 分别代入  $df(X) = \text{tr}(A \text{d}X)$  的微分, 知实值标量函数  $f(X)$  的二阶微分  $d^2 f(X) = \text{tr}(\text{d}A \text{d}X)$  取形式

$$d^2 f(X) = \text{tr}(B(\text{d}X)C \text{d}X) \quad (3.3.25)$$

或

$$d^2 f(X) = \text{tr}(V(\text{d}X)U(\text{d}X)^T) \quad (3.3.26)$$

利用迹函数的性质  $\text{tr}(\mathbf{ABCD}) = (\text{vec}\mathbf{D}^T)^T(\mathbf{C}^T \otimes \mathbf{A})\text{vec}\mathbf{B}$ , 易知

$$\begin{aligned}\text{tr}(\mathbf{B}(\text{d}\mathbf{X})\mathbf{C}\text{d}\mathbf{X}) &= (\text{vec}(\text{d}\mathbf{X})^T)^T(\mathbf{C}^T \otimes \mathbf{B})\text{vec}(\text{d}\mathbf{X}) \\ &= (\text{d}(\mathbf{K}_{mn}\text{vec}\mathbf{X}))^T(\mathbf{C}^T \otimes \mathbf{B})\text{d}(\text{vec}\mathbf{X}) \\ &= (\text{d}(\text{vec}\mathbf{X}))^T\mathbf{K}_{nm}(\mathbf{C}^T \otimes \mathbf{B})\text{d}(\text{vec}\mathbf{X})\end{aligned}\quad (3.3.27)$$

$$\begin{aligned}\text{tr}(\mathbf{V}\text{d}\mathbf{X}\mathbf{U}(\text{d}\mathbf{X})^T) &= (\text{vec}(\text{d}\mathbf{X}))^T(\mathbf{U}^T \otimes \mathbf{V})\text{vec}(\text{d}\mathbf{X}) \\ &= (\text{d}\text{vec}\mathbf{X})^T(\mathbf{U}^T \otimes \mathbf{V})\text{d}\text{vec}\mathbf{X}\end{aligned}\quad (3.3.28)$$

式中使用了关系式  $\mathbf{K}_{mn}\text{vec}(\mathbf{A}_{m \times n}) = \text{vec}(\mathbf{A}_{m \times n}^T)$  和  $\mathbf{K}_{mn}^T = \mathbf{K}_{nm}$ 。

表达式  $\text{d}^2f(\mathbf{X}) = \text{tr}(\mathbf{B}(\text{d}\mathbf{X})\mathbf{C}\text{d}\mathbf{X})$  和  $\text{d}^2f(\mathbf{X}) = \text{tr}(\mathbf{V}(\text{d}\mathbf{X})\mathbf{U}(\text{d}\mathbf{X})^T)$  称为实标量函数的二阶矩阵微分的两种规范形式。

由于二阶矩阵微分的规范形式  $\text{tr}(\mathbf{V}(\text{d}\mathbf{X})\mathbf{U}(\text{d}\mathbf{X})^T)$  或  $\text{tr}(\mathbf{B}(\text{d}\mathbf{X})\mathbf{C}\text{d}\mathbf{X})$  可以分别等价表示成 Hessian 矩阵的辨识命题 3.3.1 所要求的二次型规范形式, 所以命题 3.3.1 可以等价叙述为 Hessian 矩阵的下述辨识定理。

**定理 3.3.1** [328,p.192] 令  $f(\mathbf{X})$  是  $m \times n$  实矩阵  $\mathbf{X}$  的实值函数, 并可二次微分, 则实函数  $f(\mathbf{X})$  在  $\mathbf{X}$  的二阶实微分矩阵与 Hessian 矩阵之间存在下面的对应关系

$$\text{d}^2f(\mathbf{X}) = \text{tr}(\mathbf{V}(\text{d}\mathbf{X})\mathbf{U}(\text{d}\mathbf{X})^T) \iff \mathbf{H}[f(\mathbf{X})] = \frac{1}{2}(\mathbf{U}^T \otimes \mathbf{V} + \mathbf{U} \otimes \mathbf{V}^T) \quad (3.3.29)$$

或者

$$\text{d}^2f(\mathbf{X}) = \text{tr}(\mathbf{B}(\text{d}\mathbf{X})\mathbf{C}\text{d}\mathbf{X}) \iff \mathbf{H}[f(\mathbf{X})] = \frac{1}{2}\mathbf{K}_{nm}(\mathbf{C}^T \otimes \mathbf{B} + \mathbf{B}^T \otimes \mathbf{C}) \quad (3.3.30)$$

式中,  $\mathbf{K}_{nm}$  为交换矩阵。

定理 3.3.1 表明, Hessian 矩阵辨识的基本问题就是如何将给定的实标量函数的二阶矩阵微分表示成两种规范形式之一。

下面举几个例子说明如何应用定理 3.3.1 求实值函数的 Hessian 矩阵。

**例 3.3.1** 考虑实值函数  $f(\mathbf{X}) = \text{tr}(\mathbf{X}^{-1})$ , 其中  $\mathbf{X}$  是一个  $n \times n$  矩阵。由于

$$\text{d}f(\mathbf{X}) = -\text{tr}(\mathbf{X}^{-1}(\text{d}\mathbf{X})\mathbf{X}^{-1})$$

求上述一阶微分的微分, 并利用  $\text{d}(\text{tr}\mathbf{U}) = \text{tr}(\text{d}\mathbf{U})$ , 得二阶微分矩阵

$$\begin{aligned}\text{d}^2f(\mathbf{X}) &= -\text{tr}((\text{d}\mathbf{X}^{-1})(\text{d}\mathbf{X})\mathbf{X}^{-1}) - \text{tr}(\mathbf{X}^{-1}(\text{d}\mathbf{X})(\text{d}\mathbf{X}^{-1})) \\ &= 2\text{tr}(\mathbf{X}^{-1}(\text{d}\mathbf{X})\mathbf{X}^{-1}(\text{d}\mathbf{X})\mathbf{X}^{-1}) \\ &= 2\text{tr}(\mathbf{X}^{-2}(\text{d}\mathbf{X})\mathbf{X}^{-1}\text{d}\mathbf{X})\end{aligned}$$

利用定理 3.3.1, 即可得到 Hessian 矩阵

$$\mathbf{H}[f(\mathbf{X})] = \frac{\partial^2 \text{tr}(\mathbf{X}^{-1})}{\partial \text{vec}\mathbf{X} \partial (\text{vec}\mathbf{X})^T} = \mathbf{K}_{nn}[\mathbf{X}^{-T} \otimes \mathbf{X}^{-2} + (\mathbf{X}^{-2})^T \otimes \mathbf{X}^{-1}]$$

**例 3.3.2** 对于二次型函数  $f(\mathbf{X}) = \text{tr}(\mathbf{X}^T \mathbf{A} \mathbf{X})$ , 其一阶微分为

$$df(\mathbf{X}) = \text{tr}(\mathbf{X}^T (\mathbf{A} + \mathbf{A}^T) d\mathbf{X})$$

再次求微分, 得二阶微分

$$d^2 f(\mathbf{X}) = \text{tr}((\mathbf{A} + \mathbf{A}^T)(d\mathbf{X})(d\mathbf{X})^T)$$

由定理 3.3.1 知 Hessian 矩阵为

$$\mathbf{H}[f(\mathbf{X})] = \frac{\partial^2 \text{tr}(\mathbf{X}^T \mathbf{A} \mathbf{X})}{\partial \text{vec} \mathbf{X} \partial (\text{vec} \mathbf{X})^T} = \mathbf{I} \otimes (\mathbf{A} + \mathbf{A}^T)$$

**例 3.3.3** 函数  $\log |\mathbf{X}_{n \times n}|$  的一阶微分为  $d \log |\mathbf{X}| = \text{tr}(\mathbf{X}^{-1} d\mathbf{X})$ 。由此得二阶微分  $-\text{tr}(\mathbf{X}^{-1}(d\mathbf{X})\mathbf{X}^{-1} d\mathbf{X})$ 。由定理 3.3.1 得 Hessian 矩阵

$$\mathbf{H}[f(\mathbf{X})] = \frac{\partial^2 \log |\mathbf{X}|}{\partial \text{vec} \mathbf{X} \partial (\text{vec} \mathbf{X})^T} = -\mathbf{K}_{nn}(\mathbf{X}^{-T} \otimes \mathbf{X}^{-1})$$

### 3.4 共轭梯度与复 Hessian 矩阵

在阵列信号处理和移动通信中, 当处理窄带信号时, 通常都采用等效复基带表示, 将发射和接收信号以及系统参数表示成复值向量。在这些应用中, 最优化问题的目标函数是复向量或者复矩阵的二次型或其他形式的实值函数, 优化问题的求解必须计算目标函数相对于复向量或者复矩阵的梯度。很显然, 这类梯度会有以下两种形式:

- (1) 梯度 目标函数相对于复向量或者复矩阵本身的梯度;
- (2) 共轭梯度 目标函数相对于复共轭向量或者复共轭矩阵的梯度。

#### 3.4.1 全纯函数与复变函数的偏导

在讨论目标函数相对于复变元向量或者复变元矩阵的梯度和共轭梯度之前, 有必要先复习一下复变函数的有关知识。

为了方便叙述, 首先对变元和函数作统一的符号规定:

$\mathbf{z} = [z_1, \dots, z_m]^T \in \mathbb{C}^m$  为复向量变元, 其复共轭为  $\mathbf{z}^*$ ;

$\mathbf{Z} = [z_1, \dots, z_n] \in \mathbb{C}^{m \times n}$  为复矩阵变元, 其复共轭为  $\mathbf{Z}^*$ ;

$f(\mathbf{z}) \in \mathbb{C}$  为复标量函数, 变元为  $m \times 1$  复向量  $\mathbf{z}$  及  $\mathbf{z}^*$ , 记作  $f : \mathbb{C}^m \rightarrow \mathbb{C}$ ;

$f(\mathbf{Z}) \in \mathbb{C}$  为复标量函数, 变元为  $m \times n$  复矩阵  $\mathbf{Z}$  及  $\mathbf{Z}^*$ , 记作  $f : \mathbb{C}^{m \times n} \rightarrow \mathbb{C}$ ;

$\mathbf{f}(\mathbf{z}) \in \mathbb{C}^p$  为  $p \times 1$  复向量函数, 变元为  $m \times 1$  复向量  $\mathbf{z}$  及  $\mathbf{z}^*$ , 记作  $\mathbf{f} : \mathbb{C}^m \rightarrow \mathbb{C}^p$ ;

$\mathbf{f}(\mathbf{Z}) \in \mathbb{C}^p$  为  $p \times 1$  复向量函数, 变元为  $m \times n$  复矩阵  $\mathbf{Z}$  及  $\mathbf{Z}^*$ , 记作  $\mathbf{f} : \mathbb{C}^{m \times n} \rightarrow \mathbb{C}^p$ ;

$\mathbf{F}(\mathbf{z}) \in \mathbb{C}^{p \times q}$  为  $p \times q$  复矩阵函数, 变元为  $m \times 1$  复向量  $\mathbf{z}$  及  $\mathbf{z}^*$ , 记作  $\mathbf{F} : \mathbb{C}^m \rightarrow \mathbb{C}^{p \times q}$ ;

$F(Z) \in \mathbb{C}^{p \times q}$  为  $p \times q$  复矩阵函数, 变元为  $m \times n$  复矩阵  $Z$  及  $Z^*$ , 记作  $F : \mathbb{C}^{m \times n} \rightarrow \mathbb{C}^{p \times q}$ 。

表 3.4.1 汇总了以上复值函数的分类。

表 3.4.1 复值函数的分类

函数类型	标量变元 $z, z^* \in \mathbb{C}$	向量变元 $z, z^* \in \mathbb{C}^m$	矩阵变元 $Z, Z^* \in \mathbb{C}^{m \times n}$
标量函数 $f \in \mathbb{C}$	$f(z, z^*)$ $f : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{C}$	$f(z, z^*)$ $f : \mathbb{C}^m \times \mathbb{C}^m \rightarrow \mathbb{C}$	$f(Z, Z^*)$ $f : \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \rightarrow \mathbb{C}$
向量函数 $f \in \mathbb{C}^p$	$f(z, z^*)$ $f : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{C}^p$	$f(z, z^*)$ $f : \mathbb{C}^m \times \mathbb{C}^m \rightarrow \mathbb{C}^p$	$f(Z, Z^*)$ $f : \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \rightarrow \mathbb{C}^p$
矩阵函数 $F \in \mathbb{C}^{p \times q}$	$F(z, z^*)$ $F : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{C}^{p \times q}$	$F(z, z^*)$ $F : \mathbb{C}^m \times \mathbb{C}^m \rightarrow \mathbb{C}^{p \times q}$	$F(Z, Z^*)$ $F : \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \rightarrow \mathbb{C}^{p \times q}$

**定义 3.4.1**<sup>[284]</sup> 令  $D \subseteq \mathbb{C}$  是函数  $f : D \rightarrow \mathbb{C}$  的定义域。以复数  $z$  为变元的函数  $f(z)$  是在  $D$  域的复解析函数, 若  $f(z)$  是复数可微分的, 即  $\lim_{\Delta z \rightarrow 0} \frac{f(z + \Delta z) - f(z)}{\Delta z}$  对所有  $z \in D$  存在。

术语“(复) 解析”在现代数学中常用完全同义的术语“全纯”(holomorphic) 代替。因此, 复解析函数常称为全纯函数(holomorphic function)。全纯函数和实解析函数的区别是: 一个函数在实变量  $x$  和  $y$  域内都是(实) 解析的, 但在复变量  $z = x + jy$  域内不一定是全纯的, 即可能是非复解析的。

令复变函数  $f(z)$  可以用实部  $u(x, y)$  和虚部  $v(x, y)$  写作

$$f(z) = u(x, y) + jv(x, y)$$

式中  $z = x + jy$ , 并且  $u(x, y)$  和  $v(x, y)$  分别是实值函数。

关于全纯函数, 以下四种叙述等价<sup>[167]</sup>:

1. 复变函数  $f(z)$  是全纯函数(即复解析函数);
2. 复变函数的导数  $f'(z)$  存在, 并且连续;
3. 复变函数  $f(z)$  满足 Cauchy-Riemann 条件

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} \quad \text{和} \quad \frac{\partial v}{\partial x} = -\frac{\partial u}{\partial y} \quad (3.4.1)$$

4. 复变函数  $f(z)$  的所有导数存在, 并且具有一个收敛的幂级数。

Cauchy-Riemann 条件也称为 Cauchy-Riemann 方程, 它的一个直接结果是: 函数  $f(z) = u(x, y) + jv(x, y)$  为全纯函数, 仅当实变函数  $u(x, y)$  和  $v(x, y)$  同时满足 Laplace 方程

$$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0 \quad \text{和} \quad \frac{\partial^2 v(x, y)}{\partial x^2} + \frac{\partial^2 v(x, y)}{\partial y^2} = 0 \quad (3.4.2)$$

满足 Laplace 方程

$$\frac{\partial^2 g(x, y)}{\partial x^2} + \frac{\partial^2 g(x, y)}{\partial y^2} = 0 \quad (3.4.3)$$

的实变函数  $g(x, y)$  称为调和函数 (harmonic function)。

一个复变函数  $f(z) = u(x, y) + jv(x, y)$  只要其中任何一个实变函数  $u(x, y)$  或者  $v(x, y)$  不满足 Cauchy-Riemann 条件或者 Laplace 条件, 那么它就不是一个全纯函数。

虽然幂函数  $z^n$ 、指数函数  $e^z$ 、对数函数  $\ln z$ 、正弦函数  $\sin z$  和余弦函数  $\cos z$  等许多函数都是全纯函数, 即全复平面上的解析函数。但是, 实际经常遇到的一些常用函数却不是全纯函数:

(1) 复变函数  $f(z) = z^* = x - jy = u(x, y) + jv(x, y)$  中的实变函数  $u(x, y) = x$  和  $v(x, y) = -y$  显然不满足 Cauchy-Riemann 条件  $\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}$ 。

(2) 任何一个非常数的实值复变函数  $f(z) \in \mathbb{R}$  都不满足 Cauchy-Riemann 条件  $\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}$  和  $\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}$ , 因为  $f(z) = u(x, y) + jv(x, y)$  中的实变函数  $v(x, y) = 0$ 。特别地, 实值函数  $f(z) = |z| = \sqrt{x^2 + y^2}$  是不可微分的, 而  $f(z) = |z|^2 = x^2 + y^2 = u(x, y) + jv(x, y)$  中的实变函数  $u(x, y) = x^2 + y^2$  不是调和函数, 因为它不满足 Laplace 条件  $\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0$ 。

(3) 复变函数  $f(z) = \operatorname{Re}(z) = x$  和  $f(z) = \operatorname{Im}(z) = y$  都不满足 Cauchy-Riemann 条件。

既然很多常用的复变函数用  $f(z)$  表示时不是全纯函数, 自然会产生一个问题便是: 是否采用其他表示形式, 能够保证任何一个复变函数是全纯 (即复解析) 函数, 从而求出它关于  $z$  或者  $z^*$  的偏导呢? 为了回答这个问题, 有必要复习复变函数论中关于复数  $z$  和共轭复数  $z^*$  的导数的定义。

形式偏导 (formal partial derivatives) 定义为

$$\frac{\partial}{\partial z} = \frac{1}{2} \left( \frac{\partial}{\partial x} - j \frac{\partial}{\partial y} \right) \quad (3.4.4)$$

$$\frac{\partial}{\partial z^*} = \frac{1}{2} \left( \frac{\partial}{\partial x} + j \frac{\partial}{\partial y} \right) \quad (3.4.5)$$

上述形式偏导是 Wirtinger 于 1927 年提出的<sup>[515]</sup>, 有时也叫 Wirtinger 偏导。

关于复变量  $z = x + jy$  的偏导, 有一个实部和虚部的独立性基本假设

$$\frac{\partial x}{\partial y} = 0 \quad \text{和} \quad \frac{\partial y}{\partial x} = 0 \quad (3.4.6)$$

由形式偏导的定义及上述独立性假设, 容易求出

$$\frac{\partial z}{\partial z^*} = \frac{\partial x}{\partial z^*} + j \frac{\partial y}{\partial z^*} = \frac{1}{2} \left( \frac{\partial x}{\partial x} + j \frac{\partial x}{\partial y} \right) + j \frac{1}{2} \left( \frac{\partial y}{\partial x} + j \frac{\partial y}{\partial y} \right) = \frac{1}{2}(1 + 0) + j \frac{1}{2}(0 + j)$$

$$\frac{\partial z^*}{\partial z} = \frac{\partial x}{\partial z} - j \frac{\partial y}{\partial z} = \frac{1}{2} \left( \frac{\partial x}{\partial x} - j \frac{\partial x}{\partial y} \right) - j \frac{1}{2} \left( \frac{\partial y}{\partial x} - j \frac{\partial y}{\partial y} \right) = \frac{1}{2}(1 - 0) - j \frac{1}{2}(0 - j)$$

即有

$$\frac{\partial z}{\partial z^*} = 0 \quad \text{和} \quad \frac{\partial z^*}{\partial z} = 0 \quad (3.4.7)$$

式 (3.4.7) 揭示了复变函数理论的一个基本结果：复变量  $z$  和复共轭变量  $z^*$  是两个独立的变量。

在标准的复变函数框架内，一个复变函数  $f(z)$  (其中  $z = x + jy$ ) 使用实(极)坐标  $r \stackrel{\text{def}}{=} (x, y)^T$  表示为  $f(r) = f(x, y)$ 。然而，在复导数的框架内，基于复变量的实部与虚部相互独立的基本假设，则使用共轭坐标  $c \stackrel{\text{def}}{=} (z, z^*)^T$  替代实坐标  $r = (x, y)^T$ ，将复变函数  $f(z)$  写成  $f(c) = f(z, z^*)$  的形式。于是，在求函数  $f(z, z^*)$  的偏导时，复变量  $z$  和复共轭变量  $z^*$  可以当作两个相互独立的变量处理，即任何一个相对于另一个都可认为是常数

$$\nabla_z f(z, z^*) = \left. \frac{\partial f(z, z^*)}{\partial z} \right|_{z^*=\text{常数}}, \quad \nabla_{z^*} f(z, z^*) = \left. \frac{\partial f(z, z^*)}{\partial z^*} \right|_{z=\text{常数}} \quad (3.4.8)$$

这意味着，任何一个非全纯的复变函数  $f(z)$  写成  $f(z, z^*)$  之后，都变成了全纯函数，因为对于固定的  $z^*$ ，复变函数  $f(z, z^*)$  在  $z = x + jy$  全平面是解析的；而且对于固定的  $z$  值，复变函数  $f(z, z^*)$  在  $z^* = x - jy$  全平面上也是解析的<sup>[167, 283]</sup>。

例如，复变量  $z$  的实值函数  $f(z, z^*) = |z|^2 = zz^*$  的一阶偏导数  $\frac{\partial |z|^2}{\partial z} = z^*$  和  $\frac{\partial |z|^2}{\partial z^*} = z$  存在，并且连续。也就是说，虽然  $f(z) = |z|^2$  不是全纯函数，但  $f(z, z^*) = |z|^2 = zz^*$  是在  $z = x + jy$  全平面上解析的 ( $z^*$  固定为常数时) 以及在  $z^* = x - jy$  全平面上解析的 ( $z$  固定为常数时)。

非全纯函数与全纯函数的比较如表 3.4.2 所示。

表 3.4.2 非全纯函数与全纯函数的比较

函 数	非全纯函数	全纯函数
坐 标	极坐标 $\begin{cases} r \stackrel{\text{def}}{=} (x, y)^T \in \mathbb{R} \times \mathbb{R} \\ z = x + jy \end{cases}$	共轭坐标 $\begin{cases} c \stackrel{\text{def}}{=} (z, z^*)^T \in \mathbb{C} \times \mathbb{C} \\ z = x + jy, z^* = x - jy \end{cases}$
函数表示	$f(r) = f(x, y)$	$f(c) = f(z, z^*)$

下面是复变函数偏导的常用公式与法则<sup>[283]</sup>：

(1) 复变函数共轭  $f^*(z, z^*)$  关于  $z$  变量共轭  $z^*$  的偏导等于原复变函数  $f(z, z^*)$  关于  $z$  变量的偏导的共轭，即

$$\frac{\partial f^*(z, z^*)}{\partial z^*} = \left( \frac{\partial f(z, z^*)}{\partial z} \right)^* \quad (3.4.9)$$

(2) 复变函数共轭  $f^*(z, z^*)$  关于  $z$  变量的偏导等于原复变函数  $f(z, z^*)$  关于共轭变量  $z^*$  的偏导的共轭，即

$$\frac{\partial f^*(z, z^*)}{\partial z} = \left( \frac{\partial f(z, z^*)}{\partial z^*} \right)^* \quad (3.4.10)$$

(3) 复微分法则

$$df(z, z^*) = \frac{\partial f(z, z^*)}{\partial z} dz + \frac{\partial f(z, z^*)}{\partial z^*} dz^* \quad (3.4.11)$$

(4) 链式法则

$$\frac{\partial h(g(z, z^*))}{\partial z} = \frac{\partial h(g(z, z^*))}{\partial g(z, z^*)} \frac{\partial g(z, z^*)}{\partial z} + \frac{\partial h(g(z, z^*))}{\partial g^*(z, z^*)} \frac{\partial g^*(z, z^*)}{\partial z} \quad (3.4.12)$$

$$\frac{\partial h(g(z, z^*))}{\partial z^*} = \frac{\partial h(g(z, z^*))}{\partial g(z, z^*)} \frac{\partial g(z, z^*)}{\partial z^*} + \frac{\partial h(g(z, z^*))}{\partial g^*(z, z^*)} \frac{\partial g^*(z, z^*)}{\partial z^*} \quad (3.4.13)$$

### 3.4.2 复矩阵微分

复标量变元  $z$  的复变函数  $f(z)$  和全纯函数  $f(z, z^*)$  的概念很容易推广到以复矩阵作变元的矩阵函数  $\mathbf{F}(Z)$  和全纯矩阵函数  $\mathbf{F}(Z, Z^*)$ 。

关于全纯函数，下面的叙述等价<sup>[69]</sup>：

(1) 矩阵函数  $\mathbf{F}(Z)$  是复矩阵变元  $Z$  的全纯函数；

(2) 矩阵微分  $d \text{vec}(\mathbf{F}(Z)) = \frac{\partial \text{vec}(\mathbf{F}(Z))}{\partial (\text{vec}Z)^T} d \text{vec}Z$ ；

(3)  $\frac{\partial \text{vec}(\mathbf{F}(Z))}{\partial (\text{vec}Z^*)^T} = \mathbf{O}$  (零矩阵) 对所有  $Z$  恒成立；

(4)  $\frac{\partial \text{vec}(\mathbf{F}(Z))}{\partial (\text{vec}(\text{Re}Z))^T} + j \frac{\partial \text{vec}(\mathbf{F}(Z))}{\partial (\text{vec}(\text{Im}Z))^T} = \mathbf{O}$  对所有  $Z$  恒成立。

由于满足上述条件，所以矩阵函数  $\mathbf{F}(Z, Z^*)$  为全纯函数，其矩阵微分

$$d \text{vec}(\mathbf{F}(Z, Z^*)) = \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}Z)^T} d \text{vec}Z + \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}Z^*)^T} d \text{vec}Z^* \quad (3.4.14)$$

全纯函数  $\mathbf{F}(Z, Z^*)$  相对于矩阵变元实部  $\text{Re}(Z)$  的偏导

$$\frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}(\text{Re}Z))^T} = \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}Z)^T} + \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))^T}{\partial (\text{vec}Z^*)^T}$$

而相对于矩阵变元虚部  $\text{Im}(Z)$  的偏导

$$\frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}(\text{Im}Z))^T} = j \left( \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))}{\partial (\text{vec}Z)^T} - \frac{\partial \text{vec}(\mathbf{F}(Z, Z^*))^T}{\partial (\text{vec}Z^*)^T} \right)$$

复矩阵微分  $dZ = [dZ_{ij}]_{i=1, j=1}^{m, n}$  具有以下常用性质<sup>[69]</sup>：

(1) 转置  $dZ^T = d(Z^T) = (dZ)^T$

(2) Hermitian 转置  $dZ^H = d(Z^H) = (dZ)^H$

(3) 共轭  $dZ^* = d(Z^*) = (dZ)^*$

(4) 线性 (加法法则)  $d(Y + Z) = dY + dZ$

(5) 链式法则 若  $\mathbf{F}$  是  $\mathbf{Y}$  的函数，而  $\mathbf{Y}$  又是  $\mathbf{Z}$  的函数，则

$$d \text{vec} \mathbf{F} = \frac{\partial \text{vec} \mathbf{F}}{\partial (\text{vec} \mathbf{Y})^T} d \text{vec} \mathbf{Y} = \frac{\partial \text{vec} \mathbf{F}}{\partial (\text{vec} \mathbf{Y})^T} \frac{\partial \text{vec} \mathbf{Y}}{\partial (\text{vec} \mathbf{Z})^T} d \text{vec} \mathbf{Z}$$

式中  $\frac{\partial \text{vec} F}{\partial (\text{vec } Y)^T}$  和  $\frac{\partial \text{vec} F}{\partial (\text{vec } Z)^T}$  分别称为正规复偏导和广义复偏导。

### (6) 乘法法则

$$\begin{aligned} d(\mathbf{U}\mathbf{V}) &= (d\mathbf{U})\mathbf{V} + \mathbf{U}(d\mathbf{V}) \\ d\text{vec}(\mathbf{U}\mathbf{V}) &= (\mathbf{V}^T \otimes \mathbf{I})d\text{vec}\mathbf{U} + (\mathbf{I} \otimes \mathbf{U})d\text{vec}\mathbf{V} \end{aligned}$$

$$(7) \text{ Kronecker 积 } d(\mathbf{Y} \otimes \mathbf{Z}) = d\mathbf{Y} \otimes \mathbf{Z} + \mathbf{Y} \otimes d\mathbf{Z}$$

$$(8) \text{ Hadamard 积 } d(\mathbf{Y} * \mathbf{Z}) = d\mathbf{Y} * \mathbf{Z} + \mathbf{Y} * d\mathbf{Z}$$

下面从单变量的复微分法则出发，具体推导复矩阵微分与复偏导的关系。

### 单个变量的复微分法则

$$df(z, z^*) = \frac{\partial f(z, z^*)}{\partial z} dz + \frac{\partial f(z, z^*)}{\partial z^*} dz^* \quad (3.4.15)$$

很容易推广为多元实标量函数  $f(\cdot) = f((z_1, z_1^*), \dots, (z_m, z_m^*))$  的复微分法则

$$df(\cdot) = \frac{\partial f(\cdot)}{\partial z_1} dz_1 + \dots + \frac{\partial f(\cdot)}{\partial z_m} dz_m + \frac{\partial f(\cdot)}{\partial z_1^*} dz_1^* + \dots + \frac{\partial f(\cdot)}{\partial z_m^*} dz_m^* \quad (3.4.16)$$

这一复微分法则是复矩阵微分的基础。

将  $m \times 1$  复变元向量  $\mathbf{z} = [z_1, \dots, z_m]^T$  的每个元素视为多元实标量函数的复变量，则由多元实标量函数的复微分法则，得到复微分法则的向量形式

$$\begin{aligned} df(\mathbf{z}, \mathbf{z}^*) &= \left[ \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_1}, \dots, \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_m} \right] \begin{bmatrix} dz_1 \\ \vdots \\ dz_m \end{bmatrix} + \left[ \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_1^*}, \dots, \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_m^*} \right] \begin{bmatrix} dz_1^* \\ \vdots \\ dz_m^* \end{bmatrix} \\ &= \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^T} d\mathbf{z} + \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^H} d\mathbf{z}^* \end{aligned}$$

或简记作

$$df(\mathbf{z}, \mathbf{z}^*) = D_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*) d\mathbf{z} + D_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*) d\mathbf{z}^* \quad (3.4.17)$$

式中  $d\mathbf{z} = [dz_1, \dots, dz_m]^T$  和  $d\mathbf{z}^* = [dz_1^*, \dots, dz_m^*]^T$ ，而

$$D_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*) = \left. \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^T} \right|_{\mathbf{z}^*=\text{常数向量}} = \left[ \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_1}, \dots, \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_m} \right] \quad (3.4.18)$$

$$D_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*) = \left. \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^H} \right|_{\mathbf{z}=\text{常数向量}} = \left[ \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_1^*}, \dots, \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial z_m^*} \right] \quad (3.4.19)$$

分别是实标量函数  $f(\mathbf{z}, \mathbf{z}^*)$  的协梯度向量和共轭协梯度向量。其中

$$D_{\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}^T} \stackrel{\text{def}}{=} \left[ \frac{\partial}{\partial z_1}, \dots, \frac{\partial}{\partial z_m} \right] \quad (3.4.20)$$

$$D_{\mathbf{z}^*} = \frac{\partial}{\partial \mathbf{z}^H} \stackrel{\text{def}}{=} \left[ \frac{\partial}{\partial z_1^*}, \dots, \frac{\partial}{\partial z_m^*} \right] \quad (3.4.21)$$

分别称作复变元列向量  $\mathbf{z} \in \mathbb{C}^m$  的协梯度算子 (cogradient operator) 和共轭协梯度算子 (conjugate cogradient operator)。

令  $\mathbf{z} = \mathbf{x} + j\mathbf{y} = [z_1, \dots, z_m]^T \in \mathbb{C}^m$ , 其中  $\mathbf{x} = [x_1, \dots, x_m]^T \in \mathbb{R}^m$ ,  $\mathbf{y} = [y_1, \dots, y_m]^T \in \mathbb{R}^m$ , 即  $z_i = x_i + jy_i, i = 1, \dots, m$ , 并且实部  $x_i$  与虚部  $y_i$  是相互独立的变元。

对行向量  $\mathbf{z}^T = [z_1, \dots, z_m]$  的每个元素运用实标量函数的偏导算子

$$\mathbf{D}_{\mathbf{z}} = \frac{\partial}{\partial z_i} = \frac{1}{2} \left( \frac{\partial}{\partial x_i} - j \frac{\partial}{\partial y_i} \right) \quad \text{和} \quad \mathbf{D}_{\mathbf{z}^*} = \frac{\partial}{\partial z_i^*} = \frac{1}{2} \left( \frac{\partial}{\partial x_i} + j \frac{\partial}{\partial y_i} \right) \quad (3.4.22)$$

便得到用复变向量  $\mathbf{z}$  的实部  $\mathbf{x}$  与虚部  $\mathbf{y}$  表示的协梯度算子

$$\mathbf{D}_{\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}^T} = \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}^T} - j \frac{\partial}{\partial \mathbf{y}^T} \right) \quad (3.4.23)$$

和共轭协梯度算子

$$\mathbf{D}_{\mathbf{z}^*} = \frac{\partial}{\partial \mathbf{z}^H} = \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}^T} + j \frac{\partial}{\partial \mathbf{y}^T} \right) \quad (3.4.24)$$

类似地, 梯度算子 (gradient operator) 和共轭梯度算子 (conjugate gradient operator) 采用列向量形式, 分别定义为

$$\nabla_{\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}} \stackrel{\text{def}}{=} \left[ \frac{\partial}{\partial z_1}, \dots, \frac{\partial}{\partial z_m} \right]^T \quad (3.4.25)$$

$$\nabla_{\mathbf{z}^*} = \frac{\partial}{\partial \mathbf{z}^*} \stackrel{\text{def}}{=} \left[ \frac{\partial}{\partial z_1^*}, \dots, \frac{\partial}{\partial z_m^*} \right]^T \quad (3.4.26)$$

于是, 实标量函数  $f(\mathbf{z}, \mathbf{z}^*)$  的梯度向量和共轭梯度向量分别定义为

$$\nabla_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*) = \left. \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}} \right|_{\mathbf{z}^*=\text{常数向量}} = (\mathbf{D}_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*))^T \quad (3.4.27)$$

$$\nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*) = \left. \frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^*} \right|_{\mathbf{z}=\text{常数向量}} = (\mathbf{D}_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*))^T \quad (3.4.28)$$

对复列向量  $\mathbf{z} = [z_1, \dots, z_m]^T$  的每个元素运用实标量函数的偏导算子, 立即得到用复变向量  $\mathbf{z}$  的实部  $\mathbf{x}$  与虚部  $\mathbf{y}$  表示的梯度算子

$$\nabla_{\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}} = \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}} - j \frac{\partial}{\partial \mathbf{y}} \right) \quad (3.4.29)$$

和共轭梯度算子

$$\nabla_{\mathbf{z}^*} = \frac{\partial}{\partial \mathbf{z}^*} = \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}} + j \frac{\partial}{\partial \mathbf{y}} \right) \quad (3.4.30)$$

利用梯度算子和共轭梯度算子的定义公式, 不难求出

$$\frac{\partial \mathbf{z}^T}{\partial \mathbf{z}} = \frac{\partial \mathbf{x}^T}{\partial \mathbf{z}} + j \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}} = \frac{1}{2} \left( \frac{\partial \mathbf{x}^T}{\partial \mathbf{x}} - j \frac{\partial \mathbf{x}^T}{\partial \mathbf{y}} \right) + j \frac{1}{2} \left( \frac{\partial \mathbf{y}^T}{\partial \mathbf{x}} - j \frac{\partial \mathbf{y}^T}{\partial \mathbf{y}} \right) = \mathbf{I}_{m \times m}$$

$$\frac{\partial \mathbf{z}^T}{\partial \mathbf{z}^*} = \frac{\partial \mathbf{x}^T}{\partial \mathbf{z}^*} + j \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}^*} = \frac{1}{2} \left( \frac{\partial \mathbf{x}^T}{\partial \mathbf{x}} + j \frac{\partial \mathbf{x}^T}{\partial \mathbf{y}} \right) + j \frac{1}{2} \left( \frac{\partial \mathbf{y}^T}{\partial \mathbf{x}} + j \frac{\partial \mathbf{y}^T}{\partial \mathbf{y}} \right) = \mathbf{O}_{m \times m}$$

式中使用了  $\frac{\partial \mathbf{x}^T}{\partial \mathbf{x}} = \mathbf{I}_{m \times m}$ ,  $\frac{\partial \mathbf{x}^T}{\partial \mathbf{y}} = \mathbf{O}_{m \times m}$  和  $\frac{\partial \mathbf{y}^T}{\partial \mathbf{y}} = \mathbf{I}_{m \times m}$ ,  $\frac{\partial \mathbf{y}^T}{\partial \mathbf{x}} = \mathbf{O}_{m \times m}$  等结果, 因为  $\mathbf{z}$  的实部  $\mathbf{x}$  与虚部  $\mathbf{y}$  相互独立。

将上述结果以及它们的共轭、转置和复共轭转置一并写出, 便有下列重要结果

$$\frac{\partial \mathbf{z}^T}{\partial \mathbf{z}} = \mathbf{I}, \quad \frac{\partial \mathbf{z}^H}{\partial \mathbf{z}^*} = \mathbf{I}, \quad \frac{\partial \mathbf{z}}{\partial \mathbf{z}^T} = \mathbf{I}, \quad \frac{\partial \mathbf{z}^*}{\partial \mathbf{z}^H} = \mathbf{I} \quad (3.4.31)$$

$$\frac{\partial \mathbf{z}^T}{\partial \mathbf{z}^*} = \mathbf{O}, \quad \frac{\partial \mathbf{z}^H}{\partial \mathbf{z}} = \mathbf{O}, \quad \frac{\partial \mathbf{z}}{\partial \mathbf{z}^H} = \mathbf{O}, \quad \frac{\partial \mathbf{z}^*}{\partial \mathbf{z}^T} = \mathbf{O} \quad (3.4.32)$$

上述结果揭示了复微分的一个重要事实: 在复向量的实部与虚部相互独立的基本假设下, 复向量变元  $\mathbf{z}$  与其复共轭向量变元  $\mathbf{z}^*$  可以视为两个相互独立的变元。这一重要事实一点也不奇怪: 因为这两个向量之间的夹角为  $\pi/2$ , 相互正交。于是, 可以总结出协梯度算子和梯度算子的下列应用法则:

(1) 无论是使用协梯度算子  $\frac{\partial}{\partial \mathbf{z}^T}$  还是梯度算子  $\frac{\partial}{\partial \mathbf{z}}$ , 复共轭变元向量  $\mathbf{z}^*$  都可以视为一常数向量;

(2) 无论是使用共轭协梯度算子  $\frac{\partial}{\partial \mathbf{z}^H}$  还是共轭梯度算子  $\frac{\partial}{\partial \mathbf{z}^*}$ , 向量  $\mathbf{z}$  均可以当作一常数向量处理。

不妨称上述法则为复偏导算子的独立法则: 当使用复偏导算子(协梯度算子、共轭协梯度算子、梯度算子和共轭梯度算子)时, 复变元向量  $\mathbf{z}$  和  $\mathbf{z}^*$  可以当作两个相互独立的变元向量处理, 即其中一个向量作为变元时, 另一个向量便可视为常数向量。

现在考虑以矩阵作变元的实标量函数  $f(\mathbf{Z}, \mathbf{Z}^*)$ , 其中  $\mathbf{Z} \in \mathbb{C}^{m \times n}$ 。将矩阵变元  $\mathbf{Z}$  和  $\mathbf{Z}^*$  分别向量化, 则由式 (3.4.17) 得到实标量函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的(一阶)复微分法则

$$\begin{aligned} df(\mathbf{Z}, \mathbf{Z}^*) &= \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})^T} d(\text{vec } \mathbf{Z}) + \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)^T} d(\text{vec } \mathbf{Z}^*) \\ &= \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})^T} d(\text{vec } \mathbf{Z}) + \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)^T} d(\text{vec } \mathbf{Z}^*) \end{aligned} \quad (3.4.33)$$

式中

$$\begin{aligned} \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})^T} &= \left[ \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}} \right] \\ \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)^T} &= \left[ \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}^*}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}^*}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}^*}, \dots, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}^*} \right] \end{aligned}$$

定义

$$\mathbf{D}_{\text{vec } \mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})^T}, \quad \mathbf{D}_{\text{vec } \mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)^T} \quad (3.4.34)$$

分别为函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的协梯度向量和共轭协梯度向量。

函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的梯度向量和共轭梯度向量分别用符号  $\nabla_{\text{vec } \mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*)$  和  $\nabla_{\text{vec } \mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*)$  表示, 并分别定义为

$$\nabla_{\text{vec } \mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec } \mathbf{Z}}, \quad \nabla_{\text{vec } \mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec } \mathbf{Z}^*} \quad (3.4.35)$$

共轭梯度向量  $\nabla_{\text{vec}Z^*} f(Z, Z^*)$  具有以下性质 [69]:

- (1) 共轭梯度向量在函数  $f(Z, Z^*)$  的极值点等于零向量, 即  $\nabla_{\text{vec}Z^*} f(Z, Z^*) = \mathbf{0}$ 。
- (2) 共轭梯度向量  $\nabla_{\text{vec}Z^*} f(Z, Z^*)$  指向函数  $f(Z, Z^*)$  的最陡增大斜率方向 (direction of steepest slope), 而负共轭梯度向量  $-\nabla_{\text{vec}Z^*} f(Z, Z^*)$  则指向函数  $f(Z, Z^*)$  的最陡下降斜率方向。
- (3) 最陡增大斜率的幅值等于  $\|\nabla_{\text{vec}Z^*} f(Z, Z^*)\|_2$ 。
- (4) 共轭梯度向量  $\nabla_{\text{vec}Z^*} f(Z, Z^*)$  是曲面  $f(Z, Z^*) = \text{const}$  的法线。这意味着, 共轭梯度向量  $\nabla_{\text{vec}Z^*} f(Z, Z^*)$  和负共轭梯度向量  $-\nabla_{\text{vec}Z^*} f(Z, Z^*)$  可以分别用于梯度上升算法和梯度下降算法。

以上性质对函数  $f(z, z^*)$  的共轭梯度向量  $\nabla_{z^*} f(z, z^*)$  同样适用。

另一方面, 实标量函数  $f(Z, Z^*)$  的 Jacobian 矩阵和共轭 Jacobian 矩阵分别定义为

$$D_Z f(Z, Z^*) \stackrel{\text{def}}{=} \left. \frac{\partial f(Z, Z^*)}{\partial Z^T} \right|_{Z^*=\text{常数矩阵}} = \begin{bmatrix} \frac{\partial f(Z, Z^*)}{\partial Z_{11}} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{m1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(Z, Z^*)}{\partial Z_{1n}} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{mn}} \end{bmatrix} \quad (3.4.36)$$

$$D_{Z^*} f(Z, Z^*) \stackrel{\text{def}}{=} \left. \frac{\partial f(Z, Z^*)}{\partial Z^H} \right|_{Z=\text{常数矩阵}} = \begin{bmatrix} \frac{\partial f(Z, Z^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{m1}^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(Z, Z^*)}{\partial Z_{1n}^*} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{mn}^*} \end{bmatrix} \quad (3.4.37)$$

类似地, 实标量函数  $f(Z, Z^*)$  的复梯度矩阵和共轭梯度矩阵分别定义为

$$\nabla_Z f(Z, Z^*) \stackrel{\text{def}}{=} \left. \frac{\partial f(Z, Z^*)}{\partial Z} \right|_{Z^*=\text{常数矩阵}} = \begin{bmatrix} \frac{\partial f(Z, Z^*)}{\partial Z_{11}} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(Z, Z^*)}{\partial Z_{m1}} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{mn}} \end{bmatrix} \quad (3.4.38)$$

$$\nabla_{Z^*} f(Z, Z^*) \stackrel{\text{def}}{=} \left. \frac{\partial f(Z, Z^*)}{\partial Z^*} \right|_{Z=\text{常数矩阵}} = \begin{bmatrix} \frac{\partial f(Z, Z^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{1n}^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(Z, Z^*)}{\partial Z_{m1}^*} & \dots & \frac{\partial f(Z, Z^*)}{\partial Z_{mn}^*} \end{bmatrix} \quad (3.4.39)$$

综合以上定义, 实标量函数  $f(Z, Z)$  的各种偏导有下列关系:

- (1) 共轭(协)梯度向量等于(协)梯度向量的复数共轭, 共轭 Jacobian 矩阵等于 Jacobian 矩阵的复数共轭, 共轭梯度矩阵等于复梯度矩阵的复数共轭。
- (2) (共轭)梯度向量等于(共轭)协梯度向量的转置

$$\nabla_{\text{vec}Z} f(Z, Z^*) = D_{\text{vec}Z}^T f(Z, Z^*) \quad (3.4.40)$$

$$\nabla_{\text{vec}Z^*} f(Z, Z^*) = D_{\text{vec}Z^*}^T f(Z, Z^*) \quad (3.4.41)$$

(3) (共轭) 协梯度向量等于 (共轭) Jacobian 矩阵的向量化的转置

$$\mathbf{D}_{\text{vec}Z} f(\mathbf{Z}, \mathbf{Z}) = \text{vec}^T (\mathbf{D}_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*)) \quad (3.4.42)$$

$$\mathbf{D}_{\text{vec}Z^*} f(\mathbf{Z}, \mathbf{Z}) = \text{vec}^T (\mathbf{D}_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*)) \quad (3.4.43)$$

(4) (共轭) 梯度矩阵等于 (共轭) Jacobian 矩阵的转置

$$\nabla_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{D}_{\mathbf{Z}}^T f(\mathbf{Z}, \mathbf{Z}^*) \quad (3.4.44)$$

$$\nabla_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{D}_{\mathbf{Z}^*}^T f(\mathbf{Z}, \mathbf{Z}^*) \quad (3.4.45)$$

下面是有关梯度的运算法则：

(1) 若  $f(\mathbf{Z}, \mathbf{Z}^*) = c$  为常数，则梯度矩阵和共轭梯度矩阵均等于零矩阵，即  $\frac{\partial c}{\partial \mathbf{Z}} = \mathbf{O}$  和  $\frac{\partial c}{\partial \mathbf{Z}^*} = \mathbf{O}$ 。

(2) 线性法则 若  $f(\mathbf{Z}, \mathbf{Z}^*)$  和  $g(\mathbf{Z}, \mathbf{Z}^*)$  都是实标量函数，而  $c_1$  和  $c_2$  为复常数，则

$$\frac{\partial [c_1 f(\mathbf{Z}, \mathbf{Z}^*) + c_2 g(\mathbf{Z}, \mathbf{Z}^*)]}{\partial \mathbf{Z}^*} = c_1 \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} + c_2 \frac{\partial g(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*}$$

(3) 乘积法则

$$\frac{\partial f(\mathbf{Z}, \mathbf{Z}^*) g(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} = g(\mathbf{Z}, \mathbf{Z}^*) \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} + f(\mathbf{Z}, \mathbf{Z}^*) \frac{\partial g(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*}$$

(4) 商法则 若  $g(\mathbf{Z}, \mathbf{Z}^*) \neq 0$ ，则

$$\frac{\partial f(\mathbf{Z}, \mathbf{Z}^*) / g(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} = \frac{1}{g^2(\mathbf{Z}, \mathbf{Z}^*)} \left[ g(\mathbf{Z}, \mathbf{Z}^*) \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} - f(\mathbf{Z}, \mathbf{Z}^*) \frac{\partial g(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} \right]$$

若  $h(\mathbf{Z}, \mathbf{Z}^*) = g(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*), \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))$ ，则

$$\begin{aligned} \frac{\partial h(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec} \mathbf{Z}} &= \frac{\partial g(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*), \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))}{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T} \cdot \frac{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}} + \\ &\quad \frac{\partial g(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*), \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))}{\partial (\text{vec} \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))^T} \cdot \frac{\partial (\text{vec} \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}} \end{aligned} \quad (3.4.46)$$

$$\begin{aligned} \frac{\partial h(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec} \mathbf{Z}^*} &= \frac{\partial g(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*), \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))}{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T} \cdot \frac{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}^*} + \\ &\quad \frac{\partial g(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*), \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))}{\partial (\text{vec} \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))^T} \cdot \frac{\partial (\text{vec} \mathbf{F}^*(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}^*} \end{aligned} \quad (3.4.47)$$

### 3.4.3 复 Hessian 矩阵

考虑以复矩阵为变元的实值函数  $f = f(\mathbf{Z}, \mathbf{Z}^*)$ ，其微分可以等价写作

$$df = (\mathbf{D}_{\mathbf{Z}} f) d \text{vec} \mathbf{Z} + (\mathbf{D}_{\mathbf{Z}^*} f) d \text{vec} \mathbf{Z}^* \quad (3.4.48)$$

式中

$$D_Z f = \frac{\partial f}{\partial(\text{vec } Z)^T}, \quad D_{Z^*} f = \frac{\partial f}{\partial(\text{vec } Z^*)^T} \quad (3.4.49)$$

注意到其微分为行向量, 故

$$\begin{aligned} d(D_Z f) &= \left( \frac{\partial D_Z f}{\partial \text{vec } Z} d \text{vec } Z + \frac{\partial D_Z f}{\partial \text{vec } Z^*} d \text{vec } Z^* \right)^T \\ &= (d \text{vec } Z)^T \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z)^T} + (d \text{vec } Z^*)^T \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z)^T} \end{aligned} \quad (3.4.50)$$

$$\begin{aligned} d(D_{Z^*} f) &= \left( \frac{\partial D_{Z^*} f}{\partial \text{vec } Z} d \text{vec } Z + \frac{\partial D_{Z^*} f}{\partial \text{vec } Z^*} d \text{vec } Z^* \right)^T \\ &= (d \text{vec } Z)^T \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z^*)^T} + (d \text{vec } Z^*)^T \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z^*)^T} \end{aligned} \quad (3.4.51)$$

由于  $d(\text{vec } Z)$  不是  $\text{vec } Z$  的函数, 且  $d(\text{vec } Z^*)$  不是  $\text{vec } Z^*$  的函数, 故有

$$d^2 \text{vec } Z = d(d \text{vec } Z) = 0 \quad \text{和} \quad d^2 \text{vec } Z^* = d(d \text{vec } Z^*) = 0 \quad (3.4.52)$$

于是, 实值函数  $f = f(Z, Z^*)$  的二次微分

$$\begin{aligned} d^2 f &= d(D_Z f) d \text{vec } Z + d(D_{Z^*} f) d \text{vec } Z^* \\ &= (d \text{vec } Z)^T \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z)^T} d \text{vec } Z + (d \text{vec } Z^*)^T \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z)^T} d \text{vec } Z \\ &\quad + (d \text{vec } Z)^T \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z^*)^T} d \text{vec } Z^* + (d \text{vec } Z^*)^T \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z^*)^T} d \text{vec } Z^* \end{aligned}$$

或写作

$$\begin{aligned} d^2 f &= [(d \text{vec } Z^*)^T, (d \text{vec } Z)^T] \begin{bmatrix} \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z)^T} & \frac{\partial^2 f}{\partial \text{vec } Z^* \partial(\text{vec } Z^*)^T} \\ \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z)^T} & \frac{\partial^2 f}{\partial \text{vec } Z \partial(\text{vec } Z^*)^T} \end{bmatrix} \begin{bmatrix} d \text{vec } Z \\ d \text{vec } Z^* \end{bmatrix} \\ &= [(d \text{vec } Z^*)^T, (d \text{vec } Z)^T] \begin{bmatrix} H_{Z^*, Z} & H_{Z^*, Z^*} \\ H_{Z, Z} & H_{Z, Z^*} \end{bmatrix} \begin{bmatrix} d \text{vec } Z \\ d \text{vec } Z^* \end{bmatrix} \\ &= \begin{bmatrix} d \text{vec } Z \\ d \text{vec } Z^* \end{bmatrix}^H H \begin{bmatrix} d \text{vec } Z \\ d \text{vec } Z^* \end{bmatrix} \end{aligned} \quad (3.4.53)$$

式中

$$H = \begin{bmatrix} H_{Z^*, Z} & H_{Z^*, Z^*} \\ H_{Z, Z} & H_{Z, Z^*} \end{bmatrix} \quad (3.4.54)$$

称为函数  $f(Z, Z^*)$  的全 Hessian 矩阵, 其四个分块矩阵

$$\left. \begin{aligned} H_{Z^*, Z} &= \frac{\partial^2 f(Z, Z^*)}{\partial \text{vec } Z^* \partial(\text{vec } Z)^T} \\ H_{Z^*, Z^*} &= \frac{\partial^2 f(Z, Z^*)}{\partial \text{vec } Z^* \partial(\text{vec } Z^*)^T} \\ H_{Z, Z} &= \frac{\partial^2 f(Z, Z^*)}{\partial \text{vec } Z \partial(\text{vec } Z)^T} \\ H_{Z, Z^*} &= \frac{\partial^2 f(Z, Z^*)}{\partial \text{vec } Z \partial(\text{vec } Z^*)^T} \end{aligned} \right\} \quad (3.4.55)$$

分别称为函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的部分 Hessian 矩阵, 其中  $\mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}}$  是函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的主 Hessian 矩阵。

根据上述定义公式, 容易证明标量函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的 Hessian 矩阵的以下性质:

- (1) 部分 Hessian 矩阵  $\mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}}$  和  $\mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*}$  分别是 Hermitian 矩阵, 并且相互共轭

$$\mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}} = \mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}}^H, \quad \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*} = \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*}^H, \quad \mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}} = \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*}^*$$
 (3.4.56)

- (2) 另外两个部分 Hessian 矩阵分别是对称的, 并且互为共轭

$$\mathbf{H}_{\mathbf{Z}, \mathbf{Z}} = \mathbf{H}_{\mathbf{Z}, \mathbf{Z}}^T, \quad \mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}^*} = \mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}^*}^T \quad \mathbf{H}_{\mathbf{Z}, \mathbf{Z}} = \mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}^*}^* \quad (3.4.57)$$

- (3) 全 Hessian 矩阵是 Hermitian 矩阵, 即有  $\mathbf{H} = \mathbf{H}^H$ 。

另由式 (3.4.53) 易知, 由于实值函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的二阶微分  $d^2f$  为二次型函数, 故全 Hessian 矩阵的正定性由  $d^2f$  决定:

- (1) 若  $d^2f > 0$  对所有  $\text{vec}\mathbf{Z}$  恒成立, 则全 Hessian 矩阵为正定矩阵;
- (2) 若  $d^2f \geq 0$  对所有  $\text{vec}\mathbf{Z}$  恒成立, 则全 Hessian 矩阵为半正定矩阵;
- (3) 若  $d^2f < 0$  对所有  $\text{vec}\mathbf{Z}$  恒成立, 则全 Hessian 矩阵为负定矩阵;
- (4) 若  $d^2f \leq 0$  对所有  $\text{vec}\mathbf{Z}$  恒成立, 则全 Hessian 矩阵为半负定矩阵。

对于实值函数  $f = f(\mathbf{z}, \mathbf{z}^*)$ , 其二阶微分

$$d^2f = \begin{bmatrix} d\mathbf{z} \\ d\mathbf{z}^* \end{bmatrix}^H \mathbf{H} \begin{bmatrix} d\mathbf{z} \\ d\mathbf{z}^* \end{bmatrix} \quad (3.4.58)$$

其中  $\mathbf{H}$  是函数  $f(\mathbf{z}, \mathbf{z}^*)$  的全 Hessian 矩阵, 定义为

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{\mathbf{z}^*, \mathbf{z}} & \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*} \\ \mathbf{H}_{\mathbf{z}, \mathbf{z}} & \mathbf{H}_{\mathbf{z}, \mathbf{z}^*} \end{bmatrix} \quad (3.4.59)$$

四个部分 Hessian 矩阵分别为

$$\left. \begin{aligned} \mathbf{H}_{\mathbf{z}^*, \mathbf{z}} &= \frac{\partial^2 f(\mathbf{z}, \mathbf{Z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^T} \\ \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*} &= \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^H} \\ \mathbf{H}_{\mathbf{z}, \mathbf{z}} &= \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^T} \\ \mathbf{H}_{\mathbf{z}, \mathbf{z}^*} &= \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^H} \end{aligned} \right\} \quad (3.4.60)$$

显然, 全 Hessian 矩阵为 Hermitian 矩阵, 即有  $\mathbf{H} = \mathbf{H}^H$ , 并且四个部分 Hessian 矩阵之间有下列关系

$$\mathbf{H}_{\mathbf{z}^*, \mathbf{z}} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}}^H, \quad \mathbf{H}_{\mathbf{z}, \mathbf{z}^*} = \mathbf{H}_{\mathbf{z}, \mathbf{z}^*}^H, \quad \mathbf{H}_{\mathbf{z}^*, \mathbf{z}} = \mathbf{H}_{\mathbf{z}, \mathbf{z}^*}^* \quad (3.4.61)$$

$$\mathbf{H}_{\mathbf{z}, \mathbf{z}} = \mathbf{H}_{\mathbf{z}, \mathbf{z}}^T, \quad \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*}^T \quad \mathbf{H}_{\mathbf{z}, \mathbf{z}} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*}^* \quad (3.4.62)$$

### 3.5 复梯度矩阵与复 Hessian 矩阵的辨识

上一节定义了以复矩阵为变元的实标量函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的复梯度矩阵与复 Hessian 矩阵。本节介绍复梯度矩阵和复 Hessian 矩阵的辨识方法。

#### 3.5.1 实标量函数的复梯度矩阵辨识

若令

$$\mathbf{A} = \mathbf{D}_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) \quad \text{和} \quad \mathbf{B} = \mathbf{D}_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) \quad (3.5.1)$$

则

$$\frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}^T(\mathbf{Z})} = \text{rvec}(\mathbf{D}_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*)) = \text{rvec}(\mathbf{A}) = \text{vec}^T(\mathbf{A}^T) \quad (3.5.2)$$

$$\frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}^H(\mathbf{Z})} = \text{rvec}(\mathbf{D}_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*)) = \text{rvec}(\mathbf{B}) = \text{vec}^T(\mathbf{B}^T) \quad (3.5.3)$$

于是, 一阶复矩阵微分公式 (3.4.33) 可以写成

$$df(\mathbf{Z}, \mathbf{Z}^*) = \text{vec}^T(\mathbf{A}^T) d(\text{vec} \mathbf{Z}) + \text{vec}^T(\mathbf{B}^T) d(\text{vec} \mathbf{Z}^*) \quad (3.5.4)$$

利用  $\text{tr}(\mathbf{C}^T \mathbf{D}) = \text{vec}^T(\mathbf{C}) \text{vec}(\mathbf{D})$ , 式 (3.5.4) 可以用迹函数形式表示为

$$df(\mathbf{Z}, \mathbf{Z}^*) = \text{tr}(\mathbf{A} d\mathbf{Z} + \mathbf{B} d\mathbf{Z}^*) \quad (3.5.5)$$

由式 (3.5.1) 和式 (3.5.5), 即可得到复 Jacobian 矩阵和复梯度矩阵的辨识命题如下。

**命题 3.5.1** 给定一标量函数  $f(\mathbf{Z}, \mathbf{Z}^*) : \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \rightarrow \mathbb{C}$ , 则该函数关于复矩阵变元的 Jacobian 矩阵和共轭 Jacobian 矩阵可以辨识如下

$$df(\mathbf{Z}, \mathbf{Z}^*) = \text{tr}(\mathbf{A} d\mathbf{Z} + \mathbf{B} d\mathbf{Z}^*) \iff \begin{cases} \mathbf{D}_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{A} \\ \mathbf{D}_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{B} \end{cases} \quad (3.5.6)$$

或有

$$df(\mathbf{Z}, \mathbf{Z}^*) = \text{tr}(\mathbf{A} d\mathbf{Z} + \mathbf{B} d\mathbf{Z}^*) \iff \begin{cases} \nabla_{\mathbf{Z}} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{A}^T \\ \nabla_{\mathbf{Z}^*} f(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{B}^T \end{cases} \quad (3.5.7)$$

即复梯度矩阵和共轭梯度矩阵分别由矩阵  $\mathbf{A}$  和  $\mathbf{B}$  的转置唯一辨识。

上述命题表明, 复 Jacobian 矩阵和复梯度矩阵的辨识关键是将标量函数的矩阵微分表示成规范形式  $df(\mathbf{Z}, \mathbf{Z}^*) = \text{tr}(\mathbf{A} d\mathbf{Z} + \mathbf{B} d\mathbf{Z}^*)$ 。特别地, 若  $f(\mathbf{Z}, \mathbf{Z}^*)$  为实值函数, 则  $\mathbf{B} = \mathbf{A}^*$ 。

#### 例 3.5.1 迹函数 $\text{tr}(\mathbf{Z} \mathbf{A} \mathbf{Z}^* \mathbf{B})$ 的复矩阵微分

$$\begin{aligned} d[\text{tr}(\mathbf{Z} \mathbf{A} \mathbf{Z}^* \mathbf{B})] &= \text{tr}((d\mathbf{Z}) \mathbf{A} \mathbf{Z}^* \mathbf{B}) + \text{tr}(\mathbf{Z} \mathbf{A} (d\mathbf{Z}^*) \mathbf{B}) \\ &= \text{tr}(\mathbf{A} \mathbf{Z}^* \mathbf{B} d\mathbf{Z}) + \text{tr}(\mathbf{B} \mathbf{Z} \mathbf{A} d\mathbf{Z}^*) \end{aligned}$$

由此得迹函数  $\text{tr}(ZAZ^*B)$  的梯度矩阵和共轭梯度矩阵分别为

$$\nabla_Z \text{tr}(ZAZ^*B) = (AZ^*B)^T = B^T Z^H A^T \quad (3.5.8)$$

$$\nabla_{Z^*} \text{tr}(ZAZ^*B) = (BZA)^T = A^T Z^T B^T \quad (3.5.9)$$

表 3.5.1 列出了几种迹函数的微分、梯度矩阵与共轭梯度矩阵。

表 3.5.1 几种迹函数的微分、梯度矩阵与共轭梯度矩阵

$f(Z, Z^*)$	微分 $df$	梯度矩阵 $\partial f / \partial Z$	共轭梯度矩阵 $\partial f / \partial Z^*$
$\text{tr}(AZ)$	$\text{tr}(AdZ)$	$A^T$	$O$
$\text{tr}(AZ^H)$	$\text{tr}(A^T dZ^*)$	$O$	$A$
$\text{tr}(ZAZ^T B)$	$\text{tr}((AZ^T B + A^T Z^T B^T)dZ)$	$B^T Z A^T + B Z A$	$O$
$\text{tr}(ZAZB)$	$\text{tr}((AZB + BZA)dZ)$	$(AZB + BZA)^T$	$O$
$\text{tr}(ZAZ^*B)$	$\text{tr}(AZ^*BdZ + BZA dZ^*)$	$B^T Z^H A^T$	$A^T Z^T B^T$
$\text{tr}(ZAZ^H B)$	$\text{tr}(AZ^H B dZ + A^T Z^T B^T dZ^*)$	$B^T Z^H A^T$	$BZA$
$\text{tr}(AZ^{-1})$	$-\text{tr}(Z^{-1} AZ^{-1} dZ)$	$-Z^{-T} A^T Z^{-T}$	$O$
$\text{tr}(Z^k)$	$k\text{tr}(Z^{k-1} dZ)$	$k(Z^T)^{k-1}$	$O$

例 3.5.2 行列式  $|ZZ^*|$  和  $|ZZ^H|$  的复矩阵微分分别为

$$\begin{aligned} d|ZZ^*| &= |ZZ^*| \text{tr}[(ZZ^*)^{-1} d(ZZ^*)] \\ &= |ZZ^*| \text{tr}[(ZZ^*)^{-1} (dZ) Z^*] + |ZZ^*| \text{tr}[(ZZ^*)^{-1} Z dZ^*] \\ &= |ZZ^*| \text{tr}[Z^* (ZZ^*)^{-1} dZ] + |ZZ^*| \text{tr}[(ZZ^*)^{-1} Z dZ^*] \end{aligned}$$

$$\begin{aligned} d|ZZ^H| &= |ZZ^H| \text{tr}[(ZZ^H)^{-1} d(ZZ^H)] \\ &= |ZZ^H| \text{tr}[(ZZ^H)^{-1} (dZ) Z^H] + |ZZ^H| \text{tr}[(ZZ^H)^{-1} Z dZ^H] \\ &= |ZZ^H| \text{tr}[Z^H (ZZ^H)^{-1} dZ] + |ZZ^H| \text{tr}\{[(ZZ^H)^{-1} Z]^T dZ^*\} \end{aligned}$$

故梯度矩阵与共轭梯度矩阵分别为

$$\nabla_Z |ZZ^*| = |ZZ^*| (Z^H Z^T)^{-1} Z^* \quad (3.5.10)$$

$$\nabla_{Z^*} |ZZ^*| = |ZZ^*| Z^T (Z^H Z^T)^{-1} \quad (3.5.11)$$

和

$$\nabla_Z |ZZ^H| = |ZZ^H| (Z^* Z^T)^{-1} Z^* \quad (3.5.12)$$

$$\nabla_{Z^*} |ZZ^H| = |ZZ^H| Z^T (Z^* Z^T)^{-1} \quad (3.5.13)$$

例 3.5.3 矩阵整数幂的行列式  $|Z^k|$  的微分

$$d|Z^k| = |Z^k| \text{tr}(Z^{-k} dZ^k) = |Z|^k \text{tr}(Z^{-k} k Z^{k-1} dZ) = k |Z|^k \text{tr}(Z^{-1} dZ)$$

由此得梯度矩阵与共轭梯度矩阵

$$\nabla_Z |Z^k| = k|Z|^k Z^{-T}, \quad \nabla_{Z^*} |Z^k| = O \quad (3.5.14)$$

表 3.5.2 罗列了几种行列式函数的微分、梯度矩阵与共轭梯度矩阵。

表 3.5.2 几种行列式函数的微分、梯度矩阵与共轭梯度矩阵

函数 $f$	微分 $df$	$\partial f / \partial Z$	$\partial f / \partial Z^*$
$ Z $	$ Z  \text{tr}(Z^{-1} dZ)$	$ Z  Z^{-T}$	$O$
$ ZZ^T $	$2 ZZ^T  \text{tr}(Z^T(ZZ^T)^{-1} dZ)$	$2 ZZ^T (ZZ^T)^{-1} Z$	$O$
$ Z^T Z $	$2 Z^T Z  \text{tr}((Z^T Z)^{-1} Z^T dZ)$	$2 Z^T Z  Z(Z^T Z)^{-1}$	$O$
$ ZZ^* $	$ZZ^* \text{tr}(Z^*(ZZ^*)^{-1} dZ + (ZZ^*)^{-1} Z dZ^*)$	$ ZZ^* (Z^H Z^T)^{-1} Z^H$	$ ZZ^*  Z^T (Z^H Z^T)^{-1}$
$ Z^* Z $	$Z^* Z \text{tr}((Z^* Z)^{-1} Z^* dZ + Z(Z^* Z)^{-1} dZ^*)$	$ Z^* Z  Z^H (Z^T Z^H)^{-1}$	$ Z^* Z  (Z^T Z^H)^{-1} Z^T$
$ ZZ^H $	$ ZZ^H  \text{tr}(Z^H(ZZ^H)^{-1} dZ + Z^T(Z^* Z^T)^{-1} dZ^*)$	$ ZZ^H (Z^* Z^T)^{-1} Z^*$	$ ZZ^H (ZZ^H)^{-1} Z$
$ Z^H Z $	$ Z^H Z  \text{tr}((Z^H Z)^{-1} Z^H dZ + (Z^T Z^*)^{-1} Z^T dZ^*)$	$ Z^H Z  Z^* (Z^T Z^*)^{-1}$	$ Z^H Z  Z(Z^H Z)^{-1}$
$ Z^k $	$k Z ^k \text{tr}(Z^{-1} dZ)$	$k Z ^k Z^{-T}$	$O$

### 3.5.2 矩阵函数的复梯度矩阵辨识

若  $\mathbf{f}(z, z^*) = [f_1(z, z^*), \dots, f_n(z, z^*)]^T$  是以  $m \times 1$  复向量  $z$  为变元的  $n \times 1$  复向量函数, 则有

$$\begin{aligned} \begin{bmatrix} df_1(z, z^*) \\ \vdots \\ df_n(z, z^*) \end{bmatrix} &= \begin{bmatrix} D_z f_1(z, z^*) \\ \vdots \\ D_z f_n(z, z^*) \end{bmatrix} dz + \begin{bmatrix} D_{z^*} f_1(z, z^*) \\ \vdots \\ D_{z^*} f_n(z, z^*) \end{bmatrix} dz^* \\ &= \begin{bmatrix} \frac{\partial f_1(z, z^*)}{\partial z_1} & \dots & \frac{\partial f_1(z, z^*)}{\partial z_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(z, z^*)}{\partial z_1} & \dots & \frac{\partial f_n(z, z^*)}{\partial z_m} \end{bmatrix} dz + \begin{bmatrix} \frac{\partial f_1(z, z^*)}{\partial z_1^*} & \dots & \frac{\partial f_1(z, z^*)}{\partial z_m^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(z, z^*)}{\partial z_1^*} & \dots & \frac{\partial f_n(z, z^*)}{\partial z_m^*} \end{bmatrix} dz^* \end{aligned}$$

或简记为

$$df(z, z^*) = D_z \mathbf{f}(z, z^*) dz + D_{z^*} \mathbf{f}(z, z^*) dz^* \quad (3.5.15)$$

式中  $df(z, z^*) = [df_1(z, z^*), \dots, df_n(z, z^*)]^T$ , 而

$$D_z \mathbf{f}(z, z^*) = \frac{\partial \mathbf{f}(z, z^*)}{\partial z^T} = \begin{bmatrix} \frac{\partial f_1(z, z^*)}{\partial z_1} & \dots & \frac{\partial f_1(z, z^*)}{\partial z_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(z, z^*)}{\partial z_1} & \dots & \frac{\partial f_n(z, z^*)}{\partial z_m} \end{bmatrix} \in \mathbb{C}^{n \times m} \quad (3.5.16)$$

$$D_{z^*} \mathbf{f}(z, z^*) = \frac{\partial \mathbf{f}(z, z^*)}{\partial z^H} = \begin{bmatrix} \frac{\partial f_1(z, z^*)}{\partial z_1^*} & \dots & \frac{\partial f_1(z, z^*)}{\partial z_m^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(z, z^*)}{\partial z_1^*} & \dots & \frac{\partial f_n(z, z^*)}{\partial z_m^*} \end{bmatrix} \in \mathbb{C}^{n \times m} \quad (3.5.17)$$

分别是复向量函数  $\mathbf{f}(\mathbf{z}, \mathbf{z}^*)$  相对于复向量变元  $\mathbf{z}$  和  $\mathbf{z}^*$  的 Jacobian 矩阵。

考查以  $m \times n$  复矩阵  $\mathbf{Z}$  为变元的  $p \times q$  矩阵函数  $\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)$ 。若记  $p \times q$  矩阵函数  $\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = [\mathbf{f}_1(\mathbf{Z}, \mathbf{Z}^*), \dots, \mathbf{f}_q(\mathbf{Z}, \mathbf{Z}^*)]$ , 则

$$d\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = [df_1(\mathbf{Z}, \mathbf{Z}^*), \dots, df_q(\mathbf{Z}, \mathbf{Z}^*)]$$

并且对于列向量函数  $\mathbf{f}_i(\mathbf{Z}, \mathbf{Z}^*), i = 1, \dots, q$ , 式 (3.5.15) 均成立。这意味着

$$\begin{bmatrix} df_1(\mathbf{Z}, \mathbf{Z}^*) \\ \vdots \\ df_q(\mathbf{Z}, \mathbf{Z}^*) \end{bmatrix} = \begin{bmatrix} D_{\text{vec}(\mathbf{Z})}\mathbf{f}_1(\mathbf{Z}, \mathbf{Z}^*) \\ \vdots \\ D_{\text{vec}(\mathbf{Z})}\mathbf{f}_q(\mathbf{Z}, \mathbf{Z}^*) \end{bmatrix} d(\text{vec}\mathbf{Z}) + \begin{bmatrix} D_{\text{vec}(\mathbf{Z}^*)}\mathbf{f}_1(\mathbf{Z}, \mathbf{Z}^*) \\ \vdots \\ D_{\text{vec}(\mathbf{Z}^*)}\mathbf{f}_q(\mathbf{Z}, \mathbf{Z}^*) \end{bmatrix} d(\text{vec}\mathbf{Z}^*) \quad (3.5.18)$$

式中

$$D_{\text{vec}(\mathbf{Z})}\mathbf{f}_i(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial \mathbf{f}_i(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}^T(\mathbf{Z})} \in \mathbb{C}^{p \times mn} \quad (3.5.19)$$

$$D_{\text{vec}(\mathbf{Z}^*)}\mathbf{f}_i(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial \mathbf{f}_i(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}^T(\mathbf{Z}^*)} \in \mathbb{C}^{p \times mn} \quad (3.5.20)$$

式 (3.5.18) 又可简写为

$$d(\text{vec}\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = \mathbf{A}d(\text{vec}\mathbf{Z}) + \mathbf{B}d(\text{vec}\mathbf{Z}^*) \in \mathbb{C}^{pq} \quad (3.5.21)$$

式中

$$d(\text{vec}\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = [df_{11}(\mathbf{Z}, \mathbf{Z}^*), \dots, df_{p1}(\mathbf{Z}, \mathbf{Z}^*), \dots, df_{1q}(\mathbf{Z}, \mathbf{Z}^*), \dots, df_{pq}(\mathbf{Z}, \mathbf{Z}^*)]^T$$

$$d(\text{vec}\mathbf{Z}) = [dZ_{11}, \dots, dZ_{m1}, \dots, dZ_{1n}, \dots, dZ_{mn}]^T$$

$$d(\text{vec}\mathbf{Z}^*) = [dZ_{11}^*, \dots, dZ_{m1}^*, \dots, dZ_{1n}^*, \dots, dZ_{mn}^*]^T$$

并且

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}} \end{bmatrix} \\ &= \frac{d\text{vec}\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)}{d(\text{vec}\mathbf{Z})^T} \\ &= D_{\text{vec}(\mathbf{Z})}\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) \end{aligned} \quad (3.5.22)$$

和

$$\begin{aligned}
 \mathbf{B} &= \begin{bmatrix} \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}^*} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}^*} & \dots & \frac{\partial f_{11}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}^*} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}^*} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}^*} & \dots & \frac{\partial f_{p1}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}^*} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}^*} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}^*} & \dots & \frac{\partial f_{1q}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}^*} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{11}^*} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{m1}^*} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{1n}^*} & \dots & \frac{\partial f_{pq}(\mathbf{Z}, \mathbf{Z}^*)}{\partial Z_{mn}^*} \end{bmatrix} \\
 &= \frac{\partial \text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec} \mathbf{Z}^*)^T} \\
 &= D_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) \tag{3.5.23}
 \end{aligned}$$

显然, 矩阵  $\mathbf{A}$  和  $\mathbf{B}$  分别是矩阵函数  $\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)$  相对于复矩阵变元的 Jacobian 矩阵和共轭 Jacobian 矩阵。

矩阵函数  $\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)$  相对于复矩阵变元的梯度矩阵和共轭梯度矩阵分别定义为

$$\nabla_{\text{vec}(\mathbf{Z})} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}} = (D_{\text{vec}(\mathbf{Z})} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T \tag{3.5.24}$$

$$\nabla_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \frac{\partial (\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T}{\partial \text{vec} \mathbf{Z}^*} = (D_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*))^T \tag{3.5.25}$$

特别地, 对于实标量函数  $f(\mathbf{Z}, \mathbf{Z}^*)$ , 式 (3.5.21) 简化为式 (3.4.33)。

综合以上讨论, 由式 (3.5.21) 可得以下命题。

**命题 3.5.2** 对于复变矩阵函数  $\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) \in \mathbb{C}^{p \times q}$  (其中  $\mathbf{Z}, \mathbf{Z}^* \in \mathbb{C}^{m \times n}$ ), 其 Jacobian 矩阵和共轭 Jacobian 矩阵可以辨识如下

$$d(\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = \mathbf{Ad}(\text{vec} \mathbf{Z}) + \mathbf{Bd}(\text{vec} \mathbf{Z}^*) \Leftrightarrow \begin{cases} D_{\text{vec}(\mathbf{Z})} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{A} \\ D_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{B} \end{cases} \tag{3.5.26}$$

或梯度矩阵和共轭梯度矩阵的辨识公式为

$$d(\text{vec} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = \mathbf{Ad}(\text{vec} \mathbf{Z}) + \mathbf{Bd}(\text{vec} \mathbf{Z}^*) \Leftrightarrow \begin{cases} \nabla_{\text{vec}(\mathbf{Z})} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{A}^T \\ \nabla_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{B}^T \end{cases} \tag{3.5.27}$$

若

$$d(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = \mathbf{A}(d\mathbf{Z})\mathbf{B} + \mathbf{C}(d\mathbf{Z}^*)\mathbf{D}$$

则有向量化

$$d \text{vec}(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = (\mathbf{B}^T \otimes \mathbf{A})d(\text{vec} \mathbf{Z}) + (\mathbf{D}^T \otimes \mathbf{C})d(\text{vec} \mathbf{Z}^*)$$

由命题 3.5.2 得以下辨识公式

$$d(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) = \mathbf{A}(d\mathbf{Z})\mathbf{B} + \mathbf{C}(d\mathbf{Z}^*)\mathbf{D} \Leftrightarrow \begin{cases} D_{\text{vec}(\mathbf{Z})} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{B}^T \otimes \mathbf{A} \\ D_{\text{vec}(\mathbf{Z}^*)} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = \mathbf{D}^T \otimes \mathbf{C} \end{cases} \tag{3.5.28}$$

类似地, 若

$$d(F(Z, Z^*)) = A(dZ)^T B + C(dZ^*)^T D$$

则有向量化

$$\begin{aligned} d \operatorname{vec}(F(Z, Z^*)) &= (B^T \otimes A)d(\operatorname{vec}Z^T) + (D^T \otimes C)d(\operatorname{vec}Z^H) \\ &= (B^T \otimes A)K_{mn}d(\operatorname{vec}Z) + (D^T \otimes C)K_{mn}d(\operatorname{vec}Z^*) \end{aligned}$$

式中利用了向量化性质  $\operatorname{vec}(X_{m \times n}) = K_{mn}\operatorname{vec}(X)$ 。由命题 3.5.2 即得辨识公式

$$\begin{aligned} d(F(Z, Z^*)) &= A(dZ)^T B + C(dZ^*)^T D \\ \Leftrightarrow \begin{cases} D_{\operatorname{vec}(Z)} F(Z, Z^*) = (B^T \otimes A)K_{mn} \\ D_{\operatorname{vec}(Z^*)} F(Z, Z^*) = (D^T \otimes C)K_{mn} \end{cases} \end{aligned} \quad (3.5.29)$$

上式表明, 矩阵函数  $F(Z, Z^*)$  的梯度矩阵和共轭梯度矩阵的辨识的关键在于将矩阵函数的矩阵微分表示成规范形式  $d(F(Z, Z^*)) = A(dZ)^T B + C(dZ^*)^T D$ 。

表 3.5.3 总结了一阶复矩阵微分与 Jacobian 矩阵的对应关系, 其中  $z \in \mathbb{C}^m, Z \in \mathbb{C}^{m \times n}, F \in \mathbb{C}^{p \times q}$ 。

表 3.5.3 一阶复矩阵微分与 Jacobian 矩阵的对应关系

函数	一阶复矩阵微分规范形式	Jacobian 矩阵
$f(z, z^*)$	$df(z, z^*) = adz + bdz^*$	$\frac{\partial f}{\partial z} = a, \frac{\partial f}{\partial z^*} = b$
$f(z, z^*)$	$df(z, z^*) = a^T dz + b^T dz^*$	$\frac{\partial f}{\partial z^T} = a^T, \frac{\partial f}{\partial z^H} = b^T$
$f(Z, Z^*)$	$df(Z, Z^*) = \operatorname{tr}(AdZ + BdZ^*)$	$\frac{\partial f}{\partial Z^T} = A, \frac{\partial f}{\partial Z^H} = B$
$F(Z, Z^*)$	$d \operatorname{vec}F = Ad \operatorname{vec}Z + Bd \operatorname{vec}Z^*$	$\frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z)^T} = A, \frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z^*)^T} = B$
	$dF = A(dZ)B + C(dZ^*)D$	$\frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z)^T} = B^T \otimes A, \frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z^*)^T} = D^T \otimes C$
	$dF = A(dZ)^T B + C(dZ^*)^T D$	$\frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z)^T} = (B^T \otimes A)K_{mn}, \frac{\partial \operatorname{vec}F}{\partial (\operatorname{vec}Z^*)^T} = (D^T \otimes C)K_{mn}$

### 3.5.3 复 Hessian 矩阵辨识

由命题 3.5.1 知, 实值标量函数  $f(Z, Z^*)$  的微分可以写作规范形式  $df(Z, Z^*) = \operatorname{tr}(AdZ + A^*dZ^*)$ , 其中  $A = A(Z, Z^*)$  通常是变元矩阵  $Z, Z^*$  的复矩阵函数。于是,  $A$  的微分矩阵可以写作

$$dA = C(dZ)D + E(dZ^*)F \quad (3.5.30)$$

或者

$$dA = D(dZ)^T C + F(dZ^*)^T E \quad (3.5.31)$$

将式(3.5.30)代入到二阶微分

$$d^2 f(\mathbf{Z}, \mathbf{Z}^*) = d(df(\mathbf{Z}, \mathbf{Z}^*)) = \text{tr}(d\mathbf{A}d\mathbf{Z} + d\mathbf{A}^*d\mathbf{Z}^*) \quad (3.5.32)$$

易得

$$\begin{aligned} d^2 f(\mathbf{Z}, \mathbf{Z}^*) &= \text{tr}(\mathbf{C}(d\mathbf{Z})\mathbf{D}d\mathbf{Z}) + \text{tr}(\mathbf{E}(d\mathbf{Z}^*)\mathbf{F}d\mathbf{Z}) \\ &\quad + \text{tr}(\mathbf{C}^*(d\mathbf{Z}^*)\mathbf{D}^*d\mathbf{Z}^*) + \text{tr}(\mathbf{E}^*(d\mathbf{Z})\mathbf{F}^*d\mathbf{Z}^*) \end{aligned} \quad (3.5.33)$$

利用迹函数的性质  $\text{tr}(\mathbf{XYUV}) = (\text{vec}\mathbf{V}^T)^T(\mathbf{U}^T \otimes \mathbf{X})\text{vec}\mathbf{Y}$ 、向量化性质  $\text{vec}\mathbf{Z}^T = \mathbf{K}_{mn}\text{vec}\mathbf{Z}$  以及交换矩阵性质  $\mathbf{K}_{mn}^T = \mathbf{K}_{nm}$ , 易知

$$\begin{aligned} \text{tr}(\mathbf{C}(d\mathbf{Z})\mathbf{D}d\mathbf{Z}) &= (\mathbf{K}_{mn}\text{vec } d\mathbf{Z})^T(\mathbf{D}^T \otimes \mathbf{C})\text{vec } d\mathbf{Z} \\ &= (d\text{ vec}\mathbf{Z})^T \mathbf{K}_{nm}(\mathbf{D}^T \otimes \mathbf{C})d\text{ vec}\mathbf{Z} \end{aligned}$$

类似地, 有

$$\begin{aligned} \text{tr}(\mathbf{E}(d\mathbf{Z}^*)\mathbf{F}d\mathbf{Z}) &= (d\text{ vec}\mathbf{Z})^T \mathbf{K}_{nm}(\mathbf{F}^T \otimes \mathbf{E})d\text{ vec}\mathbf{Z}^* \\ \text{tr}(\mathbf{E}^*(d\mathbf{Z})\mathbf{F}^*d\mathbf{Z}^*) &= (d\text{ vec}\mathbf{Z}^*)^T \mathbf{K}_{nm}(\mathbf{F}^H \otimes \mathbf{E}^*)d\text{ vec}\mathbf{Z} \\ \text{tr}(\mathbf{C}^*(d\mathbf{Z}^*)\mathbf{D}^*d\mathbf{Z}^*) &= (d\text{ vec}\mathbf{Z}^*)^T \mathbf{K}_{nm}(\mathbf{D}^H \otimes \mathbf{C}^*)d\text{ vec}\mathbf{Z}^* \end{aligned}$$

于是, 函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的二阶微分公式(3.5.33)可以改写为

$$\begin{aligned} d^2 f(\mathbf{Z}, \mathbf{Z}^*) &= (d\text{ vec}\mathbf{Z})^T \mathbf{K}_{nm}(\mathbf{D}^T \otimes \mathbf{C})d\text{ vec}\mathbf{Z} \\ &\quad + (d\text{ vec}\mathbf{Z})^T \mathbf{K}_{nm}(\mathbf{F}^T \otimes \mathbf{E})d\text{ vec}\mathbf{Z}^* \\ &\quad + (d\text{ vec}\mathbf{Z}^*)^T \mathbf{K}_{nm}(\mathbf{F}^H \otimes \mathbf{E}^*)d\text{ vec}\mathbf{Z} \\ &\quad + (d\text{ vec}\mathbf{Z}^*)^T \mathbf{K}_{nm}(\mathbf{D}^H \otimes \mathbf{C}^*)d\text{ vec}\mathbf{Z} \end{aligned} \quad (3.5.34)$$

另外, 将式(3.5.31)代入到二阶微分公式(3.5.32)中, 则有

$$\begin{aligned} d^2 f(\mathbf{Z}, \mathbf{Z}^*) &= \text{tr}(\mathbf{C}(d\mathbf{Z})\mathbf{D}(d\mathbf{Z})^T) + \text{tr}(\mathbf{E}(d\mathbf{Z})\mathbf{F}(d\mathbf{Z}^*)^T) \\ &\quad + \text{tr}(\mathbf{E}^*(d\mathbf{Z}^*)\mathbf{F}^*(d\mathbf{Z})^T) + \text{tr}(\mathbf{C}^*(d\mathbf{Z}^*)\mathbf{D}^*(d\mathbf{Z}^*)^T) \end{aligned} \quad (3.5.35)$$

再次利用迹函数的性质  $\text{tr}(\mathbf{XYUV}) = (\text{vec}\mathbf{V}^T)^T(\mathbf{U}^T \otimes \mathbf{X})\text{vec}\mathbf{Y}$  易知, 函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的二阶微分公式(3.5.34)可以改写为

$$\begin{aligned} d^2 f(\mathbf{Z}, \mathbf{Z}^*) &= (d\text{ vec}\mathbf{Z})^T(\mathbf{D}^T \otimes \mathbf{C})d\text{ vec}\mathbf{Z} \\ &\quad + (d\text{ vec}\mathbf{Z}^*)^T(\mathbf{F}^T \otimes \mathbf{E})d\text{ vec}\mathbf{Z}^* \\ &\quad + (d\text{ vec}\mathbf{Z})^T(\mathbf{F}^H \otimes \mathbf{E}^*)d\text{ vec}\mathbf{Z}^* \\ &\quad + (d\text{ vec}\mathbf{Z}^*)^T(\mathbf{D}^H \otimes \mathbf{C}^*)d\text{ vec}\mathbf{Z} \end{aligned} \quad (3.5.36)$$

综合式(3.5.36)、式(3.5.35)和式(3.4.53),即得复Hessian矩阵的辨识定理如下。

**定理3.5.1** 令 $f(\mathbf{Z}, \mathbf{Z}^*)$ 是 $m \times n$ 变元矩阵 $\mathbf{Z}, \mathbf{Z}^*$ 的二次可微分的实值函数,则其二阶微分和复Hessian矩阵之间存在下列对应关系

$$\text{tr}(\mathbf{C}(\mathbf{d}\mathbf{Z})\mathbf{D}(\mathbf{d}\mathbf{Z})^T) \Leftrightarrow \mathbf{H}_{\mathbf{Z}, \mathbf{Z}} = \frac{1}{2}(\mathbf{D}^T \otimes \mathbf{C} + \mathbf{D} \otimes \mathbf{C}^T) \quad (3.5.37)$$

$$\text{tr}(\mathbf{E}(\mathbf{d}\mathbf{Z})\mathbf{F}(\mathbf{d}\mathbf{Z}^*)^T) \Leftrightarrow \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*} = \frac{1}{2}(\mathbf{F}^T \otimes \mathbf{E} + \mathbf{F}^* \otimes \mathbf{E}^H) \quad (3.5.38)$$

和

$$\text{tr}(\mathbf{C}(\mathbf{d}\mathbf{Z})\mathbf{D}\mathbf{d}\mathbf{Z}) \Leftrightarrow \mathbf{H}_{\mathbf{Z}, \mathbf{Z}} = \frac{1}{2}\mathbf{K}_{nm}(\mathbf{D}^T \otimes \mathbf{C} + \mathbf{C}^T \otimes \mathbf{D}) \quad (3.5.39)$$

$$\text{tr}(\mathbf{E}(\mathbf{d}\mathbf{Z})\mathbf{F}\mathbf{d}\mathbf{Z}^*) \Leftrightarrow \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*} = \frac{1}{2}\mathbf{K}_{nm}(\mathbf{F}^T \otimes \mathbf{E} + \mathbf{E}^H \otimes \mathbf{F}^*) \quad (3.5.40)$$

在得到部分Hessian矩阵 $\mathbf{H}_{\mathbf{Z}, \mathbf{Z}}$ 和 $\mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*}$ 后,即可分别得到另外两个部分Hessian矩阵 $\mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}^*} = \mathbf{H}_{\mathbf{Z}, \mathbf{Z}}^*$ 和 $\mathbf{H}_{\mathbf{Z}^*, \mathbf{Z}} = \mathbf{H}_{\mathbf{Z}, \mathbf{Z}^*}^*$ 。

共轭梯度矩阵与全Hessian矩阵在以复矩阵为变元的目标函数的最优化算法中起着重要的作用,第4章将具体介绍共轭梯度矩阵与全Hessian矩阵在最优化中的应用。

## 本章小结

本章首先针对实矩阵微分,依次介绍了实函数相对于实矩阵变元的Jacobian矩阵和梯度矩阵、一阶实矩阵微分与Jacobian矩阵辨识、二阶实矩阵微分与Hessian矩阵辨识。然后,重点讨论了复矩阵微分,分别介绍了标量函数相对于复矩阵变元的共轭梯度与复Hessian矩阵、复梯度矩阵与复Hessian矩阵的辨识。矩阵微分是解决梯度矩阵辨识和Hessian矩阵辨识的有效数学工具。如第4章所述,梯度矩阵和Hessian矩阵又是最优化算法设计的关键数学工具。

## 习题

**3.1 证明:**若 $\phi(\mathbf{x}) = (\mathbf{f}(\mathbf{x}))^T \mathbf{g}(\mathbf{x})$ ,则

$$\mathbf{D}_{\mathbf{x}}\phi(\mathbf{x}) = (\mathbf{g}(\mathbf{x}))^T \mathbf{D}_{\mathbf{x}}\mathbf{f}(\mathbf{x}) + (\mathbf{f}(\mathbf{x}))^T \mathbf{D}_{\mathbf{x}}\mathbf{g}(\mathbf{x})$$

**3.2 证明:**若 $\phi(\mathbf{x}) = (\mathbf{f}(\mathbf{x}))^T \mathbf{A}\mathbf{g}(\mathbf{x})$ ,则

$$\mathbf{D}_{\mathbf{x}}\phi(\mathbf{x}) = (\mathbf{g}(\mathbf{x}))^T \mathbf{A}^T \mathbf{D}_{\mathbf{x}}\mathbf{f}(\mathbf{x}) + (\mathbf{f}(\mathbf{x}))^T \mathbf{A} \mathbf{D}_{\mathbf{x}}\mathbf{g}(\mathbf{x})$$

**3.3 证明**Kronecker积的矩阵微分

$$\mathbf{d}(\mathbf{X} \otimes \mathbf{Y}) = (\mathbf{d}\mathbf{X})\mathbf{Y} + \mathbf{X} \otimes \mathbf{d}\mathbf{Y}$$

### 3.4 证明

$$d(\mathbf{X} * \mathbf{Y}) = (d\mathbf{X})\mathbf{Y} + \mathbf{X} * d\mathbf{X}$$

其中  $\mathbf{X} * \mathbf{Y}$  表示  $\mathbf{X}$  与  $\mathbf{Y}$  的 Hadamard 积。

**3.5** 令实值实标量函数  $f(\mathbf{X}) = \mathbf{a}^T \mathbf{X} \mathbf{X}^T \mathbf{b}$ , 其中  $\mathbf{X} \in \mathbb{R}^{m \times n}, \mathbf{a}, \mathbf{b} \in \mathbb{R}^{n \times 1}$ 。利用矩阵变元的元素之间的独立性假设, 证明

$$\frac{\partial \mathbf{a}^T \mathbf{X}^T \mathbf{X} \mathbf{b}}{\partial \mathbf{X}} = \mathbf{X} (\mathbf{a} \mathbf{b}^T + \mathbf{b} \mathbf{a}^T)$$

### 3.6 证明

$$d(\mathbf{U} \mathbf{V} \mathbf{W}) = (d\mathbf{U}) \mathbf{V} \mathbf{W} + \mathbf{U} (d\mathbf{V}) \mathbf{W} + \mathbf{U} \mathbf{V} (d\mathbf{W})$$

### 3.7 证明

$$d[\text{tr}(\mathbf{X}^T \mathbf{X})] = 2\text{tr}(\mathbf{X}^T d\mathbf{X})$$

**3.8** 求迹函数  $\text{tr}[(\mathbf{X}^T \mathbf{C} \mathbf{X})^{-1} \mathbf{A}]$  的微分矩阵与梯度矩阵。

### 3.9 证明

$$\begin{aligned} & \frac{\partial \text{tr}[(\mathbf{X}^T \mathbf{C} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{B} \mathbf{X})]}{\partial \mathbf{X}} \\ &= -(\mathbf{C} + \mathbf{C}^T) \mathbf{X} (\mathbf{X}^T \mathbf{C}^T \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{B}^T \mathbf{X}) (\mathbf{X}^T \mathbf{C}^T \mathbf{X})^{-1} \\ &+ (\mathbf{B} + \mathbf{B}^T) \mathbf{X} (\mathbf{X}^T \mathbf{C}^T \mathbf{X})^{-1} \end{aligned}$$

**3.10** 求迹函数  $\text{tr}[(\mathbf{A} + \mathbf{X}^T \mathbf{C} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{B} \mathbf{X})]$  的微分矩阵和梯度矩阵。

**3.11** 求实标量函数  $f(\mathbf{x}) = \mathbf{a}^T \mathbf{x}$  和  $f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x}$  的 Hessian 矩阵。

**3.12** 证明实标量函数  $f(\mathbf{X}) = \text{tr}(\mathbf{A} \mathbf{X} \mathbf{B} \mathbf{X}^T)$  的 Hessian 矩阵为

$$\mathbf{H}[f(\mathbf{X})] = \mathbf{B}^T \otimes \mathbf{A} + \mathbf{B} \otimes \mathbf{A}^T$$

**3.13** 求行列式对数  $\log |\mathbf{X}^T \mathbf{A} \mathbf{X}|, \log |\mathbf{X} \mathbf{A} \mathbf{X}^T|$  和  $\log |\mathbf{X} \mathbf{A} \mathbf{X}|$  的 Hessian 矩阵。

**3.14** 求矩阵函数  $\mathbf{A} \mathbf{X} \mathbf{B}$  和  $\mathbf{A} \mathbf{X}^{-1} \mathbf{B}$  的 Jacobian 矩阵。

**3.15** 求下列迹函数的梯度矩阵:

(1)  $\text{tr}(\mathbf{A} \mathbf{X}^{-1} \mathbf{B})$ 。

(2)  $\text{tr}(\mathbf{A} \mathbf{X}^T \mathbf{B} \mathbf{X} \mathbf{C})$ 。

**3.16** 求行列式  $|\mathbf{X}^T \mathbf{A} \mathbf{X}|, |\mathbf{X} \mathbf{A} \mathbf{X}^T|, |\mathbf{X} \mathbf{A} \mathbf{X}|$  和  $|\mathbf{X}^T \mathbf{A} \mathbf{X}^T|$  的梯度矩阵。

**3.17** 求行列式对数  $\log |\mathbf{X}^T \mathbf{A} \mathbf{X}|, \log |\mathbf{X} \mathbf{A} \mathbf{X}^T|$  和  $\log |\mathbf{X} \mathbf{A} \mathbf{X}|$  的 Jacobian 矩阵与梯度矩阵。

**3.18** 证明: 若  $\mathbf{F}$  是矩阵函数, 并且可二次微分, 则

$$d^2 \log |\mathbf{F}| = -\text{tr}(\mathbf{F}^{-1} d\mathbf{F})^2 + \text{tr}(\mathbf{F}^{-1}) d^2 \mathbf{F}$$

**3.19** 通过  $d(\mathbf{X}^{-1} \mathbf{X})$  求逆矩阵的微分  $d\mathbf{X}^{-1}$ 。

**3.20** 证明: 对于非奇异矩阵  $\mathbf{X} \in \mathbb{R}^{n \times n}$ , 其逆矩阵  $\mathbf{X}^{-1}$  是无穷次可微分的, 并且其  $r$  次微分

$$d^{(r)}(\mathbf{X}^{-1}) = (-1)^r r! (\mathbf{X}^{-1} d\mathbf{X})^r \mathbf{X}^{-1}, \quad r = 1, 2, \dots$$

**3.21** 证明:  $m \times n$  实矩阵  $\mathbf{X}$  的 Moore-Penrose 广义逆的微分为

$$\begin{aligned} d(\mathbf{X}^\dagger) &= -\mathbf{X}^\dagger(d\mathbf{X})\mathbf{X}^\dagger + \mathbf{X}^\dagger(\mathbf{X}^\dagger)^T(d\mathbf{X}^T)(\mathbf{I}_m - \mathbf{Z}\mathbf{Z}^\dagger) \\ &\quad + (\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X})(d\mathbf{X}^T)(\mathbf{X}^\dagger)^T\mathbf{X}^\dagger \end{aligned}$$

**3.22** 令  $\mathbf{X} \in \mathbb{R}^{m \times n}$ , 且  $\mathbf{X}^\dagger$  是  $\mathbf{X}$  的 Moore-Penrose 广义逆。证明:

$$d(\mathbf{X}^\dagger\mathbf{X}) = \mathbf{X}^\dagger(d\mathbf{X})(\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X}) + [\mathbf{X}^\dagger(d\mathbf{X})(\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X})]^T$$

和

$$d(\mathbf{X}\mathbf{X}^\dagger) = (\mathbf{I}_m - \mathbf{X}\mathbf{X}^\dagger)(d\mathbf{X})\mathbf{X}^\dagger + [(\mathbf{I}_m - \mathbf{X}\mathbf{X}^\dagger)(d\mathbf{X})\mathbf{X}^\dagger]^T$$

提示: 矩阵  $\mathbf{X}^\dagger\mathbf{X}$  和  $\mathbf{X}\mathbf{X}^\dagger$  均是幂等矩阵和对称矩阵。

**3.23** 求下列实值函数的 Hessian 矩阵:

- (1)  $(\mathbf{A}\mathbf{x} + \mathbf{b})^T(\mathbf{D}\mathbf{x} + \mathbf{e})$ 。
- (2)  $(\mathbf{A}\mathbf{x} + \mathbf{b})^T\mathbf{C}(\mathbf{D}\mathbf{x} + \mathbf{e})$ 。
- (3)  $(\mathbf{A}\mathbf{x} + \mathbf{b})^T\mathbf{C}(\mathbf{A}\mathbf{x} + \mathbf{b})$ 。

**3.24** 若  $\mathbf{X} \in \mathbb{R}^{n \times n}$ , 证明: 迹函数  $\text{tr}(\mathbf{X}^2)$  的 Hessian 矩阵

$$\frac{\partial^2 \text{tr}(\mathbf{X}^2)}{\partial \text{vec} \mathbf{X} \partial (\text{vec} \mathbf{X})^T} = 2\mathbf{K}_{nn}$$

**3.25** 证明

$$\begin{aligned} d(\mathbf{F}(\mathbf{Z}, \mathbf{Z}^*)) &= \mathbf{A}(d\mathbf{Z})^T\mathbf{B} + \mathbf{C}(d\mathbf{Z}^*)^T\mathbf{D} \\ \iff & \begin{cases} \mathbf{D}_\mathbf{Z} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = (\mathbf{B}^T \otimes \mathbf{A})\mathbf{K}_{mn} \\ \mathbf{D}_{\mathbf{Z}^*} \mathbf{F}(\mathbf{Z}, \mathbf{Z}^*) = (\mathbf{D}^T \otimes \mathbf{A})\mathbf{K}_{mn} \end{cases} \end{aligned}$$

**3.26** 证明全 Hessian 矩阵为 Hermitian 矩阵, 即  $\mathbf{H}^H = \mathbf{H}$ 。

**3.27** 证明部分 Hessian 矩阵的以下性质:

$$\mathbf{H}_{\mathbf{z}^*, \mathbf{z}} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}}^H, \quad \mathbf{H}_{\mathbf{z}, \mathbf{z}^*} = \mathbf{H}_{\mathbf{z}, \mathbf{z}^*}^H \quad \text{和} \quad \mathbf{H}_{\mathbf{z}^*, \mathbf{z}} = \mathbf{H}_{\mathbf{z}, \mathbf{z}^*}^*$$

和

$$\mathbf{H}_{\mathbf{z}, \mathbf{z}} = \mathbf{H}_{\mathbf{z}, \mathbf{z}}^T, \quad \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*}^T \quad \text{和} \quad \mathbf{H}_{\mathbf{z}, \mathbf{z}} = \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*}^*$$

**3.28** 令  $\mathbf{x}$  为复向量, 求下列实值函数的复 Hessian 矩阵:

- (1)  $(\mathbf{A}\mathbf{x} + \mathbf{b})^H(\mathbf{D}\mathbf{x} + \mathbf{e})$ 。
- (2)  $(\mathbf{A}\mathbf{x} + \mathbf{b})^H\mathbf{C}(\mathbf{D}\mathbf{x} + \mathbf{e})$ 。

(3)  $(Ax + b)^H C(Ax + b)$ 。

**3.29** 令  $z_1, z_2 \in \mathbb{C}^n$ , 并且  $A \in \mathbb{C}^{n \times n}$ 。若  $f(z) = z_1^H A z_2$ , 其中  $z = [z_1, z_2]^T$ , 试求

(1) 复偏导向量  $D_{z_1} f(z) = \frac{\partial f(z)}{\partial z_1^T}$  和共轭偏导向量  $D_{z_1^*} f(z) = \frac{\partial f(z)}{\partial z_2^H}$ ;

(2) 复偏导向量  $D_{z_2} f(z) = \frac{\partial f(z)}{\partial z_2^T}$  和共轭偏导向量  $D_{z_2^*} f(z) = \frac{\partial f(z)}{\partial z_2^H}$ ;

(3) 复偏导向量  $D_z f(z) = \frac{\partial f(z)}{\partial z^T}$  和共轭偏导向量  $D_{z^*} f(z) = \frac{\partial f(z)}{\partial z^H}$ 。

**3.30** 证明

$$\text{tr}(C(dZ)DdZ^T) = (\text{d vec } Z)^T (D^T \otimes C) \text{d vec } Z$$

$$\text{tr}(E(dZ^*)FdZ^T) = (\text{d vec } Z)^T (F^T \otimes E) \text{d vec } Z^*$$

$$\text{tr}(E^*(dZ)F^*dZ^H) = (\text{d vec } Z^*)^T (F^H \otimes E^*) \text{d vec } Z$$

$$\text{tr}(C^*(dZ^*)D^*dZ^H) = (\text{d vec } Z^*)^T (D^H \otimes C^*) \text{d vec } Z^*$$

## 第4章 梯度分析与最优化

最优化理论主要研究一个函数的极值：极大值或极小值。这一函数称为最优化问题的目标函数，通常是实向量或实矩阵变元的某个实值函数，但在很多工程应用中往往是复向量或复矩阵变元的实值函数。最优化理论主要讨论：(1) 极值的存在性条件 (梯度分析)；(2) 优化算法的设计及收敛性分析。

本章将主要从梯度分析的角度，讨论全局最优化和非线性最优化的一般理论；而优化方法的介绍重点则是凸优化和内点法。

### 4.1 实变函数无约束优化的梯度分析

考虑典型的最优化问题

$$\min_{\mathbf{x} \in \mathcal{D}} f(\mathbf{x}) \quad (4.1.1)$$

其中， $\mathcal{D} = \text{dom } f(\mathbf{x})$  表示函数  $f(\mathbf{x})$  的定义域；变元向量  $\mathbf{x} \in \mathbb{R}^n$  称为最优化问题的优化向量，代表需要作出的一种选择；函数  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  称为目标函数 (objective function)，表示选择优化向量  $\mathbf{x}$  时所付出的成本或者代价，故又常称为代价函数 (cost function)。相反，代价函数的负值  $-f(\mathbf{x})$  则可理解成选择  $\mathbf{x}$  所得到的价值 (value) 或者效益 (utility)。于是，最优化问题式 (4.1.1) 的求解对应于使代价最小化或者使效益最大化。因此，极小化问题  $\min_{\mathbf{x} \in \mathcal{D}} f(\mathbf{x})$  与负目标函数的极大化问题  $\max_{\mathbf{x} \in \mathcal{D}} -f(\mathbf{x})$  二者等价。

上述优化问题没有约束条件，故称为无约束优化问题。求解无约束优化问题的大多数非线性规划方法都是基于松弛和逼近的思想 [363]。

**松弛** 称序列  $\{a_k\}_{k=0}^{\infty}$  为松弛序列 (relaxation sequence)，若  $a_{k+1} \leq a_k, \forall k \geq 0$ 。因此，在迭代求解最优化问题式 (4.1.1) 的过程中，需要产生一个松弛序列

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k), \quad k = 0, 1, \dots$$

**逼近** 逼近一个目标函数意味着，使用一个接近原始目标的简化目标函数代替原目标函数。

于是，利用松弛和逼近，可以实现以下目的：

- (1) 如果目标函数  $f(\mathbf{x})$  在实数域  $\mathbb{R}^n$  是下有界的，则序列  $\{f(\mathbf{x}_k)\}_{k=0}^{\infty}$  一定收敛。
- (2) 在任何情况下，都可以改善目标函数  $f(\mathbf{x})$  的初始值。
- (3) 一个非线性目标函数  $f(\mathbf{x})$  的极小化可以用数值方法实现，并且逼近精度足够高。

### 4.1.1 单变量函数 $f(x)$ 的平稳点与极值点

目标函数的平稳点和极值点在最优化问题中起着关键作用。平稳点的分析依赖于目标函数的梯度向量（一阶梯度），而极值点的分析则取决于目标函数的 Hessian 矩阵（二阶梯度）。因此，目标函数的梯度分析分为一阶梯度分析（平稳点分析）和二阶梯度分析（极值点分析）。

在最优化中，通常希望得到目标函数的全局极小点，函数  $f(x)$  在该点取最小值。

**定义 4.1.1** 定义域  $\mathcal{D}$  中的点  $x^*$  称为函数  $f(x)$  的全局极小点（global minimum point），若

$$f(x^*) \leq f(x), \quad \forall x \in \mathcal{D}, x \neq x^* \quad (4.1.2)$$

全局极小点也称绝对极小点（absolute minimum point），函数在该点的取值  $f(x_0)$  称为函数  $f(x)$  在定义域  $\mathcal{D}$  中的全局极小值（global minimum）或绝对极小值（absolute minimum）。

若

$$f(x^*) < f(x), \quad \forall x \in \mathcal{D} \quad (4.1.3)$$

则称  $x^*$  是函数  $f(x)$  的严格全局极小点（strict global minimum point）或严格绝对极小点（strict absolute minimum point）。

毋庸待言，最小化的理想目标是求出全局极小点。然而，这一理想目标却往往很难实现，其原因是：(1) 通常很难知道一个函数  $f(x)$  在其整个定义域  $\mathcal{D}$  的全局或者整体信息；(2) 设计一种识别全局极值点的算法往往不切实际，因为将  $f(x^*)$  的值与函数在整个定义域  $\mathcal{D}$  内的所有取值  $f(x)$  进行逐一比较往往是困难的。相比之下，了解目标函数  $f(x)$  在某点  $c$  附近区域的局部信息却要容易得多；并且设计一种算法，对某点的函数值  $f(c)$  与该点附近的函数值进行比较，也要简单得多。因此，大多数的最小化算法只能求出局部极小点：函数在该点的取值达到函数在该点邻域所有取值的最小值。

**定义 4.1.2** 给定一个点  $c \in \mathcal{D}$  和正数  $r$ ，则满足  $|x - c| < r$  的所有点  $x$  的集合称为点  $c$  以  $r$  为半径的（开）邻域，记为  $B_o(c; r) = \{x | x \in \mathcal{D}, |x - c| < r\}$ 。若  $B_c(c; r) = \{x | x \in \mathcal{D}, |x - c| \leq r\}$ ，则称其为闭邻域。

邻域  $B(c; r)$  有时也简记为  $B(c)$ 。

**定义 4.1.3** 点  $c$  称为函数  $f(x)$  的一个局部极小（或极大）点，若  $f(c) \leq f(c + \Delta x)$ （或  $f(c) \geq f(c + \Delta x)$ ）对满足  $0 < |\Delta x| \leq r$  的所有  $\Delta x$  均成立。

函数  $f(x)$  在局部极小点和局部极大点  $c$  的取值  $f(c)$  分别称为  $f(x)$  在定义域  $\mathcal{D}$  内的局部极小值（local minimum）和局部极大值（local maximum）。

**定义 4.1.4** 点  $c$  称为函数  $f(x)$  的严格局部极小点（strictly local minimum point），若  $f(c) < f(c + \Delta x)$  对所有满足  $0 < |\Delta x| \leq r$  的  $\Delta x$  均成立。

一个函数的极小点和极大点合称该函数的极值点（extreme point），而极小值和极大值合称该函数的极值（extremum 或 extreme value）。

局部极小点和严格局部极小点有时也分别称为函数  $f(x)$  的弱局部极小点 (weak local minimum point) 和强局部极小点 (strong local minimum point)。

例如, 对于常数函数  $f(x) = 3$ , 每一个点  $x$  都是一个 (弱) 局部极小点; 而函数  $f(x) = (x - 3)^2$  在  $x = 3$  有一个严格局部极小点。

特别地, 如果某个点  $x_0$  是函数  $f(x)$  在邻域  $B(c; r)$  内的唯一局部极值点, 则称为孤立局部极值点 (isolated local extremum)。

在实际应用中, 直接比较一个目标函数  $f(x)$  在某点及其邻域的所有取值仍然显得很麻烦。幸好, 函数的 Taylor 级数展开为解决这个问题提供了一种简单的方法。

如果函数  $f(x)$  具有连续的各阶导数, 则  $f(x)$  在  $c$  点的 Taylor 级数展开为

$$f(c + \Delta x) = f(c) + f'(c)\Delta x + \frac{1}{2}f''(c)(\Delta x)^2 + \cdots + \frac{1}{k!}f^k(c)(\Delta x)^k + \cdots \quad (4.1.4)$$

式中,  $f^k(c) = f^k(x)|_{x=c}$ , 而  $f^k(x) = \frac{d^k f(x)}{dx^k}$ ,  $k = 1, 2, 3, \dots$  是函数  $f(x)$  的  $k$  阶导数。

当半径  $r$  足够小时, 在邻域  $B(c; r)$  内, 高次项  $(\Delta x)^k$ ,  $k \geq 3$  常可忽略。于是, 函数  $f(x)$  在点  $c$  的邻域内可以用二阶 Taylor 级数展开

$$f(c + \Delta x) \approx f(c) + f'(c)\Delta x + \frac{1}{2}f''(c)(\Delta x)^2 \quad (4.1.5)$$

进行逼近。

首先, 将邻域  $B(c; r)$  缩小为一个非常小的区域  $|\Delta x| < \varepsilon$ , 其中  $\varepsilon$  足够小, 以至于二次项  $(\Delta x)^2$  也可以忽略不计。此时, 有函数的一阶逼近  $f(c + \Delta x) \approx f(c) + f'(c)\Delta x$ 。显然, 如果  $f'(c) > 0$ , 则  $f(c) \leq f(c + \Delta x)$  只有对  $\Delta x > 0$  成立。反之, 若  $f'(c) < 0$ , 则  $f(c) < f(c + \Delta x)$  只对  $\Delta x < 0$  成立。因此, 为了保证  $f(c) \leq f(c + \Delta x)$  对邻域  $|\Delta x| < \varepsilon$  内的所有  $\Delta x$  恒成立, 唯一合理的选择是令  $f'(c) = 0$ 。

满足  $f'(c) = 0$  的点  $x = c$  称为函数  $f(x)$  的平稳点 (stationary point)。

平稳点只是极值点的候选点。为了进一步确定一个平稳点是否确实为一个极小点, 有必要在一个稍大一些的邻域  $|\Delta x| < r$  内考虑函数  $f(c + \Delta x)$  的取值。由于  $f'(c) = 0$ , 故  $f(c + \Delta x) = f(c) + \frac{1}{2}f''(c)(\Delta x)^2$ 。显然, 若  $f''(c) \geq 0$ , 则一定有  $f(c) \leq f(c + \Delta x)$  对邻域  $B(c; r)$  内的所有  $\Delta x$  恒成立。因此, 函数  $f(x)$  在点  $c$  有局部极小值的条件为

$$f'(c) = 0 \quad \text{和} \quad f''(c) = \left. \frac{d^2 f(x)}{dx^2} \right|_{x=c} \geq 0 \quad (4.1.6)$$

**注释 1** 若  $f'(c) = 0$  和  $f''(c) > 0$  同时满足, 则  $c$  是函数  $f(x)$  在邻域  $B(c; r)$  内的一个严格局部极小点。

**注释 2** 若  $f'(c) = 0$  和  $f''(c) \leq 0$  同时满足, 则一定有  $f(c) \geq f(x)$  对位于邻域  $B(c; r)$  的所有  $f(c + \Delta x)$  成立。因此,  $c$  是函数  $f(x)$  在邻域  $B(c; r)$  内的一个局部极大点。特别地, 若  $f'(c) = 0$  和  $f''(c) < 0$  同时满足, 则一定有  $f(c) > f(x)$  对位于邻域  $B(c; r)$  的所有  $f(c + \Delta x)$  成立, 即  $c$  是函数  $f(x)$  在定义域  $D$  内的一个严格局部极大点。

**注释 3** 若  $f'(c) = 0$  和  $f''(c) = 0$ , 并且  $f''(c + \Delta x) \geq 0$  对位于邻域  $B(c; r)$  内的某些  $f(c + \Delta x)$  满足, 而对另一些  $f(c + \Delta x)$  却有  $f''(c + \Delta x) \leq 0$ , 则  $c$  不可能是函数  $f(x)$  在定义域  $\mathcal{D}$  内的一个极值点。这样的平稳点称为函数  $f(x)$  的一个鞍点 (saddle point)。

为方便理解平稳点和极值点之间的关系, 图 4.1.1 画出了一个单变量函数  $f(x)$  的曲线, 函数的定义域为  $\mathcal{D} = [0, 6]$ 。点  $x = 0$  和  $x = 6$  分别是该函数的严格全局极大点和严格全局极小点,  $x = 1$  和  $x = 4$  分别为一个 (非严格) 局部极小点和一个 (非严格) 局部极大点,  $x = 2$  和  $x = 3$  分别是一个严格局部极大点和一个严格局部极小点, 而  $x = 5$  则只是一个鞍点。注意,  $x = 0$  和  $x = 6$  虽然分别是函数  $f(x)$  在定义域  $[0, 6]$  的严格全局极大点和严格局部极小点, 但一阶导数  $f'(0)$  和  $f'(6)$  显然都不等于零。

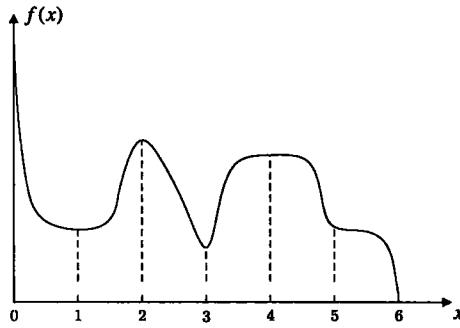


图 4.1.1 单变量函数的平稳点与极值点

函数  $f(x)$  在某点  $x = c$  的一阶导数  $f'(c) = f'(x)|_{x=c}$  反映函数在该点的变化率, 故一阶导数  $f'(x)$  称为函数  $f(x)$  的梯度函数,  $f'(c)$  称为函数  $f(x)$  在点  $x = c$  的梯度值。

#### 4.1.2 多变量函数 $f(\mathbf{x})$ 的平稳点与极值点

考虑以实向量  $\mathbf{x} = [x_1, \dots, x_n]^T$  作变元的实值函数  $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$  的无约束极小化问题

$$\min_{\mathbf{x} \in S} f(\mathbf{x}) \quad (4.1.7)$$

式中  $S \in \mathbb{R}^n$  是  $n$  维向量空间  $\mathbb{R}^n$  的一个子集合。

**定义 4.1.5** 给定一个点  $\bar{\mathbf{x}} \in \mathbb{R}^n$ , 点  $\bar{\mathbf{x}}$  的一 (闭合) 邻域记作  $B(\bar{\mathbf{x}}; r)$ , 是满足  $\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leq r$  (其中  $r > 0$ ) 的所有点  $\mathbf{x}$  的集合, 即

$$B(\bar{\mathbf{x}}; r) = \{\mathbf{x} | \|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leq r\} \quad (4.1.8)$$

**定义 4.1.6** 给定一集合  $S$ , 点  $\bar{\mathbf{x}}$  称为集合  $S$  的内点 (interior point), 若  $\bar{\mathbf{x}} \in S$ , 并且存在  $\bar{\mathbf{x}}$  的一邻域, 该邻域完全包含在集合  $S$  内。集合  $S$  的内集 (interior) 记作  $\text{int}(S)$ , 它是  $S$  的所有内点的合集。

**定义 4.1.7** 给定一集合  $S$ , 点  $\mathbf{x}$  是  $S$  的边界点 (boundary point), 若  $\mathbf{x}$  的每一个邻域至少有一个点在  $S$  内, 并且至少有一个点不在  $S$  内。集合  $S$  的边界记作  $\text{bnd}(S)$ , 是  $S$  的所有边界点的合集。一个闭集包含其所有边界点。

令  $\mathbf{c} = [c_1, \dots, c_n]^T$  是向量空间  $\mathbb{R}^n$  内的一个点, 且  $r$  为某个正数。向量空间  $\mathbb{R}^n$  内与点  $\mathbf{c}$  的距离  $\|\mathbf{x} - \mathbf{c}\|_2$  小于  $r$  的所有向量  $\mathbf{x}$  的集合称作以  $\mathbf{c}$  为中心,  $r$  为半径的  $n$  维球体 ( $n$ -ball), 记为  $B(\mathbf{c}; r)$  或者  $B(\mathbf{c})$ , 即有<sup>[328]</sup>

$$B(\mathbf{c}; r) = \{\mathbf{x} | \mathbf{x} \in \mathbb{R}^n, \|\mathbf{x} - \mathbf{c}\|_2 < r\} \quad (4.1.9)$$

$n$  维球体  $B(\mathbf{c}; r)$  也称向量  $\mathbf{c}$  的邻域。

令  $\Delta\mathbf{x} = \mathbf{x} - \mathbf{c}$ , 则在半径  $r$  足够小的邻域  $B(\mathbf{c}; r)$  内, 实变函数  $f(\mathbf{x})$  在点  $\mathbf{c}$  的二阶 Taylor 级数逼近为

$$f(\mathbf{c} + \Delta\mathbf{x}) = f(\mathbf{c}) + \left( \frac{\partial f(\mathbf{c})}{\partial \mathbf{c}} \right)^T \Delta\mathbf{x} + \frac{1}{2} (\Delta\mathbf{c})^T \frac{\partial^2 f(\mathbf{c})}{\partial \mathbf{c} \partial \mathbf{c}^T} \Delta\mathbf{x} \quad (4.1.10)$$

$$= f(\mathbf{c}) + (\nabla f(\mathbf{c}))^T \Delta\mathbf{x} + \frac{1}{2} (\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} \quad (4.1.11)$$

式中

$$\nabla f(\mathbf{c}) = \frac{\partial f(\mathbf{c})}{\partial \mathbf{c}} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} \quad (4.1.12)$$

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{c})}{\partial \mathbf{c} \partial \mathbf{c}^T} = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \quad (4.1.13)$$

分别是函数  $f(\mathbf{x})$  在点  $\mathbf{c}$  的梯度向量和 Hessian 矩阵。

将单变量函数的极值点的定义加以推广, 即可得到以实向量为变元的实值函数  $f(\mathbf{x})$  的极小点的定义如下。

**定义 4.1.8** 令标量  $r > 0$ , 并且  $\mathbf{x} = \mathbf{c} + \Delta\mathbf{x}$  是向量空间  $\mathbb{R}^n$  的子集合  $S$  的点。若

$$f(\mathbf{c}) \leq f(\mathbf{c} + \Delta\mathbf{x}) \quad \forall 0 < \|\Delta\mathbf{x}\|_2 \leq r; \quad (4.1.14)$$

则称点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个局部极小点。若

$$f(\mathbf{c}) < f(\mathbf{c} + \Delta\mathbf{x}) \quad \forall 0 < \|\Delta\mathbf{x}\|_2 \leq r; \quad (4.1.15)$$

则称点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个严格局部极小点。若

$$f(\mathbf{c}) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in S; \quad (4.1.16)$$

则称点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  在定义域  $S$  的一个全局极小点。若

$$f(\mathbf{c}) < f(\mathbf{x}) \quad \forall \mathbf{x} \in S, \mathbf{x} \neq \mathbf{c} \quad (4.1.17)$$

则称点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  在定义域  $S$  的一个严格全局极小点。

由式 (4.1.13) 易知, 在邻域  $B(\mathbf{c}; r)$  的一个足够小的内部区域  $\|\Delta\mathbf{x}\|_2 < \varepsilon$ , 二阶项可以忽略的情况下, 函数的一阶 Taylor 级数逼近为

$$f(\mathbf{c} + \Delta\mathbf{x}) \approx f(\mathbf{c}) + (\nabla f(\mathbf{c}))^T \Delta\mathbf{x} \quad (4.1.18)$$

显然, 为了保证  $f(\mathbf{c}) \leq f(\mathbf{c} + \Delta\mathbf{x})$  对满足  $\|\Delta\mathbf{x}\|_2 < \varepsilon$  的所有  $\Delta\mathbf{x}$  恒成立, 必须选择

$$\nabla f(\mathbf{c}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} = \mathbf{0}, \quad \forall 0 < \|\Delta\mathbf{x}\|_2 < r \quad (4.1.19)$$

在  $\nabla f(\mathbf{c}) = \mathbf{0}$  的选择下, 对于二阶项不能忽略的邻域  $\|\Delta\mathbf{x}\|_2 < r$ , 函数  $f(\mathbf{x})$  有二阶 Taylor 级数逼近

$$f(\mathbf{c} + \Delta\mathbf{x}) \approx f(\mathbf{c}) + \frac{1}{2}(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} \quad (4.1.20)$$

于是, 我们容易得出以下结论:

- (1) 若二次型  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} \geq 0$  对  $0 < \|\Delta\mathbf{x}\|_2 < r$  或者  $\Delta\mathbf{x} \in B(\mathbf{c}; r)$  的所有  $\Delta\mathbf{x}$  恒成立, 或 Hessian 矩阵半正定

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \succeq 0 \quad (4.1.21)$$

则  $f(\mathbf{c}) \leq f(\mathbf{c} + \Delta\mathbf{x}), \forall \Delta\mathbf{x} \in B(\mathbf{c}; r)$ , 即点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个局部极小点,

- (2) 若二次型  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} > 0$  对所有  $\Delta\mathbf{x} \in B(\mathbf{c}; r)$  恒成立, 或 Hessian 矩阵正定

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \succ 0 \quad (4.1.22)$$

则点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个严格局部极小点。

- (3) 若二次型  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} \leq 0$  对所有  $\Delta\mathbf{x} \in B(\mathbf{c}; r)$  恒成立, 或 Hessian 矩阵半负定

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \preceq 0 \quad (4.1.23)$$

则  $f(\mathbf{c}) \geq f(\mathbf{c} + \Delta\mathbf{x}), \forall \Delta\mathbf{x} \in B(\mathbf{c}; r)$ , 即点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个局部极大点。

- (4) 若二次型  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} < 0$  对所有  $\Delta\mathbf{x} \in B(\mathbf{c}; r)$  成立, 或 Hessian 矩阵负定

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \prec 0 \quad (4.1.24)$$

则点  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个严格局部极大点

- (5) 若二次型  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} \leq 0$  对邻域  $B(\mathbf{c}; r)$  的某些点  $\Delta\mathbf{x}$  成立, 而  $(\Delta\mathbf{x})^T \mathbf{H}(f(\mathbf{c})) \Delta\mathbf{x} > 0$  对邻域  $B(\mathbf{c}; r)$  的另一些点  $\Delta\mathbf{x}$  成立, 或 Hessian 矩阵

$$\mathbf{H}(f(\mathbf{c})) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \text{ 不定} \quad (4.1.25)$$

则点  $\mathbf{c}$  只是函数  $f(\mathbf{x})$  的一个鞍点。

### 4.1.3 多变量函数 $f(\mathbf{X})$ 的平稳点与极值点

现在考虑以矩阵为变元的实值函数  $f(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ 。此时, 需要先通过向量化, 将变元矩阵  $\mathbf{X} \in \mathbb{R}^{m \times n}$ , 变成一个  $mn \times 1$  向量  $\text{vec}(\mathbf{X})$ 。

令  $S$  是矩阵空间  $\mathbb{R}^{m \times n}$  的一个子集合, 它是  $m \times n$  矩阵变元  $\mathbf{X}$  的定义域, 即  $\mathbf{X} \in S$ 。

函数  $f(\mathbf{X})$  以点  $\text{vec}(\mathbf{C})$  为中心,  $r$  为半径的邻域记作  $B(\mathbf{C}; r)$ , 定义为

$$B(\mathbf{C}; r) = \{\mathbf{X} | \mathbf{X} \in \mathbb{R}^{m \times n}, \|\text{vec}(\mathbf{X}) - \text{vec}(\mathbf{C})\|_2 < r\} \quad (4.1.26)$$

于是, 由式 (4.1.13) 知, 函数  $f(\mathbf{X})$  在点  $\mathbf{C}$  的二阶 Taylor 级数逼近公式为

$$\begin{aligned} f(\mathbf{C} + \Delta \mathbf{X}) &= f(\mathbf{C}) + \left( \frac{\partial f(\mathbf{C})}{\partial \text{vec}(\mathbf{C})} \right)^T \text{vec}(\Delta \mathbf{X}) \\ &\quad + \frac{1}{2} (\text{vec}(\Delta \mathbf{X}))^T \frac{\partial^2 f(\mathbf{C})}{\partial \text{vec}(\mathbf{C}) \partial (\text{vec} \mathbf{C})^T} \text{vec}(\Delta \mathbf{X}) \\ &= f(\mathbf{C}) + (\nabla_{\text{vec} \mathbf{C}} f(\mathbf{C}))^T \text{vec}(\Delta \mathbf{X}) \\ &\quad + \frac{1}{2} (\text{vec}(\Delta \mathbf{X}))^T \mathbf{H}(f(\mathbf{C})) \text{vec}(\Delta \mathbf{X}) \end{aligned} \quad (4.1.27)$$

式中

$$\nabla_{\text{vec} \mathbf{C}} f(\mathbf{C}) = \left. \frac{\partial f(\mathbf{X})}{\partial \text{vec}(\mathbf{X})} \right|_{\mathbf{X}=\mathbf{C}} \in \mathbb{R}^{mn} \quad (4.1.28)$$

$$\mathbf{H}(f(\mathbf{C})) = \left. \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \right|_{\mathbf{X}=\mathbf{C}} \in \mathbb{R}^{mn \times mn} \quad (4.1.29)$$

分别是函数  $f(\mathbf{X})$  在点  $\mathbf{C}$  的梯度向量和 Hessian 矩阵。

如果  $0 < \|\text{vec}(\Delta \mathbf{X})\|_2 < \varepsilon$ , 且  $\varepsilon$  足够小, 以至于二次项  $(\text{vec}(\Delta \mathbf{X}))^T \mathbf{H}(f(\mathbf{C})) \text{vec}(\Delta \mathbf{X})$  可以忽略, 则有一阶 Tarloy 级数逼近

$$f(\mathbf{C} + \Delta \mathbf{X}) \approx f(\mathbf{C}) + (\nabla_{\text{vec} \mathbf{C}} f(\mathbf{C}))^T \text{vec}(\Delta \mathbf{X}) \quad (4.1.30)$$

显然, 为了使得  $f(\mathbf{C} + \Delta \mathbf{X}) \geq f(\mathbf{C})$  对于满足  $0 < \|\Delta \mathbf{X}\|_2 < \varepsilon$  的所有  $\Delta \mathbf{X}$  均成立, 必须选择

$$\nabla_{\text{vec} \mathbf{C}} f(\mathbf{C}) = \left. \frac{\partial f(\mathbf{X})}{\partial \text{vec} \mathbf{X}} \right|_{\mathbf{X}=\mathbf{C}} = \mathbf{0} \quad (4.1.31)$$

在选择  $\nabla_{\text{vec} \mathbf{C}} f(\mathbf{C}) = \mathbf{0}$  的条件下, 考虑二次项  $(\text{vec}(\Delta \mathbf{X}))^T \mathbf{H}(f(\mathbf{C})) \text{vec}(\Delta \mathbf{X})$  不可忽略的邻域  $B(\mathbf{C}; r)$ 。此时, 有二阶 Taylor 级数逼近

$$f(\mathbf{C} + \Delta \mathbf{X}) \approx f(\mathbf{C}) + \frac{1}{2} (\text{vec}(\Delta \mathbf{X}))^T \mathbf{H}(f(\mathbf{C})) \text{vec}(\Delta \mathbf{X}) \quad (4.1.32)$$

由此容易得出以下结论:

(1)  $f(\mathbf{C}) \leq f(\mathbf{C} + \Delta \mathbf{X}), \forall \Delta \mathbf{X} \in B(\mathbf{C}; r)$ , 即点  $\mathbf{C}$  是函数  $f(\mathbf{X})$  的一个局部极小点, 若二次型  $(\text{vec} \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec} \Delta \mathbf{X} \geq 0$  对满足  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  的所有  $\Delta \mathbf{X}$  恒成立, 或 Hessian 矩阵半正定

$$\mathbf{H}(f(\mathbf{C})) = \left. \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \right|_{\mathbf{X}=\mathbf{C}} \succeq 0 \quad (4.1.33)$$

(2) 点  $\mathbf{C}$  是  $f(\mathbf{X})$  的一个严格局部极小点, 若二次型  $(\text{vec} \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec} \Delta \mathbf{X} > 0$  对所有  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  恒成立, 或 Hessian 矩阵正定

$$\mathbf{H}(f(\mathbf{C})) = \left. \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \right|_{\mathbf{X}=\mathbf{C}} \succ 0 \quad (4.1.34)$$

(3)  $f(\mathbf{C}) \geq f(\mathbf{C} + \Delta \mathbf{X}), \forall \Delta \mathbf{X} \in B(\mathbf{C}; r)$ , 即点  $\mathbf{C}$  是  $f(\mathbf{X})$  的一个局部极大点, 若二次型  $(\text{vec } \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec } \Delta \mathbf{X} \leq 0$  对所有  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  恒成立, 或 Hessian 矩阵半负定

$$\mathbf{H}(f(\mathbf{C})) = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big|_{\mathbf{X}=\mathbf{C}} \preceq 0 \quad (4.1.35)$$

(4) 点  $\mathbf{C}$  是  $f(\mathbf{X})$  的一个严格局部极大点, 若二次型  $(\text{vec } \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec } \Delta \mathbf{X} < 0$  对所有  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  恒成立, 或 Hessian 矩阵负定

$$\mathbf{H}(f(\mathbf{C})) = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big|_{\mathbf{X}=\mathbf{C}} \prec 0 \quad (4.1.36)$$

(5) 点  $\mathbf{C}$  只是函数  $f(\mathbf{X})$  的一个鞍点, 若二次型  $(\text{vec } \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec } \Delta \mathbf{X} \leq 0$  对某些  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  成立, 而对另一些  $\Delta \mathbf{X} \in B(\mathbf{C}; r)$  有  $(\text{vec } \Delta \mathbf{X})^T \mathbf{H}(f(\mathbf{C})) \text{vec } \Delta \mathbf{X} \geq 0$ , 或 Hessian 矩阵

$$\mathbf{H}(f(\mathbf{C})) = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big|_{\mathbf{X}=\mathbf{C}} \text{ 不定} \quad (4.1.37)$$

表 4.1.1 归纳了无约束优化函数的平稳点和极值点的条件。

表 4.1.1 实变函数的平稳点和极值点的条件

实变函数	$f(x) : \mathbb{R} \rightarrow \mathbb{R}$	$f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$	$f(\mathbf{X}) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$
平稳点	$\frac{\partial f(x)}{\partial x} \Big _{x=c} = 0$	$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big _{\mathbf{x}=c} = \mathbf{0}$	$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} \Big _{\mathbf{X}=\mathbf{C}} = \mathbf{O}_{m \times n}$
局部极小点	$\frac{\partial^2 f(x)}{\partial x^2} \Big _{x=c} \geq 0$	$\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big _{\mathbf{x}=c} \succeq 0$	$\frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big _{\mathbf{X}=\mathbf{C}} \succeq 0$
严格局部极小点	$\frac{\partial^2 f(x)}{\partial x^2} \Big _{x=c} > 0$	$\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big _{\mathbf{x}=c} \succ 0$	$\frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big _{\mathbf{X}=\mathbf{C}} \succ 0$
局部极大点	$\frac{\partial^2 f(x)}{\partial x^2} \Big _{x=c} \leq 0$	$\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big _{\mathbf{x}=c} \preceq 0$	$\frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big _{\mathbf{X}=\mathbf{C}} \preceq 0$
严格局部极大点	$\frac{\partial^2 f(x)}{\partial x^2} \Big _{x=c} < 0$	$\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big _{\mathbf{x}=c} \prec 0$	$\frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big _{\mathbf{X}=\mathbf{C}} \prec 0$
鞍点	$\frac{\partial^2 f(x)}{\partial x^2} \Big _{x=c} \text{ 不定}$	$\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big _{\mathbf{x}=c} \text{ 不定}$	$\frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial (\text{vec} \mathbf{X})^T} \Big _{\mathbf{X}=\mathbf{C}} \text{ 不定}$

#### 4.1.4 实变函数的梯度分析

多变量函数  $f(\mathbf{x})$  的平稳点与极值点分析可以总结为局部极值点的下列必要条件。

**定理 4.1.1 (极值点一阶必要条件)**<sup>[372]</sup> 若  $c$  是  $f(\mathbf{x})$  的局部极值点, 并且  $f(\mathbf{x})$  在点  $c$  的邻域  $B(c; r)$  内是连续可微分的, 则

$$\nabla_c f(c) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=c} = \mathbf{0} \quad (4.1.38).$$

事实上, 式 (4.1.38) 只是平稳点的一阶必要条件, 而非一阶充分条件, 因为正如图 4.1.1 所示的那样, 有的平稳点可能只是一个鞍点。

**定理 4.1.2 (局部极小点二阶必要条件)** [328, 372] 若  $\mathbf{c}$  是  $f(\mathbf{x})$  的局部极小点,  $f(\mathbf{x})$  在  $\mathbf{c}$  点是可微分的,  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  在  $\mathbf{c}$  的邻域  $B(\mathbf{c}; r)$  内连续, 则

$$\nabla_{\mathbf{c}} f(\mathbf{c}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} = \mathbf{0} \quad \text{和} \quad \nabla_{\mathbf{c}}^2 f(\mathbf{c}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \succeq 0 \quad (4.1.39)$$

式中,  $\nabla_{\mathbf{c}}^2 f(\mathbf{c}) \succeq 0$  表示 Hessian 矩阵  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  在  $\mathbf{c}$  点的值  $\nabla_{\mathbf{c}}^2 f(\mathbf{c})$  是一个半正定矩阵。

注释 1 如果将定理 4.1.2 的条件式 (4.1.39) 换成

$$\nabla_{\mathbf{c}} f(\mathbf{c}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} = \mathbf{0} \quad \text{和} \quad \nabla_{\mathbf{c}}^2 f(\mathbf{c}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \preceq 0 \quad (4.1.40)$$

则定理 4.1.2 给出  $\mathbf{c}$  点是函数  $f(\mathbf{x})$  的局部极大点的二阶必要条件。式中,  $\nabla_{\mathbf{c}}^2 f(\mathbf{c}) \preceq 0$  表示在  $\mathbf{c}$  点的 Hessian 矩阵  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  半负定。

注释 2 对于一个以  $m \times n$  矩阵  $\mathbf{X}$  为变元的实变函数  $f(\mathbf{X})$ , 定理 4.1.2 的相应叙述为: 若  $\mathbf{C}$  是函数  $f(\mathbf{X})$  的一个局部极小点,  $f(\mathbf{X})$  在  $\mathbf{X}$  点是可微分的, 并且  $\nabla_{\text{vec } \mathbf{X}}^2 f(\mathbf{X})$  在  $\mathbf{C}$  的邻域  $B(\mathbf{C}; r)$  内连续, 则

$$\nabla_{\text{vec } \mathbf{C}} f(\mathbf{C}) = \frac{\partial f(\mathbf{X})}{\partial \text{vec } (\mathbf{X})} \Big|_{\mathbf{X}=\mathbf{C}} = \mathbf{0}_{mn \times 1} \quad (4.1.41)$$

和

$$\nabla_{\text{vec } \mathbf{C}}^2 f(\mathbf{C}) = \frac{\partial^2 f(\mathbf{X})}{\partial (\text{vec } \mathbf{X}) \partial (\text{vec } \mathbf{X})^T} \Big|_{\mathbf{X}=\mathbf{C}} \succeq 0 \quad (4.1.42)$$

需要强调的是: 定理 4.1.2 只是实变函数  $f(\mathbf{x})$  的局部极小点的必要条件, 而不是充分条件。然而, 对于一个无约束优化算法, 我们往往希望能够直接判断算法收敛的点  $\mathbf{c}$  或者  $\mathbf{C}$  是否就是给定的目标函数  $f(\mathbf{x})$  或者  $f(\mathbf{X})$  的一个极值点。下面的定理提供了这一问题的解决途径。

**定理 4.1.3 (局部极小点二阶充分条件)** [328, 372] 假设  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  在  $\mathbf{c}$  的开邻域内连续, 并且

$$\nabla_{\mathbf{c}} f(\mathbf{c}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} = \mathbf{0} \quad \text{和} \quad \nabla_{\mathbf{c}}^2 f(\mathbf{c}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \succ 0 \quad (4.1.43)$$

则  $\mathbf{c}$  是函数  $f(\mathbf{x})$  的一个严格局部极小点。式中,  $\nabla_{\mathbf{x}}^2 f(\mathbf{c}) \succ 0$  表示  $\mathbf{c}$  点的 Hessian 矩阵  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  正定。

注释 1 如果将定理 4.1.3 的条件式 (4.1.43) 换成

$$\nabla_{\mathbf{c}} f(\mathbf{c}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{c}} = \mathbf{0} \quad \text{和} \quad \nabla_{\mathbf{c}}^2 f(\mathbf{c}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \Big|_{\mathbf{x}=\mathbf{c}} \prec 0 \quad (4.1.44)$$

则定理 4.1.3 给出  $\mathbf{c}$  点是函数  $f(\mathbf{x})$  的一个严格局部极大点的二阶充分条件。式中  $\nabla_{\mathbf{c}}^2 f(\mathbf{c}) \prec 0$  表示在  $\mathbf{c}$  点的 Hessian 矩阵  $\nabla_{\mathbf{x}}^2 f(\mathbf{x})$  负定。

注释 2 对于一个以  $m \times n$  矩阵  $\mathbf{X}$  为变元的实变函数  $f(\mathbf{X})$ , 定理 4.1.3 的相应叙述

如下：假定  $f(\mathbf{X})$  在  $\mathbf{X}$  点是可微分的， $\nabla_{\text{vec } \mathbf{X}}^2 f(\mathbf{X})$  在  $\mathbf{C}$  的邻域  $B(\mathbf{C}; r)$  内连续，并且

$$\nabla_{\text{vec } \mathbf{C}} f(\mathbf{C}) = \left. \frac{\partial f(\mathbf{X})}{\partial \text{vec } (\mathbf{X})} \right|_{\mathbf{X}=\mathbf{C}} = \mathbf{0}_{mn \times 1} \quad (4.1.45)$$

$$\nabla_{\text{vec } \mathbf{C}}^2 f(\mathbf{C}) = \left. \frac{\partial^2 f(\mathbf{X})}{\partial (\text{vec } \mathbf{X}) \partial (\text{vec } \mathbf{X})^T} \right|_{\mathbf{X}=\mathbf{C}} \succ 0 \quad (4.1.46)$$

则  $\mathbf{C}$  是函数  $f(\mathbf{X})$  的一个严格局部极小点。

**注释 3** 只要 Hessian 矩阵  $\nabla_{\mathbf{c}}^2 f(\mathbf{c})$  或  $\nabla_{\text{vec } \mathbf{C}}^2 f(\mathbf{C})$  不定，则  $\mathbf{c}$  或  $\mathbf{C}$  点就不能保证是函数  $f(\mathbf{x})$  或  $f(\mathbf{X})$  的一个极值点，它有可能只是一个鞍点。

## 4.2 复变函数无约束优化的梯度分析

在大量的工程应用（例如无线通信、雷达、声呐等）中，信号往往表现为复向量形式。本节考察复向量为变元的实值目标函数的无约束最优化问题。

### 4.2.1 多变量复变函数 $f(z, z^*)$ 的平稳点与极值点

现在考虑以  $n \times 1$  复数向量为变元的实值函数  $f(z, z^*) : \mathbb{C}^n \times \mathbb{C}^n \rightarrow \mathbb{R}$  的最优化。

由一阶微分

$$df(z, z^*) = \frac{\partial f(z, z^*)}{\partial z^T} dz + \frac{\partial f(z, z^*)}{\partial z^H} dz^* = \left[ \frac{\partial f(z, z^*)}{\partial z^T}, \frac{\partial f(z, z^*)}{\partial z^H} \right] \begin{bmatrix} dz \\ dz^* \end{bmatrix} \quad (4.2.1)$$

和二阶微分

$$\begin{aligned} d^2 f(z, z^*) &= \left( \frac{\partial^2 f(z, z^*)}{\partial z \partial z^T} dz + \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^T} dz^* \right)^T dz \\ &\quad + \left( \frac{\partial^2 f(z, z^*)}{\partial z \partial z^H} dz + \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^H} dz^* \right)^T dz^* \\ &= [dz^H, dz^T] \begin{bmatrix} \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^H} \\ \frac{\partial^2 f(z, z^*)}{\partial z \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z \partial z^H} \end{bmatrix} \begin{bmatrix} dz \\ dz^* \end{bmatrix} \end{aligned} \quad (4.2.2)$$

易知实值函数  $f(z, z^*)$  在  $\mathbf{c}$  点的二阶 Taylor 级数逼近为

$$\begin{aligned} f(z, z^*) &\approx f(\mathbf{c}, \mathbf{c}^*) + \left[ \frac{\partial f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c}}, \frac{\partial f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c}^*} \right] \begin{bmatrix} \Delta \mathbf{c} \\ \Delta \mathbf{c}^* \end{bmatrix} \\ &\quad + \frac{1}{2} [\Delta \mathbf{c}^H, \Delta \mathbf{c}^T] \begin{bmatrix} \frac{\partial^2 f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c}^* \partial \mathbf{c}^T} & \frac{\partial^2 f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c}^* \partial \mathbf{c}^H} \\ \frac{\partial^2 f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c} \partial \mathbf{c}^T} & \frac{\partial^2 f(\mathbf{c}, \mathbf{c}^*)}{\partial \mathbf{c} \partial \mathbf{c}^H} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{c} \\ \Delta \mathbf{c}^* \end{bmatrix} \\ &= f(\mathbf{c}, \mathbf{c}^*) + (\nabla f(\mathbf{c}, \mathbf{c}^*))^T \Delta \tilde{\mathbf{c}} + \frac{1}{2} (\Delta \tilde{\mathbf{c}})^H \mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) \Delta \tilde{\mathbf{c}} \end{aligned} \quad (4.2.3)$$

式中  $\Delta\tilde{c} = \begin{bmatrix} \Delta c \\ \Delta c^* \end{bmatrix} \in \mathbb{C}^{2n}$  代表复自变量  $z$  在  $c$  点的偏改变量  $\Delta c = z - c$  和  $\Delta c^* = z^* - c^*$  组成的向量, 而  $\nabla f(c, c^*)$  和  $H(f(c, c^*))$  分别是实值函数  $f(z, z^*)$  的梯度向量

$$\nabla f(z, z^*) = \begin{bmatrix} \frac{\partial f(z, z^*)}{\partial z} \\ \frac{\partial f(z, z^*)}{\partial z^*} \end{bmatrix} \in \mathbb{C}^{2n} \quad (4.2.4)$$

和 Hessian 矩阵

$$H(f(z, z^*)) = \begin{bmatrix} \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^H} \\ \frac{\partial^2 f(z, z^*)}{\partial z \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z \partial z^H} \end{bmatrix} \in \mathbb{C}^{2n \times 2n} \quad (4.2.5)$$

在  $z = c$  点的值。

考虑  $c$  点的邻域  $B(c; r)$ , 其中  $\|\Delta c\|_2$  足够小, 以至于式 (4.2.3) 中的二次项可以忽略, 从而有函数的一阶逼近

$$f(z, z^*) \approx f(c, c^*) + (\nabla f(c, c^*))^T \Delta \tilde{c} \quad (\|\Delta c\|_2 \text{ 足够小}) \quad (4.2.6)$$

显然, 为了使  $f(c, c^*)$  取极小值或者极大值, 必须要求  $c$  是一个平稳点, 即  $\nabla f(c, c^*) = 0_{2n \times 1}$ 。这一平稳点条件等价为  $\frac{\partial f(c, c^*)}{\partial c} = 0_{n \times 1}$  和  $\frac{\partial f(c, c^*)}{\partial c^*} = 0_{n \times 1}$ , 或简化为在  $c$  点的共轭偏导向量为零向量

$$\frac{\partial f(c, c^*)}{\partial c^*} = \left. \frac{\partial f(z, z^*)}{\partial z^*} \right|_{z=c} = 0_{n \times 1} \quad (4.2.7)$$

实值函数  $f(z, z^*)$  在平稳点  $c$  的二阶 Taylor 级数逼近为

$$f(z, z^*) \approx f(c, c^*) + \frac{1}{2}(\Delta \tilde{c})^H H(f(c, c^*)) \Delta \tilde{c} \quad (4.2.8)$$

由此容易得出实值函数  $f(z, z^*)$  的极值点条件如下:

(1) 平稳点  $c$  是  $f(z, z^*)$  的一个局部极小点, 若二次型  $(\Delta \tilde{c})^H H(f(c, c^*)) \Delta \tilde{c} \geq 0$  对所有  $\Delta c \in B(c; r)$  恒成立, 或等价于 Hessian 矩阵半正定

$$H(f(c, c^*)) = \left[ \begin{array}{cc} \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^H} \\ \frac{\partial^2 f(z, z^*)}{\partial z \partial z^T} & \frac{\partial^2 f(z, z^*)}{\partial z \partial z^H} \end{array} \right]_{z=c} \succeq 0 \quad (4.2.9)$$

(2)  $c$  是一个严格局部极小点, 若 Hessian 矩阵正定, 即  $H(f(z, z^*))|_{z=c} \succ 0$ 。

(3)  $c$  是一个局部极大点, 若 Hessian 矩阵半负定, 即  $H(f(z, z^*))|_{z=c} \preceq 0$ 。

(4)  $c$  是一个严格局部极大点, 若 Hessian 矩阵负定, 即  $H(f(z, z^*))|_{z=c} \prec 0$ 。

(5)  $c$  只是一个鞍点, 若 Hessian 矩阵  $H(f(z, z^*))|_{z=c}$  不定。

### 4.2.2 多变量复变函数 $f(\mathbf{Z}, \mathbf{Z}^*)$ 的平稳点与极值点

对于以  $m \times n$  复数矩阵为变元的实值函数  $f(\mathbf{Z}, \mathbf{Z}^*) : \mathbb{C}^n \times \mathbb{C}^n \rightarrow \mathbb{R}$  的最优化，需要将变元矩阵  $\mathbf{Z}$  和  $\mathbf{Z}^*$  分别向量化为  $\text{vec}(\mathbf{Z})$  和  $\text{vec}(\mathbf{Z}^*)$ 。

由实值函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的一阶微分

$$\begin{aligned} d\mathbf{f}(\mathbf{Z}, \mathbf{Z}^*) &= \left( \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}(\mathbf{Z})} \right)^T \text{vec}(d\mathbf{Z}) + \left( \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \text{vec}(\mathbf{Z}^*)} \right)^T \text{vec}(d\mathbf{Z}^*) \\ &= \left[ \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})^T}, \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)^T} \right] \begin{bmatrix} \text{vec}(d\mathbf{Z}) \\ \text{vec}(d\mathbf{Z}^*) \end{bmatrix} \end{aligned} \quad (4.2.10)$$

和二阶微分

$$\begin{aligned} d^2\mathbf{f}(\mathbf{Z}, \mathbf{Z}^*) &= \left( \frac{\partial f^2(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z})^T} \text{vec}(d\mathbf{Z}) + \frac{\partial f^2(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z})^T} \text{vec}(d\mathbf{Z}^*) \right)^T \text{vec}(d\mathbf{Z}) \\ &\quad + \left( \frac{\partial f^2(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z}^*)^T} \text{vec}(d\mathbf{Z}) + \frac{\partial f^2(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z}^*)^T} \text{vec}(d\mathbf{Z}^*) \right)^T \text{vec}(d\mathbf{Z}^*) \\ &= [(\text{vec}(d\mathbf{Z}^*))^T, (\text{vec}(d\mathbf{Z}))^T] \begin{bmatrix} \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z}^*)^T} \\ \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z}^*)^T} \end{bmatrix} \begin{bmatrix} \text{vec}(d\mathbf{Z}) \\ \text{vec}(d\mathbf{Z}^*) \end{bmatrix} \end{aligned} \quad (4.2.11)$$

易知实值函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  在  $\mathbf{C}$  点的二阶 Taylor 级数逼近为

$$\begin{aligned} f(\mathbf{Z}, \mathbf{Z}^*) &= f(\mathbf{C}, \mathbf{C}^*) + \left[ \frac{\partial f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C})^T}, \frac{\partial f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C}^*)^T} \right] \begin{bmatrix} \text{vec}(\Delta \mathbf{C}) \\ \text{vec}(\Delta \mathbf{C}^*) \end{bmatrix} \\ &\quad + \frac{1}{2} [(\text{vec}(\Delta \mathbf{C}^*))^T, (\text{vec}(\Delta \mathbf{C}))^T] \begin{bmatrix} \frac{\partial^2 f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C}^*) \partial (\text{vec } \mathbf{C})^T} & \frac{\partial^2 f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C}^*) \partial (\text{vec } \mathbf{C}^*)^T} \\ \frac{\partial^2 f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C}) \partial (\text{vec } \mathbf{C})^T} & \frac{\partial^2 f(\mathbf{C}, \mathbf{C}^*)}{\partial (\text{vec } \mathbf{C}) \partial (\text{vec } \mathbf{C}^*)^T} \end{bmatrix} \begin{bmatrix} \text{vec}(\Delta \mathbf{C}) \\ \text{vec}(\Delta \mathbf{C}^*) \end{bmatrix} \\ &= f(\mathbf{C}, \mathbf{C}^*) + (\nabla f(\mathbf{C}, \mathbf{C}^*))^T \text{vec}(\Delta \tilde{\mathbf{C}}) + \frac{1}{2} (\text{vec}(\Delta \tilde{\mathbf{C}}))^H \mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) \text{vec}(\Delta \tilde{\mathbf{C}}) \end{aligned} \quad (4.2.12)$$

式中  $\Delta \tilde{\mathbf{C}} = \begin{bmatrix} \Delta \mathbf{C} \\ \Delta \mathbf{C}^* \end{bmatrix} = \begin{bmatrix} \mathbf{Z} - \mathbf{C} \\ \mathbf{Z}^* - \mathbf{C}^* \end{bmatrix} \in \mathbb{C}^{2n}$ ，而  $\nabla f(\mathbf{C}, \mathbf{C}^*)$  和  $\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*))$  则分别是复变函数  $f(\mathbf{Z}, \mathbf{Z}^*)$  的梯度向量

$$\nabla_{\text{vec}(\mathbf{Z})} f(\mathbf{Z}, \mathbf{Z}^*) = \begin{bmatrix} \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z})} \\ \frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*)} \end{bmatrix} \in \mathbb{C}^{2mn} \quad (4.2.13)$$

和 Hessian 矩阵

$$\mathbf{H}(f(\mathbf{Z}, \mathbf{Z}^*)) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z}^*)^T} \\ \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z}^*)^T} \end{bmatrix} \in \mathbb{C}^{2mn \times 2mn} \quad (4.2.14)$$

在  $Z = C$  点的值。

当  $\|\text{vec}(\Delta C)\|_2$  或者  $C$  点的邻域

$$B(C; r) = \{Y \in \mathbb{C}^{m \times n} \mid \|C - Y\|_2 < r\} \quad (4.2.15)$$

足够小时, 式 (4.2.12) 中的二次项可以忽略, 从而有

$$f(Z, Z^*) = f(C, C^*) + (\nabla f(C, C^*))^\text{T} \text{vec}(\Delta \tilde{C}) \quad (\|\text{vec}(\Delta C)\|_2 \text{ 足够小}) \quad (4.2.16)$$

显然, 为了使  $f(C, C^*)$  取极小值或者极大值, 必须要求  $C$  是一个平稳点, 即  $\nabla f(C, C^*) = \mathbf{0}_{2n \times 1}$ 。这一平稳点条件等价为  $\frac{\partial f(C, C^*)}{\partial \text{vec}(C)} = \mathbf{0}_{n \times 1}$  和  $\frac{\partial f(C, C^*)}{\partial \text{vec}(C^*)} = \mathbf{0}_{nn \times 1}$ , 又可简化为在  $C$  点的共轭偏导向量为零向量

$$\frac{\partial f(C, C^*)}{\partial \text{vec}(C^*)} = \left. \frac{\partial f(Z, Z^*)}{\partial \text{vec}(Z^*)} \right|_{Z=C} = \mathbf{0}_{nn \times 1} \quad (4.2.17)$$

或者等价于实值函数  $f(Z, Z^*)$  在  $Z = C$  点的共轭梯度矩阵为零矩阵

$$\frac{\partial f(C, C^*)}{\partial C^*} = \left. \frac{\partial f(Z, Z^*)}{\partial Z^*} \right|_{Z=C} = \mathbf{O}_{n \times n} \quad (4.2.18)$$

于是, 实值函数  $f(Z, Z^*)$  在平稳点  $Z = C$  的二阶 Taylor 级数逼近为

$$f(Z, Z^*) = f(C, C^*) + \frac{1}{2} (\text{vec}(\Delta \tilde{C}))^\text{H} \mathbf{H}(f(C, C^*)) \text{vec}(\Delta \tilde{C}) \quad (4.2.19)$$

根据二次型  $(\text{vec}(\Delta \tilde{C}))^\text{H} \mathbf{H}(f(C, C^*)) \text{vec}(\Delta \tilde{C}) \geq 0$  的取值即 Hessian 矩阵  $\mathbf{H}(f(C, C^*))$  的正定性, 容易得出实值函数  $f(Z, Z^*)$  的极值点条件如下:

(1) 平稳点  $C$  是  $f(Z, Z^*)$  的一个局部极小点, 若 Hessian 矩阵半正定

$$\mathbf{H}(f(C, C^*)) = \begin{bmatrix} \frac{\partial^2 f(Z, Z^*)}{\partial (\text{vec } Z^*) \partial (\text{vec } Z)^T} & \frac{\partial^2 f(Z, Z^*)}{\partial (\text{vec } Z^*) \partial (\text{vec } Z^*)^T} \\ \frac{\partial^2 f(Z, Z^*)}{\partial (\text{vec } Z) \partial (\text{vec } Z)^T} & \frac{\partial^2 f(Z, Z^*)}{\partial (\text{vec } Z) \partial (\text{vec } Z^*)^T} \end{bmatrix} \Big|_{Z=C} \succeq 0 \quad (4.2.20)$$

(2) 平稳点  $C$  是  $f(Z, Z^*)$  的一严格局部极小点, 若 Hessian 矩阵正定。

(3) 平稳点  $C$  是  $f(Z, Z^*)$  的一局部极大点, 若 Hessian 矩阵半负定。

(4) 平稳点  $C$  是  $f(Z, Z^*)$  的一严格局部极大点, 若 Hessian 矩阵负定。

(5) 平稳点  $C$  是  $f(Z, Z^*)$  的一鞍点, 若二次型  $(\text{vec}(\Delta \tilde{C}))^\text{H} \mathbf{H}(f(C, C^*)) \text{vec}(\Delta \tilde{C}) > 0$  对某些点  $\Delta C \in B(C; r)$  成立, 而  $(\text{vec}(\Delta \tilde{C}))^\text{H} \mathbf{H}(f(C, C^*)) \text{vec}(\Delta \tilde{C}) < 0$  对另一些点  $\Delta C \in B(C; r)$  成立, 或者等价于 Hessian 矩阵  $\mathbf{H}(f(Z, Z^*))|_{Z=C}$  为不定矩阵。

表 4.2.1 汇总了复变函数的平稳点和极值点条件。

表 4.2.1 复变函数的平稳点和极值点条件

复变函数	$f(z, z^*) : \mathbb{C} \rightarrow \mathbb{R}$	$f(\mathbf{z}, \mathbf{z}^*) : \mathbb{C}^n \rightarrow \mathbb{R}$	$f(\mathbf{Z}, \mathbf{Z}^*) : \mathbb{C}^{m \times n} \rightarrow \mathbb{R}$
平稳点	$\frac{\partial f(z, z^*)}{\partial z^*} \Big _{z=c} = 0$	$\frac{\partial f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^*} \Big _{\mathbf{z}=\mathbf{c}} = \mathbf{0}_{n \times 1}$	$\frac{\partial f(\mathbf{Z}, \mathbf{Z}^*)}{\partial \mathbf{Z}^*} \Big _{\mathbf{Z}=\mathbf{C}} = \mathbf{0}_{m \times n}$
局部极小点	$\mathbf{H}(f(c, c^*)) \succeq 0$	$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) \succeq 0$	$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) \succeq 0$
全局极小点	$\mathbf{H}(f(c, c^*)) \succ 0$	$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) \succ 0$	$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) \succ 0$
局部极大点	$\mathbf{H}(f(c, c^*)) \preceq 0$	$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) \preceq 0$	$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) \preceq 0$
全局极大点	$\mathbf{H}(f(c, c^*)) \prec 0$	$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) \prec 0$	$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) \prec 0$
鞍点	$\mathbf{H}(f(c, c^*))$ 不定	$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*))$ 不定	$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*))$ 不定

表中

$$\mathbf{H}(f(c, c^*)) = \begin{bmatrix} \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z} & \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^*} \\ \frac{\partial^2 f(z, z^*)}{\partial z \partial z} & \frac{\partial^2 f(z, z^*)}{\partial z \partial z^*} \end{bmatrix} \Big|_{z=c} \in \mathbb{C}^{2 \times 2} \quad (4.2.21)$$

$$\mathbf{H}(f(\mathbf{c}, \mathbf{c}^*)) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^T} & \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^H} \\ \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^T} & \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^H} \end{bmatrix} \Big|_{\mathbf{z}=\mathbf{c}} \in \mathbb{C}^{2n \times 2n} \quad (4.2.22)$$

$$\mathbf{H}(f(\mathbf{C}, \mathbf{C}^*)) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}^*) \partial (\text{vec } \mathbf{Z}^*)^T} \\ \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z})^T} & \frac{\partial^2 f(\mathbf{Z}, \mathbf{Z}^*)}{\partial (\text{vec } \mathbf{Z}) \partial (\text{vec } \mathbf{Z}^*)^T} \end{bmatrix} \Big|_{\mathbf{Z}=\mathbf{C}} \in \mathbb{C}^{2mn \times 2mn} \quad (4.2.23)$$

### 4.2.3 无约束最小化问题的梯度分析

给定一个实值目标函数  $f(\mathbf{w}, \mathbf{w}^*)$  或  $f(\mathbf{W}, \mathbf{W}^*)$ , 其无约束最小化问题的梯度分析可以归纳总结如下。

1. 共轭梯度矩阵决定最小化问题的闭式解。
2. 共轭梯度矩阵与 Hessian 矩阵给出局部极小点辨识的必要条件或充分条件。
3. 共轭梯度向量的负方向决定求解最小化问题的最速下降迭代算法。
4. Hessian 矩阵给出求解最小化问题的 Newton 算法。

下面依次对这些梯度分析展开讨论。

#### 1. 无约束最小化问题的闭式解

通过令目标函数的共轭梯度向量(或矩阵)为零向量(或矩阵), 可以求出无约束最小化问题的闭式解。

例 4.2.1 考察求解超定矩阵方程  $\mathbf{A}\mathbf{z} = \mathbf{b}$  的最小二乘方法。定义误差平方和

$$\begin{aligned} J(\mathbf{z}) &= \|\mathbf{A}\mathbf{z} - \mathbf{b}\|_2^2 = (\mathbf{A}\mathbf{z} - \mathbf{b})^H(\mathbf{A}\mathbf{z} - \mathbf{b}) \\ &= \mathbf{z}^H \mathbf{A}^H \mathbf{A}\mathbf{z} - \mathbf{z}^H \mathbf{A}^H \mathbf{b} - \mathbf{b}^H \mathbf{A}\mathbf{z} + \mathbf{b}^H \mathbf{b} \end{aligned}$$

为准则函数。令其共轭梯度向量  $\nabla_{\bar{z}^*} J(\bar{z}) = \mathbf{A}^H \mathbf{A} \bar{z} - \mathbf{A}^H \mathbf{b}$  等于零向量，易知：若  $\mathbf{A}^H \mathbf{A}$  非奇异，则

$$\bar{z} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b} \quad (4.2.24)$$

这就是超定矩阵方程  $\mathbf{A} \bar{z} = \mathbf{b}$  的最小二乘解。

**例 4.2.2** 考察求解超定矩阵方程  $\mathbf{A} z = \mathbf{b}$  的最大似然方法。定义对数似然函数

$$l(\hat{z}) = C - \frac{1}{\sigma^2} \mathbf{e}^H \mathbf{e} = C - \frac{1}{\sigma^2} (\mathbf{b} - \mathbf{A} \hat{z})^H (\mathbf{b} - \mathbf{A} \hat{z}) \quad (4.2.25)$$

式中， $C$  为一实常数。求对数似然函数

$$l(\hat{z}) = C - \frac{1}{\sigma^2} \mathbf{b}^H \mathbf{b} + \frac{1}{\sigma^2} \mathbf{b}^H \mathbf{A} \hat{z} + \frac{1}{\sigma^2} \hat{z}^H \mathbf{A}^H \mathbf{b} - \frac{1}{\sigma^2} \hat{z}^H \mathbf{A}^H \mathbf{A} \hat{z} \quad (4.2.26)$$

相对于  $z$  的共轭梯度，得

$$\nabla_{\bar{z}^*} l(\hat{z}) = \frac{1}{\sigma^2} \mathbf{A}^H \mathbf{b} - \frac{1}{\sigma^2} \mathbf{A}^H \mathbf{A} \hat{z}$$

令其等于零，得  $\mathbf{A}^H \mathbf{b} - \mathbf{A}^H \mathbf{A} z_{\text{opt}} = \mathbf{0}$  或  $\mathbf{A}^H \mathbf{A} z_{\text{opt}} = \mathbf{A}^H \mathbf{b}$ ，其中  $z_{\text{opt}}$  是使对数似然函数  $l(\hat{z})$  极大化的  $\hat{z}$  值。于是，若  $\mathbf{A}^H \mathbf{A}$  非奇异，则

$$z_{\text{opt}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b} \quad (4.2.27)$$

这就是矩阵方程  $\mathbf{A} w = \mathbf{b}$  的最大似然解。可见，矩阵方程  $\mathbf{A} w = \mathbf{b}$  的最大似然解与最小二乘解等价。

## 2. 局部极小点辨识的必要条件或充分条件

作为必要条件式 (4.1.39) 和充分条件式 (4.1.43) 在复向量情况下的推广，判断实目标函数  $f(z, z^*)$  的局部极小点的条件如下：

(1) 必要条件 若  $z_0$  或  $Z_0$  是  $f(z, z^*)$  或  $f(Z, Z^*)$  的局部极小点，则该函数在点  $z_0$  或  $Z_0$  的共轭梯度为零向量或零矩阵，并且全 Hessian 矩阵半正定，即

$$\left. \frac{\partial f(z, z^*)}{\partial z^*} \right|_{z=z_0} = \mathbf{0}, \quad \begin{bmatrix} \mathbf{H}_{z^*, z} & \mathbf{H}_{z^*, z^*} \\ \mathbf{H}_{z, z} & \mathbf{H}_{z, z^*} \end{bmatrix}_{z=z_0} \succeq 0 \quad (4.2.28)$$

或者

$$\left. \frac{\partial f(Z, Z^*)}{\partial Z^*} \right|_{Z=Z_0} = \mathbf{0}, \quad \begin{bmatrix} \mathbf{H}_{Z^*, z} & \mathbf{H}_{Z^*, z^*} \\ \mathbf{H}_{Z, z} & \mathbf{H}_{Z, z^*} \end{bmatrix}_{Z=Z_0} \succeq 0 \quad (4.2.29)$$

(2) 充分条件 若函数  $f(z, z^*)$  在  $z_0$  的共轭梯度向量为零向量，或者  $f(Z, Z^*)$  在  $Z_0$  的共轭梯度矩阵为零矩阵，并且全 Hessian 矩阵正定，即

$$\left. \frac{\partial f(z, z^*)}{\partial z^*} \right|_{z=z_0} = \mathbf{0}, \quad \begin{bmatrix} \mathbf{H}_{z^*, z} & \mathbf{H}_{z^*, z^*} \\ \mathbf{H}_{z, z} & \mathbf{H}_{z, z^*} \end{bmatrix}_{z=z_0} \succ 0 \quad (4.2.30)$$

或者

$$\left. \frac{\partial f(Z, Z^*)}{\partial Z^*} \right|_{Z=Z_0} = \mathbf{0}, \quad \begin{bmatrix} \mathbf{H}_{Z^*, z} & \mathbf{H}_{Z^*, z^*} \\ \mathbf{H}_{Z, z} & \mathbf{H}_{Z, z^*} \end{bmatrix}_{Z=Z_0} \succ 0 \quad (4.2.31)$$

则  $z_0$  是函数  $f(z, z^*)$  的严格局部极小点, 或  $Z_0$  是  $f(Z, Z^*)$  的严格局部极小点。

对于以复向量为变元的凸函数  $f(z, z^*)$ , 它的任何局部极小点  $z_0$  都是该函数的一个全局极小点。若凸函数  $f(z, z^*)$  是可微分的, 则满足  $\frac{\partial f(z, z^*)}{\partial z^*} \Big|_{z=z_0} = 0$  的平稳点  $z_0$  就是函数  $f(z, z^*)$  的一个全局极小点。

### 3. 实值目标函数的最速下降方向

一个以复矩阵为变元的实值目标函数的平稳点的确定存在

$$\frac{\partial f(Z, Z^*)}{\partial Z} \Big|_{Z=C} = O_{m \times n} \quad \text{或} \quad \frac{\partial f(Z, Z^*)}{\partial Z^*} \Big|_{Z=C} = O_{m \times n} \quad (4.2.32)$$

两种选择。那么, 在设计最优化问题的学习算法时, 应该选用哪一种梯度呢?为此, 需要引出曲率方向的定义。

**定义 4.2.1**<sup>[171]</sup> 当矩阵  $H$  是非线性函数  $f(x)$  的 Hessian 矩阵时, 称满足  $p^H H p > 0$  的向量  $p$  为函数  $f$  的正曲率方向 (direction of positive curvature), 满足  $p^H H p < 0$  的向量  $p$  为函数  $f$  的负曲率方向 (direction of negative curvature)。标量  $p^H H p$  则称为函数  $f$  沿着方向  $p$  的曲率 (curvature)。

曲率方向也就是函数的最大变化率方向。

**定理 4.2.1**<sup>[58]</sup> 令  $f(z)$  是复向量  $z$  的实值函数。通过将  $z$  和  $z^*$  视为独立的变元, 实目标函数  $f(z)$  的曲率方向由共轭梯度向量  $\nabla_{z^*} f(z)$  给出。

定理 4.2.1 表明, 共轭梯度向量  $\nabla_{z^*} f(z, z^*)$  或  $\nabla_{\text{vec}(Z^*)} f(Z, Z^*)$  的每个分量给出了目标函数  $f(z, z^*)$  或  $f(Z, Z^*)$  在该分量方向上的变化率:

- (1) 共轭梯度向量  $\nabla_{z^*} f(z, z^*)$  或  $\nabla_{\text{vec}(Z^*)} f(Z, Z^*)$  给出目标函数增长最快的方向;
- (2) 负共轭梯度向量  $-\nabla_{z^*} f(z, z^*)$  或  $-\nabla_{\text{vec}(Z^*)} f(Z, Z^*)$  给出目标函数最陡减小的方向。

因此, 求函数的最小值时, 沿着负的共轭梯度方向走, 可以最快到达极小点。这种优化算法称为梯度下降算法, 又称最速下降算法。

作为定理 4.2.1 的几何解释, 图 4.2.1 画出了函数  $f(z) = |z|^2$  在  $c_1$  和  $c_2$  两点的梯度和共轭梯度, 其中  $\nabla_{c_i^*} f = \frac{\partial f(z)}{\partial z^*} \Big|_{z=c_i}$ ,  $i = 1, 2$ 。

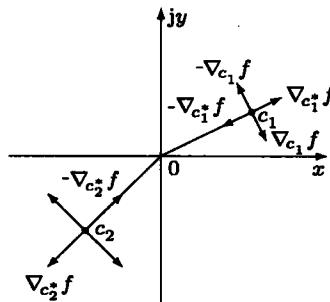


图 4.2.1 函数  $f(z) = |z|^2$  在  $c_1$  和  $c_2$  点的梯度与共轭梯度

显然, 只有共轭梯度的负方向指向函数的全局最小值  $z^* = 0$ 。因此, 在无约束的最小化问题中, 通常使用共轭梯度向量的负方向  $-\nabla_{z^*} f(z)$  作为更新方向:

$$z_k = z_{k-1} - \mu \nabla_{z^*} f(z), \quad \mu > 0 \quad (4.2.33)$$

就是说, 候补解在迭代过程中的校正量与目标函数的负共轭梯度成正比。上式称为优化问题候补解的学习算法。由于负共轭梯度向量总是指向目标函数减小的方向, 所以这种学习算法被称为最速下降法。

最速下降法中的常数  $\mu$  称为学习步长, 它决定候补解趋向最优解的收敛速率。

#### 4. 求解极小化问题的 Newton 算法

共轭梯度向量只是目标函数的一阶微分信息。如果进一步利用 Hessian 矩阵提供的目标函数的二阶微分信息, 则有望设计出性能更好的优化算法。利用 Hessian 矩阵设计的优化算法称为 Newton 法。

Newton 法是一种简单而有效的约束优化算法, 应用广泛。特别地, Newton 法已经成为现代内点法的一种代表性算法, 详见后述。

### 4.3 凸优化理论

4.2 节讨论了无约束优化问题, 本节将讨论约束优化问题。求解约束优化问题的基本思想是将其变为无约束优化问题。

#### 4.3.1 标准约束优化问题

考虑标准形式的约束最优化问题

$$\min_{\mathbf{x}} f_0(\mathbf{x}) \quad \text{subject to } f_i(\mathbf{x}) \leq 0, i = 1, \dots, m; \mathbf{A}\mathbf{x} = \mathbf{b} \quad (4.3.1)$$

或写作

$$\min_{\mathbf{x}} f_0(\mathbf{x}) \quad \text{subject to } f_i(\mathbf{x}) \leq 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, q \quad (4.3.2)$$

式中, subject to 表示“约束为”。有些文献则使用 such that (使得, 满足) 表示约束条件, 或者直接使用两者共用的缩略符号“s.t.”。本书统一采用 subject to。

约束优化问题中的变量  $\mathbf{x}$  为优化变量或决策变量, 函数  $f_0(\mathbf{x})$  称为目标函数 (或代价函数), 而

$$f_i(\mathbf{x}) \leq 0, \mathbf{x} \in \mathcal{I} \quad \text{和} \quad h_i(\mathbf{x}) = 0, \mathbf{x} \in \mathcal{E} \quad (4.3.3)$$

分别称为不等式约束条件和等式约束条件。其中,  $\mathcal{I}$  和  $\mathcal{E}$  分别是不等式约束函数和等式约束函数的定义域

$$\mathcal{I} = \bigcap_{i=1}^m \text{dom } f_i \quad \text{和} \quad \mathcal{E} = \bigcap_{i=1}^q \text{dom } h_i \quad (4.3.4)$$

不等式约束和等式约束合称显式约束 (explicit constraints)，无显式约束 ( $m = q = 0$ ) 的优化问题退化为无约束优化问题。

不等式约束  $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$  和等式约束  $h_i(\mathbf{x}) = 0, i = 1, \dots, q$  表示对  $\mathbf{x}$  的可能选择进行限制的  $m+q$  个严格要求或规定。目标函数  $f_0(\mathbf{x})$  表示选择  $\mathbf{x}$  所付出的代价。相反，负目标函数  $-f_0(\mathbf{x})$  则可以理解成选择  $\mathbf{x}$  所获得的价值或者收获的效益。因此，约束优化问题式 (4.3.2) 的解对应于选择  $\mathbf{x}$ ，以便在满足  $m+q$  个严格要求的所有被选  $\mathbf{x}$  中，使代价最小化或者使效益最大化。

约束优化问题式 (4.3.2) 的最优值记作  $p^*$ ，定义为目标函数  $f_0(\mathbf{x})$  的下确界

$$p^* = \inf\{f_0(\mathbf{x}) | f_i(\mathbf{x}) \leq 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, q\} \quad (4.3.5)$$

若  $p^* = \infty$ ，则称约束优化问题式 (4.3.2) 是不可行的 (找不到任何一个点  $\mathbf{x}$  满足约束条件)。若  $p^* = -\infty$ ，则约束优化问题式 (4.3.2) 是下无界的。下面是求解约束优化式 (4.3.2) 的几个关键问题：

- (1) 寻找使约束优化问题可行的点。
- (2) 寻找使约束优化问题达到最优值的点。
- (3) 避免或者转变下无界的优化问题。

满足所有不等式约束和等式约束的点  $\mathbf{x}$  称为一个可行点。所有可行点组成的集合称为可行域或可行集，定义为

$$\mathcal{F} \stackrel{\text{def}}{=} \mathcal{I} \cap \mathcal{E} = \{\mathbf{x} | f_i(\mathbf{x}) \leq 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, q\} \quad (4.3.6)$$

可行集以外的点统称非可行点 (infeasible point)。于是，目标函数的定义域  $\text{dom } f_0$  和可行域  $\mathcal{F}$  的交集

$$\mathcal{D} = \text{dom } f_0 \cap \bigcap_{i=1}^m \text{dom } f_i \cap \bigcap_{i=1}^q \text{dom } h_i = \text{dom } f_0 \cap \mathcal{F} \quad (4.3.7)$$

称为优化问题的定义域。

一个可行点  $\mathbf{x}$  是最优点，若  $f_0(\mathbf{x}) = p^*$ 。

寻找一个最优点有时可能是困难的。此时，可以寻找两种弱化的最优点。

(1) 局部最优点 一个可行点  $\mathbf{x}$  称为局部最优点，若存在一个常数  $\delta > 0$ ，使得

$$\begin{aligned} \min_{\mathbf{z}} \quad & f_0(\mathbf{z}) \\ \text{subject to} \quad & f_i(\mathbf{z}) \leq 0, i = 1, \dots, m; h_i(\mathbf{z}) = 0, i = 1, \dots, q \\ & \|\mathbf{z} - \mathbf{x}\|_2 < \delta \end{aligned}$$

(2) 次最优点 给定一个允许误差  $\varepsilon > 0$ ，一个可行点  $\mathbf{x}$  称为  $\varepsilon$ -次最优点，若

$$|f_0(\mathbf{x}) - f_0(\mathbf{x}^*)| = |f_0(\mathbf{x}) - p^*| \leq \varepsilon \quad (4.3.8)$$

获得约束优化问题的一个可行点  $\mathbf{x}$  并不困难，但问题是如何判断这个可行点是否是一个最优点。显然，如果直接根据最优点的定义作出判断，无疑是不切实际的。因此，希望有其他条件可用于直接判断。这些条件称为优化条件。

变分不等式 (variational inequality, VI) 给定一个 Banach 空间  $E$ ,  $E$  的一个子集  $K$  以及一个从  $K$  到  $E$  的对偶空间  $E^*$  的映射函数  $F : K \rightarrow E^*$ , 变分不等式问题的提法是<sup>[21]</sup>: 求一向量  $\mathbf{x}$  (称为变分不等式问题的一个解), 使其满足

$$\langle F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \geq 0, \quad \forall \mathbf{y} \in K \quad (4.3.9)$$

求解一个变分不等式问题通常包含以下三个步骤<sup>[514]</sup>:

- (1) 证明解的存在性, 这意味着问题的数学正确性。
- (2) 证明解的唯一性, 这意味着变分不等式可以描述物理现象, 具有物理正确性。
- (3) 求变分不等式的解。

当映射函数直接取作凸优化问题的目标函数的梯度向量时, 变分不等式问题便简化为最小值原理 (minimum principle), 它给出凸优化问题的最优解必须满足的条件。

**最小值原理** 若  $f(\mathbf{x})$  是凸优化问题的目标函数, 则可行点  $\mathbf{x}$  是最优解点的充分必要条件是

$$\langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \geq 0, \quad \forall \mathbf{y} \in K \quad (4.3.10)$$

然而, 当映射函数不可能表示成某个“潜在函数”的梯度向量时, 变分不等式问题与一个优化问题是不同的。事实上, 并不是所有的连续函数  $F$  都可以表示成一个合适标量函数的梯度。

通常, 一个约束最优化问题是很难求解的, 特别是当  $\mathbf{x}$  中的决策变量个数很大时, 问题的求解尤为困难。这一困难的产生有以下几个主要原因<sup>[232]</sup>:

- (1) 优化问题的定义区域可能弥漫着局部最优解。
- (2) 可能非常难于求出一个可行点。
- (3) 一般优化算法中使用的停止准则往往在约束优化问题中失效。
- (4) 优化算法的收敛速率可能很差。
- (5) 数值问题可能使极小化算法要么完全停止不前, 要么徘徊不止, 无法正常收敛。

约束优化问题的上述困难可以借助凸优化技术加以克服。本质上, 凸优化就是凸集约束下凸(或凹)目标函数的极小化(或极大化)。凸优化是最优化、凸分析和数值计算三门学科的融合。

### 4.3.2 凸集与凸函数

先介绍凸优化的有关基本概念。考虑  $n$  维实向量空间  $\mathbb{R}^n$ 。

**定义 4.3.1** 一个集合  $S \in \mathbb{R}^n$  称为凸集(合), 若对任意两个点  $\mathbf{x}, \mathbf{y} \in S$ , 连接它们的线段也在集合  $S$  内, 即

$$\mathbf{x}, \mathbf{y} \in S, \quad \theta \in [0, 1] \implies \theta\mathbf{x} + (1 - \theta)\mathbf{y} \in S \quad (4.3.11)$$

图 4.3.1 画出了凸集和非凸集的示意图。

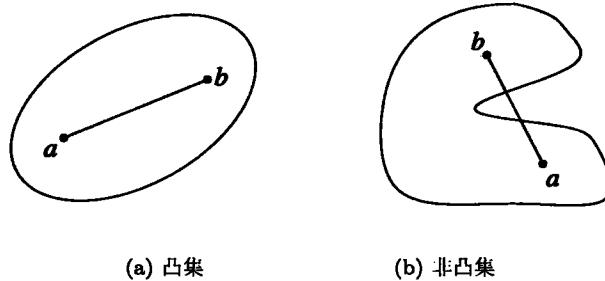


图 4.3.1 凸集与非凸集

许多熟悉的集合都是凸集，例如单位球体 (unit ball)  $S = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$ 。然而，单位球面 (unit sphere)  $S = \{\mathbf{x} : \|\mathbf{x}\|_2 = 1\}$  却不是凸集，因为连接球面上两点的线段显然不在球面上。

凸集具有以下重要性质<sup>[363, Theorem 2.2.4]</sup>：令  $S_1 \subseteq \mathbb{R}^n$  和  $S_2 \subseteq \mathbb{R}^m$  是凸集，并且  $\mathcal{A}(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$  为线性算子  $\mathcal{A}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$ ，则

- (1) 交集  $S_1 \cap S_2 = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \in S_1, \mathbf{x} \in S_2\}$  (其中  $m = n$ ) 为凸集。
- (2) 和集  $S_1 + S_2 = \{\mathbf{z} = \mathbf{x} + \mathbf{y} | \mathbf{x} \in S_1, \mathbf{y} \in S_2\}$  (其中  $m = n$ ) 为凸集。
- (3) 直和  $S_1 \oplus S_2 = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n+m} | \mathbf{x} \in S_1, \mathbf{y} \in S_2\}$  为凸集。
- (4) 锥包 (conic hull)  $\mathcal{K}(S_1) = \{\mathbf{z} \in \mathbb{R}^n | \mathbf{z} = \beta\mathbf{x}, \mathbf{x} \in S_1, \beta \geq 0\}$  为凸集。
- (5) 仿射象 (affine image)  $\mathcal{A}(S_1) = \{\mathbf{y} \in \mathbb{R}^m | \mathbf{y} = \mathcal{A}(\mathbf{x}), \mathbf{x} \in S_1\}$  为凸集。
- (6) 逆仿射象  $\mathcal{A}^{-1}(S_2) = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} = \mathcal{A}^{-1}(\mathbf{y}), \mathbf{y} \in S_2\}$  为凸集。
- (7) 下列凸包 (convex hull) 为凸集

$$\text{conv}(S_1, S_2) = \{\mathbf{z} \in \mathbb{R}^n | \mathbf{z} = \alpha\mathbf{x} + (1 - \alpha)\mathbf{y}, \mathbf{x} \in S_1, \mathbf{y} \in S_2; \alpha \in [0, 1]\}$$

凸集最重要的性质是性质 (1) 的推广：任意多个（甚至不可数）凸集的交集仍然是凸集。例如，两个凸集——单位球体和非负象限 (nonnegative orthant)  $\mathbb{R}_+^n$  的交集  $S = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1, x_i \geq 0\}$  保留了凸性。然而，两个凸集的并集却往往是非凸的。例如，两个单位球体  $S_1 = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq 1\}$  和  $S_2 = \{\mathbf{x} : \|\mathbf{x} - 3 \times 1\|_2 \leq 1\}$  (其中 1 表示所有元素都等于 1 的向量) 都是凸集，但它们的并集  $S_1 \cup S_2$  却不是一个凸集，因为连接这两个球体任意两点的线段显然都不在并集  $S_1 \cup S_2$  内。

给定向量  $\mathbf{x} \in \mathbb{R}^n$  和  $\rho > 0$ ，则

$$B_o(\mathbf{x}, \rho) = \{\mathbf{y} \in \mathbb{R}^n | \|\mathbf{y} - \mathbf{x}\|_2 < \rho\} \quad (4.3.12)$$

$$B_c(\mathbf{x}, \rho) = \{\mathbf{y} \in \mathbb{R}^n | \|\mathbf{y} - \mathbf{x}\|_2 \leq \rho\} \quad (4.3.13)$$

分别称为以  $\mathbf{x}$  为中心， $\rho$  为半径的开球体 (open ball) 和闭球体 (closed ball)。

一个凸集  $S \subseteq \mathbb{R}^n$  称为凸锥 (convex cone), 若从原点发出, 并且通过该集合中任意一点的所有射线以及连接这些射线的任意两点的所有线段仍然在该凸集中, 即

$$\mathbf{x}, \mathbf{y} \in S, \lambda, \mu \geq 0 \implies \lambda\mathbf{x} + \mu\mathbf{y} \in S \quad (4.3.14)$$

非负象限  $\mathbb{R}_+^n = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \succeq 0\}$  是一个凸锥。半正定矩阵  $\mathbf{X} \succeq 0$  的集合  $S_+^n = \{\mathbf{X} \in \mathbb{R}^{n \times n} : \mathbf{X} \succeq 0\}$  也是一个凸锥, 因为任意个半正定矩阵的正的组合仍然是半正定的。因此, 常将  $S_+^n$  称为半正定锥 (positive semidefinite cone)。

**定义 4.3.2** 向量函数  $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^m$  称为仿射函数 (affine function), 若它具有线性加常数向量的形式

$$f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b} \quad (4.3.15)$$

类似地, 矩阵函数  $F(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^{p \times q}$  称为仿射函数, 若它具有形式

$$F(\mathbf{x}) = \mathbf{A}_0 + x_1 \mathbf{A}_1 + \cdots + x_n \mathbf{A}_n \quad (4.3.16)$$

式中  $\mathbf{A}_i \in \mathbb{R}^{p \times q}$ 。仿射函数有时也粗略地称为线性函数。

**定义 4.3.3**<sup>[232]</sup> 给定  $n \times 1$  向量点  $\mathbf{x}_i \in \mathbb{R}^n$  和实数  $\theta_i \in \mathbb{R}$ , 则  $\mathbf{y} = \theta_1 \mathbf{x}_1 + \cdots + \theta_k \mathbf{x}_k$  称为:

- (1) 线性组合 (对任意实数  $\theta_i$ );
- (2) 仿射组合 (affine combination), 若  $\sum_i \theta_i = 1$ ;
- (3) 凸组合 (convex combination), 若  $\sum_i \theta_i = 1$ , 并且所有  $\theta_i \geq 0$ ;
- (4) 锥组合 (conic combination), 若  $\theta_i \geq 0, i = 1, \dots, k$ 。

令  $\mathcal{A}$  是一任意标签集合 (可能包含无穷多个标签), 并且  $\{S_\alpha | \alpha \in \mathcal{A}\}$  表示一批集合, 则这些集合的交集具有以下重要性质<sup>[232]</sup>

$$S_\alpha \text{是} \begin{pmatrix} \text{子空间} \\ \text{仿射函数} \\ \text{凸集} \\ \text{凸锥} \end{pmatrix} \implies \bigcap_{\alpha \in \mathcal{A}} S_\alpha \text{是} \begin{pmatrix} \text{子空间} \\ \text{仿射函数} \\ \text{凸集} \\ \text{凸锥} \end{pmatrix} \quad (4.3.17)$$

**定义 4.3.4**<sup>[363]</sup> 给定一个凸集  $S \subseteq \mathbb{R}^n$  和函数  $f : S \rightarrow \mathbb{R}$ , 则:

(1) 函数  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  称为凸函数 (convex function), 当且仅当  $S = \text{dom}(f)$  是凸集, 并且对于所有  $\mathbf{x}, \mathbf{y} \in S$  和每一个标量  $\alpha \in (0, 1)$ , 函数满足 Jensen 不等式

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}) \quad (4.3.18)$$

(2) 函数  $f(\mathbf{x})$  称为严格凸函数 (strictly convex function), 当且仅当  $S = \text{dom}(f)$  是凸集, 并且对于所有  $\mathbf{x}, \mathbf{y} \in S$  和每一个标量  $\alpha \in (0, 1)$ , 函数满足不等式

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) < \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}) \quad (4.3.19)$$

在凸优化中，常要求目标函数为强凸函数 (strongly convex function)，它有以下三种定义：

(1) 函数  $f(\mathbf{x})$  称为强凸函数，若<sup>[363]</sup>

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}) - \frac{\mu}{2}\alpha(1 - \alpha)\|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.3.20)$$

对所有  $\mathbf{x}, \mathbf{y} \in S$  及  $\alpha \in [0, 1]$  成立。

(2) 函数  $f(\mathbf{x})$  称为强凸函数，若<sup>[46]</sup>

$$(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^\top (\mathbf{x} - \mathbf{y}) \geq \mu \|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.3.21)$$

对所有  $\mathbf{x}, \mathbf{y} \in S$  及某个  $\mu > 0$  成立。

(3) 函数  $f(\mathbf{x})$  称为强凸函数，若<sup>[363]</sup>

$$f(\mathbf{y}) \geq f(\mathbf{x}) + [\nabla f(\mathbf{x})]^\top (\mathbf{y} - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|_2^2 \quad (4.3.22)$$

上述三种定义中，常数  $\mu (> 0)$  称为强凸函数  $f(\mathbf{x})$  的凸性参数 (convexity parameter)。三种凸函数之间存在以下的关系

$$\text{强凸函数} \Rightarrow \text{严格凸函数} \Rightarrow \text{凸函数} \quad (4.3.23)$$

下面是关于拟凸函数的定义。

**定义 4.3.5**<sup>[469]</sup> 函数  $f(\mathbf{x})$  称为拟凸函数 (quasi-convex function)，若对所有  $\mathbf{x}, \mathbf{y} \in E$  和  $\alpha \in [0, 1]$ ，不等式

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \max\{f(\mathbf{x}), f(\mathbf{y})\} \quad (4.3.24)$$

成立。函数  $f(\mathbf{x})$  称为强拟凸函数 (strongly quasi-convex function)，若对所有  $\mathbf{x}, \mathbf{y} \in E$ ,  $\mathbf{x} \neq \mathbf{y}$  和  $\alpha \in (0, 1)$ ，严格不等式

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) < \max\{f(\mathbf{x}), f(\mathbf{y})\} \quad (4.3.25)$$

成立。函数  $f(\mathbf{x})$  称为严格拟凸函数 (strictly quasi-convex function)，若严格不等式 (4.3.25) 对所有  $\mathbf{x}, \mathbf{y} \in E$ ,  $f(\mathbf{x}) \neq f(\mathbf{y})$  和  $\alpha \in (0, 1)$  成立。

### 4.3.3 凸函数辨识的充分必要条件

给定一个定义在凸集  $S$  上的目标函数  $f(\mathbf{x}) : S \rightarrow \mathbb{R}$ ，一个自然会问的问题是如何判断该函数是否是凸函数？凸函数辨识的方法分为一阶梯度辨识法和二阶梯度辨识法。

**定义 4.3.6**<sup>[443]</sup> 给定一个凸集  $S \in \mathbb{R}^n$ ，则映射函数  $\mathbf{F}(\mathbf{x}) : S \rightarrow \mathbb{R}^n$  称为

(1) 在凸集  $S$  上单调 (monotone)

$$\langle \mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq 0, \quad \forall \mathbf{x}, \mathbf{y} \in S \quad (4.3.26)$$

(2) 在凸集  $S$  上严格单调 (strictly monotone), 若

$$\langle \mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle > 0, \quad \forall \mathbf{x}, \mathbf{y} \in S \text{ 和 } \mathbf{x} \neq \mathbf{y} \quad (4.3.27)$$

(3) 在凸集  $S$  上强单调 (strongly monotone), 若

$$\langle \mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu \|\mathbf{x} - \mathbf{y}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in S \quad (4.3.28)$$

特别地, 若将函数  $f(\mathbf{x})$  的梯度向量取作映射函数, 即  $\mathbf{F}(\mathbf{x}) = \nabla_{\mathbf{x}} f(\mathbf{x})$ , 则可以得到凸函数辨识的一阶充分必要条件。

### 1. 凸函数辨识的一阶充分必要条件

**定理 4.3.1**<sup>[443]</sup> 令  $f : S \rightarrow \mathbb{R}$  是一个定义在  $n$  维向量空间  $\mathbb{R}^n$  内的凸集  $S$  上的函数, 并且可微分, 则

$$f(\mathbf{x}) \text{ 凸} \Leftrightarrow \langle \nabla_{\mathbf{x}} f(\mathbf{x}) - \nabla_{\mathbf{x}} f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq 0, \quad \forall \mathbf{x}, \mathbf{y} \in S \quad (4.3.29)$$

$$f(\mathbf{x}) \text{ 严格凸} \Leftrightarrow \langle \nabla_{\mathbf{x}} f(\mathbf{x}) - \nabla_{\mathbf{x}} f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle > 0, \quad \forall \mathbf{x}, \mathbf{y} \in S \text{ 和 } \mathbf{x} \neq \mathbf{y} \quad (4.3.30)$$

$$f(\mathbf{x}) \text{ 强凸} \Leftrightarrow \langle \nabla_{\mathbf{x}} f(\mathbf{x}) - \nabla_{\mathbf{x}} f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu \|\mathbf{x} - \mathbf{y}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in S \quad (4.3.31)$$

**定理 4.3.2**<sup>[55]</sup> 若  $f : S \rightarrow \mathbb{R}$  在凸定义域是可微分的, 则  $f$  为凸函数, 当且仅当

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla_{\mathbf{x}} f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \quad (4.3.32)$$

### 2. 凸函数辨识的二阶充分必要条件

**定理 4.3.3**<sup>[328]</sup> 令  $f : S \rightarrow \mathbb{R}$  是一个定义在  $n$  维向量空间  $\mathbb{R}^n$  内的凸集  $S$  上的函数, 并且可二次微分, 则  $f(\mathbf{x})$  是凸函数, 当且仅当 Hessian 矩阵半正定

$$\mathbf{H}_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \succeq 0, \quad \forall \mathbf{x} \in S \quad (4.3.33)$$

**注释** 令  $f : S \rightarrow \mathbb{R}$  是一个定义在  $n$  维向量空间  $\mathbb{R}^n$  内的凸集  $S$  上的函数, 并且可二次微分, 则  $f(\mathbf{x})$  是严格凸函数, 当且仅当 Hessian 矩阵正定

$$\mathbf{H}_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \succ 0, \quad \forall \mathbf{x} \in S \quad (4.3.34)$$

与严格极小点的充分条件要求 Hessian 矩阵在  $\mathbf{c}$  一点正定不同, 这里要求 Hessian 矩阵在整个凸集  $S$  的所有点均正定。

下面的基本性质对于判断一个函数的凸性非常有用<sup>[232]</sup>:

(1) 函数  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  是凸函数, 当且仅当它在所有线段上是凸的, 即  $\tilde{f}(t) \triangleq f(\mathbf{x}_0 + t\mathbf{h})$  对  $t \in \mathbb{R}$  和所有  $\mathbf{x}_0, \mathbf{h} \in \mathbb{R}^n$  都是凸的。

(2) 凸函数的非负求和是凸函数

$$\alpha_1, \alpha_2 \geq 0 \text{ 且 } f_1(\mathbf{x}), f_2(\mathbf{x}) \text{ 为凸函数} \implies \alpha_1 f_1(\mathbf{x}) + \alpha_2 f_2(\mathbf{x}) \text{ 是凸函数}$$

(3) 凸函数的无穷求和、积分为凸函数

$$p(y) \geq 0, q(\mathbf{x}, y) \text{ 在 } \mathbf{x} \in S \text{ 是凸函数} \implies \int p(y)q(\mathbf{x}, y)dy \text{ 在 } \mathbf{x} \in S \text{ 是凸函数}$$

(4) 凸函数各点的上确界(最大值)为凸函数

$$f_\alpha(\mathbf{x}) \text{ 为凸函数} \implies \sup_{\alpha \in A} f_\alpha(\mathbf{x}) \text{ 是凸函数}$$

(5) 凸函数的仿射变换为凸函数

$$f(\mathbf{x}) \text{ 为凸函数} \implies f(A\mathbf{x} + \mathbf{b}) \text{ 为凸函数}$$

值得指出的是,除  $L_0$  范数以外,向量的所有范数

$$\|\mathbf{x}\|_p = \left( \sum_{i=1}^n |x_i|^p \right)^{1/p}, p \geq 1; \quad \|\mathbf{x}\|_\infty = \max_i |x_i| \quad (4.3.35)$$

都是凸函数。

#### 4.3.4 凸优化方法及其梯度分析

目标函数为凸函数,并且其定义域为凸集的优化问题称为无约束凸优化问题。目标函数和不等式约束函数均为凸函数,等式约束函数为仿射函数,并且定义域为凸集的优化问题则称为约束凸优化问题。

根据目标函数是否为平滑函数,凸优化问题分为平滑凸优化问题和非平滑凸优化问题。如果凸结构被利用,则相应的优化方法称为结构优化方法 (structural optimization method)。不使用任何凸结构的优化称为黑盒优化 (black-box optimization)。业已证明<sup>[364]</sup>,结构优化的梯度型算法优于黑盒优化的梯度型算法一个数量级。近几年的研究表明,在某些情况下,黑盒优化方法是不可替代的。这主要是因为:凸问题的结构太复杂,以至于很难构造一个好的自我和谐的障碍函数 (self-concordant barrier) 以及难于应用平滑技术<sup>[364]</sup>。本书主要讨论黑盒优化方法。

**定理 4.3.4**<sup>[372,p.16]</sup> 无约束凸函数  $f(\mathbf{x})$  的任何局部极小点  $\mathbf{x}^*$  都是该函数的一个全局极小点。若凸函数  $f(\mathbf{x})$  是可微分的,则满足  $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = 0$  的平稳点  $\mathbf{x}^*$  是  $f(\mathbf{x})$  的一个全局极小点。

**引理 4.3.1**<sup>[366]</sup> 如果  $f(\mathbf{x})$  是强凸函数,则极小化问题  $\min_{\mathbf{x} \in Q} f(\mathbf{x})$  是可解的,且其解  $\mathbf{x}$  是唯一的,并有

$$f(\mathbf{x}) \geq f(\mathbf{x}^*) + \frac{1}{2}\mu\|\mathbf{x} - \mathbf{x}^*\|_2^2, \quad \forall \mathbf{x} \in Q \quad (4.3.36)$$

其中  $\mu$  是强凸函数  $f(\mathbf{x})$  的凸性参数。

定理 4.3.4 表明,任何一个约束极小化问题如果能够转变成一个无约束的凸优化问题,则无约束凸优化问题的任何一个局部极小点都是原约束极小化问题的一个全局极小点。引理 4.3.1 进一步表明,如果转变后的无约束极小化问题的目标函数为强凸函数,则

极小化问题的任何一个平稳点都是原约束极小化问题的一个全局极小点。这为求解约束优化问题指明了方向：将它转变为一个无约束的凸优化问题。

考虑标准约束极小化问题式 (4.3.2)，并作如下假定：

- (1) 不等式约束函数  $f_i(\mathbf{x}), i = 1, \dots, m$  均为凸函数。
- (2) 等式约束函数  $h_i(\mathbf{x})$  具有仿射函数形式  $h(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}$ 。
- (3) 原始目标函数  $f_0(\mathbf{x})$  为平滑函数 (可微分)，但不是凸函数。

利用 Lagrangian 乘子法 (有时也称 Lagrangian 松弛法)，约束优化问题式 (4.3.2) 可以松弛为无约束优化问题

$$\min L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x}) \quad (4.3.37)$$

式中， $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  称为 Lagrangian 函数； $\lambda_i, i = 1, \dots, m$  和  $\nu_i, i = 1, \dots, q$  分别是针对不等式约束  $f_i(\mathbf{x}) \leq 0$  和等式约束  $h_i(\mathbf{x}) = 0$  的 Lagrangian 乘子， $\boldsymbol{\lambda}, \boldsymbol{\nu}$  为 Lagrangian 乘子向量。向量  $\mathbf{x}$  有时称为优化变量 (optimization variable) 或决策变量 (decision variable) 或原始变量 (primal variable)，而  $\boldsymbol{\lambda}$  也称对偶变量 (dual variable)。原约束优化问题式 (4.3.2) 称为原始问题，而无约束优化问题式 (4.3.37) 则称为对偶问题。

这里对 Lagrangian 乘子  $\lambda_i$  作一个关键的非负性约束： $\lambda_i \geq 0, i = 1, \dots, m$ ；而对另一个 Lagrangian 乘子  $\nu_i, i = 1, \dots, q$ ，则不作任何约束。

记  $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_m]^T$ ,  $\boldsymbol{\nu} = [\nu_1, \dots, \nu_q]^T$ ，则 Lagrangian 乘子  $\lambda_i$  的非负性约束可以用向量的分量不等式表示为  $\boldsymbol{\lambda} \succeq \mathbf{0}$ 。

观察式 (4.3.37) 知，在不等式  $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$  的约束下，当  $\lambda_i$  取很大的正值时，式 (4.3.37) 的第 2 项可能趋于负无穷，从而导致 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  负无穷。因此，需要将 Lagrangian 函数极大化

$$J_1(\mathbf{x}) = \max_{\boldsymbol{\lambda} \succeq \mathbf{0}, \boldsymbol{\nu}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \max_{\boldsymbol{\lambda} \succeq \mathbf{0}, \boldsymbol{\nu}} \left( f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x}) \right) \quad (4.3.38)$$

无约束极大化问题式 (4.3.38) 仍然存在一个问题：无法避免违法约束  $f_i(\mathbf{x}) > 0$ 。这有可能导致  $J_1(\mathbf{x})$  正无穷大，即有

$$J_1(\mathbf{x}) = \begin{cases} f_0(\mathbf{x}), & \text{若 } \mathbf{x} \text{ 满足原始全部约束} \\ (f_0(\mathbf{x}), +\infty), & \text{否则} \end{cases} \quad (4.3.39)$$

由式 (4.3.39) 易知，为了得到全部不等式和等式约束条件下原始目标函数  $f(\mathbf{x})$  的极小解  $\min_{\mathbf{x}} f_0(\mathbf{x}) = f_0(\mathbf{x}^*)$ ，必须将函数  $J_1(\mathbf{x})$  极小化

$$J_P(\mathbf{x}) = \min_{\mathbf{x}} J_1(\mathbf{x}) = \min_{\mathbf{x}} \max_{\boldsymbol{\lambda} \succeq \mathbf{0}, \boldsymbol{\nu}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \quad (4.3.40)$$

这是一个极小-极大化问题，其解就是 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  的上确界 (supremum) 即最小上界，故有

$$J_P(\mathbf{x}) = \sup \left( f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x}) \right) \quad (4.3.41)$$

这是原始约束极小化问题变成无约束极小化问题后的代价函数，简称原始代价函数。

由式(4.3.41)和式(4.3.39)易知，原始约束极小化问题的最优值

$$p^* = J_P(\mathbf{x}^*) = \min_{\mathbf{x}} f_0(\mathbf{x}) = f_0(\mathbf{x}^*) \quad (4.3.42)$$

简称最优原始值(optimal primal value)。

问题是：一个非凸目标函数的极小化不能转化为另一个凸函数的极小化。因此，若  $f_0(\mathbf{x})$  不是凸函数，则即使我们设计了一种优化算法，可以得到原始代价函数的某个局部极值点  $\tilde{\mathbf{x}}$ ，也不能保证它是一个全局极值点。

幸运的是，一个凸函数  $f(\mathbf{x})$  的极小化与凹函数  $-f(\mathbf{x})$  的极大化等价。基于凸函数极小化与凹函数极大化之间的这一对偶关系，容易引出解决非凸函数优化问题的对偶方法：将非凸目标函数的极小化转换成凹目标函数的极大化。

为此，考虑由 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  构造另一个目标函数

$$J_2(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \min_{\mathbf{x}} \left( f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x}) \right) \quad (4.3.43)$$

由式(4.3.43)知

$$\min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \begin{cases} \min_{\mathbf{x}} f_0(\mathbf{x}), & \text{若 } \mathbf{x} \text{ 满足原始全部约束} \\ (-\infty, \min_{\mathbf{x}} f_0(\mathbf{x})), & \text{否则} \end{cases} \quad (4.3.44)$$

其极大化函数

$$J_D(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \max_{\mathbf{x} \geq 0, \boldsymbol{\nu}} J_2(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \max_{\mathbf{x} \geq 0, \boldsymbol{\nu}} \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \quad (4.3.45)$$

称为原始问题的对偶目标函数，它是 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  的极大-极小化问题。

由于 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  的极大-极小化问题就是该函数的下确界(infimum)即最大下界，故有

$$J_D(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf \left( f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x}) \right) \quad (4.3.46)$$

由式(4.3.46)定义的对偶目标函数具有以下特点：

- (1) 对偶目标函数  $J_D(\boldsymbol{\lambda}, \boldsymbol{\nu})$  是 Lagrangian 函数  $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  的极大-极小化函数即最大下界。
- (2) 对偶目标函数  $J_D(\boldsymbol{\lambda}, \boldsymbol{\nu})$  是极大化的目标函数，因此它不是代价函数，而是价值或收益函数。
- (3) 对偶目标函数  $J_D(\boldsymbol{\lambda}, \boldsymbol{\nu})$  是下无界的：其下界为  $-\infty$ 。因此， $J_D(\boldsymbol{\lambda}, \boldsymbol{\nu})$  是变元  $\mathbf{x}$  的凹函数，即使  $f_0(\mathbf{x})$  不是凸函数。

根据定理 4.3.4 知，凹函数的任何一个局部极值点都是一个全局极值点。因此，标准约束极小化问题式(4.3.2)的算法设计变成了对偶目标函数的无约束极大化算法的设计。

由于约束极小化问题通过 Lagrangian 乘子法, 变成了无约束凹函数的极大化问题, 所以常称这一方法为 Lagrangian 对偶法。

记对偶目标函数的最优值 (简称为最优对偶值) 为

$$d^* = J_D(\lambda^*, \nu^*) \quad (4.3.47)$$

由式 (4.3.44) 和式 (4.3.45) 立即知

$$d^* \leq \min_{\mathbf{x}} f_0(\mathbf{x}) = p^* \quad (4.3.48)$$

最优原始值与最优对偶值之差  $p^* - d^*$  称为原始极小化问题与对偶极大化问题的对偶 (性) 间隙 (duality gap)。

式 (4.3.48) 是 Lagrangian 函数  $L(\mathbf{x}, \lambda, \nu)$  的极大-极小化与极小-极大化之间的关系。事实上, 对于任何一个非负的实值函数  $f(\mathbf{x}, \mathbf{y})$ , 其极大-极小化与极小-极大化之间都存在以下不等式关系 (见习题 4.15)

$$\max_{\mathbf{x}} \min_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) \leq \min_{\mathbf{y}} \max_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) \quad (4.3.49)$$

若  $d^* \leq p^*$ , 则称 Lagrangian 对偶法具有弱对偶性 (weak duality); 而当  $d^* = p^*$  时, 则称 Lagrangian 对偶法满足强对偶性 (strong duality)。

给定一允许对偶间隙  $\varepsilon$ , 满足

$$p^* - f_0(\mathbf{x}) \leq \varepsilon \quad (4.3.50)$$

的点  $\mathbf{x}$  和  $(\lambda, \nu)$  分别称为对偶凹极大化问题的  $\varepsilon$ -次最优原始点和  $\varepsilon$ -次最优对偶点。

令  $\mathbf{x}^*$  和  $(\lambda^*, \nu^*)$  分别表示具有零对偶间隙  $\varepsilon = 0$  的任意原始最优点和对偶最优点。由于  $\mathbf{x}^*$  使 Lagrangian 目标函数  $L(\mathbf{x}, \lambda^*, \nu^*)$  在所有原始可行点  $\mathbf{x}$  中最小化, 所以 Lagrangian 目标函数在点  $\mathbf{x}^*$  的梯度向量必然等于零向量

$$\nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{i=1}^q \nu_i^* \nabla h_i(\mathbf{x}^*) = \mathbf{0}$$

于是, Lagrangian 对偶无约束优化问题的 Karush-Kuhn-Tucker (KKT) 条件 (局部极小解的一阶必要条件) 为<sup>[372]</sup>

$$\left. \begin{array}{l} f_i(\mathbf{x}^*) \leq 0, \quad i = 1, \dots, m \quad (\text{原始不等式约束}) \\ h_i(\mathbf{x}^*) = 0, \quad i = 1, \dots, q \quad (\text{原始等式约束}) \\ \lambda_i^* \geq 0, \quad i = 1, \dots, m \quad (\text{非负性}) \\ \lambda_i^* f_i(\mathbf{x}^*) = 0, \quad i = 1, \dots, m \quad (\text{互补松弛性}) \\ \nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{i=1}^q \nu_i^* \nabla h_i(\mathbf{x}^*) = \mathbf{0} \end{array} \right\} \quad (4.3.51)$$

满足 KKT 条件的点  $\mathbf{x}$  称为 KKT 点。

**注释** 如果约束优化问题式 (4.3.2) 中的不等式约束  $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$  改变为  $c_i(\mathbf{x}) \geq 0, i = 1, \dots, m$ , 则 Lagrangian 函数需修改为

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) - \sum_{i=1}^m \lambda_i c_i(\mathbf{x}) + \sum_{i=1}^q \nu_i h_i(\mathbf{x})$$

并且 KKT 条件式 (4.3.51) 中所有的不等式约束函数  $f_i(\mathbf{x})$  均需要替换为  $-c_i(\mathbf{x})$ 。

下面对各个 KKT 条件进行解读。

- (1) 第 1 个和第 2 个 KKT 条件分别是原始不等式和等式约束条件。
- (2) 第 3 个 KKT 条件是 Lagrangian 乘子  $\lambda_i$  的非负性条件, 它是 Lagrangian 对偶法的一个关键约束。
- (3) 第 4 个 KKT 条件 (互补松弛性, complementary slackness) 也称双互补性 (dual complementary), 是 Lagrangian 对偶法的另一个关键约束。这一条件意味着, 对于违法约束  $f_i(\mathbf{x}) > 0$ , 对应的 Lagrangian 乘子  $\lambda_i$  必须等于零, 从而可以完全避免违法约束。这一作用宛如在不等式约束条件的边界  $f_i(\mathbf{x}) = 0, i = 1, \dots, m$  树立起一道障碍, 阻止违法约束  $f_i(\mathbf{x}) > 0$  的发生。

- (4) 第 5 个 KKT 条件是极小化  $\min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$  的平稳点条件。

以下是运用 Lagrangian 对偶法需要注意的几个问题。

### 1. 线性无关约束限制

**定义 4.3.7** 对于不等式约束  $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$ , 若在点  $\bar{\mathbf{x}}$  有  $f_i(\bar{\mathbf{x}}) = 0$ , 则称第  $i$  个约束是在  $\bar{\mathbf{x}}$  点的积极约束 (active constraint); 若  $f_i(\bar{\mathbf{x}}) < 0$ , 则称第  $i$  个约束是在  $\bar{\mathbf{x}}$  点的非积极约束 (inactive constraint)。若  $f_i(\bar{\mathbf{x}}) > 0$ , 则称第  $i$  个约束是在  $\bar{\mathbf{x}}$  点的违法约束 (violated constraint)。在  $\bar{\mathbf{x}}$  点的所有积极约束的指标集  $\mathcal{A}(\bar{\mathbf{x}}) = \{i | f_i(\bar{\mathbf{x}}) = 0\}$  称为  $\bar{\mathbf{x}}$  点的作用集 (active set)。

令  $m$  个不等式约束  $f_i(\mathbf{x}), i = 1, \dots, m$  在某个 KKT 点  $\mathbf{x}^*$  共有  $k$  个积极约束  $f_{A1}(\mathbf{x}^*), \dots, f_{Ak}(\mathbf{x}^*)$  和  $m - k$  个非积极约束。

为了满足 KKT 条件中的互补性  $\lambda_i f_i(\mathbf{x}^*) = 0$ , 与非积极约束  $f_i(\mathbf{x}^*) < 0$  对应的 Lagrangian 乘子  $\lambda_i^*$  必须等于零。这意味着, 式 (4.3.51) 中的最后一个 KKT 条件变为

$$\nabla f_0(\mathbf{x}^*) + \sum_{i \in \mathcal{A}} \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{i=1}^q \nu_i^* \nabla h_i(\mathbf{x}^*) = \mathbf{0}$$

或者

$$\begin{bmatrix} \frac{\partial f_0(\mathbf{x}^*)}{\partial x_1^*} \\ \vdots \\ \frac{\partial f_0(\mathbf{x}^*)}{\partial x_n^*} \end{bmatrix} + \begin{bmatrix} \frac{\partial h_1(\mathbf{x}^*)}{\partial x_1^*} & \dots & \frac{\partial h_q(\mathbf{x}^*)}{\partial x_1^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_1(\mathbf{x}^*)}{\partial x_n^*} & \dots & \frac{\partial h_q(\mathbf{x}^*)}{\partial x_n^*} \end{bmatrix} \begin{bmatrix} \nu_1^* \\ \vdots \\ \nu_q^* \end{bmatrix} = - \begin{bmatrix} \frac{\partial f_{A1}(\mathbf{x}^*)}{\partial x_1^*} & \dots & \frac{\partial f_{Ak}(\mathbf{x}^*)}{\partial x_1^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{A1}(\mathbf{x}^*)}{\partial x_n^*} & \dots & \frac{\partial f_{Ak}(\mathbf{x}^*)}{\partial x_n^*} \end{bmatrix} \begin{bmatrix} \lambda_{A1}^* \\ \vdots \\ \lambda_{Ak}^* \end{bmatrix}$$

即有

$$\nabla f_0(\mathbf{x}^*) + (\mathbf{J}_h(\mathbf{x}^*))^\top \boldsymbol{\nu}^* = -(\mathbf{J}_{\mathcal{A}}(\mathbf{x}^*))^\top \boldsymbol{\lambda}_{\mathcal{A}}^* \quad (4.3.52)$$

式中  $\mathbf{J}_h(\mathbf{x}^*)$  是等式约束  $h_i(\mathbf{x}) = 0, i = 1, \dots, q$  在点  $\mathbf{x}^*$  的 Jacobian 矩阵, 而

$$\mathbf{J}_{\mathcal{A}}(\mathbf{x}^*) = \begin{bmatrix} \frac{\partial f_{\mathcal{A}1}(\mathbf{x}^*)}{\partial x_1^*} & \dots & \frac{\partial f_{\mathcal{A}1}(\mathbf{x}^*)}{\partial x_n^*} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{\mathcal{A}k}(\mathbf{x}^*)}{\partial x_1^*} & \dots & \frac{\partial f_{\mathcal{A}k}(\mathbf{x}^*)}{\partial x_n^*} \end{bmatrix} \in \mathbb{R}^{k \times n} \quad (4.3.53)$$

$$\boldsymbol{\lambda}_{\mathcal{A}}^* = [\lambda_{\mathcal{A}1}^*, \dots, \lambda_{\mathcal{A}k}^*] \in \mathbb{R}^k \quad (4.3.54)$$

分别是积极约束的 Jacobian 矩阵和 Lagrangian 乘子向量。

式 (4.3.52) 表明, 若积极约束在可行点  $\bar{\mathbf{x}}$  的 Jacobian 矩阵  $\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})$  满行秩, 则积极约束的 Lagrangian 乘子向量可由

$$\boldsymbol{\lambda}_{\mathcal{A}}^* = -(\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})^\top)^{-1}\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})[\nabla f_0(\bar{\mathbf{x}}) + (\mathbf{J}_h(\bar{\mathbf{x}}))^\top \boldsymbol{\nu}^*] \quad (4.3.55)$$

唯一确定。为此, 对积极约束的梯度向量有以下规定。

**定义 4.3.8** 考虑不等式约束  $f_i(\mathbf{x}) \leq 0$ 。称线性无关约束规定 (LICQ: linear independence constraint qualification) 在可行点  $\bar{\mathbf{x}}$  成立, 若积极约束的梯度  $\nabla f_{\mathcal{A}i}(\bar{\mathbf{x}}), i \in \mathcal{A}$  线性无关, 或积极约束的 Jacobian 矩阵  $\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})$  满行秩。

**定义 4.3.9** 考虑不等式约束  $f_i(\mathbf{x}) \leq 0$ 。称 Mangasarian-Fromovitz 约束规定 (MFC-Q) 在可行点  $\bar{\mathbf{x}}$  成立, 若  $\bar{\mathbf{x}}$  是严格可行点, 或者若存在一向量  $\mathbf{p}$  使得  $\nabla f_i(\bar{\mathbf{x}})^\top \mathbf{p} < 0$  对所有  $i \in \mathcal{A}$  成立, 即若  $\mathbf{J}_{\mathcal{A}}(\bar{\mathbf{x}})^\top \mathbf{p} < 0$ 。

## 2. Slater 定理 (强对偶性的判断)

在算法设计中, 总是希望强对偶性能够成立。判断强对偶性是否成立的一种简单方法是 Slater 定理。

定义原始不等式约束的可行域  $\mathcal{F}$  的相对内域为

$$\text{relint}(\mathcal{F}) = \{\mathbf{x} | f_i(\mathbf{x}) < 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, p\} \quad (4.3.56)$$

位于可行域的相对内域的点  $\bar{\mathbf{x}} \in \text{relint}(\mathcal{F})$  称为相对内点 (relative interior point)。

优化过程中, 迭代点位于可行域的内域的约束规定称为 Slater 条件。Slater 定理说的是: 如果 Slater 条件满足, 并且原始不等式优化问题式 (4.3.2) 为凸优化问题, 则对偶无约束优化问题式 (4.3.45) 的最优值  $d^*$  与原始优化问题的最优值  $p^*$  相等, 即强对偶性成立。

下面列举出原始约束优化问题与 Lagrangian 对偶无约束凸优化问题的最优解之间的几点关系。

(1) 只有当不等式约束函数  $f_i(\mathbf{x}), i = 1, \dots, m$  均为凸函数, 且等式约束函数  $h_i(\mathbf{x}), i = 1, \dots, q$  均为仿射函数时, 一个原始约束优化问题才能借助 Lagrangian 松弛方法, 转换成一个凹函数的对偶无约束极大化问题。

(2) 凹函数的极大化等价于凸函数的极小化。

(3) 若原始约束优化问题的目标函数  $f_0(\mathbf{x})$  不是凸函数, 但不等式约束函数  $f_i(\mathbf{x}), i = 1, \dots, m$  均为凸函数, 并且等式约束函数  $h_i(\mathbf{x}), i = 1, \dots, q$  均为仿射函数, 则 Lagrangian 目标函数满足 KKT 条件的点  $\mathbf{x}^*$  和  $(\lambda^*, \nu^*)$  一般不会分别是原始最优点和对偶最优点, 即 Lagrangian 对偶无约束优化问题的最优解不是原始约束优化问题的最优解, 而是  $\varepsilon$ -次最优解, 其中  $\varepsilon = f_0(\mathbf{x}^*) - J_D(\lambda^*, \nu^*)$ 。

(4) 若  $f_0(\mathbf{x})$  和  $f_i(\mathbf{x})$  均为凸函数, 并且等式约束函数  $h_i(\mathbf{x})$  均为仿射函数, 即原始约束优化问题为凸优化问题, 则 Lagrangian 目标函数满足 KKT 条件的点  $\tilde{\mathbf{x}}$  和  $(\tilde{\lambda}, \tilde{\nu})$  分别是具有零对偶间隙的原始最优点和对偶最优点。换言之, Lagrangian 对偶无约束优化问题的最优解  $\mathbf{d}^*$  就是原始约束凸优化问题的最优解  $\mathbf{p}^*$ 。

## 4.4 平滑凸优化的一阶算法

非凸函数的约束极小化问题可以利用 Lagrangian 对偶法, 变成凹函数的无约束极大化问题求解。本节讨论无约束凸优化的最优化算法。

最优化算法分为一阶算法和二阶算法。本节以平滑函数为对象, 介绍平滑凸优化的一阶算法: 梯度法和投影梯度法、共轭梯度法、Nesterov 最优梯度法。

### 4.4.1 梯度法与梯度投影法

考虑目标函数  $f : Q \rightarrow \mathbb{R}$  (其中  $\mathbf{x} \in Q \subset \mathbb{R}^n$ ) 的无约束优化问题

$$\min_{\mathbf{x} \in Q} f(\mathbf{x}) \quad (4.4.1)$$

如图 4.4.1 所示, 向量  $\mathbf{x}$  为黑盒的输入, 其输出为函数  $f(\mathbf{x})$  及其梯度函数  $\nabla f(\mathbf{x})$ 。

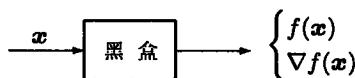


图 4.4.1 一阶黑盒优化方法

令  $\mathbf{x}_{\text{opt}}$  表示  $\min f(\mathbf{x})$  的最优解, 一阶黑盒优化 (first-order black-box optimization) 就是只利用  $f(\mathbf{x})$  和  $\nabla f(\mathbf{x})$ , 求解向量  $\mathbf{y} \in Q$  满足

$$\mathbf{y} : f(\mathbf{y}) - f(\mathbf{x}_{\text{opt}}) \leq \varepsilon$$

其中  $\varepsilon$  是给定的精度误差。满足这一条件的解  $\mathbf{y}$  称为目标函数  $f(\mathbf{x})$  的  $\varepsilon$ -次最优解。

一阶黑盒优化方法包括两个基本任务：

- (1) 一阶迭代优化算法的设计。
- (2) 优化算法的收敛速率或复杂度分析。

下面先介绍优化算法的设计。

下降法 (descent method) 是一种最简单的一阶优化方法，其求解无约束凸函数最小化问题  $\min f(\mathbf{x})$  的基本思想是：当  $Q = \mathbb{R}^n$  时，利用优化序列

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mu_k \Delta \mathbf{x}_k, \quad k = 1, 2, \dots \quad (4.4.2)$$

寻找最优点  $\mathbf{x}_{\text{opt}}$ 。式中， $k = 1, 2, \dots$  表示迭代次数， $\mu_k \geq 0$  称为第  $k$  次迭代的步长 (step size 或 step length)，用于控制更新  $\mathbf{x}$  寻优的步伐； $\Delta$  和  $\mathbf{x}$  的连体符号 (concatenated symbols)  $\Delta \mathbf{x}$  表示  $\mathbb{R}^n$  内的一个向量，称为步行方向 (step direction) 或搜索方向 (search direction)，而  $\Delta \mathbf{x}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$  表示目标函数  $f(\mathbf{x})$  在第  $k$  次迭代的搜索方向。

由于最小化算法设计要求迭代过程中目标函数是下降的

$$f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k) \quad (4.4.3)$$

所以这种方法称为下降法。这就要求对所有  $k$ ，必须有  $\mathbf{x}_k \in \text{dom } f$ 。

由目标函数在  $\mathbf{x}_k$  的一阶 Taylor 近似表达式

$$f(\mathbf{x}_{k+1}) \approx f(\mathbf{x}_k) + (\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k \quad (4.4.4)$$

易知，若

$$(\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k < 0 \quad (4.4.5)$$

则  $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$ ，故满足  $(\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k < 0$  的搜索方向  $\Delta \mathbf{x}_k$  称为目标函数  $f(\mathbf{x})$  在第  $k$  次迭代的下降步 (descent step) 或下降方向 (descent direction)。

显然，为使  $(\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k < 0$  成立，应当取

$$\Delta \mathbf{x}_k = -\nabla f(\mathbf{x}_k) \cos \theta \quad (4.4.6)$$

其中  $0 \leq \theta < \pi/2$  是下降方向与负梯度方向  $-\nabla f(\mathbf{x}_k)$  之间的夹角，为锐角。

$\theta = 0$  意味着  $\Delta \mathbf{x}_k = -\nabla f(\mathbf{x}_k)$ ，即搜索方向直接取目标函数  $f$  在点  $\mathbf{x}_k$  的负梯度方向。此时，下降步的长度  $\|\Delta \mathbf{x}_k\|_2 = \|\nabla f(\mathbf{x}_k)\|_2$  取最大值，故称下降方向  $\Delta \mathbf{x}_k$  具有最大的下降步伐或速率，与此对应的下降法则称为最速下降法 (steepest descent method)

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mu_k \nabla f(\mathbf{x}_k), \quad k = 1, 2, \dots \quad (4.4.7)$$

最速下降法也可利用函数的二次逼近解释。函数  $f(\mathbf{x})$  在  $\mathbf{y}$  点的二次 Taylor 展开为

$$f(\mathbf{y}) \approx f(\mathbf{x}) + (\nabla f(\mathbf{x}))^T (\mathbf{y} - \mathbf{x}) + \frac{1}{2} (\mathbf{y} - \mathbf{x})^T \nabla^2 f(\mathbf{x}) (\mathbf{y} - \mathbf{x}) \quad (4.4.8)$$

若用  $\frac{1}{t}I$  代替 Hessian 矩阵  $\nabla^2 f(\mathbf{x})$ , 则有

$$f(\mathbf{y}) \approx f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top(\mathbf{y} - \mathbf{x}) + \frac{1}{2t}\|\mathbf{y} - \mathbf{x}\|_2^2 \quad (4.4.9)$$

上式为函数  $f(\mathbf{x})$  在点  $\mathbf{y}$  的二次逼近 (quadratic approximation, QA)。易求得梯度向量

$$\nabla f(\mathbf{y}) = \frac{\partial f(\mathbf{y})}{\partial \mathbf{y}} = \nabla f(\mathbf{x}) + \frac{1}{t}(\mathbf{y} - \mathbf{x})$$

令  $\nabla f(\mathbf{y}) = 0$ , 便得到解为  $\mathbf{y} = \mathbf{x} - t\nabla f(\mathbf{x})$ 。令  $\mathbf{y} = \mathbf{x}_{k+1}$  和  $\mathbf{x} = \mathbf{x}_k$ , 立即得最速下降法的更新公式  $\mathbf{x}_{k+1} = \mathbf{x}_k - t\nabla f(\mathbf{x}_k)$ 。

最速下降方向  $\Delta \mathbf{x} = -\nabla f(\mathbf{x})$  只使用目标函数  $f(\mathbf{x})$  的一阶梯度信息。如果能够再利用目标函数的二阶梯度即 Hessian 矩阵  $\nabla^2 f(\mathbf{x}_k)$ , 则有望找到更好的下降方向。此时, 最优下降方向  $\Delta \mathbf{x}$  应该是使  $f(\mathbf{x})$  的二阶 Taylor 逼近函数最小化问题的解

$$\min_{\Delta \mathbf{x}} f(\mathbf{x} + \Delta \mathbf{x}) = f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top \Delta \mathbf{x} + \frac{1}{2}(\Delta \mathbf{x})^\top \nabla^2 f(\mathbf{x}) \Delta \mathbf{x} \quad (4.4.10)$$

在最优点, 相对于参数向量  $\Delta \mathbf{x}$  的梯度必须等于零, 即

$$\begin{aligned} \frac{\partial f(\mathbf{x} + \Delta \mathbf{x})}{\partial \Delta \mathbf{x}} &= \nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x}) \Delta \mathbf{x} = \mathbf{0} \\ \iff \Delta \mathbf{x}_{\text{nt}} &= -(\nabla^2 f(\mathbf{x}))^{-1} \nabla f(\mathbf{x}) \end{aligned} \quad (4.4.11)$$

其中  $\Delta \mathbf{x}_{\text{nt}}$  称为 Newton 步或 Newton 下降方向, 相应的寻优方法称为 Newton 法。Newton 法也称 Newton-Raphson 法。

#### 算法 4.4.1 梯度下降算法及其变型

初始化 选择一个起始点  $\mathbf{x}_1 \in \text{dom } f$  和允许精度  $\varepsilon > 0$ , 并且令  $k = 1$ 。

步骤 1 计算目标函数在点  $\mathbf{x}_k$  的梯度  $\nabla f(\mathbf{x}_k)$  (以及 Hessian 矩阵  $\nabla^2 f(\mathbf{x}_k)$ ), 并选择下降方向

$$\Delta \mathbf{x}_k = \begin{cases} -\nabla f(\mathbf{x}_k) & \text{(最速下降法)} \\ -(\nabla^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k) & \text{(Newton 法)} \end{cases}$$

步骤 2 选择步长  $\mu_k > 0$ 。

步骤 3 进行更新

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mu_k \Delta \mathbf{x}_k \quad (4.4.12)$$

步骤 4 判断停止准则是否满足: 若  $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| \leq \varepsilon$ , 则停止迭代, 并输出  $\mathbf{x}_k$ ; 若不满足, 则令  $k \leftarrow k + 1$ , 并返回步骤 1, 进行下一轮迭代, 直至停止准则满足为止。

根据步长  $\mu_k$  的选择不同, 梯度算法有以下几种常用变型<sup>[363]</sup>:

(1) 梯度算法执行之前, 选择步长序列  $\{\mu_k\}_{k=0}^{\infty}$ 。例如

$$\mu_k = \mu \quad (\text{固定步长}) \quad \text{或} \quad \mu_k = \frac{h}{\sqrt{k+1}}$$

(2) 全松弛 (full relaxation)

$$\mu_k = \arg \min_{\mu \geq 0} f(\mathbf{x}_k - \mu \nabla f(\mathbf{x}_k))$$

(3) Goldstein-Armijo 规则 求  $\mathbf{x}_{k+1} = \mathbf{x}_k - \mu \nabla f(\mathbf{x}_k)$ , 使得

$$\begin{aligned}\alpha \langle \nabla f(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}_{k+1} \rangle &\leq f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \\ \beta \langle \nabla f(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}_{k+1} \rangle &\geq f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})\end{aligned}$$

式中,  $0 < \alpha < \beta < 1$  是两个固定的参数。

注意, 梯度算法中的变元向量  $\mathbf{x}$  是无约束的, 即  $\mathbf{x} \in \mathbb{R}^n$ 。若  $\mathbf{x} \in C$  是有约束的, 其中  $C \subset \mathbb{R}^n$ , 则梯度算法中的更新公式应该用投影代替

$$\mathbf{x}_{k+1} = \mathcal{P}_C(\mathbf{x}_k - \mu_k \nabla f(\mathbf{x}_k)) \quad (4.4.13)$$

这一算法称为梯度投影法 (gradient-projection method) 法。梯度投影法也称投影梯度法 (projected gradient method)。 $\mathcal{P}_C(\mathbf{y})$  称为投影算子 (projection operator), 定义为

$$\mathcal{P}_C(\mathbf{y}) = \arg \min_{\mathbf{x} \in C} \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.4.14)$$

投影算子也可等价表示为

$$\mathcal{P}_C(\mathbf{y}) = \mathbf{P}_C \mathbf{y} \quad (4.4.15)$$

其中,  $\mathbf{P}_C$  是到子空间  $C$  上的投影矩阵。若  $C$  是矩阵  $\mathbf{A}$  的列空间, 则

$$\mathbf{P}_A = \mathbf{A}(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \quad (4.4.16)$$

关于投影矩阵, 将在第 9 章 (投影分析) 中详细讨论。

特别地, 若  $C = \mathbb{R}^n$  即变元向量  $\mathbf{x}$  是无约束的, 则投影算子等于单位矩阵, 即  $\mathcal{P}_C = \mathbf{I}$ , 故有

$$\mathcal{P}_{\mathbb{R}^n}(\mathbf{y}) = \mathbf{P} \mathbf{y} = \mathbf{y}, \quad \forall \mathbf{y} \in \mathbb{R}^n$$

此时, 梯度投影算法退化为梯度算法。

下面是向量  $\mathbf{x}$  到一些典型集合上的投影 [498]。

(1) 到超平面  $C = \{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$  (其中  $\mathbf{a} \neq \mathbf{0}$ ) 上的投影

$$\mathcal{P}_C(\mathbf{x}) = \mathbf{x} + \frac{b - \mathbf{a}^T \mathbf{x}}{\|\mathbf{a}\|_2^2} \mathbf{a} \quad (4.4.17)$$

(2) 到仿射集  $C = \{\mathbf{x} | \mathbf{Ax} = \mathbf{b}\}$  (其中  $\mathbf{A} \in \mathbb{R}^{p \times n}$ ,  $\text{rank}(\mathbf{A}) = p$ ) 上的投影

$$\mathcal{P}_C(\mathbf{x}) = \mathbf{x} + \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1} (\mathbf{b} - \mathbf{A} \mathbf{x}) \quad (4.4.18)$$

若  $p \ll n$  或者  $\mathbf{A} \mathbf{A}^T = \mathbf{I}$ , 则投影  $\mathcal{P}_C(\mathbf{x})$  是低成本的。

(3) 到非负象限  $C = \mathbb{R}_+^n$  上的投影

$$\mathcal{P}_C(\mathbf{x}) = (\mathbf{x})^+ \Leftrightarrow [(\mathbf{x})^+]_i = \max\{x_i, 0\} \quad (4.4.19)$$

(4) 到半空间 (halfspace)  $C = \{\mathbf{x} | \mathbf{a}^\top \mathbf{x} \leq b\}$  (其中  $\mathbf{a} \neq \mathbf{0}$ ) 上的投影

$$\mathcal{P}_C(\mathbf{x}) = \begin{cases} \mathbf{x} + \frac{b - \mathbf{a}^\top \mathbf{x}}{\|\mathbf{a}\|_2^2} \mathbf{a}, & \text{若 } \mathbf{a}^\top \mathbf{x} > b \\ \mathbf{x}, & \text{若 } \mathbf{a}^\top \mathbf{x} \leq b \end{cases} \quad (4.4.20)$$

(5) 到矩形集  $C = [\mathbf{a}, \mathbf{b}]$  (其中  $a_i \leq x_i \leq b_i$ ) 上的投影

$$\mathcal{P}_C(\mathbf{x}) = \begin{cases} a_i, & \text{若 } x_i \leq a_i \\ x_i, & \text{若 } a_i \leq x_i \leq b_i \\ b_i, & \text{若 } x_i \geq b_i \end{cases} \quad (4.4.21)$$

(6) 到 Euclidean 球  $C = \{\mathbf{x} | \|\mathbf{x}\|_2 \leq 1\}$  上的投影

$$\mathcal{P}_C(\mathbf{x}) = \begin{cases} \frac{1}{\|\mathbf{x}\|_2} \mathbf{x}, & \text{若 } \|\mathbf{x}\|_2 > 1 \\ \mathbf{x}, & \text{若 } \|\mathbf{x}\|_2 \leq 1 \end{cases} \quad (4.4.22)$$

(7) 到  $L_1$  范数球  $C = \{\mathbf{x} | \|\mathbf{x}\|_1 \leq 1\}$  上的投影

$$\mathcal{P}_C(\mathbf{x})_i = \begin{cases} x_i - \lambda, & \text{若 } x_i > \lambda \\ 0, & \text{若 } -\lambda \leq x_i \leq \lambda \\ x_i + \lambda, & \text{若 } x_i < -\lambda \end{cases} \quad (4.4.23)$$

其中  $\lambda = 0$ , 若  $\|\mathbf{x}\|_1 \leq 1$ ; 否则  $\lambda$  是下列方程的解

$$\sum_{i=1}^n \max\{|x_i| - \lambda, 0\} = 1$$

(8) 到二阶锥  $C = \{(\mathbf{x}, t) | \|\mathbf{x}\|_2 \leq t, \mathbf{x} \in \mathbb{R}^n\}$  上的投影

$$\mathcal{P}_C(\mathbf{x}) = \begin{cases} (\mathbf{x}, t), & \text{若 } \|\mathbf{x}\|_2 \leq t \\ \frac{t + \|\mathbf{x}\|_2}{2\|\mathbf{x}\|_2} \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix}, & \text{若 } -t < \|\mathbf{x}\|_2 < t \\ (0, 0), & \text{若 } \|\mathbf{x}\|_2 \leq -t, \mathbf{x} \neq \mathbf{0} \end{cases} \quad (4.4.24)$$

(9) 到半正定锥  $C = \mathbb{S}_+^n$  上的投影

$$\mathcal{P}_C(\mathbf{X}) = \sum_{i=1}^n \max\{0, \lambda_i\} \mathbf{q}_i \mathbf{q}_i^\top \quad (4.4.25)$$

式中,  $\mathbf{X} = \sum_{i=1}^n \lambda_i \mathbf{q}_i \mathbf{q}_i^\top$  是半正定矩阵  $\mathbf{X}$  的特征值分解。

### 4.4.2 共轭梯度算法

考虑矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的迭代求解，其中  $\mathbf{A} \in \mathbb{R}^{n \times n}$  是一非奇异矩阵。这一矩阵方程可以等价写作

$$\mathbf{x} = (\mathbf{I} - \mathbf{A})\mathbf{x} + \mathbf{b} \quad (4.4.26)$$

由此启发了下面的迭代算法

$$\mathbf{x}_{k+1} = (\mathbf{I} - \mathbf{A})\mathbf{x}_k + \mathbf{b} \quad (4.4.27)$$

这一迭代称为 Richardson 迭代，它可以写作更加一般的形式

$$\mathbf{x}_{k+1} = \mathbf{M}\mathbf{x}_k + \mathbf{c} \quad (4.4.28)$$

其中  $\mathbf{M}$  是一个  $n \times n$  矩阵，称为迭代矩阵 (iteration matrix)。

具有式 (4.4.28) 一类形式的迭代称为固定迭代法 (stationary iterative methods)，但它没有非固定迭代法 (nonstationary iterative methods) 有效。

所谓非固定迭代法就是  $\mathbf{x}_{k+1}$  与前面的迭代  $\mathbf{x}_k, \mathbf{x}_{k-1}, \dots, \mathbf{x}_0$  都有关的一种迭代方法。最典型的非固定迭代法是 Krylov 子空间方法

$$\mathbf{x}_{k+1} = \mathbf{x}_0 + \mathcal{K}_k \quad (4.4.29)$$

式中

$$\mathcal{K}_k = \text{span}(\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{k-1}\mathbf{r}_0) \quad (4.4.30)$$

称为第  $k$  次 Krylov 子空间，其中  $\mathbf{x}_0$  是迭代的初始值，而  $\mathbf{r}_0$  表示初始残差 (向量)。

Krylov 子空间方法存在多种形式，下面依次介绍三种最常用的 Krylov 子空间方法：共轭梯度法、双共轭梯度法和预处理共轭梯度法。

#### 1. 共轭梯度法

共轭梯度法 (conjugate gradient method) 使用  $\mathbf{r}_0 = \mathbf{A}\mathbf{x}_0 - \mathbf{b}$  作为初始残差向量。

共轭梯度法的适用对象限定为对称正定方程组  $\mathbf{A}\mathbf{x} = \mathbf{b}$ ，其中  $\mathbf{A}$  是一个  $n \times n$  的对称正定矩阵。

称非零向量组合  $\{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_k\}$  是  $\mathbf{A}$ -正交或共轭的，若

$$\mathbf{p}_i^T \mathbf{A} \mathbf{p}_j = 0, \quad \forall i \neq j \quad (4.4.31)$$

式 (4.4.31) 描述的性质常常简称为一组向量的  $\mathbf{A}$ -正交性或共轭性 (conjugacy)。显然，若  $\mathbf{A} = \mathbf{I}$ ，则向量的共轭性退化为普通的向量正交。

凡使用共轭向量作为更新方向的算法统称共轭方向算法。若共轭向量  $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_{n-1}$  不是预先设定的，而是在迭代过程中利用梯度下降法更新，则称目标函数  $f(\mathbf{x})$  的最小化算法为共轭梯度算法。

**算法 4.4.2 共轭梯度 (CG) 算法** [198, 263]

输入  $n \times n$  对称矩阵  $A$  和  $n \times 1$  向量  $b$ , 最大迭代步数  $k_{\max}$ , 允许误差  $\varepsilon$ 。

初始化 选择  $x_0 \in \mathbb{R}^n$ , 令  $r = Ax_0 - b$  和  $\rho_0 = \|r\|_2^2$ 。

迭代  $k = 1, 2, \dots, k_{\max}$

1. 若  $k = 1$ , 则  $p = r$ 。否则, 令  $\beta = \rho_{k-1}/\rho_{k-2}$  和  $p = r + \beta p$ ;

2.  $w = Ap$ ;

3.  $\alpha = \rho_{k-1}/p^T w$ ;

4.  $x = x + \alpha p$ ;

5.  $r = r - \alpha w$ ;

6.  $\rho_k = \|r\|_2^2$ ;

7. 若  $\sqrt{\rho_k} < \varepsilon \|b\|_2$  或者  $k = k_{\max}$ , 则停止迭代, 并输出  $x$ ; 否则, 令  $k = k + 1$ , 并返回步骤 1, 继续迭代。

由上述算法可以看出, 在共轭梯度法的迭代过程中, 矩阵方程  $Ax = b$  解为

$$x_k = \sum_{i=1}^k \alpha_i p_i = \sum_{i=1}^k \frac{\langle r_{i-1}, r_{i-1} \rangle}{\langle p_i, Ap_i \rangle} p_i \quad (4.4.32)$$

即  $x_k$  属于第  $k$  次 Krylov 子空间

$$x_k \in \text{span}\{p_1, p_2, \dots, p_k\} = \text{span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}$$

与固定迭代法的迭代矩阵  $M$  需要构造和存储不同, 算法 4.4.2 中的矩阵  $A$  只是参与矩阵与向量的相乘  $Ap$ 。因此, Krylov 子空间法又称为无矩阵 (matrix-free) 方法<sup>[263]</sup>。

## 2. 双共轭梯度法

若矩阵  $A$  不是实对称矩阵, 则可使用 Fletcher<sup>[168]</sup> 提出的双共轭梯度法 (biconjugate gradient method) 求解矩阵方程  $Ax = b$ 。顾名思义, 在这种方法中, 有两个搜索方向  $p, \bar{p}$  与矩阵  $A$  共轭。

$$\left. \begin{array}{l} \bar{p}_i^T A p_j = p_i^T A \bar{p}_j = 0, \quad i \neq j \\ \bar{r}_i^T r_j = r_i^T \bar{r}_j^T = 0, \quad i \neq j \\ \bar{r}_i^T p_j = r_i^T \bar{p}_j^T = 0, \quad j < i \end{array} \right\} \quad (4.4.33)$$

### 算法 4.4.3 双共轭梯度法<sup>[168, 252]</sup>

初始化  $p_1 = r_1, \bar{p}_1 = \bar{r}_1$ 。

迭代 对  $k = 0, 1, \dots, k_{\max}$ , 计算

$$\begin{aligned}\alpha_k &= \bar{r}_k^T r_k / (\bar{p}_k^T A p_k) \\ r_{k+1} &= r_k - \alpha_k A p_k \\ \bar{r}_{k+1} &= \bar{r}_k - \alpha_k A^T \bar{p}_k \\ \beta_k &= \bar{r}_{k+1}^T r_{k+1} / (\bar{r}_k^T r_k) \\ p_{k+1} &= r_{k+1} + \beta_k p_k \\ \bar{p}_{k+1} &= \bar{r}_{k+1} + \beta_k \bar{p}_k\end{aligned}$$

输出

$$x_{k+1} = x_k + \alpha_k p_k, \quad \bar{x}_{k+1} = \bar{x}_k + \alpha_k \bar{p}_k$$

### 3. 预处理共轭梯度法

1988 年, Bramble 与 Pasciak<sup>[57]</sup> 针对对称不定鞍点问题 (symmetric indefinite saddle point problems)

$$\begin{bmatrix} A & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ q \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}$$

开创性地提出了预处理共轭梯度 (preconditioned conjugate gradient) 迭代。其中,  $A$  是一个  $n \times n$  实对称正定矩阵,  $B$  是一个  $m \times n$  实矩阵, 具有满行秩  $m$  ( $\leq n$ ), 而  $O$  是一个  $m \times m$  零矩阵。

预处理共轭梯度迭代的基本思想是: 通过灵巧选择标量积的形式, 使得预处理后的鞍点矩阵变成对称正定矩阵。

为了简化讨论, 假定需要将一个条件数大的矩阵方程  $Ax = b$  转换成另外一个具有相同解的对称正定方程。令  $M$  是一个可以逼近  $A$  的对称正定矩阵, 但比  $A$  更容易求逆。于是, 原矩阵方程  $Ax = b$  可以转换成  $M^{-1}Ax = M^{-1}b$ , 两者具有相同的解。然而, 矩阵方程  $M^{-1}Ax = M^{-1}b$  存在一个隐患:  $M^{-1}A$  一般既不是对称的, 也不是正定的, 即使  $M$  和  $A$  都是对称正定的。因此, 直接用矩阵  $M^{-1}$  作为矩阵方程  $Ax = b$  的预处理器是不可靠的。

令  $S$  是对称矩阵  $M$  的平方根, 即  $M = SS^T$ , 其中  $S$  是对称正定的。现在, 使用  $S^{-1}$  代替  $M^{-1}$  作为预处理器, 将原矩阵方程  $Ax = b$  变成  $S^{-1}Ax = S^{-1}b$ 。若令  $x = S^{-T}\hat{x}$ , 则预处理后的矩阵方程为

$$S^{-1}AS^{-T}\hat{x} = S^{-1}b \tag{4.4.34}$$

与矩阵  $M^{-1}A$  一般缺乏对称正定性不同,  $S^{-1}AS^{-T}$  一定是对称正定的, 若  $A$  是对称正定的。 $S^{-1}AS^{-T}$  的对称性很容易看出, 正定性也容易验证: 检验二次型函数易知,  $y^T(S^{-1}AS^{-T})y = z^TAz$ , 其中  $z = S^{-T}y$ 。由于  $A$  是对称正定的, 故  $z^TAz > 0$ ,  $\forall z \neq 0$ , 从而  $y^T(S^{-1}AS^{-T})y > 0, \forall y \neq 0$ 。即是说,  $S^{-1}AS^{-T}$  一定是正定的。

现在, 共轭梯度法可以应用于求解矩阵方程式 (4.4.34), 得到  $\hat{x}$ , 然后再由  $x = S^{-T}\hat{x}$  即可恢复  $x$ 。

#### 算法 4.4.4 使用预处理器的预处理共轭梯度算法<sup>[452]</sup>

输入  $A, b$ , 预处理器  $S^{-1}$  (也许隐定义), 最大迭代步数  $k_{\max}$ , 容许误差  $\varepsilon < 1$ 。

初始化  $k = 0, r = Ax - b, d = S^{-1}r, \delta_{\text{new}} = r^T r, \delta_0 = \delta_{\text{new}}$ 。

迭代 若  $k = k_{\max}$  或者  $\delta_{\text{new}} < \varepsilon^2 \delta_0$ , 则停止迭代; 否则, 进行以下计算。

1.  $q = Ad;$
2.  $\alpha = \delta_{\text{new}} / (d^T q);$
3.  $x = x + \alpha d;$
4. 若  $k$  能够被 50 整除, 则  $r = b - Ax$ 。否则,  $r = r - \alpha q$ ;
5.  $s = S^{-1}r;$
6.  $\delta_{\text{old}} = \delta_{\text{new}};$
7.  $\delta_{\text{new}} = r^T s;$
8.  $\beta = \delta_{\text{new}} / \delta_{\text{old}};$
9.  $d = s + \beta d;$
10.  $k = k + 1$ , 并重复以上迭代。

预处理器可以避免被使用, 因为容易看出矩阵方程  $Ax = b$  与  $S^{-1}AS^{-T}\hat{x} = S^{-1}b$  的变元之间存在下列对应关系<sup>[263]</sup>

$$x_k = S^{-1}\hat{x}_k, \quad r_k = S\hat{r}_k, \quad p_k = S^{-1}\hat{p}_k, \quad z_k = S^{-1}\hat{r}_k$$

利用这些对应关系, 可以由共轭梯度法得到下面的预处理共轭梯度算法。

#### 算法 4.4.5 不用预处理器的预处理共轭梯度算法<sup>[263]</sup>

输入  $n \times n$  对称矩阵  $A$  和  $n \times 1$  向量  $b$ , 最大迭代步数  $k_{\max}$ , 允许误差  $\varepsilon$ 。

初始化  $x_0 \in \mathbb{R}^n, r = Ax_0 - b, \rho_0 = \|r\|_2^2$ 。

迭代  $k = 1, 2, \dots, k_{\max}$

1.  $z = Mr;$
2.  $\tau_{k-1} = z^T r;$
3. 若  $k = 1$ , 则  $\beta = 0$  和  $p = z$ 。否则, 令  $\beta = \tau_{k-1} / \tau_{k-2}$  和  $p = z + \beta p$ ;
4.  $w = Ap;$
5.  $\alpha = \tau_{k-1} / p^T w;$
6.  $x = x + \alpha p;$
7.  $r = r - \alpha w;$
8.  $\rho_k = r^T r;$
9. 若  $\sqrt{\rho_k} < \varepsilon \|b\|_2$  或者  $k = k_{\max}$ , 则停止迭代, 并输出  $x$ ; 否则, 令  $k = k + 1$ , 并返回步骤 1, 继续迭代。

文献 [452] 给出了共轭梯度法的精彩导论。

对于复矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$ , 其中  $\mathbf{A} \in \mathbb{C}^{n \times n}$ ,  $\mathbf{x} \in \mathbb{C}^n$ ,  $\mathbf{b} \in \mathbb{C}^n$ , 可以将它按照实部和虚部变成下面的实矩阵方程

$$\begin{bmatrix} \mathbf{A}_R & -\mathbf{A}_I \\ \mathbf{A}_I & \mathbf{A}_R \end{bmatrix} = \begin{bmatrix} \mathbf{b}_R \\ \mathbf{b}_I \end{bmatrix} \quad (4.4.35)$$

若  $\mathbf{A} = \mathbf{A}_R + j\mathbf{A}_I$  是 Hermitian 正定矩阵, 则式 (4.4.35) 是对称正定矩阵。因此, 共轭梯度算法和预处理共轭梯度算法即可用于求出  $(\mathbf{x}_R, \mathbf{x}_I)$ 。

预处理共轭梯度法是求解偏微分方程的一种广泛适用的技术, 在优化控制中有着重要的应用 [227]。事实上, 正如后面将介绍的那样, 求解优化问题的 KKT 方程和 Newton 方程也需要利用预处理共轭梯度法, 以提高求解优化搜索方向的数值稳定性。

除了以上三种共轭梯度算法之外, 还有投影共轭梯度 (projected conjugate gradients) 算法 [198]。这种算法需要使用一个种子空间 (seed space)  $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$  产生矩阵方程  $\mathbf{A}\mathbf{x}_q = \mathbf{b}_q, q = 1, 2, \dots$  的解, 所以选择一个好的种子空间对投影共轭梯度起着关键的作用。很多情况下, 种子空间本身也可能需要进行更新。

#### 4.4.3 收敛速率

一优化算法的收敛速率是指: 优化算法需要多少次迭代, 才能使目标函数的估计误差达到所要求的精度? 或者给定一个迭代步数  $K$ , 优化算法能够达到何种精度? 收敛速率的逆函数称为优化算法的复杂度 (complexity)。

令  $\mathbf{x}^*$  代表一个局部或全局极小点, 最优化算法的估计误差定义为迭代点  $\mathbf{x}_k$  的目标函数值与该全局极小点的最小目标函数值之差

$$\delta_k = f(\mathbf{x}_k) - f(\mathbf{x}^*)$$

我们自然会对一最优化算法的收敛问题感兴趣:

- (1) 给定一迭代次数  $K$ , 期望的精度  $\lim_{1 \leq k \leq K} \delta_k$  如何?
- (2) 给定一允许精度  $\varepsilon$ , 需要多少次迭代才能达到  $\min_k \delta_k \leq \varepsilon$ ?

在分析优化算法的收敛问题时, 常常着眼于目标函数变元的更新序列  $\{\mathbf{x}_k\}$  收敛到其理想极小点  $\mathbf{x}^*$  的速度。在数值分析中, 一个序列达到其极限的速度称为收敛速率。

##### 1. Q 收敛速率 [382]

假定一序列  $\{\mathbf{x}_k\}$  收敛到  $\mathbf{x}^*$ 。若存在实数  $\alpha \geq 1$  和与迭代次数  $k$  无关的正常数  $\mu$ , 使得

$$\mu = \lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2}{\|\mathbf{x}_k - \mathbf{x}^*\|_2^\alpha} \quad (4.4.36)$$

则称  $\{\mathbf{x}_k\}$  具有  $\alpha$ -阶  $Q$  收敛速率。 $Q$  收敛速率意即商 (Quotient) 收敛速率。

$Q$  收敛速率有以下几种典型速率:

- (1) 当  $\alpha = 1$  时,  $Q$  收敛速率称为序列  $\{\mathbf{x}_k\}$  的极限收敛速率

$$\mu = \lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2}{\|\mathbf{x}_k - \mathbf{x}^*\|_2} \quad (4.4.37)$$

根据  $\mu$  值的大小, 序列  $\{\mathbf{x}_k\}$  的极限收敛速率又可分为以下三种:

- ① 次线性收敛速率 (sublinear rate of convergence):  $\alpha = 1, \mu = 1$ 。
- ② 线性收敛速率 (linear rate of convergence):  $\alpha = 1, \mu \in (0, 1)$ 。
- ③ 超线性收敛速率 (superlinear rate of convergence):  $\alpha = 1, \mu = 0$  或者  $1 < \alpha < 2, \mu = 0$ 。

(2) 当  $\alpha = 2$  时, 称  $\{\mathbf{x}_k\}$  具有  $Q$  二次收敛速率。

(3) 当  $\alpha = 3$  时, 称  $\{\mathbf{x}_k\}$  具有  $Q$  三次收敛速率。

若  $\{\mathbf{x}_k\}$  是次线性收敛的, 并且

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+2} - \mathbf{x}_{k+1}\|_2}{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2} = 1$$

则称序列  $\{\mathbf{x}_k\}$  是对数收敛的 (logarithmical convergence)。

次线性速率是一类慢的收敛速率; 线性速率是一类比较快的收敛速率; 超线性速率是一类非常快的收敛速率, 而二次收敛速率则是一类极快的收敛速率。设计优化算法时, 常常要求它至少是线性速率收敛的, 最好是二次速率收敛的。超快的三次收敛速率一般说来较难实现。

## 2. 局部收敛速率

序列  $\{\mathbf{x}_k\}$  的局部收敛速率记作  $r_k$ , 定义为

$$r_k = \left\| \frac{\mathbf{x}_{k+1} - \mathbf{x}^*}{\mathbf{x}_k - \mathbf{x}^*} \right\| \quad (4.4.38)$$

一优化算法的解析复杂度定义为更新变量的局部收敛速率的逆函数。

下面是局部收敛速率的分类 [363]:

(1) 次线性速率 这一速率用迭代次数  $k$  的幂函数描述。例如, 若局部收敛速率  $r_k \leq \frac{c}{\sqrt{k}}$ , 则相应的优化算法的复杂度上界为  $\left(\frac{c}{\epsilon}\right)^2$ 。次线性速率的收敛是相当缓慢的。常数  $c$  在复杂度中起着关键作用。

(2) 线性速率 这种收敛速率用迭代次数  $k$  的指数函数表示。例如, 若收敛速率  $r_k \leq c(1-q)^k$ , 则对应的复杂度为  $\frac{1}{q} \left( \ln c + \ln \frac{1}{\epsilon} \right)$ 。线性速率是快的。

(3) 二次速率 此速率具有迭代次数  $k$  的双指数组形式。例如, 若收敛速率  $r_{k+1} \leq cr_k^2$ , 则相应的复杂度为期望精度  $\epsilon$  的双对数函数  $\ln \ln \frac{1}{\epsilon}$ 。二次速率是一种极快的收敛速率。常数  $c$  只对二次速率的起始时刻重要。

例如, 收敛速率  $O(1/k^2)$  意味着: 达到  $f(\mathbf{x}^{(k)}) - f(\mathbf{x}^*) \leq \epsilon$  的逼近精度, 需要  $O(1/\sqrt{\epsilon})$  次迭代; 而达到同样的逼近精度, 收敛速率  $O(1/k)$  却要求  $O(1/\epsilon)$  次迭代。以  $\epsilon = 10^{-4}$  为例, 收敛速率  $O(1/k^2)$  只要求上百次迭代, 而  $O(1/k)$  的收敛速率却需要上万次迭代。

### 4.4.4 Nesterov 最优梯度法

令  $Q \subset \mathbb{R}^n$  是向量空间  $\mathbb{R}^{n'}$  的一个凸集。考虑无约束优化问题  $\min_{\mathbf{x} \in Q} f(\mathbf{x})$ 。

**定义 4.4.1** [363] 称目标函数  $f(\mathbf{x})$  在定义域  $Q$  上是 Lipschitz 连续的, 若

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq L\|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{x}, \mathbf{y} \in Q \quad (4.4.39)$$

对某个  $L > 0$  (Lipschitz 常数) 成立。类似地, 称一个可微分的函数  $f(\mathbf{x})$  的梯度  $\nabla f(\mathbf{x})$  在定义域  $Q$  上是 Lipschitz 连续的, 若

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L\|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{x}, \mathbf{y} \in Q \quad (4.4.40)$$

对某个 Lipschitz 常数  $L > 0$  成立。

### 1. Lipschitz 连续函数与连续函数的关系

称函数  $f(\mathbf{x})$  在点  $\mathbf{x}_0$  连续, 若  $\lim_{\mathbf{x} \rightarrow \mathbf{x}_0} f(\mathbf{x}) = f(\mathbf{x}_0)$ 。当我们称  $f(\mathbf{x})$  是连续函数时, 意指  $f(\mathbf{x})$  在定义域每一点都连续。

一个 Lipschitz 连续的函数  $f(\mathbf{x})$  一定是连续函数, 但是一个连续函数不一定是 Lipschitz 连续函数。例如, 函数  $f(\mathbf{x}) = \frac{1}{\sqrt{\mathbf{x}}}$  是一个在开区间  $(0, 1)$  上的连续函数。若假定它也是一个 Lipschitz 连续函数, 则必须满足  $|f(\mathbf{x}_1) - f(\mathbf{x}_2)| = \left| \frac{1}{\sqrt{\mathbf{x}_1}} - \frac{1}{\sqrt{\mathbf{x}_2}} \right| \leq L \left| \frac{1}{\mathbf{x}_1} - \frac{1}{\mathbf{x}_2} \right|$  即  $\left| \frac{1}{\sqrt{\mathbf{x}_1}} + \frac{1}{\sqrt{\mathbf{x}_2}} \right| \leq L$ 。但是, 对于  $\mathbf{x} \rightarrow 0$  点, 并且  $\mathbf{x}_1 = \frac{1}{n^2}, \mathbf{x}_2 = \frac{9}{n^2}$  时, 却有  $L \geq \frac{n}{4} \rightarrow \infty$ 。因此, 连续函数  $f(\mathbf{x}) = \frac{1}{\sqrt{\mathbf{x}}}$  在开区间  $(0, 1)$  不是 Lipschitz 连续函数。

### 2. Lipschitz 连续函数与可微分函数的关系

一个处处可微分的函数称为平滑函数。一个平滑函数一定是连续函数, 但连续函数不一定可微分。一个典型的例子是在某点尖锐的连续函数不可微分。因此, 一个 Lipschitz 函数不一定可微分, 但是在定义域  $Q$  上具有 Lipschitz 连续梯度的函数  $f(\mathbf{x})$  一定是定义域  $Q$  上的平滑函数, 因为定义规定  $f(\mathbf{x})$  在定义域  $Q$  上是可微分的。

在凸优化中, 与目标函数  $f(\mathbf{x})$  本身是否 Lipschitz 连续相比, 其  $p$  阶导数是否 Lipschitz 连续更加重要<sup>①</sup>。因此, 在凸优化中, 常使用符号  $C_L^{k,p}(Q)$  (其中  $Q \subseteq \mathbb{R}^n$ ) 表示具有以下性质的 Lipschitz 连续函数类<sup>[363]</sup>:

- (1) 函数  $f \in C_L^{k,p}(Q)$  是在  $Q$  上可  $k$  次连续微分的。
- (2) 函数  $f \in C_L^{k,p}(Q)$  的  $p$  阶导数都是 Lipschitz 常数为  $L$  的 Lipschitz 连续函数

$$\|f^{(p)}(\mathbf{x}) - f^{(p)}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{x}, \mathbf{y} \in Q$$

若  $k \neq 0$ , 则称  $f \in C_L^{k,p}(Q)$  是可微分函数。显然, 总是有  $p \leq k$ 。若  $q > k$ , 则有  $C_L^{q,p}(Q) \subseteq C_L^{k,p}(Q)$ 。例如,  $C_L^{2,1}(Q) \subseteq C_L^{1,1}(Q)$ 。

以下是几种常用的函数类  $C_L^{k,p}(Q)$ :

<sup>①</sup> 这里限定  $p = 0, 1, 2$ 。函数  $f(\mathbf{x})$  的零阶导数为函数本身, 一阶导数是其梯度向量  $\nabla f(\mathbf{x})$ , 二阶导数则为其 Hessian 矩阵。

(1)  $f(\mathbf{x}) \in \mathcal{C}_L^0(Q)$  表示  $f(\mathbf{x})$  是在定义域  $Q$  上的 Lipschitz 连续函数 (常数  $L \neq 0$ ), 但不可微分。

(2)  $f(\mathbf{x}) \in \mathcal{C}_L^{1,0}(Q)$  表示  $f(\mathbf{x})$  是在定义域  $Q$  上具有常数  $L$  的 Lipschitz 连续函数, 但其梯度不是 Lipschitz 连续的。

(3)  $f(\mathbf{x}) \in \mathcal{C}_L^{1,1}(Q)$  表示  $f(\mathbf{x})$  的梯度  $\nabla f(\mathbf{x})$  是定义域  $Q$  上的 Lipschitz 连续函数。

$\mathcal{C}_L^{k,p}(Q)$  函数类的基本性质是: 若  $f_1 \in \mathcal{C}_{L_1}^{k,p}(Q)$ ,  $f_2 \in \mathcal{C}_{L_2}^{k,p}(Q)$ , 并且  $\alpha, \beta \in \mathbb{R}$ , 则

$$\alpha f_1 + \beta f_2 \in \mathcal{C}_{L_3}^{k,p}(Q)$$

其中  $L_3 = |\alpha| L_1 + |\beta| L_2$ 。

在所有 Lipschitz 连续函数中, 具有 Lipschitz 连续梯度的  $\mathcal{C}_L^{1,1}(Q)$  是最重要的函数类, 广泛应用于凸优化中。

关于  $\mathcal{C}_L^{1,1}(Q)$  函数类, 有下面两个重要的引理<sup>[363]</sup>。

**引理 4.4.1** 函数  $f(\mathbf{x})$  属于  $\mathcal{C}_L^{2,1}(\mathbb{R}^n)$ , 当且仅当

$$\|f''(\mathbf{x})\|_{\text{F}} \leq L, \quad \forall \mathbf{x} \in \mathbb{R}^n \quad (4.4.41)$$

**引理 4.4.2** 若  $f(\mathbf{x}) \in \mathcal{C}_L^{1,1}(Q)$ , 则

$$|f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle| \leq \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in Q \quad (4.4.42)$$

由于一个  $\mathcal{C}_L^{2,1}$  函数一定是一个  $\mathcal{C}_L^{1,1}$  函数, 所以引理 4.4.1 直接给出了判断一个函数是否属于  $\mathcal{C}_L^{1,1}$  函数类的简单方法。

下面是应用引理 4.4.1 判断  $\mathcal{C}_L^{1,1}(Q)$  函数类的几个例子。

线性函数  $f(\mathbf{x}) = \langle \mathbf{a}, \mathbf{x} \rangle + b$  属于  $\mathcal{C}_0^{1,1}$  函数类, 即线性函数的梯度不是 Lipschitz 连续的, 因为

$$f'(\mathbf{x}) = \mathbf{a}, \quad f''(\mathbf{x}) = \mathbf{O} \implies \|f''(\mathbf{x})\|_{\text{F}} = 0$$

二次型函数  $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{a}^T \mathbf{x} + b$  属于  $\mathcal{C}_{\|\mathbf{A}\|_{\text{F}}}^{1,1}(\mathbb{R}^n)$  函数类, 因为

$$\nabla f(\mathbf{x}) = \mathbf{A} \mathbf{x} + \mathbf{a}, \quad \nabla^2 f(\mathbf{x}) = \mathbf{A} \implies \|f''(\mathbf{x})\|_{\text{F}} = \|\mathbf{A}\|_{\text{F}}$$

对数函数  $f(x) = \ln(1 + e^x)$  属  $\mathcal{C}_{1/2}^{1,1}(\mathbb{R})$  函数类, 因为

$$f'(x) = \frac{e^x}{1 + e^x}, \quad f''(x) = \frac{e^x}{(1 + e^x)^2} \implies |f''(x)| = \frac{1}{2} \left| 1 - \frac{1 + e^{2x}}{(1 + e^x)^2} \right| \leq \frac{1}{2}$$

式中, 为了求  $f''(x)$ , 可令  $y = f'(x)$ , 则  $y' = f''(x) = y(\ln y)'$ 。

函数  $f(x) = \sqrt{1 + x^2}$  属  $\mathcal{C}_1^{1,1}(\mathbb{R})$  函数类, 因为

$$f'(x) = \frac{x}{\sqrt{1 + x^2}}, \quad f''(x) = \frac{1}{(1 + x^2)^{3/2}} \implies |f''(x)| \leq 1$$

引理 4.4.2 是分析梯度算法对一个  $C_L^{1,1}$  函数  $f(\mathbf{x})$  的收敛速率的关键不等式。由引理 4.4.2 有<sup>[534]</sup>

$$\begin{aligned} f(\mathbf{x}_{k+1}) &\leq f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{L}{2} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^2 \\ &\leq f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{x}_k\|_2^2 - \frac{L}{2} \|\mathbf{x} - \mathbf{x}_{k+1}\|_2^2 \\ &\leq f(\mathbf{x}) + \frac{L}{2} \|\mathbf{x} - \mathbf{x}_k\|_2^2 - \frac{L}{2} \|\mathbf{x} - \mathbf{x}_{k+1}\|_2^2 \end{aligned}$$

令  $\mathbf{x} = \mathbf{x}^*$  和  $\delta_k = f(\mathbf{x}_k) - f(\mathbf{x}^*)$ , 则

$$\begin{aligned} 0 &\leq \frac{L}{2} \|\mathbf{x}^* - \mathbf{x}_{k+1}\|_2^2 \leq -\delta_{k+1} + \frac{L}{2} \|\mathbf{x}^* - \mathbf{x}_k\|_2^2 \\ &\leq \cdots \leq -\sum_{i=1}^{k+1} \delta_i + \frac{L}{2} \|\mathbf{x}^* - \mathbf{x}_0\|_2^2 \end{aligned}$$

由于梯度投影法的估计误差  $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_{k+1}$ , 所以  $-(\delta_1 + \cdots + \delta_{k+1}) \leq -(k+1)\delta_{k+1}$ , 从而上述不等式可以简化为

$$0 \leq \frac{L}{2} \|\mathbf{x}^* - \mathbf{x}_{k+1}\|_2^2 \leq -(k+1)\delta_k + \frac{L}{2} \|\mathbf{x}^* - \mathbf{x}_0\|_2^2$$

即得梯度投影法的收敛速率上界<sup>[534]</sup>

$$\delta_k = f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{L \|\mathbf{x}^* - \mathbf{x}_0\|_2^2}{2(k+1)} \quad (4.4.43)$$

由函数类  $C_L^{k,p}(Q)$ , 可以进一步引出凸优化中常用的两类目标函数:

(1)  $\mathcal{F}_L^{k,p}(Q)$  表示可  $k$  次微分, 其  $p$  阶导数为 Lipschitz 连续的凸函数, 并且 Lipschitz 常数为  $L$ 。

(2)  $\mathcal{S}_{\mu,L}^{k,p}(Q)$  表示可  $k$  次微分, 其  $p$  阶导数为 Lipschitz 连续的强凸函数。其中,  $\mathcal{S}_{\mu,L}^{1,1}(\mathbb{R}^n)$  是最重要的强凸函数类

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu \|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.4.44)$$

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.4.45)$$

比值  $\kappa = L/\mu (\geq 1)$  称为强凸函数  $f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^{k,p}(Q)$  的“条件数”, 因为它描述了强凸函数  $f(\mathbf{x})$  的 Hessian 矩阵的半正定性:  $\mu I_n \preceq \nabla^2 f(\mathbf{x}) \preceq L I_n$ , 即  $\nabla^2 f(\mathbf{x}) - \mu I_n$  和  $L I_n - \nabla^2 f(\mathbf{x})$  分别为半正定矩阵。

无约束凸优化中最重要的函数类型是具有 Lipschitz 连续梯度的凸函数  $\mathcal{F}_L^{1,1}(\mathbb{R}^n)$  和强凸函数  $\mathcal{S}_{\mu,L}^{1,1}(\mathbb{R}^n)$ 。类似地, 约束凸优化问题

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \text{ subject to } f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$$

中最重要的函数类型则是具有 Lipschitz 连续梯度的凸函数  $\mathcal{F}_L^{1,1}(\mathcal{X})$  和强凸函数  $\mathcal{S}_{\mu,L}^{1,1}(\mathcal{X})$ , 其中  $\mathcal{X}$  为闭集

$$\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n | f_i(\mathbf{x}) \leq 0, i = 1, \dots, m\}$$

由文献 [363] 的定理 2.1.7 和定理 2.1.13, 得到一阶优化方法的最优收敛速率如下。

**定理 4.4.1** 对于凸函数  $f(\mathbf{x}) \in \mathcal{F}_L^{\infty,1}(\mathbb{R}^\infty)$  和强凸函数  $f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^{\infty,1}(\mathbb{R}^\infty)$ , 更新序列  $\{\mathbf{x}_k\}$  满足

$$\mathbf{x}_k \in \mathbf{x}_0 + \text{span}\{\mathbf{x}_0, \dots, \mathbf{x}_{k-1}\}$$

的任何一阶方法所能够达到的估计误差  $\varepsilon = f(\mathbf{x}_k) - f(\mathbf{x}^*)$  的下界分别为

$$\mathcal{F}_L^{\infty,1}(\mathbb{R}^\infty): \quad f(\mathbf{x}_k) - f(\mathbf{x}^*) \geq \frac{3L\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{32(k+1)^2} \quad (4.4.46)$$

$$\mathcal{S}_{\mu,L}^{\infty,1}(\mathbb{R}^\infty): \quad f(\mathbf{x}_k) - f(\mathbf{x}^*) \geq \frac{\mu}{2} \left( \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^{2k} \|\mathbf{x}_0 - \mathbf{x}^*\|_2^2 \quad (4.4.47)$$

式中,  $\kappa = \frac{L}{\mu} > 1$ ,  $\text{span}\{\mathbf{u}_0, \dots, \mathbf{u}_{k-1}\}$  表示向量  $\mathbf{u}_0, \dots, \mathbf{u}_{k-1}$  的线性子空间;  $\mathbf{x}_0$  是梯度法的初始值; 而  $f(\mathbf{x}^*)$  代表函数  $f$  的极小值。

文献 [363] 的推论 2.1.2 和定理 2.1.15 可以综合成关于梯度法的收敛速率如下。

**定理 4.4.2** 对于凸函数  $f(\mathbf{x}) \in \mathcal{F}_L^{1,1}(\mathbb{R}^n)$  和强凸函数  $f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^{1,1}(\mathbb{R}^n)$ , 梯度法  $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k)$  产生的序列  $\{\mathbf{x}_k\}$  给出的目标函数的估计误差  $\varepsilon = f(\mathbf{x}_k) - f(\mathbf{x}^*)$  的上界分别为

$$\mathcal{F}_L^{1,1}(\mathbb{R}^n): \quad f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{2L\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{k+4} \quad (4.4.48)$$

$$\mathcal{S}_{\mu,L}^{1,1}(\mathbb{R}^n): \quad f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{L}{2} \left( \frac{\kappa - 1}{\kappa + 1} \right)^{2k} \|\mathbf{x}_0 - \mathbf{x}^*\|_2^2 \quad (4.4.49)$$

式中,  $\kappa = \frac{L}{\mu} > 1$ 。

比较式 (4.4.48) 和式 (4.4.46) 知, 梯度法对凸目标函数远没有达到最优收敛速率, 因为一阶方法的最优收敛速率为  $O\left(\frac{1}{k^2}\right)$ , 而梯度法收敛速率只是  $O\left(\frac{1}{k}\right)$ 。

下面比较强凸目标函数情况下一阶优化算法和梯度算法的收敛速率。

给定一允许的精度  $f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \varepsilon$ , 并令达到这一精度所需要的最少迭代次数为  $K^*$ , 则由一阶优化算法的收敛速率下界公式 (4.4.47) 得

$$\ln \varepsilon \geq \ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 - 2K^* \ln \frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1}$$

由此得

$$\begin{aligned} K^* &\geq \frac{1}{2 \ln \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}} \left( \ln \frac{1}{\varepsilon} + \ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 \right) \\ &\geq \frac{\sqrt{\kappa}}{4} \left( \ln \frac{1}{\varepsilon} + \ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 \right) \\ &\approx \frac{\sqrt{\kappa}}{4} \ln \frac{1}{\varepsilon} \end{aligned}$$

其中的近似等式是因为  $\ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 \ll \ln \frac{1}{\varepsilon}$ ; 而第二个不等式则利用了  $\ln \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1} > \frac{2}{\sqrt{\kappa}}$ , 因为

$$\ln \frac{x+1}{x-1} = 2 \left( \frac{1}{x} + \frac{1}{3x^3} + \frac{1}{5x^5} + \dots \right) > \frac{2}{x}, \quad x > 1$$

由  $K^* \geq \frac{\sqrt{\kappa}}{4} \ln \frac{1}{\varepsilon}$  知, 一阶优化算法的最优收敛速率为  $O\left(\frac{\sqrt{\kappa}}{4} \ln \frac{1}{\varepsilon}\right)$ 。类似地, 对于式 (4.4.49) 所示的优化算法的收敛速率的上界, 则有

$$\begin{aligned} K &\leq \frac{1}{2 \ln \frac{\kappa+1}{\kappa-1}} \left( \ln \frac{1}{\varepsilon} + \ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 \right) \\ &\leq \frac{\kappa}{4} \left( \ln \frac{1}{\varepsilon} + \ln \frac{\mu}{2} + 2 \ln \|\mathbf{x}_0 - \mathbf{x}\|_2 \right) \approx \frac{\kappa}{4} \ln \frac{1}{\varepsilon} \end{aligned}$$

即是说, 当  $K \leq \frac{\kappa}{4} \ln \frac{1}{\varepsilon}$  时, 梯度算法的估计精度  $|f(\mathbf{x}_K) - f(\mathbf{x}^*)| \geq \varepsilon$ 。为了保证  $|f(\mathbf{x}_K) - f(\mathbf{x}^*)| \leq \varepsilon$ , 迭代次数必须满足下列条件

$$K \geq \frac{\kappa}{4} \ln \frac{1}{\varepsilon}$$

因此, 对于强凸目标函数, 梯度算法的收敛速率为  $O\left(\frac{\kappa}{4} \ln \frac{1}{\varepsilon}\right)$ , 明显比一阶优化算法的最优收敛速率为  $O\left(\frac{\sqrt{\kappa}}{4} \ln \frac{1}{\varepsilon}\right)$  慢, 若条件数  $\kappa$  明显比 1 大。

梯度法远不是最优的这一缺陷促使人们致力于如何加速梯度法。收敛速率比梯度法快速的一类方法为重球法。特别地, 20 世纪 80 年代初期, Nesterov 提出了一种最优梯度法 (optimal gradient method), 优美地解决了梯度法的收敛问题。

重球法 (heavy ball method, HBM) 是一种二步方法 (two-step method): 令  $\mathbf{p}_0$  和  $\mathbf{x}_0$  是两个初始向量,  $\alpha_k$  和  $\beta_k$  是两个正值的序列, 则求解无约束最小化  $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$  的一阶方法可以采用二步更新<sup>[411]</sup>

$$\mathbf{p}_k = -\nabla f(\mathbf{x}_k) + \beta_k \mathbf{p}_{k-1} \quad (4.4.50)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \quad (4.4.51)$$

特别地, 若令  $\mathbf{p}_0 = \mathbf{0}$ , 则上述二步更新可以改写为一步更新

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k) + \beta_k (\mathbf{x}_k - \mathbf{x}_{k-1}) \quad (4.4.52)$$

式中  $\mathbf{x}_k - \mathbf{x}_{k-1}$  称为动量 (momentum)。

下面介绍 Nesterov 最优梯度法, 它与重球法不谋而合。

**定义 4.4.2**<sup>[363]</sup> 一对序列  $\{\phi_k(\mathbf{x})\}_{k=0}^{\infty}$  和  $\{\lambda_k\}_{k=0}^{\infty}, \lambda_k \geq 0$  称为函数  $f(\mathbf{x})$  的估计序列 (estimation sequence), 若  $\lambda_k \rightarrow 0$ , 并且对于任何  $\mathbf{x} \in \mathbb{R}^n$  和所有  $k \geq 0$ , 有

$$\phi_k(\mathbf{x}) \leq (1 - \lambda_k)f(\mathbf{x}) + \lambda_k \phi_0(\mathbf{x}) \quad (4.4.53)$$

**引理 4.4.3**<sup>[363]</sup> 若对某个序列  $\{\mathbf{x}_k\}$ , 有

$$f(\mathbf{x}_k) \leq \phi_k^* \equiv \min_{\mathbf{x} \in \mathbb{R}^n} \phi_k(\mathbf{x}) \quad (4.4.54)$$

则  $f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \lambda_k [\phi_0(\mathbf{x}^*) - f(\mathbf{x}^*)] \rightarrow 0$ 。

引理 4.4.3 表明, 一个估计序列如果满足条件式 (4.4.54), 则  $f(\mathbf{x}_k)$  将收敛为目标函数  $f(\mathbf{x})$  的最小值  $f(\mathbf{x}^*)$ 。于是, 凸目标函数  $f(\mathbf{x})$  的最小化需要解决以下两个问题:

- (1) 如何构造估计序列  $\{\phi_k(\mathbf{x})\}$  和  $\{\lambda_k\}$ ?
- (2) 如何保证估计序列满足条件式 (4.4.54)?

问题 (1) 的答案比较简单。假定目标函数  $f(\mathbf{x})$  (其中  $\mathbf{x} \in B_n$ ) 是一个具有凸性参数  $\mu$  的闭强凸函数

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f$$

并且  $f(\mathbf{x})$  的梯度  $\nabla f(\mathbf{x})$  是 Lipschitz 连续的 (Lipschitz 常数为  $L$ )

$$|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})| \leq L \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in B_n$$

则由

$$\lambda_{k+1} = (1 - \alpha_k) \lambda_k \tag{4.4.55}$$

$$\phi_{k+1}(\mathbf{x}) = (1 - \alpha_k) \phi_k(\mathbf{x}) + \alpha_k \left[ f(\mathbf{y}_k) + \langle f'(\mathbf{y}_k), \mathbf{x} - \mathbf{y}_k \rangle + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}_k\|_2^2 \right] \tag{4.4.56}$$

递推产生的序列  $\{\phi_k(\mathbf{x})\}$  和  $\{\lambda_k\}$  是目标函数  $f(\mathbf{x})$  的估计序列<sup>[363]</sup>, 其中

- ①  $\{\mathbf{y}_k\}_{k=0}^{\infty}$  是向量空间  $\mathbb{R}^n$  的一任意序列;
- ②  $\alpha_k \in (0, 1)$ ,  $\sum_{k=0}^{\infty} \alpha_k = \infty$ ;
- ③  $\lambda_0 = 1$ ;
- ④  $\phi_0(\mathbf{x})$  是向量空间  $\mathbb{R}^n$  的一任意函数。

问题 (2) 的解决取决于凸性参数  $\mu$  和 Lipschitz 常数  $L$  的灵活应用。

令  $Q_f = L/\mu$  表示目标函数  $f(\mathbf{x})$  的“条件数”。Nesterov 提出选择

$$\begin{aligned} \mathbf{x}_k &= \mathbf{y}_k - t_k f'(\mathbf{y}_k), \quad t_k = \frac{1}{L} \\ \alpha_k^2 &= (1 - \alpha_{k+1}) \alpha_k^2 + \frac{\mu}{L} \alpha_{k+1} \\ \mathbf{y}_k &= \mathbf{x}_k + \beta_k (\mathbf{x}_k - \mathbf{x}_{k-1}) \end{aligned}$$

显然, 第 1 式和第 3 式取重球法的二步更新公式的形式。

Nesterov 的第 1 种最优梯度法如下。

**算法 4.4.6 Nesterov 第 1 最优梯度法**<sup>[361]</sup>

初始化 令  $\mathbf{y}_0 = \mathbf{x}_{-1} \in \mathbb{R}^n$  和  $\alpha_0 = 1$ 。

对  $k = 1, 2, \dots$ , 进行以下迭代, 直到  $\mathbf{x}_k$  收敛:

$$\mathbf{x}_k = \mathbf{y}_k - \frac{1}{L} \nabla f(\mathbf{y}_k) \tag{4.4.57}$$

$$\alpha_{k+1} = \frac{1}{2} \left( 1 + \sqrt{4\alpha_k^2 + 1} \right) \tag{4.4.58}$$

$$\mathbf{y}_{k+1} = \mathbf{x}_k + \frac{\alpha_k - 1}{\alpha_{k+1}} (\mathbf{x}_k - \mathbf{x}_{k-1}) \tag{4.4.59}$$

Nesterov 最优梯度法产生两个序列  $\{\mathbf{x}_k\}$  和  $\{\mathbf{y}_k\}$ , 其中  $\{\mathbf{x}_k\}$  是逼近解序列, 而  $\{\mathbf{y}_k\}$  是搜索点序列。

下面是 Nesterov 第 1 最优梯度法的收敛性能。

**定理 4.4.3** [363, Theorem 2.2.2] 令  $\{\mathbf{x}_k\}$  由 Nesterov 第 1 最优梯度法产生, 其中  $\alpha_0 = \hat{\alpha} + \sqrt{1 + \hat{\alpha}^2}$ , 并且  $\hat{\alpha} = -\frac{1}{2} + \frac{\mu}{L}$ , 则

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq L \left( \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa}} \right)^k \|\mathbf{x}_0 - \mathbf{x}^*\|_2^2, \quad \kappa = \frac{L}{\mu} \quad (4.4.60)$$

欲使估计精度  $f(\mathbf{x}_k) - f(\mathbf{x}^*) \geq \epsilon$ , 则 Nesterov 最优梯度法所需的迭代次数

$$\begin{aligned} K &\leq \ln \frac{1}{\ln \frac{\sqrt{\kappa}}{\sqrt{\kappa}-1}} \left( \ln \frac{1}{\epsilon} + \ln L + 2 \ln \|\mathbf{x}_0 - \mathbf{x}^*\| \right) \\ &\leq \frac{2\sqrt{\kappa} - 1}{2} \left( \ln \frac{1}{\epsilon} + \ln L + 2 \ln \|\mathbf{x}_0 - \mathbf{x}^*\| \right) \\ &\leq \sqrt{\kappa} \left( \ln \frac{1}{\epsilon} + \ln L + 2 \ln \|\mathbf{x}_0 - \mathbf{x}^*\| \right) \end{aligned}$$

这表明, 为了达到  $f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \epsilon$ , Nesterov 最优梯度法的迭代次数为  $O\left(\sqrt{\kappa} \ln \frac{1}{\epsilon}\right)$ , 是一阶最优方法的迭代次数  $O\left(\frac{\sqrt{\kappa}}{4} \ln \frac{1}{\epsilon}\right)$  的 4 倍, 属同一数量级。从这个意义上讲, Nesterov 方法称得上是一种最优的一阶方法。

上述 Nesterov 最优梯度法只适用于无约束最小化  $\min f(\mathbf{x})$ , 其中  $\mathbf{x} \in \mathbb{R}^n$ , 并且  $f$  是具有 Lipschitz 连续梯度的凸函数。

当目标函数的定义域不是向量空间  $\mathbb{R}^n$ , 而是某个凸集  $Q \subset \mathbb{R}^n$  时, Nesterov 最优梯度法需要使用梯度映射作适当修正。

**定义 4.4.3 (梯度映射)** [363] 称  $\mathbf{g}(\bar{\mathbf{x}}; L) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  是具有 Lipschitz 连续梯度的凸函数  $f(\mathbf{x})$  在凸集  $Q$  上的梯度映射, 若

$$\mathbf{x}_Q(\bar{\mathbf{x}}; L) = \arg \min_{\mathbf{x} \in Q} \left( f(\bar{\mathbf{x}}) + \langle \nabla f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle + \frac{L}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|_2^2 \right) \quad (4.4.61)$$

$$\mathbf{g}_Q(\bar{\mathbf{x}}; L) = L(\bar{\mathbf{x}} - \mathbf{x}_Q(\bar{\mathbf{x}}; L)) \quad (4.4.62)$$

Nesterov 最优梯度法应该选择初始值  $\mathbf{x}_0 \in Q$ , 并且式 (4.4.57) 替换为梯度投影

$$\mathbf{x}_{k+1} = \mathcal{P}_Q \left( \mathbf{y}_k - \frac{1}{L} \nabla f(\mathbf{y}_k) \right) \quad (4.4.63)$$

或者替换为梯度映射

$$\mathbf{x}_{k+1} = \mathbf{g}_Q(\mathbf{y}_k; L) \quad (4.4.64)$$

若  $Q \equiv \mathbb{R}^n$ , 则对于梯度投影有  $\mathcal{P}_Q(\mathbf{u}) = \mathbf{u}$ , 而对于梯度映射有  $\mathbf{x}_Q(\mathbf{y}; L) = \mathbf{y} - \frac{1}{L} \nabla f(\mathbf{y})$  和  $\mathbf{g}_Q(\mathbf{y}; L) = \nabla f(\mathbf{y})$ , 故式 (4.4.63) 退化为式 (4.4.57)。

Nesterov 第 1 最优梯度法的两个主要缺点是：①  $\mathbf{y}_k$  有可能位于  $Q$  外，因此要求  $f(\mathbf{x})$  必须是在每一点都是明确定义的 (well-defined)。② 只适用于 Euclidean 范数。为了克服这两个缺点，Nesterov 又提出了以下两种算法。

#### 算法 4.4.7 Nesterov 第 2 最优梯度法<sup>[362]</sup>

$$\begin{aligned}\mathbf{y}_k &= \theta_k \mathbf{z}_k + (1 - \theta_k) \mathbf{x}_k \\ \mathbf{z}_{k+1} &= \arg \min_{\mathbf{z} \in Q} [f(\mathbf{y}_k) + \langle \nabla f(\mathbf{y}_k), \mathbf{z} - \mathbf{y}_k \rangle + \theta_k \cdot L \cdot D(\mathbf{z}, \mathbf{z}_k)] \\ \mathbf{x}_{k+1} &= \theta_k \mathbf{z}_{k+1} + (1 - \theta_k) \mathbf{x}_k\end{aligned}$$

#### 算法 4.4.8 Nesterov 第 3 最优梯度法<sup>[364]</sup>

$$\begin{aligned}\mathbf{y}_k &= \theta_k \mathbf{z}_k + (1 - \theta_k) \mathbf{x}_k \\ \mathbf{z}_{k+1} &= \arg \min_{\mathbf{z} \in Q} \left( \sum_{i=1}^k \frac{1}{\alpha_i} [f(\mathbf{y}_i) + \langle \nabla f(\mathbf{y}_i), \mathbf{z} - \mathbf{y}_i \rangle] + \theta_k \cdot L \cdot d(\mathbf{z}) \right) \\ \mathbf{x}_{k+1} &= \theta_k \mathbf{z}_{k+1} + (1 - \theta_k) \mathbf{x}_k\end{aligned}$$

Nesterov 的上述三种最优梯度法实际上均为投影梯度与重球法的结合算法，它们都具有  $O(1/k^2)$  的收敛速率。

## 4.5 非平滑凸优化的次梯度法

梯度法要求目标函数  $f(\mathbf{x})$  在点  $\mathbf{x}$  存在梯度  $\nabla f(\mathbf{x})$ ；Nesterov 最优梯度法则进一步要求目标函数具有 Lipschitz 连续梯度。因此，梯度法和 Nesterov 最优法只适用于平滑的目标函数。

### 4.5.1 次梯度与次微分

现在考虑非平滑凸目标函数的最小化  $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$ ，其中  $f$  为凸函数，但是非平滑函数，不可微分。

非平滑目标函数的常见例子如  $\|\mathbf{x}\|_1, \|\mathbf{x}\|_*, \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1$  等。

由于非平滑函数  $f(\mathbf{x})$  在  $\mathbf{x}$  的梯度向量不存在，所以梯度算法和 Nesterov 最优梯度法不适用。一个自然会问的问题是：非平滑函数是否存在类似于梯度向量的某种“广义梯度”？

对于一个可二次连续微分的函数  $f(\mathbf{x})$ ，其二阶逼近为

$$f(\mathbf{x} + \Delta\mathbf{x}) \approx f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top \Delta\mathbf{x} + (\Delta\mathbf{x})^\top \mathbf{H} \Delta\mathbf{x}$$

若 Hessian 矩阵  $\mathbf{H}$  半正定或者正定，则有不等式

$$f(\mathbf{x} + \Delta\mathbf{x}) \geq f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top \Delta\mathbf{x}$$

或者

$$f(\mathbf{y}) \geq f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top(\mathbf{y} - \mathbf{x}), \quad \forall \mathbf{x}, \mathbf{y} \in \text{dom } f(\mathbf{x}) \quad (4.5.1)$$

虽然非平滑函数  $f(\mathbf{x})$  不存在梯度向量  $\nabla f(\mathbf{x})$ , 但是有可能找到另外一个向量  $\mathbf{g}$  代替梯度向量之后, 能够满足不等式 (4.5.1)。这种向量虽然不是梯度向量, 却具有类似于梯度向量的作用。为了区别, 称这样的向量为次梯度向量 (subgradient vector)。

**定义 4.5.1** 一向量  $\mathbf{g} \in \mathbb{R}^n$  是函数  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  在点  $\mathbf{x} \in \mathbb{R}^n$  的次梯度向量, 若对所有向量  $\mathbf{y} \in \text{dom}(f)$ , 有

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \quad (4.5.2)$$

显然, 若  $f$  是凸函数和可微分, 则  $f$  在  $\mathbf{x}$  的梯度  $\nabla f(\mathbf{x})$  即是一个次梯度向量。因此, 梯度向量是次梯度向量的特例。一般说来, 一函数在某点  $\mathbf{x}$  的次梯度可能有多个。

下面是次梯度向量  $\mathbf{g}$  的基本性质:

- (1)  $f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x})$  是  $f(\mathbf{y})$  的全局下界 (global lower bound)。
- (2) 若  $f(\mathbf{x})$  是可微分的, 则梯度向量  $\nabla f(\mathbf{x})$  是函数  $f$  在  $\mathbf{x}$  的次梯度。

函数  $f$  在点  $\mathbf{x}$  的所有次梯度的集合称为函数  $f$  在点  $\mathbf{x}$  的次微分

$$\partial f(\mathbf{x}) \stackrel{\text{def}}{=} \bigcap_{\mathbf{y} \in \text{dom } f} \{ \mathbf{g} | f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \} \quad (4.5.3)$$

函数  $f(\mathbf{x})$  称为在点  $\mathbf{x}$  是可次微分的 (subdifferentiable), 若它至少存在一个次梯度向量。函数  $f$  在定义域上是可次微分的, 若它在所有点  $\mathbf{x} \in \text{dom } f$  是可次微分的。

**例 4.5.1** 函数  $f(x) = |x|$  在  $x$  非平滑, 不存在梯度  $\nabla|x|$ , 但存在次梯度。为了求次梯度, 将函数改写为  $f(s, x) = |x| = s \cdot x$ 。由此立即知函数  $f(s, x)$  的梯度  $\frac{\partial f(s, x)}{\partial x} = s$ , 并且  $s = -1$  若  $x < 0$ ;  $s = +1$  若  $x > 0$ 。此外, 若  $x = 0$ , 则由次梯度定义知, 应该满足  $|y| \geq gy$ , 即  $g \in [-1, +1]$ 。因此, 函数  $|x|$  的次微分

$$\partial|x| = \begin{cases} \{-1\}, & x < 0 \\ \{+1\}, & x > 0 \\ [-1, 1], & x = 0 \end{cases}$$

次微分的基本性质如下 [56]:

- (1) 次微分的凸性  $\partial f(\mathbf{x})$  总是闭凸集, 即使  $f(\mathbf{x})$  不是凸函数。
- (2) 非空与有界性 若  $\mathbf{x} \in \text{int}(\text{dom } f)$ , 则次微分  $\partial f(\mathbf{x})$  是非空的和有界的。
- (3) 凸函数的次微分 若  $f$  在点  $\mathbf{x}$  是凸的和可微分的, 则次微分是单元素集  $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$ , 即其梯度是其唯一的次梯度。反之, 若  $f$  是凸函数, 并且  $\partial f(\mathbf{x}) = \{\mathbf{g}\}$ , 则  $f$  在  $\mathbf{x}$  是可微分的, 并且  $\mathbf{g} = \nabla f(\mathbf{x})$ 。
- (4) 非负因子 若  $\alpha > 0$ , 则  $\partial(\alpha f(\mathbf{x})) = \alpha \partial f(\mathbf{x})$ 。

(5) 不可微分函数的极小点 点  $\mathbf{x}^*$  是凸函数  $f$  的一个极小点, 当且仅当  $f$  在  $\mathbf{x}^*$  可次微分, 并且

$$\mathbf{0} \in \partial f(\mathbf{x}^*) \quad (4.5.4)$$

即  $\mathbf{g}(\mathbf{x}^*) = \mathbf{0}$  是  $f$  在  $\mathbf{x}^*$  的次梯度。若  $f$  是可微分的, 则一阶条件  $\mathbf{0} \in \partial f(\mathbf{x})$  退化为  $\nabla f(\mathbf{x}) = \mathbf{0}$ 。

(6) 凸函数之和的次微分 若  $f_1, \dots, f_m$  均为凸函数, 则函数  $f(\mathbf{x}) = f_1(\mathbf{x}) + \dots + f_m(\mathbf{x})$  的次微分

$$\partial f(\mathbf{x}) = \partial f_1(\mathbf{x}) + \dots + \partial f_m(\mathbf{x})$$

(7) 仿射变换的次微分 若  $h(\mathbf{x}) = f(A\mathbf{x} + \mathbf{b})$ , 则次微分  $\partial h(\mathbf{x}) = A^T \partial f(A\mathbf{x} + \mathbf{b})$ 。

(8) 逐点极大函数的次微分 令  $f$  是凸函数  $f_1, \dots, f_m$  的逐点极大函数, 即

$$f(\mathbf{x}) = \max_{i=1, \dots, m} f_i(\mathbf{x})$$

则

$$\partial f(\mathbf{x}) = \text{conv} \left( \bigcup \{ \partial f_i(\mathbf{x}) \mid f_i(\mathbf{x}) = f(\mathbf{x}) \} \right)$$

即逐点极大函数  $f$  的次微分是“作用函数”(active function)  $f_i(\mathbf{x})$  在点  $\mathbf{x}$  的次微分的并集的凸包。

例如, 若  $f = \max_{i=1, \dots, m} f_i(\mathbf{x})$ , 并且  $f_i$  是凸函数和可微分的, 则

$$\partial f(\mathbf{x}) = \text{conv} \{ \nabla f_i(\mathbf{x}) \mid f_i(\mathbf{x}) = f(\mathbf{x}) \}$$

**例 4.5.2**  $L_1$  范数  $f(\mathbf{x}) = \|\mathbf{x}\|_1 = |x_1| + \dots + |x_m|$  是在  $\mathbf{x}$  不可微分的凸函数。根据定义 4.5.1, 次梯度应该满足  $f(\mathbf{x}) \geq f(\mathbf{0}) + \mathbf{g}^T(\mathbf{x} - \mathbf{0})$  即  $\|\mathbf{x}\|_1 \geq \mathbf{g}^T \mathbf{x}$ , 或用元素形式写作

$$\sum_{i=1}^m |x_i| \geq \sum_{i=1}^m g_i x_i$$

显然, 为了满足  $|x_i| \geq g_i x_i$ , 次梯度向量的元素  $g_i$  的取值必须满足  $g_i \in [-1, 1]$ 。于是, 有

$$g_i = \begin{cases} 1, & \text{若 } x_i > 0 \\ -1, & \text{若 } x_i < 0 \\ [-1, 1], & \text{若 } x_i = 0 \end{cases}$$

上式表明: ① 任何一个次梯度向量的元素最大绝对值都不可能大于 1, 即  $\|\mathbf{g}\|_\infty \leq 1$ ; ② 任何一个次梯度向量与变元向量  $\mathbf{x}$  的内积都等于  $L_1$  范数  $\|\mathbf{x}\|_1$ , 即有  $\mathbf{g}^T \mathbf{x} = \|\mathbf{x}\|_1$ 。因此,  $L_1$  范数  $\|\mathbf{x}\|_1$  的次微分为

$$\partial \|\mathbf{x}\|_1 = \{ \mathbf{g} \mid \|\mathbf{g}\|_\infty \leq 1, \mathbf{g}^T \mathbf{x} = 1 \} \quad (4.5.5)$$

**例 4.5.3** 和例 4.5.2 一样, Euclidean 范数平方

$$f(\mathbf{x}) = \|\mathbf{x}\|_2^2 = \sum_{i=1}^m x_i^2$$

的次梯度向量也必须满足  $f(\mathbf{x}) \geq f(\mathbf{0}) + \mathbf{g}^T(\mathbf{x} - \mathbf{0})$ , 即有

$$\sum_{i=1}^m x_i^2 \geq \sum_{i=1}^m g_i x_i$$

显然, 为了让所有  $x_i$  都满足  $x_i^2 \geq g_i x_i$ ,  $g_i$  应取值  $g_i = x_i$ 。换言之, 次梯度向量  $\mathbf{g} = \mathbf{x}$ , 即 Euclidean 范数平方的次微分

$$\partial\|\mathbf{x}\|_2^2 = \{\mathbf{x}\} = \{\nabla\|\mathbf{x}\|_2^2\} \quad (4.5.6)$$

令  $\mathbf{X} \in \mathbb{R}^{m \times n}$  是一任意矩阵, 其奇异值分解为  $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$ , 则矩阵  $\mathbf{X}$  的核范数(即所有奇异值之和)的次微分为<sup>[86, 314, 509]</sup>

$$\partial\|\mathbf{X}\|_* = \{\mathbf{U}\mathbf{V}^T + \mathbf{W} | \mathbf{W} \in \mathbb{R}^{m \times n}, \mathbf{U}^T\mathbf{W} = \mathbf{O}, \mathbf{W}\mathbf{V} = \mathbf{O}, \|\mathbf{W}\|_{\text{spec}} \leq 1\} \quad (4.5.7)$$

$\|\mathbf{x}\|_\infty$  的次微分为<sup>[428, 107]</sup>

$$\partial\|\mathbf{x}\|_\infty = \begin{cases} \{\mathbf{y} : \|\mathbf{y}\|_1 \leq 1\}, & \mathbf{x} = \mathbf{0} \\ \text{conv}\{\text{sgn}(x_i)\mathbf{e}_i : |x_i| = \|\mathbf{x}\|_\infty\}, & \mathbf{x} \neq \mathbf{0} \end{cases} \quad (4.5.8)$$

其中 conv 表示凸锥,  $\mathbf{e}_i$  是一基本向量(其元素  $e_i = 1$ , 其他元素全部为零的向量)。

#### 4.5.2 迫近函数

**定义 4.5.2**<sup>[498]</sup> 函数  $d(\mathbf{x})$  称为闭合凸集  $C$  的迫近函数(proximity function), 若:

(1)  $d(\mathbf{x})$  在  $C$  上连续, 并且是  $C$  上的强凸函数。

(2)  $C \subseteq \text{dom}(d(\mathbf{x}))$ 。

令  $C$  是向量空间  $E$  内的一闭合凸集。若  $d(\mathbf{x})$  是集合  $C$  的一迫近函数, 则  $d(\mathbf{x})$  是一连续的强凸函数

$$d(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha d(\mathbf{x}) + (1 - \alpha)d(\mathbf{y}) - \frac{1}{2}\mu\alpha(1 - \alpha)\|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.5.9)$$

其中  $\mu \geq 0$  为凸性参数。

迫近函数的中心称为集合  $C$  的迫近中心(prox-center), 用符号  $\mathbf{x}_0$  表示, 定义为

$$\mathbf{x}_0 = \arg \min_{\mathbf{x}} \{d(\mathbf{x}) | \mathbf{x} \in C\} \quad (4.5.10)$$

迫近函数  $d(\mathbf{x})$  度量  $\mathbf{x}$  到迫近中心的“距离”。

不失一般性, 假定迫近中心的迫近函数等于零, 即  $d(\mathbf{x}_0) = 0$ 。

称迫近函数是归一化迫近函数, 若以下两个条件满足: ① 强凸性常数  $\mu = 1$ , 即  $\inf_{\mathbf{x} \in C} d(\mathbf{x}) = 0$ 。②  $d(\mathbf{x}) \geq \frac{1}{2}\|\mathbf{x} - \mathbf{x}_0\|_2^2, \forall \mathbf{x} \in C$ 。

下面是几个典型的迫近函数<sup>[366, 498]</sup>:

(1) Euclidean 范数  $f(\mathbf{x}) = \|\mathbf{x}\|_2^2$  的逼近函数

$$d(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}_0\|_2^2$$

式中  $\mathbf{x}_0 \in C$  为逼近中心。

(2)  $L_1$  范数  $f(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$  的逼近函数

$$d(\mathbf{x}) = \|\mathbf{x} - \mathbf{x}_0\|_1 = \sum_{i=1}^n |x^{(i)} - x_0^{(i)}|, \quad \mathbf{x}_0 \in C$$

(3) Frobenius 范数  $f(\mathbf{X}) = \|\mathbf{X}\|_F$  的逼近函数

$$d(\mathbf{X}) = \frac{1}{2} \|\mathbf{X} - \mathbf{X}_0\|_F^2$$

(4) 熵函数  $f(\mathbf{x}) = \|\mathbf{x}\|_1 = 1$  的逼近函数 若  $C \stackrel{\text{def}}{=} \{\mathbf{x} \in \mathbb{R}_+^n : \sum_{i=1}^n x_i = 1\} = \{\mathbf{x} \succeq \mathbf{0} | \mathbf{1}^T \mathbf{x} = 1\}$ , 则

$$d(\mathbf{x}) = \ln n + \sum_{i=1}^n x_i \ln x_i$$

是在  $C$  上的强凸函数, 且凸性参数  $\mu = 1$ , 逼近中心  $\mathbf{x}_0 = \frac{1}{n} \mathbf{1} = [\frac{1}{n}, \dots, \frac{1}{n}]^T$ 。

### 4.5.3 共轭函数

若令  $E^*$  表示  $E$  上所有线性函数构造的空间, 则  $E$  和  $E^*$  分别称为原始向量空间和对偶向量空间。

**定义 4.5.3** 实值函数  $g(\mathbf{x})$  的共轭函数记作  $g^*(\mathbf{y})$ , 定义为

$$g^*(\mathbf{y}) \stackrel{\text{def}}{=} \sup_{\mathbf{x} \in \text{dom } g} (\mathbf{y}^T \mathbf{x} - g(\mathbf{x})) = \sup_{\mathbf{x} \in \text{dom } g} (\langle \mathbf{y}, \mathbf{x} \rangle - g(\mathbf{x})) \quad (4.5.11)$$

共轭函数也称对偶函数。共轭函数有着有趣的经济意义解释<sup>[541]</sup>: 令  $\mathbf{x} = [x_1, \dots, x_n]^T$  表示  $n$  件产品组成的产量向量,  $g(\mathbf{x})$  表示生产这些产品的成本。若  $y_i$  代表产品  $x_i$  的价格, 则收入与成本之差  $\sum_{i=1}^n y_i x_i - g(\mathbf{x}) = \mathbf{y}^T \mathbf{x} - g(\mathbf{x})$  即代表生产这  $n$  件产品的利润。利润的最大可能值作为价格的函数, 由共轭函数  $g^*(\mathbf{y}) = \sup(\mathbf{y}^T \mathbf{x} - g(\mathbf{x}))$  决定, 使共轭函数最大化的变元向量  $\mathbf{y}$  就是使利润最大化的  $n$  个产品的价格向量 (price vector)。因此, 共轭函数  $g^*(\mathbf{y})$  可以理解为一收益函数。

对偶空间上的函数  $s \in E^*$  在原始空间的点  $\mathbf{x} \in E$  的值记为  $\langle s, \mathbf{x} \rangle$ 。

给定一正定的自伴算子 (self-adjoint operator) 或自共轭算子 (self-conjugate operator)  $\mathbf{B} : E \rightarrow E^*$ 。由于  $s = \mathbf{B}\mathbf{x}$ , 原始空间和对偶空间上向量的 Euclidean 范数分别定义为

$$\|\mathbf{x}\| = \langle \mathbf{x}, \mathbf{B}\mathbf{x} \rangle^{1/2} = \langle \mathbf{x}, s \rangle^{1/2}, \quad \mathbf{x} \in E \quad (4.5.12)$$

$$\|s\|^* = \langle s, \mathbf{B}^{-1/2} s \rangle^{1/2} = \langle s, \mathbf{x} \rangle^{1/2}, \quad s \in E^* \quad (4.5.13)$$

在坐标向量空间  $E = \mathbb{R}^n$  的特殊情况下, 由于  $E = E^*$  或  $B = I$ , 故上述 Euclidean 范数退化为标准形式的 Euclidean 范数:  $\|\mathbf{x}\| = \langle \mathbf{x}, \mathbf{x} \rangle^{1/2}, \forall \mathbf{x} \in \mathbb{R}^n$ 。

若  $g^*(\mathbf{y})$  是函数  $g(\mathbf{x})$  的共轭函数, 则原始函数  $g(\mathbf{x})$  反过来也可以视为  $g^*(\mathbf{y})$  的共轭函数。换言之, 函数  $g(\mathbf{x})$  的二次共轭函数 (共轭函数的共轭) 就是函数  $g(\mathbf{x})$  本身:  $g^{**}(\mathbf{x}) = (g^*(\mathbf{x}))^* = g(\mathbf{x})$ 。

函数及其共轭函数服从 Fenchel 不等式

$$g(\mathbf{x}) + g^*(\mathbf{y}) \geq \mathbf{x}^T \mathbf{y}, \quad \forall \mathbf{x}, \mathbf{y} \quad (4.5.14)$$

共轭函数  $g^*(\mathbf{y})$  具有以下重要性质 [498]:

- (1) 共轭函数  $g^*(\mathbf{y})$  一定是闭凸函数, 即使  $g$  不是。
- (2) 若  $g(\mathbf{x})$  是闭强凸函数, 则共轭函数  $g^*(\mathbf{y})$  是明确定义的, 并且在所有  $\mathbf{y}$  点都是可微分的, 其梯度向量为

$$\nabla g^*(\mathbf{y}) = \arg \min_{\mathbf{x}} (\mathbf{y}^T \mathbf{x} - g(\mathbf{x})) \quad (4.5.15)$$

(3) 梯度向量  $\nabla g^*(\mathbf{y})$  是 Lipschitz 连续的, 并且 Lipschitz 常数为  $1/L$ , 即有

$$\|\nabla g^*(\mathbf{u}) - \nabla g^*(\mathbf{v})\|_2 \leq \frac{1}{L} \|\mathbf{u} - \mathbf{v}\|_2 \quad (4.5.16)$$

下面是几个共轭函数的例子 [498]:

- (1) 负对数函数  $g(x) = -\log x$  的共轭函数为

$$g^*(\mathbf{y}) = \sup_{x>0} (xy + \log x) = \begin{cases} -1 - \log(-y), & y < 0 \\ \infty, & \text{其他} \end{cases}$$

- (2) 二次函数  $g(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x}$  (其中  $\mathbf{Q}$  正定) 的共轭函数为

$$g^*(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } g} (\mathbf{y}^T \mathbf{x} - \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x}) = \frac{1}{2} \mathbf{y}^T \mathbf{Q}^{-1} \mathbf{y}$$

- (3) 集合  $C$  的指示函数 (indicator function)

$$I_C(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in C \\ +\infty, & \text{其他} \end{cases} \quad (4.5.17)$$

的共轭函数是集合  $C$  的支撑函数  $S_C(\mathbf{y})$ , 即有

$$I_C^*(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } f} (\mathbf{y}^T \mathbf{x} - I_C(\mathbf{x})) = \sup_{\mathbf{x} \in C} \mathbf{y}^T \mathbf{x} = S_C(\mathbf{y}) \quad (4.5.18)$$

特别地, 向量范数  $\|\mathbf{x}\|$  的共轭函数  $\|\mathbf{y}\|^*$  称为范数  $\|\mathbf{x}\|$  的对偶范数

$$\|\mathbf{y}\|^* = \sup_{\|\mathbf{x}\| \leq 1} \mathbf{y}^T \mathbf{x} \quad (4.5.19)$$

因此, 对偶范数  $\|\mathbf{y}\|^*$  是单位球范数  $\|\mathbf{x}\| \leq 1$  的支撑函数。

(4) 向量范数  $g(\mathbf{x}) = \|\mathbf{x}\|$  的共轭函数  $g^*(\mathbf{y}) = \|\mathbf{y}\|^*$  是对偶单位范数球  $\|\mathbf{y}\|^* \leq 1$  上的指示函数

$$g^*(\mathbf{y}) = \sup_{\mathbf{x}} (\mathbf{y}^T \mathbf{x} - \|\mathbf{x}\|) = \begin{cases} 0, & \|\mathbf{y}\|^* \leq 1 \\ +\infty, & \text{其他} \end{cases} \quad (4.5.20)$$

下面是三组常用的向量范数-对偶向量范数对

$$(\|\mathbf{x}\|_2, \|\mathbf{y}\|_2), \quad (\|\mathbf{x}\|_1, \|\mathbf{y}\|_\infty), \quad \left( \sqrt{\mathbf{x}^T \mathbf{Q} \mathbf{x}}, \sqrt{\mathbf{y}^T \mathbf{Q}^{-1} \mathbf{y}} \right) \quad (\mathbf{Q} \text{ 正定})$$

以及两组常用的矩阵范数-对偶矩阵范数对

$$(\|\mathbf{X}\|_F, \|\mathbf{Y}\|_F), \quad \left( \|\mathbf{X}\|_2 = \sigma_{\max}(\mathbf{X}), \|\mathbf{Y}\|_* = \sum_{i=1}^n \sigma_i(\mathbf{X}) \right)$$

共轭函数与次梯度之间的关系<sup>[498]</sup> 若  $f(\mathbf{x})$  为闭凸函数，则

$$\mathbf{y} \in \partial f(\mathbf{x}) \iff \mathbf{x} \in \partial f^*(\mathbf{y}) \iff \mathbf{x}^T \mathbf{y} = f(\mathbf{x}) + f^*(\mathbf{y}) \quad (4.5.21)$$

即是说，若  $\mathbf{y}$  是函数  $f(\mathbf{x})$  的次梯度，则  $\mathbf{x}$  一定是共轭函数  $f^*(\mathbf{y})$  的次梯度。因此，向量  $\mathbf{x}$  与  $\mathbf{y}$  的内积等于函数  $f(\mathbf{x})$  与共轭函数  $f^*(\mathbf{y})$  之和。

指示函数  $I_C(\mathbf{x})$  的次微分是  $C$  在  $\mathbf{x}$  的正规锥 (normal cone)  $N_C(\mathbf{x})$ <sup>[498]</sup>

$$\partial I_C(\mathbf{x}) = N_C(\mathbf{x}) = \{ \mathbf{s} | \mathbf{s}^T (\mathbf{y} - \mathbf{x}) \leq 0, \forall \mathbf{y} \in C \}$$

#### 4.5.4 原始-对偶次梯度算法

将平滑凸函数极小化的梯度算法中的梯度  $\nabla f(\mathbf{x})$  换成次梯度  $\mathbf{g}(\mathbf{x})$ ，即得非平滑凸函数  $f(\mathbf{x})$  极小化的次梯度算法

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k, \quad k \geq 0 \quad (4.5.22)$$

或搜索方向归一化的次梯度算法

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k / \|\mathbf{g}_k\|_2, \quad k \geq 0 \quad (4.5.23)$$

其中， $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$  为非平滑函数  $f(\mathbf{x})$  在  $\mathbf{x}_k$  的次梯度向量，而步长序列  $\{\alpha_k\}_{k=0}^\infty$  必须满足发散级数规则 (divergent-series rule)<sup>[366]</sup>

$$\alpha_k > 0, \quad \alpha_k \rightarrow 0, \quad \sum_{k=0}^{\infty} \alpha_k = \infty \quad (4.5.24)$$

注意，次梯度算法不是一种梯度下降法，因为次梯度算法不能保证  $f(\mathbf{x}_k) \leq f(\mathbf{x}_{k-1})$ ，而只能跟踪截至  $k$  步迭代时的最优点

$$f_k^{\text{best}} = \min\{f(\mathbf{x}_1), \dots, f(\mathbf{x}_k)\}$$

式(4.5.22)和式(4.5.23)分别称为原始次梯度算法和原始归一化次梯度算法。由于目标函数  $f(\mathbf{x})$  非平滑, 所以不能指望其次梯度在最优解点的邻域等于零向量。于是, 为了保证原始序列  $\{\mathbf{x}_k\}_{k=0}^{\infty}$  的收敛, 原始次梯度算法(4.5.22)或原始归一化次梯度算法(4.5.23)中的步长必须满足收敛条件  $\alpha_k \rightarrow 0$ 。这意味着, 对原始次梯度加权的系数  $\alpha_k$  必须随迭代次数  $k$  而递减。

“权系数  $\alpha_k$  必须是递减的”这一规则与迭代算法的一般原理相矛盾: 因为随着迭代的进行, 新的信息应该比旧的信息更加重要, 所以对新信息的加权不应该递减。

为了克服权系数递减带来的问题, Nesterov 提出了原始-对偶次梯度算法<sup>[366]</sup>。这一算法的基本思想是使用两个权系数序列作为控制序列: 第一个序列负责控制对偶空间的支撑函数(support function), 第二个序列控制对偶空间与原始空间之间的动态更新。

定义闭凸集  $C$  的支撑函数的逼近型逼近<sup>[366]</sup>

$$V_{\beta}(\mathbf{s}) = \max_{\mathbf{x} \in C} \{ \langle \mathbf{s}, \mathbf{x} - \mathbf{x}_0 \rangle - \beta d(\mathbf{x}) \} = \min_{\mathbf{x} \in C} \{ -\langle \mathbf{s}, \mathbf{x} - \mathbf{x}_0 \rangle + \beta d(\mathbf{x}) \} \quad (4.5.25)$$

**引理 4.5.1**<sup>[366]</sup> 函数  $V_{\beta}(\mathbf{s})$  是在对偶空间  $E^*$  的凸函数, 并且可微分。此外, 其梯度是 Lipschitz 连续的 (Lipschitz 常数为  $\frac{1}{\beta\mu}$ ), 即有

$$\|\nabla V_{\beta}(\mathbf{s}_1) - \nabla V_{\beta}(\mathbf{s}_2)\| \leq \frac{1}{\beta\mu} \|\mathbf{s}_1 - \mathbf{s}_2\|_*, \quad \forall \mathbf{s}_1, \mathbf{s}_2 \in E^* \quad (4.5.26)$$

式中  $\mu$  是逼近函数  $d(\mathbf{x})$  的凸性参数。对任何  $\mathbf{s} \in E^*$ , 梯度向量  $\nabla V_{\beta}(\mathbf{s})$  属于  $C$ , 即

$$\nabla V_{\beta}(\mathbf{s}) = \pi_{\beta}(\mathbf{s}) - \mathbf{x}_0 \quad (4.5.27)$$

其中

$$\pi_{\beta}(\mathbf{s}) \stackrel{\text{def}}{=} \arg \min_{\mathbf{x} \in C} \{ -\langle \mathbf{s}, \mathbf{x} \rangle + \beta d(\mathbf{x}) \} \quad (4.5.28)$$

**算法 4.5.1** Nesterov 原始-对偶次梯度算法(对偶平均算法)<sup>[366]</sup>

初始化  $\mathbf{s}_0 = \mathbf{0} \in E^*$ , 选择  $\beta_0 > 0$ 。

迭代 ( $k \geq 0$ ):

- (1) 计算次梯度向量  $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ ;
- (2) 选择  $\alpha_k > 0$ , 令  $\mathbf{s}_{k+1} = \mathbf{s}_k + \alpha_k \mathbf{g}_k$ ;
- (3) 选择  $\beta_{k+1} \geq \beta_k$ , 计算

$$\mathbf{x}_{k+1} = \pi_{\beta_{k+1}}(-\mathbf{s}_{k+1}) = \arg \min_{\mathbf{x} \in C} \{ \langle \mathbf{s}_{k+1}, \mathbf{x} \rangle + \beta_{k+1} d(\mathbf{x}) \}$$

下面是原始次梯度算法与原始-对偶次梯度算法之间的主要区别:

(1) 原始次梯度算法求解原始问题  $\min_{\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\}$ , 而原始-对偶次梯度算法求解对偶问题  $V_{\beta}(\mathbf{s}) = \max_{\mathbf{x} \in C} \{ \langle \mathbf{s}, \mathbf{x} - \mathbf{x}_0 \rangle - \beta d(\mathbf{x}) \} = \min_{\mathbf{x} \in C} \{ -\langle \mathbf{s}, \mathbf{x} - \mathbf{x}_0 \rangle + \beta d(\mathbf{x}) \}$ 。

(2) 原始次梯度算法只用步长序列  $\{\alpha_k\}_{k=0}^{\infty}$  控制原始变量的迭代  $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k$ 。原始-对偶次梯度法则使用两个控制序列: 步长序列  $\{\alpha_k\}_{k=0}^{\infty}$  控制对偶变量  $\mathbf{s}$  的迭代  $\mathbf{s}_{k+1} = \mathbf{s}_k + \alpha_k \mathbf{g}_k$ ; 参数序列  $\{\beta_k\}_{k=0}^{\infty}$  控制对偶变量  $\mathbf{s}_k$  和原始变量  $\mathbf{x}_k$  之间的动态更新。

(3) 在原始次梯度算法中, 步长序列  $\{\alpha_k\}_{k=0}^{\infty}$  必须是递减的; 但在原始-对偶次梯度算法中, 步长序列  $\{\alpha_k\}_{k=0}^{\infty}$  和参数序列  $\{\beta_k\}_{k=0}^{\infty}$  允许是非递减的。

对偶平均算法另有下面的变型。

**算法 4.5.2 Nesterov 原始-对偶次梯度算法 (加权对偶平均算法)** [366]

初始化  $s_0 = \mathbf{0} \in E^*$ , 选择  $\rho > 0$ 。

迭代 ( $k \geq 0$ ):

(1) 计算次梯度向量  $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ ;

(2) 令  $s_{k+1} = s_k + \mathbf{g}_k / \|\mathbf{g}_k\|_*$ ;

(3) 选择  $\beta_{k+1} = \frac{\hat{\beta}_{k+1}}{\rho\sqrt{\mu}}$ , 其中

$$\sqrt{2k-1} \leq \hat{\beta}_k \leq \frac{1}{1+\sqrt{3}} + \sqrt{2k-1}, \quad k \geq 1$$

计算  $\mathbf{x}_{k+1} = \pi_{\beta_{k+1}}(-s_{k+1}) = \arg \min_{\mathbf{x} \in C} \{\langle s_{k+1}, \mathbf{x} \rangle + \beta_{k+1} d(\mathbf{x})\}$ 。

#### 4.5.5 投影次梯度法

令  $f(\mathbf{x})$  是一凸函数, 可能平滑或非平滑。最小化  $f(\mathbf{x})$  的投影次梯度法的更新公式可统一表示为 [53]

$$\mathbf{x}_{k+1} = \mathcal{P}(\mathbf{x}_k - \alpha_k \mathbf{g}_k) \quad (4.5.29)$$

式中  $\mathcal{P}(\mathbf{z})$  表示向量  $\mathbf{z}$  到定义域  $C$  上的投影。

(1) 对于线性等式约束凸优化问题

$$\min f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad (4.5.30)$$

向量  $\mathbf{z}$  到定义域  $\{\mathbf{x} | \mathbf{A}\mathbf{x} = \mathbf{b}\}$  上的投影为

$$\mathcal{P}(\mathbf{z}) = \mathbf{z} - \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} (\mathbf{A}\mathbf{z} - \mathbf{b}) \quad (4.5.31)$$

$$= (\mathbf{I} - \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{A}) \mathbf{z} + \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{b} \quad (4.5.32)$$

将投影  $\mathcal{P}(\mathbf{z})$  代入式 (4.5.29), 并利用  $\mathbf{A}\mathbf{x}_k = \mathbf{b}$ , 立即有 [53]

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \left[ \mathbf{I} - \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{A} \right] \mathbf{g}_k = \mathbf{x}_k - \alpha_k \mathbf{P}_A^\perp \mathbf{g}_k \quad (4.5.33)$$

其中

$$\mathbf{P}_A^\perp = \mathbf{I} - \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{A} \quad (4.5.34)$$

表示到  $\mathbf{A}$  的列空间上的正交投影矩阵。

例如, 若  $f(\mathbf{x}) = \|\mathbf{x}\|_1$ , 则由次梯度  $\mathbf{g} = \text{sgn}(\mathbf{x})$  得

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{P}_A^\perp \text{sgn}(\mathbf{x}_k) \quad (4.5.35)$$

(2) 对于不等式约束凸优化问题

$$\min f_0(\mathbf{x}) \quad \text{subject to} \quad f_i(\mathbf{x}) \leq 0, i = 1, \dots, m \quad (4.5.36)$$

则需要考虑其对偶优化问题

$$\max g(\boldsymbol{\lambda}) \quad \text{subject to} \quad \boldsymbol{\lambda} \succeq \mathbf{0} \quad (4.5.37)$$

若  $\mathbf{h}$  是 Lagrangian 对偶函数  $g(\boldsymbol{\lambda})$  的次梯度，即  $\mathbf{h} \in \partial g(\boldsymbol{\lambda})$ ，则投影次梯度算法的更新公式为<sup>[53]</sup>

$$\boldsymbol{\lambda}_{k+1} = \mathcal{P}_{\mathbb{R}_+}(\boldsymbol{\lambda}_k - \alpha_k \mathbf{h}) = (\boldsymbol{\lambda}_k - \alpha_k \mathbf{h})_+ \quad (4.5.38)$$

式中  $(z)_+ = [\max\{z_1, 0\}, \dots, \max\{z_n, 0\}]^T$ 。

## 4.6 非平滑凸函数的平滑凸优化

4.5 节讨论了求解非平滑函数优化问题的次梯度法，由于目标函数非平滑，不可能是 Lipschitz 连续的，所以次梯度算法不能采用 Nesterov 最优梯度算法。为了能够应用 Nesterov 最优梯度法，必须考虑非平滑函数的平滑优化：将一个非平滑的目标函数用一个平滑函数逼近。

### 4.6.1 非平滑函数的平滑逼近

考虑一组合优化问题

$$\min_{\mathbf{x} \in E} F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}) \quad (4.6.1)$$

其中  $E \subset \mathbb{R}^n$  是一有限维实向量空间，并且

$g : E \rightarrow \mathbb{R}$  为凸函数，在  $E$  上不可微分即非平滑。

$f : \mathbb{R}^n \rightarrow \mathbb{R}$  为连续的平滑凸函数，可微分，其梯度为 Lipschitz 连续函数

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L \|\mathbf{x} - \mathbf{y}\|_2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$$

其中  $L > 0$  为梯度  $\nabla f(\mathbf{x})$  的 Lipschitz 常数。

非平滑目标函数的平滑最小化包含两个基本过程：

(1) 用一可微分函数  $g_\mu$  (被  $\mu$  参数化) 逼近非平滑函数  $g$ 。

(2) 使用 (快速) 梯度算法最小化  $g_\mu$ 。

为了将  $E$  上的非平滑函数  $g$  用另一个平滑函数  $g^*$  逼近，新函数  $g^*$  必须以另一个有限维向量空间  $E^* \subset \mathbb{R}^n$  的向量  $\mathbf{y}$  作为变元，即待确定的平滑函数为  $g^*(\mathbf{y})$ 。

令非平滑函数  $g(\mathbf{x})$  是一闭凸函数，其定义域有界。现在考虑如何利用共轭函数，将非平滑函数转换成平滑函数。这一转换由以下三个步骤组成：

## (1) 定义共轭函数

$$G(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } g} ((\mathbf{A}\mathbf{y} + \mathbf{b})^T \mathbf{x} - g(\mathbf{x})) = g^*(\mathbf{A}\mathbf{y} + \mathbf{b}) \quad (4.6.2)$$

(2) 构造一凸性参数为  $\mu$  的逼近函数  $d(\mathbf{x})$ 。

(3) 用共轭函数  $G(\mathbf{y})$  和逼近函数  $d(\mathbf{x})$  构造平滑逼近函数

$$g_\mu(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } g} ((\mathbf{A}\mathbf{y} + \mathbf{b})^T \mathbf{x} - g(\mathbf{x}) - \mu d(\mathbf{x})) = (g + \mu d)^*(\mathbf{A}\mathbf{y} + \mathbf{b}) \quad (4.6.3)$$

由式 (4.6.3) 知, 平滑逼近函数具有如下性质:

(1)  $g_\mu(\mathbf{y})$  是可微分的, 其梯度

$$\nabla g_\mu(\mathbf{y}) = \frac{\partial g_\mu(\mathbf{y})}{\partial \mathbf{y}} = \mathbf{A}^T \arg \min_{\mathbf{x} \in \text{dom } g} ((\mathbf{A}\mathbf{y} + \mathbf{b})^T \mathbf{x} - g(\mathbf{x}) - \mu d(\mathbf{x})) \quad (4.6.4)$$

(2) 梯度  $\nabla g_\mu(\mathbf{y})$  是 Lipschitz 连续的, 且 Lipschitz 常数为  $\|\mathbf{A}\|_2^2/\mu$ 。

这些性质表明, Nesterov 最优梯度法对平滑逼近函数  $g_\mu(\mathbf{y})$  适用。

## 1. 平滑逼近的精度

如果使用平滑逼近函数  $g_\mu(\mathbf{y})$  逼近原非平滑的函数  $g(\mathbf{x})$ , 则平滑逼近的精度为

$$g(\mathbf{x}) - \mu D \leq g_\mu(\mathbf{y}) \leq g(\mathbf{x})$$

式中  $D = \sup_{\mathbf{x} \in \text{dom } g} d(\mathbf{x}) < \infty$ , 因为  $\text{dom } g$  是有界的, 并且  $\text{dom } g \subseteq \text{dom } d$ 。

非平滑函数的平滑优化的复杂度取决于梯度算法: (快速) 梯度算法达到收敛所需要的迭代次数为  $O(L_\mu/\varepsilon_\mu)$ , 其中

(1)  $L_\mu$  是平滑逼近函数  $g_\mu$  的梯度  $\nabla g_\mu$  的 Lipschitz 常数。

(2)  $\varepsilon_\mu$  是使平滑逼近函数  $g_\mu$  最小化所要求的精度。

Lipschitz 常数  $L_\mu$  与优化精度  $\varepsilon_\mu$  之间的权衡:

(1) 较大的 Lipschitz 常数意味着较弱的平滑度, 给出较精确的逼近。

(2) 较小的 Lipschitz 常数意味着较强的平滑度, 提供较快的收敛。

## 2. 平滑逼近函数的实现

逼近函数常使用 Huber 函数 [247]

$$h_\mu(t) = \begin{cases} t^2/(2\mu), & |t| \leq \mu \\ |t| - \mu/2, & |t| \geq \mu \end{cases} \quad (4.6.5)$$

实现。Huber 函数  $h_\mu(t)$  可以逼近  $|t|$

$$h_\mu(t) \leq |t| \leq h_\mu(t) + \mu/2$$

其中  $\mu$  控制逼近的精度和平滑度:

(1) 逼近的精度  $|t| - \frac{\mu}{2} \leq h_\mu(t) \leq |t|$ 。

(2) 逼近函数的平滑度  $h_\mu''(t) \leq \frac{1}{\mu}$ 。

Huber 函数为平滑函数，其梯度

$$\nabla h_\mu(t) = \begin{cases} t/\mu, & |t| \leq \mu \\ \text{sgn}(t), & |t| > \mu \end{cases}$$

Huber 函数的梯度  $h'_\mu$  是 Lipschitz 连续的 (Lipschitz 常数为  $1/\mu$ )。

下面是几种非平滑函数的逼近表示<sup>[498]</sup>。

(1) 分段线性函数  $g(\mathbf{x}) = \max_{i=1,\dots,m} (\mathbf{a}_i^T \mathbf{x} + b_i)$  的平滑逼近

① 共轭表示  $G(\mathbf{y}) = \sup_{y_i \geq 0, \mathbf{1}^T \mathbf{y} = 1} (\mathbf{A}\mathbf{x} + \mathbf{b})^T \mathbf{y}$

② 逼近函数  $d(\mathbf{y}) = \sum_{i=1}^m y_i \log y_i + \log m$

③ 平滑逼近  $g_\mu(\mathbf{x}) = \mu \log \left( \sum_{i=1}^m e^{(\mathbf{a}_i^T \mathbf{x} + b_i)/\mu} \right) - \mu \log m$

(2)  $L_1$  范数  $g(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1$  的平滑逼近

① 共轭表示  $G(\mathbf{y}) = \sup_{\|\mathbf{y}\|_\infty \leq 1} (\mathbf{A}\mathbf{x} - \mathbf{b})^T \mathbf{y}$

② 逼近函数  $d(\mathbf{y}) = \sum_{i=1}^m w_i y_i^2$  ( $w_i > 1$ )

③ 平滑逼近 (Huber 逼近)  $g_\mu(\mathbf{x}) = \sum_{i=1}^m h_\mu(\mathbf{a}_i^T \mathbf{x} + b_i)$ , 其中  $h_\mu(t)$  为 Huber 函数

(3) 核函数  $g(\mathbf{X}) = \|\mathbf{X}\|_*$  的平滑逼近

① 共轭表示  $G(\mathbf{Y}) = \sup_{\|\mathbf{Y}\|_2 \leq 1} \text{tr}(\mathbf{X}^T \mathbf{Y})$

② 逼近函数  $d(\mathbf{Y}) = \frac{1}{2} \|\mathbf{Y}\|_F^2$

③ 平滑逼近 (Huber 逼近)  $g_\mu(\mathbf{X}) = \sum_i h_\mu(\sigma_i(\mathbf{X}))$

(4) 最大特征值  $g(\mathbf{X}) = \lambda_{\max}(\mathbf{X})$  的平滑逼近

① 共轭表示  $G(\mathbf{Y}) = \sup_{y_i \geq 0, \text{tr}(\mathbf{Y})=1} \text{tr}(\mathbf{X}\mathbf{Y})$

② 逼近函数  $d(\mathbf{Y}) = \sum_{i=1}^n \lambda_i(\mathbf{Y}) \log(\lambda_i(\mathbf{Y})) + \log n$

③ 平滑逼近  $g_\mu(\mathbf{X}) = \mu \log \left( \sum_{i=1}^n e^{\lambda_i(\mathbf{X})/\mu} \right) - \mu \log n$

(5) Chebyshev 逼近  $g(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_\infty$  (其中  $\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$ ) 的平滑逼近

① 共轭表示  $G(\mathbf{u}, \mathbf{v}) = \sup_{(\mathbf{u}, \mathbf{v}) \in Q} \langle \mathbf{u} - \mathbf{v}, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle$ , 其中

$$Q = \{(\mathbf{u}, \mathbf{v}) | \mathbf{u} \succeq \mathbf{0}, \mathbf{v} \succeq \mathbf{0}, \mathbf{1}^T \mathbf{u} + \mathbf{1}^T \mathbf{v} = 1\}$$

② 逼近函数  $d(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^m u_i \log u_i + \sum_{i=1}^m v_i \log v_i + \log 2m$

$$\textcircled{3} \text{ 平滑逼近 } g_\mu(\mathbf{x}) = \mu \sum_{i=1}^m \log \left[ \cosh \left( \frac{\mathbf{a}_i^\top \mathbf{x} - b_i}{\mu} \right) \right]$$

一旦非平滑函数用平滑函数逼近，即可使用 Nesterov 最优梯度法进行平滑函数的最小化。

#### 4.6.2 近似梯度法

令  $C_i = \text{dom } f_i(\mathbf{x}), i = 1, \dots, I$  为  $m$  维 Euclidean 空间  $\mathbb{R}^m$  内的闭凸集， $C = \bigcap_{i=1}^I C_i$  为这些闭凸集的交集。考虑组合优化问题

$$\min_{\mathbf{x} \in C} \sum_{i=1}^I f_i(\mathbf{x}) \quad (4.6.6)$$

其中，闭凸集  $C_i, i = 1, \dots, I$  表示对组合优化问题的解  $\mathbf{x}$  施加的约束。

交集  $C$  分三种情况 [73]：

- (1) 交集  $C$  非空且“小”( $C$  的所有分集非常类似)；
- (2) 交集  $C$  非空且“大”( $C$  的各个分集差异大)；
- (3) 交集  $C$  为空集，这意味着各个约束条件相互矛盾。

组合优化问题式 (4.6.6) 的直接求解一般比较困难。然而，若

$$f_1(\mathbf{x}) = \|\mathbf{x} - \mathbf{x}_0\|, \quad f_i(\mathbf{x}) = I_{C_i}(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in C_i \\ +\infty, & \mathbf{x} \notin C_i \end{cases}$$

则组合优化问题可分割成

$$\min_{\mathbf{x} \in \bigcap_{i=2}^I C_i} \|\mathbf{x} - \mathbf{x}_0\| \quad (4.6.7)$$

与组合优化问题式 (4.6.6) 不同，分割优化问题式 (4.6.7) 可以用投影方法求解。特别地， $C_i$  为凸集时，一个凸目标函数到这些凸集的交集的投影与该目标函数的近似映射密切相关。

**定义 4.6.1** 凸函数  $h(\mathbf{x})$  的近似映射 (proximal mapping) 定义为

$$\text{prox}_h(\mathbf{x}) = \arg \min_{\mathbf{u}} \left( h(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|_2^2 \right) \quad (4.6.8)$$

或者

$$\text{prox}_{\mu h}(\mathbf{x}) = \arg \min_{\mathbf{u}} \left( h(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{u} - \mathbf{x}\|_2^2 \right) \quad (4.6.9)$$

近似映射也称近似算子。近似映射具有以下重要性质 [498]：

- (1) 存在性与唯一性 近似映射总是存在，并且对于所有  $\mathbf{x}$  是唯一的。
- (2) 次梯度特性 (subgradient characterization) 近似映射与次梯度之间存在对应关系

$$\mathbf{u} = \text{prox}_h(\mathbf{x}) \iff \mathbf{x} - \mathbf{u} \in \partial h(\mathbf{u}) \quad (4.6.10)$$

(3) 非扩张映射 (nonexpansive mapping) 近似映射是具有常数 1 的非扩张映射：若  $\mathbf{u} = \text{prox}_h(\mathbf{x})$  和  $\hat{\mathbf{u}} = \text{prox}_h(\hat{\mathbf{x}})$ ，则

$$(\mathbf{u} - \hat{\mathbf{u}})^\top (\mathbf{x} - \hat{\mathbf{x}}) \geq \| \mathbf{u} - \hat{\mathbf{u}} \|_2^2$$

(4) 可分离求和 (separable sum) 函数的逼近映射 若  $h : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \rightarrow \mathbb{R}$  是可分离求和函数, 即  $h(\mathbf{x}_1, \mathbf{x}_2) = h_1(\mathbf{x}_1) + h_2(\mathbf{x}_2)$ , 则

$$\text{prox}_h(\mathbf{x}_1, \mathbf{x}_2) = (\text{prox}_{h_1}(\mathbf{x}_1), \text{prox}_{h_2}(\mathbf{x}_2))$$

(5) 变元的缩放和平移 (scaling and translation of argument) 若  $h(\mathbf{x}) = f(\alpha\mathbf{x} + \mathbf{b})$ , 其中  $\alpha \neq 0$ , 则

$$\text{prox}_h(\mathbf{x}) = \frac{1}{\alpha} (\text{prox}_{\alpha^2 f}(\alpha\mathbf{x} + \mathbf{b}) - \mathbf{b})$$

(6) 共轭函数的逼近映射 若  $h^*(\mathbf{x})$  是函数  $h(\mathbf{x})$  的共轭, 则对于任何  $\mu > 0$ , 共轭函数的逼近映射为

$$\text{prox}_{\mu h^*}(\mathbf{x}) = \mathbf{x} - \mu \text{prox}_{h/\mu}(\mathbf{x}/\mu)$$

若  $\mu = 1$ , 则上式简化为

$$\mathbf{x} = \text{prox}_h(\mathbf{x}) + \text{prox}_{h^*}(\mathbf{x}) \quad (4.6.11)$$

这一分解称为 Moreau 分解。

实变量  $x \in \mathbb{R}$  的软阈值化算子 (soft thresholding operator), 定义为

$$S_\tau[x] = \begin{cases} x - \tau, & x > \tau \\ 0, & |x| \leq \tau \\ x + \tau, & x < -\tau \end{cases} \quad (4.6.12)$$

式中,  $\tau > 0$  称为实变量  $x$  的软阈值。软阈值化算子也可等价写作

$$\begin{aligned} S_\tau[x] &= (x - \tau)_+ - (-x - \tau)_+ = \max\{x - \tau, 0\} - \max\{-x - \tau, 0\} \\ &= (x - \tau)_+ + (x + \tau)_- = \max\{x - \tau, 0\} + \min\{x + \tau, 0\} \end{aligned}$$

实向量  $\mathbf{x} \in \mathbb{R}^n$  的软阈值化算子  $S_\tau[\mathbf{x}]$  是一个  $n$  维实向量, 其元素定义为

$$S_\tau[\mathbf{x}]_i = \max\{x_i - \tau, 0\} + \min\{x_i + \tau, 0\} = \begin{cases} x_i - \tau, & x_i > \tau \\ 0, & |x_i| \leq \tau \\ x_i + \tau, & x_i < -\tau \end{cases}$$

实矩阵  $\mathbf{X} \in \mathbb{R}^{m \times n}$  的软阈值化算子  $S_\tau[\mathbf{X}]$  是一个  $m \times n$  实矩阵, 其元素

$$S_\tau[\mathbf{X}]_{ij} = \max\{X_{ij} - \tau, 0\} + \min\{X_{ij} + \tau, 0\} = \begin{cases} X_{ij} - \tau, & X_{ij} > \tau \\ 0, & |X_{ij}| \leq \tau \\ X_{ij} + \tau, & X_{ij} < -\tau \end{cases}$$

软阈值化算子也称收缩算子 (shrinkage operator), 因为它能够使变量  $x$ 、向量  $\mathbf{x}$  和矩阵  $\mathbf{X}$  的元素向零移动, 从而收缩元素的取值范围。因此, 软阈值化算子有时也写作 [34, 54]

$$S_\tau[x] = (|x| - \tau)_+ \text{sgn}(x) = (1 - \tau/|x|)_+ x \quad (4.6.13)$$

表 4.6.1 列出了一些典型函数的逼近映射 [122]。

表 4.6.1 一些典型函数的逼近映射

编号	函 数	逼近映射
1	凸函数 $h(\mathbf{x}) = 0$	$\text{prox}_h(\mathbf{x}) = \mathbf{x}$
2	位移函数 $h(\mathbf{x}) = \phi(\mathbf{x} - \mathbf{z})$	$\text{prox}_h(\mathbf{x}) = \mathbf{z} + \text{prox}_{\phi}(\mathbf{x} - \mathbf{z})$
3	比例函数 $h(\mathbf{x}) = \phi(\mathbf{x}/\rho)$ , $\rho \neq 0$	$\text{prox}_h(\mathbf{x}) = \rho \text{prox}_{\phi/\rho^2}(\mathbf{x}/\rho)$
4	反身函数 $h(\mathbf{x}) = \phi(-\mathbf{x})$	$\text{prox}_h(\mathbf{x}) = -\text{prox}_{\phi}(-\mathbf{x})$
5	共轭函数 $h(\mathbf{x}) = \phi^*(\mathbf{x})$	$\text{prox}_h(\mathbf{x}) = \mathbf{x} - \text{prox}_{\phi}(\mathbf{x})$
6	指示函数 $h(\mathbf{x}) = I_C(\mathbf{x})$	$\text{prox}_h(\mathbf{x}) = \mathcal{P}_C(\mathbf{x}) = \arg \min_{\mathbf{u} \in C} \ \mathbf{u} - \mathbf{x}\ _2^2$
7	支撑函数 $h(\mathbf{x}) = S_C(\mathbf{x})$	$\text{prox}_h(\mathbf{x}) = \mathbf{x} - \mathcal{P}_C(\mathbf{x})$
8	二次扰动函数 $h(\mathbf{x}) = \phi(\mathbf{x}) + \frac{\alpha}{2} \ \mathbf{x}\ ^2 + \mathbf{u}^T \mathbf{x} + \gamma$	$\text{prox}_h(\mathbf{x}) = \text{prox}_{\phi/(\alpha+1)}((\mathbf{x} - \mathbf{u})/(\alpha+1))$
9	凸函数 $h(\mathbf{x}) = \tau \ \mathbf{x}\ _1$ (其中 $\tau \geq 0$ )	$\text{prox}_h(\mathbf{x})_i = S_\tau[\mathbf{x}]_i = \begin{cases} \mathbf{x}_i - \tau, & \mathbf{x}_i > \tau \\ 0, &  \mathbf{x}_i  \leq \tau \\ \mathbf{x}_i + \tau, & \mathbf{x}_i < -\tau \end{cases}$
10	二次函数 $h(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c$	$\text{prox}_{\mu h}(\mathbf{x}) = (\mathbf{A} + \mu \mathbf{I})^{-1}(\mathbf{x} - \mu \mathbf{b})$
11	Euclidean 范数 $h(\mathbf{x}) = \ \mathbf{x}\ _2$	$\text{prox}_{\mu h}(\mathbf{x}) = \begin{cases} (1 - \mu/\ \mathbf{x}\ _2) \mathbf{x}, & \ \mathbf{x}\ _2 \geq \mu \\ 0, & \text{其他} \end{cases}$
12	对数障碍函数 $h(\mathbf{x}) = -\sum_{i=1}^n \log x_i$	$\text{prox}_{\mu h}(\mathbf{x})_i = \frac{x_i}{2} \sqrt{x_i^2 + 4\mu}, \quad i = 1, \dots, n$

无约束最小化问题式 (4.6.1) 的逼近梯度算法 (proximal gradient algorithm) 为

$$\mathbf{x}^{(k)} = \text{prox}_{\mu_k g} \left( \mathbf{x}^{(k-1)} - \mu_k \nabla f(\mathbf{x}^{(k-1)}) \right) \quad (4.6.14)$$

其中  $\mu_k$  是步长, 取常数或者由直线搜索确定。

下面是无约束最小化问题式 (4.6.1) 的几个典型例子。

### 1. 梯度法

若  $g(\mathbf{x}) = 0$ , 则最小化问题式 (4.6.1) 简化为无约束最小化  $\min f(\mathbf{x})$ 。由于  $\text{prox}_g(\mathbf{x}) = \mathbf{x}$ , 故逼近梯度算法式 (4.6.14) 退化为普通的梯度算法

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} - \mu_k \nabla f(\mathbf{x}^{(k-1)})$$

因此, 梯度算法是当凸函数  $g(\mathbf{x}) = 0$  时逼近梯度算法的一个特例; 而逼近梯度算法则是梯度算法的一种推广。

### 2. 梯度投影法 (gradient projection method)

对于指示函数  $g(\mathbf{x}) = I_C(\mathbf{x})$ , 最小化问题式 (4.6.1) 变为无约束最小化  $\min_{\mathbf{x} \in C} f(\mathbf{x})$ 。由于  $\text{prox}_g(\mathbf{x}) = \mathcal{P}_C(\mathbf{x})$ , 故逼近梯度法为

$$\mathbf{x}^{(k)} = \mathcal{P}_C \left( \mathbf{x}^{(k-1)} - \mu_k \nabla f(\mathbf{x}^{(k-1)}) \right) \quad (4.6.15)$$

$$= \arg \min_{\mathbf{u} \in C} \left\| \mathbf{u} - \mathbf{x}^{(k-1)} + \mu_k \nabla f(\mathbf{x}^{(k-1)}) \right\|_2^2 \quad (4.6.16)$$

这一算法称为梯度投影法。

### 3. 迭代软阈值化法 (iterative soft-thresholding method)

当  $g(\mathbf{x}) = \|\mathbf{x}\|_1$  时, 最小化问题式 (4.6.1) 变为无约束最小化  $\min f(\mathbf{x}) + \|\mathbf{x}\|_1$ 。此时, 迫近梯度算法

$$\mathbf{x}^{(k)} = \text{prox}_{\mu_k g} \left( \mathbf{x}^{(k-1)} - \mu_k \nabla f(\mathbf{x}^{(k-1)}) \right) \quad (4.6.17)$$

称为迭代软阈值化法, 其中

$$\text{prox}_{\mu g}(\mathbf{u})_i = \begin{cases} u_i - \mu, & u_i > \mu \\ 0, & -\mu \leq u_i \leq \mu \\ u_i + \mu, & u_i < -\mu \end{cases}$$

### 4. 奇异值阈值化法

若  $g(\mathbf{X}) = \|\mathbf{X}\|_* = \sum_{i=1}^{\min\{m,n\}} \sigma_i(\mathbf{X})$ , 则最小化问题式 (4.6.1) 变为无约束最小化  $\min \| \mathbf{X} \|_* + f(\mathbf{X})$ 。与之对应的迫近梯度法为

$$\mathbf{X}^{(k)} = \text{prox}_{\mu_k} \left( \mathbf{X}^{(k-1)} - \mu_k \nabla f(\mathbf{X}^{(k-1)}) \right) \quad (4.6.18)$$

若  $\mathbf{W} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T$ , 则

$$\text{prox}_\mu(\mathbf{W}) = \mathbf{U} \mathcal{D}_\mu(\boldsymbol{\Sigma}) \mathbf{V}^T \quad (4.6.19)$$

其中

$$[\mathcal{D}_\mu(\boldsymbol{\Sigma})]_i = \begin{cases} \sigma_i(\mathbf{X}) - \mu, & \text{若 } \sigma_i(\mathbf{X}) > \mu \\ 0, & \text{其他} \end{cases} \quad (4.6.20)$$

称为奇异值阈值化 (运算)。

下面是 Beck 与 Teboulle<sup>[34]</sup> 针对 Nesterov 最优梯度法提出的迫近梯度算法——快速迭代收缩-阈值化算法 (fast iterative shrinkage-thresholding algorithm, FISTA)。

#### 算法 4.6.1 具有固定步长的 FISTA 算法<sup>[34]</sup>

输入  $\nabla f(\mathbf{x})$  的 Lipschitz 常数  $L = L(f)$ 。

初始化  $\mathbf{y}_1 = \mathbf{x}_0 \in \mathbb{R}^n$ ,  $t_1 = 1$ 。

第  $k$  步迭代计算

$$\begin{aligned} \mathbf{x}_k &= \arg \min_{\mathbf{x}} \left\{ g(\mathbf{x}) + \frac{L}{2} \left| \mathbf{x} - \left( \mathbf{y}_k - \frac{1}{L} \nabla f(\mathbf{y}_k) \right) \right|^2 \right\} \\ t_{k+1} &= \frac{1 + \sqrt{1 + 4t_k^2}}{2} \\ \mathbf{y}_{k+1} &= \mathbf{x}_k + \left( \frac{t_k - 1}{t_{k+1}} \right) (\mathbf{x}_k - \mathbf{x}_{k-1}) \end{aligned}$$

**定理 4.6.1**<sup>[34]</sup> 令  $\{\mathbf{x}_k\}, \{\mathbf{y}_k\}$  是由 FISTA 算法产生的序列, 则对于任何迭代次数  $k \geq 1$ , 有下列结果

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{2L(f)\|\mathbf{x} - \mathbf{x}^*\|_2^2}{(k+1)^2}, \quad \forall \mathbf{x}^* \in X_*$$

式中  $\mathbf{x}^*$  和  $X_*$  分别表示  $\min F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x})$  的最优解点和最优解点集。

定理 4.6.1 表明, 若要求 FISTA 算法给出  $\varepsilon$  最优解  $F(\bar{\mathbf{x}}) - F(\mathbf{x}^*) \leq \varepsilon$ , 则 FISTA 最多需要  $\lceil C/\sqrt{\varepsilon} - 1 \rceil$  次迭代, 其中  $C = \sqrt{2L(f)\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}$ 。

## 4.7 约束优化算法

前面几节讨论了无约束优化的主要算法。在实际应用中, 常常遇到约束优化问题。

求解约束优化问题的标准方法是将约束优化问题转化为无约束优化问题。转化的方法主要有三种: ① Lagrangian 乘子法; ② 罚函数法; ③ Lagrangian 乘子法与罚函数法的结合 (增广 Lagrangian 乘子法)。

### 4.7.1 Lagrangian 乘子法与对偶上升法

考虑一等式约束的凸优化问题

$$\min f(\mathbf{x}) \quad \text{subject to } \mathbf{A}\mathbf{x} = \mathbf{b} \quad (4.7.1)$$

式中,  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , 并且目标函数  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  是凸函数。

Lagrangian 乘子法将式 (4.7.1) 变成无约束最小化问题, 其 Lagrangian 目标函数为

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) \quad (4.7.2)$$

原始优化问题式 (4.7.1) 的对偶目标函数为

$$g(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) = -f^*(-\mathbf{A}^T \boldsymbol{\lambda}) - \mathbf{b}^T \boldsymbol{\lambda} \quad (4.7.3)$$

其中  $\boldsymbol{\lambda}$  为对偶变量或 Lagrangian 乘子向量,  $f^*$  是  $f$  的凸共轭函数。

借助 Lagrangian 乘子法, 原始等式约束极小化问题式 (4.7.1) 变为对偶极大化问题

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}^m} g(\boldsymbol{\lambda}) = -f^*(-\mathbf{A}^T \boldsymbol{\lambda}) - \mathbf{b}^T \boldsymbol{\lambda} \quad (4.7.4)$$

假定强对偶性满足, 则原始问题和对偶问题的最优解相同。此时, 原始极小化问题式 (4.7.1) 的最优解点  $\mathbf{x}^*$  即可由下式恢复

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}^*) \quad (4.7.5)$$

在对偶上升法 (dual ascent method) 中, 利用梯度上升法求解极大化问题式 (4.7.4)。对偶上升法由两个步骤组成

$$\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}_k) \quad (4.7.6)$$

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \mu_k (\mathbf{A}\mathbf{x}_{k+1} - \mathbf{b}) \quad (4.7.7)$$

其中, 式 (4.7.6) 为原始变量  $\boldsymbol{x}$  极小化步骤, 式 (4.7.7) 则是对偶变量  $\boldsymbol{\lambda}$  更新步骤, 其步长为  $\mu_k$ 。

由于对偶变量  $\boldsymbol{\lambda} \succeq 0$  可解释为一价格向量, 所以对偶变量的更新也叫价格上升 (price ascent) 或价格调整 (price adjustment) 步骤。价格上升的目的就是使收益函数  $g(\boldsymbol{\lambda}^k)$  趋于最大化。

对偶上升法包含有两层含义: ① 对偶变量  $\boldsymbol{\lambda}$  的更新采用梯度上升法。② 通过步长  $\mu^k$  的适当选择, 保证对偶目标函数的上升, 即  $g(\boldsymbol{\lambda}_{k+1}) > g(\boldsymbol{\lambda}_k)$ 。

## 4.7.2 罚函数法

罚函数法是一种被广泛采用的约束优化方法, 其基本原理是: 通过罚函数与/或障碍函数, 将约束优化问题变成一反映原目标函数和约束条件的合成函数的无约束极小化。

考虑约束优化问题

$$\min f_0(\boldsymbol{x}) \quad \text{subject to } f_i(\boldsymbol{x}) \geq 0, i = 1, \dots, m; \quad h_j(\boldsymbol{x}) = 0, j = 1, \dots, q \quad (4.7.8)$$

式中  $\boldsymbol{x} \in \mathcal{F} \subseteq \mathcal{S} \subseteq \mathbb{R}^n$ , 且  $\mathcal{F}$  表示变元向量  $\boldsymbol{x}$  的可行集,  $\mathcal{S}$  代表整个搜索空间。

将约束优化问题变为无约束优化问题有两种惩罚方式<sup>[526]</sup>。第 1 种方式使用加性罚函数项

$$L(\boldsymbol{x}) = \begin{cases} f_0(\boldsymbol{x}), & \boldsymbol{x} \in \mathcal{F} \\ f_0(\boldsymbol{x}) + p(\boldsymbol{x}), & \text{其他} \end{cases} \quad (4.7.9)$$

其中  $p(\boldsymbol{x})$  称为罚函数 (penalty function)。如果变元  $\boldsymbol{x}$  没有违背可行集  $\mathcal{F}$  的约束, 罚函数  $p(\boldsymbol{x}) = 0$ , 否则  $p(\boldsymbol{x}) > 0$ 。

第 2 种方式使用乘性罚函数项

$$L(\boldsymbol{x}) = \begin{cases} f_0(\boldsymbol{x}), & \boldsymbol{x} \in \mathcal{F} \\ f_0(\boldsymbol{x})p(\boldsymbol{x}), & \text{其他} \end{cases} \quad (4.7.10)$$

其中  $p(\boldsymbol{x}) = 1$ , 若  $\boldsymbol{x}$  没有脱离可行集  $\mathcal{F}$ ; 否则  $p(\boldsymbol{x}) > 1$ 。

通常多采用加性惩罚方式。

罚函数有时被简称为惩罚。罚函数的主要性质是: 若  $p_1(\boldsymbol{x})$  是对闭集  $\mathcal{F}_1$  的惩罚,  $p_2(\boldsymbol{x})$  是对闭集  $\mathcal{F}_2$  的惩罚, 则  $p_1(\boldsymbol{x}) + p_2(\boldsymbol{x})$  是对交集  $\mathcal{F}_1 \cap \mathcal{F}_2$  的惩罚。

令

$$\mathcal{F} = \{\boldsymbol{x} \in \mathbb{R}^n | f_i(\boldsymbol{x}) \geq 0, i = 1, \dots, m\}$$

则下列函数是闭集  $\mathcal{F}$  上的罚函数:

(1) 二次罚函数

$$p(\boldsymbol{x}) = \sum_{i=1}^m (\max\{0, -f_i(\boldsymbol{x})\})^2 \quad (4.7.11)$$

(2) 非平滑罚函数

$$p(\boldsymbol{x}) = \sum_{i=1}^M \max\{0, -f_i(\boldsymbol{x})\} \quad (4.7.12)$$

罚函数法将原始约束优化问题转换成无约束优化问题

$$\min_{\mathbf{x} \in \mathcal{S}} L_\rho(\mathbf{x}) = f_0(\mathbf{x}) + \rho \cdot p(\mathbf{x}) \quad (4.7.13)$$

式中, 系数  $\rho$  为惩罚参数, 通过对罚函数  $p(\mathbf{x})$  的加权, 体现惩罚的力度。转换后的优化问题式 (4.7.13) 常称为原约束优化问题的辅助优化问题。下面分三种情况加以讨论。

### 1. 等式约束优化的罚函数法

先考虑等式约束优化问题

$$\min_{\mathbf{x}} f_0(\mathbf{x}) \quad \text{subject to} \quad h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q \quad (4.7.14)$$

定义函数

$$p(\mathbf{x}) = \sum_{i=1}^q |h_i(\mathbf{x})|^2 \quad (4.7.15)$$

显然, 这一函数具有以下性质

$$p(\mathbf{x}) \begin{cases} = 0, & h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q \\ > 0, & h_i(\mathbf{x}) \neq 0, \quad i = 1, \dots, q \end{cases} \quad (4.7.16)$$

这表明,  $p(\mathbf{x})$  是等式约束优化问题式 (4.7.14) 的罚函数, 因为它对满足等式约束条件的点  $\mathbf{x}$  无任何影响, 而对违反等式约束条件的点则予以惩罚。

### 2. 不等式约束优化的罚函数法

再考虑只有不等式约束的优化问题

$$\min_{\mathbf{x} \in \mathbb{R}^n} f_0(\mathbf{x}) \quad \text{subject to} \quad f_i(\mathbf{x}) \geq 0, \quad i = 1, \dots, m \quad (4.7.17)$$

不等式约束优化问题式 (4.7.17) 的罚函数分为以下两大类:

#### (1) 外罚函数

罚函数取

$$p(\mathbf{x}) = \sum_{i=1}^m (\max\{0, -f_i(\mathbf{x})\})^r \quad (4.7.18)$$

式中  $r$  常取 1 或 2。这种函数对违背不等式约束的点即可行集外部的所有点进行处罚, 称为外罚函数 (exterior penalty function)。

#### (2) 内罚函数

罚函数取

$$p(\mathbf{x}) = \sum_{i=1}^m \frac{1}{f_i(\mathbf{x})} \quad \text{或} \quad p(\mathbf{x}) = \sum_{i=1}^m \frac{1}{f_i(\mathbf{x})} \log(f_i(\mathbf{x})) \quad (4.7.19)$$

这种罚函数相当于在可行集边界  $bnd(\mathcal{F})$  上树立起一道围墙, 对于企图从可行内集  $\text{int}(\mathcal{F})$  穿越到可行集边界  $bnd(\mathcal{F})$  的点  $\mathbf{x}$  进行阻挡, 故称为内罚函数 (interior penalty function), 也称障碍函数 (barrier function)。

一连续函数  $\phi(\mathbf{x})$  称为具有非空内集的闭集  $\mathcal{F} = \text{int}(\mathcal{F})$  上的障碍函数, 若

$$\phi(\mathbf{x}) \begin{cases} = 0, & \mathbf{x} \in \text{int}(\mathcal{F}) \\ \rightarrow \infty, & \mathbf{x} \rightarrow \text{bnd}(\mathcal{F}) \end{cases} \quad (4.7.20)$$

障碍函数有时简称为障碍。

与罚函数类似, 障碍函数的主要性质是: 若  $\phi_1(\mathbf{x})$  是对闭集  $\mathcal{F}_1$  的障碍,  $\phi_2(\mathbf{x})$  是对闭集  $\mathcal{F}_2$  的障碍, 则  $\phi_1(\mathbf{x}) + \phi_2(\mathbf{x})$  是对交集  $\mathcal{F}_1 \cap \mathcal{F}_2$  的障碍。

令

$$\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \geq 0; i = 1, \dots, m\} \quad (4.7.21)$$

$$\text{strict}(\mathcal{F}) = \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) > 0; i = 1, \dots, m\} \quad (4.7.22)$$

分别代表不等式约束函数的可行区和严格可行区。

下面是闭集  $\mathcal{F}$  上的几种典型障碍函数 [363]:

(1) 幂函数障碍函数 (power-function barrier function)  $\phi(\mathbf{x}) = \sum_{i=1}^m \frac{1}{(f_i(\mathbf{x}))^p}, p \geq 1$ 。

(2) 对数障碍函数 (logarithmic barrier function)  $\phi(\mathbf{x}) = \frac{1}{f_i(\mathbf{x})} \sum_{i=1}^m \log(f_i(\mathbf{x}))$ 。

(3) 指数障碍函数 (exponential barrier function)  $\phi(\mathbf{x}) = \sum_{i=1}^m \exp\left(\frac{1}{f_i(\mathbf{x})}\right)$ 。

特别地,  $p = 1$  的幂函数障碍函数  $\phi(\mathbf{x}) = \sum_{i=1}^m \frac{1}{f_i(\mathbf{x})}$  称为逆障碍函数, 它是 Carroll 于 1961 年提出的 [91]; 而

$$\phi(\mathbf{x}) = \mu \sum_{i=1}^m \frac{1}{\log(f_i(\mathbf{x}))} \quad (4.7.23)$$

称为经典 Fiacco-McCormick 对数障碍函数, 是 Fiacco 与 McCormick 于 1968 年在他们著名的开创性著作中提出的 [164]。其中,  $\mu$  为障碍参数。

采用外罚函数和内罚函数的优化方法分别称为外罚函数法和内罚函数法, 它们之间的比较如下:

(1) 外罚函数法常称罚函数法; 内罚函数法习惯称障碍函数法, 简称障碍法。罚函数  $\frac{1}{f_i(\mathbf{x})}$  和  $\frac{1}{f_i(\mathbf{x})} \log f_i(\mathbf{x})$  分别称为逆障碍函数和对数障碍函数。

(2) 外罚函数法对可行集以外的所有点进行惩罚, 求出的解满足全部不等式约束条件  $f_i(\mathbf{x}) \leq 0, i = 1, \dots, m$ , 是不等式约束优化问题的精确解, 因而是一种最优设计方案; 而内罚函数法或障碍法阻挡了可行集边界的点, 得到的解只满足严格不等式  $f_i(\mathbf{x}) > 0, i = 1, \dots, m$ , 是原始优化问题的近似解, 因而是一种次优设计方案。

(3) 外罚函数法可以用不可行点启动, 通常收敛慢; 而内罚函数法要求初始点是可行内点, 其选择比较困难, 但却具有很好的收敛和逼近性能。

在进化计算中，通常采用外罚函数法，因为内罚函数法要求的可行初始点搜索往往是 NP 难题。工程设计人员尤其是过程控制人员偏爱使用内罚函数法，因为这种方法可以使设计者观察到“优化过程中在可行集内的设计点所对应的目标函数值的变化情况”，而这是外罚函数法无法提供的。

### 3. 等式和不等式约束优化的混合罚函数法

对于标准形式的不等式约束优化问题式 (4.7.8)，可行集定义为满足所有不等式和等式约束的点集

$$\mathcal{F} = \{\mathbf{x} | f_i(\mathbf{x}) \leq 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, q\} \quad (4.7.24)$$

既满足严格不等式约束  $f_i(\mathbf{x}) < 0$ ，又同时满足等式约束  $h_i(\mathbf{x}) = 0$  的点集

$$\text{relint}(\mathcal{F}) = \{\mathbf{x} | f_i(\mathbf{x}) < 0, i = 1, \dots, m; h_i(\mathbf{x}) = 0, i = 1, \dots, q\} \quad (4.7.25)$$

称为相对可行内点集 (relative feasible interior set) 或相对严格可行集 (relative strictly feasible set)。相对可行内点集的点称为相对内点。可行集与相对可行内点集的差集  $\mathcal{F} \setminus \text{relint}(\mathcal{F})$  称为相对可行集边界 (relative boundary of the feasible set)。

综合等式约束优化问题和只有不等式约束优化问题的罚函数法，很容易得到同时有等式和不等式约束的优化问题式 (4.7.8) 的混合罚函数法：

#### (1) 混合外罚函数法

$$\min_{\mathbf{x}} f_0(\mathbf{x}) + \rho_1 \sum_{i=1}^m (\max\{0, f_i(\mathbf{x})\})^2 + \rho_2 \sum_{i=1}^q |h_i(\mathbf{x})|^2 \quad (4.7.26)$$

#### (2) 混合内罚函数法

$$\min_{\mathbf{x}} f_0(\mathbf{x}) + \rho_1 \sum_{i=1}^m \frac{1}{-f_i(\mathbf{x})} \log(-f_i(\mathbf{x})) + \rho_2 \sum_{i=1}^q |h_i(\mathbf{x})|^2 \quad (4.7.27)$$

从以上定义知，混合外罚函数法惩罚的是可行集  $\mathcal{F}$  以外的点即不可行集里的点，而混合内部罚函数法惩罚的则是相对可行内集以外的点。因此，混合外罚函数法可以用不可行点启动，而混合内罚函数通常则需要用相对内点启动。

在罚函数的严格分类的意义上，上述各种罚函数属“死亡”惩罚 (death penalty)，即通过惩罚函数  $p(\mathbf{x}) = +\infty$  完全排除非可行解点  $\mathbf{x} \in S \setminus \mathcal{F}$  (搜索空间  $S$  与可行集的差集)<sup>[526]</sup>。如果可行搜索空间是凸的或者是整个搜索空间的合理部分，这种罚函数法可以工作得很好<sup>[342]</sup>。然而，对于遗传算法和进化计算，很多问题的可行集和不可行集的边界是未知的，因此很难确定可行集的精确位置。在这些情况下，常采用其他的罚函数<sup>[526]</sup>：静态惩罚 (static penalties)、动态惩罚 (dynamic penalties)、退火惩罚 (annealing penalties)、自适应惩罚 (adaptive penalties) 和协同进化惩罚 (co-evolutionary penalties)。

### 4.7.3 增广 Lagrangian 乘子法

前面分别介绍了约束优化的 Lagrangian 乘子法和罚函数法，它们的主要不足如下。

Lagrangian 乘子法的主要缺点是<sup>[297, 44]</sup>：①只有当约束优化问题具有局部凸结构时，对偶的无约束优化问题才是良好定义的，并且 Lagrangian 乘子的更新  $\lambda_{k+1} = \lambda_k + \alpha_k h(\mathbf{x}_k)$  才有意义。② Lagrangian 目标函数的收敛比较慢，因为 Lagrangian 乘子的更新是一种上升迭代 (ascent iteration)，只能适度地快速收敛。

罚函数法的不足是<sup>[44]</sup>：收敛慢，大的惩罚参数容易引起转化后的无约束优化问题的病态，从而造成算法的数值不稳定性。

减缓这两种方法缺点的一种简单而有效的途径是将两种方法结合起来。

下面分等式约束和不等式约束两种情况加以讨论。

#### 1. 等式约束优化的增广 Lagrangian 乘子法

考虑等式约束最小化问题式 (4.7.14)。记  $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \dots, h_q(\mathbf{x})]^T$ 。

对 Lagrangian 目标函数  $L(\mathbf{x}, \boldsymbol{\lambda})$  加惩罚函数，组建 Lagrangian 乘子法与罚函数法相结合的目标函数  $L : \mathbb{R}^n \times \mathbb{R}^q \times \mathbb{R} \rightarrow (-\infty, +\infty]$

$$L_\rho(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) + \rho \phi(\mathbf{h}(\mathbf{x})) = f_0(\mathbf{x}) + \sum_{i=1}^q y_i h_i(\mathbf{x}) + \rho \sum_{i=1}^q \phi(h_i(\mathbf{x})) \quad (4.7.28)$$

式中  $\rho$  为惩罚参数。

这种将罚函数与 Lagrangian 函数相结合，构造出更合适的目标函数的方法称为增广 Lagrangian 乘子法 (augmented Lagrangian multiplier method)，简称增广乘子法，或称广义乘子法。增广 Lagrangian 乘子法是 Hestenes<sup>[229]</sup> 和 Powell<sup>[414]</sup> 于 1960 年代后期讨论和分析的。

由式 (4.7.28) 容易看出：

- (1) 若惩罚因子  $\rho = 0$ ，则增广 Lagrangian 乘子法退化为标准的 Lagrangian 乘子法。
- (2) 若 Lagrangian 乘子向量  $\boldsymbol{\lambda} = \mathbf{0}$ ，则增广 Lagrangian 乘子法退化为标准罚函数法。

求解无约束优化问题  $\min L_\rho(\mathbf{x}, \boldsymbol{\lambda})$  的对偶上升法由以下两个更新组成<sup>[44]</sup>

$$\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} L_\rho(\mathbf{x}, \boldsymbol{\lambda}_k) \quad (4.7.29)$$

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho_k \nabla_{\boldsymbol{\lambda}} L_\rho(\mathbf{x}_{k+1}, \boldsymbol{\lambda}_k) \quad (4.7.30)$$

式中  $\nabla_{\boldsymbol{\lambda}} L_\rho(\mathbf{x}, \boldsymbol{\lambda})$  是增广 Lagrangian 函数关于对偶向量  $\boldsymbol{\lambda}$  的梯度向量。

特别地，若等式约束为仿射函数  $\mathbf{h}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}$ ，且罚函数取  $\phi(\mathbf{h}(\mathbf{x})) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ ，则增广 Lagrangian 函数

$$L_\rho(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \boldsymbol{\lambda}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) + \frac{\rho}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \quad (4.7.31)$$

相应的对偶上升法的更新公式为

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \arg \min_{\boldsymbol{x}} L_{\rho}(\boldsymbol{x}, \boldsymbol{\lambda}_k) \\ \boldsymbol{\lambda}_{k+1} &= \boldsymbol{\lambda}_k + \rho_k (\boldsymbol{A}\boldsymbol{x}_{k+1} - \boldsymbol{b})\end{aligned}$$

关于增广 Lagrangian 乘子法，通常作以下假设<sup>[44]</sup>：

(1) 优化问题  $\min L_{\rho}(\boldsymbol{x}, \boldsymbol{\lambda})$  存在一局部极小点  $\bar{\boldsymbol{x}}$ ，它是可行集  $\mathcal{F}$  的内点，并且满足孤立局部极小点的二阶充分条件

①  $f_0(\boldsymbol{x})$  和  $h_i(\boldsymbol{x})$  在  $\bar{\boldsymbol{x}}$  的邻域是二次可微分的；

② 梯度  $\nabla h_i(\bar{\boldsymbol{x}}), i = 1, \dots, q$  是线性无关的；

③ 存在一个对偶向量  $\bar{\boldsymbol{\lambda}}$  满足条件  $\nabla L_0(\bar{\boldsymbol{x}}, \bar{\boldsymbol{\lambda}}) = 0$  和  $\boldsymbol{z}^T \nabla^2 L_{\rho}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{\lambda}}) \boldsymbol{z} > 0, \forall \boldsymbol{z} \neq \mathbf{0} \in \mathbb{R}^n; (\nabla L_0(\bar{\boldsymbol{x}}, \bar{\boldsymbol{\lambda}}))^T \boldsymbol{z} = 0, i = 1, \dots, q$ 。其中， $L_0(\boldsymbol{x}, \boldsymbol{\lambda}) = L_{\rho}(\boldsymbol{x}, \boldsymbol{\lambda})|_{\rho=0} = f_0(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x})$ 。

(2) 罚函数  $\phi: \mathbb{R} \rightarrow [0, +\infty]$  在包括零在内的一个开区间里是二次可微分的，并且其在零点的二阶导数  $\phi''(0) = 1$ 。

增广 Lagrangian 乘子法的极小化点与 Lagrangian 乘子向量  $\boldsymbol{\lambda}$ 、惩罚参数  $\rho$  有关，记为  $\boldsymbol{x}(\boldsymbol{\lambda}, \rho)$ 。令  $\bar{\boldsymbol{\lambda}}$  是使增广 Lagrangian 目标函数  $L_{\rho}(\boldsymbol{x}, \boldsymbol{\lambda})$  最小化的解点  $\bar{\boldsymbol{x}}$  所对应的增广 Lagrangian 乘子向量，Bertsekas 证明了<sup>[44]</sup>：在假设条件(1)、(2)及其他相对温和的假设下，由式(4.7.30)更新的增广 Lagrangian 乘子向量序列  $\{\boldsymbol{\lambda}^k\}$  具有以下收敛速率：

① 若  $\rho_k \rightarrow \bar{\rho} < \infty, \boldsymbol{\lambda}^k \neq \bar{\boldsymbol{\lambda}}, \forall k$ ，则

$$\limsup_{k \rightarrow \infty} \frac{\|\boldsymbol{\lambda}_{k+1} - \bar{\boldsymbol{\lambda}}\|_2}{\|\boldsymbol{\lambda}_k - \bar{\boldsymbol{\lambda}}\|_2} \leq \frac{M}{\bar{\rho}} \quad (\text{线性收敛}) \quad (4.7.32)$$

式中  $M > 0$  为标量。

② 若  $\rho_k \rightarrow \infty, \boldsymbol{\lambda}_k \neq \bar{\boldsymbol{\lambda}}, \forall k$ ，则

$$\lim_{k \rightarrow \infty} \frac{\|\boldsymbol{\lambda}_{k+1} - \bar{\boldsymbol{\lambda}}\|_2}{\|\boldsymbol{\lambda}_k - \bar{\boldsymbol{\lambda}}\|_2} = 0 \quad (\text{超线性收敛}) \quad (4.7.33)$$

上述分析表明，增广 Lagrangian 乘子法具有如下优点：

(1) 无须将惩罚因子  $\rho_k$  增加至无穷大，只需要使用式(4.7.30)更新 Lagrangian 乘子向量，增广乘子法即可收敛。因此，罚函数法的病态条件在增广乘子法中不复存在。

(2) 迭代式(4.7.30)产生的增广 Lagrangian 乘子向量序列在相对温和的假设下快速收敛，比普通的 Lagrangian 乘子法的收敛快得多。

(3) 不再像标准 Lagrangian 乘子法那样要求目标函数  $f_0(\boldsymbol{x})$  具有局部凸结构。换言之，增广 Lagrangian 乘子法的适用范围更为广泛。

虽然增广 Lagrangian 乘子法与标准 Lagrangian 乘子法的对偶上升法取相同的形式，但它们之间存在以下主要区别：

(1) 标准 Lagrangian 乘子法的  $\mathbf{x}$  更新是标准 Lagrangian 目标函数  $L(\mathbf{x}, \boldsymbol{\lambda}_k)$  的极小化结果, 而增广 Lagrangian 乘子法的  $\mathbf{x}$  更新则是 Lagrangian 目标函数与惩罚函数之和  $L_\rho(\mathbf{x}, \boldsymbol{\lambda}_k)$  的极小化结果。

(2) 标准 Lagrangian 乘子法的 Lagrangian 乘子向量  $\boldsymbol{\lambda}$  更新中的参数  $\mu_k$  为步长, 而增广 Lagrangian 乘子法的  $\boldsymbol{\lambda}$  更新中的参数  $\rho_k$  为惩罚参数。

## 2. 混合约束优化的增广 Lagrangian 乘子法

考虑不等式约束和等式约束同时存在的混合约束优化问题

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to } \mathbf{Ax} = \mathbf{b}, \mathbf{Bx} \leq \mathbf{h} \quad (4.7.34)$$

令非负向量  $\mathbf{s} \geq \mathbf{0}$  为松弛变量 (slack variables), 使得  $\mathbf{Bx} + \mathbf{s} = \mathbf{h}$ 。于是, 混合约束中的不等式约束变成了等式约束。若取惩罚函数  $\phi(\mathbf{g}(\mathbf{x})) = \frac{1}{2} \|\mathbf{g}(\mathbf{x})\|_2^2$ , 则增广 Lagrangian 目标函数

$$\begin{aligned} L_\rho(\mathbf{x}, \mathbf{s}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = & f(\mathbf{x}) + \boldsymbol{\lambda}^T (\mathbf{Ax} - \mathbf{b}) + \boldsymbol{\nu}^T (\mathbf{Bx} + \mathbf{s} - \mathbf{h}) \\ & + \frac{\rho}{2} (\|\mathbf{Ax} - \mathbf{b}\|_2^2 + \|\mathbf{Bx} + \mathbf{s} - \mathbf{h}\|_2^2) \end{aligned} \quad (4.7.35)$$

式中, 两个 Lagrangian 乘子向量  $\boldsymbol{\lambda} \geq \mathbf{0}$  和  $\boldsymbol{\nu} \geq \mathbf{0}$ , 并且惩罚参数  $\rho > 0$ 。

这样一来, 等式约束优化问题的对偶上升法即可推广应用于式 (4.7.35), 从而得到混合约束优化问题的对偶上升法

$$\mathbf{x}_{k+1} = \arg \min_{\mathbf{x}} L_\rho(\mathbf{x}, \mathbf{s}_k, \boldsymbol{\lambda}_k, \boldsymbol{\nu}_k) \quad (4.7.36)$$

$$\mathbf{s}_{k+1} = \arg \min_{\mathbf{s} \geq \mathbf{0}} L_\rho(\mathbf{x}_{k+1}, \mathbf{s}, \boldsymbol{\lambda}_k, \boldsymbol{\nu}_k) \quad (4.7.37)$$

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho_k (\mathbf{Ax}_{k+1} - \mathbf{b}) \quad (4.7.38)$$

$$\boldsymbol{\nu}_{k+1} = \boldsymbol{\nu}_k + \rho_k (\mathbf{Bx}_{k+1} + \mathbf{s}_{k+1} - \mathbf{h}) \quad (4.7.39)$$

其中, 式 (4.7.36) 和式 (4.7.37) 分别为原始变量  $\mathbf{x}$  和中间变量  $\mathbf{s}$  的更新, 式 (4.7.38) 和式 (4.7.39) 则分别是对应于等式约束  $\mathbf{Ax} = \mathbf{b}$  和不等式约束  $\mathbf{Bx} \leq \mathbf{h}$  的 Lagrangian 乘子向量  $\boldsymbol{\lambda}$  和  $\boldsymbol{\nu}$  的对偶更新。

## 4.7.4 交替方向乘子法

在应用统计学和机器学习中, 经常会遇到大尺度的等式约束优化问题, 其中  $\mathbf{x} \in \mathbb{R}^n$  的维数  $n$  很大。如果向量  $\mathbf{x}$  可以分解为几个子向量, 即  $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_r)$ , 并且目标函数也可分解为

$$f(\mathbf{x}) = \sum_{i=1}^r f_i(\mathbf{x}_i)$$

其中  $\mathbf{x}_i \in \mathbb{R}^{n_i}$ , 并且  $\sum_{i=1}^r n_i = n$ , 则大尺度的优化问题可转变为分布式优化 (distributed optimization) 问题。

交替方向乘子法 (alternating direction method of multipliers, ADMM) 是一种非常适合于分布式凸优化的简单而有效的方法。

ADMM 采用一种分解坐标法的方式, 将优化问题的求解变成较小的局部子问题的求解, 然后这些局部子问题的解以协同的方式, 用于恢复或重构大尺度优化问题的解。

ADMM 是 20 世纪 70 年代中期由 Gabay 和 Mercier<sup>[179]</sup>, Glowinski 和 Marrocco<sup>[183]</sup>独立提出的。

与目标函数  $f(\mathbf{x})$  的分解相对应, 等式约束的矩阵也分块为

$$\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_r], \quad \mathbf{Ax} = \sum_{i=1}^r \mathbf{A}_i \mathbf{x}_i$$

于是, 增广 Lagrangian 目标函数可写作<sup>[54]</sup>

$$L_\rho(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^r L_i(\mathbf{x}_i, \boldsymbol{\lambda}) = \sum_{i=1}^r \left( f_i(\mathbf{x}_i) + \boldsymbol{\lambda}^T \mathbf{A}_i \mathbf{x}_i \right) - \boldsymbol{\lambda}^T \mathbf{b} + \frac{\rho}{2} \left\| \sum_{i=1}^r (\mathbf{A}_i \mathbf{x}_i) - \mathbf{b} \right\|_2^2$$

对增广 Lagrangian 目标函数应用对偶上升法, 即可得到能够进行并行运算的分散算法 (decentralized algorithm)<sup>[54]</sup>

$$\mathbf{x}_i^{k+1} = \arg \min_{\mathbf{x}_i \in \mathbb{R}^{n_i}} L_i(\mathbf{x}_i, \boldsymbol{\lambda}_k), \quad i = 1, \dots, r \quad (4.7.40)$$

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho_k \left( \sum_{i=1}^r \mathbf{A}_i \mathbf{x}_i^{k+1} - \mathbf{b} \right) \quad (4.7.41)$$

其中,  $\mathbf{x}_i$  更新 ( $i = 1, \dots, r$ ) 可独立地并行运行。由于  $\mathbf{x}_i, i = 1, \dots, r$  以一种交替或序贯的方式进行更新, 故这种增广 Lagrangian 乘子法称为“交替方向”乘子法。

在实际中最简单而有用的目标函数分解

$$\min f(\mathbf{x}) + g(\mathbf{z}) \quad \text{subject to} \quad \mathbf{Ax} + \mathbf{Bz} = \mathbf{c} \quad (4.7.42)$$

式中  $\mathbf{x} \in \mathbb{R}^n, \mathbf{z} \in \mathbb{R}^m, \mathbf{A} \in \mathbb{R}^{p \times n}, \mathbf{B} \in \mathbb{R}^{p \times m}, \mathbf{c} \in \mathbb{R}^p$ 。

优化问题式 (4.7.42) 的增广 Lagrangian 目标函数为

$$L_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda}) = f(\mathbf{x}) + g(\mathbf{z}) + \boldsymbol{\lambda}^T (\mathbf{Ax} + \mathbf{Bz} - \mathbf{c}) + \frac{\rho}{2} \|\mathbf{Ax} + \mathbf{Bz} - \mathbf{c}\|_2^2 \quad (4.7.43)$$

由此易知, 其最优化条件分为原始可行性

$$\mathbf{Ax} + \mathbf{Bz} - \mathbf{c} = \mathbf{0} \quad (4.7.44)$$

和对偶可行性

$$\mathbf{0} \in \partial f(\mathbf{x}) + \mathbf{A}^T \mathbf{x} + \rho(\mathbf{Ax} + \mathbf{Bz} - \mathbf{c}) = \partial f(\mathbf{x}) + \mathbf{A}^T \boldsymbol{\lambda} \quad (4.7.45)$$

$$\mathbf{0} \in \partial g(\mathbf{z}) + \mathbf{B}^T \mathbf{z} + \rho(\mathbf{Ax} + \mathbf{Bz} - \mathbf{c}) = \partial g(\mathbf{z}) + \mathbf{B}^T \boldsymbol{\lambda} \quad (4.7.46)$$

式中  $\partial f(\mathbf{x})$  和  $\partial g(\mathbf{z})$  分别是子目标函数  $f(\mathbf{x})$  和  $g(\mathbf{z})$  的次微分。

优化问题  $\min L_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda})$  的交替方向乘子法的更新公式为

$$\mathbf{x}_{k+1} = \arg \min_{\mathbf{x} \in \mathbb{R}^n} L_\rho(\mathbf{x}, \mathbf{z}_k, \boldsymbol{\lambda}_k) \quad (4.7.47)$$

$$\mathbf{z}_{k+1} = \arg \min_{\mathbf{z} \in \mathbb{R}^m} L_\rho(\mathbf{x}_{k+1}, \mathbf{z}, \boldsymbol{\lambda}_k) \quad (4.7.48)$$

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho_k (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1} - \mathbf{c}) \quad (4.7.49)$$

原始可行性不可能严格满足，其误差

$$\mathbf{r}_k = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{z}_k - \mathbf{c} \quad (4.7.50)$$

称为第  $k$  次迭代的原始残差（向量）。于是，Lagrangian 乘子向量的更新可以简写为

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \rho_k \mathbf{r}_{k+1} \quad (4.7.51)$$

同样地，对偶可行性也不可能严格满足。由于  $\mathbf{x}_{k+1}$  是  $L_\rho(\mathbf{x}, \mathbf{z}_k, \boldsymbol{\lambda}_k)$  的极小化变量，故有

$$\begin{aligned} \mathbf{0} &\in \partial f(\mathbf{x}_{k+1}) + \mathbf{A}^\top \boldsymbol{\lambda}_k + \rho(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_k - \mathbf{c}) \\ &= \partial f(\mathbf{x}_{k+1}) + \mathbf{A}^\top [\boldsymbol{\lambda}_k + \rho \mathbf{r}_{k+1} + \rho \mathbf{B}(\mathbf{z}_k - \mathbf{z}_{k+1})] \\ &= \partial f(\mathbf{x}_{k+1}) + \mathbf{A}^\top \boldsymbol{\lambda}_{k+1} + \rho \mathbf{A}^\top \mathbf{B}(\mathbf{z}_k - \mathbf{z}_{k+1}) \end{aligned}$$

与对偶可行性公式 (4.7.45) 比较，易知

$$\mathbf{s}_{k+1} = \rho \mathbf{A}^\top \mathbf{B}(\mathbf{z}_k - \mathbf{z}_{k+1}) \quad (4.7.52)$$

为对偶可行性的误差，故称为第  $k+1$  次迭代的对偶残差（向量）。

交替方向乘子法的停止准则是第  $k+1$  次迭代的原始残差和对偶残差都应该非常小，即满足<sup>[54]</sup>

$$\|\mathbf{r}_{k+1}\|_2 \leq \varepsilon_{\text{pri}}, \quad \|\mathbf{s}_{k+1}\|_2 \leq \varepsilon_{\text{dual}} \quad (4.7.53)$$

式中  $\varepsilon_{\text{pri}}$  和  $\varepsilon_{\text{dual}}$  分别是原始可行性和对偶可行性的允许扰动。

若令  $\boldsymbol{\nu} = (1/\rho)\boldsymbol{\lambda}$  是经过比例  $1/\rho$  缩放的 Lagrangian 乘子向量（简称缩放对偶向量），则式 (4.7.47)~式 (4.7.49) 变为<sup>[54]</sup>

$$\mathbf{x}_{k+1} = \arg \min_{\mathbf{x} \in \mathbb{R}^n} (f(\mathbf{x}) + (\rho/2)\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}_k - \mathbf{c} + \boldsymbol{\nu}_k\|_2^2) \quad (4.7.54)$$

$$\mathbf{z}_{k+1} = \arg \min_{\mathbf{z} \in \mathbb{R}^m} (g(\mathbf{z}) + (\rho/2)\|\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z} - \mathbf{c} + \boldsymbol{\nu}_k\|_2^2) \quad (4.7.55)$$

$$\boldsymbol{\nu}_{k+1} = \boldsymbol{\nu}_k + \mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{z}_{k+1} - \mathbf{c} = \boldsymbol{\nu}_k + \mathbf{r}_{k+1} \quad (4.7.56)$$

缩放对偶向量具有有趣的解释<sup>[54]</sup>：由第  $k$  次迭代的残差  $\mathbf{r}_k = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{z}_k - \mathbf{c}$  易知

$$\boldsymbol{\nu}_k = \boldsymbol{\nu}^0 + \sum_{i=1}^k \mathbf{r}^i \quad (4.7.57)$$

也就是说，第  $k$  次迭代的缩放对偶向量是所有  $k$  次迭代的原始残差的运行之和。

式(4.7.54)~式(4.7.56)称为缩放形式的交替方向乘子法，而式(4.7.47)~式(4.7.49)则为无缩放的交替方向乘子法。

## 4.8 Newton 法

前面几节主要介绍了一阶优化算法。一阶优化算法只使用目标函数的零阶信息  $f(\mathbf{x})$  和一阶信息  $\nabla f(\mathbf{x})$ 。如果目标函数是二次可微分的，则利用 Hessian 矩阵的 Newton 法是二次或更快速收敛的<sup>[322]</sup>。因此，Newton 法是求解最优化问题的一种简单而有效的具体算法。本节主要介绍无约束最优化和等式约束最优化的 Newton 法。

在介绍 Newton 法时，将先讨论实数向量为变元的目标函数最小化的 Newton 法，然后推广到复数向量为变元的目标函数最小化的复 Newton 法。

### 4.8.1 无约束优化的 Newton 法

由于利用了 Hessian 矩阵提供的目标函数的二阶信息，求解无约束优化  $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$  的 Newton 法具有比梯度下降法和最速下降法更优的收敛性能。

对于无约束优化  $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$ ，若 Hessian 矩阵  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  正定，则由 Newton 矩阵方程  $\nabla^2 f(\mathbf{x}) \Delta \mathbf{x} = -\nabla f(\mathbf{x})$  可得 Newton 步  $\Delta \mathbf{x} = -(\nabla^2 f(\mathbf{x}))^{-1} \nabla f(\mathbf{x})$ ，这导致了下面的梯度下降算法

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mu_k (\nabla^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k) \quad (4.8.1)$$

这就是著名的 Newton 法。

Newton 法在应用中可能会遇到两个棘手的问题：

- (1) Hessian 矩阵  $\mathbf{H} = \nabla^2 f(\mathbf{x})$  难于求出。
- (2) 即便 Hessian 矩阵可以求出，但其求逆  $\mathbf{H}^{-1} = (\nabla^2 f(\mathbf{x}))^{-1}$  却有可能是数值不稳定的。

解决这两个棘手问题的方法有以下三种。

#### 1. 截尾 Newton 法

不直接利用 Hessian 矩阵的逆矩阵求 Newton 步长，而采用迭代方法由 Newton 矩阵方程  $\nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} = -\nabla f(\mathbf{x})$  求 Newton 步  $\Delta \mathbf{x}_{nt}$  的近似解。

使用迭代方法近似求解 Newton 方程的 Newton 法称为截尾 Newton 法 (truncated Newton method)<sup>[134]</sup>，在这种方法里，共轭梯度和预处理共轭梯度算法是近似求解 Newton 方程的主流方法。

截尾 Newton 法特别适合于大型无约束优化和约束优化问题以及内点法。

## 2. 修正 Newton 法

当 Hessian 矩阵不是正定矩阵时，可以对 Newton 方程进行修正<sup>[171]</sup>

$$(\nabla^2 f(\mathbf{x}) + \mathbf{E})\Delta\mathbf{x}_{\text{nt}} = -\nabla f(\mathbf{x}) \quad (4.8.2)$$

其中  $\mathbf{E}$  为半正定矩阵，通常取对角矩阵，使得  $\nabla^2 f(\mathbf{x}) + \mathbf{E}$  为对称正定矩阵。这一方法称为修正 Newton 法。典型的修正 Newton 法取  $\mathbf{E} = \delta \mathbf{I}$ ，其中  $\delta > 0$  很小。

## 3. 拟 Newton 法

在 Newton 法中，若使用一对称正定矩阵  $\mathbf{B}_k$  逼近 Hessian 矩阵的逆矩阵  $\mathbf{H}_k^{-1}$ ，便得到下面的算法

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{B}_k \nabla f(\mathbf{x}_k) \quad (4.8.3)$$

使用一对称矩阵近似 Hessian 矩阵的逆矩阵的 Newton 法称为拟 Newton 法 (quasi-Newton methods)。用对称矩阵  $\Delta\mathbf{H}_k = \mathbf{H}_{k+1} - \mathbf{H}_k$  近似 Hessian 矩阵，并记  $\rho_k = f'(\mathbf{x}_{k+1}) - f'(\mathbf{x}_k)$  和  $\delta_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ 。

根据 Hessian 矩阵近似的不同，拟 Newton 法有以下三种常用算法<sup>[363]</sup>：

### (1) 秩 1 更新算法

$$\Delta\mathbf{H}_k = \frac{(\delta_k - \mathbf{H}_k)(\delta_k - \mathbf{H}_k)^T}{\langle \delta_k - \mathbf{H}_k \rho_k, \rho_k \rangle}$$

### (2) DFP (Davidon-Fletcher-Powell) 算法

$$\Delta\mathbf{H}_k = \frac{\delta_k \delta_k^T}{\langle \rho_k, \delta_k \rangle} - \frac{\mathbf{H}_k \rho_k \rho_k^T \mathbf{H}_k}{\langle \mathbf{H}_k \rho_k, \rho_k \rangle}$$

### (3) BFGS (Broyden-Fletcher-Goldfarb-Shanno) 算法

$$\Delta\mathbf{H}_k = \frac{\mathbf{H}_k \rho_k \delta_k^T + \delta_k \rho_k^T \mathbf{H}_k}{\langle \mathbf{H}_k \rho_k, \rho_k \rangle} - \beta_k \frac{\mathbf{H}_k \rho_k \rho_k^T \mathbf{H}_k}{\langle \mathbf{H}_k \rho_k, \rho_k \rangle}$$

其中  $\beta_k = 1 + \langle \rho_k, \delta_k \rangle / \langle \mathbf{H}_k \rho_k, \rho_k \rangle$ 。

在各种 Newton 法的迭代过程中，通常都需要沿着直线  $\{\mathbf{x} + \mu \Delta\mathbf{x} | \mu \geq 0\}$  方向寻找最优点，这一步骤称为直线搜索 (linear search)。步长  $\mu$  的选择只是使目标函数沿着射线  $\{\mathbf{x} + \mu \Delta\mathbf{x} | \mu \geq 0\}$  近似最小化或者使目标函数“足够”减小，这种达到近似最小化的搜索称为不精确直线搜索 (inexact line search)。

不精确直线搜索的一个通用条件是：搜索直线  $\{\mathbf{x} + \mu \Delta\mathbf{x} | \mu \geq 0\}$  上的步长  $\mu_k$  首先必须充分降低目标函数  $f(\mathbf{x}_k)$ ，即保证

$$f(\mathbf{x}_k + \mu \Delta\mathbf{x}_k) < f(\mathbf{x}_k) + \alpha \mu (\nabla f(\mathbf{x}_k))^T \Delta\mathbf{x}_k, \quad \alpha \in (0, 1) \quad (4.8.4)$$

不等式条件式 (4.8.4) 有时称为 Armijo 条件<sup>[45, 169, 322, 372]</sup>。通常，对于比较大的步长  $\mu$ ，Armijo 条件往往不满足。因此，可以从单位步长  $\mu = 1$  开始搜索，若 Armijo 条件不满足，则需要通过一个回调因子  $\beta \in (0, 1)$ ，将步长下调至  $\mu = \beta \mu$ 。回调后，若 Armijo

条件仍不满足，则需要进一步回调步长  $\mu = \beta\mu$ 。如此反复，直至找到一个合适的步长  $\mu$ ，使得 Armijo 条件满足为止。这样一种搜索方法习惯称为 Newton 直线搜索或回溯直线搜索 (backtracking line search)。

回溯直线搜索方法可以确保目标函数  $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$ ，而且步长  $\mu$  又不至于太小。

#### 算法 4.8.1 无约束优化的 Newton 算法 (回溯直线搜索)

初始化 给定某个初始点  $\mathbf{x}_1 \in \text{dom } f(\mathbf{x})$  以及参数  $\alpha \in (0, 0.5), \beta \in (0, 1)$ 。令  $k = 1$ 。

步骤 1 计算目标函数的梯度  $\mathbf{b}_k = \nabla f(\mathbf{x}_k)$  和 Hessian 矩阵  $\mathbf{H}_k = \nabla^2 f(\mathbf{x}_k)$ ，求解 Newton 方程  $\mathbf{H}_k \Delta \mathbf{x}_k = -\mathbf{b}_k$ ，得到 Newton 步  $\Delta \mathbf{x}_k$ 。

步骤 2 回溯直线搜索：令  $\mu = 1$ ，若 Armijo 条件不满足，即  $f(\mathbf{x}_k + \mu \Delta \mathbf{x}_k) > f(\mathbf{x}_k) + \alpha \mu \mathbf{b}_k^T \Delta \mathbf{x}_k$ ，则回调步长  $\mu = \beta\mu$ ，并判断步长  $\mu$  回调之后，Armijo 条件是否还不满足，直至找到一个合适的步长  $\mu$ ，使得 Armijo 条件满足。

步骤 3 利用回溯直线搜索确定的步长  $\mu$ ，进行更新  $\mathbf{x}_{k+1} = \mathbf{x}_k + \mu \Delta \mathbf{x}_k$ 。

步骤 4 判断停止准则是否满足：若  $|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < \varepsilon$ ，则停止迭代，输出  $\mathbf{x}_k$ ；否则，令  $k \leftarrow k + 1$ ，并返回步骤 1，进行下一轮迭代，直至停止准则满足为止。

如果步骤 1 中的 Newton 方程采用其他方法求解，其他步骤不变，则算法 4.8.1 分别给出截尾 Newton 算法、修正 Newton 算法和拟 Newton 算法。

#### 4.8.2 无约束优化的复 Newton 法

考虑以复向量为变元的实函数的最小化  $\min f(\mathbf{z})$ ，其中  $\mathbf{z} \in \mathbb{C}^n, f : \mathbb{C}^n \rightarrow \mathbb{R}$ 。

由二元复变函数的二阶 Taylor 级数逼近

$$\begin{aligned} f(x + h_1, y + h_2) &= f(x, y) + h_1 \frac{\partial f(x, y)}{\partial x} + h_2 \frac{\partial f(x, y)}{\partial y} \\ &\quad + \frac{1}{2!} \left[ h_1^2 \frac{\partial^2 f(x, y)}{\partial x \partial x} + 2h_1 h_2 \frac{\partial^2 f(x, y)}{\partial x \partial y} + h_2^2 \frac{\partial^2 f(x, y)}{\partial y \partial y} \right] \end{aligned} \quad (4.8.5)$$

易知，全纯函数  $f(\mathbf{z}, \mathbf{z}^*)$  的二阶 Taylor 级数逼近为

$$\begin{aligned} f(\mathbf{z} + \Delta \mathbf{z}, \mathbf{z}^* + \Delta \mathbf{z}^*) &= f(\mathbf{z}, \mathbf{z}^*) + [(\nabla_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*))^T, (\nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*))^T] \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{z}^* \end{bmatrix} \\ &\quad + \frac{1}{2} [(\Delta \mathbf{z})^H, (\Delta \mathbf{z})^T] \begin{bmatrix} \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^T} & \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z}^* \partial \mathbf{z}^H} \\ \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^T} & \frac{\partial^2 f(\mathbf{z}, \mathbf{z}^*)}{\partial \mathbf{z} \partial \mathbf{z}^H} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{z}^* \end{bmatrix} \end{aligned} \quad (4.8.6)$$

由一阶优化条件  $\frac{\partial f(\mathbf{z} + \Delta \mathbf{z}, \mathbf{z}^* + \Delta \mathbf{z}^*)}{\partial \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{z}^* \end{bmatrix}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$ ，立即得到无约束优化的复 Newton 步满足的方程

$$\begin{bmatrix} \mathbf{H}_{\mathbf{z}^*, \mathbf{z}} & \mathbf{H}_{\mathbf{z}^*, \mathbf{z}^*} \\ \mathbf{H}_{\mathbf{z}, \mathbf{z}} & \mathbf{H}_{\mathbf{z}, \mathbf{z}^*} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z}_{\text{nt}} \\ \Delta \mathbf{z}_{\text{nt}}^* \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*) \\ \nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*) \end{bmatrix} \quad (4.8.7)$$

式中

$$\left. \begin{aligned} H_{z^*, z} &= \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^T}, & H_{z^*, z^*} &= \frac{\partial^2 f(z, z^*)}{\partial z^* \partial z^H} \\ H_{z, z} &= \frac{\partial^2 f(z, z^*)}{\partial z \partial z^T}, & H_{z, z^*} &= \frac{\partial^2 f(z, z^*)}{\partial z \partial z^H} \end{aligned} \right\} \quad (4.8.8)$$

分别是全纯函数  $f(z, z^*)$  的部分 Hessian 矩阵。于是，复 Newton 法的更新公式为

$$\begin{bmatrix} z_{k+1} \\ z_{k+1}^* \end{bmatrix} = \begin{bmatrix} z_k \\ z_k^* \end{bmatrix} + \mu \begin{bmatrix} \Delta z_k \\ \Delta z_k^* \end{bmatrix} \quad (4.8.9)$$

### 4.8.3 等式约束优化的 Newton 法

考虑等式约束优化问题

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to } A\mathbf{x} = \mathbf{b} \quad (4.8.10)$$

式中  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  为凸函数，可二次连续微分；而  $A \in \mathbb{R}^{p \times n}$ ，且  $\text{rank}(A) = p$ ，其中  $p < n$ 。

令  $\Delta \mathbf{x}_{nt}$  代表 Newton 搜索方向，则目标函数  $f(\mathbf{x})$  的二阶 Taylor 近似为

$$f(\mathbf{x} + \Delta \mathbf{x}_{nt}) = f(\mathbf{x}) + (\nabla f(\mathbf{x}))^T \Delta \mathbf{x}_{nt} + \frac{1}{2} (\Delta \mathbf{x}_{nt})^T \nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt}$$

其约束条件为

$$A(\mathbf{x} + \Delta \mathbf{x}_{nt}) = \mathbf{b} \quad \text{或} \quad A\Delta \mathbf{x}_{nt} = \mathbf{0}$$

换言之，Newton 搜索方向可以通过等式约束优化问题确定

$$\min_{\Delta \mathbf{x}_{nt}} f(\mathbf{x}) + (\nabla f(\mathbf{x}))^T \Delta \mathbf{x}_{nt} + \frac{1}{2} (\Delta \mathbf{x}_{nt})^T \nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} \quad \text{subject to } A\Delta \mathbf{x}_{nt} = \mathbf{0} \quad (4.8.11)$$

令  $\lambda$  是与等式约束  $A\Delta \mathbf{x}_{nt} = \mathbf{0}$  对应的 Lagrangian 乘子向量，可得到 Lagrangian 目标函数

$$L(\Delta \mathbf{x}_{nt}, \lambda) = f(\mathbf{x}) + (\nabla f(\mathbf{x}))^T \Delta \mathbf{x}_{nt} + \frac{1}{2} (\Delta \mathbf{x}_{nt})^T \nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} + \lambda^T A \Delta \mathbf{x}_{nt} \quad (4.8.12)$$

由一阶最优化条件  $\frac{\partial L(\Delta \mathbf{x}_{nt}, \lambda)}{\partial \Delta \mathbf{x}_{nt}} = \mathbf{0}$  和约束条件  $A\Delta \mathbf{x}_{nt} = \mathbf{0}$ ，易得

$$\nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} + A^T \lambda = \mathbf{0} \quad \text{和} \quad A\Delta \mathbf{x}_{nt} = \mathbf{0}$$

或合并写作

$$\begin{bmatrix} \nabla^2 f(\mathbf{x}) & A^T \\ A & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_{nt} \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}) \\ \mathbf{O} \end{bmatrix} \quad (4.8.13)$$

令等式约束优化问题式 (4.8.10) 的最优解  $\mathbf{x}^*$  存在，对应的目标函数  $f(\mathbf{x})$  的最优值

$$p^* = \inf \{f(\mathbf{x}) | A\mathbf{x} = \mathbf{b}\} = f(\mathbf{x}^*) \quad (4.8.14)$$

若令

$$\lambda^2(\mathbf{x}) = (\Delta \mathbf{x}_{nt})^T \nabla^2 f(\mathbf{x}) \Delta \mathbf{x}_{nt} \quad (4.8.15)$$

则可以证明<sup>[55]</sup>,  $\lambda^2(\mathbf{x})/2$  给出 Newton 算法收敛点  $\tilde{\mathbf{x}}$  的凸代价函数值  $f(\tilde{\mathbf{x}})$  与最优值  $p^*$  之间的偏差  $f(\tilde{\mathbf{x}}) - p^*$  的估计, 因此  $\lambda^2(\mathbf{x})$  可以用作 Newton 算法的停止准则。

#### 算法 4.8.2 可行点启动 Newton 算法 (等式约束优化)<sup>[55]</sup>

初始化 选择一个可行起始点  $\mathbf{x}_1 \in \text{dom } f$  且  $\mathbf{A}\mathbf{x}_1 = \mathbf{b}$ , 允许误差  $\epsilon > 0$ 。给定参数  $\alpha \in (0, 0.5), \beta \in (0, 1)$ 。令  $k = 1$ 。

步骤 1 计算目标函数在点  $\mathbf{x}_k$  的梯度向量  $\nabla f(\mathbf{x}_k)$  和 Hessian 矩阵  $\nabla^2 f(\mathbf{x}_k)$ 。

步骤 2 用预处理共轭梯度算法求解 KKT 方程

$$\begin{bmatrix} \nabla^2 f(\mathbf{x}_k) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_{\text{nt}}^{(k)} \\ \boldsymbol{\lambda}_{\text{nt}} \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}_k) \\ \mathbf{O} \end{bmatrix} \quad (4.8.16)$$

得到 Newton 搜索方向  $\Delta \mathbf{x}_{\text{nt}}^{(k)}$ 。

步骤 3 计算

$$\lambda^2(\mathbf{x}_k) = (\Delta \mathbf{x}_{\text{nt}}^{(k)})^T \nabla^2 f(\mathbf{x}_k) \Delta \mathbf{x}_{\text{nt}}^{(k)} \quad (4.8.17)$$

判断停止准则是否满足: 若  $\lambda^2(\mathbf{x}_k) < \epsilon$ , 则输出最优解  $\mathbf{x}_k$ , 并停止迭代; 否则, 转下一步, 进行回溯直线搜索。

步骤 4 回溯直线搜索: 令  $\mu = 1$ , 若  $f(\mathbf{x}_k + \mu \Delta \mathbf{x}_{\text{nt}}^{(k)}) > f(\mathbf{x}_k) + \alpha \mu (\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_{\text{nt}}^{(k)}$ , 则令  $\mu = \beta \mu$ , 再用回调后的  $\mu$  判断上述不等式是否还成立, 直至找到一个合适的步长  $\mu$ , 使得 Armijo 条件满足, 即上述不等式反向成立。

步骤 5 进行 Newton 更新  $\mathbf{x}_{k+1} = \mathbf{x}_k + \mu \Delta \mathbf{x}_{\text{nt}}^{(k)}$ 。令  $k \leftarrow k + 1$ , 并返回步骤 1, 进行新一轮 Newton 搜索方向更新和回溯直线搜索, 直至停止准则满足为止。

算法 4.8.2 以可行点作为初始点, 称为可行点启动 Newton 法。

可行点启动 Newton 法具有以下特点:

(1) 它是一种下降方法, 因为步骤 4 的回溯直线搜索能够保证目标函数在每一步迭代都是下降的, 并且每一个迭代点  $\mathbf{x}_k$  都满足等式约束。

(2) 该方法需要一可行点作为启动点。

然而, 有些情况下, 不容易找到一个可行点作为初始点。下面考虑将可行点启动 Newton 法推广为不可行点启动 Newton 法。

当  $\mathbf{x}_k$  是一个不可行点时, 考虑等式约束优化问题

$$\begin{aligned} \min_{\Delta \mathbf{x}_k} \quad & f(\mathbf{x}_k + \Delta \mathbf{x}_k) = f(\mathbf{x}_k) + (\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k + \frac{1}{2} (\Delta \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) \Delta \mathbf{x}_k \\ \text{subject to} \quad & \mathbf{A}(\mathbf{x}_k + \Delta \mathbf{x}_k) = \mathbf{b} \end{aligned}$$

令  $\boldsymbol{\lambda}_{k+1} (= \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k)$  是与等式约束  $\mathbf{A}(\mathbf{x}_k + \Delta \mathbf{x}_k) = \mathbf{b}$  对应的 Lagrangian 乘子向量, 可得 Lagrangian 目标函数

$$\begin{aligned} L(\Delta \mathbf{x}_k, \boldsymbol{\lambda}_{k+1}) = & f(\mathbf{x}_k) + (\nabla f(\mathbf{x}_k))^T \Delta \mathbf{x}_k + \frac{1}{2} (\Delta \mathbf{x}_k)^T \nabla^2 f(\mathbf{x}_k) \Delta \mathbf{x}_k \\ & + \boldsymbol{\lambda}_{k+1}^T [\mathbf{A}(\mathbf{x}_k + \Delta \mathbf{x}_k) - \mathbf{b}] \end{aligned}$$

由一阶最优化条件  $\frac{\partial L(\Delta\mathbf{x}_k, \lambda_{k+1})}{\partial \Delta\mathbf{x}_k} = \mathbf{0}$  和  $\frac{\partial L(\Delta\mathbf{x}_k, \lambda_{k+1})}{\partial \lambda_{k+1}} = \mathbf{0}$ , 易得

$$\nabla f(\mathbf{x}_k) + \nabla^2 f(\mathbf{x}_k) \Delta\mathbf{x}_k + \mathbf{A}^T \lambda_{k+1} = \mathbf{0}$$

$$\mathbf{A} \Delta\mathbf{x}_k = -(\mathbf{A}\mathbf{x}_k - \mathbf{b})$$

或合并写作

$$\begin{bmatrix} \nabla^2 f(\mathbf{x}_k) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_k \\ \lambda_{k+1} \end{bmatrix} = - \begin{bmatrix} \nabla f(\mathbf{x}_k) \\ \mathbf{A}\mathbf{x}_k - \mathbf{b} \end{bmatrix} \quad (4.8.18)$$

将  $\lambda_{k+1} = \lambda_k + \Delta\lambda_k$  代入上式, 立即有

$$\begin{bmatrix} \nabla^2 f(\mathbf{x}_k) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_k \\ \Delta\lambda_k \end{bmatrix} = - \begin{bmatrix} \nabla f(\mathbf{x}_k) + \mathbf{A}^T \lambda_k \\ \mathbf{A}\mathbf{x}_k - \mathbf{b} \end{bmatrix} \quad (4.8.19)$$

式 (4.8.19) 启示了 Newton 算法的停止准则之一。定义残差向量

$$\mathbf{r}(\mathbf{x}_k, \lambda_k) = \begin{bmatrix} \mathbf{r}_{\text{dual}}(\mathbf{x}_k, \lambda_k) \\ \mathbf{r}_{\text{pri}}(\mathbf{x}_k, \lambda_k) \end{bmatrix} = \begin{bmatrix} \nabla f(\mathbf{x}_k) + \mathbf{A}^T \lambda_k \\ \mathbf{A}\mathbf{x}_k - \mathbf{b} \end{bmatrix} \quad (4.8.20)$$

式中

$$\mathbf{r}_{\text{dual}}(\mathbf{x}_k, \lambda_k) = \nabla f(\mathbf{x}_k) + \mathbf{A}^T \lambda_k \quad (4.8.21)$$

$$\mathbf{r}_{\text{pri}}(\mathbf{x}_k, \lambda_k) = \mathbf{A}\mathbf{x}_k - \mathbf{b} \quad (4.8.22)$$

分别表示对偶残差向量和原始残差向量。

显然, 不可行点启动 Newton 算法的停止准则之一可以归纳为

$$\begin{bmatrix} \Delta\mathbf{x}_k \\ \Delta\lambda_k \end{bmatrix} \approx \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \Leftrightarrow \begin{bmatrix} \mathbf{r}_{\text{dual}}(\mathbf{x}_k, \lambda_k) \\ \mathbf{r}_{\text{pri}}(\mathbf{x}_k, \lambda_k) \end{bmatrix} \approx \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \Leftrightarrow \|\mathbf{r}(\mathbf{x}_k, \lambda_k)\|_2 < \varepsilon \quad (4.8.23)$$

对很小的扰动误差  $\varepsilon > 0$  成立。

不可行点启动 Newton 算法的另一个停止准则是等式约束条件必须满足。这一停止准则保证了 Newton 算法的收敛点一定是可行点, 虽然其初始点和多数迭代点允许是不可行的。

### 算法 4.8.3 不可行点启动 Newton 算法 (等式约束优化)<sup>[55]</sup>

初始化 选择一个不可行起始点  $\mathbf{x}_1 \in \mathbb{R}^n$ 、任意初始 Lagrangian 乘子向量  $\lambda_1 \in \mathbb{R}^p$  和允许误差  $\varepsilon > 0$ 。给定参数  $\alpha \in (0, 0.5)$ ,  $\beta \in (0, 1)$ , 令  $k = 1$ 。

步骤 1 计算目标函数在点  $\mathbf{x}_k$  的梯度向量  $\nabla f(\mathbf{x}_k)$  和 Hessian 矩阵  $\nabla^2 f(\mathbf{x}_k)$ 。

步骤 2 判断停止准则是否满足: 若  $\mathbf{A}\mathbf{x}_k = \mathbf{b}$ , 并且  $\|\mathbf{r}(\mathbf{x}_k, \lambda_k)\|_2 < \varepsilon$ , 则输出  $\mathbf{x}_k, \lambda_k$ , 并停止迭代; 否则, 转至下一步。

步骤 3 用预处理共轭梯度算法求解 KKT 方程式 (4.8.19), 求 Newton 步  $(\Delta\mathbf{x}_k, \Delta\lambda_k)$ 。

步骤 4 回溯直线搜索: 令  $\mu = 1$ , 若  $f(\mathbf{x}_k + \Delta\mathbf{x}_k) > f(\mathbf{x}_k) + \alpha\mu(\nabla f(\mathbf{x}_k))^T \Delta\mathbf{x}_k$ , 则令  $\mu = \beta\mu$ , 再用回调后的  $\mu$  判断此不等式是否还成立, 直至找到一个合适的  $\mu$ , 使得这一不等式条件反向成立。

### 步骤5 进行 Newton 更新

$$\begin{bmatrix} \mathbf{z}_{k+1} \\ \boldsymbol{\lambda}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_k \\ \boldsymbol{\lambda}_k \end{bmatrix} + \mu \begin{bmatrix} \Delta \mathbf{z}_k \\ \Delta \boldsymbol{\lambda}_k \end{bmatrix} \quad (4.8.24)$$

令  $k \leftarrow k + 1$ , 返回步骤1, 并重复以上步骤, 直至停止准则满足为止。

#### 4.8.4 等式约束优化的复 Newton 法

考虑等式约束下复向量为变元的实目标函数的最小化问题

$$\min_{\mathbf{z}} f(\mathbf{z}) \quad \text{subject to } A\mathbf{z} = \mathbf{b} \quad (4.8.25)$$

式中  $\mathbf{z} \in \mathbb{C}^n$ ,  $f : \mathbb{C}^n \rightarrow \mathbb{R}$  为凸函数, 可二次连续微分; 而  $A \in \mathbb{C}^{p \times n}$ , 且  $\text{rank}(A) = p$ , 其中  $p < n$ 。

令  $(\Delta \mathbf{z}_k, \Delta \boldsymbol{\lambda}_k^*)$  代表第  $k$  次迭代的复搜索方向, 则全纯函数  $f(\mathbf{z}_k, \mathbf{z}_k^*)$  的二阶 Taylor 展开为

$$\begin{aligned} f(\mathbf{z}_k + \Delta \mathbf{z}_k, \mathbf{z}_k^* + \Delta \mathbf{z}_k^*) &= f(\mathbf{z}_k, \mathbf{z}_k^*) + (\nabla_{\mathbf{z}_k} f(\mathbf{z}_k, \mathbf{z}_k^*))^T \Delta \mathbf{z}_k + (\nabla_{\mathbf{z}_k^*} f(\mathbf{z}_k, \mathbf{z}_k^*))^T \Delta \mathbf{z}_k^* \\ &\quad + \frac{1}{2} [(\Delta \mathbf{z}_k)^H, (\Delta \mathbf{z}_k)^T] \begin{bmatrix} \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^H} \\ \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^H} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z}_k \\ \Delta \mathbf{z}_k^* \end{bmatrix} \end{aligned}$$

其约束条件为

$$A(\mathbf{z}_k + \Delta \mathbf{z}_k) = \mathbf{b} \quad \text{或} \quad A\Delta \mathbf{z}_k = \mathbf{0} \quad (\text{若 } \mathbf{z}_k \text{ 为可行点})$$

换言之, Newton 搜索方向可以通过等式约束优化问题确定

$$\begin{array}{ll} \min_{\Delta \mathbf{z}_k, \Delta \mathbf{z}_k^*} & f(\mathbf{z}_k + \Delta \mathbf{z}_k, \mathbf{z}_k^* + \Delta \mathbf{z}_k^*) \\ \text{subject to} & A\Delta \mathbf{z}_k = \mathbf{0} \end{array} \quad (4.8.26)$$

令  $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k \in \mathbb{R}^p$  是与等式约束条件  $A\Delta \mathbf{z}_k = \mathbf{0}$  对应的 Lagrangian 乘子向量, 则等式约束优化问题式 (4.8.26) 可转换为无约束优化问题

$$\min_{\Delta \mathbf{z}_k, \Delta \mathbf{z}_k^*; \Delta \boldsymbol{\lambda}_k} f(\mathbf{z}_k + \Delta \mathbf{z}_k, \mathbf{z}_k^* + \Delta \mathbf{z}_k^*) + (\boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k)^T A\Delta \mathbf{z}_k$$

由一阶最优化条件  $\frac{\partial L(\Delta \mathbf{z}_k, \Delta \mathbf{z}_k^*; \boldsymbol{\lambda}_{k+1})}{\partial \begin{bmatrix} \Delta \mathbf{z}_k \\ \Delta \mathbf{z}_k^* \end{bmatrix}} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$  和  $\frac{\partial L(\Delta \mathbf{z}_k, \Delta \mathbf{z}_k^*; \boldsymbol{\lambda}_{k+1})}{\partial \Delta \boldsymbol{\lambda}_k} = \mathbf{0}$  易得 Newton 方程

$$\begin{bmatrix} \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^H} & \mathbf{A}^T \\ \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^H} & \mathbf{O} \\ \mathbf{A} & \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z}_k \\ \Delta \mathbf{z}_k^* \\ \Delta \boldsymbol{\lambda}_k \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{z}_k} f(\mathbf{z}_k, \mathbf{z}_k^*) + \mathbf{A}^T \boldsymbol{\lambda}_k \\ \nabla_{\mathbf{z}_k^*} f(\mathbf{z}_k, \mathbf{z}_k^*) \\ \mathbf{0} \end{bmatrix} \quad (4.8.27)$$

定义残差向量

$$\mathbf{r}(\mathbf{z}_k, \mathbf{z}_k^*, \boldsymbol{\lambda}_k) = \begin{bmatrix} \nabla_{\mathbf{z}_k} f(\mathbf{z}_k, \mathbf{z}_k^*) + \mathbf{A}^T \boldsymbol{\lambda}_k \\ \nabla_{\mathbf{z}_k^*} f(\mathbf{z}_k, \mathbf{z}_k^*) \end{bmatrix} \quad (4.8.28)$$

#### 算法 4.8.4 可行点启动复 Newton 算法 (等式约束优化)

初始化 选择一个可行起始点  $\mathbf{z}_1 \in \text{dom } f$  且  $\mathbf{A}\mathbf{z}_1 = \mathbf{b}$ , 允许误差  $\varepsilon > 0$ . 给定参数  $\alpha \in (0, 0.5), \beta \in (0, 1)$ . 令  $k = 1$ .

步骤 1 由式 (4.8.28) 计算残差向量, 判断停止准则是否满足: 若  $\|\mathbf{r}(\mathbf{z}_k, \mathbf{z}_k^*, \boldsymbol{\lambda}_k)\|_2 < \varepsilon$ , 则输出最优解  $\mathbf{z}_k$ , 并停止迭代; 否则, 继续下面的步骤。

步骤 2 计算目标函数在点  $\mathbf{z}_k$  的梯度向量  $\nabla_{\mathbf{z}} f(\mathbf{z}_k, \mathbf{z}_k^*)$ , 共轭梯度向量  $\nabla_{\mathbf{z}^*} f(\mathbf{z}_k, \mathbf{z}_k^*)$  以及全 Hessian 矩阵

$$\mathbf{H}_k = \begin{bmatrix} \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^H} \\ \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^H} \end{bmatrix} \quad (4.8.29)$$

步骤 3 用共轭梯度或者预处理共轭梯度算法解 Newton 方程式 (4.8.27), 得 Newton 步  $(\Delta \mathbf{z}_{\text{nt},k}, \Delta \mathbf{z}_{\text{nt},k}^*, \Delta \boldsymbol{\lambda}_{\text{nt},k})$ .

步骤 4 回溯直线搜索: 令  $\mu = 1$ , 若

$$f(\mathbf{z}_k + \mu \Delta \mathbf{z}_{\text{nt},k}, \mathbf{z}_k^* + \mu \Delta \mathbf{z}_{\text{nt},k}^*) > f(\mathbf{z}_k, \mathbf{z}_k^*) + \alpha \mu [(\nabla_{\mathbf{z}} f(\mathbf{z}, \mathbf{z}^*))^T, (\nabla_{\mathbf{z}^*} f(\mathbf{z}, \mathbf{z}^*))^T] \begin{bmatrix} \Delta \mathbf{z}_{\text{nt},k} \\ \Delta \mathbf{z}_{\text{nt},k}^* \end{bmatrix}$$

则下调步长  $\mu = \beta \mu$ , 然后再判断上述不等式是否满足: 若满足, 则进一步下调步长  $\mu = \beta \mu$ ; 直至找到一个合适的  $\mu$ , 使上式的不等式号反向成立。

步骤 5 进行 Newton 更新

$$\begin{bmatrix} \mathbf{z}_{k+1} \\ \mathbf{z}_{k+1}^* \end{bmatrix} = \begin{bmatrix} \mathbf{z}_k \\ \mathbf{z}_k^* \end{bmatrix} + \mu \begin{bmatrix} \Delta \mathbf{z}_{\text{nt},k} \\ \Delta \mathbf{z}_{\text{nt},k}^* \end{bmatrix}, \quad \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k$$

令  $k \leftarrow k + 1$ , 并返回步骤 1, 重复以上步骤, 直至停止准则满足为止。

下面讨论用不可行点作为启动点的复 Newton 算法。此时, 与式 (4.8.26) 对应的等式约束优化问题为

$$\begin{array}{ll} \min_{\Delta \mathbf{z}_k, \Delta \mathbf{z}_k^*} & f(\mathbf{z}_k + \Delta \mathbf{z}_k, \mathbf{z}_k^* + \Delta \mathbf{z}_k^*) \\ \text{subject to} & \mathbf{A}(\mathbf{z}_k + \Delta \mathbf{z}_k) = \mathbf{b} \end{array} \quad (4.8.30)$$

此时, Newton 方程式 (4.8.27) 修正为

$$\begin{bmatrix} \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k^* \partial \mathbf{z}_k^H} & \mathbf{A}^T \\ \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^T} & \frac{\partial^2 f(\mathbf{z}_k, \mathbf{z}_k^*)}{\partial \mathbf{z}_k \partial \mathbf{z}_k^H} & \mathbf{O} \\ \mathbf{A} & \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z}_k \\ \Delta \mathbf{z}_k^* \\ \Delta \boldsymbol{\lambda}_k \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{z}_k} f(\mathbf{z}_k, \mathbf{z}_k^*) + \mathbf{A}^T \boldsymbol{\lambda}_k \\ \nabla_{\mathbf{z}_k^*} f(\mathbf{z}_k, \mathbf{z}_k^*) \\ \mathbf{A}\mathbf{z}_k - \mathbf{b} \end{bmatrix} \quad (4.8.31)$$

与之对应的残差向量为

$$\mathbf{r}(\mathbf{z}_k, \mathbf{z}_k^*, \boldsymbol{\lambda}_k) = \begin{bmatrix} \nabla_{\mathbf{z}_k} f(\mathbf{z}_k, \mathbf{z}_k^*) + \mathbf{A}^T \boldsymbol{\lambda}_k \\ \nabla_{\mathbf{z}_k^*} f(\mathbf{z}_k, \mathbf{z}_k^*) \\ \mathbf{A}\mathbf{z}_k - \mathbf{b} \end{bmatrix} \quad (4.8.32)$$

将可行点启动 Newton 算法 4.8.4 中步骤 1 的残差计算公式式 (4.8.28) 替换成式 (4.8.32), 步骤 3 的待求解 Newton 方程式 (4.8.27) 换成式 (4.8.31), 即可得到不可行点启动 Newton 算法。注意, 由式 (4.8.32) 知, 残差向量足够逼近零向量时,  $\mathbf{A}\mathbf{z}_k$  即充分逼近  $\mathbf{b}$ , 因而算法的收敛点一定是一个可行点。

## 4.9 原始-对偶内点法

业已公认<sup>[171]</sup>, 内点法 (interior point method) 这一术语是 Fiacco 和 McCormick 最早于 1968 年在他们具有开创性的著作 (文献 [164, p.41]) 中提出的。然而, 苦于缺乏低复杂度的优化算法, 内点法在此后的十几年间并未获得较大的发展; 只是直到 Karmarkar<sup>[259]</sup> 于 1984 年提出了具有多项式复杂度的线性规划算法之后, 连续优化的领域才发生了巨大的变化, 这一变化被形容为“内点革命” (interior-point revolution)<sup>[171]</sup>。内点革命导致了连续优化问题的思考方法的根本性转变: 以前多年认为互不相关的优化领域实际具有统一的理论框架。

### 4.9.1 非线性优化的原始-对偶问题

考虑标准形式的非线性优化问题

$$\min f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{0}, \quad \mathbf{z} \succeq \mathbf{0} \quad (4.9.1)$$

其中  $\mathbf{x} \in \mathbb{R}^n, f: \mathbb{R}^n \rightarrow \mathbb{R}, \mathbf{d}: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^m, \mathbf{z} \in \mathbb{R}_+^m$ ; 并且  $f(\mathbf{x})$  和  $d_i(\mathbf{x}, \mathbf{z}), i = 1, \dots, m$  均为凸函数, 且二次可连续微分。

非线性优化模型式 (4.9.1) 包括了以下几种常见优化模型:

(1) 若  $\mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{h}(\mathbf{x}) - \mathbf{z}$ , 其中  $\mathbf{z} \succeq \mathbf{0}$ , 则式 (4.9.1) 给出文献 [500] 的非线性优化模型

$$\min f(\mathbf{x}) \quad \text{subject to} \quad h_i(\mathbf{x}) \geq 0, \quad i = 1, \dots, m \quad (4.9.2)$$

(2) 令  $\mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{h}(\mathbf{x}) + \mathbf{z}$ , 且  $z_i = 0, i = 1, \dots, p; h_{p+i} = g_i, z_{p+i} \geq 0, i = 1, \dots, m-p$ , 则式 (4.9.1) 与文献 [72] 的非线性优化模型一致

$$\min f(\mathbf{x}) \quad \text{subject to} \quad h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p; \quad g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m-p \quad (4.9.3)$$

(3) 若  $\mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{c}(\mathbf{x}) - \mathbf{z}$ , 并且  $z_i = 0, i \in \mathcal{E}; z_i \geq 0, i \in \mathcal{I}$ , 则式 (4.9.1) 给出文献 [171] 的非线性优化模型

$$\min f(\mathbf{x}) \quad \text{subject to} \quad c_i(\mathbf{x}) = 0, \quad i \in \mathcal{E}; \quad c_i(\mathbf{x}) \geq 0, \quad i \in \mathcal{I}; \quad i = 1, \dots, m \quad (4.9.4)$$

(4) 若  $\mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{c}(\mathbf{x}), \mathbf{z} = \mathbf{x} \succeq \mathbf{0}$ , 则式 (4.9.1) 与文献 [506] 的非线性优化模型一致

$$\min f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{c}(\mathbf{x}) = \mathbf{0}, \quad \mathbf{x} \succeq \mathbf{0} \quad (4.9.5)$$

(5) 若取  $f(\mathbf{x}) = \mathbf{c}^T \mathbf{x}$ ,  $\mathbf{z} = \mathbf{x} \succeq \mathbf{0}$ ,  $m = n$  以及  $\mathbf{d}(\mathbf{x}, \mathbf{z}) = \mathbf{A}\mathbf{x} - \mathbf{b}$ , 则式 (4.9.1) 给出文献 [429] 的线性规划模型

$$\min \mathbf{c}^T \mathbf{x} \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \succeq \mathbf{0} \quad (4.9.6)$$

对于不等式约束非线性优化的原始问题

$$(P) \quad \min f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{h}(\mathbf{x}) \succeq \mathbf{0} \quad (\text{其中 } \mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m) \quad (4.9.7)$$

其对偶问题可表示为<sup>[500]</sup>

$$(D) \quad \max L(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) - \mathbf{y}^T \mathbf{h}(\mathbf{x}) + [(\nabla \mathbf{h}(\mathbf{x}))^T \mathbf{y} - \nabla f(\mathbf{x})]^T \mathbf{x} \quad (4.9.8)$$

$$\text{subject to} \quad (\nabla \mathbf{h}(\mathbf{x}))^T \mathbf{y} = \nabla f(\mathbf{x}), \quad \mathbf{y} \succeq \mathbf{0}$$

式中  $\mathbf{x} \in \mathbb{R}^n$  为原始变量,  $\mathbf{y} \in \mathbb{R}^m$  为对偶变量。

#### 4.9.2 一阶原始-对偶内点法

为了更好地理解内点法, 下面先以线性规划问题

$$\min \{\mathbf{c}^T \mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \succeq \mathbf{0}\} \quad (4.9.9)$$

为讨论对象。

由式 (4.9.8) 知, 上述线性规划的对偶问题为

$$\max \{\mathbf{b}^T \mathbf{y} : \mathbf{A}^T \mathbf{y} + \mathbf{z} = \mathbf{c}, \quad \mathbf{z} \succeq \mathbf{0}\} \quad (4.9.10)$$

式中,  $\mathbf{x} \in \mathbb{R}_+^n$ ,  $\mathbf{y} \in \mathbb{R}^m$ ,  $\mathbf{z} \in \mathbb{R}_+^m$  分别为原始变量、对偶变量和松弛变量, 矩阵  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , 向量  $\mathbf{b} \in \mathbb{R}^m$ ,  $\mathbf{c} \in \mathbb{R}^n$ 。不失一般性, 假定矩阵  $\mathbf{A}$  满行秩, 即  $\text{rank}(\mathbf{A}) = m$ 。

利用一阶优化条件易得原始-对偶问题的 KKT 方程

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b}, \quad \mathbf{x} \succeq \mathbf{0} \\ \mathbf{A}^T \mathbf{y} + \mathbf{z} &= \mathbf{c}, \quad \mathbf{z} \succeq \mathbf{0} \\ x_i z_i &= 0, \quad i = 1, \dots, n \end{aligned}$$

前两个条件分别是原始问题和对偶问题的一阶优化条件及可行性条件, 第三个为互补性条件。由于前两个条件包含有非负性要求, 决定了上述 KKT 方程只能用迭代方法求解。

互补性条件  $x_i z_i = 0$  可以用中心化条件  $x_i z_i = \mu, i = 1, \dots, n$  代替, 其中  $\mu > 0$ 。关系式  $x_i z_i = \mu, \forall i = 1, \dots, n$  也称扰动互补性 (perturbed complementarity) 条件或者互补松弛度 (complementary slackness)。显然, 若  $\mu \rightarrow 0$ , 则  $x_i z_i \rightarrow 0$ 。

利用扰动互补性代替互补性条件, 可以将 KKT 方程等价写作

$$\left. \begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b}, \quad \mathbf{x} \succeq \mathbf{0} \\ \mathbf{A}^T \mathbf{y} + \mathbf{z} &= \mathbf{c}, \quad \mathbf{z} \succeq \mathbf{0} \\ \mathbf{x}^T \mathbf{z} &= n\mu \end{aligned} \right\} \quad (4.9.11)$$

**定义 4.9.1** <sup>[430]</sup> 若原始问题有一个可行解  $\mathbf{x} \succ 0$ , 对偶问题也有一个解  $(\mathbf{y}, \mathbf{z})$ , 并且  $\mathbf{z} \succ 0$ , 则称原始-对偶问题满足内点条件 (interior-point condition, IPC)。

若内点条件满足, 则将式 (4.9.11) 的解记作  $(\mathbf{x}(\mu), \mathbf{y}(\mu), \mathbf{z}(\mu))$ , 并称之为原始问题  $(P)$  和对偶问题  $(D)$  的  $\mu$ -中心。所有  $\mu$ -中心的集合称作  $(P)$  和  $(D)$  的中心路径。

将  $\mathbf{x}_k = \mathbf{x}_{k-1} + \Delta \mathbf{x}_k, \mathbf{y}_k = \mathbf{y}_{k-1} + \Delta \mathbf{y}_k$  和  $\mathbf{z}_k = \mathbf{z}_{k-1} + \Delta \mathbf{z}_k$  代入式 (4.9.11), 则有<sup>[429]</sup>

$$\begin{aligned} \mathbf{A}\Delta \mathbf{x}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_{k-1} \\ \mathbf{A}^T \Delta \mathbf{y}_k + \Delta \mathbf{z}_k &= \mathbf{c} - \mathbf{A}^T \mathbf{y}_{k-1} - \mathbf{z}_{k-1} \\ \mathbf{z}_{k-1}^T \Delta \mathbf{x}_k + \mathbf{x}_k^T \Delta \mathbf{z}_{k-1} &= n\mu - \mathbf{x}_{k-1}^T \mathbf{z}_{k-1} \end{aligned}$$

或等价写作

$$\begin{bmatrix} \mathbf{A} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{A}^T & \mathbf{I} \\ \mathbf{z}_{k-1}^T & \mathbf{0}^T & \mathbf{x}_{k-1}^T \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_k \\ \Delta \mathbf{y}_k \\ \Delta \mathbf{z}_k \end{bmatrix} = \begin{bmatrix} \mathbf{b} - \mathbf{A}\mathbf{x}_{k-1} \\ \mathbf{c} - \mathbf{A}^T \mathbf{y}_{k-1} - \mathbf{z}_{k-1} \\ n\mu - \mathbf{x}_{k-1}^T \mathbf{z}_{k-1} \end{bmatrix} \quad (4.9.12)$$

由一阶优化条件得到的内点法称为一阶原始-对偶内点法, 其关键步骤是求解式 (4.9.12), 得到 Newton 步  $(\Delta \mathbf{x}_k, \Delta \mathbf{y}_k, \Delta \mathbf{z}_k)$ 。这一步骤可以直接求逆矩阵, 但数值性能更好的方法是使用迭代方法求解式 (4.9.12)。然而, 由于最左边的矩阵不是实对称矩阵, 所以共轭梯度法和预处理共轭梯度法都无法使用。

#### 算法 4.9.1 可行点启动原始-对偶内点法<sup>[429]</sup>

输入 精度参数  $\epsilon > 0$ ; 障碍更新参数  $\theta, 0 < \theta < 1$ : 可行点  $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ , 且  $\mathbf{x}_0^T \mathbf{z}_0 = n\mu_0$ 。

初始化  $k = 1$ 。

步骤 1 求解 KKT 方程式 (4.9.12), 得解  $(\Delta \mathbf{x}_k, \Delta \mathbf{y}_k, \Delta \mathbf{z}_k)$ 。

步骤 2  $\mu$  更新  $\mu_k = (1 - \theta)\mu_{k-1}$ 。

步骤 3 进行原始变量、对偶变量和 Lagrangian 乘子的更新

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \rho \Delta \mathbf{x}_k, \quad \mathbf{y}_k = \mathbf{y}_{k-1} + \rho \Delta \mathbf{y}_k, \quad \mathbf{z}_k = \mathbf{z}_{k-1} + \rho \Delta \mathbf{z}_k$$

步骤 4 判断收敛准则是否满足: 若  $\mathbf{x}_k^T \mathbf{z}_k < \epsilon$ , 则输出  $(\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k)$ ; 否则, 令  $k \leftarrow k + 1$ , 并返回步骤 1, 继续迭代, 直到收敛准则满足。

上述算法要求  $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$  为可行点。这一可行点可以用下面的算法迭代确定。

#### 算法 4.9.2 可行点算法<sup>[429]</sup>

输入 精度参数  $\epsilon > 0$ ; 障碍更新参数  $\theta, 0 < \theta < 1$ ; 阈值参数  $\tau > 0$ 。

初始化  $\mathbf{x}_0 \succ 0, \mathbf{z}_0 \succ 0, \mathbf{y}_0, \mathbf{x}_0^T \mathbf{z}_0 = n\mu_0$ , 令  $k = 1$ 。

步骤 1 计算残差向量  $r_b^{k-1} = \mathbf{b} - \mathbf{A}^T \mathbf{x}_{k-1}$  和  $r_c^{k-1} = \mathbf{c} - \mathbf{A}^T \mathbf{y}_{k-1} - \mathbf{z}_{k-1}$ 。

步骤 2  $\mu$  更新  $\nu_{k-1} = \mu_{k-1}/\mu_0$ 。

步骤 3 求解 KKT 方程

$$\begin{bmatrix} \mathbf{A} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{A}^T & \mathbf{I} \\ \mathbf{z}_{k-1}^T & \mathbf{0}^T & \mathbf{x}_{k-1}^T \end{bmatrix} \begin{bmatrix} \Delta^f \mathbf{x}_k \\ \Delta^f \mathbf{y}_k \\ \Delta^f \mathbf{z}_k \end{bmatrix} = \begin{bmatrix} \theta \nu_{k-1} r_b^0 \\ \theta \nu_{k-1} r_c^0 \\ n\mu - \mathbf{x}_{k-1}^T \mathbf{z}_{k-1} \end{bmatrix}$$

步骤4 变量更新  $(\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k) = (\mathbf{x}_{k-1}, \mathbf{y}_{k-1}, \mathbf{z}_{k-1}) + (\Delta^f \mathbf{x}_k, \Delta^f \mathbf{y}_k, \Delta^f \mathbf{z}_k)$ 。

步骤5 收敛准则检验：若  $\max\{\mathbf{x}_k^T \mathbf{z}_k, \mathbf{b} - \mathbf{A}^T \mathbf{x}_k, \mathbf{c} - \mathbf{A}^T \mathbf{y}_k - \mathbf{z}_k\} \leq \varepsilon$ ，则输出  $(\mathbf{x}, \mathbf{y}, \mathbf{z}) = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{z}_k)$ ；否则，令  $k \leftarrow k + 1$ ，并返回步骤1，继续以上迭代，直至收敛准则满足。

### 4.9.3 二阶原始-对偶内点法

一阶原始-对偶内点法存在以下缺点：① KKT 方程的矩阵不是对称矩阵，难于采用共轭梯度或者预处理共轭梯度等有效算法求解。② KKT 方程只由一阶优化条件得到，未使用 Hessian 矩阵提供的二阶统计信息。③ 难于确保将迭代点控制为内点。④ 不方便推广到一般的非线性优化问题。

为了克服一阶原始-对偶内点法的缺点，非线性优化的原始-对偶内点法由三个基本要素组成：

(1) 障碍函数 将变量  $\mathbf{x}$  限定为可行内点。

(2) Newton 法 等式约束最小化的 Newton 法用于有效求解 KKT 方程。

(3) 回溯直线搜索 用于确定一个合适的步长。

这种障碍函数与 Newton 法相结合的内点法称为二阶原始-对偶内点法。

定义松弛变量  $\mathbf{z} \in \mathbb{R}^m$ ，它是一个满足  $\mathbf{h}(\mathbf{x}) - \mathbf{z} = \mathbf{0}$  的非负变量  $\mathbf{z} \succeq \mathbf{0}$ 。借助松弛变量，不定式约束的原始问题  $(P)$  可以等价表示成等式约束的优化问题

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{h}(\mathbf{x}) - \mathbf{z}, \mathbf{z} \succeq \mathbf{0} \quad (4.9.13)$$

为了进一步消去不等式  $\mathbf{z} \succeq \mathbf{0}$ ，引入经典 Fiacco-McCormick 对数障碍函数

$$b_\mu(\mathbf{x}, \mathbf{z}) = f(\mathbf{x}) - \mu \sum_{i=1}^m \log(z_i) \quad (4.9.14)$$

式中  $\mu > 0$  为障碍参数。对于非常小的  $\mu$ ，除了接近约束等于零的点之外，障碍函数  $b_\mu(\mathbf{x}, \mathbf{z})$  与原目标函数  $f(\mathbf{x})$  二者的作用相像。

等式约束的优化问题式 (4.9.13) 现在可表示为无约束优化问题

$$\min_{\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda}} L_\mu(\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \mu \sum_{i=1}^m \log(z_i) - \boldsymbol{\lambda}^T (\mathbf{h}(\mathbf{x}) - \mathbf{z}) \quad (4.9.15)$$

式中  $L_\mu(\mathbf{x}, \mathbf{z}, \boldsymbol{\lambda})$  为 Lagrangian 目标函数， $\boldsymbol{\lambda} \in \mathbb{R}^m$  为 Lagrangian 乘子，或叫对偶变量。

记

$$\mathbf{Z} = \text{diag}(z_1, \dots, z_m), \quad \mathbf{A} = \text{diag}(\lambda_1, \dots, \lambda_m) \quad (4.9.16)$$

令  $\nabla_{\mathbf{x}} L_\mu = \frac{\partial L_\mu}{\partial \mathbf{x}^T} = \mathbf{0}$ ,  $\nabla_{\mathbf{z}} L_\mu = \frac{\partial L_\mu}{\partial \mathbf{z}^T} = \mathbf{0}$  和  $\nabla_{\boldsymbol{\lambda}} L_\mu = \frac{\partial L_\mu}{\partial \boldsymbol{\lambda}^T} = \mathbf{0}$ ，并用  $\mathbf{A}$  左乘第二个等式两边，即得一阶优化条件

$$\left. \begin{aligned} \nabla f(\mathbf{x}) - (\nabla h(\mathbf{x}))^T \boldsymbol{\lambda} &= \mathbf{0} \\ -\mu \mathbf{1} + \mathbf{Z} \mathbf{A} \mathbf{1} &= \mathbf{0} \\ \mathbf{h}(\mathbf{x}) - \mathbf{z} &= \mathbf{0} \end{aligned} \right\} \quad (4.9.17)$$

式中  $\mathbf{1}$  是一个全部元素为 1 的  $m$  维向量。

为了推导无约束优化问题式 (4.9.15) 的 Newton 方程, 考虑 Lagrangian 目标函数

$$\begin{aligned} L_\mu(\mathbf{x} + \Delta\mathbf{x}, \mathbf{z} + \Delta\mathbf{z}, \boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}) = & f(\mathbf{x} + \Delta\mathbf{x}) - \mu \sum_{i=1}^m \log(z_i + \Delta z_i) \\ & - (\boldsymbol{\lambda} + \Delta\boldsymbol{\lambda})^\top [\mathbf{h}(\mathbf{x} + \Delta\mathbf{x}) - \mathbf{z} - \Delta\mathbf{z}] \end{aligned}$$

式中

$$\begin{aligned} f(\mathbf{x} + \Delta\mathbf{x}) &= f(\mathbf{x}) + (\nabla f(\mathbf{x}))^\top \Delta\mathbf{x} + \frac{1}{2} (\Delta\mathbf{x})^\top \nabla^2 f(\mathbf{x}) \Delta\mathbf{x} \\ h_i(\mathbf{x} + \Delta\mathbf{x}) &= h_i(\mathbf{x}) + (\nabla h_i(\mathbf{x}))^\top \Delta\mathbf{x} + \frac{1}{2} (\Delta\mathbf{x})^\top \nabla^2 h_i(\mathbf{x}) \Delta\mathbf{x}, \quad i = 1, \dots, m \end{aligned}$$

令  $\nabla_{\Delta\mathbf{x}} L_\mu = \frac{\partial L_\mu}{\partial(\Delta\mathbf{x})^\top} = \mathbf{0}$ ,  $\nabla_{\Delta\mathbf{z}} L_\mu = \frac{\partial L_\mu}{\partial(\Delta\mathbf{z})^\top} = \mathbf{0}$  和  $\nabla_{\Delta\boldsymbol{\lambda}} L_\mu = \frac{\partial L_\mu}{\partial(\Delta\boldsymbol{\lambda})^\top} = \mathbf{0}$ , 即得 Newton 方程 [500]

$$\begin{bmatrix} \mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) & \mathbf{O} & -(\mathbf{A}(\mathbf{x}))^\top \\ \mathbf{O} & \mathbf{A} & \mathbf{Z} \\ \mathbf{A}(\mathbf{x}) & -\mathbf{I} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{z} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}) + (\nabla h(\mathbf{x}))^\top \boldsymbol{\lambda} \\ \mu \mathbf{1} - \mathbf{Z} \boldsymbol{\Lambda} \mathbf{1} \\ \mathbf{z} - \mathbf{h}(\mathbf{x}) \end{bmatrix} \quad (4.9.18)$$

式中

$$\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) = \nabla^2 f(\mathbf{x}) - \sum_{i=1}^m \lambda_i \nabla^2 h_i(\mathbf{x}), \quad \mathbf{A}(\mathbf{x}) = \nabla \mathbf{h}(\mathbf{x}) \quad (4.9.19)$$

式 (4.9.18) 中,  $\Delta\mathbf{x}$  称为优化方向 (optimality direction),  $\Delta\mathbf{z}$  为中央方向 (centrality direction), 而  $\Delta\boldsymbol{\lambda}$  为可行方向 (feasibility direction) [500]。三元组  $(\Delta\mathbf{x}, \Delta\mathbf{z}, \Delta\boldsymbol{\lambda})$  组成内点法更新的 Newton 步即搜索方向。

式 (4.9.18) 的第一个方程两边同乘  $-1$ , 第二个方程两边左乘  $-\mathbf{Z}^{-1}$ , 则有 [500]

$$\begin{bmatrix} -\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) & \mathbf{O} & (\mathbf{A}(\mathbf{x}))^\top \\ \mathbf{O} & -\mathbf{Z}^{-1} \boldsymbol{\Lambda} & -\mathbf{I} \\ \mathbf{A}(\mathbf{x}) & -\mathbf{I} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{z} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha} \\ -\boldsymbol{\beta} \\ \gamma \end{bmatrix} \quad (4.9.20)$$

其中

$$\boldsymbol{\alpha} = \nabla f(\mathbf{x}) - (\nabla h(\mathbf{x}))^\top \boldsymbol{\lambda} \quad (4.9.21)$$

$$\boldsymbol{\beta} = \mu \mathbf{Z}^{-1} \mathbf{1} - \boldsymbol{\lambda} \quad (4.9.22)$$

$$\gamma = \mathbf{z} - \mathbf{h}(\mathbf{x}) \quad (4.9.23)$$

以上三个变量具有以下含义:

(1)  $\gamma$  度量原始不可行性,  $\gamma = \mathbf{0}$  意味着  $\mathbf{x}$  是满足等式约束  $\mathbf{h}(\mathbf{x}) - \mathbf{z} = \mathbf{0}$  的可行点; 否则,  $\mathbf{x}$  是不可行点。

(2)  $\boldsymbol{\alpha}$  度量对偶不可行性,  $\boldsymbol{\alpha} = \mathbf{0}$  意味着对偶变量  $\boldsymbol{\lambda}$  满足一阶优化条件, 是可行的; 否则, 违背一阶优化条件, 是不可行的。

(3)  $\beta$  测量互补松弛度,  $\beta = \mathbf{0}$  即  $\mu Z^{-1} \mathbf{1} = \lambda$  意味着互补松弛性  $z_i \lambda_i = \mu, \forall i = 1, \dots, m$ , 并且  $\mu = 0$  时, 互补性完全满足; 而  $\mu$  偏离 0 值越小, 互补松弛度越小; 反之, 互补松弛度则越大。

式 (4.9.20) 可以分解成两部分

$$\Delta z = \mathbf{A}^{-1} \mathbf{Z}(\beta - \Delta \lambda) \quad (4.9.24)$$

和

$$\begin{bmatrix} -\mathbf{H}(\mathbf{x}, \lambda) & (\mathbf{A}(\mathbf{x}))^T \\ \mathbf{A}(\mathbf{x}) & \mathbf{Z} \mathbf{A}^{-1} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} \alpha \\ \gamma + \mathbf{Z} \mathbf{A}^{-1} \beta \end{bmatrix} \quad (4.9.25)$$

重要的是, 原 Newton 方程式 (4.9.20) 变成了维数更小的子 Newton 方程式 (4.9.25)。

现在, Newton 方程式 (4.9.25) 很容易通过预处理共轭梯度算法迭代求解。一旦 Newton 步  $(\Delta \mathbf{x}, \Delta z, \Delta \lambda)$  由式 (4.9.25) 的解和式 (4.9.24) 求出之后, 即可进行下列更新

$$\left. \begin{array}{l} \mathbf{x}_{k+1} = \mathbf{x}_k + \eta \Delta \mathbf{x}_k \\ \mathbf{z}_{k+1} = \mathbf{z}_k + \eta \Delta \mathbf{z}_k \\ \lambda_{k+1} = \lambda_k + \eta \Delta \lambda_k \end{array} \right\} \quad (4.9.26)$$

式中  $\eta$  为更新的共同步长。

### 1. 凸优化问题的修正

内点法的关键是保证迭代点  $\mathbf{x}_k$  为内点, 即满足  $\mathbf{h}(\mathbf{x}_k) \succ \mathbf{0}$ 。然而, 对于非二次型凸优化, 只是简单地选择步长并不足于保证非负变量的正性。为此, 有必要对凸优化问题进行适当的修正。

凸优化问题的一种简单修正是引入评价函数 (merit function)。与单纯的代价函数最小化不同, 评价函数的最小化有两个目的: 既要促使迭代点向目标函数的局部极小点靠拢, 又要保证迭代点是等式约束的可行点。因此, 评价函数的主要作用有两个: ① 将约束优化问题变成无约束约束问题; ② 当评价函数的值越小时, 迭代点越靠近原始约束优化问题的最优解。

考虑经典的 Fiacco-McCormick 评价函数 [164]

$$\Psi_{\rho, \mu}(\mathbf{x}, \mathbf{z}) = f(\mathbf{x}) - \mu \sum_{i=1}^m \log(z_i) + \frac{\rho}{2} \|\mathbf{h}(\mathbf{x}) - \mathbf{z}\|_2^2 \quad (4.9.27)$$

若定义对偶正规矩阵 (dual normal matrix)

$$\mathbf{N}(\mathbf{x}, \lambda, \mathbf{z}) = \mathbf{H}(\mathbf{x}, \lambda) + (\mathbf{A}(\mathbf{x}))^T \mathbf{Z}^{-1} \mathbf{A}(\mathbf{x}) \quad (4.9.28)$$

则下列定理成立。

**定理 4.9.1** <sup>[500]</sup> 令  $b(\mathbf{x}, \lambda) = f(\mathbf{x}) - \sum_{i=1}^m \lambda_i \log(z_i)$  表示障碍函数。假定对偶正规矩阵  $\mathbf{N}(\mathbf{x}, \lambda, \mathbf{z})$  正定, 则由式 (4.9.20) 确定的搜索方向  $(\Delta \mathbf{x}, \Delta \lambda)$  具有以下性质:

(1) 若  $\gamma = \mathbf{0}$ , 则

$$\begin{bmatrix} \nabla_x b(\mathbf{x}, \boldsymbol{\lambda}) \\ \nabla_{\boldsymbol{\lambda}} b(\mathbf{x}, \boldsymbol{\lambda}) \end{bmatrix}^T \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \boldsymbol{\lambda} \end{bmatrix} \leq 0$$

(2) 存在一个  $\rho_{\min} \geq 0$ , 使得对于每一个  $\rho > \rho_{\min}$ , 下列不等式成立

$$\begin{bmatrix} \nabla_x \Psi_{\rho, \mu}(\mathbf{x}, \mathbf{z}) \\ \nabla_{\boldsymbol{\lambda}} \Psi_{\rho, \mu}(\mathbf{x}, \mathbf{z}) \end{bmatrix}^T \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \boldsymbol{\lambda} \end{bmatrix} \leq 0$$

在上述两种情况下, 等式成立当且仅当  $(\mathbf{x}, \mathbf{z})$  对某个  $\boldsymbol{\lambda}$  满足式 (4.9.17)。

## 2. 非凸优化问题的修正

对于非凸优化问题, Hessian 矩阵  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda})$  可能不是半正定的, 从而使对偶正规矩阵  $\mathbf{N}(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{z})$  可能不是正定的。在这种情况下, 可以对 Hessian 矩阵加一个很小的扰动, 用  $\tilde{\mathbf{H}}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) + \delta \mathbf{I}$  代替对偶正规矩阵里的 Hessian 矩阵  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda})$ , 得

$$\tilde{\mathbf{N}}(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{z}) = \mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) + \delta \mathbf{I} + (\mathbf{A}(\mathbf{x}))^T \mathbf{Z}^{-1} \boldsymbol{\Lambda} \mathbf{A}(\mathbf{x})$$

**定理 4.9.2**<sup>[499]</sup> 若对偶正规矩阵  $\tilde{\mathbf{N}}(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{z})$  正定, 则使用  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) + \delta \mathbf{I}$  代替  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda})$  之后, 由式 (4.9.20) 确定的搜索方向  $(\Delta \mathbf{x}, \Delta \boldsymbol{\lambda}, \Delta \mathbf{z})$ :

- (1) 是函数  $\|\mathbf{h}(\mathbf{x}) - \mathbf{z}\|_2^2$  的下降方向。
- (2) 是互补松驰性  $z_i \lambda_i = \mu, i = 1, \dots, m$  或  $\mathbf{Z} \boldsymbol{\Lambda} \mathbf{1} = \mu \mathbf{1}$  的下降方向。

定理 4.9.2 表明, 对于非凸优化问题, 用小扰动的 Hessian 矩阵  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda}) + \delta \mathbf{I}$  代替原 Hessian 矩阵  $\mathbf{H}(\mathbf{x}, \boldsymbol{\lambda})$ , 可以保证内点法收敛至满足约束条件  $\mathbf{h}(\mathbf{x}) \succ \mathbf{0}$  和互补性  $x_i \lambda_i = 0, i = 1, \dots, m$ 。

## 本章小结

最优化问题的求解取决于标量目标函数关于自变元 (矩阵或向量) 的梯度和 Hessian 矩阵。本章首先分别以实矩阵 (含实向量) 和复矩阵 (含复向量) 为目标函数的变元, 讨论了梯度、共轭梯度以及 Hessian 矩阵的计算方法。特别地, 矩阵微分在求梯度矩阵和 Hessian 矩阵中起着重要的作用。

围绕优化理论和算法, 本章重点介绍了凸优化理论、一阶与二阶优化算法。针对一阶优化算法, 本章依次介绍了平滑函数优化的梯度法和 Nesterov 最优梯度法, 非平滑函数优化的次梯度法、共轭函数法和逼近梯度法以及约束优化算法等。然后, 介绍了二阶优化算法 (Newton 法和原始-对偶内点法)。

## 习题

4.1 令  $\mathbf{y}$  是一实值观测数据向量, 由  $\mathbf{y} = \alpha \mathbf{x} + \mathbf{v}$  给出, 其中,  $\alpha$  为实标量,  $\mathbf{x}$  代表

一实值确定性过程，而加性噪声向量  $\mathbf{v}$  具有零均值向量，协方差矩阵  $\mathbf{R}_v = \mathbf{E}\{\mathbf{v}\mathbf{v}^T\}$ 。试求一最优滤波器向量  $\mathbf{w}$ ，使得估计子  $\hat{\alpha} = \mathbf{w}^T \mathbf{y}$  是一个方差最小的无偏估计子。

**4.2** 令  $f(t)$  为一已知函数。考虑二次型函数的最小化

$$\text{minimize } Q(x) = \text{minimize} \int_0^1 [f(t) - x_0 - x_1 t - \cdots - x_n t^n]^2 dt$$

判断与线性方程组对应的矩阵是否病态？

**4.3** 考虑方程  $\mathbf{y} = \mathbf{A}\theta + \mathbf{e}$ ，其中， $\mathbf{e}$  为误差向量。定义加权误差平方和

$$E_w \stackrel{\text{def}}{=} \mathbf{e}^H \mathbf{W} \mathbf{e}$$

其中， $\mathbf{W}$  为一 Hermitian 正定矩阵，它对误差起加权作用。

(1) 求使  $E_w$  最小化的参数向量  $\theta$  的解。这一解称为  $\theta$  的加权最小二乘估计。

(2) 利用  $LDL^H$  分解  $\mathbf{W} = LDL^H$ ，证明加权最小二乘准则相当于使误差或数据向量进行预白化。

**4.4** 令代价函数为  $f(\mathbf{w}) = \mathbf{w}^H \mathbf{R}_e \mathbf{w}$ ，并且给滤波器加约束条件  $\mathbf{R}(\mathbf{w}^H \mathbf{x}) = b$ ，其中  $b$  为一常数。试求最优滤波器  $\mathbf{w}$ 。

**4.5** 解释下列有约束最优化问题是否有解：

- (1)  $\min\{x_1 + x_2\}$ , 约束条件为  $x_1^2 + x_2^2 = 2$ ,  $0 \leq x_1 \leq 1$ ,  $0 \leq x_2 \leq 1$ ;
- (2)  $\min\{x_1 + x_2\}$ , 约束条件为  $x_1^2 + x_2^2 \leq 1$ ,  $x_1 + x_2 = 4$ ;
- (3)  $\min\{x_1 x_2\}$ , 约束条件为  $x_1 + x_2 = 3$ 。

**4.6** 考虑约束优化问题

$$\min(x - 1)(y + 1) \quad \text{subject to } x - y = 0$$

利用 Lagrangian 乘子法证明极小点为  $(1, 1)$ ，且 Lagrangian 乘子  $\lambda = 1$ 。若 Lagrangian 函数取

$$\psi(x, y) = (x - 1)(y + 1) - \lambda(x - y)$$

证明  $\phi(x, y)$  在  $(0, 0)$  有一个鞍点，即点  $(0, 0)$  不能使  $\psi(x, y)$  极小化。

**4.7** 求解约束优化问题  $\min J(x, y, z) = x^2 + y^2 + z^2$ ，约束条件为  $3x + 4y - z = 25$ 。

**4.8** 假定  $\mathbf{x}$  是  $N$  维数据或文本向量，现在希望寻找一  $n \times N$  线性变换矩阵  $\mathbf{W}$  对  $\mathbf{x}$  进行数据压缩： $\mathbf{y} = \mathbf{W}\mathbf{x}$ ，使得  $n \ll N$ 。定义目标函数  $J_W = \text{tr}[(\mathbf{W}\mathbf{S}_{zw}\mathbf{W}^T)^{-1}\mathbf{W}\mathbf{S}_{zb}\mathbf{W}^T]$ ，其中， $\mathbf{S}_{zw}$  和  $\mathbf{S}_{zb}$  分别是原数据向量  $\mathbf{x}$  的类内和类间散布矩阵。线性变换矩阵的优化准则是使目标函数  $J_W$  极大化。设  $\mathbf{S}_{zw}^{-1}\mathbf{S}_{zb}$  的特征值为  $\lambda_1, \lambda_2, \dots, \lambda_N$  ( $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$ )，并且  $\mathbf{u}_i$  是与特征值  $\lambda_i$  对应的特征向量。证明  $\mathbf{W} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]$ ，并且  $\max J_W = \sum_{i=1}^n \lambda_i$ 。  
(提示：使用矩阵微分求梯度矩阵  $\partial J_W / \partial \mathbf{W}$ )

**4.9** 证明<sup>[328]</sup>

$$\begin{aligned} d(\mathbf{F}^\dagger \mathbf{F}) &= \mathbf{F}^\dagger (d\mathbf{F})(\mathbf{I} - \mathbf{F}^\dagger \mathbf{F}) + [\mathbf{F}^\dagger (d\mathbf{F})(\mathbf{I} - \mathbf{F}^\dagger \mathbf{F})]^\dagger \\ d(\mathbf{FF}^\dagger) &= (\mathbf{I} - \mathbf{FF}^\dagger)(d\mathbf{F})\mathbf{F}^\dagger + [(\mathbf{I} - \mathbf{FF}^\dagger)(d\mathbf{F})\mathbf{F}^\dagger]^\dagger \end{aligned}$$

式中,  $\mathbf{A}^\dagger$  是  $\mathbf{A}$  的 Moore-Penrose 逆矩阵。

#### 4.10 已知线性方程

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ \lambda_1 & \lambda_2 & \cdots & \lambda_n \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^{n-1} & \lambda_2 & \cdots & \lambda_n^{n-1} \end{bmatrix} \begin{bmatrix} d\lambda_1 \\ d\lambda_2 \\ \vdots \\ d\lambda_n \end{bmatrix} = \begin{bmatrix} \text{tr}(d\mathbf{X}) \\ \text{tr}(\mathbf{X}_0 d\mathbf{X}) \\ \vdots \\ \text{tr}(\mathbf{X}_0^{n-1} d\mathbf{X}) \end{bmatrix}$$

求  $d\lambda_i$ 。

#### 4.11 求标量函数 $f(\mathbf{X}) = \mathbf{a}^T \mathbf{X} \mathbf{X}^T \mathbf{a}$ 的 Hessian 矩阵。

#### 4.12 令观测数据向量由线性回归模型

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad E\{\boldsymbol{\varepsilon}\} = \mathbf{0}, \quad E\{\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T\} = \sigma^2 \mathbf{I}$$

产生。现在希望设计一个滤波器矩阵  $\mathbf{A}$ , 其输出向量  $\mathbf{e} = \mathbf{A}\mathbf{y}$  满足  $E\{\mathbf{e} - \boldsymbol{\varepsilon}\} = \mathbf{0}$ , 并且可以使得  $E\{(\mathbf{e} - \boldsymbol{\varepsilon})^T(\mathbf{e} - \boldsymbol{\varepsilon})\}$  最小化。证明这个最优化问题等效为

$$\min [\text{tr}(\mathbf{A}^T \mathbf{A}) - 2\text{tr}(\mathbf{A})]$$

约束条件为  $\mathbf{A}\mathbf{X} = \mathbf{O}$ , 其中,  $\mathbf{O}$  为零矩阵。

#### 4.13 证明最优化问题

$$\min [\text{tr}(\mathbf{A}^T \mathbf{A}) - 2\text{tr}(\mathbf{A})] \quad \text{subject to } \mathbf{A}\mathbf{X} = \mathbf{O} \quad (\text{零矩阵})$$

的解矩阵为  $\hat{\mathbf{A}} = \mathbf{I} - \mathbf{X}\mathbf{X}^\dagger$ 。

#### 4.14 证明无约束问题

$$\min [(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T (\mathbf{V} + \mathbf{X}\mathbf{X}^T)^\dagger (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]$$

与下面的约束问题具有相同的解向量  $\boldsymbol{\beta}$ :

$$\min [(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\mathbf{V}^\dagger(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})] \quad \text{subject to } (\mathbf{I} - \mathbf{V}\mathbf{V}^\dagger)\mathbf{X}\boldsymbol{\beta} = (\mathbf{I} - \mathbf{V}\mathbf{V}^\dagger)\mathbf{y}$$

4.15 令  $f(\mathbf{x}, \mathbf{y}) \geq 0$ 。证明: 其极大-极小化函数与极小-极大化函数之间存在以下关系

$$\max_{\mathbf{x}} \min_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) \leq \min_{\mathbf{y}} \max_{\mathbf{x}} f(\mathbf{x}, \mathbf{y})$$

提示: 可先证明  $\max_{\mathbf{x}} \left( \min_t f(\mathbf{x}, t) \right) \leq \min_{\mathbf{y}} \left( \max_s f(s, \mathbf{y}) \right)$ 。

#### 4.16 求解下列关于 $\mathbf{A}$ 的约束最优化问题

$$\min [\text{tr}(\mathbf{A}^T \mathbf{A}) - 2\text{tr}(\mathbf{A})], \quad \text{subject to } \mathbf{A}\mathbf{x} = \mathbf{0}$$

#### 4.17 求解下列关于 $\mathbf{x}$ 的约束最优化问题

$$\min \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r, \quad \text{subject to } -1 \leq x_i \leq 1, \quad i = 1, 2, 3,$$

其中

$$\mathbf{P} = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, \quad \mathbf{q} = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, \quad r = 1$$

#### 4.18 证明约束最优化问题

$$\min \frac{1}{2} \mathbf{x}^T \mathbf{x} \quad \text{subject to } \mathbf{C}\mathbf{x} = \mathbf{b}$$

具有唯一解  $\mathbf{x}^* = \mathbf{C}^\dagger \mathbf{b}$ 。

4.19 令矩阵  $\mathbf{Y} \in \mathbb{R}^{n \times m}$ ,  $\mathbf{Z} \in \mathbb{R}^{n \times (n-m)}$ , 它们的列构成线性无关的集合。如果将服从约束条件  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的解向量表示为  $\mathbf{x} = \mathbf{Y}\mathbf{x}_Y + \mathbf{Z}\mathbf{x}_Z$ , 其中,  $\mathbf{x}_Y$  和  $\mathbf{x}_Z$  分别是某个  $m \times 1$  和  $(n-m) \times 1$  向量。证明解向量为  $\mathbf{x} = \mathbf{Y}(\mathbf{AY})^{-1}\mathbf{b} + \mathbf{Z}\mathbf{x}_Z$ 。

4.20 若约束最优化问题为  $\min \text{tr}(\mathbf{AV}\mathbf{A}^T)$ , 约束条件是  $\mathbf{AX} = \mathbf{W}$ , 证明

$$\mathbf{A} = \mathbf{W}(\mathbf{X}^T \mathbf{V}_0^\dagger \mathbf{X})^\dagger \mathbf{X}^T \mathbf{V}_0^\dagger + \mathbf{Q}(\mathbf{I} - \mathbf{V}_0 \mathbf{V}_0^\dagger)$$

其中,  $\mathbf{V}_0 = \mathbf{V} + \mathbf{XX}^T$ ,  $\mathbf{Q}$  是一任意矩阵。

4.21 约束最优化问题为  $\min \text{tr}(\mathbf{AS}^T)$ , 约束条件为  $\mathbf{AX} = \mathbf{O}$ ,  $\mathbf{AA}^T = \mathbf{I}$ 。证明最优化问题的解为 [328, pp.302~303]

$$\mathbf{A} = (\mathbf{SMS}^T)^{-1/2} \mathbf{SM}, \quad \mathbf{M} = \mathbf{I} - \mathbf{XX}^\dagger$$

4.22 [328, p.367] 令  $\mathbf{S}$  和  $\Phi$  是两个已知的  $m \times m$  正定矩阵, 并且  $\Phi$  为对角矩阵。令  $f$  是一实值函数, 由

$$f(\mathbf{A}) = \log |\mathbf{AA}^T + \Phi| + \text{tr}((\mathbf{AA}^T + \Phi)^{-1} \mathbf{S})$$

定义, 其中,  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $1 \leq n \leq m$ 。证明:

(1) 当  $\mathbf{A} = \Phi^{1/2} \mathbf{T}(\mathbf{A} - \mathbf{I}_n)^{1/2}$  时, 实值函数  $f$  达到最小。式中,  $\Phi$  是一个  $n \times n$  对角矩阵, 它包含了矩阵  $\Phi^{-1/2} \mathbf{S} \Phi^{-1/2}$  的  $n$  个最大特征值, 而矩阵  $\mathbf{T}$  是一个  $m \times n$  矩阵, 由矩阵  $\Phi^{-1/2} \mathbf{S} \Phi^{-1/2}$  的  $n$  个主特征向量组成。

(2) 实值函数  $f$  的最小值为

$$m + \log |\mathbf{S}| + \sum_{i=n+1}^m (\lambda_i - \log \lambda_i - 1)$$

式中,  $\lambda_{n+1}, \lambda_{n+2}, \dots, \lambda_m$  表示矩阵  $\Phi^{-1/2} \mathbf{S} \Phi^{-1/2}$  的  $m-n$  个最小的特征值。

4.23 令  $p > 1$ ,  $a_i \geq 0$ ,  $i = 1, 2, \dots, n$ , 证明: 对每一组满足  $\sum_{i=1}^n x_i^q = 1$  ( $q = p/(p-1)$ ) 的非负实数  $x_1, x_2, \dots, x_n$ , 不等式

$$\sum_{i=1}^n a_i x_i \leq \left( \sum_{i=1}^n a_i^p \right)^{1/p}$$

成立。这一不等式称为  $\left(\sum_{i=1}^n a_i^p\right)^{1/p}$  的表示定理 [328, p.218]，并且等号成立，当且仅当  $a_1 = a_2 = \dots = a_n = 0$  或者  $x_i^q = a_i^p \left(\sum_{k=1}^n a_k^p\right)^{-1}$ ,  $i = 1, 2, \dots, n$ 。

**4.24** 考虑  $M$  个实谐波信号的 Pisarenko 谐波分解的下列推广<sup>[291]</sup>。令噪声子空间的维数大于 1，于是张成噪声子空间的矩阵  $V_n$  的每一个列向量的元素都满足

$$\sum_{k=0}^{2M} v_k e^{j\omega_i k} = \sum_{k=0}^{2M} v_k e^{-j\omega_i k} = 0, \quad 1 \leq i \leq M$$

令  $\bar{p} = V_n \alpha$  表示  $V_n$  的列向量的非退化线性组合。所谓非退化，乃是指由向量  $\bar{p} = [\bar{p}_0, \bar{p}_1, \dots, \bar{p}_{2M}]^T$  的元素构造的多项式  $p(z)$  至少具有  $2M$  阶，即  $p(z) = \bar{p}_0 + \bar{p}_1 z + \dots + \bar{p}_{2M} z^{2M}$ ,  $\bar{p}_{2M} \neq 0$ 。于是，这一多项式也满足上面的式子。这意味着，所有谐波频率均可由多项式  $p(z)$  位于单位圆上的  $2M$  个根求出。现在希望选择系数向量  $\alpha$  满足条件:  $p_0 = 1$  和  $\sum_{k=1}^K p_k^2 = \min$ 。

(1) 令  $v^T$  是矩阵  $V_n$  的第一行，而  $V$  是由  $V_n$  的其他所有行组成的矩阵。若  $p$  是由  $\bar{p}$  除第一个元素以外的其他元素组成的向量，试证明

$$\alpha = \arg \min \alpha^T V^T V \alpha \quad \text{subject to } v^T \alpha = 1$$

(2) 利用 Lagrangian 乘子法证明约束优化问题的解为

$$\alpha = \frac{(V^T V)^{-1} v}{v^T (V^T V)^{-1} v}, \quad p = \frac{V(V^T V)^{-1} v}{v^T (V^T V)^{-1} v}$$

**4.25** 设矩阵  $X$  的秩  $\text{rank}(X) \leq r$ ，证明：对所有满足  $A^T A = I_r$  的半正交矩阵  $A$  和所有矩阵  $Z \in \mathbb{R}^{n \times r}$ ，恒有

$$\min \text{tr} \left( (X - ZA^T)(X - ZA^T)^T \right) = 0$$

# 第5章 奇异值分析

Beltrami (1835–1899) 和 Jordan (1838–1921) 二位学者被公认为是奇异值分解的创始人: Beltrami 于 1873 年发表了奇异值分解的第一篇论文<sup>[38]</sup>, 一年后 Jordan 发表了自己对奇异值分解的独立推导<sup>[257]</sup>。现在, 奇异值分解 (包括各种推广) 已是数值线性代数的最有用和最有效的工具之一, 它在统计分析、物理和应用科学 (如信号与图像处理、系统理论和控制、通信、计算机视觉等) 中被广泛地应用。

本章首先介绍数值算法的数值稳定性与条件数的概念, 以引出矩阵奇异值分解的必要性; 然后详细讨论奇异值分解和广义奇异值分解的数值计算及应用。接着将介绍奇异值分解的最新推广——奇异值阈值化和奇异值投影以及它们在应用科学的热门领域——矩阵完备化、低秩与稀疏矩阵分解等中的应用。

## 5.1 数值稳定性与条件数

在信息科学与工程等许多应用中, 在对数据进行处理时, 常常需要考虑一个重要问题: 实际的观测数据存在某种程度的不确定性或误差, 而且对数据进行的数值计算也总是伴随有误差。误差有何影响? 数据处理和数值分析的算法稳定吗? 为了回答这些问题, 下面两个概念是极其重要的:

- (1) 一种算法的数值稳定性;
- (2) 所涉及问题的条件或扰动分析。

假定  $f$  表示用数学定义的某个问题, 它作用于数据  $d \in D$  (其中  $D$  表示某个数据组), 并产生一个解  $f(d) \in F$  ( $F$  代表某个解集)。给定  $d \in D$ , 我们希望计算  $f(d)$ 。通常, 只能够已知  $d$  的某个近似值  $d^*$ , 我们所能够做到的就是计算  $f(d^*)$ 。如果  $f(d^*)$  “逼近”  $f(d)$ , 那么问题就是“良性”的。若  $d^*$  接近  $d$  时,  $f(d^*)$  有可能与  $f(d)$  相差很大, 我们就称问题是“病态”的。如果没有有关问题的更详细的信息, 术语“逼近”就不可能准确地描述问题。

在扰动理论中, 称求解  $f(d)$  的某种算法是数值上稳定的, 若它引入的对扰动的敏感度不会比原问题本身固有的敏感度更大。稳定性可以保证稍有扰动时问题的解接近无扰动时的解。更确切地说, 令  $f^*$  表示用于实现或近似  $f$  的一算法, 则  $f^*$  是稳定的, 若对所有  $d \in D$ , 存在一接近  $d$  的  $d^* \in D$  使得  $f(d^*)$  (稍有扰动的问题的解) 接近解  $f^*(d)$ 。

当然, 我们不可能期望求解病态问题的一种稳定算法会具有比数据无扰动时更高的精确度。然而, 一种不稳定的算法甚至会对良性问题给出差的结果。因此, 在确定某个解的精度时, 有两个不同的因素必须考虑: 首先, 若算法是稳定的, 则  $f^*(d)$  应该接近

$f(d^*)$ : 其次, 若问题是良性的, 则  $f(d^*)$  应该接近  $f(d)$ 。这样,  $f^*(d)$  就会接近  $f(d)$ 。

下面讨论数值稳定性的数学描述。

在工程中, 经常会遇到线性方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$ , 其中,  $n \times n$  矩阵  $\mathbf{A}$  是一个元素为已知数值的系数矩阵,  $n \times 1$  向量  $\mathbf{b}$  为已知向量, 而  $n \times 1$  向量  $\mathbf{x}$  是一个待求解的未知参数向量。系数矩阵  $\mathbf{A}$  非奇异时, 由于独立的方程个数和未知参数的个数相等, 故方程具有唯一解, 称为适定方程。很自然地, 我们会对这个方程解的稳定性产生兴趣: 如果系数矩阵  $\mathbf{A}$  与 (或) 向量  $\mathbf{b}$  发生扰动, 那么方程的解向量  $\mathbf{x}$  会如何变化呢? 还能够保持一定的稳定性吗? 研究方程的解向量  $\mathbf{x}$  如何受系数矩阵  $\mathbf{A}$  和系数向量  $\mathbf{b}$  的元素微小变化 (扰动) 的影响, 将得到描述矩阵  $\mathbf{A}$  的一个重要特征的数值, 称为条件数 (condition number)。

为了分析的方便, 先假定只存在向量  $\mathbf{b}$  的扰动  $\delta\mathbf{b}$ , 而矩阵  $\mathbf{A}$  是稳定不变的。此时, 精确的解向量  $\mathbf{x}$  就会扰动为  $\mathbf{x} + \delta\mathbf{x}$ , 即有

$$\mathbf{A}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b} + \delta\mathbf{b} \quad (5.1.1)$$

这意味着

$$\delta\mathbf{x} = \mathbf{A}^{-1}\delta\mathbf{b} \quad (5.1.2)$$

因为  $\mathbf{A}\mathbf{x} = \mathbf{b}$ 。对式 (5.1.2) 应用矩阵范数的性质, 得

$$\|\delta\mathbf{x}\|_2 \leq \|\mathbf{A}^{-1}\|_2 \|\delta\mathbf{b}\|_2 \quad (5.1.3)$$

对线性方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  也使用矩阵范数的相同性质, 又有

$$\|\mathbf{b}\|_2 \leq \|\mathbf{A}\|_2 \|\mathbf{x}\|_2 \quad (5.1.4)$$

由式 (5.1.3) 和式 (5.1.4), 立即得到

$$\frac{\|\delta\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leq (\|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2) \frac{\|\delta\mathbf{b}\|_2}{\|\mathbf{b}\|_2} \quad (5.1.5)$$

然后, 一并考虑扰动  $\delta\mathbf{A}$  的影响。此时, 线性方程变为

$$(\mathbf{A} + \delta\mathbf{A})(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b}$$

由上式可推导出

$$\begin{aligned} \delta\mathbf{x} &= [(\mathbf{A} + \delta\mathbf{A})^{-1} - \mathbf{A}^{-1}]\mathbf{b} \\ &= \{\mathbf{A}^{-1}[\mathbf{A} - (\mathbf{A} + \delta\mathbf{A})](\mathbf{A} + \delta\mathbf{A})^{-1}\}\mathbf{b} \\ &= -\mathbf{A}^{-1}\delta\mathbf{A}(\mathbf{A} + \delta\mathbf{A})^{-1}\mathbf{b} \\ &= -\mathbf{A}^{-1}\delta\mathbf{A}(\mathbf{x} + \delta\mathbf{x}) \end{aligned} \quad (5.1.6)$$

由此得

$$\|\delta\mathbf{x}\|_2 \leq \|\mathbf{A}^{-1}\|_2 \|\delta\mathbf{A}\|_2 \|\mathbf{x} + \delta\mathbf{x}\|_2$$

即有

$$\frac{\|\delta \mathbf{x}\|_2}{\|\mathbf{x} + \delta \mathbf{x}\|_2} \leq (\|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2) \frac{\|\delta \mathbf{A}\|_2}{\|\mathbf{A}\|_2} \quad (5.1.7)$$

式 (5.1.5) 和式 (5.1.7) 表明, 解向量  $\mathbf{x}$  的相对误差与数值

$$\text{cond}(\mathbf{A}) = \|\mathbf{A}\|_2 \cdot \|\mathbf{A}^{-1}\|_2 \quad (5.1.8)$$

成正比。式中,  $\text{cond}(\mathbf{A})$  称为矩阵  $\mathbf{A}$  的条件数, 有时也用符号  $\kappa(\mathbf{A})$  表示。

当系数矩阵  $\mathbf{A}$  一个很小的扰动只引起解向量  $\mathbf{x}$  很小的扰动时, 就称矩阵  $\mathbf{A}$  是“良态”矩阵 (well-conditioned matrix)。若系数矩阵  $\mathbf{A}$  一个很小的扰动会引起解向量  $\mathbf{x}$  很大的扰动, 则称矩阵  $\mathbf{A}$  是“病态”矩阵 (ill-conditioned matrix)。条件数刻画了求解线性方程时, 误差经过矩阵  $\mathbf{A}$  的传播扩大为解向量的误差的程度, 因此是衡量线性方程数值稳定性的一个重要指标。

进一步地, 我们来分析误差在线性最小二乘问题中对解的影响。考虑超定的线性方程  $\mathbf{Ax} = \mathbf{b}$  的求解: 与前面的适定方程不同, 这里  $\mathbf{A}$  是一个  $m \times n$  矩阵, 且  $m > n$ 。由于方程个数多于未知参数个数, 这类方程统称超定方程。超定方程存在唯一的线性最小二乘解, 由

$$\mathbf{A}^H \mathbf{Ax} = \mathbf{A}^H \mathbf{b} \quad (5.1.9)$$

即  $\mathbf{x} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$  给出。容易证明 (详见后面的 5.2.2 节)

$$\text{cond}(\mathbf{A}^H \mathbf{A}) = [\text{cond}(\mathbf{A})]^2 \quad (5.1.10)$$

由式 (5.1.5) 和式 (5.1.7) 可知,  $\mathbf{b}$  的误差  $\delta \mathbf{b}$  和  $\mathbf{A}$  的误差  $\delta \mathbf{A}$  对超定方程式 (5.1.9) 的解  $\mathbf{x}$  的误差的影响分别与  $\mathbf{A}$  的条件数的平方成正比。也就是说, 超定方程 (5.1.9) 的条件数将呈平方关系增大。例如, 考虑

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ \delta & 0 \\ 0 & \delta \end{bmatrix}$$

的情况, 其中,  $\delta$  很小。 $\mathbf{A}$  的条件数为  $\delta^{-1}$  数量级。由于

$$\mathbf{B} = \mathbf{A}^H \mathbf{A} = \begin{bmatrix} 1 + \delta^2 & 1 \\ 1 & 1 + \delta^2 \end{bmatrix}$$

因而条件数变为  $\delta^{-2}$  数量级。

另外, 如果我们利用  $\mathbf{A}$  的 QR 分解  $\mathbf{A} = \mathbf{QR}$  来解超定方程  $\mathbf{Ax} = \mathbf{b}$  的话, 那么由于  $\mathbf{Q}^H \mathbf{Q} = \mathbf{I}$ , 故有

$$\text{cond}(\mathbf{Q}) = 1, \quad \text{cond}(\mathbf{A}) = \text{cond}(\mathbf{Q}^H \mathbf{A}) = \text{cond}(\mathbf{R}) \quad (5.1.11)$$

此时,  $\mathbf{b}$  和  $\mathbf{A}$  的误差的影响将分别如式 (5.1.5) 和式 (5.1.7) 所示, 与  $\mathbf{A}$  的条件数成正比。

以上事实告诉我们, 求解超定方程问题的 QR 分解方法具有比最小二乘方法更好的数值稳定性 (更小的条件数)。

若条件数“很大”，线性方程问题便称为（相对于范数  $\|\cdot\|_2$ ）病态的。此时，对于一接近真实  $\mathbf{b}$  的  $\mathbf{b}^*$ ，由于条件数很大，所以与  $\mathbf{b}^*$  对应的解就会远离对应于  $\mathbf{b}$  的解。解决这类病态问题的一种比 QR 分解更加有效的方法是总体最小二乘法（将在第 6 章介绍），它的基础就是 5.2 节要讨论的矩阵的奇异值分解。事实上，正如以后几节将看到的那样，矩阵的奇异值分解已被广泛应用于解决工程学科中的许多重要问题。

## 5.2 奇异值分解

奇异值分解（singular value decomposition, SVD）是现代数值分析（尤其是数值计算）的最基本和最重要的工具之一。本节介绍奇异值分解的定义、几何解释以及奇异值的性质。

### 5.2.1 奇异值分解及其解释

奇异值分解最早是 Beltrami 于 1873 年对实正方矩阵提出来的<sup>[38]</sup>。Beltrami 从双线性函数

$$f(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{A} \mathbf{y}, \quad \mathbf{A} \in \mathbb{R}^{n \times n}$$

出发，通过引入线性变换  $\mathbf{x} = \mathbf{U}\boldsymbol{\xi}$ ,  $\mathbf{y} = \mathbf{V}\boldsymbol{\eta}$ ，将双线性函数变为  $f(\mathbf{x}, \mathbf{y}) = \boldsymbol{\xi}^T \mathbf{S} \boldsymbol{\eta}$ ，其中

$$\mathbf{S} = \mathbf{U}^T \mathbf{A} \mathbf{V} \tag{5.2.1}$$

Beltrami 观测到，如果约束  $\mathbf{U}$  和  $\mathbf{V}$  为正交矩阵，则它们的选择各存在  $n^2 - n$  个自由度。他提出利用这些自由度使矩阵  $\mathbf{S}$  的对角线以外的元素全部为零，即矩阵  $\mathbf{S} = \boldsymbol{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$  为对角矩阵。于是，用  $\mathbf{U}$  和  $\mathbf{V}^T$  分别左乘和右乘式 (5.2.1)，并利用  $\mathbf{U}$  和  $\mathbf{V}$  的正交性，立即得到

$$\mathbf{A} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T \tag{5.2.2}$$

这就是 Beltrami 于 1873 年得到的实正方矩阵的奇异值分解<sup>[38]</sup>。1874 年，Jordan 也独立地推导出了实正方矩阵的奇异值分解<sup>[257]</sup>。有关奇异值分解的这段发明历史，可参见 MacDuffee 的书<sup>[326,p.78]</sup> 或 Stewart 的评述论文<sup>[465]</sup>。文献 [465] 还详细地评述了奇异值分解的整个早期历史。

后来，Autonne<sup>[25]</sup> 于 1902 年把奇异值分解推广到复正方矩阵；Eckart 与 Young<sup>[152]</sup> 于 1939 年又进一步把奇异值分解推广到一般的复长方形矩阵。因此，现在常将任意复长方矩阵的奇异值分解定理称为 Autonne-Eckart-Young 定理，详见下述。

**定理 5.2.1**（矩阵的奇异值分解）令  $\mathbf{A} \in \mathbb{R}^{m \times n}$ （或  $\mathbb{C}^{m \times n}$ ），则存在正交（或酉）矩阵  $\mathbf{U} \in \mathbb{R}^{m \times m}$ （或  $\mathbb{C}^{m \times m}$ ）和  $\mathbf{V} \in \mathbb{R}^{n \times n}$ （或  $\mathbb{C}^{n \times n}$ ）使得

$$\mathbf{A} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T \text{ (或 } \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^H \text{)} \tag{5.2.3}$$

式中  $\Sigma = \begin{bmatrix} \Sigma_1 & O \\ O & O \end{bmatrix}$ , 且  $\Sigma_1 = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$ , 其对角元素按照顺序

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0, \quad r = \text{rank}(\mathbf{A}) \quad (5.2.4)$$

排列。

以上定理最早是 Eckart 与 Young [152] 于 1939 年证明的, 但证明较繁杂, 而 Klema 与 Laub [271] 的证明则比较简单。

数值  $\sigma_1, \sigma_2, \dots, \sigma_r$  连同  $\sigma_{r+1} = \sigma_{r+2} = \dots = \sigma_n = 0$  一起称作矩阵  $\mathbf{A}$  的奇异值。

**定义 5.2.1** 矩阵  $\mathbf{A}_{m \times n}$  的奇异值  $\sigma_i$  称为单奇异值, 若  $\sigma_i \neq \sigma_j, \forall j \neq i$ 。

下面是关于奇异值和奇异值分解的几点解释和标记。

(1)  $n \times n$  矩阵  $\mathbf{V}$  为酉矩阵, 用  $\mathbf{V}$  右乘式 (5.2.3), 得  $\mathbf{AV} = \mathbf{U}\Sigma$ , 其列向量形式为

$$\mathbf{Av}_i = \begin{cases} \sigma_i \mathbf{u}_i, & i = 1, 2, \dots, r \\ 0, & i = r + 1, r + 2, \dots, n \end{cases} \quad (5.2.5)$$

因此,  $\mathbf{V}$  的列向量  $\mathbf{v}_i$  称为矩阵  $\mathbf{A}$  的右奇异向量 (right singular vector),  $\mathbf{V}$  称为  $\mathbf{A}$  的右奇异向量矩阵 (right singular vector matrix)。

(2)  $m \times m$  矩阵  $\mathbf{U}$  是酉矩阵, 用  $\mathbf{U}^H$  左乘式 (5.2.3), 得到  $\mathbf{U}^H \mathbf{A} = \Sigma \mathbf{V}$ , 其列向量形式为

$$\mathbf{u}_i^H \mathbf{A} = \begin{cases} \sigma_i \mathbf{v}_i^T, & i = 1, 2, \dots, r \\ 0, & i = r + 1, r + 2, \dots, n \end{cases} \quad (5.2.6)$$

因此,  $\mathbf{U}$  的列向量  $\mathbf{u}_i$  称为矩阵  $\mathbf{A}$  的左奇异向量 (left singular vector), 并称  $\mathbf{U}$  为  $\mathbf{A}$  的左奇异向量矩阵 (left singular vector matrix)。

(3) 矩阵  $\mathbf{A}$  的奇异值分解式 (5.2.3) 可以改写成向量表达形式

$$\mathbf{A} = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^H \quad (5.2.7)$$

这种表达有时称为  $\mathbf{A}$  的并向量 (奇异值) 分解 (dyadic decomposition) [198]。

(4) 当矩阵  $\mathbf{A}$  的秩  $r = \text{rank}(\mathbf{A}) < \min\{m, n\}$  时, 由于奇异值  $\sigma_{r+1} = \dots = \sigma_h = 0, h = \min\{m, n\}$ , 故奇异值分解式 (5.2.3) 可以简化为

$$\mathbf{A} = \mathbf{U}_r \Sigma_r \mathbf{V}_r^H \quad (5.2.8)$$

式中

$$\mathbf{U}_r = [\mathbf{u}_1, \dots, \mathbf{u}_r], \quad \mathbf{V}_r = [\mathbf{v}_1, \dots, \mathbf{v}_r], \quad \Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r)$$

式 (5.2.8) 称为矩阵  $\mathbf{A}$  的截尾奇异值分解 (truncated SVD) 或薄奇异值分解 (thin SVD)。与之形成对照, 式 (5.2.3) 则称为全奇异值分解 (full SVD)。

(5) 用  $\mathbf{u}_i^H$  左乘式 (5.2.5), 并注意到  $\mathbf{u}_i^H \mathbf{u}_i = 1$ , 易得

$$\mathbf{u}_i^H \mathbf{A} \mathbf{v}_i = \sigma_i, \quad i = 1, 2, \dots, \min\{m, n\} \quad (5.2.9)$$

或用矩阵形式写成

$$\mathbf{U}^H \mathbf{A} \mathbf{V} = \begin{bmatrix} \Sigma_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix}, \quad \Sigma_1 = \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n \end{bmatrix} \quad (5.2.10)$$

式 (5.2.3) 和式 (5.2.9) 是矩阵奇异值分解的两种定义方式。事实上, 式 (5.2.3) 很容易由式 (5.2.9) 导出。由于  $\mathbf{U}$  和  $\mathbf{V}$  分别是  $m \times m$  和  $n \times n$  酉矩阵, 满足  $\mathbf{U}\mathbf{U}^H = \mathbf{I}_m$  和  $\mathbf{V}\mathbf{V}^H = \mathbf{I}_n$ , 所以在式 (5.2.9) 两边左乘  $\mathbf{U}$  和右乘  $\mathbf{V}^H$  后, 立即得式 (5.2.3)。这也可以看作是定理 5.2.1 的另一种推导。

(6) 由式 (5.2.3) 易得

$$\mathbf{A}\mathbf{A}^H = \mathbf{U}\Sigma^2\mathbf{U}^H \quad (5.2.11)$$

这表明,  $m \times n$  矩阵  $\mathbf{A}$  的奇异值  $\sigma_i$  是矩阵乘积  $\mathbf{A}\mathbf{A}^H$  的特征值 (这些特征值是非负的) 的正平方根。

(7) 如果矩阵  $\mathbf{A}_{m \times n}$  具有秩  $r$ , 则:

- ①  $m \times m$  酉矩阵  $\mathbf{U}$  的前  $r$  列组成矩阵  $\mathbf{A}$  的列空间的标准正交基。
- ②  $n \times n$  酉矩阵  $\mathbf{V}$  的前  $r$  列组成矩阵  $\mathbf{A}$  的行空间或  $\mathbf{A}^H$  的列空间的标准正交基。
- ③  $\mathbf{V}$  的后  $n - r$  列组成矩阵  $\mathbf{A}$  的零空间的标准正交基。
- ④  $\mathbf{U}$  的后  $m - r$  列组成矩阵  $\mathbf{A}^H$  的零空间的标准正交基。

顾名思义, 矩阵  $\mathbf{A}$  的奇异值应该能够描述  $\mathbf{A}$  的奇异性质。下面的定理从数学上严格地叙述了这一事实。

**定理 5.2.2**<sup>[198]</sup> 令  $\mathbf{A} \in \mathbb{C}^{m \times n}$  ( $m > n$ ) 的奇异值为

$$\sigma_1 \geq \cdots \geq \sigma_n \geq 0$$

则

$$\sigma_k = \min_{\mathbf{E} \in \mathbb{C}^{m \times n}} \{\|\mathbf{E}\|_{\text{spec}} : \text{rank}(\mathbf{A} + \mathbf{E}) \leq k - 1\}, \quad k = 1, \dots, n \quad (5.2.12)$$

并且存在一满足  $\|\mathbf{E}_k\|_{\text{spec}} = \sigma_k$  的误差矩阵  $\mathbf{E}$  使得

$$\text{rank}(\mathbf{A} + \mathbf{E}_k) = k - 1, \quad k = 1, \dots, n$$

定理 5.2.2 表明, 如果原  $n \times n$  矩阵  $\mathbf{A}$  是正方的, 并且具有一个零奇异值, 则该矩阵的秩减小 1 的误差矩阵  $\mathbf{E}$  的谱范数等于零。这意味着, 误差矩阵必然是一个零矩阵。换句话说, 根据定理 5.2.2, 当原  $n \times n$  矩阵  $\mathbf{A}$  有一个零奇异值时, 该矩阵的秩  $\text{rank}(\mathbf{A}) \leq n - 1$ ,

即原矩阵  $A$  本来就不是满秩的。因此，如果一个正方矩阵具有零奇异值，则该矩阵必定是奇异矩阵。从这个角度讲，零奇异值刻画了矩阵  $A$  的奇异性质。一个正方矩阵只要有一个奇异值接近零，那么这个矩阵就接近于奇异矩阵。推而广之，一个非正方的矩阵如果有奇异值为零，则说明这个长方矩阵一定不是满列秩的或者满行秩的。这种情况称为矩阵的秩亏缺，它相对于矩阵的满秩亦是一种奇异现象。总之，无论是正方还是长方矩阵，零奇异值都刻画矩阵的奇异性质。这就是矩阵奇异值的内在含义。

对于矩阵方程式 (5.1.2)，可以把

$$\tilde{x} = V^H x \quad \text{或} \quad x = V \tilde{x} \quad (5.2.13)$$

看作是利用  $V$  进行的一种正交变换（也可认为是一种旋转），将  $x$  的各点旋转为  $\tilde{x}$  的各点。同样地，也可以利用  $U^H$  对  $b$  作正交变换

$$\tilde{b} = U^H b \quad (5.2.14)$$

即将  $b$  的各点旋转一定角度后变为  $\tilde{b}$  上的各点。现在，将奇异值分解式 (5.2.3) 代入方程式  $Ax = b$ ，并利用式 (5.2.13) 和式 (5.2.14)，可得到

$$\tilde{b} = \Sigma \tilde{x} \longrightarrow \tilde{x} = \Sigma^\dagger \tilde{b}$$

于是，线性方程式 (5.1.2) 的求解过程可以解释为一系列的线性变换操作，即

$$b \xrightarrow{U^H} U^H b = \tilde{b} \xrightarrow{\Sigma} \Sigma^\dagger \tilde{b} = \tilde{x} \xrightarrow{V} V \tilde{x} = x$$

注意， $\Sigma$  的广义逆矩阵  $\Sigma^\dagger$  可直接计算为

$$\Sigma^\dagger = \begin{bmatrix} \Sigma^{-1} & O \\ O & O \end{bmatrix} \quad (5.2.15)$$

其中

$$\Sigma^{-1} = \text{diag}(1/\sigma_1, 1/\sigma_2, \dots, 1/\sigma_r) \quad (5.2.16)$$

把  $m \times n$  矩阵  $A$  视作从  $n$  维（复数）向量空间  $\mathbb{C}^n$  到  $m$  维（复数）向量空间  $\mathbb{C}^m$  的线性映射有时是很方便的。此时，关于奇异值分解的唯一性，有以下结果<sup>[543]</sup>：

- (1) 非零奇异值的个数  $r$  和它们的值  $\sigma_1, \sigma_2, \dots, \sigma_r$  相对于矩阵  $A$  是唯一确定的。
- (2) 若  $\text{rank}(A) = r$ ，则满足  $Ax = \mathbf{0}$  的  $x (\in \mathbb{C}^n)$  的集合即  $A$  的零空间  $\text{Null } A (\subseteq \mathbb{C}^n)$  是  $n - r$  维的，因此可选择正交基  $\{v_{r+1}, v_{r+2}, \dots, v_n\}$  作为  $A$  在  $\mathbb{C}^n$  内的零空间。从这个意义上讲， $V$  的列向量张成的  $\mathbb{C}^n$  的子空间  $\text{Null}(A)$  是唯一确定的，但是各个向量只要能组成该子空间的正交基，它们就可以自由地选择。
- (3) 可以表示成  $y = Ax$  的  $y (\in \mathbb{C}^m)$  的集合组成  $A$  的像空间  $\text{Im } A$ ，它是  $r$  维的，而  $\text{Im } A$  的正交补空间  $(\text{Im } A)^\perp$  是  $m - r$  维的，因此可选择  $\{u_{r+1}, u_{r+2}, \dots, u_m\}$  作为  $\text{Im } A$  在  $\mathbb{C}^m$  内的正交补空间内的正交基。由  $U$  的列向量  $u_{r+1}, u_{r+2}, \dots, u_m$  张成的  $\mathbb{C}^m$  的子空间  $(\text{Im } A)^\perp$  是唯一确定的。

(4) 若  $\sigma_i$  是单奇异值 (即  $\sigma_i \neq \sigma_j, \forall j \neq i$ ), 则  $v_i$  和  $u_i$  除相差一相角 ( $A$  为实数矩阵时, 相差一符号) 外是唯一确定的。也就是说,  $v_i$  和  $u_i$  同时乘以  $e^{j\theta}$  ( $j = \sqrt{-1}$ , 且  $\theta$  为实数) 后, 它们仍然分别是矩阵  $A$  的右和左奇异向量。

### 5.2.2 奇异值的性质

矩阵的各种变形与奇异值的变化有以下关系:

(1)  $m \times n$  矩阵  $A$  的共轭转置  $A^H$  的奇异值分解为

$$A^H = V \Sigma^T U^H \quad (5.2.17)$$

即矩阵  $A$  和  $A^H$  具有完全相同的奇异值。

(2)  $A^H A, AA^H$  的奇异值分解分别为

$$A^H A = V \Sigma^T \Sigma V^H, \quad AA^H = U \Sigma^T \Sigma U^H \quad (5.2.18)$$

其中

$$\Sigma^T \Sigma = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_r^2, \overbrace{0, \dots, 0}^{n-r}) \quad (5.2.19)$$

$$\Sigma \Sigma^T = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_r^2, \overbrace{0, \dots, 0}^{m-r}) \quad (5.2.20)$$

(3)  $P$  和  $Q$  分别为  $m \times m$  和  $n \times n$  酉矩阵时,  $PAQ^H$  的奇异值分解由

$$PAQ^H = \tilde{U} \Sigma \tilde{V}^H \quad (5.2.21)$$

给出, 其中,  $\tilde{U} = PU$ ,  $\tilde{V} = QV$ 。也就是说, 矩阵  $PAQ^H$  与  $A$  具有相同的奇异值, 即奇异值具有酉不变性, 但奇异向量不同。

(4)  $m \times n$  矩阵  $A$  的奇异值分解与  $n \times m$  维 Moore-Penrose 广义逆矩阵  $A^\dagger$  之间存在下列关系

$$A^\dagger = V \Sigma^\dagger U^H \quad (5.2.22)$$

其中,  $\Sigma^\dagger$  由式 (5.2.15) 给定。

虽然  $U$  和  $V$  相对于  $A$  不是唯一确定的, 但广义逆矩阵  $A^\dagger$  是唯一确定的。特别地, 若  $A$  是一个正方的非奇异矩阵, 则  $A^\dagger = A^{-1}$ 。因此, 在这一情况下, 如果  $A$  的奇异值是  $\sigma_1, \dots, \sigma_n$ , 那么  $A^{-1}$  的奇异值就是  $1/\sigma_1, \dots, 1/\sigma_n$ 。

关于矩阵和它的子矩阵的奇异值之间的关系, 有下面的定理, 常被称为奇异值交织定理 (interlacing theorem for singular values)。

**定理 5.2.3**<sup>[239, 248]</sup> 令  $A$  是一个  $m \times n$  矩阵, 其奇异值  $\sigma_1 \geq \dots \geq \sigma_r$ , 其中,  $r =$

$\min\{m, n\}$ 。若  $p \times q$  矩阵  $B$  是  $A$  的子矩阵，其奇异值  $\gamma_1 \geq \cdots \geq \gamma_{\min\{p, q\}}$ ，则

$$\sigma_i \geq \gamma_i, \quad i = 1, \dots, \min\{p, q\} \quad (5.2.23)$$

并且

$$\gamma_i \geq \sigma_{i+(m-p)+(n-q)}, \quad i \leq \min\{p+q-m, p+q-n\} \quad (5.2.24)$$

矩阵的奇异值与矩阵的范数、行列式、条件数、特征值等有着密切的关系。

### 1. 奇异值与范数的关系

矩阵  $A$  的谱范数等于  $A$  的最大奇异值，即

$$\|A\|_{\text{spec}} = \sigma_1 \quad (5.2.25)$$

注意到矩阵  $A$  的 Frobenius 范数  $\|A\|_F$  是酉不变的，即  $\|U^H A V\|_F = \|A\|_F$ ，故有

$$\|A\|_F = \left[ \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right]^{1/2} = \|U^H A V\|_F = \|\Sigma\|_F = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_r^2} \quad (5.2.26)$$

即是说，任何一个矩阵的 Frobenius 范数等于该矩阵所有非零奇异值平方和的正平方根。

### 2. 奇异值与行列式的关系

设  $A$  是  $n \times n$  正方矩阵。由于酉矩阵的行列式之绝对值等于 1，所以由定理 5.2.1 有

$$|\det(A)| = |\det \Sigma| = \sigma_1 \sigma_2 \cdots \sigma_n \quad (5.2.27)$$

若所有  $\sigma_i$  都不等于零，则  $|\det(A)| \neq 0$ ，这表明  $A$  是非奇异的。若至少有一个  $\sigma_i (i > r)$  等于零，则  $\det(A) = 0$ ，即  $A$  奇异。这就是之所以把全部  $\sigma_i$  值统称为奇异值的原因。

### 3. 奇异值与条件数的关系

对于一个  $m \times n$  矩阵  $A$ ，其条件数也可以利用奇异值定义为

$$\text{cond}(A) = \sigma_1 / \sigma_p, \quad p = \min\{m, n\} \quad (5.2.28)$$

由定义式 (5.2.28) 可以看出，条件数是一个大于或等于 1 的正数，因为  $\sigma_1 \geq \sigma_p$ 。显然，由于至少有一个奇异值  $\sigma_p = 0$ ，故奇异矩阵的条件数为无穷大，而条件数虽然不是无穷大，但很大时，就称  $A$  是接近奇异的。这意味着，当条件数很大时， $A$  的行向量或列向量的线性相关性很强。另由定义式 (5.1.8) 易知，正交或酉矩阵  $V$  的条件数等于 1。从这个意义上讲，正交或酉矩阵是“理想条件”的。式 (5.2.28) 也可用作条件数  $\text{cond}(A)$  的评价。

考虑超定方程  $Ax = b$ 。此时，由于  $A^H A$  的奇异值分解为

$$A^H A = V \Sigma^2 V^H \quad (5.2.29)$$

即矩阵  $\mathbf{A}^H \mathbf{A}$  的最大和最小奇异值分别是矩阵  $\mathbf{A}$  的最大和最小奇异值的平方，故

$$\text{cond}(\mathbf{A}^H \mathbf{A}) = \frac{\sigma_1^2}{\sigma_n^2} = [\text{cond}(\mathbf{A})]^2 \quad (5.2.30)$$

换言之，矩阵  $\mathbf{A}^H \mathbf{A}$  的条件数是矩阵  $\mathbf{A}$  的条件数的平方倍。

#### 4. 奇异值与特征值的关系

设  $n \times n$  正方对称矩阵  $\mathbf{A}$  的特征值为  $\lambda_1, \dots, \lambda_n$  ( $|\lambda_1| \geq \dots \geq |\lambda_n|$ )，奇异值为  $\sigma_1, \dots, \sigma_n$  ( $\sigma_1 \geq \dots \geq \sigma_n \geq 0$ )，则  $\sigma_i \geq |\lambda_i| \geq \sigma_n$  ( $i = 1, \dots, n$ )， $\text{cond}(\mathbf{A}) \geq |\lambda_1|/|\lambda_n|$ 。

下面是奇异值的性质汇总。

##### 1. 奇异值服从的等式关系 [324]

- (1) 矩阵  $\mathbf{A}_{m \times n}$  和其 Hermitian 矩阵  $\mathbf{A}^H$  具有相同的奇异值。
- (2) 矩阵  $\mathbf{A}_{m \times n}$  的非零奇异值是  $\mathbf{A}\mathbf{A}^H$  或者  $\mathbf{A}^H\mathbf{A}$  的非零特征值的正平方根。
- (3)  $\sigma > 0$  是矩阵  $\mathbf{A}_{m \times n}$  的单奇异值，当且仅当  $\sigma^2$  是  $\mathbf{A}\mathbf{A}^H$  或  $\mathbf{A}^H\mathbf{A}$  的单特征值。
- (4) 若  $p = \min\{m, n\}$ ，且  $\sigma_1, \dots, \sigma_p$  是矩阵  $\mathbf{A}_{m \times n}$  的奇异值，则

$$\text{tr}(\mathbf{A}^H \mathbf{A}) = \sum_{i=1}^p \sigma_i^2$$

- (5) 矩阵行列式的绝对值等于矩阵奇异值之乘积，即  $|\det(\mathbf{A})| = \sigma_1 \cdots \sigma_n$ 。
- (6) 矩阵  $\mathbf{A}$  的谱范数等于  $\mathbf{A}$  的最大奇异值，即  $\|\mathbf{A}\|_{\text{spec}} = \sigma_{\max}$ 。
- (7) 若  $m \geq n$ ，则对于矩阵  $\mathbf{A}_{m \times n}$ ，有

$$\begin{aligned} \sigma_{\min}(\mathbf{A}) &= \min \left\{ \left( \frac{\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{x}} \right)^{1/2} : \mathbf{x} \neq \mathbf{0} \right\} \\ &= \min \left\{ (\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x})^{1/2} : \mathbf{x}^H \mathbf{x} = 1, \mathbf{x} \in \mathbb{C}^n \right\} \end{aligned}$$

- (8) 若  $m \geq n$ ，则对于矩阵  $\mathbf{A}_{m \times n}$ ，有

$$\begin{aligned} \sigma_{\max}(\mathbf{A}) &= \max \left\{ \left( \frac{\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{x}} \right)^{1/2} : \mathbf{x} \neq \mathbf{0} \right\} \\ &= \max \left\{ (\mathbf{x}^H \mathbf{A}^H \mathbf{A} \mathbf{x})^{1/2} : \mathbf{x}^H \mathbf{x} = 1, \mathbf{x} \in \mathbb{C}^n \right\} \end{aligned}$$

- (9) 若  $m \times m$  矩阵  $\mathbf{A}$  非奇异，则

$$\frac{1}{\sigma_{\min}(\mathbf{A})} = \max \left\{ \left( \frac{\mathbf{x}^H (\mathbf{A}^{-1})^H \mathbf{A}^{-1} \mathbf{x}}{\mathbf{x}^H \mathbf{x}} \right)^{1/2} : \mathbf{x} \neq \mathbf{0}, \mathbf{x} \in \mathbb{C}^n \right\}$$

- (10) 若  $\sigma_1, \dots, \sigma_p$  是矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$  的非零奇异值 (其中， $p = \min\{m, n\}$ )，则矩阵  $\begin{bmatrix} \mathbf{O} & \mathbf{A} \\ \mathbf{A}^H & \mathbf{O} \end{bmatrix}$  有  $2p$  个非零奇异值  $\sigma_1, \dots, \sigma_p, -\sigma_1, \dots, -\sigma_p$  及  $|m - n|$  个零奇异值。

(11) 若  $\mathbf{A} = \mathbf{U} \begin{bmatrix} \Sigma_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \mathbf{V}^H$  是  $m \times n$  矩阵  $\mathbf{A}$  的奇异值分解, 则  $\mathbf{A}$  的 Moore-Penrose 逆矩阵

$$\mathbf{A}^\dagger = \mathbf{V} \begin{bmatrix} \Sigma_1^{-1} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \mathbf{U}^H$$

## 2. 奇异值服从的不等式关系 [238, 239, 294, 101, 324]

(1) 若  $\mathbf{A}$  和  $\mathbf{B}$  是  $m \times n$  矩阵, 则对于 ( $p = \min\{m, n\}$ ), 有

$$\sigma_{i+j-1}(\mathbf{A} + \mathbf{B}) \leq \sigma_i(\mathbf{A}) + \sigma_j(\mathbf{B}), \quad 1 \leq i, j \leq p, \quad i + j \leq p + 1$$

特别地, 当  $j = 1$  时,  $\sigma_i(\mathbf{A} + \mathbf{B}) \leq \sigma_i(\mathbf{A}) + \sigma_1(\mathbf{B})$  对  $i = 1, \dots, p$  成立。

(2) 对矩阵  $\mathbf{A}_{m \times n}, \mathbf{B}_{m \times n}$ , 有  $\sigma_{\max}(\mathbf{A} + \mathbf{B}) \leq \sigma_{\max}(\mathbf{A}) + \sigma_{\max}(\mathbf{B})$ 。

(3) 若  $\mathbf{A}$  和  $\mathbf{B}$  是  $m \times n$  矩阵, 则

$$\sum_{j=1}^p [\sigma_j(\mathbf{A} + \mathbf{B}) - \sigma_j(\mathbf{A})]^2 \leq \|\mathbf{B}\|_F^2, \quad p = \min\{m, n\}$$

(4) 若  $\mathbf{A}_{m \times m} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m]$  的奇异值  $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \dots \geq \sigma_m(\mathbf{A})$ , 则

$$\sum_{j=1}^k [\sigma_{m-k+j}(\mathbf{A})]^2 \leq \sum_{j=1}^k \mathbf{a}_j^H \mathbf{a}_j \leq \sum_{j=1}^k [\sigma_j(\mathbf{A})]^2, \quad k = 1, 2, \dots, m$$

(5) 若  $p = \min\{m, n\}$ , 且  $\mathbf{A}_{m \times n}$  和  $\mathbf{B}_{m \times n}$  的奇异值排列为  $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \dots \geq \sigma_p(\mathbf{A})$ ,  $\sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{B}) \geq \dots \geq \sigma_p(\mathbf{B})$  和  $\sigma_1(\mathbf{A} + \mathbf{B}) \geq \sigma_2(\mathbf{A} + \mathbf{B}) \geq \dots \geq \sigma_p(\mathbf{A} + \mathbf{B})$ , 则

$$\sigma_{i+j-1}(\mathbf{A} \mathbf{B}^H) \leq \sigma_i(\mathbf{A}) \sigma_j(\mathbf{B}), \quad 1 \leq i, j \leq p, \quad i + j \leq p + 1$$

(6) 设  $m \times (n - 1)$  矩阵  $\mathbf{B}$  是删去  $m \times n$  矩阵  $\mathbf{A}$  任意一列得到的矩阵, 并且它们的奇异值都按照非降顺序排列, 则

$$\sigma_1(\mathbf{A}) \geq \sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{A}) \geq \sigma_2(\mathbf{B}) \geq \dots \geq \sigma_h(\mathbf{A}) \geq \sigma_h(\mathbf{B}) \geq 0$$

式中,  $h = \min\{m, n - 1\}$ 。

(7) 矩阵  $\mathbf{A}_{m \times n}$  的最大奇异值满足不等式

$$\sigma_{\max}(\mathbf{A}) \geq \left[ \frac{1}{n} \text{tr}(\mathbf{A}^H \mathbf{A}) \right]^{1/2}$$

(8) 设  $(m - 1) \times n$  矩阵  $\mathbf{B}$  是删去  $m \times n$  矩阵  $\mathbf{A}$  任意一行得到的矩阵, 并且它们的奇异值都按照非降顺序排列, 则

$$\sigma_1(\mathbf{A}) \geq \sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{A}) \geq \sigma_2(\mathbf{B}) \geq \dots \geq \sigma_h(\mathbf{A}) \geq \sigma_h(\mathbf{B}) \geq 0$$

式中,  $h = \min\{m, n - 1\}$ 。

### 5.2.3 秩亏缺最小二乘解

在奇异值分析的应用中，常常需要用一个低秩的矩阵逼近一个含噪声或扰动的矩阵。下面的定理给出了逼近质量的评价。

**定理 5.2.4** 令  $A \in \mathbb{R}^{m \times n}$  的奇异值分解由  $A = \sum_{i=1}^p \sigma_i u_i v_i^T$  给出，其中  $p = \text{rank}(A)$ 。

若  $k < p$ ，并且  $A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$ ，则逼近质量可分别使用谱范数和 Frobenius 范数度量

$$\min_{\text{rank}(\mathbf{B})=k} \|A - \mathbf{B}\|_{\text{spec}} = \|A - A_k\|_{\text{spec}} = \sigma_{k+1} \quad (5.2.31)$$

$$\min_{\text{rank}(\mathbf{B})=k} \|A - \mathbf{B}\|_F = \|A - A_k\|_F = \sqrt{\sum_{i=k+1}^q \sigma_i^2} \quad (5.2.32)$$

式中， $q = \min\{m, n\}$ 。

证明 详见文献 [151, 347, 248]。

在信号处理和系统理论中，最常见的线性方程组  $Ax = b$  是超定的和非满秩即秩亏缺的，也就是说，矩阵  $A \in \mathbb{C}^{m \times n}$  的行数  $m$  比列数  $n$  大，且  $r = \text{rank}(A) < n$ 。令  $A$  的奇异值分解由式  $A = U \Sigma V^H$  给出，其中， $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0)$ 。考察

$$G = V \Sigma^\dagger U^H \quad (5.2.33)$$

式中， $\Sigma^\dagger = \text{diag}(1/\sigma_1, \dots, 1/\sigma_r, 0, \dots, 0)$ 。由奇异值的性质 (4) 知， $G$  是  $A$  的 Moore-Penrose 广义逆矩阵。因此

$$\hat{x} = Gb = V \Sigma^\dagger U^H b \quad (5.2.34)$$

或表示为

$$x_{\text{LS}} = \sum_{i=1}^r (\mathbf{u}_i^H b / \sigma_i) \mathbf{v}_i$$

它是最小二乘问题

$$\min \|Ax - b\|_2 \quad (5.2.35)$$

的最小范数解，相应的最小残差为

$$\rho_{\text{LS}} = \|Ax_{\text{LS}} - b\|_2 = \|[\mathbf{u}_{r+1}, \dots, \mathbf{u}_m]^H b\|_2 \quad (5.2.36)$$

应用奇异值分解求解最小二乘问题的方法常简称为奇异值分解方法。虽然在理论上，当  $i > r$  时奇异值  $\sigma_i = 0$ ，但是计算出来的奇异值  $\hat{\sigma}_i$ ， $i > r$  并不会等于零，有时甚至表现出比较大的扰动。因此，需要有计算秩  $r$  的估计值  $\hat{r}$  的方法。在信号处理和系统理论中，常将该估计值称为“有效秩”。

有效秩确定有以下两种常用方法。

#### 1. 归一化奇异值方法

计算归一化奇异值

$$\bar{\sigma}_i = \frac{\hat{\sigma}_i}{\hat{\sigma}_1} \quad (5.2.37)$$

选择满足准则

$$\bar{\sigma}_i \geq \epsilon \quad (5.2.38)$$

的最大整数作为有效秩的估计值  $\hat{r}$ 。显然，这一准则等价于选择满足

$$\hat{\sigma}_i \geq \epsilon \cdot \hat{\sigma}_1 \quad (5.2.39)$$

的最大整数  $\hat{r}$ 。式中， $\epsilon$  是某个很小的正数，它根据计算机精度与（或）数据精度选取。例如，选取  $\epsilon = 0.1$  或者  $\epsilon = 0.05$  等。

## 2. 范数比方法

令  $m \times n$  矩阵  $A_k$  是原  $m \times n$  矩阵  $A$  的秩  $k$  近似，定义该近似矩阵与原矩阵的 Frobenius 范数比为

$$\nu(k) = \frac{\|A_k\|_F}{\|A\|_F} = \frac{\sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_k^2}}{\sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_h^2}}, \quad h = \min\{m, n\} \quad (5.2.40)$$

并选择满足

$$\nu(k) \geq \alpha \quad (5.2.41)$$

的最大整数作为有效秩估计  $\hat{r}$ ，其中， $\alpha$  是接近于 1 的阈值，例如  $\alpha = 0.997$  等。

采用以上两种准则确定出有效秩  $\hat{r}$  后，可将

$$\hat{x}_{LS} = \sum_{i=1}^{\hat{r}} (\hat{u}_i^H b / \hat{\sigma}_i) \hat{v}_i \quad (5.2.42)$$

看作是真实最小二乘解  $x_{LS}$  的一个合理近似。显而易见，这种解就是方程组  $A_{\hat{r}}x = b$  的最小二乘解，其中

$$A_{\hat{r}} = \sum_{i=1}^{\hat{r}} \sigma_i u_i v_i^H \quad (5.2.43)$$

在最小二乘问题中，用  $A_{\hat{r}}$  代替  $A$  相当于过滤掉小的奇异值。当  $A$  是从有噪声的观测数据得到时，这种过滤能够起很大的作用。容易观察出，式 (5.2.42) 给出的最小二乘解  $\hat{x}_{LS}$  仍然包含了  $n$  个参数。然而，由于线性方程  $Ax = b$  秩亏缺意味着  $x$  中只有  $r$  个参数是独立的，其他参数是这  $r$  个独立参数的重复作用或线性相关的结果。在许多应用中，当然希望能够求出这  $r$  个线性无关的参数，而不是包含了冗余因素的  $n$  个参数。换言之，我们的目的是只估计主要因素，并剔除掉次要因素。这一问题可以借助低秩总体最小二乘方法，将在后面章节讨论。

### 5.3 乘积奇异值分解

5.2节介绍了一般矩阵的奇异值分解。从本节开始，将依次讨论几种特殊情况下矩阵的奇异值分解，它们分别是乘积奇异值分解、广义奇异值分解和结构奇异值分解。本节介绍乘积奇异值分解的有关理论和实现算法。

#### 5.3.1 乘积奇异值分解问题

所谓乘积奇异值分解 (product singular value decomposition, PSVD)，顾名思义就是两个矩阵乘积  $\mathbf{B}^T \mathbf{C}$  的奇异值分解。考虑矩阵乘积

$$\mathbf{A} = \mathbf{B}^T \mathbf{C}, \quad \mathbf{B} \in \mathbb{R}^{p \times m}, \quad \mathbf{C} \in \mathbb{R}^{p \times n}, \quad \text{rank}(\mathbf{B}) = \text{rank}(\mathbf{C}) = p \quad (5.3.1)$$

从原理上讲，乘积奇异值分解等价于直接对矩阵的乘积进行普通的奇异值分解。然而，事先直接计算矩阵的乘积，再计算矩阵乘积的奇异值分解往往会让小的奇异值产生大的扰动。为了说明这一点，请看一个例子。

**例 5.3.1**<sup>[145]</sup> 令

$$\mathbf{B}^T = \begin{bmatrix} 1 & \xi \\ -1 & \xi \end{bmatrix}, \quad \mathbf{C} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{B}^T \mathbf{C} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1-\xi & 1+\xi \\ -1-\xi & -1+\xi \end{bmatrix} \quad (5.3.2)$$

显然， $\mathbf{C}$  是一个正交矩阵，而  $\mathbf{B}^T$  的两列  $[1, -1]^T$  和  $[\xi, \xi]^T$  相互正交。矩阵乘积  $\mathbf{B}^T \mathbf{C}$  的真实奇异值为  $\sigma_1 = \sqrt{2}$  和  $\sigma_2 = \sqrt{2}|\xi|$ 。然而，若  $|\xi|$  小于截止误差  $\varepsilon$ ，式 (5.3.2) 的浮点计算结果为  $\mathbf{B}^T \mathbf{C} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$ ，其奇异值为  $\sigma_1 = \sqrt{2}$  和  $\sigma_2 = 0$ 。若  $|\xi| > 1/\varepsilon$ ，则浮点运算得到的矩阵乘积  $\mathbf{B}^T \mathbf{C} = \frac{1}{\sqrt{2}} \begin{bmatrix} -\xi & \xi \\ -\xi & \xi \end{bmatrix}$ ，其奇异值为  $\sigma_1 = 0$  和  $\sigma_2 = \sqrt{2}|\xi|$ 。因此，矩阵乘积  $\mathbf{B}^T \mathbf{C}$  的两个实际的奇异值  $\sigma_1 = \sqrt{2}$  和  $\sigma_2 = \sqrt{2}|\xi|$  在经过浮点算法计算后，最小的奇异值被扰动为 0，与实际的奇异值相差明显。Laub 等人<sup>[304]</sup> 指出，当线性系统接近不可控和不可观测时，小奇异值的精确计算显得十分重要，因为如果一个非零的小奇异值被计算为零值，则会导致错误的结论，将一个最小系统判断为非最小系统。

上述例子说明，直接对两个矩阵的乘积  $\mathbf{B}^T \mathbf{C}$  进行奇异值分解在数值上是不可取的。因此，有必要考虑一个更加困难的问题：能否使得计算式 (5.3.1) 中  $\mathbf{A} = \mathbf{B}^T \mathbf{C}$  的奇异值分解尽可能与给定的  $\mathbf{B}$  和  $\mathbf{C}$  具有接近的精度？这就是所谓的(矩阵)乘积奇异值分解问题。

乘积奇异值分解是由 Fernando 与 Hammarling 于 1988 年首先提出来的<sup>[163]</sup>，它可以用下面的定理来表述。

**定理 5.3.1 (乘积奇异值分解)**<sup>[163]</sup> 令  $\mathbf{B}^T \in \mathbb{C}^{m \times p}$ ,  $\mathbf{C} \in \mathbb{C}^{p \times n}$ ，则存在酉矩阵  $\mathbf{U} \in \mathbb{C}^{m \times m}$ ,  $\mathbf{V} \in \mathbb{C}^{n \times n}$  和非奇异矩阵  $\mathbf{Q} \in \mathbb{C}^{p \times p}$  使得

$$\mathbf{U} \mathbf{B}^H \mathbf{Q} = \begin{bmatrix} \mathbf{I} & & \\ & \mathbf{O}_B & \\ & & \Sigma_B \end{bmatrix}, \quad \mathbf{Q}^{-1} \mathbf{C} \mathbf{V}^H = \begin{bmatrix} \mathbf{O}_C & & \\ & \mathbf{I} & \\ & & \Sigma_C \end{bmatrix} \quad (5.3.3)$$

式中

$$\begin{aligned}\boldsymbol{\Sigma}_B &= \text{diag}(s_1, s_2, \dots, s_r), & 1 > s_1 \geq \dots \geq s_r > 0 \\ \boldsymbol{\Sigma}_C &= \text{diag}(t_1, t_2, \dots, t_r), & 1 > t_1 \geq \dots \geq t_r > 0\end{aligned}$$

且

$$s_i^2 + t_i^2 = 1, \quad i = 1, 2, \dots, r$$

有关本定理的证明, 可参见文献 [163]。根据定理 5.3.1, 不难验证

$$\mathbf{U}\mathbf{B}^H\mathbf{C}\mathbf{V}^H = \text{diag}(\mathbf{O}_C, \mathbf{O}_B, \boldsymbol{\Sigma}_B \boldsymbol{\Sigma}_C)$$

因此, 矩阵乘积  $\mathbf{B}^H\mathbf{C}$  的奇异值由零奇异值和非零奇异值两部分组成, 其非零奇异值由  $s_i t_i, i = 1, 2, \dots, r$  给出。

### 5.3.2 乘积奇异值分解的精确计算

Drmac 于 1998 年提出了乘积奇异值分解的精确计算算法<sup>[145]</sup>, 其基本思路如下: 任何一个矩阵  $\mathbf{A}$  与正交矩阵相乘, 其奇异值保持不变。因此, 若令

$$\mathbf{B}' = \mathbf{T}\mathbf{B}\mathbf{U}, \quad \mathbf{C}' = (\mathbf{T}^T)^{-1}\mathbf{C}\mathbf{V} \quad (5.3.4)$$

其中,  $\mathbf{T}$  非奇异,  $\mathbf{U}, \mathbf{V}$  为正交矩阵, 则  $\mathbf{B}'^T\mathbf{C}' = \mathbf{U}^T\mathbf{B}^T\mathbf{C}\mathbf{V}$  与  $\mathbf{B}^T\mathbf{C}$  具有完全相同的奇异值 (包括零奇异值在内), 并且很容易由  $\mathbf{B}'^T\mathbf{C}'$  的奇异值分解得到  $\mathbf{B}^T\mathbf{C}$  的奇异值分解, 因为

$$\mathbf{B}'^T\mathbf{C}' = \mathbf{U}^T\mathbf{B}^T\mathbf{T}^T(\mathbf{T}^T)^{-1}\mathbf{C}\mathbf{V} = \mathbf{U}^T(\mathbf{B}^T\mathbf{C})\mathbf{V}$$

给定矩阵  $\mathbf{B} \in \mathbb{R}^{p \times m}, \mathbf{C} \in \mathbb{R}^{p \times n}, p \leq \min\{m, n\}$ , 并记矩阵  $\mathbf{B}$  的行向量为  $\mathbf{b}_i^T, i = 1, 2, \dots, p$ 。Drmac 的乘积奇异值分解算法如下。

#### 算法 5.3.1 乘积奇异值分解 PSVD(B, C)<sup>[145]</sup>

步骤 1 计算  $\mathbf{B}_r = \text{diag}(\|\mathbf{b}_1^T\|_2, \|\mathbf{b}_2^T\|_2, \dots, \|\mathbf{b}_r^T\|_2)$ , 令  $\mathbf{B}_1 = \mathbf{B}_r^\dagger \mathbf{B}, \mathbf{C}_1 = \mathbf{B}_r \mathbf{C}$ 。

步骤 2 计算  $\mathbf{C}_1^T$  的 QR 分解, 即

$$\mathbf{C}_1^T \mathbf{\Pi} = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{O}_{(n-r) \times p} \end{bmatrix}$$

其中,  $\mathbf{R} \in \mathbb{R}^{r \times p}, \text{rank}(\mathbf{R}) = r$ ;  $\mathbf{Q}$  为正交矩阵。

步骤 3 利用标准矩阵乘法计算矩阵  $\mathbf{F} = \mathbf{B}_1^T \mathbf{\Pi} \mathbf{R}^T$ 。

步骤 4 计算矩阵  $\mathbf{F}$  的 QR 分解 (最好使用列旋转的 Householder QR 分解算法)

$$\mathbf{F} \mathbf{\Pi}_F = \mathbf{Q}_F \begin{bmatrix} \mathbf{R}_F \\ \mathbf{O} \end{bmatrix}$$

步骤 5 对转置矩阵  $\mathbf{R}_F^T$  应用奇异值分解的右边 Jacobi 算法 (算法 5.3.1), 计算  $\mathbf{R}_F$  的奇异值分解  $\boldsymbol{\Sigma} = \mathbf{V}^T \mathbf{R}_F \mathbf{W}$ 。

输出 矩阵乘积  $\mathbf{B}^T \mathbf{C}$  的奇异值分解结果为

$$\begin{bmatrix} \Sigma \oplus O \\ O \end{bmatrix} = \begin{bmatrix} V^T & \\ & I \end{bmatrix} Q_F^T (\mathbf{B}^T \mathbf{C}) [Q(W \oplus I_{n-p})]$$

式中,  $A \oplus D$  表示矩阵  $A$  与  $D$  的直和。

在上述算法中, 对角矩阵  $D = \text{diag}(d_1, d_2, \dots, d_p)$  的广义逆矩阵  $D^\dagger$  仍然为对角矩阵, 其对角元素为  $1/d_i$  ( $d_i \neq 0$ ) 或  $0$  ( $d_i = 0$ )。

计算矩阵乘积  $\mathbf{B}^T \mathbf{C}$  的奇异值分解的上述算法已被推广到三个矩阵乘积的奇异值分解的精确计算<sup>[147]</sup>。令

$$\mathbf{A} = \mathbf{B}^T \mathbf{S} \mathbf{C} \quad (5.3.5)$$

式中,  $\mathbf{B} \in \mathbb{R}^{p \times m}$ ,  $\mathbf{S} \in \mathbb{R}^{p \times q}$ ,  $\mathbf{C} \in \mathbb{R}^{q \times n}$ ,  $p \leq m$ ,  $q \leq n$ 。

满足正则条件

$$\text{rank}(\mathbf{B}) = p, \text{rank}(\mathbf{C}) = q, \text{rank}(\mathbf{S}) = \rho = \min\{p, q\} \quad (5.3.6)$$

的三个矩阵  $(\mathbf{B}, \mathbf{S}, \mathbf{C})$  称为正则矩阵三元组 (regular matrix triplet)<sup>[147]</sup>。在这种情况下, 矩阵  $\mathbf{A}$  将有  $\min\{m, n\} - \rho = \min\{m, n\} - \min\{p, q\}$  个确定的零奇异值。现在的问题是, 用尽可能高的相对精度计算其他非零奇异值。

下面是 Drmac 于 2000 年提出的两种算法<sup>[147]</sup>。

### 算法 5.3.2 三矩阵乘积 $\mathbf{B}^T \mathbf{S} \mathbf{C}$ 的奇异值分解 PSVD (B,S,C))<sup>[147]</sup>

输入  $\mathbf{B} \in \mathbb{R}^{p \times m}$ ,  $\mathbf{S} \in \mathbb{R}^{p \times q}$ ,  $\mathbf{C} \in \mathbb{R}^{q \times n}$ ,  $p \leq m$ ,  $q \leq n$ 。

步骤 1 计算  $\mathbf{B}_\tau = \text{diag}(\|\mathbf{b}_1^\top\|_2, \dots, \|\mathbf{b}_p^\top\|_2)$ ,  $\mathbf{C}_\tau = \text{diag}(\|\mathbf{c}_1^\top\|_2, \dots, \|\mathbf{c}_q^\top\|_2)$ , 其中,  $\mathbf{b}_i^\top (i = 1, \dots, p)$  和  $\mathbf{c}_j^\top (j = 1, \dots, q)$  分别是矩阵  $\mathbf{B}$  和  $\mathbf{C}$  的行向量。然后, 令  $\mathbf{B}_1 = \mathbf{B}_\tau^\dagger \mathbf{B}$ ,  $\mathbf{C}_1 = \mathbf{C}_\tau^\dagger \mathbf{C}$ ,  $\mathbf{S}_1 = \mathbf{B}_\tau \mathbf{S} \mathbf{C}_\tau$ 。

步骤 2 利用行和列旋转计算矩阵  $\mathbf{S}_1$  的 LU 分解

$$\Pi_1 \mathbf{S}_1 \Pi_2 = \mathbf{L} \mathbf{U}$$

式中

$$\mathbf{L} \in \mathbb{R}^{p \times p}, \mathbf{U} \in \mathbb{R}^{p \times q}, \rho = \text{rank}(\mathbf{L}) = \text{rank}(\mathbf{U}), L_{ii} = 1, 1 \leq i \leq \rho$$

步骤 3 利用标准的矩阵乘法运算计算

$$\mathbf{M} = \mathbf{L}^T \Pi_1 \mathbf{B}_1, \quad \mathbf{N} = \mathbf{U} \Pi_2^T \mathbf{C}_1$$

应用算法 5.3.1 直接得到  $\mathbf{M}^T \mathbf{N}$  的奇异值分解。

输出 三矩阵乘积  $\mathbf{B}^T \mathbf{S} \mathbf{C}$  的奇异值分解为

$$\begin{bmatrix} \Sigma \oplus O \\ O \end{bmatrix} = \begin{bmatrix} V^T & \\ & I \end{bmatrix} Q_F^T (\mathbf{B}^T \mathbf{S} \mathbf{C}) (Q(W \oplus I_{n-p}))$$

式中,  $Q, Q_F, V$  和  $W$  为在步骤 3 中使用算法 5.3.1 得到的结果。

**算法 5.3.3** 三矩阵乘积  $\mathbf{B}^T \mathbf{S}^{-1} \mathbf{C}$  的奇异值分解 PSVD ( $\mathbf{B}, \mathbf{S}^{-1}, \mathbf{C}$ )<sup>[147]</sup>

输入  $\mathbf{B} \in \mathbb{R}^{p \times m}$ ,  $\mathbf{S} \in \mathbb{R}^{p \times p}$ ,  $\mathbf{C} \in \mathbb{R}^{p \times n}$ ,  $\text{rank}(\mathbf{S}) = p$ 。

步骤 1 计算

$$\begin{aligned}\mathbf{B}_\tau &= \text{diag}(\|b_1^\tau\|_2, \|b_2^\tau\|_2, \dots, \|b_p^\tau\|_2) \\ \mathbf{C}_\tau &= \text{diag}(\|c_1^\tau\|_2, \|c_2^\tau\|_2, \dots, \|c_q^\tau\|_2)\end{aligned}$$

其中,  $b_i^\tau (i = 1, 2, \dots, p)$  和  $c_j^\tau (j = 1, 2, \dots, q)$  分别是矩阵  $\mathbf{B}$  和  $\mathbf{C}$  的行向量。然后, 令  $\mathbf{B}_1 = \mathbf{B}_\tau^{-1} \mathbf{B}, \mathbf{C}_1 = \mathbf{C}_\tau^{-1} \mathbf{C}, \mathbf{S}_1 = \mathbf{C}_\tau^{-1} \mathbf{S} \mathbf{B}_\tau^{-1}$ 。

步骤 2 利用行和列旋转计算矩阵  $\mathbf{S}_1$  的 LU 分解

$$\mathbf{\Pi}_1 \mathbf{S}_1 \mathbf{\Pi}_2 = \mathbf{L} \mathbf{U}, \quad L_{ii} = 1, \quad 1 \leq i \leq p$$

步骤 3 利用标准的矩阵乘法运算计算

$$\mathbf{M} = \mathbf{U}^{-T} \mathbf{\Pi}_2 \mathbf{B}_1, \quad \mathbf{N} = \mathbf{L}^{-1} \mathbf{\Pi}_1^T \mathbf{C}_1$$

应用算法 5.3.1 直接得到  $\mathbf{M}^T \mathbf{N}$  的奇异值分解。

输出 三矩阵乘积  $\mathbf{B}^T \mathbf{S}^{-1} \mathbf{C}$  的奇异值分解为

$$\begin{bmatrix} \Sigma \oplus O \\ O \end{bmatrix} = \begin{bmatrix} V^T & \\ & I \end{bmatrix} Q_F^T (\mathbf{B}^T \mathbf{S}^{-1} \mathbf{C}) (Q(W \oplus I_{n-p}))$$

式中,  $Q, Q_F, V$  和  $W$  为在步骤 3 中使用算法 5.3.1 得到的结果。

## 5.4 奇异值分解的应用

奇异值分解已广泛应用于许多工程问题的解决中。例如, 仅奇异值分解与信号处理的国际学术专题讨论会的论文集就有多部(例如文献 [350], 文献 [197] 等)。本节选择系统辨识和信号处理中的几个典型例子介绍奇异值分解的应用。

### 5.4.1 静态系统的奇异值分解

以电子器件为例, 我们来考虑静态系统的奇异值分解。假定某电子器件的电压  $v$  和电流  $i$  之间存在下列关系(即静态系统模型为)

$$\underbrace{\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}}_{F} \begin{bmatrix} v_1 \\ v_2 \\ i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5.4.1)$$

矩阵  $F$  的元素限定取  $v_1, v_2, i_1, i_2$  的允许值。

如果所用的电压和电流测量装置具有相同的精度 (比如 1%), 那么我们就可以很容易检测任何一组测量值是或不是式 (5.4.1) 在期望的精度范围内的解。假定用各种方法得到另外一个矩阵表达式

$$\begin{bmatrix} 1 & -1 & 10^6 & 10^6 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5.4.2)$$

显然, 只有当电流非常精确测量时, 一组  $v_1, v_2, i_1, i_2$  测量值才会以合适的精度满足式 (5.4.2); 而对于电流测量有 1% 测量误差的一般情况, 式 (5.4.2) 与静态系统模型 (5.4.1) 是大相径庭的: 式 (5.4.1) 给出的电压关系为  $v_1 - v_2 = 0$ , 而由于  $i_1 + i_2 = 0.01$  的测量误差, 式 (5.4.2) 给出的电压关系则是  $v_1 - v_2 + 10^4 = 0$ 。然而, 从代数的角度看, 式 (5.4.1) 和式 (5.4.2) 是完全等价的。因此, 我们希望能够有某些手段来比较几种代数等价的模型表示, 以确定哪一个是希望的、适用一般而不是特殊情况的通用静态系统模型。解决这个问题的基本数学工具就是奇异值分解。

更一般地, 我们考虑  $n$  个电阻的静态系统方程<sup>[108]</sup>

$$\mathbf{F} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \mathbf{0} \quad (5.4.3)$$

式中,  $\mathbf{F}$  是一个  $m \times n$  矩阵。为了简化表示, 我们将一些不变的补偿项撤去了。这样一种表达式是非常通用的, 它可以来自某些物理装置 (例如线性化的物理方程) 和网络方程。矩阵  $\mathbf{F}$  对数据的精确部分和非精确部分的作用可以利用奇异值分解来进行分析。令  $\mathbf{F}$  的奇异值分解为

$$\mathbf{F} = \mathbf{U}^T \boldsymbol{\Sigma} \mathbf{V} \quad (5.4.4)$$

于是, 精确部分和非精确部分的各个分量被矩阵  $\mathbf{F}$  的奇异值  $\sigma_1, \sigma_2, \dots, \sigma_r, 0, \dots, 0$  做不同的大小改变。如果式 (5.4.3) 是物理装置设计的准确规格, 那么矩阵  $\mathbf{F}$  的奇异值分解将提供一个代数等价, 但在数值上是最可靠的设计方程。注意到  $\mathbf{U}$  是一正交矩阵, 所以由式 (5.4.3) 和式 (5.4.4) 有

$$\boldsymbol{\Sigma} \mathbf{V} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \mathbf{0} \quad (5.4.5)$$

若将对角矩阵  $\boldsymbol{\Sigma}$  分块为

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix}$$

并将正交矩阵  $\mathbf{V}$  作相应的分块, 即

$$\mathbf{V} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}$$

其中,  $[\mathbf{A}, \mathbf{B}]$  是  $\mathbf{V}$  最上面的  $r$  行, 则式 (5.4.5) 可以写作

$$\begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \mathbf{0}$$

从而，我们可以得到与式 (5.4.3) 在代数上等价，但在数值上是最可靠的表达式

$$[\mathbf{A}, \mathbf{B}] \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \mathbf{0} \quad (5.4.6)$$

如果式 (5.4.3) 是物理装置的不精确模型，则对角矩阵的对角线上就不会出现零奇异值。这时，我们就不能够直接使用式 (5.4.6)。在这种情况下，我们需要对模型进行修正，方法是令所有奇异值  $\sigma_s, \sigma_{s+1}, \dots$  等于零，其中， $s$  是满足  $\sigma_s/\sigma_1$  小于矩阵  $\mathbf{F}$  的元素所允许的精确度 (即物理装置的测量精确度) 的最小整数。于是，式 (5.4.6) 中的  $[\mathbf{A}, \mathbf{B}]$  修正为  $\mathbf{V}$  的最上面  $s - 1$  行。有关结果表明，这样一种修正可以使参数的变化限制在预先设定的误差范围内<sup>[108]</sup>。

现在考虑一个电阻性的多端对 (电阻、电导、混合参数、传导和散射等) 的不同表达式，目的是寻找一个尽可能最优的表达式。例如，使用端对坐标  $x$  和  $y$  时，电阻性多端对的显式表示则为<sup>[108]</sup>

$$\mathbf{y} = \mathbf{Ax}, \quad \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \boldsymbol{\Omega} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} \quad (5.4.7)$$

通过选择合适的坐标变换  $\boldsymbol{\Omega}$ ，就可以得到电阻、电导、任意混合参数或传导的表达式。于是，矩阵  $\mathbf{A}$  的条件数就代表从  $\mathbf{x}$  到  $\mathbf{y}$  的信噪比放大倍数的上限。如果  $\mathbf{A}$  可逆，则该条件数也是从  $\mathbf{y}$  到  $\mathbf{x}$  的信噪比放大倍数的上限。因此，不同的表达式就可以根据它们的条件数进行排队。这就使得所有参数化表达式一目了然。显然，最优的情况是条件数  $\text{cond}(\mathbf{A}) = 1$  或  $\mathbf{A}$  是一正交矩阵 (包含一比例因子)。一个自然会问的问题是，任何一个多端对的电阻器是否有一个最优的表达式？也就是说，是否存在使得  $\text{cond}(\mathbf{A}) = 1$  的正交矩阵  $\mathbf{A}$ ？为此，让我们来看一个  $n$  维  $n$  端对的电阻器的隐含表达式

$$\mathbf{F} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} = \mathbf{0}, \quad \text{rank}(\mathbf{F}) = n \quad (5.4.8)$$

应用  $\mathbf{F}$  的奇异值分解式 (5.4.4)，即可得到式 (5.4.6)，其中， $r = n$ 。选择正交坐标变换

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \underbrace{\begin{bmatrix} I/\sqrt{2} & I/\sqrt{2} \\ -I/\sqrt{2} & I/\sqrt{2} \end{bmatrix}}_{\boldsymbol{\Omega}} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} \quad (5.4.9)$$

这样一来，就可以利用  $\boldsymbol{\Omega}$  的正交性  $\boldsymbol{\Omega}^{-1} = \boldsymbol{\Omega}^T$ ，将隐含表达式 (5.4.6) 表示成

$$\begin{aligned} [\mathbf{A}, \mathbf{B}] \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix} &= [\mathbf{A}, \mathbf{B}] \begin{bmatrix} \mathbf{A}^T & \mathbf{C}^T \\ \mathbf{B}^T & \mathbf{D}^T \end{bmatrix} \begin{bmatrix} I/\sqrt{2} & -I/\sqrt{2} \\ I/\sqrt{2} & I/\sqrt{2} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} \\ &= [\mathbf{I}, \mathbf{O}] \begin{bmatrix} I/\sqrt{2} & -I/\sqrt{2} \\ I/\sqrt{2} & I/\sqrt{2} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \mathbf{0} \end{aligned}$$

即有

$$\begin{bmatrix} I/\sqrt{2} & -I/\sqrt{2} \\ I/\sqrt{2} & I/\sqrt{2} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \mathbf{0} \Rightarrow \mathbf{y} = \mathbf{x} \quad (5.4.10)$$

于是，可以得出结论：利用式 (5.4.4) 的奇异值分解可以得到式 (5.4.9) 的正交变换，而通过此正交变换，即可得到一个在数值上最优的显式关系  $\mathbf{y} = \mathbf{x}$ 。

### 5.4.2 图像压缩

奇异值分解在图像处理中有着重要应用。假定一幅图像有  $n \times n$  个像素，如果将这  $n^2$  个数据一起传送，往往会觉得数据量太大。因此，我们希望能够改为传送另外一些比较少的数据，并且在接收端还能够利用这些传送的数据重构原图像。

不妨用  $n \times n$  矩阵  $\mathbf{A}$  表示要传送的原  $n \times n$  个像素。假定对矩阵  $\mathbf{A}$  进行奇异值分解，便得到  $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$ ，其中，奇异值按照从大到小的顺序排列。如果从中选择  $k$  个大奇异值以及与这些奇异值对应的左和右奇异向量逼近原图像，便可以共使用  $k(2n+1)$  个数值代替原来的  $n \times n$  个图像数据。这  $k(2n+1)$  个被选择的新数据是矩阵  $\mathbf{A}$  的前  $k$  个奇异值、 $n \times n$  左奇异向量矩阵  $\mathbf{U}$  的前  $k$  列和  $n \times n$  右奇异向量矩阵  $\mathbf{V}$  的前  $k$  列的元素。

比率

$$\rho = \frac{n^2}{k(2n+1)} \quad (5.4.11)$$

称为图像的压缩比。显然，被选择的大奇异值的个数  $k$  应该满足条件  $k(2n+1) < n^2$  即  $k < \frac{n^2}{2n+1}$ 。因此，我们在传送图像的过程中，就无须传送  $n \times n$  个原始数据，而只需要传送  $k(2n+1)$  个有关奇异值和奇异向量的数据即可。在接收端，在接收到奇异值  $\sigma_1, \sigma_2, \dots, \sigma_k$  以及左奇异向量  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k$  和右奇异向量  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  后，即可通过截尾的奇异值分解公式

$$\hat{\mathbf{A}} = \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i^T \quad (5.4.12)$$

重构出原图像。

一个容易理解的事实是：若  $k$  值偏小，即压缩比  $\rho$  偏大，则重构的图像的质量有可能不能令人满意。反之，过大的  $k$  值又会导致压缩比过小，从而降低图像压缩和传送的效率。因此，需要根据不同种类的图像，选择合适的压缩比，以兼顾图像传送效率和重构质量。

## 5.5 广义奇异值分解

前面介绍了一个矩阵的奇异值分解和两个矩阵乘积的奇异值分解。本节将讨论两个矩阵组成的矩阵束  $(\mathbf{A}, \mathbf{B})$  的奇异值分解。这种分解称为广义奇异值分解。

### 5.5.1 广义奇异值分解的定义与性质

广义奇异值分解 (GSVD) 方法是 Van Loan 于 1976 年最早提出来<sup>[493]</sup>。

**定理 5.5.1 (广义奇异值分解 1)** <sup>[493]</sup> 若  $\mathbf{A} \in \mathbb{C}^{m \times n}$ ,  $m \geq n$  和  $\mathbf{B} \in \mathbb{C}^{p \times n}$ , 则存在酉

矩阵  $\mathbf{U} \in \mathbb{C}^{m \times m}$  和  $\mathbf{V} \in \mathbb{C}^{p \times p}$  以及非奇异矩阵  $\mathbf{Q} \in \mathbb{C}^{n \times n}$ , 使得

$$\mathbf{U}\mathbf{A}\mathbf{Q} = [\Sigma_A \quad \mathbf{O}], \quad \Sigma_A = \begin{bmatrix} \mathbf{I}_r & & \\ & \mathbf{S}_A & \\ & & \mathbf{O}_A \end{bmatrix} \quad (5.5.1)$$

$$\mathbf{V}\mathbf{B}\mathbf{Q} = [\Sigma_B \quad \mathbf{O}], \quad \Sigma_B = \begin{bmatrix} \mathbf{O}_B & & \\ & \mathbf{S}_B & \\ & & \mathbf{I}_{k-r-s} \end{bmatrix} \quad (5.5.2)$$

式中

$$\mathbf{S}_A = \text{diag}(\alpha_{r+1}, \alpha_{r+2}, \dots, \alpha_{r+s}), \quad \mathbf{S}_B = \text{diag}(\beta_{r+1}, \beta_{r+2}, \dots, \beta_{r+s}) \quad (5.5.3)$$

$$\left. \begin{array}{l} 1 > \alpha_{r+1} \geq \dots \geq \alpha_{r+s} > 0 \\ 0 < \beta_{r+1} \leq \dots \leq \beta_{r+s} < 1 \\ \alpha_i^2 + \beta_i^2 = 1, \quad i = r+1, r+2, \dots, r+s \end{array} \right\} \quad (5.5.4)$$

整数  $k, r$  和  $s$  分别为

$$k = \text{rank} \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}, \quad r = \text{rank} \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} - \text{rank}(\mathbf{B})$$

和

$$s = \text{rank}(\mathbf{A}) + \text{rank}(\mathbf{B}) - \text{rank} \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}$$

本定理有多种证明方法, 可参见 Van Loan [493], Paige 与 Saunders [389], Golub 与 Van Loan [198] 和 Zha [537] 的证明。

根据文献 [493], 式 (5.5.1) 的对角矩阵  $\Sigma_A$  和式 (5.5.2) 的对角矩阵  $\Sigma_B$  的对角线上的元素组成广义奇异值对  $(\alpha_i, \beta_i)$ 。由  $\Sigma_A$  和  $\Sigma_B$  的形式, 前  $k$  个广义奇异值对分为三种情况

$$\alpha_i = 1, \quad \beta_i = 0, \quad i = 1, 2, \dots, r$$

$$\alpha_i, \beta_i \quad (\mathbf{S}_A \text{ 和 } \mathbf{S}_B \text{ 的元素}), \quad i = r+1, r+2, \dots, r+s$$

$$\alpha_i = 0, \quad \beta_i = 1, \quad i = r+s+1, r+s+2, \dots, k$$

这  $k$  个奇异值对  $(\alpha_i, \beta_i)$  统称矩阵束  $(\mathbf{A}, \mathbf{B})$  的非平凡广义奇异值对; 而  $\alpha_i/\beta_i (i = 1, 2, \dots, k)$  称为矩阵束  $(\mathbf{A}, \mathbf{B})$  的非平凡广义奇异值 (包括无穷大, 有限值和零)。反之, 对应于式 (5.5.1) 和式 (5.5.2) 中零列向量的另外  $n-k$  对广义奇异值则称为矩阵束  $(\mathbf{A}, \mathbf{B})$  的平凡广义奇异值对。

定理 5.5.1 限制矩阵  $\mathbf{A}$  的列数不得大于行数。当矩阵  $\mathbf{A}$  的维数不满足这一限制时, 定理 5.5.1 便不能适用。Paige 与 Saunders [389] 推广了定理 5.5.1, 提出了具有相同列数的任意矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义奇异值分解。

**定理 5.5.2** (广义奇异值分解 2)<sup>[389]</sup> 假定矩阵  $A \in \mathbb{C}^{m \times n}$  和  $B \in \mathbb{C}^{p \times n}$ , 则对于分块矩阵

$$K = \begin{bmatrix} A \\ B \end{bmatrix}, \quad t = \text{rank}(K)$$

存在酉矩阵

$$U \in \mathbb{C}^{m \times m}, \quad V \in \mathbb{C}^{p \times p}, \quad W \in \mathbb{C}^{t \times t}, \quad Q \in \mathbb{C}^{n \times n}$$

使得

$$\begin{aligned} U^H A Q &= \Sigma_A [\underbrace{W^H R}_t, \underbrace{O}_{n-t}] \\ V^H B Q &= \Sigma_B [\underbrace{W^H R}_t, \underbrace{O}_{n-t}] \end{aligned}$$

式中

$$\Sigma_A = \begin{bmatrix} I_A & & \\ & D_A & \\ & & O_A \end{bmatrix}, \quad \Sigma_B = \begin{bmatrix} I_B & & \\ & D_B & \\ & & O_B \end{bmatrix} \quad (5.5.5)$$

并且  $R \in \mathbb{C}^{t \times t}$  非奇异, 其奇异值等于矩阵  $K$  的非零奇异值。矩阵  $I_A$  为  $r \times r$  单位矩阵,  $I_B$  为  $(t-r-s) \times (t-r-s)$  单位矩阵, 其中,  $r$  和  $s$  的值与所给数据有关, 且  $O_A$  和  $O_B$  分别为  $(m-r-s) \times (t-r-s)$  维和  $(p-t+r) \times r$  维零矩阵 (这两个零矩阵有可能没有任何行或任何列), 而

$$D_A = \text{diag}(\alpha_{r+1}, \alpha_{r+2}, \dots, \alpha_{r+s}), \quad D_B = \text{diag}(\beta_{r+1}, \beta_{r+2}, \dots, \beta_{r+s})$$

满足

$$1 > \alpha_{r+1} \geq \alpha_{r+2} \geq \dots \geq \alpha_{r+s} > 0, \quad 0 < \beta_{r+1} \leq \beta_{r+2} \leq \dots \leq \beta_{r+s} < 1$$

和

$$\alpha_i^2 + \beta_i^2 = 1, \quad i = r+1, r+2, \dots, r+s$$

**证明** 参见文献 [243]。

下面是有关广义奇异值分解的几点注释。

**注释 1** 由  $(A, B)$  的广义奇异值分解与  $AB^{-1}$  的奇异值分解之间的等价性显见, 若矩阵  $B$  为单位矩阵 ( $B = I$ ), 则广义奇异值分解简化为普通的奇异值分解。这一观察结果也可从广义奇异值的定义直接得出。这是因为, 单位矩阵的奇异值全部等于 1, 从而矩阵束  $(A, I)$  的广义奇异值与  $A$  的奇异值等价。

**注释 2** 当矩阵  $B$  非奇异时, 矩阵束  $\{A, B\}$  的广义奇异值分解等同于矩阵乘积  $AB^{-1}$  的奇异值分解。由于  $AB^{-1}$  具有类似于商的形式, 以及广义奇异值本身就是矩阵  $A$  和  $B$  的奇异值之商, 所以广义奇异值分解有时也被称作商奇异值分解 (quotient singular value decomposition, QSVD)。

**注释 3** 如果矩阵  $\mathbf{B}$  不是正方的, 或者  $\mathbf{B}$  是奇异的正方矩阵, 则  $\mathbf{AB}^\dagger$  (其中,  $\mathbf{B}^\dagger$  是  $\mathbf{B}$  的 Moore-Penrose 广义逆) 的奇异值不一定对应为矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义奇异值。更严格地, 有以下结论。

**定理 5.5.3**<sup>[537]</sup> 定义

$$\mathbf{B}_A^\dagger = \mathbf{Q} \begin{bmatrix} \mathbf{O}_B^H & & \\ & \mathbf{S}_B^{-1} & \\ & & I \end{bmatrix} \mathbf{V}$$

若  $\text{rank}[\mathbf{A}^H, \mathbf{B}^H]^H = n$ , 则  $\mathbf{B}_A^\dagger$  是唯一定义的, 并且  $\mathbf{AB}_A^\dagger$  的奇异值包含了矩阵束  $(\mathbf{A}, \mathbf{B})$  的全部有限大的广义奇异值。

应当注意,  $\mathbf{B}_A^\dagger$  并不是  $\mathbf{B}$  的 Moore-Penrose 广义逆矩阵, 因为它只满足 Moore-Penrose 广义逆矩阵的四个条件中的三个条件

$$\mathbf{BB}_A^\dagger \mathbf{B} = \mathbf{B} \quad (5.5.6)$$

$$\mathbf{B}_A^\dagger \mathbf{BB}_A^\dagger = \mathbf{B}_A^\dagger \quad (5.5.7)$$

$$(\mathbf{BB}_A^\dagger)^H = \mathbf{BB}_A^\dagger \quad (5.5.8)$$

下面的定理表明,  $\mathbf{B}_A^\dagger$  是一种约束最小化问题的唯一解。

**定理 5.5.4**<sup>[537]</sup> 若  $[\mathbf{A}^H, \mathbf{B}^H]^H$  满列秩, 则  $\mathbf{B}_A^\dagger$  是下列约束极小化问题的唯一解

$$\min_{X \in \mathbb{C}^{n \times q}} \|\mathbf{AX}\|_F \quad (5.5.9)$$

subject to  $\mathbf{BXB} = \mathbf{B}$ ,  $\mathbf{XBX} = \mathbf{X}$ ,  $(\mathbf{BX})^H = \mathbf{BX}$

且  $\|\mathbf{AX}\|_F$  的极小化值为  $\sqrt{\sum_{i=r+1}^{r+s} (\alpha_i/\beta_i)^2}$ 。

## 5.5.2 广义奇异值分解的实际算法

如果  $\mathbf{A}$  或  $\mathbf{B}$  相对于方程求解是病态的, 那么计算  $\mathbf{AB}^{-1}$  通常会导致非常大的数值误差, 所以对  $\mathbf{AB}^{-1}$  本身进行奇异值分解一般并不值得推荐采用。一个自然会问的问题是, 能否绕开计算  $\mathbf{AB}^{-1}$  这一步, 而直接得到  $\mathbf{C} = \mathbf{AB}^{-1}$  的奇异值分解? 这是完全可能的, 因为  $\mathbf{C} = \mathbf{AB}^{-1}$  的奇异值分解实质上就是两个矩阵乘积的奇异值分解。

Paige<sup>[390]</sup> 根据  $\mathbf{C} = \mathbf{AB}^{-1}$  的奇异值分解与矩阵乘积的奇异值分解形式上的一致, 提出了一种实际的广义奇异值分解算法。这种算法的关键是如何避免矩阵求逆  $\mathbf{B}^{-1}$  以及如何适用于矩阵  $\mathbf{B}$  奇异的一般情况。

先讨论矩阵  $\mathbf{B}$  非奇异的情况。令  $\mathbf{A}_{ij}$  和  $\mathbf{B}_{ij}$  均代表  $2 \times 2$  矩阵, 它们的元素分别位于  $\mathbf{A}$  的第  $i, j$  行和  $\mathbf{B}$  的第  $i, j$  列。如果选择酉矩阵  $\mathbf{U}$  和  $\mathbf{V}$  使得

$$\mathbf{U}^H \mathbf{A}_{ij} \mathbf{B}_{ij}^{-1} \mathbf{V} = \mathbf{S} \quad (5.5.10)$$

是对角矩阵，则

$$\mathbf{U}^H \mathbf{A}_{ij} = \mathbf{S} \mathbf{V}^H \mathbf{B}_{ij} \quad (5.5.11)$$

结果是， $\mathbf{U}^H \mathbf{A}_{ij}$  的第 1 行与  $\mathbf{V}^H \mathbf{B}_{ij}$  的第 1 行平行， $\mathbf{U}^H \mathbf{A}_{ij}$  的第 2 行与  $\mathbf{V}^H \mathbf{B}_{ij}$  的第 2 行平行。因此，如果  $\mathbf{Q}$  是使得  $\mathbf{V}^H \mathbf{B}_{ij} \mathbf{Q}$  为下三角矩阵的酉矩阵，即

$$(\mathbf{V}^H \mathbf{B}_{ij}) \mathbf{Q} = \begin{bmatrix} \times & \otimes \\ \times & \times \end{bmatrix} = \begin{bmatrix} \times & \\ \times & \times \end{bmatrix} \quad (5.5.12)$$

则  $\mathbf{U}^H \mathbf{A}_{ij} \mathbf{Q}$  也是下三角矩阵。对于  $n \times n$  上三角矩阵  $\mathbf{C} = \mathbf{AB}^{-1}$ ，可以执行  $n(n-1)/2$  次  $2 \times 2$  Kogbetliantz 算法，使矩阵  $\mathbf{A}, \mathbf{B}$  和  $\mathbf{C}$  在上三角和下三角形式之间来回变换，最后收敛为对角矩阵形式。

广义奇异值分解也可等价叙述为以下定理<sup>[198]</sup>。

**定理 5.5.5** 若  $\mathbf{A} \in \mathbb{C}^{m_1 \times n} (m_1 \geq n)$ ,  $\mathbf{B} \in \mathbb{C}^{m_2 \times n} (m_2 \geq n)$ , 则存在一非奇异矩阵  $\mathbf{X} \in \mathbb{C}^{n \times n}$  使得

$$\begin{aligned} \mathbf{X}^H (\mathbf{A}^H \mathbf{A}) \mathbf{X} &= \mathbf{D}_A = \text{diag}(\alpha_1, \alpha_2, \dots, \alpha_n), \quad \alpha_k \geq 0 \\ \mathbf{X}^H (\mathbf{B}^H \mathbf{B}) \mathbf{X} &= \mathbf{D}_B = \text{diag}(\beta_1, \beta_2, \dots, \beta_n), \quad \beta_k \geq 0 \end{aligned}$$

式中， $\sigma_k = \sqrt{\alpha_k / \beta_k}$  称为矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义奇异值，且  $\mathbf{X}$  的列  $\mathbf{x}_k$  称为与  $\sigma_k$  对应的广义奇异向量。

定理 5.5.5 给出了计算矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义奇异值分解的多种算法。特别地，我们对寻求使  $\mathbf{D}_B$  为单位矩阵的广义奇异向量矩阵  $\mathbf{X}$  更加感兴趣，因为在这一情况下，广义奇异值  $\sigma_k$  由  $\sqrt{\alpha_k}$  直接给出。下面就是这样的两种实际算法。

#### 算法 5.5.1 GSVD 算法 1<sup>[494]</sup>

步骤 1 计算矩阵的内积  $\mathbf{S}_1 = \mathbf{A}^H \mathbf{A}$  和  $\mathbf{S}_2 = \mathbf{B}^H \mathbf{B}$ 。

步骤 2 计算  $\mathbf{S}_2$  的特征值分解  $\mathbf{U}_2^H \mathbf{S}_2 \mathbf{U}_2 = \mathbf{D} = \text{diag}(\gamma_1, \dots, \gamma_n)$ 。

步骤 3 计算  $\mathbf{Y} = \mathbf{U}_2 \mathbf{D}^{-1/2}$  和  $\mathbf{C} = \mathbf{Y}^H \mathbf{S}_1 \mathbf{Y}$ 。

步骤 4 计算  $\mathbf{C}$  的特征值分解  $\mathbf{Q}^H \mathbf{C} \mathbf{Q} = \text{diag}(\alpha_1, \dots, \alpha_n)$ ，其中， $\mathbf{Q}^H \mathbf{Q} = \mathbf{I}$ 。

步骤 5 广义奇异向量矩阵为  $\mathbf{X} = \mathbf{Y} \mathbf{Q}$ ，且广义奇异值为  $\sqrt{\alpha_k}$ ,  $k = 1, \dots, n$ 。

#### 算法 5.5.2 GSVD 算法 2<sup>[494]</sup>

步骤 1 计算  $\mathbf{B}$  的奇异值分解  $\mathbf{U}_2^H \mathbf{B} \mathbf{V}_2 = \mathbf{D} = \text{diag}(\gamma_1, \dots, \gamma_n)$ 。

步骤 2 计算  $\mathbf{Y} = \mathbf{V}_2 \mathbf{D}^{-1} \mathbf{V}_2 = \text{diag}(1/\gamma_1, \dots, 1/\gamma_n)$ 。

步骤 3 计算  $\mathbf{C} = \mathbf{A} \mathbf{Y}$ 。

步骤 4 计算矩阵  $\mathbf{C}$  的奇异值分解  $\mathbf{U}_1^H \mathbf{C} \mathbf{V}_1 = \mathbf{D}_A = \text{diag}(\alpha_1, \dots, \alpha_n)$ 。

步骤 5  $\mathbf{X} = \mathbf{Y} \mathbf{V}_1$  为广义奇异向量矩阵，而  $\alpha_k, k = 1, 2, \dots, n$  直接是矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义奇异值。

算法 5.5.1 与算法 5.5.2 的主要区别在于：前者需要计算矩阵乘积  $\mathbf{A}^H \mathbf{A}$  和  $\mathbf{B}^H \mathbf{B}$ ，而后者则完全避免了这一计算。正如前面已说明的那样，在计算两个矩阵乘积时会发生信

息的丢失，并会使条件数变坏。因此，算法 5.5.2 具有比算法 5.5.1 更好的数值性能。但是，由于需要矩阵求逆或矩阵乘积的计算，算法 5.5.1 和算法 5.5.2 的性能或多或少都会遭到损害。

一种可以避免任何矩阵求逆或矩阵内积运算的广义奇异值分解算法由 Speiser 与 Van Loan<sup>[458]</sup> 提出（也见文献 [494]）。

### 算法 5.5.3 GSVD 算法 3

#### 步骤 1 计算 QR 分解

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \mathbf{R}$$

其中， $\mathbf{Q}_1$  和  $\mathbf{Q}_2$  分别与  $\mathbf{A}$  和  $\mathbf{B}$  具有相同的维数，且  $\mathbf{R} \in \mathbb{C}^{n \times n}$  为上三角矩阵。假定  $\mathbf{R}$  非奇异，即  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B}) = \{0\}$ 。

#### 步骤 2 计算 CS 分解

$$\begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \mathbf{C} \\ \mathbf{S} \end{bmatrix} \mathbf{V}$$

其中， $\mathbf{U}_1$ ， $\mathbf{U}_2$  和  $\mathbf{V}$  为酉矩阵， $\mathbf{C} = \text{diag}(\cos(\theta_k))$ ， $\mathbf{S} = \text{diag}(\sin(\theta_k))$ ，且  $0 \leq \theta_1 \leq \dots \leq \theta_n \leq \pi/2$ 。由此可知，若  $\mathbf{X} = \mathbf{R}^{-1}\mathbf{V}$ ，则  $\mathbf{X}^H(\mathbf{A}^H\mathbf{A} - \mu^2\mathbf{B}^H\mathbf{B})\mathbf{X} = \mathbf{C}^H\mathbf{C} - \lambda\mathbf{S}^H\mathbf{S}$ ，因此，广义奇异值由  $\mu_k = \cot(\theta_k)$  给出。

步骤 3 利用  $c_d > \epsilon + c_n \geq c_{d+1} \geq \dots \geq c_n \geq 0$  ( $\epsilon > 0$  为小的扰动)，其中， $c_k = \cos(\theta_k)$ 。

步骤 4 计算乘积  $\mathbf{ZT} = \mathbf{R}^H\mathbf{V}$  的 QR 分解，其中， $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$  为酉矩阵， $\mathbf{T} \in \mathbb{C}^{n \times n}$  为上三角矩阵。由于

$$\mathbf{X} = \mathbf{R}^{-1}\mathbf{V} = (\mathbf{V}^H\mathbf{R})^{-1} = (\mathbf{R}^H\mathbf{V})^{-H} = (\mathbf{ZT})^{-H} = \mathbf{ZT}^{-H}$$

且  $\mathbf{T}^{-H}$  为下三角矩阵，故有  $\text{Span}\{\mathbf{z}_{d+1}, \mathbf{z}_{d+2}, \dots, \mathbf{z}_n\} = \text{Span}\{\mathbf{x}_{d+1}, \mathbf{x}_{d+2}, \dots, \mathbf{x}_n\}$ 。

1998 年，Drmac 提出了计算广义奇异值分解的正切算法 (tangent algorithm)<sup>[146]</sup>。这种算法分两个阶段进行：第一阶段将矩阵束  $(\mathbf{A}, \mathbf{B})$  简化为一个矩阵  $\mathbf{F}$ ；第二阶段计算矩阵  $\mathbf{F}$  的奇异值分解。正切算法的理论基础是，广义奇异值分解在等价变换下是不变的，即有

$$(\mathbf{A}, \mathbf{B}) \rightarrow (\mathbf{A}', \mathbf{B}') = (\mathbf{U}^T \mathbf{AS}, \mathbf{V}^T \mathbf{BS}) \quad (5.5.13)$$

式中， $\mathbf{U}, \mathbf{V}$  是任意的正交矩阵，且  $\mathbf{S}$  是任意的非奇异矩阵。因此，根据定义，两个矩阵束  $(\mathbf{A}, \mathbf{B})$  和  $(\mathbf{A}', \mathbf{B}')$  具有相同的广义奇异值分解。

### 算法 5.5.4 广义奇异值分解的正切算法 [146]

输入 矩阵  $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \in \mathbb{R}^{m \times n}$ ， $\mathbf{B} \in \mathbb{R}^{p \times n}$ ， $m \geq n$ ， $\text{rank}(\mathbf{B}) = n$ 。

#### 步骤 1 计算

$$\Delta_A = \text{diag}(\|\mathbf{a}_1\|_2, \|\mathbf{a}_2\|_2, \dots, \|\mathbf{a}_n\|_2)$$

$$\mathbf{A}_c = \mathbf{A}\Delta_A^{-1}, \quad \mathbf{B}_1 = \mathbf{B}\Delta_A^{-1}$$

步骤 2 利用具有列旋转的 Householder QR 分解算法计算

$$\begin{bmatrix} \mathbf{R} \\ \mathbf{O} \end{bmatrix} = \mathbf{Q}^T \mathbf{B}_1 \mathbf{\Pi}$$

步骤 3 通过求解矩阵方程  $\mathbf{F}\mathbf{R} = \mathbf{A}_c \mathbf{\Pi}$ , 计算  $\mathbf{F} = \mathbf{A}_c \mathbf{\Pi} \mathbf{R}^{-1}$ 。

步骤 4 计算矩阵  $\mathbf{F}$  的奇异值分解

$$\begin{bmatrix} \Sigma \\ \mathbf{O} \end{bmatrix} = \mathbf{V}^T \mathbf{F} \mathbf{U}$$

步骤 5 计算矩阵

$$\mathbf{X} = \mathbf{\Delta}_A^{-1} \mathbf{\Pi} \mathbf{R}^{-1} \mathbf{U}, \quad \mathbf{W} = \mathbf{Q} \begin{bmatrix} \mathbf{U} & \mathbf{O} \\ \mathbf{O} & I_{p-n} \end{bmatrix}$$

输出  $(\mathbf{A}, \mathbf{B})$  的广义奇异值分解读作

$$\begin{bmatrix} \mathbf{V}^T & \mathbf{A} \\ \mathbf{W}^T & \mathbf{B} \end{bmatrix} \mathbf{X} = \begin{bmatrix} \Sigma & \mathbf{O} \\ \mathbf{I} & \mathbf{O} \end{bmatrix}$$

### 5.5.3 高阶广义奇异值分解

广义奇异值分解是两个矩阵组成的矩阵束  $(\mathbf{A}, \mathbf{B})$  的奇异值分解。针对由  $\mathbf{A}_i \in \mathbb{R}^{m_i \times n}$  组成的  $N$  元矩阵组  $(\mathbf{A}_1, \dots, \mathbf{A}_N)$ , Ponnappalli 等人<sup>[412]</sup>于 2011 年提出了高阶广义奇异值分解 (higher-order generalized singular value decomposition, HO GSVD)

$$\mathbf{A}_i = \mathbf{U}_i \boldsymbol{\Sigma}_i \mathbf{V}^T, \quad i = 1, \dots, N \quad (5.5.14)$$

式中

$$\mathbf{U}_i = [\mathbf{u}_{i,1}, \dots, \mathbf{u}_{i,n}] \in \mathbb{R}^{m_i \times n}, \quad \|\mathbf{u}_{i,k}\|_2 = 1 \quad (5.5.15)$$

$$\boldsymbol{\Sigma}_i = \text{diag}(\sigma_{i,1}, \dots, \sigma_{i,n}) \in \mathbb{R}^{n \times n}, \quad \sigma_{i,k} > 0 \quad (5.5.16)$$

$$\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{R}^{n \times n}, \quad \|\mathbf{v}_k\|_2 = 1 \quad (5.5.17)$$

$\sigma_{i,k}$  称为矩阵  $\mathbf{A}_i$  的第  $k$  个高阶广义奇异值, 反映第  $k$  个右基向量  $\mathbf{v}_k$  在矩阵  $\mathbf{A}_i$  中的重要程度。

高阶广义奇异值分解可以用向量的外积形式等价写作

$$\mathbf{A}_i = \mathbf{U}_i \boldsymbol{\Sigma}_i \mathbf{V}^T = \sum_{k=1}^n \sigma_{i,k} \mathbf{u}_{i,k} \mathbf{v}_k^T = \sum_{k=1}^n \sigma_{i,k} \mathbf{u}_{i,k} \circ \mathbf{v}_k \quad (5.5.18)$$

其中  $\|\mathbf{u}_{i,k} \mathbf{v}_k^T\|_F = 1$ 。

令

$$\begin{aligned} \mathbf{S} &= \frac{1}{N(N-1)} \sum_{i=1, i \neq j}^N \sum_{j=1}^N (\mathbf{A}_i \mathbf{A}_j^{-1} + \mathbf{A}_j \mathbf{A}_i^{-1}) \\ &= \frac{2}{N(N-1)} \sum_{i=1, i \neq j}^N \sum_{j=1}^N S_{ij} \end{aligned} \quad (5.5.19)$$

其中

$$S_{ij} = \frac{1}{2} (\mathbf{A}_i \mathbf{A}_j^{-1} + \mathbf{A}_j \mathbf{A}_i^{-1}), \quad i \neq j \quad (5.5.20)$$

由式 (5.5.18) 和式 (5.5.19) 易知,  $\mathbf{S}$  为对称矩阵。令  $\mathbf{S}$  的特征值分解  $\mathbf{S}\mathbf{V} = \mathbf{V}\mathbf{A}$  即

$$\mathbf{S}\mathbf{v}_i = \lambda_i \mathbf{v}_i, \quad i = 1, \dots, n \quad (5.5.21)$$

并构成矩阵  $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n]$  和对角矩阵  $\mathbf{A} = \text{diag}(\lambda_1, \dots, \lambda_n)$ 。

对称矩阵  $\mathbf{S}$  的特征值分解具有以下性质:

(1) 矩阵  $\mathbf{S}$  具有  $N$  个独立的特征向量, 并且特征值  $\lambda_i$  和特征向量  $\mathbf{v}_i$  都是实的。

(2)  $\mathbf{S}$  的特征值  $\lambda_i \geq 1$ 。

一旦确定了矩阵  $\mathbf{V}$ , 即可令  $\mathbf{X}_i = \mathbf{U}_i \boldsymbol{\Sigma}_i$ , 将高阶广义奇异值分解  $\mathbf{A}_i = \mathbf{U}_i \boldsymbol{\Sigma}_i \mathbf{V}^T$  变成  $\mathbf{A}_i = \mathbf{X}_i \mathbf{V}^T$ 。于是, 未知的矩阵  $\mathbf{X}_i$  可以通过求解  $N$  个独立的线性方程组

$$\mathbf{V} \mathbf{X}_i^T = \mathbf{A}_i^T, \quad i = 1, \dots, N \quad (5.5.22)$$

求出。

获得  $\mathbf{X}_i$  之后, 又可利用

$$\mathbf{X}_i = [\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,n}] = \mathbf{U}_i \boldsymbol{\Sigma}_i = [\sigma_{i,1} \mathbf{u}_{i,1}, \dots, \sigma_{i,n} \mathbf{u}_{i,n}]$$

得到关系式

$$\mathbf{x}_{i,k} = \sigma_{i,k} \mathbf{u}_{i,k}, \quad k = 1, \dots, n; \quad i = 1, \dots, N \quad (5.5.23)$$

由于高阶广义奇异值分解要求  $\|\mathbf{u}_{i,k}\|_2 = 1$ , 故由式 (5.5.23) 立即得到由  $\mathbf{X}_i$  重构  $\boldsymbol{\Sigma}$  和  $\mathbf{U}_i$  的公式

$$\sigma_{i,k} = \|\mathbf{x}_{i,k}\|_2, \quad \mathbf{u}_{i,k} = \mathbf{x}_{i,k} / \sigma_{i,k}, \quad k = 1, \dots, n; \quad i = 1, \dots, N \quad (5.5.24)$$

以上讨论与结果可以综合得到下面高阶广义奇异值分解算法。

#### 算法 5.5.5 高阶广义奇异值分解算法<sup>[412]</sup>

已知  $N$  个矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_N$ 。

步骤 1 利用式 (5.5.19) 计算矩阵  $\mathbf{S}$ 。

步骤 2 计算  $\mathbf{S}$  的特征值分解, 并由特征向量组成  $N$  个矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_N$  共同的右基向量矩阵  $\mathbf{V}$ 。

步骤 3 求解  $N$  个独立的线性方程组  $\mathbf{V}\mathbf{X}_i^T = \mathbf{A}_i^T$ ,  $i = 1, \dots, N$ , 得到  $\mathbf{X}_i$ ,  $i = 1, \dots, N$ 。

步骤 4 利用  $\sigma_{i,k} = \|\mathbf{x}_{i,k}\|_2$ ,  $\mathbf{u}_{i,k} = \mathbf{x}_{i,k}/\sigma_{i,k}$  ( $k = 1, \dots, n$ ;  $i = 1, \dots, N$ ) 重构高阶广义奇异值矩阵  $\Sigma_i = \text{diag}(\sigma_{i,1}, \dots, \sigma_{i,n})$  和左基向量矩阵  $\mathbf{U}_i = [\mathbf{u}_{i,1}, \dots, \mathbf{u}_{i,n}]$ ,  $i = 1, \dots, N$ 。

输出 高阶广义奇异值分解结果  $\mathbf{V}$  和  $\mathbf{U}_i$ ,  $\Sigma_i$ ,  $i = 1, \dots, N$ 。

#### 5.5.4 应用

多麦克风在离散时间  $k$  采集的含噪声语音信号可以用观测模型

$$\mathbf{y}[k] = \mathbf{x}[k] + \mathbf{v}[k]$$

描述。式中,  $\mathbf{x}[k]$  和  $\mathbf{v}[k]$  分别为语音信号向量和加性噪声向量。若令  $\mathbf{R}_{yy} = E\{\mathbf{y}[k]\mathbf{y}^T[k]\}$ ,  $\mathbf{R}_{vv} = E\{\mathbf{v}[k]\mathbf{v}^T[k]\}$  分别代表观测数据的自相关矩阵和加性噪声的自相关矩阵, 则可以对它们进行联合对角化, 即

$$\begin{aligned} \mathbf{R}_{yy} &= \mathbf{Q} \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_m^2) \mathbf{Q}^T \\ \mathbf{R}_{vv} &= \mathbf{Q} \text{diag}(\eta_1^2, \eta_2^2, \dots, \eta_m^2) \mathbf{Q}^T \end{aligned} \quad (5.5.25)$$

2002 年, Doclo 与 Moonen<sup>[137]</sup> 证明了, 为了实现多麦克风语音增强, 使均方误差最小的最优滤波器为

$$\mathbf{W}[k] = \mathbf{R}_{yy}^{-1}[k] \mathbf{R}_{xx}[k] = \mathbf{R}_{yy}^{-1}[k] (\mathbf{R}_{yy}[k] - \mathbf{R}_{vv}[k]) \quad (5.5.26)$$

$$= \mathbf{Q}^{-T} \text{diag} \left( 1 - \frac{\sigma_1^2}{\eta_1^2}, 1 - \frac{\sigma_2^2}{\eta_2^2}, \dots, 1 - \frac{\sigma_m^2}{\eta_m^2} \right) \mathbf{Q} \quad (5.5.27)$$

构造  $p \times m$  观测数据矩阵  $\mathbf{Y}[k]$  和  $q \times m$  加性噪声矩阵  $\mathbf{V}[k']$  如下

$$\mathbf{Y}[k] = \begin{bmatrix} \mathbf{y}^T[k-p+1] \\ \vdots \\ \mathbf{y}^T[k-1] \\ \mathbf{y}^T[k] \end{bmatrix}, \quad \mathbf{V}[k'] = \begin{bmatrix} \mathbf{v}^T[k'-q+1] \\ \vdots \\ \mathbf{v}^T[k'-1] \\ \mathbf{v}^T[k'] \end{bmatrix} \quad (5.5.28)$$

式中,  $\mathbf{V}[k']$  是平时在无语音信号时测量得到的相同环境下的加性噪声数据矩阵。于是, 只要计算矩阵束  $(\mathbf{Y}[k], \mathbf{V}[k'])$  的广义奇异值分解, 得到  $\mathbf{Q}$  和广义奇异值  $\sigma_i/\eta_i$ , 即可直接获得最优滤波器  $\mathbf{W}^T[k]$ 。理论和仿真结果表明, 这种基于广义奇异值分解的最优滤波器显示了波束形成器的空间指向特性, 有着很好的多麦克风语音增强效果。

在信息恢复系统中, 降维技术对处理大批量数据是至关重要的。为此, 数据的低维表示必须是全部文本数据一个很好的逼近。模式识别通过使类内散布最小、类间散布最大, 对数据进行聚类。然而, 这种识别分析要求类内散布矩阵或类间散布矩阵必须有一个是非奇异的。但是, 文本数据矩阵往往不能满足这一要求。2003 年, Howland 等人<sup>[243]</sup> 证

明了, 利用广义奇异值分解, 无论文本数据维数多少, 都可以实现聚类; 并且直接使用数据矩阵的广义奇异值分解, 还可避免使用散布矩阵带来的数值稳定性问题。基于广义奇异值分解, 文献 [243] 提出了聚类文本数据的降维方法, 这种方法能够有效保持文本数据的结构。

在生物信息学中, 广义奇异值分解已应用于两个不同生物体的基因组范围内表达数据集的比较分析<sup>[15]</sup>, 而高阶广义奇异值分解被应用于多种生物全球基因的比较<sup>[412]</sup>。

在模式识别和机器学习中, 判别分析 (discriminant analysis) 广泛用于抽取保留类型可分性的特征, 而广义奇异值分解已被推广到判别分析<sup>[244]</sup>。

## 5.6 矩阵完备

在应用科学和工程的领域 (例如图像、语音和视频处理、生物信息学、网络搜索、电子商务等) 中, 数据集往往是高维的, 其维数甚至达到百万数量级。发现和利用高维数据中的低维结构, 在这些应用中显得尤为重要。另外, 在这些领域的诸多应用中, 人们只能够观测到一个数据矩阵的少量元素, 但希望只根据这些有限的信息, 能够猜测出未看到的大量元素, 从而恢复一个未知的低秩矩阵或近似低秩矩阵。此外, 高维数据矩阵的元素还可能含有很大的观测误差, 甚至遭到篡改。于是, 从数学和应用科学的角度就提出了一个重要的问题: 如何从少数 (可能被污染的) 矩阵元素精确地恢复一个低秩的矩阵, 而且还能够纠正可能的观测误差甚至错误。这个问题称为矩阵完备 (matrix completion)。

矩阵完备是 Candès 与 Recht 于 2009 年提出的<sup>[86]</sup>, 近几年已经成为矩阵分析与应用的一个非常活跃的研究热点。本节将介绍矩阵完备的主要理论、实现算法及应用。

### 5.6.1 矩阵恢复与矩阵分解

假定已知数据已排列成一高维数据或者样本矩阵  $\mathbf{D} \in \mathbb{R}^{m \times n}$ 。估计一低维子空间的问题称为低秩矩阵逼近, 系求一低秩矩阵  $\mathbf{A}$ , 使得  $\mathbf{D}$  与  $\mathbf{A}$  的差异  $\mathbf{E} = \mathbf{D} - \mathbf{A}$  最小化

$$\min_{\mathbf{A}} \|\mathbf{E}\|_F^2 = \|\mathbf{D} - \mathbf{A}\|_F^2 \quad \text{subject to } r \leq \text{rank}(\mathbf{A}) \quad (5.6.1)$$

其中,  $r \ll \min\{m, n\}$ 。求解低秩矩阵逼近问题的著名方法是主分量分析 (principal component analysis, PCA)<sup>[240, 151, 256]</sup>: 计算数据矩阵的奇异值分解  $\mathbf{D} = \mathbf{U}\Sigma\mathbf{V}^T$ , 确定  $r$  个主 (要) 奇异值以及与之对应的主 (要) 左奇异向量  $\mathbf{u}_1, \dots, \mathbf{u}_r$ 。这  $r$  个主左奇异向量反映待分析或识别的模式或信号的主要特征, 故基于主左奇异向量的模式与信号分析称为主分量分析。

当两个或多个信号或图像的主要特征 (轮廓) 相同, 而它们的次要特征 (细节) 不同时, 则需要使用与那些次 (要) 奇异值对应的次 (要) 左奇异向量  $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$  作为特征向量, 进行模式或信号的分析。这样的方法称为次分量分析 (minimal component analysis, MCA)。

若观测数据被独立同分布 (i.i.d.) 高斯噪声污染时, 经典 PCA 能够给出低秩矩阵逼近问题的最优解。然而, 若观测数据被高度污染 (例如误差或扰动矩阵  $\mathbf{E}$  的很多元素取值很大, 或者虽然大多数元素的数值不大, 但少数元素为异常值), 则经典 PCA 估计的矩阵  $\hat{\mathbf{A}}$  将远离真实的低秩矩阵  $\mathbf{A}$ 。此时, 必须考虑矩阵恢复问题。

矩阵恢复 (matrix recovery) 就是当低秩矩阵  $\mathbf{A}$  的观测或样本矩阵  $\mathbf{D} = \mathbf{A} + \mathbf{E}$  的某些元素被严重损坏时, 能够自动识别被损坏的元素, 精确地恢复原低秩矩阵  $\mathbf{A}$ 。

在工程和应用科学的许多领域 (例如机器学习、控制、系统工程、信号处理、模式识别和计算机视觉) 中, 将一个数据矩阵分解为一个低秩矩阵与一个误差 (或扰动) 矩阵之和, 旨在恢复低秩矩阵是远远不够的, 而是需要将一个数据矩阵  $\mathbf{D}$  分解为一个低秩矩阵  $\mathbf{A}$  与一个稀疏矩阵  $\mathbf{E}$  之和  $\mathbf{D} = \mathbf{A} + \mathbf{E}$ , 并且希望同时恢复低秩矩阵与稀疏矩阵。矩阵的这类分解称为低秩与稀疏矩阵分解。

下面是低秩与稀疏矩阵分解的几个具有代表性的典型应用领域<sup>[100, 88]</sup>:

(1) 图形化建模 (graphical modeling) 在许多应用中, 由于少量的特征因子可以解释大多数的观测统计量, 所以大协方差矩阵常用低秩矩阵逼近 (如 PCA)。另一类模型是图形化模型<sup>[305]</sup>, 协方差矩阵的逆矩阵 (也称信息矩阵) 假定相对于某个图形是稀疏的。因此, 在统计模型选择设定中, 常将数据矩阵分解为低秩矩阵与稀疏矩阵之和, 以分别刻画未被观测的隐匿变量的作用和图形化模型。

(2) 组合系统辨识 (composite system identification) 在系统辨识中, 常使用由低秩 Hankel 矩阵与稀疏 Hankel 矩阵之和表示一组合系统。其中, 稀疏 Hankel 矩阵对应于一个具有稀疏冲激响应的线性时不变系统, 而低秩 Hankel 矩阵则对应于具有小的模型阶数的最小实现系统。

(3) 视频监控 (video surveillance) 给定一监控视频帧的序列, 通常需要辨识脱离背景的活动。利用图像帧与帧之间的相似性, 若将视频帧的序列排列成一个数据矩阵  $\mathbf{D}$ , 则可以将图像分解为一个低秩矩阵  $\mathbf{A}$  与一个稀疏矩阵  $\mathbf{E}$  之和, 从而达到背景与前景的分离。其中, 低秩矩阵反映图像每帧之间的相似部分 (对应于平稳的图像背景), 而稀疏矩阵则反映图像中的特有部分 (对应于前景中的运动体)。

(4) 人脸识别 (face recognition) 由于不同光照下的凸朗伯表面的图像张成一个低维子空间<sup>[33]</sup>, 所以低维模型对图像数据是最有效的。特别地, 人脸的图像可以利用低维子空间很好地逼近。因此, 准确地恢复这一子空间在人脸识别和校准中显得尤为关键。然而, 实际的人脸图像通常会遭受阴影、高光 (镜面反射或者亮度的饱和度) 等污染或者部分图像被损。因此, 通过矩阵分解, 可以将人脸图像中的阴影、高光或被损坏部分去除。

(5) 潜在语义检索 (latent semantic indexing, LSI) 网络搜索引擎经常需要检索巨大的文档集之中的某些内容。常用的方法是潜在语义检索<sup>[136]</sup>, 其基本思想是将一个词与文档的相关性 (如频次) 进行编码, 作为文档-词矩阵 (document-versus-term matrix)  $\mathbf{D}$  的元素。传统的 PCA (或 SVD) 将矩阵  $\mathbf{D}$  分解为一个低秩矩阵与一残差矩阵之和, 但残差矩阵不一定是稀疏的。如果将  $\mathbf{D}$  分解为低秩矩阵  $\mathbf{A}$  与稀疏矩阵  $\mathbf{E}$  之和, 则  $\mathbf{A}$  就可以

捕获在所有文档中共同使用的常见单词，而  $E$  则能够捕获每一个文档与其他文档相区别的少数几个关键词。

(6) 评分与协同筛选 (ranking and collaborative filtering) 预测用户的喜好在电子商务和广告中越来越重要。商家现在经常定期收集各种产品 (例如书籍、电影、游戏和网络工具等) 的排名。所谓评分和协同筛选，就是利用用户对某些产品的不完整评分，预测任何一个特定用户对任何一个产品的喜好。评分与协同筛选最有名的实现是 Netflix 推荐系统，其目的是对未公演的电影作评分预测。在这种情况下，数据矩阵是不完整的：只观测到一部分矩阵元素，大部分矩阵元素需要精确预测和补充。这样一个数学问题称为低秩矩阵的完备。由于数据采集过程往往缺乏控制，有时甚至是特定条件下采集的，所以少部分可用的评分可能误差比较大，甚至有可能遭到篡改。因此，需要在完备低秩矩阵的同时，还能够矫正错误。

### 5.6.2 矩阵完备及其可辨识性

秩最小化问题的数学模型为

$$\min \operatorname{rank}(\mathbf{D}) \quad \text{subject to } \mathbf{D} \in \mathcal{C} \quad (5.6.2)$$

式中  $\mathcal{C}$  为一凸集。

令  $\mathbf{D} \in \mathbb{R}^{m \times n}$  是一高维不完全数据矩阵：只已知或者观测到少量的矩阵元素，这些已知元素或样本元素的指标集为  $\Omega$ ，即只有  $D_{ij}, (i, j) \in \Omega$  是已知或被观测的矩阵元素。

指标集的支撑也称基数，记为  $|\Omega|$ ，表示样本元素的个数；而样本数目与矩阵维数之比  $p = \frac{|\Omega|}{mn}$  称为数据矩阵的样本密度。在一些典型应用 (例如 Netflix 推荐系统等) 中，样本密度往往只有 1% 甚至更低。

从一个不完全数据矩阵恢复一个低秩矩阵和一个稀疏矩阵的数学问题称为矩阵完备 (matrix completion)。注意，矩阵完备包含有两个目的：

- (1) 矩阵填充 补充或者填补低秩矩阵的所有未知元素。
- (2) 矩阵纠正 对某些误差大甚至被篡改的样本矩阵元素进行纠正。

表 5.6.1 比较了矩阵完备与低秩矩阵逼近之间的区别。

表 5.6.1 矩阵完备与低秩矩阵逼近的比较

方法	矩阵完备	低秩矩阵逼近
已知	数据矩阵的少数元素	整个数据矩阵
目的	重构整个高维矩阵	抽取高维矩阵的低秩特性
问题	完备能力受非相干性和采样率限制	可逼近性受真实秩和采样方法限制

低秩矩阵完备的数学问题是：恢复一个低秩  $\operatorname{rank}(\mathbf{X}) \ll \min\{m, n\}$  的矩阵  $\mathbf{X}$ ，使得

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \operatorname{rank}(\mathbf{X}) \quad \text{subject to } \mathcal{P}_{\Omega}(\mathbf{X}) = \mathcal{P}_{\Omega}(\mathbf{D}) \quad (5.6.3)$$

式中,  $\mathcal{P}_\Omega : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$  是到指标集  $\Omega$  的投影

$$[\mathcal{P}_\Omega(\mathbf{D})]_{ij} = \begin{cases} D_{ij}, & (i, j) \in \Omega \\ 0, & \text{否则} \end{cases} \quad (5.6.4)$$

若令  $\mathbf{D} = \mathbf{A} + \mathbf{E}$  中的“噪声矩阵”  $\mathbf{E} \in \mathbb{R}^{m \times n}$  的平均平方元素值  $\sigma^2 = \frac{1}{mn} \|\mathbf{E}\|_2^2$ , 并且被恢复的矩阵为  $\mathbf{X}$ , 则矩阵恢复分为以下四种类型<sup>[174]</sup>:

- (1) 低秩矩阵  $\mathbf{A}$  的精确恢复 (exact recovery)  $\hat{\mathbf{X}} = \mathbf{A}^*$ 。
- (2) 低秩矩阵  $\mathbf{A}$  的近精确恢复 (near-exact recovery)  $\frac{1}{mn} \|\hat{\mathbf{X}} - \mathbf{A}^*\|_2^2 \leq \epsilon \cdot \sigma^2$ 。
- (3) 低秩矩阵  $\mathbf{A}$  的逼近恢复 (approximate recovery)  $\frac{1}{mn} \|\hat{\mathbf{X}} - \mathbf{A}^*\|_2^2 \leq \epsilon \cdot \text{scale}(\mathbf{A})$ , 其中  $\text{scale}(\mathbf{A}) = \frac{1}{mn} \|\mathbf{A}\|_{\text{F}}^2$  (平均平方元素幅值) 或者  $\text{scale}(\mathbf{A}) = \|\mathbf{A}\|_\infty = \max\{A_{ij}\}$  (最大元素幅值)。
- (4) 样本矩阵  $\mathbf{D}$  的逼近恢复  $\frac{1}{mn} \|\hat{\mathbf{X}} - \mathbf{D}\|_2^2 \leq \sigma^2 + \epsilon \cdot \text{scale}(\mathbf{A})$ 。

注意, 精确和近精确恢复要求低秩矩阵  $\mathbf{A}$  满足严格的非相干性假设, 而一般的低秩矩阵很难满足这一条件。因此, 对于不满足严格的非相干条件的低秩矩阵, 只能实现逼近恢复。

式 (5.6.3) 所示矩阵完备问题是一个 NP 难题 (NP-hard problem)。为了使矩阵完备问题可解, 必须将矩阵的秩最小化予以松弛。这一松弛与矩阵的 Schatten 范数密切相关。

若  $\mathbf{U} \in \mathbb{C}^{m \times m}$  和  $\mathbf{V} \in \mathbb{C}^{n \times n}$  是两个酉矩阵, 满足  $\|\mathbf{A}\| = \|\mathbf{UAV}\|$  的范数称为酉不变范数 (unitarily invariant norms)<sup>[505, 314]</sup>。

令矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$  有奇异值分解  $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H$ 。显然,  $\|\mathbf{A}\| = \|\mathbf{U}^H\mathbf{A}\mathbf{V}\| = \|\Sigma\|$  是一酉不变范数。

令  $\boldsymbol{\sigma} = [\sigma_1, \dots, \sigma_k]^T$ ,  $k = \min\{m, n\}$  表示全部奇异值组成的向量, 则酉不变范数  $\|\mathbf{A}\| = \|\Sigma\|$  可以用奇异值向量的范数形式定义:  $\|\mathbf{A}\| = \|\boldsymbol{\sigma}\|$ 。特别地, 称

$$\|\mathbf{A}\|_p = \|\boldsymbol{\sigma}\|_p = \left( \sum_{i=1}^{\min\{m,n\}} \sigma_i^p \right)^{1/p} \quad (5.6.5)$$

是矩阵  $\mathbf{A}$  的 Schatten  $p$  范数。

最常用的 Schatten 范数是  $p = 1, 2, \infty$  三种情况:

(1)  $p = 1$  时的 Schatten 范数称为核范数 (nuclear norm), 定义为一矩阵的所有奇异值之和

$$\|\mathbf{A}\|_* = \sum_{i=1}^{\min\{m,n\}} \sigma_i = \text{tr}(\sqrt{\mathbf{A}^H \mathbf{A}}) \quad (5.6.6)$$

式中,  $\mathbf{B} = \sqrt{\mathbf{A}^H \mathbf{A}}$  满足  $\mathbf{B}^H \mathbf{B} = \mathbf{A}^H \mathbf{A}$ 。

(2)  $p = 2$  时的 Schatten 范数与 Frobenius 范数等价

$$\|\mathbf{A}\|_2 = \|\mathbf{A}\|_{\text{F}} = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2} = \sqrt{\text{tr}(\mathbf{A}^H \mathbf{A})} = \sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \quad (5.6.7)$$

(3)  $p = \infty$  时的 Schatten 范数与诱导  $l_2$  范数(谱范数)相同, 即  $\|\mathbf{A}\|_\infty = \sigma_{\max}(\mathbf{A})$ 。

因此, 核范数、Frobenius 范数和谱范数都是酉不变范数。

下面是精确地恢复低秩矩阵  $\mathbf{A}$  和稀疏矩阵  $\mathbf{E}$  的两个要点:

(1) 大多数的低秩矩阵都可以由样本元素的集合精确恢复, 这些集合甚至可以只有少得惊人的元素数目。

(2) 秩  $r$  的矩阵  $\mathbf{M} \in \mathbb{R}^{n \times n}$  可以通过求解下列优化问题完好地恢复

$$\min \|\mathbf{X}\|_* \quad \text{subject to } X_{ij} = M_{ij}, \quad (i, j) \in \Omega \quad (5.6.8)$$

只要样本元素的个数

$$m \geq Cn^{6/5}r \log n \quad (5.6.9)$$

对某个正的常数  $C$  成立。式 (5.6.8) 中,  $\|\mathbf{X}\|_* = \sum_i \sigma_i(\mathbf{X})$  表示矩阵的核范数即矩阵所有奇异值之和。

令  $\mathbf{P}_\Omega$  是到矩阵  $\mathbf{X}$  的列空间的正交投影矩阵

$$\mathbf{P}_\Omega = \mathbf{X}(\mathbf{X}^T \mathbf{X})^\dagger \mathbf{X}^T, \quad \mathbf{X} \in \Omega$$

则有  $\mathbf{P}_\Omega \mathbf{X} = \mathbf{X}$ ,  $\mathbf{X} \in \Omega$ , 相应的元素形式为

$$[\mathbf{P}_\Omega \mathbf{X}]_{ij} = \begin{cases} X_{ij}, & X_{ij} \in \Omega \\ 0, & X_{ij} \notin \Omega \end{cases} \quad (5.6.10)$$

于是, 核范数最小化问题式 (5.6.8) 可以等价写作

$$\min \|\mathbf{X}\|_* \quad \text{subject to } \mathbf{P}_\Omega \mathbf{X} = \mathbf{P}_\Omega \mathbf{M} \quad (5.6.11)$$

为了求解矩阵完备问题, Candes 等人<sup>[88]</sup> 提出了稳健主分量分析 (robust principal component analysis) 法: 将 NP 难题的秩最小化松弛为核范数的最小化, 利用主分量追踪 (principal component pursuit), 通过求解约束最小化问题

$$\min_{\mathbf{A}, \mathbf{E}} f(\mathbf{A}, \mathbf{E}) = \|\mathbf{A}\|_* + \lambda \|\mathbf{E}\|_1 \quad \text{subject to } \mathbf{D} = \mathbf{A} + \mathbf{E} \quad (5.6.12)$$

从数据矩阵  $\mathbf{D} \in \mathbb{R}^{m \times n}$  恢复一个未知的低秩矩阵  $\mathbf{A}$  与一个未知的稀疏矩阵  $\mathbf{E}$ 。式中,  $\|\mathbf{A}\|_* = \sum_i^{\min\{m, n\}} \sigma_i(\mathbf{A})$  表示矩阵的核范数即所有奇异值之和, 反映低秩矩阵的代价或者成本;  $\|\mathbf{E}\|_1 = \sum_{i=1}^m \sum_{j=1}^n |E_{ij}|$  为矩阵  $\mathbf{E}$  的所有元素的绝对值之和, 描述稀疏矩阵的代价; 常数  $\lambda > 0$  的作用是平衡低秩要求与稀疏要求之间的矛盾。

在没有关于稀疏模式与/或矩阵秩的附加信息的情况下, 矩阵分解  $\mathbf{D} = \mathbf{A} + \mathbf{E}$  无疑是一个病态问题, 存在着低秩与稀疏之间的不确定性, 从而引发下面两个可辨识性问题:

(1) 低秩矩阵本身有可能是非常稀疏的。

(2) 稀疏矩阵的非零元素有可能只集中在矩阵的某个列, 该列元素就有可能否定低秩矩阵的对应列的元素, 从而改变低秩矩阵的秩。

为了解决低秩与稀疏之间的上述两种不确定性, Chandrasekaran 等人<sup>[100]</sup>于 2011 年提出了秩-稀疏非相干性 (rank-sparsity incoherence) 条件。

令  $\mathbf{L} \in \mathbb{R}^{n \times n}$  是一低秩矩阵, 即  $\text{rank}(\mathbf{L}) \leq k$ 。定义秩约束矩阵集合为

$$\mathcal{P}(\mathbf{L}) = \{\mathbf{L} \in \mathbb{R}^{n \times n} \mid \text{rank}(\mathbf{L}) \leq k\} \quad (5.6.13)$$

若  $\mathbf{L} = \mathbf{U}\Sigma\mathbf{V}^T$  是  $n \times n$  矩阵  $\mathbf{L}$  的奇异值分解, 其中  $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n \times k}, k = \text{rank}(\mathbf{L})$ , 则所有矩阵  $\mathbf{U}\mathbf{X}^T + \mathbf{V}\mathbf{Y}^T$  的集合称为矩阵  $\mathbf{M}$  相对于秩约束矩阵  $\mathcal{P}(\mathbf{L})$  在  $\mathbf{L}$  的切空间 (tangent space), 记作  $T(\mathbf{L})$ , 即有

$$T(\mathbf{L}) = \{\mathbf{U}\mathbf{X}^T + \mathbf{V}\mathbf{Y}^T \mid \mathbf{X}, \mathbf{Y} \in \mathbb{R}^{n \times k}\} \quad (5.6.14)$$

显然,  $\mathbf{L} \in T(\mathbf{L})$ , 并且  $T(\mathbf{L})$  是  $\mathbb{R}^{n \times n}$  内的一个子空间。

定义系数

$$\xi(\mathbf{L}) \stackrel{\text{def}}{=} \max_{\mathbf{N} \in T(\mathbf{L}), \|\mathbf{N}\|_{\text{spec}} \leq 1} \|\mathbf{N}\|_{\infty} \quad (5.6.15)$$

其中,  $\|\mathbf{N}\|_{\text{spec}}$  是矩阵  $\mathbf{N}$  的谱范数即最大奇异值, 而  $\|\mathbf{N}\|_{\infty}$  为矩阵元素的最大绝对值。因此, 系数  $\xi(\mathbf{L})$  描述秩约束矩阵集合中秩小于或者等于  $k$  的所有低秩矩阵的最大绝对值元素。因此, 若  $\xi(\mathbf{L})$  小, 则意味着所有低秩矩阵  $\mathbf{L}$  的元素的最大绝对值都比较小, 故  $\mathbf{L}$  不可能是非常稀疏的。

另外, 令  $m$  为一正整数,  $\Omega(m)$  表示所有满足稀疏度条件  $\|\mathbf{S}\|_1 \leq m$  的所有稀疏矩阵的集合, 即

$$\Omega(m) = \{\mathbf{S} \in \mathbb{R}^{n \times n} \mid \|\mathbf{S}\|_1 \leq m\} \quad (5.6.16)$$

若定义系数

$$\mu(\mathbf{S}) \stackrel{\text{def}}{=} \max_{\mathbf{N} \in \Omega(m), \|\mathbf{N}\|_{\infty} \leq 1} \|\mathbf{N}\|_{\text{spec}} \quad (5.6.17)$$

则系数  $\mu(\mathbf{S})$  代表稀疏度小于或者等于  $m$  的所有稀疏矩阵的最大奇异值。显然, 若系数  $\mu(\mathbf{S})$  小, 则所有稀疏矩阵的奇异值都小, 因此它们都不可能是低秩矩阵。

综上所述, 对于同一个矩阵  $\mathbf{M} \in \mathbb{R}^{n \times n}$ , 系数  $\xi(\mathbf{M})$  和  $\mu(\mathbf{M})$  不可能同时都小。文献 [100] 已证明: 对于任何非零  $n \times n$  矩阵  $\mathbf{M}$ , 其秩和稀疏度满足以下不确定性原理

$$\xi(\mathbf{M})\mu(\mathbf{M}) \geq 1 \quad (5.6.18)$$

文献 [100] 进一步证明: 若真实的矩阵分解  $\mathbf{D} = \mathbf{A}^* + \mathbf{E}^*$ , 并且  $(\hat{\mathbf{A}}, \hat{\mathbf{E}})$  是矩阵分解问题式 (5.6.12) 的解

$$(\hat{\mathbf{A}}, \hat{\mathbf{E}}) = \arg \min_{\mathbf{A}, \mathbf{E}} \|\mathbf{A}\|_* + \lambda \|\mathbf{E}\|_1 \quad \text{subject to } \mathbf{D} = \mathbf{A} + \mathbf{E}$$

则  $(\hat{\mathbf{A}}, \hat{\mathbf{E}}) = (\mathbf{A}^*, \mathbf{E}^*)$  是矩阵分解问题式 (5.6.12) 的唯一最优解, 若系数  $\xi(\mathbf{A}^*)$  和  $\mu(\mathbf{E}^*)$  满足不等式

$$\xi(\mathbf{A}^*)\mu(\mathbf{E}^*) \leq \frac{1}{6} \quad (5.6.19)$$

并且 Lagrangian 乘子  $\lambda$  的取值范围满足

$$\lambda \in \left( \frac{\xi(\mathbf{A}^*)}{1 - 4\xi(\mathbf{A}^*)\mu(\mathbf{E}^*)}, \frac{1 - 3\xi(\mathbf{A}^*)\mu(\mathbf{E}^*)}{\mu(\mathbf{E}^*)} \right) \quad (5.6.20)$$

特别地,  $\lambda = \frac{(3\xi(\mathbf{A}^*))^p}{(2\mu(\mathbf{E}^*))^{1-p}}$  (其中  $p \in [0, 1]$ ) 总是位于上述取值范围, 因而总能够保证真实低秩矩阵  $\mathbf{A}^*$  和真实稀疏矩阵  $\mathbf{E}^*$  的精确恢复。

### 5.6.3 矩阵完备的奇异值阈值化法

考虑低秩矩阵  $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$  的截尾奇异值分解

$$\mathbf{Y} = \mathbf{U}\Sigma\mathbf{V}^T, \quad \Sigma = \text{diag}(\sigma_1, \dots, \sigma_r) \quad (5.6.21)$$

式中,  $r = \text{rank}(\mathbf{Y}) \ll \min\{n_1, n_2\}$ ,  $\mathbf{U} \in \mathbb{R}^{n_1 \times r}$ ,  $\mathbf{V} \in \mathbb{R}^{n_2 \times r}$ 。

令  $\tau \geq 0$ , 则

$$\mathcal{D}_\tau(\mathbf{Y}) = \mathbf{U}\mathcal{D}_\tau(\Sigma)\mathbf{V}^T \quad (5.6.22)$$

称为矩阵  $\mathbf{Y}$  的奇异值阈值化 (singular value thresholding, SVT), 其中

$$\mathcal{D}_\tau(\Sigma) = \text{diag}((\sigma_1 - \tau)_+, \dots, (\sigma_r - \tau)_+) \quad (5.6.23)$$

为奇异值的软阈值化, 并且

$$(\sigma_i - \tau)_+ = \begin{cases} \sigma_i - \tau, & \text{若 } \sigma_i > \tau \\ 0, & \text{其他} \end{cases}$$

为软阈值运算。

奇异值阈值化与奇异值分解的关系如下:

- (1) 若阈值  $\tau = 0$ , 则奇异值阈值化退化为截尾奇异值分解式 (5.6.21)。
- (2) 所有奇异值以常数  $\tau > 0$  进行软阈值运算, 并不改变左和右奇异向量矩阵  $\mathbf{U}$  和  $\mathbf{V}$ , 只是改变奇异值的大小。

恰当地选择阈值  $\tau$ , 能够有效地将部分奇异值向零收缩。在这个意义上, 又称奇异值阈值化这一变换为奇异值收缩算子 (singular value shrinkage operator)。需要注意, 如果阈值  $\tau$  比大多数奇异值大, 则奇异值阈值化算子  $\mathcal{D}(\mathbf{Y})$  的秩将比原矩阵  $\mathbf{Y}$  的秩小得多。

奇异值阈值化的关键是如何选择软阈值  $\tau$ ? 下面的定理给出了这个问题的答案。

**定理 5.6.1** <sup>[75]</sup> 对于每一个软阈值  $\tau \geq 0$  和矩阵  $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$ , 奇异值收缩算子式 (5.6.22) 服从

$$\mathcal{D}_\tau(\mathbf{Y}) = \arg \min_{\mathbf{X}} \left\{ \frac{1}{2} \|\mathbf{X} - \mathbf{Y}\|_{\text{F}}^2 + \tau \|\mathbf{X}\|_* \right\} \quad (5.6.24)$$

由定理 5.6.1 知, 对于无约束问题

$$\min_{\mathbf{A}} \tau \|\mathbf{A}\|_* + \frac{1}{2} \|\mathbf{A} - \mathbf{D}\|_{\text{F}}^2 \quad (5.6.25)$$

由于目标函数  $\|\mathbf{A}\|_*$  和  $\frac{1}{2}\|\mathbf{A} - \mathbf{D}\|_F^2$  分别是严格凸函数, 所以上述矩阵完备问题存在唯一最优解, 并由已知数据矩阵  $\mathbf{D}$  的奇异值阈值化直接给出

$$\hat{\mathbf{A}} = \mathcal{D}_\tau(\mathbf{D}) = \mathbf{U}\mathcal{D}_\tau(\Sigma)\mathbf{V}^H \quad (5.6.26)$$

因此, 问题是如何将一个矩阵分解问题变换为式 (5.6.25) 所示的规范形式。下面是两个典型的应用例子。

### 1. 矩阵完备问题

考虑矩阵完备问题

$$\min \frac{1}{2}\|\mathbf{X}\|_F^2 + \tau\|\mathbf{X}\|_* \quad \text{subject to } \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{D}) \quad (5.6.27)$$

使用 Lagrangian 乘子法, 并注意到  $\lambda^T c = \langle \Lambda, \mathbf{C} \rangle = (\text{vec}(\Lambda))^T \text{vec}(\mathbf{C})$ , 其中  $\text{vec}(\Lambda) = \lambda$  为 Lagrangian 乘子矩阵, 且  $\text{vec}(\mathbf{C}) = \mathbf{c}$ 。于是, Lagrangian 目标函数

$$\begin{aligned} L(\mathbf{X}, \Lambda) &= \frac{1}{2}\|\mathbf{X}\|_F^2 + \tau\|\mathbf{X}\|_* + \lambda^T \text{vec}(\mathcal{P}_\Omega(\mathbf{X} - \mathbf{D})) \\ &= \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2 + \langle \Lambda, \mathcal{P}_\Omega(\mathbf{X} - \mathbf{D}) \rangle \end{aligned} \quad (5.6.28)$$

注意到

$$\arg \min_{\mathbf{X}} \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2 + \langle \Lambda, \mathcal{P}_\Omega(\mathbf{X} - \mathbf{D}) \rangle = \arg \min_{\mathbf{X}} \|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathcal{P}_\Omega(\mathbf{Y})\|_F^2$$

由定理 5.6.1 立即知

$$\mathbf{X}_k = \mathcal{D}_\tau(\mathcal{P}_\Omega(\Lambda_{k-1})) = \mathcal{D}_\tau(\mathbf{Y}_{k-1}) \quad (5.6.29)$$

因为  $\Lambda_k = \mathcal{P}_\Omega(\Lambda_k), \forall k \geq 0$ 。另由式 (5.6.28) 易知, Lagrangian 函数在  $\Lambda$  点的梯度

$$\frac{\partial L(\mathbf{X}, \Lambda)}{\partial \Lambda^T} = \mathcal{P}_\Omega(\mathbf{X} - \mathbf{D})$$

于是,  $\Lambda_k$  更新的梯度下降法为

$$\Lambda_k = \Lambda_{k-1} + \mu \mathcal{P}_\Omega(\mathbf{D} - \mathbf{X}_k) \quad (5.6.30)$$

其中  $\mu$  为步长。

式 (5.6.29) 和式 (5.6.30) 一起给出迭代序列<sup>[75]</sup>

$$\begin{cases} \mathbf{X}_k = \mathcal{D}_\tau(\Lambda_{k-1}) \\ \Lambda_k = \Lambda_{k-1} + \mu \mathcal{P}_\Omega(\mathbf{D} - \mathbf{X}_k) \end{cases}$$

上述迭代为线性化的 Bregman 迭代, 它是 Uzawa 算法<sup>[24]</sup>的特例。

上述迭代序列具有以下特点:

- (1) 稀疏性 对每一个  $k \geq 0$ ,  $\Lambda_k$  在  $\Omega$  以外的元素都等于零, 因此  $\mathbf{Y}_k$  为稀疏矩阵。
- (2) 低秩性 矩阵  $\mathbf{X}_k$  具有低秩。由于只需要存储主要的特征因子, 因此算法需要的存储小。

## 2. 仿射约束的矩阵完备问题

对于仿射约束的矩阵完备问题

$$\min \tau \|X\|_* + \frac{1}{2} \|X\|_F^2 \quad \text{subject to } A(X) = b \quad (5.6.31)$$

由于 Lagrangian 函数

$$L(X, \lambda) = \tau \|X\|_* + \frac{1}{2} \|X\|_F^2 + \langle \lambda, b - A(X) \rangle$$

所以迭代序列为<sup>[75]</sup>

$$\begin{cases} X_k = D_\tau(A^*(\lambda_{k-1})) \\ \lambda_k = \lambda_{k-1} + \mu(b - A(X_k)) \end{cases}$$

其中  $A^*$  是满足  $A^*A = I$  的仿射变换  $A$  的伴随算子。

Cai 与 Osher 针对奇异值阈值化的实现，提出了无须进行奇异值分解的快速奇异值阈值化算法<sup>[76]</sup>。

数据矩阵  $D$  的奇异值分解  $D = U \Sigma V^T$  可以分解为两部分之和

$$D = U \begin{bmatrix} (\sigma_1 - \tau)_+ & & 0 \\ & \ddots & \\ 0 & & (\sigma_r - \tau)_+ \end{bmatrix} V^T + U \begin{bmatrix} \min\{\sigma_1, \tau\} & & 0 \\ & \ddots & \\ 0 & & \min\{\sigma_r, \tau\} \end{bmatrix} V^T$$

或者写作

$$D = D_\tau(D) + P_\tau(D) \quad (5.6.32)$$

式中

$$P_\tau(D) = U \begin{bmatrix} \min\{\sigma_1, \tau\} & & 0 \\ & \ddots & \\ 0 & & \min\{\sigma_r, \tau\} \end{bmatrix} V^T \quad (5.6.33)$$

表示数据矩阵  $D$  到 2-范数球的投影。

若数据矩阵的极式分解为  $D = WZ$ ，其中  $W$  是酉矩阵， $Z$  为对称的非负定矩阵，则由于 Frobenius 范数是酉不变范数，故有<sup>[76]</sup>

$$P_\tau(D) = \arg \min_{\|A\|_2 \leq \tau} \|D - A\|_F = W \arg \min_{\|A\|_2 \leq \tau} \|Z - A\|_F = WP_\tau(Z) \quad (5.6.34)$$

式中

$$P_\tau(Z) = \arg \min_{\|A\|_2 \leq \tau} \|Z - A\|_F \quad (5.6.35)$$

于是，奇异值阈值化  $D_\tau(D) = D - WP_\tau(Z)$  的计算转换成投影  $P_\tau(Z)$  的计算，从而构成了快速奇异值阈值化的三步算法<sup>[76]</sup>：

- (1) 利用文献 [230, 231] 的方法计算数据矩阵的极式分解  $D = WZ$ 。
- (2) 计算投影  $P_\tau(Z) = \arg \min_{\|A\|_2 \leq \tau} \|Z - A\|_F$ 。
- (3) 令  $D_\tau(D) = D - WP_\tau(Z)$ 。

下面分别是极式分解和投影计算的算法。

#### 算法 5.6.1 $D = WZ$ 的极式分解算法 [230, 231]

输入 数据矩阵  $D$ 。

输出 极式因子矩阵  $W$  和对称非负定矩阵  $Z$ 。

步骤 1 若数据矩阵奇异或者为  $m \times n$  非正方矩阵，则使用 QR 分解计算数据矩阵的完全正交分解

$$D = U \begin{bmatrix} R & O \\ O & O \end{bmatrix} Q$$

其中， $U \in \mathbb{R}^{m \times m}$ ,  $Q \in \mathbb{R}^{n \times n}$  均为正交矩阵， $R \in \mathbb{R}^{r \times r}$  为可逆的上三角矩阵， $O$  为零矩阵。取  $W_0 = R$ ；否则，取  $W_0 = D$ 。

步骤 2 对  $k = 0, 1, \dots, k_{\max}$  (最大迭代次数)，执行以下运算：

(1) 计算  $W_k^{-T}$ ；

(2) 令  $\gamma_k = \left( \frac{\|W_k^{-1}\|_1 \|W_k^{-1}\|_\infty}{\|W_k\|_1 \|W_k\|_\infty} \right)^{1/4}$ ；

(3) 令  $W_{k+1} = \frac{1}{2}(\gamma_k W_k + \gamma_k^{-1} W_k^{-T})$ ；

(4) 检验  $W_{k+1}$  是否收敛？若  $\|W_{k+1} - W_k\|_F \leq \epsilon \|D\|_F$ ，则停止迭代，并输出  $W = W_{k+1}$  和  $Z = W_{k+1}^T D$ ；否则，令  $k \leftarrow k + 1$ ，并返回 (1)，重复以上运算，直至  $W_{k+1}$  收敛或达到最大迭代次数  $k_{\max}$ 。

#### 算法 5.6.2 投影 $P_\tau(Z)$ 的算法 [76]

输入 对称非负定矩阵  $Z$ ，实数  $\delta$ 。

输出 投影  $P = P_\tau(Z)$ 。

步骤 1 计算  $Z$  位于  $[\tau(1 - \delta), \tau(1 + \delta)]$  区间的特征值矩阵  $\Sigma_1$  及与这些特征值对应的特征向量  $V_1$ 。

步骤 2 令  $Z \leftarrow Z - U_1 \Sigma_1 V_1^T$  及  $P_k = O$ 。

步骤 3 对  $k = 0, 1, \dots, k_{\max}$  (最大迭代次数)，执行以下运算：

(1) 计算

$$P_{k+1} = \frac{1}{2}P_k + \frac{1}{4}Z + \frac{3\tau}{4}I - (2P_k - Z - \tau I)^{-1} \left( \tau P_k - \frac{1}{4}Z^2 - \frac{3\tau^2}{4}I \right)$$

(2) 检验  $P_{k+1}$  是否收敛？若  $\|P_{k+1} - P_k\|_F \leq \epsilon \|Z\|_F$ ，则停止迭代，并输出

$$P = P_{k+1} - V_1 P_\tau(\Sigma_1) V_1^T$$

否则，令  $k \leftarrow k + 1$ ，并重复 (1) 和 (2)，直至  $P_{k+1}$  收敛或者达到最大迭代次数  $k_{\max}$ 。

## 本章小结

本章首先分析了单个矩阵的(普通)奇异值分解、奇异值的性质以及奇异值分解的数值计算。然后,以两个矩阵作为对象,介绍了奇异值分解的两种推广——乘积奇异值分解和广义奇异值分解。又以多个矩阵为对象,介绍了高阶广义奇异值分解。本章还分别介绍了奇异值分解和广义奇异值分解的应用。

作为奇异值分解的最新发展,本章最后介绍了矩阵完备和奇异值阈值化。

## 习 题

### 5.1 已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$

通过计算  $\mathbf{AA}^T$  和  $\mathbf{A}^T\mathbf{A}$  的特征值和特征向量,求矩阵  $\mathbf{A}$  的奇异值分解。

### 5.2 分别计算矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & -1 \\ 3 & -3 \\ -3 & 3 \end{bmatrix} \quad \text{和} \quad \mathbf{A} = \begin{bmatrix} 3 & 4 & 5 \\ 2 & 1 & 7 \end{bmatrix}$$

的奇异值分解。

### 5.3 已知矩阵

$$\mathbf{A} = \begin{bmatrix} -149 & -50 & -154 \\ 537 & 180 & 546 \\ -27 & 9 & -25 \end{bmatrix}$$

求  $\mathbf{A}$  的奇异值以及与最小奇异值  $\sigma_1$  相对应的左、右奇异向量。

5.4 令  $\mathbf{A} = \mathbf{x}\mathbf{p}^H + \mathbf{y}\mathbf{q}^H$ , 其中,  $\mathbf{x} \perp \mathbf{y}$  和  $\mathbf{p}^\perp \mathbf{q}$ 。求矩阵  $\mathbf{A}$  的 Frobenius 范数  $\|\mathbf{A}\|_F$ 。  
(提示: 计算  $\mathbf{A}^H\mathbf{A}$ , 并求  $\mathbf{A}$  的奇异值。)

5.5 已知  $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H$  是矩阵  $\mathbf{A}$  的奇异值分解, 矩阵  $\mathbf{A}^H$  的奇异值与  $\mathbf{A}$  的奇异值有何关系?

5.6 证明: 若  $\mathbf{A}$  为正方矩阵, 则  $|\det(\mathbf{A})|$  等于  $\mathbf{A}$  的奇异值之积。

5.7 假定  $\mathbf{A}$  为可逆矩阵, 求  $\mathbf{A}^{-1}$  的奇异值分解。

5.8 证明: 若  $\mathbf{A}$  为  $n \times n$  正定矩阵, 则  $\mathbf{A}$  的奇异值与  $\mathbf{A}$  的特征值相同。

5.9 令  $\mathbf{A}$  为  $m \times n$  矩阵, 且  $\mathbf{P}$  为  $m \times m$  正交矩阵。证明  $\mathbf{PA}$  与  $\mathbf{A}$  的奇异值相同。矩阵  $\mathbf{PA}$  与  $\mathbf{A}$  的左、右奇异向量有何关系?

5.10 令  $\mathbf{A}$  是一个  $m \times n$  矩阵, 并且  $\lambda_1, \dots, \lambda_n$  是矩阵  $\mathbf{A}^T\mathbf{A}$  的特征值, 相对应的特征向量为  $\mathbf{u}_1, \dots, \mathbf{u}_n$ 。证明  $\mathbf{A}$  的奇异值  $\sigma_i$  等于范数  $\|\mathbf{Au}_i\|$ , 即  $\sigma_i = \|\mathbf{Au}_i\|$ ,  $i = 1, \dots, n$ 。

**5.11** 令  $\lambda_1, \lambda_2, \dots, \lambda_n$  和  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$  分别是矩阵  $\mathbf{A}^T \mathbf{A}$  的特征值和特征向量。假定矩阵  $\mathbf{A}$  有  $r$  个非零的奇异值, 证明  $\{\mathbf{A}\mathbf{u}_1, \mathbf{A}\mathbf{u}_2, \dots, \mathbf{A}\mathbf{u}_r\}$  是列空间  $\text{Col}(\mathbf{A})$  的一组正交基, 并且  $\text{rank}(\mathbf{A}) = r$ 。

**5.12** 令  $\mathbf{B}, \mathbf{C} \in \mathbb{R}^{m \times n}$ , 求复矩阵  $\mathbf{A} = \mathbf{B} + j\mathbf{C}$  与实分块矩阵  $\begin{bmatrix} \mathbf{B} & -\mathbf{C} \\ \mathbf{C} & \mathbf{B} \end{bmatrix}$  的奇异值和奇异向量之间的关系。

**5.13** 用矩阵  $\mathbf{A} \in \mathbb{R}^{m \times n}$  ( $m \geq n$ ) 的奇异向量表示  $\begin{bmatrix} \mathbf{O} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{O} \end{bmatrix}$  的特征向量。

**5.14** 利用 MATLAB 函数  $[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{X})$  求解方程  $\mathbf{Ax} = \mathbf{b}$ , 其中

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 3 & 1 & 3 \\ 1 & 0 & 1 \\ 2 & 2 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 4 \\ 3 \\ 2 \end{bmatrix}$$

**5.15** 假定计算机仿真的观测数据为

$$x(n) = \sqrt{20} \sin(2\pi 0.2n) + \sqrt{2} \sin(2\pi 0.215n) + w(n)$$

产生, 其中,  $w(n)$  是一高斯白噪声, 其均值为 0, 方差为 1, 并取  $n = 1, 2, \dots, 128$ 。试针对 10 次独立的仿真实验数据, 分别确定自相关矩阵

$$\mathbf{R} = \begin{bmatrix} r(0) & r(-1) & \cdots & r(-2p) \\ r(1) & r(0) & \cdots & r(-2p+1) \\ \vdots & \vdots & \ddots & \vdots \\ r(M) & r(M-1) & \cdots & r(M-2p) \end{bmatrix}$$

的有效秩。式中,  $r(k) = \frac{1}{128} \sum_{i=1}^{128-k} x(i)x(i+k)$  表示观测信号的样本自相关函数 (未知的观测数据皆令其等于 0), 并取  $M = 50$ ,  $p = 10$ 。

**5.16** [198] 使用奇异值分解证明: 若  $\mathbf{A} \in \mathbb{R}^{m \times n}$  ( $m \geq n$ ), 则存在  $\mathbf{Q} \in \mathbb{R}^{m \times n}$  和  $\mathbf{P} \in \mathbb{R}^{n \times n}$ , 使得  $\mathbf{A} = \mathbf{QP}$ , 其中,  $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}_n$ , 并且  $\mathbf{P}$  是对称的和非负定的。这一分解有时称为极分解 (polar decomposition), 因为它与复数分解  $z = |z|e^{j\arg(z)}$  类似。

# 第6章 矩阵方程求解

在众多科学与工程学科，如物理、化学工程、统计学、经济学、生物学、信号处理、自动控制、系统理论、医学和军事工程等中，许多问题都可用数学建模成矩阵方程  $\mathbf{Ax} = \mathbf{b}$ 。根据数据向量  $\mathbf{b} \in \mathbb{R}^{m \times 1}$  和数据矩阵  $\mathbf{A} \in \mathbb{R}^{m \times n}$  的不同，矩阵方程有以下三种主要类型（参见图 6.0.1）：

- (1) 超定矩阵方程  $m > n$ ，并且数据矩阵  $\mathbf{A}$  和数据向量  $\mathbf{b}$  均已知，其中之一或者二者可能存在误差或者干扰。
- (2) 盲矩阵方程 仅数据向量  $\mathbf{b}$  已知，数据矩阵  $\mathbf{A}$  未知。
- (3) 欠定稀疏矩阵方程  $m < n$ ，数据矩阵  $\mathbf{A}$  和数据向量  $\mathbf{b}$  均已知，但未知向量  $\mathbf{x}$  为稀疏向量。

本章将依次详细讨论上述矩阵方程的求解方法。

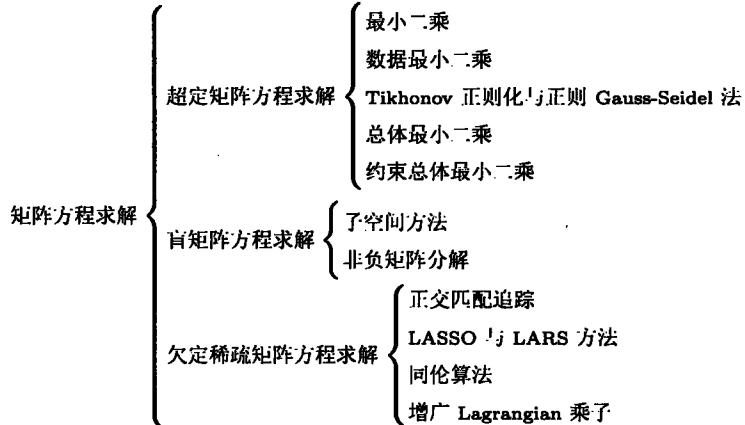


图 6.0.1 矩阵方程的三种主要类型

## 6.1 最小二乘方法

线性参数估计问题广泛存在于科学与技术问题中，最小二乘方法是最常用的线性参数估计方法。实际上，早在高斯的年代，最小二乘方法就用来对平面上的点拟合线，对高维空间的点拟合超平面。本节分析最小二乘方法的工作原理、最优解的条件及其不足。

### 6.1.1 普通最小二乘

考虑超定矩阵方程  $\mathbf{Ax} = \mathbf{b}$ ，其中  $\mathbf{b}$  为  $m \times 1$  数据向量， $\mathbf{A}$  为  $m \times n$  数据矩阵，并且  $m > n$ 。

假定数据向量存在加性观测误差或噪声, 即  $\mathbf{b} = \mathbf{b}_0 + \mathbf{e}$ , 其中  $\mathbf{b}_0$  和  $\mathbf{e}$  分别是无误差的数据向量和误差向量。

为了抵制误差对矩阵方程求解的影响, 引入一校正向量  $\Delta\mathbf{b}$ , 并用它去“扰动”有误差的数据向量  $\mathbf{b}$ 。我们的目标是, 使校正项  $\Delta\mathbf{b}$  “尽可能小”, 同时通过强令  $\mathbf{A}\mathbf{x} = \mathbf{b} + \Delta\mathbf{b}$  补偿存在于数据向量  $\mathbf{b}$  中的不确定性(噪声或误差), 使得  $\mathbf{b} + \Delta\mathbf{b} = \mathbf{b}_0 + \mathbf{e} + \Delta\mathbf{b} \rightarrow \mathbf{b}_0$ , 从而实现

$$\mathbf{A}\mathbf{x} = \mathbf{b} + \Delta\mathbf{b} \implies \mathbf{A}\mathbf{x} = \mathbf{b}_0 \quad (6.1.1)$$

的转换。也就是说, 如果直接选择校正向量  $\Delta\mathbf{b} = \mathbf{A}\mathbf{x} - \mathbf{b}$ , 并且使校正向量“尽可能小”, 则可以实现无误差的矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}_0$  的求解。

矩阵方程的这一求解思想可以用下面的优化问题进行描述

$$\min_{\mathbf{x}} \|\Delta\mathbf{b}\|^2 = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = (\mathbf{A}\mathbf{x} - \mathbf{b})^T(\mathbf{A}\mathbf{x} - \mathbf{b}) \quad (6.1.2)$$

这一方法称为普通最小二乘 (ordinary least squares, OLS) 法, 常简称为最小二乘法。

事实上, 校正向量  $\Delta\mathbf{b} = \mathbf{A}\mathbf{x} - \mathbf{b}$  恰好是矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  两边的误差向量。因此, 最小二乘方法的核心思想是求出的解向量  $\mathbf{x}$  能够使矩阵方程两边的误差平方和最小化。于是, 矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的普通最小二乘解为

$$\hat{\mathbf{x}}_{\text{LS}} = \arg \min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \quad (6.1.3)$$

为了推导  $\mathbf{x}$  的解析解, 展开式 (6.1.2) 得

$$\phi = \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{b}$$

求  $\phi$  相对于  $\mathbf{x}$  的导数, 并令其结果等于零, 则有

$$\frac{d\phi}{d\mathbf{x}} = 2\mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{A}^T \mathbf{b} = 0$$

也就是说, 解  $\mathbf{x}$  必然满足

$$\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{b} \quad (6.1.4)$$

当  $m \times n$  矩阵  $\mathbf{A}$  具有不同的秩时, 上述方程的解有两种不同的情况。

情况 1 超定方程 ( $m > n$ ) 满列秩, 即  $\text{rank}(\mathbf{A}) = n$ 。

由于  $\mathbf{A}^T \mathbf{A}$  非奇异, 所以方程有唯一的解

$$\mathbf{x}_{\text{LS}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (6.1.5)$$

这恰好就是我们在第 1 章证明过的最小二乘解。在参数估计理论中, 称这种可以唯一确定的未知参数  $\mathbf{x}$  是(唯一)可辨识的。

对于秩亏缺 ( $\text{rank}(\mathbf{A}) < n$ ) 的超定方程, 则最小二乘解为

$$\mathbf{x}_{\text{LS}} = (\mathbf{A}^T \mathbf{A})^\dagger \mathbf{A}^T \mathbf{b} \quad (6.1.6)$$

其中  $\mathbf{B}^\dagger$  代表矩阵  $\mathbf{B}$  的 Moore-Penrose 逆矩阵。

**情况 2 欠定方程  $\text{rank}(\mathbf{A}) = m < n$ 。**

在这种情况下, 由  $\mathbf{x}$  的不同解均得到相同的  $\mathbf{Ax}$  值。显而易见, 虽然数据向量  $\mathbf{b}$  可以提供有关  $\mathbf{Ax}$  的某些信息, 但是无法区别对应于相同  $\mathbf{Ax}$  值的各个不同的未知参数向量  $\mathbf{x}$ 。因此, 称这样的参数向量是不可辨识的。更一般地, 如果某参数的不同值给出在抽样空间上的相同分布, 则称该参数是不可辨识的<sup>[456]</sup>。

### 6.1.2 Gauss-Markov 定理

在参数估计理论中, 称参数向量  $\boldsymbol{\theta}$  的估计  $\hat{\boldsymbol{\theta}}$  为无偏估计, 若它的数学期望值等于真实的未知参数向量, 即  $E\{\hat{\boldsymbol{\theta}}\} = \boldsymbol{\theta}$ 。进一步地, 如果一个无偏估计还具有最小方差, 则称这一无偏估计为最优无偏估计。类似地, 对于数据向量  $\mathbf{b}$  含有加性噪声或者扰动的超定方程  $\mathbf{A}\boldsymbol{\theta} = \mathbf{b} + \mathbf{e}$ , 若最小二乘解  $\hat{\boldsymbol{\theta}}_{\text{LS}}$  的数学期望等于真实参数向量  $\boldsymbol{\theta}$ , 便称最小二乘解是无偏的。如果它还具有最小方差, 则称最小二乘解是最优无偏的。

**定理 6.1.1 (Gauss-Markov 定理)** 考虑线性方程组

$$\mathbf{Ax} = \mathbf{b} + \mathbf{e} \quad (6.1.7)$$

式中,  $m \times n$  矩阵  $\mathbf{A}$  和  $n \times 1$  向量  $\mathbf{x}$  分别为常数矩阵和参数向量;  $\mathbf{b}$  为  $m \times 1$  向量, 它存在随机误差向量  $\mathbf{e} = [e_1, e_2, \dots, e_m]^T$ 。误差向量的均值向量和协方差矩阵分别为

$$E\{\mathbf{e}\} = \mathbf{0}, \quad \text{Cov}(\mathbf{e}) = E\{\mathbf{ee}^H\} = \sigma^2 \mathbf{I}$$

$n \times 1$  参数向量  $\mathbf{x}$  的最优无偏解  $\hat{\mathbf{x}}$  存在, 当且仅当  $\text{rank}(\mathbf{A}) = n$ 。此时, 最优无偏解由最小二乘解

$$\hat{\mathbf{x}}_{\text{LS}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b} \quad (6.1.8)$$

给出, 其方差

$$\text{Var}(\hat{\mathbf{x}}_{\text{LS}}) \leq \text{Var}(\tilde{\mathbf{x}}) \quad (6.1.9)$$

式中,  $\tilde{\mathbf{x}}$  是矩阵方程  $\mathbf{Ax} = \mathbf{b} + \mathbf{e}$  的任何一个其他解。

**证明** 由假设条件  $E\{\mathbf{e}\} = \mathbf{0}$  立即有

$$E\{\mathbf{b}\} = E\{\mathbf{Ax}\} - E\{\mathbf{e}\} = \mathbf{Ax} \quad (1)$$

利用已知条件  $\text{Cov}(\mathbf{e}) = E\{\mathbf{ee}^H\} = \sigma^2 \mathbf{I}$ , 并注意到  $\mathbf{Ax}$  与误差向量  $\mathbf{e}$  统计不相关, 又有

$$E\{\mathbf{bb}^H\} = E\{(\mathbf{Ax} - \mathbf{e})(\mathbf{Ax} - \mathbf{e})^H\} = E\{\mathbf{Axx}^H \mathbf{A}^H\} + E\{\mathbf{ee}^H\} = \mathbf{Axx}^H \mathbf{A}^H + \sigma^2 \mathbf{I} \quad (2)$$

由于  $\text{rank}(\mathbf{A}) = n$ , 矩阵乘积  $\mathbf{A}^H \mathbf{A}$  非奇异, 因此有

$$E\{\hat{\mathbf{x}}_{\text{LS}}\} = E\{(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}\} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H E\{\mathbf{b}\} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{Ax} = \mathbf{x}$$

即最小二乘解  $\hat{x}_{LS} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$  是矩阵方程  $\mathbf{Ax} = \mathbf{b} + \mathbf{e}$  的无偏解。

下面证明  $\hat{x}_{LS}$  具有最小方差。为此，假定  $\mathbf{x}$  还有另外一个候补解  $\tilde{\mathbf{x}}$ ，则可以将它表示成

$$\tilde{\mathbf{x}} = \hat{x}_{LS} + \mathbf{C}\mathbf{b} + \mathbf{d}$$

式中， $\mathbf{C}$  和  $\mathbf{d}$  分别为常数矩阵和常数向量。解  $\tilde{\mathbf{x}}$  是无偏的，即

$$\mathbb{E}\{\tilde{\mathbf{x}}\} = \mathbb{E}\{\hat{x}_{LS}\} + \mathbb{E}\{\mathbf{C}\mathbf{b}\} + \mathbf{d} = \mathbf{x} + \mathbf{C}\mathbf{A}\mathbf{x} + \mathbf{d} = \mathbf{x} + \mathbf{C}\mathbf{A}\mathbf{x} + \mathbf{d}, \quad \forall \mathbf{x}$$

当且仅当

$$\mathbf{C}\mathbf{A} = \mathbf{O} \text{ (零矩阵),} \quad \mathbf{d} = \mathbf{0} \quad (3)$$

利用这两个无偏约束条件，易知  $\mathbb{E}\{\mathbf{C}\mathbf{b}\} = \mathbf{C}\mathbb{E}\{\mathbf{b}\} = \mathbf{C}\mathbf{A}\boldsymbol{\theta} = \mathbf{0}$ 。于是，得

$$\begin{aligned} \text{cov}(\tilde{\mathbf{x}}) &= \text{Cov}(\hat{x}_{LS} + \mathbf{C}\mathbf{b}) = \mathbb{E}\{[(\hat{x}_{LS} - \mathbf{x}) + \mathbf{C}\mathbf{b}] [(\hat{x}_{LS} - \mathbf{x}) + \mathbf{C}\mathbf{b}]^H\} \\ &= \text{Cov}(\hat{x}_{LS}) + \mathbb{E}\{(\hat{x}_{LS} - \mathbf{x})(\mathbf{C}\mathbf{b})^H\} + \mathbb{E}\{\mathbf{C}\mathbf{b}(\hat{x}_{LS} - \mathbf{x})^H\} + \mathbb{E}\{\mathbf{C}\mathbf{b}\mathbf{b}^H\mathbf{C}^H\} \end{aligned} \quad (4)$$

但是，由式(1)~式(3)，易知

$$\begin{aligned} \mathbb{E}\{(\hat{x}_{LS} - \mathbf{x})(\mathbf{C}\mathbf{b})^H\} &= \mathbb{E}\{(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b} \mathbf{b}^H \mathbf{C}^H\} - \mathbb{E}\{\mathbf{x} \mathbf{b}^H \mathbf{C}^H\} \\ &= (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbb{E}\{\mathbf{b} \mathbf{b}^H\} \mathbf{C}^H - \mathbf{x} \mathbb{E}\{\mathbf{b}^H\} \mathbf{C}^H \\ &= (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H (\mathbf{A} \mathbf{x} \mathbf{x}^H \mathbf{A}^H + \sigma^2 \mathbf{I}) \mathbf{C}^H - \mathbf{x} \mathbf{x}^H \mathbf{A}^H \mathbf{C}^H \\ &= \mathbf{O} \\ \mathbb{E}\{\mathbf{C}\mathbf{b}(\hat{x}_{LS} - \mathbf{x})^H\} &= [\mathbb{E}\{(\hat{x}_{LS} - \mathbf{x})(\mathbf{C}\mathbf{b})^H\}]^H = \mathbf{O} \\ \mathbb{E}\{\mathbf{C}\mathbf{b}\mathbf{b}^H\mathbf{C}^H\} &= \mathbf{C}\mathbb{E}\{\mathbf{b} \mathbf{b}^H\} \mathbf{C}^H = \mathbf{C}(\mathbf{A} \mathbf{x} \mathbf{x}^H \mathbf{A}^H + \sigma^2 \mathbf{I}) \mathbf{C}^H = \sigma^2 \mathbf{C} \mathbf{C}^H \end{aligned}$$

故式(4)可化简为

$$\text{Cov}(\tilde{\mathbf{x}}) = \text{Cov}(\hat{x}_{LS}) + \sigma^2 \mathbf{C} \mathbf{C}^H \quad (5)$$

上式取迹函数后，利用迹函数的性质  $\text{tr}(\mathbf{A} + \mathbf{B}) = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{B})$ ，并注意到对于具有零均值向量的随机向量  $\mathbf{x}$ ，有  $\text{tr}[\text{Cov}(\mathbf{x})] = \text{Var}(\mathbf{x})$ ，即可将式(5)改写作

$$\text{Var}(\tilde{\mathbf{x}}) = \text{Var}(\hat{x}_{LS}) + \sigma^2 \text{tr}(\mathbf{C} \mathbf{C}^H) \geq \text{Var}(\hat{x}_{LS})$$

因为  $\text{tr}(\mathbf{C} \mathbf{C}^H) \geq 0$ 。这就证明了  $\hat{x}_{LS}$  具有最小方差，从而是最优无偏解。 ■

注意，定理 6.1.1 的条件  $\text{Cov}(\mathbf{e}) = \sigma^2 \mathbf{I}$  意味着加性误差向量  $\mathbf{e}$  的各个分量互不相关，并且具有相同的方差  $\sigma^2$ 。只有在这种情况下，最小二乘解才是无偏的和最优的。这正是 Gauss-Markov 定理的物理含义所在。

### 6.1.3 普通最小二乘解与最大似然解的等价性

若加性误差向量  $\mathbf{e} = [e_1, \dots, e_m]^T$  为独立同分布的复高斯随机向量，则由式 (1.5.35) 知，其概率密度函数为

$$f(\mathbf{e}) = \frac{1}{\pi^m |\boldsymbol{\Gamma}_e|} \exp [-(\mathbf{e} - \boldsymbol{\mu}_e)^H \boldsymbol{\Gamma}_e^{-1} (\mathbf{e} - \boldsymbol{\mu}_e)] \quad (6.1.10)$$

式中， $|\boldsymbol{\Gamma}_e|$  表示协方差矩阵  $\boldsymbol{\Gamma}_e = \text{diag}(\sigma_1^2, \dots, \sigma_m^2)$  的行列式，即有  $|\boldsymbol{\Gamma}_e| = \sigma_1^2 \cdots \sigma_m^2$ 。

在 Gauss-Markov 定理的条件（即误差向量的各个独立同分布的高斯随机变量均具有零均值和相同方差  $\sigma^2$ ）下，加性误差向量的概率密度函数简化为

$$f(\mathbf{e}) = \frac{1}{(\pi\sigma^2)^m} \exp \left( -\frac{1}{\sigma^2} \mathbf{e}^H \mathbf{e} \right) = \frac{1}{(\pi\sigma^2)^m} \exp \left( -\frac{1}{\sigma^2} \|\mathbf{e}\|_2^2 \right) \quad (6.1.11)$$

其似然函数

$$L(\mathbf{e}) = \log f(\mathbf{e}) = -\frac{1}{\pi^m \sigma^{2(m+1)}} \|\mathbf{e}\|_2^2 = -\frac{1}{\pi^m \sigma^{2(m+1)}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \quad (6.1.12)$$

于是，矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的最大似然解

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \max_{\mathbf{x}} \frac{-1}{\pi^m \sigma^{2(m+1)}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = \hat{\mathbf{x}}_{\text{LS}} \quad (6.1.13)$$

即是说，在 Gauss-Markov 定理的条件下，矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的最大似然解  $\hat{\mathbf{x}}_{\text{ML}}$  与最小二乘解  $\hat{\mathbf{x}}_{\text{LS}}$  等价。

容易看出，当误差向量  $\mathbf{e}$  为零均值的高斯随机向量，但其元素具有不同方差时，由于协方差矩阵  $\boldsymbol{\Gamma}_e$  不等于  $\sigma^2 \mathbf{I}$ ，所以这种情况下的最大似然解

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \max_{\mathbf{x}} \frac{1}{\pi^m \sigma_1^2 \cdots \sigma_m^2} \exp (-\mathbf{e}^H \boldsymbol{\Gamma}_e^{-1} \mathbf{e})$$

将不可能等于最小二乘解  $\hat{\mathbf{x}}_{\text{LS}}$ ，即最小二乘解不再是最优的。

### 6.1.4 数据最小二乘

考虑超定矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$ ，但与普通最小二乘问题不同，这里假定数据向量  $\mathbf{b}$  无观测误差或噪声，只有数据矩阵  $\mathbf{A} = \mathbf{A}_0 + \mathbf{E}$  有观测误差或噪声，并且误差矩阵  $\mathbf{E}$  的每一个误差元素服从零均值、等方差的独立高斯分布。

考虑用校正矩阵  $\Delta \mathbf{A}$  干扰有误差的数据矩阵  $\mathbf{A}$ ，使得  $\mathbf{A} + \Delta \mathbf{A} = \mathbf{A}_0 + \mathbf{E} + \Delta \mathbf{A} \rightarrow \mathbf{A}_0$ 。与普通最小二乘方法相类似，通过强令  $(\mathbf{A} + \Delta \mathbf{A})\mathbf{x} = \mathbf{b}$ ，补偿数据矩阵中存在的误差矩阵，实现

$$(\mathbf{A} + \Delta \mathbf{A})\mathbf{x} = \mathbf{b} \implies \mathbf{A}_0\mathbf{x} = \mathbf{b}$$

此时， $\mathbf{x}$  的最优解为

$$\hat{\mathbf{x}}_{\text{DLS}} = \arg \min_{\mathbf{x}} \|\Delta \mathbf{A}\|_2^2 \quad \text{subject to } \mathbf{b} \in \text{Range}(\mathbf{A} - \Delta \mathbf{A}) \quad (6.1.14)$$

这一方法称为数据最小二乘 (data least squares, DLS) 法。其中, 约束条件  $\mathbf{b} \in \text{Range}(\mathbf{A} + \Delta\mathbf{A})$  意味着, 对于每一个给定的精确数据向量  $\mathbf{b} \in \mathbb{C}^m$  和有误差的数据矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$ , 总可以找到一个向量  $\mathbf{x} \in \mathbb{C}^n$ , 使得  $(\mathbf{A} + \Delta\mathbf{A})\mathbf{x} = \mathbf{b}$ 。因此, 两个约束条件  $\mathbf{b} \in \text{Range}(\mathbf{A} + \Delta\mathbf{A})$  和  $(\mathbf{A} + \Delta\mathbf{A})\mathbf{x} = \mathbf{b}$  的表述等价。

利用 Lagrange 乘子法, 可以将约束的数据最小二乘问题式 (6.1.14) 转变成无约束优化问题

$$\min L(\mathbf{x}) = \text{tr}(\Delta\mathbf{A}(\Delta\mathbf{A})^H) + \lambda^H(\mathbf{Ax} + \Delta\mathbf{Ax} - \mathbf{b}) \quad (6.1.15)$$

令共轭梯度矩阵  $\partial L(\mathbf{x})/\partial \Delta\mathbf{A}^H$  等于零矩阵, 立即得  $\Delta\mathbf{A} = -\lambda\mathbf{x}^H$ 。将  $\Delta\mathbf{A} = -\lambda\mathbf{x}^H$  代入约束条件  $(\mathbf{A} + \Delta\mathbf{A})\mathbf{x} = \mathbf{b}$ , 即有  $\lambda = \frac{\mathbf{Ax} - \mathbf{b}}{\mathbf{x}^H \mathbf{x}}$ , 从而有  $\Delta\mathbf{A} = -\frac{(\mathbf{Ax} - \mathbf{b})\mathbf{x}^H}{\mathbf{x}^H \mathbf{x}}$ 。于是, 原目标函数

$$J(\mathbf{x}) = \|\Delta\mathbf{A}\|_2^2 = \text{tr}(\Delta\mathbf{A}(\Delta\mathbf{A})^H) = \text{tr}\left(\frac{(\mathbf{Ax} - \mathbf{b})\mathbf{x}^H}{\mathbf{x}^H \mathbf{x}} \frac{\mathbf{x}(\mathbf{Ax} - \mathbf{b})^H}{\mathbf{x}^H \mathbf{x}}\right)$$

利用迹函数性质  $\text{tr}(\mathbf{BC}) = \text{tr}(\mathbf{CB})$ , 立即有

$$J(\mathbf{x}) = \frac{(\mathbf{Ax} - \mathbf{b})^H(\mathbf{Ax} - \mathbf{b})}{\mathbf{x}^H \mathbf{x}} \quad (6.1.16)$$

由此得

$$\hat{\mathbf{x}}_{\text{DLS}} = \arg \min_{\mathbf{x}} \frac{(\mathbf{Ax} - \mathbf{b})^H(\mathbf{Ax} - \mathbf{b})}{\mathbf{x}^H \mathbf{x}} \quad (6.1.17)$$

这就是超定矩阵方程  $\mathbf{Ax} = \mathbf{b}$  的数据最小二乘解。

## 6.2 Tikhonov 正则化与正则 Gauss-Seidel 法

在求解超定矩阵方程  $\mathbf{A}_{m \times n} \mathbf{x}_{n \times 1} = \mathbf{b}_{m \times 1}$  (其中  $m > n$ ) 的时候, 普通最小二乘法和数据最小二乘法有两个基本的假设: ① 数据矩阵  $\mathbf{A}$  非奇异或者满列秩; ② 数据向量  $\mathbf{b}$  或者数据矩阵  $\mathbf{A}$  存在加性噪声或误差。

本节介绍数据矩阵秩亏缺或者存在误差时超定矩阵方程求解的正则化方法。

### 6.2.1 Tikhonov 正则化

如 6.1 节所述, 当  $m = n$ , 并且  $\mathbf{A}$  非奇异时, 矩阵方程  $\mathbf{Ax} = \mathbf{b}$  的解为  $\hat{\mathbf{x}} = \mathbf{A}^{-1}\mathbf{b}$ ; 而当  $m > n$ , 并且  $\mathbf{A}_{m \times n}$  满列秩时, 方程组的解由  $\hat{\mathbf{x}}_{\text{LS}} = \mathbf{A}^\dagger \mathbf{b} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$  给出。

问题是, 在工程应用中, 矩阵  $\mathbf{A}$  往往是秩亏缺的。在这些情况下, 解  $\hat{\mathbf{x}} = \mathbf{A}^{-1}\mathbf{b}$  或者  $\hat{\mathbf{x}}_{\text{LS}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$  要么发散, 要么即使存在, 也只是对  $\mathbf{x}$  的毫无意义的质量很差的逼近。即便幸运的话, 碰巧找到一个对  $\mathbf{x}$  的合理逼近, 但是误差估计值  $\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{A}\hat{\mathbf{x}} - \mathbf{b}\|$  或  $\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \|\mathbf{A}^\dagger\| \|\mathbf{A}\hat{\mathbf{x}} - \mathbf{b}\|$  也令人大失所望<sup>[367]</sup>。观察易知, 问题出在数据矩阵  $\mathbf{A}$  的协方差矩阵  $\mathbf{A}^H \mathbf{A}$  的求逆。

作为最小二乘方法的代价函数  $\frac{1}{2}\|\mathbf{Ax} - \mathbf{b}\|_2^2$  的改进, Tikhonov<sup>[472]</sup> 于 1963 年提出使用正则化最小二乘代价函数

$$J(\mathbf{x}) = \frac{1}{2} (\|\mathbf{Ax} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_2^2) \quad (6.2.1)$$

式中  $\lambda > 0$  称为正则化参数 (regularization parameters)。

代价函数关于变元  $\mathbf{x}$  的共轭梯度

$$\frac{\partial J(\mathbf{x})}{\partial \mathbf{x}^H} = \frac{\partial}{\partial \mathbf{x}^H} ((\mathbf{Ax} - \mathbf{b})^H(\mathbf{Ax} - \mathbf{b}) + \lambda \mathbf{x}^H \mathbf{x}) = \mathbf{A}^H \mathbf{A} \mathbf{x} - \mathbf{A}^H \mathbf{b} + \lambda \mathbf{x}$$

令  $\frac{\partial J(\mathbf{x})}{\partial \mathbf{x}^H} = \mathbf{0}$ , 立即得解

$$\hat{\mathbf{x}}_{\text{Tik}} = (\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b} \quad (6.2.2)$$

这种使用  $(\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1}$  代替协方差矩阵的直接求逆  $(\mathbf{A}^H \mathbf{A})^{-1}$  的方法常称为 Tikhonov 正则化 (Tikhonov regularization), 或简称正则化方法 (regularized method)。在信号处理与图像处理的文献中, 有时把正则化法称为松弛法 (relaxation method)。

Tikhonov 正则化方法的本质是: 通过对秩亏缺的矩阵  $\mathbf{A}$  的协方差矩阵  $\mathbf{A}^H \mathbf{A}$  的每一个对角元素加一个很小的扰动  $\lambda$ , 使得奇异的协方差矩阵  $\mathbf{A}^H \mathbf{A}$  的求逆变成非奇异矩阵  $\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I}$  的求逆, 从而大大改善求解秩亏缺矩阵方程  $\mathbf{Ax} = \mathbf{b}$  的数值稳定性。

显然, 若数据矩阵  $\mathbf{A}$  满列秩, 但存在误差或者噪声时, 就需要采用与 Tikhonov 正则化相反的做法, 对被噪声污染的协方差矩阵  $\mathbf{A}^H \mathbf{A}$  加一个很小的负扰动矩阵  $-\lambda \mathbf{I}$ , 使  $\mathbf{A}^H \mathbf{A}$  去干扰。这种使用负的正则化参数  $-\lambda$  的 Tikhonov 正则化称为反正则化方法 (deregularized method), 其解由

$$\hat{\mathbf{x}} = (\mathbf{A}^H \mathbf{A} - \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b} \quad (6.2.3)$$

给出。6.3 节将介绍的总体最小二乘方法就是一种典型的反正则化方法。

如前所述, 正则化参数  $\lambda$  应该取很小的值, 这样既可以使  $(\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1}$  更好地逼近  $(\mathbf{A}^H \mathbf{A})^{-1}$ , 又可避免  $\mathbf{A}^H \mathbf{A}$  的奇异, 从而使 Tikhonov 正则法可以明显改进奇异和病态方程组求解的数值稳定性。这是因为, 矩阵  $\mathbf{A}^H \mathbf{A}$  是半正定的, 故  $\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I}$  的特征值位于区间  $[\lambda, \lambda + \|\mathbf{A}\|_F^2]$ , 这使得条件数

$$\text{cond}(\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I}) \leq (\lambda + \|\mathbf{A}_F\|^2)/\lambda \quad (6.2.4)$$

相比  $\mathbf{A}^H \mathbf{A}$  的条件数  $\leq \infty$ , 有明显的改善。

为了进一步改善 Tikhonov 正则化求解奇异和病态方程组的结果, 可以使用迭代 Tikhonov 正则化 (iterated Tikhonov regularization)<sup>[367]</sup>: 令初始解向量  $\mathbf{x}_0 = \mathbf{0}$  和初始残差向量  $\mathbf{r}_0 = \mathbf{b}$ , 则解向量和残差向量可以用以下迭代公式进行更新

$$\left. \begin{aligned} \mathbf{x}_k &= \mathbf{x}_{k-1} + (\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{r}_{k-1} \\ \mathbf{r}_k &= \mathbf{b} - \mathbf{A} \mathbf{x}_k \end{aligned} \right\}, \quad k = 1, 2, \dots \quad (6.2.5)$$

令  $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H$  是矩阵  $\mathbf{A}$  的奇异值分解，则  $\mathbf{A}^H\mathbf{A} = \mathbf{V}\Sigma^2\mathbf{V}^H$ ，从而得普通最小二乘解和 Tikhonov 正则化解分别为

$$\hat{\mathbf{x}}_{LS} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{b} = \mathbf{V}\Sigma^{-1}\mathbf{U}^H\mathbf{b} \quad (6.2.6)$$

$$\hat{\mathbf{x}}_{Tik} = (\mathbf{A}^H\mathbf{A} + \sigma_{min}^2\mathbf{I})^{-1}\mathbf{A}^H\mathbf{b} = \mathbf{V}(\Sigma^2 + \sigma_{min}^2\mathbf{I})^{-1}\Sigma\mathbf{U}^H\mathbf{b} \quad (6.2.7)$$

其中  $\sigma_{min}$  是矩阵  $\mathbf{A}$  最小的非零奇异值。若矩阵  $\mathbf{A}$  奇异或者病态，即  $\sigma_n = 0$ ，则由于  $\Sigma^{-1}$  的对角元素中会出现  $\frac{1}{\sigma_n} = \infty$  的项，从而导致最小二乘解发散。相反，基于奇异值分解的 Tikhonov 正则化解  $\hat{\mathbf{x}}_{Tik}$  却具有很好的数值稳定性，因为

$$(\Sigma^2 + \delta^2\mathbf{I})^{-1}\Sigma = \text{diag}\left(\frac{\sigma_1}{\sigma_1^2 + \sigma_{min}^2}, \dots, \frac{\sigma_n}{\sigma_n^2 + \sigma_{min}^2}\right) \quad (6.2.8)$$

的对角元素介于 0 和  $\sigma_1/(\sigma_1^2 + \sigma_{min}^2)$  之间。

当正则化参数  $\lambda$  在定义区间  $[0, \infty)$  内变化时，一个正则化最小二乘问题的解族称为该正则化问题的正则化路径 (regularization path)。

Tikhonov 正则化解具有以下重要性质 [270]：

(1) 线性 Tikhonov 正则化最小二乘问题的解  $\hat{\mathbf{x}}_{Tik} = (\mathbf{A}^H\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^H\mathbf{b}$  是观测数据向量  $\mathbf{b}$  的线性函数。

(2)  $\lambda \rightarrow 0$  时的极限特性 当正则化参数  $\lambda \rightarrow 0$  时，Tikhonov 正则化最小二乘问题的解收敛为普通最小二乘解或 Moore-Penrose 解  $\lim_{\lambda \rightarrow 0} \hat{\mathbf{x}}_{Tik} = \hat{\mathbf{x}}_{LS} = \mathbf{A}^\dagger\mathbf{b} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{b}$ 。解点  $\hat{\mathbf{x}}_{Tik}$  在满足  $\mathbf{A}^H(\mathbf{A}\mathbf{x} - \mathbf{b}) = \mathbf{0}$  的所有可行点中具有最小  $L_2$  范数

$$\hat{\mathbf{x}}_{Tik} = \arg \min_{\mathbf{A}^T(\mathbf{b} - \mathbf{A}\mathbf{x}) = \mathbf{0}} \|\mathbf{x}\|_2 \quad (6.2.9)$$

(3)  $\lambda \rightarrow \infty$  时的极限特性 当  $\lambda \rightarrow \infty$  时，Tikhonov 正则化最小二乘问题的最优解收敛为零向量，即  $\lim_{\lambda \rightarrow \infty} \hat{\mathbf{x}}_{Tik} = \mathbf{0}$ 。

(4) 正则化路径 当正则化参数  $\lambda$  在  $[0, \infty)$  区间变化时，Tikhonov 正则化最小二乘问题的最优解是正则化参数的光滑函数，即当  $\lambda$  减小为零时，最优解收敛为 Moore-Penrose 解；而当  $\lambda$  增大时，最优解收敛为零向量解。

Tikhonov 正则化可以有效防止矩阵  $\mathbf{A}$  秩亏缺时最小二乘解  $\hat{\mathbf{x}}_{LS} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$  的发散，明显改善最小二乘和交替最小二乘算法的收敛性能，因而被广泛应用。

## 6.2.2 正则 Gauss-Seidel 法

令  $X_i \subseteq \mathbb{R}^{n_i}$  是  $n_i$  维列向量  $\mathbf{x}_i$  的可行集。考虑非线性最小化问题

$$\min_{\mathbf{x} \in X} f(\mathbf{x}) = f(\mathbf{x}_1, \dots, \mathbf{x}_m) \quad (6.2.10)$$

式中， $\mathbf{x} \in X = X_1 \times X_2 \times \dots \times X_m \subseteq \mathbb{R}^n$  为闭合的、非空的凸集  $X_i \subseteq \mathbb{R}^{n_i}, i = 1, \dots, m$  的笛卡儿积，并且  $\sum_{i=1}^m n_i = n$ 。

式 (6.2.10) 是一个  $m$  个变元向量耦合在一起的无约束优化问题。求解这类耦合优化问题的有效方法是分块非线性 Gauss-Seidel 法<sup>[45, 209]</sup>, 常简称为 GS 法。

在 GS 法的每一步迭代中, 固定  $m-1$  个变元向量为已知, 对剩下的另一个待优化变元向量进行最小化。这一思想构成了非线性无约束优化问题式 (6.2.10) 的 GS 法的基本框架, 具体步骤如下:

- (1) 初始化  $m-1$  个变元向量  $\mathbf{x}_i, i = 2, \dots, m$ , 并令  $k = 0$ 。
- (2) 求分离的子优化问题的解

$$\mathbf{x}_i^{k+1} = \arg \min_{\mathbf{y} \in X_i} f(\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}, \mathbf{y}, \mathbf{x}_{i+1}^k, \dots, \mathbf{x}_m^k), \quad i = 1, \dots, m \quad (6.2.11)$$

在更新  $\mathbf{x}_i$  的第  $k+1$  步迭代,  $\mathbf{x}_1, \dots, \mathbf{x}_{i-1}$  业已更新为  $\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}$ , 故这些子向量和尚待进行  $k+1$  步迭代的子向量  $\mathbf{x}_{i+1}^k, \dots, \mathbf{x}_m^k$  被固定为已知向量。

(3) 检验  $m$  个变元向量是否均收敛。若收敛, 则输出优化结果  $(\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_m^{k+1})$ ; 否则, 令  $k \leftarrow k+1$ , 返回步骤 (2), 并继续迭代, 直至收敛准则满足为止。

GS 方法有两种主要的变型:

- (1) 分块协同下降法

当  $n$  维变元向量  $\mathbf{x} \in \mathbb{R}^n$  的分块数  $m = n$  时, 分块非线性 GS 方法常称为分块协同下降 (block coordinate descent, BCD) 法<sup>[415, 484]</sup>。

- (2) 交替最小二乘法

若优化问题式 (6.2.10) 的目标函数  $f(\mathbf{x})$  为最小二乘误差函数 (例如  $\|\mathbf{Ax} - \mathbf{b}\|_2^2$ ), 则 GS 法习惯称为交替最小二乘 (alternating least squares, ALS) 法。

**例 6.2.1** 考虑  $m \times n$  已知数据矩阵  $\mathbf{X}$  的满秩分解  $\mathbf{X} = \mathbf{AB}$ , 其中  $m \times r$  矩阵  $\mathbf{A}$  满列秩,  $r \times n$  矩阵  $\mathbf{B}$  满行秩。令矩阵满秩分解的代价函数

$$f(\mathbf{A}, \mathbf{B}) = \frac{1}{2} \|\mathbf{X} - \mathbf{AB}\|_F^2 \quad (6.2.12)$$

交替最小二乘算法首先初始化矩阵  $\mathbf{A}$ 。在第  $k+1$  次迭代中, 由固定的矩阵  $\mathbf{A}_k$ , 即可更新矩阵  $\mathbf{B}$  的最小二乘解

$$\mathbf{B}_{k+1} = (\mathbf{A}_k^T \mathbf{A}_k)^{-1} \mathbf{A}_k^T \mathbf{X} \quad (6.2.13)$$

然后, 由矩阵分解的转置  $\dot{\mathbf{X}}^T = \mathbf{B}^T \mathbf{A}^T$ , 立即又可以更新矩阵  $\mathbf{A}^T$  的最小二乘解

$$\mathbf{A}_{k+1}^T = (\mathbf{B}_{k+1} \mathbf{B}_{k+1}^T)^{-1} \mathbf{B}_{k+1} \mathbf{X}^T \quad (6.2.14)$$

以上两种最小二乘方法交替进行。一旦算法收敛, 即可得到矩阵分解的优化结果。

下面分析 GS 法的收敛性能。为此, 先引入极限点和临界点的概念。

令  $S$  是拓补空间  $X$  的一个子集, 称空间  $X$  内的点  $x$  是子集  $S$  的一个极限点 (limit point), 若  $x$  的每一个邻域至少总含有  $S$  中的一个点 (不包含  $x$  本身)。换言之, 极限点  $x$  是拓补空间  $X$  内可以用  $S$  中的点 (不包含  $x$  本身) 进行“逼近”的点。

导数等于零或者导数不存在的函数曲线上的点在优化问题中起着重要的作用。点  $x$  称为函数  $f(x)$  的一个临界点 (critical point)，若  $x$  位于该函数的定义域，并且函数在该点的导数  $f'(x) = 0$  或者  $f'(x)$  不存在。一个临界点的几何解释是：在曲线上该点的切线要么是水平的，抑或是垂直的，要么根本不存在。

**例 6.2.2** 考虑一实值函数  $f(x) = x^4 - 4x^2$ ，其导数  $f'(x) = 4x^3 - 8x$ ，由  $f'(x) = 0$  立即得到函数  $f(x)$  的三个临界点  $x = 0, -\sqrt{2}$  和  $\sqrt{2}$ 。

上述关于标量变元的极限点和临界点的概念很容易推广到向量变元。

**定义 6.2.1 (极限点)** 称向量  $\bar{x} \in \mathbb{R}^n$  是向量序列  $\{\bar{x}_k\}_{k=1}^{\infty}$  在向量空间  $\mathbb{R}^n$  的一个极限点，若存在  $\{\bar{x}_k\}_{k=1}^{\infty}$  的一个子序列收敛为  $\bar{x}$ 。

**定义 6.2.2 (临界点)** 令  $f : X \rightarrow \mathbb{R}$  (其中  $X \subset \mathbb{R}^n$ ) 是一实值函数，称  $\bar{x} \in \mathbb{R}^n$  是函数  $f(\bar{x})$  的一个临界点，若下列条件满足

$$\mathbf{g}^T(\bar{x})(\mathbf{y} - \bar{x}) \geq 0, \quad \forall \mathbf{y} \in X \quad (6.2.15)$$

其中  $\mathbf{g}^T(\bar{x})$  表示向量函数  $\mathbf{g}(\bar{x})$  的转置，并且  $\mathbf{g}(\bar{x}) = \nabla f(\bar{x})$  表示连续可微分函数  $f(\bar{x})$  在点  $\bar{x}$  的梯度向量，或者  $\mathbf{g}(\bar{x}) \in \partial f(\bar{x})$  表示不可微分的非平滑函数  $f(\bar{x})$  在点  $\bar{x}$  的次梯度向量。若  $X = \mathbb{R}^n$  或者  $\bar{x}$  是  $X$  的内点，则临界点条件式 (6.2.15) 退化为无约束最小化问题  $\min f(\bar{x})$  的平稳点条件  $\nabla f(\bar{x}) = \mathbf{0}$  (对连续可微分的目标函数) 或者  $\mathbf{0} \in \partial f(\bar{x})$  (对非平滑的目标函数)。

令  $\bar{x}^k = (\bar{x}_1^k, \dots, \bar{x}_m^k)$  表示 GS 算法产生的迭代结果，自然希望迭代序列  $\{\bar{x}^k\}_{k=1}^{\infty}$  有极限点，并且每一个极限点都是目标函数  $f$  的一临界点。对于优化问题式 (6.2.10)，GS 算法的这一收敛性能取决于目标函数  $f$  的拟凸性。

**定义 6.2.3 (拟凸函数与严格拟凸函数)** 令  $S$  是一实向量空间的凸子集。函数  $f : S \rightarrow \mathbb{R}$  称为  $S$  内的拟凸函数 (quasiconvex function)，若对每一个向量  $\mathbf{x}, \mathbf{y} \in S$  和常数  $\alpha \in (0, 1)$ ，下列条件满足

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \max\{f(\mathbf{x}), f(\mathbf{y})\} \quad (6.2.16)$$

称  $f$  是严格拟凸函数 (strictly quasiconvex function)，若

$$f(\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}) < \max\{f(\mathbf{x}), f(\mathbf{y})\} \quad (6.2.17)$$

令  $\alpha \in (0, 1)$  和  $\mathbf{y}_i \neq \mathbf{x}_i$ 。类似于定义 6.2.3，交替最小二乘问题式 (6.2.10) 的目标函数  $f(\mathbf{x})$  称作相对于  $\mathbf{x}_i \in X_i$  的拟凸函数，若

$$\begin{aligned} & f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \alpha\mathbf{x}_i + (1 - \alpha)\mathbf{y}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m) \\ & \leq \max\{f(\mathbf{x}), f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \mathbf{y}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m)\} \end{aligned}$$

称  $f(\mathbf{x})$  是严格拟凸函数，若

$$\begin{aligned} & f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \alpha\mathbf{x}_i + (1 - \alpha)\mathbf{y}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m) \\ & < \max\{f(\mathbf{x}), f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \mathbf{y}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m)\} \end{aligned}$$

下面是在不同假设条件下 GS 算法的收敛性能。

**定理 6.2.1**<sup>[209]</sup> 假定函数  $f$  相对于  $X$  上的向量  $\mathbf{x}_i$  ( $i = 1, \dots, m-2$ ) 是一严格拟凸函数，并且由 GS 法产生的序列  $\{\mathbf{x}^k\}$  具有极限点，则  $\{\mathbf{x}^k\}$  的每一个极限点  $\bar{\mathbf{x}}$  都是优化问题式 (6.2.10) 的一个临界点。

**定理 6.2.2**<sup>[45]</sup> 令  $f$  是式 (6.2.10) 中的目标函数。假定对每一个  $i$  和  $\mathbf{x} \in X$ ，优化算法

$$\min_{\mathbf{y} \in X_i} f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \mathbf{y}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m)$$

的极小点唯一得到。若  $\{\mathbf{x}^k\}$  是由 GS 算法产生的迭代序列，则  $\mathbf{x}^k$  的每一个极限点都是一个临界点。

然而，在实际应用中，优化问题式 (6.2.10) 的目标函数常常可能不满足上述两个定理的条件。例如，根据定理 6.2.1 知，在矩阵  $A$  列秩亏缺的情况下，二次目标函数  $\|\mathbf{Ax} - \mathbf{b}\|_2^2$  不是拟凸函数，所以交替最小二乘法的收敛性能将无法保证。

GS 法产生的序列虽然含有极限点，但它们可能不是优化问题式 (6.2.10) 的临界点。GS 算法有可能不收敛这一事实早在 1973 年就被 Powell 观察到，并称为 GS 法的“徘徊”(circle) 现象<sup>[415]</sup>。最近，文献 [356, 315] 通过大量仿真实验观察到，即使能够收敛，交替最小二乘法的迭代过程也很容易陷入“泥沼”(swamp) 之中：异常高的迭代次数导致收敛速度大幅放缓。特别地，当  $m$  个变元矩阵之中只要有一个变元矩阵的列秩是亏缺的，或者  $m$  个变元矩阵虽然都满列秩，但某些变元矩阵的列向量之间存在共线性(collinearity) 时，很容易观察到这种泥沼现象<sup>[356, 315]</sup>。

避免 GS 法的徘徊和泥沼现象的一种简单而有效的方法是将优化问题式 (6.2.10) 的目标函数进行 Tikhonov 正则化，将分离的子优化算法式 (6.2.11) 正则化为

$$\mathbf{x}_i^{k+1} = \arg \min_{\mathbf{y} \in X_i} f(\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}, \mathbf{y}, \mathbf{x}_{i+1}^k, \dots, \mathbf{x}_m^k) + \frac{1}{2} \tau_i \|\mathbf{y} - \mathbf{x}_i^k\|_2^2 \quad (6.2.18)$$

式中  $i = 1, \dots, m$ 。

上述算法称为 GS 算法的迫近点版本 (proximal point versions)<sup>[26, 47]</sup>，缩写为 PGS。正则项  $\|\mathbf{y} - \mathbf{x}_i^k\|_2^2$  的作用是迫使更新后的向量  $\mathbf{x}_i^{k+1} = \mathbf{y}$  接近  $\mathbf{x}_i^k$ ，不致偏离太多，避免迭代过程的剧烈震荡，防止算法发散。这大概就是“迫近点版本”这一术语的本质所在。

称 GS 或 PGS 方法是良好定义的 (well-defined)，若每个子问题都有一个最优解<sup>[209]</sup>。

**定理 6.2.3 (PGS 的收敛性)**<sup>[209]</sup> 假定 PGS 方法是良好定义的，并且序列  $\{\mathbf{x}^k\}$  有极限点，则  $\{\mathbf{x}^k\}$  的每一个极限点  $\bar{\mathbf{x}}$  都是优化问题式 (6.2.10) 的一个临界点。

定理 6.2.3 表明，PGS 法的收敛性能确实优于 GS 法。文献 [315] 通过大量仿真实验表明，在达到相同误差的条件下，限于泥沼之中的 GS 法的迭代次数异常大，而 PGS 法往往收敛很快。PGS 法在有些文献中也称为正则 GS 算法。

交替最小二乘法和正则交替最小二乘法在非负矩阵分解和张量分解中有着重要的应用，将在后面具体介绍。

## 6.3 总体最小二乘

尽管最初的称呼不同，总体最小二乘 (total least squares, TLS) 实际上已有相当长的历史了。有关总体最小二乘最早的思想可追溯到 Pearson 于 1901 年发表的论文<sup>[399]</sup>，当时他考虑的是  $A$  和  $b$  同时存在误差时矩阵方程  $Ax = b$  的近似求解方法。但是，只是到了 1980 年，才由 Golub 和 Van Loan<sup>[196]</sup> 从数值分析的观点首次对这种方法进行了整体分析，并正式称为总体最小二乘。在数理统计中，这种方法称为正交回归 (orthogonal regression) 或变量误差回归 (errors-in-variables regression)<sup>[190]</sup>。在系统辨识中，总体最小二乘称为特征向量法或 Koopmans-Levin 方法<sup>[496]</sup>。现在，总体最小二乘方法已经广泛应用于信号处理、自动控制、系统科学、统计学、物理学、经济学、生物学和医学等众多学科与领域。

### 6.3.1 总体最小二乘问题

令  $A_0$  和  $b_0$  分别代表不可观测的无误差数据矩阵和无误差数据向量，实际观测的数据矩阵和数据向量分别为

$$A = A_0 + E, \quad b = b_0 + e \quad (6.3.1)$$

其中， $E$  和  $e$  分别表示误差数据矩阵和误差数据向量。

总体最小二乘的基本思想是：不仅用校正向量  $\Delta b$  去干扰数据向量  $b$ ，同时用校正矩阵  $\Delta A$  去干扰数据矩阵  $A$ ，以便对  $A$  和  $b$  二者内存在的误差或噪声进行联合补偿

$$\begin{aligned} b + \Delta b &= b_0 + e + \Delta b \rightarrow b_0 \\ A + \Delta A &= A_0 + E + \Delta A \rightarrow A_0 \end{aligned}$$

以抑制观测误差或噪声对矩阵方程求解的影响，从而实现有误差的矩阵方程求解向精确矩阵方程的求解的转换

$$(A + \Delta A)x = b + \Delta b \implies A_0x = b_0 \quad (6.3.2)$$

自然地，我们希望校正数据矩阵和校正数据向量都尽可能小。因此，总体最小二乘问题可以用约束优化问题叙述为

$$\text{TLS: } \min_{\Delta A, \Delta b, x} \|[\Delta A, \Delta b]\|_2^2 = \|\Delta A\|_2^2 + \|\Delta b\|_2^2 \quad (6.3.3)$$

$$\text{subject to } (A + \Delta A)x = b + \Delta b \quad (6.3.4)$$

约束条件  $(A + \Delta A)x = b + \Delta b$  有时也表示为  $(b + \Delta b) \in \text{Range}(A + \Delta A)$ 。

由式 (6.3.2) 知，原矩阵方程  $Ax = b$  可以改写为

$$([A, b] + [\Delta A, \Delta b]) \begin{bmatrix} x \\ -1 \end{bmatrix} = 0 \quad (6.3.5)$$

或等价为

$$(\mathbf{B} + \mathbf{D})\mathbf{z} = \mathbf{0} \quad (6.3.6)$$

式中, 增广数据矩阵  $\mathbf{B} = [\mathbf{A}, \mathbf{b}]$  和增广校正矩阵  $\mathbf{D} = [\Delta\mathbf{A}, \Delta\mathbf{b}]$  均为  $m \times (n+1)$  维矩阵, 而  $\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}$  为  $(n+1) \times 1$  向量。

由式 (6.3.5) 知, 解向量  $\begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}$  是与增广矩阵  $[\mathbf{A}, \mathbf{b}]$  的最小奇异值  $\sigma_{\min}$  对应的右奇异向量, 亦即与  $[\mathbf{A}, \mathbf{b}]^H[\mathbf{A}, \mathbf{b}]$  的最小特征值  $\lambda_{\min}$  对应的特征向量。换言之, 解向量  $\begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}$  是下列 Rayleigh 商最小化的无约束优化问题的解

$$\min_{\mathbf{x}} J(\mathbf{x}) = \frac{\begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}^H [\mathbf{A}, \mathbf{b}]^H [\mathbf{A}, \mathbf{b}] \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}}{\begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}^H \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}} = \frac{\|\mathbf{Ax} - \mathbf{b}\|_2^2}{\|\mathbf{x}\|_2^2 + 1} \quad (6.3.7)$$

### 6.3.2 总体最小二乘解

在超定方程的总体最小二乘解中, 有两种可能的情况。

情况 1 矩阵  $\mathbf{B}$  的奇异值  $\sigma_n$  明显比  $\sigma_{n+1}$  大, 即最小的奇异值只有一个。

式 (6.3.3) 表明, 总体最小二乘问题可以归结为: 求一具有最小范数平方的扰动矩阵  $\mathbf{D} \in \mathbb{C}^{m \times (n+1)}$ , 使得  $\mathbf{B} + \mathbf{D}$  是非满秩的 (如果满秩, 则只有平凡解  $\mathbf{z} = \mathbf{0}$ )。

事实上, 如果约束最小二乘解  $\mathbf{z}$  是一个单位范数的向量, 并且将式 (6.3.6) 改写为  $\mathbf{Bz} = \mathbf{r} = -\mathbf{Dz}$ , 则总体最小二乘问题式 (6.3.3) 又可以等价写作一个带约束的标准最小二乘问题

$$\min \|\mathbf{Bz}\|_2^2 = \min \|\mathbf{r}\|_2^2 \quad \text{subject to} \quad \mathbf{z}^H \mathbf{z} = 1 \quad (6.3.8)$$

因为  $\mathbf{r}$  可以视为矩阵方程  $\mathbf{Bz} = \mathbf{0}$  的总体最小二乘解  $\mathbf{z}$  的误差向量。换言之, 总体最小二乘解  $\mathbf{z}$  是使得误差平方和  $\|\mathbf{r}\|_2^2$  为最小的最小二乘解。

上述约束最小二乘问题很容易用 Lagrange 乘数法求解。定义目标函数

$$J(\mathbf{z}) = \|\mathbf{Bz}\|_2^2 + \lambda(1 - \mathbf{z}^H \mathbf{z}) \quad (6.3.9)$$

式中,  $\lambda$  为 Lagrange 乘数。注意到  $\|\mathbf{Bz}\|_2^2 = \mathbf{z}^H \mathbf{B}^H \mathbf{Bz}$ , 故由  $\frac{\partial J(\mathbf{z})}{\partial \mathbf{z}^*} = 0$ , 得到

$$\mathbf{B}^H \mathbf{Bz} = \lambda \mathbf{z} \quad (6.3.10)$$

这表明, Lagrange 乘数应该选择为矩阵  $\mathbf{B}^H \mathbf{B}$  的最小特征值 (即  $\mathbf{B}$  的最小奇异值的平方), 而总体最小二乘解  $\mathbf{z}$  是与最小奇异值  $\sigma_{\min} = \sqrt{\lambda_{\min}}$  对应的右奇异向量。

令  $m \times (n+1)$  增广矩阵  $\mathbf{B}$  的奇异值分解为

$$\mathbf{B} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^H \quad (6.3.11)$$

并且其奇异值按照顺序  $\sigma_1 \geq \dots \geq \sigma_{n+1}$  排列，与这些奇异值对应的右奇异向量为  $v_1, \dots, v_{n+1}$ 。于是，根据上面的分析，总体最小二乘解为  $z = v_{n+1}$ 。也就是说，原矩阵方程  $Ax = b$  的最小二乘解由下式给出

$$\mathbf{x}_{\text{TLS}} = \frac{1}{v(1, n+1)} \begin{bmatrix} v(2, n+1) \\ \vdots \\ v(n+1, n+1) \end{bmatrix} \quad (6.3.12)$$

其中， $v(i, n+1)$  是  $V$  的第  $n+1$  列的第  $i$  个元素。

总结以上讨论，可以得到求解约束优化问题

$$\text{TLS : } \min_{\Delta A, \Delta b, x} \|\Delta A\|_2^2 + \alpha^2 \|\Delta b\|_2^2 \quad (6.3.13)$$

$$(A + \Delta A)x = b + \Delta b \quad (6.3.14)$$

的总体最小二乘算法  $\text{TLS}(A, b, \alpha) = (\Delta A, \Delta b, x)$  如下。

**算法 6.3.1** <sup>[196]</sup> TLS 算法  $\text{TLS}(A, b, \alpha) = (\Delta A, \Delta b, x)$

输入  $A \in \mathbb{C}^{m \times n}, b \in \mathbb{C}^m, \alpha > 0$ 。

输出  $\Delta A \in \mathbb{C}^{m \times n}, \Delta b \in \mathbb{C}^m, x \in \mathbb{C}^m$ 。

步骤 1 计算 SVD  $[A, ab] = U \Sigma V^H$ ，其中  $\Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix}$ ,  $\Sigma_1 = \text{diag}(\sigma_1, \dots, \sigma_{n+1})$ 。

步骤 2 若  $\sigma_n(A) > \sigma_{n+1}$  (其中  $\sigma_n(A)$  是数据矩阵  $A$  的第  $n$  个奇异值)，则总体最小二乘问题的解由下式给出

$$\begin{aligned} (\Delta A, \Delta b) &= \sigma_{n+1} u_{n+1} v_{n+1}^T \underbrace{\text{diag}(1, \dots, 1, \alpha)}_{n \uparrow} \\ x &= -\frac{1}{\alpha V_{n+1, n+1}} [V_{1, n+1}, \dots, V_{n, n+1}]^T \end{aligned}$$

式中， $u_{n+1}$  和  $v_{n+1}$  分别是  $U$  和  $V$  的第  $n+1$  列，而  $V_{i,j}$  是  $V$  的第  $(i, j)$  元素。

情况 2 矩阵  $B$  的最小奇异值多重 (最后面若干个奇异值重复或非常接近)。

不妨令

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p > \sigma_{p+1} \approx \dots \approx \sigma_{n+1} \quad (6.3.15)$$

且  $v_i$  是子空间

$$S = \text{Span}\{v_{p+1}, v_{p+2}, \dots, v_{n+1}\}$$

中的任一列向量，则上述任一右奇异向量  $v_i$  都给出一组总体最小二乘解

$$x = y_i / \alpha_i, \quad i = p+1, p+2, \dots, n+1$$

其中， $\alpha_i$  是向量  $v_i$  的第一个元素，而其他的元素组成向量  $y_i$ ，也即  $v_i = \begin{bmatrix} \alpha_i \\ y_i \end{bmatrix}$ 。因此，会有  $n+1-p$  个总体最小二乘解。然而，可以找出在某种意义上唯一的总体最小二乘解。可能的唯一解有两种：

- (1) 最小范数解: 解向量由  $n$  个参数组成。
- (2) 最优最小二乘近似解: 解向量仅包含  $p$  个参数。

下面分别给予介绍。

### 1. 最小范数解

最小范数解为  $n$  个参数的总体最小二乘解。求解最小范数解的总体最小二乘算法由 Golub 和 Van Loan<sup>[196]</sup> 提出。

#### 算法 6.3.2 最小范数解的 TLS 算法

步骤 1 计算增广矩阵的奇异值分解  $\mathbf{B} = \mathbf{U}\Sigma\mathbf{V}^H$ , 并存储矩阵  $\mathbf{V}$  和所有奇异值。

步骤 2 确定主奇异值的个数  $p$ 。

步骤 3 令  $\mathbf{V}_1 = [\mathbf{v}_{p+1}, \mathbf{v}_{p+2}, \dots, \mathbf{v}_{n+1}]$  是  $\mathbf{V}$  的列分块形式, 并计算 Householder 变换矩阵  $\mathbf{Q}$  使得

$$\mathbf{V}_1 \mathbf{Q} = \left[ \begin{array}{c|cccc} \alpha & 0 & \cdots & 0 \\ \hline \cdots & \cdots & \cdots & \cdots \\ \mathbf{y} & \times \end{array} \right]$$

其中,  $\alpha$  是一个标量,  $\times$  代表其数值在下一步不起作用的块。

步骤 4 若  $\alpha \neq 0$ , 则  $\mathbf{x}_{\text{TLS}} = \mathbf{y}/\alpha$ ; 若  $\alpha = 0$ , 则对原设定的  $p$  无 TLS 解, 应减小  $p$ , 即使用  $p \leftarrow p - 1$ , 并重复以上步骤, 直至求出唯一的 TLS 解。

步骤 4 表明, 确定  $\mathbf{x}_{\text{TLS}}$  只需要使用  $[\alpha, \mathbf{y}^T]^T$ , 因此在步骤 3, 没有必要计算整个矩阵  $\mathbf{Q}$ , 只需要计算出  $\mathbf{Q}$  的第 1 列即可。具体说来,  $[\alpha, \mathbf{y}^T]^T$  可以通过使  $\mathbf{Q}$  的第 1 列取  $\mathbf{V}_1$  的第 1 行的复数共轭直接获得 (还有其他方法, 但这是最简单的一种)。如果令向量  $\bar{\mathbf{v}}_1$  是矩阵  $\mathbf{V}_1$  的第 1 行, 即对  $\mathbf{V}_1$  作如下分块

$$\mathbf{V}_1 = \begin{bmatrix} \bar{\mathbf{v}}_1 \\ \bar{\mathbf{V}} \end{bmatrix} \quad (6.3.16)$$

即可将 TLS 解最终写作

$$\mathbf{x}_{\text{TLS}} = \frac{\bar{\mathbf{V}} \bar{\mathbf{v}}_1^H}{\bar{\mathbf{v}}_1 \bar{\mathbf{v}}_1^H} = \alpha^{-1} \bar{\mathbf{V}} \bar{\mathbf{v}}_1^H \quad (6.3.17)$$

显然,  $\alpha \approx 0$  对应于  $\mathbf{V}_1$  的第 1 行均为数值很小的元素。在这种情况下, 应该减小  $p$  即增加  $\mathbf{V}_1$  的维数, 以便得到一个非零的  $\alpha = \bar{\mathbf{v}}_1 \bar{\mathbf{v}}_1^H$  (注意, 这里的  $\bar{\mathbf{v}}_1$  是一个行向量)。

应当注意的是, 最小范数解  $\mathbf{x}_{\text{TLS}}$  和原方程  $\mathbf{Ax} = \mathbf{b}$  的未知参数向量  $\mathbf{x}$  一样, 含有  $n$  个参数。由此可见, 尽管  $\mathbf{B}$  的有效秩  $p$  小于  $n$ , 但是最小范数解仍然假定在向量  $\mathbf{x}$  中的  $n$  个未知参数是相互独立的。事实上, 由于增广矩阵  $\mathbf{B} = [\mathbf{A}, \mathbf{b}]$  与原数据矩阵  $\mathbf{A}$  具有相同的秩, 故  $\mathbf{A}$  的秩也是  $p$ 。这意味着,  $\mathbf{A}$  中仅有  $p$  列是线性无关的, 从而原方程  $\mathbf{Ax} = \mathbf{b}$  中起主导作用的参数个数是  $p$ , 而不是  $n$ 。概而言之, TLS 问题的最小范数解中包含了一些冗余的参数, 它们与另外一些参数是线性相关的。在信号处理和系统理论中, 往往对不含冗余参数的唯一 TLS 解更加感兴趣, 这就是最优最小二乘近似解。

### 2. 最优最小二乘近似解

首先, 令  $m \times (n+1)$  矩阵  $\hat{\mathbf{B}}$  是增广矩阵  $\mathbf{B}$  的一个秩  $p$  的最佳逼近, 即

$$\hat{\mathbf{B}} = \mathbf{U} \boldsymbol{\Sigma}_p \mathbf{V}^H \quad (6.3.18)$$

式中,  $\boldsymbol{\Sigma}_p = \text{diag}(\sigma_1, \dots, \sigma_p, 0, \dots, 0)$ 。

再令  $m \times (p+1)$  矩阵  $\hat{\mathbf{B}}_j^{(p)}$  是  $m \times (n+1)$  最优逼近矩阵  $\hat{\mathbf{B}}$  中的一个子矩阵, 定义为

$$\hat{\mathbf{B}}_j^{(p)}: \text{由 } \hat{\mathbf{B}} \text{ 的第 } j \text{ 列到第 } p+j \text{ 列组成的子矩阵} \quad (6.3.19)$$

显然, 这样的子矩阵共有  $n+1-p$  个, 即  $\hat{\mathbf{B}}_1^{(p)}, \hat{\mathbf{B}}_2^{(p)}, \dots, \hat{\mathbf{B}}_{n+1-p}^{(p)}$ 。

如前所述,  $\mathbf{B}$  的有效秩为  $p$  意味着参数向量  $\mathbf{x}$  中只有  $p$  个是线性独立的。不妨令  $(p+1) \times 1$  向量  $\mathbf{a} = \begin{bmatrix} \mathbf{x}^{(p)} \\ -1 \end{bmatrix}$ , 其中,  $\mathbf{x}^{(p)}$  是由向量  $\mathbf{x}$  中的  $p$  个线性独立的未知参数组成的列向量。这样一来, 原总体最小二乘问题的求解就变成了下列  $n+1-p$  个 TLS 问题的求解

$$\hat{\mathbf{B}}_j^{(p)} \mathbf{a} = 0, \quad j = 1, 2, \dots, n+1-p \quad (6.3.20)$$

或等价为合成的 TLS 问题的求解

$$\begin{bmatrix} \hat{\mathbf{B}}(1:p+1) \\ \hat{\mathbf{B}}(2:p+2) \\ \vdots \\ \hat{\mathbf{B}}(n+1-p:n+1) \end{bmatrix} \mathbf{a} = \mathbf{0} \quad (6.3.21)$$

式中,  $\hat{\mathbf{B}}(i:p+i)$  代表式 (6.3.19) 定义的  $\hat{\mathbf{B}}_i^{(p)}$ 。不难证明

$$\hat{\mathbf{B}}(i:p+i) = \sum_{k=1}^p \sigma_k \mathbf{u}_k (\mathbf{v}_k^i)^H \quad (6.3.22)$$

式中,  $\mathbf{v}_k^i$  是酉矩阵  $\mathbf{V}$  的第  $k$  列向量的一个加窗段, 定义为

$$\mathbf{v}_k^i = [v(i, k), v(i+1, k), \dots, v(i+p, k)]^T \quad (6.3.23)$$

这里,  $v(i, k)$  是酉矩阵  $\mathbf{V}$  第  $i$  行第  $k$  列上的元素。

根据最小二乘原理, 求方程组式 (6.3.21) 的最小二乘解等价于使测度 (或代价) 函数

$$\begin{aligned} f(\mathbf{a}) &= [\hat{\mathbf{B}}(1:p+1)\mathbf{a}]^H \hat{\mathbf{B}}(1:p+1)\mathbf{a} + [\hat{\mathbf{B}}(2:p+2)\mathbf{a}]^H \hat{\mathbf{B}}(2:p+2)\mathbf{a} + \dots + \\ &\quad [\hat{\mathbf{B}}(n+1-p:n+1)\mathbf{a}]^H \hat{\mathbf{B}}(n+1-p:n+1)\mathbf{a} \\ &= \mathbf{a}^H \left[ \sum_{i=1}^{n+1-p} [\hat{\mathbf{B}}(i:p+i)]^H \hat{\mathbf{B}}(i:p+i) \right] \mathbf{a} \end{aligned} \quad (6.3.24)$$

极小化。

定义  $(p+1) \times (p+1)$  矩阵

$$\mathbf{S}^{(p)} = \sum_{i=1}^{n+1-p} [\hat{\mathbf{B}}(i:p+i)]^H \hat{\mathbf{B}}(i:p+i) \quad (6.3.25)$$

则测度函数可简写为

$$f(\mathbf{a}) = \mathbf{a}^H \mathbf{S}^{(p)} \mathbf{a} \quad (6.3.26)$$

$f(\mathbf{a})$  的极小化变量  $\mathbf{a}$  由  $\partial f(\mathbf{a}) / \partial \mathbf{a}^* = 0$  给出, 其结果为

$$\mathbf{S}^{(p)} \mathbf{a} = \alpha \mathbf{e}_1 \quad (6.3.27)$$

式中,  $\mathbf{e}_1 = [1, 0, \dots, 0]^T$ , 而常数  $\alpha > 0$  表示误差能量。由定义式 (6.3.25) 和式 (6.3.22) 可以求得

$$\mathbf{S}^{(p)} = \sum_{j=1}^p \sum_{i=1}^{n+1-p} \sigma_j^2 \mathbf{v}_j^i (\mathbf{v}_j^i)^H \quad (6.3.28)$$

方程式 (6.3.27) 的求解是简单的, 它与未知的常数  $\alpha$  无关。如果我们令  $\mathbf{S}^{-(p)}$  为矩阵  $\mathbf{S}^{(p)}$  的逆矩阵, 则解向量  $\mathbf{a}$  仅取决于逆矩阵  $\mathbf{S}^{-(p)}$  的第 1 列。易知, TLS 解向量  $\mathbf{a} = \begin{bmatrix} \mathbf{x}^{(p)} \\ -1 \end{bmatrix}$  中的  $\mathbf{x}^{(p)} = [x_{\text{TLS}}(1), \dots, x_{\text{TLS}}(p)]^T$  的元素由

$$x_{\text{TLS}}(i) = -\mathbf{S}^{-(p)}(i, 1) / \mathbf{S}^{-(p)}(p+1, 1), \quad i = 1, \dots, p \quad (6.3.29)$$

给出。通常称这种解为最优最小二乘近似解。由于这种解的参数个数与有效秩相同, 故又称为低阶模型或低秩总体最小二乘解<sup>[74]</sup>。

注意, 若增广矩阵  $\mathbf{B} = [-\mathbf{b}, \mathbf{A}]$ , 则

$$x_{\text{TLS}}(i) = \mathbf{S}^{-(p)}(i+1, 1) / \mathbf{S}^{-(p)}(1, 1), \quad i = 1, 2, \dots, p \quad (6.3.30)$$

因为在这种情况下, 解向量  $\mathbf{a} = \begin{bmatrix} 1 \\ \mathbf{x}^{(p)} \end{bmatrix}$ 。

归纳起来, 求最优最小二乘近似解的具体算法如下。

### 算法 6.3.3 SVD-TLS 算法

步骤 1 计算增广矩阵  $\mathbf{B}$  的 SVD, 并存储右奇异矩阵  $\mathbf{V}$ 。

步骤 2 确定  $\mathbf{B}$  的有效秩  $p$ 。

步骤 3 利用式 (6.3.28) 和式 (6.3.23) 计算  $(p+1) \times (p+1)$  矩阵  $\mathbf{S}^{(p)}$ 。

步骤 4 求  $\mathbf{S}^{(p)}$  的逆矩阵  $\mathbf{S}^{-(p)}$ , 并由式 (6.3.29) 求最优最小二乘近似解。

上述算法的基本思想是由 Cadzow<sup>[74]</sup> 提出来的。

### 6.3.3 总体最小二乘解的性能

总体最小二乘有两个非常有趣的解释: 一个是它的几何解释<sup>[196]</sup>, 另一个是它的闭式解<sup>[513]</sup>。

#### 1. 总体最小二乘解的几何解释

令  $\mathbf{a}_i^T$  是矩阵的第  $i$  行,  $b_i$  是向量  $\mathbf{b}$  的第  $i$  个元素, 则总体最小二乘解  $\mathbf{x}_{\text{TLS}}$  是使

$$\min_{\mathbf{x}} \frac{\|\mathbf{Ax} - \mathbf{b}\|_2^2}{\|\mathbf{x}\|_2^2 + 1} = \sum_{i=1}^n \frac{|\mathbf{a}_i^T \mathbf{x} - b_i|^2}{\mathbf{x}^T \mathbf{x} + 1} \quad (6.3.31)$$

的极小化变量, 其中  $|a_i^T x - b_i| / (x^T x + 1)$  是从点  $\begin{pmatrix} a_i \\ b_i \end{pmatrix} \in \mathbb{C}^{n+1}$  到子空间  $P_x$  内的最近点的距离, 且子空间  $P_x$  定义为

$$P_x = \left\{ \begin{pmatrix} a \\ b \end{pmatrix} \mid a \in \mathbb{C}^{n \times 1}, b \in C, b = x^T a \right\} \quad (6.3.32)$$

因此, 总体最小二乘解可以用子空间  $P_x$  表征<sup>[196]</sup>: 总体最小二乘问题等价于求到  $m$  个二元组  $\begin{pmatrix} a_i \\ b_i \end{pmatrix}, i = 1, 2, \dots, m$  的最近的子空间  $P_x$ , 即解点  $\begin{pmatrix} a_i \\ b_i \end{pmatrix}$  到  $P_x$  的距离的平方和为最小。图 6.3.1 画出了一维情况下 LS 解与 TLS 解的比较。

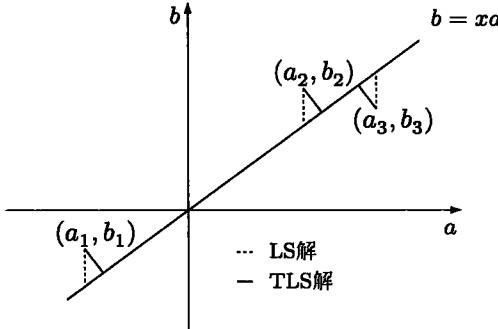


图 6.3.1 LS 解与 TLS 解

图 6.3.1 中, 虚线表示的是 LS 解, 它是 (与  $b$  轴) 平行的竖直距离; 实线所示为 TLS 解, 它始于点  $(a_i, b_i)$ , 是到直线  $b = xa$  的垂直距离。从这一几何解释, 可以得出结论: 总体最小二乘方法比最小二乘方法好, 因为前者在曲线拟合中的残差最小。

## 2. 总体最小二乘解的闭式解

若增广矩阵  $\mathbf{B}$  的奇异值为  $\sigma_1 \geq \dots \geq \sigma_{n+1}$ , 则总体最小二乘解可表示成<sup>[513]</sup>

$$\mathbf{x}_{\text{TLS}} = (\mathbf{A}^H \mathbf{A} - \sigma_{n+1}^2 \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b} \quad (6.3.33)$$

与 Tikhonov 正则化比较知, 总体最小二乘是一种反正则化方法, 可以解释为一种具有噪声清除作用的最小二乘方法: 先从协方差矩阵  $\mathbf{A}^T \mathbf{A}$  中减去噪声影响项  $\sigma_{n+1}^2 \mathbf{I}$ , 然后再矩阵求逆, 得到最小二乘解。

令含误差的数据矩阵  $\mathbf{A} = \mathbf{A}_0 + \mathbf{E}$ , 则其协方差矩阵  $\mathbf{A}^H \mathbf{A} = \mathbf{A}_0^H \mathbf{A}_0 + \mathbf{E}^H \mathbf{A}_0 + \mathbf{A}_0^H \mathbf{E} + \mathbf{E}^H \mathbf{E}$ 。显然, 当误差矩阵  $\mathbf{E}$  具有零均值时, 协方差矩阵的数学期望  $E\{\mathbf{A}^H \mathbf{A}\} = E\{\mathbf{A}_0^H \mathbf{A}_0\} + E\{\mathbf{E}^H \mathbf{E}\} = \mathbf{A}_0^H \mathbf{A}_0 + E\{\mathbf{E}^H \mathbf{E}\}$ 。若误差矩阵的列向量统计不相关, 并且具有相同方差, 即  $E\{\mathbf{E}^T \mathbf{E}\} = \sigma^2 \mathbf{I}$ , 则  $(n+1) \times (n+1)$  协方差矩阵  $\mathbf{A}^H \mathbf{A}$  的最小特征值  $\lambda_{n+1} = \sigma_{n+1}^2$  就是误差矩阵  $\mathbf{E}$  的奇异值的平方。由于奇异值平方  $\sigma_{n+1}^2$  恰巧体现了误差矩阵各个列向量共同的方差  $\sigma^2$ , 使得通过  $\mathbf{A}^H \mathbf{A} - \sigma_{n+1}^2 \mathbf{I}$  之运算, 可以恢复原来无误差数据矩阵的协方差矩阵, 即有  $\mathbf{A}^T \mathbf{A} - \sigma_{n+1}^2 \mathbf{I} = \mathbf{A}_0^H \mathbf{A}_0$ 。换言之, 总体最小二乘方法有效地抑制了未知误差矩阵的影响。

应当指出, 求解矩阵方程  $\mathbf{A}_{m \times n} \mathbf{x}_n = \mathbf{b}_m$  的总体最小二乘方法与 Tikhonov 正则化方法的主要区别在于: 总体最小二乘解可以只包含  $p = \text{rank}([\mathbf{A}, \mathbf{b}])$  个主要参数在内, 将冗余参数剔除; 而 Tikhonov 正则化方法求得的解包含了所有  $n$  个参数, 没有抓主舍次的参数选择功能。

以下是求解超定矩阵方程  $\mathbf{Ax} = \mathbf{b}$  的普通最小二乘、数据最小二乘、Tikhonov 正则化和总体最小二乘四种方法之间的比较。

### 1. 解向量的比较

$$\hat{\mathbf{x}}_{\text{LS}} = \arg \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2^2 = \arg \min_{\mathbf{x}} (\mathbf{Ax} - \mathbf{b})^H (\mathbf{Ax} - \mathbf{b}) \quad (6.3.34)$$

$$\hat{\mathbf{x}}_{\text{DLS}} = \arg \min_{\mathbf{x}} \frac{\|\mathbf{Ax} - \mathbf{b}\|_2^2}{\|\mathbf{x}\|_2^2} = \arg \min_{\mathbf{x}} \frac{(\mathbf{Ax} - \mathbf{b})^H (\mathbf{Ax} - \mathbf{b})}{\mathbf{x}^H \mathbf{x}} \quad (6.3.35)$$

$$\hat{\mathbf{x}}_{\text{Tik}} = \arg \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{x}\|_2^2 = \arg \min_{\mathbf{x}} (\mathbf{Ax} - \mathbf{b})^H (\mathbf{Ax} - \mathbf{b}) + \lambda \mathbf{x}^H \mathbf{x} \quad (6.3.36)$$

$$\hat{\mathbf{x}}_{\text{TLS}} = \arg \min_{\mathbf{x}} \frac{\|\mathbf{Ax} - \mathbf{b}\|_2^2}{\|\mathbf{x}\|_2^2 + 1} = \arg \min_{\mathbf{x}} \frac{(\mathbf{Ax} - \mathbf{b})^H (\mathbf{Ax} - \mathbf{b})}{\mathbf{x}^H \mathbf{x} + 1} \quad (6.3.37)$$

### 2. 扰动方法的比较

(1) 普通最小二乘方法: 用尽可能小的校正项  $\Delta \mathbf{b}$  “扰动”数据向量  $\mathbf{b}$ , 使得  $\mathbf{b} - \Delta \mathbf{b} \approx \mathbf{b}_0$ , 从而补偿  $\mathbf{b}$  中存在的观测噪声或误差  $\mathbf{e}$ 。校正向量选择  $\Delta \mathbf{b} = \mathbf{Ax} - \mathbf{b}$ , 解析解为  $\hat{\mathbf{x}}_{\text{LS}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$ 。

(2) 数据最小二乘方法: 校正矩阵  $\Delta \mathbf{A} = \frac{(\mathbf{Ax} - \mathbf{b}) \mathbf{x}^H}{\mathbf{x}^H \mathbf{x}}$ , 其目的是补偿数据矩阵  $\mathbf{A}$  中存在的观测误差矩阵  $\mathbf{E}$ 。数据最小二乘解为  $\hat{\mathbf{x}}_{\text{DLS}} = \arg \min_{\mathbf{x}} \frac{(\mathbf{Ax} - \mathbf{b})^H (\mathbf{Ax} - \mathbf{b})}{\mathbf{x}^H \mathbf{x}}$ 。

(3) Tikhonov 正则化方法: 解析解为  $(\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b}$ , 通过给矩阵  $\mathbf{A}^H \mathbf{A}$  的每个对角元素加相同的扰动项  $\lambda > 0$ , 可以避免最小二乘解  $(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b}$  的数值不稳定性。

(4) 总体最小二乘方法: 存在三种不同的解: 最小范数解、含全部  $n$  个元素的反正则化解  $\hat{\mathbf{x}}_{\text{TLS}} = (\mathbf{A}^H \mathbf{A} - \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b}$  以及只有  $p = \text{rank}([\mathbf{A}, \mathbf{b}])$  个主要参数的 SVD-TLS 解。

特别地, 上述四种方法的适用范围不同:

(1) 最小二乘方法适用于数据矩阵  $\mathbf{A}$  满列秩且精确已知, 数据向量  $\mathbf{b}$  存在独立同分布的高斯误差的情况。

(2) 数据最小二乘适用于数据矩阵  $\mathbf{A}$  满列秩, 且存在独立同分布的高斯误差以及数据向量  $\mathbf{b}$  无误差的情况。

(3) Tikhonov 正则化适用于数据矩阵  $\mathbf{A}$  的列秩亏缺的情况。

(4) 总体最小二乘适用于满列秩的数据矩阵  $\mathbf{A}$  和数据向量  $\mathbf{b}$  均存在独立同分布的高斯误差的情况。

### 6.3.4 总体最小二乘拟合

举凡需要求解线性方程  $\mathbf{Ax} = \mathbf{b}$  的工程问题, 由于矩阵  $\mathbf{A}$  和向量  $\mathbf{b}$  的元素都是实测数据, 总是存在误差。因此, 总体最小二乘方法在这些场合都可以使用。事实上, 总体最小二乘方法已在工程问题中获得了广泛的应用。

在科学与工程问题的数值分析中, 经常需要对给定的一些数据点, 拟合一条曲线或一曲面。由于这些数据点通常是观测得到的, 不可避免地会含有误差或被噪声污染, 总体最小二乘方法可望给出比一般最小二乘方法更好的拟合结果。

考虑数据拟合问题: 给定  $n$  个数据点  $(x_1, y_1), \dots, (x_n, y_n)$ , 希望对这些点拟合一直线。假定直线方程为  $ax + by - c = 0$ 。若直线通过点  $(x_0, y_0)$ , 则  $c = ax_0 + by_0$ 。

现在考虑让拟合直线通过已知  $n$  个数据点的中心

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \quad (6.3.38)$$

若将  $c = a\bar{x} + b\bar{y}$  代入, 则可将直线方程写作

$$a(x - \bar{x}) + b(y - \bar{y}) = 0 \quad (6.3.39)$$

或者用斜率形式等价写为

$$m(x - \bar{x}) + (y - \bar{y}) = 0 \quad (6.3.40)$$

参数向量  $[a, b]^T$  称为拟合直线的法向量 (normal vector), 而  $-m = -a/b$  称为拟合直线的斜率。于是, 直线拟合问题便变成了法向量  $[a, b]^T$  或者斜率参数  $m$  的求解。

显然, 将  $n$  个已知数据点代入直线方程后, 直线方程不可能严格满足, 会存在拟合误差。最小二乘拟合就是使拟合误差的平方和最小化, 即最小二乘拟合的代价函数取为

$$D_{LS}^{(1)}(m, \bar{x}, \bar{y}) = \sum_{i=1}^n [(x_i - \bar{x}) + m(y_i - \bar{y})]^2 \quad (6.3.41)$$

$$D_{LS}^{(2)}(m, \bar{x}, \bar{y}) = \sum_{i=1}^n [m(x_i - \bar{x}) - (y_i - \bar{y})]^2 \quad (6.3.42)$$

令  $\frac{\partial D_{LS}^{(i)}(m, \bar{x}, \bar{y})}{\partial m} = 0$ ,  $i = 1, 2$ , 即可求出直线斜率  $m$ 。然后, 只要将  $m$  代入式 (6.3.40), 便可得到拟合直线的方程。

与最小二乘拟合不同, 总体最小二乘拟合则考虑使各个已知数据点到直线方程  $a(x - x_0) + b(y - y_0) = 0$  的距离平方和最小化。

点  $(p, q)$  到直线  $ax + by - c = 0$  的距离  $d$  由

$$d^2 = \frac{(ap + bq - c)^2}{a^2 + b^2} = \frac{[a(p - x_0) + b(q - y_0)]^2}{a^2 + b^2} \quad (6.3.43)$$

确定。于是, 已知的  $n$  个数据点到直线  $a(x - \bar{x}) + b(y - \bar{y}) = 0$  的距离平方和为

$$D(a, b, \bar{x}, \bar{y}) = \sum_{i=1}^n \frac{[a(x_i - \bar{x}) + b(y_i - \bar{y})]^2}{a^2 + b^2} \quad (6.3.44)$$

**引理 6.3.1** <sup>[370]</sup> 对过直线的数据点  $(x_0, y_0)$  和数据点集合  $(x_1, y_1), \dots, (x_n, y_n)$ , 恒有不等式

$$D(a, b, \bar{x}, \bar{y}) \leq D(a, b, x_0, y_0) \quad (6.3.45)$$

等号成立, 当且仅当  $x_0 = \bar{x}$  和  $y_0 = \bar{y}$ 。

引理 6.3.1 表明, 总体最小二乘拟合的直线必须通过  $n$  个数据点的中心  $(\bar{x}, \bar{y})$ , 才能使偏差  $D$  最小。

为了求拟合直线的法向量  $[a, b]^T$  或斜率  $-m = -a/b$ , 下面考虑如何使偏差  $D$  最小。为此, 将  $D$  写成  $2 \times 1$  单位向量  $t = (a^2 + b^2)^{-1/2}[a, b]^T$  与  $n \times 2$  矩阵  $M$  的乘积, 即

$$D(a, b, \bar{x}, \bar{y}) = \|Mt\|_2^2 = \left\| \begin{bmatrix} x_1 - \bar{x} & y_1 - \bar{y} \\ x_2 - \bar{x} & y_2 - \bar{y} \\ \vdots & \vdots \\ x_n - \bar{x} & y_n - \bar{y} \end{bmatrix} \frac{1}{\sqrt{a^2 + b^2}} \begin{bmatrix} a \\ b \end{bmatrix} \right\|_2^2 \quad (6.3.46)$$

式中

$$M = \begin{bmatrix} x_1 - \bar{x} & y_1 - \bar{y} \\ x_2 - \bar{x} & y_2 - \bar{y} \\ \vdots & \vdots \\ x_n - \bar{x} & y_n - \bar{y} \end{bmatrix} \quad (6.3.47)$$

由式 (6.3.46) 直接可得下面的结果 <sup>[370]</sup>。

**命题 6.3.1** 距离平方和  $D(a, b, \bar{x}, \bar{y})$  在单位法向量  $t = (a^2 + b^2)^{-1/2}[a, b]^T$  达到最小值。此时, 映射  $t \mapsto \|Mt\|_2$  在单位球面  $S^1 = \{t \in \mathbb{R}^2 \mid \|t\|_2 = 1\}$  达到最小值。

命题 6.3.1 表明, 距离平方和  $D(a, b, \bar{x}, \bar{y})$  有一个最小值。下面的定理给出了如何获得这一最小距离平方和的方法。

**定理 6.3.1** <sup>[370]</sup> 若  $2 \times 1$  法向量  $t$  取作与  $2 \times 2$  矩阵  $M^T M$  的最小特征值  $\sigma_2^2$  对应的特征向量, 则距离平方和  $D(a, b, \bar{x}, \bar{y})$  取最小值  $\sigma_2^2$ 。

文献 [370] 给出的定理证明不严密。事实上, 定理的证明是简单的: 利用  $\|t\|_2 = 1$  的约定, 距离平方和  $D(a, b, \bar{x}, \bar{y})$  可以写作

$$D(a, b, \bar{x}, \bar{y}) = \frac{t^T M^T M t}{t^T t} \quad (6.3.48)$$

这是典型的 Rayleigh 商形式。显然,  $D(a, b, \bar{x}, \bar{y})$  取最小值的条件是: 法向量  $t$  取作与矩阵  $M^T M$  的最小特征值对应的特征向量。

下面的例子有助于我们进一步理解总体最小二乘拟合与一般的最小二乘拟合之间的差别。

**例 6.3.1** 已知三个数据点  $(2, 1), (2, 4), (5, 1)$ 。计算中心点, 得

$$\bar{x} = \frac{1}{3}(2 + 2 + 5) = 3, \quad \bar{y} = \frac{1}{3}(1 + 4 + 1) = 2$$

减去这些均值后，得到零均值的数据矩阵

$$\mathbf{M} = \begin{bmatrix} 2-3 & 1-2 \\ 2-3 & 4-2 \\ 5-3 & 1-2 \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ -1 & 2 \\ 2 & -1 \end{bmatrix}$$

从而有

$$\mathbf{M}^T \mathbf{M} = \begin{bmatrix} 6 & -3 \\ -3 & 6 \end{bmatrix}$$

其特征值分解为

$$\mathbf{M}^T \mathbf{M} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 9 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 6 & -3 \\ -3 & 6 \end{bmatrix}$$

因此，法向量  $\mathbf{t} = [\mathbf{a}, \mathbf{b}]^T = [1/\sqrt{2}, 1/\sqrt{2}]^T$ 。最后，得总体最小二乘拟合的直线方程为

$$a(x - \bar{x}) + b(y - \bar{y}) = 0 \implies \frac{1}{\sqrt{2}}(x - 3) + \frac{1}{\sqrt{2}}(y - 2) = 0$$

即  $y = -x + 5$ 。此时，距离平方和为

$$D_{\text{TLS}}(\mathbf{a}, \mathbf{b}, \bar{x}, \bar{y}) = \|\mathbf{M}\mathbf{t}\|_2^2 = \left\| \begin{bmatrix} -1 & -1 \\ -1 & 2 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \right\|_2^2 = 3$$

与总体最小二乘拟合不同，若最小二乘拟合的代价函数取

$$\begin{aligned} D_{\text{LS}}^{(1)}(m, \bar{x}, \bar{y}) &= \frac{1}{m^2 + 1} \sum_{i=1}^3 [m(x_i - \bar{x}) + (y_i - \bar{y})]^2 \\ &= \frac{1}{m^2 + 1} [(-m - 1)^2 + (-m + 2)^2 + (2m - 1)^2] \end{aligned}$$

令

$$\frac{\partial D_{\text{LS}}^{(1)}(m, \bar{x}, \bar{y})}{\partial m} = 6m - 3 = 0$$

得  $m = 1/2$ ，即斜率为  $-1/2$ 。此时，最小二乘拟合的直线方程为  $\frac{1}{2}(x - 3) + (y - 2) = 0$ ，即  $x + 2y - 7 = 0$ ，相应的距离平方和  $D_{\text{LS}}^{(1)}(m, \bar{x}, \bar{y}) = 3.6$ 。

类似地，若最小二乘拟合采用代价函数

$$\begin{aligned} D_{\text{LS}}^{(2)}(m, \bar{x}, \bar{y}) &= \frac{1}{m^2 + 1} \sum_{i=1}^3 [m(y_i - \bar{y}) + (x_i - \bar{x})]^2 \\ &= \frac{1}{m^2 + 1} [(-m - 1)^2 + (2m - 1)^2 + (-m + 2)^2] \end{aligned}$$

则使得  $D_{\text{LS}}^{(2)}(m, \bar{x}, \bar{y})$  最小的  $m = \frac{1}{2}$ ，即拟合的直线方程为  $2x - y - 4 = 0$ ，相应的距离平方和  $D_{\text{LS}}^{(2)}(m, \bar{x}, \bar{y}) = 3.6$ 。

图 6.3.2 画出了使用总体最小二乘方法和两种最小二乘方法拟合直线的结果。

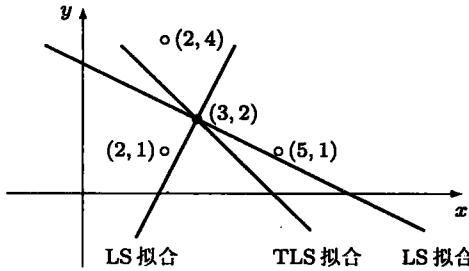


图 6.3.2 最小二乘拟合直线与总体最小二乘拟合直线

这个例子表明,  $D_{\text{LS}}^{(1)}(m, \bar{x}, \bar{y}) = D_{\text{LS}}^{(2)}(m, \bar{x}, \bar{y}) > D_{\text{TLS}}(a, b, \bar{x}, \bar{y})$ , 即两种最小二乘拟合具有相同的拟合误差偏差, 它们比总体最小二乘的拟合偏差大。可见, 总体最小二乘拟合确实比最小二乘拟合的精度高。

**定理 6.3.1** 很容易推广到高维数据情况。令  $n$  个数据向量  $\mathbf{x}_i = [x_{1i}, \dots, x_{mi}]^T, i = 1, 2, \dots, n$  分别为  $m$  维数据, 并且

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_m]^T \quad (6.3.49)$$

为均值 (即中心) 向量, 式中,  $\bar{x}_j = \sum_{i=1}^n x_{ji}$ 。现在考虑使用  $m$  维法向量  $\mathbf{r} = [r_1, \dots, r_m]^T$  对已知的数据向量, 拟合超平面 (hyperplane)  $\mathbf{x}$ , 即  $\mathbf{x}$  满足法方程

$$\langle \mathbf{x} - \bar{\mathbf{x}}, \mathbf{r} \rangle = 0 \quad (6.3.50)$$

构造  $n \times m$  矩阵

$$\mathbf{M} = \begin{bmatrix} \mathbf{x}_1 - \bar{\mathbf{x}} \\ \vdots \\ \mathbf{x}_n - \bar{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} x_{11} - \bar{x}_1 & x_{12} - \bar{x}_2 & \cdots & x_{1m} - \bar{x}_m \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} - \bar{x}_1 & x_{n2} - \bar{x}_2 & \cdots & x_{nm} - \bar{x}_m \end{bmatrix} \quad (6.3.51)$$

则可以得到拟合  $m$  维超平面的总体最小二乘算法如下。

**算法 6.3.4**  $m$  维超平面拟合的总体最小二乘算法 [370]

已知  $n$  个数据向量  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ 。

步骤 1 计算均值向量  $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$ 。

步骤 2 利用式 (6.3.51) 构造  $n \times m$  矩阵  $\mathbf{M}$ 。

步骤 3 计算  $m \times m$  矩阵  $\mathbf{M}^T \mathbf{M}$  的最小特征值及其对应的特征向量  $\mathbf{u}$ , 并令  $\mathbf{r} = \mathbf{u}$ 。

结果 由法方程  $\langle \mathbf{x} - \bar{\mathbf{x}}, \mathbf{r} \rangle = 0$  确定的超平面可以使得距离平方和  $D(\mathbf{r}, \bar{\mathbf{x}})$  最小。

距离平方和  $D(\mathbf{r}, \bar{\mathbf{x}})$  实际上代表了各个已知数据向量 (点) 到达超平面的距离平方和。因此, 距离平方和最小, 意味着拟合误差平方和最小。

需要注意的是，如果矩阵  $\mathbf{M}^T \mathbf{M}$  的最小特征值（或者  $\mathbf{M}$  的最小奇异值）具有多重度，则与之对应的特征向量也有多个，从而导致拟合超平面存在多个解。这种情况的发生或许昭示线性数据拟合模型可能不合适，而应该尝试其他的非线性拟合模型。

总体最小二乘已在下列领域获得了广泛的应用：信号处理 [261, 546]，生物医学信号处理 [492]，图像处理 [369]，变量误差建模 [491, 467]，频域系统辨识 [408, 441]，线性系统的子空间辨识 [495]，天文学 [59]，通信 [385, 545]，雷达系统 [162] 和故障检测 [246] 等。

## 6.4 约束总体最小二乘

求解矩阵方程  $\mathbf{A}\mathbf{x} = \mathbf{b}$  的数据最小二乘法和总体最小二乘法虽然考虑了数据矩阵存在观测误差或噪声的情况，但都假定误差随机变量是独立同分布的，并且具有相同的方差。然而，在一些重要的应用中，数据矩阵  $\mathbf{A}$  的噪声分量可能是统计相关的；或者虽然统计不相关，但却具有不同的方差。本节讨论噪声矩阵的列向量统计相关情况下，超定矩阵方程的求解。

### 6.4.1 约束总体最小二乘方法

矩阵方程  $\mathbf{A}_{m \times n} \mathbf{x}_n = \mathbf{b}_m$  可以改写为

$$[\mathbf{A}, \mathbf{b}] \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0} \quad \text{或} \quad \mathbf{C} \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0} \quad (6.4.1)$$

其中  $\mathbf{C} = [\mathbf{A}, \mathbf{b}] \in \mathbb{C}^{m \times (n+1)}$  为增广数据矩阵。

考虑存在于增广数据矩阵中的噪声矩阵  $\mathbf{D} = [\mathbf{E}, \mathbf{e}]$ 。在噪声矩阵的列向量之间存在统计相关的情况下，与总体最小二乘方法一样，有必要使用增广校正矩阵  $\Delta\mathbf{C} = [\Delta\mathbf{A}, \Delta\mathbf{b}]$  抑制噪声矩阵  $\mathbf{D} = [\mathbf{E}, \mathbf{e}]$  的影响，并且校正矩阵的列向量之间也应该统计相关。

使校正矩阵  $\Delta\mathbf{C}$  列向量之间统计相关的简单方法是令每个列向量都与同一个向量（例如  $\mathbf{u}$ ）线性相关

$$\Delta\mathbf{C} = [\mathbf{G}_1 \mathbf{u}, \dots, \mathbf{G}_{n+1} \mathbf{u}] \in \mathbb{R}^{m \times (n+1)} \quad (6.4.2)$$

式中， $\mathbf{G}_i \in \mathbb{R}^{m \times m}$ ,  $i = 1, \dots, n+1$  为已知矩阵，而  $\mathbf{u}$  待确定。

约束总体最小二乘问题可以叙述如次<sup>[1]</sup>：确定一解向量  $\mathbf{x}$  和最小范数扰动向量  $\mathbf{u}$ ，使得

$$(\mathbf{C} + [\mathbf{G}_1 \mathbf{u}, \dots, \mathbf{G}_{n+1} \mathbf{u}]) \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0} \quad (6.4.3)$$

或等价表示成约束优化问题

$$\left. \begin{array}{l} \min_{\mathbf{u}, \mathbf{x}} \mathbf{u}^T \mathbf{W} \mathbf{u} \\ \text{subject to } (\mathbf{A} + \Delta\mathbf{A})\mathbf{x} = \mathbf{b} + \Delta\mathbf{b} \\ [\Delta\mathbf{A}, \Delta\mathbf{b}] = [\mathbf{G}_1 \mathbf{u}, \dots, \mathbf{G}_{n+1} \mathbf{u}] \end{array} \right\} \quad (6.4.4)$$

或者更简洁地写作

$$\min_{\mathbf{u}, \mathbf{x}} \mathbf{u}^T \mathbf{W} \mathbf{u} \quad \text{subject to} \quad (\mathbf{C} + [\mathbf{G}_1 \mathbf{u}, \dots, \mathbf{G}_{n+1} \mathbf{u}]) \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0} \quad (6.4.5)$$

式中,  $\mathbf{W}$  为加权矩阵, 通常取对角矩阵或者单位矩阵。

与总体最小二乘不同, 校正矩阵  $\Delta \mathbf{A}$  约束为  $\Delta \mathbf{A} = [\mathbf{G}_1 \mathbf{u}, \dots, \mathbf{G}_n \mathbf{u}]$ , 而校正向量  $\Delta \mathbf{b}$  约束为  $\Delta \mathbf{b} = \mathbf{G}_{n+1} \mathbf{u}$ 。在约束总体最小二乘问题里, 增广校正矩阵  $[\Delta \mathbf{A}, \Delta \mathbf{b}]$  的列向量之间的线性相关结构通过选择适当的矩阵  $\mathbf{G}_i$  ( $i = 1, \dots, n+1$ ) 得以保持。方法应用的关键是如何根据应用对象, 选择合适的基本矩阵  $\mathbf{G}_i$ 。

式 (6.4.5) 是一个在二次型方程约束下的二次型函数的极小化问题, 它可能没有闭式解, 但是在适当的条件下, 该极小化问题可以转换成一个对极小化变量  $\mathbf{x}$  的无约束极小化问题。

**定理 6.4.1<sup>[1]</sup>** 令

$$\mathbf{W}_x = \sum_{i=1}^n x_i \mathbf{G}_i - \mathbf{G}_{n+1} \quad (6.4.6)$$

则约束总体最小二乘的解向量就是满足下列函数极小化的变量  $\mathbf{x}$

$$\min_{\mathbf{x}} F(\mathbf{x}) = \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}^H \mathbf{C}^H (\mathbf{W}_x \mathbf{W}_x^H)^{\dagger} \mathbf{C} \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} \quad (6.4.7)$$

式中,  $\mathbf{W}_x^{\dagger}$  是  $\mathbf{W}_x$  的 Moore-Penrose 逆矩阵。

文献 [1] 提出了计算约束总体最小二乘解的一种复数形式的 Newton 方法: 将矩阵  $F(\mathbf{x})$  视为  $2n$  个复变量  $x_1, \dots, x_n, x_1^*, \dots, x_n^*$  的复解析函数。

Newton 递推公式如下

$$\mathbf{x} = \mathbf{x}_0 + (\mathbf{A}^* \mathbf{B}^{-1} \mathbf{A} - \mathbf{B}^*)^{-1} (\mathbf{a}^* - \mathbf{A}^* \mathbf{B}^{-1} \mathbf{a}) \quad (6.4.8)$$

式中

$$\left. \begin{aligned} \mathbf{a} &= \frac{\partial F}{\partial \mathbf{x}} = \left[ \frac{\partial F}{\partial x_1}, \frac{\partial F}{\partial x_2}, \dots, \frac{\partial F}{\partial x_n} \right]^T = F \text{ 的复梯度} \\ \mathbf{A} &= \frac{\partial^2 F}{\partial \mathbf{x} \partial \mathbf{x}^T} = F \text{ 的无共轭复 Hessian 矩阵} \\ \mathbf{B} &= \frac{\partial^2 F}{\partial \mathbf{x}^* \partial \mathbf{x}^T} = F \text{ 的共轭复 Hessian 矩阵} \end{aligned} \right\} \quad (6.4.9)$$

两个  $n \times n$  部分 Hessian 矩阵的第  $(k, l)$  元素定义为

$$\left[ \frac{\partial^2 F}{\partial \mathbf{x} \partial \mathbf{x}^T} \right]_{k,l} = \frac{\partial^2 F}{\partial x_k \partial x_l} = \frac{1}{4} \left( \frac{\partial F}{\partial x_{kR}} - j \frac{\partial F}{\partial x_{kI}} \right) \left( \frac{\partial F}{\partial x_{lR}} - j \frac{\partial F}{\partial x_{lI}} \right) \quad (6.4.10)$$

$$\left[ \frac{\partial^2 F}{\partial \mathbf{x}^* \partial \mathbf{x}^T} \right]_{k,l} = \frac{\partial^2 F}{\partial x_k^* \partial x_l} = \frac{1}{4} \left( \frac{\partial F}{\partial x_{kR}} + j \frac{\partial F}{\partial x_{kI}} \right) \left( \frac{\partial F}{\partial x_{lR}} - j \frac{\partial F}{\partial x_{lI}} \right) \quad (6.4.11)$$

式中,  $x_{kR}$  和  $x_{kI}$  分别表示  $x_k$  的实部和虚部。

令

$$\tilde{\mathbf{u}} = (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \mathbf{C} \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} \quad (6.4.12)$$

$$\tilde{\mathbf{B}} = \mathbf{C} \mathbf{I}_{n+1,n} - [\mathbf{G}_1 \mathbf{W}_x^H \tilde{\mathbf{u}}, \dots, \mathbf{G}_n \mathbf{W}_x^H \tilde{\mathbf{u}}] \quad (6.4.13)$$

$$\tilde{\mathbf{G}} = [\mathbf{G}_1^H \tilde{\mathbf{u}}, \dots, \mathbf{G}_n^H \tilde{\mathbf{u}}] \quad (6.4.14)$$

其中,  $\mathbf{I}_{n+1,n}$  是一个  $(n+1) \times n$  对角矩阵, 其对角线元素为 1。于是,  $\mathbf{a}$ ,  $\mathbf{A}$  和  $\mathbf{B}$  可以分别计算如下

$$\mathbf{a} = (\mathbf{u}^H \tilde{\mathbf{B}})^T \quad (6.4.15)$$

$$\mathbf{A} = -\tilde{\mathbf{G}}^H \mathbf{W}_x^H (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \tilde{\mathbf{B}} - (\tilde{\mathbf{G}}^H \mathbf{W}_x^H (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \tilde{\mathbf{B}})^T \quad (6.4.16)$$

$$\mathbf{B} = [\tilde{\mathbf{B}}^H (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \tilde{\mathbf{B}}]^T + \tilde{\mathbf{G}}^H [\mathbf{W}_x^H (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \mathbf{W}_x - \mathbf{I}] \tilde{\mathbf{G}} \quad (6.4.17)$$

业已证明<sup>[1]</sup>, 约束总体最小二乘估计与约束极大似然估计等价。

#### 6.4.2 超分辨谐波恢复

Abatzoglou 等人<sup>[1]</sup> 以谐波信号的超分辨恢复为例, 介绍了约束总体最小二乘的应用。假定有  $L$  个窄带波前信号照射到  $N$  个线性均匀阵列上。阵列信号满足下面的前向线性预测方程<sup>[292]</sup>

$$\mathbf{C}_k \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0}, \quad k = 1, 2, \dots, M \quad (6.4.18)$$

其中

$$\mathbf{C}_k = \left[ \begin{array}{cccc} y_k(1) & y_k(2) & \cdots & y_k(L+1) \\ y_k(2) & y_k(3) & \cdots & y_k(L+2) \\ \vdots & \vdots & \vdots & \vdots \\ y_k(N-L) & y_k(N-L+1) & \cdots & y_k(N) \\ \hline y_k^*(L+1) & y_k^*(L) & \cdots & y_k^*(1) \\ \vdots & \vdots & \vdots & \vdots \\ y_k^*(N) & y_k^*(N-1) & \cdots & y_k^*(N-L) \end{array} \right] \quad (6.4.19)$$

这里,  $y_k(i)$  是第  $k$  个阵列在  $i$  时刻的输出观测值。矩阵  $\mathbf{C}_k$  称为第  $k$  个快拍的数据矩阵。将所有的数据矩阵合成一个数据矩阵  $\mathbf{C}$ , 即

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 \\ \vdots \\ \mathbf{C}_M \end{bmatrix} \quad (6.4.20)$$

于是, 超分辨谐波恢复问题归结为利用约束总体最小二乘求解矩阵方程, 即

$$\mathbf{C} \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} = \mathbf{0}$$

而  $\mathbf{W}_x$  的估计结果由下式给出

$$\hat{\mathbf{W}}_x = \begin{bmatrix} \mathbf{W}_1 & 0 \\ 0 & \mathbf{W}_2 \end{bmatrix}$$

其中

$$\mathbf{W}_1 = \begin{bmatrix} x_1 & x_2 & \cdots & x_L & -1 & 0 \\ \ddots & \ddots & & \ddots & \ddots & \ddots \\ 0 & & x_1 & x_2 & \cdots & x_L & -1 \end{bmatrix}$$

$$\mathbf{W}_2 = \begin{bmatrix} -1 & x_L & \cdots & x_2 & x_1 & 0 \\ \ddots & \ddots & & \ddots & \ddots & \ddots \\ 0 & & -1 & x_L & \cdots & x_2 & x_1 \end{bmatrix}$$

另外

$$\begin{aligned} F(\mathbf{x}) &= \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}^H \mathbf{C}^H (\mathbf{W}_x \mathbf{W}_x^H)^{-1} \mathbf{C} \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix}^H \sum_{m=1}^M \mathbf{C}_m^H (\hat{\mathbf{W}}_x \hat{\mathbf{W}}_x^H)^{-1} \mathbf{C}_m \begin{bmatrix} \mathbf{x} \\ -1 \end{bmatrix} \end{aligned} \quad (6.4.21)$$

为了估计相关的波数  $\phi_i$ , 约束总体最小二乘方法分为三步:

- (1) 利用 6.4.1 节介绍的 Newton 方法求解式 (6.4.21), 得到  $\mathbf{x}$ ;
- (2) 计算线性预测系数多项式

$$\sum_{k=1}^L x_k z^{k-1} - z^L = 0 \quad (6.4.22)$$

- (3) 估计对应的角度  $\phi_i = \arg(z_i)$ ,  $i = 1, \dots, L$ 。

### 6.4.3 正则化约束总体最小二乘图像恢复

退化图像的恢复是一个重要的问题, 因为它能够从观测到的退化图像数据恢复丢失的信息。图像恢复的目的是在已知记录数据和某些先验知识的情况下, 求原始图像的最优解。

令  $N \times 1$  点扩展函数 (point-spread function, PSF) 表示为

$$\mathbf{h} = \bar{\mathbf{h}} + \Delta\mathbf{h} \quad (6.4.23)$$

式中,  $\bar{\mathbf{h}}, \Delta\mathbf{h} \in \mathbb{R}^N$  分别是点扩展函数的已知部分和 (未知的) 误差部分。误差分量  $\Delta\mathbf{h} = [\Delta h(0), \Delta h(1), \dots, \Delta h(N-1)]^T$  为独立同分布噪声, 均值为 0, 方差为  $\sigma_h$ 。

观测到的退化图像用向量  $\mathbf{g}$  表示, 成像方程可用矩阵-向量形式表示为

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \Delta\mathbf{g} \quad (6.4.24)$$

式中,  $f$  和  $\Delta g \in \mathbb{R}^N$  分别表示原始图像和观测图像的加性噪声。加性噪声  $\Delta g = [\Delta g(0), \Delta g(1), \dots, \Delta g(N-1)]^T$  也是独立同分布噪声, 并与点扩展函数的误差分量  $\Delta h$  统计不相关。矩阵  $H \in \mathbb{R}^{N \times N}$  表示点扩展矩阵, 由已知部分  $\bar{H}$  和误差部分组成, 即

$$H = \bar{H} + \Delta H \quad (6.4.25)$$

式 (6.4.24) 的总体最小二乘解为

$$f = \arg \min_{[\bar{H}, \hat{g}] \in \mathbb{R}^{N \times (N+1)}} \| [H, g] - [\bar{H}, \hat{g}] \|_F^2 \quad (6.4.26)$$

式中,  $\hat{g}$  服从约束条件

$$\hat{g} \in \text{Range}(\bar{H}) \quad (6.4.27)$$

通过定义未知的归一化噪声向量  $u \in \mathbb{R}^{2N}$  (由  $\Delta h$  和  $\Delta g$  组成), 即

$$u = \left[ \frac{\Delta h(0)}{\sigma_h}, \dots, \frac{\Delta h(N-1)}{\sigma_h}, \frac{\Delta g(0)}{\sigma_g}, \dots, \frac{\Delta g(N-1)}{\sigma_g} \right]^T \quad (6.4.28)$$

Mesarovic 等人<sup>[341]</sup> 提出了基于约束总体最小二乘的图像恢复算法

$$f = \arg \min_f \left\{ \|u\|_2^2 \right\} \quad \text{subject to } \bar{H}f - g + Lu = 0 \quad (6.4.29)$$

式中,  $L$  是一个  $N \times 2N$  矩阵, 定义为

$$L = \begin{bmatrix} \sigma_h f(0) & \sigma_h f(N-1) & \cdots & \sigma_h f(1) & | & \sigma_g & 0 & \cdots & 0 \\ \sigma_h f(1) & \sigma_h f(0) & \cdots & \sigma_h f(2) & | & 0 & \sigma_g & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots \\ \sigma_h f(N-1) & \sigma_h f(N-2) & \cdots & \sigma_h f(0) & | & 0 & 0 & \cdots & \sigma_g \end{bmatrix} \quad (6.4.30)$$

在给定被记录的数据向量  $g$  和点扩展矩阵已知部分  $\bar{H}$  的情况下, 式 (6.4.24) 的原始图像  $f$  的求解是一个典型的逆问题。因此, 图像恢复问题的求解在数学上对应为式 (6.4.24) 的逆变换的存在性和唯一性。若逆变换不存在, 则称图像恢复这一逆问题为奇异的。另外, 逆变换虽然存在, 但是其解有可能不唯一, 而是有一组解。对一个实际的物理问题而言, 这种非唯一解是不可接受的。此时, 称图像恢复为病态的逆问题。这意味着, 观测数据向量  $g$  中的小扰动有可能导致恢复图像  $g$  中的大扰动<sup>[20, 473]</sup>。

克服图像恢复病态问题的有效方法之一是使用正则化方法<sup>[473, 135]</sup>, 得到正则化约束总体最小二乘算法<sup>[135, 180]</sup>。

正则化约束总体最小二乘图像恢复算法的基本思想是引入正则化算子  $Q$  和正则化参数  $\lambda > 0$ , 将最小化的目标函数替换为两个互补函数之和, 即式 (6.4.29) 变成

$$f = \arg \min_f \left\{ \|u\|_2^2 + \lambda \|Qf\|_2^2 \right\} \quad \text{subject to } \bar{H}f - g + Lu = 0 \quad (6.4.31)$$

正则化参数的选择需要兼顾观测数据的保真度和解的平滑性。为了进一步改善正则化约束总体最小二乘图像恢复算法的性能, Chen 等人<sup>[106]</sup> 提出了自适应选择正则化参

数  $\lambda$  的方法，并称为自适应正则化约束总体最小二乘图像恢复算法。这一算法的解为

$$\mathbf{f} = \arg \min_{\mathbf{f}} \left\{ \|\mathbf{u}\|_2^2 + \lambda(\mathbf{f}) \|\mathbf{Qf}\|_2^2 \right\} \quad \text{subject to } \bar{\mathbf{H}}\mathbf{f} - \mathbf{g} + \mathbf{L}\mathbf{u} = \mathbf{0} \quad (6.4.32)$$

以上介绍了约束总体最小二乘在图像恢复中的应用原理。限于篇幅，对以上两种算法的实现，不再赘述，感兴趣的读者可分别参考文献 [341] 和文献 [106]。

## 6.5 盲矩阵方程求解的子空间方法

考虑盲矩阵方程

$$\mathbf{X} = \mathbf{AS} \quad (6.5.1)$$

其中  $\mathbf{X}_{N \times M}$  为复矩阵，其元素为观测数据；而复矩阵  $\mathbf{A}_{N \times d}$  和  $\mathbf{S}_{d \times M}$  均未知。盲矩阵方程求解的问题是：在只已知  $\mathbf{X}$  的情况下，能够求出未知矩阵  $\mathbf{S}$  吗？答案是肯定的，但需要假定两个条件：矩阵  $\mathbf{A}$  满列秩和  $\mathbf{S}$  满行秩。这两个假设条件在工程问题中往往是满足的。例如，在阵列信号处理中，矩阵  $\mathbf{A}$  满列秩意味着各个信号的波达方向是独立的，而矩阵  $\mathbf{S}$  满行秩则要求各个源信号是独立发射的。

假定  $N$  为数据长度， $d$  为源信号个数， $M$  为传感器个数，通常取  $M \geq d$  和  $N > M$ 。定义数据矩阵  $\mathbf{X}$  的截尾奇异值分解为

$$\mathbf{X} = \hat{\mathbf{U}} \hat{\Sigma} \hat{\mathbf{V}}^H \quad (6.5.2)$$

式中， $\hat{\Sigma}$  是包含了  $d$  个主要奇异值的  $d \times d$  对角矩阵。由于  $\text{Col}(\mathbf{A}) = \text{Col}(\hat{\mathbf{U}})$ ，即矩阵  $\mathbf{A}$  和  $\hat{\mathbf{U}}$  二者张成同一个信号子空间，故有

$$\hat{\mathbf{U}} = \mathbf{AT} \quad (6.5.3)$$

式中， $\mathbf{T}$  是一个  $d \times d$  非奇异矩阵。

令  $\mathbf{W}$  是一个  $d \times N$  复矩阵，它代表一神经网络或者滤波器。用  $\mathbf{W}$  左乘式 (6.5.1)，得到

$$\mathbf{WX} = \mathbf{WAS}$$

若调整矩阵  $\mathbf{W}$  使得  $\mathbf{WA} = \mathbf{I}_d$ ，则立即得到方程式 (6.5.1) 的解

$$\mathbf{S} = \mathbf{WX} \quad (6.5.4)$$

为了求  $\mathbf{W}$ ，计算

$$\mathbf{W}\hat{\mathbf{U}} = \mathbf{WAT} = \mathbf{T}$$

立即有

$$\mathbf{W} = \mathbf{T}\hat{\mathbf{U}}^H \quad (6.5.5)$$

总结以上讨论, 可以得到求解盲矩阵方程式 (6.5.1) 的以下方法

$$\left. \begin{array}{ll} \text{数据模型} & \mathbf{X} = \mathbf{AS} \\ \text{截尾 SVD} & \mathbf{X} = \hat{\mathbf{U}}\hat{\Sigma}\hat{\mathbf{V}} \\ \text{求解} & \hat{\mathbf{U}} = \mathbf{AT} \text{ 得到 } \mathbf{T} \\ \text{方程的解} & \mathbf{S} = (\mathbf{T}\hat{\mathbf{U}}^T)\mathbf{X} \end{array} \right\} \quad (6.5.6)$$

由于这种方法利用了信号子空间, 故称为子空间方法。因此, 盲矩阵方程求解的关键问题是如何在  $\mathbf{A}$  和  $\mathbf{T}$  均未知的情况下, 从  $\hat{\mathbf{U}} = \mathbf{AT}$  求出非奇异矩阵  $\mathbf{T}$ 。

下面以无线通信为例, 介绍盲矩阵方程式 (6.5.1) 的求解。

在不考虑无线通信中的多径传输的情况下, 方程式 (6.5.1) 中的矩阵为<sup>[503]</sup>

$$\mathbf{A} = \mathbf{A}_\theta \mathbf{B} \quad (6.5.7)$$

式中

$$\mathbf{A}_\theta = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_d)] = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \theta_1 & \theta_2 & \cdots & \theta_d \\ \vdots & \vdots & \ddots & \vdots \\ \theta_1^{N-1} & \theta_2^{N-1} & \cdots & \theta_d^{N-1} \end{bmatrix}$$

$$\mathbf{B} = \text{daig}(\beta_1, \beta_2, \dots, \beta_d)$$

式中,  $\theta_i$  和  $\beta_i$  分别是第  $i$  个用户信号的波达方向和衰减系数, 它们都是未知的。

定义对角矩阵

$$\boldsymbol{\Theta} = \text{diag}(\theta_1, \theta_2, \dots, \theta_d) \quad (6.5.8)$$

和  $(M-1) \times M$  维选择矩阵

$$\mathbf{J}_1 = [\mathbf{I}_{M-1}, \mathbf{0}], \quad \mathbf{J}_2 = [\mathbf{0}, \mathbf{I}_{M-1}]$$

它们分别选出矩阵  $\mathbf{A}_\theta$  的上面  $M-1$  行和下面  $M-1$  行。易知

$$(\mathbf{J}_1 \mathbf{A}_\theta) \boldsymbol{\Theta} = \mathbf{J}_2 \mathbf{A}_\theta \quad (6.5.9)$$

于是, 有

$$\hat{\mathbf{U}} = \mathbf{AT} = \mathbf{A}_\theta \mathbf{BT} \quad (6.5.10)$$

为了求出非奇异矩阵  $\mathbf{T}$ , 若用选择矩阵  $\mathbf{J}_1$  和  $\mathbf{J}_2$  分别左乘式 (6.5.10), 并令  $\mathbf{A}'_\theta = \mathbf{J}_1 \mathbf{A}_\theta$ , 然后利用式 (6.5.9), 则得

$$\hat{\mathbf{U}}_1 = \mathbf{J}_1 \hat{\mathbf{U}} = (\mathbf{J}_1 \mathbf{A}_\theta) \mathbf{BT} = \mathbf{A}'_\theta \mathbf{BT} \quad (6.5.11)$$

$$\hat{\mathbf{U}}_2 = \mathbf{J}_2 \hat{\mathbf{U}} = (\mathbf{J}_2 \mathbf{A}_\theta) \mathbf{BT} = \mathbf{A}'_\theta \boldsymbol{\Theta} \mathbf{BT} \quad (6.5.12)$$

由于  $\mathbf{B}$  和  $\boldsymbol{\Theta}$  都是对角矩阵, 故  $\boldsymbol{\Theta}\mathbf{B} = \mathbf{B}\boldsymbol{\Theta}$ , 从而有

$$\hat{\mathbf{U}}_2 = \mathbf{A}'_{\theta}\mathbf{B}\boldsymbol{\Theta}\mathbf{T} = \mathbf{A}'_{\theta}\mathbf{B}\mathbf{T}\mathbf{T}^{-1}\boldsymbol{\Theta}\mathbf{T} = \hat{\mathbf{U}}_1\mathbf{T}^{-1}\boldsymbol{\Theta}\mathbf{T}$$

或写作

$$\hat{\mathbf{U}}_1^{\dagger}\hat{\mathbf{U}}_2 = \mathbf{T}^{-1}\boldsymbol{\Theta}\mathbf{T} \quad (6.5.13)$$

式中,  $\hat{\mathbf{U}}_1^{\dagger} = (\hat{\mathbf{U}}_1^H\hat{\mathbf{U}}_1)^{-1}\hat{\mathbf{U}}_1^H$  是矩阵  $\hat{\mathbf{U}}_1$  的广义逆矩阵。

由于  $\boldsymbol{\Theta}$  为对角矩阵, 易知式 (6.5.13) 是一典型的相似变换。因此, 通过对矩阵  $\hat{\mathbf{U}}_1^{\dagger}\hat{\mathbf{U}}_2$  进行相似变换, 即可得到非奇异矩阵  $\mathbf{T}$ 。

以上讨论可以总结为求解盲矩阵方程  $\mathbf{X} = \mathbf{A}_{\theta}\mathbf{B}$  的下列算法:

- (1) 计算矩阵  $\mathbf{X}$  的截尾奇异值分解  $\mathbf{X} = \hat{\mathbf{U}}\hat{\Sigma}\hat{\mathbf{V}}^H$ 。
- (2) 抽取矩阵  $\hat{\mathbf{U}}$  的上面  $M - 1$  行组成  $\hat{\mathbf{U}}_1$ , 下面  $M - 1$  行组成  $\hat{\mathbf{U}}_2$ 。
- (3) 对矩阵  $\hat{\mathbf{U}}_1^{\dagger}\hat{\mathbf{U}}_2$  进行相似变换, 得到非奇异矩阵  $\mathbf{T}$ 。
- (4) 矩阵方程  $\mathbf{X} = \mathbf{A}_{\theta}\mathbf{B}$  的解为

$$\mathbf{B} = (\hat{\mathbf{U}}^H\mathbf{T})\mathbf{X}$$

虽然上面只是介绍了单路径传输的情况, 但是求解矩阵方程  $\mathbf{X} = \mathbf{AS}$  的子空间方法也适用于多径传输的情况。不同的只是矩阵  $\mathbf{A}$  的形式不同, 从而使得求非奇异矩阵  $\mathbf{T}$  的方法也有所不同。限于篇幅, 这里不再赘述, 有兴趣的读者可进一步参考文献 [503] 或文献 [545, p.365-367]。

## 6.6 非负矩阵分解的优化理论

全部元素为非负实数的矩阵称为非负矩阵。已知矩阵  $\mathbf{X}$  以及两个未知矩阵  $\mathbf{A}, \mathbf{S}$  均为非负矩阵的矩阵方程  $\mathbf{X} = \mathbf{AS}$  称为盲非负矩阵方程, 它广泛存在于工程应用问题中。

### 6.6.1 非负性约束与稀疏性约束

在很多工程应用问题中, 常常需要对数据施加两个约束: 非负性约束和稀疏性约束。

非负性约束就是约束数据是非负的。实际的数据很多本来就是非负的, 它们组成非负矩阵。非负矩阵广泛存在于日常生活中, 下面是非负矩阵的四种重要的实际例子 [296]:

(1) 在文本采集中, 文本被存储为一个个向量, 每个文本向量的元素是某个相关的术语 (term) 在该文本中出现的次数的计数。将文本向量一个接一个堆栈起来, 就构成了一个非负的“术语  $\times$  文本”矩阵。

(2) 在图像采集中, 每一图像都用向量表示, 向量的每一个元素对应为一个像素。像素的强度和颜色由非负数值给出, 由此形成了非负的“像素  $\times$  图像”矩阵。

(3) 对于商品设定 (item sets) 或推荐系统, 顾客的购买记录或评分以非负的稀疏矩阵的形式存储。

(4) 在基因表示分析中, “基因  $\times$  实验” 矩阵是通过观测在某些实验条件下所产生的基因序列构造的。

此外, 在模式识别和信号处理中, 对于某个特定的模式或目标信号而言, 所有特征向量的线性组合很可能不太合适。相反, 某些特征向量的部分组合则更合适。例如, 人脸识别中, 强调眼睛、鼻子、嘴唇等特定部分的组合往往更加有效。在所有组合中, 正的和负的组合系数分别强调部分特征的正面和负面作用, 而零组合系数意味着某些特征不起作用。与之不同, 在部分组合中, 只有起作用和不起作用的两类特征。因此, 为了强调某些主要特征的作用, 很自然地会对系数向量中的元素加上非负性约束。

稀疏性约束就是约束数据不是稠密的, 而是稀疏的, 即大多数的数据取零值, 只有很少的数据取非零值。一个大多数元素取零, 少部分元素取非零值的矩阵称为稀疏矩阵; 而元素只取非负值的稀疏矩阵称为非负稀疏矩阵。例如, 商品推荐系统中的顾客的购买或者评分组成的矩阵就是非负稀疏矩阵。在经济学中, 很多变量和数据 (例如, 成交量和价格等) 既是稀疏的, 又是非负的。稀疏性约束可以增加投资组合的有效性, 而非负性约束则既可以提高投资的有效性, 又能够降低投资风险 [535, 448]。另外, 很多自然界的信号与图像本身虽然不是稀疏的, 但是经过某种变换之后, 在变换域内却是稀疏的。例如, 人脸和医学图像的离散余弦变换 (DCT) 就是典型的稀疏数据; 语音信号的短时 Fourier 变换在时频域内是稀疏的。

### 6.6.2 非负矩阵分解的数学模型及解释

线性数据分析的基本问题是: 通过某种适当的变换或分解, 将高维的原始数据向量表示成一组低维向量的线性组合。由于抽取了原数据向量的本质或特征, 可以用来进行模式识别, 所以这组低维向量常称为原数据的“模式向量”或“基(本)向量”或“特征向量”。

在进行数据分析、建模和处理时, 通常必须考虑模式向量的两个基本要求:

(1) 可解释性 (interpretability) 模式向量的分量应该具有明确的物理或者生理意义和含义。

(2) 统计保真度 (statistical fidelity) 当数据一致和没有太多误差或噪声时, 模式向量的分量应该可以解释数据的方差 (主要能量分布)。

矢量量化 (VQ: vector quantization) 和主分量分析是两种广泛被使用的非监督学习算法, 它们采用根本不同的方式对数据进行编码。

#### 1. 矢量量化法

矢量量化法使用存储的模式 (prototype) 向量作为码矢 (codevector)。令  $c_n$  是  $k$  维

码矢，并共存储有  $N$  个码矢，即

$$\mathbf{c}_n = [c_{n,1}, c_{n,2}, \dots, c_{n,k}]^T, \quad n = 1, \dots, N$$

$N$  个码矢的集合  $\{\mathbf{c}_1, \dots, \mathbf{c}_N\}$  组成码书 (codebook)。

所有与存储的模式向量即码矢  $\mathbf{c}_n$  最接近的数据向量组成的区域称为码矢  $\mathbf{c}_n$  的编码区 (encoding region)，定义为

$$S_n = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{c}_n\|^2 \leq \|\mathbf{x} - \mathbf{c}_{n'}\|^2, \forall n' = 1, \dots, N\} \quad (6.6.1)$$

矢量量化问题的提法是：给定  $M$  个  $k$  维数据向量  $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \dots, x_{i,k}]^T, i = 1, \dots, M$ ，确定这些向量所在的编码区，即这些向量各自对应的码矢。

令  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_M] \in \mathbb{R}^{k \times M}$  为数据矩阵， $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_N] \in \mathbb{R}^{k \times N}$  表示码书矩阵，则数据矩阵的矢量量化可以用以下模型描述

$$\mathbf{X} = \mathbf{CS} \quad (6.6.2)$$

其中， $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_M] \in \mathbb{R}^{N \times M}$  为量化系数矩阵，其列称为量化系数向量。

从最优化的角度看问题，矢量量化的优化准则是“胜者赢得一切”(winner-take-all)，将输入数据聚类到互相排斥的模式<sup>[188, 308]</sup>。从编码的观点出发，矢量量化为“祖母细胞编码”(grandmother cell coding)，每一个数据只由一个基向量解释 (即数据被聚类)<sup>[512]</sup>。具体而言，每一个量化系数向量都是一个只有一个元素为 1，其他元素皆等于零的  $N$  维基本向量。因此，码书矩阵第  $j$  列的第  $i$  个元素等于 1，表明数据向量  $\mathbf{x}_j$  被判断为与码矢  $\mathbf{c}_i$  最接近，即一个数据向量只对应一个码矢。矢量量化可以捕获输入数据的非线性结构，但是其捕获能力比较弱，因为矢量量化法中数据向量与码矢是一对一对应的。如果数据的维数很大，就需要大量的码矢才能表示输入数据。

## 2. 主分量分析法

线性数据模型是广泛使用的一种数据模型，包括主分量分析 (principal component analysis, PCA)，线性判别分析 (linear discriminant analysis, LDA) 和独立分量分析 (independent component analysis, ICA) 等多元数据分析方法 (multivariate data analysis) 都采用这一线性数据模型。

与矢量量化由码矢组成码书类似，主分量分析方法由一组与主要分量对应的相互正交的基向量  $\mathbf{a}_i$  组成基矩阵  $\mathbf{A}$ 。这些基向量称为模式或特征向量。对于一数据向量  $\mathbf{x}$ ，主分量分析采用分享的约束原则进行优化，用模式向量的线性组合  $\mathbf{x} = \mathbf{As}$  表示输入数据。从编码的角度看，主分量分析是一种分布式编码。与祖母细胞编码的矢量量化法相比，主分量分析法由于采用分布式编码，所以只需要较少的基向量就可以表示大维数的数据。

主分量分析法的缺点是：

- (1) 不能捕获输入数据的非线性结构。

(2) 虽然基向量可以统计解释为最大差异的方向, 但许多方向并没有一个明显的视觉解释, 这是因为基矩阵  $A$  和量化系数向量  $s$  的元素可以取零、正和负的符号。由于基向量用于线性组合, 而这种组合涉及正、负数之间的复杂对消, 所以许多单个的基向量由于被对消而失掉了直观的物理意义, 对于非负数据(例如彩色图像的像素值)不具有可解释性。这是因为, 非负数据的模式向量的元素应该都是非负的数值, 但是相互正交的特征向量不可能都含有非负的元素: 如果与最大特征值对应的特征向量  $u_1$  的元素全部是非负的, 则其他与之正交的特征向量  $u_j, j \neq 1$  就必然含有负的元素, 否则两个向量的正交条件  $\langle u_1, u_j \rangle = 0, j \neq 1$  不可能成立。这一事实表明, 相互正交的特征向量不能用作非负数据分析的模式向量或基向量。

在主分量分析、线性判别分析和独立分量分析等方法中, 系数向量的元素通常多取正和负值, 鲜有取零值。这意味着在这些方法中, 所有基向量都参与观测数据向量的拟合或者回归。与这些方法不同, 非负矩阵分解 (non-negative matrix factorization, NMF) 中, 由于对基向量和系数向量的元素均作非负的约束, 所以容易想象, 此时参与拟合或者回归观测数据向量的基向量的个数肯定比较少。从这一角度讲, 非负矩阵分解有抽取主要基向量的作用。

非负矩阵分解的另一个突出优点是: 对组合因子的非负约束有利于产生稀疏的编码, 即很多编码值为零。在生物学中, 人脑就是以这种稀疏编码的方式对信息进行编码的<sup>[165]</sup>。

因此, 作为线性数据分析的另一类方法, 应该在使数据重构误差最小化时, 撤销对基向量的正交化约束, 而改为非负性约束。

非负矩阵分解是一种线性、非负逼近的数据表示。令  $x(j) = [x_1(j), \dots, x_I(j)]^T \in \mathbb{R}_+^{I \times 1}$  和  $s(j) = [s_1(j), \dots, s_K(j)]^T \in \mathbb{R}_+^{K \times 1}$  分别代表用  $I$  个传感器测得的离散时间  $j$  的非负数据向量和  $K$  维非负系数向量, 其中  $\mathbb{R}_+$  表示非负象限。非负数据向量的数学模型如下式所示

$$\begin{bmatrix} x_1(j) \\ \vdots \\ x_I(j) \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{I1} & \cdots & a_{IK} \end{bmatrix} \begin{bmatrix} s_1(j) \\ \vdots \\ s_K(j) \end{bmatrix} \quad \text{或 } x(j) = As(j) \quad (6.6.3)$$

式中,  $A = [a_1, \dots, a_K] \in \mathbb{R}^{I \times K}$  称为基矩阵,  $s$  称为系数向量。

基矩阵的各个列向量  $a_k, k = 1, \dots, K$  称为基向量。由于不同时刻的测量向量  $x(j)$  都用相同的一组基向量  $a_k, k = 1, \dots, K$  表示, 所以这些  $I$  维基向量可以想象成数据表示的积木块, 而  $K$  维系数向量  $s(j)$  的元素  $s_k(j)$  则表示第  $k$  个基向量(积木块)  $a_k$  在数据向量  $x(j)$  中的存在强度, 体现对应的基向量  $a_k$  在观测向量  $x$  的拟合或回归中的贡献。因此, 系数向量的元素  $s_k(j)$  常称为拟合系数、回归系数或组合系数等。 $s_k(j) > 0$  表示基向量  $a_k$  的贡献为加法组合即正面作用; $s_k(j) = 0$  表示相对应的基向量的零贡献, 即不参与拟合或回归; 而负的系数  $s_k(j) < 0$  则意味着基向量的减法组合, 起着负面的组合作用。

如果将  $j = 1, \dots, J$  个离散时间的非负观测数据向量排列成一个非负的观测矩阵,

则有

$$[\mathbf{x}(1), \dots, \mathbf{x}(J)] = \mathbf{A}[\mathbf{s}(1), \dots, \mathbf{s}(J)] \implies \mathbf{X} = \mathbf{AS} \quad (6.6.4)$$

矩阵  $\mathbf{S}$  称为系数矩阵。系数矩阵本质上是基矩阵的编码矩阵。

盲非负矩阵方程  $\mathbf{X} = \mathbf{AS}$  的求解问题可以叙述为：给定一个非负矩阵  $\mathbf{X} \in \mathbb{R}_+^{I \times J}$  (其元素  $x_{ij} \geq 0$ ) 和一个低秩  $r < \min\{I, J\}$ , 由  $\mathbf{X}$  求未知的稀疏矩阵  $\mathbf{A} \in \mathbb{R}_+^{I \times r}$  及  $\mathbf{S} \in \mathbb{R}_+^{r \times J}$ , 使得

$$\mathbf{X} = \mathbf{AS} + \mathbf{N} \quad (6.6.5)$$

或者

$$\mathbf{X}_{ij} = [\mathbf{AS}]_{ij} + \mathbf{N}_{ij} = \sum_{k=1}^r a_{ik} s_{kj} + n_{ij} \quad (6.6.6)$$

其中  $\mathbf{N} \in \mathbb{R}^{I \times J}$  为逼近误差矩阵。系数矩阵  $\mathbf{S}$  也称编码变量矩阵 (encoding variable matrix), 其元素为未知的隐匿非负分量。

将一个非负的数据矩阵分解为非负的基矩阵与非负的系数矩阵的乘积这一问题习惯称为非负矩阵分解。非负矩阵分解由 Lee 和 Seung 于 1999 年在 Nature 上提出 [307], 它本质上是一种线性的、非负的数据表示。如果数据矩阵  $\mathbf{X}$  是正的矩阵, 则要求基矩阵  $\mathbf{A}$  和系数矩阵  $\mathbf{S}$  也都是正的矩阵。这样的矩阵分解称为正矩阵分解 (positive matrix factorization, PMF), 由 Paatero 与 Tapper 于 1994 年提出 [386]。在非负矩阵分解的范畴里, 矩阵  $\mathbf{X}$ 、 $\mathbf{A}$  和  $\mathbf{S}$  分别称为数据矩阵、基矩阵和系数矩阵。

式 (6.6.5) 表明, 当数据矩阵  $\mathbf{X}$  的秩  $r = \text{rank}(\mathbf{X}) < \min\{I, J\}$  时, 非负矩阵逼近  $\mathbf{AS}$  可以想象成数据矩阵  $\mathbf{X}$  的一种压缩和去噪形式。

非负矩阵分解的另一个重要特征是它的分布式非负编码和部位组合能力。

非负矩阵分解不允许矩阵分解因子  $\mathbf{A}$  和  $\mathbf{S}$  中出现负的元素, 其优点是: 与矢量量化的单一约束不同, 非负约束允许采用多个基图像或特征脸的组合表示一张人脸图像; 与主分量分析不同, 非负矩阵分解只允许加法组合, 因为  $\mathbf{A}$  和  $\mathbf{S}$  的非零元素全部都是正的, 从而避免了主分量分析中的基图像之间的任何减法组合的发生。就优化准则而言, 非负矩阵分解采用分享约束加非负约束。从编码的观点看, 非负矩阵分解是一种分布式的非负编码, 常常可以导致稀疏编码。

由于这些优点, 使得非负矩阵分解给人的直觉印象是: 不是将所有特征进行组合, 而是将部分特征 (简称部位, parts) 组合成一个 (目标) 整体。从机器学习的角度看, 非负矩阵分解是一种基于部位组合表示的机器学习方法, 具有抽取主要特征的能力。

非负矩阵分解的第三个主要特征是它的多线性数据分析能力。主分量分析使用所有特征基向量的线性组合表示数据, 只能提取数据的线性结构。与之不同, 非负矩阵分解使用不同数量和不同标记的基向量 (部位) 的组合表示数据, 所以可以抽取数据的多线性结构, 具有一定的非线性数据分析能力。

表 6.6.1 有助于理解矢量量化、主分量分析与非负矩阵分解之间的联系与区别。

表 6.6.1 矢量量化、主分量分析与非负矩阵分解的比较

方法	矢量量化 (VQ)	主分量分析 (PCA)	非负矩阵分解 (NMF)
约束条件	胜者赢得一切 (独享)	全体分享	少数个体分享 + 非负性
组成	码书矩阵 $C$ , 量化系数矩阵 $S$	基矩阵 $A$ , 系数矩阵 $S$	基矩阵 $A$ , 系数矩阵 $S$
数学模型	模式聚类: $X = CS$	线性组合: $X = AS$	非负分解: $X = AS$
结构特点	量化系数向量 $s_j$ : 码矢 $c_i$	基向量 $a_k$ : 相互正交	基矩阵 $A$ 、系数矩阵 $S$ : 非负矩阵
分析能力	非线性分析	线性分析	多线性分析
编码方式	祖母细胞编码	分布式编码	分布式非负编码 (稀疏编码)
机器学习	单一模式学习	分布式学习	部位组合学习

### 6.6.3 散度与变形对数

非负矩阵分解本质上是一个最优化问题，常采用某种散度作为代价函数。

两个概率密度  $p$  和  $q$  之间的距离  $D(p\|q)$  称为散度 (divergences)，若它只满足非负性和正定性条件  $D(p\|q) \geq 0$  (等号成立当且仅当  $p = q$ )。

根据噪声统计分布的先验知识，非负矩阵分解常用的散度有 Kullback-Leibler 散度和 Alpha-Beta (AB)-散度等。

#### 1. 平方 Euclidean 距离

当逼近误差服从正态分布时，非负矩阵分解一般使用误差矩阵的平方 Euclidean 距离作代价函数

$$D_E(X\|AS) = \|X - AS\|_2^2 = \frac{1}{2} \sum_{i=1}^I \sum_{j=1}^J (x_{ij} - [AS]_{ij})^2 \quad (6.6.7)$$

其中  $[AS]_{ij}$  表示矩阵乘积  $AS$  第  $i$  行、第  $j$  列的元素。虽然优化问题对  $A$  和  $S$  分别是凸的，但是当两个矩阵同时作为变元时，优化问题却不是凸的。

在很多应用中，除了非负约束  $a_{ik} \geq 0; s_{kj} \geq 0, \forall i, j, k$  之外，往往还会要求期望解  $S$  或  $A$  分别具有某种与应用有关的特性  $J_S(S)$  和  $J_A(A)$ 。这些特性分别称为系数矩阵  $X$  和基矩阵  $A$  的正则化函数，所得到的代价函数

$$D_E(X\|AS) = \frac{1}{2} \|X - AS\|_2^2 + \alpha_A J_A(A) + \alpha_S J_S(S) \quad (6.6.8)$$

称为正则化平方 Euclidean 距离代价函数。其中， $\alpha_S$  和  $\alpha_A$  为正则化参数。

#### 2. Kullback-Leibler 散度

令  $\phi: \mathcal{D} \rightarrow \mathbb{R}$  为定义在闭合凸集  $\mathcal{D} \subseteq \mathbb{R}_+^K$  的一连续可微分凸函数。与函数  $\phi$  对应的两个向量  $x, g \in \mathcal{D}$  之间的 Bregman 距离记作  $B_\phi(x\|g)$ ，定义为

$$B_\phi(x\|g) \stackrel{\text{def}}{=} \phi(x) - \phi(g) - \langle \nabla \phi(g), x - g \rangle \quad (6.6.9)$$

其中  $\nabla\phi(\mathbf{x})$  是函数  $\phi$  在  $\mathbf{x}$  的梯度。特别地, 若  $\phi$  取凸函数

$$\phi(\mathbf{x}) = \sum_{i=1}^K x_i \ln x_i \quad (6.6.10)$$

则 Bregman 距离改称为 Kullback-Leibler 散度, 记作  $D_{KL}(\mathbf{x}\|\mathbf{g})$ 。

在概率论和信息论中, Kullback-Leibler 散度又称信息散度 (information divergence)、信息增益、相对熵 (relative entropy), 常被简称为 KL 散度或 I 散度。

对于一随机过程的两个概率分布矩阵  $\mathbf{P}, \mathbf{G} \in \mathbb{R}^{I \times J}$ , 假定它们是两个非负矩阵, 则它们之间的 KL 散度用符号  $D_{KL}(\mathbf{P}\|\mathbf{G})$  表示, 定义为

$$D_{KL}(\mathbf{P}\|\mathbf{G}) = \sum_{i=1}^I \sum_{j=1}^J \left( p_{ij} \ln \frac{p_{ij}}{g_{ij}} - p_{ij} + g_{ij} \right) \quad (6.6.11)$$

容易验证, KL 散度  $D_{KL}(\mathbf{P}\|\mathbf{G}) \geq 0$ , 并且  $D_{KL}(\mathbf{P}\|\mathbf{G}) \neq D_{KL}(\mathbf{G}\|\mathbf{P})$ , 不具有对称性。

### 3. AB-散度

Alpha-Beta 散度简称 AB-散度, 是一个以  $\alpha$  和  $\beta$  为参数的距离函数。AB-散度包括了大多数的散度为特例。

令  $\mathbf{P}, \mathbf{G} \in \mathbb{R}^{I \times J}$  是两个非负的测量矩阵, 则它们之间的 AB-散度定义为 [17, 110, 111]

$$D_{AB}^{(\alpha, \beta)}(\mathbf{P}\|\mathbf{G}) = \begin{cases} -\frac{1}{\alpha\beta} \sum_{i=1}^I \sum_{j=1}^J \left( p_{ij}^\alpha g_{ij}^\beta - \frac{\alpha}{\alpha+\beta} p_{ij}^{\alpha+\beta} - \frac{\beta}{\alpha+\beta} g_{ij}^{\alpha+\beta} \right), & \alpha, \beta, \alpha + \beta \neq 0 \\ \frac{1}{\alpha^2} \sum_{i=1}^I \sum_{j=1}^J \left( p_{ij}^\alpha \ln \frac{p_{ij}^\alpha}{g_{ij}^\alpha} - p_{ij}^\alpha + g_{ij}^\alpha \right), & \alpha \neq 0, \beta = 0 \\ \frac{1}{\alpha^2} \sum_{i=1}^I \sum_{j=1}^J \left( \ln \frac{g_{ij}^\alpha}{p_{ij}^\alpha} + \left( \frac{g_{ij}^\alpha}{p_{ij}^\alpha} \right)^{-1} - 1 \right), & \alpha = -\beta \neq 0 \\ \frac{1}{\beta^2} \sum_{i=1}^I \sum_{j=1}^J \left( g_{ij}^\beta \ln \frac{g_{ij}^\beta}{p_{ij}^\beta} - g_{ij}^\beta + p_{ij}^\beta \right), & \alpha = 0, \beta \neq 0 \\ \frac{1}{2} \sum_{i=1}^I \sum_{j=1}^J (\ln p_{ij} - \ln g_{ij})^2, & \alpha = 0, \beta = 0 \end{cases} \quad (6.6.12)$$

以下是 AB-散度的几个特例。

(1) Alpha-散度 当  $\alpha + \beta = 1$  时, AB-散度退化为 Alpha-散度 (或称  $\alpha$  散度)

$$D_\alpha(\mathbf{P}\|\mathbf{G}) = D_{AB}^{(\alpha, 1-\alpha)}(\mathbf{P}\|\mathbf{G}) = \frac{1}{\alpha(\alpha-1)} \sum_{i=1}^I \sum_{j=1}^J [p_{ij}^\alpha g_{ij}^{1-\alpha} - \alpha p_{ij} + (\alpha-1)g_{ij}] \quad (6.6.13)$$

其中  $\alpha \neq 0$  和  $\alpha \neq 1$ 。

(2) Beta-散度 当  $\alpha = 1$  时, AB-散度给出 Beta-散度 (或称  $\beta$  散度)

$$D_\beta(\mathbf{P}\|\mathbf{G}) = D_{AB}^{(1, \beta)}(\mathbf{P}\|\mathbf{G}) = -\frac{1}{\beta} \sum_{i=1}^I \sum_{j=1}^J \left( p_{ij} g_{ij}^\beta - \frac{1}{1+\beta} p_{ij}^{1+\beta} - \frac{\beta}{1+\beta} g_{ij}^{1+\beta} \right) \quad (6.6.14)$$

其中  $\beta \neq 0$ 。特别地, 若  $\beta = 1$ , 则

$$D_{\beta=1}(\mathbf{P}\|\mathbf{G}) = D_{AB}^{(1,1)}(\mathbf{P}\|\mathbf{G}) = \frac{1}{2} \sum_{i=1}^I \sum_{j=1}^J (p_{ij} - g_{ij})^2 \quad (6.6.15)$$

退化为平方 Euclidean 距离。

(3) Kullback-Leibler (KL) 散度 当  $\alpha = 1$  和  $\beta = 0$  时, AB-散度退化为标准的 Kullback-Leibler 散度, 即  $D_{AB}^{(1,0)}(\mathbf{P}\|\mathbf{G}) = D_{\beta=0}(\mathbf{P}\|\mathbf{G}) = D_{KL}(\mathbf{P}\|\mathbf{G})$ 。

(4) Itakura-Saito (IS) 散度 当  $\alpha = 1$  和  $\beta = -1$  时, AB-散度给出标准的 Itakura-Saito 散度

$$D_{IS}(\mathbf{P}\|\mathbf{G}) = D_{AB}^{(1,-1)}(\mathbf{P}\|\mathbf{G}) = \sum_{i=1}^I \sum_{j=1}^J \left( \ln \frac{g_{ij}}{p_{ij}} + \frac{p_{ij}}{g_{ij}} - 1 \right) \quad (6.6.16)$$

以下是几种常用的  $\alpha$  散度 [111]:

- ① 当  $\alpha = 2$  时,  $\alpha$  散度退化为 Pearson  $\chi^2$  距离;
- ② 当  $\alpha = 0.5$  时,  $\alpha$  散度退化为 Hellinger 距离;
- ③ 当  $\alpha = -1$  时,  $\alpha$  散度退化为 Neyman  $\chi^2$  距离;
- ④ 当  $\alpha$  趋于零时,  $\alpha$  散度的极限是从  $\mathbf{G}$  到  $\mathbf{P}$  的 KL 散度, 即

$$\lim_{\alpha \rightarrow 0} D_\alpha(\mathbf{P}\|\mathbf{G}) = D_{KL}(\mathbf{G}\|\mathbf{P})$$

- ⑤ 当  $\alpha$  趋于 1 时,  $\alpha$  散度的极限是从  $\mathbf{P}$  到  $\mathbf{G}$  的 KL 散度, 即

$$\lim_{\alpha \rightarrow 1} D_\alpha(\mathbf{P}\|\mathbf{G}) = D_{KL}(\mathbf{P}\|\mathbf{G})$$

在数理统计和信息论中, 对于一组概率的离散集合  $\{p_i\}$  (其满足条件  $\sum_i p_i = 1$ ), Shannon 熵定义为

$$S = - \sum_i p_i \log_2(p_i) \quad (6.6.17)$$

Boltzmann-Gibbs 熵定义为

$$S_{BG} = -k \sum_i p_i \ln(p_i) \quad (6.6.18)$$

对于两个独立的系统  $A$  和  $B$ , 它们的联合概率密度  $p(A, B) = p(A)p(B)$ , 且 Shannon 熵和 Boltzmann-Gibbs 熵都具有可加性 (additivity)

$$S(A + B) = S(A) + S(B), \quad S_{BG}(A + B) = S_{BG}(A) + S_{BG}(B)$$

所以称为扩张熵 (extensive entropy)。

在物理学中, Tsallis 熵是标准 Boltzmann-Gibbs 熵的推广。Tsallis 熵是 Tsallis 于 1988 年提出的, 并称为  $q$  熵 [481]

$$S_q(p_i) = \frac{1 - \sum_i^q (p_i)^q}{q - 1} \quad (6.6.19)$$

Boltzmann-Gibbs 熵是  $q \rightarrow 1$  时 Tsallis 熵的极限，即有

$$S_{\text{BG}} = \lim_{q \rightarrow 1} D_q(p_i)$$

Tsallis 熵具有伪可加性 (pseudo additivity)

$$S_q(A + B) = S_q(A) + S_q(B) + (1 - q)S_q(A)S_q(B)$$

因此是一种非扩张熵 (non-extensive entropy) 或非加性熵 (nonadditive Entropy)<sup>[482]</sup>。

采用 Tsallis 熵的数理统计常称为 Tsallis 数理统计。Tsallis 数理统计的主要数学工具是  $q$  对数 (q logarithm) 和  $q$  指数 (q exponential)。特别地，重要的  $q$ -高斯分布由  $q$ -指数定义。

对于非负的实数  $q$  和  $x$ ，函数

$$\ln_q(x) = \begin{cases} \frac{x^{(1-q)} - 1}{1 - q}, & q \neq 1 \\ \ln(x), & q = 1 \end{cases} \quad (6.6.20)$$

称为  $x$  的 Tsallis 对数<sup>[481]</sup>，也称  $q$ -对数。对所有  $x \geq 0$ ，Tsallis 对数是解析的、递增的凹函数。 $q$ -对数的逆函数称为  $q$ -指数，定义为

$$\exp_q(x) = \begin{cases} [1 + (1 - q)x]^{\frac{1}{1-q}}, & 1 + (1 - q)x > 0 \\ 0, & q < 1 \\ +\infty, & q > 1 \\ \exp(x), & q = 1 \end{cases} \quad (6.6.21)$$

$q$ -指数与 Tsallis 对数的关系为

$$\exp_q(\ln_q(x)) = x \quad (6.6.22)$$

$$\ln_q(\exp_q(x)) = x \quad (6.6.23)$$

概率密度分布  $f(x)$  称为  $q$ -高斯分布，若

$$f(x) = \frac{\sqrt{\beta}}{C_q} \exp_q(-\beta x^2) \quad (6.6.24)$$

其中  $\exp_q(x) = [1 + (1 - q)x]^{\frac{1}{1-q}}$  为  $q$ -指数，归一化因子  $C_q$  由下式决定

$$C_q = \begin{cases} \frac{2\sqrt{\pi} \Gamma(\frac{1}{1-q})}{(3-q)\sqrt{1-q} \Gamma(\frac{1}{1-q})}, & -\infty < q < 1 \\ \sqrt{\pi}, & q = 1 \\ \frac{\sqrt{\pi} \Gamma(\frac{3-q}{2(q-1)})}{\sqrt{q-1} \Gamma(\frac{1}{q-1})}, & 1 < q < 3 \end{cases}$$

当  $q \rightarrow 1$  时， $q$ -高斯分布的极限即为高斯分布； $q$ -高斯分布已经应用于统计力学、地质学、解剖学、天文学、经济学、金融与机器学习中。

与高斯分布相比, 取  $1 < q < 3$  的  $q$ -高斯分布的一个突出特点是具有明显的拖尾。由于这一特点,  $q$ -对数即 Tsallis 对数和  $q$ -指数非常适合于用 AB-散度作为非负矩阵分解优化问题的代价函数。

为了方便 Tsallis 对数和  $q$ -指数在非负矩阵分解中的应用, 定义变形对数 (deformed logarithm)

$$\phi(x) = \ln_{1-\alpha}(x) = \begin{cases} \frac{x^\alpha - 1}{\alpha}, & \alpha \neq 0 \\ \ln(x), & \alpha = 0 \end{cases} \quad (6.6.25)$$

变形对数的逆变换

$$\phi^{-1}(x) = \exp_{1-\alpha}(x) = \begin{cases} \exp(x), & \alpha = 0 \\ (1 + \alpha x)^{1/\alpha}, & \alpha \neq 0; 1 + \alpha x \geq 0 \\ 0, & \alpha \neq 0; 1 + \alpha x < 0 \end{cases} \quad (6.6.26)$$

称为变形指数 (deformed exponential)。

变形对数和变形指数在非负矩阵分解的最优化算法中有重要的应用。

## 6.7 非负矩阵分解算法

非负矩阵分解是一个具有非负约束的最小化问题。根据 Berry 等人<sup>[43]</sup> 的分类, 非负矩阵分解有三种基本算法:

- (1) 乘法算法;
- (2) 梯度下降法;
- (3) 交替最小二乘算法。

这些算法都属于一阶优化算法。其中, 乘法算法本质上也是梯度下降法, 但它通过步长的聪明选择, 将一般梯度下降法的减法更新规则转变为乘法更新。后来, Cichocki, Zdunek 与 Amari<sup>[112]</sup> 在 Berry 等人分类的基础上, 又增加了拟牛顿法 (二阶优化算法) 和多层分解法两类方法。

下面依次介绍这五种代表性方法。

### 6.7.1 非负矩阵分解的乘法算法

梯度下降算法是一种被广泛使用的优化算法, 其基本思想是被优化的变元加上适当的校正项之后, 即给出被优化变元的更新。步长是这类算法的一个关键的选择参数, 它决定了校正量的大小。

考虑无约束最小化问题  $\min f(\mathbf{X})$  的梯度下降算法的一般更新规则

$$x_{ij} \leftarrow x_{ij} - \eta_{ij} \nabla f(x_{ij}), \quad i = 1, \dots, I; j = 1, \dots, J \quad (6.7.1)$$

其中  $x_{ij}$  是变元矩阵  $\mathbf{X}$  的元素,  $\nabla f(x_{ij})$  为代价函数  $f(\mathbf{X})$  的梯度矩阵  $\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}$  在点  $x_{ij}$  的值。如果适当地选择步长  $\eta_{ij}$ , 使得上述加法更新规则中的加法项  $x_{ij}$  能够被消去, 就可

将原来为加法运算的梯度下降算法变成乘法运算的梯度下降算法。这种乘法算法是 Lee 和 Seung 针对非负矩阵分解专门提出的<sup>[308]</sup>，但对其他很多优化问题同样适用，因此具有广泛的应用性。

### 1. 平方 Euclidean 距离最小化的乘法算法

考虑使用典型的平方 Euclidean 距离作代价函数的无约束优化问题  $\min D_E(\mathbf{X} \parallel \mathbf{AS}) = \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_2^2$ ，其梯度下降算法为

$$a_{ik} \leftarrow a_{ik} - \mu_{ik} \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} \quad (6.7.2)$$

$$s_{kj} \leftarrow s_{kj} - \eta_{kj} \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} \quad (6.7.3)$$

式中

$$\begin{aligned} \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} &= -[(\mathbf{X} - \mathbf{AS})\mathbf{S}^T]_{ik} \\ \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} &= -[\mathbf{A}^T(\mathbf{X} - \mathbf{AS})]_{kj} \end{aligned}$$

分别是代价函数关于变元矩阵的元素  $a_{ik}$  和  $s_{kj}$  的梯度。

若令

$$\mu_{ik} = \frac{a_{ik}}{[\mathbf{AS}^T]_{ik}}, \quad \eta_{kj} = \frac{s_{kj}}{[\mathbf{A}^T \mathbf{AS}]_{kj}} \quad (6.7.4)$$

则梯度下降算法变成乘法算法：

$$a_{ik} \leftarrow a_{ik} \frac{[\mathbf{XS}^T]_{ik}}{[\mathbf{AS}^T]_{ik}}, \quad i = 1, \dots, I; k = 1, \dots, K \quad (6.7.5)$$

$$s_{kj} \leftarrow s_{kj} \frac{[\mathbf{A}^T \mathbf{X}]_{kj}}{[\mathbf{A}^T \mathbf{AS}]_{kj}}, \quad k = 1, \dots, K; j = 1, \dots, J \quad (6.7.6)$$

关于乘法算法，有四点重要的注释。

**注释 1** 乘法算法的理论基础是期望最大化 (expectation-maximization, EM) 算法中的辅助函数。因此，乘法算法实际上是一种期望最大化的极大似然 (expectation maximization maximum likelihood, EMML) 算法<sup>[111]</sup>。

**注释 2** 上述元素形式的乘法算法很容易改写为矩阵形式的乘法算法

$$\mathbf{A} \leftarrow \mathbf{A} * [(\mathbf{XS}^T) \oslash (\mathbf{AS}^T)] \quad (6.7.7)$$

$$\mathbf{S} \leftarrow \mathbf{S} * [(\mathbf{A}^T \mathbf{X}) \oslash (\mathbf{A}^T \mathbf{AS})] \quad (6.7.8)$$

式中， $\mathbf{B} * \mathbf{C}$  表示两个矩阵的元素乘积 (components-wise product) 即 Hadamard 积，而  $\mathbf{B} \oslash \mathbf{C}$  代表两个矩阵的元素除法 (element-wise division)，即有

$$[\mathbf{B} * \mathbf{C}]_{ik} = b_{ik} c_{ik}, \quad [\mathbf{B} \oslash \mathbf{C}]_{ik} = b_{ik} / c_{ik} \quad (6.7.9)$$

**注释3** 梯度下降算法中，通常取固定的步长或者自适应的步长，与被更新的变元的下标无关。换言之，步长一般随更新的时间而变。在同一更新时间内，变元矩阵的不同元素实际上采用相同的步长进行更新。与之形成强烈对比的是，乘法法则针对变元矩阵的元素取不同的步长  $\mu_{ik}$ 。因此，这种步长既是时间自适应的，又是与变元元素自适应的。这正是乘法算法能够提高梯度下降算法性能的重要原因。

**注释4** 散度  $D(\mathbf{X} \parallel \mathbf{AS})$  在不同的乘法更新规则中是非递增的<sup>[308]</sup>，而这一非递增性有可能导致算法最终收敛不到平稳点<sup>[199]</sup>。

对于正则化非负矩阵分解的代价函数  $J(\mathbf{A}, \mathbf{S}) = \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_2^2 + \alpha_A J_A(\mathbf{A}) + \alpha_S J_S(\mathbf{S})$ ，由于梯度分别为

$$\begin{aligned}\nabla_{a_{ik}} J(\mathbf{A}, \mathbf{S}) &= \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} + \alpha_A \frac{\partial J_A(\mathbf{A})}{\partial a_{ik}} \\ \nabla_{s_{kj}} J(\mathbf{A}, \mathbf{S}) &= \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} + \alpha_S \frac{\partial J_S(\mathbf{S})}{\partial s_{kj}}\end{aligned}$$

所以乘法算法是式(6.7.6)的适当修正<sup>[308]</sup>

$$a_{ik} \leftarrow a_{ik} \frac{[ \mathbf{X} \mathbf{S}^T ]_{ik} - \alpha_A \nabla J_A(a_{ik}) ]_+}{[ \mathbf{A} \mathbf{S} \mathbf{S}^T ]_{ik} + \epsilon} \quad (6.7.10)$$

$$s_{kj} \leftarrow s_{kj} \frac{[ \mathbf{A}^T \mathbf{X} ]_{kj} - \alpha_S \nabla J_S(s_{kj}) ]_+}{[ \mathbf{A}^T \mathbf{A} \mathbf{S} ]_{kj} + \epsilon} \quad (6.7.11)$$

式中， $\nabla J_A(a_{ik}) = \frac{\partial J_A(\mathbf{A})}{\partial a_{ik}}$ ， $\nabla J_S(s_{kj}) = \frac{\partial J_S(\mathbf{S})}{\partial s_{kj}}$ ，并且  $[u]_+ = \max\{u, \epsilon\}$ 。参数  $\epsilon$  是一个很小的正数，主要为了防止出现近似等于零的分母，以保证算法的收敛性能和数值稳定性。

上述元素形式的乘法算法也可以改写为矩阵形式

$$\mathbf{A} \leftarrow \mathbf{A} * [ (\mathbf{X} \mathbf{S}^T - \alpha_A \Psi_A) \oslash (\mathbf{A} \mathbf{S} \mathbf{S}^T + \epsilon \mathbf{I}) ] \quad (6.7.12)$$

$$\mathbf{S} \leftarrow \mathbf{S} * [ (\mathbf{A}^T \mathbf{X} - \alpha_S \Psi_S) \oslash (\mathbf{A}^T \mathbf{A} \mathbf{S} + \epsilon \mathbf{I}) ] \quad (6.7.13)$$

式中  $\Psi_A = \frac{\partial J_A(\mathbf{A})}{\partial \mathbf{A}}$  和  $\Psi_S = \frac{\partial J_S(\mathbf{S})}{\partial \mathbf{S}}$  是两个梯度矩阵，而  $\mathbf{I}$  为单位矩阵。

## 2. KL 散度最小化的乘法算法

考虑 KL 散度

$$D_{\text{KL}}(\mathbf{X} \parallel \mathbf{AS}) = \sum_{i_1=1}^I \sum_{j_1=1}^J \left( x_{i_1 j_1} \ln \frac{x_{i_1 j_1}}{[\mathbf{AS}]_{i_1 j_1}} - x_{i_1 j_1} + [\mathbf{AS}]_{i_1 j_1} \right) \quad (6.7.14)$$

的最小化问题。由于

$$\frac{\partial D_{\text{KL}}(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} = - \sum_{j=1}^J (s_{kj} x_{ij} / [\mathbf{AS}]_{ij} + s_{kj})$$

$$\frac{\partial D_{\text{KL}}(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} = - \sum_{i=1}^I (a_{ik} x_{ij} / [\mathbf{AS}]_{ij} + a_{ik})$$

所以梯度下降算法为

$$\begin{aligned} a_{ik} &\leftarrow a_{ik} - \mu_{ik} \times \left[ -\sum_{j=1}^J (s_{kj} x_{ij} / [\mathbf{AS}]_{ij} + s_{kj}) \right] \\ s_{kj} &\leftarrow s_{kj} - \eta_{kj} \times \left[ -\sum_{i=1}^I (a_{ik} x_{ij} / [\mathbf{AS}]_{ij} + a_{ik}) \right] \end{aligned}$$

若令

$$\mu_{ik} = \frac{1}{\sum_{j=1}^J s_{kj}}, \quad \eta_{kj} = \frac{1}{\sum_{i=1}^I a_{ik}}$$

则梯度下降算法可以改写为乘法算法 [308]

$$a_{ik} \leftarrow a_{ik} \frac{\sum_{j=1}^J s_{kj} x_{ik} / [\mathbf{AS}]_{ik}}{\sum_{j=1}^J s_{kj}} \quad (6.7.15)$$

$$s_{kj} \leftarrow s_{kj} \frac{\sum_{i=1}^I a_{ik} x_{ik} / [\mathbf{AS}]_{ik}}{\sum_{i=1}^I a_{ik}} \quad (6.7.16)$$

其矩阵形式为

$$\mathbf{A} \leftarrow [\mathbf{A} \oslash [\mathbf{1}_I \otimes (\mathbf{S}\mathbf{1}_K)^T]] * [[\mathbf{X} \oslash (\mathbf{AS})] \mathbf{S}^T] \quad (6.7.17)$$

$$\mathbf{S} \leftarrow [\mathbf{S} \oslash [(\mathbf{A}^T \mathbf{1}_I) \otimes \mathbf{1}_K]] * [\mathbf{A}^T [\mathbf{X} \oslash (\mathbf{AS})]] \quad (6.7.18)$$

式中  $\mathbf{1}_I$  是全部元素为 1 的  $I$  维列向量。

### 3. AB 散度最小化的乘法算法

对于 AB 散度 (其中  $\alpha, \beta, \alpha + \beta \neq 0$ )

$$D_{AB}^{(\alpha, \beta)}(\mathbf{X} \parallel \mathbf{AS}) = -\frac{1}{\alpha \beta} \sum_{i=1}^I \sum_{j=1}^J \left( x_{ij}^\alpha [\mathbf{AS}]_{ij}^\beta - \frac{\alpha}{\alpha + \beta} x_{ij}^{\alpha+\beta} - \frac{\beta}{\alpha + \beta} [\mathbf{AS}]_{ij}^{\alpha+\beta} \right) \quad (6.7.19)$$

其梯度

$$\frac{\partial D_{AB}^{(\alpha, \beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} = -\frac{1}{\alpha} \sum_{j=1}^J \left( s_{kj} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{1-\beta} - [\mathbf{AS}]_{ij}^{\alpha+\beta-1} s_{kj} \right) \quad (6.7.20)$$

$$\frac{\partial D_{AB}^{(\alpha, \beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} = -\frac{1}{\beta} \sum_{i=1}^I \left( a_{ik} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{1-\beta} - [\mathbf{AS}]_{ij}^{\alpha+\beta-1} a_{ik} \right) \quad (6.7.21)$$

因此, 若令步长

$$\mu_{ik} = \frac{\alpha a_{ik}}{\sum_{j=1}^J s_{kj} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}}, \quad \eta_{kj} = \frac{\alpha s_{kj}}{\sum_{i=1}^I a_{ik} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}}$$

则得到 AB 散度最小化的乘法算法

$$a_{ik} \leftarrow a_{ik} \left( \frac{\sum_{j=1}^J s_{kj} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{\beta-1}}{\sum_{j=1}^J s_{kj} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}} \right), \quad s_{kj} \leftarrow s_{kj} \left( \frac{\sum_{i=1}^I a_{ik} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{\beta-1}}{\sum_{i=1}^I a_{ik} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}} \right)$$

为了加快期望最大化的极大似然算法的收敛，文献 [110] 提出使用更新规则

$$a_{ik} \leftarrow a_{ik} \left( \frac{\sum_{j=1}^J s_{kj} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{\beta-1}}{\sum_{j=1}^J s_{kj} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}} \right)^{1/\alpha} \quad (6.7.22)$$

$$s_{kj} \leftarrow s_{kj} \left( \frac{\sum_{i=1}^I a_{ik} x_{ij}^\alpha [\mathbf{AS}]_{ij}^{\beta-1}}{\sum_{i=1}^I a_{ik} [\mathbf{AS}]_{ij}^{\alpha+\beta-1}} \right)^{1/\alpha} \quad (6.7.23)$$

其中  $1/\alpha$  为正的松弛参数，有助于改善算法的收敛。

当  $\beta = 1 - \alpha$  时，由 AB 散度最小化的乘法算法，立即得到代价函数为 Alpha 散度时的乘法算法

$$a_{ik} \leftarrow a_{ik} \left( \frac{\sum_{j=1}^J (x_{ij}/[\mathbf{AS}]_{ij})^\alpha s_{kj}}{\sum_{j=1}^J s_{kj}} \right)^{1/\alpha} \quad (6.7.24)$$

$$s_{kj} \leftarrow s_{kj} \left( \frac{\sum_{i=1}^I a_{ik} (x_{ij}/[\mathbf{AS}]_{ij})^\alpha}{\sum_{i=1}^I a_{ik}} \right)^{1/\alpha} \quad (6.7.25)$$

对于包含  $\alpha = 0$  与/或  $\beta = 0$  在内的一般情况，AB 散度的梯度为<sup>[110]</sup>

$$\frac{\partial D_{AB}^{(\alpha,\beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial a_{ik}} = - \sum_{j=1}^J [\mathbf{AS}]_{ij}^{\lambda-1} s_{kj} \ln_{1-\alpha} \left( \frac{x_{ij}}{[\mathbf{AS}]_{ij}} \right) \quad (6.7.26)$$

$$\frac{\partial D_{AB}^{(\alpha,\beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial s_{kj}} = - \sum_{i=1}^I [\mathbf{AS}]_{ij}^{\lambda-1} a_{ik} \ln_{1-\alpha} \left( \frac{x_{ij}}{[\mathbf{AS}]_{ij}} \right) \quad (6.7.27)$$

其中  $\lambda = \alpha + \beta$ 。于是，利用变形对数与变形指数的关系  $\exp_{1-\alpha}(\ln_{1-\alpha}(x)) = x$  易知，AB 散度函数最小化的梯度算法为

$$a_{ik} \leftarrow \exp_{1-\alpha} \left( \ln_{1-\alpha}(a_{ik}) - \mu_{ik} \frac{\partial D_{AB}^{(\alpha,\beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial \ln_{1-\alpha}(a_{ik})} \right) \quad (6.7.28)$$

$$s_{kj} \leftarrow \exp_{1-\alpha} \left( \ln_{1-\alpha}(s_{kj}) - \eta_{kj} \frac{\partial D_{AB}^{(\alpha,\beta)}(\mathbf{X} \parallel \mathbf{AS})}{\partial \ln_{1-\alpha}(s_{kj})} \right) \quad (6.7.29)$$

若选择

$$\mu_{ik} = \frac{a_{ik}^{2\alpha-1}}{\sum_{j=1}^J s_{kj} [\mathbf{AS}]_{ij}^{\lambda-1}}, \quad \eta_{kj} = \frac{s_{kj}^{2\alpha-1}}{\sum_{i=1}^I a_{ik} [\mathbf{AS}]_{ij}^{\lambda-1}}$$

则可得到 Cichocki, Cruces 与 Amari 提出的 AB 乘法的非负矩阵分解算法 [110]

$$a_{ik} \leftarrow a_{ik} \exp_{1-\alpha} \left( \sum_{j=1}^J \frac{s_{kj} [\mathbf{AS}]_{ij}^{\lambda-1}}{\sum_{j=1}^J s_{kj} [\mathbf{AS}]_{ij}^{\lambda-1}} \ln_{1-\alpha} \left( \frac{x_{ij}}{[\mathbf{AS}]_{ij}} \right) \right) \quad (6.7.30)$$

$$s_{kj} \leftarrow s_{kj} \exp_{1-\alpha} \left( \underbrace{\sum_{i=1}^I \frac{a_{ik} [\mathbf{AS}]_{ij}^{\lambda-1}}{\sum_{i=1}^I a_{ik} [\mathbf{AS}]_{ij}^{\lambda-1}} \ln_{1-\alpha} \left( \frac{x_{ij}}{[\mathbf{AS}]_{ij}} \right)}_{\text{权系数}} \underbrace{}_{\alpha-\text{变焦}} \right) \quad (6.7.31)$$

由于参数  $\alpha$  的变焦作用, 基矩阵  $\mathbf{A} = [a_{ik}]$  和系数矩阵  $\mathbf{S} = [s_{kj}]$  的元素更新的相对误差主要由变形对数的比值  $x_{ij}/[\mathbf{AS}]_{ij}$  控制: 当  $\alpha > 1$  时, 变形对数  $\ln_{1-\alpha}(x_{ij}/[\mathbf{AS}]_{ij})$  具有缩小功能, 强调的是较大比值  $x_{ij}/[\mathbf{AS}]_{ij}$  的作用, 因为较小比值被缩小后, 更加被忽视; 而当  $\alpha < 1$  时, 变形对数的放大功能则相对突出了较小比值  $x_{ij}/[\mathbf{AS}]_{ij}$  的影响。

由于 AB 散度包含了许多散度为特例, 所以 AB 乘法非负矩阵分解算法融合了多种非负矩阵分解算法。

在以上各种乘法更新公式中, 通常都需要在每一个分母项加一很小的扰动 (例如  $\epsilon = 10^{-9}$ ), 以防止被零除。

## 6.7.2 投影梯度法和 Nesterov 最优梯度法

非负矩阵分解的梯度下降法由两个梯度下降算法组成

$$\mathbf{A}_{k+1} = \mathbf{A}_k - \mu_A \frac{\partial f(\mathbf{A}_k, \mathbf{S}_k)}{\partial \mathbf{A}_k}$$

$$\mathbf{S}_{k+1} = \mathbf{S}_k - \mu_S \frac{\partial f(\mathbf{A}_k, \mathbf{S}_k)}{\partial \mathbf{S}_k}$$

为了保证矩阵  $\mathbf{A}_k$  和  $\mathbf{S}_k$  的非负性, 在每一步更新中都需要将得到的更新矩阵  $\mathbf{A}_{k+1}$  和  $\mathbf{S}_{k+1}$  的所有元素向非负象限进行投影, 这就构成了非负矩阵分解的投影梯度算法 [317]

$$\mathbf{A}_{k+1} = \left[ \mathbf{A}_k - \mu_A \frac{\partial f(\mathbf{A}_k, \mathbf{S}_k)}{\partial \mathbf{A}_k} \right]_+ \quad (6.7.32)$$

$$\mathbf{S}_{k+1} = \left[ \mathbf{S}_k - \mu_S \frac{\partial f(\mathbf{A}_k, \mathbf{S}_k)}{\partial \mathbf{S}_k} \right]_+ \quad (6.7.33)$$

由于步长  $\mu_A$  和  $\mu_S$  没有像乘法算法那样经过精心选择, 加之非负性投影, 所以投影梯度算法的收敛分析比较困难。

Nesterov 最优梯度法为改善梯度下降法的收敛性能提供了一个有效的途径。

考虑低秩非负矩阵  $\mathbf{X} \in \mathbb{R}^{m \times n}$  的分解

$$\min \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_F^2 \quad \text{subject to } \mathbf{A} \in \mathbb{R}_+^{m \times r}, \mathbf{S} \in \mathbb{R}_+^{r \times n} \quad (6.7.34)$$

其中  $r = \text{rank}(\mathbf{X}) < \min\{m, n\}$ 。

由于式 (6.7.34) 是一个非凸最小化问题, 故可以使用块协同下降法即交替非负最小二乘表示式 (6.7.63) 与式 (6.7.64) 求解最小化问题式 (6.7.34) 的局部解

$$\mathbf{S}_{t+1} = \arg \min_{\mathbf{S} \geq 0} F(\mathbf{A}_t, \mathbf{S}) = \frac{1}{2} \|\mathbf{X} - \mathbf{A}_t \mathbf{S}\|_F^2 \quad (6.7.35)$$

$$\mathbf{A}_{t+1}^T = \arg \min_{\mathbf{A} \geq 0} F(\mathbf{S}_{t+1}^T, \mathbf{A}^T) = \frac{1}{2} \|\mathbf{X}^T - \mathbf{S}_{t+1}^T \mathbf{A}^T\|_F^2 \quad (6.7.36)$$

式中  $t$  表示第  $t$  个数据块。

最近, Guan 等人<sup>[210]</sup>证明了非负矩阵分解的目标函数满足 Nesterov 最优梯度法的条件:

- (1) 目标函数  $F(\mathbf{A}_t, \mathbf{S}) = \frac{1}{2} \|\mathbf{X} - \mathbf{A}_t \mathbf{S}\|_F^2$  是凸函数。
- (2) 目标函数  $F(\mathbf{A}_t, \mathbf{S})$  的梯度是 Lipschitz 连续的, 且 Lipschitz 常数为  $L = \|\mathbf{A}_t^T \mathbf{A}_t\|_F$ 。据此, 文献 [210] 提出了低秩非负矩阵分解的 Nesterov 最优梯度算法 (NeNMF)。

#### 算法 6.7.1 Nesterov 非负矩阵分解 (NeNMF) 算法

输入 数据矩阵  $\mathbf{X} \in \mathbb{R}_+^{m \times n}, 1 \leq r \leq \min\{m, n\}$ 。

输出 基矩阵  $\mathbf{A} \in \mathbb{R}_+^{m \times r}$  和系数矩阵  $\mathbf{S} \in \mathbb{R}_+^{r \times n}$ 。

初始化  $\mathbf{A}_1 \geq 0, \mathbf{S}_1 \geq 0, t = 1$ 。

步骤 1 更新  $\mathbf{A}_{t+1}$  和  $\mathbf{S}_{t+1}$

$$\mathbf{S}_{t+1} = \text{OGM}(\mathbf{A}_t, \mathbf{S}), \quad \mathbf{A}_{t+1} = \text{OGM}(\mathbf{S}_{t+1}^T, \mathbf{A}^T)$$

步骤 2 检验迭代算法的停止准则

$$\nabla_S^P F(\mathbf{A}_t, \mathbf{S}_t) = 0, \quad \nabla_A^P F(\mathbf{A}_t, \mathbf{S}_t) = 0$$

其中

$$\begin{aligned} \nabla_S^P F(\mathbf{A}_t, \mathbf{S}_t)_{ij} &= \begin{cases} \nabla_S F(\mathbf{A}_t, \mathbf{S}_t)_{ij}, & (\mathbf{S}_t)_{ij} > 0 \\ \min \{0, \nabla_S F(\mathbf{A}_t, \mathbf{S}_t)_{ij}\}, & (\mathbf{S}_t)_{ij} = 0 \end{cases} \\ \nabla_A^P F(\mathbf{A}_t, \mathbf{S}_t)_{ij} &= \begin{cases} \nabla_A F(\mathbf{A}_t, \mathbf{S}_t)_{ij}, & (\mathbf{A}_t)_{ij} > 0 \\ \min \{0, \nabla_A F(\mathbf{A}_t, \mathbf{S}_t)_{ij}\}, & (\mathbf{A}_t)_{ij} = 0 \end{cases} \end{aligned}$$

分别是目标函数  $F(\mathbf{A}_t, \mathbf{S}_t)$  关于  $\mathbf{S}$  和  $\mathbf{A}$  的投影梯度。

步骤 3 若停止准则满足, 则停止迭代, 并输出矩阵  $\mathbf{A}_t$  和  $\mathbf{S}_t$ ; 否则, 令  $t \leftarrow t + 1$ , 并返回步骤 1, 继续迭代, 直至停止准则满足。

算法 6.7.1 中的函数  $\text{OGM}(\mathbf{A}_t, \mathbf{S})$  和  $\text{OGM}(\mathbf{S}_{t+1}^T, \mathbf{A}^T)$  分别是求解交替非负最小二乘问题式 (6.7.58) 和式 (6.7.59) 的 Nesterov 最优梯度法(OGM)。

#### 算法 6.7.2 最优梯度法 OGM( $\mathbf{A}_t, \mathbf{S}$ )<sup>[210]</sup>

输入  $\mathbf{A}_t, \mathbf{S}_t$ 。

输出  $\mathbf{S}_{t+1}$ 。

初始化  $\mathbf{Y}_0 = \mathbf{S}_t \geq 0, \alpha_0 = 1, L = \|\mathbf{A}_t^T \mathbf{A}_t\|_F, k = 0$ 。

步骤 1 更新

$$\begin{aligned}\mathbf{S}_k &= \mathcal{P}_+ \left( \mathbf{Y}_k - \frac{1}{L} \nabla_{\mathbf{S}} F(\mathbf{A}_t, \mathbf{Y}_k) \right) \\ \alpha_{k+1} &= \frac{1 + \sqrt{4\alpha_k^2 + 1}}{2} \\ \mathbf{Y}_{k+1} &= \mathbf{S}_k + \frac{\alpha_k - 1}{\alpha_{k+1}} (\mathbf{S}_k - \mathbf{S}_{k-1})\end{aligned}$$

步骤 2 检验收敛准则是否满足

$$\nabla_{\mathbf{S}}^P F(\mathbf{A}_t, \mathbf{S}_k) = 0$$

若满足，则停止迭代，进入步骤 3；否则，返回步骤 1，继续迭代，直至收敛准则满足。

步骤 3 输出  $\mathbf{S}_{t+1} = \mathbf{S}_k$ 。

注释 将算法 6.7.2 中的矩阵作如下置换： $\mathbf{A}_t \rightarrow \mathbf{S}_t^T$  和  $\mathbf{S}_t \rightarrow \mathbf{A}_t^T$ ，并且 Lipschitz 常数更换为  $L = \|\mathbf{S}_t \mathbf{S}_t^T\|_F$ ，即得到最优梯度算法 OGM( $\mathbf{S}_{t+1}^T, \mathbf{A}^T$ )，其输出为  $\mathbf{A}_{t+1}^T = \mathbf{A}_k^T$ 。

文献 [210] 还分别介绍了 NeNMF 算法对  $L_1$  正则化、 $L_2$  正则化以及流形正则化非负矩阵分解的应用。

由于在优化中引入了结构信息，所以 NeNMF 算法在固定一个矩阵而优化另外一个矩阵时，能够以  $O\left(\frac{1}{k^2}\right)$  的速率收敛 [210]。

### 6.7.3 交替非负最小二乘算法

交替最小二乘方法最早由 Paatero 与 Tapper<sup>[386]</sup> 用于非负矩阵分解。由于这种方法约束矩阵是非负的，所以现在习惯称为交替非负最小二乘 (alternating nonnegative least squares, ANLS) 算法。

非负矩阵分解  $\mathbf{X}_{I \times J} = \mathbf{A}_{I \times K} \mathbf{S}_{K \times J}$  的优化问题

$$\min_{\mathbf{A}, \mathbf{S}} \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_F^2 \quad \text{subject to } \mathbf{A}, \mathbf{S} \geq 0 \quad (6.7.37)$$

可以分解为两个交替非负最小二乘子问题<sup>[386]</sup>

$$\text{ANLS1} \quad \min_{\mathbf{S} \geq 0} f_1(\mathbf{S}) = \frac{1}{2} \|\mathbf{AS} - \mathbf{X}\|_F^2 \quad (\mathbf{A} \text{ 固定}) \quad (6.7.38)$$

$$\text{ANLS2} \quad \min_{\mathbf{A} \geq 0} f_2(\mathbf{A}^T) = \frac{1}{2} \|\mathbf{S}^T \mathbf{A}^T - \mathbf{X}^T\|_F^2 \quad (\mathbf{S} \text{ 固定}) \quad (6.7.39)$$

这两个交替非负最小二乘子问题相当于使用最小二乘方法交替求解矩阵方程  $\mathbf{AS} = \mathbf{X}$  和  $\mathbf{S}^T \mathbf{A}^T = \mathbf{X}^T$ ，其最小二乘解分别为

$$\mathbf{S} = \mathcal{P}_+ \left( (\mathbf{A}^T \mathbf{A})^\dagger \mathbf{A}^T \mathbf{X} \right) \quad (6.7.40)$$

$$\mathbf{A}^T = \mathcal{P}_+ \left( (\mathbf{S} \mathbf{S}^T)^\dagger \mathbf{S} \mathbf{X}^T \right) \quad (6.7.41)$$

当  $\mathbf{A}$  与/或  $\mathbf{S}$  在迭代过程中奇异时，算法将无法收敛。

为了克服交替最小二乘算法的数值稳定性差的缺点，Langville 等人<sup>[296]</sup> 和 Pauca 等人<sup>[397]</sup> 于 2006 年独立地提出了约束非负矩阵分解 (constrained nonnegative matrix factorization, CNMF)

$$\text{CNMF} \quad \min_{\mathbf{A}, \mathbf{S}} \frac{1}{2} (\|\mathbf{X} - \mathbf{AS}\|_F^2 + \alpha\|\mathbf{A}\|_F^2 + \beta\|\mathbf{S}\|_F^2) \quad \text{subject to } \mathbf{A}, \mathbf{S} \geq 0 \quad (6.7.42)$$

式中， $\alpha \geq 0$  和  $\beta \geq 0$  是两个正则化参数，分别起到压制  $\|\mathbf{A}\|_F^2$  和  $\|\mathbf{S}\|_F^2$  的作用。

上述约束优化实际上是 Tikhonov 于 1963 年提出的正则化最小二乘问题<sup>[472]</sup> 的一个典型应用。

正则化非负矩阵分解问题可以分解为两个交替正则化非负最小二乘 (alternating regularization nonnegative least squares, ARNLS) 问题

$$\text{ARNLS1} \quad \min_{\mathbf{S} \in \mathbb{R}_+^{J \times K}} J_1(\mathbf{S}) = \frac{1}{2} \|\mathbf{AS} - \mathbf{X}\|_F^2 + \frac{1}{2} \beta \|\mathbf{S}\|_F^2 \quad (\mathbf{A} \text{ 固定}) \quad (6.7.43)$$

$$\text{ARNLS2} \quad \min_{\mathbf{A} \in \mathbb{R}_+^{I \times J}} J_2(\mathbf{A}^T) = \frac{1}{2} \|\mathbf{S}^T \mathbf{A}^T - \mathbf{X}^T\|_F^2 + \frac{1}{2} \alpha \|\mathbf{A}\|_F^2 \quad (\mathbf{S} \text{ 固定}) \quad (6.7.44)$$

或者等价写作

$$\text{ARNLS1} \quad \min_{\mathbf{S} \in \mathbb{R}_+^{J \times K}} J_1(\mathbf{S}) = \frac{1}{2} \left\| \begin{bmatrix} \mathbf{A} \\ \sqrt{\beta} \mathbf{I}_J \end{bmatrix} \mathbf{S} - \begin{bmatrix} \mathbf{X} \\ \mathbf{O}_{J \times K} \end{bmatrix} \right\|_F^2 \quad (6.7.45)$$

$$\text{ARNLS2} \quad \min_{\mathbf{A} \in \mathbb{R}_+^{I \times J}} J_2(\mathbf{A}^T) = \frac{1}{2} \left\| \begin{bmatrix} \mathbf{S}^T \\ \sqrt{\alpha} \mathbf{I}_J \end{bmatrix} \mathbf{A}^T - \begin{bmatrix} \mathbf{X}^T \\ \mathbf{O}_{J \times I} \end{bmatrix} \right\|_F^2 \quad (6.7.46)$$

由矩阵微分

$$\begin{aligned} dJ_1(\mathbf{S}) &= \frac{1}{2} d \left( \text{tr}[(\mathbf{AS} - \mathbf{X})^T (\mathbf{AS} - \mathbf{X})] + \beta \text{tr}(\mathbf{S}^T \mathbf{S}) \right) \\ &= \text{tr} \left( (\mathbf{S}^T \mathbf{A}^T \mathbf{A} - \mathbf{X}^T \mathbf{A} + \beta \mathbf{S}^T) d\mathbf{S} \right) \\ dJ_2(\mathbf{A}^T) &= \frac{1}{2} d \left( \text{tr}[(\mathbf{AS} - \mathbf{X})(\mathbf{AS} - \mathbf{X})^T] + \alpha \text{tr}(\mathbf{A}^T \mathbf{A}) \right) \\ &= \text{tr} \left( (\mathbf{A} \mathbf{S} \mathbf{S}^T - \mathbf{X} \mathbf{S}^T + \alpha \mathbf{A}) d\mathbf{A}^T \right) \end{aligned}$$

由此得梯度矩阵

$$\frac{\partial J_1(\mathbf{S})}{\partial \mathbf{S}} = -\mathbf{A}^T \mathbf{X} + \mathbf{A}^T \mathbf{AS} + \beta \mathbf{S} \quad (6.7.47)$$

$$\frac{\partial J_2(\mathbf{A}^T)}{\partial \mathbf{A}} = -\mathbf{S} \mathbf{X}^T + \mathbf{S} \mathbf{S}^T \mathbf{A}^T + \alpha \mathbf{A}^T \quad (6.7.48)$$

由  $\frac{\partial J_1(\mathbf{S})}{\partial \mathbf{S}} = 0$  和  $\frac{\partial J_2(\mathbf{A}^T)}{\partial \mathbf{A}^T} = 0$  分别得到两个正则化最小二乘子问题的解为

$$(\mathbf{A}^T \mathbf{A} + \beta \mathbf{I}_J) \mathbf{S} = \mathbf{A}^T \mathbf{X} \quad \text{或} \quad \mathbf{S} = (\mathbf{A}^T \mathbf{A} + \beta \mathbf{I}_J)^{-1} \mathbf{A}^T \mathbf{X} \quad (6.7.49)$$

$$(\mathbf{S} \mathbf{S}^T + \alpha \mathbf{I}_J) \mathbf{A}^T = \mathbf{S} \mathbf{X}^T \quad \text{或} \quad \mathbf{A}^T = (\mathbf{S} \mathbf{S}^T + \alpha \mathbf{I}_J)^{-1} \mathbf{S} \mathbf{X}^T \quad (6.7.50)$$

求解上述问题的最小二乘方法称为交替约束最小二乘方法，其基本框架如下<sup>[296]</sup>：

(1) 用非负元素初始化  $\mathbf{A} \in \mathbb{R}^{I \times K}$ 。

(2) 迭代求正则化最小二乘解式 (6.7.49) 和式 (6.7.50)，并且强制矩阵  $\mathbf{S}$  和  $\mathbf{A}$  非负化

$$s_{kj} = S_{kj} = \max\{0, s_{kj}\} \quad \text{和} \quad a_{ik} = A_{ik} = \max\{0, a_{ik}\} \quad (6.7.51)$$

(3) 将  $\mathbf{A}$  的各列和  $\mathbf{S}$  的各行分别归一化为单位 Frobenius 范数。然后，返回 (2)，并重复迭代，直至某个收敛准则满足。

更好的方法是使用乘法算法求解两个交替最小二乘问题。由梯度表达式 (6.7.47) 和式 (6.7.48) 立即得交替梯度算法

$$\begin{aligned} S_{kj} &\leftarrow S_{kj} + \eta_{kj} \left[ \mathbf{A}^T \mathbf{X} - \mathbf{A} \mathbf{A}^T \mathbf{S} - \beta \mathbf{S} \right]_{kj} \\ \mathbf{A}_{ik}^T &\leftarrow \mathbf{A}_{ik}^T + \mu_{ik} \left[ \mathbf{X} \mathbf{S}^T - \mathbf{A} \mathbf{S} \mathbf{S}^T - \alpha \mathbf{A} \right]_{ik} \end{aligned}$$

若选择

$$\eta_{kj} = \frac{S_{kj}}{[\mathbf{A} \mathbf{A}^T \mathbf{S} + \beta \mathbf{S}]_{kj}}, \quad \mu_{ik} = \frac{\mathbf{A}_{ik}^T}{[\mathbf{A} \mathbf{S} \mathbf{S}^T + \alpha \mathbf{A}]_{ik}}$$

则梯度算法变为乘法算法

$$S_{kj} \leftarrow S_{kj} \frac{[\mathbf{A}^T \mathbf{X}]_{ik}}{[\mathbf{A} \mathbf{A}^T \mathbf{S} + \beta \mathbf{S}]_{ik}} \quad (6.7.52)$$

$$\mathbf{A}_{ik}^T \leftarrow \mathbf{A}_{ik}^T \frac{[\mathbf{X} \mathbf{S}^T]_{kj}}{[\mathbf{A} \mathbf{S} \mathbf{S}^T + \alpha \mathbf{A}]_{kj}} \quad (6.7.53)$$

只要矩阵  $\mathbf{A}$  和  $\mathbf{S}$  使用非负数值初始化，上述迭代即可保证这两个矩阵的非负性。

文献 [397] 通过选择步长

$$\eta_{kj} = \frac{S_{kj}}{[\mathbf{A} \mathbf{A}^T \mathbf{S}]_{kj}}, \quad \mu_{ik} = \frac{\mathbf{A}_{ik}^T}{[\mathbf{A} \mathbf{S} \mathbf{S}^T]_{ik}}$$

得到乘法算法

$$S_{kj} \leftarrow S_{kj} \frac{[\mathbf{X} \mathbf{S}^T - \beta \mathbf{S}]_{kj}}{[\mathbf{A} \mathbf{A}^T \mathbf{S}]_{kj} + \epsilon} \quad (6.7.54)$$

$$\mathbf{A}_{ik}^T \leftarrow \mathbf{A}_{ik}^T \frac{[\mathbf{A}^T \mathbf{X} - \alpha \mathbf{A}]_{ik}}{[\mathbf{A} \mathbf{S} \mathbf{S}^T]_{ik} + \epsilon} \quad (6.7.55)$$

注意，上述算法由于分子存在减法运算，故一般不能保证矩阵元素的非负性。

#### 6.7.4 拟牛顿法与多层分解法

##### 1. 拟牛顿法 [536]

假定需要求解超定矩阵方程  $\mathbf{S}^T \mathbf{A}^T = \mathbf{X}^T$  (其中  $J \gg K$ ) 中的未知矩阵  $\mathbf{A}^T$ ，有效方法之一是拟牛顿法。

代价函数  $D_E(\mathbf{X} \parallel \mathbf{AS}) = \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_2^2$  的 Hessian 矩阵  $\mathbf{H}_A = \nabla_A^2(D_E) = \mathbf{I}_{I \times I} \otimes \mathbf{SS}^T \in \mathbb{R}^{IK \times IK}$  是一个分块对角矩阵，对角线上的块矩阵为  $\mathbf{SS}^T$ 。于是，拟牛顿法取

$$\mathbf{A} \leftarrow [\mathbf{A} - \nabla_A(D_E(\mathbf{X} \parallel \mathbf{AS})) \mathbf{H}_A^{-1}]$$

其中  $\nabla_A(D_E(\mathbf{X} \parallel \mathbf{AS})) = (\mathbf{AS} - \mathbf{X})\mathbf{S}^T$  是代价函数  $D_E(\mathbf{X} \parallel \mathbf{AS})$  的梯度矩阵。因此，拟牛顿算法具体为

$$\mathbf{A} \leftarrow \left[ \mathbf{A} - (\mathbf{AS} - \mathbf{X})\mathbf{S}^T \left( \mathbf{SS}^T \right)^{-1} \right]$$

为了防止矩阵  $\mathbf{SS}^T$  奇异或者条件数很大，可以采用松弛法

$$\mathbf{A} \leftarrow \left[ \mathbf{A} - (\mathbf{AS} - \mathbf{X})\mathbf{S}^T \left( \mathbf{SS}^T + \lambda \mathbf{I}_{K \times K} \right)^{-1} \right]$$

## 2. 多层分解法 [113~115]

多层非负矩阵分解的基本思想是：认为第 1 次分解的结果  $\mathbf{X} \approx \mathbf{A}^{(1)}\mathbf{S}^{(1)}$  存在较大的误差，因而将  $\mathbf{S}^{(1)}$  视为新的数据矩阵，再作第 2 层的非负矩阵分解  $\mathbf{S}^{(1)} \approx \mathbf{A}^{(2)}\mathbf{S}^{(2)}$ 。第 2 层分解的结果仍然存在误差，又作第 3 层非负矩阵分解  $\mathbf{S}^{(2)} \approx \mathbf{A}^{(3)}\mathbf{S}^{(3)}$ ，如此继续，形成一个  $L$  层的非负矩阵分解

$$\begin{aligned} \mathbf{X} &\approx \mathbf{A}^{(1)}\mathbf{S}^{(1)} \in \mathbb{R}^{I \times J} \quad (\text{其中 } \mathbf{A}^{(1)} \in \mathbb{R}^{I \times K}) \\ \mathbf{S}^{(1)} &\approx \mathbf{A}^{(2)}\mathbf{S}^{(2)} \in \mathbb{R}^{K \times J} \quad (\text{其中 } \mathbf{A}^{(2)} \in \mathbb{R}^{K \times K}) \\ &\vdots \\ \mathbf{S}^{(L-1)} &\approx \mathbf{A}^{(L)}\mathbf{S}^{(L)} \in \mathbb{R}^{K \times J} \quad (\text{其中 } \mathbf{A}^{(L)} \in \mathbb{R}^{K \times K}) \end{aligned}$$

每层分解都是非负矩阵分解，可以采用前述任何一种算法进行。最后的多层分解结果为

$$\mathbf{X} \approx \mathbf{A}^{(1)}\mathbf{A}^{(2)} \dots \mathbf{A}^{(L)}\mathbf{S}^{(L)}$$

由此得非负矩阵分解  $\mathbf{A} = \mathbf{A}^{(1)}\mathbf{A}^{(2)} \dots \mathbf{A}^{(L)}$  和  $\mathbf{S} = \mathbf{S}^{(L)}$ 。

### 6.7.5 稀疏非负矩阵分解

非负矩阵分解最有用的性质之一是它往往会产生数据的稀疏表示。因此，在希望利用非负矩阵分解得到数据的稀疏表示时，有必要考虑具有稀疏度约束的非负矩阵分解。

给定一向量  $\mathbf{x} \in \mathbb{R}^n$ ，Hoyer<sup>[245]</sup> 提出使用  $L_1$  范数和  $L_2$  范数之比

$$\text{sparseness}(\mathbf{x}) = \frac{\sqrt{n} - \|\mathbf{x}\|_1 / \|\mathbf{x}\|_2}{\sqrt{n} - 1} \quad (6.7.56)$$

作为该向量的稀疏度测度 (sparseness measure)。显然，若  $\mathbf{x}$  只有一个非零元素，则其稀疏度等于 1；当且仅当  $\mathbf{x}$  的所有元素的绝对值相等，其稀疏度为零。一向量的稀疏度介于这两个边界值之间。

具有稀疏度约束的非负矩阵分解的定义如下<sup>[245]</sup>: 给定一个非负数据矩阵  $\mathbf{X} \in \mathbb{R}_+^{I \times J}$ , 求非负基矩阵  $\mathbf{A} \in \mathbb{R}_+^{I \times K}$  和非负系数矩阵  $\mathbf{S} \in \mathbb{R}_+^{K \times J}$ , 使得

$$L(\mathbf{A}, \mathbf{S}) = \|\mathbf{X} - \mathbf{AS}\|_F^2 \quad (6.7.57)$$

最小化, 并且  $\mathbf{A}$  和  $\mathbf{S}$  满足以下稀疏度约束

$$\begin{aligned} \text{sparsereness}(\mathbf{a}_k) &= S_a, \quad k = 1, \dots, K \\ \text{sparsereness}(\mathbf{s}_k) &= S_s, \quad k = 1, \dots, K \end{aligned}$$

其中,  $\mathbf{a}_k$  和  $\mathbf{s}_k$  分别是非负矩阵  $\mathbf{A}$  的第  $k$  列和  $\mathbf{S}$  的第  $k$  行。另外,  $K$  为分量的个数,  $S_a$  和  $S_s$  分别是  $\mathbf{A}$  的各列和  $\mathbf{S}$  的各行的(期望)稀疏度, 这三个参数由用户根据应用对象决定。

稀疏约束非负矩阵分解的要点是:

- (1) 约束非负基矩阵  $\mathbf{A}$  的各个列向量为稀疏向量, 并且具有相同的列稀疏度。
- (2) 约束非负系数矩阵  $\mathbf{S}$  的各个行向量为稀疏向量, 并且具有相同的行稀疏度。

下面是具有稀疏度约束的非负矩阵分解的基本框架<sup>[245]</sup>:

1. 用两个随机正矩阵分别初始化矩阵  $\mathbf{A}$  和  $\mathbf{S}$ 。

2. 若对矩阵  $\mathbf{A}$  运用稀疏度约束, 则

(1) 令  $\mathbf{A} \leftarrow \mathbf{A} - \mu_{\mathbf{A}}(\mathbf{AS} - \mathbf{X})\mathbf{S}^T$ 。

(2) 将  $\mathbf{A}$  的每一个列向量投影成一个  $L_2$  范数不变, 但  $L_1$  范数与期望的稀疏度相等的新的非负列向量。

若对矩阵  $\mathbf{A}$  不运用稀疏度约束, 则取标准的乘法算法  $\mathbf{A} = \mathbf{A} * (\mathbf{XS}^T) \oslash (\mathbf{ASS}^T)$ 。

3. 若对矩阵  $\mathbf{S}$  运用稀疏度约束, 则

(1) 令  $\mathbf{S} \leftarrow \mathbf{S} - \mu_{\mathbf{S}}\mathbf{A}^T(\mathbf{AS} - \mathbf{X})$ 。

(2) 将  $\mathbf{S}$  的每一个行向量投影成一个  $L_2$  范数不变, 但  $L_1$  范数与期望的稀疏度相等的新的非负行向量。

若对矩阵  $\mathbf{S}$  不运用稀疏度约束, 则取标准的乘法算法  $\mathbf{S} = \mathbf{S} * (\mathbf{A}^T \mathbf{X}) \oslash (\mathbf{A}^T \mathbf{AS})$ 。

给定任一向量  $\mathbf{x} \in \mathbb{R}^n$ , 下面的算法<sup>[245]</sup>求与  $\mathbf{x}$  的 Euclidean 距离最近的非负向量  $\mathbf{s}$ , 它具有给定的  $L_1$  范数  $L_1$  和给定的  $L_2$  范数  $L_2$ 。

1. 初始化 令  $Z = \{\}$  和  $s_i = x_i + (L_1 - \sum x_i)/\dim(\mathbf{x})$ ,  $i = 1, \dots, n$ 。

2. 迭代

(1) 令  $m_i = \begin{cases} L_1/(\dim(\mathbf{x}) - \text{size}(Z)), & \text{若 } i \notin Z \\ 0, & \text{若 } i \in Z \end{cases}$

(2) 选择  $\alpha \geq 0$  满足  $\alpha\|\mathbf{s}\|_2 + (1 - \alpha)\|\mathbf{m}\|_2 = L_2$ , 并令  $\mathbf{s} = \mathbf{m} + \alpha(\mathbf{s} - \mathbf{m})$ 。

(3) 若  $\mathbf{s}$  的所有元素都是非负的, 则输出  $\mathbf{s}$ , 并停止迭代; 否则, 进行下一步。

(4) 令  $Z = Z \cup \{i | s_i < 0\}$ 。

(5) 令  $s_i = 0, \forall i \in Z$ 。

(6) 计算  $c = (\sum s_i - L_1) / (\dim(\mathbf{x}) - \text{size}(Z))$ 。

(7) 令  $s_i = s_i - c, \forall i \notin Z$ 。

(8) 返回 (1), 并继续迭代。

以上算法适用于对非负基矩阵  $\mathbf{A}$  的列向量与/或非负系数矩阵  $\mathbf{S}$  的行向量有  $L_1$  范数和  $L_2$  范数约束的非负矩阵分解。

下面考虑对矩阵  $\mathbf{S}$  的列加稀疏性约束的正则化非负矩阵分解

$$\min_{\mathbf{A}, \mathbf{S}} \frac{1}{2} \left( \|\mathbf{AS} - \mathbf{X}\|_F^2 + \alpha \|\mathbf{A}\|_F^2 + \beta \sum_{j=1}^J \|S_{:,j}\|_1^2 \right) \quad \text{subject to } \mathbf{A}, \mathbf{S} \geq 0 \quad (6.7.58)$$

其中  $S_{:,j}$  表示矩阵  $\mathbf{S}$  的第  $j$  列。

稀疏非负矩阵分解问题式 (6.7.58) 可以分解为两个交替最小二乘子问题 [269]

$$\min_{\mathbf{A} \in \mathbb{R}_+^{I \times K}} J_3(\mathbf{S}) = \frac{1}{2} \left\| \begin{bmatrix} \mathbf{A} \\ \sqrt{\beta} \mathbf{1}_K^\top \end{bmatrix} \mathbf{S} - \begin{bmatrix} \mathbf{X} \\ \mathbf{O}_{K \times J} \end{bmatrix} \right\|_F^2 \quad (6.7.59)$$

$$\min_{\mathbf{S} \in \mathbb{R}_+^{K \times J}} J_4(\mathbf{A}^\top) = \frac{1}{2} \left\| \begin{bmatrix} \mathbf{S}^\top \\ \sqrt{\alpha} \mathbf{I}_K \end{bmatrix} \mathbf{A}^\top - \begin{bmatrix} \mathbf{X}^\top \\ \mathbf{O}_{K \times I} \end{bmatrix} \right\|_F^2 \quad (6.7.60)$$

式中  $\mathbf{1}_K$  是一个全部元素为 1 的  $K$  维列向量。式 (6.7.60) 的最小化相当于使矩阵  $\mathbf{S} \in \mathbb{R}^{K \times J}$  的各列的  $L_1$  范数最小化, 即对  $\mathbf{S}$  规定稀疏度。

由矩阵微分

$$\begin{aligned} dJ_3(\mathbf{S}) &= \frac{1}{2} d \left( \text{tr}[(\mathbf{AS} - \mathbf{X})^\top (\mathbf{AS} - \mathbf{X}) + \beta \mathbf{S}^\top \mathbf{1}_K \mathbf{1}_K^\top \mathbf{S}] \right) \\ &= \text{tr} \left( (\mathbf{S}^\top \mathbf{A}^\top \mathbf{A} - \mathbf{X}^\top \mathbf{A} + \beta \mathbf{S}^\top \mathbf{E}_K) d\mathbf{S} \right) \\ dJ_4(\mathbf{A}^\top) &= dJ_2(\mathbf{A}^\top) = \text{tr} \left( (\mathbf{A} \mathbf{S} \mathbf{S}^\top - \mathbf{X} \mathbf{S}^\top + \alpha \mathbf{A}) d\mathbf{A}^\top \right) \end{aligned}$$

即得稀疏非负最小二乘问题的目标函数的梯度矩阵

$$\frac{\partial J_3(\mathbf{S})}{\partial \mathbf{S}} = -\mathbf{A}^\top \mathbf{X} + \mathbf{A}^\top \mathbf{AS} + \beta \mathbf{E}_J \mathbf{S} \quad (6.7.61)$$

$$\frac{\partial J_4(\mathbf{A}^\top)}{\partial \mathbf{A}^\top} = -\mathbf{SX}^\top + \mathbf{SS}^\top \mathbf{A}^\top + \alpha \mathbf{A}^\top \quad (6.7.62)$$

式中,  $\mathbf{E}_K = \mathbf{1}_K \mathbf{1}_K^\top$  是一个全部元素为 1 的  $K \times K$  矩阵。

于是, 交替稀疏非负最小二乘解分别由

$$(\mathbf{A}^\top \mathbf{A} + \beta \mathbf{E}_J) \mathbf{S} = \mathbf{A}^\top \mathbf{X} \quad \text{或} \quad \mathbf{S} = (\mathbf{A}^\top \mathbf{A} + \beta \mathbf{E}_J)^{-1} \mathbf{A}^\top \mathbf{X} \quad (6.7.63)$$

$$(\mathbf{SS}^\top + \alpha \mathbf{I}_J) \mathbf{A}^\top = \mathbf{SX}^\top \quad \text{或} \quad \mathbf{A}^\top = (\mathbf{SS}^\top + \alpha \mathbf{I}_J)^{-1} \mathbf{SX}^\top \quad (6.7.64)$$

给出。

另一方面, 稀疏非负矩阵分解的梯度算法为

$$S_{kj} \leftarrow S_{kj} + \eta_{kj} (\mathbf{A}^\top \mathbf{X} - \mathbf{A}^\top \mathbf{AS} - \beta \mathbf{E}_K \mathbf{S})$$

$$A_{ik}^\top \leftarrow A_{ik}^\top + \mu_{ik} (\mathbf{SX}^\top - \mathbf{SS}^\top \mathbf{A}^\top - \alpha \mathbf{A}^\top)$$

若选择步长

$$\eta_{kj} = \frac{S_{kj}}{[A^T AS + \beta E_J S]_{kj}}, \quad \mu_{ik} = \frac{A_{ik}^T}{[SS^T A^T + \alpha A^T]_{ik}},$$

则梯度算法变成乘法算法

$$S_{kj} \leftarrow S_{kj} \frac{[A^T X]_{kj}}{[A^T AS + \beta E_J S]_{kj}} \quad (6.7.65)$$

$$A_{ik}^T \leftarrow A_{ik}^T \frac{[SX^T]_{ik}}{[SS^T A^T + \alpha A^T]_{ik}} \quad (6.7.66)$$

这就是稀疏非负矩阵分解的交替最小二乘乘法算法 [269]。

## 6.8 稀疏矩阵方程求解：优化理论

在稀疏表示与压缩感知中，需要求解欠定的稀疏矩阵方程  $Ax = b$ ，其中  $x$  为稀疏向量，只有少量元素不等于零。本节讨论求解稀疏矩阵方程的优化理论。

### 6.8.1 $L_1$ 范数最小化

如第 1 章所述，稀疏表示和压缩感知的核心问题是  $L_0$  拟范数最小化

$$(P_0) \quad \min_x \|x\|_0 \quad \text{subject to } y = \Phi x \quad (6.8.1)$$

其中  $\Phi \in \mathbb{R}^{m \times n}$ ,  $x \in \mathbb{R}^n$ ,  $y \in \mathbb{R}^m$ 。

由于观测信号通常被噪声污染，所以上述优化问题中的等式约束常松弛为允许某个误差扰动  $\epsilon \geq 0$  的不等式约束的  $L_0$  拟范数最小化问题

$$\min_x \|x\|_0 \quad \text{subject to} \quad \|\Phi x - y\|_2 \leq \epsilon \quad (6.8.2)$$

直接求解优化问题式  $(P_0)$  或式  $(6.8.2)$ ，必须筛选出系数向量  $x$  中所有可能的非零元素。此方法是不可跟踪的 (intractable) 或 NP 困难的，因为搜索空间过于庞大 [331, 129, 355]。

向量  $x = [x_1, \dots, x_N]^T$  的非零元素的指标集合称为支撑区，用符号  $\text{supp}(x) = \{i | x_i \neq 0\}$  表示，支撑区的长度即非零元素的个数用  $L_0$  拟范数

$$\|x\|_0 = |\text{supp}(x)| \quad (6.8.3)$$

度量。一个向量  $x \in \mathbb{C}^N$  称为  $K$ -稀疏的，若  $\|x\|_0 \leq K$ ，其中  $K \in \{1, \dots, N\}$ 。

$K$ -稀疏向量的集合记为

$$\Sigma_K = \{x \in \mathbb{C}^N \mid \|x\|_0 \leq K\} \quad (6.8.4)$$

若  $\hat{x} \in \Sigma_K$ ，则称向量  $\hat{x} \in \mathbb{C}^N$  是  $x \in \mathbb{C}^N$  的  $K$ -项逼近或者  $K$ -稀疏逼近。

对于任何正数  $p > 0$ , 定义向量的  $L_p$  范数

$$\|\mathbf{x}\|_p = \left( \sum_{i \in \text{supp}(\mathbf{x})} |x_i|^p \right)^{1/p} \quad (6.8.5)$$

给定  $L$  个  $M$  维实数输入向量  $\{\mathbf{y}_1, \dots, \mathbf{y}_L\}$ , 它们组成数据矩阵  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_L] \in \mathbb{R}^{M \times L}$ 。稀疏编码 (sparse coding) 问题的提法是: 确定  $N$  个  $M$  维基向量  $\mathbf{a}_1, \dots, \mathbf{a}_N \in \mathbb{R}^M$ , 以及对每一个输入向量  $\mathbf{y}_l$ , 确定一个  $N$  维稀疏的权向量或系数向量  $\mathbf{s}_l \in \mathbb{R}^N$ , 使得少数基向量的加权线性组合即可逼近原输入向量

$$\mathbf{y}_l = \sum_{i=1}^N s_{l,i} \mathbf{a}_i = \mathbf{A} \mathbf{s}_l, \quad l = 1, \dots, L \quad (6.8.6)$$

式中  $s_{l,i}$  表示稀疏权向量  $\mathbf{s}_l$  的第  $i$  个元素。

稀疏编码可视为神经编码的一种形式: 由于权向量是稀疏的, 所以对于每一个输入向量, 只有少量的神经元 (基向量) 被强激励; 而且输入向量不同时, 被激励的神经元也各异。

称  $\hat{\mathbf{x}}$  是  $\mathbf{x}$  在  $L_p$  范数条件下的最优  $K$ -稀疏逼近, 若逼近误差向量  $\mathbf{x} - \hat{\mathbf{x}}$  的  $L_p$  范数达到下确界, 即

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_p = \inf_{\mathbf{z} \in \Sigma_K} \|\mathbf{x} - \mathbf{z}\|_p$$

显然,  $L_0$  范数定义式 (6.8.3) 与  $L_p$  范数定义式 (6.8.5) 之间存在密切的关系: 当  $p \rightarrow 0$  时,  $\|\mathbf{x}\|_0 = \lim_{p \rightarrow 0} \|\mathbf{x}\|_p^p$ 。由于当且仅当  $p \geq 1$  时  $\|\mathbf{x}\|_p$  为凸函数, 所以  $L_1$  范数是最接近于  $L_0$  拟范数的凸目标函数。于是, 从最优化的角度, 称  $L_1$  范数是  $L_0$  拟范数的凸松弛。因此,  $L_0$  拟范数最小化问题 ( $P_0$ ) 便可转变为凸松弛的  $L_1$  范数最小化问题

$$(P_1) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to } \mathbf{y} = \Phi \mathbf{x} \quad (6.8.7)$$

这是一个凸优化问题, 因为作为目标函数的  $L_1$  范数  $\|\mathbf{x}\|_1$  本身是凸函数, 而等式约束  $\mathbf{y} = \Phi \mathbf{x}$  又是仿射函数。

存在观测噪声的实际情况下, 等式约束的最优化问题 ( $P_1$ ) 又可松弛为不等式约束的最优化问题

$$(P_{10}) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - \Phi \mathbf{x}\|_2 \leq \epsilon \quad (6.8.8)$$

$L_1$  范数下的最优化问题又称为基追踪 (base pursuit, BP)。这是一个二次约束线性规划 (quadratically constrained linear program, QCLP) 问题。

若  $\mathbf{x}_1$  是 ( $P_1$ ) 的解, 且  $\mathbf{x}_0$  是 ( $P_0$ ) 的解, 则 [141]

$$\|\mathbf{x}_1\|_1 \leq \|\mathbf{x}_0\|_1 \quad (6.8.9)$$

因为  $\mathbf{x}_0$  只是  $(P_1)$  的可行解, 而  $\mathbf{x}_1$  则是  $(P_1)$  的最优解; 同时有

$$\Phi \mathbf{x}_1 = \Phi \mathbf{x}_0 \quad (6.8.10)$$

与不等式约束  $L_0$  范数最小化式 (6.8.2) 相类似, 不等式约束  $L_1$  范数最小化表达式 (6.8.8) 也有两种变型:

(1) 利用  $\mathbf{x}$  是  $K$  稀疏向量的约束, 将不等式约束  $L_1$  范数最小化变成不等式约束的  $L_2$  范数最小化

$$(P_{11}) \quad \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_1 \leq q \quad (6.8.11)$$

这是一个二次规划 (quadratic program, QP) 问题。

(2) 利用 Lagrangian 乘子法, 将不等式约束的  $L_1$  范数最小化变成

$$(P_{12}) \quad \min_{\lambda, \mathbf{x}} \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \quad (6.8.12)$$

这一最小化问题称为基追踪去噪 (basis pursuit denoising, BPDN)<sup>[105]</sup>。其中, Lagrangian 乘子称为正则化参数, 用于控制稀疏解的稀疏度:  $\lambda$  取值越大, 解  $\mathbf{x}$  越稀疏。当正则化参数  $\lambda$  足够大时, 解  $\mathbf{x}$  为零向量; 随着  $\lambda$  的逐渐减小, 解向量  $\mathbf{x}$  的稀疏度也逐渐减小; 当  $\lambda$  逐渐减小至 0 时, 解向量  $\mathbf{x}$  便变成使得  $\|\mathbf{y} - \Phi \mathbf{x}\|_2^2$  最小化的向量。也就是说,  $\lambda > 0$  可以平衡双重目标函数 (twin objectives)

$$J(\lambda, \mathbf{x}) = \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \quad (6.8.13)$$

中的误差平方和代价函数  $\frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2$  及  $L_1$  范数代价函数  $\|\mathbf{x}\|_1$ 。

优化问题  $(P_{10})$  和  $(P_{11})$  分别称为误差约束的  $L_1$ -最小化和  $L_1$ -惩罚最小化<sup>[479]</sup>。

在基于小波的图像/信号重构和恢复 (即解卷积) 中, 也会遇到优化问题  $(P_{12})$ 。此时, 矩阵  $\Phi$  具有形式  $\mathbf{R}\mathbf{W}$ , 其中  $\mathbf{R}$  是观测算子的一种矩阵表示, 而  $\mathbf{W}$  则由小波基或冗余字典组成, 并且  $\mathbf{x}$  为未知图像/信号的表示系数<sup>[156, 166]</sup>。

$L_1$  范数最小化也称  $L_1$  线性规划或  $L_1$  范数正则化最小二乘。

## 6.8.2 RIP 条件

$L_1$  范数最小化问题  $(P_1)$  是  $L_0$  范数最小化  $(P_0)$  某种程度的凸松弛。与  $L_0$  范数最小化问题具有不可跟踪性不同,  $L_1$  范数最小化问题具有可跟踪性 (trackability)。一个自然会问的问题是“这两种优化问题的解之间究竟有何关系”?

**定义 6.8.1** 约束等距性 (restricted isometry property, RIP) 条件: 称矩阵  $\Phi$  满足  $K$  阶 RIP 条件, 若

$$\|\mathbf{x}\|_0 \leq K \implies (1 - \delta_K) \|\mathbf{x}\|_2^2 \leq \|\Phi_K \mathbf{x}\|_2^2 \leq (1 + \delta_K) \|\mathbf{x}\|_2^2 \quad (6.8.14)$$

式中  $0 \leq \delta_K < 1$  是一个与稀疏度  $K$  有关的常数，而  $\Phi_K$  是由字典矩阵  $\Phi$  的任意  $K$  列组成的子矩阵。

RIP 条件由 Candes 和 Tao 于 2006 年提出<sup>[79]</sup>，后经 Foucart 和 Lai 于 2009 年加以细化<sup>[173]</sup>。

当 RIP 条件满足时，非凸的  $L_0$  范数最小化 ( $P_0$ ) 与凸的  $L_1$  范数最小化 ( $P_1$ ) 等价。

令  $I = \{i|x_i \neq 0\} \subset \{1, \dots, n\}$  表示稀疏向量  $x$  的非零元素的支撑区， $|I|$  表示支撑区的长度即稀疏向量  $x$  的非零元素的个数。

具有参数  $\delta_K$  的  $K$  阶 RIP 条件常简记为  $\text{RIP}(K, \delta_K)$ ，而  $\delta_K$  常称为约束等距常数 (restricted isometry constants, RIC)，定义为所有使  $\text{RIP}(K, \delta_K)$  成立的参数  $\delta$  的下确界

$$\delta_K = \inf \left\{ \delta \mid (1 - \delta) \|z\|_2^2 \leq \|\Phi_I z\|_2^2 \leq (1 + \delta) \|z\|_2^2, \forall |I| \leq K, \forall z \in \mathbb{R}^{|I|} \right\} \quad (6.8.15)$$

由定义 6.8.1 不难看出，若矩阵  $\Phi_K$  为正交矩阵，则  $\delta_K = 0$ ，因为  $\|\Phi_K x\|_2 = \|x\|_2$ 。于是，一个矩阵的约束等距常数  $\delta_K$  的非零值实际上可以评价该矩阵的非正交程度。另外，由于  $\Phi_K$  是由  $\Phi$  的  $K$  列的任意抽取构成的，故要求信号  $x$  在  $\Phi$  的每一列上的能量投影都尽可能地均匀。这就是限制等距的物理含义。

若  $K + K' < P$ ，则字典矩阵  $\Phi$  的约束正交常数 (restricted orthogonality constant, ROC)  $\theta_{K, K'}$  定义为满足下列不等式的最小常数<sup>[77]</sup>

$$\langle \Phi z, \Phi z' \rangle \leq \theta_{K, K'} \|z\|_2 \|z'\|_2 \quad (6.8.16)$$

式中  $z$  和  $z'$  分别是  $K$  稀疏和  $K'$  稀疏的任意两个向量，并且它们的支撑区无交连。

下面是约束等距常数的性质。

(1) 稀疏信号精确重构的充分条件 若字典矩阵  $\Phi$  分别满足具有常数  $\delta_K, \delta_{2K}, \delta_{3K}$  的 RIP 条件，并且

$$\delta_K + \delta_{2K} + \delta_{3K} < 1 \quad (6.8.17)$$

则  $L_1$  范数最小化可以精确重构所有  $K$  稀疏的信号<sup>[78]</sup>。上述充分条件也可以改善为<sup>[84]</sup>

$$\delta_{2K} < \sqrt{2} - 1 \quad (6.8.18)$$

文献 [77] 最近证明了约束等距常数的新下界为

$$\delta_K < 0.307 \quad (6.8.19)$$

在此条件下，若无噪声存在，则  $K$  稀疏信号可以确保由  $L_1$  范数最小化精确恢复；并且在有噪声情况下  $K$  稀疏信号则可由  $L_1$  范数最小化稳定地估计。

(2) 约束等距常数与特征值的关系 若字典矩阵  $\Phi \in \mathbb{R}^{m \times n}$  满足  $\text{RIP}(K, \delta_K)$ ，则约束等距常数与特征值之间存在下列不等式关系<sup>[126]</sup>

$$1 - \delta_K \leq \lambda_{\min}(\Phi_I^T \Phi_I) \leq \lambda_{\max}(\Phi_I^T \Phi_I) \leq 1 + \delta_K \quad (6.8.20)$$

式中  $\lambda_{\min}(\Phi_I^T \Phi_I)$  和  $\lambda_{\max}(\Phi_I^T \Phi_I)$  分别表示  $\Phi_I^T \Phi_I$  的最小和最大特征值。

(3) 约束等距常数  $\delta_K$  和约束正交常数  $\theta_{K,K'}$  的单调性<sup>[77]</sup>

$$\delta_K \leq \delta_{K_1} \quad (\text{若 } K \leq K_1) \quad (6.8.21)$$

$$\theta_{K,K'} \leq \theta_{K_1,K'_1} \quad (\text{若 } K \leq K_1; K' \leq K'_1) \quad (6.8.22)$$

RIP 条件与矩阵  $\Phi$  的列之间的相干统计量

$$\mu = \max_{j \neq k} |\langle \phi_j, \phi_k \rangle| \quad (6.8.23)$$

密切相关。Donoho 与 Elad [139] 借助 Gershgorin 圆盘定理证明了  $\delta_K \leq \mu(K-1)$ 。在信号处理应用中，常取  $\mu \approx m^{-1/2}$ ，由此得非平凡的 RIP 临界  $K \approx \sqrt{m}$ 。对于某些随机矩阵，它们具有更高的 RIP 临界，例如高斯随机矩阵和 Bernoulli 随机矩阵的 RIP 临界为  $K = m / \log(n/m)$ ，这就解释了在压缩感知中将随机矩阵作为测量矩阵的优势。

$L_0$  范数稀疏逼近算法通常收敛缓慢，除非字典矩阵  $\Phi$  允许有快速的矩阵-向量乘法。幸运的是，对在某个变换域(如频域和小波域)可压缩的自然信号或图像，压缩后的系数(如 Fourier 变换系数和小波变换系数)是稀疏的，相应的变换矩阵  $\Phi$  为部分正交矩阵(partial orthogonal matrix)。部分正交矩阵有快速的矩阵-向量乘法。

对一个  $n \times n$  正交矩阵随机抽取  $m$  个行向量得到的矩阵称为  $m \times n$  部分正交矩阵(partial orthogonal matrices)。常用的部分正交矩阵有以下三类：

(1) 部分 Fourier 矩阵(partial Fourier matrix, PFM) 从  $n \times n$  维 Fourier 矩阵随机抽取  $m$  行得到的  $m \times n$  维部分正交矩阵。由于部分 Fourier 矩阵可用于从部分频域信息对医学数据进行模型重构，所以在医学成像和光谱学中具有重要的应用。

(2) 部分小波矩阵(partial wavelet matrix, PWM) 每一行由  $n$  维正交小波基构造的  $m \times n$  维矩阵。

(3) 部分 Hadamard 矩阵(partial Hadamard matrix, PHM) 从  $n \times n$  维 Hadamard 矩阵随机抽取  $m$  行组成的  $m \times n$  维部分正交矩阵。

当字典矩阵  $\Phi$  是部分 Fourier 矩阵或者部分小波矩阵时，矩阵-向量乘法  $\Phi \mathbf{x}$  可以利用快速 Fourier 变换(FFT)算法或快速小波变换(FWT)算法。若字典矩阵  $\Phi$  是部分 Hadamard 矩阵，则矩阵-向量乘法  $\Phi \mathbf{x}$  实质上变成了向量元素的加法，因为部分 Hadamard 矩阵的每一行的元素只取 +1 或者 -1。

### 6.8.3 与 Tikhonov 正则化最小二乘的关系

在 Tikhonov 正则化最小二乘问题中，用未知系数向量  $\mathbf{x}$  的  $L_1$  范数代替正则项中的  $L_2$  范数，即得到  $L_1$  正则化最小二乘问题

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \quad (6.8.24)$$

$L_1$  正则化最小二乘问题总是有解，但不一定是唯一解。

令  $\mathbf{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^n$ ，则  $L_1$  范数和  $L_2$  范数之间有不等式<sup>[77]</sup>

$$0 \leq \| \mathbf{x} \|_2 - \frac{\| \mathbf{x} \|_1}{\sqrt{n}} \leq \frac{\sqrt{n}}{4} \left( \max_{1 \leq i \leq n} x_i - \min_{1 \leq i \leq n} x_i \right) \quad (6.8.25)$$

等号成立，当且仅当  $|x_1| = \dots = |x_n|$ 。这一不等式描述了  $L_1$  范数最小化的解向量与 Tikhonov 正则化最小二乘解向量之间的关系。

下面是  $L_1$  正则化的性质，反映了  $L_1$  正则化与 Tikhonov 正则化二者之间的类似点与不同点<sup>[270]</sup>：

(1) 非线性 与 Tikhonov 正则化问题的解向量  $\mathbf{x}$  是观测数据向量  $\mathbf{y}$  的线性函数不同， $L_1$  正则化问题的解向量不是观测数据向量的线性函数。

(2)  $\lambda \rightarrow 0$  时的极限特性 当  $\lambda \rightarrow 0$  时，Tikhonov 正则化问题的解的极限点在满足  $\Phi^H(\mathbf{y} - \Phi\mathbf{x}) = \mathbf{0}$  的所有可行点中具有最小  $L_2$  范数  $\| \mathbf{x} \|_2$ 。与之不同，当  $\lambda \rightarrow 0$  时， $L_1$  正则化问题的解的极限点在满足  $\Phi^H(\mathbf{y} - \Phi\mathbf{x}) = \mathbf{0}$  的所有可行点中具有最小  $L_1$  范数  $\| \mathbf{x} \|_1$ 。

(3) 当  $\lambda \geq \lambda_{\max}$  有限大时的极限特性 当  $\lambda \rightarrow \infty$  时，Tikhonov 正则化问题的最优解收敛为零向量；然而，只要

$$\lambda \geq \lambda_{\max} = \| \Phi^H \mathbf{y} \|_{\infty} \quad (6.8.26)$$

则  $L_1$  正则化问题的解便收敛为零向量。式中  $\| \mathbf{w} \|_{\infty} = \max\{w_i\}$  为向量  $\mathbf{w}$  的  $L_{\infty}$  范数。

(4) 正则化路径 当正则化参数  $\lambda$  在  $[0, \infty)$  区间变化时，Tikhonov 正则化问题的最优解是正则化参数的光滑函数。与之不同，当  $\lambda$  在定义域  $[0, \infty)$  内变化时， $L_1$  正则化最小二乘问题的解族则具有分段线性的求解路径性质<sup>[154]</sup>：存在正则化参数  $\lambda_1, \dots, \lambda_k$ （其中  $0 = \lambda_k < \dots < \lambda_1 = \lambda_{\max}$ ），使得  $L_1$  正则化问题的解向量是分段线性的

$$\mathbf{x}_{L_1} = \frac{\lambda_i - \lambda}{\lambda_i - \lambda_{i+1}} \mathbf{x}_{L_1}^{(i+1)} - \frac{\lambda - \lambda_{i+1}}{\lambda_i - \lambda_{i+1}} \mathbf{x}_{L_1}^{(i)} \quad (6.8.27)$$

其中  $\lambda_{i+1} \leq \lambda \leq \lambda_i ; i = 1, \dots, k-1$ ，并且  $\mathbf{x}_{L_1}^{(i)}$  表示正则化参数取  $\lambda_i$  时  $L_1$  正则化问题的解向量，而  $\mathbf{x}_{L_1}$  为  $L_1$  正则化问题的最优解。因此有  $\mathbf{x}_{L_1}^{(1)} = \mathbf{0}$  和  $\mathbf{x}_{L_1} = \mathbf{0}$ ，当  $\lambda \geq \lambda_1$  时。

$L_1$  正则化最小二乘问题与 Tikhonov 正则化最小二乘问题的根本区别是： $L_1$  正则化最小二乘问题的解向量通常是稀疏向量，而 Tikhonov 正则化最小二乘问题的解中的所有的系数一般是非零的。

#### 6.8.4 $L_1$ 范数最小化的梯度分析

一个实值变元  $t \in \mathbb{R}$  的正负号函数 (signum function) 定义为

$$\operatorname{sgn}(t) = \begin{cases} +1, & t > 0 \\ 0, & t = 0 \\ -1, & t < 0 \end{cases} \quad (6.8.28)$$

$t \in \mathbb{R}$  的正负号多值函数 (signum multifunction) 又称集 (合) 值函数 (set-valued function), 定义为<sup>[213]</sup>

$$\text{SGN}(t) = \frac{\partial|t|}{\partial t} = \begin{cases} \{+1\}, & t > 0 \\ [-1, +1], & t = 0 \\ \{-1\}, & t < 0 \end{cases} \quad (6.8.29)$$

正负号多值函数也称  $|t|$  的次微分 (subdifferential)。

对于  $L_1$  范数优化问题

$$\min_{\mathbf{x}} J(\lambda, \mathbf{x}) = \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \quad (6.8.30)$$

其目标函数的梯度向量

$$\nabla_{\mathbf{x}} J(\lambda, \mathbf{x}) = \frac{\partial J(\lambda, \mathbf{x})}{\partial \mathbf{x}} = -\Phi^T(\mathbf{y} - \Phi \mathbf{x}) + \lambda \nabla_{\mathbf{x}} \|\mathbf{x}\|_1 = -\mathbf{c} + \lambda \nabla_{\mathbf{x}} \|\mathbf{x}\|_1 \quad (6.8.31)$$

其中  $\mathbf{c} = \Phi^T(\mathbf{y} - \Phi \mathbf{x})$  称为残差相关向量 (vector of residual correlations), 并且  $\nabla_{\mathbf{x}} \|\mathbf{x}\|_1 = [\nabla_{x_1} \|\mathbf{x}\|_1, \dots, \nabla_{x_n} \|\mathbf{x}\|_1]^T$  是  $L_1$  范数  $\|\mathbf{x}\|_1$  的梯度向量, 其第  $i$  个元素

$$\nabla_{x_i} \|\mathbf{x}\|_1 = \frac{\partial \|\mathbf{x}\|_1}{\partial x_i} = \begin{cases} \{+1\}, & x_i > 0 \\ \{-1\}, & x_i < 0 \\ [-1, +1], & x_i = 0 \end{cases} \quad (i = 1, \dots, n) \quad (6.8.32)$$

由式 (6.8.31) 知,  $L_1$  范数最小化问题 ( $P_{12}$ ) 的平稳点由条件  $\nabla_{\mathbf{x}} J(\lambda, \mathbf{x}) = -\mathbf{c} + \lambda \nabla_{\mathbf{x}} \|\mathbf{x}\|_1 = \mathbf{0}$  即

$$\mathbf{c} = \lambda \nabla_{\mathbf{x}} \|\mathbf{x}\|_1 \quad (6.8.33)$$

确定。

若记  $\mathbf{c} = [c(1), \dots, c(n)]^T$ , 并将式 (6.8.32) 代入式 (6.8.33), 则可以将平稳点条件改写为

$$c(i) = \begin{cases} \{+\lambda\}, & x_i > 0 \\ \{-\lambda\}, & x_i < 0 \\ [-\lambda, \lambda], & x_i = 0 \end{cases} \quad (i = 1, \dots, n) \quad (6.8.34)$$

由于  $L_1$  范数最小化是一个凸函数优化问题, 所以上述平稳点条件实际上就是  $L_1$  范数最小化的最优解的充分与必要条件。

平稳点条件式 (6.8.34) 可以用残差相关向量表示为

$$c(I) = \lambda \cdot \text{sgn}(\mathbf{x}) \quad \text{和} \quad |c(I^c)| \leq \lambda \quad (6.8.35)$$

其中  $I^c = \{1, \dots, n\} - I$  是支撑区  $I$  的补集。这表明, 支撑区内的残差相关的幅值大小等于  $\lambda$ , 符号则与向量  $\mathbf{x}$  的相应元素的符号一致。

式 (6.8.35) 可以等价写作

$$|c(j)| = \lambda \quad \forall j \in I \quad \text{和} \quad |c(j)| \leq \lambda \quad \forall j \in I^c \quad (6.8.36)$$

也就是说, 支撑区内的残差相关的绝对值等于  $\lambda$ ; 而支撑区以外的残差相关的绝对值则小于或者等于  $\lambda$ , 即有  $\|\mathbf{c}\|_\infty = \max\{c(j)\} = \lambda$ 。

## 6.9 稀疏矩阵方程求解：优化算法

6.8 节讨论了稀疏矩阵方程求解的  $L_1$  范数最小化理论，本节讨论求解稀疏矩阵方程的  $L_1$  范数最小化的具体算法。尽管优化算法各不相同，但是它们有共同的基本思想：利用式 (6.8.36)，通过稀疏向量的支撑区的识别，将欠定的稀疏矩阵方程变换为超定的（非稀疏）矩阵方程的求解。

### 6.9.1 正交匹配追踪法

正交匹配追踪法是信号处理文献中拟合稀疏模型的一种贪婪分步最小二乘 (greedy stepwise least squares) 法。

求解欠定矩阵方程  $\Phi_{m \times n} \mathbf{x}_{n \times 1} = \mathbf{y}_{m \times 1}$  ( $m \ll n$ ) 具有稀疏度  $s$  的整体最优解的一般方法是：先求超定方程  $\mathbf{A}_{m \times s} \tilde{\mathbf{x}}_{s \times 1} = \mathbf{y}$  (通常  $m \gg s$ ) 的最小二乘解，并从中确定最优解。其中， $\mathbf{A}$  由矩阵  $\Phi$  的  $s$  个列向量组成， $\tilde{\mathbf{x}}$  则由  $\mathbf{x}$  中与矩阵  $\Phi$  被抽取列标号对应的元素组成。由于超定方程共有  $C_m^s$  种组合形式，整体求解既费时，又费事。

贪婪算法 (greedy algorithm)<sup>[478]</sup> 的基本思想是：不求整体最优解，而是试图尽快找到在某种意义上的局部最优解。贪婪法虽然不能够对所有问题得到整体最优解，但对范围相当广泛的许多问题能产生整体最优解或者整体最优解的近似解。

典型的贪婪算法有以下匹配追踪算法：

(1) 匹配追踪 (matching pursuit, MP) 法 由 Mallat 和 Zhang 于 1993 年提出<sup>[331]</sup>，其基本思想是，不是针对某个代价函数进行最小化，而是考虑迭代地构造一个稀疏解  $\mathbf{x}$ ：只使用字典矩阵  $\Phi$  的少数列向量（简称原子）的线性组合对观测向量  $\mathbf{x}$  实现稀疏逼近  $\Phi \mathbf{x} = \mathbf{y}$ ，其中字典矩阵  $\Phi$  被选择的列向量所组成的作用集是以逐列的方式建立的。在每一步迭代，字典矩阵中同当前残差向量  $r = \Phi \mathbf{x} - \mathbf{y}$  最相似的列向量被选择作为作用集的新的一列。如果残差随着迭代的进行递减，则可以保证算法收敛。

(2) 正交匹配追踪 (orthogonal matching pursuit, OMP)<sup>[396, 129, 182]</sup> 匹配追踪只能保证残差向量与每一步迭代所选择的字典矩阵列向量正交，但与以前选择的列向量一般不正交。正交匹配追踪则能够保证每步迭代后残差向量与以前选择的所有列向量正交，以保证迭代的最优化，从而减少了迭代次数，性能也更稳健。正交匹配追踪算法复杂度为  $O(mn)$ ，可以得到稀疏度  $K \leq m/(2 \log n)$  的系数向量。

(3) 正则正交匹配追踪 (ROMP)<sup>[358, 359]</sup> 在 OMP 算法基础上，加入正则化过程。首先根据相关原子挑选多个原子作为候选集，然后从候选集中按照正则化原则挑选出部分原子，最后将其并入最终的支撑集，实现原子的快速、有效选择。

(4) 分段正交匹配追踪 (StOMP)<sup>[143]</sup> 将 OMP 算法进行了一定程度的简化，以牺牲逼近精度为代价，进一步提高了计算速度，复杂度为  $O(n)$ ，更适合求解大规模稀疏逼近问题。

(5) 压缩采样匹配追踪 (CoSaMP)<sup>[360]</sup> 引入了回退筛选的思想，是对 ROMP 结果的改进。与 OMP 算法相比，该算法逼近精度更高，复杂度更低，为  $O(n \log^2 n)$ ，稀疏系数向量的稀疏度  $K \leq m/(2 \log(1 + n/K))$ 。

此外，还有梯度追踪 (gradient pursuit) 算法<sup>[50]</sup> 和子空间追踪算法<sup>[126]</sup> 等。

虽然正交匹配追踪与  $L_1$  范数最小化公式 ( $P_1$ ) 无关，但在某些应用中可以成功获得  $L_1$  范数最小化问题 ( $P_1$ ) 的解。

#### 算法 6.9.1 正交匹配追踪算法<sup>[396, 129]</sup>

输入 观测数据向量  $y \in \mathbb{R}^m$  和字典矩阵  $\Phi \in \mathbb{R}^{m \times n}$ 。

输出 稀疏的系数向量  $x \in \mathbb{R}^n$ 。

步骤 1 初始化 令标签集  $\Omega_0 = \emptyset$ ，初始残差向量  $r_0 = y$ ，令  $k = 1$ 。

步骤 2 辨识 求矩阵  $\Phi$  中与残差向量  $r_{k-1}$  最强相关的列

$$j_k \in \arg \max_j |\langle r_{k-1}, \phi_j \rangle|, \quad \Omega_k = \Omega_{k-1} \cup \{j_k\} \quad (6.9.1)$$

步骤 3 估计 最小化问题  $\min_x \|y - \Phi_{\Omega_k} x\|_2$  的解由

$$x_k = (\Phi_{\Omega_k}^H \Phi_{\Omega_k})^{-1} \Phi_{\Omega_k}^H y \quad (6.9.2)$$

给出，其中  $\Phi_{\Omega_k} = [\varphi_{\omega_1}, \dots, \varphi_{\omega_k}]$ ， $\omega_1, \dots, \omega_k \in \Omega_k$ 。

步骤 4 更新残差

$$r_k = y - \Phi_{\Omega_k} x_k \quad (6.9.3)$$

步骤 5 令  $k \leftarrow k + 1$ ，并重复步骤 2 至步骤 4。若某个停止判据满足，则停止迭代。

步骤 6 输出系数向量

$$x(i) = \begin{cases} x_k(i), & i \in \Omega_k \\ 0, & \text{其他} \end{cases} \quad (6.9.4)$$

Sparsify toolbox (<http://www.see.ed.ac.uk/tblumens/sparsify>) 提供了 `greed_omp_qr` 函数。该函数基于 QR 分解，并要求矩阵的每一列都具有单位范数。

下面是三种常用的停止判据<sup>[480]</sup>：

(1) 运行到某个固定的迭代步数后停止。

(2) 残差能量小于某个预先给定值  $\varepsilon$

$$\|r_k\|_2 \leq \varepsilon \quad (6.9.5)$$

(3) 当字典矩阵  $\Phi$  的任何一列都没有残差向量  $r_k$  的明显能量时

$$\|\Phi^H r_k\|_\infty \leq \varepsilon \quad (6.9.6)$$

在第  $k$  步迭代中，正交匹配追踪、分段正交匹配追踪和正则正交匹配追踪等方法均将字典矩阵  $\Phi$  中的新候选列的标签集与第  $k-1$  步迭代的标签集  $\Omega_{k-1}$  合并。一旦一个候选列入选，它将保留在被选列的列表中，直至算法结束。

与之不同，压缩感知信号重构的子空间追踪算法<sup>[126]</sup> 则对  $K$  稀疏信号，保留  $K$  个候选列的标签集不变，而允许其中的候选列在迭代过程中不断更新。

### 算法 6.9.2 子空间追踪算法<sup>[126]</sup>

输入 稀疏度  $K$ ，字典矩阵  $\Phi \in \mathbb{R}^{m \times n}$ ，观测向量  $y \in \mathbb{R}^m$ 。

初始化 (1)  $\Omega_0 = \{\text{向量 } \Phi^T y \text{ 中具有最大幅值的 } K \text{ 个元素的标签集合}\}$ 。

(2) 残差  $r_0 = y - \Phi_{\Omega_0} \Phi_{\Omega_0}^\dagger y$ 。

迭代 对  $k = 1, 2, \dots$ ，执行以下运算。

步骤 1  $\tilde{\Omega}_k = \Omega_{k-1} \cup \{\text{向量 } \Phi_{\Omega_{k-1}}^T r_{k-1} \text{ 中具有最大幅值的 } K \text{ 个标签集合}\}$ 。

步骤 2 计算系数向量  $x_p = \Phi_{\tilde{\Omega}_k}^\dagger y$ 。

步骤 3  $\Omega_k = \{\text{向量 } x_p \text{ 中具有最大幅值的 } K \text{ 个标签集合}\}$ 。

步骤 4  $r_k = y - \Phi_{\Omega_k} \Phi_{\Omega_k}^\dagger y$ 。

步骤 5 若  $\|r_k\|_2 > \|r_{k-1}\|_2$ ，则令  $\Omega_k = \Omega_{k-1}$ ，并退出迭代；否则，令  $k \leftarrow k + 1$ ，并返回步骤 1，继续新一轮迭代。

业已证明<sup>[478, 140]</sup>，正交匹配追踪在某些情况下可以成功地求出最稀疏的解。然而，在  $L_1$  范数最小化成功的某些应用中，正交匹配追踪却可能找不到最稀疏的解<sup>[104, 478, 141]</sup>。

## 6.9.2 LASSO 算法与 LARS 算法

在信号处理界研究有关稀疏表示的优化算法的同时，数理统计界则从统计拟合的角度研究这个问题，并取得重要进展。

为了减少直接求解  $(P_1)$  的计算复杂度，考虑观测向量  $y$  的线性回归问题  $\hat{y} = \Phi \hat{x}$ ，其中向量  $y \in \mathbb{R}^m$  已经零均值化，并且矩阵  $\Phi \in \mathbb{R}^{m \times n}$  的列已经零均值化和  $L_2$  范数已经单位化

$$\sum_{i=1}^m y_i = 0, \quad \sum_{i=1}^m \phi_{ij} = 0, \quad \sum_{i=1}^m \phi_{ij}^2 = 1, \quad j = 1, \dots, n \quad (6.9.7)$$

假定一候选回归系数向量  $\hat{x} = [\hat{x}_1, \dots, \hat{x}_n]^T$  给出预测向量

$$\hat{y} = \sum_{i=1}^n \hat{x}_i \phi_i = \Phi \hat{x} \quad (6.9.8)$$

相对应的误差能量即误差平方和为

$$\|y - \hat{y}\|_2^2 = \sum_{i=1}^m (y_i - \hat{y}_i)^2 \quad (6.9.9)$$

Tibshirani<sup>[471]</sup> 于 1996 年提出了求解线性回归的的最小绝对收缩与选择算子 (least absolute shrinkage and selection operator, LASSO) 算法，其求最优预测向量的基本思想

是：通过约束预测向量的  $L_1$  范数不超过某个上限  $q$ ，使预测误差平方和最小化，即

$$\text{LASSO : } \min_{\mathbf{x}} \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i| \leq q \quad (6.9.10)$$

可见  $L_1$  范数最小化问题的变型 ( $P_{11}$ ) 与 LASSO 算法的模型具有完全相同的形式。

如式 (6.9.10) 所示，LASSO 算法是一种不等式约束的普通最小二乘方法。LASSO 算法的显著特点是它具有的收缩和选择两种基本功能：

(1) 收缩功能 与每一步估计所有未知参数的迭代算法不同，LASSO 算法收缩待估计的参数的范围，每一步只对入选的少数参数进行估计。

(2) 选择功能 使用  $L_1$  范数作为惩罚项，LASSO 算法会自动地选择很少一部分变量进行线性回归。

求解 LASSO 问题的有效方法是 Efron 等人的最小角度回归 (least angle regression, LARS) 算法<sup>[154]</sup>。LARS 算法是一种逐步回归的方法，其基本思想是：以保证当前残差和已入选变量之间的相关系数相等的方式，选择当前残差在已入选变量的构成空间的投影作为求解路径 (solution path)。然后，在这一求解路径上继续搜索，吸收新的变量加入，然后调整求解路径。

令标签集  $\Omega_0 = \emptyset$ ，且残差向量的初始值  $\mathbf{r} = \mathbf{y}$ 。第 1 步迭代找出字典矩阵  $\Phi$  中与残差向量 (此时  $\mathbf{r} = \mathbf{y}$ ) 相关系数  $\hat{c}_i$  最大的列  $\phi_i^{(1)}$ ，并将其加入作用集，即将该列向量的编号计入标签集  $\Omega_1$ ，并得到第 1 个回归变量集  $\Phi_{\Omega_1}$ 。

LARS 算法的基本步骤如下：假设经过  $k-1$  个 LARS 步骤，已经得到一个回归变量集  $\Phi_{\Omega_{k-1}}$ 。于是，可以得到一个向量表达式

$$\mathbf{w}_{\Omega_{k-1}} = (\mathbf{1}_{\Omega_{k-1}}^T (\Phi_{\Omega_{k-1}}^T \Phi_{\Omega_{k-1}})^{-1} \mathbf{1}_{\Omega_{k-1}})^{-1/2} (\Phi_{\Omega_{k-1}}^T \Phi_{\Omega_{k-1}})^{-1} \mathbf{1}_{\Omega_{k-1}} \quad (6.9.11)$$

$\Phi_{\Omega_{k-1}} \mathbf{w}_{\Omega_{k-1}}$  就是 LARS 算法在当前回归变量集  $\Omega_{k-1}$  下的求解路径，而  $\mathbf{w}_{\Omega_{k-1}}$  则是  $\mathbf{x}$  的继续搜索的路径。

为了搜索  $\mathbf{x}$ ，Efron 等人定义了一个向量  $\hat{\mathbf{d}}$ ，其元素为  $s_i w_i, i \in \Omega_{k-1}$ ，其中  $w_i$  是向量  $\mathbf{w}_{\Omega_{k-1}}$  的第  $i$  个元素，而  $s_i$  则是入选变量向量  $\phi_i$  与当前残差  $\mathbf{y} - \hat{\mathbf{y}}_{k-1}$  的相关系数的符号，也就是  $\hat{x}_i$  的符号。那些没有入选的变量对应在  $\hat{\mathbf{d}}$  中的元素为 0，即有

$$x_j(\gamma) = \hat{x}_j + \gamma \hat{d}_j$$

很显然， $x_j(\gamma)$  会在  $\gamma_j = -\hat{x}_j / \hat{d}_j$  处变号。对于业已得到的 LASSO 估计  $\mathbf{x}(\gamma)$ ，其中的元素会在某个大于 0 的最小  $\gamma_j$  处变号，将这个最小  $\gamma_j$  记作  $\tilde{\gamma}$ 。如果没有  $\gamma_j$  大于 0，则记  $\tilde{\gamma}$  为无穷大。根据这一观察，可以对 LARS 算法实施 LASSO 修正。

具有 LASSO 修正的 LARS 算法的核心是“一步一个”(one at a time)，即每一步迭代都要增加或删掉一个回归变量。具体做法是：在现有回归变量集和当前残差的基础上，会有一条求解路径，在此路径上前进的最大步记为  $\hat{\gamma}$ ，而找到一个新变量的最大步记为  $\tilde{\gamma}$ 。

如果  $\tilde{\gamma} < \hat{\gamma}$ , 则对于 LARS 估计的那个新变量  $x_j(\gamma)$  便不会成为一个 LASSO 估计, 应该将这个变量从回归变量集中删去; 反之, 若  $\tilde{\gamma} > \hat{\gamma}$ , 则对于 LARS 估计的那个  $x_j(\gamma)$  应该成为一个新 LASSO 估计, 此时需要将此变量加入回归变量集中。去掉一个变量或者增加一个变量后, 都需要停止在原求解路径上前进, 通过重新计算当前残差和当前这些新变量集之间的相关系数, 确定出一条新的求解路径, 并继续进行“一步一个”的 LARS 迭代步骤。如此重复, 即可通过 LARS 算法得到所有的 LASSO 估计。

下面是具有 LASSO 修正的 LARS 算法的具体步骤<sup>[154]</sup>。

### 算法 6.9.3 具有 LASSO 修正的 LARS 算法

输入 观测数据向量  $y \in \mathbb{R}^m$  和字典矩阵  $\Phi \in \mathbb{R}^{m \times n}$ 。

输出 系数向量  $x \in \mathbb{R}^n$ 。

初始化 标签集  $\Omega_0 = \emptyset$ , 初始拟合观测向量  $\hat{y} = 0$ , 字典矩阵  $\Phi_{\Omega_0} = \Phi$ 。

对  $k = 1, 2, \dots$ , 执行以下运算。

步骤 1 计算相关向量

$$\hat{c}_k = \Phi_{\Omega_{k-1}}^T (y - \hat{y}_{k-1}) \quad (6.9.12)$$

步骤 2 记最大相关系数  $C = \max\{|\hat{c}_k(1)|, \dots, |\hat{c}_k(n)|\}$ , 更新标签集

$$\Omega_k = \Omega_{k-1} \cup \{j^{(k)} \mid |\hat{c}_k(j)| = C\} \quad (6.9.13)$$

步骤 3 将字典矩阵与标签集  $\Omega_k$  的所有元素对应的列向量组成矩阵  $\Phi_{\Omega_k} = [s_j \phi_j, j \in \Omega_k]$ , 其中  $s_j = \text{sgn}(\hat{c}_k(j))$  是相关系数  $\hat{c}_k(j)$  的符号函数。

步骤 4 求当前最小角度方向即角平分线方向  $\mu_k$

$$G_{\Omega_k} = \Phi_{\Omega_k}^T \Phi_{\Omega_k} \in \mathbb{R}^{k \times k} \quad (6.9.14)$$

$$\alpha_{\Omega_k} = (\mathbf{1}_k^T G_{\Omega_k} \mathbf{1}_k)^{-1/2} \quad (6.9.15)$$

$$w_{\Omega_k} = \alpha_{\Omega_k} G_{\Omega_k}^{-1} \mathbf{1}_k \in \mathbb{R}^k \quad (6.9.16)$$

$$\mu_k = \Phi_{\Omega_k} w_{\Omega_k} \in \mathbb{R}^k \quad (6.9.17)$$

步骤 5 使用最小二乘法估计系数向量

$$\hat{x}_k = (\Phi_{\Omega_k}^T \Phi_{\Omega_k})^{-1} \Phi_{\Omega_k}^T = G_{\Omega_k}^{-1} \Phi_{\Omega_k}^T \quad (6.9.18)$$

并计算

$$b = \Phi_{\Omega_k}^T \mu_k = [b_1, \dots, b_m]^T \quad (6.9.19)$$

步骤 6 计算

$$\hat{\gamma} = \min_{j \in \Omega_k^c} + \left\{ \frac{C - \hat{c}_k(j)}{\alpha_{\Omega_k} - b_j}, \frac{C + \hat{c}_k(j)}{\alpha_{\Omega_k} + b_j} \right\} \quad (6.9.20)$$

$$\tilde{\gamma} = \min_{j \in \Omega_k} + \left\{ -\frac{x_j}{w_j} \right\} \quad (6.9.21)$$

式中  $w_j$  是向量  $\mathbf{w}_{\Omega_k} = [w_1, \dots, w_n]^T$  的第  $j$  个元素，而  $\min^+ \{\cdot\}$  表示只取括号中正的最小项。若无正的项存在，则  $\min^+$  取无穷大。

**步骤 7** 若  $\tilde{\gamma} < \hat{\gamma}$ ，则拟合向量和标签集分别修正为

$$\hat{\mathbf{y}}_k = \hat{\mathbf{y}}_{k-1} + \tilde{\gamma} \boldsymbol{\mu}_k \quad \text{和} \quad \Omega_k = \Omega_{k-1} - \{\tilde{j}\} \quad (6.9.22)$$

式中去除的一个候选变量的下标  $\tilde{j}$  是式 (6.9.21) 取最小值所对应的变量下标  $j \in \Omega_k$ 。反之，若  $\hat{\gamma} < \tilde{\gamma}$ ，则拟合向量和标签集分别修正为

$$\hat{\mathbf{y}}_k = \hat{\mathbf{y}}_{k-1} + \hat{\gamma} \boldsymbol{\mu}_k \quad \text{和} \quad \Omega_k = \Omega_{k-1} \cup \{\hat{j}\} \quad (6.9.23)$$

其中，增加的一个候选变量的指标  $\hat{j}$  是式 (6.9.20) 取最小值所对应的变量下标  $j \in \Omega_k$ 。

**步骤 8**  $k \leftarrow k + 1$ ，并重复步骤 1 ~ 步骤 7，直至算法满足某个停止准则。

**步骤 9** 输出  $\hat{\mathbf{x}}_k$ 。

SparseLab toolbox (<http://www.sparselab.stanford.edu>) 提供了 SolveLasso 函数和 SolveOMP 函数。

### 6.9.3 同伦算法

在拓扑中，同伦的概念描述两个对象间的“连续变化”。同伦算法 (homotopy algorithm) 是一种从一个简单解开始，通过迭代计算，变化到所希望的复杂解的搜索算法。因此，同伦算法的关键是初始简单解的确定。

考虑  $L_1$  范数最小化问题  $(P_1)$  和无约束  $L_2$  最小化问题  $(P_{12})$  之间的关系，假定对每一个最小化问题  $(P_{12}) : \lambda \in [0, \infty)$ ，有一个相应的唯一解  $\mathbf{x}_\lambda$ 。于是，集合  $\{\mathbf{x}_\lambda | \lambda \in [0, \infty)\}$  便确定一个求解路径，并且对于足够大的  $\lambda$  值有  $\mathbf{x}_\lambda = \mathbf{0}$ ，而当  $\lambda \rightarrow 0$  时， $(P_{12})$  的解  $\tilde{\mathbf{x}}_\lambda$  收敛为  $L_1$  范数最小化问题  $(P_1)$  的解。因此， $\mathbf{x}_\lambda = \mathbf{0}$  就是求解最小化问题  $(P_1)$  的同伦算法的初始解。

求解无约束  $L_2$  范数最小化问题  $(P_{12})$  的同伦算法从初始值  $\mathbf{x}_0 = \mathbf{0}$  开始，以一种迭代的方式运行，计算  $k = 1, 2, \dots$  各步的解  $\mathbf{x}_k$ 。在整个运算中，保持作用集

$$I = \{j | |c_k(j)| = \|c_k\|_\infty = \lambda\} \quad (6.9.24)$$

不变。

下面是求解  $L_1$  范数最小化问题的同伦算法<sup>[144]</sup>。

#### 算法 6.9.4 同伦算法

输入 观测向量  $\mathbf{y} \in \mathbb{R}^m$ ，字典矩阵  $\Phi$ ，参数  $\lambda$ 。

初始化  $\mathbf{x}_0 = \mathbf{0}$ ,  $c_0 = \Phi^T \mathbf{y}$ 。

迭代  $k = 1, 2, \dots$

**步骤 1** 用式 (6.9.24) 构造作用集  $I$ ，组成支撑区的残差相关向量  $\mathbf{c}_k(I) = [c_k(i), i \in I]$  和字典矩阵  $\Phi_I = [\phi_i, i \in I]$ 。

步骤 2 计算残差相关向量  $\mathbf{c}_k(I) = \Phi_I^T(\mathbf{y} - \Phi_I \mathbf{x}_k)$ 。

步骤 3 通过求解方程

$$\Phi_I^T \Phi_I \mathbf{d}_k(I) = \text{sgn}(\mathbf{c}_k(I)) \quad (6.9.25)$$

得到更新方向向量  $\mathbf{d}_k(I)$ 。

步骤 4 计算

$$\gamma_k^+ = \min_{i \in I^c} + \left\{ \frac{\lambda - c_k(i)}{1 - \phi_i^T \mathbf{v}_k}, \frac{\lambda + c_k(i)}{1 + \phi_i^T \mathbf{v}_k} \right\} \quad (6.9.26)$$

$$\gamma_k^- = \min_{i \in I} \{-x_k(i)/d_k(i)\} \quad (6.9.27)$$

步骤 5 确定断点 (breakpoint)

$$\gamma_k = \min\{\gamma_k^+, \gamma_k^-\} \quad (6.9.28)$$

步骤 6 更新解向量

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \gamma_k \mathbf{d}_k \quad (6.9.29)$$

步骤 7 若  $\|\mathbf{x}_k\|_\infty = 0$ , 则算法停止, 并输出稀疏向量结果  $\mathbf{x}_k$ ; 否则, 返回步骤 1, 并继续以上迭代。

随着  $\lambda$  的减小,  $(P_{11})$  的目标函数将经历一个从  $L_2$  范数约束到  $L_1$  范数目标函数的同伦过程。这就是同伦算法可以求解  $L_1$  优化问题  $(P_{11})$  的原理所在。

业已证明<sup>[154, 383]</sup>, 同伦算法是求解  $L_1$  最小化问题  $(P_1)$  的一种正确解法。

#### 6.9.4 Bregman 迭代算法

前面讨论的求解矩阵方程  $\mathbf{A}\mathbf{u} = \mathbf{b}$  的稀疏优化模型可以归纳为:

(1)  $L_0$  极小化模型  $\min_{\mathbf{u}} \|\mathbf{u}\|_0$  subject to  $\mathbf{A}\mathbf{u} = \mathbf{b}$ ;

(2) 基追踪 (BP)/压缩感知模型  $\min_{\mathbf{u}} \|\mathbf{u}\|_1$  subject to  $\mathbf{A}\mathbf{u} = \mathbf{b}$ ;

(3) 基追踪去噪 (basis-pursuit de-noising) 模型  $\min_{\mathbf{u}} \|\mathbf{u}\|_1$  subject to  $\|\mathbf{A}\mathbf{u} - \mathbf{b}\|_2 < \epsilon$ ;

(4) LASSO 模型  $\min_{\mathbf{u}} \|\mathbf{A}\mathbf{u} - \mathbf{b}\|_2$  subject to  $\|\mathbf{u}\|_1 \leq s$ 。

其中, 基追踪是  $L_1$  极小化的松弛形式, LASSO 是基追踪去噪的等价线性预测表示。

下面讨论以上优化模型的一般形式

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} J(\mathbf{u}) + \lambda H(\mathbf{u}) \quad (6.9.30)$$

其中  $J : X \rightarrow \mathbb{R}$  和  $H : X \rightarrow \mathbb{R}$  均为非负的凸函数 ( $X$  为闭凸集), 但  $J$  为非平滑函数, 而  $H$  可微分。

形象地讲, 一个函数被称作有界变差函数 (function of bounded variation), 若其图形的振荡 (即摆动、变差) 在一个特定的区间内一定程度上是可控的 (manageable) 或温顺的 (tame)。在数学分析中, 有界变差函数就是其变差有界的实函数。

向量  $\mathbf{u}$  的有界变差范数 (bounde-variation norm, BV norm) 记作  $\|\mathbf{u}\|_{\text{BV}}$ , 定义为<sup>[10]</sup>

$$\|\mathbf{u}\|_{\text{BV}} = \|\mathbf{u}\|_1 + J_0(\mathbf{u}) = \|\mathbf{u}\|_1 + \int_{\Omega} |\nabla \mathbf{u}| dx \quad (6.9.31)$$

其中  $J_0(\mathbf{u})$  表示  $\mathbf{u}$  的全变分 (total variation, 即总变差)。

若令  $J(\mathbf{u}) = \|\mathbf{u}\|_{\text{BV}}$  和  $H(\mathbf{u}) = \frac{1}{2}\|\mathbf{u} - \mathbf{f}\|_2^2$ , 则优化模型式 (6.9.30) 变成全变分/Rudin-Osher-Fatemi (ROF) 去噪模型<sup>[432]</sup>

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} \|\mathbf{u}\|_{\text{BV}} + \frac{\lambda}{2} \|\mathbf{u} - \mathbf{f}\|_2^2 \quad (6.9.32)$$

求解优化问题式 (6.9.30) 的一种著名迭代方法为 Bregman 迭代, 它的主要数学工具是 Bregman 距离<sup>[60]</sup>。

**定义 6.9.1** 令  $J(\mathbf{u})$  为凸函数, 向量  $\mathbf{u}, \mathbf{v} \in X$ , 且  $\mathbf{g} \in \partial J(\mathbf{v})$  是函数  $J$  在点  $\mathbf{v}$  的次梯度向量。点  $\mathbf{u}$  和  $\mathbf{v}$  之间的 Bregman 距离记作  $D_J^g(\mathbf{u}, \mathbf{v})$ , 并定义为

$$D_J^g(\mathbf{u}, \mathbf{v}) = J(\mathbf{u}) - J(\mathbf{v}) - \langle \mathbf{g}, \mathbf{u} - \mathbf{v} \rangle \quad (6.9.33)$$

Bregman 距离不是传统意义下的一种距离, 因为  $D_J^g(\mathbf{u}, \mathbf{v}) \neq D_J^g(\mathbf{v}, \mathbf{u})$ 。然而, Bregman 距离却具有几个很好的性质, 这使得它成为求解  $L_1$  正则化优化问题的一种有效工具。

**性质 1** 对于所有  $\mathbf{u}, \mathbf{v} \in X$  和  $\mathbf{g} \in \partial J(\mathbf{v})$  而言, Bregman 距离是非负的, 即有  $D_J^g(\mathbf{u}, \mathbf{v}) \geq 0$ 。

**性质 2** 相同两点之间的 Bregman 距离为零  $D_J^g(\mathbf{v}, \mathbf{v}) = 0$ 。

**性质 3** Bregman 距离可度量两个点  $\mathbf{u}$  和  $\mathbf{v}$  之间的接近度 (closeness), 因为对于连接  $\mathbf{u}$  和  $\mathbf{v}$  的直线段内的任何点  $\mathbf{w}$  而言, 均有  $D_J^g(\mathbf{u}, \mathbf{v}) \geq D_J^g(\mathbf{w}, \mathbf{v})$ 。

考虑非平滑函数  $J(\mathbf{u})$  在第  $k$  次迭代点  $\mathbf{u}^k$  的一阶 Taylor 级数逼近  $J(\mathbf{u}) = J(\mathbf{u}^k) + \langle \mathbf{g}^k, \mathbf{u} - \mathbf{u}^k \rangle$ 。逼近误差由 Bregman 距离

$$D_J^{\mathbf{g}^k}(\mathbf{u}, \mathbf{u}^k) = J(\mathbf{u}) - J(\mathbf{u}^k) - \langle \mathbf{g}^k, \mathbf{u} - \mathbf{u}^k \rangle \quad (6.9.34)$$

度量。

Bergman 于 1965 年提出, 无约束优化问题式 (6.9.30) 可以修正为<sup>[60]</sup>

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} D_J^{\mathbf{g}^k}(\mathbf{u}) + \lambda H(\mathbf{u}) \quad (6.9.35)$$

$$= \arg \min_{\mathbf{u}} J(\mathbf{u}) - \langle \mathbf{g}^k, \mathbf{u} - \mathbf{u}^k \rangle + \lambda H(\mathbf{u}) \quad (6.9.36)$$

这就是著名的 Bregman 迭代。

下面介绍 Bregman 迭代的具体算法及其推广。

### 1. Bregman 迭代算法

记 Bregman 迭代优化问题的目标函数  $L(\mathbf{u}) = J(\mathbf{u}) - \langle \mathbf{g}^k, \mathbf{u} - \mathbf{u}^k \rangle + \lambda H(\mathbf{u})$ 。由平稳点条件  $\mathbf{0} \in \partial L(\mathbf{u})$  得  $\mathbf{0} \in \partial J(\mathbf{u}) - \mathbf{g}^k + \lambda \nabla H(\mathbf{u})$ 。因此, 在第  $k+1$  次迭代点  $\mathbf{u}^{k+1}$ , 有

$$\mathbf{g}^{k+1} = \mathbf{g}^k - \lambda \nabla H(\mathbf{u}^{k+1}), \quad \mathbf{g}^{k+1} \in \partial J(\mathbf{u}^{k+1}) \quad (6.9.37)$$

式(6.9.36)和式(6.9.37)一起组成了Bregman迭代算法(Bregman iterative algorithm),它是Osher等人<sup>[384]</sup>于2005年针对图像处理提出的。

#### 算法 6.9.5 Bregman 迭代算法

初始化  $k = 0, \mathbf{u}^0 = \mathbf{0}, \mathbf{g}^0 = \mathbf{0}$ 。

迭代

$$\begin{aligned}\mathbf{u}^{k+1} &= \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + \lambda H(\mathbf{u}) \\ \mathbf{g}^{k+1} &= \mathbf{g}^k - \lambda \nabla H(\mathbf{u}^{k+1}) \in \partial J(\mathbf{u}^{k+1})\end{aligned}$$

若  $\mathbf{u}^k$  未收敛, 则令  $k = k + 1$ , 返回迭代。

由于  $\mathbf{u}^1 = \arg \min_{\mathbf{u}} J(\mathbf{u}) + H(\mathbf{u})$ , 所以第1步迭代求解的优化问题为原始问题。从第2步迭代开始, 执行的是基于Bregman距离的Bregman迭代。

文献[384]证明了上述Bergman迭代算法的收敛性能。

**定理 6.9.1** 假定  $J$  和  $H$  都是凸函数, 并且  $H$  可微分。若式(6.9.36)的解存在, 则下列收敛结果为真:

- (1) 函数  $H$  在迭代过程中是单调下降的, 即  $H(\mathbf{u}^{k+1}) \leq H(\mathbf{u}^k)$ 。
- (2) 函数  $H$  将收敛为最优解  $H(\mathbf{u}^*)$ , 因为  $H(\mathbf{u}^k) \leq H(\mathbf{u}^*) + J(\mathbf{u}^*)/k$ 。

Bregman迭代算法有两种常用版本<sup>[530]</sup>:

版本1:  $k = 0, \mathbf{u}^0 = \mathbf{0}, \mathbf{g}^0 = \mathbf{0}$ 。

迭代

$$\begin{aligned}\mathbf{u}^{k+1} &= \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + \frac{1}{2} \|\mathbf{A}\mathbf{u} - \mathbf{b}\|_2^2 \\ \mathbf{g}^{k+1} &= \mathbf{g}^k - \mathbf{A}^T(\mathbf{A}\mathbf{u}^{k+1} - \mathbf{b})\end{aligned}$$

若  $\mathbf{u}^k$  未收敛, 则令  $k \leftarrow k + 1$ , 并返回迭代。

版本2:  $k = 0, \mathbf{b}^0 = \mathbf{0}, \mathbf{u}^0 = \mathbf{0}$ 。

迭代

$$\begin{aligned}\mathbf{b}^{k+1} &= \mathbf{b} + (\mathbf{b}^k - \mathbf{A}\mathbf{u}^k) \\ \mathbf{u}^{k+1} &= \arg \min_{\mathbf{u}} J(\mathbf{u}) + \frac{1}{2} \|\mathbf{A}\mathbf{u} - \mathbf{b}^{k+1}\|_2^2\end{aligned}$$

若  $\mathbf{u}^k$  未收敛, 则令  $k \leftarrow k + 1$ , 并返回迭代。

文献[530]证明了以上两种版本是等价的。

#### 2. 线性化 Bregman 迭代算法

Bregman迭代算法提供了优化问题的一种有效工具, 但是由于每一步都需要进行目标函数  $D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + H(\mathbf{u})$  的最小化, 所以运算比较费时。为了提高Bregman迭代算法的计算效率, Yin等人<sup>[530]</sup>于2008年提出了线性化Bregman迭代算法。

线性化Bregman迭代的基本思想是: 在Bregman迭代的基础上, 再使用一阶Taylor级数展开将非线性函数  $H(\mathbf{u})$  在点  $\mathbf{u}^k$  线性化为  $H(\mathbf{u}) = H(\mathbf{u}^k) + \langle \nabla H(\mathbf{u}^k), \mathbf{u} - \mathbf{u}^k \rangle$ 。于

是，具有  $\lambda = 1$  的优化问题式 (6.9.30) 变为

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + H(\mathbf{u}^k) + \langle \nabla H(\mathbf{u}^k), \mathbf{u} - \mathbf{u}^k \rangle$$

注意到一阶 Taylor 级数展开只是对  $\mathbf{u}$  位于点  $\mathbf{u}^k$  的邻域时才精确，并且相对于  $\mathbf{u}$  的优化而言， $H(\mathbf{u}^k)$  作为相加的常数项，可以省去，故上述优化问题的更精确的表达形式是

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + \langle \nabla H(\mathbf{u}^k), \mathbf{u} - \mathbf{u}^k \rangle + \frac{1}{2\delta} \|\mathbf{u} - \mathbf{u}^k\|_2^2 \quad (6.9.38)$$

重要的是，上式又可等价写作

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + \frac{1}{2\delta} \|\mathbf{u} - (\mathbf{u}^k - \delta \nabla H(\mathbf{u}^k))\|_2^2 \quad (6.9.39)$$

因为式 (6.9.38) 与式 (6.9.39) 只是相差一个与  $\mathbf{u}$  无关的常数项。

特别地，若  $H(\mathbf{u}) = \frac{1}{2} \|\mathbf{A}\mathbf{u} - \mathbf{b}\|_2^2$ ，则由  $\nabla H(\mathbf{u}) = \mathbf{A}^T(\mathbf{A}\mathbf{u} - \mathbf{b})$ ，式 (6.9.39) 可写作

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} D_J^{g^k}(\mathbf{u}, \mathbf{u}^k) + \frac{1}{2\delta} \left\| \mathbf{u} - \left( \mathbf{u}^k - \delta \mathbf{A}^T(\mathbf{A}\mathbf{u}^k - \mathbf{b}) \right) \right\|_2^2 \quad (6.9.40)$$

考察式 (6.9.40) 的目标函数

$$L(\mathbf{u}) = J(\mathbf{u}) - J(\mathbf{u}^k) - \langle \mathbf{g}^k, \mathbf{u} - \mathbf{u}^k \rangle + \frac{1}{2\delta} \left\| \mathbf{u} - \left( \mathbf{u}^k - \delta \mathbf{A}^T(\mathbf{A}\mathbf{u}^k - \mathbf{b}) \right) \right\|_2^2$$

由平稳点的次微分条件  $\mathbf{0} \in \partial L(\mathbf{u})$  知

$$\mathbf{0} \in \partial J(\mathbf{u}) - \mathbf{g}^k + \frac{1}{\delta} \left[ \mathbf{u} - \left( \mathbf{u}^k - \delta \mathbf{A}^T(\mathbf{A}\mathbf{u}^k - \mathbf{b}) \right) \right]$$

记  $\mathbf{g}^{k+1} \in \partial J(\mathbf{u}^{k+1})$ ，则由上式有<sup>[530]</sup>

$$\mathbf{g}^{k+1} = \mathbf{g}^k - \mathbf{A}^T(\mathbf{A}\mathbf{u}^k - \mathbf{b}) - \frac{(\mathbf{u}^{k+1} - \mathbf{u}^k)}{\delta} = \dots = \sum_{i=1}^k \mathbf{A}^T(\mathbf{b} - \mathbf{A}\mathbf{u}^i) - \frac{\mathbf{u}^{k+1}}{\delta} \quad (6.9.41)$$

若令

$$\mathbf{v}^k = \sum_{i=1}^k \mathbf{A}^T(\mathbf{b} - \mathbf{A}\mathbf{u}^i) \quad (6.9.42)$$

则可以得到两个重要的迭代公式。

首先，由式 (6.9.41) 和式 (6.9.42) 得变元  $\mathbf{u}$  第  $k$  次迭代的更新公式

$$\mathbf{u}^{k+1} = \delta(\mathbf{v}^k - \mathbf{g}^{k+1}) \quad (6.9.43)$$

其次，由式 (6.9.42) 直接得到中间变元  $\mathbf{v}^k$  的迭代公式

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \mathbf{A}^T(\mathbf{b} - \mathbf{A}\mathbf{u}^{k+1}) \quad (6.9.44)$$

式 (6.9.43) 和式 (6.9.44) 一起组成了求解优化问题

$$\min_{\mathbf{u}} J(\mathbf{u}) + \frac{1}{2} \|\mathbf{A}\mathbf{u} - \mathbf{b}\|_2^2 \quad (6.9.45)$$

的线性化 Bregman 迭代算法 (linearized Bregman iterative algorithm) [530]。

如果限定  $J(\mathbf{u}) = \mu \|\mathbf{u}\|_1$ , 则由于

$$\delta(\|\mathbf{u}\|_1)_i = \begin{cases} \{+1\}, & \text{若 } u_i > 0 \\ [-1, +1], & \text{若 } u_i = 0 \\ \{-1\}, & \text{若 } u_i < 0 \end{cases} \quad (6.9.46)$$

所以式 (6.9.43) 可写作分量形式

$$u_i^{k+1} = \delta(v_i^k - g_i^{k+1}) = \delta \cdot \text{shrink}(v_i^k, \mu), \quad i = 1, \dots, n \quad (6.9.47)$$

式中

$$\text{shrink}(y, \alpha) = \text{sgn}(y) \max\{|y| - \alpha, 0\} = \begin{cases} y - \alpha, & y \in (\alpha, \infty) \\ 0, & y \in [-\alpha, \alpha] \\ y + \alpha, & y \in (-\infty, -\alpha) \end{cases}$$

为收缩算子。

以上结果可以总结为求解基追踪去噪/全变分去噪的线性化 Bregman 迭代算法 [530]。

#### 算法 6.9.6 基追踪/压缩感知的线性化 Bregman 迭代算法

初始化  $k = 0, \mathbf{u}^0 = \mathbf{0}, \mathbf{v}^0 = \mathbf{0}$ 。

迭代

$$\begin{aligned} u_i^{k+1} &= \delta \cdot \text{shrink}(v_i^k, \mu), \quad i = 1, \dots, n \\ v^{k+1} &= \mathbf{v}^k + \mathbf{A}^T(\mathbf{b} - \mathbf{A}\mathbf{u}^{k+1}) \end{aligned}$$

若  $\mathbf{u}^k$  未收敛, 则令  $k = k + 1$ , 返回迭代。

### 3. 分割 Bregman 算法

考虑  $J(\mathbf{u}) = \|\Phi(\mathbf{u})\|_1$  时的优化问题

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} \|\Phi(\mathbf{u})\|_1 + H(\mathbf{u}) \quad (6.9.48)$$

引入中间变量  $\mathbf{z} = \Phi(\mathbf{u})$ , 则无约束优化问题式 (6.9.30) 可写成约束优化问题

$$(\mathbf{u}^{k+1}, \mathbf{z}^{k+1}) = \arg \min_{\mathbf{u}, \mathbf{z}} \|\mathbf{z}\|_1 + H(\mathbf{u}) \quad \text{subject to } \mathbf{z} = \Phi(\mathbf{u}) \quad (6.9.49)$$

增加一个  $L_2$  惩罚项, 即可将这一约束优化问题变成无约束优化问题

$$(\mathbf{u}^{k+1}, \mathbf{z}^{k+1}) = \arg \min_{\mathbf{u}, \mathbf{z}} \|\mathbf{z}\|_1 + H(\mathbf{u}) + \frac{\lambda}{2} \|\mathbf{z} - \Phi(\mathbf{u})\|_2^2 \quad (6.9.50)$$

Goldstein 与 Osher<sup>[192]</sup> 已经证明: 看似比较复杂的 Bergman 迭代

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg \min_{\mathbf{x}} D_E^g(\mathbf{x}, \mathbf{x}^k) + \frac{\lambda}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2 \\ &= \arg \min_{\mathbf{x}} E(\mathbf{x}) - \langle \mathbf{g}^k, \mathbf{x} - \mathbf{x}^k \rangle + \frac{\lambda}{2} \|\mathbf{Ax} - \mathbf{b}\|_2^2 \end{aligned} \quad (6.9.51)$$

$$\mathbf{g}^{k+1} = \mathbf{g}^k - \lambda \mathbf{A}^T(\mathbf{Ax}^{k+1} - \mathbf{b}) \quad (6.9.52)$$

等价于简化的 Bergman 迭代

$$\mathbf{x}^{k+1} = \arg \min_{\mathbf{x}} E(\mathbf{x}) + \frac{\lambda}{2} \|\mathbf{Ax} - \mathbf{b}^k\|_2^2 \quad (6.9.53)$$

$$\mathbf{b}^{k+1} = \mathbf{b}^k + \mathbf{b} - \mathbf{Ax}^k \quad (6.9.54)$$

将这一等价关系应用于无约束优化问题式 (6.9.50)，即得到下列分割 Bergman 迭代算法 (split Bregman iterative algorithm)<sup>[192]</sup>

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} H(\mathbf{u}) + \frac{\lambda}{2} \|\mathbf{z}^k - \Phi(\mathbf{u}) - \mathbf{b}^k\|_2^2 \quad (6.9.55)$$

$$\mathbf{z}^{k+1} = \arg \min_{\mathbf{z}} \|\mathbf{z}\|_1 + \frac{\lambda}{2} \|\mathbf{z} - \Phi(\mathbf{u}^{k+1}) - \mathbf{b}^k\|_2^2 \quad (6.9.56)$$

$$\mathbf{b}^{k+1} = \mathbf{b}^k + [\Phi(\mathbf{u}^{k+1}) - \mathbf{z}^{k+1}] \quad (6.9.57)$$

分割 Bregman 迭代算法的三个迭代具有以下特点：

- (1) 第 1 个迭代是一个可微分的优化问题，可以使用 Gauss-Seidel 方法求解。
- (2) 第 2 个迭代可以使用收缩算子有效求解。
- (3) 第 3 个迭代为显式计算。

## 本章小结

本章集中讨论了超定矩阵方程、盲矩阵方程以及欠定稀疏矩阵方程求解的线性代数方法。

超定矩阵方程求解的主要方法有：① Tikhonov 正则化，可有效防止矩阵  $\mathbf{A}$  秩亏缺对解向量的影响。② 正则 Gauss-Seidel 法，基本思想是对多个变元向量进行解耦。主要有分块协同下降法和交替最小二乘法。③ 总体最小二乘法（同时考虑数据矩阵  $\mathbf{A}$  和数据向量  $\mathbf{b}$  的独立高斯白噪声）。④ 约束总体最小二乘（克服  $\mathbf{A}$  的误差矩阵的列向量之间的相关性）。

盲矩阵方程求解：主要介绍了子空间方法和非负矩阵分解的典型算法——乘法算法、投影梯度法、Nesterov 最优梯度法、交替非负最小二乘算法和非负稀疏矩阵方程求解。

稀疏矩阵方程求解：首先讨论了  $L_0$  范数优化的松弛及正则化理论，然后重点介绍了稀疏优化的正交匹配追踪法、LASSO 算法与 LARS 算法、同伦算法、Bregman 迭代（包括线性化 Bergman 迭代和分割 Bergman 迭代算法）。

围绕矩阵方程求解的应用，主要介绍了总体最小二乘拟合、超分辨谐波恢复、正则化约束总体最小二乘图像恢复、非负矩阵分解在模式识别中的应用、基追踪去噪、压缩感知和全变分去噪等。

## 习 题

**6.1** 考虑线性方程  $\mathbf{A}\mathbf{x} + \boldsymbol{\epsilon} = \mathbf{x}$ , 其中,  $\boldsymbol{\epsilon}$  为加性有色噪声向量, 满足条件  $E\{\boldsymbol{\epsilon}\} = \mathbf{0}$  和  $E\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T\} = \mathbf{R}$ 。令  $\mathbf{R}$  已知, 并使用加权误差函数  $Q(\mathbf{x}) = \boldsymbol{\epsilon}^T \mathbf{W} \boldsymbol{\epsilon}$  作为求参数向量  $\mathbf{x}$  最优估计  $\hat{\mathbf{x}}_{\text{WLS}}$  的代价函数。这种方法称为加权最小二乘方法。证明

$$\hat{\mathbf{x}}_{\text{WLS}} = (\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \mathbf{x}$$

其中, 加权矩阵  $\mathbf{W}$  的最优选择为  $\mathbf{W}_{\text{opt}} = \mathbf{R}^{-1}$ 。

**6.2** 已知超定的线性方程  $\mathbf{Z}_t^T \mathbf{X}_t = \mathbf{Z}_t^T \mathbf{Y}_t \mathbf{x}$ , 其中,  $\mathbf{Z}_t \in \mathbb{R}^{(t+1) \times K}$  称为辅助变量矩阵, 并且  $t+1 > K$ 。

(1) 令参数向量  $\mathbf{x}$  在  $t$  时刻的估计为  $\hat{\mathbf{x}}$ , 求其表达式。这一方法称为辅助变量方法 (instrumental variable method)。

(2) 令

$$\mathbf{Y}_{t+1} = \begin{bmatrix} \mathbf{Y}_t \\ \mathbf{y}_{t+1} \end{bmatrix}, \quad \mathbf{Z}_{t+1} = \begin{bmatrix} \mathbf{Z}_t \\ \mathbf{z}_{t+1} \end{bmatrix}, \quad \mathbf{X}_{t+1} = \begin{bmatrix} \mathbf{X}_t \\ \mathbf{x}_{t+1} \end{bmatrix}$$

求  $\mathbf{x}_{t+1}$  的递推计算公式。

**6.3** [540] 给定  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{b} \in \mathbb{R}^m$ ,  $\mathbf{C} \in \mathbb{R}^{p \times n}$ ,  $\mathbf{d} \in \mathbb{R}^p$ , 并且  $\tau$  是一个大于零的数。现在希望求解带有二次约束的最小二乘问题

$$\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2, \quad \mathbf{x} \in S(\tau)$$

其中,  $S(\tau)$  是一个向量集合, 定义为

$$S(\tau) = \{\mathbf{x} \mid \|\mathbf{C}\mathbf{x} - \mathbf{d}\|_2 \leq \tau\}$$

(1) 证明: 若  $\|(I - \mathbf{C}\mathbf{C}^\dagger)\mathbf{d}\|_2 > \tau$ , 则上述两式所表述的二次约束最小二乘问题无解。

(2) 二次约束最小二乘问题存在显式解, 当且仅当存在  $\mathbf{z} \in \mathbb{R}^n$  使得

$$\|\mathbf{C}[\mathbf{A}^\dagger \mathbf{b} + (I - \mathbf{A}^\dagger \mathbf{A})\mathbf{z}] - \mathbf{d}\|_2 \leq \tau$$

成立, 并且对应的显式解由  $\mathbf{x} = \mathbf{A}^\dagger + (I - \mathbf{A}^\dagger \mathbf{A})\mathbf{z}$  给出。(提示: 无约束最小二乘问题  $\min \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$  的通解为  $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b} + \text{Null}(\mathbf{A}) = \mathbf{A}^\dagger \mathbf{b} + \text{Range}(I - \mathbf{A}^\dagger \mathbf{A})$ 。)

**6.4** 令  $\lambda > 0$ , 并且  $\mathbf{A}\mathbf{x} = \mathbf{b}$  为超定方程。证明: 反 Tikhonov 正则化优化问题

$$\min \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 - \frac{1}{2} \lambda \|\mathbf{x}\|_2^2$$

的最优解为

$$\mathbf{x} = (\mathbf{A}^H \mathbf{A} - \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b}$$

6.5 [198] 求解线性方程  $\mathbf{Ax} = \mathbf{b}$  的总体最小二乘问题也可以表示为

$$\min_{\mathbf{b}+\mathbf{e} \in \text{Range}(\mathbf{A}+\mathbf{E})} \|\mathbf{D}[\mathbf{E}, \mathbf{e}] \mathbf{T}\|_{\text{F}}, \quad \mathbf{E} \in \mathbb{R}^{m \times n}, \mathbf{e} \in \mathbb{R}^m$$

式中,  $\mathbf{D} = \text{diag}(d_1, \dots, d_m)$  和  $\mathbf{T} = \text{diag}(t_1, \dots, t_{n+1})$  非奇异。

(1) 证明: 若  $\text{rank}(\mathbf{A}) < n$ , 则上述总体最小二乘问题有一个解, 当且仅当  $\mathbf{b} \in \text{Range}(\mathbf{A})$ 。

(2) 证明: 若  $\text{rank}(\mathbf{A}) = n$ ,  $\mathbf{A}^T \mathbf{D}^2 \mathbf{b} = \mathbf{0}$ ,  $|t_{n+1}| \|\mathbf{D}\mathbf{b}\|_2 \geq \sigma_n(\mathbf{D}\mathbf{A}\mathbf{T}_1)$ ,  $\mathbf{T}_1 = \text{diag}(t_1, \dots, t_n)$ , 则总体最小二乘问题无解。其中,  $\sigma_n(\mathbf{C})$  表示矩阵  $\mathbf{C}$  的第  $n$  个奇异值。

6.6 考虑上题所述的总体最小二乘问题。证明: 若  $\mathbf{C} = \mathbf{D}[\mathbf{A}, \mathbf{b}] \mathbf{T} = [\mathbf{A}_1, \mathbf{d}]$ , 并且  $\sigma_n(\mathbf{C}) > \sigma_{n+1}(\mathbf{C})$ , 则总体最小二乘解满足  $(\mathbf{A}_1^T \mathbf{A}_1 - \sigma_{n+1}^2(\mathbf{C}) \mathbf{I}) \mathbf{x} = \mathbf{A}_1^T \mathbf{d}$ 。

6.7 已知数据点  $(1, 3), (3, 1), (5, 7), (4, 6), (7, 4)$ , 分别求总体最小二乘和一般最小二乘的拟合直线, 并分析它们的距离平方和。

6.8 考虑加性白噪声中的谐波恢复问题

$$x(n) = \sum_{i=1}^p A_i \sin(2\pi f_i n + \phi_i) + e(n)$$

其中,  $A_i, f_i, \phi_i$  分别是第  $i$  个谐波的幅值、频率和相位, 而  $e(n)$  为加性高斯白噪声。已知上述谐波过程服从特殊 ARMA 模型

$$x(n) + \sum_{i=1}^{2p} a_i x(n-i) = e(n) + \sum_{i=1}^{2p} a_i e(n-i), \quad n = 1, 2, \dots$$

和差分方程 (修正 Yule-Walker 方程)

$$R_x(k) + \sum_{i=1}^{2p} a_i R_x(k-i) = 0, \quad \forall k$$

并且谐波频率可以通过

$$f_i = \arctan[\text{Im}(z_i)/\text{Re}(z_i)]/2\pi, \quad i = 1, 2, \dots, p$$

恢复, 其中,  $z_i$  是特征多项式

$$A(z) = 1 + \sum_{i=1}^{2p} a_i z^{-i}$$

的共轭根对  $(z_i, z_i^*)$  的一个根。若

$$x(n) = \sqrt{20} \sin(2\pi 0.2n) + \sqrt{2} \sin(2\pi 0.213n) + e(n)$$

其中,  $e(n)$  是均值为 0, 方差为 1 的标准高斯白噪声, 并取  $n = 1, 2, \dots, 128$ 。试使用一般的最小二乘方法和奇异值-总体最小二乘(SVD-TLS) 算法分别估计观测数据的 ARMA

模型的 AR 参数  $a_i$ , 并估计谐波频率  $f_1$  和  $f_2$ 。假定差分方程个数取为 40, 使用最小二乘方法时分别取  $p = 2$  和  $p = 3$ , 而总体最小二乘算法取未知参数个数为 14, 通过有效奇异值个数的判断, 确定谐波个数, 然后计算特征多项式的根。从这一计算机仿真实验, 你能够得出最小二乘方法和总体最小二乘方法的某些比较结果吗?

### 6.9 通过令

$$\mathbf{X}_{:,j} = \mathbf{c}_1 \mathbf{a}_1^T b_{j1} + \cdots + \mathbf{c}_R \mathbf{a}_R^T b_{jR}$$

证明

$$\mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{:,1}, \dots, \mathbf{X}_{:,J}] = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$$

### 6.10 证明 KL 散度

$$D_{\text{KL}}(\mathbf{P} \parallel \mathbf{Q}) = \sum_{i=1}^I \sum_{j=1}^K \left( p_{ij} \log \frac{p_{ij}}{q_{ij}} - p_{ij} + q_{ij} \right)$$

的非负性, 并且 KL 散度等于零, 当且仅当  $\mathbf{P} = \mathbf{Q}$ 。

### 6.11 令 $D_E(\mathbf{X} \parallel \mathbf{AS}) = \frac{1}{2} \|\mathbf{X} - \mathbf{AS}\|_2^2$ , 证明

$$\nabla D_E(\mathbf{X} \parallel \mathbf{AS}) = \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial \mathbf{A}} = -(\mathbf{X} - \mathbf{AS})\mathbf{S}^T$$

$$\nabla D_E(\mathbf{X} \parallel \mathbf{AS}) = \frac{\partial D_E(\mathbf{X} \parallel \mathbf{AS})}{\partial \mathbf{S}} = -\mathbf{A}^T(\mathbf{X} - \mathbf{AS})$$

# 第7章 特征分析

对一个已知的量确定描述其特征的坐标系，称为特征分析 (eigenanalysis)。特征分析在数学和工程应用中都具有重要的实际意义。本章将围绕矩阵的特征分析，首先详细讨论矩阵的特征值分解。然后围绕特征值分解的以下推广分别展开专题介绍：矩阵束的广义特征值分解、Rayleigh 商、广义 Rayleigh 商、二次特征值问题以及多个矩阵的联合对角化。最后，将讨论特征分析与 Fourier 分析之间的联系。为了方便读者进一步理解这些理论，还将重点介绍有关典型应用。

## 7.1 特征值问题与特征方程

特征值问题既是一个理论上非常有意义的问题，同时又有着广泛的应用。

### 7.1.1 特征值问题

若对任意非零向量  $w$  恒有  $\mathcal{L}[w] = w$ ，则称  $\mathcal{L}$  为恒等变换 (identity transformation)。当一个线性算子作用于一向量时，如果仍然输出此向量，便称该线性算子具有输入重生 (input-reproducing) 特性。输入重生有两种情况：

- (1) 对任何非零输入向量，线性算子的输出向量都与输入向量完全相同 (恒等算子即属这种情况)。
- (2) 只是对某些特定的输入向量，线性算子的输出向量才与输入向量相同，并且还相差一个常数因子。

**定义 7.1.1** 若非零向量  $u$  作为线性算子  $\mathcal{L}$  的输入时，所产生的输出与输入相同 (顶多相差一个常数因子  $\lambda$ )，即

$$\mathcal{L}[u] = \lambda u, \quad u \neq 0 \tag{7.1.1}$$

则称向量  $u$  是线性算子  $\mathcal{L}$  的特征向量，称标量  $\lambda$  为线性算子  $\mathcal{L}$  的特征值。

工程应用中最常用的线性算子或线性变换当属线性时不变系统，其一连串的输入为向量，对应的输出也为向量形式。由上述定义知，若将每一个特征向量  $u$  视为线性时不变系统的输入，那么与每一个特征向量对应的特征值  $\lambda$  就相当于线性系统  $\mathcal{L}$  输入该特征向量时的增益。由于只有当特征向量  $u$  作线性系统  $\mathcal{L}$  的输入时，系统的输出才具有与输入相同 (除相差一个倍数因子外) 这一重要特征，所以特征向量 (eigenvector) 可以看作是表征系统特征的向量，其英文名又叫 characteristic vector。这就是从线性系统的观点，给出的特征向量的物理解释。

一个线性变换  $w = \mathcal{L}(x)$  若能够表示为  $w = Ax$ , 则称  $A$  是线性变换的标准矩阵 (standard matrix)。显然, 如果  $A$  是线性变换的标准矩阵, 则线性变换的特征值问题的表达式 (7.1.1) 可以写作

$$Au = \lambda u, \quad u \neq 0 \quad (7.1.2)$$

这样的标量  $\lambda$  称为矩阵  $A$  的特征值 (eigenvalue), 向量  $u$  称为与  $\lambda$  对应的特征向量 (eigenvector)。式 (7.1.2) 有时也被称为特征值–特征向量方程式。

由式 (7.1.2) 易知, 若  $A \in \mathbb{C}^{n \times n}$  为 Hermitian 矩阵, 则其特征值  $\lambda$  一定是实数, 并且有

$$A = U \Sigma U^H \quad (7.1.3)$$

式中,  $U = [u_1, \dots, u_n]^T$  和  $\Sigma = \text{diag}(\lambda_1, \dots, \lambda_n)$ 。式 (7.1.3) 称为 Hermitian 矩阵  $A$  的特征值分解。

由于特征值  $\lambda$  和特征向量  $u$  经常成对出现, 因此常将  $(\lambda, u)$  称为矩阵  $A$  的特征对 (eigenpair)。虽然特征值可以取零值, 但是特征向量不可以是零向量。

由上述分析有下列结论:

(1) 标量  $\lambda$  是线性变换  $\mathcal{L}$  的特征值, 当且仅当  $\lambda$  是该线性变换的标准矩阵  $A$  的特征值。

(2) 向量  $u$  是线性变换  $\mathcal{L}$  与特征值  $\lambda$  对应的特征向量, 当且仅当  $u$  是该线性变换的标准矩阵  $A$  与特征值  $\lambda$  的特征向量。

式 (7.1.2) 意味着, 使用矩阵  $A$  对向量  $u$  所作的线性变换  $Au$  不改变向量  $u$  的方向。因此, 线性变换  $Au$  是一种“保持方向不变”的映射。为了确定向量  $u$ , 不妨将式 (7.1.2) 改写作

$$(A - \lambda I)u = 0 \quad (7.1.4)$$

由于上式对任意向量  $u$  均应该成立, 故式 (7.1.4) 存在非零解  $u \neq 0$  的唯一条件是矩阵  $A - \lambda I$  的行列式等于零, 即

$$\det(A - \lambda I) = 0 \quad (7.1.5)$$

应当指出, 一个特征值不一定是唯一的, 有可能多个特征值取相同的值。同一特征值重复的次数称为特征值的多重度 (multiplicity)。例如,  $n \times n$  单位矩阵的  $n$  个特征值都等于 1, 其多重度为  $n$ 。

观察式 (7.1.5), 很容易直接得出下面的重要结果: 若特征值问题具有非零解  $x \neq 0$ , 则标量  $\lambda$  必然使  $n \times n$  矩阵  $A - \lambda I$  奇异。因此, 特征值问题的求解由以下两步组成:

- (1) 求出所有使矩阵  $A - \lambda I$  奇异的标量  $\lambda$  (特征值);
- (2) 给出一个使矩阵  $A - \lambda I$  奇异的特征值  $\lambda$ , 求出所有满足  $(A - \lambda I)x = 0$  的非零向量  $x$ , 它就是与  $\lambda$  对应的特征向量。

### 7.1.2 特征多项式

根据矩阵的奇偶性和行列式之间的关系知, 矩阵  $(A - \lambda I)$  是奇异矩阵, 当且仅当  $\det(A - \lambda I) = 0$ , 即

$$(A - \lambda I) \text{ 奇异} \iff \det(A - \lambda I) = 0 \quad (7.1.6)$$

因此, 矩阵  $(A - \lambda I)$  称为  $A$  的特征矩阵 (characteristic matrix)<sup>[424]</sup>。当  $A$  是  $n \times n$  矩阵时, 展开式 (7.1.6) 左端的行列式, 即得到显式的  $n$  次多项式方程

$$\alpha_0 + \alpha_1 \lambda + \cdots + \alpha_{n-1} \lambda^{n-1} + (-1)^n \lambda^n = 0 \quad (7.1.7)$$

称为矩阵  $A$  的特征方程, 多项式  $\det(A - \lambda I)$  称为特征多项式。为了避免矩阵  $A$  的特征值  $\lambda$  计算与特征多项式求根问题之间的混淆, 常在特征多项式中用  $x$  代替  $\lambda$ 。

**定义 7.1.2** 令  $A$  是一个  $n \times n$  矩阵, 则  $n$  阶多项式

$$p(x) = \det(A - xI) = \begin{vmatrix} a_{11} - x & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - x & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - x \end{vmatrix} = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0 \quad (7.1.8)$$

称为矩阵  $A$  的特征多项式。方程

$$p(x) = \det(A - xI) = 0 \quad (7.1.9)$$

称为矩阵  $A$  的特征方程。特征方程的根称为矩阵  $A$  的特征值 (eigenvalues, characteristic values, latent values) 或特征根 (characteristic roots, latent roots)。

显然, 矩阵  $A$  的  $n$  个特征值  $\lambda$  的计算与特征方程  $p(x) = 0$  的求根是两个等价的问题。由于特征多项式  $p(x)$  是变量  $x$  的  $n$  阶多项式, 特征方程  $p(x) = 0$  不可能有多于  $n$  个不同的根。即是说, 矩阵  $A_{n \times n}$  共有  $n$  个特征值。一个  $n \times n$  矩阵  $A$  能够产生一个特征多项式。同样, 每一个  $n$  次多项式也可以写成一个  $n \times n$  矩阵的特征多项式<sup>[36]</sup>。

**定理 7.1.1** 任何一个多项式

$$p(\lambda) = \lambda^n + a_1 \lambda^{n-1} + \cdots + a_{n-1} \lambda + a_n$$

都可以写成  $n \times n$  矩阵

$$A = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n \\ -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & -1 & 0 \end{bmatrix}$$

的特征多项式, 即有  $p(\lambda) = \det(\lambda I - A)$ 。

## 7.2 特征值与特征向量

本节重点讨论矩阵  $A$  的特征值与特征向量的有关计算及性质。

### 7.2.1 特征值

根据代数学基本定理知, 即使矩阵  $A$  是实的, 特征方程的根也可能是复的, 而且根的多重数可以是任意的, 甚至可以是  $n$  重根。这些根统称矩阵  $A$  的特征值。

关于特征值, 有必要先集中介绍以下术语<sup>[434, p.15]</sup>:

(1) 称  $A$  的特征值  $\lambda$  具有代数多重度 (algebraic multiplicity)  $\mu$ , 若  $\lambda$  是特征多项式  $\det(A - zI) = 0$  的  $\mu$  重根。

(2) 若特征值  $\lambda$  的代数多重度为 1, 则称该特征值为单特征值 (simple eigenvalue)。非单的特征值称为多重特征值 (multiple eigenvalue)。

(3) 称  $A$  的特征值  $\lambda$  具有几何多重度 (geometric multiplicity)  $\gamma$ , 若与  $\lambda$  对应的线性无关特征向量的个数为  $\gamma$ 。换言之, 几何多重度  $\gamma$  是特征空间  $\text{Null}(A - \lambda I)$  的维数。

(4) 矩阵  $A$  称为减次矩阵 (derogatory matrix), 若至少有一个特征值的几何多重度大于 1。

(5) 一特征值称为半单特征值 (semi-simple eigenvalue), 若它的代数多重度等于它的几何多重度。不是半单的特征值称为亏损特征值 (defective eigenvalue)。

几何多重度还有另外一种定义, 将在稍后介绍。

众所周知, 任何一个  $n$  阶多项式  $p(x)$  都可以写成因式分解形式

$$p(x) = a(x - x_1)(x - x_2) \cdots (x - x_n) \quad (7.2.1)$$

注意, 特征多项式  $p(x)$  的  $n$  个根  $x_1, x_2, \dots, x_n$  不一定是各不相同的, 也不一定就是实的。

一般说来, 矩阵  $A$  的特征值是各不相同的。若特征多项式存在多重根, 则称矩阵  $A$  具有退化特征值 (degenerate eigenvalue)。

需要注意的是, 即使矩阵  $A$  是实矩阵, 其特征值也可能是复的。以 Givens 旋转矩阵

$$A = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

为例, 其特征方程

$$\det(A - \lambda I) = \begin{vmatrix} \cos \theta - \lambda & -\sin \theta \\ \sin \theta & \cos \theta - \lambda \end{vmatrix} = (\cos \theta - \lambda)^2 + \sin^2 \theta = 0$$

然而, 若  $\theta$  不是  $\pi$  的整数倍, 则  $\sin^2 \theta > 0$ 。此时, 特征方程不可能有  $\lambda$  的实根, 即 Givens 旋转矩阵的两个特征值都为复数, 与它们对应的特征向量也是复向量。

下面是特征值的一些重要性质。

**性质 1** 矩阵  $A$  奇异, 当且仅当至少有一个特征值  $\lambda = 0$ 。

**性质 2** 矩阵  $A$  和  $A^T$  具有相同的特征值。

**性质 3** 若  $\lambda$  是  $n \times n$  矩阵  $A$  的特征值, 则有

- (1)  $\lambda^k$  是矩阵  $A^k$  的特征值。
- (2) 若  $A$  非奇异, 则  $A^{-1}$  具有特征值  $1/\lambda$ 。
- (3) 矩阵  $A + \sigma^2 I$  的特征值为  $\lambda + \sigma^2$ 。

### 7.2.2 特征向量

若矩阵  $A_{n \times n}$  是一个一般的复矩阵, 并且  $\lambda$  是其特征值, 则满足

$$(A - \lambda I)v = 0 \quad \text{或} \quad Av = \lambda v \quad (7.2.2)$$

的向量  $v$  称为  $A$  与特征值  $\lambda$  对应的右特征向量, 而满足

$$u^H(A - \lambda I) = 0^T \quad \text{或} \quad u^H A = \lambda u^H \quad (7.2.3)$$

的向量  $u$  称为  $A$  与特征值  $\lambda$  对应的左特征向量。

若矩阵  $A$  为 Hermitian 矩阵, 则由于其所有特征值为实数, 立即知  $v = u$ , 即 Hermitian 矩阵的左和右特征向量相同。

有必要对矩阵的奇异值分解与特征值分解之间的联系与区别作一番比较:

- (1) 奇异值分解适用于任何  $m \times n$  长方形矩阵 ( $m \geq n$  或者  $m < n$  均可), 特征值分解只适用于正方矩阵。
- (2) 即使是同一个  $n \times n$  非 Hermitian 矩阵  $A$ , 奇异值和特征值的定义也是完全不同的: 奇异值定义为使原矩阵  $A$  的秩减小 1 的误差矩阵  $E_k$  的谱范数

$$\sigma_k = \min_{E \in \mathbb{C}^{m \times n}} \{\|E\|_{\text{spec}} : \text{rank}(A + E) \leq k - 1\}, \quad k = 1, \dots, \min\{m, n\} \quad (7.2.4)$$

而特征值定义为特征多项式  $\det(A - \lambda I) = 0$  的根。同一正方矩阵的奇异值和特征值之间无内在的关系, 但  $m \times n$  矩阵  $A$  的非零奇异值是  $n \times n$  Hermitian 矩阵  $A^H A$  或  $m \times m$  Hermitian 矩阵  $AA^H$  的非零特征值的正平方根。

- (3)  $m \times n$  矩阵  $A$  与奇异值  $\sigma_i$  对应的左奇异向量  $u_i$  和右奇异向量  $v_i$  定义为满足  $u_i^H A v_i = \sigma_i$  的两个向量, 而  $n \times n$  矩阵  $A$  的左和右特征向量则分别由  $u^H A = \lambda_i u^H$  和  $A v_i = \lambda_i v_i$  定义。因此, 对于同一个  $n \times n$  非 Hermitian 矩阵  $A$ , 它的(左和右)奇异向量与(左和右)特征向量之间也没有内在的关系。然而, 矩阵  $A \in \mathbb{C}^{m \times n}$  的左奇异向量  $u_i$  和右奇异向量  $v_i$  分别是  $m \times m$  Hermitian 矩阵  $AA^H$  和  $A^H A$  的特征向量。

**命题 7.2.1** 令  $\mathbf{u}_1, \dots, \mathbf{u}_k$  是  $n \times n$  矩阵  $\mathbf{A}$  与不同特征值  $\lambda_1, \dots, \lambda_k$  相对应的特征向量, 即

$$\mathbf{A}\mathbf{u}_i = \lambda_i \mathbf{u}_i, \quad i = 1, 2, \dots, k; \quad k \leq n \quad (7.2.5)$$

$$\lambda_i \neq \lambda_j, \quad i \neq j; \quad 1 \leq i, j \leq k \quad (7.2.6)$$

则这  $k$  个特征向量的集合  $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$  是一个线性无关集合。

**证明** 参见文献 [255]。

若  $k = n$ , 则命题 7.2.1 的结果为下列推论。

**推论 7.2.1** 令  $\mathbf{A}$  是一个  $n \times n$  矩阵。若  $\mathbf{A}$  具有不同的  $n$  个特征值, 则  $\mathbf{A}$  具有  $n$  个线性无关的特征向量。

另一方面, 从式 (7.1.2) 容易看出, 一个特征向量乘以任一非零的标量后, 仍然满足式 (7.1.2), 即还是特征向量。为了避免特征向量的多值性, 通常定义特征向量总是具有单位内积 (或者单位范数), 即约定  $\mathbf{u}^H \mathbf{u} = 1$ 。

**命题 7.2.2** 若  $(\lambda, \mathbf{u})$  是  $n \times n$  实矩阵  $\mathbf{A}$  的特征对, 则  $(\lambda^*, \mathbf{u}^*)$  也是实矩阵  $\mathbf{A}$  的特征对。换言之, 若实矩阵存在复特征值与/或复特征向量, 则它们一定分别以复共轭对的形式出现。

**证明** 由于  $\mathbf{A}$  是实矩阵, 故有  $(\mathbf{A}\mathbf{u})^* = \mathbf{A}\mathbf{u}^*$ , 式中,  $*$  为复数共轭。利用这一结果和已知条件  $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$ , 易知  $\mathbf{A}\mathbf{u}^* = (\mathbf{A}\mathbf{u})^* = (\lambda\mathbf{u})^* = \lambda^*\mathbf{u}^*$ 。这一结果表明,  $(\lambda^*, \mathbf{u}^*)$  也是实矩阵  $\mathbf{A}$  的特征对。 ■

如果一个普通的  $n \times n$  矩阵  $\mathbf{A}$  已求出了不同的特征值, 那么只要求解矩阵方程  $(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}$ , 即可得到与每个已知  $\lambda$  对应的特征向量  $\mathbf{x}$ 。

下面通过一个例子说明如何分步求出一个  $n \times n$  矩阵  $\mathbf{A}$  的特征值、对应的特征向量和对角化。

**例 7.2.1** 已知一个  $3 \times 3$  实矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 3 & 3 \\ -2 & 1 & 1 \end{bmatrix}$$

是非对称的一般矩阵。直接计算知, 特征多项式

$$\det(\mathbf{A} - \lambda\mathbf{I}) = \begin{vmatrix} 1 - \lambda & 1 & 1 \\ 0 & 3 - \lambda & 3 \\ -2 & 1 & 1 - \lambda \end{vmatrix} = -\lambda(\lambda - 2)(\lambda - 3)$$

求解特征方程  $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$  得到矩阵  $\mathbf{A}$  的 3 个特征值  $\lambda = 0, 2, 3$ 。

(1) 对于特征值  $\lambda = 0$ , 有  $(\mathbf{A} - 0\mathbf{I})\mathbf{x} = \mathbf{0}$ , 即有

$$x_1 + x_2 + x_3 = 0$$

$$3x_2 + 3x_3 = 0$$

$$-2x_1 + x_2 + x_3 = 0$$

其解为  $x_1 = 0$  和  $x_2 = -x_3$ , 其中,  $x_3$  任意。因此, 与特征值  $\lambda = 0$  对应的特征向量为

$$\mathbf{x} = \begin{bmatrix} 0 \\ -a \\ a \end{bmatrix} = a \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix}, \quad a \neq 0$$

取  $a = 1$ , 得特征向量为  $\mathbf{x}_1 = [0, -1, 1]^T$ 。

(2) 对于特征值  $\lambda = 2$ , 有  $(\mathbf{A} - 2I)\mathbf{x} = 0$ , 即

$$-x_1 + x_2 + x_3 = 0$$

$$x_2 + 3x_3 = 0$$

$$-2x_1 + x_2 - x_3 = 0$$

其解为  $x_1 = -2x_3, x_2 = -3x_3$ , 其中,  $x_3$  任意。因此, 与特征值  $\lambda = 2$  对应的特征向量为

$$\mathbf{x} = \begin{bmatrix} -2a \\ -3a \\ a \end{bmatrix} = a \begin{bmatrix} -2 \\ -3 \\ 1 \end{bmatrix}, \quad a \neq 0$$

取  $a = 1$ , 得特征向量为  $\mathbf{x}_2 = [-2, -3, 1]^T$ 。

(3) 类似地, 与  $\lambda = 3$  对应的特征向量为  $\mathbf{x}_3 = [1, 2, 0]^T$ 。三个特征向量组成矩阵

$$\mathbf{U} = \begin{bmatrix} 0 & -2 & 1 \\ -1 & -3 & 2 \\ 1 & 1 & 0 \end{bmatrix}$$

其逆矩阵为

$$\mathbf{U}^{-1} = \begin{bmatrix} 1 & -1/2 & 1/2 \\ -1 & 1/2 & 1/2 \\ -1 & 1 & 1 \end{bmatrix}$$

于是, 矩阵  $\mathbf{A}$  的对角化结果为

$$\begin{aligned} \mathbf{U}^{-1}\mathbf{AU} &= \begin{bmatrix} 1 & -1/2 & 1/2 \\ -1 & 1/2 & 1/2 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 3 & 3 \\ -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & -2 & 1 \\ -1 & -3 & 2 \\ 1 & 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 & -1/2 & 1/2 \\ -1 & 1/2 & 1/2 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & -4 & 3 \\ 0 & -6 & 6 \\ 0 & 2 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \end{aligned}$$

它恰好就是由矩阵  $\mathbf{A}$  的三个不同特征值 0, 2, 3 构成的对角矩阵。

### 7.2.3 与其他矩阵函数的关系

一个矩阵的特征值与矩阵的其他标量函数有着密切的关系。这里先引出特征值的条件数的定义。

**定义 7.2.1** [434, p.93] 任意一个矩阵  $A$  的单个特征值  $\lambda$  的条件数定义为

$$\text{cond}(\lambda) = \frac{1}{\cos \theta(u, v)} \quad (7.2.7)$$

式中,  $\theta(u, v)$  表示与特征值  $\lambda$  对应的左特征向量  $u$  和右特征向量  $v$  之间的夹角 (锐角)。

**例 7.2.2** [434, p.93] 考虑矩阵

$$A = \begin{bmatrix} -149 & -50 & -154 \\ 537 & 180 & 546 \\ -27 & -9 & -25 \end{bmatrix}$$

其特征值为  $\{1, 2, 3\}$ 。与特征值  $\lambda = 1$  对应的左和右特征向量分别为

$$u = \begin{bmatrix} 0.6810 \\ 0.2253 \\ 0.6967 \end{bmatrix} \quad \text{和} \quad v = \begin{bmatrix} 0.3162 \\ -0.9487 \\ 0.0000 \end{bmatrix}$$

相应的条件数  $\text{cond}(\lambda_1) \approx 603.64$ 。这说明矩阵元素 0.01 数量级的扰动将引起特征值  $\lambda_1$  最大 6 倍的变化。例如, 元素  $a_{11}$  扰动到  $-149.01$ , 则矩阵  $A$  的特征值变为

$$\{0.2287, 3.2878, 2.4735\}$$

下面讨论一个矩阵的所有特征值的集合与该矩阵的谱、行列式、迹之间的关系。

### 1. 与矩阵的谱的关系

**定义 7.2.2** 矩阵  $A \in \mathbb{C}^{n \times n}$  的所有特征值  $\lambda \in \mathbb{C}$  的集合称为矩阵  $A$  的谱, 记作  $\lambda(A)$ 。矩阵  $A$  的谱半径是非负实数, 定义为

$$\rho(A) = \max |\lambda| : \lambda \in \lambda(A) \quad (7.2.8)$$

由于  $\rho(A)$  是包含  $A$  的所有特征值在圆内或圆上的最小圆盘的半径, 圆心在复平面的原点, 故名谱半径。

令  $\lambda(A) = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$ , 则

$$\det(A) = \lambda_1 \lambda_2 \cdots \lambda_n \quad (7.2.9)$$

显然, 若  $A$  具有零特征值, 则  $\det(A) = 0$ , 即矩阵  $A$  奇异。反之, 若  $A$  的所有特征值都不等于零, 则  $\det(A) \neq 0$ , 即矩阵  $A$  非奇异。

### 2. 与矩阵的行列式和迹的关系

矩阵  $A$  的迹等于其所有特征值之和, 而行列式  $|A|$  等于矩阵  $A$  所有特征值的乘积, 即有

$$\text{tr}(A) = \sum_{i=1}^n \lambda_i \quad (7.2.10)$$

$$\det(A) = \prod_{i=1}^n \lambda_i \quad (7.2.11)$$

令  $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$  表示  $n$  个信号组成的向量, 且  $\mathbf{R}_s = \mathbb{E}\{\mathbf{s}(t)\mathbf{s}^H(t)\}$  表示信号向量  $\mathbf{s}(t)$  的相关矩阵, 即

$$\mathbf{R}_s = \begin{bmatrix} \mathbb{E}\{|s_1(t)|^2\} & \mathbb{E}\{s_1(t)s_2^*(t)\} & \cdots & \mathbb{E}\{s_1(t)s_n^*(t)\} \\ \mathbb{E}\{s_2(t)s_1^*(t)\} & \mathbb{E}\{|s_2(t)|^2\} & \cdots & \mathbb{E}\{s_2(t)s_n^*(t)\} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{E}\{s_n(t)s_1^*(t)\} & \mathbb{E}\{s_n(t)s_2^*(t)\} & \cdots & \mathbb{E}\{|s_n(t)|^2\} \end{bmatrix}$$

假定其  $n$  个特征特征值为  $\lambda_1, \lambda_2, \dots, \lambda_n$ 。利用矩阵的迹的定义知

$$\text{tr}(\mathbf{R}_s) = \sum_{i=1}^n \mathbb{E}\{|s_i(t)|^2\} = \sum_{i=1}^n \lambda_i \quad (7.2.12)$$

即是说, 相关矩阵  $\mathbf{R}_s$  的特征值之和反映  $n$  个信号功率之和。

**定义 7.2.3** 对称矩阵  $\mathbf{A} \in \mathbb{R}^{n \times n}$  的惯性  $\text{In}(\mathbf{A})$  定义为三元组

$$\text{In}(\mathbf{A}) = (i_+(\mathbf{A}), i_-(\mathbf{A}), i_0(\mathbf{A}))$$

其中,  $i_+(\mathbf{A}), i_-(\mathbf{A})$  和  $i_0(\mathbf{A})$  分别是  $\mathbf{A}$  的正、负和零特征值的个数 (多重特征值分别计算多重数在内)。另外, 量  $i_+(\mathbf{A}) - i_-(\mathbf{A})$  叫做  $\mathbf{A}$  的符号差 (signature)。

显然, 对称矩阵  $\mathbf{A}$  的秩由  $\text{rank}(\mathbf{A}) = i_+(\mathbf{A}) + i_-(\mathbf{A})$  决定。

### 3. 矩阵多项式的特征值

考虑矩阵  $\mathbf{A}$  的  $n$  次多项式

$$f(\mathbf{A}) = \mathbf{A}^n + c_1 \mathbf{A}^{n-1} + \cdots + c_{n-1} \mathbf{A} + c_n \mathbf{I} \quad (7.2.13)$$

若矩阵  $\mathbf{A}$  有特征对  $(\lambda, \mathbf{u})$ , 即  $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$ , 则由于

$$\mathbf{A}^2\mathbf{u} = \lambda\mathbf{A}\mathbf{u} = \lambda^2\mathbf{u}, \quad \mathbf{A}^3\mathbf{u} = \lambda\mathbf{A}^2\mathbf{u} = \lambda^3\mathbf{u}, \quad \dots, \quad \mathbf{A}^n\mathbf{u} = \lambda^n\mathbf{u}$$

立即有

$$\begin{aligned} & (\mathbf{A}^n + c_1 \mathbf{A}^{n-1} + \cdots + c_{n-1} \mathbf{A} + c_n \mathbf{I})\mathbf{u} \\ &= \mathbf{A}^n\mathbf{u} + c_1 \mathbf{A}^{n-1}\mathbf{u} + \cdots + c_{n-1} \mathbf{A}\mathbf{u} + c_n \mathbf{u} \\ &= \lambda^n\mathbf{u} + c_1 \lambda^{n-1}\mathbf{u} + \cdots + c_{n-1} \lambda\mathbf{u} + c_n \mathbf{u} \\ &= (\lambda^n + c_1 \lambda^{n-1} + \cdots + c_{n-1} \lambda + c_n)\mathbf{u} \end{aligned} \quad (7.2.14)$$

令矩阵多项式  $f(\mathbf{A})$  的特征值为  $f(\lambda)$ , 即  $f(\mathbf{A})\mathbf{u} = f(\lambda)\mathbf{u}$ 。将这一关系式代入式 (7.2.14), 易知

$$f(\lambda) = \lambda^n + c_1 \lambda^{n-1} + \cdots + c_{n-1} \lambda + c_n \quad (7.2.15)$$

是矩阵多项式  $f(\mathbf{A})$  的特征值。

标量  $x$  的幂级数定义为  $e^x = 1 + x + x^2/2! + x^3/3! + \dots$ 。类似地, 矩阵  $\mathbf{A}$  的幂级数定义为

$$e^{\mathbf{A}} = \mathbf{I} + \mathbf{A} + \frac{1}{2!}\mathbf{A}^2 + \frac{1}{3!}\mathbf{A}^3 + \dots = \sum_{i=0}^{\infty} \frac{1}{i!} \mathbf{A}^i \quad (7.2.16)$$

假定级数收敛。若矩阵  $\mathbf{A}$  的特征值为  $\lambda$ , 则由矩阵多项式的特征值表示式 (7.2.15), 立即知矩阵的指数函数  $e^{\mathbf{A}}$  的特征值为

$$f(\lambda) = \sum_{i=0}^{\infty} \lambda^i / i! = e^{\lambda} \quad (7.2.17)$$

#### 7.2.4 特征值和特征向量的性质

前面分析了特征值和特征向量的一些典型性质。事实上, 一个  $n \times n$  矩阵 (不一定是 Hermitian 矩阵)  $\mathbf{A}$  的特征值具有广泛的性质, 详见下面的汇总 [238]。

- (1)  $n \times n$  矩阵  $\mathbf{A}$  共有  $n$  个特征值, 其中, 多重特征值按照其多重度计数。
- (2) 若非对称的实矩阵  $\mathbf{A}$  存在复特征值与/或复特征向量, 则它们一定分别以复共轭对的形式出现。
- (3) 若  $\mathbf{A}$  是实对称矩阵或 Hermitian 矩阵, 则其所有特征值都是实数。
- (4) 关于对角矩阵与三角矩阵的特征值:
  - ① 若  $\mathbf{A} = \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$ , 则其特征值为  $a_{11}, a_{22}, \dots, a_{nn}$ 。
  - ② 若  $\mathbf{A}$  为三角矩阵, 则其对角元素是所有的特征值。
- (5) 对一个  $n \times n$  矩阵  $\mathbf{A}$ :
  - ① 若  $\lambda$  是  $\mathbf{A}$  的特征值, 则  $\lambda$  也是  $\mathbf{A}^T$  的特征值。
  - ② 若  $\lambda$  是  $\mathbf{A}$  的特征值, 则  $\lambda^*$  是  $\mathbf{A}^H$  的特征值。
  - ③ 若  $\lambda$  是  $\mathbf{A}$  的特征值, 则  $\lambda + \sigma^2$  是  $\mathbf{A} + \sigma^2 \mathbf{I}$  的特征值。
  - ④ 若  $\lambda$  是矩阵  $\mathbf{A}$  的特征值, 则  $1/\lambda$  是逆矩阵  $\mathbf{A}^{-1}$  的特征值。
- (6) 幂等矩阵  $\mathbf{A}^2 = \mathbf{A}$  的所有特征值取 0 或者 1。
- (7) 若  $\mathbf{A}$  是实正交矩阵, 则其所有特征值位于单位圆上。
- (8) 特征值与矩阵奇异性关系:
  - ① 若  $\mathbf{A}$  奇异, 则它至少有一个特征值为 0。
  - ② 若  $\mathbf{A}$  非奇异, 则它所有的特征值非零。
- (9) 特征值与迹的关系: 矩阵  $\mathbf{A}$  的特征值之和等于该矩阵的迹, 即  $\sum_{i=1}^n \lambda_i = \text{tr}(\mathbf{A})$ 。
- (10) 与不同特征值  $\lambda_1, \lambda_2, \dots, \lambda_n$  对应的非零特征向量  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$  线性无关。
- (11) 一个 Hermitian 矩阵  $\mathbf{A}$  是正定 (或半正定) 的, 当且仅当它的特征值是正 (或者非负) 的。

(12) 特征值与行列式的关系: 矩阵  $A$  所有特征值的乘积等于该矩阵的行列式, 即  $\prod_{i=1}^n \lambda_i = \det(A) = |A|$ 。

(13) 特征值与秩的关系:

① 若  $n \times n$  矩阵  $A$  有  $r$  个非零特征值, 则  $\text{rank}(A) \geq r$ 。

② 若 0 是  $n \times n$  矩阵  $A$  的无多重的特征值, 则  $\text{rank}(A) = n - 1$ 。

③ 若  $\text{rank}(A - \lambda I) \leq n - 1$ , 则  $\lambda$  是矩阵  $A$  的特征值。

(14) 若  $A$  的特征值不相同, 则一定可以找到一个相似矩阵  $S^{-1}AS = D$  (对角矩阵), 其对角元素即是矩阵  $A$  的特征值。

(15)  $n \times n$  矩阵  $A$  的任何一个特征值  $\lambda$  的几何多重度都不可能大于  $\lambda$  的代数多重度。

(16) Cayley-Hamilton 定理: 若  $\lambda_1, \lambda_2, \dots, \lambda_n$  是  $n \times n$  矩阵  $A$  的特征值, 则

$$\prod_{i=1}^n (A - \lambda_i I) = 0$$

(17) 关于相似矩阵的特征值:

① 若  $\lambda$  是  $n \times n$  矩阵  $A$  的一个特征值, 并且  $n \times n$  矩阵  $B$  非奇异, 则  $\lambda$  也是矩阵  $B^{-1}AB$  的一个特征值, 但对应的特征向量一般不相同。

② 若  $\lambda$  是  $n \times n$  矩阵  $A$  的一个特征值, 并且  $n \times n$  矩阵  $B$  是酉矩阵, 则  $\lambda$  也是矩阵  $B^HAB$  的一个特征值, 但对应的特征向量一般不相同。

③ 若  $\lambda$  是  $n \times n$  矩阵  $A$  的一个特征值, 并且  $n \times n$  矩阵  $B$  是正交矩阵, 则  $\lambda$  也是矩阵  $B^TAB$  的一个特征值, 但对应的特征向量一般不相同。

(18) 一个  $n \times n$  矩阵  $A = [a_{ij}]$  的最大特征值以该矩阵的列元素之和的最大值为界, 即  $\lambda_{\max} \leq \max_i \sum_{j=1}^n a_{ij}$ 。

(19) 随机向量  $x(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T$  的相关矩阵  $R = E\{x(t)x^H(t)\}$  的特征值以信号的最大功率  $P_{\max} = \max_i E\{|x_i(t)|^2\}$  和最小功率  $P_{\min} = \min_i E\{|x_i(t)|^2\}$  为界, 即有

$$P_{\min} \leq \lambda_i \leq P_{\max} \quad (7.2.18)$$

(20) 随机向量  $x(t)$  的相关矩阵  $R$  的特征值散布 (eigenvalue spread) 为

$$\chi(R) = \frac{\lambda_{\max}}{\lambda_{\min}} \quad (7.2.19)$$

(21) 关于绝对值小于 1 的特征值:

① 若  $|\lambda_i| < 1, i = 1, \dots, n$ , 则矩阵  $A \pm I_n$  非奇异。

②  $|\lambda_i| < 1, i = 1, \dots, n \Leftrightarrow \det(A - zI_n) = 0$  的根只可能位于单位圆内。

(22) 关于  $m \times n$  ( $n \geq m$ ) 矩阵  $\mathbf{A}$  与  $n \times m$  矩阵  $\mathbf{B}$  乘积的特征值:

① 若  $\lambda$  是矩阵乘积  $\mathbf{AB}$  的特征值, 则  $\lambda$  也是  $\mathbf{BA}$  的特征值。

② 若  $\lambda \neq 0$  是矩阵乘积  $\mathbf{BA}$  的特征值, 则  $\lambda$  也是  $\mathbf{AB}$  的特征值。

③ 若  $\lambda_1, \lambda_2, \dots, \lambda_m$  是矩阵乘积  $\mathbf{AB}$  的特征值, 则矩阵乘积  $\mathbf{BA}$  的  $n$  个特征值为  $\lambda_1, \dots, \lambda_m, 0, \dots, 0$ 。

(23) 若矩阵  $\mathbf{A}$  的特征值为  $\lambda$ , 则矩阵多项式  $f(\mathbf{A}) = \mathbf{A}^n + c_1\mathbf{A}^{n-1} + \dots + c_{n-1}\mathbf{A} + c_n\mathbf{I}$  的特征值为

$$f(\lambda) = \lambda^n + c_1\lambda^{n-1} + \dots + c_{n-1}\lambda + c_n \quad (7.2.20)$$

(24) 若矩阵  $\mathbf{A}$  的特征值为  $\lambda$ , 则矩阵指数函数  $e^{\mathbf{A}}$  的特征值为  $e^\lambda$ 。

性质 (14) 给出了求矩阵  $\mathbf{A}$  的特征值的相似变换方法。此时, 通常选择相似变换矩阵  $\mathbf{S}$  为正交矩阵。

下面概括了特征值  $\lambda$  与特征向量  $\mathbf{u}$  组成的特征对  $(\lambda, \mathbf{u})$  具有的性质 [238]。

(1) 若  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{A}$  的特征对, 则  $(c\lambda, \mathbf{u})$  是矩阵  $c\mathbf{A}$  的特征对, 其中,  $c$  为非零的常数。

(2) 若  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{A}$  的特征对, 则  $(\lambda, c\mathbf{u})$  也是矩阵  $\mathbf{A}$  的一个特征对, 其中,  $c$  为非零的常数。

(3) 若  $(\lambda_i, \mathbf{u}_i)$  和  $(\lambda_j, \mathbf{u}_j)$  分别是矩阵  $\mathbf{A}$  的特征对, 并且  $\lambda_i \neq \lambda_j$ , 则特征向量  $\mathbf{u}_i$  与  $\mathbf{u}_j$  线性无关。

(4) Hermitian 矩阵与不同特征值对应的特征向量相互正交, 即对于  $\lambda_i \neq \lambda_j$ , 有  $\mathbf{u}_i^H \mathbf{u}_j = 0$ 。

(5) 若  $\lambda$  是矩阵  $\mathbf{A}$  的特征值, 向量  $\mathbf{u}_1$  和  $\mathbf{u}_2$  分别是与  $\lambda$  对应的特征向量, 则  $c_1\mathbf{u}_1 + c_2\mathbf{u}_2$  是矩阵  $\mathbf{A}$  与特征值  $\lambda$  对应的特征向量, 其中,  $c_1$  和  $c_2$  为常数, 并且至少有一个不等于 0。

(6) 若  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{A}$  的特征对, 并且  $\alpha_0, \alpha_1, \dots, \alpha_p$  为复常数, 则  $f(\lambda) = \alpha_0 + \alpha_1\lambda + \dots + \alpha_p\lambda^p$  是矩阵多项式  $f(\mathbf{A}) = \alpha_0\mathbf{I} + \alpha_1\mathbf{A} + \dots + \alpha_p\mathbf{A}^p$  的特征值, 与之对应的特征向量仍然为  $\mathbf{u}$ 。

(7) 若  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{A}$  的特征对, 则  $(\lambda^k, \mathbf{u})$  是矩阵  $\mathbf{A}^k$  的特征对。

(8) 若  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{A}$  的特征对, 则  $(e^\lambda, \mathbf{u})$  是矩阵指数函数  $e^{\mathbf{A}}$  的特征对。

(9) 若  $n \times n$  矩阵  $\mathbf{A}$  有  $n$  个线性无关的特征向量, 则  $\mathbf{A}$  的特征值分解为

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{U}^{-1} \quad (7.2.21)$$

(10) 若  $\lambda(A)$  和  $\lambda(B)$  分别是矩阵  $A$  和  $B$  的特征值, 而  $u(A)$  和  $u(B)$  分别是与特征值  $\lambda(A)$  和  $\lambda(B)$  对应的特征向量, 则

①  $\lambda(A)\lambda(B)$  是矩阵 Kronecker 积  $A \otimes B$  的特征值, 并且  $u(A) \otimes u(B)$  是与特征值  $\lambda(A)\lambda(B)$  对应的特征向量。

②  $\lambda(A)$  和  $\lambda(B)$  分别是矩阵直和  $A \oplus B$  的特征值, 与它们对应的特征向量分别为  $\begin{bmatrix} u(A) \\ 0 \end{bmatrix}$  和  $\begin{bmatrix} 0 \\ u(B) \end{bmatrix}$ 。

(11) 令  $B$  是一个秩等于 1 的  $n \times n$  矩阵, 其特征值为  $\lambda$ , 特征向量为  $u_1$ , 则

$$\begin{aligned} (B + \alpha I)^{-1} &= \frac{1}{\alpha + \lambda} u_1 u_1^H + \frac{1}{\alpha} I - \frac{1}{\alpha} u_1 u_1^H \\ &= \frac{1}{\alpha} I - \frac{\lambda}{\alpha(\alpha + \lambda)} u_1 u_1^H \end{aligned} \quad (7.2.22)$$

矩阵  $A$  的奇异值问题往往转化为相应矩阵的特征值问题求解。实现这一转化有两种主要方法:

方法 1 矩阵  $A_{m \times n}$  的非零奇异值是  $m \times m$  矩阵  $AA^T$  或者  $n \times n$  矩阵  $A^T A$  的非零特征值  $\lambda_i$  的正平方根, 并且  $A$  与  $\sigma_i$  对应的左奇异向量  $u_j$  和右奇异向量  $v_i$  分别是矩阵  $AA^T$  和  $A^T A$  与非零特征值  $\lambda_i$  对应的特征向量。

方法 2 矩阵  $A_{m \times n}$  的奇异值分解转化为  $(m+n) \times (m+n)$  增广矩阵

$$\begin{bmatrix} O & A \\ A^T & O \end{bmatrix} \quad (7.2.23)$$

的特征值分解。

**定理 7.2.1** (Jordan-Wielandt 定理)<sup>[463, Theorem I.4.2]</sup> 若  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{p-1} \geq \sigma_p$  是  $A_{m \times n}$  的奇异值 (其中,  $p = \min\{m, n\}$ ), 则上述增广矩阵具有特征值

$$-\sigma_1, \dots, -\sigma_p, \underbrace{0, \dots, 0}_{|m-n|\uparrow}, \sigma_p, \dots, \sigma_1$$

与  $\pm \sigma_j$  相对应的特征向量为

$$\begin{bmatrix} u_j \\ \pm v_j \end{bmatrix}, \quad j = 1, 2, \dots, p$$

若  $m \neq n$ , 则另有特征向量

$$\begin{bmatrix} u_j \\ 0 \end{bmatrix}, \quad n+1 \leq j \leq m \quad \text{或} \quad \begin{bmatrix} 0 \\ v_j \end{bmatrix}, \quad m+1 \leq j \leq n$$

分别取决于  $m > n$  或者  $m < n$ 。

定理 7.2.1 启迪了使用增广矩阵的特征值分解计算矩阵  $A$  的奇异值分解的一类方法。例如, 通过对 Jacobi-Davidson 算法加以推广, Hochstenbach 于 2001 年提出了 Jacobi-Davidson 型奇异值分解算法<sup>[236]</sup>。

关于矩阵之和  $\mathbf{A} + \mathbf{B}$  的特征值，有下面的结果。

**定理 7.2.2** (Weyl 定理)<sup>[294]</sup> 设  $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$  是 Hermitian 矩阵，且特征值按照递增顺序排列

$$\begin{aligned}\lambda_1(\mathbf{A}) &\leq \lambda_2(\mathbf{A}) \leq \cdots \leq \lambda_n(\mathbf{A}) \\ \lambda_1(\mathbf{B}) &\leq \lambda_2(\mathbf{B}) \leq \cdots \leq \lambda_n(\mathbf{B}) \\ \lambda_1(\mathbf{A} + \mathbf{B}) &\leq \lambda_2(\mathbf{A} + \mathbf{B}) \leq \cdots \leq \lambda_n(\mathbf{A} + \mathbf{B})\end{aligned}$$

则

$$\lambda_i(\mathbf{A} + \mathbf{B}) \geq \begin{cases} \lambda_i(\mathbf{A}) + \lambda_1(\mathbf{B}) \\ \lambda_{i-1}(\mathbf{A}) + \lambda_2(\mathbf{B}) \\ \vdots \\ \lambda_1(\mathbf{A}) + \lambda_i(\mathbf{B}) \end{cases} \quad (7.2.24)$$

和

$$\lambda_i(\mathbf{A} + \mathbf{B}) \leq \begin{cases} \lambda_i(\mathbf{A}) + \lambda_n(\mathbf{B}) \\ \lambda_{i+1}(\mathbf{A}) + \lambda_{n-1}(\mathbf{B}) \\ \vdots \\ \lambda_n(\mathbf{A}) + \lambda_i(\mathbf{B}) \end{cases} \quad (7.2.25)$$

式中， $i = 1, 2, \dots, n$ 。

特别地，当  $\mathbf{A}$  为实对称矩阵，并且  $\mathbf{B} = a\mathbf{z}\mathbf{z}^T$ ，则有下面的交织特征值定理 (interlacing eigenvalue theorem) [198, Theorem 8.1.8]。

**定理 7.2.3** 令  $\mathbf{A} \in \mathbb{R}^{n \times n}$  是一对称矩阵，其特征值  $\lambda_1, \lambda_2, \dots, \lambda_n$  满足

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \quad (7.2.26)$$

并令  $\mathbf{z} \in \mathbb{R}^n$  是一向量，其范数  $\|\mathbf{z}\| = 1$ 。假定  $a$  为一实数，并且矩阵  $\mathbf{A} + a\mathbf{z}\mathbf{z}^T$  的特征值

$$\xi_1 \geq \xi_2 \geq \cdots \geq \xi_n \quad (7.2.27)$$

则

$$\xi_1 \geq \lambda_1 \geq \xi_2 \geq \lambda_2 \geq \cdots \geq \xi_n \geq \lambda_n, \quad \text{若 } a > 0 \quad (7.2.28)$$

或者

$$\lambda_1 \geq \xi_1 \geq \lambda_2 \geq \xi_2 \geq \cdots \geq \lambda_n \geq \xi_n, \quad \text{若 } a < 0 \quad (7.2.29)$$

并且无论  $a > 0$  还是  $a < 0$ ，均有

$$\sum_{i=1}^n (\xi_i - \lambda_i) = a \quad (7.2.30)$$

### 7.2.5 矩阵的可对角化定理

一个矩阵的规范化表示称为该矩阵的范式。现在考虑使用特征值和特征向量表示的矩阵范式。为此，先分析矩阵的秩与多重特征值之间的重要关系。

**引理 7.2.1** 若  $n \times n$  矩阵  $A$  的秩为  $r_A$ ，并且具有  $z_A$  个零特征值，则

$$r_A \geq n - z_A \quad (7.2.31)$$

即非零特征值的个数不会超过矩阵的秩。

**引理 7.2.2** 若  $\lambda_k$  是  $n \times n$  矩阵  $A$  的多重特征值，并且其多重度为  $m_k$ ，则

$$\text{rank}(A - \lambda_k I) \geq n - m_k \quad (7.2.32)$$

以上两个引理的证明可参考文献 [444, p.306]。

下面考虑如何对一个正方矩阵进行相似变换，将它变成对角矩阵。

如前所述，任意正方矩阵  $A$  的每一个特征值  $\lambda_i$  都有一个相对应的特征向量  $u_i$  满足

$$Au_i = \lambda_i u_i, \quad i = 1, 2, \dots, n \quad (7.2.33)$$

这一方程组也可合写为

$$A[u_1, \dots, u_n] = [u_1, \dots, u_n] \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \quad (7.2.34)$$

定义矩阵

$$U = [u_1, \dots, u_n], \quad \Sigma = \text{diag}(\lambda_1, \dots, \lambda_n) \quad (7.2.35)$$

则式 (7.2.34) 可以简写作

$$AU = U\Sigma \quad (7.2.36)$$

若矩阵  $U$  非奇异，则有

$$U^{-1}AU = \Sigma = \text{diag}(\lambda_1, \dots, \lambda_n) \quad (7.2.37)$$

通过相似变换得到的对角矩阵  $\Sigma$  称为矩阵  $A$  在相似下的范式 (canonical form under similarity) 或简称为相似范式 (similar canonical form) [444, p.283]。

**定义 7.2.4** 一个  $n \times n$  实矩阵  $A$  若与一个对角矩阵相似，则称矩阵  $A$  是可对角化的 (diagonalizable)。

**定理 7.2.4** 一个  $n \times n$  实矩阵  $A$  是可对角化的，当且仅当  $A$  具有  $n$  个线性无关的特征向量。

**证明** (1) 充分条件的证明 假设  $u_1, \dots, u_n$  是矩阵  $A$  的  $n$  个线性无关的特征向量，即有  $Au_i = \lambda_i u_i, i = 1, \dots, n$ 。令矩阵  $S = [u_1, \dots, u_n]$  由特征向量  $u_1, \dots, u_n$  组

成。由于这些特征向量相互线性无关，矩阵  $\mathbf{S}$  为非奇异矩阵，其逆矩阵  $\mathbf{S}^{-1}$  存在。根据逆矩阵的定义知  $\mathbf{S}^{-1}\mathbf{S} = [\mathbf{S}^{-1}\mathbf{u}_1, \dots, \mathbf{S}^{-1}\mathbf{u}_n] = \mathbf{I}$ 。另外，由  $\mathbf{A}\mathbf{u}_i = \lambda_i\mathbf{u}_i$  易知

$$\mathbf{AS} = [\mathbf{A}\mathbf{u}_1, \dots, \mathbf{A}\mathbf{u}_n] = [\lambda_1\mathbf{u}_1, \dots, \lambda_n\mathbf{u}_n]$$

上式左乘逆矩阵  $\mathbf{S}^{-1}$ ，则有

$$\begin{aligned}\mathbf{S}^{-1}\mathbf{AS} &= [\lambda_1\mathbf{S}^{-1}\mathbf{u}_1, \dots, \lambda_n\mathbf{S}^{-1}\mathbf{u}_n] \\ &= \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} [\mathbf{S}^{-1}\mathbf{u}_1, \dots, \mathbf{S}^{-1}\mathbf{u}_n] \\ &= \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \mathbf{I} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}\end{aligned}$$

即充分性得证。

(2) 必要条件的证明 令矩阵  $\mathbf{A}$  与对角矩阵  $\mathbf{D}$  相似，即  $\mathbf{S}^{-1}\mathbf{AS} = \mathbf{D}$ 。由此得  $\mathbf{AS} = \mathbf{SD}$ 。记  $\mathbf{S} = [s_1, \dots, s_n]$ ,  $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ ，则  $\mathbf{AS} = \mathbf{SD}$  可以写作

$$[\mathbf{As}_1, \dots, \mathbf{As}_n] = [d_1\mathbf{s}_1, \dots, d_n\mathbf{s}_n]$$

立即有

$$\mathbf{As}_i = d_i\mathbf{s}_i, \quad i = 1, \dots, n$$

这说明矩阵  $\mathbf{S}$  的列向量  $\mathbf{s}_i$  是矩阵  $\mathbf{A}$  的特征向量  $\mathbf{u}_i$ ，即  $\mathbf{s}_i = \mathbf{u}_i$ ,  $i = 1, \dots, n$ 。但是，由于矩阵  $\mathbf{S}$  非奇异，所以其所有列向量线性无关，从而知  $\mathbf{u}_1, \dots, \mathbf{u}_n$  线性无关。这就证明了必要条件。 ■

由于一个  $n \times n$  矩阵有  $n$  个不同的特征值时，它的  $n$  个特征向量线性无关，所以定理 7.2.4 给出下面的推论。

**推论 7.2.2** 若  $n \times n$  矩阵  $\mathbf{A}$  有  $n$  个不同的特征值，则  $\mathbf{A}$  是可对角化的。

更一般地，即使矩阵  $\mathbf{A}$  具有多重根，它仍然有可能是可对角化的，因为  $\mathbf{A}$  的  $n$  个特征向量有可能是线性无关的。下面的定理给出了矩阵的所有特征向量线性无关的充分必要条件，从而也是一个矩阵可对角化的充分必要条件。这一定理常被称为可对角化定理 (diagonability theorem)。

**定理 7.2.5**<sup>[444,p.307]</sup> 若矩阵  $\mathbf{A} \in \mathbb{C}^{n \times n}$  的特征值  $\lambda_k$  具有代数多重度  $m_k$ ,  $k = 1, \dots, p$ ，并且  $\sum_{k=1}^p m_k = n$ ，则矩阵  $\mathbf{A}$  具有  $n$  个线性无关的特征向量，当且仅当  $\text{rank}(\mathbf{A} - \lambda_k \mathbf{I}) = n - m_k$ ,  $k = 1, \dots, p$ 。此时， $\mathbf{AU} = \mathbf{U}\Sigma$  中的矩阵  $\mathbf{U}$  是非奇异的，而且  $\mathbf{A}$  可对角化为  $\mathbf{U}^{-1}\mathbf{AU} = \Sigma$ 。

### 7.3 Cayley-Hamilton 定理及其应用

如 7.2 节所述, 一个  $n \times n$  矩阵  $A$  的特征值由特征多项式  $\det(A - \lambda I)$  决定。不仅如此, 特征多项式还与矩阵的求逆、矩阵幂和矩阵指数函数的计算密切相关, 所以有必要对特征多项式作更深入的讨论与分析。

#### 7.3.1 Cayley-Hamilton 定理

Cayley-Hamilton 定理是关于一般矩阵的特征多项式的重要结果。从这一定理出发, 很容易解决矩阵的求逆、矩阵幂和矩阵指数函数的计算等问题。为了引出 Cayley-Hamilton 定理, 先介绍几个与多项式有关的重要概念。

当  $p_n \neq 0$  时,  $n$  称为多项式  $p(x) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0$  的阶数。一个  $n$  阶多项式称为首一多项式 (monic polynomial), 若  $x^n$  的系数等于 1。

若  $p(A) = p_n A^n + p_{n-1} A^{n-1} + \cdots + p_1 A + p_0 I = O$ , 则称  $p(x) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0$  是使矩阵  $A$  零化的多项式, 简称零化多项式 (annihilating polynomial)。

对于一个  $n \times n$  矩阵  $A$ , 令  $m$  是使得幂  $I, A, \dots, A^m$  线性相关的最小整数。于是, 有方程式

$$p_m A^m + p_{m-1} A^{m-1} + \cdots + p_1 A + p_0 I = O_{n \times n} \quad (7.3.1)$$

式中,  $A^m$  的系数不为零。多项式  $p(x) = p_m x^m + p_{m-1} x^{m-1} + \cdots + p_1 x + p_0$  称为矩阵  $A$  的最小多项式。

下面的定理表明, 特征多项式  $p(x) = \det(A - xI)$  是使矩阵  $A_{n \times n}$  零化的多项式。

**定理 7.3.1 (Cayley-Hamilton 定理)** 每一个正方矩阵  $A_{n \times n}$  都满足其特征方程, 即若特征多项式具有式 (7.1.8) 的形式, 则

$$p_n A^n + p_{n-1} A^{n-1} + \cdots + p_1 A + p_0 I = O \quad (7.3.2)$$

式中,  $I$  和  $O$  分别为  $n \times n$  单位矩阵和零矩阵。

**证明** 逆矩阵的定义公式  $B^{-1} = \frac{1}{\det(B)} \text{adj}(B)$  可以等价写作  $B \text{adj}(B) = \det(B)I_n$ , 故有

$$(A - xI) \text{adj}(A - xI) = \det(A - xI)I$$

将  $p(x) = \det(A - xI) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0$  代入上式, 便有

$$(A - xI) \text{adj}(A - xI) = (p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0)I \quad (1)$$

上式表明, 伴随矩阵  $\text{adj}(A - xI)$  必然是一个关于  $x$  的  $n - 1$  次矩阵多项式, 不妨令其为

$$\text{adj}(A - xI) = x^{n-1} B_{n-1} + x^{n-2} B_{n-2} + \cdots + x B_1 + B_0 \quad (2)$$

式中,  $\mathbf{B}_{n-1}, \mathbf{B}_{n-2}, \dots, \mathbf{B}_1$  均为  $n \times n$  常数矩阵。将式(2)代入式(1), 得

$$\begin{aligned} & (\mathbf{A} - x\mathbf{I})(x^{n-1}\mathbf{B}_{n-1} + x^{n-2}\mathbf{B}_{n-2} + \dots + x\mathbf{B}_1 + \mathbf{B}_0) \\ &= (p_n x^n + p_{n-1} x^{n-1} + \dots + p_1 x + p_0) \mathbf{I} \end{aligned}$$

比较上式两边同幂次项  $x^k$  的系数, 即可得到  $n+1$  个方程

$$\begin{aligned} -\mathbf{B}_{n-1} &= p_n \mathbf{I} \\ -\mathbf{B}_{n-2} + \mathbf{A}\mathbf{B}_{n-1} &= p_{n-1} \mathbf{I} \\ &\vdots \\ -\mathbf{B}_0 + \mathbf{A}\mathbf{B}_1 &= p_1 \mathbf{I} \\ -\mathbf{A}\mathbf{B}_0 &= p_0 \mathbf{I} \end{aligned}$$

在上述前  $n$  个方程中, 两边分别左乘矩阵  $\mathbf{A}^n, \mathbf{A}^{n-1}, \dots, \mathbf{A}$ , 然后将所有  $n+1$  个方程相加, 立即有  $\mathbf{O} = p_n \mathbf{A}^n + p_{n-1} \mathbf{A}^{n-1} + \dots + p_1 \mathbf{A} + p_0 \mathbf{I}$ , 这正是所期望的结果。 ■

以上证明是大多数文献采用的传统证明方法。Jain 与 Gunawardena 从矩阵分解的角度, 给出了另外一种证明。对此证明感兴趣的读者可参考文献 [251, p.180]。

Cayley-Hamilton 定理有很多非常有趣和重要的应用。例如, 利用 Cayley-Hamilton 定理, 也能够直接证明两个相似矩阵具有相同的特征值。

考查两个相似矩阵的特征多项式。令  $\mathbf{B} = \mathbf{S}^{-1} \mathbf{A} \mathbf{S}$  是  $\mathbf{A}$  的相似矩阵, 并且已知矩阵  $\mathbf{A}$  的特征多项式  $p(\mathbf{x}) = \det(\mathbf{A} - x\mathbf{I}) = p_n x^n + p_{n-1} x^{n-1} + \dots + p_1 x + p_0$ 。根据 Cayley-Hamilton 定理知  $p(\mathbf{A}) = p_n \mathbf{A}^n + p_{n-1} \mathbf{A}^{n-1} + \dots + p_1 \mathbf{A} + p_0 \mathbf{I} = \mathbf{O}$ 。

对于相似矩阵  $\mathbf{B}$ , 由于

$$\mathbf{B}^k = (\mathbf{S}^{-1} \mathbf{A} \mathbf{S})(\mathbf{S}^{-1} \mathbf{A} \mathbf{S}) \cdots (\mathbf{S}^{-1} \mathbf{A} \mathbf{S}) = \mathbf{S}^{-1} \mathbf{A}^k \mathbf{S}$$

故有

$$\begin{aligned} p(\mathbf{B}) &= p_n \mathbf{B}^n + p_{n-1} \mathbf{B}^{n-1} + \dots + p_1 \mathbf{B} + p_0 \mathbf{I} \\ &= p_n \mathbf{S}^{-1} \mathbf{A}^n \mathbf{S} + p_{n-1} \mathbf{S}^{-1} \mathbf{A}^{n-1} \mathbf{S} + \dots + p_1 \mathbf{S}^{-1} \mathbf{A} \mathbf{S} + p_0 \mathbf{I} \\ &= \mathbf{S}^{-1} (p_n \mathbf{A}^n + p_{n-1} \mathbf{A}^{n-1} + \dots + p_1 \mathbf{A} + p_0 \mathbf{I}) \mathbf{S} \\ &= \mathbf{S}^{-1} p(\mathbf{A}) \mathbf{S} \\ &= \mathbf{O} \end{aligned}$$

在得到最后一个式子时, 代入了 Cayley-Hamilton 定理的结果  $p(\mathbf{A}) = \mathbf{O}$ 。换言之, 两个相似矩阵  $\mathbf{A} \sim \mathbf{B}$  具有相同的特征多项式, 从而它们具有相同的特征值。

下面再介绍 Cayley-Hamilton 定理的几个重要应用。

### 7.3.2 逆矩阵和广义逆矩阵的计算

若矩阵  $A_{n \times n}$  非奇异, 则用  $A^{-1}$  右乘(或左乘)式(7.3.2)两边, 立即有

$$p_n A^{n-1} + p_{n-1} A^{n-2} + \cdots + p_2 A + p_1 I + p_0 A^{-1} = O$$

由此即可得到逆矩阵的计算公式

$$A^{-1} = -\frac{1}{p_0}(p_n A^{n-1} + p_{n-1} A^{n-2} + \cdots + p_2 A + p_1 I) \quad (7.3.3)$$

**例 7.3.1** 已知矩阵

$$A = \begin{bmatrix} 1 & 5 \\ 4 & 6 \end{bmatrix}$$

其特征多项式为

$$\det(A - xI) = \begin{vmatrix} 1-x & 5 \\ 4 & 6-x \end{vmatrix} = (1-x)(6-x) - 5 \times 4 = x^2 - 7x - 14$$

即  $p_0 = -14$ ,  $p_1 = -7$ ,  $p_2 = 1$ 。将这些值代入式(7.3.3), 立即得

$$A^{-1} = \frac{1}{14}(A - 7I) = \frac{1}{14} \left( \begin{bmatrix} 1 & 5 \\ 4 & 6 \end{bmatrix} - 7 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) = \begin{bmatrix} -\frac{3}{7} & \frac{5}{14} \\ \frac{2}{7} & -\frac{1}{14} \end{bmatrix}$$

**例 7.3.2** 由矩阵

$$A = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

得矩阵的二次幂

$$A^2 = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 5 \\ 0 & 4 & 0 \\ 0 & 0 & 9 \end{bmatrix}$$

和特征多项式

$$\begin{aligned} \det(A - xI) &= \begin{vmatrix} 2-x & 0 & 1 \\ 0 & 2-x & 0 \\ 0 & 0 & 3-x \end{vmatrix} = (2-x)^2(3-x) \\ &= -x^3 + 7x^2 - 16x + 12 \end{aligned}$$

即  $p_0 = 12$ ,  $p_1 = -16$ ,  $p_2 = 7$ ,  $p_3 = -1$ 。将这些值连同矩阵  $A$ ,  $A^2$  一起代入矩阵求逆公式(7.3.3), 则有

$$\begin{aligned} A^{-1} &= -\frac{1}{12}(-A^2 + 7A - 16I) \\ &= -\frac{1}{12} \left( - \begin{bmatrix} 4 & 0 & 5 \\ 0 & 4 & 0 \\ 0 & 0 & 9 \end{bmatrix} + 7 \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} - 16 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \\ &= \frac{1}{6} \begin{bmatrix} 3 & 0 & -1 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{bmatrix} \end{aligned}$$

Cayley-Hamilton 定理还可用于求任意一个复矩阵的广义逆矩阵。这一结果是 Decell 于 1965 年得到的<sup>[131]</sup>。

**定理 7.3.2** 令矩阵  $\mathbf{A}$  是任意一个  $m \times n$  矩阵，并且令

$$f(\lambda) = (-1)^m(a_0\lambda^m + a_1\lambda^{m-1} + \cdots + a_{m-1}\lambda + a_m), \quad a_0 = 1 \quad (7.3.4)$$

是矩阵乘积  $\mathbf{AA}^H$  的特征多项式  $\det(\mathbf{AA}^H - \lambda\mathbf{I})$ 。若  $k$  是满足  $a_k \neq 0$  的最大整数，则  $\mathbf{A}$  的广义逆矩阵由

$$\mathbf{A}^\dagger = -a_k^{-1}\mathbf{A}^H \left[ (\mathbf{AA}^H)^{k-1} + a_1(\mathbf{AA}^H)^{k-2} + \cdots + a_{k-2}(\mathbf{AA}^H) + a_{k-1}\mathbf{I} \right] \quad (7.3.5)$$

确定。当  $k=0$  是使  $a_k \neq 0$  的最大整数时，广义逆矩阵  $\mathbf{A}^\dagger = \mathbf{O}$ 。

**证明** 参见文献 [131]。

在有些文献（例如文献 [460]）中，称上述定理为 Decell 定理。注意，Decell 定理中的整数  $k$  就是矩阵  $\mathbf{A}$  的秩，即  $k = \text{rank}(\mathbf{A})$ 。

根据上述定理，Decell 提出了计算 Moore-Penrose 逆矩阵的下列方法<sup>[131]</sup>。

(1) 构造矩阵序列  $\mathbf{A}_0, \mathbf{A}_1, \dots, \mathbf{A}_k$

$$\begin{aligned} \mathbf{A}_0 &= \mathbf{O}, & -1 &= q_0, & \mathbf{B}_0 &= \mathbf{I} \\ \mathbf{A}_1 &= \mathbf{AA}^H, & \text{tr}(\mathbf{A}_1) &= q_1, & \mathbf{B}_1 &= \mathbf{A}_1 - q_1\mathbf{I} \\ \mathbf{A}_2 &= \mathbf{AA}^H\mathbf{B}_1, & \frac{\text{tr}(\mathbf{A}_2)}{2} &= q_2, & \mathbf{B}_2 &= \mathbf{A}_2 - q_2\mathbf{I} \\ &\vdots & &\vdots & &\vdots \\ \mathbf{A}_{k-1} &= \mathbf{AA}^H\mathbf{B}_{k-2}, & \frac{\text{tr}(\mathbf{A}_{k-1})}{k-1} &= q_{k-1}, & \mathbf{B}_{k-1} &= \mathbf{A}_{k-1} - q_{k-1}\mathbf{I} \\ \mathbf{A}_k &= \mathbf{AA}^H\mathbf{B}_{k-1}, & \frac{\text{tr}(\mathbf{A}_k)}{k} &= q_k, & \mathbf{B}_k &= \mathbf{A}_k - q_k\mathbf{I} \end{aligned}$$

Faddeev 证明了<sup>[161, pp.260~265]</sup>，按此方法构造的系数  $q_i = -a_i, i = 1, \dots, k$ 。

(2) 计算 Moore-Penrose 逆矩阵

$$\begin{aligned} \mathbf{A}^\dagger &= -a_k^{-1}\mathbf{A}^H \left[ (\mathbf{AA}^H)^{k-1} + a_1(\mathbf{AA}^H)^{k-2} + \cdots + a_{k-2}(\mathbf{AA}^H) + a_{k-1}\mathbf{I} \right] \\ &= -a_k^{-1}\mathbf{A}^H\mathbf{B}_{k-1} \end{aligned} \quad (7.3.6)$$

### 7.3.3 矩阵幂的计算

给定任意一个矩阵  $\mathbf{A}_{n \times n}$  和一个整数  $k$ ，称  $\mathbf{A}^k$  是矩阵  $\mathbf{A}$  的  $k$  次幂。如果  $k$  比较大，显然矩阵幂  $\mathbf{A}^k$  的计算是一件很繁琐的事。幸运的是，Cayley-Hamilton 定理为这个问题提供了一种简单的解决方法。

考查多项式除法  $f(x)/g(x)$ ，其中， $g(x) \neq 0$ 。根据 Euclidean 除法知，存在两个多项式  $q(x)$  和  $r(x)$ ，使得

$$f(x) = g(x)q(x) + r(x)$$

式中,  $q(x)$  和  $r(x)$  分别称为商和余项, 并且余项  $r(x)$  的阶数小于  $p(x)$  的阶数或  $r(x) = 0$ 。

令矩阵  $A$  的特征多项式为

$$p(x) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0$$

则对于任何一个  $x$ , 其  $K$  次幂

$$x^K = p(x)q(x) + r(x) \quad (7.3.7)$$

当  $x$  是特征方程  $p(x) = 0$  的一个根时, 上式变为

$$x^K = r(x) = r_0 + r_1 x + \cdots + r_{n-1} x^{n-1} \quad (7.3.8)$$

因为  $r(x)$  的阶数小于  $p(x)$  的阶数  $n$ 。

用  $A$  代替标量  $x$ , 则式 (7.3.7) 变为

$$A^K = p(A)q(A) + r(A) \quad (7.3.9)$$

根据 Cayley-Hamilton 定理知, 若  $p(x)$  是矩阵  $A$  的特征多项式, 则  $p(A) = O$ 。因此, 式 (7.3.9) 简化为

$$A^K = r(A) = r_0 I + r_1 A + \cdots + r_{n-1} A^{n-1} \quad (7.3.10)$$

式 (7.3.10) 给出了计算矩阵幂  $A^K$  的方法。

**算法 7.3.1** 矩阵幂的计算 [251, p.172]

步骤 1 构造特征多项式

$$p(x) = \det(A - xI) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0 \quad (7.3.11)$$

步骤 2 计算特征方程  $p(x) = 0$  的  $n$  个特征根即特征值  $\lambda_1, \lambda_2, \dots, \lambda_n$ 。

步骤 3 将特征值代入式 (7.3.8), 得到一组线性方程

$$\left. \begin{array}{l} r_0 + \lambda_1 r_1 + \cdots + \lambda_1^{n-1} r_{n-1} = \lambda_1^K \\ r_0 + \lambda_2 r_1 + \cdots + \lambda_2^{n-1} r_{n-1} = \lambda_2^K \\ \vdots \\ r_0 + \lambda_n r_1 + \cdots + \lambda_n^{n-1} r_{n-1} = \lambda_n^K \end{array} \right\} \quad (7.3.12)$$

解之, 得  $r_0, r_1, \dots, r_{n-1}$ 。

步骤 4 计算矩阵幂

$$A^K = r_0 I + r_1 A + \cdots + r_{n-1} A^{n-1}$$

下面举例加以说明。

例 7.3.3 已知

$$\mathbf{A} = \begin{bmatrix} 1 & 1/2 \\ 2 & 1 \end{bmatrix}$$

计算  $\mathbf{A}^{731}$ 。

解 利用式 (7.3.11) 构造特征多项式

$$p(x) = \det(\mathbf{A} - x\mathbf{I}) = \begin{vmatrix} 1-x & 1/2 \\ 2 & 1-x \end{vmatrix} = x^2 - 2x$$

令  $p(x) = 0$ , 求出  $\mathbf{A}$  的特征值  $\lambda_1 = 0$  和  $\lambda_2 = 2$ 。将特征值代入式 (7.3.12) 得线性方程组

$$r_0 + 0r_1 = 0^{731}$$

$$r_0 + 2r_1 = 2^{731}$$

解之, 得  $r_0 = 0$  和  $r_1 = 2^{730}$ 。计算矩阵幂, 得

$$\mathbf{A}^{731} = 2^{730}\mathbf{A} = 2^{730} \begin{bmatrix} 1 & 1/2 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 2^{730} & 2^{729} \\ 2^{731} & 2^{730} \end{bmatrix}$$

#### 7.3.4 矩阵指数函数的计算

类似于标量指数函数

$$e^{at} = 1 + at + \frac{1}{2!}a^2t^2 + \cdots + \frac{1}{k!}a^kt^k + \cdots$$

对一个已知矩阵  $\mathbf{A}$ , 可以定义矩阵指数函数 (matrix exponential function)

$$e^{\mathbf{At}} = \mathbf{I} + \mathbf{At} + \frac{1}{2!}\mathbf{A}^2t^2 + \cdots + \frac{1}{k!}\mathbf{A}^kt^k + \cdots \quad (7.3.13)$$

矩阵指数函数可以用来表示一阶微分方程的解。在工程应用中, 经常会遇到线性一阶微分方程组

$$\dot{\mathbf{x}}(t) = \mathbf{Ax}(t), \quad \mathbf{x}(0) = \mathbf{x}_0$$

其中,  $\mathbf{A}$  为常数矩阵。上述一阶微分方程组的解可以写作  $\mathbf{x}(t) = e^{\mathbf{At}}\mathbf{x}_0$ 。因此, 线性一阶微分方程组的求解等价于计算矩阵指数函数  $e^{\mathbf{At}}$ 。

利用 Cayley-Hamilton 定理, 可以证明  $n$  阶线性矩阵微分方程的解的唯一性。

**定理 7.3.3**<sup>[310]</sup> 令  $\mathbf{A}$  是一个  $n \times n$  常数矩阵, 其特征多项式为

$$p(\lambda) = \det(\lambda\mathbf{I} - \mathbf{A}) = \lambda^n + c_{n-1}\lambda^{n-1} + \cdots + c_1\lambda + c_0 \quad (7.3.14)$$

并且  $n$  阶矩阵微分方程

$$\Phi^{(n)}(t) + c_{n-1}\Phi^{(n-1)}(t) + \cdots + c_1\Phi'(t) + c_0\Phi(t) = \mathbf{O} \quad (7.3.15)$$

满足初始条件

$$\Phi(0) = \mathbf{I}, \quad \Phi'(0) = \mathbf{A}, \quad \Phi''(0) = \mathbf{A}^2, \quad \dots, \quad \Phi^{(n-1)}(0) = \mathbf{A}^{n-1} \quad (7.3.16)$$

则  $\Phi(t) = e^{\mathbf{A}t}$  是  $n$  阶矩阵微分方程式 (7.3.15) 的唯一解。

定理 7.3.3 保证了矩阵微分方程的唯一解的存在性。下面的定理给出了求这一唯一解的方法。

**定理 7.3.4** [310] 令  $\mathbf{A}$  是一个  $n \times n$  常数矩阵, 其特征多项式为

$$p(\lambda) = \det(\lambda\mathbf{I} - \mathbf{A}) = \lambda^n + c_{n-1}\lambda^{n-1} + \cdots + c_1\lambda + c_0$$

则满足初始条件式 (7.3.16) 的矩阵微分方程式 (7.3.15) 的解由

$$\Phi(t) = e^{\mathbf{A}t} = x_1(t)\mathbf{I} + x_2(t)\mathbf{A} + x_3(t)\mathbf{A}^2 + \cdots + x_n(t)\mathbf{A}^{n-1} \quad (7.3.17)$$

给出, 式中,  $x_k(t)$  是  $n$  阶标量微分方程

$$x^{(n)}(t) + c_{n-1}x^{(n-1)}(t) + \cdots + c_1x'(t) + c_0x(t) = 0 \quad (7.3.18)$$

满足初始条件

$$\left. \begin{array}{l} x_1(0) = 1 \\ x'_1(0) = 0 \\ \vdots \\ x_1^{(n-1)}(0) = 0 \end{array} \right\}, \quad \left. \begin{array}{l} x_2(0) = 0 \\ x'_2(0) = 1 \\ \vdots \\ x_2^{(n-1)}(0) = 0 \end{array} \right\}, \quad \dots, \quad \left. \begin{array}{l} x_n(0) = 0 \\ x'_n(0) = 0 \\ \vdots \\ x_n^{(n-1)}(0) = 1 \end{array} \right\}$$

的解。

式 (7.3.17) 给出了计算矩阵指数函数  $e^{\mathbf{A}t}$  的一种有效方法。

**例 7.3.4** 已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

利用定理 7.3.4 求  $e^{\mathbf{A}t}$ 。

解 由特征方程

$$\det(\lambda\mathbf{I} - \mathbf{A}) = \begin{vmatrix} \lambda - 1 & 0 & -1 \\ 0 & \lambda - 1 & 0 \\ 0 & 0 & \lambda - 2 \end{vmatrix} = (\lambda - 1)^2(\lambda - 2) = 0$$

求得矩阵  $\mathbf{A}$  的特征值  $\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 2$ , 即特征值 1 的多重度为 2。

由定理 7.3.4 知, 矩阵指数函数

$$e^{\mathbf{A}t} = x_1(t)\mathbf{I} + x_2(t)\mathbf{A} + x_3(t)\mathbf{A}^2$$

式中,  $x_1(t), x_2(t), x_3(t)$  是三阶标量微分方程

$$x'''(t) + c_2x''(t) + c_1x'(t) + c_0x(t) = 0$$

满足初始条件

$$\left. \begin{array}{l} x_1(0) = 1 \\ x'_1(0) = 0 \\ x''_1(0) = 0 \end{array} \right\}, \quad \left. \begin{array}{l} x_2(0) = 0 \\ x'_2(0) = 1 \\ x''_2(0) = 0 \end{array} \right\}, \quad \left. \begin{array}{l} x_3(0) = 0 \\ x'_3(0) = 0 \\ x''_3(0) = 1 \end{array} \right\}$$

的解。

由矩阵  $A$  的特征值  $\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 2$  知, 标量微分方程  $x'''(t) + c_2x''(t) + c_1x'(t) + c_0x(t) = 0$  的通解由

$$x(t) = a_1te^t + a_2e^t + a_3e^{2t}$$

给出。

由上述通解公式易得以下结果:

(1) 将初始条件  $x_1(0) = 1, x'_1(0) = 0, x''_1(0) = 0$  代入通解公式, 得

$$\left\{ \begin{array}{l} a_2 + a_3 = 1 \\ a_1 + a_2 + 2a_3 = 0 \\ 2a_1 + a_2 + 4a_3 = 0 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} a_1 = -2 \\ a_2 = 0 \\ a_3 = 1 \end{array} \right.$$

即满足初始条件  $x_1(0) = 1, x'_1(0) = 0, x''_1(0) = 0$  的特解为

$$x_1(t) = -2te^t + e^{2t}$$

(2) 由初始条件  $x_2(0) = 0, x'_2(0) = 1, x''_2(0) = 0$  得

$$\left\{ \begin{array}{l} a_2 + a_3 = 0 \\ a_1 + a_2 + 2a_3 = 1 \\ 2a_1 + a_2 + 4a_3 = 0 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} a_1 = 1 \\ a_2 = 2 \\ a_3 = -1 \end{array} \right.$$

从而得满足初始条件  $x_2(0) = 0, x'_2(0) = 1, x''_2(0) = 0$  的特解为

$$x_2(t) = te^t + 2e^t - e^{2t}$$

(3) 由初始条件  $x_3(0) = 0, x'_3(0) = 0, x''_3(0) = 1$  得

$$\left\{ \begin{array}{l} a_2 + a_3 = 0 \\ a_1 + a_2 + 2a_3 = 0 \\ 2a_1 + a_2 + 4a_3 = 1 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} a_1 = -1 \\ a_2 = -1 \\ a_3 = 1 \end{array} \right.$$

换言之, 满足初始条件  $x_3(0) = 0, x'_3(0) = 0, x''_3(0) = 1$  的特解为

$$x_3(t) = -te^t - e^t + e^{2t}$$

计算知

$$\mathbf{A}^2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix}$$

因此,由定理 7.3.4 可求得

$$\begin{aligned} e^{\mathbf{At}} &= x_1(t)\mathbf{I} + x_2(t)\mathbf{A} + x_3(t)\mathbf{A}^2 \\ &= (-2te^t + e^{2t}) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + (te^t + 2e^t - e^{2t}) \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} + \\ &\quad (-te^t - e^t + e^{2t}) \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix} \end{aligned}$$

即有

$$e^{\mathbf{At}} = \begin{bmatrix} -2te^t + e^t + e^{2t} & 0 & -2te^t - e^t + 2e^{2t} \\ 0 & -2te^t + e^t + e^{2t} & 0 \\ 0 & 0 & -4te^t + 3e^{2t} \end{bmatrix}$$

总结以上讨论,可以得出结论:Cayley-Hamilton 定理为矩阵求逆、矩阵幂的计算、线性微分方程的求解(或等价于矩阵指数函数的计算)提供了非常有效的工具。

## 7.4 特征值分解的几种典型应用

特征值分解有着广泛的应用。本节以信号处理和模式识别中的问题为例,介绍几个典型的应用:标准正交变换、迷向圆变换、Pisarenko 谐波分解、主分量分析。

### 7.4.1 标准正交变换与迷向圆变换

给定一组彼此相关的随机变量,常常希望通过线性变换,把它变成另外一组统计不相关的随机变量。甚至更进一步,希望变换后的一组统计不相关随机变量各个分量还具有单位方差。这两个任务可以通过标准正交变换和迷向圆变换分别完成。

令  $\mathbf{x}$  为一  $m \times 1$  随机向量,其均值向量为  $\mathbf{m}_x$ ,协方差矩阵为  $\mathbf{C}_x$ 。

首先,使用线性变换  $\mathbf{x}_0 = \mathbf{x} - \mathbf{m}_x$  将  $\mathbf{x}$  变成零均值的随机向量  $\mathbf{x}_0$ 。此时,随机向量  $\mathbf{x}_0$  的自相关矩阵与  $\mathbf{x}$  的协方差矩阵相同,即  $\mathbf{R}_{\mathbf{x}_0} = \mathbf{C}_x$ 。

#### 1. 标准正交变换

令  $\mathbf{C}_x$  的特征值分解为

$$\mathbf{C}_x = \mathbf{U}_x \boldsymbol{\Sigma}_x \mathbf{U}_x^H \quad (7.4.1)$$

利用  $\mathbf{U}_x^H$  对  $\mathbf{x}_0$  进行线性变换,其结果为

$$\mathbf{w} = \mathbf{U}_x^H \mathbf{x}_0 = \mathbf{U}_x^H (\mathbf{x} - \mathbf{m}_x) \quad (7.4.2)$$

于是, 变换结果  $\mathbf{w}$  的均值向量

$$\mathbf{m}_w = \mathbf{U}_x^H \mathbf{E}\{\mathbf{x}_0\} = \mathbf{U}_x^H \mathbf{E}\{\mathbf{x} - \mathbf{m}_x\} = \mathbf{0} \quad (7.4.3)$$

且  $\mathbf{w}$  的协方差矩阵

$$\begin{aligned} \mathbf{C}_w &= \mathbf{R}_w = \mathbf{E}\{\mathbf{w}\mathbf{w}^H\} = \mathbf{E}\{\mathbf{U}_x^H \mathbf{x}_0 \mathbf{x}_0^H \mathbf{U}_x\} \\ &= \mathbf{U}_x^H \mathbf{R}_{x_0} \mathbf{U}_x = \mathbf{U}_x^H \mathbf{C}_x \mathbf{U}_x = \boldsymbol{\Sigma}_x \end{aligned} \quad (7.4.4)$$

由于  $\boldsymbol{\Sigma}_x$  是对角矩阵, 故  $\mathbf{C}_w$  也是对角矩阵。

总结以上讨论, 使用特征值分解的线性变换  $\mathbf{w} = \mathbf{U}_x \mathbf{x}_0 = \mathbf{U}_x (\mathbf{x} - \mathbf{m}_x)$  具有以下有趣的性质 [332]:

(1) 随机向量  $\mathbf{w}$  具有零均值, 各个分量彼此统计不相关 (因而正交)。进一步地, 若  $\mathbf{x}$  是具有均值向量  $\mathbf{m}_x$  和协方差矩阵  $\mathbf{C}_x$  的正态或高斯分布  $N(\mathbf{m}_x, \mathbf{C}_x)$ , 则  $\mathbf{w}$  是一个具有零均值向量和协方差矩阵为对角矩阵的正态分布  $N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ , 即它的各个分量相互统计独立 (注: 对于具有零均值向量的正态或高斯随机向量, 正交、统计不相关和统计独立三者等价)。

- (2) 随机变量  $w_i (i = 1, 2, \dots, m)$  的方差等于协方差矩阵  $\mathbf{C}_x$  的特征值。
- (3) 由于线性变换矩阵  $\mathbf{U}_x$  是标准正交矩阵, 所以线性变换  $\mathbf{w} = \mathbf{U}_x^H \mathbf{x}_0$  称为标准正交变换 (orthonormal transformation), 且距离函数的平方

$$d^2(\mathbf{x}_0) \stackrel{\text{def}}{=} \mathbf{x}_0^H \mathbf{C}_{x_0}^{-1} \mathbf{x}_0 = (\mathbf{x}^H \mathbf{U}_x) \boldsymbol{\Sigma}_x^{-1} \mathbf{U}_x^H \mathbf{x} = \mathbf{x}^H \mathbf{C}_x^{-1} \mathbf{x} = d^2(\mathbf{x}) \quad (7.4.5)$$

在标准正交变换下保持不变。距离测度  $d^2(\mathbf{x}) = \mathbf{x}^H \mathbf{C}_x^{-1} \mathbf{x}$  称为 Mahalanobis 距离。在正态随机向量的情况下, Mahalanobis 距离与对数似然函数有关。

## 2. 迷向圆变换

在上面的标准正交变换中, 线性变换  $\mathbf{w} = \mathbf{U}_x^H \mathbf{x}_0$  的自相关矩阵 (与协方差矩阵相等)  $\mathbf{R}_w$  为对角矩阵, 但不是单位矩阵  $\mathbf{I}$ 。要使  $\mathbf{w}$  的自相关矩阵为单位矩阵, 就需要对  $\mathbf{w}$  再作另一个线性变换

$$\mathbf{y} = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{w} = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{U}_x^H \mathbf{x}_0 = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{U}_x^H (\mathbf{x} - \mathbf{m}_x) \quad (7.4.6)$$

由上式和式 (7.4.4), 得

$$\mathbf{R}_y = \mathbf{E}\{\mathbf{y}\mathbf{y}^H\} = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{C}_w \boldsymbol{\Sigma}_x^{-1/2} = \boldsymbol{\Sigma}_x^{-1/2} \boldsymbol{\Sigma}_x \boldsymbol{\Sigma}_x^{-1/2} = \mathbf{I}$$

线性变换  $\mathbf{y} = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{w} = \boldsymbol{\Sigma}_x^{-1/2} \mathbf{U}_x^H \mathbf{x}_0$  称为迷向圆变换 (isotropic circular transformation), 因为向量  $\mathbf{y}$  的所有分量都是零均值的、具有单位方差的统计不相关随机变量。图 7.4.1 画出了二维情况下迷向圆变换的几何解释 [332]。

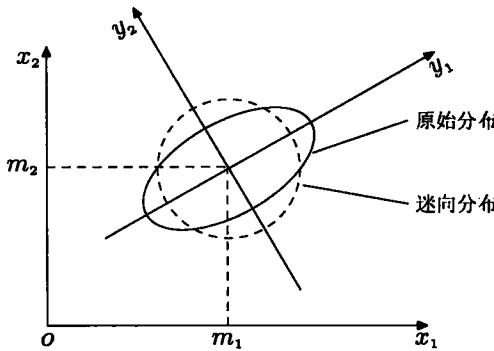


图 7.4.1 迷向圆变换的几何解释

图 7.4.1 清楚地表明，在迷向圆变换中不仅存在坐标轴的平移和旋转，而且还存在坐标轴的伸缩。结果是，向量  $\mathbf{y}$  的分布变成了圆周型的分布，在所有方向上都相同，即分布是方向不变的。这样一种分布称为各向同性分布或迷向分布 (isotropic distribution)。

从图 7.4.1 可以看出，以二维向量  $\mathbf{x} = \overrightarrow{AB} = (a, b)$  为例，经过标准正交变换  $\mathbf{w} = \mathbf{U}_x(\mathbf{x} - \mathbf{m}_x)$  后，描述  $\mathbf{w}$  的新坐标系具有以下几何特点：

- (1) 新坐标系是描述向量  $\mathbf{x}$  的原坐标系的平移，即原点从  $(0, 0)$  平移至  $(a, b)$ 。
- (2) 新坐标系是原坐标系的旋转。

总结以上讨论，可以得出迷向圆变换所具有的重要性质：

- (1) 经过迷向圆变换之后，自相关矩阵变成单位矩阵。这意味着，迷向圆变换得到的随机向量的所有分量具有单位方差，而且彼此统计不相关。
- (2) 对具有零均值向量的随机向量  $\mathbf{x}_0$  所进行的迷向圆变换  $\mathbf{y} = \Sigma_x^{-1/2} \mathbf{U}_x^H \mathbf{x}_0$  的矩阵 (称为迷向圆变换矩阵)  $\mathbf{A} = \Sigma_x^{-1/2} \mathbf{U}_x^H$  是正交的，但不是标准正交的。
- (3) Mahalanobis 距离  $d^2(\mathbf{x}_0)$  在迷向圆变换下不能保持不变，即  $d^2(\mathbf{y}) \neq d^2(\mathbf{x}_0)$ 。

下面从信号处理的观点观看标准正交变换和迷向圆变换。

若将  $m$  个相关的零均值随机过程  $\{x_1(n), x_2(n), \dots, x_m(n)\}$  视为一随机向量  $\mathbf{x}(n) = [x_1(n), x_2(n), \dots, x_m(n)]^T$ ，则随机向量  $\mathbf{x}(n)$  的标准正交变换等价于将  $m$  个相关的随机过程  $x_1(n), x_2(n), \dots, x_m(n)$  分别变换为白 (色) 噪声过程，简称 (预) 白化。在这类情况下， $m$  个白噪声过程具有不同的方差。

对随机向量  $\mathbf{x}(n)$  进行迷向圆变换时， $m$  个相关随机过程  $x_1(n), x_2(n), \dots, x_m(n)$  分别白化为具有单位方差的白噪声，常称标准白化。

因此，若将数学语言翻译成信号处理语言，即有： $m$  维随机向量的标准正交变换和迷向圆变换分别是  $m$  个随机过程的白化和标准白化。白化或标准白化是信号处理、模式识别和机器学习等中经常使用的数据预处理手段。

### 7.4.2 Pisarenko 谐波分解

谐波过程在很多工程应用中会经常遇到，并需要确定这些谐波的频率和功率（合称谐波恢复）。谐波恢复的关键任务是估计谐波的个数及频率。下面介绍谐波恢复的 Pisarenko 谐波分解方法，它是俄罗斯数学家 Pisarenko 提出的<sup>[409]</sup>。

考虑由  $p$  个实正弦波组成的谐波过程

$$x(n) = \sum_{i=1}^p A_i \sin(2\pi f_i n + \theta_i) \quad (7.4.7)$$

当相位  $\theta_i$  为常数时，上述谐波过程是一确定性过程，它是非平稳的。为了保证谐波过程的平稳性，通常假定相位  $\theta_i$  是在  $[-\pi, \pi]$  内均匀分布的随机数。此时，谐波过程是一随机过程。

谐波过程可以使用差分方程描述。先考虑单个正弦波的情况。为简单计，令谐波信号  $x(n) = \sin(2\pi f n + \theta)$ 。回忆三角函数恒等式

$$\sin(2\pi f n + \theta) + \sin[2\pi f(n-2) + \theta] = 2 \cos(2\pi f) \sin[2\pi f(n-1) + \theta]$$

若将  $x(n) = \sin(2\pi f n + \theta)$  代入上式，便得到二阶差分方程

$$x(n) - 2 \cos(2\pi f) x(n-1) + x(n-2) = 0$$

对上式作  $z$  变换，得

$$[1 - 2 \cos(2\pi f) z^{-1} + z^{-2}] X(z) = 0$$

于是，得到特征多项式

$$1 - 2 \cos(2\pi f) z^{-1} + z^{-2} = 0$$

它有一对共轭复数根，即

$$z = \cos(2\pi f) \pm j \sin(2\pi f) = e^{\pm j 2\pi f}$$

注意，共轭根的模为 1，即  $|z_1| = |z_2| = 1$ 。由特征多项式的根可决定正弦波的频率，即有

$$f_i = \arctan[\text{Im}(z_i)/\text{Re}(z_i)]/2\pi \quad (7.4.8)$$

通常，只取正的频率。显然，如果  $p$  个实的正弦波信号没有重复频率的话，则这  $p$  个频率应该由特征多项式

$$\prod_{i=1}^p (z - z_i)(z - z_i^*) = \sum_{i=0}^{2p} a_i z^{2p-i} = 0$$

或

$$1 + a_1 z^{-1} + \cdots + a_{2p-1} z^{-(2p-1)} + a_{2p} z^{-2p} = 0 \quad (7.4.9)$$

的根决定。易知，这些根的模全部等于 1。由于所有根都是以共轭对的形式出现，所以特征多项式 (7.4.9) 的系数存在对称性，即

$$a_i = a_{2p-i}, \quad i = 0, 1, \dots, p \quad (7.4.10)$$

与式 (7.4.10) 对应的差分方程为

$$x(n) + \sum_{i=1}^{2p} a_i x(n-i) = 0 \quad (7.4.11)$$

正弦波过程一般是在加性白噪声中被观测的，设加性白噪声为  $e(n)$ ，即观测过程

$$y(n) = x(n) + e(n) = \sum_{i=1}^p A_i \sin(2\pi f_i n + \theta_i) + e(n) \quad (7.4.12)$$

式中， $e(n) \sim N(0, \sigma_e^2)$  为高斯白噪声，它与正弦波信号  $x(n)$  统计独立。将  $x(n) = y(n) - e(n)$  代入式 (7.4.11)，立即得到白噪声中的正弦波过程所满足的差分方程

$$y(n) + \sum_{i=1}^{2p} a_i y(n-i) = e(n) + \sum_{i=1}^{2p} a_i e(n-i) \quad (7.4.13)$$

这是一个特殊的自回归-滑动平均 (ARMA) 过程，不仅自回归 (AR) 阶数与滑动平均 (MA) 阶数相等，而且 AR 参数也与 MA 参数完全相同。

现在推导这一特殊 ARMA 过程的 AR 参数满足的法方程。为此，定义向量

$$\left. \begin{aligned} \mathbf{y}(n) &= [y(n), y(n-1), \dots, y(n-2p)]^T \\ \mathbf{w} &= [1, a_1, \dots, a_{2p}]^T \\ \mathbf{e}(n) &= [e(n), e(n-1), \dots, e(n-2p)]^T \end{aligned} \right\} \quad (7.4.14)$$

于是，式 (7.4.13) 可写成

$$\mathbf{y}^T(n)\mathbf{w} = \mathbf{e}^T(n)\mathbf{w} \quad (7.4.15)$$

用向量  $\mathbf{y}(n)$  左乘式 (7.4.15)，并取数学期望，即得

$$\mathbb{E}\{\mathbf{y}(n)\mathbf{y}^T(n)\}\mathbf{w} = \mathbb{E}\{\mathbf{y}(n)\mathbf{e}^T(n)\}\mathbf{w} \quad (7.4.16)$$

令  $R_y(k) = \mathbb{E}\{y(n+k)y(n)\}$  表示观测数据  $y(n)$  的自相关函数，则

$$\mathbb{E}\{\mathbf{y}(n)\mathbf{y}^T(n)\} = \begin{bmatrix} R_y(0) & R_y(-1) & \cdots & R_y(-2p) \\ R_y(1) & R_y(0) & \cdots & R_y(-2p+1) \\ \vdots & \vdots & \ddots & \vdots \\ R_y(2p) & R_y(2p-1) & \cdots & R_y(0) \end{bmatrix} \stackrel{\text{def}}{=} \mathbf{R}$$

$$\mathbb{E}\{\mathbf{y}(n)\mathbf{e}^T(n)\} = \mathbb{E}\{[\mathbf{x}(n) + \mathbf{e}(n)]\mathbf{e}^T(n)\} = \mathbb{E}\{\mathbf{e}(n)\mathbf{e}^T(n)\} = \sigma_e^2 \mathbf{I}$$

其中，使用了  $x(n)$  与  $e(n)$  统计独立的假设。将以上两个关系式代入式 (7.4.16)，便得到一个重要的法方程

$$\mathbf{R}\mathbf{w} = \sigma_e^2 \mathbf{w} \quad (7.4.17)$$

这表明,  $\sigma_e^2$  是观测过程  $\{y(n)\}$  的自相关矩阵  $\mathbf{R} = E\{\mathbf{y}(n)\mathbf{y}^T(n)\}$  的特征值, 而特征多项式的系数向量  $\mathbf{w}$  是对应于该特征值的特征向量。这就是 Pisarenko 谐波分解方法的理论基础。

注意, 自相关矩阵  $\mathbf{R}$  的特征值  $\sigma_e^2$  是噪声  $e(n)$  的方差, 其他特征值则与各个谐波信号的功率对应。当信噪比较大时, 特征值  $\sigma_e^2$  明显小于其他特征值。因此, Pisarenko 谐波分解启迪我们, 谐波恢复问题可以转化为自相关矩阵  $\mathbf{R}$  的特征值分解: 谐波过程的特征多项式的系数向量  $\mathbf{w}$  就是自相关矩阵中与最小特征值  $\sigma_e^2$  对应的那个特征向量。

### 7.4.3 离散 Karhunen-Loeve 变换

在许多信号处理和模式识别应用中, 常常需要将随机信号的观测样本用另外一组数(或系数)表示, 同时使这种新的表示具有某些所希望的性质。例如, 对于编码而言, 希望信号可以用少数系数表示, 同时这些系数集中了原信号的功率。又如, 对于最优滤波, 则希望变换后的样本统计不相关, 这样就可以降低滤波器的复杂度, 或者提高信噪比。实现上述目标的通用做法是将信号展开成正交基函数的线性组合, 使得信号相对于基函数的各个分量不会相互干扰。

如果正交基函数根据信号观测样本的协方差矩阵适当选择, 就有可能在所有正交基函数中, 获得具有最小均方误差的信号表示。在均方误差最小的意义上, 这样一种信号表示是最优的信号表示, 它在随机信号的分析与编码中具有重要的意义和应用。这种信号变换是 Karhunen 和 Loeve 针对连续随机信号提出的, 称为 Kauhunen-Loeve 变换。后来, Hotelling 把它推广到离散随机信号, 所以也叫 Hotelling 变换。不过, 在大多数文献中, 仍习惯称为离散 Karhunen-Loeve 变换。

令  $\mathbf{x} = [x_1, \dots, x_M]^T$  是一个零均值的随机向量, 其自相关矩阵  $\mathbf{R}_x = E\{\mathbf{x}\mathbf{x}^H\}$ 。现在, 希望使用线性变换

$$\mathbf{w} = \mathbf{Q}^H \mathbf{x} \quad (7.4.18)$$

其中,  $\mathbf{Q}$  是一酉矩阵, 即  $\mathbf{Q}^{-1} = \mathbf{Q}^H$ 。于是, 原随机信号向量  $\mathbf{x}$  可以用线性正交变换矩阵  $\mathbf{Q}$  表示成  $\mathbf{w}$  的线性组合, 即

$$\mathbf{x} = \mathbf{Q}\mathbf{w} = \sum_{i=1}^M w_i \mathbf{q}_i, \quad \mathbf{q}_i^H \mathbf{q}_j = 0, \quad i \neq j \quad (7.4.19)$$

为了减小变换后的系数  $w_i$  的个数, 假定在上式中只使用  $\mathbf{w}$  的前  $m$  个系数  $w_1, \dots, w_m$  ( $m = 1, \dots, M$ ) 逼近随机信号向量  $\mathbf{x}$ , 即

$$\hat{\mathbf{x}} = \sum_{i=1}^m w_i \mathbf{q}_i, \quad 1 \leq m \leq M \quad (7.4.20)$$

于是, 随机信号向量的  $m$  阶逼近的误差由

$$\mathbf{e}_m = \mathbf{x} - \hat{\mathbf{x}} = \sum_{i=1}^M w_i \mathbf{q}_i - \sum_{i=1}^m w_i \mathbf{q}_i = \sum_{i=m+1}^M w_i \mathbf{q}_i \quad (7.4.21)$$

给出。由此可以得到均方误差

$$E_m = \mathbb{E}\{\mathbf{e}_m^H \mathbf{e}_m\} = \sum_{i=m+1}^M \mathbf{q}_i^H \mathbb{E}\{|w_i|^2\} \mathbf{q}_i = \sum_{i=m+1}^M \mathbb{E}\{|w_i|^2\} \mathbf{q}_i^H \mathbf{q}_i \quad (7.4.22)$$

由  $w_i = \mathbf{q}_i^H \mathbf{x}$  易知  $\mathbb{E}\{|w_i|^2\} = \mathbf{q}_i^H \mathbf{R}_x \mathbf{q}_i$ 。若进一步约束  $\mathbf{q}_i^H \mathbf{q}_i = 1$ , 则式 (7.4.22) 表示的均方误差可以重新写为

$$E_m = \sum_{i=m+1}^M \mathbb{E}\{|w_i|^2\} = \sum_{i=m+1}^M \mathbf{q}_i^H \mathbf{R}_x \mathbf{q}_i \quad (7.4.23)$$

约束条件为

$$\mathbf{q}_i^H \mathbf{q}_i = 1, \quad i = m+1, m+2, \dots, M$$

为了使均方误差最小化, 使用 Lagrangian 乘子法构造代价函数

$$J = \sum_{i=m+1}^M \mathbf{q}_i^H \mathbf{R}_x \mathbf{q}_i + \sum_{i=m+1}^M \lambda_i (1 - \mathbf{q}_i^H \mathbf{q}_i),$$

令  $\frac{\partial J}{\partial \mathbf{q}_i^*} = 0, i = m+1, m+2, \dots, M$ , 即

$$\frac{\partial}{\partial \mathbf{q}_i^*} \left[ \sum_{i=m+1}^M \mathbf{q}_i^H \mathbf{R}_x \mathbf{q}_i + \sum_{i=m+1}^M \lambda_i (1 - \mathbf{q}_i^H \mathbf{q}_i) \right] = \mathbf{R}_x \mathbf{q}_i - \lambda_i \mathbf{q}_i = 0 \\ i = m+1, m+2, \dots, M \quad (7.4.24)$$

即得

$$\mathbf{R}_x \mathbf{q}_i = \lambda_i \mathbf{q}_i, \quad i = m+1, m+2, \dots, M \quad (7.4.25)$$

这一变换称为 Karhunen-Loeve 变换。

上述讨论说明, 当使用式 (7.4.20) 逼近一个随机信号向量  $\mathbf{x}$  时, 为了使逼近的均方误差为最小, 应该选择 Lagrangian 乘子  $\lambda_i$  和代价函数中的正交基向量  $\mathbf{g}_i$  分别是信号自相关矩阵  $\mathbf{R}_x$  后面的  $M-m$  个特征值和特征向量。换言之, 式 (7.4.20) 中用作随机信号向量的正交基应该是  $\mathbf{R}_x$  的前  $m$  个特征向量。

令  $M \times M$  自相关矩阵  $\mathbf{R}_x$  的特征值分解为

$$\mathbf{R}_x = \sum_{i=1}^M \lambda_i \mathbf{u}_i \mathbf{u}_i^H \quad (7.4.26)$$

因此, 式 (7.4.20) 中被选择的正交基为  $\mathbf{g}_i = \mathbf{u}_i, i = 1, \dots, m$ 。

如果自相关矩阵  $\mathbf{R}_x$  只有  $K$  个大特征值, 并且其他  $M-K$  个特征值可以忽略, 则式 (7.4.20) 中信号逼近的阶数应该取  $m = K$ , 从而得到信号的  $K$  阶离散 Karhunen-Loeve 展开式

$$\hat{\mathbf{x}} = \sum_{i=1}^K w_i \mathbf{u}_i \quad (7.4.27)$$

其中,  $w_i, i = 1, \dots, K$  是  $K \times 1$  向量

$$\mathbf{w} = \mathbf{U}_1^H \mathbf{x} \quad (7.4.28)$$

的第  $i$  个元素。式中,  $\mathbf{U}_1 = [\mathbf{u}_1, \dots, \mathbf{u}_K]$  由自相关矩阵中与  $K$  个大特征值对应的特征向量组成。此时,  $K$  阶离散 Karhunen-Loeve 展开的均方误差为

$$E_K = \sum_{i=K+1}^M \mathbf{u}_i^H \mathbf{R}_x \mathbf{u}_i = \sum_{i=K+1}^M \mathbf{u}_i^H \left( \sum_{j=1}^M \lambda_j \mathbf{u}_j \mathbf{u}_j^H \right) \mathbf{u}_i = \sum_{i=K+1}^M \lambda_i \quad (7.4.29)$$

由于  $\lambda_i, i = K+1, \dots, M$  都是自相关矩阵  $\mathbf{R}_x$  的次特征值, 均方误差  $E_K$  很小。

如果原数据  $x_1, \dots, x_M$  是需要发射的  $M$  个数据, 在发射端直接发射这些数据, 会带来两个问题: 这些数据很容易被他人接收; 在很多情况下, 数据长度  $M$  可能很大。例如, 一幅图像需要先按行转换为数据, 然后将各行的数据合成一个很长的数据段。利用离散 Karhunen-Loeve 展开, 则可以避免直接发射原数据的这两个缺陷。假定需要发送的图像或者语音信号的  $M$  个离散样本为  $x_c(0), x_c(1), \dots, x_c(M-1)$ , 其中,  $M$  很大。如果分析给定数据  $x_c(0), x_c(1), \dots, x_c(M-1)$  的自相关矩阵, 并确定其大特征值的个数  $K$ , 就可以得到  $K$  个线性变换系数  $w_1, \dots, w_K$  和  $K$  个正交的特征向量  $\mathbf{u}_1, \dots, \mathbf{u}_K$ 。这样, 就只需要在发射端发射  $K$  个系数  $w_1, \dots, w_K$ 。如果在接收端有这  $K$  个特征向量的信息, 则可利用

$$\hat{\mathbf{x}} = \sum_{i=1}^K w_i \mathbf{u}_i \quad (7.4.30)$$

重构被发射的  $M$  个数据  $x_c(i), i = 0, 1, \dots, M-1$ 。

将  $M$  个信号数据  $x_c(0), x_c(1), \dots, x_c(M-1)$  变换成  $K$  个系数  $w_1, \dots, w_K$  的过程称为信号编码或数据压缩; 而从这  $K$  个系数重构  $M$  个信号数据的过程则称为信号解码。图 7.4.2 画出了信号编码和解码的原理图。

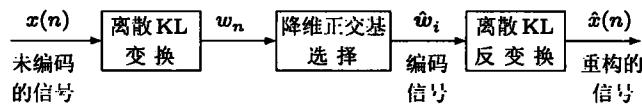


图 7.4.2 利用离散 Karhunen-Loeve 变换的信号编码和解码原理图

比率  $M/K$  称为压缩比。若  $K$  比  $M$  小得多时, 即可得到大的压缩比。显然, 经过离散 Karhunen-Loeve 变换对原数据进行编码后, 不仅可以大大压缩发射数据的长度, 而且即使  $K$  个编码系数被他人接收, 由于没有  $K$  个特征向量的信息, 他人也难于准确重构原数据。

#### 7.4.4 主分量分析

假定有  $P$  个统计相关的性质指标集合  $\{x_1, \dots, x_P\}$ , 由于它们之间的相关性, 在这

$P$  个性质指标中存在信息的冗余。现在希望通过正交变换，从中获得  $K$  个新特征集合  $\{\tilde{x}_1, \dots, \tilde{x}_K\}$ 。这些新特征相互正交。由于彼此正交，新特征之间不再有信息的冗余。这一过程称为特征提取。从空间变换的角度，特征提取的实质就是从  $P$  个原始变量的  $C^P$  空间内，提取出彼此正交的  $K$  个新变量，组成  $C^K$  空间。将一个存在信息冗余的多维空间变成一个无信息冗余的较低维空间，这样一种线性变换称为降维 (reduced dimension)。作为降维处理的典型一例，下面介绍主分量分析 (principal component analysis, PCA)。

通过正交变换，可以将存在统计相关的  $P$  个原始性质指标变成  $P$  个彼此正交的新性质指标。在这  $P$  个新的性质指标中，具有较大功率的  $K$  个性质指标可以视为  $P$  个原始性质指标的主要成分，简称主分量或者主成分。只利用数据向量的  $K$  个主分量进行的数据或者信号分析称为主分量分析。

主分量分析的主要目的是用  $K (< P)$  个主分量概括表达统计相关的  $P$  个变量。为了全面反映  $P$  个原始变量所携带的有用信息，每一个主分量都应该是  $P$  个原始变量的线性组合方式。

**定义 7.4.1** 令  $\mathbf{R}_x$  是数据向量  $\mathbf{x}$  的自相关矩阵，它有  $K$  个主特征值，与这些主特征值对应的  $K$  个特征向量称为数据向量  $\mathbf{x}$  的主分量。

主分量分析的主要步骤及思想如下。

(1) 降维 将  $P$  个变量综合成  $K$  个主分量

$$\tilde{x}_j = \sum_{i=1}^P a_{ij}^* x_i = \mathbf{a}_j^H \mathbf{x}, \quad j = 1, 2, \dots, K \quad (7.4.31)$$

式中， $\mathbf{a}_j = [a_{1j}, \dots, a_{Pj}]^T$  和  $\mathbf{x} = [x_1, \dots, x_P]^T$ 。

(2) 正文化 欲使主分量正交归一，即

$$\langle \tilde{x}_i, \tilde{x}_j \rangle = \mathbf{x}^H \mathbf{x} \mathbf{a}_i^H \mathbf{a}_j = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

必须选择系数向量  $\mathbf{a}_i$  满足正交归一条件  $\mathbf{a}_i^H \mathbf{a}_j = \delta_{i-j}$  (Kronecker  $\delta$  函数)，因为  $\mathbf{x}$  各个元素统计相关，即  $\mathbf{x}^H \mathbf{x} \neq 0$ 。

(3) 功率最大化 若选择  $\mathbf{a}_i = \mathbf{u}_i$ ,  $i = 1, \dots, K$ ，其中， $\mathbf{u}_i$  ( $i = 1, \dots, K$ ) 是自相关矩阵  $\mathbf{R}_x = E\{\mathbf{x}\mathbf{x}^H\}$  与  $K$  个大特征值  $\lambda_1 \geq \dots \geq \lambda_K$  对应的特征向量，则容易计算出各个无冗余分量的能量为

$$\begin{aligned} E_{\tilde{x}_i} &= E\{|\tilde{x}_i|^2\} = E\{\mathbf{a}_i^H \mathbf{x} (\mathbf{a}_i^H \mathbf{x})^*\} = \mathbf{u}_i^H E\{\mathbf{x}\mathbf{x}^H\} \mathbf{u}_i = \mathbf{u}_i^H \mathbf{R}_x \mathbf{u}_i \\ &= \mathbf{u}_i^H [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_P] \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_P \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^H \\ \mathbf{u}_2^H \\ \vdots \\ \mathbf{u}_P^H \end{bmatrix} \mathbf{u}_i \\ &= \lambda_i \end{aligned}$$

由于特征值按照非降顺序排列, 故

$$E_{\tilde{x}_1} \geq E_{\tilde{x}_2} \geq \cdots \geq E_{\tilde{x}_k} \quad (7.4.32)$$

因此, 按照能量的大小, 常称  $\tilde{x}_1$  为第一主分量,  $\tilde{x}_2$  为第二主分量, 等等。

注意到  $P \times P$  自相关矩阵

$$\mathbf{R}_x = E\{\mathbf{x}\mathbf{x}^H\} = \begin{bmatrix} E\{|x_1|^2\} & E\{x_1 x_2^*\} & \cdots & E\{x_1 x_P^*\} \\ E\{x_2 x_1^*\} & E\{|x_2|^2\} & \cdots & E\{x_2 x_P^*\} \\ \vdots & \vdots & \ddots & \vdots \\ E\{x_P x_1^*\} & E\{x_P x_2^*\} & \cdots & E\{|x_P|^2\} \end{bmatrix} \quad (7.4.33)$$

利用矩阵迹的定义和性质知

$$\text{tr}(\mathbf{R}_x) = E\{|x_1|^2\} + E\{|x_2|^2\} + \cdots + E\{|x_P|^2\} = \lambda_1 + \lambda_2 + \cdots + \lambda_P \quad (7.4.34)$$

但是, 若自相关矩阵  $\mathbf{R}_x$  只有  $K$  个大的特征值, 则有

$$E\{|x_1|^2\} + E\{|x_2|^2\} + \cdots + E\{|x_P|^2\} \approx \lambda_1 + \lambda_2 + \cdots + \lambda_K \quad (7.4.35)$$

总结以上讨论, 可以得出结论: 主分量分析的基本思想是通过降维、正交化和能量最大化这三个步骤, 将原来统计相关的  $P$  个随机数据转换成  $K$  个相互正交的主分量, 这些主分量的能量之和近似等于原  $P$  个随机数据的能量之和。

**定义 7.4.2**<sup>[521]</sup> 令  $\mathbf{R}_x$  是  $P$  维数据向量  $\mathbf{x}$  的自相关矩阵, 它有  $K$  个主特征值和  $P - K$  个次特征值(即小特征值), 与这些次特征值对应的  $P - K$  个特征向量称为数据向量  $\mathbf{x}$  的次分量。

只利用数据向量的  $P - K$  个次分量进行的数据或者信号分析称为次分量分析(minor component analysis, MCA)。

主分量分析可以给出被分析信号和图像的轮廓和主要信息。与之不同, 次分量分析则可以提供信号的细节和图像的纹理。次分量分析在很多领域中有着广泛的应用。例如, 次分量分析已用于频率估计<sup>[337, 338]</sup>、盲波束形成<sup>[208]</sup>、动目标显示<sup>[272]</sup>、杂波对消<sup>[30]</sup>等。在模式识别中, 当主分量分析不能识别两个对象信号时, 应进一步作次分量分析, 比较它们所含信息的细节部分。

## 7.5 广义特征值分解

前面几节讨论了单个  $n \times n$  矩阵的特征值分解及其应用。从这一节开始, 我们的注意力将陆续转移到特征值问题的各种推广。本节先考虑两个矩阵组成的矩阵对的特征值分解, 习惯称其为广义特征值分解。事实上, 单个矩阵的特征值分解是广义特征值分解的一种特例。

### 7.5.1 广义特征值分解及其性质

特征值分解的基础是线性变换  $\mathcal{L}[\mathbf{u}] = \lambda\mathbf{u}$  表示的特征系统 (eigensystem): 取线性变换  $\mathcal{L}[\mathbf{u}] = \mathbf{A}\mathbf{u}$ , 即得特征值分解  $\mathbf{A}\mathbf{u} = \lambda\mathbf{u}$ 。

现在考虑特征系统的推广: 它由两个线性系统  $\mathcal{L}_a$  和  $\mathcal{L}_b$  共同组成, 两个线性系统都以向量  $\mathbf{u}$  作为输入, 但第一个系统  $\mathcal{L}_a$  的输出  $\mathcal{L}_a[\mathbf{u}]$  是第二个系统  $\mathcal{L}_b$  的输出  $\mathcal{L}_b[\mathbf{u}]$  的某个常数 (例如  $\lambda$ ) 倍, 即特征系统推广为<sup>[254]</sup>

$$\mathcal{L}_a[\mathbf{u}] = \lambda\mathcal{L}_b[\mathbf{u}], \quad \mathbf{u} \neq \mathbf{0} \quad (7.5.1)$$

称为广义特征系统, 记作  $(\mathcal{L}_a, \mathcal{L}_b)$ 。式中的常数  $\lambda$  和非零向量  $\mathbf{u}$  分别称为广义特征系统的特征值 (即广义特征值) 和特征向量 (即广义特征向量)。

特别地, 若两个线性变换分别取

$$\mathcal{L}_a[\mathbf{u}] = \mathbf{A}\mathbf{u}, \quad \mathcal{L}_b[\mathbf{u}] = \mathbf{B}\mathbf{u} \quad (7.5.2)$$

则广义特征系统变为

$$\mathbf{A}\mathbf{u} = \lambda\mathbf{B}\mathbf{u} \quad (7.5.3)$$

广义特征系统的两个  $n \times n$  矩阵  $\mathbf{A}$  和  $\mathbf{B}$  组成一矩阵束 (matrix pencil) 或矩阵对 (matrix pair), 记作  $(\mathbf{A}, \mathbf{B})$ ; 常数  $\lambda$  和非零向量  $\mathbf{u}$  分别称为矩阵束的广义特征值 (generalized eigenvalue) 和广义特征向量 (generalized eigenvector)。

一个广义特征值和与之对应的广义特征向量合称广义特征对, 记作  $(\lambda, \mathbf{u})$ 。式 (7.5.3) 也称广义特征方程。观察知, 特征值问题是当矩阵束取作  $(\mathbf{A}, \mathbf{I})$  时广义特征值问题的一个特例。

虽然广义特征值和广义特征向量总是成对出现, 但是广义特征值可以单独求出。这一情况与特征值可以单独求出类似。为了单独求出广义特征值, 将广义特征方程式 (7.5.3) 稍加改写, 即有

$$(\mathbf{A} - \lambda\mathbf{B})\mathbf{u} = \mathbf{0} \quad (7.5.4)$$

如果上式括号内的矩阵  $\mathbf{A} - \lambda\mathbf{B}$  是非奇异的, 则广义特征方程只有唯一的零解  $\mathbf{u} = \mathbf{0}$ 。显然, 这种解是平凡的, 毫无意义。为了求出非零的有用解, 矩阵  $\mathbf{A} - \lambda\mathbf{B}$  不能是非奇异的。这意味着, 它们的行列式必须等于零

$$(\mathbf{A} - \lambda\mathbf{B}) \text{ 奇异} \Leftrightarrow \det(\mathbf{A} - \lambda\mathbf{B}) = 0 \quad (7.5.5)$$

$\det(\mathbf{A} - \lambda\mathbf{B}) = 0$  称为广义特征多项式。鉴于此, 矩阵束  $(\mathbf{A}, \mathbf{B})$  又常表示成  $\mathbf{A} - \lambda\mathbf{B}$ 。

对于  $n \times n$  维的矩阵束  $(\mathbf{A}, \mathbf{B})$ , 式 (7.5.5) 是一个  $n$  阶多项式, 称为广义特征多项式。因此, 矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值  $\lambda$  是满足广义特征多项式  $\det(\mathbf{A} - z\mathbf{B}) = 0$  的所有解  $z$  (包括零值在内)。显然, 若矩阵  $\mathbf{B}$  为单位矩阵, 则广义特征多项式退化为  $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$

即特征多项式 (7.1.6)。从这一角度讲, 广义特征多项式是特征多项式的推广, 而特征多项式是广义特征多项式在  $\mathbf{B} = \mathbf{I}$  时的一个特例。

若将矩阵束的广义特征值记作  $\lambda(\mathbf{A}, \mathbf{B})$ , 则广义特征值定义为

$$\lambda(\mathbf{A}, \mathbf{B}) = \{z \in \mathbb{C} : \det(\mathbf{A} - z\mathbf{B}) = 0\} \quad (7.5.6)$$

**定理 7.5.1**<sup>[540]</sup>  $\lambda \in \mathbb{C}$  和  $\mathbf{u} \in \mathbb{C}^n$  分别是矩阵束  $(\mathbf{A}, \mathbf{B})_{n \times n}$  的广义特征值和广义特征向量, 当且仅当

- (1)  $\det(\mathbf{A} - \lambda\mathbf{B}) = 0$ 。
- (2)  $\mathbf{u} \in \text{Null}(\mathbf{A} - \lambda\mathbf{B})$ , 并且  $\mathbf{u} \neq \mathbf{0}$ 。

下面是关于广义特征值问题  $\mathbf{Ax} = \lambda\mathbf{Bx}$  的一些性质 [252, pp.176~177]:

(1) 若矩阵  $\mathbf{A}$  和  $\mathbf{B}$  互换, 则广义特征值将变为其倒数, 但广义特征向量保持不变, 即有

$$\mathbf{Ax} = \lambda\mathbf{Bx} \Rightarrow \mathbf{Bx} = \frac{1}{\lambda}\mathbf{Ax}$$

- (2) 若  $\mathbf{B}$  非奇异, 则广义特征值分解简化为标准的特征值分解

$$\mathbf{Ax} = \lambda\mathbf{Bx} \Rightarrow (\mathbf{B}^{-1}\mathbf{A})\mathbf{x} = \lambda\mathbf{x}$$

- (3) 若  $\mathbf{A}$  和  $\mathbf{B}$  均为实对称的正定矩阵, 则广义特征值一定是正的。

- (4) 如果  $\mathbf{A}$  奇异, 则  $\lambda = 0$  必定是一个广义特征值。

(5) 若  $\mathbf{A}$  和  $\mathbf{B}$  均为正定的 Hermitian 矩阵, 则广义特征值必定是实的, 并且与不同广义特征值对应的广义特征向量相对于正定矩阵  $\mathbf{A}$  和  $\mathbf{B}$  分别正交, 即有

$$\mathbf{x}_i^H \mathbf{A} \mathbf{x}_j = \mathbf{x}_i^H \mathbf{B} \mathbf{x}_j = 0, \quad i \neq j$$

(6) 若  $\mathbf{A}$  和  $\mathbf{B}$  均为实对称矩阵, 并且  $\mathbf{B}$  正定, 则广义特征值问题  $\mathbf{Ax} = \lambda\mathbf{Bx}$  可以变换为标准的对称特征值问题

$$(\mathbf{L}^{-1}\mathbf{A}\mathbf{L}^{-T})(\mathbf{L}^T\mathbf{x}) = \lambda(\mathbf{L}^T\mathbf{x})$$

式中,  $\mathbf{L}$  为下三角矩阵, 它是 Cholesky 分解  $\mathbf{B} = \mathbf{LL}^T$  的因子。

(7) 若  $\tilde{\mathbf{B}} = \mathbf{B} + (1/\alpha)\mathbf{A}$ , 其中,  $\alpha$  是任意一个不等于零的标量, 则修正的广义特征值问题  $\mathbf{Ax} = \tilde{\lambda}\tilde{\mathbf{B}}\mathbf{x}$  的广义特征值  $\tilde{\lambda}$  与原广义特征值  $\lambda$  之间存在下列关系, 即  $\tilde{\lambda}^{-1} = \lambda^{-1} + \alpha^{-1}$ 。

严格地说, 上面介绍的广义特征向量  $\mathbf{u}$  称为矩阵束的右广义特征向量。与广义特征值  $\lambda$  对应的左特征向量定义为满足

$$\mathbf{v}^H \mathbf{A} = \lambda \mathbf{v}^H \mathbf{B} \quad (7.5.7)$$

的列向量  $\mathbf{v}$ 。令  $\mathbf{X}$  和  $\mathbf{Y}$  均为非奇异矩阵，则由式 (7.5.3) 和式 (7.5.7) 立即知

$$\mathbf{X}\mathbf{A}\mathbf{u} = \lambda\mathbf{X}\mathbf{B}\mathbf{u}, \quad \mathbf{v}^H\mathbf{A}\mathbf{Y} = \lambda\mathbf{v}^H\mathbf{B}\mathbf{Y} \quad (7.5.8)$$

这表明，矩阵束  $(\mathbf{A}, \mathbf{B})$  左乘非奇异矩阵，不改变矩阵束的右广义特征向量；而矩阵束右乘非奇异矩阵，则不改变左广义特征向量。

在很多应用中，往往只使用广义特征值（如稍后将介绍的 ESPRIT 方法），在这种情况下，等价矩阵束是一个非常有用的概念。

**定义 7.5.1** 所有广义特征值相同的两个矩阵束称为等价矩阵束。

由广义特征值的定义  $\det(\mathbf{A} - \lambda\mathbf{B}) = 0$  和行列式的性质，易知

$$\det(\mathbf{X}\mathbf{A}\mathbf{Y} - \lambda\mathbf{X}\mathbf{B}\mathbf{Y}) = 0 \iff \det(\mathbf{A} - \lambda\mathbf{B}) = 0$$

因此，矩阵束左乘任意一个非奇异矩阵与（或）右乘任意一个非奇异矩阵，都不会改变矩阵束的广义特征值。这一结果可以总结为下面的命题。

**命题 7.5.1** 若  $\mathbf{X}$  和  $\mathbf{Y}$  是两个非奇异矩阵，则  $(\mathbf{X}\mathbf{A}\mathbf{Y}, \mathbf{X}\mathbf{B}\mathbf{Y})$  和  $(\mathbf{A}, \mathbf{B})$  是两个等价的矩阵束。

### 7.5.2 广义特征值分解算法

下面的算法使用压缩映射计算  $n \times n$  实对称矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征对  $(\lambda, \mathbf{u})$ 。

**算法 7.5.1** 广义特征值分解的 Lanczos 算法 [434,p.298]

步骤 1 初始话

选择范数满足  $\mathbf{u}_1^H \mathbf{B} \mathbf{u}_1 = 1$  的向量  $\mathbf{u}_1$ ，并令  $\alpha_1 = 0$ ,  $\mathbf{z}_0 = \mathbf{u}_0 = \mathbf{0}$ ,  $\mathbf{z}_1 = \mathbf{B}\mathbf{u}_1$ 。

步骤 2 对  $i = 1, 2, \dots, n$ , 计算

$$\begin{aligned} \mathbf{u} &= \mathbf{A}\mathbf{u}_i - \alpha_i \mathbf{z}_{i-1} \\ \beta_i &= \langle \mathbf{u}, \mathbf{u}_i \rangle \\ \mathbf{u} &= \mathbf{u} - \beta_i \mathbf{z}_i \\ \mathbf{w} &= \mathbf{B}^{-1}\mathbf{u} \\ \alpha_{i+1} &= \sqrt{\langle \mathbf{w}, \mathbf{u} \rangle} \\ \mathbf{u}_{i+1} &= \mathbf{w}/\alpha_{i+1} \\ \mathbf{z}_{i+1} &= \mathbf{u}/\alpha_{i+1} \\ \lambda_i &= \beta_i/\alpha_{i+1} \end{aligned}$$

广义特征值问题也可等价写作

$$\alpha\mathbf{A}\mathbf{u} = \beta\mathbf{B}\mathbf{u} \quad (7.5.9)$$

此时，广义特征值定义为  $\lambda = \beta/\alpha$ 。

下面是计算  $n \times n$  对称正定矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值分解的正切算法，它是 Drmac 于 1998 年提出的 [146]。

### 算法 7.5.2 对称正定矩阵束的广义特征值分解

步骤 1 计算  $\Delta_A = \text{diag}(A_{11}, A_{22}, \dots, A_{nn})^{-1/2}$ ,  $A_s = \Delta_A A \Delta_A$  和  $B_1 = \Delta_A B \Delta_A$ 。

步骤 2 计算 Cholesky 分解  $R_A^T R_A = A_s$  和  $R_B^T R_B = B_1 \Pi$ 。

步骤 3 通过求解矩阵方程  $F R_B = A \Pi$ , 计算  $F = A \Pi R_B^{-1}$ 。

步骤 4 求  $F$  的奇异值分解  $\Sigma = V F U^T$ 。

步骤 5 计算  $X = \Delta_A \Pi R_B^{-1} U$ 。

输出 矩阵  $X$  和  $\Sigma$  满足  $AX = BX\Sigma^2$ 。

当矩阵  $B$  奇异时, 以上两种算法将是不稳定的。矩阵  $B$  奇异时的矩阵束  $(A, B)$  的广义特征值分解算法由 Nour-Omid 等人<sup>[373]</sup> 提出。这种算法的主要思想是: 通过引入一移位因子  $\sigma$ , 使  $(A - \sigma B)$  非奇异。

### 算法 7.5.3 $B$ 奇异时的广义特征值分解算法<sup>[373, 434]</sup>

步骤 1 初始化

选择 Range $[(A - \sigma B)^{-1} B]$  的基向量  $w$ , 计算  $z_1 = Bw$ ,  $\alpha_1 = \sqrt{\langle w, z_1 \rangle}$ , 令  $u_0 = 0$ 。

步骤 2 对  $i = 1, 2, \dots, n$ , 计算

$$u_i = w / \alpha_i$$

$$z_i = (A - \sigma B)^{-1} u_i$$

$$w = w - \alpha_i u_{i-1}$$

$$\beta_i = \langle w, z_i \rangle$$

$$z_{i+1} = Bw$$

$$\alpha_{i+1} = \sqrt{\langle z_{i+1}, w \rangle}$$

$$\lambda_i = \beta_i / \alpha_{i+1}$$

### 7.5.3 广义特征值分解的总体最小二乘方法

在广义特征值分解的应用中, 我们往往只对非零的广义特征值感兴趣, 因为这些非零的广义特征值的个数反映了信号分量的个数, 而广义特征值本身则往往隐含了信号参数的有用信息。然而, 在实际应用中, 信号分量的个数常常是不知道的, 需要估计。通常, 矩阵束  $(A, B)$  的维数往往比信号分量的实际个数大。另外, 矩阵  $A$  和  $B$  又常常分别由观测数据向量的自相关矩阵和互相关矩阵构成。在实际应用中, 这些相关矩阵中的自相关函数和互相关函数往往由比较短的观测样本数据估计得到, 存在比较大的估计误差。矩阵束的实际维数取大和相关矩阵存在较大估计误差这两个事实, 使得矩阵束的非零广义特征值的估计成为最小二乘算子。

Roy 和 Kailath 指出<sup>[431]</sup>, 最小二乘算子会导致在求解广义特征值问题的某些潜在的数值困难。前面两章已详细分析了奇异值分解 (SVD) 和总体最小二乘 (TLS) 的应用可以将一个较大维数 ( $m \times m$ ) 病态最小二乘问题转化为一个较小维数 ( $p \times p$ ) 的无病态总体最小二乘问题。因此, 求解广义特征值问题的总体最小二乘方法成为广义特征值分解应

用中的一种自然选择。

已提出了多种求解广义特征值问题的总体最小二乘方法。这些方法的中心思想都是在不改变矩阵束的非零广义特征值的前提下，利用截尾的奇异值分解，将一个大维数的矩阵束转化为一个小微数的矩阵束。这些方法需要的奇异值分解次数各不相同。其中，Zhang 和 Liang<sup>[538]</sup> 提出的方法只需要 1 次奇异值分解，是计算最简单的。

考虑矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值分解。令  $\mathbf{A}$  的奇异值分解为

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^H = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \Sigma_1 & \mathbf{O} \\ \mathbf{O} & \Sigma_2 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad (7.5.10)$$

式中， $\Sigma_1$  由  $p$  个主奇异值组成，在不改变广义特征值的条件下，可以用  $\mathbf{U}_1^H$  左乘和用  $\mathbf{V}_1$  右乘矩阵  $\mathbf{A} - \gamma \mathbf{B}$ ，得到

$$\Sigma_1 - \gamma \mathbf{U}_1^H \mathbf{B} \mathbf{V}_1 \quad (7.5.11)$$

原较大维数的矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值问题变成了较小维数 ( $p \times p$ ) 的矩阵束  $(\Sigma_1, \mathbf{U}_1^H \mathbf{B} \mathbf{V}_1)$  的广义特征值问题。这一方法称为广义特征值分解的总体最小二乘方法。

#### 7.5.4 应用举例——ESPRIT 方法

ESPRIT 是借助旋转不变技术估计信号参数 (estimating signal parameter via rotational invariance techniques) 的英文缩写。ESPRIT 方法最早是由 Roy 等人<sup>[431]</sup> 于 1989 年提出的，现已成为现代信号处理中的一种主要方法，并得到了广泛的应用。

考虑白噪声中的  $p$  个谐波信号

$$x(n) = \sum_{i=1}^p s_i e^{j n \omega_i} + w(n) \quad (7.5.12)$$

式中， $s_i$  和  $\omega_i \in (-\pi, \pi)$  分别为第  $i$  个谐波信号的复幅值和频率。假定  $w(n)$  是一零均值、方差为  $\sigma^2$  的复值高斯白噪声过程，即

$$\mathbb{E}\{w(k)w^*(l)\} = \sigma^2 \delta(k-l), \quad \mathbb{E}\{w(k)w(l)\} = 0, \quad \forall k, l$$

问题是，只根据观测数据  $x(1), \dots, x(N)$ ，估计谐波信号的个数  $p$  和频率  $\omega_1, \dots, \omega_p$ 。

定义一个新的过程  $y(n) \stackrel{\text{def}}{=} x(n+1)$ 。选择  $m > p$ ，并引入以下  $m \times 1$  维向量

$$\mathbf{x}(n) \stackrel{\text{def}}{=} [x(n), x(n+1), \dots, x(n+m-1)]^T \quad (7.5.13)$$

$$\mathbf{w}(n) \stackrel{\text{def}}{=} [w(n), w(n+1), \dots, w(n+m-1)]^T \quad (7.5.14)$$

$$\mathbf{y}(n) \stackrel{\text{def}}{=} [y(n), y(n+1), \dots, y(n+m-1)]^T \quad (7.5.15)$$

$$= [x(n+1), x(n+2), \dots, x(n+m)]^T \quad (7.5.15)$$

$$\mathbf{a}(\omega_i) \stackrel{\text{def}}{=} [1, e^{j \omega_i}, \dots, e^{j(m-1)\omega_i}]^T \quad (7.5.16)$$

于是，式 (7.5.12) 可以写作向量形式

$$\mathbf{x}(n) = \mathbf{A}\mathbf{s}(n) + \mathbf{w}(n) \quad (7.5.17)$$

另有

$$\mathbf{y}(n) = \mathbf{A}\Phi\mathbf{s}(n) + \mathbf{w}(n+1) \quad (7.5.18)$$

式中

$$\mathbf{A} \stackrel{\text{def}}{=} [\mathbf{a}(\omega_1), \mathbf{a}(\omega_2), \dots, \mathbf{a}(\omega_p)] \quad (7.5.19)$$

$$\mathbf{s}(n) \stackrel{\text{def}}{=} [s_1 e^{j\omega_1 n}, s_2 e^{j\omega_2 n}, \dots, s_p e^{j\omega_p n}]^T \quad (7.5.20)$$

$$\Phi \stackrel{\text{def}}{=} \text{diag}(e^{j\omega_1}, e^{j\omega_2}, \dots, e^{j\omega_p}) \quad (7.5.21)$$

注意,  $\Phi$  是一酉矩阵, 即有  $\Phi^H \Phi = \Phi \Phi^H = \mathbf{I}$ , 它将空间的向量  $\mathbf{x}(n)$  和  $\mathbf{y}(n)$  联系在一起; 矩阵  $\mathbf{A}$  是一个  $m \times p$  维 Vandermonde 矩阵。由于  $\mathbf{y}(n) = \mathbf{x}(n+1)$ , 故  $\mathbf{y}(n)$  可以看作是  $\mathbf{x}(n)$  的平移结果。鉴于此, 矩阵  $\Phi$  被称作旋转算符, 因为平移是最简单的旋转。

观测向量  $\mathbf{x}(n)$  的自相关矩阵为

$$\mathbf{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}^H(n)\} = \mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma^2 \mathbf{I} \quad (7.5.22)$$

式中  $\mathbf{P} = E\{\mathbf{s}(n)\mathbf{s}^H(n)\}$  是信号向量的相关矩阵。

向量  $\mathbf{x}(n)$  和  $\mathbf{y}(n)$  的互相关矩阵为

$$\mathbf{R}_{xy} = E\{\mathbf{x}(n)\mathbf{y}^H(n)\} = \mathbf{A}\mathbf{P}\Phi^H\mathbf{A}^H + \sigma^2 \mathbf{Z} \quad (7.5.23)$$

式中,  $\sigma^2 \mathbf{Z} = E\{\mathbf{w}(n)\mathbf{w}^H(n+1)\}$ 。容易验证,  $\mathbf{Z}$  是一个  $m \times m$  特殊矩阵

$$\mathbf{Z} = \begin{bmatrix} 0 & & & 0 \\ 1 & 0 & & \\ & \ddots & \ddots & \\ 0 & & 1 & 0 \end{bmatrix} \quad (7.5.24)$$

即主对角线下面的对角线上的元素全部为 1, 而其他元素皆等于 0。

现在的问题是: 已知自相关矩阵  $\mathbf{R}_{xx}$  和互相关矩阵  $\mathbf{R}_{xy}$ , 如何估计谐波信号的个数  $p$ 、谐波频率  $\omega_i$  以及谐波功率  $|s_i|^2$  ( $i = 1, 2, \dots, p$ )。

向量  $\mathbf{x}(n)$  经过平移, 变为  $\mathbf{y}(n) = \mathbf{x}(n+1)$ , 但是这种平移却保持了  $\mathbf{x}(n)$  和  $\mathbf{y}(n)$  对应的信号子空间的不变性。这是因为  $\mathbf{R}_{xx} \stackrel{\text{def}}{=} E\{\mathbf{x}(n)\mathbf{x}^H(n)\} = E\{\mathbf{x}(n+1)\mathbf{x}^H(n+1)\} \stackrel{\text{def}}{=} \mathbf{R}_{yy}$ , 它们完全相同!

对  $\mathbf{R}_{xx}$  作特征值分解, 可以得到其最小特征值  $\lambda_{\min} = \sigma^2$ 。构造一对新的矩阵

$$\mathbf{C}_{xx} = \mathbf{R}_{xx} - \lambda_{\min} \mathbf{I} = \mathbf{R}_{xx} - \sigma^2 \mathbf{I} = \mathbf{A}\mathbf{P}\mathbf{A}^H \quad (7.5.25)$$

$$\mathbf{C}_{xy} = \mathbf{R}_{xy} - \lambda_{\min} \mathbf{Z} = \mathbf{R}_{xy} - \sigma^2 \mathbf{Z} = \mathbf{A}\mathbf{P}\Phi^H\mathbf{A}^H \quad (7.5.26)$$

用  $(\mathbf{C}_{xx}, \mathbf{C}_{xy})$  组成一矩阵束。

考查矩阵束

$$\mathbf{C}_{xx} - \gamma \mathbf{C}_{xy} = \mathbf{A}\mathbf{P}(\mathbf{I} - \gamma \Phi^H)\mathbf{A}^H \quad (7.5.27)$$

由于  $\mathbf{A}$  满列秩和  $\mathbf{P}$  非奇异, 所以从矩阵秩的角度, 式(7.5.27)可以写作

$$\text{rank}(\mathbf{C}_{xx} - \gamma \mathbf{C}_{xy}) = \text{rank}(\mathbf{I} - \gamma \boldsymbol{\Phi}^H) \quad (7.5.28)$$

当  $\gamma \neq e^{j\omega_i}, i = 1, \dots, p$  时, 矩阵  $(\mathbf{I} - \gamma \boldsymbol{\Phi})$  是非奇异的, 而当  $\gamma = e^{j\omega_i}$  时, 由于  $\gamma e^{-j\omega_i} = 1$ , 所以矩阵  $(\mathbf{I} - \gamma \boldsymbol{\Phi})$  奇异, 即秩亏缺。这说明,  $e^{j\omega_i}, i = 1, \dots, p$  都是矩阵束  $(\mathbf{C}_{xx}, \mathbf{C}_{xy})$  的广义特征值。这一结果可以用下面的定理加以归纳。

**定理 7.5.2**<sup>[431]</sup> 定义  $\boldsymbol{\Gamma}$  为矩阵束  $(\mathbf{C}_{xx}, \mathbf{C}_{xy})$  的广义特征值矩阵, 其中,  $\mathbf{C}_{xx} = \mathbf{R}_{xx} - \lambda_{\min} \mathbf{I}$ ,  $\mathbf{C}_{xy} = \mathbf{R}_{xy} - \lambda_{\min} \mathbf{Z}$ , 且  $\lambda_{\min}$  是自相关矩阵  $\mathbf{R}_{xx}$  的最小特征值。若矩阵  $\mathbf{P}$  非奇异, 则矩阵  $\boldsymbol{\Gamma}$  与旋转算符矩阵  $\boldsymbol{\Phi}$  之间有下列关系

$$\boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{\Phi} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \quad (7.5.29)$$

即  $\boldsymbol{\Gamma}$  的非零元素是旋转算符矩阵  $\boldsymbol{\Phi}$  的各元素的一个排列。

基本的 ESPRIT 算法可总结如下。

#### 算法 7.5.4 基本 ESPRIT 算法 1<sup>[431]</sup>

步骤 1 利用已知观测数据  $x(1), \dots, x(N)$  估计自相关函数  $R_{xx}(0), R_{xx}(1), \dots, R_{xx}(m)$ 。

步骤 2 由估计的自相关函数构造  $m \times m$  自相关矩阵  $\mathbf{R}_{xx}$  和  $m \times m$  互相关矩阵  $\mathbf{R}_{xy}$ 。

步骤 3 求  $\mathbf{R}_{xx}$  的特征值分解。对于  $m > p$ , 最小特征值为噪声方差  $\sigma^2$  的估计。

步骤 4 利用  $\sigma^2$  计算  $\mathbf{C}_{xx} = \mathbf{R}_{xx} - \sigma^2 \mathbf{I}$  和  $\mathbf{C}_{xy} = \mathbf{R}_{xy} - \sigma^2 \mathbf{Z}$ 。

步骤 5 求矩阵束  $(\mathbf{C}_{xx}, \mathbf{C}_{xy})$  的广义特征值分解, 得到位于单位圆上的  $p$  个广义特征值  $e^{j\omega_i}, i = 1, \dots, p$ , 它们直接给出谐波频率。

以上介绍的基本 ESPRIT 方法可以看作是一种最小二乘算子, 其作用是将原  $m$  维观测空间约束到一个子空间 (其维数等于波达方向个数  $p$ )。因此, 这种基本 ESPRIT 方法有时称作 LS-ESPRIT 算法。前已分析过, 将总体最小二乘方法的思想应用于广义特征值分解, 可以改善其数值性能。因此, 对 ESPRIT 方法有必要使用下面的总体最小二乘算法。

#### 算法 7.5.5 TLS-ESPRIT 算法<sup>[538]</sup>

步骤 1 进行矩阵  $\mathbf{R}_{xx}$  的特征值分解。

步骤 2 利用最小特征值  $\sigma^2$  计算  $\mathbf{C}_{xx} = \mathbf{R}_{xx} - \sigma^2 \mathbf{I}$  和  $\mathbf{C}_{xy} = \mathbf{R}_{xy} - \sigma^2 \mathbf{Z}$ 。

步骤 3 作矩阵  $\mathbf{C}_{xx}$  的奇异值分解, 确定其有效秩, 并存储与  $p$  个主奇异值对应的  $\boldsymbol{\Sigma}_1, \mathbf{U}_1$  和  $\mathbf{V}_1$ 。

步骤 4 计算  $\mathbf{U}_1^H \mathbf{C}_{xy} \mathbf{V}_1$ 。

步骤 5 求矩阵束  $(\boldsymbol{\Sigma}_1, \mathbf{U}_1^H \mathbf{C}_{xy} \mathbf{V}_1)$  的广义特征值分解, 得到单位圆上的广义特征值, 它们直接给出谐波频率。

业已证明, 虽然 LS-ESPRIT 和 TLS-ESPRIT 给出相同的渐近 (对大样本) 估计精度, 但是在小样本时 TLS-ESPRIT 总是比 LS-ESPRIT 好。此外, 与 LS-ESPRIT 不同, TLS-ESPRIT 考虑了  $\mathbf{C}_{xx}$  和  $\mathbf{C}_{xy}$  二者的噪声影响, 所以比 LS-ESPRIT 更合理。

### 7.5.5 相似变换在广义特征值分解中的应用

考查一个由  $m$  个阵元组成的等距线阵。如图 7.5.1 所示，现在将这个等距线阵分为两个子阵列，其中，子阵列 1 由第 1 个至第  $m-1$  个阵元组成，子阵列 2 由第 2 个至第  $m$  个阵元组成。

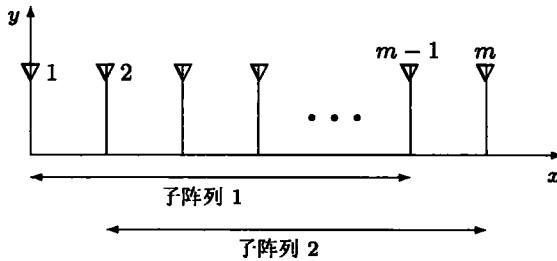


图 7.5.1 阵列分成两个子阵列

令  $m \times N$  矩阵

$$\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)] \quad (7.5.30)$$

代表原阵列的观测数据矩阵，其中， $\mathbf{x}(n) = [x_1(n), x_2(n), \dots, x_m(n)]^T$  是  $m$  个阵元在  $n$  时刻的观测信号组成的观测数据向量；而  $N$  为数据长度，即  $n = 1, 2, \dots, N$ 。

若令

$$\mathbf{s} = [s(1), s(2), \dots, s(N)] \quad (7.5.31)$$

代表信号矩阵，式中， $s(n) = [s_1(n), s_2(n), \dots, s_p(n)]^T$  表示信号向量，则对于  $N$  个快拍的数据，式 (7.5.12) 可以用矩阵形式表示成

$$\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)] = \mathbf{A}\mathbf{s} \quad (7.5.32)$$

式中， $\mathbf{A}$  是  $m \times p$  阵列方向矩阵。

令  $\mathbf{J}_1$  和  $\mathbf{J}_2$  是两个  $(m-1) \times m$  选择矩阵，且有

$$\mathbf{J}_1 = [\mathbf{I}_{m-1} \mid \mathbf{0}_{m-1}] \quad (7.5.33)$$

$$\mathbf{J}_2 = [\mathbf{0}_{m-1} \mid \mathbf{I}_{m-1}] \quad (7.5.34)$$

式中， $\mathbf{I}_{m-1}$  代表  $(m-1) \times (m-1)$  单位矩阵， $\mathbf{0}_{m-1}$  表示  $(m-1) \times 1$  零向量。

用选择矩阵  $\mathbf{J}_1$  和  $\mathbf{J}_2$  分别左乘观测数据矩阵  $\mathbf{X}$ ，得到

$$\mathbf{X}_1 = \mathbf{J}_1 \mathbf{X} = [\mathbf{x}_1(1), \mathbf{x}_1(2), \dots, \mathbf{x}_1(N)] \quad (7.5.35)$$

$$\mathbf{X}_2 = \mathbf{J}_2 \mathbf{X} = [\mathbf{x}_2(1), \mathbf{x}_2(2), \dots, \mathbf{x}_2(N)] \quad (7.5.36)$$

式中

$$\mathbf{x}_1(n) = [x_1(n), x_2(n), \dots, x_{m-1}(n)]^T, \quad n = 1, 2, \dots, N \quad (7.5.37)$$

$$\mathbf{x}_2(n) = [x_2(n), x_3(n), \dots, x_m(n)]^T, \quad n = 1, 2, \dots, N \quad (7.5.38)$$

即是说, 观测数据子矩阵  $\mathbf{X}_1$  由观测数据矩阵  $\mathbf{X}$  的前  $m-1$  行组成, 相当于子阵列 1 的观测数据矩阵;  $\mathbf{X}_2$  则由  $\mathbf{X}$  的后  $m-1$  行组成, 相当于子阵列 2 的观测数据矩阵。

令

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \text{最后一行} \end{bmatrix} = \begin{bmatrix} \text{第 1 行} \\ \mathbf{A}_2 \end{bmatrix} \quad (7.5.39)$$

则根据等距线阵的阵列响应矩阵  $\mathbf{A}$  的结构知, 子矩阵  $\mathbf{A}_1$  和  $\mathbf{A}_2$  之间存在以下关系

$$\mathbf{A}_2 = \mathbf{A}_1 \Phi \quad (7.5.40)$$

容易验证

$$\mathbf{X}_1 = \mathbf{A}_1 \mathbf{S} \quad (7.5.41)$$

$$\mathbf{X}_2 = \mathbf{A}_2 \mathbf{S} = \mathbf{A}_1 \Phi \mathbf{S} \quad (7.5.42)$$

由于  $\Phi$  是一酉矩阵, 所以  $\mathbf{X}_1$  和  $\mathbf{X}_2$  具有相同的信号子空间和噪声子空间, 即子阵列 1 和子阵列 2 具有相同的观测空间 (信号子空间 + 噪声子空间)。这就是等距线阵的平移不变性的物理解释。

由式 (7.5.22) 得

$$\begin{aligned} \mathbf{R}_{xx} &= \mathbf{A} \mathbf{P} \mathbf{A}^H + \sigma^2 \mathbf{I} = [\mathbf{U}_s, \mathbf{U}_n] \begin{bmatrix} \Sigma_s & \mathbf{O} \\ \mathbf{O} & \sigma^2 \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U}_s^H \\ \mathbf{U}_n^H \end{bmatrix} \\ &= [\mathbf{U}_s \Sigma_s, \sigma^2 \mathbf{U}_n] \begin{bmatrix} \mathbf{U}_s^H \\ \mathbf{U}_n^H \end{bmatrix} = \mathbf{U}_s \Sigma_s \mathbf{U}_s^H + \sigma^2 \mathbf{U}_n \mathbf{U}_n^H \end{aligned} \quad (7.5.43)$$

由于  $\mathbf{I} - \mathbf{U}_n \mathbf{U}_n^H = \mathbf{U}_s \mathbf{U}_s^H$ , 故由式 (7.5.43) 得

$$\mathbf{A} \mathbf{P} \mathbf{A}^H + \sigma^2 \mathbf{U}_s \mathbf{U}_s^H = \mathbf{U}_s \Sigma_s \mathbf{U}_s^H \quad (7.5.44)$$

用  $\mathbf{U}_s$  右乘上式两边, 注意到  $\mathbf{U}_s^H \mathbf{U}_s = \mathbf{I}$ , 并加以重排, 即得

$$\mathbf{U}_s = \mathbf{A} \mathbf{T} \quad (7.5.45)$$

式中,  $\mathbf{T}$  是一个非奇异矩阵, 且

$$\mathbf{T} = \mathbf{P} \mathbf{A}^H \mathbf{U}_s (\Sigma_s - \sigma^2 \mathbf{I})^{-1} \quad (7.5.46)$$

虽然  $\mathbf{T}$  是一未知矩阵, 但它只是下面分析中的一个“虚拟参数”, 我们只用到它的非奇异性。用  $\mathbf{T}$  右乘式 (7.5.39), 则有

$$\mathbf{A} \mathbf{T} = \begin{bmatrix} \mathbf{A}_1 \mathbf{T} \\ \text{最后一行} \end{bmatrix} = \begin{bmatrix} \text{第 1 行} \\ \mathbf{A}_2 \mathbf{T} \end{bmatrix} \quad (7.5.47)$$

采用相同的分块形式, 将  $\mathbf{U}_s$  也分块成

$$\mathbf{U}_s = \begin{bmatrix} \mathbf{U}_1 \\ \text{最后一行} \end{bmatrix} = \begin{bmatrix} \text{第 1 行} \\ \mathbf{U}_2 \end{bmatrix} \quad (7.5.48)$$

由于  $\mathbf{A}\mathbf{T} = \mathbf{U}_s$ , 故比较式(7.5.47)与式 (7.5.48), 立即有

$$\mathbf{U}_1 = \mathbf{A}_1\mathbf{T} \quad \text{和} \quad \mathbf{U}_2 = \mathbf{A}_2\mathbf{T} \quad (7.5.49)$$

将式 (7.5.40) 代入式 (7.5.49), 即有

$$\mathbf{U}_2 = \mathbf{A}_1\Phi\mathbf{T} \quad (7.5.50)$$

由式 (7.5.49) 及式 (7.5.50), 又有

$$\mathbf{U}_1\mathbf{T}^{-1}\Phi\mathbf{T} = \mathbf{A}_1\mathbf{T}\mathbf{T}^{-1}\Phi\mathbf{T} = \mathbf{A}_1\Phi\mathbf{T} = \mathbf{U}_2 \quad (7.5.51)$$

定义

$$\Psi = \mathbf{T}^{-1}\Phi\mathbf{T} \quad (7.5.52)$$

矩阵  $\Psi$  称为矩阵  $\Phi$  的相似变换, 因此它们具有相同的特征值, 即  $\Psi$  的特征值也为  $e^{j\phi_m}$ ,  $m = 1, 2, \dots, M$ 。

将式 (7.5.52) 代入式 (7.5.51), 则得到一个重要的关系式, 即

$$\mathbf{U}_2 = \mathbf{U}_1\Psi \quad (7.5.53)$$

式 (7.5.53) 启迪了基本 ESPRIT 算法的另一种算法。

#### 算法 7.5.6 (基本 ESPRIT 算法 2)

步骤 1 计算阵列协方差矩阵  $\hat{\mathbf{R}}_{xx}$  的特征值分解  $\hat{\mathbf{R}}_{xx} = \hat{\mathbf{U}}\Sigma\hat{\mathbf{U}}^H$ 。

步骤 2 矩阵  $\hat{\mathbf{U}}$  与  $\hat{\mathbf{R}}_{xx}$  的  $p$  个主特征值对应的部分组成  $\hat{\mathbf{U}}_s$ 。

步骤 3 抽取  $\hat{\mathbf{U}}_s$  的前面  $m - 1$  行组成矩阵  $\hat{\mathbf{U}}_1$ , 后面  $m - 1$  行组成矩阵  $\hat{\mathbf{U}}_2$ 。计算  $\Psi = (\hat{\mathbf{U}}_1^H\hat{\mathbf{U}}_1)^{-1}\hat{\mathbf{U}}_1^H\hat{\mathbf{U}}_2$  的特征值分解。矩阵  $\Psi$  的特征值  $e^{j\omega_i}$  ( $i = 1, 2, \dots, p$ ) 给出估计值  $\hat{\omega}_i$ ,  $i = 1, 2, \dots, p$ 。

ESPRIT 方法在通信信号处理尤其是在空时二维处理中有着重要的应用, 感兴趣的读者可参考文献 [545]。

## 7.6 Rayleigh 商

在物理和信息技术中, 常常会遇到 Hermitian 矩阵的二次型函数的商的最大化或者最小化。这种商有两种形式, 它们分别是一个 Hermitian 矩阵的 Rayleigh 商 (有时也叫 Rayleigh-Ritz 比) 和两个 Hermitian 矩阵的广义 Rayleigh 商 (或广义 Rayleigh-Ritz 比)。

### 7.6.1 Rayleigh 商的定义及性质

在研究振动系统的小振荡时, 为了找到合适的广义坐标, Rayleigh 于 20 世纪 30 年代提出了一种特殊形式的商<sup>[425]</sup>, 被后人称为 Rayleigh 商。下面是现在被广泛采用的 Rayleigh 商定义。

**定义 7.6.1** Hermitian 矩阵  $A \in \mathbb{C}^{n \times n}$  的 Rayleigh 商或 Rayleigh-Ritz 比  $R(\mathbf{x})$  是一个标量, 定义为

$$R(\mathbf{x}) = R(\mathbf{x}, A) = \frac{\mathbf{x}^H A \mathbf{x}}{\mathbf{x}^H \mathbf{x}} \quad (7.6.1)$$

其中,  $\mathbf{x}$  是待选择的向量, 其目的是使 Rayleigh 商最大化或者最小化。

Rayleigh 商的重要性质如下<sup>[393, 394, 101, 224]</sup>。

**性质 1 (齐次性)** 若  $\alpha$  和  $\beta$  为标量, 则

$$R(\alpha \mathbf{x}, \beta A) = \beta R(\mathbf{x}, A) \quad (7.6.2)$$

**性质 2 (平移不变性)**

$$R(\mathbf{x}, A - \alpha I) = R(\mathbf{x}, A) - \alpha \quad (7.6.3)$$

**性质 3 (正交性)**

$$\mathbf{x} \perp (A - R(\mathbf{x})I)\mathbf{x} \quad (7.6.4)$$

**性质 4 (有界性)** 当向量  $\mathbf{x}$  在所有非零向量的范围变化时, Rayleigh 商  $R(\mathbf{x})$  落在一复平面的区域 (称为矩阵  $A$  的值域) 内, 这一区域是闭合的、有界的和凸的。若  $A$  是 Hermitian 的, 即满足  $A = A^H$ , 则这一区域是一个闭区间  $[\lambda_1, \lambda_n]$ 。

**性质 5 (最小残差)** 对于所有向量  $\mathbf{x} \neq 0$  和所有标量  $\mu$ , 恒有

$$\|[A - R(\mathbf{x})I]\mathbf{x}\| \leq \|[A - \mu I]\mathbf{x}\| \quad (7.6.5)$$

关于有界性, 可进一步参考文献 [333]。

Hermitian 矩阵的 Rayleigh 商的有界性可以用下面的定理严格叙述。

**定理 7.6.1 (Rayleigh-Ritz 定理)** 令  $A \in \mathbb{C}^{n \times n}$  是 Hermitian 的, 并令  $A$  的特征值按递增次序

$$\lambda_{\min} = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{n-1} \leq \lambda_n = \lambda_{\max} \quad (7.6.6)$$

排列, 则

$$\max_{\mathbf{x} \neq 0} \frac{\mathbf{x}^H A \mathbf{x}}{\mathbf{x}^H \mathbf{x}} = \max_{\mathbf{x}^H \mathbf{x}=1} \frac{\mathbf{x}^H A \mathbf{x}}{\mathbf{x}^H \mathbf{x}} = \lambda_{\max}, \quad \text{若 } A \mathbf{x} = \lambda_{\max} \mathbf{x} \quad (7.6.7)$$

和

$$\min_{\mathbf{x} \neq 0} \frac{\mathbf{x}^H A \mathbf{x}}{\mathbf{x}^H \mathbf{x}} = \min_{\mathbf{x}^H \mathbf{x}=1} \frac{\mathbf{x}^H A \mathbf{x}}{\mathbf{x}^H \mathbf{x}} = \lambda_{\min}, \quad \text{若 } A \mathbf{x} = \lambda_{\min} \mathbf{x} \quad (7.6.8)$$

更一般地, 矩阵  $A$  的所有特征向量和特征值分别称为 Rayleigh 商  $R(\mathbf{x})$  的临界点 (critical point) 和临界值 (critical value)。

这个定理的证明方法有多种, 如参考文献 [198, 224, 117]。

下面考虑 Rayleigh 商的梯度与 Hessian 矩阵<sup>[224, 117]</sup>。为简便计, 将 Rayleigh 商  $R(\mathbf{x})$  简记作  $R$ 。

Rayleigh 商的梯度为

$$\nabla_{\mathbf{x}} = \frac{\partial R}{\partial \mathbf{x}^T} = \frac{2}{\|\mathbf{x}\|_2^2} (\mathbf{A} - R\mathbf{I})\mathbf{x} \quad (7.6.9)$$

而 Rayleigh 商的 Hessian 矩阵为

$$\mathbf{H}_R = \frac{\partial^2 R}{\partial \mathbf{x} \partial \mathbf{x}^T} = \frac{2}{\|\mathbf{x}\|_2^2} [\mathbf{A} - \nabla_{\mathbf{x}}(R)\mathbf{x}^T - \mathbf{x}\nabla_{\mathbf{x}}^T(R)\mathbf{x} - R\mathbf{I}] \quad (7.6.10)$$

令  $\mathbf{u}_i, \lambda_i, i = 1, \dots, n$  分别是矩阵  $\mathbf{A}$  的特征向量和特征值, 即它们分别是 Rayleigh 商的临界点和临界值, 即有

$$R(\mathbf{u}_i) = \lambda_i, \quad i = 1, \dots, n \quad (7.6.11)$$

计算 Hessian 矩阵在临界点  $\mathbf{u}_i$  的值, 易得

$$\mathbf{H}_R(\mathbf{u}_i) = \mathbf{A} - \lambda_i \mathbf{I}, \quad i = 1, \dots, n \quad (7.6.12)$$

由式 (7.6.12) 可得两个重要结果:

(1) Hessian 矩阵的行列式

$$|\mathbf{H}_R(\mathbf{u}_i)| = |\mathbf{A} - \lambda_i \mathbf{I}| = 0 \quad (7.6.13)$$

因为  $\mathbf{A} - z\mathbf{I}$  是矩阵  $\mathbf{A}$  的特征多项式。上式意味着 Hessian 矩阵  $\mathbf{H}_R(\mathbf{u}_i)$  对于所有临界点  $\mathbf{u}_i$  都是奇异矩阵。

(2) 用向量  $\mathbf{u}_j$  右乘式 (7.6.12), 立即有

$$\mathbf{H}_R(\mathbf{u}_i)\mathbf{u}_j = (\mathbf{A} - \lambda_i \mathbf{I})\mathbf{u}_j = \mathbf{A}\mathbf{u}_j - \lambda_i \mathbf{u}_j = \lambda_j \mathbf{u}_j - \lambda_i \mathbf{u}_j$$

因为  $\mathbf{A}\mathbf{u}_j = \lambda_j \mathbf{u}_j$ 。上式即是

$$\mathbf{H}_R(\mathbf{u}_i)\mathbf{u}_j = \begin{cases} 0, & j = i \\ (\lambda_j - \lambda_i)\mathbf{u}_j, & j \neq i \end{cases} \quad (7.6.14)$$

这说明, 由 Rayleigh 商的临界点计算得到的 Hessian 矩阵  $\mathbf{H}_R(\mathbf{u}_i)$  与矩阵  $\mathbf{A}$  具有相同的特征向量, 但特征值不同。此外, 由于  $\lambda_j - \lambda_{\min} \geq 0$ , 故只有在临界点  $\mathbf{u}_{\min}$  的 Hessian 矩阵是半正定的, 满足  $\mathbf{H}_R(\mathbf{u}_{\min}) \succeq 0$ 。

## 7.6.2 Rayleigh 商迭代

令  $\mathbf{A} \in \mathbb{C}^{n \times n}$  是一个可对角化的矩阵, 其特征值为  $\lambda_i$ , 与之对应的特征向量为  $\mathbf{u}_i$ 。为方便计, 假定矩阵  $\mathbf{A}$  非奇异, 第一个特征值比其他特征值都大, 并且  $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_n$ , 则特征值  $\lambda_1$  和与之对应的特征向量  $\mathbf{u}_1$  分别称为矩阵  $\mathbf{A}$  的主特征值和主特征向量。

乘幂法使用

$$\mathbf{x}_k = \frac{\mathbf{A}\mathbf{x}_{k-1}}{\|\mathbf{A}\mathbf{x}_{k-1}\|_2} = \frac{\mathbf{A}^k \mathbf{x}_0}{\|\mathbf{A}^k \mathbf{x}_0\|_2} \quad (7.6.15)$$

迭代计算向量  $\mathbf{x}$ , 同时希望它收敛为主特征向量, 即

$$\lim_{k \rightarrow \infty} \mathbf{x}_k = \lambda \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} \quad \text{对某个满足 } |\lambda| = 1 \text{ 的复常数 } \lambda \quad (7.6.16)$$

乘幂法的魅力在于它的简单性, 而非计算有效性。

Rayleigh 观察到  $\lambda_i = R(\mathbf{x}_i)$ , 并提出了当矩阵  $\mathbf{A}$  是 Hermitian 矩阵时, 计算其主特征值和主特征向量的迭代方法, 现在习惯称为 Rayleigh 商迭代。

Rayleigh 商迭代是逆迭代的一种变型, 其标准算法如下。选择一个单位长度的初始向量  $\mathbf{x}_0$ , 对  $k = 0, 1, \dots$  执行以下运算:

- (1) 构造  $R_k = R(\mathbf{x}_k) = \mathbf{x}_k^H \mathbf{A} \mathbf{x}_k$ 。
- (2) 若  $(\mathbf{A} - R_k \mathbf{I})$  奇异, 则求  $(\mathbf{A} - R_k \mathbf{I}) \mathbf{x}_{k+1} = \mathbf{0}$  的非零解  $\mathbf{x}_{k+1} \neq \mathbf{0}$ , 并停止迭代。若  $(\mathbf{A} - R_k \mathbf{I})$  非奇异, 则继续下面的运算。
- (3) 计算

$$\mathbf{x}_{k+1} = \frac{(\mathbf{A}_k - R_k \mathbf{I})^{-1} \mathbf{x}_k}{\|(\mathbf{A}_k - R_k \mathbf{I})^{-1} \mathbf{x}_k\|} \quad (7.6.17)$$

迭代结果  $\{R_k, \mathbf{x}_k\}$  称为由 Rayleigh 迭代产生的 Rayleigh 序列。

Rayleigh 序列具有以下性质 [393]:

- (1) 尺度不变性 矩阵  $\alpha \mathbf{A}$  ( $\alpha \neq 0$ ) 产生与矩阵  $\mathbf{A}$  相同的 Rayleigh 序列。
- (2) 平移不变性 矩阵  $\mathbf{A} - \alpha \mathbf{I}$  产生的 Rayleigh 序列为  $\{R_k - \alpha, \mathbf{x}_k\}$ 。
- (3)酉相似性 矩阵  $\mathbf{U} \mathbf{A} \mathbf{U}^H$  ( $\mathbf{U}$  是酉矩阵) 产生的 Rayleigh 序列为  $\{R_k, \mathbf{U} \mathbf{x}_k\}$ 。

Rayleigh 商迭代算法还有下面的推广 [393], 它适用于一般矩阵  $\mathbf{A}$ 。选择一个单位长度的初始向量  $\mathbf{x}_0$ , 并针对  $k = 0, 2, 4, \dots$  执行下面的迭代运算:

- (1) 构造  $R_k = R(\mathbf{x}_k)$ 。
- (2) 求解  $\mathbf{x}_{k+1}^H (\mathbf{A} - R_k \mathbf{I}) = \tau_k \mathbf{x}_k^H$ , 并使  $\|\mathbf{x}_{k+1}\| = 1$ 。
- (3) 构造  $R_{k+1} = R(\mathbf{x}_{k+1})$ 。
- (4) 求解  $(\mathbf{A} - R_{k+1} \mathbf{I}) \mathbf{x}_{k+2} = \tau_{k+1} \mathbf{x}_{k+1}$ , 并使  $\|\mathbf{x}_{k+2}\| = 1$ 。

如果碰巧  $(\mathbf{A} - R_k \mathbf{I})$  或者  $(\mathbf{A} - R_{k+1} \mathbf{I})$  奇异, 则求解齐次方程, 直接得到相应的特征向量。若矩阵  $\mathbf{A} = \mathbf{A}^H$ , 则上述推广算法退化为原标准 Rayleigh 商迭代方法。

### 7.6.3 Rayleigh 商问题求解的共轭梯度算法

取 Rayleigh 商的梯度的负方向作为向量  $\mathbf{x}$  的梯度流, 即

$$\dot{\mathbf{x}} = -[\mathbf{A} - R(\mathbf{x}) \mathbf{I}] \mathbf{x} \quad (7.6.18)$$

则向量  $\mathbf{x}$  可以利用梯度算法迭代计算<sup>[224]</sup>

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mu \dot{\mathbf{x}}_k = \mathbf{x}_k - \mu [\mathbf{A} - R(\mathbf{x}_k)I]\mathbf{x}_k \quad (7.6.19)$$

正如下面的定理所述, Rayleigh 商问题求解的梯度算法具有比标准 Rayleigh 商迭代算法更快的收敛速率。

**定理 7.6.2** 假定  $\lambda_1 > \lambda_2$ 。对于几乎所有满足  $\|\mathbf{x}_0\| = 1$  的初始值  $\mathbf{x}_0$ , 由梯度算法迭代计算的向量  $\mathbf{x}_k$  以速率  $\lambda_1 - \lambda_2$  指数收敛为矩阵  $\mathbf{A}$  的最大特征向量  $\mathbf{u}_1$  或  $-\mathbf{u}_1$ 。

**证明** 参见文献 [224, p.18]。

下面介绍求解 Rayleigh 商问题

$$R(\mathbf{x}) = \frac{\mathbf{x}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{x}} \quad (7.6.20)$$

的共轭梯度算法, 式中,  $\mathbf{A}$  为实对称矩阵。

从某个初始向量  $\mathbf{x}_0$  出发, 共轭梯度算法使用迭代公式

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \quad (7.6.21)$$

更新和逼近对称矩阵的最小(或最大)特征向量。实系数  $\alpha_k$  由下式给出<sup>[449, 525]</sup>

$$\alpha_k = \pm \frac{1}{2D} \left( -B + \sqrt{B^2 - 4CD} \right) \quad (7.6.22)$$

式中, 正号适用于最小特征向量的更新, 负号对应于最大特征向量的更新。

式 (7.6.22) 的参数  $D, B, C$  的计算公式如下

$$D = P_b(k)P_c(k) - P_a(k)P_d(k) \quad (7.6.23)$$

$$B = P_b(k) - \lambda_k P_d(k) \quad (7.6.24)$$

$$C = P_a(k) - \lambda_k P_c(k) \quad (7.6.25)$$

$$P_a(k) = \mathbf{p}_k^T \mathbf{A} \mathbf{x}_k / (\mathbf{x}_k^T \mathbf{x}_k) \quad (7.6.26)$$

$$P_b(k) = \mathbf{p}_k^T \mathbf{A} \mathbf{p}_k / (\mathbf{x}_k^T \mathbf{x}_k) \quad (7.6.27)$$

$$P_c(k) = \mathbf{p}_k^T \mathbf{x}_k / (\mathbf{x}_k^T \mathbf{x}_k) \quad (7.6.28)$$

$$P_d(k) = \mathbf{p}_k^T \mathbf{p}_k / (\mathbf{x}_k^T \mathbf{x}_k) \quad (7.6.29)$$

$$\lambda_k = R(\mathbf{x}_k) = \mathbf{x}_k^T \mathbf{A} \mathbf{x}_k / (\mathbf{x}_k^T \mathbf{x}_k) \quad (7.6.30)$$

在第  $k+1$  步迭代, 搜索方向按照下列方式选择

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + b(k) \mathbf{p}_k \quad (7.6.31)$$

式中,  $b(-1) = 0$ , 且  $\mathbf{r}_{k+1}$  为  $k+1$  步迭代的残差向量, 由

$$\mathbf{r}_{k+1} = -\frac{1}{2} \nabla_{\mathbf{x}} R(\mathbf{x}_{k+1}) = (\lambda_{k+1} \mathbf{x}_{k+1} - \mathbf{A} \mathbf{x}_{k+1}) / (\mathbf{x}_{k+1}^T \mathbf{x}_{k+1}) \quad (7.6.32)$$

式 (7.6.31) 中的  $b(k)$  的选择应该使搜索方向  $\mathbf{p}_{k+1}$  与  $\mathbf{p}_k$  是相对于 Rayleigh 商的 Hessian 矩阵  $\mathbf{H}$  共轭的或者  $\mathbf{H}$  正交的, 即

$$\mathbf{p}_{k+1}^T \mathbf{H} \mathbf{p}_k = 0 \quad (7.6.33)$$

关于  $\mathbf{H}$  的选择, Chen 等人<sup>[103]</sup> 使用矩阵  $\mathbf{A}$  作  $\mathbf{H}$ 。此时

$$b(k) = -\frac{\mathbf{r}_{k+1}^T \mathbf{A} \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \quad (7.6.34)$$

但是, 这种选择只适用于二次型目标函数  $\mathbf{x}^H \mathbf{A} \mathbf{x}$ , 因为这种函数的 Hessian 矩阵等于  $\mathbf{A}$ 。

由于 Rayleigh 商不是二次型函数, 直接计算 Hessian 矩阵, 得<sup>[525]</sup>

$$\mathbf{H}(\mathbf{x}) = \frac{2}{\mathbf{x}^T \mathbf{x}} \left[ \mathbf{A} - \frac{\partial R(\mathbf{x})}{\partial \mathbf{x}} \mathbf{x}^T - \mathbf{x} \left( \frac{\partial R(\mathbf{x})}{\partial \mathbf{x}} \right)^T - R(\mathbf{x}) \mathbf{I} \right] \quad (7.6.35)$$

式中

$$\frac{\partial R(\mathbf{x})}{\partial \mathbf{x}} = -\frac{\mathbf{x}}{(\mathbf{x}^T \mathbf{x})^2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \frac{2}{\mathbf{x}^T \mathbf{x}} \mathbf{A} \mathbf{x} = -\frac{R(\mathbf{x})}{\mathbf{x}^T \mathbf{x}} \mathbf{x} + \frac{2}{\mathbf{x}^T \mathbf{x}} \mathbf{A} \mathbf{x} \quad (7.6.36)$$

当  $\mathbf{x} = \mathbf{x}_{k+1}$  时, 将 Hessian 矩阵简记为  $\mathbf{H}_{k+1} = \mathbf{H}(\mathbf{x}_{k+1})$ 。此时, 参数

$$b(k) = -\frac{\mathbf{r}_{k+1}^T \mathbf{H}_{k+1} \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{H}_{k+1} \mathbf{s}_j \mathbf{p}_k} \quad (7.6.37)$$

将  $\mathbf{H}_{k+1}$  代入后, 得

$$b(k) = -\frac{\mathbf{r}_{k+1}^T \mathbf{A} \mathbf{p}_k + (\mathbf{r}_{k+1}^T \mathbf{r}_{k+1})(\mathbf{x}_{k+1}^T \mathbf{p}_k)}{\mathbf{p}_k^T (\mathbf{A} \mathbf{p}_k - \lambda_{k+1} \mathbf{I}) \mathbf{p}_k} \quad (7.6.38)$$

式 (7.6.21) ~ 式 (7.6.30) 以及式 (7.6.38) 一起组成了求解 Rayleigh 商问题式 (7.6.20) 的共轭梯度算法。如果对更新的  $\mathbf{x}_k$  进行归一化, 并且当式 (7.6.22) 前面取正号时, 算法求出的是对称矩阵  $\mathbf{A}$  的最小特征值和对应的最小特征向量。若希望求  $\mathbf{A}$  的最大特征值和相应的最大特征向量, 则只要在式 (7.6.22) 前面取负号即可。这种算法是 Yang 等人提出的<sup>[525]</sup>。

## 7.7 广义 Rayleigh 商

Rayleigh 商的推广形式称为广义 Rayleigh 商。本节讨论广义 Rayleigh 商的定义和求解方法, 并以模式识别和移动通信为例, 介绍广义 Rayleigh 商的典型应用。

### 7.7.1 广义 Rayleigh 商的定义及性质

**定义 7.7.1** 令  $\mathbf{A}$  和  $\mathbf{B}$  均为  $n \times n$  维 Hermitian 矩阵, 且  $\mathbf{B}$  是正定矩阵。矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义 Rayleigh 商或广义 Rayleigh-Ritz 比  $R(\mathbf{x})$  是一个标量 (函数), 定义为

$$R(\mathbf{x}) = \frac{\mathbf{x}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{B} \mathbf{x}} \quad (7.7.1)$$

其中,  $\mathbf{x}$  是待选择的向量, 其目的是使广义 Rayleigh 商最大化或者最小化。

为了求解广义 Rayleigh 商, 定义一个新向量  $\tilde{\mathbf{x}} = \mathbf{B}^{1/2}\mathbf{x}$ , 其中,  $\mathbf{B}^{1/2}$  表示正定矩阵  $\mathbf{B}$  的平方根。用  $\mathbf{x} = \mathbf{B}^{-1/2}\tilde{\mathbf{x}}$  代入广义 Rayleigh 商定义式(7.7.1), 则有

$$R(\tilde{\mathbf{x}}) = \frac{\tilde{\mathbf{x}}^H (\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2}) \tilde{\mathbf{x}}}{\tilde{\mathbf{x}}^H \tilde{\mathbf{x}}} \quad (7.7.2)$$

这表明, 矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义 Rayleigh 商等价为矩阵乘积  $(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2})$  的 Rayleigh 商。由 Rayleigh-Ritz 定理知, 当选择向量  $\tilde{\mathbf{x}}$  是与矩阵乘积  $(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2})$  的最小特征值  $\lambda_{\min}$  对应的特征向量时, 广义 Rayleigh 商取最小值  $\lambda_{\min}$ ; 而当选择向量  $\tilde{\mathbf{x}}$  是与矩阵乘积  $(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2})$  的最大特征值  $\lambda_{\max}$  对应的特征向量时, 广义 Rayleigh 商取最大值  $\lambda_{\max}$ 。考查矩阵乘积  $(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2})$  的特征值分解

$$(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2}) \tilde{\mathbf{x}} = \lambda \tilde{\mathbf{x}} \quad (7.7.3)$$

若  $\mathbf{B} = \sum_{i=1}^n \beta_i \mathbf{v}_i \mathbf{v}_i^H$  是矩阵  $\mathbf{B}$  的特征值分解, 则

$$\mathbf{B}^{1/2} = \sum_{i=1}^n \sqrt{\beta_i} \mathbf{v}_i \mathbf{v}_i^H$$

并且  $\mathbf{B}^{1/2} \mathbf{B}^{1/2} = \mathbf{B}$ 。由于矩阵  $\mathbf{B}^{1/2}$  和其逆矩阵  $\mathbf{B}^{-1/2}$  具有相同的特征向量和互为倒数的特征值, 故

$$\mathbf{B}^{-1/2} = \sum_{i=1}^n \frac{1}{\sqrt{\beta_i}} \mathbf{v}_i \mathbf{v}_i^H \quad (7.7.4)$$

说明  $\mathbf{B}^{-1/2}$  也是 Hermitian 矩阵, 即有  $(\mathbf{B}^{-1/2})^H = \mathbf{B}^{-1/2}$ 。

用矩阵  $\mathbf{B}^{-1/2}$  左乘式 (7.7.3) 两边, 并代入  $(\mathbf{B}^{-1/2})^H = \mathbf{B}^{-1/2}$ , 即得

$$\mathbf{B}^{-1} \mathbf{A} \mathbf{B}^{-1/2} \tilde{\mathbf{x}} = \lambda \mathbf{B}^{-1/2} \tilde{\mathbf{x}} \quad \text{或} \quad \mathbf{B}^{-1} \mathbf{A} \mathbf{x} = \lambda \mathbf{x}$$

因为  $\mathbf{x} = \mathbf{B}^{-1/2} \tilde{\mathbf{x}}$ 。因此, 矩阵乘积  $(\mathbf{B}^{-1/2})^H \mathbf{A} (\mathbf{B}^{-1/2})$  的特征值分解与矩阵  $\mathbf{B}^{-1} \mathbf{A}$  的特征值分解等价。由于  $\mathbf{B}^{-1} \mathbf{A}$  的特征值分解就是矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值分解, 所以上述讨论可归结为: 广义 Rayleigh 商取最大值和最小值的条件是

$$R(\mathbf{x}) = \frac{\mathbf{x}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{B} \mathbf{x}} = \lambda_{\max}, \quad \text{若选择 } \mathbf{A} \mathbf{x} = \lambda_{\max} \mathbf{B} \mathbf{x} \quad (7.7.5)$$

$$R(\mathbf{x}) = \frac{\mathbf{x}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{B} \mathbf{x}} = \lambda_{\min}, \quad \text{若选择 } \mathbf{A} \mathbf{x} = \lambda_{\min} \mathbf{B} \mathbf{x} \quad (7.7.6)$$

即是说, 欲使广义 Rayleigh 商最大化, 向量  $\mathbf{x}$  必须选取与矩阵束  $(\mathbf{A}, \mathbf{B})$  最大广义特征值对应的特征向量; 反之, 需要使广义 Rayleigh 商最小化时, 则应该取与矩阵束  $(\mathbf{A}, \mathbf{B})$  最小广义特征值对应的特征向量作  $\mathbf{x}$ 。

### 7.7.2 应用举例 1: 类鉴别有效性的评估

模式识别广泛应用于人的特征(如人脸、指纹、虹膜等)识别和各种雷达目标(如飞机、舰船等)识别。在这些应用中,信号特征的提取是至关重要的。例如,将目标视为一个线性系统,系统的参数就是目标信号的一种特征。

散布(divergence)是两类信号间“距离”或相异度的一种测度,常常用于进行特征评价和类鉴别有效性的评估。

令 $Q$ 是待评估的几种方法抽取的信号特征向量的共同维数。假设共有 $c$ 类信号,Fisher类鉴别测度需要比较 $c-1$ 个类鉴别函数。作为Fisher测度的推广,现在考虑所有 $Q$ 维特征向量在 $c-1$ 维类鉴别空间上的投影。

令 $N=N_1+\cdots+N_c$ ,其中, $N_i$ 表示在训练阶段提取的第*i*类信号的特征向量的个数。假定

$$\mathbf{s}_{i,k} = [s_{i,k}(1), \dots, s_{i,k}(Q)]^T$$

表示在训练阶段由第*i*类信号的第*k*组观测数据得到的 $Q$ 维特征向量,而

$$\mathbf{m}_i = [m_i(1), \dots, m_i(Q)]^T$$

为第*i*类信号的特征向量的样本均值向量,其中

$$m_i(q) = \frac{1}{N_i} \sum_{k=1}^{N_i} s_{i,k}(q), \quad i = 1, \dots, c, \quad q = 1, \dots, Q$$

类似地,令

$$\mathbf{m} = [m(1), \dots, m(Q)]^T$$

表示由全体观测数据得到的所有特征向量的总体均值向量,其中

$$m(q) = \frac{1}{c} \sum_{i=1}^c m_i(q), \quad q = 1, \dots, Q$$

有了以上向量后,即可定义 $Q \times Q$ 类内散布矩阵(within-class scatter matrix)<sup>[149]</sup>

$$\mathbf{S}_w \stackrel{\text{def}}{=} \frac{1}{c} \sum_{i=1}^c \left[ \frac{1}{N_i} \sum_{k=1}^{N_i} (\mathbf{s}_{i,k} - \mathbf{m}_i)(\mathbf{s}_{i,k} - \mathbf{m}_i)^T \right] \quad (7.7.7)$$

和 $Q \times Q$ 类间散布矩阵(between-class scatter matrix)<sup>[149]</sup>

$$\mathbf{S}_b \stackrel{\text{def}}{=} \frac{1}{c} \sum_{i=1}^c (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \quad (7.7.8)$$

令 $\text{Span}(\mathbf{U})$ 为 $Q \times Q$ 矩阵 $\mathbf{U}$ 的列张成的 $Q$ 维子空间。定义准则函数

$$J(\mathbf{U}) \stackrel{\text{def}}{=} \frac{\prod_{\text{diag}} \mathbf{U}^T \mathbf{S}_b \mathbf{U}}{\prod_{\text{diag}} \mathbf{U}^T \mathbf{S}_w \mathbf{U}} \quad (7.7.9)$$

式中,  $\prod_{\text{diag}} \mathbf{A}$  表示矩阵  $\mathbf{A}$  的对角元素的乘积。作为评估类鉴别能力的测度, 应该使  $J$  最大化。称  $\text{Span}(\mathbf{U})$  是类鉴别空间, 若

$$\mathbf{U} = \underset{\mathbf{U} \in \mathbb{R}^{Q \times Q}}{\operatorname{argmax}} J(\mathbf{U}) = \frac{\prod_{\text{diag}} \mathbf{U}^T \mathbf{S}_b \mathbf{U}}{\prod_{\text{diag}} \mathbf{U}^T \mathbf{S}_w \mathbf{U}} \quad (7.7.10)$$

这一优化问题又可等价写作

$$[\mathbf{u}_1, \dots, \mathbf{u}_Q] = \underset{\mathbf{u}_i \in \mathbb{R}^Q}{\operatorname{argmax}} \frac{\prod_{i=1}^Q \mathbf{u}_i^T \mathbf{S}_b \mathbf{u}_i}{\prod_{i=1}^Q \mathbf{u}_i^T \mathbf{S}_w \mathbf{u}_i} = \prod_{i=1}^Q \frac{\mathbf{u}_i^T \mathbf{S}_b \mathbf{u}_i}{\mathbf{u}_i^T \mathbf{S}_w \mathbf{u}_i} \quad (7.7.11)$$

其解为

$$\mathbf{u}_i = \underset{\mathbf{u}_i \in \mathbb{R}^Q}{\operatorname{argmax}} \frac{\mathbf{u}_i^T \mathbf{S}_b \mathbf{u}_i}{\mathbf{u}_i^T \mathbf{S}_w \mathbf{u}_i}, \quad i = 1, \dots, Q \quad (7.7.12)$$

这恰好就是广义 Rayleigh 商的最大化。上式有着明确的物理意义: 构成最优秀类鉴别子空间的矩阵  $\mathbf{U}$  的列向量  $\mathbf{u}$  应该同时使得类间散布最大和类内散布最小, 即广义 Rayleigh 商最大化。

对于  $c$  类信号的分类, 最优的类鉴别子空间是  $c - 1$  维的。因此, 式 (7.7.12) 只需要对  $c - 1$  个广义 Rayleigh 商最大化。换言之, 只需要求解广义特征值问题

$$\mathbf{S}_b \mathbf{u}_i = \lambda_i \mathbf{S}_w \mathbf{u}_i, \quad i = 1, 2, \dots, c - 1 \quad (7.7.13)$$

得到  $c - 1$  个广义特征向量  $\mathbf{u}_1, \dots, \mathbf{u}_{c-1}$ 。这些广义特征向量构成的  $Q \times (c - 1)$  矩阵

$$\mathbf{U}_{c-1} = [\mathbf{u}_1, \dots, \mathbf{u}_{c-1}] \quad (7.7.14)$$

它的列张成最优的类鉴别子空间。

获得了矩阵  $Q \times (c - 1)$  矩阵  $\mathbf{U}_{c-1}$  后, 即可以对在训练阶段获得的每一个信号特征向量  $\mathbf{s}_{i,k}$ , 求出它在最优类鉴别子空间的投影

$$\mathbf{y}_{i,k} = \mathbf{U}_{c-1}^T \mathbf{s}_{i,k}, \quad i = 1, \dots, c, \quad k = 1, \dots, N_i \quad (7.7.15)$$

当只有三类信号 ( $c = 3$ ) 时, 最优的类鉴别子空间是一平面, 每个特征向量在最优类鉴别子空间上的投影为一个点。这些投影图直观地反映出不同特征向量在信号分类中的鉴别能力。

### 7.7.3 应用举例 2: 干扰抑制的鲁棒波束形成

在无线通信中, 基站若使用由多个天线 (称为阵元) 组成的天线阵列, 便可以通过空间处理, 达到分离多个同信道的用户, 从而检测出期望用户的信号。

考虑由  $M$  个全向性天线组成的阵列，并且  $K$  个窄带信号位于远场。由  $M$  个阵元接收到的观测信号向量为

$$\mathbf{y}(n) = \mathbf{d}(n) + \mathbf{i}(n) + \mathbf{e}(n) \quad (7.7.16)$$

式中， $\mathbf{e}(n)$  为  $M$  个阵元上的加性白噪声组成的向量，而

$$\mathbf{d}(n) = \mathbf{a}(\theta_0(n))s_0(n) \quad (7.7.17)$$

$$\mathbf{i}(n) = \sum_{k=1}^{K-1} \mathbf{a}(\theta_k(n))s_k(n) \quad (7.7.18)$$

分别为  $M$  个阵元接收到的期望信号向量和其他  $K-1$  个用户的干扰信号向量。其中， $\theta_0(n)$  和  $\theta_k(n)$  分别代表期望信号和第  $k$  个干扰信号的波达方向(角)，向量  $\mathbf{a}(\theta_0(n))$  和  $\mathbf{a}(\theta_k(n))$  分别是期望信号和第  $k$  个干扰信号的阵列响应向量。

假定所有信号源彼此统计不相关，各个阵元的加性白噪声统计不相关，并且具有相同方差  $\sigma^2$ 。于是，观测信号的自相关矩阵为

$$\mathbf{R}_y = \mathbb{E}\{\mathbf{y}(n)\mathbf{y}^H(n)\} = \mathbf{R}_d + \mathbf{R}_{i+e} \quad (7.7.19)$$

式中

$$\mathbf{R}_d = \mathbb{E}\{\mathbf{d}(n)\mathbf{d}^H(n)\} = P_0 \mathbf{a}(\theta_0) \mathbf{a}^H(\theta_0) \quad (7.7.20)$$

$$\mathbf{R}_{i+e} = \mathbb{E}\{[\mathbf{i}(n) + \mathbf{e}(n)][\mathbf{i}(n) + \mathbf{e}(n)]^H\} = \sum_{k=1}^{K-1} P_k \mathbf{a}(\theta_k) \mathbf{a}^H(\theta_k) + \sigma^2 \mathbf{I} \quad (7.7.21)$$

其中，常数  $P_0$  和  $P_k$  分别表示期望信号和第  $k$  个干扰信号的功率。

令  $\mathbf{w}(n)$  是波束形成器在  $n$  时刻的权向量，其输出

$$\mathbf{z}(n) = \mathbf{w}^H(n) \mathbf{y}(n) \quad (7.7.22)$$

容易求得波束形成器输出的信干噪比 (signal-to-interference-plus-noise-ratio, SINR) 为

$$\text{SINR}(\mathbf{w}) = \frac{\mathbb{E}\{|\mathbf{w}^H \mathbf{d}(n)|^2\}}{\mathbb{E}\{|\mathbf{w}^H [\mathbf{i}(n) + \mathbf{e}(n)]|^2\}} = \frac{\mathbf{w}^H \mathbf{R}_d \mathbf{w}}{\mathbf{w}^H \mathbf{R}_{i+e} \mathbf{w}} \quad (7.7.23)$$

为了达到干扰抑制之目的，应该使 Rayleigh 商 SINR ( $\mathbf{w}$ ) 最大化。即是说，干扰抑制的最优波束形成器应该选择为矩阵束  $\{\mathbf{R}_d, \mathbf{R}_{i+e}\}$  与最大广义特征值对应的广义特征向量。然而，这需要分别计算期望信号的自相关矩阵  $\mathbf{R}_d$  和干扰加噪声的自相关矩阵  $\mathbf{R}_{i+e}$ 。这是难于直接做到的。

将  $\mathbf{R}_d = P_0 \mathbf{a}(\theta_0) \mathbf{a}^H(\theta_0)$  代入信干噪比公式 (7.7.23)，无约束最优化问题可表示为

$$\max \text{SINR}(\mathbf{w}) = \max \frac{P_0 |\mathbf{w}^H \mathbf{a}(\theta_0)|^2}{\mathbf{w}^H \mathbf{R}_{i+e} \mathbf{w}} \quad (7.7.24)$$

若增加约束条件  $\mathbf{w}^H \mathbf{a}(\theta_0) = 1$ ，则无约束最优化问题等价为下列约束最优化问题

$$\min \mathbf{w}^H \mathbf{R}_{i+e} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \mathbf{a}(\theta_0) = 1 \quad (7.7.25)$$

这个最优化问题仍然不方便求解，因为  $\mathbf{R}_{i+e}$  不可能计算。

注意到

$$\mathbf{w}^H \mathbf{R}_y \mathbf{w} = \mathbf{w}^H \mathbf{R}_d \mathbf{w} + \mathbf{w}^H \mathbf{R}_{i+e} \mathbf{w} = P_0 + \mathbf{w}^H \mathbf{R}_{i+e} \mathbf{w}$$

而期望信号功率  $P_0$  是与波束形成器无关的常数，所以式 (7.7.25) 的约束最优化问题又等价为

$$\min \mathbf{w}^H \mathbf{R}_y \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \mathbf{a}(\theta_0) = 1 \quad (7.7.26)$$

与式 (7.7.25) 不同的是，观测信号向量的自相关矩阵  $\mathbf{R}_y$  容易估计。使用 Lagrangian 乘子法，容易求出约束最优化问题式 (7.7.26) 的解为

$$\mathbf{w}_{\text{opt}}(n) = \frac{\mathbf{R}_y^{-1} \mathbf{a}(\theta_0(n))}{\mathbf{a}^H(\theta_0(n)) \mathbf{R}_y^{-1} \mathbf{a}(\theta_0(n))} \quad (7.7.27)$$

无线通信干扰抑制的这一鲁棒波束形成器是文献 [427] 提出的。

## 7.8 二次特征值问题

在流体力学中流量的线性稳定性研究<sup>[62, 223]</sup>、声学系统的动态分析、结构力学中结构系统的振动分析、电路仿真<sup>[474]</sup>、微电子力学系统 (microelectronic mechanical system, MEMS) 的数学建模<sup>[118]</sup>、生物医学信号处理、时间序列预报、语音的线性预测编码<sup>[130]</sup>、多输入-多输出 (multiple input-multiple output, MIMO) 系统分析<sup>[474]</sup>、工业应用的偏微分方程的有限元分析<sup>[279, 381]</sup>以及线性代数问题的一些应用中，常常会遇到一个共同的问题——二次特征值问题 (quadratic eigenvalue problem, QEP)。

二次特征值问题与标准的特征值问题尤其是广义特征值问题，既存在密切的联系，又有着明显的不同。鉴于理论、方法及应用的重要性，本节对二次特征值问题进行专门讨论与介绍。

### 7.8.1 二次特征值问题的描述

具有粘滞阻尼和无外力作用的结构系统，其运动方程为微分方程<sup>[252, 434]</sup>

$$\mathbf{M} \ddot{\mathbf{x}} + \mathbf{C} \dot{\mathbf{x}} + \mathbf{K} \mathbf{x} = \mathbf{0} \quad (7.8.1)$$

式中， $\mathbf{M}$ 、 $\mathbf{C}$ 、 $\mathbf{K}$  分别为质量矩阵、阻尼矩阵和刚度矩阵；而向量  $\ddot{\mathbf{x}}$ 、 $\dot{\mathbf{x}}$  和  $\mathbf{x}$  分别为加速度、速度和位移向量。

在振动分析中，齐次线性方程式 (7.8.1) 的通解形式为

$$\mathbf{x} = e^{\lambda t} \mathbf{u} \quad (7.8.2)$$

式中,  $\mathbf{u}$  通常为复向量, 而  $\lambda$  为特征值, 它一般也是复数。将式 (7.8.2) 及其关于时间的导数代入式 (7.8.1), 便得到特征方程

$$(\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K})\mathbf{u} = \mathbf{0}$$

由于某些矩阵是非对称的, 上式左边也存在形如

$$\mathbf{v}^H (\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K}) = \mathbf{0}^T$$

的解。显然, 以上两个方程是关于特征值的二次方程, 简称二次特征值问题。

抽去物理含义, 可以将二次特征值问题叙述为<sup>[295]</sup>: 求标量  $\lambda$  和非零向量  $\mathbf{u}, \mathbf{v}$ , 使它们满足方程

$$(\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K})\mathbf{u} = \mathbf{0}, \quad \mathbf{v}^H (\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K}) = \mathbf{0}^T \quad (7.8.3)$$

式中,  $\mathbf{M}, \mathbf{C}, \mathbf{K}$  为  $n \times n$  复矩阵。满足上述方程的标量  $\lambda$  称为特征值, 非零向量  $\mathbf{u}$  和  $\mathbf{v}$  分别称为与特征值  $\lambda$  对应的右和左特征向量。特征值和特征向量组成二次特征值问题的特征对。

特征值问题  $\mathbf{A}_{n \times n}\mathbf{u} = \lambda\mathbf{u}$  的特征方程为  $|\mathbf{A} - \lambda\mathbf{I}| = 0$ , 广义特征值问题  $\mathbf{A}_{n \times n}\mathbf{u} = \lambda\mathbf{B}_{n \times n}\mathbf{u}$  的特征方程为  $|\mathbf{A} - \lambda\mathbf{B}| = 0$  都是关于特征值的一次方程。与之不同, 式 (7.8.3) 的特征值由二次特征方程  $|\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K}| = 0$  决定, 故称为二次特征值。显然, 二次特征值问题存在  $2n$  个特征值 (有限大或者无穷大),  $2n$  个右特征向量以及  $2n$  个左特征向量。在工程应用中, 特征值通常为复数, 其虚部为谐振频率, 实部表示指数阻尼, 并且希望得到在频率范围内所有的特征值。一般情况下, 特征对有几十个到几百个之多。

二次特征值问题是非线性特征值问题的一个重要子类。令

$$\mathbf{Q}(\lambda) = \lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K} \quad (7.8.4)$$

这是一个二次  $n \times n$  矩阵多项式。换言之, 矩阵  $\mathbf{Q}(\lambda)$  的系数是  $\lambda$  的二次多项式。通常称  $\mathbf{Q}(\lambda)$  为  $\lambda$  矩阵<sup>[293]</sup>。于是, 特征值  $\lambda$  是特征方程

$$|\mathbf{Q}(z)| = |z^2 \mathbf{M} + z \mathbf{C} + \mathbf{K}| = 0 \quad (7.8.5)$$

的根, 并称  $|\mathbf{Q}(\lambda)|$  为特征多项式。

**定义 7.8.1**<sup>[474]</sup> 矩阵  $\mathbf{Q}(\lambda)$  称为正则  $\lambda$  矩阵, 若特征多项式  $|\mathbf{Q}(z)|$  对所有  $z$  值不恒等于零。反之, 若  $|\mathbf{Q}(z)| \equiv 0, \forall z$ , 则称矩阵  $\mathbf{Q}(z)$  是非正则  $\lambda$  矩阵。

在非正则矩阵的情况下, 存在无穷多个特征值, 因此这种矩阵不在考虑之列。下面假定矩阵  $\mathbf{Q}(z)$  为正则矩阵, 或等价假定  $\lambda$  矩阵  $\mathbf{Q}(\lambda)$  为正则矩阵。对于一个正则的  $\lambda$  矩阵  $\mathbf{Q}(\lambda)$ , 两个不同的特征值可能有同一个特征向量。

表 7.8.1 汇总了二次特征值问题的特征值与特征向量的性质。

表 7.8.1 二次特征值问题的特征值与特征向量的性质<sup>[474]</sup>

编号	矩阵性质	特征值性质	特征向量性质
1	$\mathbf{M}$ 非奇异	$2n$ 个有限大特征值	
2	$\mathbf{M}$ 奇异	有限大和无穷大特征值	
3	$\mathbf{M}, \mathbf{C}, \mathbf{K}$ 为实矩阵	实或共轭成对 $(\lambda, \lambda^*)$	若 $\mathbf{u}$ 是 $\lambda$ 的右特征向量, 则 $\mathbf{u}^*$ 是 $\lambda^*$ 的右特征向量
4	$\mathbf{M}, \mathbf{C}, \mathbf{K}$ 为 Hermitian 矩阵	实或共轭成对 $(\lambda, \lambda^*)$	若 $\mathbf{u}$ 是 $\lambda$ 的右特征向量, 则 $\mathbf{u}^*$ 是 $\lambda^*$ 的右特征向量
5	$\mathbf{M}$ : Hermitian 正定 $\mathbf{C}, \mathbf{K}$ : Hermitian 半正定	$\operatorname{Re}(\lambda) \leq 0$	
6	$\mathbf{M}, \mathbf{C}$ 对称正定 $\mathbf{K}$ 对称半正定 $\gamma(\mathbf{M}, \mathbf{C}, \mathbf{K}) > 0$	$\lambda$ 取正和负, $n$ 个大特征值, $n$ 个小特征值	$n$ 个大特征值 (或 $n$ 个小特征值) 对应的 $n$ 个特征向量线性无关
7	$\mathbf{M}, \mathbf{K}$ : Hermitian 矩阵 $\mathbf{M}$ 正定, $\mathbf{C} = -\mathbf{C}^H$	特征值为纯虚数 或共轭成对 $(\lambda, \lambda^*)$	若 $\mathbf{u}$ 是 $\lambda$ 的右特征向量, 则 $\mathbf{u}$ 是 $-\lambda^*$ 的左特征向量
8	$\mathbf{M}, \mathbf{K}$ 实对称正定 $\mathbf{C} = -\mathbf{C}^T$	特征值为纯虚数	

表中,  $\gamma(\mathbf{M}, \mathbf{C}, \mathbf{K}) = \min \left\{ (\mathbf{u}^H \mathbf{C} \mathbf{u})^2 - 4(\mathbf{u}^H \mathbf{M} \mathbf{u})(\mathbf{u}^H \mathbf{K} \mathbf{u}) : \|\mathbf{u}\|_2 = 1 \right\}$ 。

需要指出, 根据二次型函数的大小, 二次特征值问题又可进一步分类如下<sup>[215]</sup>:

- (1) 二次型函数满足  $(\mathbf{u}^H \mathbf{C} \mathbf{u})^2 < 4(\mathbf{u}^H \mathbf{M} \mathbf{u})(\mathbf{u}^H \mathbf{K} \mathbf{u})$  的二次特征值问题式 (7.8.1) 称为椭圆二次特征值问题 (elliptic QEP)。
- (2) 二次型函数满足  $(\mathbf{u}^H \mathbf{C} \mathbf{u})^2 > 4(\mathbf{u}^H \mathbf{M} \mathbf{u})(\mathbf{u}^H \mathbf{K} \mathbf{u})$  的二次特征值问题式 (7.8.1) 称为双曲线二次特征值问题 (hyperbolic QEP)。

## 7.8.2 二次特征值问题求解

求解二次特征值问题有以下两种主要方法:

- (1) 分解法 基于广义 Bezout 定理, 将二次特征值问题分解为两个一次特征值子问题。
- (2) 线性化方法 通过线性化手段, 将非线性的二次特征值问题变为线性广义特征值问题。

下面分别介绍这两种方法。

### 1. 分解法

定义矩阵

$$\mathbf{Q}(S) = \mathbf{MS}^2 + \mathbf{CS} + \mathbf{K}, \quad S \in \mathbb{C}^{n \times n} \quad (7.8.6)$$

则它与  $\mathbf{Q}(\lambda)$  之差为

$$\mathbf{Q}(\lambda) - \mathbf{Q}(S) = \mathbf{M}(\lambda^2 \mathbf{I} - \mathbf{S}^2) + \mathbf{C}(\lambda \mathbf{I} - \mathbf{S}) = (\lambda \mathbf{M} + \mathbf{MS} + \mathbf{C})(\lambda \mathbf{I} - \mathbf{S}) \quad (7.8.7)$$

这一结果称为二次矩阵多项式的广义 Bezout 定理<sup>[186]</sup>。

如果二次矩阵方程

$$\mathbf{Q}(S) = \mathbf{M}S^2 + \mathbf{C}S + \mathbf{K} = \mathbf{O} \quad (7.8.8)$$

存在一个解  $S \in \mathbb{C}^{n \times n}$ , 则称这一解为二次矩阵方程的(右)解(right solvent)。类似地, 方程  $\mathbf{S}^2\mathbf{M} + \mathbf{SC} + \mathbf{K} = \mathbf{O}$  的解称为二次矩阵方程的左解(left solvent)<sup>[474]</sup>。显然, 若  $S$  是二次矩阵方程  $\mathbf{Q}(S) = \mathbf{O}$  的解, 则广义 Bezout 定理的公式(7.8.7)简化为

$$\mathbf{Q}(\lambda) = (\lambda\mathbf{M} + \mathbf{MS} + \mathbf{C})(\lambda\mathbf{I} - \mathbf{S}) \quad (7.8.9)$$

式(7.8.9)表明, 若  $S$  是二次矩阵方程(7.8.8)的解, 则二次特征值问题  $|\mathbf{Q}(\lambda)| = |\lambda^2\mathbf{M} + \lambda\mathbf{C} + \mathbf{K}| = 0$  等价为  $|\mathbf{Q}(\lambda)| = |(\lambda\mathbf{M} + \mathbf{MS} + \mathbf{C})(\lambda\mathbf{I} - \mathbf{S})| = 0$ 。由矩阵乘积的行列式性质  $|\mathbf{AB}| = |\mathbf{A}||\mathbf{B}|$  知, 二次特征值问题变成了以下两个(一次)特征值子问题:

- (1)  $n$ 个二次特征值是特征方程  $|\lambda\mathbf{M} + \mathbf{MS} + \mathbf{C}| = 0$  的解;
- (2) 另外  $n$ 个二次特征值是特征方程  $|\lambda\mathbf{I} - \mathbf{S}| = 0$  的解。

由于  $|\lambda\mathbf{M} + \mathbf{MS} + \mathbf{C}| = |\mathbf{MS} + \mathbf{C} - \lambda(-\mathbf{M})|$ , 故特征值问题  $|\lambda\mathbf{M} + \mathbf{MS} + \mathbf{C}| = 0$  与广义特征值问题  $(\mathbf{MS} + \mathbf{C})\mathbf{u} = \lambda(-\mathbf{M})\mathbf{u}$  等价。

总结以上讨论知, 从广义 Bezout 定理出发, 二次特征值问题  $(\lambda^2\mathbf{M} + \lambda\mathbf{C} + \mathbf{K})\mathbf{u} = \mathbf{0}$  可以分解为两个一次特征值问题:

- (1) 矩阵束  $(\mathbf{MS} + \mathbf{C}, -\mathbf{M})$  的广义特征值问题, 即  $(\mathbf{MS} + \mathbf{C})\mathbf{u} = -\lambda\mathbf{Mu}$ ;
- (2) 矩阵  $S$  的标准特征值问题  $S\mathbf{u} = \lambda\mathbf{u}$ 。

也就是说, 二次特征值问题的  $2n$  个特征对  $(\lambda_i, \mathbf{u}_i)$  由广义特征值问题  $(\mathbf{MS} + \mathbf{C})\mathbf{u} = -\lambda\mathbf{Mu}$  的  $n$  个广义特征对以及二次矩阵方程式(7.8.8)的解矩阵  $S$  的  $n$  个特征对共同组成。

## 2. 线性化方法

求解非线性方程的常用思路之一是将非线性方程线性化, 变为线性方程后再求解。这一思想同样适用于二次特征值问题的求解, 因为二次特征值问题本身就是非线性特征问题的一个重要子类。

若令  $\mathbf{z} = \begin{bmatrix} \lambda\mathbf{x} \\ \mathbf{x} \end{bmatrix}$ , 则特征方程  $(\lambda^2\mathbf{M} + \lambda\mathbf{C} + \mathbf{K})\mathbf{x} = \mathbf{0}$  可以写成等价形式  $\mathbf{L}_c(\lambda)\mathbf{z} = \mathbf{0}$ , 其中

$$\mathbf{L}_c(\lambda) = \lambda \begin{bmatrix} \mathbf{M} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} - \begin{bmatrix} -\mathbf{C} & -\mathbf{K} \\ \mathbf{I} & \mathbf{O} \end{bmatrix} \quad (7.8.10)$$

或者

$$\mathbf{L}_c(\lambda) = \lambda \begin{bmatrix} \mathbf{M} & \mathbf{C} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} - \begin{bmatrix} \mathbf{O} & -\mathbf{K} \\ \mathbf{I} & \mathbf{O} \end{bmatrix} \quad (7.8.11)$$

式中,  $\mathbf{L}_c(\lambda)$  称为  $\mathbf{Q}(\lambda)$  的友型(companion form)或线性化  $\lambda$  矩阵。

友型矩阵分为第1友型(first companion form)

$$L1: \quad \mathbf{A} = \begin{bmatrix} -\mathbf{C} & -\mathbf{K} \\ \mathbf{I} & \mathbf{O} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{M} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \quad (7.8.12)$$

和第 2 友型 (second companion form)

$$L2: \quad A = \begin{bmatrix} O & -K \\ I & O \end{bmatrix}, \quad B = \begin{bmatrix} M & C \\ O & I \end{bmatrix} \quad (7.8.13)$$

于是, 经过线性化, 二次特征值问题  $Q(\lambda)x = 0$  变成了广义特征值问题  $L_c(\lambda)z = 0$  或  $Az = \lambda Bz$ 。

如 7.5 节所述, 为了保证矩阵束  $(A, B)$  的广义特征值分解是唯一确定的, 矩阵  $B$  必须是非奇异矩阵。由式 (7.8.12) 和式 (7.8.13) 知, 这相当于要求矩阵  $M$  非奇异。

#### 算法 7.8.1 矩阵 $M$ 非奇异时求解二次特征值问题的线性化算法 [474]

输入  $\lambda$  矩阵  $P(\lambda) = \lambda^2 M + \lambda C + K$ 。

步骤 1 利用线性化公式 (7.8.12) 或者式 (7.8.13) 构造矩阵束  $(A, B)$ 。

步骤 2 使用 QZ 分解计算广义 Schur 分解

$$T = Q^H A Z, \quad S = Q^H B Z \quad (7.8.14)$$

式中,  $T$  和  $S$  为上三角矩阵 (对角元素分别为  $t_{kk}$  和  $s_{kk}$ ), 而  $Q$  和  $Z$  为酉矩阵。

步骤 3 计算二次特征值及其对应的特征向量

for  $k = 1 : 2n$

$$\lambda_k = t_{kk}/s_{kk}$$

求解  $(T - \lambda_k S)\phi = 0$ , 并令  $\xi = Z\phi$

$$\xi_1 = \xi(1 : n); \quad \xi_2 = \xi(n + 1 : 2n)$$

$$r_1 = P(\lambda_k)\xi_1/\|\xi_1\|; \quad r_2 = P(\lambda_k)\xi_2/\|\xi_2\|$$

$$u_k = \begin{cases} \xi(1 : n), & \text{若 } \|r_1\| \leq \|r_2\| \\ \xi(n + 1 : 2n), & \text{其他} \end{cases}$$

endfor

步骤 2 的 QZ 分解可以直接使用 MATLAB 程序的 `qz` 函数运行。顺便指出, 当上三角矩阵  $S$  的某个对角元素  $s_{ii} = 0$  时, 则特征值  $\lambda = \infty$ 。

如果矩阵  $M, C$  和  $K$  均为对称矩阵, 并且  $A - \lambda B$  是  $P(\lambda) = \lambda^2 M + \lambda C + K$  的对称线性化时, 算法 7.8.1 将不能保证  $A - \lambda B$  的对称性, 这是因为步骤 2 采用的 QZ 分解不能保证  $(A, B)$  的对称性。在这种情况下, 需要改用以下方法 [474]:

(1) 若  $B$  为确定的矩阵, 则先计算  $B$  的 Cholesky 分解  $B = LL^T$ , 其中,  $L$  为下三角矩阵。

(2) 将  $B = LL^T$  代入对称的广义特征值问题  $A\xi = \lambda B\xi$ , 变成对称的标准特征值问题  $L^{-1}AL^{-T}\phi = \lambda\phi$ , 其中,  $\phi = L^T\xi$ 。

(3) 利用对称 QR 分解, 计算对称矩阵  $L^{-1}AL^{-T}$  的特征对  $(\lambda, \phi)$ 。这些特征对即是二次特征值问题待求的特征对。

文献 [339] 介绍了线性化后二次特征值分解的广义 Davidson 算法、修正 Davidson 算法、二次残差迭代法等几种方法。

算法 7.8.1 只适用于矩阵  $\mathbf{M}$  非奇异的情况。然而，在一些工业应用中，矩阵  $\mathbf{M}$  往往是奇异矩阵。例如，在阻尼结构的有限元分析中，经常遇到所谓的无质量自由度 (massless degree of freedom)，它们对应为质量矩阵  $\mathbf{M}$  的某些列为零向量 [279]。

当矩阵  $\mathbf{M}$  奇异，从而使  $\mathbf{B}$  也奇异时，有两种方法可以改进原二次特征值问题 [279]。

一种方法使用谱变换 (spectral transformation)，即引入一个适当的特征值位移量  $\lambda_0$ ，变为

$$\mu = \lambda - \lambda_0 \quad (7.8.15)$$

于是，典型的 I 型线性化变为

$$\begin{bmatrix} -\mathbf{C} - \lambda_0 \mathbf{M} & -\mathbf{K} \\ \mathbf{I} & -\lambda_0 \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \mathbf{u} \end{bmatrix} = \mu \begin{bmatrix} \mathbf{M} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \mathbf{u} \end{bmatrix} \quad (7.8.16)$$

注意，位移  $\lambda_0$  的适当选择可以保证矩阵  $\begin{bmatrix} -\mathbf{C} - \lambda_0 \mathbf{M} & -\mathbf{K} \\ \mathbf{I} & -\lambda_0 \mathbf{I} \end{bmatrix}$  非奇异。

另一改进是令

$$\alpha = \frac{1}{\mu} \quad (7.8.17)$$

将  $\mu = 1/\alpha$  代入式 (7.8.16)，并予以重排，即得

$$\begin{bmatrix} -\mathbf{C} - \lambda_0 \mathbf{M} & -\mathbf{K} \\ \mathbf{I} & -\lambda_0 \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{M} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \mathbf{u} \end{bmatrix} = \alpha \begin{bmatrix} \dot{\mathbf{u}} \\ \mathbf{u} \end{bmatrix} \quad (7.8.18)$$

上式是标准的特征值分解

$$\mathbf{A}\mathbf{x} = \alpha\mathbf{x} \quad (7.8.19)$$

式中

$$\mathbf{A} = \begin{bmatrix} -\mathbf{C} - \lambda_0 \mathbf{M} & -\mathbf{K} \\ \mathbf{I} & -\lambda_0 \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{M} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \quad (7.8.20)$$

因此，求解特征值问题  $\mathbf{A}\mathbf{x} = \alpha\mathbf{x}$ ，即可得到特征值  $\alpha$  和与之对应的特征向量。然后，又可由

$$\lambda = \mu + \lambda_0 = \frac{1}{\alpha} + \lambda_0 \quad (7.8.21)$$

确定二次特征值问题的特征值。

有必要指出，任何一个高次特征值问题

$$(\lambda^m \mathbf{A}_m + \lambda^{m-1} \mathbf{A}_{m-1} + \cdots + \lambda \mathbf{A}_1 + \mathbf{A}_0) \mathbf{u} = \mathbf{0} \quad (7.8.22)$$

都可以线性化，例如 [339]

$$\begin{bmatrix} -\mathbf{A}_0 & & & \\ & \mathbf{I} & & \\ & & \ddots & \\ & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda\mathbf{u} \\ \vdots \\ \lambda^{m-1}\mathbf{u} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \cdots & \mathbf{A}_m \\ \mathbf{I} & \mathbf{O} & \cdots & \mathbf{O} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{O} & \cdots & \mathbf{I} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda\mathbf{u} \\ \vdots \\ \lambda^{m-1}\mathbf{u} \end{bmatrix} \quad (7.8.23)$$

也就是说， $m$  次特征值问题也可以线性化为标准的广义特征值问题  $\mathbf{Ax} = \lambda\mathbf{Bx}$  求解。

### 7.8.3 应用举例

下面介绍二次特征值的几个应用例子。

#### 1. AR 参数估计

考虑将实随机过程建模成自回归 (AR) 过程

$$x(n) + a(1)x(n-1) + \cdots + a(p)x(n-p) = e(n) \quad (7.8.24)$$

式中,  $a(1), \dots, a(p)$  为 AR 参数,  $p$  为 AR 阶数,  $e(n)$  为不可观测的激励信号, 通常为白噪声, 其方差为  $\sigma_e^2$ 。

用  $x(n-\tau), \tau \geq 1$  同乘上式两边, 并取数学期望, 得线性法方程

$$R_x(\tau) + a(1)R_x(\tau-1) + \cdots + a(p)R_x(\tau-p) = 0, \quad \tau = 1, 2, \dots \quad (7.8.25)$$

式中,  $R_x(\tau) = E\{x(n)x(n-\tau)\}$  表示 AR 过程的自相关函数。这一法方程称为 Yule-Walker 方程。

取  $\tau = 1, \dots, p$ , 则式 (7.8.25) 可以写作

$$\begin{bmatrix} R_x(0) & R_x(-1) & \cdots & R_x(-p+1) \\ R_x(1) & R_x(0) & \cdots & R_x(-p+2) \\ \vdots & \vdots & \ddots & \vdots \\ R_x(p-1) & R_x(p-2) & \cdots & R_x(0) \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{bmatrix} = - \begin{bmatrix} R_x(1) \\ R_x(2) \\ \vdots \\ R_x(p) \end{bmatrix} \quad (7.8.26)$$

然而, 在许多情况下存在观测噪声  $v(n)$ , 即实际观测信号为

$$y(n) = x(n) + v(n)$$

式中,  $v(n)$  为白噪声, 其方差为  $\sigma^2$ , 并与  $x(n)$  统计不相关。在这一假设下, 观测信号  $y(n)$  与 AR 随机过程  $x(n)$  的自相关函数之间存在下列关系

$$R_x(\tau) = R_y(\tau) - \sigma^2 \delta(\tau) = \begin{cases} R_y(0) - \sigma^2, & \tau = 0 \\ R_y(\tau), & \tau \neq 0 \end{cases}$$

将这一关系代入式 (7.8.26) 后, Yule-Walker 方程变为

$$\begin{bmatrix} R_y(0) - \sigma^2 & R_y(-1) & \cdots & R_y(-p+1) \\ R_y(1) & R_y(0) - \sigma^2 & \cdots & R_y(-p+2) \\ \vdots & \vdots & \ddots & \vdots \\ R_y(p-1) & R_y(p-2) & \cdots & R_y(0) - \sigma^2 \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{bmatrix} = - \begin{bmatrix} R_y(1) \\ R_y(2) \\ \vdots \\ R_y(p) \end{bmatrix} \quad (7.8.27)$$

称为噪声补偿的 Yule-Walker 方程<sup>[474]</sup>。

令  $\mathbf{a} = [-a(1), -a(2), \dots, -a(p)]^T$ ,  $\mathbf{r}_1 = [R_y(1), R_y(2), \dots, R_y(p)]^T$  和

$$\mathbf{R}_y = \begin{bmatrix} R_y(0) & R_y(-1) & \cdots & R_y(-p+1) \\ R_y(1) & R_y(0) & \cdots & R_y(-p+2) \\ \vdots & \vdots & \ddots & \vdots \\ R_y(p-1) & R_y(p-2) & \cdots & R_y(0) \end{bmatrix}$$

则式 (7.8.27) 可改写为

$$(\mathbf{R}_y - \sigma^2 \mathbf{I}_p) \mathbf{a} = \mathbf{r}_1 \quad (7.8.28)$$

由于  $R_y(\tau) = R_x(\tau)$ ,  $\tau \geq 1$ , 故若令  $\tau = p+1, p+2, \dots, p+q$ , 则由式 (7.8.25) 得

$$\left. \begin{array}{l} \mathbf{g}_1^T \mathbf{a} = R_y(p+1) \\ \mathbf{g}_2^T \mathbf{a} = R_y(p+2) \\ \vdots \\ \mathbf{g}_q^T \mathbf{a} = R_y(p+q) \end{array} \right\} \quad (7.8.29)$$

式中,  $\mathbf{g}_i = [R_y(p+i-1), R_y(p+i-2), \dots, R_y(i)]^T$ 。将式 (7.8.28) 和式(7.8.29) 合并, 即可得到矩阵方程

$$(\bar{\mathbf{R}}_y - \lambda \mathbf{D}) \mathbf{v} = \mathbf{0}_{p+q} \quad (7.8.30)$$

式中,  $\bar{\mathbf{R}}_y$  和  $\mathbf{D}$  均为  $(p+q) \times (p+1)$  矩阵, 且  $\mathbf{v}$  是  $(p+1) \times 1$  向量, 它们定义为

$$\bar{\mathbf{R}}_y = \begin{bmatrix} R_y(1) & R_y(0) & R_y(-1) & \cdots & R_y(-p+1) \\ R_y(2) & R_y(1) & R_y(0) & \cdots & R_y(-p+2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_y(p) & R_y(p-1) & R_y(p-2) & \cdots & R_y(0) \\ R_y(p+1) & R_y(p) & R_y(2) & \cdots & R_y(1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_y(p+q) & R_y(p+q-1) & R_y(p+q-2) & \cdots & R_y(q) \end{bmatrix}$$

和

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 1 \\ a(1) \\ a(2) \\ \vdots \\ a(p) \end{bmatrix}$$

用  $(\bar{\mathbf{R}}_y - \lambda \mathbf{D})^T$  左乘式 (7.8.30) 两边, 即有

$$(\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K}) \mathbf{v} = \mathbf{0}_{p+1} \quad (7.8.31)$$

式中

$$\mathbf{M} = \bar{\mathbf{R}}_y^T \bar{\mathbf{R}}_y, \quad \mathbf{C} = -(\bar{\mathbf{R}}_y^T \mathbf{D} + \mathbf{D}^T \bar{\mathbf{R}}_y), \quad \mathbf{K} = \mathbf{D}^T \mathbf{D} \quad (7.8.32)$$

由于矩阵  $\mathbf{M}, \mathbf{C}$  和  $\mathbf{K}$  均为对称矩阵, 故式 (7.8.31) 为对称二次特征值问题。

令

$$\mathbf{A} = \begin{bmatrix} \mathbf{K} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -\mathbf{C} & -\mathbf{M} \\ \mathbf{I} & \mathbf{O} \end{bmatrix} \quad (7.8.33)$$

则二次特征值问题变为  $2(p+1)$  维广义特征值问题

$$(\mathbf{A} - \lambda \mathbf{B})\mathbf{u} = \mathbf{0} \quad (7.8.34)$$

由于特征值是实的或共轭成对  $(\lambda, \lambda^*)$  出现，并且与  $(\lambda, \lambda^*)$  对应的右特征向量也为共轭对  $(\mathbf{u}, \mathbf{u}^*)$ 。在  $2(p+1)$  个特征对  $(\lambda_i, \mathbf{u}_i)$  中，只有同时满足式 (7.8.28) 和式 (7.8.29) 的特征值  $\lambda$  及其对应的特征向量  $\mathbf{u}$  才分别是待求的观测噪声方差  $\sigma^2$  和 AR 随机过程参数向量  $[1, a(1), \dots, a(p)]^T$  的估计。

由被加性白噪声污染的观测数据估计 AR 随机过程参数的上述方法是 Davila 于 1998 年提出的<sup>[130]</sup>。

## 2. 约束最小二乘

考虑下面的约束最小二乘问题

$$\mathbf{x} = \arg \min \left\{ \mathbf{x}^T \mathbf{A} \mathbf{x} - 2 \mathbf{b}^T \mathbf{x} \right\} \quad (7.8.35)$$

约束条件为  $\mathbf{x}^T \mathbf{x} = c^2$ 。其中， $\mathbf{A} \in \mathbb{R}^{n \times n}$  为对称矩阵； $c$  为不等于 0 的实常数，在很多应用中常取  $c = 1$ 。

这个约束最优化问题可以用 Lagrangian 乘子法求解。令代价函数

$$J(\mathbf{x}, \lambda) = \mathbf{x}^T \mathbf{A} \mathbf{x} - 2 \mathbf{b}^T \mathbf{x} + \lambda(c^2 - \mathbf{x}^T \mathbf{x}) \quad (7.8.36)$$

由  $\frac{\partial J(\mathbf{x}, \lambda)}{\partial \mathbf{x}} = \mathbf{0}$  和  $\frac{\partial J(\mathbf{x}, \lambda)}{\partial \lambda} = 0$  分别得

$$(\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{b}, \quad \mathbf{x}^T \mathbf{x} = c^2 \quad (7.8.37)$$

令  $\mathbf{x} = (\mathbf{A} - \lambda \mathbf{I})\mathbf{y}$ ，将这一假设代入式(7.8.37)的第一个式子，即得

$$(\mathbf{A} - \lambda \mathbf{I})^2 \mathbf{y} = \mathbf{b} \quad \text{或} \quad (\lambda^2 \mathbf{I} - 2\lambda \mathbf{A} + \mathbf{A}^2)\mathbf{y} - \mathbf{b} = \mathbf{0} \quad (7.8.38)$$

利用  $\mathbf{x} = (\mathbf{A} - \lambda \mathbf{I})\mathbf{y}$  及  $\mathbf{A}$  为对称矩阵之假设，易知

$$\begin{aligned} \mathbf{x}^T \mathbf{x} &= \mathbf{y}^T (\mathbf{A} - \lambda \mathbf{I})^T (\mathbf{A} - \lambda \mathbf{I}) \mathbf{y} = \mathbf{y}^T (\mathbf{A} - \lambda \mathbf{I}) [(\mathbf{A} - \lambda \mathbf{I}) \mathbf{y}] \\ &= \mathbf{y}^T (\mathbf{A} - \lambda \mathbf{I}) \mathbf{x} = \mathbf{y}^T \mathbf{b} \end{aligned}$$

于是，式 (7.8.37) 的第二个式子可以等价写作

$$\mathbf{y}^T \mathbf{b} = c^2 \quad \text{或} \quad 1 = \mathbf{y}^T \mathbf{b} / c^2$$

由此得

$$\mathbf{b} = \mathbf{b} \mathbf{y}^T \mathbf{b} / c^2 = \mathbf{b} \mathbf{b}^T \mathbf{y} / c^2$$

因为  $\mathbf{y}^T \mathbf{b} = \mathbf{b}^T \mathbf{y}$ 。将上式代入式 (7.8.38)，即有

$$\left[ \lambda^2 \mathbf{I} - 2\lambda \mathbf{A} + \left( \mathbf{A}^2 - c^{-2} \mathbf{b} \mathbf{b}^T \right) \right] \mathbf{y} = \mathbf{0} \quad (7.8.39)$$

这恰好是一个对称的二次特征值问题。

Gander 等人<sup>[181]</sup> 业已证明, 约束最小二乘问题的求解需要 Lagrangian 乘子  $\lambda$  的最小二乘解。

总结以上讨论, 可以得出约束最小二乘问题式 (7.8.35) 的求解步骤如下:

- (1) 求解二次特征值问题式 (7.8.39), 得到特征值  $\lambda_i$  (特征向量  $\mathbf{y}_i$  可以不需要)。
- (2) 确定最小的特征值  $\lambda_{\min}$ 。
- (3) 约束最小二乘问题式 (7.8.35) 的解由  $\mathbf{x} = (\mathbf{A} - \lambda_{\min} \mathbf{I})^{-1} \mathbf{b}$  给出。

### 3. 多输入-多输出系统

多输入-多输出 (multiple input-multiple output, MIMO) 系统是通信、雷达、信号处理、自动控制和系统工程中经常遇到的线性系统。考虑  $m$  个输入和  $n$  个输出的线性受控系统

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{B}\mathbf{u}(t) \quad (7.8.40)$$

$$\mathbf{y}(t) = \mathbf{L}\mathbf{q}(t) \quad (7.8.41)$$

式中,  $\mathbf{u}(t) \in \mathbb{C}^m$ ,  $m \leq r$  为某个输入信号向量;  $\mathbf{q}(t) \in \mathbb{C}^r$  为系统的状态向量;  $\mathbf{y}(t) \in \mathbb{C}^n$  为系统的输出向量;  $\mathbf{B} \in \mathbb{C}^{r \times m}$  为系统的输入作用矩阵;  $\mathbf{L} \in \mathbb{C}^{n \times r}$  为系统的输出作用矩阵;  $\mathbf{M}, \mathbf{C}, \mathbf{K}$  为  $r \times r$  矩阵。

取 MIMO 系统的 Laplace 变换, 并假定零初始条件, 得

$$s^2 \mathbf{M} \bar{\mathbf{q}}(s) + s \mathbf{C} \bar{\mathbf{q}}(s) + \mathbf{K} \bar{\mathbf{q}}(s) = \mathbf{B} \bar{\mathbf{u}}(s) \quad (7.8.42)$$

$$\bar{\mathbf{y}}(s) = \mathbf{L} \bar{\mathbf{q}}(s) \quad (7.8.43)$$

于是, 系统的传递函数矩阵 (transfer function matrix)

$$\mathbf{G}(s) = \frac{\bar{\mathbf{y}}(s)}{\bar{\mathbf{u}}(s)} = \mathbf{L}(s^2 \mathbf{M} + s \mathbf{C} + \mathbf{K})^{-1} \mathbf{B} \quad (7.8.44)$$

令  $\mathbf{Q}(s) = s^2 \mathbf{M} + s \mathbf{C} + \mathbf{K}$ , 则由式 (7.8.12) 和式 (7.8.13), 得第 1 友型  $L1$  和第 2 友型  $L2$  情况下的逆矩阵

$$L1: \quad (s^2 \mathbf{M} + s \mathbf{C} + \mathbf{K})^{-1} = \mathbf{U}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{A} \mathbf{V}^H = \sum_{i=1}^{2n} \frac{\lambda_i \mathbf{u}_i \mathbf{v}_i^H}{s - \lambda_i} \quad (7.8.45)$$

$$L2: \quad (s^2 \mathbf{M} + s \mathbf{C} + \mathbf{K})^{-1} = \mathbf{U}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{V}^H = \sum_{i=1}^{2n} \frac{\mathbf{u}_i \mathbf{v}_i^H}{s - \lambda_i} \quad (7.8.46)$$

即是说, 二次特征多项式  $s^2 \mathbf{M} + s \mathbf{C} + \mathbf{K}$  的特征值给出受控 MIMO 系统传递函数的极点。因此, 二次特征值为研究受控 MIMO 系统的控制性能和响应性能提供了依据。

文献 [474] 介绍了二次特征值问题的诸多应用, 是一篇关于二次特征值问题的精彩综述。

## 7.9 联合对角化

特征值分解是一个 Hermitian 矩阵的对角化，广义特征值分解可视为两个矩阵的联合对角化。一个自然会问的问题是：能否对多个矩阵同时对角化或联合对角化？这正是本节的主题。

### 7.9.1 联合对角化问题

考虑阵列接收信号模型

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{v}(t), \quad t = 1, 2, \dots \quad (7.9.1)$$

其中， $\mathbf{x}(t) = [x_1(t), \dots, x_m(t)]^T$  为观测信号向量， $m$  是观测信号的传感器数目； $\mathbf{s}(t) = [s_1(t), \dots, s_n(t)]^T$  为源信号向量，并且  $m \geq n$ ； $\mathbf{v}(t) = [v_1(t), \dots, v_m(t)]^T$  为传感器阵列上的加性噪声向量；而  $\mathbf{A} \in \mathbb{C}^{m \times n}$  表示信号源混合状况的矩阵，称为混合矩阵。

盲信号分离问题的提法是：利用观测信号向量  $\mathbf{x}(t)$  辨识未知的混合矩阵  $\mathbf{A}$ ，然后利用分离矩阵  $\mathbf{W} = \mathbf{A}^\dagger \in \mathbb{C}^{n \times m}$ ，恢复源信号向量  $\mathbf{s}(t) = \mathbf{W}\mathbf{x}(t)$ ，达到信号分离之目的。因此，盲信号分离的关键是混合矩阵  $\mathbf{A}$  的辨识。为此，需要对盲信号分离模型作以下假设：

(1) 加性噪声是时域白色，空域有色的，即其自相关矩阵

$$\mathbf{R}_v(k) = \mathbb{E}\{\mathbf{v}(t)\mathbf{v}^H(t-k)\} = \delta(k)\mathbf{R}_v = \begin{cases} \mathbf{R}_v, & k=0 \text{ (空域有色)} \\ \mathbf{O}, & k \neq 0 \text{ (时域白色)} \end{cases}$$

其中，时域白色是指每个传感器上的加性噪声为白噪声，而空域有色系指不同传感器的加性白噪声可能相关。

(2)  $n$  个源信号统计独立，即有  $\mathbb{E}\{\mathbf{s}(t)\mathbf{s}^H(t-k)\} = \mathbf{D}_k$  (对角矩阵)。

(3) 源信号与加性噪声统计独立，即有  $\mathbb{E}\{\mathbf{s}(t)\mathbf{v}^H(t-k)\} = \mathbf{O}$  (零矩阵)。

在上述假设条件下，阵列输出向量的协方差矩阵为

$$\begin{aligned} \mathbf{C}_x(k) &= \mathbb{E}\{\mathbf{x}(t)\mathbf{x}^H(t-k)\} \\ &= \mathbb{E}\{[\mathbf{A}\mathbf{s}(t) + \mathbf{v}(t)][\mathbf{A}\mathbf{s}(t-k) + \mathbf{v}(t-k)]^H\} \\ &= \mathbf{A}\mathbb{E}\{\mathbf{s}(t)\mathbf{s}^H(t-k)\}\mathbf{A}^H + \mathbb{E}\{\mathbf{v}(t)\mathbf{v}^H(t-k)\} \\ &= \begin{cases} \mathbf{A}\mathbf{D}_0\mathbf{A}^H + \mathbf{R}_v, & k=0 \\ \mathbf{A}\mathbf{D}_k\mathbf{A}^H, & k \neq 0 \end{cases} \end{aligned} \quad (7.9.2)$$

这一结果表明，若采用无噪声影响 (滞后  $k \neq 0$ ) 的  $K$  个协方差矩阵  $\mathbf{C}_x(k)$ ， $k = 1, \dots, K$ ，并且对这  $K$  个矩阵进行联合对角化

$$\mathbf{C}_x(k) = \mathbf{U}\boldsymbol{\Sigma}_k\mathbf{U}^H, \quad k = 1, \dots, K \quad (7.9.3)$$

就有可能辨识出混合矩阵  $\mathbf{A}$ 。

应当注意的是,  $\mathbf{U}$  不一定就是  $\mathbf{A}$ , 对角矩阵  $\boldsymbol{\Sigma}_k$  也不一定就是原对角矩阵  $\mathbf{D}_k$ , 因为对于任何一个广义置换矩阵  $\mathbf{G}$  而言,  $\mathbf{U} = \mathbf{AG}$  都能够满足式(7.9.2)。

**定义 7.9.1** (本质相等矩阵) 两个  $n \times m$  矩阵  $\mathbf{A}$  和  $\mathbf{U}$  称为本质相等矩阵, 记作  $\mathbf{A} \doteq \mathbf{U}$ , 若  $\mathbf{U} = \mathbf{AG}$ , 其中  $\mathbf{G}$  为  $m \times m$  广义置换矩阵。

由式(7.9.3)不能精确辨识混合矩阵  $\mathbf{A}$ , 只能辨识与之本质相等的矩阵  $\mathbf{AG}$ 。这一数学结果也可从盲信号分离的数学模型直接得到解释: 由

$$\mathbf{x}(t) = \mathbf{As}(t) = \sum_{i=1}^m \frac{\mathbf{a}_i^T}{\alpha} \alpha s_i(t) \quad (7.9.4)$$

知, 交换源信号的排列顺序和幅值的固定因子变化, 只要混合矩阵的列向量作相应的排列和相反的幅度变化, 得到的混合信号则完全相同。换言之, 只根据观测信号  $\mathbf{x}(t)$  或其自相关矩阵, 只能辨识与  $\mathbf{A}$  本质相等的矩阵  $\mathbf{AG}$ 。这一现象称为盲信号分离的模糊性或不确定性。不过, 从信号分离的角度, 分离信号的排序和幅度的不确定性是完全允许的。

多个矩阵的联合对角化最早是 Flury 于 1984 年考虑  $K$  个协方差矩阵的共同主分量分析时提出的。后来, Cardoso 与 Souloumiac<sup>[90]</sup> 于 1996 年, Belouchrani 等人<sup>[37]</sup> 于 1997 年从盲信号分离的角度分别提出了多个累积量矩阵和协方差矩阵的近似联合对角化。从此, 联合对角化在盲信号分离领域获得了广泛的研究与应用。

联合对角化的数学问题是: 给定  $K$  个  $m \times m$  对称矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_K$ , 寻求一  $m \times n$  满列秩矩阵  $\mathbf{U}$ , 使得这  $K$  个矩阵同时对角化(联合对角化)

$$\mathbf{A}_k = \mathbf{U} \mathbf{A}_k \mathbf{U}^H, \quad k = 1, \dots, K \quad (7.9.5)$$

其中  $\mathbf{U} \in \mathbb{C}^{m \times n}$  称为联合对角化器(joint diagonalizer), 对角矩阵  $\mathbf{A}_k \in \mathbb{R}^{n \times n}, k = 1, \dots, K$ 。

值得指出的是, 两个  $n \times n$  Hermitian 矩阵  $\mathbf{A}$  和  $\mathbf{B}$  的(精确)联合对角化与 Hermitian 矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值分解等价。

联合对角化  $\mathbf{A}_k = \mathbf{U} \mathbf{A}_k \mathbf{U}^H$  为精确联合对角化。然而, 实际的联合对角化为近似联合对角化: 给定矩阵集合  $\mathcal{A} = \{\mathbf{A}_1, \dots, \mathbf{A}_K\}$ , 希望求一个联合对角化器  $\mathbf{U} \in \mathbb{C}^{m \times n}$  和  $K$  个对应的  $n \times n$  对角矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_K$ , 使目标函数最小化<sup>[89, 90]</sup>

$$\min J_1(\mathbf{U}, \mathbf{A}_1, \dots, \mathbf{A}_K) = \min \sum_{k=1}^K w_k \left\| \mathbf{U}^H \mathbf{A}_k \mathbf{U} - \mathbf{A}_k \right\|_F^2 \quad (7.9.6)$$

或者<sup>[510, 527]</sup>

$$\min J_2(\mathbf{U}, \mathbf{A}_1, \dots, \mathbf{A}_K) = \min \sum_{k=1}^K w_k \left\| \mathbf{A}_k - \mathbf{U} \mathbf{A}_k \mathbf{U}^H \right\|_F^2 \quad (7.9.7)$$

式中,  $w_1, \dots, w_K$  为正的权系数。为简化叙述, 下面假定  $w_1 = \dots = w_K = 1$ 。

### 7.9.2 正交近似联合对角化

在很多工程应用中, 只使用联合对角化矩阵  $\mathbf{U}$ , 无须使用对角矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_K$ 。因此, 如何将近似联合对角化问题的目标函数转换成只包含联合对角化矩阵  $\mathbf{U}$  的函数, 便是一个有着实际意义的问题。这种单一优化问题有以下几种求解方法。

#### 1. 非对角函数最小化方法

在数值分析中, 一个  $m \times m$  正方矩阵  $\mathbf{B} = [B_{ij}]$  所有非主对角线元素的绝对值的平方和定义为该矩阵的非对角 (off) 函数, 即有

$$\text{off}(\mathbf{B}) \stackrel{\text{def}}{=} \sum_{i=1, i \neq j}^m \sum_{j=1}^n |B_{ij}|^2 \quad (7.9.8)$$

如果将抽取正方矩阵  $\mathbf{M}$  所有非主对角线元素组成的矩阵

$$[\mathbf{M}_{\text{off}}]_{ij} = \begin{cases} 0, & i = j \\ M_{ij}, & i \neq j \end{cases} \quad (7.9.9)$$

称为 off 矩阵, 则 off 函数就是 off 矩阵的 Frobenius 范数的平方

$$\text{off}(\mathbf{M}) = \|\mathbf{M}_{\text{off}}\|_F^2 \quad (7.9.10)$$

利用 off 函数, 可以将正交近似联合对角化问题表示为<sup>[89, 90]</sup>

$$\min J_{1a}(\mathbf{U}) = \sum_{k=1}^K \text{off}(\mathbf{U}^H \mathbf{A}_k \mathbf{U}) = \sum_{k=1}^K \sum_{i=1, i \neq j}^n \sum_{j=1}^n |(\mathbf{U}^H \mathbf{A}_k \mathbf{U})_{ij}|^2 \quad (7.9.11)$$

对矩阵  $\mathbf{A}_1, \dots, \mathbf{A}_K$  的非对角元素实施一系列的 Given 旋转, 即可实现这些矩阵的正交联合对角化。所有 Givens 旋转矩阵的乘积即给出联合对角化器  $\mathbf{U}$ 。这就是 Cardoso 等人提出的近似联合对角化的 Jacobi 算法<sup>[89, 90]</sup>。

#### 2. 对角函数最大化方法

一个正方矩阵的对角函数可以是标量函数、向量函数或矩阵函数。

(1) 对角函数  $\text{diag}(\mathbf{B}) \in \mathbb{R}$  是  $m \times m$  正方矩阵  $\mathbf{B}$  的对角函数, 定义为

$$\text{diag}(\mathbf{B}) \stackrel{\text{def}}{=} \sum_{i=1}^m |B_{ii}|^2 \quad (7.9.12)$$

(2) 对角向量函数  $m \times m$  正方矩阵  $\mathbf{B}$  的对角向量化函数记作  $\text{diag}(\mathbf{B}) \in \mathbb{C}^m$ , 是一个将矩阵  $\mathbf{B}$  的对角元素排列的列向量, 即有

$$\text{diag}(\mathbf{B}) \stackrel{\text{def}}{=} [B_{11}, \dots, B_{mm}]^T \quad (7.9.13)$$

(3) 对角矩阵函数  $m \times m$  正方矩阵  $\mathbf{B}$  的对角矩阵函数记作  $\text{Diag}(\mathbf{B}) \in \mathbb{C}^{m \times m}$ , 是一个抽取矩阵  $\mathbf{B}$  的对角元素组成的对角矩阵, 即有

$$\text{Diag}(\mathbf{B}) \stackrel{\text{def}}{=} \begin{bmatrix} B_{11} & & & 0 \\ & \ddots & & \\ 0 & & B_{mm} & \end{bmatrix} \quad (7.9.14)$$

使  $\text{off}(\mathbf{B})$  最小化, 又可等价为对角函数  $\text{diag}(\mathbf{B})$  的最大化, 即有

$$\min \text{off}(\mathbf{B}) = \max \text{diag}(\mathbf{B}) \quad (7.9.15)$$

所以式 (7.9.11) 又可改写为<sup>[510]</sup>

$$\max J_{1b}(\mathbf{U}) = \sum_{k=1}^K \text{diag}(\mathbf{U}^H \mathbf{A}_k \mathbf{U}) = \sum_{k=1}^K \sum_{i=1}^n |(\mathbf{U}^H \mathbf{A}_k \mathbf{U})_{ii}|^2 \quad (7.9.16)$$

事实上, 对角函数  $\text{diag}(\mathbf{B})$  实际上就是正方矩阵  $\mathbf{B}$  与自己的内积, 也就是矩阵乘积  $\mathbf{B}^H \mathbf{B}$  的迹函数

$$\text{diag}(\mathbf{B}) = \langle \mathbf{B}, \mathbf{B} \rangle = \text{tr}(\mathbf{B}^H \mathbf{B}) \quad (7.9.17)$$

于是, 优化问题式 (7.9.16) 可以用迹函数写作

$$\max J_{1b}(\mathbf{U}) = \sum_{k=1}^K \text{tr}(\mathbf{U}^H \mathbf{A}_k^H \mathbf{U} \mathbf{U}^H \mathbf{A}_k \mathbf{U}) \quad (7.9.18)$$

特别地, 在正交近似联合对角化的情况下, 由于  $\mathbf{U} \mathbf{U}^H = \mathbf{U}^H \mathbf{U} = \mathbf{I}$  和  $\mathbf{A}_k^H = \mathbf{A}_k, k = 1, \dots, K$ , 故上式可简化为

$$\max J_{1b}(\mathbf{U}) = \sum_{k=1}^K \text{tr}(\mathbf{U}^H \mathbf{A}_k^2 \mathbf{U}) \quad (7.9.19)$$

### 3. 子空间方法

在正交近似联合对角化的情况下, 联合对角化器  $\mathbf{U}$  为酉矩阵, 其列向量具有单位范数, 即  $\|\mathbf{u}_k\|_F = 1$ 。

定义  $\mathbf{m}_k = \text{diag}(\mathbf{A}_k)$  是由对角矩阵  $\mathbf{A}$  的对角元素组成的向量, 并令

$$\hat{\mathbf{A}} = [\text{vec}(\mathbf{A}_1), \dots, \text{vec}(\mathbf{A}_K)], \quad \mathbf{M} = [\mathbf{m}_1, \dots, \mathbf{m}_K]$$

则联合对角化问题的代价函数可以等价写为

$$\begin{aligned} \sum_{k=1}^K \left\| \mathbf{A}_k - \mathbf{U} \mathbf{A}_k \mathbf{U}^H \right\|_F^2 &= \sum_{k=1}^K \left\| \text{vec}(\mathbf{A}_k) - (\mathbf{U}^* \odot \mathbf{U}) \text{diag}(\mathbf{A}_k) \right\|_F^2 \\ &= \left\| \hat{\mathbf{A}} - (\mathbf{U}^* \odot \mathbf{U}) \mathbf{M} \right\|_F^2 \end{aligned} \quad (7.9.20)$$

式中,  $\mathbf{C} \odot \mathbf{D}$  是矩阵的 Khatri-Rao 积, 它是矩阵列分量的 Kronecker 积

$$\mathbf{C} * \mathbf{D} = [c_1 \otimes d_1, c_2 \otimes d_2, \dots, c_n \otimes d_n] \quad (7.9.21)$$

于是, 联合对角化问题的解式 (7.9.20) 可以换写为

$$\{\mathbf{U}, \mathbf{M}\} = \arg \min_{\mathbf{U}, \mathbf{M}} \left\| \hat{\mathbf{A}} - \mathbf{B} \mathbf{M} \right\|_F^2, \quad \mathbf{B} = \mathbf{U}^* \odot \mathbf{U} \quad (7.9.22)$$

这一最优化问题是可分离的，因为  $\mathbf{M}$  的最小二乘解为

$$\mathbf{M} = (\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H \hat{\mathbf{A}} \quad (7.9.23)$$

将式 (7.9.23) 代入式 (7.9.22) 后，可以消去  $\mathbf{M}$ ，得到

$$\mathbf{U} = \arg \min_{\mathbf{U}} \left\| \hat{\mathbf{A}} - \mathbf{B}(\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H \hat{\mathbf{A}} \right\|_F^2 = \arg \min_{\mathbf{U}} \left\| \mathbf{P}_B^\perp \hat{\mathbf{A}} \right\|_F^2 \quad (7.9.24)$$

式中， $\mathbf{P}_B^\perp = \mathbf{I} - \mathbf{B}(\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H$  为正交投影矩阵。求解最优化问题式 (7.9.24) 的具体算法可参考文献 [490]。

上面介绍的三种方法都属于正交近似联合对角化算法，因为它们给出的联合对角化器为酉矩阵。

### 7.9.3 非正交近似联合对角化

正交联合对角化的优点是不会出现无平凡解 ( $\mathbf{U} = \mathbf{0}$ ) 和退化解 ( $\mathbf{U}$  奇异)；缺点是正交联合对角化必须先对观测数据向量预白化。

预白化有两个主要缺点：

- (1) 预白化严重影响分离信号的性能，因为预白化的误差在后面的信号分离中得不到纠正。
- (2) 白化会破坏加权最小二乘准则，造成某个矩阵被精确对角化，而其他矩阵的对角化可能很差。

非正交联合对角化就是没有约束  $\mathbf{U}^H \mathbf{U} = \mathbf{I}$  的联合对角化，已经成为主流的联合对角化方法。非正交联合对角化的优点是没有白化的两个缺点，而缺点则是可能存在平凡解和退化解。

非正交联合对角化的典型算法有：Pham<sup>[404]</sup> 的基于信息论准则最小化的迭代算法，van der Veen<sup>[490]</sup> 的 Newton 型迭代的子空间拟合算法，Yeredor<sup>[527]</sup> 的 AC-DC 算法等。

AC-DC 算法将耦合的优化问题

$$\begin{aligned} J_{WLS2}(\mathbf{U}, \Lambda_1, \dots, \Lambda_K) &= \sum_{k=1}^K w_k \|\mathbf{A}_k - \mathbf{U} \Lambda_k \mathbf{U}^H\|_F^2 \\ &= \sum_{k=1}^K w_k \left\| \mathbf{A}_k - \sum_{n=1}^N \lambda_n^{[k]} \mathbf{u}_k \mathbf{u}_k^H \right\|_F^2 \end{aligned}$$

分离成两个单独的优化问题。这一算法由两个阶段组成：

- (1) 交替列 (AC, alternating columns) 阶段 固定  $\mathbf{U}$  的其他列和矩阵  $\Lambda_1, \dots, \Lambda_K$ ，使目标函数  $J_{WLS2}(\mathbf{U})$  相对于  $\mathbf{U}$  的某个列向量最小化。
- (2) 对角中心 (DC, diagonal centers) 阶段 固定  $\mathbf{U}$ ，使  $J_{WLS2}(\mathbf{U}, \Lambda_1, \dots, \Lambda_K)$  相对于所有  $\Lambda_1, \dots, \Lambda_K$  最小化。

避免平凡解的简单方法是加约束条件  $\text{Diag}(\mathbf{U}) = \mathbf{I}$ 。然而，非正交联合对角化的主要缺点是联合对角化器  $\mathbf{U}$  有可能奇异或者条件数很大。奇异或条件数很大的解称为退化解。非正交联合对角化问题的退化解是文献 [316] 提出并解决的。

为了同时避免非正交联合对角化问题的平凡解和退化解，Li 和 Zhang 提出使用下面的目标函数<sup>[316]</sup>

$$\min f(\mathbf{U}) = \sum_{k=1}^K \alpha_k \sum_{i=1}^N \sum_{j=1, j \neq i}^N |[\mathbf{U}^H \mathbf{A}_k \mathbf{U}]_{ij}|^2 - \beta \ln |\det(\mathbf{U})| \quad (7.9.25)$$

其中  $\alpha_k (1 \leq k \leq K)$  为正的权重系数， $\beta$  为一正数， $\ln$  表示自然对数。

上述代价函数可分为平方对角化误差函数

$$f_1(\mathbf{U}) = \sum_{k=1}^K \alpha_k \sum_{i=1}^N \sum_{j=1, j \neq i}^N |[\mathbf{U}^H \mathbf{A}_k \mathbf{U}]_{ij}|^2 \quad (7.9.26)$$

与负对数行列式项

$$f_2(\mathbf{U}) = -\ln |\det(\mathbf{U})| \quad (7.9.27)$$

之和。

代价函数 (7.9.25) 的一个明显优点是：当  $\mathbf{U} = \mathbf{O}$  或者奇异时， $f_2(\mathbf{U}) \rightarrow +\infty$ 。因此，代价函数  $f(\mathbf{U})$  的最小化可以同时避免平凡解和退化解。

此外，文献 [316] 还证明了以下两个重要结果：

(1) 当且仅当非奇异矩阵  $\mathbf{U}$  使得所有矩阵  $\mathbf{A}_k, k = 1, \dots, K$  精确联合对角化时， $f_1(\mathbf{U})$  是下无界的。换言之，在近似联合对角化时， $f(\mathbf{U})$  是下有界的。

(2) 代价函数  $f(\mathbf{U})$  的最小化与惩罚参数  $\beta$  的数值无关，这意味着， $\beta$  可以选择有限大的任意值，通常可直接选  $\beta = 1$ ，从而避免了罚函数法性能取决于惩罚参数的选择。

联合对角化已广泛应用于盲信号分离<sup>[12, 37, 352, 527]</sup>、盲波束形成<sup>[89]</sup>、时延估计<sup>[528]</sup>、频率估计<sup>[343]</sup>、阵列信号处理<sup>[511]</sup>、多输入-多输出 (MIMO) 盲均衡<sup>[125]</sup> 以及盲 MIMO 系统辨识<sup>[102]</sup> 等问题中。

## 7.10 Fourier 分析与特征分析

实际应用的信号或函数要么是周期函数，要么是非周期函数。周期函数和非周期函数的正交展开是函数分析的简单而有效的方法。本节分别讨论周期函数的 Fourier 分析和非周期函数的特征分析。

### 7.10.1 周期函数的 Fourier 分析

Fourier 分析是一种非常有用的数学工具，广泛应用于数学、物理、信息科学和诸多工程学科。Fourier 分析由 Fourier 级数和 Fourier 积分变换两部分组成。

考虑使用复指数函数或复谐波信号  $e^{j\omega n}$  作为线性时不变系统  $\mathcal{L}$  的输入。令线性系统的传递函数为  $H(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h(k)e^{-j\omega k}$ , 其中  $h(k)$  称为系统的冲激响应系数。由于系统的输出是系统输入与系统冲激响应的卷积和, 故有

$$\mathcal{L}[e^{j\omega n}] = \sum_{k=-\infty}^{\infty} h(n-k)e^{j\omega k} = \sum_{k=-\infty}^{\infty} h(k)e^{j\omega(n-k)} = H(e^{j\omega})e^{j\omega n} \quad (7.10.1)$$

令  $n = 0, 1, \dots, N-1$ , 则有

$$\mathcal{L} \begin{bmatrix} 1 \\ e^{j\omega} \\ \vdots \\ e^{j\omega(N-1)} \end{bmatrix} = H(e^{j\omega}) \begin{bmatrix} 1 \\ e^{j\omega} \\ \vdots \\ e^{j\omega(N-1)} \end{bmatrix}$$

或简写为

$$\mathcal{L}[\mathbf{w}(\omega)] = H(e^{j\omega})\mathbf{w}(\omega) \quad (7.10.2)$$

式中

$$\mathbf{w}(\omega) = [1, e^{j\omega}, \dots, e^{j\omega(N-1)}]^T \quad (7.10.3)$$

式 (7.10.2) 表明, 向量  $\mathbf{w}(\omega) = [1, e^{j\omega}, \dots, e^{j\omega(N-1)}]^T$  是线性时不变系统的特征向量, 而系统传递函数  $H(e^{j\omega})$  是与  $\mathbf{w}(\omega)$  相对应的特征值。由于对于每个频率  $\omega$ , 式 (7.10.2) 都成立, 所以线性时不变系统有无穷多个特征向量  $\mathbf{w}(\omega)$ ,  $\omega = -\infty, \dots, \infty$ , 相对应的特征值也有无穷多个, 它们是  $H(e^{j\omega})$ ,  $\omega = -\infty, \dots, \infty$ 。

复指数函数  $e^{j\omega n}$  常称为线性系统的本征函数, 它也是周期函数的 Fourier 级数展开的基函数。

一个周期为  $T$  的平稳的周期随机过程  $x(t)$  可以用指数函数展开为 Fourier 级数

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\omega_0 t} \quad (7.10.4)$$

式中,  $\omega_0 = 2\pi/T$  为角频率,  $c_k$  称为展开系数或 Fourier 系数, 由 Fourier 变换

$$c_k = \frac{1}{T} \int_0^T x(t) e^{-jk\omega_0 t} dt \quad (7.10.5)$$

确定。

对随机过程的周期性要求可以保证展开系数  $c_k$  和  $c_n$  在  $k \neq n$  的情况下相互正交。其证明如下。

由式 (7.10.5) 易求得

$$E\{c_k c_n^*\} = \frac{1}{T^2} E \left\{ \int_0^T \int_0^T x(t) x^*(u) e^{jk\omega_0 t} e^{-jn\omega_0 u} dt du \right\} \quad (7.10.6)$$

$$= \frac{1}{T^2} \int_0^T \int_0^T R(t-u) e^{j\omega_0(kt-nu)} dt du \quad (7.10.7)$$

式中,  $R(t-u) = E\{x(t)x^*(u)\}$  是周期随机过程  $x(t)$  的相关函数, 它也是周期函数。令滞后 (lag)  $\tau = t - u$ , 则

$$R(\tau) = \sum_{m=-\infty}^{\infty} b_m e^{j m \omega_0 \tau} \quad (7.10.8)$$

将上式代入式 (7.10.7), 得

$$\begin{aligned} E\{c_k c_n^*\} &= \frac{1}{T^2} \int_0^T \int_0^T \sum_{m=-\infty}^{\infty} b_m e^{j m \omega_0 (t-u)} e^{j \omega_0 (nu - kt)} dt du \\ &= \sum_{m=-\infty}^{\infty} b_m \frac{1}{T} \int_0^T e^{j \omega_0 (m-k)t} dt \frac{1}{T} \int_0^T e^{j \omega_0 (m-n)u} du \\ &= \begin{cases} 1, & k = n \\ 0, & k \neq n \end{cases} \end{aligned}$$

即当  $k \neq n$  时, 展开系数  $c_k$  和  $c_n$  正交。反之, 也可以证明, 欲使展开系数正交, 则平稳的随机过程必须是周期函数。

### 7.10.2 非周期函数的特征分析

一个平稳的非周期随机过程或函数不可能用复指数函数作基函数展开为 Fourier 级数形式, 但可以用正交函数  $\phi_k(t)$  为基函数展开为级数形式。这种方法称为 Karhunen-Loeve 展开, 简称 KL 展开。

假定一个平稳的非周期随机过程  $x(t)$  的时间定义域为区间  $[a, b]$ , 则  $x(t)$  可以用级数形式展开为

$$x(t) = \sum_{k=1}^{\infty} \alpha_k c_k \phi_k(t) \quad (7.10.9)$$

式中,  $\alpha_k$  是实或复常数, 并且

$$\int_a^b \phi_k(t) \phi_n^*(t) dt = \begin{cases} 1, & k = n \\ 0, & k \neq n \end{cases} \quad (7.10.10)$$

$$E\{c_k c_n^*\} = \begin{cases} 1, & k = n \\ 0, & k \neq n \end{cases} \quad (7.10.11)$$

用  $\phi_n^*(t)$  乘式 (7.10.9) 的两边, 再对时间  $t$  在区间  $[a, b]$  内积分, 并且使用式 (7.10.10), 易知

$$c_k = \frac{1}{\alpha_k} \int_a^b x(t) \phi_k^*(t) dt \quad (7.10.12)$$

式 (7.10.9) 称为随机过程  $x(t)$  的 KL 展开, 是一种正交展开; 而积分变换式 (7.10.12) 称为  $x(t)$  的 KL 变换。因此, 使用 KL 展开和 KL 变换对随机过程进行分析时, 需要确定复常数  $\alpha_k$  和正交基函数  $\phi_k(t)$ 。

计算随机过程的自相关函数，并利用展开系数的正交式(7.10.11)，立即有

$$\begin{aligned} R(t, u) &= E\{x(t)x^*(u)\} \\ &= E\left\{\sum_{k=1}^{\infty} \alpha_k c_k \phi_k(t) \sum_{n=1}^{\infty} \alpha_n^* c_n^* \phi_n^*(u)\right\} \\ &= \sum_{k=1}^{\infty} |\alpha_k|^2 \phi_k(t) \phi_k^*(u) \end{aligned} \quad (7.10.13)$$

由上式及式(7.10.10)得

$$\int_a^b R(t, u) \phi_i(u) du = \sum_{k=1}^{\infty} |\alpha_k|^2 \phi_k(t) \int_a^b \phi_i(u) \phi_k^*(u) du = |\alpha_i|^2 \phi_i(t) \quad (7.10.14)$$

这表明，复常数的模平方  $|\alpha_i|^2$  是积分方程的特征值  $\lambda$ ，基函数  $\phi_i(t)$  是与该特征值对应的特征函数  $\phi(t)$ ，即

$$\int_a^b R(t, u) \phi(u) du = \lambda \phi(t) \quad (7.10.15)$$

以上讨论的是连续时间的单个平稳非周期随机过程的情况。下面考查  $m$  个离散时间的平稳非周期随机过程  $x_i(n)$  ( $n = 1, \dots, N; i = 1, \dots, m$ ) 的级数展开表示。为此，我们先来讨论任意一个向量的坐标表示。

**定理 7.10.1** 令  $V$  是一向量空间，并且  $B = \{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  是  $V$  空间的一组基向量。对于  $V$  空间内的每一个向量  $\mathbf{w}$ ，存在唯一的一组标量，使得  $\mathbf{w}$  可以表示成

$$\mathbf{w} = w_1 \mathbf{u}_1 + \dots + w_p \mathbf{u}_p \quad (7.10.16)$$

**证明** 假定  $\mathbf{w} \in V$  有两种不同的表示方式

$$\mathbf{w} = w_1 \mathbf{u}_1 + \dots + w_p \mathbf{u}_p$$

$$\mathbf{w} = \alpha_1 \mathbf{u}_1 + \dots + \alpha_p \mathbf{u}_p$$

两式相减，得

$$\mathbf{0} = (w_1 - \alpha_1) \mathbf{u}_1 + \dots + (w_p - \alpha_p) \mathbf{u}_p$$

由于  $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  为基向量，它们线性无关，因此  $w_1 - \alpha_1 = 0, \dots, w_p - \alpha_p = 0$ ，即  $\alpha_1 = w_1, \dots, \alpha_p = w_p$ 。即是说，当利用基  $B$  表示向量  $\mathbf{w}$  时，不可能有两种不同的表示方式。 ■

定理 7.10.1 表明，向量空间  $V$  内任一向量  $\mathbf{w}$  的唯一表示决定于基  $B = \{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  的选择和标量  $w_1, \dots, w_p$  的确定。基向量  $\mathbf{u}_1, \dots, \mathbf{u}_p$  组成了向量表示的坐标系(coordinate system)，而标量  $w_1, \dots, w_p$  称为向量  $\mathbf{w}$  相对于基  $B$  的坐标(coordinates)，这些坐标组成的向量

$$[\mathbf{w}]_B = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix} \quad (7.10.17)$$

称为向量  $w$  相对于基  $B$  的坐标向量 (coordinate vector)。

很自然地, 与坐标轴通常应该相互垂直类似, 坐标系  $u_1, \dots, u_p$  最好相互正交。标准正交基  $\{u_1, \dots, u_p\}$  的使用给坐标向量的确定带来很大的方便: 用  $u_i^H$  左乘式 (7.10.16), 并注意到正交性  $u_i^H u_j = \delta(i - j)$ , 立即有

$$w_i = u_i^H w = \langle u_i, w \rangle \quad (7.10.18)$$

换句话说, 若使用标准正交基  $\{u_1, u_2, \dots, u_p\}$  作为向量  $w$  表示的坐标系, 则  $w$  的坐标可以利用式 (7.10.18) 直接确定。

定理 7.10.1 构成了著名的 KL 展开的理论基础: 为了寻找  $m$  个离散时间的平稳非周期随机过程  $x_i(n) (n = 1, \dots, N; i = 1, \dots, m)$  的级数展开, 关键是如何选择一组适合于  $x_i(n) (n = 1, \dots, N; i = 1, \dots, m)$  的标准正交基作为坐标系。

令  $x_i = [x_i(1), \dots, x_i(N)]^T$  表示第  $i$  个随机过程的观测数据向量,  $\phi_k = [\phi_k(1), \dots, \phi_k(N)]^T$  表示第  $k$  个正交基向量, 即

$$E\{\phi_k^H \phi_j\} = \begin{cases} 1, & k = j \\ 0, & k \neq j \end{cases} \quad (7.10.19)$$

于是, KL 展开式 (7.10.9) 变为

$$x_i = \sum_{k=1}^N c_{ik} \phi_k \quad (7.10.20)$$

若令  $c_i = [c_{i1}, \dots, c_{iN}]^T$ , 则式 (7.10.20) 可以写成更加紧凑的形式, 即

$$x_i = \Phi c_i \quad (7.10.21)$$

式中,  $\Phi = [\phi_1, \dots, \phi_N]$ 。

将式 (7.10.21) 代入  $M$  个随机信号向量  $x_1, \dots, x_M$  的相关矩阵, 得

$$R = \sum_{i=1}^M E\{x_i x_i^H\} = \Phi \left( \sum_{i=1}^M E\{c_i c_i^H\} \right) \Phi^H \quad (7.10.22)$$

但由于要求展开系数必须相互正交, 故

$$\sum_{i=1}^M E\{c_i c_i^H\} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_N \end{bmatrix} = D \quad (7.10.23)$$

于是, 式 (7.10.22) 可写为

$$R = \Phi D \Phi^H \quad \text{或} \quad R \Phi = \Phi D$$

即有

$$R \phi_i = \lambda_i \phi_i, \quad i = 1, \dots, N \quad (7.10.24)$$

这说明, 基向量  $\phi_i$  是  $N \times N$  相关矩阵  $R$  的特征向量, 与之对应的特征值为  $\lambda_i$ 。

式 (7.10.21) 两边左乘矩阵  $\Phi^H$ , 并注意到正交基向量  $\phi_i$  满足  $\Phi^H \Phi = I$ , 即得

$$\mathbf{c}_i = \Phi^H \mathbf{x}_i \quad (7.10.25)$$

综合定理 7.10.1 和以上讨论, 利用相关矩阵的特征值和特征向量对  $M$  个随机过程得观测向量  $\mathbf{x}_1, \dots, \mathbf{x}_M$  进行分析的方法可以总结如下:

- (1) 利用式 (7.10.21) 计算  $N \times N$  相关矩阵  $R$ 。
- (2) 对相关矩阵  $R$  进行特征值分解, 得到特征值  $\lambda_i$  及对应的特征向量  $\mathbf{u}_i$ ,  $i = 1, \dots, N$ 。取  $P$  个大特征值及其对应的特征向量, 组成  $P \times N$  矩阵  $U = [\mathbf{u}_1, \dots, \mathbf{u}_P]$ 。
- (3) 利用这  $P$  个特征向量做标准正交基, 对第  $i$  个随机向量的级数展开式为

$$\mathbf{x}_i = \sum_{k=1}^P c_{ik} \mathbf{u}_k, \quad i = 1, \dots, M \quad (7.10.26)$$

称为随机向量的线性特征向量展开 (linear eigenvector expansion)。展开式中的展开系数  $c_{ik}$  由

$$\mathbf{c}_i = U^H \mathbf{x}_i, \quad i = 1, \dots, M \quad (7.10.27)$$

确定, 其中,  $\mathbf{c}_i = [c_{i1}, \dots, c_{iP}]^T$ ,  $i = 1, \dots, M$ 。

如前所述, Fourier 级数和 Fourier 展开共同构成了 Fourier 分析的理论框架。与此类似, 式 (7.10.26) 和式 (7.10.27) 一起形成了对平稳非周期随机过程的分析方法。由于这种方法是基于特征值和特征向量导出的, 所以很自然地可称为随机过程的特征分析。

对于单个随机向量, 线性特征向量展开式 (7.10.26) 简化为

$$\mathbf{x} = \sum_{k=1}^P c_k \mathbf{u}_k \quad (7.10.28)$$

式中, 展开系数  $c_k$  由向量的内积

$$c_k = \langle \mathbf{u}_k, \mathbf{x} \rangle, \quad k = 1, \dots, P \quad (7.10.29)$$

确定。这恰好就是式 (7.10.18)。

线性特征向量展开式 (7.10.26) 表明, 特征向量可以定义一种新的坐标系。这一坐标系由  $P$  个相互垂直 (即正交) 的坐标组成。当随机向量  $\mathbf{x}$  作为线性系统  $\mathcal{L}$  的输入时, 由式 (7.10.28) 知, 线性系统的输出为

$$\mathcal{L}[\mathbf{x}] = \mathcal{L} \left[ \sum_{k=1}^P c_k \mathbf{u}_k \right] = \sum_{k=1}^P c_k \mathcal{L}[\mathbf{u}_k] \quad (7.10.30)$$

由于  $\mathcal{L}[\mathbf{u}_k] = \lambda_k \mathbf{u}_k$ , 上式给出结果

$$\mathcal{L}[\mathbf{x}] = \sum_{k=1}^P \lambda_k c_k \mathbf{u}_k \quad (7.10.31)$$

这说明, 如果使用特征值和特征向量, 线性系统的输出  $\mathcal{L}[\mathbf{x}]$  将变得容易计算。

综合以上讨论, 可以看出, 特征值和特征向量不仅是随机向量的线性展开的有力工具, 而且也对线性系统输出的分析有着重要的作用。

式 (7.10.28) 可以用来解释随机信号的功率谱。考虑离散随机信号  $x(0), x(1), \dots, x(N-1)$  的功率谱, 定义 Fourier 向量为

$$\mathbf{w} = [1, e^{j\omega}, \dots, e^{j(N-1)\omega}]^H \quad (7.10.32)$$

则  $x(0), x(1), \dots, x(N-1)$  的离散 Fourier 变换即信号的频谱为

$$X(\omega) = \sum_{k=0}^{N-1} x(k) e^{-jk\omega} = \mathbf{w}^H \mathbf{x} \quad (7.10.33)$$

由于信号的功率谱  $P(\omega)$  定义为频谱模值的平方, 故

$$P(\omega) = |X(\omega)|^2 = |\mathbf{w}^H \mathbf{x}|^2 = \mathbf{w}^H (\mathbf{x} \mathbf{x}^H) \mathbf{w} = \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} \quad (7.10.34)$$

式中,  $N \times N$  矩阵  $\hat{\mathbf{R}} = \mathbf{x} \mathbf{x}^H$  是自相关矩阵  $\mathbf{R} = E\{\mathbf{x}(t) \mathbf{x}^H(t)\}$  的瞬时估计。令  $\hat{\mathbf{R}}$  的特征值分解为

$$\hat{\mathbf{R}} = \sum_{k=1}^N \lambda_k \mathbf{u}_k \mathbf{u}_k^H \quad (7.10.35)$$

式中,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$ 。于是, 式 (7.10.34) 定义的功率谱可以写作

$$P(\omega) = \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} = \sum_{k=1}^N \lambda_k |\mathbf{w}^H \mathbf{u}_k|^2 \geq 0 \quad (7.10.36)$$

因为特征值  $\lambda_i \geq 0$ 。

由于 Fourier 向量的元素以角频率  $\omega$  为变量, 功率谱  $P(\omega)$  取连续函数形式。此时, Rayleigh 商为

$$\lambda_N \leq \frac{\mathbf{w}^H \hat{\mathbf{R}} \mathbf{w}}{\mathbf{w}^H \mathbf{w}} \leq \lambda_1 \quad (7.10.37)$$

注意, 对于  $N \times 1$  维 Fourier 向量  $\mathbf{w}$ , 有  $\mathbf{w}^H \mathbf{w} = N$ 。将这以一结果代入式 (7.10.37), 立即有

$$N \lambda_N \leq P(\omega) \leq N \lambda_1 \quad (7.10.38)$$

从以上分析可以得出功率谱分析的以下结论:

- (1) 由于 Rayleigh 商的性质, 功率谱  $P(\omega)$  的取值位于区间  $[N \lambda_N, N \lambda_1]$ 。
- (2) 当 Fourier 向量碰巧是  $\hat{\mathbf{R}}$  的一个特征向量, 并且对应的特征值非零时, 功率谱  $P(\omega)$  取极大值即峰值。

应当指出, 对于一般的随机信号, Fourier 向量不会碰巧与信号相关矩阵的特征向量一致。然而, 对于等距离布置的直线阵列, 这一情况是会发生的。此时, 各个阵元观测信号的空间自相关矩阵是 Hermitian 矩阵, 其理想的特征向量取 Fourier 向量的形式。对此感兴趣的读者可进一步参考文献 [254]。

### 本章小结

矩阵的特征分析包含了丰富多彩的内容。不仅标准的特征值分解有许多有趣的性质和广泛的应用, 而且它还有各类既有趣又极为重要的推广: 广义特征值分解、Rayleigh 商、广义 Rayleigh 商、二次特征值和多个矩阵的联合对角化, 它们之间有以下关系:

- (1) 对称的正定矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值分解等价为  $\mathbf{B}^{-1}\mathbf{A}$  的特征值分解;
  - (2) 满足对称矩阵  $\mathbf{A}$  的 Rayleigh 商的极小值和极大值的解  $(\lambda, \mathbf{u})$  分别是与  $\mathbf{A}$  的最小和最大的特征值对应的特征对;
  - (3) 满足对称矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义 Rayleigh 商的极小值和极大值的解  $(\lambda, \mathbf{u})$  分别是与矩阵束  $(\mathbf{A}, \mathbf{B})$  的最小和最大的广义特征值对应的广义特征对;
  - (4) 二次特征值问题通过线性化, 可以转换为标准的特征值问题求解;
  - (5) 两个对称矩阵的联合对角化等同于这两个矩阵组成的矩阵束的广义特征值分解。
- 围绕这些推广的特征值分解, 本章分别列举了一些典型的应用例子。
- 最后, 本章还讨论了特征分析与 Fourier 分析之间的关系。

### 习 题

**7.1 证明特征值的以下性质:** 若  $\lambda$  是  $n \times n$  矩阵  $\mathbf{A}$  的特征值, 则有

- (1)  $\lambda^k$  是矩阵  $\mathbf{A}^k$  的特征值。
- (2) 若  $\mathbf{A}$  非奇异, 则  $\mathbf{A}^{-1}$  具有特征值  $1/\lambda$ 。
- (3) 矩阵  $\mathbf{A} + \sigma^2 \mathbf{I}$  的特征值为  $\lambda + \sigma^2$ 。

**7.2 证明当  $\mathbf{A}$  为幂等矩阵时, 矩阵  $\mathbf{BA}$  的特征值与  $\mathbf{ABA}$  的特征值相同。**

**7.3 设  $n$  阶矩阵  $\mathbf{A}$  的全部元素为 1, 求  $\mathbf{A}$  的  $n$  个特征值。**

**7.4 设矩阵**

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & y & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

(1) 已知  $\mathbf{A}$  的一个特征值为 3, 试求  $y$  值。

(2) 求矩阵  $\mathbf{P}$ , 使  $(\mathbf{AP})^T \mathbf{AP}$  为对角矩阵。

7.5 令初始值  $u(0) = 2, v(0) = 8$ 。利用特征值求解微分方程

$$u'(t) = 3u(t) + v(t)$$

$$v'(t) = -2u(t) + v(t)$$

7.6 令  $4 \times 4$  维 Hessenberg 矩阵

$$\mathbf{H} = \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ 0 & b_3 & c_3 & d_3 \\ 0 & 0 & c_4 & d_4 \end{bmatrix}$$

证明以下两个结果：

(1) 若  $a_2, b_3, c_4$  均不等于零，并且  $\mathbf{H}$  的任意特征值  $\lambda$  为实数，则  $\lambda$  的几何多重度必定等于 1。

(2) 若  $\mathbf{H}$  与对称矩阵  $\mathbf{A}$  相似，并且  $\mathbf{A}$  的某个特征值  $\lambda$  的代数多重度大于 1，则  $a_2, b_3, c_4$  至少有一个等于零。

7.7 证明以下各题：

(1) 若  $\mathbf{A} = \mathbf{A}^2 = \mathbf{A}^H \in \mathbb{C}^{n \times n}$ ，并且  $\text{rank}(\mathbf{A}) = r < n$ ，则存在  $n \times n$  酉矩阵  $\mathbf{V}$ ，使得

$$\mathbf{V}^H \mathbf{A} \mathbf{V} = \text{diag}(\mathbf{I}_r, \mathbf{0})$$

(2) 若  $\mathbf{A} = \mathbf{A}^H \in \mathbb{C}^{n \times n}$ ，并且  $\mathbf{A}^2 = \mathbf{I}_n$ ，则存在酉矩阵  $\mathbf{V}$ ，使得

$$\mathbf{V}^H \mathbf{A} \mathbf{V} = \text{diag}(\mathbf{I}_r, \mathbf{I}_{n-r})$$

7.8 令  $\mathbf{H} = \mathbf{I} - 2\mathbf{u}\mathbf{u}^H$  为 Householder 变换矩阵。

(1) 证明：具有单位范数的向量  $\mathbf{u}$  是 Householder 变换矩阵的特征向量，并求与之相对应的特征值。

(2) 若向量  $\mathbf{w}$  与  $\mathbf{u}$  正交，证明  $\mathbf{w}$  是矩阵  $\mathbf{H}$  的特征向量，并求与之对应的特征向量。

7.9 设矩阵  $\mathbf{A}$  和  $\mathbf{B}$  相似，其中

$$\mathbf{A} = \begin{bmatrix} -2 & 0 & 0 \\ 2 & x & 2 \\ 3 & 1 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & y \end{bmatrix}$$

(1) 求  $x$  和  $y$  的值。

(2) 求可逆矩阵  $\mathbf{P}$ ，使得  $\mathbf{P}^T \mathbf{A} \mathbf{P} = \mathbf{B}$ 。

7.10 利用分块矩阵，可以将矩阵的特征值问题降维。令

$$\mathbf{A} = \begin{bmatrix} \mathbf{B} & \mathbf{X} \\ \mathbf{O} & \mathbf{C} \end{bmatrix} \quad (\mathbf{O} \text{ 为零矩阵})$$

证明： $\det(\mathbf{A} - \lambda \mathbf{I}) = \det(\mathbf{B} - \lambda \mathbf{I}) \det(\mathbf{C} - \lambda \mathbf{I})$ 。

7.11 令  $\mathbf{u}$  是矩阵  $\mathbf{A}$  与特征值  $\lambda$  对应的一个特征向量。

(1) 证明  $\mathbf{u}$  是矩阵  $\mathbf{A} - 3\mathbf{I}$  的一个特征向量。

(2) 证明  $\mathbf{u}$  是矩阵  $\mathbf{A}^2 - 4\mathbf{A} + 3\mathbf{I}$  的一个特征向量。

7.12<sup>[251]</sup> 令

$$\mathbf{A} = \begin{bmatrix} -2 & 4 & 3 \\ 0 & 0 & 0 \\ -1 & 5 & 2 \end{bmatrix}, \quad f(x) = x^{593} - 2x^{15}$$

证明

$$\mathbf{A}^{593} - 2\mathbf{A}^{15} = -\mathbf{A} = \begin{bmatrix} 2 & -4 & -3 \\ 0 & 0 & 0 \\ 1 & -5 & -2 \end{bmatrix}$$

7.13<sup>[251]</sup> 假定  $a_0, a_1, a_2, \dots$  为正整数序列，并且满足递推关系  $a_{k+1} = a_k + 2a_{k-1}$ ,  $\forall k \geq 1$ 。若  $a_0 = 0, a_1 = 1$ , 求  $a_k$  值。（提示：建立向量  $\begin{bmatrix} a_{k+1} \\ a_k \\ a_0 \end{bmatrix}$  与  $\begin{bmatrix} a_1 \\ a_0 \end{bmatrix}$  之间的关系，并运用 Cayley-Hamilton 定理。）

7.14 已知矩阵  $\mathbf{A} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$ , 求  $e^{\mathbf{At}}$ 。

7.15<sup>[251]</sup> 已知矩阵  $\mathbf{A} = \begin{bmatrix} 1 & 1 & 2 \\ -1 & 2 & 1 \\ 0 & 1 & 3 \end{bmatrix}$ , 求非奇异矩阵  $\mathbf{S}$  使相似矩阵  $\mathbf{B} = \mathbf{S}^{-1}\mathbf{AS}$  为对角矩阵。

7.16 证明：满足  $\mathbf{A}^2 - \mathbf{A} = 2\mathbf{I}$  的矩阵  $\mathbf{A}_{n \times n}$  是可对角化的。

7.17 已知  $\mathbf{u} = [1, 1, -1]^T$  是矩阵  $\mathbf{A} = \begin{bmatrix} 2 & -1 & 2 \\ 5 & a & 3 \\ -1 & b & -2 \end{bmatrix}$  的一个特征向量。

(1) 求  $a, b$  和特征向量  $\mathbf{u}$  对应的特征值。

(2) 矩阵  $\mathbf{A}$  能否相似于对角矩阵？试说明理由。

7.18 证明：矩阵  $\mathbf{A}$  和  $\mathbf{B}$  的 Kronecker 积  $\mathbf{A} \otimes \mathbf{B}$  的非零特征值等于  $\mathbf{A}$  的特征值与  $\mathbf{B}$  的特征值的乘积，即  $\lambda(\mathbf{A} \otimes \mathbf{B}) = \lambda(\mathbf{A})\lambda(\mathbf{B})$ 。

7.19 令  $\mathbf{A}$  和  $\mathbf{B}$  均为 Hermitian 矩阵，并且  $\lambda_i$  和  $\mu_i$  分别是矩阵  $\mathbf{A}$  和  $\mathbf{B}$  的特征值。证明：若  $c_1\mathbf{A} + c_2\mathbf{B}$  具有特征值  $c_1\lambda_i + c_2\mu_i$ ，其中， $c_1, c_2$  为任意标量，则  $\mathbf{AB} = \mathbf{BA}$ 。

7.20 证明：若  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  正定，则  $|\mathbf{A}\lambda^2 + \mathbf{B}\lambda + \mathbf{C}| = 0$  的根具有负的实部。

7.21 令  $\mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{X}$  为  $2 \times 2$  矩阵。证明：矩阵方程  $\mathbf{PX}^2 + \mathbf{QX} + \mathbf{R} = \mathbf{O}$  (零矩阵) 的解  $\mathbf{X}$  的每一个特征值都是  $|\mathbf{P}\lambda^2 + \mathbf{Q}\lambda + \mathbf{C}| = 0$  的根。

7.22 令

$$\mathbf{A} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

若定义矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值  $(\alpha, \beta_i)$  是满足  $\det(\beta\mathbf{A} - \alpha\mathbf{B}) = 0$  的数值  $\alpha$  和  $\beta$ ，试求  $\alpha$  和  $\beta$ 。

7.23<sup>[434, p.286]</sup> 令矩阵束  $(\mathbf{A}, \mathbf{B})$  的广义特征值  $(\alpha, \beta)$  如上题所定义。令  $\lambda_i = (\alpha_i, \beta_i)$  和  $\lambda_j = (\alpha_j, \beta_j)$  是矩阵束  $(\mathbf{A}, \mathbf{B})$  的两个不同广义特征值，并且  $\mathbf{u}_i$  是与广义特征值  $\lambda_i$  对

应的右广义特征向量，而  $w_j$  是与  $\lambda_j$  对应的左广义特征向量，证明

$$\langle Au_i, w_j \rangle = \langle Bu_i, w_j \rangle = 0$$

**7.24** 令矩阵  $G$  是  $A$  的广义逆矩阵，并且  $A$  和  $GA$  都是对称矩阵。证明  $A$  的非零特征值的倒数是广义逆矩阵  $G$  的一个特征值。

**7.25** [36, p.226] 令  $A$  是一个  $n \times n$  复矩阵，其特征值为  $\lambda_1, \lambda_2, \dots, \lambda_n$ 。证明  $A$  为正规矩阵，当且仅当下列条件之一成立：

(1)  $AA^H$  的特征值为  $|\lambda_1|^2, |\lambda_2|^2, \dots, |\lambda_n|^2$ 。

(2)  $A + A^H$  的特征值为  $\lambda_1 + \lambda_1^*, \lambda_2 + \lambda_2^*, \dots, \lambda_n + \lambda_n^*$ 。

**7.26** 利用特征方程证明：若  $\lambda$  是矩阵  $A$  的实特征值，则  $A + A^{-1}$  的特征值的绝对值等于或大于 2。

**7.27** 设  $A_{4 \times 4}$  满足条件  $|3I_4 + A| = 0, AA^T = 2I_4$  和  $|A| < 0$ 。求矩阵  $A$  的伴随矩阵  $\text{adj}(A) = \det(A)A^{-1}$  的一个特征值。

**7.28** 证明二次型  $f = \mathbf{x}^T A \mathbf{x}$  在  $\|\mathbf{x}\| = 1$  时的最大值等于对称矩阵  $A$  的最大特征值。（提示：将  $f$  化为标准二次型。）

**7.29** 证明：若  $\lambda$  是矩阵  $AB$  的一个非零特征值，则它也是矩阵  $BA$  的非零特征值 ( $A, B$  不一定为正方矩阵，但  $AB$  和  $BA$  分别是正方的)。

**7.30** 令  $A_{n \times n}$  为对称矩阵，其特征值为  $\lambda_i, i = 1, 2, \dots, n$ 。证明

$$\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 = \sum_{k=1}^n \lambda_k^2$$

**7.31** 设  $A_{n \times n}$  的全部特征值为  $\lambda_1, \lambda_2, \dots, \lambda_n$ ，且与  $\lambda_i$  对应的特征向量为  $u_i$ 。试求

(1)  $P^{-1}AP$  的特征值与相对应的特征向量。

(2)  $(P^{-1}AP)^T$  的特征值与相对应的特征向量。

**7.32** 设  $n \times n$  矩阵  $A = \{a_{ij}\}$ ，其中  $a_{ij} = a$  ( $i = j$ ) 或  $b$  ( $i \neq j$ )。求  $A$  的特征值及特征向量。

**7.33** 令  $p(z) = a_0 + a_1 z + \dots + a_n z^n$  为一多项式。证明：矩阵  $A$  的特征向量一定是矩阵多项式  $p(A)$  的特征向量，但  $p(A)$  的特征向量不一定是  $A$  的特征向量。

**7.34** 令  $A$  为实斜对称矩阵，即其元素  $a_{ij} = -a_{ji}$ 。证明：

(1)  $A$  的特征值为纯虚数或零。

(2) 若  $u + jv$  是与特征值  $j\mu$  (其中， $\mu$  是非零的实数) 对应的特征向量，并且  $u$  和  $v$  为实向量，则  $u$  与  $v$  正交。

**7.35** 令  $A$  是一个正交矩阵， $\lambda$  是  $A$  的一个不等于  $\pm 1$ ，但其模为 1 的特征值，并且  $u + jv$  是与该特征值对应的特征向量，其中， $u$  和  $v$  为实向量。证明  $u$  和  $v$  正交。

**7.36** 一滤波器的抽头延迟线的输出为  $y(k) = \mathbf{a}^T \mathbf{x}(k)$ ，其中  $\mathbf{a} = [a_0, a_1, \dots, a_n]^T$  和  $\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-n)]^T$ 。令  $R_x \stackrel{\text{def}}{=} E\{\mathbf{x}\mathbf{x}^T\} = Q\Sigma Q^T$ ，其中  $\Sigma =$

$\text{diag}(\lambda_0, \lambda_1, \dots, \lambda_n)$ 。如果输出序列  $\{y(k)\}$  的均方值为  $J_a = \frac{1}{2}\mathbb{E}\{y^2(k)\}$ , 证明以下结果:

- (1) 在条件  $a^T a = 1$  的约束下, 使  $J_a$  最小化等价于  $J_w = \frac{1}{2} \sum_{i=0}^n w_i^2 \lambda_i$  的最小化, 其中  $w = [w_0, w_1, \dots, w_n]^T$ ,  $\sum_{i=0}^n w_i^2 = 1$ , 且  $w = Q^T a$ 。
- (2) 若取  $w = [\pm 1, 0, \dots, 0]^T$ , 则使  $J_a$  最小化的最优向量  $a = \pm a_0$ , 其中,  $a_0$  是矩阵  $R_x$  相对于最小特征值  $\lambda_0$  的特征向量。

7.37 证明: 一个  $n \times n$  实对称矩阵  $A$  可以写作

$$A = \sum_{i=1}^n \lambda_i Q_i$$

式中,  $\lambda_i$  是  $A$  的特征值;  $Q_i$  为非负定矩阵, 并且不仅满足正交条件

$$Q_i Q_j = O, \quad i \neq j$$

而且还是幂等矩阵, 即  $Q_i^2 = Q_i$ 。矩阵  $A$  的这一表示称为  $A$  的谱分解<sup>[36, p.64]</sup>。

7.38 已知

$$A = \begin{bmatrix} 3 & -1 & 0 \\ 0 & 5 & -2 \\ 0 & 0 & 9 \end{bmatrix}$$

求非奇异矩阵  $S$  使得相似变换  $S^{-1}AS = B$  为对角矩阵, 并求对角矩阵  $B$ 。

7.39 已知矩阵

$$A = \begin{bmatrix} 0 & 0 & 1 \\ x & 1 & y \\ 1 & 0 & 0 \end{bmatrix}$$

有三个线性无关的特征向量, 求  $x$  和  $y$  应该满足的条件。

7.40 设

$$A = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

试通过求矩阵  $S$  使得  $S^{-1}AS = D$  (对角矩阵), 证明  $A$  是可对角化的。

7.41 证明下列结论:

- (1) 若  $u_1, u_2, \dots, u_p$  是矩阵  $A_{n \times n}$  与同一特征值  $\lambda$  对应的特征向量, 则  $u_1, u_2, \dots, u_p$  的任意一个线性组合也是  $A$  的属于特征值  $\lambda$  的特征向量。
- (2) 若  $u_1, u_2, \dots, u_p$  是矩阵  $A_{n \times n}$  与不同特征值  $\lambda_1, \lambda_2, \dots, \lambda_p$  对应的特征向量, 则当  $c_1, c_2, \dots, c_p$  中至少有两个不为零时,  $c_1 u_1 + c_2 u_2 + \dots + c_p u_p$  必定不是  $A$  的特征向量。

7.42 设三阶实对称矩阵  $A$  的特征值为  $\lambda_1 = -1, \lambda_2 = 1, \lambda_3 = -2$ , 且与  $\lambda_1$  对应的特征向量为  $u_1 = [0, 1, 1]^T$ , 求矩阵  $A$  的表达式。

7.43 已知矩阵

$$A = \begin{bmatrix} 1 & -1 & 1 \\ x & 4 & y \\ -3 & -3 & 5 \end{bmatrix}$$

有三个线性无关的特征向量，且  $\lambda = 2$  是  $A$  的二重特征值。试求可逆矩阵  $P$ ，使得  $P^{-1}AP$  为对角矩阵。

**7.44** 设向量  $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_n]^T$  和  $\beta = [\beta_1, \beta_2, \dots, \beta_n]^T$  是两个正交的非零向量。若令  $A = \alpha\beta^T$ ，试求

(1)  $A^2$ ；

(2) 矩阵  $A$  的特征值和特征向量。

**7.45** 已知矩阵

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

且  $B = (kI + A)^2$ ，其中， $k$  为实数。

(1) 求矩阵  $B$  的对角化矩阵  $A$ 。

(2) 试问： $k$  为何值时，矩阵  $B$  是正定的？

**7.46** 设

$$|A| = \begin{vmatrix} a & -1 & c \\ 5 & b & 3 \\ 1-c & 0 & -a \end{vmatrix} = -1$$

又  $A$  的伴随矩阵  $\text{adj}(A) = |A|A^{-1}$  有一个特征值为  $\lambda$ ，与之对应的特征向量  $u = [-1, -1, 1]^T$ 。求  $a, b, c$  和  $\lambda$  的值。

**7.47** 求矩阵

$$A = \begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{bmatrix}$$

的正交基。

**7.48** 求解广义特征值问题  $Ax = \lambda Bx$  时，矩阵  $B$  必须是非奇异的。现在假定  $B$  奇异，其广义逆矩阵为  $B^\dagger$ 。

(1) 令  $(\lambda, x)$  是矩阵  $B^\dagger A$  的一个特征对。证明，该特征对是矩阵束  $(A, B)$  的一个广义特征对，若  $Ax$  是矩阵  $BB^\dagger$  与特征值 1 对应的特征向量。

(2) 令  $\lambda, x$  满足  $Ax = \lambda Bx$ 。证明：若  $x$  也是矩阵  $B^\dagger B$  与特征值 1 对应的特征向量，则  $(\lambda, x)$  是矩阵  $B^\dagger A$  的一个特征对。

**7.49** 令矩阵束  $(A, B)$  与广义特征值  $\lambda_1$  对应的右特征向量为  $u_1$ ，左特征向量为  $v_1$ ，并且  $\langle Bu_1, Bv_1 \rangle = 1$ 。试证明：矩阵束  $(A, B)$  和

$$A_1 = A - \sigma_1 Bu_1 v_1^H B^H, \quad B_1 = B - \sigma_2 A u_1 v_1^H B^H$$

具有相同的左和右特征向量。式中，假定移位因子  $\sigma_1$  和  $\sigma_2$  满足条件  $1 - \sigma_1 \sigma_2 \neq 0$ 。

**7.50** 若  $A$  和  $B$  均为正定的 Hermitian 矩阵，证明：广义特征值必定是实的，并且与不同广义特征值对应的广义特征向量相对于正定矩阵  $A$  和  $B$  分别正交，即有

$$x_i^H Ax_j = x_i^H Bx_j = 0, \quad i \neq j$$

## 7.51 假定

$$\begin{bmatrix} \mathbf{O} & \mathbf{A} \\ \mathbf{A}^T & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{z} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{B}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{B}_2 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{z} \end{bmatrix}$$

式中,  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{B}_1 \in \mathbb{R}^{m \times m}$ ,  $\mathbf{B}_2 \in \mathbb{R}^{n \times n}$ 。假定矩阵  $\mathbf{B}_1$  和  $\mathbf{B}_2$  正定, 并且分别具有 Cholesky 三角因子  $\mathbf{G}_1$  和  $\mathbf{G}_2$ 。将上式描述的广义特征值问题与  $\mathbf{G}_1^{-1}\mathbf{A}\mathbf{G}_2^{-T}$  联系起来。

## 7.52 已知矩阵

$$\begin{aligned} \frac{dx(t)}{dt} &= \omega x(t) + 0y(t) + 0z(t) \\ \frac{dy(t)}{dt} &= 0x(t) + \omega y(t) + z(t) \\ \frac{dz(t)}{dt} &= 0x(t) + y(t) + \omega z(t) \end{aligned}$$

求三阶矩阵微分方程

$$\Phi^{(n)}(t) + c_2 \Phi''(t) + c_1 \Phi'(t) + c_0 \Phi(t) = \mathbf{O}$$

满足初始条件

$$\Phi(0) = \mathbf{I}, \quad \Phi'(0) = \mathbf{A}, \quad \Phi''(0) = \mathbf{A}^2$$

的解  $\Phi(t)$ 。

7.53 若  $\mathbf{M}, \mathbf{C}, \mathbf{K}$  对称和正定, 证明

$$|\lambda^2 \mathbf{M} + \lambda \mathbf{C} + \mathbf{K}| = 0$$

的根有负的实部。

7.54 已知  $m \times m$  矩阵

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & & \ddots & \\ & & & 2 & -1 \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}$$

证明

$$|\lambda \mathbf{I} - \mathbf{A}| = \prod_{i=1}^m \left( \lambda - 2 - 2 \cos \frac{i\pi}{m+1} \right)$$

7.55 假定  $n \times n$  维 Hermitian 矩阵  $\mathbf{A}$  的特征值按照顺序  $\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \dots \geq \lambda_n(\mathbf{A})$  排列。用 Rayleigh 商证明:

- (1)  $\lambda_1(\mathbf{A} + \mathbf{B}) \geq \lambda_1(\mathbf{A}) + \lambda_n(\mathbf{B})$ 。
- (2)  $\lambda_n(\mathbf{A} + \mathbf{B}) \geq \lambda_n(\mathbf{A}) + \lambda_n(\mathbf{B})$ 。

7.56 利用 Rayleigh 商证明: 对于任何一个  $n \times n$  对称矩阵  $\mathbf{A}$  和任何一个  $n \times n$  半正定矩阵  $\mathbf{B}$ , 特征值服从不等式

$$\lambda_k(\mathbf{A} + \mathbf{B}) \geq \lambda_k(\mathbf{A}), \quad k = 1, 2, \dots, n$$

若  $\mathbf{B}$  正定, 则

$$\lambda_k(\mathbf{A} + \mathbf{B}) > \lambda_k(\mathbf{A}), \quad k = 1, 2, \dots, n$$

7.57 令  $\mathbf{A}$  和  $\mathbf{B}$  分别是  $n \times n$  对称矩阵, 证明

$$\lambda_1(\mathbf{A} + \mathbf{B}) \leq \lambda_1(\mathbf{A}) + \lambda_1(\mathbf{B})$$

和

$$\lambda_n(\mathbf{A} + \mathbf{B}) \geq \lambda_n(\mathbf{A}) + \lambda_n(\mathbf{B})$$

7.58 令  $\mathbf{A}$  是一个  $n \times n$  矩阵 (不一定对称)。证明对任意  $n \times 1$  向量  $\mathbf{x}$ , 恒有不等式

$$(\mathbf{x}^T \mathbf{A} \mathbf{x})^2 \leq (\mathbf{x}^T \mathbf{A} \mathbf{A}^T \mathbf{x})(\mathbf{x}^T \mathbf{x})$$

和

$$\frac{1}{2} \left| \frac{\mathbf{x}^T (\mathbf{A} + \mathbf{A}^T) \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \right| \leq \left( \frac{\mathbf{x}^T \mathbf{A} \mathbf{A}^T \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \right)^{1/2}$$

7.59 考虑对称矩阵序列

$$\mathbf{A}_r = [a_{ij}], \quad i, j = 1, 2, \dots, r$$

其中,  $r = 1, 2, \dots, n$ 。令  $\lambda_i(\mathbf{A}_r)$ ,  $i = 1, 2, \dots, r$  是矩阵  $\mathbf{A}_r$  的第  $i$  个特征值, 并且

$$\lambda_1(\mathbf{A}_r) \geq \lambda_2(\mathbf{A}_r) \geq \dots \geq \lambda_r(\mathbf{A}_r)$$

则

$$\lambda_{k+1}(\mathbf{A}_{i+1}) \leq \lambda_k(\mathbf{A}_i) \leq \lambda_k(\mathbf{A}_{i+1})$$

这一结果称为 Sturmian 分离定理<sup>[36, p.117]</sup>。试使用 Rayleigh 商证明这一定理。

7.60 证明式(7.8.1) 是双曲线二次特征值问题, 当且仅当  $\mathbf{Q}(\lambda)$  对某些实特征值  $\lambda \in R$  负定。

7.61 对于两个相同维数的任意正方矩阵  $\mathbf{A}, \mathbf{B}$ , 证明

$$(1) 2(\mathbf{A} \mathbf{A}^T + \mathbf{B} \mathbf{B}^T) - (\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{B})^T \text{ 半正定}.$$

$$(2) \operatorname{tr}[(\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{B})^T] \leq 2[\operatorname{tr}(\mathbf{A} \mathbf{A}^T) + \operatorname{tr}(\mathbf{B} \mathbf{B}^T)].$$

$$(3) \lambda[(\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{B})^T] \leq 2[\lambda(\mathbf{A} \mathbf{A}^T) + \lambda(\mathbf{B} \mathbf{B}^T)].$$

7.62 令  $\mathbf{A}$  和  $\mathbf{B}$  是两个维数相同的半正定矩阵, 证明<sup>[524]</sup>

$$\sqrt{\operatorname{tr}(\mathbf{AB})} \leq \frac{1}{2} [\operatorname{tr}(\mathbf{A}) + \operatorname{tr}(\mathbf{B})]$$

等号成立, 当且仅当  $\mathbf{A} = \mathbf{B}$  和  $\operatorname{rank}(\mathbf{A}) \leq 1$ 。

**7.63** 对于多项式  $p(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_1x + a_0$ , 称  $n \times n$  矩阵

$$\mathbf{C}_p = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}$$

为多项式  $p(x)$  的友矩阵。利用数学归纳法证明：对于  $n \geq 2$ , 恒有

$$\det(\mathbf{C}_p - \lambda \mathbf{I}) = (-1)^n(a_0 + a_1\lambda + \cdots + a_{n-1}\lambda^{n-1} + \lambda^n) = (-1)^n p(\lambda)$$

**7.64** 令  $p(x) = x^3 + a_2x^2 + a_1x + a_0$ , 并且  $\lambda$  是多项式  $p(x)$  的一个零点。

(1) 写出多项式  $p(x)$  的友矩阵  $\mathbf{C}_p$ 。

(2) 解释为什么  $\lambda^3 = -a_2\lambda^2 - a_1\lambda - a_0$ , 并证明  $(1, \lambda, \lambda^2)$  是多项式  $p(x)$  的友矩阵  $\mathbf{C}_p$  的特征值。

**7.65** [344] 令  $\mathbf{A}, \mathbf{B}$  和  $\mathbf{A} - \mathbf{B}$  为半正定矩阵。证明： $\mathbf{B}^\dagger - \mathbf{A}^\dagger$  是半正定的，当且仅当  $\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{B})$ 。

**7.66** 令  $\mathbf{A}$  是一个  $n \times n$  矩阵(不一定对称)。证明：对于每一个  $n \times 1$  向量  $\mathbf{x}$ , 恒有

$$(\mathbf{x}^\top \mathbf{A} \mathbf{x})^2 \leq (\mathbf{x}^\top \mathbf{A} \mathbf{A}^\top \mathbf{x})(\mathbf{x}^\top \mathbf{x})$$

因此有

$$\frac{1}{2} \left| \frac{\mathbf{x}^\top (\mathbf{A} + \mathbf{A}^\top) \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \right| \leq \left( \frac{\mathbf{x}^\top \mathbf{A} \mathbf{A}^\top \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \right)^{1/2}$$

**7.67** 证明：对于任何一个  $n \times n$  对称矩阵  $\mathbf{A}$  和任何一个半正定矩阵  $\mathbf{B}$ , 恒有特征值不等式

$$\lambda_k(\mathbf{A} + \mathbf{B}) \geq \lambda_k(\mathbf{A}), \quad k = 1, 2, \dots, n$$

并且若  $\mathbf{B}$  正定，则有

$$\lambda_k(\mathbf{A} + \mathbf{B}) > \lambda_k(\mathbf{A}), \quad k = 1, 2, \dots, n$$

(提示：利用 Rayleigh 商的极大-极小原理。)

**7.68** [328] 令  $\mathbf{A}$  是  $n \times n$  正定矩阵，其特征值  $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ 。证明：矩阵

$$(\lambda_1 + \lambda_n) \mathbf{I}_n - \mathbf{A} - (\lambda_1 \lambda_n) \mathbf{A}^{-1}$$

半正定，并且其秩  $\leq n - 2$ 。(提示：利用  $x^2 - (a + b)x + ab \leq 0, \forall x \in [a, b]$ 。)

**7.69** [328] 令  $\mathbf{A}$  是一个  $n \times n$  正定矩阵，其特征值  $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ 。利用上一习题证明：

$$1 \leq (\mathbf{x}^\top \mathbf{A} \mathbf{x})(\mathbf{x}^\top \mathbf{A}^{-1} \mathbf{x}) \leq \frac{(\lambda_1 + \lambda_n)^2}{4\lambda_1 \lambda_n}$$

对所有满足  $\mathbf{x}^\top \mathbf{x} = 1$  的实向量  $\mathbf{x}$  成立。这一不等式称为 Kantorovich 不等式 [258]。

## 第8章 子空间分析与跟踪

在涉及逼近、最优化、微分方程、通信、信号处理、系统科学等问题中，子空间起着非常重要的作用。向量子空间的框架可以帮助我们回答一些重要问题：例如，怎样才能对复杂函数获得一个好的多项式逼近？如何求微分方程好的逼近解？怎样设计一个更好的信号处理器？诸如此类的问题实际上是许多工程应用的核心问题。向量子空间为解决这些问题提供了一类有效的方法——子空间方法。

本章主要讨论子空间的分析理论，介绍子空间方法的一些典型应用。在很多复杂的工程问题中，子空间是时变的，而我们又需要对接收信号作实时处理，或者对系统进行实时控制。在这些场合，需要对子空间进行跟踪。因此，本章还将重点讨论如何对子空间进行跟踪与更新。

### 8.1 子空间的一般理论

在具体讨论各种子空间之前，有必要先介绍子空间的基本概念、子空间之间的代数关系和几何关系等。

#### 8.1.1 子空间的基

令  $V = \mathbb{C}^n$  为  $n$  维复向量空间。考虑  $m$  个  $n$  维复向量的子集合，其中  $m < n$ 。

**定义 8.1.1** 若  $S = \{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  是向量空间  $V$  的向量子集合，则  $\mathbf{u}_1, \dots, \mathbf{u}_m$  的所有线性组合的集合  $W$  称为由  $\mathbf{u}_1, \dots, \mathbf{u}_m$  张成的子空间，定义为

$$W = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_m\} = \{\mathbf{u} | \mathbf{u} = a_1\mathbf{u}_1 + \dots + a_m\mathbf{u}_m\} \quad (8.1.1)$$

张成子空间  $W$  的每个向量称为  $W$  的生成元 (generator)，而所有生成元组成的集合  $\{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  称为子空间的张成集 (spanning set)。一个只包含了零向量的向量子空间称为平凡子空间 (trivial subspace)。

**定理 8.1.1** 张成集定理 (spanning set theorem)<sup>[306, p.234]</sup> 令  $S = \{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  是向量空间  $V$  的一个子集，并且  $W = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  是由  $S$  的  $m$  个列向量张成的一个子空间。

(1) 如果  $S$  内有某个向量 (例如  $\mathbf{u}_k$ ) 是其他向量的线性组合，则从  $S$  中删去向量  $\mathbf{u}_k$  后，其他向量仍然张成子空间  $W$ 。

(2) 若  $W \neq \{\mathbf{0}\}$  即  $W$  为非平凡子空间，则在  $S$  内一定存在某个由线性无关的向量组成的子集合，它张成子空间  $W$ 。

**证明** (1) 由于子空间的生成只与张成集的向量有关, 与它们的排列顺序无关, 故不失一般性, 可以假定  $S$  内的向量经过排列, 使得向量  $\mathbf{u}_m$  是  $\mathbf{u}_1, \dots, \mathbf{u}_{m-1}$  的线性组合

$$\mathbf{u}_m = a_1 \mathbf{u}_1 + \cdots + a_{m-1} \mathbf{u}_{m-1}$$

若  $\mathbf{x}$  为子空间  $W$  内的某个向量, 则对合适的标量  $c_1, \dots, c_m$ , 可以将  $\mathbf{x}$  写作

$$\mathbf{x} = c_1 \mathbf{u}_1 + \cdots + c_{m-1} \mathbf{u}_{m-1} + c_m \mathbf{u}_m$$

将  $\mathbf{u}_m$  的线性组合表达式代入上式, 易看出  $\mathbf{x}$  是  $\mathbf{u}_1, \dots, \mathbf{u}_{m-1}$  的线性组合。因此, 删去  $\mathbf{u}_m$  后, 向量子集合  $\{\mathbf{u}_1, \dots, \mathbf{u}_{m-1}\}$  仍然张成子空间  $W$ , 因为  $\mathbf{x}$  是  $W$  的一任意元素。

(2) 如果  $S$  内仍然存在与其他向量线性相关的向量, 则可以继续删去该向量, 一直到删去所有与其他向量线性相关的向量为止。然而, 由于  $W \neq \{\mathbf{0}\}$ , 所以在  $S$  内至少会剩下一个非零向量不至于被删去。换言之, 张成集一定存在。 ■

假定从向量集合  $S = \{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  中删去与其他向量线性相关的所有多余向量后, 剩下  $p$  个线性无关的向量  $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ , 它们仍然张成子空间  $W$ 。在张成同一子空间  $W$  的意义上, 称  $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  和  $\{\mathbf{u}_1, \dots, \mathbf{u}_m\}$  为等价张成集 (equivalent spanning sets)。由此可引出子空间的基的概念。

**定义 8.1.2** 令  $W$  是一向量子空间。向量集合  $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  称为  $W$  的一组基, 若下列两个条件满足:

(1) 子空间  $W$  由向量  $\mathbf{u}_1, \dots, \mathbf{u}_p$  张成, 即

$$W = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$$

(2) 向量集合  $B = \{\mathbf{u}_1, \dots, \mathbf{u}_p\}$  是一线性无关的集合。

定理 8.1.1 的 (1) 给出了从子空间  $W$  的张成集  $S$  构造  $W$  的基的原则: 删去所有与其他向量线性有关的向量; 定理的 (2) 则保证了非平凡子空间  $W$  的基的存在性。

关于子空间的基, 有以下两点重要的观察:

(1) 当使用张成集定理从向量集合  $S$  中删去某个向量时, 一旦  $S$  变成线性无关向量的集合, 则必须立即停止从  $S$  内再删除向量。如果删去不是其他剩余向量的线性组合的额外向量, 则较小的向量集合将不再张成原子空间  $W$ 。因此, 子空间的一组基是一个尽可能小的张成集。换句话说, 张成子空间  $W$  的基向量一个也不能少。

(2) 一组基也是线性无关向量的尽可能大的集合。令  $S$  是子空间  $W$  的一组基, 如果从子空间  $W$  内, 给  $S$  再扩大一个向量 (例如  $\mathbf{w}$ ), 则新的向量集合不可能是线性无关的, 因为  $S$  张成子空间  $W$ , 并且  $W$  内的向量  $\mathbf{w}$  本身就是  $S$  内各个基向量的线性组合。因此, 张成子空间  $W$  的基向量一个也不能多。

需要注意的是, 当提及某个向量子空间的基时, 并非说它是唯一的基, 而只是强调它是其中的一组基。虽然一个向量子空间的基可能有多种选择, 但所有的基都必定含有相

同数目的线性无关向量，否则有较多向量的张成集合就不能算作一组基。从这一讨论中，很容易引出子空间的维数的概念。

**定义 8.1.3** 子空间  $W$  的任何一组基的向量个数称为  $W$  的维数，用符号  $\dim(W)$  表示。若  $W$  的任何一组基都不是由有限个线性无关的向量组成时，则称  $W$  是无限维向量子空间 (infinite-dimensional vector subspace)。

由于任何一个零向量都与其他向量线性相关，所以不失一般性，通常假定在子空间的张成集合中不含零向量。

对于给定的张成集合  $\{u_1, u_2, \dots, u_n\}$ ，很容易构成子空间  $\text{Span}\{u_1, u_2, \dots, u_n\}$  的一组基，详见 8.2 节的讨论。

给向量子空间  $W$  规定一组基的一个重要原因是：能够为子空间  $W$  提供一坐标系。下面的定理说明了坐标系的存在性。

**定理 8.1.2** 令  $B = \{b_1, b_2, \dots, b_n\}$  是  $n$  维向量子空间  $W$  的一组基，则对于  $W$  中的任何一个向量  $x$ ，都存在一组唯一的标量  $c_1, c_2, \dots, c_n$ ，使得  $x$  可以表示为

$$x = c_1 b_1 + c_2 b_2 + \cdots + c_n b_n \quad (8.1.2)$$

**证明** [306, p.240] 由于基  $B$  张成子空间  $W$ ，所以该子空间的任何一个向量都可以表示为这些基向量的线性组合，即式 (8.1.2) 成立。假定  $x$  存在另外一种表示

$$x = d_1 b_1 + d_2 b_2 + \cdots + d_n b_n$$

则有

$$\mathbf{0} = x - x = (c_1 - d_1)b_1 + (c_2 - d_2)b_2 + \cdots + (c_n - d_n)b_n$$

因为  $B$  的向量  $b_1, b_2, \dots, b_n$  线性无关，故上式成立的条件是  $c_i = d_i, i = 1, 2, \dots, n$ 。这就证明了式 (8.1.2) 的唯一性。 ■

上述定理称为子空间向量的唯一表示定理。系数  $c_1, c_2, \dots, c_n$  的唯一性，使得可以利用它们构成子空间  $W$  表示的  $n$  个坐标，从而组成子空间的坐标系。

### 8.1.2 无交连、正交与正交补

在子空间分析中，两个子空间之间的关系由这两个子空间的元素（即向量）之间的关系刻画。下面讨论子空间之间的代数关系。

子空间  $S_1, S_2, \dots, S_n$  的交

$$S = S_1 \cap S_2 \cap \cdots \cap S_n \quad (8.1.3)$$

是子空间  $S_1, S_2, \dots, S_n$  共同拥有的所有向量组成的集合。若这些子空间共同的唯一向量为零向量，即  $S = S_1 \cap S_2 \cap \cdots \cap S_n = \{\mathbf{0}\}$ ，则称子空间  $S_1, S_2, \dots, S_n$  无交连 (disjoint)。无交连的子空间的并  $S = S_1 \cup S_2 \cup \cdots \cup S_n$  称为子空间的直和，记作

$$S = S_1 \oplus S_2 \oplus \cdots \oplus S_n \quad (8.1.4)$$

此时, 每一个向量  $\mathbf{x} \in S$  具有唯一的分解表示  $\mathbf{x} = \mathbf{a}_1 + \mathbf{a}_2 + \cdots + \mathbf{a}_n$ , 其中,  $\mathbf{a}_i \in S_i$ 。

若一向量与子空间  $S$  的所有向量都正交, 则称该向量正交于子空间  $S$ 。推而广之, 称子空间  $S_1, S_2, \dots, S_n$  为正交子空间, 记作  $S_i \perp S_j$ ,  $i \neq j$ , 若  $\mathbf{a}_i \perp \mathbf{a}_j$  对所有  $\mathbf{a}_i \in S_i, \mathbf{a}_j \in S_j$  ( $i \neq j$ ) 恒成立。

特别地, 与子空间  $S$  正交的所有向量的集合组成一个向量子空间, 称为  $S$  的正交补 (orthogonal complement) 空间, 记作  $S^\perp$ 。具体而言, 令  $S$  为一向量空间, 则称向量空间  $S^\perp$  为  $S$  的正交补, 若

$$S^\perp = \{\mathbf{x} | \mathbf{x}^T \mathbf{y} = 0, \forall \mathbf{y} \in S\} \quad (8.1.5)$$

子空间  $S$  和它的正交补  $S^\perp$  的维数满足关系式

$$\dim(S) + \dim(S^\perp) = \dim(V) \quad (8.1.6)$$

顾名思义, 子空间  $S$  在向量空间  $V$  的正交补空间  $S^\perp$  含有正交和补充双重含义:

(1) 子空间  $S^\perp$  与  $S$  正交;

(2) 向量空间  $V$  是子空间  $S$  与  $S^\perp$  的直和, 即  $V = S \oplus S^\perp$ 。这表明, 向量空间  $V$  是由子空间  $S$  补充  $S^\perp$  而成。

下面是无交连子空间、正交子空间和正交补空间的关系。

(1) 无交连是比正交更弱的条件, 这是因为: 两个子空间无交连, 只是表明这两个子空间没有任何一对非零的共同向量, 并不意味着这两个向量之间的任何其他关系。与之相反, 当子空间  $S_1$  和  $S_2$  正交时, 任意两个向量  $\mathbf{x} \in S_1$  和  $\mathbf{y} \in S_2$  都是正交的, 它们之间没有任何相关的一部分, 即  $S_1$  和  $S_2$  一定是无交连的。因此, 无交连的两个子空间不一定正交, 但正交的两个子空间必定是无交连的。

(2) 正交补空间是一个比正交子空间更严格的概念: 子空间  $S$  在向量空间  $V$  的正交补  $S^\perp$  一定与  $S$  正交, 但与  $S$  正交的子空间一般不是  $S$  的正交补。例如, 向量空间  $V$  内可能会有多个子空间  $S_1, S_2, \dots, S_p$  都与子空间  $S$  正交, 只要  $\mathbf{x}_i^T \mathbf{y} = 0, \forall \mathbf{x}_i \in S_i, i = 1, 2, \dots, p; \mathbf{y} \in S$ 。因此, 不能说其中的某个正交子空间  $S_i$  是  $S$  的正交补。由于向量空间  $V$  是由它的子空间  $S$  与正交补  $S^\perp$  补充而成, 所以当向量空间  $V$  和子空间  $S$  给定之后, 正交补  $S^\perp$  便是唯一确定的。

特别地, 向量空间  $\mathbb{R}^m$  的每一个向量  $\mathbf{u}$  都可以用唯一的方式分解为子空间  $S$  的向量  $\mathbf{x}$  与正交补  $S^\perp$  的向量  $\mathbf{y}$  之和, 即

$$\mathbf{u} = \mathbf{x} + \mathbf{y}, \quad \mathbf{x} \perp \mathbf{y} \quad (8.1.7)$$

这一分解形式称为向量的正交分解。向量的正交分解在信号处理、模式识别、自动控制、系统科学等学科中有着广泛的应用。

**例 8.1.1** 函数  $u(t)$  称为严格平方可积分函数, 记作  $u(t) \in L^2(\mathbb{R})$ , 若

$$\int_{-\infty}^{\infty} |u(t)|^2 dt < \infty$$

在小波分析中，通常使用多个分辨率对平方可积分函数或信号  $u(t) \in L^2(R)$  进行逼近，称为函数或信号的多分辨率分析。在多分辨率分析中，需要构造  $L^2(R)$  空间内的一个子空间列或链  $\{V_j : j \in \mathbb{Z}\}$ ，使它具有一些所期望的性质。其中，这个子空间列必须具有包容性

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots$$

根据这一包容性知， $V_j$  是  $V_{j+1}$  的子空间。因此，一定存在  $V_j$  的正交补  $W_j$ ，使得

$$V_{j+1} = V_j \oplus W_j$$

式中， $V_j$  和  $W_j$  分别称为分辨率为  $2^{-j}$  情况下的尺度子空间和小波子空间。满足上述条件的多分辨率分析称为正交多分辨率分析。

满足关系式

$$\{0\} \subset S_1 \subset S_2 \subset \cdots \subset S_m$$

的子空间集  $\{S_1, S_2, \dots, S_m\}$  称为子空间套。

一个特征向量定义一个一维子空间，它相对于左乘矩阵  $A$  是不变的。更一般地，有不变子空间 (invariant subspace) 的下述定义 [198]。

**定义 8.1.4** 一个子空间  $S \subseteq \mathbb{C}^n$  称为 (相对于)  $A$  不变的，若

$$x \in S \implies Ax \in S$$

**例 8.1.2** 令  $n \times n$  (对称或非对称) 矩阵  $A$  的特征向量为  $u_1, u_2, \dots, u_n$ ，且  $S = \text{Span}\{u_1, u_2, \dots, u_n\}$ ，则由于  $Au_i = \lambda_i u_i, i = 1, 2, \dots, n$ ，故

$$u_i \in S \implies Au_i \in S, \quad i = 1, 2, \dots, n$$

这表明，由  $A$  的特征向量张成的子空间  $S$  是相对于  $A$  不变的子空间。

对  $n \times n$  矩阵  $A$  的任意一个特征值  $\lambda$ ，子空间  $\text{Null}(A - \lambda I)$  是相对于  $A$  不变的子空间，因为

$$u \in \text{Null}(A - \lambda I) \implies (A - \lambda I)u = \mathbf{0} \implies Au = \lambda u \in \text{Null}(A - \lambda I)$$

零空间  $\text{Null}(A - \lambda I)$  称为矩阵  $A$  与特征值  $\lambda$  对应的特征空间 (eigenspace)。

令  $A \in \mathbb{C}^{n \times n}$ ,  $B \in \mathbb{C}^{k \times k}$ ,  $X \in \mathbb{C}^{n \times k}$ ，并且  $X = [x_1, x_2, \dots, x_k]$ ，则  $AX = XB$  的第  $j$  列为

$$\begin{aligned} Ax_j &= \begin{bmatrix} b_{1j}x_{11} + b_{2j}x_{12} + \cdots + b_{kj}x_{1k} \\ b_{1j}x_{21} + b_{2j}x_{22} + \cdots + b_{kj}x_{2k} \\ \vdots \\ b_{1j}x_{n1} + b_{2j}x_{n2} + \cdots + b_{kj}x_{nk} \end{bmatrix} \\ &= b_{1j}x_1 + b_{2j}x_2 + \cdots + b_{kj}x_k \end{aligned}$$

因此, 若  $S = \text{Span}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ , 则

$$\mathbf{A}\mathbf{x}_j \in S, \quad j = 1, 2, \dots, k$$

换言之, 子空间  $S = \text{Span}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$  是相对于  $\mathbf{A}$  不变的子空间, 若

$$\mathbf{AX} = \mathbf{XB}, \quad \mathbf{A} \in \mathbb{C}^{n \times n}, \mathbf{B} \in \mathbb{C}^{k \times k}, \mathbf{X} \in \mathbb{C}^{n \times k} \quad (8.1.8)$$

此时, 若  $\mathbf{X}$  具有满列秩, 并且  $(\lambda, \mathbf{u})$  是矩阵  $\mathbf{B}$  的特征对, 即  $\mathbf{Bu} = \lambda\mathbf{u}$ , 则两边可以同时左乘满列秩矩阵  $\mathbf{X}$ , 从而有

$$\mathbf{Bu} = \lambda\mathbf{u} \implies \mathbf{XBu} = \lambda\mathbf{Xu} \implies \mathbf{A}(\mathbf{Xu}) = \lambda(\mathbf{Xu}) \quad (8.1.9)$$

即  $\lambda(\mathbf{B}) \subseteq \lambda(\mathbf{A})$ , 等号成立, 当且仅当  $\mathbf{X}$  为正方的非奇异矩阵时。也就是说, 若  $\mathbf{X}$  是非奇异矩阵, 则  $\mathbf{B} = \mathbf{X}^{-1}\mathbf{AX}$  是  $\mathbf{A}$  的相似矩阵, 并且  $\lambda(\mathbf{B}) = \lambda(\mathbf{A})$ 。这就从不变子空间的角度, 又一次证明了两个相似矩阵具有相同的特征值, 但它们的特征向量可能不同。

不变子空间的概念在利用子空间迭代跟踪和更新大的稀疏矩阵的特征值时, 起着重要的作用。

### 8.1.3 子空间的正交投影与夹角

关于子空间之间的几何关系, 我们会问: 沿着某个子空间, 到另一个子空间的投影如何描述? 两个子空间之间的距离和夹角又是如何定义的?

#### 1. 子空间的正交投影

令  $\mathbf{x} \in \mathbb{R}^n$ , 并且  $S$  和  $H$  是两个子空间。现在, 希望使用一线性矩阵变换  $\mathbf{P}$ , 将  $\mathbb{R}^n$  的向量  $\mathbf{x}$  映射为子空间  $S$  的向量  $\mathbf{x}_1$ 。这样一种线性变换称为沿着  $H$  的方向到  $S$  的投影算子 (projector onto  $S$  along  $H$ ), 常用符号  $\mathbf{P}_{S|H}$  表示。若子空间  $H$  是  $S$  的正交补, 则  $\mathbf{P}_{S|S^\perp}\mathbf{x}$  是将  $\mathbb{R}^n$  的向量  $\mathbf{x}$  沿着与子空间  $S$  垂直的方向, 到子空间  $S$  的投影, 故称  $\mathbf{P}_{S|S^\perp}\mathbf{x}$  为到子空间  $S$  的正交投影, 常用  $\mathbf{P}_S$  作数学符号。

**定义 8.1.5** [198, p.75] 矩阵  $\mathbf{P} \in \mathbb{C}^{n \times n}$  称为到子空间  $S$  的正交投影算子, 若  $\text{Range}(\mathbf{P}) = S$ ,  $\mathbf{P}^2 = \mathbf{P}$  和  $\mathbf{P}^H = \mathbf{P}$ 。

对上述定义的三个条件加以解读, 可以得到以下结果:

(1) 条件  $\text{Range}(\mathbf{P}) = S$  意味着  $\mathbf{P}$  的列空间必须等于子空间  $S$ 。若子空间  $S$  是矩阵  $\mathbf{A}_{m \times n}$  的  $n$  个列向量张成的子空间, 即  $S = \text{Span}(\mathbf{A})$ , 则  $\text{Range}(\mathbf{P}) = \text{Span}(\mathbf{A}) = \text{Range}(\mathbf{A})$ 。这意味着, 若将矩阵  $\mathbf{A}$  向子空间  $S$  作正交投影, 则其结果  $\mathbf{PA}$  必须等于原矩阵  $\mathbf{A}$ , 即有  $\mathbf{PA} = \mathbf{A}$ 。

(2) 条件  $\mathbf{P}^2 = \mathbf{P}$  意味着正交投影算子必须是幂等算子。

(3) 条件  $\mathbf{P}^H = \mathbf{P}$  表明, 正交投影算子必须具有复共轭对称性即 Hermitian 性。

应当注意的是，在有些文献中，一般定义具有 Hermitian 性的幂等算子为正交投影算子，是因为并没有强调它是到哪一个子空间的正交投影算子。当我们需要刻意强调是到子空间  $S$  的正交投影算子时，就必须加上  $\text{Range}(\mathbf{P}) = S$  这一条件。换言之，即使一线性算子满足幂等性和 Hermitian 性，但若其列空间与子空间  $S$  不一致，它便不是到子空间  $S$  的正交投影算子，而可能是到另外某个子空间的正交投影算子。

根据正交投影算子的定义知，若  $\mathbf{x} \in \mathbb{R}^n$ ，则有  $\mathbf{Px} \in S$  和  $(\mathbf{I} - \mathbf{P})\mathbf{x} \in S^\perp$ 。

假定  $\mathbf{P}_1$  和  $\mathbf{P}_2$  都是到子空间  $S$  的正交投影算子，则对于任意一个向量  $\mathbf{x} \in \mathbb{R}^n$ ，有下列结果

$$\begin{aligned}\|(\mathbf{P}_1 - \mathbf{P}_2)\mathbf{x}\|_2^2 &= (\mathbf{P}_1\mathbf{x} - \mathbf{P}_2\mathbf{x})^\text{H}(\mathbf{P}_1\mathbf{x} - \mathbf{P}_2\mathbf{x}) \\ &= (\mathbf{P}_1\mathbf{x})^\text{H}(\mathbf{I} - \mathbf{P}_2)\mathbf{x} + (\mathbf{P}_2\mathbf{x})^\text{H}(\mathbf{I} - \mathbf{P}_1)\mathbf{x} \\ &\equiv 0, \quad \forall \mathbf{x}\end{aligned}$$

这是因为  $\mathbf{P}_1\mathbf{x}$  和  $\mathbf{P}_2\mathbf{x}$  都是到子空间  $S$  的正交投影，从而有

$$\begin{aligned}\mathbf{y}_1 &= \mathbf{P}_1\mathbf{x} \in S, \quad \mathbf{z}_2 = (\mathbf{I} - \mathbf{P}_2)\mathbf{x} \in S^\perp \quad \Rightarrow \quad \mathbf{y}_1^\text{H}\mathbf{z}_2 = 0 \\ \mathbf{y}_2 &= \mathbf{P}_2\mathbf{x} \in S, \quad \mathbf{z}_1 = (\mathbf{I} - \mathbf{P}_1)\mathbf{x} \in S^\perp \quad \Rightarrow \quad \mathbf{y}_2^\text{H}\mathbf{z}_1 = 0\end{aligned}$$

由于  $\|(\mathbf{P}_1 - \mathbf{P}_2)\mathbf{x}\|_2^2 = 0$  对所有非零向量  $\mathbf{x}$  成立，故  $\mathbf{P}_1 = \mathbf{P}_2$ ，即到一个子空间的正交投影算子是唯一确定的。

对于子空间  $S = \text{Span}(\mathbf{A}_{m \times n})$ ，假定  $m \geq n$ ，并且  $\text{rank}(\mathbf{A}) = n$ 。观察知，线性变换矩阵

$$\mathbf{P}_S = \mathbf{A}(\mathbf{A}^\text{H}\mathbf{A})^{-1}\mathbf{A}^\text{H} \quad (8.1.10)$$

满足正交投影算子定义的幂等性和 Hermitian 性。另外，由于  $\mathbf{P}_S\mathbf{A} = \mathbf{A}$ ，即  $\mathbf{P}$  等价满足  $\text{Range}(\mathbf{P}_S) = \text{Span}(\mathbf{A}) = S$ 。因此，式 (8.1.10) 定义的线性变换算子  $\mathbf{P}_S$  是到由  $\mathbf{A}$  的列向量生成的子空间  $S$  上的正交投影算子。

如果子空间  $H$  与  $S$  不正交，则  $\mathbf{P}_{S|H}\mathbf{x}$  称为向量  $\mathbf{x}$  沿着子空间  $H$  的方向，到子空间  $S$  的斜投影，并称  $\mathbf{P}_{S|H}$  为斜投影算子。

关于正交投影算子和斜投影算子，将在第 9 章“投影分析”作专题讨论。

## 2. 子空间的夹角与距离

复向量空间  $\mathbb{C}^n$  内两个非零向量  $\mathbf{x}$  和  $\mathbf{y}$  之间的夹角记为  $\theta(\mathbf{x}, \mathbf{y})$ ，它们之间的锐角由

$$\cos \theta(\mathbf{x}, \mathbf{y}) = \frac{|\langle \mathbf{x}, \mathbf{y} \rangle|}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2}, \quad 0 \leq \theta(\mathbf{x}, \mathbf{y}) \leq \frac{\pi}{2} \quad (8.1.11)$$

定义。

向量  $\mathbf{x}$  与子空间  $S$  之间的锐角定义为  $\mathbf{x}$  与子空间  $S$  的所有向量  $\mathbf{y}$  之间的最小锐角，即

$$\theta(\mathbf{x}, S) = \min_{\mathbf{y} \in S} \theta(\mathbf{x}, \mathbf{y}) \quad (8.1.12)$$

正交投影算子的优化性能可以使用向量与子空间之间的锐角描述。

**定理 8.1.3** [434, p.63] 令  $P$  是到子空间  $S$  的正交投影算子，则对于复向量空间  $\mathbb{C}^n$  内的任意向量  $x$ ，有

$$\min_{y \in S} \|x - y\|_2 = \|x - Px\|_2 \quad (8.1.13)$$

或等价为

$$\theta(x, S) = \theta(x, Px) \quad (8.1.14)$$

定理 8.1.3 表明，复向量空间  $\mathbb{C}^n$  内任意一个向量  $x$  在向量空间  $S$  的最优逼近由投影  $P_S x$  决定，其他任何逼近形式都不可能比  $P_S x$  更接近  $x$ 。

**定义 8.1.6** [198, p.76] 假定  $S_1$  和  $S_2$  是  $\mathbb{C}^n$  的两个子空间，并且  $\dim(S_1) = \dim(S_2)$ ，则这两个子空间之间的距离定义为

$$\text{dist}(S_1, S_2) = \|P_{S_1} - P_{S_2}\|_F \quad (8.1.15)$$

式中， $P_{S_i}$  是到子空间  $S_i$ ,  $i = 1, 2$  的正交投影算子。

#### 8.1.4 主角与补角

当两个子空间的基向量不止一个时，子空间之间的夹角显然会有多个。此时，两个子空间之间的角度是直线与平面之间角度概念的推广。

给定  $n$  维 Hilbert 空间  $V$  的两个子空间  $H_1$  和  $H_2$ ，则两个子空间之间的夹角有多少个。这些角度的个数与两个子空间的最小维数相同。不妨令  $\dim(H_1) = p$ ,  $\dim(H_2) = q$ ，并且  $p > q$ ，则  $H_1$  和  $H_2$  之间的夹角共有  $q$  个。

**定义 8.1.7** [198] 子空间  $H_1$  与  $H_2$  之间的第  $i$  主角 (principal angle)  $\phi_i(H_1, H_2)$  是介于 0 和  $\pi/2$  之间的角度，定义为

$$\phi_i(H_1, H_2) = \arccos \left( \max_{u \in H_1} \max_{v \in H_2} u^H v \right) = \arccos (u_i^H v_i) \quad (8.1.16)$$

约束条件为

$$\left. \begin{aligned} u^H u &= v^H v = 1 \\ u^H u_j &= 0, \quad j = 1, 2, \dots, i-1 \\ v^H v_j &= 0, \quad j = 1, 2, \dots, i-1 \end{aligned} \right\} \quad (8.1.17)$$

式中， $u_i$  和  $v_i$  是  $\phi_i$  达到第  $i$  个最大值时的向量  $u$  和  $v$ 。

在这些主角之中，最小的主角称为最小角度 (minimum angle)。

**定义 8.1.8** [262] 子空间  $H_1$  与  $H_2$  之间的最小角度  $\phi(H_1, H_2)$  是介于 0 和  $\pi/2$  之间的角度，其余弦定义为

$$\cos \phi(H_1, H_2) \stackrel{\text{def}}{=} \max \{ |u^H v| : u \in H_1, v \in H_2, \|u\| = 1, \|v\| = 1 \} \quad (8.1.18)$$

显然，两个子空间之间的最小角度就是它们之间的第 1 个主角。

若将两个子空间的相交部分排除在外，即可得到与最小角度略有不同的角度定义。令  $H_{1:2} \stackrel{\text{def}}{=} H_1 \cap H_2$ 。

**定义 8.1.9** [262] 子空间  $H_2$  与  $H_1$  之间的补角 (complementary angle) 定义为

$$\phi_c(H_2, H_1) = \phi(H_2 \cap H_{1:2}^\perp, H_1 \cap H_{1:2}^\perp) \quad (8.1.19)$$

式中,  $H_{1:2}^\perp$  是  $H_{1:2}$  的正交补空间。

注意, 若两个子空间无交连, 即  $H_1 \cap H_2 = \{\mathbf{0}\}$ , 则这两个子空间之间的补角与最小角度相同, 即  $\phi_c(H_1, H_2) = \phi(H_1, H_2)$ 。

关于补角的取值, Lorch<sup>[321]</sup> 证明了以下结果。

**引理 8.1.1** 令  $H_1$  和  $H_2$  是 Hilbert 空间  $V$  的闭合子空间, 则

$$\phi_c(H_1, H_2) > 0 \quad (8.1.20)$$

当且仅当  $H_1 + H_2$  是闭合的。

### 8.1.5 子空间的旋转

在工程中经常会对同一对象进行多次测量, 并且每一次的测量数据并不完全相同。令  $\mathbf{A}$  和  $\mathbf{B}$  分别是两次测量得到的  $m \times n$  数据矩阵, 现在, 希望求一个  $n \times n$  实正交矩阵  $\mathbf{Q}$ , 在  $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$  的约束条件下, 使得

$$\min \|\mathbf{A} - \mathbf{BQ}\|_F \quad (8.1.21)$$

即通过正交矩阵  $\mathbf{Q}$ , 强迫  $\mathbf{BQ}$  与  $\mathbf{A}$  尽可能一致。

上述问题称为正交强迫一致问题 (orthogonal Procrustes problem), 它最早是 Green 于 1952 年在计量心理学杂志上提出的<sup>[204]</sup>。由于  $\mathbf{Q}$  是正交矩阵, 矩阵乘积  $\mathbf{BQ}$  并不改变  $\mathbf{B}$  的列向量之间的线性无关性, 所以列空间  $\text{Col}(\mathbf{BQ}) = \text{Col}(\mathbf{B})$ 。另外, 矩阵乘积  $\mathbf{BQ}$  相当于使矩阵  $\mathbf{B}$  旋转。因此, 从子空间的角度看问题, 正交强迫一致的运算相当于使列空间  $\text{Col}(\mathbf{B})$  旋转进入列空间  $\text{Col}(\mathbf{A})$  内。显然, 矩阵的 Frobenius 范数  $\|\mathbf{A} - \mathbf{BQ}\|_F$  起着度量正交强迫一致问题的解的质量的作用。

显而易见, 为了实现  $\|\mathbf{A} - \mathbf{BQ}\|_F^2$  的最小化, 应该选择正交矩阵  $\mathbf{Q}$  使得  $\mathbf{BQ}$  具有与  $\mathbf{A}$  完全相同的非对角元素, 并且对角元素的平方和尽可能接近。此时, 矩阵范数平方和  $\|\mathbf{A} - \mathbf{BQ}\|_F^2$  可以写成迹函数的形式

$$\|\mathbf{A} - \mathbf{BQ}\|_F^2 = \text{tr}(\mathbf{A}^T \mathbf{A}) + \text{tr}(\mathbf{B}^T \mathbf{B}) - 2\text{tr}(\mathbf{Q}^T \mathbf{B}^T \mathbf{A})$$

于是, 式 (8.1.21) 等价于使矩阵的迹  $\text{tr}(\mathbf{Q}^T \mathbf{B}^T \mathbf{A})$  最大化。

迹函数  $\text{tr}(\mathbf{Q}^T \mathbf{B}^T \mathbf{A})$  的最大化可以通过矩阵乘积  $\mathbf{B}^T \mathbf{A}$  的奇异值分解<sup>[198, p.582]</sup> 来实现。令矩阵  $\mathbf{B}^T \mathbf{A}$  的奇异值分解为  $\mathbf{B}^T \mathbf{A} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T$ , 式中,  $\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ 。若

定义正交矩阵  $Z = V^T Q^T U$ , 则有

$$\begin{aligned}\text{tr}(Q^T B^T A) &= \text{tr}(Q^T U \Sigma V^T) = \text{tr}(V^T Q^T U \Sigma) \\ &= \text{tr}(Z \Sigma) = \sum_{i=1}^n z_{ii} \sigma_i \\ &\leq \sum_{i=1}^n \sigma_i\end{aligned}$$

当且仅当  $Z = I$  即  $Q = UV^T$  时, 等号成立。换言之, 若选择  $Q = UV^T$ , 则  $\text{tr}(Q^T B^T A)$  取最大值, 从而使  $\|A - BQ\|_F$  取最小值。

以上分析表明, 若  $B^T A = U \Sigma V^T$  是矩阵乘积  $B^T A$  的奇异值分解, 则  $Q = UV^T$  就是正交强迫一致问题式 (8.1.21) 的解。

解矩阵  $Q$  称为矩阵乘积  $B^T A$  的正交极因子 (orthogonal polar factor)<sup>[198]</sup>, 因为正交强迫一致问题相当于将矩阵  $A$  分解为  $BQ$ , 而这种矩阵分解称为极式分解 (polar decomposition), 则因与复数的极坐标分解  $z = |z|e^{j\arg(z)}$  类似而得名。关于矩阵的极式分解及其应用, 读者可进一步参考文献 [49] 和 [230]。

有意思的是, 若  $B = I_n$ , 则正交强迫一致问题变为

$$Q = \min_{Q^T Q = I} \|A - Q\|_F$$

这一问题的数学描述是求与已知  $n \times n$  矩阵  $A$  最接近的正交矩阵, 根据前面的分析, 若  $A = U \Sigma V^T$  是  $A$  的奇异值分解, 则  $Q = UV^T$  是与矩阵  $A$  最接近的正交矩阵。

## 8.2 列空间、行空间与零空间

在对向量子空间进行分析之前, 有必要先了解与矩阵密切相关的基本空间: 列空间、行空间和零空间。

### 8.2.1 矩阵的列空间、行空间与零空间

为方便叙述, 对于矩阵  $A \in \mathbb{C}^{m \times n}$ , 其  $m$  个行向量记作

$$r_1 = [a_{11}, \dots, a_{1n}]$$

⋮

$$r_m = [a_{m1}, \dots, a_{mn}]$$

$n$  个列向量记作

$$a_1 = \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix}, \quad \dots, \quad a_n = \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix}$$

**定义 8.2.1** 若  $A = [a_1, a_2, \dots, a_n] \in \mathbb{C}^{m \times n}$  为复矩阵, 则其列向量的所有线性组合的集合构成一个子空间, 称为矩阵  $A$  的列空间 (column space) 或列张成 (column span), 用符号  $\text{Col}(A)$  表示, 即有

$$\text{Col}(A) = \text{Span}\{a_1, a_2, \dots, a_n\} \quad (8.2.1)$$

$$= \left\{ y \in \mathbb{C}^m \mid y = \sum_{j=1}^n \alpha_j a_j, \alpha_j \in \mathbb{C} \right\} \quad (8.2.2)$$

类似地, 矩阵  $A$  的复共轭行向量  $r_1^*, r_2^*, \dots, r_m^* \in \mathbb{C}^n$  的所有线性组合的集合称为矩阵  $A$  的行空间 (row space) 或行张成 (row span), 用符号  $\text{Row}(A)$  表示, 即有

$$\text{Row}(A) = \text{Span}\{r_1^*, r_2^*, \dots, r_m^*\} \quad (8.2.3)$$

$$= \left\{ y \in \mathbb{C}^n \mid y = \sum_{i=1}^m \beta_i r_i^*, \beta_i \in \mathbb{C} \right\} \quad (8.2.4)$$

在有些文献中, 常用符号  $\text{Span}\{A\}$  作为  $A$  的列空间的略写, 即

$$\text{Col}(A) = \text{Span}(A) = \text{Span}\{a_1, a_2, \dots, a_n\} \quad (8.2.5)$$

类似地, 符号  $\text{Span}(A^H)$  表示  $A$  的复共轭转置矩阵  $A^H$  的列空间。由于  $A^H$  的列向量就是矩阵  $A$  的复共轭行向量, 故

$$\text{Row}(A) = \text{Col}(A^H) = \text{Span}(A^H) = \text{Span}\{r_1^*, r_2^*, \dots, r_m^*\} \quad (8.2.6)$$

即复矩阵  $A$  的行空间与复共轭转置矩阵  $A^H$  的列空间等价。

将复矩阵  $A$  的行空间定义为其复共轭行向量  $r_1^*, r_2^*, \dots, r_m^*$  的所有线性组合的集合, 虽然在形式上比直接定义为  $r_1, r_2, \dots, r_m$  的所有线性组合的集合略显复杂, 但在利用矩阵的奇异值分解得到行空间时, 却会带来很大的方便, 详见 8.3 节。

行空间和列空间是直接针对矩阵  $A_{m \times n}$  本身定义的向量子空间。此外, 还有另外两个向量子空间不是直接用矩阵  $A$  定义, 而是通过矩阵变换  $Ax$  定义的。这两个子空间是映射或变换的值域和零空间。

在第 1 章中, 映射  $T$  的值域定义为  $T(x) \neq \mathbf{0}$  的所有值的集合, 而映射  $T$  的核或零空间则定义为满足  $T(x) = \mathbf{0}$  的所有非零解向量  $x$  的集合。很自然地, 若线性映射  $y = T(x)$  是从  $\mathbb{C}^n$  空间到  $\mathbb{C}^m$  空间的矩阵变换, 即  $y_{m \times 1} = A_{m \times n}x_{n \times 1}$ , 则对于一个给定的矩阵  $A$ , 矩阵变换  $Ax$  的值域定义为向量  $y = Ax$  的所有值的集合; 而零空间则定义为满足  $Ax = \mathbf{0}$  的向量  $x$  的集合。在一些文献 (特别是工程文献) 中, 常将矩阵变换  $Ax$  的值域和零空间分别直接当作矩阵  $A$  的值域和零空间, 即有以下定义。

**定义 8.2.2** 若  $A$  是一个  $m \times n$  复矩阵, 则  $A$  的值域 (range) 定义为

$$\text{Range}(A) = \{y \in \mathbb{C}^m \mid Ax = y, x \in \mathbb{C}^n\} \quad (8.2.7)$$

矩阵  $\mathbf{A}$  的零空间 (null space) 也称  $\mathbf{A}$  的核 (kernel), 定义为满足齐次线性方程  $\mathbf{Ax} = \mathbf{0}$  的所有解向量的集合, 即

$$\text{Null}(\mathbf{A}) = \text{Ker}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{C}^n | \mathbf{Ax} = \mathbf{0}\} \quad (8.2.8)$$

类似地, 复矩阵  $\mathbf{A}_{m \times n}$  的共轭转置  $\mathbf{A}^H$  的零空间定义为

$$\text{Null}(\mathbf{A}^H) = \text{Ker}(\mathbf{A}^H) = \{\mathbf{x} \in \mathbb{C}^m | \mathbf{A}^H \mathbf{x} = \mathbf{0}\} \quad (8.2.9)$$

零空间的维数称为  $\mathbf{A}$  的零化维 (nullity), 即有

$$\text{nullity}(\mathbf{A}) = \dim[\text{Null}(\mathbf{A})] \quad (8.2.10)$$

若  $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$  是  $\mathbf{A}$  的列分块, 不妨令  $\mathbf{x} = [\alpha_1, \alpha_2, \dots, \alpha_n]^T$ , 则  $\mathbf{Ax} = \sum_{j=1}^n \alpha_j \mathbf{a}_j$ , 故立即有

$$\text{Range}(\mathbf{A}) = \left\{ \mathbf{y} \in \mathbb{C}^m \mid \mathbf{y} = \sum_{j=1}^n \alpha_j \mathbf{a}_j, \alpha_j \in \mathbb{C} \right\} = \text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$$

表明矩阵  $\mathbf{A}$  的值域就是  $\mathbf{A}$  的列空间, 即有

$$\text{Range}(\mathbf{A}) = \text{Col}(\mathbf{A}) = \text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\} \quad (8.2.11)$$

类似地, 有

$$\text{Range}(\mathbf{A}^H) = \text{Col}(\mathbf{A}^H) = \text{Span}\{\mathbf{r}_1^*, \mathbf{r}_2^*, \dots, \mathbf{r}_m^*\} \quad (8.2.12)$$

**定理 8.2.1** 若  $\mathbf{A}$  是  $m \times n$  复矩阵, 则  $\mathbf{A}$  的行空间的正交补  $(\text{Row}(\mathbf{A}))^\perp$  是  $\mathbf{A}$  的零空间, 并且  $\mathbf{A}$  的列空间的正交补  $(\text{Col}(\mathbf{A}))^\perp$  是  $\mathbf{A}^H$  的零空间, 即有

$$(\text{Row}(\mathbf{A}))^\perp = \text{Null}(\mathbf{A}), \quad (\text{Col}(\mathbf{A}))^\perp = \text{Null}(\mathbf{A}^H) \quad (8.2.13)$$

**证明** 令矩阵  $\mathbf{A}$  的行向量为  $\mathbf{r}_i, i = 1, \dots, m$ 。由矩阵的乘法规则知, 若  $\mathbf{x}$  位于零空间  $\text{Null}(\mathbf{A})$ , 即满足  $\mathbf{Ax} = \mathbf{0}$ , 则  $\mathbf{r}_i \mathbf{x} = 0$ 。写成两个列向量正交的标准形式, 为  $(\mathbf{r}_i^H)^H \mathbf{x} = 0, i = 1, \dots, m$ 。这意味着, 向量  $\mathbf{x}$  与  $\mathbf{r}_1^H, \dots, \mathbf{r}_m^H$  的线性张成正交, 即

$$\mathbf{x} \perp \text{Span}\{\mathbf{r}_1^H, \dots, \mathbf{r}_m^H\} = \text{Col}(\mathbf{A}^H) = \text{Row}(\mathbf{A})$$

从而  $\mathbf{x}$  的所有集合即  $\text{Null}(\mathbf{A})$  与  $\text{Row}(\mathbf{A})$  正交。反之, 若  $\mathbf{x}$  与  $\text{Row}(\mathbf{A}) = \text{Col}(\mathbf{A}^H)$  正交, 则有  $\mathbf{x}$  与  $\mathbf{r}_i^H, i = 1, \dots, m$  正交, 即  $(\mathbf{r}_i^H)^H \mathbf{x} = 0$ , 等价为  $\mathbf{r}_i \mathbf{x} = 0, i = 1, \dots, m$  或  $\mathbf{Ax} = \mathbf{0}$ 。这表明, 与  $\text{Row}(\mathbf{A})$  正交的向量  $\mathbf{x}$  的集合是矩阵  $\mathbf{A}$  的零空间。因此, 有  $(\text{Row}(\mathbf{A}))^\perp = \text{Null}(\mathbf{A})$ 。

在  $(\text{Row}(\mathbf{B}_{n \times m}))^\perp = \text{Null}(\mathbf{B})$  中令  $\mathbf{B} = \mathbf{A}_{m \times n}^H$ , 立即有

$$(\text{Row}(\mathbf{A}^H))^\perp = \text{Null}(\mathbf{A}^H) \implies (\text{Col}(\mathbf{A}))^\perp = \text{Null}(\mathbf{A}^H)$$

这就完成了本定理的证明。 ■

总结以上讨论，即可得到与矩阵  $A$  的向量子空间之间的关系：

(1) 矩阵  $A$  的值域与列空间相等，即

$$\text{Range}(A) = \text{Col}(A) = \text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$$

(2) 矩阵  $A$  的行空间与  $A^H$  的列空间相等，即

$$\text{Row}(A) = \text{Col}(A^H) = \text{Range}(A^H)$$

(3) 矩阵  $A$  的行空间的正交补等于  $A$  的零空间，即

$$(\text{Row}(A))^{\perp} = \text{Null}(A)$$

(4) 矩阵  $A$  的列空间的正交补就是  $A^H$  的零空间，即

$$(\text{Col}(A))^{\perp} = \text{Null}(A^H)$$

既然矩阵  $A$  的列空间  $\text{Col}(A)$  是其列向量的所有线性组合的集合，那么列空间  $\text{Col}(A)$  便只由那些线性无关的列向量  $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \dots, \mathbf{a}_{i_k}$  决定，而与这些列向量线性相关的其他列向量对于列空间的生成则是多余的。

子集合  $\{\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_k}\}$  是列向量集合  $\{\mathbf{a}_1, \dots, \mathbf{a}_n\}$  的最大线性无关子集合 (maximal linearly independent subset)，若  $\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_k}$  线性无关，并且这些线性无关的列向量不包含在  $\{\mathbf{a}_1, \dots, \mathbf{a}_n\}$  的任何其他线性无关的子集合中。

若  $\{\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_k}\}$  是最大线性无关子集合，则

$$\text{Span}\{\mathbf{a}_1, \dots, \mathbf{a}_n\} = \text{Span}\{\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_k}\} \quad (8.2.14)$$

并称最大线性无关子集合  $\{\mathbf{a}_1, \dots, \mathbf{a}_n\}$  是矩阵  $A$  的列空间  $\text{Col}(A)$  的基。显然，对于一个给定的矩阵  $A_{m \times n}$ ，它的基可以有不同的组合形式，但所有基形式都必须包含相同的向量 (基向量) 个数。这个共同的向量个数称为矩阵  $A$  的列空间  $\text{Col}(A)$  的维数，用符号  $\dim[\text{Col}(A)]$  表示。又由于矩阵  $A_{m \times n}$  的秩定义为线性无关的列向量个数，故矩阵  $A$  的秩与列空间  $\text{Col}(A)$  的维数是一致的，即也可以将秩定义为

$$\text{rank}(A) = \dim[\text{Col}(A)] = \dim[\text{Range}(A)] \quad (8.2.15)$$

一个自然的问题是：矩阵的列空间和零空间之间有什么样的联系？事实上，这两个子空间存在很大的不同，详见表 8.2.1。

### 8.2.2 子空间的基构造：初等变换法

如上所述，矩阵  $A_{m \times n}$  的列空间和行空间分别由  $A$  的  $n$  个列向量和  $m$  个行向量张成。但是，如果矩阵的秩  $r = \text{rank}(A)$ ，则只需要矩阵  $A$  的  $r$  个线性无关列向量或行向

表8.2.1  $m \times n$  矩阵  $A$  的零空间与列空间的对比 [306, p.226]

零空间 $\text{Null}(A)$	列空间 $\text{Col}(A)$
$\text{Null}(A)$ 是 $\mathbb{C}^m$ 的子空间	$\text{Col}(A)$ 是 $\mathbb{C}^n$ 的子空间
$\text{Null}(A)$ 为隐含定义, !j $A$ 的列向量无直接关系	$\text{Col}(A)$ 为显式定义, 直接由 $A$ 的所有列向量张成
$\text{Null}(A)$ 的基应满足 $Ax = 0$	$\text{Col}(A)$ 的基是 $A$ 的主元列
$\text{Null}(A)$ 与矩阵 $A$ 的元素无任何明显关系	矩阵 $A$ 的每一列都在 $\text{Col}(A)$ 内
$\text{Null}(A)$ 的典型向量 $v$ 满足 $Av = 0$	$\text{Col}(A)$ 的典型向量满足 $Ax = v$ 为一致方程
$v \in \text{Null}(A)$ 的条件: $Av = 0$	$v \in \text{Col}(A)$ 的条件: $[A, v] !j A$ 具有相同的秩
$\text{Null}(A) = \{0\}$ 当且仅当 $Ax = 0$ 只有零解	$\text{Col}(A) = \{0\}$ 当且仅当 $Ax = b$ 有解
$\text{Null}(A) = \{0\}$ 当且仅当 $Ax$ 为一对一映射	$\text{Col}(A) = \{0\}$ 当且仅当 $Ax$ 为 $\mathbb{C}^n$ 到 $\mathbb{C}^m$ 的映射

量(即基), 即可分别生成列空间  $\text{Span}(A)$  和行空间  $\text{Span}(A^H)$ 。显然, 使用基向量是一种更加经济和更好的子空间表示法。那么, 如何寻找所需要的基向量呢? 下面讨论矩阵  $A$  的行空间  $\text{Row}(A)$ 、列空间  $\text{Col}(A)$  以及零空间  $\text{Null}(A)$  和  $\text{Null}(A^H)$  的基的构造。

容易证明下面的结果 [306]:

- (1) 初等行变换不改变矩阵  $A$  的行空间  $\text{Row}(A)$  和零空间  $\text{Null}(A)$ 。
- (2) 初等列变换不改变矩阵  $A$  的列空间  $\text{Col}(A)$  和矩阵  $A^H$  的零空间  $\text{Null}(A^H)$ 。

下面的定理给出了利用矩阵的初等行变换或者初等列变换构造所需要的子空间的方法。

**定理 8.2.2** [306] 令矩阵  $A_{m \times n}$  经过初等行变换后, 变成阶梯型矩阵  $B$ , 则

- (1) 阶梯型矩阵  $B$  的非零行组成矩阵  $A$  和  $B$  的行空间的一组基;
- (2) 矩阵  $A$  的主元列组成列空间  $\text{Col}(A)$  的一组基。

总结以上讨论, 可得到构造矩阵的行空间和列空间的基向量的初等变换法如下:

**初等行变换法** 令矩阵  $A$  经过初等行变换, 变为简约阶梯型矩阵  $B_r$ , 则

- (1) 简约阶梯型  $B_r$  所有主元位置所在的非零行构成行空间  $\text{Row}(A)$  的基;
- (2) 矩阵  $A$  的主元列组成列空间  $\text{Col}(A)$  的基;
- (3) 矩阵  $A$  的非主元列组成零空间  $\text{Null}(A^H)$  的基。

**初等列变换法** 令矩阵  $A$  经过初等列变换, 变为列形式的简约阶梯型矩阵  $B_c$ , 则

- (1) 列形式的阶梯型矩阵  $B_c$  所有主元位置所在的非零列构成列空间  $\text{Col}(A)$  的基;
- (2) 矩阵  $A$  的主元行组成行空间  $\text{Row}(A)$  的基;
- (3) 矩阵  $A$  的非主元行组成零空间  $\text{Null}(A)$  的基。

下面举例加以说明。

**例 8.2.1** 求  $3 \times 3$  矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 1 \\ -1 & -1 & 1 \\ 1 & 4 & 5 \end{bmatrix}$$

的行空间与列空间。

**解法 1** 依次进行初等列变换:  $C_2 - 2C_1$  (第 1 列乘  $-2$ , 与第 2 列相加),  $C_3 - C_1, C_1 + C_2, C_3 - 2C_2$ , 变换结果为

$$\mathbf{B}_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 3 & 2 & 0 \end{bmatrix}$$

由此得到两个线性无关的列向量  $\mathbf{c}_1 = [1, 0, 3]^T, \mathbf{c}_2 = [0, 1, 2]^T$ , 它们就是列空间  $\text{Col}(\mathbf{A})$  的基, 即

$$\text{Col}(\mathbf{A}) = \text{Span} \left\{ \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \right\}$$

根据列简约阶梯型矩阵  $\mathbf{B}$  的主元位置, 矩阵  $\mathbf{A}$  的主元行是第 1 行和第 2 行, 即行空间  $\text{Row}(\mathbf{A})$  可以写作

$$\text{Row}(\mathbf{A}) = \text{Span}\{[1, 2, 1], [-1, -1, 1]\}$$

**解法 2** 依次作初等行变换:  $R_2 + R_1$  (第 1 行加到第 2 行),  $R_3 - R_1, R_3 - 2R_2$ , 则变换结果为

$$\mathbf{B}_r = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

得到两个线性无关的行向量  $\mathbf{r}_1 = [1, 2, 1], \mathbf{r}_2 = [0, 1, 2]$ , 它们组成行空间  $\text{Row}(\mathbf{A})$  的基向量, 即

$$\text{Row}(\mathbf{A}) = \text{Span}\{[1, 2, 1], [0, 1, 2]\}$$

而矩阵  $\mathbf{A}$  的主元列为第 1 列和第 2 列, 它们组成列空间  $\text{Col}(\mathbf{A})$  的基, 即

$$\text{Col}(\mathbf{A}) = \text{Span} \left\{ \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ -1 \\ 4 \end{bmatrix} \right\}$$

事实上, 两种解法的结果等价, 因为对解法 2 求得的列空间的基作初等列变换, 有

$$\begin{bmatrix} 1 & 2 \\ -1 & -1 \\ 1 & 4 \end{bmatrix} \xrightarrow{-C_1+C_2} \begin{bmatrix} 1 & 1 \\ -1 & 0 \\ 1 & 3 \end{bmatrix} \xrightarrow{C_2-C_1} \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 2 & 3 \end{bmatrix} \xrightarrow{C_1 \leftrightarrow C_2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 3 & 2 \end{bmatrix}$$

与解法 1 的列空间基向量结果相同。类似地, 可以证明, 解法 1 和解法 2 得到的行空间的基向量也等价。

由于初等行变换与初等列变换得到的行空间与列空间的基向量等价, 故任意选择一种初等变换均可。习惯上使用初等行变换。不过, 若矩阵的列数明显少于行数时, 初等列变换需要较少的次数。

下面的定理描述了一个  $m \times n$  矩阵的秩与其零空间维数之间的关系，称为秩定理 (rank theorem)。

**定理 8.2.3** 矩阵  $A_{m \times n}$  的列空间与行空间的维数相等。这个共同的维数就是矩阵  $A$  的秩  $\text{rank}(A)$ ，它与零空间维数之间有下列关系

$$\text{rank}(A) + \dim[\text{Null}(A)] = n \quad (8.2.16)$$

**证明** 根据矩阵秩的定义式 (8.2.15) 知， $\text{rank}(A)$  就是矩阵  $A$  中线性无关列 (即主元列) 的个数。即是说， $\text{rank}(A)$  是经过初等行变换得到的阶梯型矩阵  $B$  的主元的个数。由于在每一个主元位置，阶梯型矩阵  $B$  的行都是线性无关的非零行，并且这些行构成矩阵  $A$  的行空间，所以矩阵的秩  $\text{rank}(A)$  也是行空间  $\text{Row}(A)$  的维数。由定理 8.2.1 知

$$\begin{aligned} \text{rank}(A) + \dim[\text{Null}(A)] &= \text{rank}(A) + \dim[(\text{Row}(A))^\perp] \\ &= \dim[\text{Row}(A)] + \dim[(\text{Row}(A))^\perp] \end{aligned}$$

本定理成立。 ■

下面的定理表明，矩阵的 QR 分解也可以用于构造列空间的基向量。

**定理 8.2.4** 若  $A = QR$  是一个满列秩矩阵  $A \in \mathbb{R}^{m \times n}$  的 QR 分解，并且  $A = [a_1, \dots, a_n]$  和  $Q = [q_1, \dots, q_m]$  是列分块的，则

$$\text{Span}\{a_1, \dots, a_k\} = \text{Span}\{q_1, \dots, q_k\}, \quad k = 1, \dots, n$$

特别地，若  $Q = [Q_1, Q_2]$ ，其中， $Q_1$  是  $Q$  的前  $n$  列组成的分块， $Q_2$  是  $Q$  的其他列组成的分块，则

$$\text{Range}(A) = \text{Range}(Q_1), \quad (\text{Range}(A))^\perp = \text{Range}(Q_2)$$

并且  $A = Q_1 R_1$ ， $R_1 = R(1:n, 1:n)$ ，即  $R_1$  是  $R$  的左上方  $n \times n$  方块。

**证明** 比较  $A = QR$  左右两边的第  $k$  列，可以得出结论

$$a_k = \sum_{i=1}^k r_{ik} q_i \in \text{Span}\{q_1, \dots, q_k\}$$

上式表明， $\text{Span}\{a_1, \dots, a_k\} \subseteq \text{Span}\{q_1, \dots, q_k\}$ 。然而，由于  $\text{rank}(A) = n$ ，故  $\text{Span}\{a_1, \dots, a_k\}$  具有维数  $k$ ，从而有  $\text{Span}\{a_1, \dots, a_k\} = \text{Span}\{q_1, \dots, q_k\}$ 。定理的剩余部分可以直接得出。 ■

### 8.2.3 基本空间的标准正交基构造：奇异值分解法

初等变换法得到的只是线性无关的基向量。然而，在很多应用中，希望获得已知矩阵的列空间、行空间和零空间的正交基。对线性无关的基向量，使用 Gram-Schmidt 正交化，可以实现这些要求。但是，更方便的方法是利用矩阵的奇异值分解。

令秩  $\text{rank}(\mathbf{A}) = r$  的矩阵  $\mathbf{A}_{m \times n}$  具有以下奇异值分解

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^H \quad (8.2.17)$$

式中

$$\mathbf{U} = [\mathbf{U}_r, \tilde{\mathbf{U}}_r], \quad \mathbf{V} = [\mathbf{V}_r, \tilde{\mathbf{V}}_r], \quad \Sigma = \begin{bmatrix} \Sigma_r & \mathbf{O}_{r \times (n-r)} \\ \mathbf{O}_{(m-r) \times (n-r)} & \mathbf{O}_{(n-r) \times (n-r)} \end{bmatrix}$$

这里,  $\mathbf{U}_r$  和  $\tilde{\mathbf{U}}_r$  分别为  $m \times r$  和  $m \times (m-r)$  矩阵,  $\mathbf{V}_r$  和  $\tilde{\mathbf{V}}_r$  分别为  $n \times r$  和  $n \times (n-r)$  矩阵, 并且  $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$ 。

显然, 矩阵  $\mathbf{A}$  的奇异值分解可简化为

$$\mathbf{A} = \mathbf{U}_r \Sigma_r \mathbf{V}_r^H = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^H \quad (8.2.18)$$

$$\mathbf{A}^H = \mathbf{V}_r \Sigma_r \mathbf{U}_r^H = \sum_{i=1}^r \sigma_i \mathbf{v}_i \mathbf{u}_i^H \quad (8.2.19)$$

下面分别讨论列空间、行空间和零空间的标准正交基的构造。

### 1. 列空间的标准正交基构造

将式 (8.2.18) 代入值域  $\text{Range}(\mathbf{A})$  的定义式, 易得

$$\begin{aligned} \text{Range}(\mathbf{A}) &= \{\mathbf{y} \in \mathbb{C}^m : \mathbf{y} = \mathbf{Ax}, \mathbf{x} \in \mathbb{C}^n\} \\ &= \left\{ \mathbf{y} \in \mathbb{C}^m : \mathbf{y} = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^H \mathbf{x}, \mathbf{x} \in \mathbb{C}^n \right\} \\ &= \left\{ \mathbf{y} \in \mathbb{C}^m : \mathbf{y} = \sum_{i=1}^r \mathbf{u}_i (\sigma_i \mathbf{v}_i^H \mathbf{x}), \mathbf{x} \in \mathbb{C}^n \right\} \\ &= \left\{ \mathbf{y} \in \mathbb{C}^m : \mathbf{y} = \sum_{i=1}^r \alpha_i \mathbf{u}_i, \alpha_i = \sigma_i \mathbf{v}_i^H \mathbf{x} \in \mathbb{C} \right\} \\ &= \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_r\} \end{aligned}$$

利用值域与列空间的等价关系, 即有

$$\text{Col}(\mathbf{A}) = \text{Range}(\mathbf{A}) = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_r\}$$

这表明, 与  $r$  个非零奇异值对应的左奇异向量  $\mathbf{u}_1, \dots, \mathbf{u}_r$  构成列空间  $\text{Col}(\mathbf{A})$  的一组基。

### 2. 行空间的标准正交基构造

计算复共轭转置矩阵  $\mathbf{A}^H$  的值域, 得

$$\begin{aligned}\text{Range}(\mathbf{A}^H) &= \{\mathbf{y} \in \mathbb{C}^n : \mathbf{y} = \mathbf{A}^H \mathbf{x}, \mathbf{x} \in \mathbb{C}^m\} \\ &= \left\{ \mathbf{y} \in \mathbb{C}^n : \mathbf{y} = \sum_{i=1}^r \sigma_i \mathbf{v}_i \mathbf{u}_i^H \mathbf{x}, \mathbf{x} \in \mathbb{C}^m \right\} \\ &= \left\{ \mathbf{y} \in \mathbb{C}^n : \mathbf{y} = \sum_{i=1}^r \alpha_i \mathbf{v}_i, \quad \alpha_i = \sigma_i \mathbf{u}_i^H \mathbf{x} \in C \right\} \\ &= \text{Span}\{\mathbf{v}_1, \dots, \mathbf{v}_r\}\end{aligned}$$

从而有

$$\text{Row}(\mathbf{A}) = \text{Range}(\mathbf{A}^H) = \text{Span}\{\mathbf{v}_1, \dots, \mathbf{v}_r\}$$

即与  $r$  个非零奇异值对应的右奇异向量  $\mathbf{v}_1, \dots, \mathbf{v}_r$  是行空间  $\text{Row}(\mathbf{A})$  的一组基。

### 3. 零空间的标准正交基构造

由于假定矩阵的秩为  $r$ , 故零空间  $\text{Null}(\mathbf{A})$  的维数等于  $n - r$ 。因此, 我们需要寻找  $n - r$  个线性无关的标准正交向量作为零空间的标准正交基。为此, 考虑满足  $\mathbf{A}\mathbf{x} = \mathbf{0}$  的向量。由奇异向量的性质得  $\mathbf{v}_i^H \mathbf{v}_j = 0, \forall i = 1, \dots, r, j = r + 1, \dots, n$ 。由此知

$$\mathbf{A}\mathbf{v}_j = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^H \mathbf{v}_j = \mathbf{0}, \quad \forall j = r + 1, r + 2, \dots, n$$

由于与零奇异值对应的  $n - r$  个右奇异向量  $\mathbf{v}_{r+1}, \dots, \mathbf{v}_n$  线性无关, 并且满足  $\mathbf{A}\mathbf{x} = \mathbf{0}$  的条件, 故它们组成了零空间  $\text{Null}(\mathbf{A})$  的基, 即有

$$\text{Null}(\mathbf{A}) = \text{Span}\{\mathbf{v}_{r+1}, \mathbf{v}_{r+2}, \dots, \mathbf{v}_n\}$$

类似地, 有

$$\mathbf{A}^H \mathbf{u}_j = \sum_{i=1}^r \sigma_i \mathbf{v}_i \mathbf{u}_i^H \mathbf{u}_j = \mathbf{0}, \quad \forall j = r + 1, r + 2, \dots, m$$

由于  $m - r$  个右奇异向量  $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$  线性无关, 并且满足  $\mathbf{A}^H \mathbf{x} = \mathbf{0}$  的条件, 故它们组成了零空间  $\text{Null}(\mathbf{A}^H)$  的基, 即有

$$\text{Null}(\mathbf{A}^H) = \text{Span}\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_m\}$$

由于矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$  的左奇异向量矩阵  $\mathbf{U}$  和右奇异向量矩阵  $\mathbf{V}$  为酉矩阵, 所以上述方法实际上分别提供了  $\mathbf{A}$  的列空间、行空间和零空间的标准正交基。总结以上讨论, 对于秩为  $r$  的复矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$ , 有以下结论:

(1) 与非零奇异值对应的  $r$  个左奇异向量  $\mathbf{u}_1, \dots, \mathbf{u}_r$  是列空间  $\text{Col}(\mathbf{A})$  的标准正交基, 即有

$$\text{Col}(\mathbf{A}) = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_r\} \quad (8.2.20)$$

(2) 与零奇异值对应的  $m - r$  个左奇异向量  $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$  是零空间  $\text{Null}(\mathbf{A}^H)$  的标准正交基, 即

$$\text{Null}(\mathbf{A}^H) = (\text{Col}(\mathbf{A}))^\perp = \text{Span}\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_m\} \quad (8.2.21)$$

(3) 与非零奇异值对应的  $r$  个右奇异向量  $\mathbf{v}_1, \dots, \mathbf{v}_r$  是行空间  $\text{Row}(\mathbf{A})$  的标准正交基, 即

$$\text{Row}(\mathbf{A}) = \text{Span}\{\mathbf{v}_1, \dots, \mathbf{v}_r\} \quad (8.2.22)$$

(4) 与零奇异值对应的  $n - r$  个右奇异向量  $\mathbf{v}_{r+1}, \dots, \mathbf{v}_n$  是零空间  $\text{Null}(\mathbf{A})$  的标准正交基, 即

$$\text{Null}(\mathbf{A}) = (\text{Row}(\mathbf{A}))^\perp = \text{Span}\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_n\} \quad (8.2.23)$$

令  $\mathbf{U}_H$  是与矩阵  $\mathbf{H}$  的  $p$  个非零奇异值对应的左奇异向量组成的矩阵。类似地,  $\mathbf{U}_S$  是与矩阵  $\mathbf{S}$  的  $q$  个非零奇异值对应的左奇异向量组成的矩阵, 其中, 假设  $p > q$ 。Golub 与 Van Loan<sup>[198]</sup> 证明了, 所有主角都可以利用奇异值分解计算: 由于  $\text{Span}(\mathbf{U}_H) = \text{Range}(\mathbf{H})$  和  $\text{Span}(\mathbf{U}_S) = \text{Range}(\mathbf{S})$ , 故子空间  $\text{Range}(\mathbf{H})$  和  $\text{Range}(\mathbf{S})$  之间的第  $i$  个主角由

$$\phi_i = \arccos \lambda_i, \quad i = 1, 2, \dots, q \quad (8.2.24)$$

给出, 式中,  $\lambda_i$  是乘积矩阵  $\mathbf{U}_H^H \mathbf{U}_S$  的第  $i$  个奇异值。

QR 分解是构造矩阵  $\mathbf{A}$  的列空间的正交基的另外一种方法<sup>[508]</sup>。令  $n \times n$  矩阵  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  非奇异, 并令  $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_n]$ 。由  $\mathbf{a}_1 = r_{11}\mathbf{q}_1, \mathbf{a}_2 = r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2$  可以写出一般形式

$$\mathbf{a}_k = r_{1k}\mathbf{q}_1 + r_{2k}\mathbf{q}_2 + \dots + r_{kk}\mathbf{q}_k, \quad k = 1, \dots, n$$

由此得  $\text{Span}\{\mathbf{a}_1\} = \text{Span}\{\mathbf{q}_1\}$ ,  $\text{Span}\{\mathbf{a}_1, \mathbf{a}_2\} = \text{Span}\{\mathbf{q}_1, \mathbf{q}_2\}$  以及一般形式

$$\text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_k\} = \text{Span}\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}, \quad k = 1, 2, \dots, n$$

最后有  $\text{Col}(\mathbf{A}) = \text{Col}(\mathbf{Q})$ 。换言之,酉矩阵  $\mathbf{Q}$  的列向量是矩阵  $\mathbf{A}$  的列空间的一组标准正交基。

### 8.2.4 构造两个零空间交的标准正交基

上面介绍了使用矩阵奇异值分解, 构造单个零空间  $\text{Null}(\mathbf{A})$  的标准正交基的方法。现在考虑对给定的两个矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$  和  $\mathbf{B} \in \mathbb{C}^{p \times n}$ , 如何构造零空间的交  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$  的标准正交基。

显然, 若令

$$\mathbf{C} = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \in \mathbb{C}^{(m+p) \times n}$$

则

$$\mathbf{C}\mathbf{x} = \mathbf{0} \iff \mathbf{Ax} = \mathbf{0} \text{ 和 } \mathbf{Bx} = \mathbf{0}$$

即  $\mathbf{C}$  的零空间等于  $\mathbf{A}$  的零空间与  $\mathbf{B}$  的零空间的交

$$\text{Null}(\mathbf{C}) = \text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$$

这表明, 若  $(m+p) \times n$  矩阵  $\mathbf{C}$  的秩为  $r = \text{rank}(\mathbf{C})$ , 则它的右奇异向量  $\mathbf{v}_1, \dots, \mathbf{v}_n$  中, 与  $n-r$  个零奇异值对应的右奇异向量  $\mathbf{v}_{r+1}, \dots, \mathbf{v}_n$  构成零空间的交  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$  的标准正交基。但是, 这涉及  $(m+p) \times n$  矩阵  $\mathbf{C}$  的奇异值分解。

**定理 8.2.5** [198,p.583] 令  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , 并且  $\{\mathbf{z}_1, \dots, \mathbf{z}_t\}$  是零空间  $\text{Null}(\mathbf{A})$  的一组正交基。记  $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_t]$ , 并定义  $\{\mathbf{w}_1, \dots, \mathbf{w}_q\}$  是零空间  $\text{Null}(\mathbf{BZ})$  的一组正交基, 其中,  $\mathbf{B} \in \mathbb{R}^{p \times n}$ 。若  $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_q]$ , 则  $\mathbf{ZW}$  的列向量构成零空间的交  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$  的一组正交基。

给定矩阵  $\mathbf{A}_{m \times n}$ ,  $\mathbf{B}_{p \times n}$ , 上述定理给出了构造  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$  的正交基的方法:

(1) 计算矩阵  $\mathbf{A}$  的奇异值分解  $\mathbf{A} = \mathbf{U}_A \boldsymbol{\Sigma}_A \mathbf{V}_A^T$ , 判断矩阵  $\mathbf{A}$  的有效秩  $r$ , 进而得到零空间  $\text{Null}(\mathbf{A})$  的正交基  $\mathbf{v}_{r+1}, \dots, \mathbf{v}_n$ , 其中  $\mathbf{v}_i$  是矩阵  $\mathbf{A}$  的右奇异向量。令  $\mathbf{Z} = [\mathbf{v}_{r+1}, \dots, \mathbf{v}_n]$ 。

(2) 计算矩阵  $\mathbf{C}_{p \times (n-r)} = \mathbf{BZ}$  和它的奇异值分解  $\mathbf{C} = \mathbf{U}_C \boldsymbol{\Sigma}_C \mathbf{V}_C^T$ , 判断其有效秩  $q$ , 进而得到零空间  $\text{Null}(\mathbf{BZ})$  的正交基  $\mathbf{w}_{q+1}, \dots, \mathbf{w}_{n-r}$ , 其中,  $\mathbf{w}_i$  是矩阵  $\mathbf{C} = \mathbf{BZ}$  的右奇异向量。令  $\mathbf{W} = [\mathbf{w}_{q+1}, \dots, \mathbf{w}_{n-r}]$ 。

(3) 计算矩阵  $\mathbf{ZW}$ , 其列向量即为零空间的交  $\text{Null}(\mathbf{A}) \cap \text{Null}(\mathbf{B})$  的正交基 (由于  $\mathbf{Z}$  和  $\mathbf{W}$  分别是矩阵  $\mathbf{A}$  和  $\mathbf{BZ}$  的右奇异向量组成的矩阵, 故  $\mathbf{ZW}$  具有正交性)。

### 8.3 子空间方法

前面介绍了矩阵的列空间、行空间与零空间。本节讨论子空间分析方法及其在工程和信号处理中的应用。由于在工程应用中, 多数情况下使用列空间, 因此本章今后将以矩阵的列空间作为主要讨论对象。

观测数据矩阵  $\mathbf{A}$  不可避免地存在观测误差或噪声。令

$$\mathbf{X} = \mathbf{A} + \mathbf{W} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{C}^{m \times n} \quad (8.3.1)$$

为观测数据矩阵, 其中,  $\mathbf{x}_i \in \mathbb{C}^{m \times 1}$  为观测数据向量, 而  $\mathbf{W}$  表示加性观测误差矩阵。

在信号处理和系统科学等领域中，观测数据矩阵的列空间

$$\text{Span}(\mathbf{X}) = \text{Span}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \quad (8.3.2)$$

称为观测数据空间，而观测误差矩阵的列空间

$$\text{Span}(\mathbf{W}) = \text{Span}\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n\} \quad (8.3.3)$$

则称为噪声子空间。

### 8.3.1 信号子空间与噪声子空间

定义相关矩阵

$$\mathbf{R}_X = E\{\mathbf{X}^H \mathbf{X}\} = E\{(\mathbf{A} + \mathbf{W})^H (\mathbf{A} + \mathbf{W})\} \quad (8.3.4)$$

假设误差矩阵  $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n]$  与真实数据矩阵  $\mathbf{A}$  统计不相关，则

$$\mathbf{R}_X = E\{\mathbf{X}^H \mathbf{X}\} = E\{\mathbf{A}^H \mathbf{A}\} + E\{\mathbf{W}^H \mathbf{W}\} \quad (8.3.5)$$

令  $\mathbf{R} = E\{\mathbf{A}^H \mathbf{A}\}$  和  $E\{\mathbf{W}^H \mathbf{W}\} = \sigma_w^2 \mathbf{I}$  (即各观测噪声相互统计不相关，并且具有相同的方差  $\sigma_w^2$ )，则

$$\mathbf{R}_X = \mathbf{R} + \sigma_w^2 \mathbf{I}$$

令  $\text{rank}(\mathbf{A}) = r$ ，则矩阵  $\mathbf{R}_X = E\{\mathbf{A}^H \mathbf{A}\}$  的特征值分解

$$\mathbf{R}_X = \mathbf{U} \boldsymbol{\Lambda} \mathbf{U}^H + \sigma_w^2 \mathbf{I} = \mathbf{U} (\mathbf{A} + \sigma_w^2 \mathbf{I}) \mathbf{U}^H = \mathbf{U} \boldsymbol{\Pi} \mathbf{U}^H$$

式中

$$\boldsymbol{\Pi} = \boldsymbol{\Sigma} + \sigma_w^2 \mathbf{I} = \text{diag}(\sigma_1^2 + \sigma_w^2, \dots, \sigma_r^2 + \sigma_w^2, \sigma_{r+1}^2, \dots, \sigma_n^2)$$

其中， $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \dots, \sigma_r^2, 0, \dots, 0)$ ，且  $\sigma_1^2 \geq \dots \geq \sigma_r^2$  为真实自相关矩阵  $E\{\mathbf{A}^H \mathbf{A}\}$  的非零特征值。

显然，如果信噪比足够大，即  $\sigma_r^2$  比  $\sigma_w^2$  明显大，则将含噪声的自相关矩阵  $\mathbf{R}_X$  的前  $r$  个大特征值

$$\lambda_1 = \sigma_1^2 + \sigma_w^2, \dots, \lambda_r = \sigma_r^2 + \sigma_w^2$$

称为主特征值 (principal eigenvalue)，而将剩余的  $n - r$  个小特征值

$$\lambda_{r+1} = \sigma_{r+1}^2, \dots, \lambda_n = \sigma_n^2$$

称为次特征值 (minor eigenvalue)。

这样，自相关矩阵  $\mathbf{R}_X$  的特征值分解即可写成

$$\mathbf{R}_X = [\mathbf{U}_s, \mathbf{U}_n] \begin{bmatrix} \boldsymbol{\Sigma}_s & \mathbf{O} \\ \mathbf{O} & \boldsymbol{\Sigma}_{n-s} \end{bmatrix} \begin{bmatrix} \mathbf{U}_s^H \\ \mathbf{U}_{n-s}^H \end{bmatrix} = \mathbf{S} \boldsymbol{\Sigma}_s \mathbf{S}^H + \mathbf{G} \boldsymbol{\Sigma}_{n-s} \mathbf{G}^H \quad (8.3.6)$$

式中

$$\begin{aligned}\mathbf{S} &\stackrel{\text{def}}{=} [\mathbf{s}_1, \dots, \mathbf{s}_r] = [\mathbf{u}_1, \dots, \mathbf{u}_r] \\ \mathbf{G} &\stackrel{\text{def}}{=} [\mathbf{g}_1, \dots, \mathbf{g}_{n-r}] = [\mathbf{u}_{r+1}, \dots, \mathbf{u}_n] \\ \boldsymbol{\Sigma}_s &= \text{diag}(\sigma_1^2 + \sigma_w^2, \dots, \sigma_r^2 + \sigma_w^2) \\ \boldsymbol{\Sigma}_n &= \text{diag}(\sigma_w^2, \dots, \sigma_w^2)\end{aligned}$$

因此,  $m \times r$  酉矩阵  $\mathbf{S}$  和  $m \times (n-r)$  酉矩阵  $\mathbf{G}$  分别是与  $r$  个主特征值和  $n-r$  个次特征值对应的特征向量构成的矩阵。

**定义 8.3.1** 令  $\mathbf{S}$  是与观测数据的自相关矩阵的  $r$  个大特征值  $\lambda_1, \dots, \lambda_r$  对应的特征向量矩阵, 其列空间  $\text{Span}(\mathbf{S}) = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_r\}$  称为观测数据空间  $\text{Span}(\mathbf{X})$  的信号子空间, 而与另外  $n-r$  个次特征值对应的特征向量矩阵  $\mathbf{G}$  的列空间  $\text{Span}(\mathbf{G}) = \text{Span}\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_n\}$  称为观测数据空间的噪声子空间。

下面分析信号子空间和噪声子空间的几何意义。

由子空间的构造方法及酉矩阵的特点知, 信号子空间与噪声子空间正交, 即

$$\text{Span}\{\mathbf{s}_1, \dots, \mathbf{s}_r\} \perp \text{Span}\{\mathbf{g}_1, \dots, \mathbf{g}_{n-r}\} \quad (8.3.7)$$

由于  $\mathbf{U}$  是酉矩阵, 故

$$\mathbf{U}\mathbf{U}^H = [\mathbf{S}, \mathbf{G}] \begin{bmatrix} \mathbf{S}^H \\ \mathbf{G}^H \end{bmatrix} = \mathbf{S}\mathbf{S}^H + \mathbf{G}\mathbf{G}^H = \mathbf{I}$$

即有

$$\mathbf{G}\mathbf{G}^H = \mathbf{I} - \mathbf{S}\mathbf{S}^H \quad (8.3.8)$$

定义信号子空间上的投影矩阵

$$\mathbf{P}_s \stackrel{\text{def}}{=} \mathbf{S}\langle \mathbf{S}, \mathbf{S} \rangle^{-1}\mathbf{S}^H = \mathbf{S}\mathbf{S}^H \quad (8.3.9)$$

式中使用了矩阵内积  $\langle \mathbf{S}, \mathbf{S} \rangle = \mathbf{S}^H \mathbf{S} = \mathbf{I}$ 。于是,  $\mathbf{P}_s \mathbf{x} = \mathbf{S}\mathbf{S}^H \mathbf{x}$  可视为向量  $\mathbf{x}$  在信号子空间上的投影, 故常将  $\mathbf{S}\mathbf{S}^H$  视为信号子空间 (signal subspace)。

另外,  $(\mathbf{I} - \mathbf{P}_s)\mathbf{x}$  则代表向量  $\mathbf{x}$  在信号子空间上的正交投影。由  $\langle \mathbf{G}, \mathbf{G} \rangle = \mathbf{G}^H \mathbf{G} = \mathbf{I}$  得噪声子空间上的投影矩阵  $\mathbf{P}_n = \mathbf{G}\langle \mathbf{G}, \mathbf{G} \rangle^{-1}\mathbf{G}^H = \mathbf{G}\mathbf{G}^H$ 。因此, 常称

$$\mathbf{G}\mathbf{G}^H = \mathbf{I} - \mathbf{S}\mathbf{S}^H = \mathbf{I} - \mathbf{P}_s \quad (8.3.10)$$

为噪声子空间 (noise subspace), 它表示信号子空间的正交投影矩阵。

只使用信号子空间  $\mathbf{S}\mathbf{S}^H$  或者噪声子空间  $\mathbf{G}\mathbf{G}^H$  的信号分析方法分别称为信号子空间方法或噪声子空间方法。在模式识别中, 信号子空间方法被称为主分量分析 (principal component analysis, PCA) 方法, 而噪声子空间方法则被称为次分量分析 (minor component analysis, MCA) 方法。

子空间应用具有以下几个特点<sup>[522]</sup>:

(1) 无论信号子空间方法, 还是噪声子空间方法, 都只需要使用少数几个奇异向量或者特征向量。若矩阵  $\mathbf{A}_{m \times n}$  的大奇异值 (或者特征值) 个数比小奇异值 (或者特征值) 个数少, 则使用维数比较小的信号子空间比噪声子空间更有效。反之, 则使用噪声子空间更方便。

(2) 在很多应用中, 并不需要奇异值或者特征值, 而只需知道矩阵的秩以及奇异向量或者特征向量即可。

(3) 多数情况下, 并不需要准确知道奇异向量或者特征向量, 而只需知道张成信号子空间或者噪声子空间的基向量即可。

(4) 信号子空间  $\mathbf{SS}^H$  和噪声子空间  $\mathbf{GG}^H$  可以通过  $\mathbf{GG}^H = \mathbf{I} - \mathbf{SS}^H$  相互转换。

下面介绍子空间方法的两个应用。

### 8.3.2 子空间方法应用 1: 多重信号分类 (MUSIC)

下面讨论如何利用子空间进行多个信号分类。

令  $\mathbf{x}(t)$  是在第  $t$  个快拍观察到的数据向量。在阵列信号处理和空间谱估计中,  $\mathbf{x}(t) = [x_1(t), x_2, \dots, x_n(t)]^T$  由  $n$  个阵元 (天线或传感器) 的观测数据组成。在时域谱估计中, 向量  $\mathbf{x}(t) = [x(t), x(t-1), \dots, x(t-n-1)]^T$  由连续的  $n$  个观察数据样本组成。

假定数据向量  $\mathbf{x}(t)$  是  $r$  个窄带信号入射到  $n$  个阵元组成的阵列的观察数据向量或者是  $r$  个不相干的复谐波的叠加, 即

$$\mathbf{x}(t) = \sum_{i=1}^r s_i(t) \mathbf{a}(\omega_i) + \mathbf{v}(t) = \mathbf{As}(t) + \mathbf{n}(t) \quad (8.3.11)$$

式中,  $\mathbf{A} = [\mathbf{a}(\omega_1), \dots, \mathbf{a}(\omega_r)]$  为  $n \times r$  阵列响应矩阵,  $\mathbf{a}(\omega_i) = [1, e^{j\omega_i}, \dots, e^{j(n-1)\omega_i}]^T$  为方向向量或者频率向量;  $\mathbf{s}(t) = [s_1(t), \dots, s_r(t)]^T$  为随机信号向量, 其均值为零向量, 协方差矩阵为  $\mathbf{R}_s = E\{\mathbf{s}(t)\mathbf{s}^H(t)\}$ ; 而  $\mathbf{v}(t) = [v_1(t), \dots, v_n(t)]^T$  为加性噪声向量, 其各个分量为高斯白噪声, 它们具有零均值和相同的方差  $\sigma^2$ 。在谐波恢复中, 参数  $\omega_i$  为复谐波的频率; 在阵列信号处理中,  $\omega_i$  是一空间参数

$$\omega_i = 2\pi \frac{d}{\lambda} \sin \theta_i$$

式中,  $d$  为相邻两个阵元之间的距离 (假定阵元等间距排列成一直线),  $\lambda$  为波长, 且  $\theta_i$  表示第  $i$  个窄带信号达到阵元的入射方向, 简称波达方向。

现在的问题是: 根据  $N$  个快拍的观测数据向量  $\mathbf{x}(t)$  ( $t = 1, 2, \dots, N$ ) 估计  $r$  个参数  $\omega_i$ 。这相当于对  $r$  个混合信号进行分类, 简称多重信号分类。

假定噪声向量  $\mathbf{v}(t)$  与信号向量  $\mathbf{s}(t)$  统计不相关, 并令观测数据向量的协方差矩阵  $\mathbf{R}_{xx} = E\{\mathbf{x}(t)\mathbf{x}^H(t)\}$  的特征值分解为

$$\mathbf{R}_{xx} = \mathbf{AP}_{ss}\mathbf{A}^H + \sigma^2 \mathbf{I} = \mathbf{U}\Sigma\mathbf{U}^H = [\mathbf{S}, \mathbf{G}] \begin{bmatrix} \Sigma & \mathbf{O} \\ \mathbf{O} & \sigma^2 \mathbf{I}_{n-r} \end{bmatrix} \begin{bmatrix} \mathbf{S}^H \\ \mathbf{G}^H \end{bmatrix} \quad (8.3.12)$$

式中,  $\mathbf{P}_{ss} = \mathbb{E}\{\mathbf{s}(t)\mathbf{s}^H(t)\}$ , 且  $\Sigma$  包含了  $r$  个大特征值, 它们比  $\sigma^2$  明显大。

考查

$$\mathbf{R}_{xx}\mathbf{G} = [\mathbf{S}, \mathbf{G}] \begin{bmatrix} \Sigma & \mathbf{O} \\ \mathbf{O} & \sigma^2 \mathbf{I}_{n-r} \end{bmatrix} \begin{bmatrix} \mathbf{S}^H \\ \mathbf{G}^H \end{bmatrix} \mathbf{G} = [\mathbf{S}, \mathbf{G}] \begin{bmatrix} \Sigma & \mathbf{O} \\ \mathbf{O} & \sigma^2 \mathbf{I}_{n-r} \end{bmatrix} \begin{bmatrix} \mathbf{O} \\ \mathbf{I} \end{bmatrix} = \sigma^2 \mathbf{G} \quad (8.3.13)$$

又由  $\mathbf{R}_{xx} = \mathbf{A}\mathbf{P}_{ss}\mathbf{A}^H + \sigma^2 \mathbf{I}$  有  $\mathbf{R}_{xx}\mathbf{G} = \mathbf{A}\mathbf{P}_{ss}\mathbf{A}^H\mathbf{G} + \sigma^2 \mathbf{G}$ , 利用式 (8.3.13) 的结果, 立即得到

$$\mathbf{A}\mathbf{P}_{ss}\mathbf{A}^H\mathbf{G} = \mathbf{O}$$

进而有

$$\mathbf{G}^H \mathbf{A}\mathbf{P}_{ss}\mathbf{A}^H \mathbf{G} = \mathbf{O} \quad (8.3.14)$$

众所周知,  $\mathbf{Q}$  非奇异时  $\mathbf{t}^H \mathbf{Q} \mathbf{t} = 0$ , 当且仅当  $\mathbf{t} = \mathbf{0}$ , 故式 (8.3.14) 成立的充分必要条件是

$$\mathbf{A}^H \mathbf{G} = \mathbf{O} \quad (8.3.15)$$

因为  $\mathbf{P}_{ss} = \mathbb{E}\{\mathbf{s}(t)\mathbf{s}^H(t)\}$  非奇异。将  $\mathbf{A} = [\mathbf{a}(\omega_1), \dots, \mathbf{a}(\omega_p)]$  代入式 (8.3.15), 即有

$$\mathbf{a}^H(\omega) \mathbf{G} = \mathbf{0}^T, \quad \omega = \omega_1, \omega_2, \dots, \omega_p \quad (8.3.16)$$

显然, 当  $\omega \neq \omega_1, \omega_2, \dots, \omega_p$  时,  $\mathbf{a}^H(\omega) \mathbf{G} \neq \mathbf{0}^T$ 。

将式 (8.3.16) 改写成标量形式, 可以定义一种类似于功率谱的函数

$$P(\omega) = \frac{1}{\mathbf{a}^H(\omega) \mathbf{G} \mathbf{G}^H \mathbf{a}(\omega)} \quad (8.3.17)$$

上式取峰值的  $p$  个  $\omega$  值  $\omega_1, \omega_2, \dots, \omega_p$  给出  $p$  个信号的波达方向  $\theta_1, \theta_2, \dots, \theta_p$ 。

由于式 (8.3.17) 定义的函数  $P(\omega)$  描述了空间参数 (即波达方向) 的分布, 故常称为空间谱。由于它能够对多个空间信号进行识别 (即分类) 故这种方法称为多重信号分类方法, 简称 MUSIC (multiple signal classification) 方法, 它是 Schmidt [438], Biemvenu 及 Kopp [48] 于 1979 年独立提出的。后来, Schmidt 于 1986 年重新发表了他的论文 [439]。

将式 (8.3.10) 代入式 (8.3.17), 又可得到

$$P(\omega) = \frac{1}{\mathbf{a}^H(\omega) (\mathbf{I} - \mathbf{S} \mathbf{S}^H) \mathbf{a}(\omega)} \quad (8.3.18)$$

因为  $\mathbf{G} \mathbf{G}^H$  和  $\mathbf{S} \mathbf{S}^H$  分别代表信号子空间和噪声子空间, 故式 (8.3.17) 和式 (8.3.18) 可分别视为噪声子空间方法和信号子空间方法。

在实际应用中, 通常将  $\omega$  划分为数百个等间距的单位, 得到

$$\omega_i = 2\pi i \Delta f \quad (8.3.19)$$

例如取  $\Delta f = \frac{0.5}{500} = 0.001$ , 然后将每个  $\omega_i$  值代入式 (8.3.17) 或式 (8.3.18), 求出所有峰值对应的  $\omega$  值。因此, MUSIC 算法需要在频率轴上进行全域搜索, 计算量比较大。另外,

执行 MUSIC 算法是选择噪声子空间还是信号子空间方式，决定于  $\mathbf{G}$  和  $\mathbf{S}$  中哪一个具有更小的维数。除了计算量有所不同外，这两种方式并没有本质上的区别。

为了改进 MUSIC 算法的性能，已提出了好几种变型，例如基于最大似然法的改进 MUSIC 算法<sup>[446]</sup>、解相干 MUSIC 算法和求根 MUSIC 算法<sup>[29]</sup> 等。有关这些变型的详细讨论，可参考文献 [546]。

### 8.3.3 子空间方法应用 2：子空间白化

令  $\mathbf{a}$  是  $m \times 1$  随机向量，具有零均值，其协方差矩阵  $\mathbf{C}_a = E\{\mathbf{a}\mathbf{a}^H\}$ 。若  $m \times m$  协方差矩阵  $\mathbf{C}_a$  非奇异，且不等于单位矩阵，则称随机向量  $\mathbf{a}$  为有色或非白随机向量。

令协方差矩阵的特征值分解为  $\mathbf{C}_a = \mathbf{V}\mathbf{D}\mathbf{V}^H$ ，并且矩阵

$$\mathbf{W} = \mathbf{V}\mathbf{D}^{-1/2}\mathbf{V}^H = \mathbf{C}_a^{-1/2} \quad (8.3.20)$$

则变换结果

$$\mathbf{b} = \mathbf{W}\mathbf{a} = \mathbf{C}_a^{-1/2}\mathbf{a} \quad (8.3.21)$$

的协方差矩阵等于单位矩阵，即有

$$\mathbf{C}_b = E\{\mathbf{b}\mathbf{b}^H\} = \mathbf{W}\mathbf{C}_a\mathbf{W}^H = \mathbf{C}_a^{-1/2}E\{\mathbf{a}\mathbf{a}^H\}[\mathbf{C}_a^{-1/2}]^H = \mathbf{I} \quad (8.3.22)$$

因为  $\mathbf{C}_a^{-1/2} = \mathbf{V}\mathbf{D}^{-1/2}\mathbf{V}^H$  为 Hermitian 矩阵。上式表明，随机向量  $\mathbf{b}$  为标准白色随机向量（随机向量的各元素相互统计不相关，并且各方差均等于 1）。换言之，原来有色的随机向量经过线性变换  $\mathbf{W}\mathbf{a}$  之后，变成了白色随机向量。线性变换矩阵  $\mathbf{W} = \mathbf{C}_a^{-1/2}$  称为随机向量  $\mathbf{a}$  的白化矩阵。

然而，若  $m \times m$  协方差矩阵  $\mathbf{C}_a$  奇异或者秩亏缺，例如  $\text{rank}(\mathbf{C}_a) = n < m$ ，则不存在使  $\mathbf{W}\mathbf{C}_a\mathbf{W}^H = \mathbf{I}$  的白化矩阵  $\mathbf{W}$ 。此时，应该考虑在秩空间  $V = \text{Range}(\mathbf{C}_a) = \text{Col}(\mathbf{C}_a)$  上使随机向量  $\mathbf{a}$  白化。这一白化称为子空间白化 (subspace whitening)，是 Eldar 和 Oppenheim 于 2003 年提出的<sup>[157]</sup>。

若秩亏缺的协方差矩阵  $\mathbf{C}_a$  的特征值分解为

$$\mathbf{C}_a = [\mathbf{V}_1, \mathbf{V}_2] \begin{bmatrix} \mathbf{D}_{n \times n} & \mathbf{O}_{n \times (m-n)} \\ \mathbf{O}_{(m-n) \times n} & \mathbf{O}_{(m-n) \times (m-n)} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad (8.3.23)$$

并令

$$\mathbf{W} = \mathbf{V}_1\mathbf{D}^{-1/2}\mathbf{V}_1^H \quad (8.3.24)$$

则易知线性变换结果

$$\mathbf{b} = \mathbf{W}\mathbf{a} = \mathbf{V}_1\mathbf{D}^{-1/2}\mathbf{V}_1^H\mathbf{a} \quad (8.3.25)$$

的协方差矩阵

$$\begin{aligned} \mathbf{C}_b &= \mathbb{E}\{\mathbf{b}\mathbf{b}^H\} = \mathbf{W}\mathbf{C}_a\mathbf{W}^H \\ &= \mathbf{V}_1\mathbf{D}^{-1/2}\mathbf{V}_1^H[\mathbf{V}_1, \mathbf{V}_2]\begin{bmatrix} \mathbf{D} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix}\begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix}\mathbf{V}_1\mathbf{D}^{-1/2}\mathbf{V}_1^H \\ &= [\mathbf{V}_1\mathbf{D}^{-1/2}, \mathbf{O}]\begin{bmatrix} \mathbf{D} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix}\begin{bmatrix} \mathbf{D}^{-1/2}\mathbf{V}_1^H \\ \mathbf{O} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_n & \mathbf{O} \\ \mathbf{O} & \mathbf{O} \end{bmatrix} \end{aligned}$$

即  $\mathbf{b} = \mathbf{W}\mathbf{a}$  是在子空间  $\text{Range}(\mathbf{C}_a)$  内的白色随机向量。因此，称  $\mathbf{W} = \mathbf{V}_1\mathbf{D}^{-1/2}\mathbf{V}_1^H$  为子空间白化矩阵。关于子空间白化及其具体实现，读者可进一步参考文献 [157]。文献 [158] 将子空间白化应用于信号检测，提出了基于正交和投影正交匹配滤波器的检测方法。

## 8.4 Grassmann 流形与 Stiefel 流形

考察目标函数  $J(\mathbf{W})$  的最小化，其中， $\mathbf{W}$  为  $n \times r$  矩阵。对  $\mathbf{W}$  的常用约束有两类：

- (1) 正交约束 (orthogonality constraint) 要求  $\mathbf{W}$  满足正交条件  $\mathbf{W}^H\mathbf{W} = \mathbf{I}_r$  ( $n \geq r$ ) 或者  $\mathbf{W}\mathbf{W}^H = \mathbf{I}_n$  ( $n < r$ )。满足这种条件的矩阵  $\mathbf{W}$  称为半正交矩阵。
- (2) 齐次性约束 (homogeneity constraint) 要求  $J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q})$ ，其中， $\mathbf{Q}$  为  $r \times r$  正交矩阵。

下面分别研究两类最优化问题：一类同时使用正交约束和齐次性约束，另一类只使用正交约束。

### 8.4.1 不变子空间

**定义 8.4.1** (线性流形) 令  $H$  是  $V$  空间的子空间， $\mathcal{L}$  代表  $H$  内有限个元素的所有线性组合的全体，即

$$\mathcal{L} = \left\{ \xi : \xi = \sum_{i=1}^n a_i \eta_i, \eta_i \in H \right\}$$

称  $\mathcal{L}$  是由  $H$  张成的线性流形。

称两个矩阵为等价矩阵，若它们的列向量张成的子空间相同。换言之，等价的矩阵集合具有相同的列空间，即子空间相对于基的任意选择是不变的。在这个意义上，这类子空间也称不变子空间。所有相同的子空间组成等价子空间类。

令  $n \times r$  矩阵  $\mathbf{W}$  具有满列秩，其列空间  $H = \text{Col}(\mathbf{W})$ ，并令  $\mathbf{x}$  是  $\mathbb{C}^n$  空间的一任意向量，则  $\mathbf{x}$  到  $H$  子空间的投影为

$$\mathbf{P}_H \mathbf{x} = \mathbf{W}(\mathbf{W}^H \mathbf{W})^{-1} \mathbf{W}^H \mathbf{x}$$

令  $r \times r$  矩阵  $\mathbf{M}$  非奇异，且  $n \times r$  矩阵  $\mathbf{WM}$  的列空间  $S = \text{Col}(\mathbf{WM})$ ，则  $\mathbf{x}$  到  $S$  子空

间的投影为

$$\begin{aligned}\mathbf{P}_S \mathbf{x} &= \mathbf{W} \mathbf{M} [(\mathbf{W} \mathbf{M})^H (\mathbf{W} \mathbf{M})]^{-1} (\mathbf{W} \mathbf{M})^H \mathbf{x} \\ &= \mathbf{W} (\mathbf{W}^H \mathbf{W})^{-1} \mathbf{W}^H \mathbf{x} = \mathbf{P}_H \mathbf{x}\end{aligned}$$

由于向量  $\mathbf{x}$  到子空间  $H$  和  $S$  的投影相等, 故称  $H$  和  $S$  是两个等价子空间, 或者称  $n \times r$  满列秩矩阵  $\mathbf{W}$  的列空间  $\text{Col}(\mathbf{W})$  是相对于  $r \times r$  非奇异矩阵  $\mathbf{M}$  不变的子空间。

类似地, 向量  $\mathbf{x}$  到具有满行秩的  $n \times r$  矩阵  $\mathbf{W}$  的行空间  $H_1 = \text{Row}(\mathbf{W})$  上的投影

$$\mathbf{x} \mathbf{P}_{H_1} = \mathbf{x} \mathbf{W}^H (\mathbf{W} \mathbf{W}^H)^{-1} \mathbf{W}$$

若  $\mathbf{N}$  是一个  $n \times n$  非奇异矩阵, 则  $\mathbf{x}$  到行空间  $S_1 = \text{Row}(\mathbf{N} \mathbf{W})$  上的投影

$$\begin{aligned}\mathbf{x} \mathbf{P}_{S_1} &= \mathbf{x} \mathbf{W}^H \mathbf{N}^H (\mathbf{N} \mathbf{W} \mathbf{W}^H \mathbf{N}^H)^{-1} \mathbf{N} \mathbf{W} \\ &= \mathbf{x} \mathbf{W}^H (\mathbf{W} \mathbf{W}^H)^{-1} \mathbf{W} = \mathbf{x} \mathbf{P}_{H_1}\end{aligned}$$

即  $H_1$  和  $S_1$  为等价子空间。也就是说,  $n \times r$  满行秩矩阵的行空间  $\text{Row}(\mathbf{W})$  是相对于  $n \times n$  非奇异矩阵  $\mathbf{N}$  不变的子空间。

下面考察不变子空间的集合。

#### 8.4.2 Grassmann 流形

围绕子空间  $H$  展开的理论分析, 其核心问题往往集中体现在另一空间  $V$  的任意向量  $\mathbf{x}$  到子空间  $H$  的投影分析, 因为这一投影涉及信号的最优滤波、最优估计、干扰对消等一系列应用。在这些应用中, 到子空间的投影矩阵  $\mathbf{P}_H$  和正交投影矩阵  $\mathbf{P}_H^\perp = \mathbf{I} - \mathbf{P}_H$  起着关键的作用。当  $H$  是矩阵  $\mathbf{W}$  的列(或者行)向量张成的子空间即列(或者行)空间时, 常将  $\mathbf{W}$  的投影矩阵视为子空间  $H$  的代表。因此, 不变子空间也可以通过投影矩阵作解释。

(1) “高瘦”半正交矩阵 (tall-skinny semi-orthogonal matrix)

当  $n \geq r$ , 并且对矩阵  $\mathbf{W}_{n \times r}$  加有正交约束  $\mathbf{W}^H \mathbf{W} = \mathbf{I}_r$  时,  $\mathbf{W}$  的列空间  $H = \text{Col}(\mathbf{W})$  常可以用投影矩阵

$$\mathbf{P}_H = \mathbf{W} (\mathbf{W}^H \mathbf{W})^{-1} \mathbf{W}^H = \mathbf{W} \mathbf{W}^H$$

等价描述。因此, 不变的列空间  $\text{Col}(\mathbf{W})$  可等价描述为矩阵乘积  $\mathbf{W} \mathbf{W}^H$  不变。也就是说, 当两个  $n \times r$  半正交矩阵  $\mathbf{W}_1 \neq \mathbf{W}_2$  不同, 但却满足条件  $\mathbf{W}_1 \mathbf{W}_1^H = \mathbf{W}_2 \mathbf{W}_2^H$  时, 矩阵  $\mathbf{W}_1$  和  $\mathbf{W}_2$  就是等价矩阵, 它们的列空间相同。

(2) “矮”半正交矩阵 (short semi-orthogonal matrix)

当  $n < r$ , 并且对矩阵  $\mathbf{W}_{n \times r}$  加有正交约束  $\mathbf{W} \mathbf{W}^H = \mathbf{I}_n$  时,  $\mathbf{W}$  的行空间  $H = \text{Row}(\mathbf{W})$  常可以用投影矩阵

$$\mathbf{P}_H = \mathbf{W}^H (\mathbf{W} \mathbf{W}^H)^{-1} \mathbf{W} = \mathbf{W}^H \mathbf{W}$$

等价描述。因此，不变的行空间  $\text{Row}(\mathbf{W})$  可等价描述为矩阵乘积  $\mathbf{W}^H \mathbf{W}$  不变。换言之，满足  $\mathbf{W}_1^H \mathbf{W}_1 = \mathbf{W}_2^H \mathbf{W}_2$  的两个不同矩阵  $\mathbf{W}_1$  和  $\mathbf{W}_2$  相互等价。

**引理 8.4.1** 假定  $n \times r$  矩阵  $\mathbf{W}_1 = \mathbf{W}_2 \mathbf{Q}$ ，其中， $n \geq r$ ， $\mathbf{Q}$  是  $r \times r$  正交矩阵，并且  $H_1 = \text{Col}(\mathbf{W}_1)$  和  $H_2 = \text{Col}(\mathbf{W}_2)$ ，则  $P_{H_1} = P_{H_2}$ ，从而  $\mathbf{W}_1$  和  $\mathbf{W}_2$  等价。

**证明** 计算到  $\mathbf{W}_1 = \mathbf{W}_2 \mathbf{Q}$  的列空间  $H_1$  的投影矩阵，得

$$P_{H_1} = (\mathbf{W}_2 \mathbf{Q}) [(\mathbf{W}_2 \mathbf{Q})^H (\mathbf{W}_2 \mathbf{Q})]^{-1} (\mathbf{W}_2 \mathbf{Q})^H = \mathbf{W}_2 (\mathbf{W}_2^H \mathbf{W}_2)^{-1} \mathbf{W}_2^H = P_{H_2}$$

由于列空间  $\text{Col}(\mathbf{W}_1)$  与  $\text{Col}(\mathbf{W}_2)$  相同，故矩阵  $\mathbf{W}_1$  与  $\mathbf{W}_2$  等价。 ■

上述引理表明，两个矩阵等价或者张成相同的列空间，若一个矩阵等于另外一个矩阵右乘一个正交矩阵。特别地，若  $\mathbf{W}_{n \times r}$  满足正交约束条件  $\mathbf{W}^H \mathbf{W} = \mathbf{I}_r$  和齐次性约束条件  $J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q})$ ，其中， $\mathbf{Q}$  为  $r \times r$  任意正交矩阵，则极小化问题  $\min J(\mathbf{W})$  的解不是一个  $\mathbf{W}$  矩阵，而是由  $\mathbf{W}\mathbf{Q}$  组成的矩阵集合。矩阵集合内的任何一个矩阵的列向量都张成相同的  $\mathbb{C}^r$  子空间。 $\mathbb{C}^n$  内的这一子空间集合称为 Grassmann 流形，用符号  $Gr(n, r)$  表示，即有

$$Gr(n, r) = \{\mathbf{W} \in \mathbb{C}^{n \times r} : \mathbf{W}^H \mathbf{W} = \mathbf{I}_r, \mathbf{W}\mathbf{W}^H = \text{同一矩阵}\} \quad (8.4.1)$$

Grassmann 流形是 Grassmann 于 1848 年提出的，但当时的表示比较模糊，以至于许多年之后，才被人们认识<sup>[2]</sup>。Grassmann 流形的原始定义可以在文献 [200, Chap.3, Sec.1] 中找到。

总结以上讨论，可以得出以下结论：对于极小化问题

$$\min J(\mathbf{W}) \quad (8.4.2)$$

约束条件为

$$\mathbf{W}^H \mathbf{W} = \mathbf{I}_r, \quad J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q}), \quad \mathbf{Q}^H \mathbf{Q} = \mathbf{Q}\mathbf{Q}^H = \mathbf{I}_r \quad (8.4.3)$$

其解不是单个矩阵，而是称为 Grassmann 流形的矩阵集合。也就是说，Grassmann 流形的任何一个点都是同时具有正交约束和齐次性约束的极小化问题的解。

Grassmann 流形在最优化算法、不变子空间计算、物理计算、子空间跟踪等中有着重要的应用，其几何特性由 Edelman 等人于 1998 年给出了比较系统的解释<sup>[153]</sup>。

### 8.4.3 Stiefel 流形

下面考虑只有正交约束的最小化问题

$$\min J(\mathbf{W}) \quad \text{subject to} \quad \mathbf{W}^H \mathbf{W} = \mathbf{I}_r, \quad (8.4.4)$$

上述最优化问题的解为  $n \times r$  半正交矩阵的集合。所有  $n \times r$  半正交矩阵的集合称为 Stiefel 流形，用符号  $St(n, r)$  表示，即

$$St(n, r) = \{\mathbf{W} \in \mathbb{C}^{n \times r} : \mathbf{W}^H \mathbf{W} = \mathbf{I}_r\} \quad (8.4.5)$$

它是 Stiefel 在 1930 年代研究拓扑时提出的<sup>[466]</sup>。Stiefel 还与 Hestens 一起于 1952 年提出了著名的共轭梯度算法<sup>[228]</sup>。

比较 Grassmann 流形与 Stiefel 流形之间的联系与区别是有趣的。

(1) Stiefel 流形  $St(n, r)$  是  $n \times r$  “高瘦”半正交矩阵的集合，该流形  $St(n, r)$  上的一个点代表  $n \times r$  半正交矩阵集合的一个半正交矩阵。

(2) Grassmann 流形  $Gr(n, r)$  由 Stiefel 流形  $St(n, r)$  中那些张成相同列空间的矩阵组成，该流形  $Gr(n, r)$  上的一个点是张成相同列空间的一个矩阵组合。张成该子空间的矩阵存在多种选择。换言之，Grassmann 流形的点是  $n \times r$  半正交矩阵的等价类，其中的任何两个矩阵都是等价的，即一个矩阵等于另外一个矩阵右乘一个  $r \times r$  正交矩阵。

所有  $r \times r$  正交矩阵  $\mathbf{Q}$  的集合称为正交群 (orthogonal group)，用符号  $O_r$  表示，即有

$$O_r = \{\mathbf{Q}_r \in \mathbb{C}^{r \times r} | \mathbf{Q}_r^H \mathbf{Q}_r = \mathbf{Q}_r \mathbf{Q}_r^H = \mathbf{I}_r\} \quad (8.4.6)$$

正交群、Grassmann 流形与 Stiefel 流形是与正交约束密切相关的三种子空间流形。下面研究这三种子空间流形之间的关系。

首先，令  $\mathbf{W}$  是 Stiefel 流形上的一个点，即  $\mathbf{W} \in St(n, r)$  是一个  $n \times r$  半正交矩阵。收集所有满足正交条件  $\mathbf{W}_\perp^H \mathbf{W}_\perp = \mathbf{I}_{n-r}$  和  $\mathbf{W}_\perp^H \mathbf{W} = \mathbf{O}_{(n-r) \times r}$  的  $n \times (n-r)$  矩阵  $\mathbf{W}_\perp \in St(n, n-r)$ ，则  $[\mathbf{W}, \mathbf{W}_\perp]$  构成一正交群  $O_n$ 。如果令  $\mathbf{Q}$  是满足  $\mathbf{W}_\perp \mathbf{Q} = \mathbf{O}$  的任意一个  $(n-r) \times (n-r)$  正交矩阵，则  $\mathbf{Q}$  的集合是另一正交群  $O_{n-r}$ 。注意到矩阵乘积

$$[\mathbf{W}, \mathbf{W}_\perp] \begin{bmatrix} \mathbf{I} \\ \mathbf{Q} \end{bmatrix} = \mathbf{W}$$

这表明，半正交矩阵  $\mathbf{W}_{n \times r}$  可以通过  $n \times n$  正交群  $O_n$  和  $(n-r) \times (n-r)$  正交群  $O_{n-r}$  识别。由上式的矩阵乘法知，Stiefel 流形  $St(n, r)$  上的一个点可以用两个正交群的商  $O_n/O_{n-r}$  作表示形式，即有

$$St(n, r) = O_n / O_{n-r} \quad (8.4.7)$$

其次，如果我们使用半正交矩阵  $\mathbf{W}_{n \times r}$  表示 Stiefel 流形上的一个点，则满足  $\mathbf{W} = \mathbf{U}_s \mathbf{Q}$  ( $\mathbf{Q}$  为  $r \times r$  任意正交矩阵) 或  $\mathbf{U}_s = \mathbf{W} \mathbf{Q}^{-1}$  的所有矩阵  $\mathbf{U}_s$  组成 Grassmann 流形  $Gr(n, r)$  的一个点 (等价子空间类)。因此，若将逆矩阵运算视为矩阵除法，则可以将 Grassmann 流形  $Gr(n, r)$  表示成 Stiefel 流形  $St(n, r)$  与正交矩阵  $\mathbf{Q}$  的商，即有

$$Gr(n, r) = St(n, r) / O_r \quad (8.4.8)$$

式中， $O_r$  表示  $r \times r$  正交群。若将式 (8.4.7) 代入式 (8.4.8)，又可将 Grassmann 流形表示为正交群的商

$$Gr(n, r) = O_n / (O_r \times O_{n-r}) \quad (8.4.9)$$

以上关于正交群、Grassmann 流形和 Stiefel 流形三种子空间流形的讨论可以总结为表 8.4.1 的形式。

表 8.4.1 子空间流形的表示<sup>[153]</sup>

子空间流形	符 号	矩阵表示	商 表 示
正交群	$O_n$	$n \times n$ 矩阵	
Stiefel 流形	$St(n, r)$	$n \times r$ 矩阵	$O_n / O_{n-r}$
Grassmann 流形	$Gr(n, r)$	无	$St(n, r) / O_r$ 或者 $O_n / (O_r \times O_{n-r})$

下面讨论 Stiefel 流形、Grassmann 流形与 Rayleigh 商之间的关系。

**定义 8.4.2**<sup>[224, 5]</sup> 令  $\mathbf{X} \in St(n, r)$  是一个  $n \times r$  半正交矩阵，且  $\mathbf{A}$  为  $n \times n$  维 Hermitian 矩阵，则

$$\mathbf{R}_A(\mathbf{X}) = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{A} \mathbf{X} \quad (8.4.10)$$

称为  $\mathbf{A}$  的矩阵 Rayleigh 商。矩阵 Rayleigh 商的迹

$$\rho_A(\mathbf{X}) = \text{tr}(\mathbf{R}_A(\mathbf{X})) = \text{tr}(\mathbf{X}^H \mathbf{A} \mathbf{X} (\mathbf{X}^H \mathbf{X})^{-1}) \quad (8.4.11)$$

$$= \text{tr}((\mathbf{X}^H \mathbf{X})^{-1/2} \mathbf{X}^H \mathbf{A} \mathbf{X} (\mathbf{X}^H \mathbf{X})^{-1/2}) \quad (8.4.12)$$

称为推广的(标量) Rayleigh 商。

与 Rayleigh 商  $\frac{\mathbf{x}^H \mathbf{A} \mathbf{x}}{\mathbf{x}^H \mathbf{x}}$  通常约定  $\mathbf{x}^H \mathbf{x} = 1$  相类似，矩阵 Rayleigh 商假设  $\mathbf{X}^H \mathbf{X} = \mathbf{I}$ ，即  $\mathbf{X}$  是 Stiefel 流形上的点。换言之，矩阵 Rayleigh 商利用 Stiefel 流形定义。

推广的 Rayleigh 商保留了经典 Rayleigh 商的以下重要特性。

**命题 8.4.1**<sup>[5]</sup> 矩阵 Rayleigh 商定义式 (8.4.10) 和推广的 Rayleigh 商定义式 (8.4.12) 满足以下性质：

(1) 齐次性 Rayleigh 商  $\rho_A(\mathbf{X}) = \rho_A(\mathbf{M}\mathbf{X})$  对所有非奇异矩阵  $\mathbf{M}$  成立。这意味着，若  $\text{Col}(\mathbf{W}_1) = \text{Col}(\mathbf{W}_2)$ ，则  $\rho_A(\mathbf{W}_1) = \rho_A(\mathbf{W}_2)$ 。换言之，推广的 Rayleigh 商定义了 Grassmann 流形上的一个标量场 (scalar field)。

(2) 平稳性 推广的 Rayleigh 商  $\rho_A(\mathbf{X})$  关于  $\mathbf{X}$  的梯度矩阵  $\nabla \rho_A(\mathbf{X}) = \mathbf{O}$ ，当且仅当  $\text{col}(\mathbf{X})$  是矩阵  $\mathbf{A}$  的不变子空间，即  $\text{Col}(\mathbf{AX}) \subset \text{Col}(\mathbf{X})$ 。

(3) 最小残差  $\|\mathbf{AX} - \mathbf{XB}\|_F^2 \geq \|\mathbf{AX}\|_F^2 - \|\mathbf{X}\mathbf{R}_A(\mathbf{X})\|_F^2$ ，等号成立，当且仅当  $\mathbf{B} = \mathbf{R}_A(\mathbf{X})$ 。因此， $\mathbf{B} = \mathbf{R}_A(\mathbf{X})$  是  $\min \|\mathbf{AX} - \mathbf{XB}\|_F^2$  的唯一解。

上述讨论可以总结为 Stiefel 流形、Grassmann 流形与 Rayleigh 商之间的下列关系：

1. 矩阵 Rayleigh 商  $\mathbf{R}_A(\mathbf{X}) = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{A} \mathbf{X}$  利用 Stiefel 流形 (即  $\mathbf{X} \in St(n, r)$ ) 定义。

2. 推广的 Rayleigh 商定义了 Grassmann 流形上的一个标量场。

## 8.5 投影逼近子空间跟踪

特征子空间的跟踪与更新主要用于实时信号处理，所以要求它们应该是快速算法。快速算法至少应该考虑到以下因素：

(1)  $n$  时刻的子空间可以通过更新  $n-1$  时刻的子空间获得。

(2)  $n-1$  时刻到  $n$  时刻的协方差矩阵的变化应该尽可能地是低秩变化（最好是秩 1 变化或秩 2 变化）。

(3) 只需要跟踪低维子空间。

特征子空间跟踪与更新方法可以分为以下四大类。

(1) 正交基跟踪 只使用噪声子空间特征向量的正交基，而无须使用特征向量本身。这一特点可以简化一类特征子空间的自适应跟踪问题。

(2) 秩 1 更新 将非平稳信号在  $k$  时刻的协方差矩阵看作是  $k-1$  时刻的协方差矩阵与另外一个秩等于 1 的矩阵（它是观测向量的共轭转置与其本身的乘积）之和。因此，协方差矩阵的特征值分解的跟踪与所谓的秩 1 更新密切相关。文献 [532] 和文献 [95] 是这类方法的两个典型代表，其中，文献 [95] 的方法将秩 1 更新与一阶扰动问题联系起来，文献 [532] 的方法则包含了基于秩 1 和秩 2 修正的修正特征值分解的递推更新。

(3) 投影逼近 将特征子空间的确定当作一个无约束最优化问题来求解，相应的方法称为投影逼近子空间跟踪 [522]。

(4) Lanczos 子空间跟踪 利用 Lanczos 型迭代和随机逼近的概念，可以进行时变数据矩阵的子空间跟踪 [177]。Xu 等人在文献 [519] 和文献 [520] 中分别提出了三 Lanczos 和双 Lanczos 子空间跟踪算法；前者适用于协方差矩阵的特征值分解，后者针对数据矩阵的奇异值分解；而且在 Lanczos 递推过程中能够对主特征值和主奇异值的个数进行检验估计。

本节介绍基于投影逼近的子空间跟踪。

### 8.5.1 投影逼近子空间跟踪的基本理论

下面证明，具有正交性约束  $\mathbf{W}_{n \times r}^H \mathbf{W}_{n \times r} = \mathbf{I}_r$  和齐次性约束  $J(\mathbf{W}) = J(\mathbf{WQ}_{r \times r})$  的极小化问题  $\min J(\mathbf{W})$  可以等价为一个无约束的最优化问题。

令  $\mathbf{C} = \mathbb{E}\{\mathbf{x}\mathbf{x}^H\}$  表示  $n \times 1$  随机向量的自相关矩阵，目标函数为

$$J(\mathbf{W}) = \mathbb{E}\{\|\mathbf{x} - \mathbf{W}\mathbf{W}^H\mathbf{x}\|^2\} \quad (8.5.1)$$

或写作

$$\begin{aligned} J(\mathbf{W}) &= \mathbb{E}\{\|\mathbf{x} - \mathbf{W}\mathbf{W}^H\mathbf{x}\|^2\} \\ &= \mathbb{E}\{(\mathbf{x} - \mathbf{W}\mathbf{W}^H\mathbf{x})^H(\mathbf{x} - \mathbf{W}\mathbf{W}^H\mathbf{x})\} \\ &= \mathbb{E}\{\mathbf{x}^H\mathbf{x}\} - 2\mathbb{E}\{\mathbf{x}^H\mathbf{W}\mathbf{W}^H\mathbf{x}\} + \mathbb{E}\{\mathbf{x}^H\mathbf{W}\mathbf{W}^H\mathbf{W}\mathbf{W}^H\mathbf{x}\} \end{aligned} \quad (8.5.2)$$

注意到

$$\begin{aligned} \mathbb{E}\{\mathbf{x}^H \mathbf{x}\} &= \sum_{i=1}^n \mathbb{E}\{|x_i|^2\} = \text{tr}(\mathbb{E}\{\mathbf{x} \mathbf{x}^H\}) = \text{tr}(\mathbf{C}) \\ \mathbb{E}\{\mathbf{x}^H \mathbf{W} \mathbf{W}^H \mathbf{x}\} &= \text{tr}(\mathbb{E}\{\mathbf{W}^H \mathbf{x} \mathbf{x}^H \mathbf{W}\}) = \text{tr}(\mathbf{W}^H \mathbf{C} \mathbf{W}) \\ \mathbb{E}\{\mathbf{x}^H \mathbf{W} \mathbf{W}^H \mathbf{W} \mathbf{W}^H \mathbf{x}\} &= \text{tr}(\mathbb{E}\{\mathbf{W}^H \mathbf{x} \mathbf{x}^H \mathbf{W} \mathbf{W}^H \mathbf{W}\}) = \text{tr}(\mathbf{W}^H \mathbf{C} \mathbf{W} \mathbf{W}^H \mathbf{W}) \end{aligned}$$

则目标函数可以用迹函数表示为

$$J(\mathbf{W}) = \text{tr}(\mathbf{C}) - 2\text{tr}(\mathbf{W}^H \mathbf{C} \mathbf{W}) + \text{tr}(\mathbf{W}^H \mathbf{C} \mathbf{W} \mathbf{W}^H \mathbf{W}) \quad (8.5.3)$$

式中,  $\mathbf{W}$  是  $n \times r$  矩阵, 假定其秩等于  $r$ 。下面考虑极小化问题  $\min J(\mathbf{W})$ , 与之相关的重要问题是:

- (1) 是否存在  $J(\mathbf{W})$  的全局极小点  $\mathbf{W}$ ?
- (2) 该极小点  $\mathbf{W}$  与自相关矩阵  $\mathbf{C}$  的信号子空间有何关系?
- (3) 是否存在  $J(\mathbf{W})$  的其他局部极小点?

Yang 证明了下面的两个定理, 给出了以上问题的答案<sup>[522]</sup>。

**定理 8.5.1**  $\mathbf{W}$  是  $J(\mathbf{W})$  的一个平稳点, 当且仅当  $\mathbf{W} = \mathbf{U}_r \mathbf{Q}$ , 其中,  $\mathbf{U}_r \in \mathbb{C}^{n \times r}$  由自相关矩阵  $\mathbf{C}$  的  $r$  个不同的特征向量组成, 并且  $\mathbf{Q} \in \mathbb{C}^{r \times r}$  为任意酉矩阵。在每一个平衡点, 目标函数  $J(\mathbf{W})$  的值等于特征向量不在  $\mathbf{U}_r$  的那些特征值之和。

**定理 8.5.2** 目标函数  $J(\mathbf{W})$  的所有平稳点都是鞍点, 除非  $\mathbf{U}_r$  由自相关矩阵  $\mathbf{C}$  的  $r$  个主特征向量组成。在这一特殊情况下,  $J(\mathbf{W})$  达到全局极小值。

定理 8.5.1 和定理 8.5.2 表明了以下事实:

(1) 虽然在定义目标函数和无约束极小化问题时, 没有要求  $\mathbf{W}$  的列正交, 但是两个定理却表明, 式 (8.5.1) 的目标函数  $J(\mathbf{W})$  的极小化将自动导致  $\mathbf{W}$  为半正交矩阵, 即满足  $\mathbf{W}^H \mathbf{W} = \mathbf{I}$ 。

(2) 定理 8.5.2 表明, 当  $\mathbf{W}$  的列空间等于信号子空间, 即  $\text{Col}(\mathbf{W}) = \text{Span}(\mathbf{U}_r)$  时, 目标函数  $J(\mathbf{W})$  达到全局极小值, 并且目标函数没有其他任何局部极小值。

(3) 由目标函数的定义式 (8.5.1) 易知,  $J(\mathbf{W}) = J(\mathbf{W} \mathbf{Q})$  对于所有  $r \times r$  酉矩阵  $\mathbf{Q}$  成立, 即目标函数自动满足齐次性约束。

(4) 由于式 (8.5.1) 定义的目标函数自动满足齐次性约束, 并且其极小化自动导致  $\mathbf{W}$  满足正交约束  $\mathbf{W}^H \mathbf{W} = \mathbf{I}$ , 故目标函数极小化的解  $\mathbf{W}$  不是唯一确定的, 而是 Grassmann 流形上的点。

(5) 虽然  $\mathbf{W}$  不是唯一确定的, 但投影矩阵  $\mathbf{P} = \mathbf{W}(\mathbf{W}^H \mathbf{W})^{-1} \mathbf{W}^H = \mathbf{W} \mathbf{W}^H = \mathbf{U}_r \mathbf{U}_r^H$  是唯一确定的。也就是说, 不同的解张成相同的列空间。

(6) 当  $r = 1$  即目标函数为向量  $\mathbf{w}$  的函数时,  $J(\mathbf{w})$  极小化的解  $\mathbf{w}$  为自相关矩阵  $\mathbf{C}$  与最大特征值对应的特征向量。

因此, 具有正交性约束和齐次性约束的目标函数  $J(\mathbf{W})$  的极小化求解变为奇异值分解或特征值分解问题:

(1) 利用观测数据向量  $\mathbf{x}(k)$  构造数据矩阵  $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)]$ , 再计算  $\mathbf{X}$  的奇异值分解, 判断数据矩阵的有效秩  $r$ , 得到  $r$  个主奇异值和与之对应的左奇异向量矩阵  $\mathbf{U}_r$ 。极小化问题的最优解为  $\mathbf{W} = \mathbf{U}_r$ 。

(2) 计算自相关矩阵  $\mathbf{C} = \mathbf{X}\mathbf{X}^H$  的特征值分解, 得到与  $r$  个主特征值对应的特征向量矩阵  $\mathbf{U}_r$ , 它便是极小化问题的最优解。

然而, 在实际应用中, 自相关矩阵  $\mathbf{C}$  有可能是随时间变化的, 从而, 其特征值和特征向量也是随时间变化的。由式 (8.5.3) 知, 在时变的情况下, 目标函数  $J(\mathbf{W}(t))$  的矩阵微分为

$$\begin{aligned} dJ(\mathbf{W}(t)) &= -2\text{tr} \left( \mathbf{W}^H(t)\mathbf{C}(t)d\mathbf{W}(t) + [\mathbf{C}(t)\mathbf{W}(t)]^T d\mathbf{W}^*(t) \right) \\ &\quad + \text{tr} \left( [\mathbf{W}^H(t)\mathbf{W}(t)\mathbf{W}^H(t)\mathbf{C}(t) + \mathbf{W}(t)\mathbf{C}(t)\mathbf{W}(t)\mathbf{W}^H(t)]d\mathbf{W}(t) \right. \\ &\quad \left. + [\mathbf{C}(t)\mathbf{W}(t)\mathbf{W}^H(t)\mathbf{W}(t) + \mathbf{W}(t)\mathbf{W}^H(t)\mathbf{C}(t)\mathbf{W}(t)]^T d\mathbf{W}^*(t) \right) \end{aligned}$$

由此得梯度矩阵

$$\begin{aligned} \nabla_{\mathbf{W}} J(\mathbf{W}(t)) &= -2\mathbf{C}(t)\mathbf{W}(t) + \mathbf{C}(t)\mathbf{W}(t)\mathbf{W}^H(t)\mathbf{W}(t) + \mathbf{W}(t)\mathbf{W}^H(t)\mathbf{C}(t)\mathbf{W}(t) \\ &= \mathbf{W}(t)\mathbf{W}^H(t)\mathbf{C}(t)\mathbf{W}(t) - \mathbf{C}(t)\mathbf{W}(t) \end{aligned}$$

式中, 利用了  $\mathbf{W}(t)$  的半正交约束条件  $\mathbf{W}^H(t)\mathbf{W}(t) = \mathbf{I}$ 。

将  $\mathbf{C}(t) = \mathbf{x}(t)\mathbf{x}^H(t)$  代入梯度矩阵公式, 即可得到求解极小化问题的梯度下降算法  $\mathbf{W}(t) = \mathbf{W}(t-1) - \mu \nabla_{\mathbf{W}} J(\mathbf{W}(t))$  如下

$$\mathbf{y}(t) = \mathbf{W}^H(t)\mathbf{x}(t) \quad (8.5.4)$$

$$\mathbf{W}(t) = \mathbf{W}(t-1) + \mu[\mathbf{x}(t) - \mathbf{W}(t-1)\mathbf{y}(t)]\mathbf{y}^H(t) \quad (8.5.5)$$

但是, 这一更新  $\mathbf{W}(t)$  的梯度下降算法收敛比较慢, 跟踪时变子空间的能力也比较差。更好的方法是下面的递推最小二乘算法。

定义指数加权的目标函数

$$J_1(\mathbf{W}(t)) = \sum_{i=1}^t \beta^{t-i} \|\mathbf{x}(i) - \mathbf{W}(t)\mathbf{W}^H(t)\mathbf{x}(i)\|^2 \quad (8.5.6)$$

$$= \sum_{i=1}^t \beta^{t-i} \|\mathbf{x}(i) - \mathbf{W}(t)\mathbf{y}(i)\|^2 \quad (8.5.7)$$

式中,  $0 < \beta \leq 1$  称为遗忘因子, 而  $\mathbf{y}(i) = \mathbf{W}^H(t)\mathbf{x}(i)$ 。

由自适应滤波理论知, 极小化问题  $\min J_1(\mathbf{W})$  的最优解为 Wiener 滤波器

$$\mathbf{W}(t) = \mathbf{C}_{xy}(t)\mathbf{C}_{yy}^{-1}(t) \quad (8.5.8)$$

式中, 互相关矩阵  $C_{xy}(t)$  和自相关矩阵  $C_{yy}(t)$  可以递推

$$C_{xy}(t) = \sum_{i=1}^t \beta^{t-i} \mathbf{x}(i) \mathbf{y}^H(i) = \beta C_{xy}(t-1) + \mathbf{x}(t) \mathbf{y}^H(t) \quad (8.5.9)$$

$$C_{yy}(t) = \sum_{i=1}^t \beta^{t-i} \mathbf{y}(i) \mathbf{y}^H(i) = \beta C_{yy}(t-1) + \mathbf{y}(t) \mathbf{y}^H(t) \quad (8.5.10)$$

### 8.5.2 投影逼近子空间跟踪算法

将式 (8.5.9) 和式 (8.5.10) 代入式 (8.5.8), 并运用矩阵求逆引理, 即可得到投影逼近的子空间跟踪 (projection approximation subspace tracking, PAST) 算法如下。

**算法 8.5.1 投影逼近子空间跟踪 (PAST) 算法** [522]

选择初始化矩阵  $\mathbf{P}(0)$  和  $\mathbf{W}(0)$ 。

对  $t = 1, 2, \dots$ , 计算

$$\begin{aligned} \mathbf{y}(t) &= \mathbf{W}^H(t-1) \mathbf{x}(t) \\ \mathbf{h}(t) &= \mathbf{P}(t-1) \mathbf{y}(t) \\ \mathbf{g}(t) &= \mathbf{h}(t) / [\beta + \mathbf{y}^H(t) \mathbf{h}(t)] \\ \mathbf{P}(t) &= \frac{1}{\beta} \text{Tri}[\mathbf{P}(t-1) - \mathbf{g}(t) \mathbf{h}^H(t)] \\ \mathbf{e}(t) &= \mathbf{x}(t) - \mathbf{W}(t-1) \mathbf{y}(t) \\ \mathbf{W}(t) &= \mathbf{W}(t-1) + \mathbf{e}(t) \mathbf{g}^H(t) \end{aligned}$$

式中,  $\text{Tri}[\mathbf{A}]$  表示只计算矩阵  $\mathbf{A}$  的上 (或下) 三角部分, 然后将上 (或下) 三角部分复制为矩阵的下 (或上) 三角部分。

PAST 算法从数据向量中提取信号子空间, 是一种主分量分析方法。特别地, 若上述算法的第一式用

$$\mathbf{y}(t) = g(\mathbf{W}^H(t-1) \mathbf{x}(t)) \quad (8.5.11)$$

取代, 其中,  $g(\mathbf{z}(t)) = [g(z_1(t)), g(z_2(t)), \dots, g(z_n(t))]^T$  为非线性函数, 则可得到一类称为非线性主分量分析的盲信号分离算法。非线性主分量分析的 LMS 算法和 RLS 算法分别由文献 [377] 和文献 [391] 提出。此外, 若  $r = 1$ , 则 PAST 算法简化为以下算法。

**算法 8.5.2 子空间跟踪的压缩映射 (PASTd) 算法** [522]

选择初始化向量  $\mathbf{d}_i(0)$  和  $\mathbf{w}_i(0)$ 。

对  $t = 1, 2, \dots$ , 计算

$$\mathbf{x}_1(t) = \mathbf{x}(t)$$

对  $i = 1, 2, \dots, r$ , 计算

$$y_i(t) = \mathbf{w}_i^H(t-1) \mathbf{x}_i(t)$$

$$d_i(t) = \beta d_i(t-1) + |y_i(t)|^2$$

$$\begin{aligned} \mathbf{e}_i(t) &= \mathbf{x}_i(t) - \mathbf{w}_i(t-1)y_i(t) \\ \mathbf{w}_i(t) &= \mathbf{w}_i(t-1) + \mathbf{e}_i(t)[y_i^*(t)/d_i(t)] \\ \mathbf{x}_{i+1}(t) &= \mathbf{x}_i(t) - \mathbf{w}_i(t)y_i(t) \end{aligned}$$

PASTd 算法又可进一步推广为秩和子空间二者同时跟踪的算法。对此推广感兴趣的读者可参考文献 [523]。

投影逼近子空间跟踪算法可以对  $\mathbf{W} = \mathbf{U}_r \mathbf{Q}$  进行跟踪。现在考虑信号子空间  $\mathbf{U}_r \mathbf{U}_r^H$  的直接跟踪。由投影矩阵的关系式  $\mathbf{P} = \mathbf{W}\mathbf{W}^H = \mathbf{U}_r \mathbf{U}_r^H$  知, 信号子空间  $\mathbf{U}_r \mathbf{U}_r^H$  的跟踪等价于投影矩阵  $\mathbf{P}$  的跟踪。使用投影矩阵代替式 (8.5.1) 的代价函数中的矩阵  $\mathbf{W}\mathbf{W}^H$ , 即可将投影逼近子空间跟踪的代价函数等价写成

$$J(\mathbf{P}) = E\{\|\mathbf{x} - \mathbf{Px}\|^2\} = \text{tr}(\mathbf{C}) - \text{tr}(\mathbf{CP}) - \text{tr}(\mathbf{CP}^H) + \text{tr}(\mathbf{CPP}^H) \quad (8.5.12)$$

为了使  $\mathbf{P}$  为投影矩阵, 必须对它加幂等矩阵的约束条件  $\mathbf{P}^2 = \mathbf{P}$  和复共轭对称的约束条件  $\mathbf{P}^H = \mathbf{P}$ 。利用这些约束条件可以简化式 (8.5.12)。于是, 便得到直接跟踪信号子空间投影矩阵的约束优化问题

$$\min J(\mathbf{P}) = \min E\{\|\mathbf{x} - \mathbf{Px}\|^2\} = \min[\text{tr}(\mathbf{C}) - \text{tr}(\mathbf{CP})] \quad (8.5.13)$$

约束条件为  $\text{rank}(\mathbf{P}) \neq n$ ,  $\mathbf{P}^2 = \mathbf{P}$  和  $\mathbf{P}^H = \mathbf{P}$ 。这一优化准则是 Utschick 提出的<sup>[488]</sup>。约束条件  $\text{rank}(\mathbf{P}) \neq n$  意味着  $\mathbf{P}$  不可以是非奇异的幂等矩阵 (即单位矩阵)。

在大多数情况下, PAST 算法收敛为半正交矩阵  $\mathbf{W}^H \mathbf{W} = \mathbf{I}$ 。但是, 在某些情况下, PAST 算法将不能收敛, 而呈振荡状态。为了克服 PAST 算法的这一缺点, 文献 [3] 提出了一种正交 PAST 算法: 在 PAST 算法的基础上, 增加一种正交化运算, 以便在每一步迭代都能够保证半正交条件  $\mathbf{W}^H(i)\mathbf{W}(i) = \mathbf{I}$ 。其结果反而简化了整个算法的运算。

### 算法 8.5.3 正交投影逼近子空间跟踪 (OPAST) 算法<sup>[3]</sup>

选择初始化矩阵  $\mathbf{P}(0)$  和  $\mathbf{W}(0)$ 。

对  $t = 1, 2, \dots$ , 计算

$$\mathbf{W}(i) = \mathbf{W}(i-1) + \tilde{\mathbf{p}}^H(i)\mathbf{q}(i) \quad (8.5.14)$$

$$\tau(i) = \frac{1}{\|\mathbf{q}(i)\|_2^2} \left( \frac{1}{\sqrt{1 + \|\mathbf{p}(i)\|_2^2 \|\mathbf{q}(i)\|_2^2}} - 1 \right) \quad (8.5.15)$$

$$\tilde{\mathbf{p}}(i) = \tau(i)\mathbf{W}(i-1)\mathbf{q}(i) + (1 + \tau(i)\|\mathbf{q}(i)\|_2^2)\mathbf{p}(i) \quad (8.5.16)$$

## 8.6 快速子空间分解

从 Krylov 子空间的角度出发, 样本协方差矩阵  $\mathbf{R}$  的信号子空间的跟踪变成  $\mathbf{R}$  的 Rayleigh-Ritz (RR) 向量的跟踪。这一方法的基本出发点是, 样本协方差矩阵  $\hat{\mathbf{R}}$  的主特征向量的张成与  $\hat{\mathbf{R}}$  的 Rayleigh-Ritz (RR) 向量的张成是  $\mathbf{R}$  的信号子空间的渐近等价估计。由于 RR 向量可以利用 Lanczos 算法有效求出, 故可以实现信号子空间的快速分解。

### 8.6.1 Rayleigh-Ritz 逼近

令  $A \in \mathbb{C}^{M \times M}$  为协方差矩阵, 它是 Hermitian 的。考虑样本协方差矩阵  $\hat{A} = \mathbf{X}\mathbf{X}^H/N$ , 其中  $\mathbf{X} = [\mathbf{x}(1), \dots, \mathbf{x}(N)]^T$  为数据矩阵。

令 Hermitian 矩阵  $A$  的特征值分解为

$$A = \sum_{k=1}^M \lambda_k \mathbf{u}_k \mathbf{u}_k^H \quad (8.6.1)$$

其中,  $(\lambda_k, \mathbf{u}_k)$  为  $A$  的第  $k$  个特征值和特征向量, 并假定  $\lambda_1 > \dots > \lambda_d > \lambda_{d+1} = \dots = \lambda_M = \sigma$ 。即是说,  $\{\lambda_k, \mathbf{u}_k\}_{k=1}^d$  为信号特征值和信号特征向量。

现在考虑信号特征值和信号特征向量的 Rayleigh-Ritz (RR) 逼近问题。为此, 先引入以下定义。

**定义 8.6.1** 对于一个  $m$  维子空间  $S^m$ , 若

$$A\mathbf{y}_i^{(m)} - \theta_i^{(m)}\mathbf{y}_i^{(m)} \perp S^m \quad (8.6.2)$$

则分别称  $\theta_i^{(m)}$  和  $\mathbf{y}_i^{(m)}$  是 Hermitian 矩阵  $A$  的 Rayleigh-Ritz (RR) 值和 RR 向量。

**定义 8.6.2** Krylov 矩阵记作  $K^m(A, f)$ , 定义为

$$K^m(A, f) = [f, Af, \dots, A^{m-1}f] \quad (8.6.3)$$

并将其张成

$$\mathcal{K}^m(A, f) = \text{Span}\{f, Af, \dots, A^{m-1}f\} \quad (8.6.4)$$

称作 Krylov 子空间。

对于 RR 值和 RR 向量, 文献 [394] 证明了以下结果。

**引理 8.6.1** 令  $(\theta_i^{(m)}, \mathbf{y}_i^{(m)})$  ( $i = 1, \dots, m$ ) 为子空间  $S^m$  的 RR 值和 RR 向量, 且  $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_m]$  为同一子空间的正交基。如果  $(\alpha_i, \mathbf{u}_i)$  是  $m \times m$  矩阵  $\mathbf{Q}^H A \mathbf{Q}$  的第  $i$  个特征对 (特征值与特征向量), 其中,  $i = 1, \dots, m$ , 则

$$\theta_i^{(m)} = \alpha_i \quad (8.6.5)$$

$$\mathbf{y}_i^{(m)} = \mathbf{Q}\mathbf{u}_i \quad (8.6.6)$$

引理 8.6.1 表明, 一个 Hermitian 矩阵的特征值和特征向量可以分别用 Krylov 子空间的 RR 值和 RR 向量逼近。这种逼近称为 Rayleigh-Ritz 逼近。

Rayleigh-Ritz 逼近的性能用 RR 值和 RR 向量的渐近性质评估: 对  $m > d$ , 它们各自的误差

$$\theta_k^{(m)} - \hat{\lambda}_k = O(N^{-m-d}), \quad k = 1, \dots, d \quad (8.6.7)$$

$$\mathbf{y}_k^{(m)} - \hat{\mathbf{u}}_k = O(N^{-(m-d)/2}), \quad k = 1, \dots, d \quad (8.6.8)$$

式中,  $N$  为数据长度。因此, 一旦  $m \geq d + 2$ , 则有

$$\lim_{N \rightarrow \infty} \sqrt{N}(\mathbf{y}_k^{(m)} - \mathbf{u}_k) = \lim_{N \rightarrow \infty} \sqrt{N}(\hat{\mathbf{u}}_k - \mathbf{u}_k), \quad k = 1, \dots, d \quad (8.6.9)$$

即  $\text{Span}\{\mathbf{y}_1^{(m)}, \dots, \mathbf{y}_d^{(m)}\}$  和  $\text{Span}\{\hat{\mathbf{u}}_1, \dots, \hat{\mathbf{u}}_d\}$  都是信号子空间  $\text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_d\}$  的渐近等价估计, 故 Hermitian 矩阵  $\mathbf{A}$  的信号子空间的求解变成  $\mathbf{A}$  的 RR 特征向量的求解。

### 8.6.2 快速子空间分解算法

进一步地, Lanczos 基通过 Hermitian 矩阵  $\mathbf{A}$  的三对角化, 将  $\mathbf{A}$  的 RR 对 (RR 值和 RR 向量) 与三对角矩阵的特征对 (特征值和特征向量) 紧密联系在一起。

令  $\mathbf{Q}_m = [\mathbf{q}_1, \dots, \mathbf{q}_m]$  是 Lanczos 基, 则由文献 [394] 知

$$\mathbf{Q}_m^H \hat{\mathbf{A}} \mathbf{Q}_m = \mathbf{T}_m = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \alpha_{m-1} & \beta_{m-1} \\ & & & \beta_{m-1} & \alpha_m \end{bmatrix} \quad (8.6.10)$$

其中,  $\mathbf{T}_m$  为  $m \times m$  实三角矩阵。

由于  $\mathbf{Q}_m^H \hat{\mathbf{A}} \mathbf{Q}_m = \mathbf{T}_m$ , 故 RR 值和 RR 向量可以根据  $m \times m$  三对角矩阵  $\mathbf{T}_m$  的特征值分解求出。于是, Krylov 子空间  $\mathcal{K}^m(\hat{\mathbf{A}}, \mathbf{f})$  的 RR 值和 RR 向量可用来逼近样本协方差矩阵  $\hat{\mathbf{A}}$  的期望特征值和特征向量。这一过程称作 Rayleigh-Ritz 逼近, 简称 RR 逼近。Lanczos 算法最吸引人的性质就是: 原来求  $M \times M$  (复值) 样本协方差 (Hermitian) 矩阵  $\hat{\mathbf{A}}$  的期望特征值和特征向量这一较大的问题, 借助 Lanczos 基后, 转变成计算  $m \times m$  (实) 三对角矩阵的特征值分解的较小的问题, 因为  $m$  通常比  $M$  小很多。

RR 值和 RR 向量与 Lanczos 算法密切相关。特别地, RR 值  $\{\theta_k^{(m)}\}$  和 RR 向量  $\{\mathbf{y}_k^{(m)}\}$  可以在 Lanczos 算法的第  $m$  步获得。Lanczos 算法分两种: 实现 Hermitian 矩阵的三对角化的三 Lanczos 迭代和实现任意矩阵双对角化的双 Lanczos 迭代。

#### 算法 8.6.1 三 Lanczos 迭代算法 [198]

给定 Hermitian 矩阵  $\mathbf{A}$ ;  $\mathbf{r}_0 = \mathbf{f}$  (单位范数向量);  $\beta_0 = 1$ ;  $j = 0$ 。

while ( $\beta_j \neq 0$ )

$$\mathbf{q}_{j+1} = \mathbf{r}_j / \beta_j;$$

$$j = j + 1;$$

$$\alpha_j = \mathbf{q}_j^H \mathbf{A} \mathbf{q}_j;$$

$$\mathbf{r}_j = \mathbf{A} \mathbf{q}_j - \alpha_j \mathbf{q}_j - \beta_{j-1} \mathbf{q}_{j-1};$$

$$\beta_j = \|\mathbf{r}_j\|_2$$

end

在三 Lanczos 迭代的第  $m$  步 (即  $j = m$ ), 将得到  $m$  个正交向量  $\{q_1, \dots, q_m\}$ , 它们组成 Krylov 子空间  $\mathcal{K}^m(A, f) = \text{Span}\{f, Af, \dots, A^{m-1}f\}$  的一组正交基  $Q_m$ , 常称为 Lanczos 基。

关于 RR 逼近, Xu 与 Kailath [520] 证明了下面的重要结果。

**定理 8.6.1** 令  $\hat{\lambda}_1 > \dots > \hat{\lambda}_{\hat{d}}$  和  $\hat{u}_1, \dots, \hat{u}_{\hat{d}}$  分别是样本协方差矩阵  $\hat{A}$  的特征值和特征向量, 其中  $\hat{A}$  是利用  $N$  个独立同正态分布  $N(0, A)$  的数据向量计算得到的, 且  $A$  是一个结构化的矩阵 (秩  $d$  矩阵  $+ \sigma I$ )。令  $\lambda_1 > \dots > \lambda_d > \lambda_{d+1} = \dots = \lambda_M = \sigma$  和  $u_1, \dots, u_M$  分别是真实协方差矩阵  $A$  的特征值和特征向量。用  $\theta_1^{(m)} \geq \dots \geq \theta_{\hat{d}}^{(m)}$  和  $y_1^{(m)}, \dots, y_{\hat{d}}^{(m)}$  分别表示从 Krylov 子空间  $\mathcal{K}^m(A, f)$  获得的 RR 值和 RR 向量。若选择  $f$  满足  $f^H \hat{u}_i \neq 0 (1 \leq i \leq d)$ , 则对于  $k = 1, \dots, \hat{d}$ , 下列结果成立:

(1) 若  $m \geq d + 2$ , 则 RR 值  $\theta_k^{(m)}$  逼近它们对应的特征值  $\lambda_k$  的精度为  $O(N^{-(m-d)})$ , 而 RR 向量  $y_k^{(m)}$  逼近它们对应的特征向量  $\hat{u}_k$  的精度为  $O(N^{(m-d)/2})$ , 即

$$\theta_k^{(m)} = \hat{\lambda}_k + O(N^{-(m-d)}) \quad (8.6.11)$$

$$y_k^{(m)} = \hat{u}_k + O(N^{-(m-d)/2}) \quad (8.6.12)$$

(2) 若  $m \geq d + 1$ , 则  $\theta_k^{(m)}$  和  $\hat{\lambda}_k$  是特征值  $\lambda_k$  的渐近等价估计。如果  $m \geq d + 2$ , 则  $y_k^{(m)}$  和  $\hat{e}_k$  也是特征向量  $u_k$  的渐近等价估计。

定理 8.6.1 表明, 从三 Lanczos 迭代的第  $m (> d + 1)$  步得到的  $d$  个比较大的 RR 值可以用来代替信号特征值。但是, 还需要先估计  $d$ 。为此, 构造检验统计量

$$\phi_{\hat{d}} = N(M - \hat{d}) \log \left[ \frac{\sqrt{\frac{1}{M-\hat{d}} \left( \|\hat{A}\|_F^2 - \sum_{k=1}^{\hat{d}} \theta_k^{(m)2} \right)}}{\frac{1}{M-\hat{d}} \left( \text{tr}(\hat{A}) - \sum_{k=1}^{\hat{d}} \theta_k^{(m)} \right)} \right] \quad (8.6.13)$$

其中

$$\text{tr}(\hat{A}) = \sum_{k=1}^M \hat{\lambda}_k, \quad \|\hat{A}\|_F^2 = \sum_{k=1}^M \hat{\lambda}_k^2 \quad (8.6.14)$$

文献 [520] 提出的快速子空间分解算法如下。

### 算法 8.6.2 快速子空间分解算法 (三 Lanczos 迭代)

步骤 1 适当选择  $r_0 = f$ , 它满足定理 8.6.1 中的条件。令  $m = 1, \beta_0 = \|r_0\| = 1$  和  $\hat{d} = 1$ 。

步骤 2 执行第  $m$  次三 Lanczos 迭代 (算法 8.6.1)。

步骤 3 利用算法 8.6.1 得到的  $\alpha$  和  $\beta$  值, 构造  $m \times m$  三对角矩阵  $T_m$ , 并求其特征值, 得到 RR 值  $\theta_i^{(m)}, i = 1, 2, \dots, m$ 。

步骤 4 对  $\hat{d} = 1, \dots, m-1$ , 用式 (8.6.13) 计算检验统计量  $\phi_{\hat{d}}$ 。若  $\phi_{\hat{d}} \leq \gamma_{\hat{d}} c(N)$ , 则令  $d = \hat{d}$  (接受  $H_0$  假设), 并转到步骤 5; 否则, 令  $m = m+1$ , 并返回步骤 2。

步骤 5 计算  $m \times m$  三对角矩阵  $\mathbf{T}_m$  的特征值分解, 得到与 Krylov 子空间  $\mathcal{K}^m(\hat{\mathbf{A}}, \mathbf{f})$  相关联的  $d$  个主 RR 向量  $\mathbf{y}_k^{(m)}$ 。最后的信号子空间估计为  $\text{Span}\{\mathbf{y}_1^{(m)}, \dots, \mathbf{y}_d^{(m)}\}$ 。

三 Lanczos 迭代仅适用于 Hermitian 矩阵的三角化, 不能够用于非正方的矩阵。下面考虑对  $N \times M$  数据矩阵  $\mathbf{X}_N$  直接求 RR 向量。

### 算法 8.6.3 双 Lanczos 迭代<sup>[198]</sup>

给定  $\mathbf{X}_N; \mathbf{p}_0 = \mathbf{f}$  (单位范数向量);  $\beta_0 = 1; \mathbf{u}_0 = \mathbf{0}; j = 0$ 。

while  $\beta_j^{(b)} \neq 0$

$$\mathbf{v}_{j+1} = \mathbf{p}_j / \beta_j^{(b)};$$

$$j = j + 1;$$

$$\mathbf{r}_j = \mathbf{X}_N \mathbf{v}_j - \beta_{j-1}^{(b)} \mathbf{u}_{j-1};$$

$$\alpha_j^{(b)} = \|\mathbf{r}_j\|;$$

$$\mathbf{u}_j = \mathbf{r}_j / \alpha_j^{(b)};$$

$$\mathbf{p}_j = \mathbf{X}_N^H \mathbf{u}_j - \alpha_j^{(b)} \mathbf{v};$$

$$\beta_j^{(b)} = \|\mathbf{p}_j\|;$$

end

类似于三 Lanczos 迭代, 双 Lanczos 迭代给出左 Lanczos 基  $\mathbf{U}_j = [\mathbf{u}_1, \dots, \mathbf{u}_j]$ , 右 Lanczos 基  $\mathbf{V}_j = [\mathbf{v}_1, \dots, \mathbf{v}_j]$  以及双对角矩阵

$$\mathbf{B}_j = \begin{bmatrix} \alpha_1^{(b)} & \beta_1^{(b)} & & \\ & \alpha_2^{(b)} & \ddots & \\ & & \ddots & \beta_{j-1}^{(b)} \\ & & & \alpha_j^{(b)} \end{bmatrix} \quad (8.6.15)$$

下面的定理表明, 对矩形的数据矩阵  $\mathbf{X}_N$  使用双 Lanczos 迭代等价于对样本协方差矩阵  $\hat{\mathbf{A}}$  使用三 Lanczos 迭代。

**定理 8.6.2**<sup>[519]</sup> 考查任一  $N \times M$  矩阵  $\mathbf{X}_N$ 。对  $\mathbf{X}_N^H \mathbf{X}_N$  应用三 Lanczos 迭代, 并对  $\mathbf{X}_N$  使用双 Lanczos 迭代。如果两个算法使用相同的初始值, 即如果  $\mathbf{q}_1 = \mathbf{v}_1$ , 则

$$(1) \mathbf{Q}_j = \mathbf{V}_j, \quad j = 1, \dots, M;$$

$$(2) \mathbf{T}_j = \mathbf{B}_j^H \mathbf{B}_j, \quad j = 1, \dots, M.$$

根据上述定理描述的等价性, 只要将算法 8.6.2 中的三 Lanczos 迭代换成双 Lanczos 迭代, 即可得到基于双 Lanczos 迭代的快速子空间分解算法。

### 算法 8.6.4 快速子空间分解算法 (双 Lanczos 迭代)<sup>[519]</sup>

步骤 1 适当选择  $\mathbf{r}_0 = \mathbf{f}$ , 它满足定理 8.6.1 中的条件。令  $m = 1, \beta_0 = \|\mathbf{r}_0\| = 1$  和  $\hat{d} = 1$ 。

步骤 2 执行第  $m$  次双 Lanczos 迭代 (算法 8.6.3)。

步骤3 利用算法8.6.3得到的 $\alpha$ 和 $\beta$ 值,构造 $m \times m$ 双对角矩阵 $B_m$ ,并求其奇异值奇异值 $\theta_i^{(m)}, i = 1, \dots, m$ 。

步骤4 对 $\hat{d} = 1, \dots, m-1$ ,计算检验统计量

$$\phi_{\hat{d}} = \sqrt{N} |\log(\hat{\sigma}_{\hat{d}}/\hat{\sigma}_{\hat{d}+1})| \quad (8.6.16)$$

式中 $\hat{\sigma}_i = \frac{1}{M-j} \left( \left\| \frac{1}{\sqrt{N}} \mathbf{X}_N \right\|_2^2 - \sum_{i=1}^j (\theta_i^{(m)})^2 \right)$ 。若 $\phi_{\hat{d}} < \gamma_{\hat{d}} \sqrt{\log N}$ ,则令 $d = \hat{d}$ (接受 $H_0$ 假设),并转到步骤5;否则,令 $m = m + 1$ ,并返回步骤2。

步骤5 计算 $m \times m$ 双对角矩阵 $B_m$ 的奇异值分解,得到与Krylov子空间 $\mathcal{K}^m(\hat{A}, f)$ 相关联的 $d$ 个主RR右奇异向量 $v_k^{(m)}$ 。最后的信号子空间估计为 $\text{Span}\{v_1^{(m)}, \dots, v_d^{(m)}\}$ 。

文献[519]介绍了快速子空间分解在信号处理和无线通信中的应用。

## 本章小结

本章从子空间的代数关系和几何关系入手,介绍了子空间的分析理论与方法:

- (1) 矩阵基本子空间(行空间、列空间和零空间)的性质与构造方法;
- (2) 信号子空间分析方法和噪声子空间分析方法。

为了适应实时信号处理的需要,本章还专门讨论了子空间的实时跟踪与更新的下列方法:

- (1) 基于优化理论的子空间跟踪;
- (2) 快速子空间分解。

特别地,围绕基于优化理论的子空间跟踪,重点介绍了Grassmann流形、Stiefel流形和投影逼近子空间跟踪等典型方法。

## 习题

**8.1** 令 $V$ 是所有 $2 \times 2$ 矩阵的向量空间,证明子空间

$$W = \left\{ \mathbf{A} : \mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, ad = 0, bc = 0 \right\}$$

不是 $V$ 的子空间。

**8.2** 令 $W$ 是所有 $3 \times 3$ 斜对称矩阵的集合。证明 $W$ 是所有 $3 \times 3$ 矩阵的向量空间 $V$ 的一个子空间,并求其张成子空间的基。

**8.3** 令 $V$ 是 $2 \times 2$ 矩阵的向量空间,并且

$$W = \left\{ \mathbf{A} : \mathbf{A} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}, a, b \text{为任意实数} \right\}$$

是  $V$  的一个子空间。若

$$\mathbf{B}_1 = \begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 & 3 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{B}_3 = \begin{bmatrix} 0 & 3 \\ 2 & 0 \end{bmatrix}$$

- (1) 证明矩阵集合  $\{\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3\}$  线性相关，并将  $\mathbf{B}_3$  表示为  $\mathbf{B}_1$  和  $\mathbf{B}_2$  的线性组合；
- (2) 证明  $\{\mathbf{B}_1, \mathbf{B}_2\}$  是一个线性无关的矩阵集合。

**8.4** 令  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p$  是有限维的非零向量空间  $V$  的向量，并且  $S = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p\}$  为一向量集合。判断下列结果的真与假：

- (1)  $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p$  的所有线性组合的集合为一向量空间；
- (2) 若  $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{p-1}\}$  线性无关，则  $S$  也是线性无关的向量集合；
- (3) 若向量集合  $S$  线性无关，则  $S$  是向量空间  $V$  的一组基；
- (4) 若  $V = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p\}$ ，则  $S$  的某个子集是  $V$  的一组基；
- (5) 若  $\dim(V) = p$  和  $V = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p\}$ ，则向量集合  $S$  不可能线性相关。

**8.5** 判断下列结果是否为真：

- (1) 矩阵  $\mathbf{A}$  的行空间与  $\mathbf{A}^T$  的列空间相同。
- (2) 矩阵  $\mathbf{A}$  的行空间和列空间的维数相同，即使  $\mathbf{A}$  不是正方矩阵。
- (3) 矩阵  $\mathbf{A}$  的行空间和零空间的维数之和等于  $\mathbf{A}$  的行数。
- (4) 矩阵  $\mathbf{A}^T$  的行空间与  $\mathbf{A}$  的列空间相同。

**8.6** 令  $\mathbf{A}$  为  $m \times n$  矩阵，在子空间  $\text{Row}(\mathbf{A})$ ,  $\text{Col}(\mathbf{A})$ ,  $\text{Null}(\mathbf{A})$ ,  $\text{Row}(\mathbf{A}^T)$ ,  $\text{Col}(\mathbf{A}^T)$  和  $\text{Null}(\mathbf{A}^T)$  内，有几个不同的子空间？哪些位于  $\mathbb{R}^m$  空间，哪些位于  $\mathbb{R}^n$  空间？

**8.7** 证明下列向量集合  $W$  为向量子空间，或举反例说明它不是向量子空间：

$$(1) W = \left\{ \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} : \begin{array}{l} 2a + b = c \\ a + b + c = d \end{array} \right\}; \quad (2) W = \left\{ \begin{bmatrix} a - b \\ 3b \\ 3a - 2b \\ a \end{bmatrix} : a, b \text{ 为实数} \right\}$$

$$(3) W = \left\{ \begin{bmatrix} 2a + 3b \\ c + a - 2b \\ 4c + a \\ 3c - a - b \end{bmatrix} : a, b, c \text{ 为实数} \right\}$$

**8.8** 已知

$$\mathbf{A} = \begin{bmatrix} 8 & 2 & 9 \\ -3 & -2 & -4 \\ 5 & 0 & 5 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} 2 \\ 1 \\ -2 \end{bmatrix}$$

判断  $\mathbf{w}$  是在列空间  $\text{Col}(\mathbf{A})$  还是零空间  $\text{Null}(\mathbf{A})$ ？

**8.9** 在统计理论中常常要求矩阵是满秩的。若矩阵  $\mathbf{A}$  是一个  $m \times n$  矩阵，其中， $m > n$ ，试解释  $\mathbf{A}$  满秩的条件是其列线性无关。

**8.10** 一个  $7 \times 10$  矩阵能否有二维的零空间？

**8.11** 试证明  $\mathbf{v}$  在矩阵  $\mathbf{A}$  的列空间  $\text{Col}(\mathbf{A})$  内，若  $\mathbf{Av} = \lambda\mathbf{v}$ ，且  $\lambda \neq 0$ 。

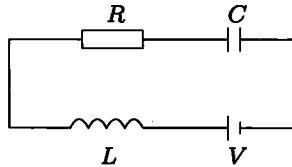
**8.12** 令  $V_1$  和  $V_2$  的列向量分别是  $\mathbb{C}^n$  的同一子空间的正交基, 证明  $V_1 V_1^H \mathbf{x} = V_2 V_2^H \mathbf{x}, \forall \mathbf{x}$ 。

**8.13** 令  $V$  是一子空间, 且  $S$  是  $V$  的生成元或张成集合。已知

$$S = \left\{ \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}, \begin{bmatrix} -1 \\ -2 \\ -3 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix} \right\}$$

求  $V$  的基, 并计算  $\dim(V)$ 。

**8.14** 题图 8.14 中的电路由电阻  $R$  (欧姆)、电感  $L$  (亨利) 和电容  $C$  (法拉) 和初始电压源  $V$  组成。令  $b = R/(2L)$ , 并假定  $R, L, C$  的值使得  $b$  的数值也等于  $1/\sqrt{LC}$  (例如, 伏特计就是这种情况)。令  $v(t)$  是在时间  $t$  测得的电容两端的瞬时电压, 而  $H$  是将  $v(t)$  映射为  $Lv''(t) + Rv'(t) + (1/C)v(t)$  的线性变换的零空间。可以证明,  $v$  位于零空间  $H$  内, 并且  $H$  由所有具有形式  $v(t) = e^{-bt}(c_1 + c_2 t)$  的函数组成。求零空间  $H$  的一组基。



题图 8.14 电路图

**8.15** 一质量为  $m$  的物体挂在一弹簧的末端。如果压紧该弹簧, 然后再释放, 这一质量—弹簧系统就会开始振荡。假定质量  $m$  与其静止位置的位移  $y(t)$  由函数

$$y(t) = c_1 \cos(\omega t) + c_2 \sin(\omega t)$$

描述, 其中,  $\omega$  是一个与质量  $m$  和弹簧有关的常数。固定  $\omega$ , 令  $c_1$  和  $c_2$  任意。

(1) 证明: 描述质量—弹簧系统振荡的函数  $y(t)$  的集合为一向量空间  $V$ 。

(2) 求向量空间  $V$  的一组基。

**8.16** 令

$$W = \left\{ \begin{bmatrix} a \\ b \\ c \end{bmatrix} : a - 3b - c = 0 \right\}$$

证明  $W$  是  $\mathbb{R}^3$  的一个子空间。

**8.17** 已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 5 & 7 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 4 \\ 2 & -7 \\ 5 & -1 \end{bmatrix}$$

求列空间  $\text{Col}(\mathbf{A})$  和  $\text{Col}(\mathbf{B})$  之间的主角的余弦。

**8.18** 假定  $\mathbf{X}$  和  $\mathbf{Y} \in \mathbb{C}^{m \times n}$  ( $m \geq n$ ), 且  $\mathbf{X}^H \mathbf{X} = \mathbf{Y}^H \mathbf{Y} = \mathbf{I}_n$ 。证明: 子空间  $\text{Col}(\mathbf{X})$  和  $\text{Col}(\mathbf{Y})$  之间的第  $i$  主角的余弦是矩阵  $\mathbf{X}^H \mathbf{Y}$  的第  $i$  个奇异值 (按递减顺序排列)。

**8.19** 令矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$ , 证明:

$$\min_{\text{rank}(\mathbf{X}) \leq k} \|\mathbf{A} - \mathbf{X}\|_F = \left( \sum_{i=k+1}^n \sigma_i^2 \right)^{1/2}$$

式中,  $\mathbf{X} \in \mathbb{C}^{m \times n}$ ,  $k < n$ , 且  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$  是矩阵  $\mathbf{A}$  的  $n$  个奇异值。

**8.20** 假定  $\mathbf{X}, \mathbf{Y} \in \mathbb{C}^{m \times n}$  ( $m \geq n$ ), 并且  $\mathbf{X}^H \mathbf{X} = \mathbf{Y}^H \mathbf{Y} = \mathbf{I}_n$ 。定义子空间  $S_1 = \text{Col}(\mathbf{X})$  和  $S_2 = \text{Col}(\mathbf{Y})$  之间的距离为

$$d(S_1, S_2) = \|\mathbf{P}_{S_1} - \mathbf{P}_{S_2}\|_F$$

(1) 求投影矩阵  $\mathbf{P}_{S_1}$  和  $\mathbf{P}_{S_2}$ ;

(2) 证明:

$$d(S_1, S_2) = (1 - \sigma_{\min}(\mathbf{X}^H \mathbf{Y}))^{1/2}$$

其中,  $\sigma_{\min}(\mathbf{X}^H \mathbf{Y})$  是矩阵  $\mathbf{X}^H \mathbf{Y}$  的最小奇异值。

**8.21** 令  $\mathbf{A}$  和  $\mathbf{B}$  是两个  $m \times n$  矩阵, 并且  $m \geq n$ 。证明

$$\min_{\mathbf{Q}^T \mathbf{Q} = \mathbf{I}_n} \|\mathbf{A} - \mathbf{BQ}\|_F^2 = \sum_{i=1}^n [(\sigma_i(\mathbf{A}))^2 - 2\sigma_i(\mathbf{B}^T \mathbf{A}) + (\sigma_i(\mathbf{B}))^2]$$

式中,  $\sigma_i(\mathbf{A})$  是矩阵  $\mathbf{A}$  的第  $i$  个奇异值。

**8.22** 假定  $T$  是一个一对一线性变换, 并且  $T(\mathbf{u}) = T(\mathbf{v})$  总是意味着  $\mathbf{u} = \mathbf{v}$ 。证明: 若像的集合  $\{T(\mathbf{u}_1), T(\mathbf{u}_2), \dots, T(\mathbf{u}_p)\}$  线性相关, 则向量集合  $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p\}$  线性相关。(注: 该命题表明, 一个一对一线性变换将线性无关的向量集合映射为线性无关的向量集合。)

**8.23** 已知矩阵

$$\mathbf{A} = \begin{bmatrix} 2 & 5 & -8 & 0 & 17 \\ 1 & 3 & -5 & 1 & 5 \\ -3 & -11 & 19 & -7 & -1 \\ 1 & 7 & -13 & 5 & -3 \end{bmatrix}$$

试求其列空间、行空间和零空间的基。

**8.24** 已知

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & -1 & 6 & -8 \\ 1 & -2 & -4 & 3 & -2 \\ 7 & -8 & -10 & -3 & 10 \\ 4 & -5 & -7 & 0 & 4 \end{bmatrix}$$

和

$$\mathbf{B} = \begin{bmatrix} 1 & -2 & -4 & 3 & -2 \\ 0 & 3 & 9 & -12 & 12 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

是两个行等价的矩阵，试求

- (1) 矩阵  $\mathbf{A}$  的秩和零空间  $\text{Null}(\mathbf{A})$  的维数；
- (2) 列空间  $\text{Col}(\mathbf{A})$  和行空间  $\text{Row}(\mathbf{A})$  的基；
- (3) 如果希望求零空间  $\text{Null}(\mathbf{A})$  的基，下一步应该执行什么运算？
- (4) 在  $\mathbf{A}^T$  的行阶梯型中有几个主元列？

**8.25** [434] 令  $\mathbf{P}$  是一投影算子，并且  $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m]$  的列向量组成值域  $\text{Range}(\mathbf{P})$  的一组基。试解释为什么总是存在  $\text{Null}(\mathbf{P})^\perp$  的一组基  $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m]$ ，使得  $\mathbf{u}_i^H \mathbf{v}_j = 0$ ？满足这一正交关系的矩阵  $\mathbf{U}$  和  $\mathbf{V}$  称为双正交的。令  $S$  和  $H$  是两个子空间，它们具有相同维数  $m$ 。是否总是存在双正交的  $\mathbf{U}$  和  $\mathbf{V}$ ，使得  $\mathbf{U}$  的列向量组成子空间  $S$  的一组基，而  $\mathbf{V}$  的列向量是  $H$  的一组基？

**8.26** 令  $\mathbf{A}$  是一个  $n \times n$  对称矩阵，证明：

- (1)  $(\text{Col}\mathbf{A})^\perp = \text{Null}(\mathbf{A})$ 。
- (2)  $\mathbb{R}^n$  内的每一个向量  $\mathbf{x}$  都可以写作  $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{z}$ ，式中， $\hat{\mathbf{x}} \in \text{Col}(\mathbf{A})$ ， $\mathbf{z} \in \text{Null}(\mathbf{A})$ 。

**8.27** 考虑一码分多址 (CDMA) 系统，它共有  $K$  个用户。假定用户 1 为期望用户，其特征波形向量  $\mathbf{s}_1$  为已知，并满足单位能量条件  $\langle \mathbf{s}_1, \mathbf{s}_1 \rangle = \mathbf{s}_1^T \mathbf{s}_1 = 1$ 。现有一接收机的观测数据向量为  $\mathbf{y}(n)$ ，它包含了  $K$  个用户信号的线性混合。为了检测期望用户的信号，希望设计一多用户检测器  $\mathbf{c}_1$ ，使检测器的输出能量最小化。若多用户检测器服从约束条件  $\mathbf{c}_1 = \mathbf{s}_1 + \mathbf{U}_i \mathbf{w}$ ，其中， $\mathbf{U}_i$  称为干扰子空间，亦即它的列张成干扰子空间。求干扰子空间  $\mathbf{U}_i$ 。

# 第9章 投影分析

在许多工程应用(例如无线通信、雷达、声纳、时间序列分析和信号处理等)中,许多问题的最优求解都可归结为:提取某个所希望的信号,而抑制掉其他所有干扰、杂波或者噪声。投影是解决这类问题的一个极为重要的数学工具。

投影分为正交投影和斜投影两类。本章将系统介绍向量与矩阵的投影分析。首先,将给出投影与正交投影的基本知识。其次,将分别从数学和信号处理的角度,引出投影矩阵与正交投影矩阵的定义公式。然后,将围绕投影矩阵与正交投影矩阵的应用,展开多方面的讨论。特别地,将介绍投影矩阵与正交投影矩阵的递推计算,以及这种递推计算的应用。最后,将聚焦于斜投影矩阵及其有趣的典型应用。

## 9.1 投影与正交投影

在学习力学的过程中,我们已经熟悉图 9.1.1 所示一方块物体在斜面上重力的分解。

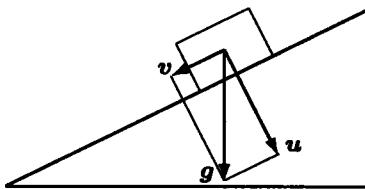


图 9.1.1 物体重力的分解

图 9.1.1 中,物体的重力  $g$  垂直向下,它可以分解为两个分量:一个与斜面垂直,为物体的压力  $u$ ;另一个与斜面平行,为物体的下滑力,即有  $g = u + v$ 。由于压力  $u$  与下滑力  $v$  相互垂直,所以重力的分解  $g = u + v$  属于所谓的正交分解。

如果定义斜面的法线(下指)向量为  $w$ ,则压力  $u$  可视为重力  $g$  在  $w$  上的投影:

$$u = \text{Proj}_w g$$

而与压力  $u$  垂直的下滑力  $v$  即是重力  $g$  在  $w$  上的正交投影,记作

$$v = \text{Proj}_{w^\perp} g$$

### 9.1.1 投影定理

更一般地，我们来考虑向量子空间中的投影与正交投影。

众所周知，在初等几何中，一个点到一直线的最短距离为垂直距离。推而广之，从一个点到一子空间的最短距离是与该子空间正交的距离。如果  $\mathbf{x} \in H$ ，而  $M$  是向量空间  $H$  的一个子空间，并且  $\mathbf{x}$  不在子空间  $M$  内，那么最短距离问题就是求向量  $\mathbf{y} \in M$  使得向量  $\mathbf{x} - \mathbf{y}$  的长度最短。如果  $\hat{\mathbf{x}} \in M$  使得 Euclidean 范数  $\|\mathbf{x} - \hat{\mathbf{x}}\|_2$  最小，则  $\hat{\mathbf{x}}$  称为向量  $\mathbf{x}$  在子空间  $M$  上的投影。类似地，向量  $\mathbf{x}$  到子空间  $M$  的正交补  $M^\perp$  上的投影则称为正交投影。

令  $M$  是  $H$  的一个子空间。已知  $V$  中的向量  $\mathbf{x}$ ，现希望求向量  $\hat{\mathbf{x}} \in M$  使得

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{y} \in M \quad (9.1.1)$$

子空间  $M$  中满足不等式 (9.1.1) 的向量  $\hat{\mathbf{x}}$  称为向量  $\mathbf{x}$  在子空间  $M$  上的投影或向量  $\mathbf{x}$  的最小二乘逼近。直观上，向量  $\hat{\mathbf{x}}$  是子空间  $M$  中与  $\mathbf{x}$  距离最近的向量。

显然，上述问题的求解过程本质上与最小二乘问题等价。需要注意的是，若  $M$  是  $H$  的一个无穷维的子空间，那么向量  $\mathbf{x}$  到子空间  $M$  的投影就有可能不存在。但是，如果  $M$  是有限维的子空间，向量  $\mathbf{x}$  到该子空间的投影就一定存在，并且唯一。

**定理 9.1.1 (投影定理)** 令  $H$  是向量空间，而  $M$  是  $H$  内的  $n$  维子空间。若对于  $H$  中的向量  $\mathbf{x}$ ，在子空间  $M$  内有一个向量  $\hat{\mathbf{x}}$ ，使得  $\mathbf{x} - \hat{\mathbf{x}}$  与  $M$  中的每一个向量  $\mathbf{y}$  都满足正交条件，即

$$\langle \mathbf{x} - \hat{\mathbf{x}}, \mathbf{y} \rangle = 0 \quad (9.1.2)$$

则不等式  $\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2$  对于所有向量  $\mathbf{y} \in M$  成立，并且等号仅当  $\mathbf{y} = \hat{\mathbf{x}}$  时成立。

**证明** 计算向量范数的平方，直接得

$$\begin{aligned} \|\mathbf{x} - \mathbf{y}\|_2^2 &= \|(\mathbf{x} - \hat{\mathbf{x}}) + (\hat{\mathbf{x}} - \mathbf{y})\|_2^2 \\ &= (\mathbf{x} - \hat{\mathbf{x}})^T(\mathbf{x} - \hat{\mathbf{x}}) + 2(\mathbf{x} - \hat{\mathbf{x}})^T(\hat{\mathbf{x}} - \mathbf{y}) + (\hat{\mathbf{x}} - \mathbf{y})^T(\hat{\mathbf{x}} - \mathbf{y}) \end{aligned}$$

由于  $(\mathbf{x} - \hat{\mathbf{x}})^T \mathbf{y} = \langle \mathbf{x} - \hat{\mathbf{x}}, \mathbf{y} \rangle = 0$  对于每一个向量  $\mathbf{y} \in M$  均成立，故对于  $M$  中的向量  $\hat{\mathbf{x}}$  自然也成立，即  $(\mathbf{x} - \hat{\mathbf{x}})^T \hat{\mathbf{x}} = 0$ 。于是，有

$$(\mathbf{x} - \hat{\mathbf{x}})^T(\hat{\mathbf{x}} - \mathbf{y}) = (\mathbf{x} - \hat{\mathbf{x}})^T \hat{\mathbf{x}} - (\mathbf{x} - \hat{\mathbf{x}})^T \mathbf{y} = 0$$

由以上两式立即有  $\|\mathbf{x} - \mathbf{y}\|_2^2 = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \|\hat{\mathbf{x}} - \mathbf{y}\|_2^2 \geq \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$ ，等号仅当  $\mathbf{y} = \hat{\mathbf{x}}$  成立。■

定理 9.1.1 表明，向量  $\mathbf{x}$  到有限维子空间  $M$  的投影  $\hat{\mathbf{x}}$  唯一存在。

向量  $\mathbf{x}$  到子空间  $M$  上的投影  $\hat{\mathbf{x}}$  常用数学符号缩写为

$$\hat{\mathbf{x}} = \mathbf{P}_M \mathbf{x}, \quad \hat{\mathbf{x}} \in M \quad (9.1.3)$$

其中,  $P_M$  代表到闭子空间  $M$  上的投影映射, 习惯称为投影算子。如图 9.1.2 所示,  $(x - \hat{x})$  是从  $x$  到  $M$  的垂线。

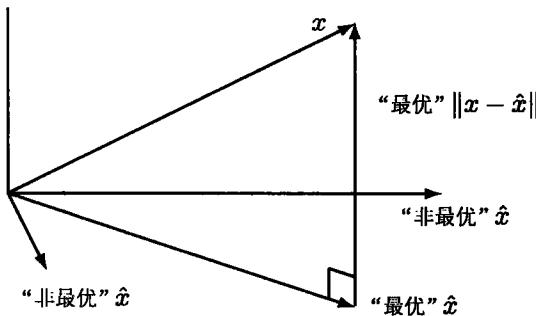


图 9.1.2 投影定理的几何解释

给定一个向量空间  $H$ , 一个子空间  $M$  和一个元素  $x \in H$ , 定理 9.1.1 表明,  $M$  内与  $x$  最接近的元素 (即  $\hat{x} \in M$ ) 是唯一的, 它满足方程

$$\langle x - \hat{x}, y \rangle = 0, \quad \forall y \in M \quad (9.1.4)$$

式 (9.1.4) 给出了  $x$  在子空间  $M$  内的最佳 (均方) 预测子  $\hat{x} = P_M x$  应该满足的方程, 称之为预测方程。

令  $M^\perp$  表示子空间  $M$  的正交补。投影  $P_M x$  具有以下性质<sup>[68]</sup>:

- (1) 齐次性  $P_M(\alpha x + \beta y) = \alpha P_M x + \beta P_M y, \quad x, y \in H; \quad \alpha, \beta \in C.$
- (2) 直角三角形等式  $\|x\|^2 = \|P_M x\|^2 + \|(I - P_M)x\|^2.$
- (3) 正交分解 每一个  $x \in M$  都具有以下的唯一表示

$$x = P_M x + (I - P_M)x \quad (9.1.5)$$

即  $x$  可以唯一分解成  $M$  的分量  $P_M x$  与  $M^\perp$  的分量  $(I - P_M)x$  之和。

- (4) 收敛性  $P_M x_n \rightarrow P_M x$ , 若  $\|x_n - x\| \rightarrow 0$ .
- (5) 自投影  $x \in M$  当且仅当  $P_M x = x$ .
- (6) 正交投影  $x \in M^\perp$  当且仅当  $P_M x = 0$ .
- (7) 包容性  $M_1 \subseteq M_2$ , 当且仅当  $P_{M_1} P_{M_2} x = P_{M_1} x$  对所有  $x \in H$  恒成立。

### 9.1.2 均方估计

一个集合  $M \subseteq L_2$  称为正交随机变量系, 若对每个  $\xi, \eta \in M (\xi \neq \eta)$  均有  $\xi \perp \eta$ 。特别地, 若对每一个  $\xi \in M$  均有  $\|\xi\| = 1$ , 则称  $M$  是一标准正交系。

许多工程问题都可以归结为: 给定  $n$  个数据向量  $\eta_1, \dots, \eta_n$ , 希望找出  $n$  个常数  $a_1, \dots, a_n$ , 使得用线性组合  $\hat{\xi} = \sum_{i=1}^n a_i \eta_i$  拟合未知的随机变量  $\xi$  时, 拟合 (或估计) 误差

向量

$$\epsilon = \xi - \hat{\xi} = \xi - \sum_{i=1}^n a_i \eta_i \quad (9.1.6)$$

的均方值

$$P = E \left\{ \left| \xi - \sum_{i=1}^n a_i \eta_i \right|^2 \right\} \quad (9.1.7)$$

为最小。在参数估计理论中，称这样的估计值为  $\xi$  的最佳线性均方估计<sup>[349]</sup>。

**定理 9.1.2** ( $L_2$  空间的投影定理) <sup>[349]</sup> 若数据向量  $\eta_1, \dots, \eta_n$  组成标准正交系，则随机变量  $\xi$  的最佳均方估计由

$$\hat{\xi} = \sum_{i=1}^n \langle \xi, \eta_i \rangle \eta_i \quad (9.1.8)$$

确定。

**定义 9.1.1** (线性流形) 令  $M$  是  $H$  空间的子空间， $L$  代表  $M$  内的有限个元素的所有线性组合的全体，即  $L = \left\{ \xi : \xi = \sum_{i=1}^n a_i \eta_i, \eta_i \in M \right\}$ ，称  $L$  是由  $M$  张成的线性流形。

下面解释最佳线性均方估计式 (9.1.8) 式的几何意义。考虑以下分解

$$\xi = \hat{\xi} + (\xi - \hat{\xi}) \quad (9.1.9)$$

可以证明，上述分解就是正交分解，即  $\hat{\xi} \perp (\xi - \hat{\xi})$ 。为此，只要等价证明  $E\{\hat{\xi}(\xi - \hat{\xi})\} = 0$  即可。证明是简单的，因为

$$\begin{aligned} E\{\hat{\xi}(\xi - \hat{\xi})\} &= E \left\{ \left[ \sum_{j=1}^n \langle \xi, \eta_j \rangle \eta_j \right] \left[ \xi - \sum_{i=1}^n \langle \xi, \eta_i \rangle \eta_i \right] \right\} \\ &= E \left\{ \xi \sum_{j=1}^n \langle \xi, \eta_j \rangle \eta_j \right\} - E \left\{ \sum_{i=1}^n \langle \xi, \eta_i \rangle \eta_i \sum_{j=1}^n \langle \xi, \eta_j \rangle \eta_j \right\} \\ &= \sum_{j=1}^n [E\{\langle \xi, \eta_j \rangle\}]^2 - \sum_{i=1}^n \sum_{j=1}^n E\{\langle \xi, \eta_i \rangle\} E\{\langle \xi, \eta_j \rangle\} E\{\langle \eta_i, \eta_j \rangle\} \\ &= \sum_{j=1}^n [E\{\langle \xi, \eta_j \rangle\}]^2 - \sum_{i=1}^n [E\{\langle \xi, \eta_i \rangle\}]^2 \\ &= 0 \end{aligned}$$

在得到倒数第二式时，利用了  $\eta_1, \dots, \eta_n$  的标准正交假设  $E\{\langle \eta_i, \eta_j \rangle\} = \delta_{ij}$ ，其中  $\delta_{ij}$  为 Kronecker  $\delta$  函数。

图 9.1.3 画出了式 (9.1.9) 的正交分解，其中， $i_1$  和  $i_2$  分别为长度为 1 的向量。

很自然地，称  $\xi - \hat{\xi}$  垂直于线性流形  $L$ ，并称  $\hat{\xi}$  是  $\xi$  在线性流形  $L$  上的投影。因此，常用投影  $\text{Proj}\{\xi | \eta_1, \dots, \eta_n\}$  表示已知数据向量  $\eta_1, \dots, \eta_n$  情况下未知参数向量  $\xi$  的均

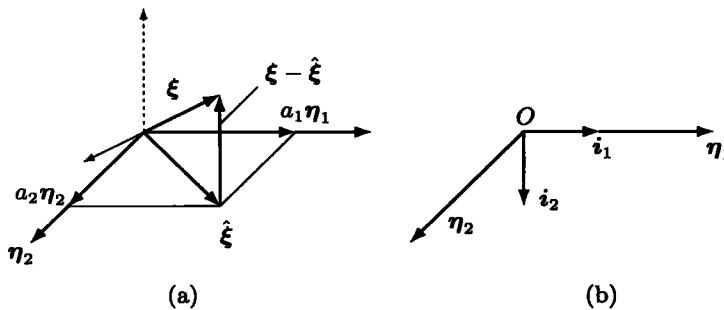


图 9.1.3 正交分解

方估计。这就是为什么把定理 9.1.2 称为  $L_2$  空间的投影定理的缘故。此外，有时也使用符号  $\hat{E}\{\xi|\eta_1, \dots, \eta_n\}$  表示由已知数据向量  $\eta_1, \dots, \eta_n$  求得的  $\xi$  均方估计。

$L_2$  空间的投影定理提供了求最佳线性均方估计的方法，但要求所给定的全部数据  $\eta_1, \dots, \eta_n$  是标准正交的。在已知数据向量不正交的一般情况下，应该利用预白化，将原来非正交的数据向量先白化成具有零均值和单位方差的标准白噪声（它们是标准正交的）。然后，对白化之后的数据向量使用投影定理求均方估计。

在某些情况下，可以很容易求得向量  $x$  到子空间  $M$  的投影。

**定理 9.1.3<sup>[255]</sup>** 令  $H$  是一内积空间， $x$  是  $H$  中的一个向量。若  $M$  是  $H$  中的  $n$  维子空间，并且  $\{u_1, \dots, u_n\}$  是子空间  $M$  的一组正交基向量，则

$$\|x - \hat{x}\| \leq \|x - y\|$$

当且仅当

$$\hat{x} = \frac{\langle x, u_1 \rangle}{\langle u_1, u_1 \rangle} u_1 + \frac{\langle x, u_2 \rangle}{\langle u_2, u_2 \rangle} u_2 + \cdots + \frac{\langle x, u_n \rangle}{\langle u_n, u_n \rangle} u_n \quad (9.1.10)$$

这一定理的意义在于：当  $M$  是内积空间  $H$  的有限维子空间时，可以先求出子空间  $M$  的一组正交基向量  $\{u_1, \dots, u_n\}$ （例如使用 Gram-Schmidt 正交化方法）；然后，再根据式 (9.1.10) 计算向量  $x$  在子空间  $M$  上的投影  $\hat{x}$ 。

## 9.2 投影矩阵与正交投影矩阵

在 9.1 节的讨论中，只是简单地提及了投影算子这一术语。本节对投影算子展开专门分析。由于投影算子与幂等矩阵密切相关，先讨论幂等矩阵。

### 9.2.1 幂等矩阵

任何一个满足幂等关系  $A^2 = A$  的矩阵  $A$  称为幂等矩阵。容易验证，单位矩阵也是幂等矩阵，但在以后的讨论中，假定幂等矩阵不取单位矩阵的形式，除非另有申明。

幂等矩阵具有以下有用性质<sup>[444]</sup>:

- (1) 幂等矩阵的特征值只取 1 和 0 两个数值。
- (2) 所有的幂等矩阵(单位矩阵除外)  $A$  都是奇异矩阵。
- (3) 所有幂等矩阵的秩与迹相等, 即  $\text{rank}(A) = \text{tr}(A)$ 。
- (4) 若  $A$  为幂等矩阵, 则  $A^H$  也为幂等矩阵, 即有  $A^H A^H = A^H$ 。
- (5) 若  $A$  为幂等矩阵, 则  $I_n - A$  也是幂等矩阵, 且  $\text{rank}(I_n - A) = n - \text{rank}(A)$ 。
- (6) 所有对称的幂等矩阵(单位矩阵除外)都是半正定的。
- (7) 令  $n \times n$  幂等矩阵  $A$  的秩为  $r_A$ , 则  $A$  有  $r_A$  个特征值 1 和  $n - r_A$  个特征值 0。
- (8) 一个对称的幂等矩阵  $A$  可以表示为  $A = LL^T$ , 其中,  $L$  满足  $L^T L = I_{r_A}$ 。
- (9) 所有的幂等矩阵  $A$  都是可对角化的

$$U^{-1}AU = \Sigma = \begin{bmatrix} I_{r_A} & O \\ O & O \end{bmatrix} \quad (9.2.1)$$

式中,  $r_A = \text{rank}(A)$ 。

虽然幂等矩阵的特征值只取 0 和 1, 但是特征值只取 0 和 1 的矩阵却不一定都是幂等矩阵。例如

$$B = \frac{1}{8} \begin{bmatrix} 11 & 3 & 3 \\ 1 & 1 & 1 \\ -12 & -4 & -4 \end{bmatrix}$$

有三个特征值 1, 0 和 0, 但它不是幂等矩阵, 因为

$$B^2 = \frac{1}{8} \begin{bmatrix} 11 & 3 & 3 \\ 0 & 0 & 0 \\ -11 & -3 & -3 \end{bmatrix} \neq B$$

与幂等矩阵的定义相类似, 满足  $A^2 = O$  (零矩阵) 的矩阵  $A$  称为幂零矩阵(nilpotent matrix), 而满足  $A^2 = I$  (单位矩阵) 的矩阵  $A$  则称为幂 1 矩阵(unipotent matrix)<sup>[444]</sup>。

矩阵  $A_{n \times n}$  称为三幂矩阵(tripotent matrix), 若  $A^3 = A$ 。

容易看出, 若  $A$  为三幂矩阵, 则  $-A$  也是三幂矩阵。

需要注意的是, 一个三幂矩阵不一定是幂等矩阵, 虽然一个幂等矩阵肯定是三幂矩阵(因为若  $A^2 = A$ , 则  $A^3 = A^2 A = AA = A$ )。为了证明这一点, 我们来考察三幂矩阵的特征值。令  $\lambda$  是三幂矩阵  $A$  的特征值, 并且  $u$  是与之对应的特征向量, 即有  $Au = \lambda u$ 。两边左乘矩阵  $A$ , 则有  $A^2 u = \lambda Au = \lambda^2 u$ 。等式两边左乘矩阵  $A$ , 立即得

$$A^3 u = \lambda^2 Au = \lambda^3 u$$

由于  $A^3 = A$  为三幂矩阵, 上式又可写作  $Au = \lambda^3 u$ , 故三幂矩阵的特征值满足关系式  $\lambda = \lambda^3$ , 即三幂矩阵的特征值有  $-1, 0, +1$  三种取值的可能, 这与幂等矩阵的特征值只取 0 和  $+1$  两种值不同。从这个意义上讲, 幂等矩阵是没有特征值为  $-1$  的特殊三幂矩阵。

### 9.2.2 投影算子与正交投影算子

**定义 9.2.1** [424] 考虑向量空间的直和分解  $\mathbb{C}^n = S \oplus H$  内的任意向量  $\mathbf{x} \in \mathbb{C}^n$ 。若  $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$  满足  $\mathbf{x}_1 \in S$  和  $\mathbf{x}_2 \in H$ , 并且  $\mathbf{x}_1$  和  $\mathbf{x}_2$  是唯一确定的, 则称映射  $\mathbf{P}\mathbf{x} = \mathbf{x}_1$  是向量  $\mathbf{x}$  沿着子空间  $H$  的方向, 到子空间  $S$  的投影, 并称  $\mathbf{P}$  是沿着  $H$  的方向, 到  $S$  的投影算子 (projector onto  $S$  along  $H$ ), 常简记为  $\mathbf{P}_{S|H}$ 。

令  $\mathbf{y} = \mathbf{P}_{S|H}\mathbf{x}$  表示向量  $\mathbf{x}$  沿着子空间  $H$  的方向到子空间  $S$  的投影, 则  $\mathbf{y} \in S$ 。如果将  $\mathbf{y}$  沿着  $H$  的方向, 再向  $S$  子空间投影, 则显然有  $\mathbf{P}_{S|H}\mathbf{y} = \mathbf{y}$ 。于是, 有

$$\mathbf{P}_{S|H}\mathbf{y} = \mathbf{y} \implies \mathbf{P}_{S|H}(\mathbf{P}_{S|H}\mathbf{x}) = \mathbf{P}_{S|H}\mathbf{P}_{S|H}\mathbf{x} = \mathbf{P}_{S|H}\mathbf{x} \implies \mathbf{P}_{S|H}^2 = \mathbf{P}_{S|H}$$

**定义 9.2.2** 齐次线性算子  $\mathbf{P}$  称为投影算子, 若它具有幂等性, 即  $\mathbf{P}^2 = \mathbf{P}\mathbf{P} = \mathbf{P}$ 。

唯一分解

$$\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2 = \mathbf{P}\mathbf{x} + (\mathbf{I} - \mathbf{P})\mathbf{x} \quad (9.2.2)$$

将  $\mathbb{C}^n$  的任意一个向量  $\mathbf{x}$  映射为  $S$  子空间的分量  $\mathbf{x}_1$  和  $H$  子空间的分量  $\mathbf{x}_2$ 。然而, 式 (9.2.2) 的唯一分解并不能保证  $\mathbf{x}_1$  和  $\mathbf{x}_2$  相互正交。在很多实际应用中, 常要求复向量空间  $\mathbb{C}^n$  的任一向量  $\mathbf{x}$  在两个子空间的投影  $\mathbf{x}_1$  和  $\mathbf{x}_2$  正交。由  $\mathbf{x}_1$  和  $\mathbf{x}_2$  正交的条件

$$\langle \mathbf{x}_1, \mathbf{x}_2 \rangle = (\mathbf{P}\mathbf{x})^\text{H}(\mathbf{I} - \mathbf{P})\mathbf{x} = 0 \implies \mathbf{x}^\text{H}\mathbf{P}^\text{H}(\mathbf{I} - \mathbf{P})\mathbf{x} = 0, \forall \mathbf{x} \neq \mathbf{0}$$

立即有

$$\mathbf{P}^\text{H}(\mathbf{I} - \mathbf{P}) = \mathbf{O} \implies \mathbf{P}^\text{H}\mathbf{P} = \mathbf{P}^\text{H} = \mathbf{P}$$

由此可引出正交投影算子的定义。

**定义 9.2.3** 映射  $\mathbf{P}^\perp = \mathbf{I} - \mathbf{P}$  称为  $\mathbf{P}$  的正交投影算子 (orthogonal projector), 若  $\mathbf{P}$  不仅是幂等矩阵, 而且还是 Hermitian 矩阵。

正交投影算子具有以下性质 [400]:

- (1) 若  $\mathbf{I} - \mathbf{P}$  是  $\mathbf{P}$  的正交投影算子, 则  $\mathbf{P}$  也是  $\mathbf{I} - \mathbf{P}$  的正交投影算子。
- (2) 若  $\mathbf{E}_1$  和  $\mathbf{E}_2$  均为正交投影算子, 且  $\mathbf{E}_1$  和  $\mathbf{E}_2$  无交叉项:  $\mathbf{E}_1\mathbf{E}_2 = \mathbf{E}_2\mathbf{E}_1 = \mathbf{O}$ , 则  $\mathbf{E}_1 + \mathbf{E}_2$  为正交投影算子。
- (3) 若  $\mathbf{E}_1$  和  $\mathbf{E}_2$  均为正交投影算子, 且  $\mathbf{E}_1\mathbf{E}_2 = \mathbf{E}_2\mathbf{E}_1 = \mathbf{E}_2$ , 则  $\mathbf{E}_1 - \mathbf{E}_2$  为正交投影算子。
- (4) 若  $\mathbf{E}_1$  和  $\mathbf{E}_2$  为正交投影算子, 且  $\mathbf{E}_1\mathbf{E}_2 = \mathbf{E}_2\mathbf{E}_1$ , 则  $\mathbf{E}_1\mathbf{E}_2$  是正交投影算子。

投影算子的幂等性和正交投影算子的 Hermitian 性具有明确的物理解释。

如图 9.2.1 所示, 将离散时间的滤波器视作一个投影算符或算子, 不妨令其为  $\mathbf{P}$ 。设滤波器在离散时间  $n$  的输入向量为

$$\mathbf{x}(n) = [x(1), x(2), \dots, x(n)]^\text{T} \quad (9.2.3)$$

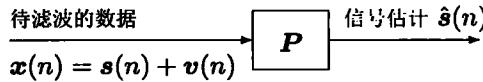


图 9.2.1 滤波器的投影算子表示

它是信号向量  $s(n)$  与加性白噪声  $v(n)$  的混合, 即  $x(n) = s(n) + v(n)$ 。

我们希望含噪声的数据向量  $x(n)$  通过滤波器  $P$  后, 得到滤波后的数据向量即信号向量的估计  $\hat{s}(n) = Px(n)$ 。下面分析对滤波器算子  $P$  应该有哪些基本要求? 为简便计, 省略向量  $s(n)$  和  $x(n)$  等中的时间变量, 将它们分别简记为  $s$  和  $x$ 。

(1) 为了保证信号通过滤波器后不致发生“畸变”, 投影算子  $P$  必须是一线性算子。

(2) 当滤波器输出  $\hat{s}(n)$  再次通过滤波器时, 信号估计  $\hat{s}(n)$  不应发生任何变化。这意味着  $PPx = Px = \hat{s}$  必须得到满足。这一条件等价为  $P^2 \stackrel{\text{def}}{=} PP = P$ , 即投影算子  $P$  必须是一个幂等算子。

(3) 由于信号估计为  $\hat{s} = Px$ , 因此  $x - Px$  代表滤波器的估计误差。根据正交性原理的引理, 当滤波器工作在最优条件时, 估计误差  $x - Px$  应该与期望响应的估计值  $Px$  正交, 即  $[x - Px] \perp Px$ 。这恰好就是正交分解的条件, 意味着正交投影算子必须具有 Hermitian 性。

下面讨论投影矩阵的构造方法。

为方便计, 令  $m \times m$  投影矩阵  $P$  有  $r$  个特征值为 1, 另外  $m - r$  个特征值为 0。于是, 投影矩阵可以写作

$$P = \sum_{i=1}^m \lambda_i u_i u_i^H = \sum_{i=1}^r u_i u_i^H \quad (9.2.4)$$

考查任意一个  $m \times 1$  向量  $x$  的投影  $y = Px$ , 则

$$y = Px = \sum_{i=1}^r u_i u_i^H x = \sum_{i=1}^r (x^H u_i)^H u_i \quad (9.2.5)$$

式 (9.2.5) 揭示了投影矩阵的本质作用:

- (1) 向量  $x$  经过投影矩阵  $P$  投影后, 向量  $x$  与投影矩阵中具有特征值 1 的特征向量相关的部分  $x^H u_i (i = 1, 2, \dots, r)$  在投影结果  $Px$  中被完整保留。
- (2) 向量  $x$  与投影矩阵中具有特征值 0 的特征向量相关的部分  $x^H u_i (i = r+1, r+2, \dots, m)$  被投影矩阵全部对消, 不出现在投影结果  $Px$  中。

因此, 当矩阵  $P$  是只具有特征值 0 和 1 的幂等矩阵时, 变换结果  $Px$  是向量  $x$  在  $P$  那些具有特征值 1 的特征向量上的投影  $(x^H u_i)^H u_i (i = 1, 2, \dots, r)$  之叠加。“投影矩阵”由此而得名。

### 9.2.3 到列空间的投影矩阵与正交投影矩阵

令  $m \times n$  维矩阵  $\mathbf{A}$  是一个满列秩矩阵, 即  $\text{rank}(\mathbf{A}) = n$ 。记矩阵  $\mathbf{A}$  的列空间  $C(\mathbf{A}) = \text{Col}(\mathbf{A}) = \text{Range}(\mathbf{A})$ 。一个自然会问的问题是: 如何构造到列空间  $C(\mathbf{A})$  的投影矩阵  $\mathbf{P}_{C(\mathbf{A})}$ ?

由于矩阵  $\mathbf{A}$  的秩为  $n$ , 只有  $n$  个非零奇异值  $\sigma_1, \dots, \sigma_n$ , 故  $\mathbf{A}$  的奇异值分解

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \Sigma_1 \\ \mathbf{O}_{(m-n) \times n} \end{bmatrix} \mathbf{V}^H = \mathbf{U}_1 \Sigma_1 \mathbf{V}^H \quad (9.2.6)$$

的对角矩阵  $\Sigma_1 = \text{diag}(\sigma_1, \dots, \sigma_n)$ , 并且

$$\mathbf{U}_1 = [\mathbf{u}_1, \dots, \mathbf{u}_n], \quad \mathbf{U}_2 = [\mathbf{u}_{n+1}, \dots, \mathbf{u}_m] \quad (9.2.7)$$

分别是与  $n$  个非零奇异值和  $m - n$  个零奇异值对应的奇异向量矩阵。

在第 8 章中, 我们曾经得到关于列空间的两个重要结果:

(1) 与非零奇异值对应的  $n$  个左奇异向量  $\mathbf{u}_1, \dots, \mathbf{u}_n$  是列空间  $\text{Col}(\mathbf{A})$  的标准正交基, 即有

$$\text{Col}(\mathbf{A}) = \text{Span}\{\mathbf{u}_1, \dots, \mathbf{u}_n\} = \text{Span}(\mathbf{U}_1) \quad (9.2.8)$$

(2) 与零奇异值对应的  $m - n$  个左奇异向量  $\mathbf{u}_{n+1}, \dots, \mathbf{u}_m$  是零空间  $\text{Null}(\mathbf{A}^H)$  的标准正交基, 即

$$\text{Null}(\mathbf{A}^H) = (\text{Col} \mathbf{A})^\perp = \text{Span}\{\mathbf{u}_{n+1}, \dots, \mathbf{u}_m\} = \text{Span}(\mathbf{U}_2) \quad (9.2.9)$$

用矩阵  $\mathbf{A}$  对  $n \times 1$  向量  $\mathbf{x}$  作线性变换, 得到  $m \times 1$  向量  $\mathbf{y} = \mathbf{Ax}$ , 则向量  $\mathbf{y}$  可以表示为

$$\mathbf{y} = \mathbf{Ax} = \sum_{i=1}^n \sigma_i \mathbf{u}_i (\mathbf{v}_i^H \mathbf{x}) = \sum_{i=1}^n (\sigma_i \alpha_i) \mathbf{u}_i \quad (9.2.10)$$

式中,  $\alpha_i = \mathbf{v}_i^H \mathbf{x}$  是特征向量  $\mathbf{v}_i$  与向量  $\mathbf{x}$  的内积。式 (9.2.10) 表明, 线性变换结果  $\mathbf{y} = \mathbf{Ax}$  是矩阵  $\mathbf{A}$  的左奇异向量的线性组合。

$m \times n$  线性变换  $\mathbf{A}$  的投影矩阵  $\mathbf{P}_{\mathbf{A}}$  将所有  $m \times 1$  向量  $\mathbf{x}$  投影到由线性变换  $\mathbf{A}$  定义的子空间。投影矩阵  $\mathbf{P}_{\mathbf{A}}$  与线性变换  $\mathbf{A}$  具有相同的特征向量:  $n$  个特征向量与非零特征值对应, 其他  $m - n$  个特征向量与零特征值对应。由于投影矩阵只有特征值 1 和 0, 因此投影矩阵的  $n$  个特征向量与特征值 1 对应, 其他  $m - n$  个特征向量与特征值 0 对应。换言之, 投影矩阵的特征值分解具有以下形式

$$\mathbf{P}_{\mathbf{A}} = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \mathbf{I}_n & \mathbf{O}_{n \times (m-n)} \\ \mathbf{O}_{(m-n) \times n} & \mathbf{O}_{(m-n) \times (m-n)} \end{bmatrix} \begin{bmatrix} \mathbf{U}_1^H \\ \mathbf{U}_2^H \end{bmatrix} = \mathbf{U}_1 \mathbf{U}_1^H \quad (9.2.11)$$

另一方面, 由式 (9.2.6) 可求得

$$\begin{aligned}
 \mathbf{A} \langle \mathbf{A}, \mathbf{A} \rangle^{-1} \mathbf{A}^H &= \mathbf{A} (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \\
 &= \mathbf{U}_1 \Sigma_1 V^H (V \Sigma_1 U_1^H U_1 \Sigma_1 V^H)^{-1} V \Sigma_1 U_1^H \\
 &= \mathbf{U}_1 \Sigma_1 V^H (V \Sigma_1^2 V^H)^{-1} V \Sigma_1 U_1^H \\
 &= \mathbf{U}_1 \Sigma_1 V^H V \Sigma_1^{-2} V^H V \Sigma_1 U_1^H \\
 &= \mathbf{U}_1 U_1^H
 \end{aligned} \tag{9.2.12}$$

比较式 (9.2.11) 和式 (9.2.12), 立即得到矩阵  $\mathbf{A}$  的投影矩阵  $\mathbf{P}_A$  的定义式

$$\mathbf{P}_A = \mathbf{A} \langle \mathbf{A}, \mathbf{A} \rangle^{-1} \mathbf{A}^H \tag{9.2.13}$$

以上介绍的投影矩阵  $\mathbf{P}_A$  的定义式 (9.2.13) 的推导是数学文献中通常采用的方法, 其关键是先猜测到投影公式  $\mathbf{A} \langle \mathbf{A}, \mathbf{A} \rangle^{-1} \mathbf{A}^H$  的形式, 再进行验证。

下面介绍另外一种推导方法, 投影公式  $\mathbf{A} \langle \mathbf{A}, \mathbf{A} \rangle^{-1} \mathbf{A}^H$  的形式可以自然地得到, 其基础是 Moore-Penrose 逆矩阵和投影的基本事实。

用  $U^\dagger U$  右乘影射函数  $\mathbf{P}U = \mathbf{V}$  两边, 由于

$$\mathbf{P}UU^\dagger U = \mathbf{P}U = \mathbf{V}U^\dagger U$$

对任意矩阵  $U$  恒成立, 故有

$$\mathbf{P} = \mathbf{V}U^\dagger = \mathbf{V}(U^H U)^\dagger U^H \tag{9.2.14}$$

令  $A = \text{Col}(\mathbf{A})$  是矩阵  $\mathbf{A} \in \mathbb{C}^{m \times n}$  的列空间,  $A^\perp = (\text{Col}(\mathbf{A}))^\perp = \text{Null}(\mathbf{A}^H)$  是列空间  $A$  的正交补。因此, 若  $S \in A^\perp$ , 则  $S$  和  $A$  相互正交, 即  $S \perp A$ 。

于是, 我们有投影的下列两个基本事实

$$\mathbf{P}_A \mathbf{A} = \mathbf{A}, \quad \mathbf{P}_A S = \mathbf{O}, \quad \forall S \in \text{Range}(\mathbf{A})^\perp \tag{9.2.15}$$

或合写为

$$\mathbf{P}_A [\mathbf{A}, \mathbf{S}] = [\mathbf{A}, \mathbf{O}] \tag{9.2.16}$$

其中  $\mathbf{O}$  为零矩阵。由式 (9.2.14) 和式 (9.2.16) 立即有

$$\begin{aligned}
 \mathbf{P}_A &= [\mathbf{A}, \mathbf{O}] ([\mathbf{A}, \mathbf{S}]^H [\mathbf{A}, \mathbf{S}])^\dagger [\mathbf{A}, \mathbf{S}]^H \\
 &= [\mathbf{A}, \mathbf{O}] \begin{bmatrix} \mathbf{A}^H \mathbf{A} & \mathbf{A}^H \mathbf{S} \\ \mathbf{S}^H \mathbf{A} & \mathbf{S}^H \mathbf{S} \end{bmatrix}^\dagger \begin{bmatrix} \mathbf{A}^H \\ \mathbf{S}^H \end{bmatrix} \\
 &= [\mathbf{A}, \mathbf{O}] \begin{bmatrix} \mathbf{A}^H \mathbf{A} & \mathbf{O} \\ \mathbf{O} & \mathbf{S}^H \mathbf{S} \end{bmatrix}^\dagger \begin{bmatrix} \mathbf{A}^H \\ \mathbf{S}^H \end{bmatrix} \\
 &= \mathbf{A} (\mathbf{A}^H \mathbf{A})^\dagger \mathbf{A}^H
 \end{aligned} \tag{9.2.17}$$

这与前面推导的结果完全相同。

容易验证, 由式 (9.2.13) 定义的投影矩阵  $\mathbf{P}_A$  具有以下性质:

(1) 幂等性

$$\mathbf{P}_A \mathbf{P}_A = \mathbf{P}_A \quad (9.2.18)$$

(2) 复共轭对称性或 Hermitian 性

$$\mathbf{P}_A^H = \mathbf{P}_A \quad (9.2.19)$$

有了投影矩阵后, 又可定义新的矩阵

$$\mathbf{P}_A^\perp = \mathbf{I} - \mathbf{P}_A = \mathbf{I} - \mathbf{A}(\mathbf{A}, \mathbf{A})^{-1}\mathbf{A}^H \quad (9.2.20)$$

由此定义式易知  $\mathbf{P}_A^\perp$  具有以下性质:

(1) 对称性

$$[\mathbf{P}_A^\perp]^H = \mathbf{P}_A^\perp \quad (9.2.21)$$

(2) 幂等性

$$\mathbf{P}_A^\perp \mathbf{P}_A^\perp = \mathbf{P}_A^\perp \quad (9.2.22)$$

(3) 与投影矩阵的正交性

$$\mathbf{P}_A^\perp \mathbf{P}_A = \mathbf{O} \quad \text{或} \quad \mathbf{P}_A \mathbf{P}_A^\perp = \mathbf{O} \quad (\text{零矩阵}) \quad (9.2.23)$$

由于  $\mathbf{P}_A^\perp$  与投影矩阵  $\mathbf{P}_A$  正交, 故  $\mathbf{P}_A^\perp$  称作正交投影矩阵。

#### 9.2.4 投影矩阵的导数

令投影矩阵

$$\mathbf{P}_A(\boldsymbol{\theta}) = \mathbf{A}(\boldsymbol{\theta})[\mathbf{A}^H(\boldsymbol{\theta})\mathbf{A}(\boldsymbol{\theta})]^{-1}\mathbf{A}^H(\boldsymbol{\theta}) = \mathbf{A}(\boldsymbol{\theta})\mathbf{A}^\dagger(\boldsymbol{\theta})$$

是某个向量  $\boldsymbol{\theta}$  的函数, 式中,  $\mathbf{A}^\dagger(\boldsymbol{\theta}) = [\mathbf{A}^H(\boldsymbol{\theta})\mathbf{A}(\boldsymbol{\theta})]^{-1}\mathbf{A}^H(\boldsymbol{\theta})$  是矩阵  $\mathbf{A}(\boldsymbol{\theta})$  的伪逆矩阵。为了书写的简洁, 将  $\mathbf{P}_A(\boldsymbol{\theta})$  简记为  $\mathbf{P}$ 。

下面介绍投影矩阵  $\mathbf{P}$  关于向量  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_n]^T$  的各个元素  $\theta_i$  的一阶与二阶导数。这些结果是由 Golub 与 Pereyra 最早给出的<sup>[195]</sup>。

定义投影矩阵关于  $\theta_i$  的一阶偏导数为

$$\mathbf{P}_i \stackrel{\text{def}}{=} \frac{\partial \mathbf{P}}{\partial \theta_i} \quad (9.2.24)$$

利用求导数的链式法则, 得

$$\mathbf{P}_i = \mathbf{A}_i \mathbf{A}^\dagger + \mathbf{A} \mathbf{A}_i^\dagger \quad (9.2.25)$$

式中

$$\mathbf{A}_i \stackrel{\text{def}}{=} \frac{\partial \mathbf{A}}{\partial \theta_i}, \quad \mathbf{A}_i^\dagger \stackrel{\text{def}}{=} \frac{\partial \mathbf{A}^\dagger}{\partial \theta_i} \quad (9.2.26)$$

分别是矩阵  $\mathbf{A}(\boldsymbol{\theta})$  及其 Moore-Penrose 逆矩阵  $\mathbf{A}^\dagger(\boldsymbol{\theta})$  关于  $\theta_i$  的偏导数。

在经过某些代数运算后, 可得伪逆矩阵  $\mathbf{A}^\dagger$  的一阶偏导数为

$$\mathbf{A}_i^\dagger = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}_i^H \mathbf{P}^\perp - \mathbf{A}^\dagger \mathbf{A}_i \mathbf{A}^\dagger \quad (9.2.27)$$

综合式 (9.2.25) 和式 (9.2.27) 得到

$$\mathbf{P}_i = \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^\dagger + (\mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^\dagger)^H \quad (9.2.28)$$

由此式容易验证, 正如所希望的那样, 有  $\text{tr}(\mathbf{P}_i) = 0$ , 因为一个投影矩阵的迹只与投影矩阵投影到的子空间的维数有关。

投影矩阵的二阶偏导数为

$$\begin{aligned} \mathbf{P}_{i,j} &= \mathbf{P}_j^\perp \mathbf{A}_i \mathbf{A}^\dagger + \mathbf{P}^\perp \mathbf{A}_{i,j} \mathbf{A}^\dagger + \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}_j^H + \\ &\quad (\mathbf{P}_j^\perp \mathbf{A}_i \mathbf{A}^\dagger + \mathbf{P}^\perp \mathbf{A}_{i,j} \mathbf{A}^\dagger + \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}_j^H)^H \end{aligned} \quad (9.2.29)$$

注意到  $\mathbf{P}_j^\perp = -\mathbf{P}_j$ , 并利用式 (9.2.27), 可以将式 (9.2.29) 表述为<sup>[504]</sup>

$$\begin{aligned} \mathbf{P}_{i,j} &= -\mathbf{P}^\perp \mathbf{A}_j \mathbf{A}^\dagger \mathbf{A}_i \mathbf{A}^\dagger - (\mathbf{A}^\dagger)^H \mathbf{A}_j^H \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^H + \mathbf{P}^\perp \mathbf{A}_{i,j} \mathbf{A}^\dagger + \\ &\quad \mathbf{P}^\perp \mathbf{A}_i (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}_j^H \mathbf{P}^\perp - \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^\dagger \mathbf{A}_j \mathbf{A}^\dagger + \\ &\quad [-\mathbf{P}^\perp \mathbf{A}_j \mathbf{A}^\dagger \mathbf{A}_i \mathbf{A}^\dagger - (\mathbf{A}^\dagger)^H \mathbf{A}_j^H \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^H + \mathbf{P}^\perp \mathbf{A}_{i,j} \mathbf{A}^\dagger + \\ &\quad \mathbf{P}^\perp \mathbf{A}_i (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}_j^H \mathbf{P}^\perp - \mathbf{P}^\perp \mathbf{A}_i \mathbf{A}^\dagger \mathbf{A}_j \mathbf{A}^\dagger]^H \end{aligned} \quad (9.2.30)$$

投影矩阵的导数公式在涉及投影矩阵的某些估计器的统计性能分析时非常有用。对此应用感兴趣的读者可参考文献 [504]。

### 9.3 投影矩阵与正交投影矩阵的应用举例

9.2 节分别从数学和信号处理角度出发, 引出了投影矩阵的概念。本节将通过举例, 介绍投影矩阵和正交投影矩阵的几个典型应用。

#### 9.3.1 投影梯度

考查一直接序列码分多址 (CDMA) 系统, 它有  $K$  个用户。在经过一系列预处理后, 接收机在第  $n$  个码元间隔的离散时间输出可用信号模型

$$y(n) = \sum_{k=1}^K A_k b_k s_k(n) + \sigma v(n), \quad n = 0, 1, \dots, N-1 \quad (9.3.1)$$

表示。式中,  $v(n)$  为信道高斯白噪声;  $A_k$ ,  $b_k$  和  $s_k(n)$  分别是第  $k$  个用户的接收幅值、信息字符序列和特征波形;  $\sigma^2$  为一常数, 表示高斯白噪声的方差。现在假定各个用户的信息字符从  $\{-1, +1\}$  中独立地、等概率地选取, 还假定特征波形的长度为  $N$ , 具有单位能量, 即

$$\sum_{n=0}^{N-1} |s_k(n)|^2 = 1 \quad \text{或} \quad \langle s_k, s_k \rangle = 1 \quad (9.3.2)$$

式中,  $s_k = [s_k(0), s_k(1), \dots, s_k(N-1)]^T$  表示用户  $k$  的特征波形向量。

盲多用户检测问题的提法是: 只已知一个码元间隔内的接收信号  $y(0), \dots, y(N-1)$  和期望用户的特征波形  $s_d(0), s_d(1), \dots, s_d(N-1)$ , 估计期望用户发射的信息字符  $b_d$ 。这里,“盲”是指我们不知道其他用户的任何信息。不失一般性, 假定用户 1 为期望用户。

定义

$$\begin{aligned} \mathbf{y}(n) &= [y(0), y(1), \dots, y(N-1)]^T \\ \mathbf{v}(n) &= [v(0), v(1), \dots, v(N-1)]^T \end{aligned}$$

分别为接收信号向量和噪声向量, 则式 (9.3.1) 可以用向量形式写作

$$\mathbf{y}(n) = A_1 b_1(n) s_1 + \sum_{k=2}^K A_k b_k(n) s_k + \sigma \mathbf{v}(n) \quad (9.3.3)$$

式中, 第一项为期望用户的信号, 第二项为所有其他用户 (统称干扰用户) 的干扰信号之和, 第三项代表信道噪声。

现在针对期望用户 1, 设计其在码元间隔  $n$  内的多用户检测器  $c_1(n)$ , 则检测器输出为  $c_1^T(n)\mathbf{y}(n) = \langle c_1, \mathbf{y} \rangle$ 。因此, 在第  $n$  个码元间隔内的期望用户的二进制信息字符 +1 或 -1 可以使用

$$\hat{b}_1(n) = \text{sgn}(\langle c_1, \mathbf{y} \rangle) = \text{sgn}(c_1^T(n)\mathbf{y}(n)) \quad (9.3.4)$$

检测。

将盲多用户检测器  $c_1$  分解为固定部分  $s_1$  与自适应调整部分  $x_1$  之和<sup>[237]</sup>

$$c_1(n) = s_1 + x_1(n) \quad (9.3.5)$$

并且这两部分正交, 即

$$\langle s_1, x_1(n) \rangle = 0 \quad (9.3.6)$$

因此, 式 (9.3.5) 是一种典型的正交分解。

现在, 在盲多用户检测器  $c_1(n)$  的设计中, 采用一种最小输出能量准则, 即使得多用户检测器的平均输出能量 (MOE)

$$\text{MOE}(c_1) = E\{\langle c_1, \mathbf{y} \rangle^2\} = E\{(c_1^T(n)\mathbf{y}(n))^2\} \quad (9.3.7)$$

最小化。求平均输出能量关于  $c_1(n)$  的无约束梯度, 得

$$\nabla \text{MOE} = 2E\{\langle \mathbf{y}, s_1 + x_1 \rangle\} \mathbf{y} \quad (9.3.8)$$

于是, 盲多用户检测器  $c_1(n)$  的自适应部分  $x_1(i)$  的随机梯度自适应算法为

$$x_1(i) = x_1(i-1) - \mu \hat{\nabla} \text{MOE} \quad (9.3.9)$$

式中,  $\hat{\nabla} \text{MOE}$  是  $\nabla \text{MOE}$  的估计, 这里采用数学期望直接用其瞬时值代替的梯度

$$\hat{\nabla} \text{MOE} = 2\langle \mathbf{y}, s_1 + x_1 \rangle \mathbf{y} \quad (9.3.10)$$

称为瞬时梯度。此时，盲多用户检测器的随机梯度算法为

$$\mathbf{x}_1(i) = \mathbf{x}_1(i-1) - \mu \langle \mathbf{y}, \mathbf{s}_1 + \mathbf{x}_1 \rangle \mathbf{y} \quad (9.3.11)$$

由正交约束式 (9.3.6) 知，在任何时刻  $i$ ，向量  $\mathbf{x}_1(i)$  都应该与特征波形向量  $\mathbf{s}_1$  正交。因此，在随机梯度算法式 (9.3.11) 中的瞬时梯度  $\langle \mathbf{y}, \mathbf{s}_1 + \mathbf{x}_1 \rangle \mathbf{y}$  应该与  $\mathbf{s}_1$  正交。这只要将式 (9.3.11) 改为

$$\mathbf{x}_1(i) = \mathbf{x}_1(i-1) - \mu \langle \mathbf{y}, \mathbf{s}_1 + \mathbf{x}_1 \rangle \mathbf{y}_1 \quad (9.3.12)$$

即可，其中， $\mathbf{y}_1$  是  $\mathbf{y}$  中与  $\mathbf{s}_1$  正交的分量，可用正交投影矩阵表示为

$$\mathbf{y}_1 = \mathbf{P}_{\mathbf{s}_1}^\perp \mathbf{y} = (\mathbf{I} - \mathbf{P}_{\mathbf{s}_1}) \mathbf{y} \quad (9.3.13)$$

梯度  $2\langle \mathbf{y}, \mathbf{s}_1 + \mathbf{x}_1 \rangle \mathbf{y}_1$  称为投影梯度，因为  $\mathbf{y}_1$  与  $\mathbf{s}_1$  正交，是原观测数据向量  $\mathbf{y}$  在  $\mathbf{s}_1$  张成的子空间上的正交投影。

注意到

$$\mathbf{P}_{\mathbf{s}_1} = \mathbf{s}_1 \langle \mathbf{s}_1, \mathbf{s}_1 \rangle^{-1} \mathbf{s}_1^T = \mathbf{s}_1 \mathbf{s}_1^T$$

式中，使用了式 (9.3.2) 即  $\langle \mathbf{s}_1, \mathbf{s}_1 \rangle = 1$ 。于是，式 (9.3.13) 为

$$\mathbf{y}_1 = (\mathbf{I} - \mathbf{s}_1 \mathbf{s}_1^T) \mathbf{y} = \mathbf{y} - \langle \mathbf{y}, \mathbf{s}_1 \rangle \mathbf{s}_1 \quad (9.3.14)$$

将式 (9.3.14) 代入式 (9.3.12)，即得盲多用户检测器的最小均方 (LMS) 型自适应算法如下<sup>[237]</sup>

$$\mathbf{x}_1(i) = \mathbf{x}_1(i-1) - \mu \langle \mathbf{y}, \mathbf{s}_1 + \mathbf{x}_1 \rangle (\mathbf{y} - \langle \mathbf{y}, \mathbf{s}_1 \rangle \mathbf{s}_1) \quad (9.3.15)$$

或写作

$$\mathbf{x}_1(i) = \mathbf{x}_1(i-1) - \mu Z(i) [\mathbf{y}(i) - Z_{\text{MF}}(i) \mathbf{s}_1] \quad (9.3.16)$$

式中

$$Z_{\text{MF}}(i) = \langle \mathbf{y}(i), \mathbf{s}_1 \rangle$$

$$Z(i) = \langle \mathbf{y}(i), \mathbf{s}_1 + \mathbf{x}_1(i-1) \rangle$$

### 9.3.2 预测滤波器的表示

假定滤波器的输入和抽头权系数均为实数。为方便叙述，先引入时移向量

$$z^{-j} \mathbf{x}(n) = [0, \dots, 0, x(1), \dots, x(n-j)]^T \quad (9.3.17)$$

注意，这里  $z^{-j}$  只是代表一个时间上移位的算子，而不要把它当成一种乘法。此外，约定离散时间变量的起点为 1，即  $x(n) = 0$  对所有  $n \leq 0$ 。

先考虑  $m$  阶前向预测滤波器

$$\hat{x}(k) = \sum_{i=1}^m w_i^f(n) x(k-i), \quad k = 1, 2, \dots, n \quad (9.3.18)$$

式中,  $w_i^f(n), i = 1, 2, \dots, m$  表示  $n$  时刻的滤波器权系数向量。将上式写成矩阵方程

$$\begin{bmatrix} 0 & 0 & \cdots & 0 \\ x(1) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x(n-1) & x(n-2) & \cdots & x(n-m) \end{bmatrix} \begin{bmatrix} w_1^f(n) \\ w_2^f(n) \\ \vdots \\ w_m^f(n) \end{bmatrix} = \begin{bmatrix} \hat{x}(1) \\ \hat{x}(2) \\ \vdots \\ \hat{x}(n) \end{bmatrix} \quad (9.3.19)$$

定义数据矩阵

$$\begin{aligned} \mathbf{X}_{1,m}(n) &\stackrel{\text{def}}{=} [z^{-1}\mathbf{x}(n), z^{-2}\mathbf{x}(n), \dots, z^{-m}\mathbf{x}(n)] \\ &= \begin{bmatrix} 0 & 0 & \cdots & 0 \\ x(1) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x(n-1) & x(n-2) & \cdots & x(n-m) \end{bmatrix} \end{aligned}$$

并分别定义  $m$  级前向预测系数向量  $\mathbf{w}_m^f(n)$  和前向预测值向量  $\hat{\mathbf{x}}(n)$  为

$$\mathbf{w}_m^f(n) \stackrel{\text{def}}{=} [w_1^f(n), w_2^f(n), \dots, w_m^f(n)]^T \quad (9.3.20)$$

$$\hat{\mathbf{x}}(n) \stackrel{\text{def}}{=} [\hat{x}(1), \hat{x}(2), \dots, \hat{x}(n)]^T \quad (9.3.21)$$

则式 (9.3.19) 可以用简洁的形式写作

$$\mathbf{X}_{1,m}(n)\mathbf{w}_m^f(n) = \hat{\mathbf{x}}(n) \quad (9.3.22)$$

为了求出前向预测形式向量的最小二乘估计, 用  $\mathbf{x}(n)$  代替上式中的  $\hat{\mathbf{x}}(n)$ , 便得到

$$\mathbf{w}_m^f(n) = \langle \mathbf{X}_{1,m}^T(n), \mathbf{X}_{1,m}(n) \rangle^{-1} \mathbf{X}_{1,m}^T(n) \mathbf{x}(n) \quad (9.3.23)$$

将式 (9.3.23) 代入式 (9.3.22), 使用投影矩阵符号可将前向预测值向量表示为

$$\hat{\mathbf{x}}(n) = \mathbf{P}_{1,m}(n)\mathbf{x}(n) \quad (9.3.24)$$

式中,  $\mathbf{P}_{1,m}(n) = \mathbf{X}_{1,m}(n)\langle \mathbf{X}_{1,m}^T(n), \mathbf{X}_{1,m}(n) \rangle^{-1} \mathbf{X}_{1,m}^T(n)$  表示数据矩阵  $\mathbf{X}_{1,m}(n)$  的投影矩阵。

若定义前向预测误差向量

$$\mathbf{e}_m^f(n) = [e_m^f(1), \dots, e_m^f(n)]^T = \mathbf{x}(n) - \hat{\mathbf{x}}(n) \quad (9.3.25)$$

式中,  $e_m^f(k), k = 1, \dots, n$  是滤波器在  $k$  时刻的前向预测误差, 则由式 (9.3.24) 及正交投影矩阵的定义立即知

$$\mathbf{e}_m^f(n) = \mathbf{P}_{1,m}^\perp(n)\mathbf{x}(n) \quad (9.3.26)$$

式 (9.3.24) 和式 (9.3.26) 的物理解释是: 前向预测值向量  $\hat{\mathbf{x}}(n)$  和前向预测误差向量  $\mathbf{e}_m^f(n)$  分别是数据向量  $\mathbf{x}(n)$  在数据矩阵  $\mathbf{X}_{1,m}(n)$  所张成的子空间上的投影和正交投影。

现在考虑后向预测滤波器

$$\hat{x}(k-m) = \sum_{i=1}^m w_i^b(n)x(k-m+i), \quad k = 1, 2, \dots, n \quad (9.3.27)$$

式中,  $w_i^b(n), i = 1, 2, \dots, m$  为  $m$  阶后向预测滤波器在  $n$  时刻的权系数。使用矩阵和向量书写上式, 得

$$\begin{bmatrix} x(1) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x(m) & x(m-1) & \cdots & 0 \\ x(m+1) & x(m) & \cdots & x(1) \\ \vdots & \vdots & \vdots & \vdots \\ x(n) & x(n-1) & \cdots & x(n-m+1) \end{bmatrix} \begin{bmatrix} w_m^b(n) \\ w_{m-1}^b(n) \\ \vdots \\ w_1^b(n) \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \hat{x}(1) \\ \vdots \\ \hat{x}(n-m) \end{bmatrix} \quad (9.3.28)$$

或

$$\mathbf{X}_{0,m-1}(n)w_m^b(n) = z^{-m}\hat{\mathbf{x}}(n) \quad (9.3.29)$$

式中

$$\begin{aligned} \mathbf{X}_{0,m-1}(n) &= [z^0\mathbf{x}(n), z^{-1}\mathbf{x}(n), \dots, z^{-m+1}\mathbf{x}(n)] \\ &= \begin{bmatrix} x(1) & 0 & \cdots & 0 \\ x(2) & x(1) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x(n) & x(n-1) & \cdots & x(n-m+1) \end{bmatrix} \end{aligned} \quad (9.3.30)$$

$$\mathbf{w}_m^b(n) = [w_m^b(n), w_{m-1}^b(n), \dots, w_1^b(n)]^T \quad (9.3.31)$$

$$\hat{\mathbf{x}}(n-m) = z^{-m}\hat{\mathbf{x}}(n) = [0, \dots, 0, \hat{x}(1), \dots, \hat{x}(n-m)]^T \quad (9.3.32)$$

在式 (9.3.29) 中用已知的数据向量  $\mathbf{x}(n)$  代替未知的预测值向量  $\hat{\mathbf{x}}(n)$ , 即可得到后向预测滤波器权向量的最小二乘解为

$$\mathbf{w}_m^b(n) = (\mathbf{X}_{0,m-1}(n), \mathbf{X}_{0,m-1}(n))^{-1} \mathbf{X}_{0,m-1}^T(n) \mathbf{x}(n-m) \quad (9.3.33)$$

将式 (9.3.33) 代入式 (9.3.29), 后向预测 (值) 向量可用投影矩阵表示为

$$\hat{\mathbf{x}}(n-m) = \mathbf{P}_{0,m-1}(n)\mathbf{x}(n-m) = \mathbf{P}_{0,m-1}(n)z^{-m}\mathbf{x}(n) \quad (9.3.34)$$

式中,  $\mathbf{P}_{0,m-1}(n) = \mathbf{X}_{0,m-1}(n)(\mathbf{X}_{0,m-1}(n), \mathbf{X}_{0,m-1}(n))^{-1} \mathbf{X}_{0,m-1}^T(n)$  是  $\mathbf{X}_{0,m-1}(n)$  的投影矩阵。

定义后向预测误差向量

$$\mathbf{e}_m^b(n) = [e_m^b(1), \dots, e_m^b(n)]^T = \mathbf{x}(n-m) - \hat{\mathbf{x}}(n-m) \quad (9.3.35)$$

式中,  $e_m^b(k) (k = 1, \dots, n)$  是  $k$  时刻的后向预测误差, 则将式 (9.3.34) 代入式 (9.3.35) 后, 又可用正交投影矩阵来表示后向预测误差向量

$$\mathbf{e}_m^b(n) = \mathbf{P}_{0,m-1}^\perp(n)z^{-m}\mathbf{x}(n) \quad (9.3.36)$$

式 (9.3.34) 和式 (9.3.35) 的物理含义如下: 后向预测向量  $\hat{\mathbf{x}}(n-m)$  和后向预测误差向量  $\mathbf{e}_m^b(n)$  分别是移位的数据向量  $z^{-m}\mathbf{x}(n)$  在数据矩阵  $\mathbf{X}_{0,m-1}(n)$  所张成子空间上的投影和正交投影。

**例 9.3.1** 假定观测数据矩阵为  $\mathbf{X}_{N \times M}$ , 现在希望设计一  $M \times 1$  阶滤波器向量  $\mathbf{w}$  拟合向量  $\mathbf{y}$ , 则观测方程可以写作

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \mathbf{e} \quad (9.3.37)$$

式中,  $\mathbf{e}$  为拟合误差向量。设最优滤波器为  $\mathbf{w}_{\text{opt}}$ , 其估计误差向量为  $\mathbf{e}_{\text{opt}}$ , 则

$$\mathbf{y} = \mathbf{X}\mathbf{w}_{\text{opt}} + \mathbf{e}_{\text{opt}} \quad (9.3.38)$$

两边同乘  $(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H$ , 即有

$$\mathbf{w}_{\text{opt}} = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{y} - (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{e}_{\text{opt}} \quad (9.3.39)$$

根据前面的分析, 滤波器的最小二乘估计为

$$\mathbf{w}_{\text{LS}} = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{y} \quad (9.3.40)$$

将式 (9.3.39) 代入上式, 即得

$$\mathbf{w}_{\text{LS}} = \mathbf{w}_{\text{opt}} + (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{e}_{\text{opt}} \quad (9.3.41)$$

于是, 有

$$\mathbf{y} - \mathbf{X}\mathbf{w}_{\text{LS}} = \mathbf{y} - \mathbf{X}\mathbf{w}_{\text{opt}} - \mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \mathbf{e}_{\text{opt}}$$

由上式、式 (9.3.39) 和式 (9.3.40), 立即得

$$(\mathbf{I} - \mathbf{P})\mathbf{y} = (\mathbf{I} - \mathbf{P})\mathbf{e}_{\text{opt}} \quad (9.3.42)$$

最小二乘估计

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}_{\text{LS}} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = \mathbf{P}\mathbf{y} \quad (9.3.43)$$

式中,  $\mathbf{P} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$  表示观测数据矩阵  $\mathbf{X}$  的投影矩阵。于是, 估计误差向量

$$\mathbf{e}_{\text{LS}} = \mathbf{y} - \hat{\mathbf{y}} = (\mathbf{I} - \mathbf{P})\mathbf{y} = (\mathbf{I} - \mathbf{P})\mathbf{e}_{\text{opt}} \quad (9.3.44)$$

式中, 使用了式 (9.3.42)。

估计误差平方和

$$E_{\text{LS}} = \mathbf{e}_{\text{LS}}^H \mathbf{e}_{\text{LS}} = \mathbf{e}_{\text{opt}}^H (\mathbf{I} - \mathbf{P})^H (\mathbf{I} - \mathbf{P}) \mathbf{e}_{\text{opt}} = \mathbf{e}_{\text{opt}}^H (\mathbf{I} - \mathbf{P}) \mathbf{e}_{\text{opt}} \quad (9.3.45)$$

估计误差平方和的数学期望值称为均方误差, 即有

$$\begin{aligned} E\{E_{\text{LS}}\} &= E\{\mathbf{e}_{\text{opt}}^H (\mathbf{I} - \mathbf{P}) \mathbf{e}_{\text{opt}}\} = E\{\text{tr}[(\mathbf{I} - \mathbf{P}) \mathbf{e}_{\text{opt}} \mathbf{e}_{\text{opt}}^H]\} \\ &= \text{tr}[(\mathbf{I} - \mathbf{P}) E\{\mathbf{e}_{\text{opt}} \mathbf{e}_{\text{opt}}^H\}] \end{aligned}$$

式中, 利用了矩阵的迹的性质  $\mathbf{x}^H \mathbf{A} \mathbf{x} = \text{tr}(\mathbf{A} \mathbf{x} \mathbf{x}^H)$ 。令  $\sigma_{\text{opt}}^2 = E\{\mathbf{e}_{\text{opt}}^H \mathbf{e}_{\text{opt}}\}$  表示最优滤波器的均方误差。于是, 上式可以写为

$$E\{E_{\text{LS}}\} = \sigma_{\text{opt}}^2 \text{tr}(\mathbf{I} - \mathbf{P}) \quad (9.3.46)$$

计算矩阵的迹, 得

$$\begin{aligned}\text{tr}(\mathbf{I} - \mathbf{P}) &= \text{tr}[\mathbf{I} - \mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H] \\ &= \text{tr}(\mathbf{I}_N) - \text{tr}[\mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H] \\ &= \text{tr}(\mathbf{I}_N) - \text{tr}[\mathbf{X}^H \mathbf{X}(\mathbf{X}^H \mathbf{X})^{-1}] \\ &= \text{tr}(\mathbf{I}_N) - \text{tr}(\mathbf{I}_M) \\ &= N - M\end{aligned}$$

将此值代入式(9.3.46), 则

$$\mathbb{E}\{E_{\text{LS}}\} = (N - M)\sigma_{\text{opt}}^2 \quad (9.3.47)$$

即最小二乘滤波器的均方误差  $\mathbb{E}\{\mathbf{e}_{\text{LS}}^H \mathbf{e}_{\text{LS}}\}$  是最优滤波器的均方误差  $\sigma_{\text{opt}}^2$  的  $(N - M)$  倍。

## 9.4 投影矩阵和正交投影矩阵的更新

自适应滤波器是滤波器系数可以随时间作自适应调节的滤波器, 这种自适应调节依靠的是简单的时间更新公式, 因为复杂的计算不能够满足信号处理的实时要求。因此, 为了将投影矩阵和正交投影矩阵应用于自适应滤波器的设计中, 需要推导出这两种矩阵的时间更新公式。

假定目前的数据空间为  $\{\mathbf{U}\}$ , 相对应的投影矩阵是  $\mathbf{P}_U$ , 正交投影矩阵为  $\mathbf{P}_U^\perp$ 。这里,  $\mathbf{U}$  取  $\mathbf{X}_{1,m}(n)$  或  $\mathbf{X}_{0,m-1}(n)$  等形式。现在假设有一个新的数据向量  $\mathbf{u}$  加入到  $\{\mathbf{U}\}$  的原向量组中。一般说来, 新数据向量  $\mathbf{u}$  将提供某些新的信息, 它们是在  $\{\mathbf{U}\}$  的原向量组中没有包含的。由于数据子空间从  $\{\mathbf{U}\}$  扩大为  $\{\mathbf{U}, \mathbf{u}\}$ , 所以应该寻找与新子空间对应的“新的”投影矩阵  $\mathbf{P}_{U,u}$  和正交投影矩阵  $\mathbf{P}_{U,u}^\perp$ 。

从自适应更新的角度出发, 由已知的投影矩阵  $\mathbf{P}_U$  求更新的投影矩阵  $\mathbf{P}_{U,u}$  的最简单方法是将  $\mathbf{P}_{U,u}$  分解为两部分: 一部分是非自适应部分或已知部分, 另一部分为自适应更新部分。存在一种特别有用的方式, 即要求非自适应部分与自适应部分彼此正交。这样一种分解称为“正交分解”。具体说来, 投影矩阵  $\mathbf{P}_{U,u}$  的正交分解为

$$\mathbf{P}_{U,u} = \mathbf{P}_U + \mathbf{P}_w \quad (9.4.1)$$

式中,  $\mathbf{P}_w$  的选择应满足正交条件

$$\langle \mathbf{P}_U, \mathbf{P}_w \rangle = \mathbf{0} \quad (9.4.2)$$

或简记作  $\mathbf{P}_w \perp \mathbf{P}_U$ 。

由于更新是通过原投影矩阵  $\mathbf{P}_U$  和新数据向量  $\mathbf{u}$  实现的, 而  $\mathbf{P}_U$  中不包含新数据向量的任何作用, 所以正交分解中的更新部分  $\mathbf{P}_w$  应该包含有新数据向量  $\mathbf{u}$ 。不妨令  $\mathbf{w} = \mathbf{X}\mathbf{u}$ , 即  $\mathbf{P}_w = \mathbf{X}\mathbf{u}\langle \mathbf{X}\mathbf{u}, \mathbf{X}\mathbf{u} \rangle^{-1}(\mathbf{X}\mathbf{u})^T$ 。将  $\mathbf{P}_w$  代入正交条件式(9.4.2), 则有

$$\langle \mathbf{P}_U, \mathbf{P}_w \rangle = \mathbf{P}_U \mathbf{X}\mathbf{u} \langle \mathbf{X}\mathbf{u}, \mathbf{X}\mathbf{u} \rangle^{-1} (\mathbf{X}\mathbf{u})^T = \mathbf{0}$$

这里使用了投影矩阵的对称性  $\mathbf{P}_U^T = \mathbf{P}_U$ 。由于  $\mathbf{w} = \mathbf{X}\mathbf{u} \neq \mathbf{0}$ , 故上式意味着  $\mathbf{P}_U\mathbf{X}\mathbf{u} = \mathbf{0}$ ,  $\forall \mathbf{u}$  恒成立, 故  $\mathbf{P}_U\mathbf{X} = \mathbf{0}$ 。这意味着  $\mathbf{X}$  应该是正交投影矩阵  $\mathbf{P}_U^\perp$ 。

综合以上讨论, 得到正交分解式 (9.4.1) 中的向量  $\mathbf{w}$  为

$$\mathbf{w} = \mathbf{P}_U^\perp \mathbf{u} \quad (9.4.3)$$

即是说,  $\mathbf{w}$  是数据向量  $\mathbf{u}$  在数据矩阵  $\mathbf{U}$  的列空间上的正交投影。

利用投影矩阵的定义式和正交投影矩阵的对称性  $[\mathbf{P}_U^\perp]^T = \mathbf{P}_U^\perp$ , 易求得

$$\mathbf{P}_w = \mathbf{w}\langle \mathbf{w}, \mathbf{w} \rangle^{-1} \mathbf{w}^T = \mathbf{P}_U^\perp \mathbf{u} \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \mathbf{u}^T \mathbf{P}_U^\perp \quad (9.4.4)$$

将式 (9.4.4) 代入正交分解式 (9.4.1), 便得到“新的”投影矩阵的更新公式如下

$$\mathbf{P}_{U,u} = \mathbf{P}_U + \mathbf{P}_U^\perp \mathbf{u} \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \mathbf{u}^T \mathbf{P}_U^\perp \quad (9.4.5)$$

再使用正交投影矩阵的定义式, 又可得到“新的”正交投影矩阵的更新公式

$$\mathbf{P}_{U,u}^\perp = \mathbf{P}_U^\perp - \mathbf{P}_U^\perp \mathbf{u} \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \mathbf{u}^T \mathbf{P}_U^\perp \quad (9.4.6)$$

式 (9.4.5) 和式 (9.4.6) 分别组成了投影矩阵和正交投影矩阵的更新。用  $\mathbf{y}$  分别右乘式 (9.4.5) 和式 (9.4.6), 得到更新公式

$$\mathbf{P}_{U,u}\mathbf{y} = \mathbf{P}_U\mathbf{y} + \mathbf{P}_U^\perp \mathbf{u} \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \langle \mathbf{u}, \mathbf{P}_U^\perp \mathbf{y} \rangle \quad (9.4.7)$$

$$\mathbf{P}_{U,u}^\perp \mathbf{y} = \mathbf{P}_U^\perp \mathbf{y} - \mathbf{P}_U^\perp \mathbf{u} \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \langle \mathbf{u}, \mathbf{P}_U^\perp \mathbf{y} \rangle \quad (9.4.8)$$

若用向量  $\mathbf{z}$  左乘式 (9.4.7) 和式 (9.4.8), 又可得到更新公式

$$\langle \mathbf{z}, \mathbf{P}_{U,u}\mathbf{y} \rangle = \langle \mathbf{z}, \mathbf{P}_U\mathbf{y} \rangle + \langle \mathbf{z}, \mathbf{P}_U^\perp \mathbf{u} \rangle \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \langle \mathbf{u}, \mathbf{P}_U^\perp \mathbf{y} \rangle \quad (9.4.9)$$

$$\langle \mathbf{z}, \mathbf{P}_{U,u}^\perp \mathbf{y} \rangle = \langle \mathbf{z}, \mathbf{P}_U^\perp \mathbf{y} \rangle - \langle \mathbf{z}, \mathbf{P}_U^\perp \mathbf{u} \rangle \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \langle \mathbf{u}, \mathbf{P}_U^\perp \mathbf{y} \rangle \quad (9.4.10)$$

式 (9.4.5) 和式 (9.4.6) 分别组成了投影矩阵和正交投影矩阵的更新公式; 式 (9.4.7) 和式 (9.4.8) 分别是数据向量的投影和正交投影的更新公式; 而式 (9.4.9) 和式 (9.4.10) 则分别是与投影矩阵和正交投影矩阵有关的标量形式的更新公式。

## 9.5 满列秩矩阵的斜投影算子

前面几节详细讨论了正交投影算子的理论与应用。概括起来, 向量  $\mathbf{z}$  到子空间  $H$  的正交投影可以分解为该向量分别到子空间  $H_1$  和  $H_2$  的正交投影之和。其中, 子空间  $H_2$  要求是  $H_1$  的正交补, 并且正交投影算子本身必须同时是幂等的和复共轭对称的。本节讨论子空间  $H_2$  不是  $H_1$  的正交补的情况下投影, 从中引出一种不具有复共轭对称性的幂等算子。这类算子统称为斜投影算子 (oblique projector)。

斜投影算子最早是在 20 世纪 30 年代由 Murray<sup>[354]</sup> 和 Lorch<sup>[321]</sup> 先后提出的。后来, Afriat<sup>[13]</sup>, Lyantse<sup>[325]</sup>, Rao 与 Mitra<sup>[424]</sup>, Halmos<sup>[214]</sup>, Kato<sup>[260]</sup> 以及 Takeuchi 等

人<sup>[470]</sup>从数学角度作了进一步的论述与介绍。1989年, Kayalar与Weinert<sup>[262]</sup>提出在阵列信号处理中使用斜投影算子, 并且推导出了计算斜投影算子的一些新的公式和迭代算法。1994年, Behrent与Scharf<sup>[35]</sup>针对到矩阵的列空间的斜投影算子, 推导了更加实际的计算公式。2000年, Vandaele和Moonen<sup>[497]</sup>利用矩阵的LQ分解, 给出了到矩阵的行空间的斜投影公式。

由于斜投影算子的广泛应用, 本节和9.6节将分别针对满列秩和满行秩矩阵, 系统地讨论斜投影算子的有关理论方法和应用。

### 9.5.1 斜投影算子的定义及性质

令  $V$  是一 Hilbert 空间,  $H$  是  $V$  中由观测数据张成的一个闭合子空间。假定数据分成两个子集, 这两个子集张成两个闭合的子空间  $H_1$  和  $H_2$ , 并且满足  $H = H_1 + H_2$ , 其中,  $H_1$  和  $H_2$  的交集  $H_1 \cap H_2 = \{0\}$ , 即子空间  $H_1$  和  $H_2$  是无重叠的 (nonoverlapping) 或无交连的 (disjoint)。注意, 两个子空间无交连只是表明它们之间没有共同的非零元素, 并不意味这两个子空间正交。正交是比无交连更强的条件: 若两个子空间正交, 则它们一定是无交连的。

令  $\mathbf{P}, \mathbf{P}_1, \mathbf{P}_2$  分别是到子空间  $H, H_1, H_2$  上的正交投影算子。对向量  $\mathbf{z} \in V$ , 现在希望计算它在子空间  $H$  上的正交投影  $\mathbf{Pz}$ 。Aronszajn<sup>[23]</sup>提出, 可以先分别求出向量  $\mathbf{z}$  到子空间  $H_1$  和  $H_2$  的正交投影  $\mathbf{P}_1\mathbf{z}$  和  $\mathbf{P}_2\mathbf{z}$ , 然后再按照综合公式

$$\mathbf{Pz} = (\mathbf{I} - \mathbf{P}_2)(\mathbf{I} - \mathbf{P}_1\mathbf{P}_2)^{-1}\mathbf{P}_1\mathbf{z} + (\mathbf{I} - \mathbf{P}_1)(\mathbf{I} - \mathbf{P}_2\mathbf{P}_1)^{-1}\mathbf{P}_2\mathbf{z} \quad (9.5.1)$$

得到  $\mathbf{Pz}$ 。

上述 Aronszajn 综合公式可以直接解决某些统计内插问题<sup>[11, 398, 435]</sup>, 并且正如文献[262]所指出的那样, 所有双滤波器平滑公式都是 Aronszajn 综合公式的特例。

应当指出, 当子空间  $H_2$  是  $H_1$  的正交补时, 正交投影算子  $\mathbf{P}_2 = \mathbf{P}_1^\perp$  或  $\mathbf{P}_1 = \mathbf{P}_2^\perp$ 。此时, Aronszajn 综合公式式(9.5.1)简化为

$$\mathbf{Pz} = \mathbf{P}_1\mathbf{z} + \mathbf{P}_1^\perp\mathbf{z} = \mathbf{P}_2\mathbf{z} + \mathbf{P}_2^\perp\mathbf{z} \quad (9.5.2)$$

即为典型的正交分解。因此, Aronszajn 综合公式是正交分解的推广。

Aronszajn 综合方法利用两个非正交的子空间的正交投影算子  $\mathbf{P}_1$  和  $\mathbf{P}_2$  计算正交投影  $\mathbf{Pz}$ 。这一方法存在以下两个缺点:

- (1) 需要比较大的计算量。
- (2) 不容易推广到子空间  $H$  分解为三个或者多个子空间的情况。

Aronszajn 综合方法的这两个固有缺点可以通过计算非正交投影加以避免。这种非正交的投影就是下面将介绍的斜投影。

欲使正交投影  $\mathbf{Pz}$  的计算变得尽可能简单, 最简便的方法莫过于使  $\mathbf{Pz}$  是向量  $\mathbf{z}$  到子空间  $H_1$  和  $H_2$  的两个投影的直和, 即

$$\mathbf{Pz} = \mathbf{E}_1\mathbf{z} + \mathbf{E}_2\mathbf{z} \quad (9.5.3)$$

虽然仍然是综合正交投影  $\mathbf{P}\mathbf{z}$ , 但是与 Aronszajn 综合公式 (9.5.2) 不同, 式 (9.5.3) 中的两个投影矩阵不再是正交的:  $\mathbf{E}_1 \neq \mathbf{E}_2^\perp$ , 即两个子空间  $H_1$  与  $H_2$  相互不正交。此时, 称

$$H = H_1 \oplus H_2 \quad (9.5.4)$$

是子空间  $H$  的直和分解。注意, 算子  $\mathbf{E}_1$  和  $\mathbf{E}_2$  不再是正交投影算子。为了便于区别, 以后将统一使用符号  $\mathbf{P}$  和  $\mathbf{E}$  分别表示正交投影算子和非正交投影算子。非正交投影算子统称斜投影算子。

应当注意, 无论是正交投影算子, 还是斜投影算子, 都必须满足任何一个投影算子所必须具有的幂等性。容易验证, Aronszajn 综合公式 (9.5.1) 中的  $(\mathbf{I} - \mathbf{P}_2)(\mathbf{I} - \mathbf{P}_1\mathbf{P}_2)^{-1}\mathbf{P}_1$  和  $(\mathbf{I} - \mathbf{P}_1)(\mathbf{I} - \mathbf{P}_2\mathbf{P}_1)^{-1}\mathbf{P}_2$  都不是斜投影算子, 因为它们都不是幂等算子。

那么, 如何构造式 (9.5.3) 中的两个斜投影算子  $\mathbf{E}_1$  和  $\mathbf{E}_2$  呢? 下面以满列秩矩阵作为讨论对象。

令  $n \times m$  矩阵  $\mathbf{H}$  是一个满列秩矩阵, 其值域 (空间) 为

$$\text{Range}(\mathbf{H}) = \{\mathbf{y} \in \mathbb{C}^m \mid \mathbf{y} = \mathbf{H}\mathbf{x}, \mathbf{x} \in \mathbb{C}^n\} \quad (9.5.5)$$

并且

$$\text{Null}(\mathbf{H}) = \{\mathbf{x} \in \mathbb{C}^n \mid \mathbf{H}\mathbf{x} = \mathbf{0}\} \quad (9.5.6)$$

是  $\mathbf{H}$  的零空间。

现在考虑  $n \times m$  满列秩矩阵  $\mathbf{H}$  和  $n \times k$  满列秩矩阵  $\mathbf{S}$  组合成一个  $n \times (m+k)$  矩阵  $[\mathbf{H}, \mathbf{S}]$ , 其中,  $m+k < n$ , 使得矩阵  $[\mathbf{H}, \mathbf{S}]$  的列秩小于行数  $n$ , 并且  $\mathbf{H}$  的列向量与  $\mathbf{S}$  的列向量线性无关。由于  $\mathbf{H}$  的列向量与  $\mathbf{S}$  的列向量线性无关, 所以两个值域  $\text{Range}(\mathbf{H})$  和  $\text{Range}(\mathbf{S})$  是无交连的。

根据投影矩阵的定义, 到值域空间  $\text{Range}(\mathbf{H}, \mathbf{S})$  的正交投影算子为

$$\begin{aligned} \mathbf{P}_{HS} &= [\mathbf{H}, \mathbf{S}] \langle [\mathbf{H}, \mathbf{S}], [\mathbf{H}, \mathbf{S}] \rangle^{-1} [\mathbf{H}, \mathbf{S}]^H \\ &= [\mathbf{H}, \mathbf{S}] \begin{bmatrix} \mathbf{H}^H \mathbf{H} & \mathbf{H}^H \mathbf{S} \\ \mathbf{S}^H \mathbf{H} & \mathbf{S}^H \mathbf{S} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}^H \\ \mathbf{S}^H \end{bmatrix} \end{aligned} \quad (9.5.7)$$

式 (9.5.7) 表明, 正交投影算子  $\mathbf{P}_{HS}$  可以分解为<sup>[35]</sup>

$$\mathbf{P}_{HS} = \mathbf{E}_{H|S} + \mathbf{E}_{S|H} \quad (9.5.8)$$

式中

$$\mathbf{E}_{H|S} = [\mathbf{H}, \mathbf{O}] \begin{bmatrix} \mathbf{H}^H \mathbf{H} & \mathbf{H}^H \mathbf{S} \\ \mathbf{S}^H \mathbf{H} & \mathbf{S}^H \mathbf{S} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}^H \\ \mathbf{S}^H \end{bmatrix} \quad (9.5.9)$$

$$\mathbf{E}_{S|H} = [\mathbf{O}, \mathbf{S}] \begin{bmatrix} \mathbf{H}^H \mathbf{H} & \mathbf{H}^H \mathbf{S} \\ \mathbf{S}^H \mathbf{H} & \mathbf{S}^H \mathbf{S} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}^H \\ \mathbf{S}^H \end{bmatrix} \quad (9.5.10)$$

这里,  $\mathbf{O}$  代表零矩阵。

利用第1章的分块矩阵求逆引理公式(1.7.11), 易求得

$$\begin{aligned}\mathbf{A}^{-1} &= \begin{bmatrix} \mathbf{H}^{\mathbb{H}}\mathbf{H} & \mathbf{H}^{\mathbb{H}}\mathbf{S} \\ \mathbf{S}^{\mathbb{H}}\mathbf{H} & \mathbf{S}^{\mathbb{H}}\mathbf{S} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} (\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1} & -(\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}}\mathbf{S}(\mathbf{S}^{\mathbb{H}}\mathbf{S})^{-1} \\ -(\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp}\mathbf{S})^{-1}\mathbf{S}^{\mathbb{H}}\mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{H})^{-1} & (\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp}\mathbf{S})^{-1} \end{bmatrix} \quad (9.5.11)\end{aligned}$$

式中

$$\begin{aligned}\mathbf{P}_S^{\perp} &= \mathbf{I} - \mathbf{P}_S = \mathbf{I} - \mathbf{S}(\mathbf{S}^{\mathbb{H}}\mathbf{S})^{-1}\mathbf{S}^{\mathbb{H}} \\ \mathbf{P}_H^{\perp} &= \mathbf{I} - \mathbf{P}_H = \mathbf{I} - \mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}}\end{aligned}$$

将式(9.5.11)代入式(9.5.9), 立即得到

$$\begin{aligned}\mathbf{E}_{H|S} &= [\mathbf{H}, \mathbf{O}] \begin{bmatrix} (\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1} & -(\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}}\mathbf{S}(\mathbf{S}^{\mathbb{H}}\mathbf{S})^{-1} \\ -(\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp}\mathbf{S})^{-1}\mathbf{S}^{\mathbb{H}}\mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{H})^{-1} & (\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp}\mathbf{S})^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{H}^{\mathbb{H}} \\ \mathbf{S}^{\mathbb{H}} \end{bmatrix} \\ &= \mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}} - \mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}}\mathbf{P}_S\end{aligned}$$

整理后, 得到

$$\mathbf{E}_{H|S} = \mathbf{H}(\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp}\mathbf{H})^{-1}\mathbf{H}^{\mathbb{H}}\mathbf{P}_S^{\perp} \quad (9.5.12)$$

类似地, 将式(9.5.11)代入式(9.5.10), 又可得到

$$\mathbf{E}_{S|H} = \mathbf{S}(\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp}\mathbf{S})^{-1}\mathbf{S}^{\mathbb{H}}\mathbf{P}_H^{\perp} \quad (9.5.13)$$

式(9.5.12)和式(9.5.13)是Behrent与Scharf于1994年得到的<sup>[35]</sup>。

观察式(9.5.12)和式(9.5.13)知, 幂等算子  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  都不是复共轭对称即Hermitian的, 所以它们虽然是投影算子, 但不是正交投影算子。

**定义 9.5.1** 一个不具有复共轭对称性的幂等算子  $\mathbf{E}$  称为斜投影算子。

根据定义, 由式(9.5.12)定义的算子  $\mathbf{E}_{H|S}$  和由式(9.5.13)定义的算子  $\mathbf{E}_{S|H}$  今后称为斜投影算子。

斜投影算子  $\mathbf{E}_{H|S}$  读作“沿着与子空间 Range( $\mathbf{S}$ )平行的方向, 到子空间 Range( $\mathbf{H}$ )上的投影算子”。类似地, 斜投影算子  $\mathbf{E}_{S|H}$  则读作“沿着与子空间 Range( $\mathbf{H}$ )平行的方向, 到子空间 Range( $\mathbf{S}$ )上的投影算子”。这一称呼给出了斜投影算子的几何解释。以斜投影算子  $\mathbf{E}_{H|S}$  为例, 其投影方向与子空间 Range( $\mathbf{S}$ )平行, 故投影算子  $\mathbf{E}_{H|S}$  的子空间与子空间 Range( $\mathbf{S}$ )不可能有任何交连。由于  $\mathbf{E}_{H|S}\mathbf{H}$  是  $\mathbf{H}$  沿着与子空间 Range( $\mathbf{S}$ )平行的方向, 到子空间 Range( $\mathbf{H}$ )上的投影, 而  $\mathbf{H}$  本身位于子空间 Range( $\mathbf{H}$ ), 所以斜投影  $\mathbf{E}_{H|S}\mathbf{H}$  的结果为  $\mathbf{H}$ , 即  $\mathbf{E}_{H|S}\mathbf{H} = \mathbf{H}$ 。

斜投影算子是正交投影算子的扩展, 而正交投影算子则是斜投影算子的一个特例。下面汇总了斜投影算子的一些重要性质<sup>[35]</sup>:

(1)  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  均为幂等算子, 即有

$$\mathbf{E}_{H|S}^2 = \mathbf{E}_{H|S}, \quad \mathbf{E}_{S|H}^2 = \mathbf{E}_{S|H}$$

(2)  $\mathbf{E}_{H|S}[\mathbf{H}, \mathbf{S}] = [\mathbf{H}, \mathbf{O}]$  和  $\mathbf{E}_{S|H}[\mathbf{H}, \mathbf{S}] = [\mathbf{O}, \mathbf{S}]$ , 或者等价为

$$\begin{aligned}\mathbf{E}_{H|S}\mathbf{H} &= \mathbf{H}, & \mathbf{E}_{H|S}\mathbf{S} &= \mathbf{O} \\ \mathbf{E}_{S|H}\mathbf{H} &= \mathbf{O}, & \mathbf{E}_{S|H}\mathbf{S} &= \mathbf{S}\end{aligned}$$

(3) 斜投影算子  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  的交叉项为零, 即有

$$\mathbf{E}_{H|S}\mathbf{E}_{S|H} = \mathbf{O}, \quad \mathbf{E}_{S|H}\mathbf{E}_{H|S} = \mathbf{O}$$

(4) 斜投影后, 再正交投影, 不会改变原斜投影

$$\mathbf{E}_{H|S} = \mathbf{P}_H\mathbf{E}_{H|S}, \quad \mathbf{E}_{S|H} = \mathbf{P}_S\mathbf{E}_{S|H}$$

(5) 令  $\mathbf{B}^\dagger = (\mathbf{B}^H\mathbf{B})^{-1}\mathbf{B}^H$  表示矩阵  $\mathbf{B}$  的广义逆矩阵, 则

$$\mathbf{H}^\dagger\mathbf{E}_{H|S} = (\mathbf{P}_S^\perp\mathbf{H})^\dagger, \quad \mathbf{S}^\dagger\mathbf{E}_{S|H} = (\mathbf{P}_H^\perp\mathbf{S})^\dagger$$

(6) 斜投影矩阵与正交投影矩阵的关系

$$\mathbf{P}_S^\perp\mathbf{E}_{H|S}\mathbf{P}_S^\perp = \mathbf{P}_S^\perp\mathbf{P}_{\mathbf{P}_S^\perp\mathbf{H}}\mathbf{P}_S^\perp = \mathbf{P}_{\mathbf{P}_S^\perp\mathbf{H}}$$

(7) 若子空间  $\text{Range}(\mathbf{H})$  与  $\text{Range}(\mathbf{S})$  正交, 即  $\text{Range}(\mathbf{H}) \perp \text{Range}(\mathbf{S})$ , 则

$$\mathbf{E}_{H|S} = \mathbf{P}_H, \quad \mathbf{E}_{S|H} = \mathbf{P}_H^\perp$$

下面是关于上述性质的几点注释。

注释 1 性质 (2) 表明:

①  $\mathbf{E}_{H|S}$  的值域是  $\text{Range}(\mathbf{H})$ , 而  $\mathbf{E}_{H|S}$  的零空间包含  $\text{Range}(\mathbf{S})$ , 即有

$$\text{Range}(\mathbf{E}_{H|S}) = \text{Range}(\mathbf{H}), \quad \text{Range}(\mathbf{S}) \subset \text{Null}(\mathbf{E}_{H|S}) \quad (9.5.14)$$

②  $\mathbf{E}_{S|H}$  的值域是  $\text{Range}(\mathbf{S})$ , 而  $\mathbf{E}_{S|H}$  的零空间包含  $\text{Range}(\mathbf{H})$ , 即有

$$\text{Range}(\mathbf{E}_{S|H}) = \text{Range}(\mathbf{S}), \quad \text{Range}(\mathbf{H}) \subset \text{Null}(\mathbf{E}_{S|H}) \quad (9.5.15)$$

换言之, 斜投影算子  $\mathbf{E}_{H|S}$  的值域在  $\mathbf{E}_{S|H}$  的零空间内, 而  $\mathbf{E}_{S|H}$  的值域则在  $\mathbf{E}_{H|S}$  的零空间内。

注释 2 由于斜投影算子  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  的交叉项为零, 故值域空间  $\text{Range}(\mathbf{E}_{H|S})$  和  $\text{Range}(\mathbf{E}_{S|H})$  是无交连的。

注释 3 性质 (4) 也是性质 (2) 的体现, 与斜投影算子的几何解释吻合: 斜投影  $\mathbf{E}_{H|S}$  是沿着与子空间  $\text{Range}(\mathbf{S})$  平行的方向, 到子空间  $\text{Range}(\mathbf{H})$  上的投影, 即有  $\text{Range}(\mathbf{E}_{H|S}) = \text{Range}(\mathbf{H})$ 。因此, 斜投影  $\mathbf{E}_{H|S}$  再向子空间  $\text{Range}(\mathbf{H})$  上投影, 其结果将不会发生任何变化。

注释 4 性质 (5) 和性质 (6) 可视为斜投影算子与正交投影算子之间的关系。

### 9.5.2 斜投影算子的几何解释

总结以上讨论，可以得到三个投影算子各自的含义如下：

- (1) 正交投影算子  $P_{HS}$  是到合成矩阵  $[H, S]$  的列向量张成的值域空间  $\text{Range}(H, S)$  上的正交投影算子。
- (2) 斜投影算子  $E_{H|S}$  是沿着值域空间  $\text{Range}(S)$ ，到另一值域空间  $\text{Range}(H)$  的投影算子。
- (3) 斜投影算子  $E_{S|H}$  是沿着值域空间  $\text{Range}(H)$ ，到值域空间  $\text{Range}(S)$  的投影算子。

前面已经解释过，斜投影算子  $E_{H|S}$  的值域是  $\text{Range}(H)$ ，其零空间包含  $\text{Range}(S)$ 。为了完整地表达斜投影算子  $E_{H|S}$  的零空间，定义矩阵  $A$  是合成矩阵  $[H, S]$  的正交矩阵，即有  $[H, S]^H A = O$ ，从而有

$$H^H A = O, \quad S^H A = O \quad (9.5.16)$$

第二式左乘满列秩矩阵  $S(S^H S)^{-1}$ ，即可将  $S^H A = O$  等价写作

$$P_S A = O \implies P_S^\perp A = A \quad (9.5.17)$$

利用这些结果，由式 (9.5.12) 得

$$\begin{aligned} E_{H|S} A &= H(H^H P_S^\perp H)^{-1} H^H P_S^\perp A \\ &= H(H^H P_S^\perp H)^{-1} H^H A = O \end{aligned}$$

这说明，矩阵  $A$  的列张成的值域子空间  $\text{Range}(A)$  也在斜投影算子  $E_{H|S}$  的零空间内，即  $\text{Range}(A) \subset \text{Null}(E_{H|S})$ 。

式 (9.5.16) 表明，值域子空间  $\text{Range}(A)$  与其他两个值域子空间  $\text{Range}(H)$  和  $\text{Range}(S)$  分别正交。于是，可以利用三个空间方向  $\text{Range}(H)$ ,  $\text{Range}(S)$  和  $\text{Range}(A)$  作为由  $n \times (m+k)$  矩阵  $[H, S]$  的列向量张成的 Euclidean 空间  $C^{m+k}$  的坐标轴，如图 9.5.1 所示。

图 9.5.1 中，坐标轴  $\text{Range}(A)$  与另外两个坐标轴  $\text{Range}(H)$ ,  $\text{Range}(S)$  垂直，但水平面上的坐标轴  $\text{Range}(H)$  和  $\text{Range}(S)$  不相互垂直。换句话说，Euclidean 空间可以分解为三个方向的直和

$$C^{m+k} = \text{Range}(H) \oplus \text{Range}(S) \oplus \text{Range}(A) \quad (9.5.18)$$

图 9.5.1 (a) 所示的正交投影已经在前面解说过，现对图 9.5.1 (b) 所示的斜投影解释如下：当向量  $y$  位于两个坐标轴  $\text{Range}(H)$  和  $\text{Range}(S)$  组成的水平面上时，向量  $y$  沿着与  $\text{Range}(S)$  平行的方向，到  $\text{Range}(H)$  的斜投影  $E_{H|S}y$  满足以下两个条件：

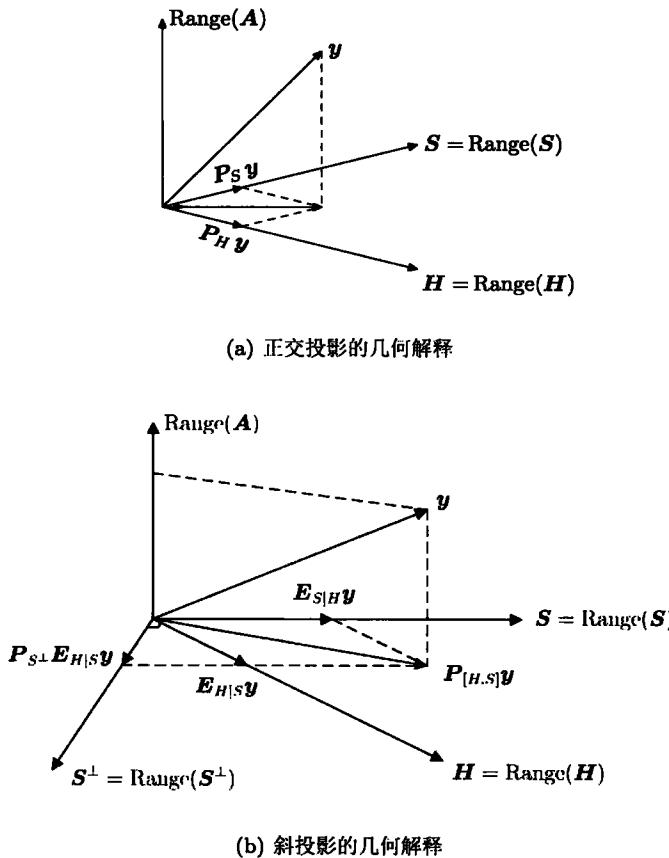


图 9.5.1 正交投影与斜投影

- ① 斜投影  $E_{H|S}y$  位于坐标轴  $\text{Range}(H)$  上, 即  $E_{H|S}y \in \text{Range}(H)$ ;
- ② 斜投影  $E_{H|S}y$  的端点到向量  $y$  的端点之间的连线与坐标轴  $\text{Range}(S)$  平行, 即  $E_{H|S}y \notin \text{Range}(S)$ .

如图 9.5.1 (b) 所示, 对于 Euclidean 空间中的向量  $y$  而言, 斜投影  $E_{H|S}y$  的构造分为以下两个步骤:

- (1) 向量  $y$  先正交投影到坐标轴  $\text{Range}(H)$  和  $\text{Range}(S)$  组成的平面上, 即投影  $P_{HS}y$  是向量  $y$  在矩阵  $[H, S]$  的列张成的值域  $\text{Range}(H, S)$  上的正交投影, 记作  $y_{HS} = P_{HS}y$ 。
- (2) 利用图 9.5.1 (a) 的方法, 求正交投影  $y_{HS}$  沿着与值域  $\text{Range}(S)$  平行的方向, 到值域  $\text{Range}(H)$  的斜投影  $E_{H|S}y_{HS}$ 。

斜投影算子的物理含义是: 向量  $y$  的斜投影  $E_{H|S}y$  是向量  $y$  沿着值域  $\text{Range}(S)$  的方向, 到值域  $\text{Range}(H)$  的投影。翻译成信号处理的语言, 即是“斜投影  $E_{H|S}y$  抽取向量  $y$  在特定方向 (值域  $\text{Range}(H)$ ) 的分量, 并完全对消掉向量  $y$  在另一个方向 (值域  $\text{Range}(S)$ ) 的所有分量”。

从图 9.5.1 还可看出, 投影  $\mathbf{P}_A\mathbf{y}$  是向量  $\mathbf{y}$  到值域  $\text{Range}(\mathbf{A})$  的正交投影。其中, 投影  $\mathbf{P}_A\mathbf{y}$  不仅与  $\mathbf{E}_{H|S}\mathbf{y}$  正交, 而且也与  $\mathbf{E}_{S|H}\mathbf{y}$  正交, 即有  $\mathbf{P}_A\mathbf{y} \perp \mathbf{E}_{H|S}\mathbf{y}$  和  $\mathbf{P}_A\mathbf{y} \perp \mathbf{E}_{S|H}\mathbf{y}$ 。

数学上, 斜投影属于平行投影的一种。一个三维点  $(x, y, z)$  到平面  $(x, y)$  的平行投影的结果为  $(x + az, y + bz, 0)$ 。常数  $a$  和  $b$  唯一决定一平行投影。若  $a = b = 0$ , 则平行投影的结果称为“正交图形”(orthographic) 或正交投影。否则, 称为斜投影。

在图像处理中, 斜投影是一种抽取图形投影的技术, 用于产生三维物体的二维画面即二维图像。

### 9.5.3 斜投影算子的递推

令  $\mathbf{H}$  为已知数据矩阵,  $H = \text{Range}(\mathbf{H})$  是  $\mathbf{H}$  的值域空间(即列空间),  $\mathbf{E}_{H|S}\mathbf{x}$  是向量  $\mathbf{x}$  沿着  $S$  空间的方向到  $H$  空间的斜投影。现在, 增加一新的数据矩阵  $\mathbf{V}$ , 问题是如何利用已经求出的斜投影  $\mathbf{E}_{H|S}\mathbf{x}$  和新数据矩阵  $\mathbf{V}$ , 递推计算新的斜投影  $\mathbf{E}_{\tilde{H}|S}\mathbf{x}$ , 其中  $\tilde{H} = \text{Range}(\tilde{\mathbf{H}})$ , 而  $\tilde{\mathbf{H}} = [\mathbf{H}, \mathbf{V}]$ 。

**定理 9.5.1**<sup>[401]</sup> 若  $\tilde{\mathbf{H}} = [\mathbf{H}, \mathbf{V}]$  和  $\tilde{H} = \text{Range}(\tilde{\mathbf{H}})$ , 并且两个值域空间  $\tilde{H}$  和  $S = \text{Range}(\mathbf{S})$  无交连, 则新的斜投影矩阵由以下递推公式给出

$$\mathbf{E}_{\tilde{H}|S} = \mathbf{E}_{H|S} + \mathbf{E}_{\tilde{V}|S} \quad (9.5.19)$$

$$\mathbf{E}_{S|\tilde{H}} = \mathbf{E}_{S|H} - \mathbf{P}_S \mathbf{E}_{\tilde{V}|S} \quad (9.5.20)$$

其中,  $\tilde{V} = \text{Range}(\tilde{\mathbf{V}})$ ,  $\tilde{\mathbf{V}} = \mathbf{V} - \mathbf{E}_{H|S}\mathbf{V}$ , 而  $\mathbf{P}_H$  是矩阵  $\mathbf{H}$  的投影矩阵。

定理 9.5.1 的下述三条注释有助于理解斜投影递推与正交投影递推之间的关系以及斜投影矩阵的作用。

**注释 1** 若子空间  $\tilde{H}$  与  $S$  正交, 则斜投影矩阵

$$\mathbf{E}_{\tilde{H}|S} = \mathbf{P}_{\tilde{H}}, \quad \mathbf{E}_{H|S} = \mathbf{P}_H$$

$$\mathbf{E}_{S|\tilde{H}} = \mathbf{P}_{\tilde{H}}^\perp, \quad \mathbf{E}_{S|H} = \mathbf{P}_H^\perp$$

此时,  $\tilde{\mathbf{V}} = (\mathbf{I} - \mathbf{P}_H)\mathbf{V} = \mathbf{P}_H^\perp\mathbf{V}$ 。于是有

$$\mathbf{E}_{\tilde{V}} = \mathbf{P}_{\tilde{V}}, \quad \mathbf{P}_S \mathbf{P}_{\tilde{V}} = \mathbf{P}_{\tilde{V}}$$

将以上关系代入式 (9.5.19) 和式 (9.5.20), 分别得

$$\mathbf{P}_{\tilde{H}} = \mathbf{P}_H + \mathbf{P}_H^\perp \mathbf{V} \langle \mathbf{P}_H^\perp, \mathbf{P}_H^\perp \mathbf{V} \rangle^{-1} \mathbf{V}^T \mathbf{P}_H^\perp$$

$$\mathbf{P}_{\tilde{H}}^\perp = \mathbf{P}_H^\perp - \mathbf{P}_H^\perp \mathbf{V} \langle \mathbf{P}_H^\perp, \mathbf{P}_H^\perp \mathbf{V} \rangle^{-1} \mathbf{V}^T \mathbf{P}_H^\perp$$

它们恰好分别是投影矩阵与正交投影矩阵的递推公式。因此, 定理 9.5.1 是投影矩阵与正交投影矩阵在  $\tilde{H}$  与  $S$  非正交情况下的推广。

**注释 2** 定理 9.5.1 可以从新息过程的角度进行解释。根据 Kalman 滤波理论, 在正交投影的情况下,  $\mathbf{P}_H\mathbf{V}$  代表数据矩阵  $\mathbf{V}$  的均方估计,  $\tilde{\mathbf{V}} = \mathbf{V} - \mathbf{P}_H\mathbf{V}$  可以视为数据矩阵

$\mathbf{V}$  的新息矩阵, 而  $\text{Range}(\tilde{\mathbf{V}})$  则表示正交投影中的新息子空间。类似地,  $\mathbf{E}_{H|S}\mathbf{V}$  是数据矩阵  $\mathbf{V}$  在子空间  $H$  内沿着无交连的子空间  $S$  的均方估计,  $\tilde{\mathbf{V}} = \mathbf{V} - \mathbf{E}_{H|S}\mathbf{V}$  可以视为数据矩阵  $\mathbf{V}$  在  $H$  内沿着  $S$  的新息矩阵, 而  $\text{Range}(\tilde{\mathbf{V}})$  则表示斜投影中的新息子空间。

注释3 从子空间的观点出发, 向量空间  $C^n = H \oplus S \oplus (H \oplus S)^\perp$ , 其中  $H, S$  和  $(H \oplus S)^\perp$  分别代表期望信号(值域)、结构性噪声(干扰)和非结构性噪声子空间。由定理 9.5.1 及  $\mathbf{P}_{(H,S)} = \mathbf{E}_{H|S} + \mathbf{E}_{S|H}$ , 易知

$$\tilde{\mathbf{V}} = (\mathbf{I} - \mathbf{E}_{H|S})\mathbf{V} = \mathbf{E}_{S|H}\mathbf{V} + \mathbf{P}_{(H,S)}^\perp\mathbf{V} = \mathbf{V}_S + \mathbf{V}_{(H,S)^\perp}$$

即是说, 数据矩阵  $\mathbf{V}$  在  $H$  内沿  $S$  的新息矩阵  $\tilde{\mathbf{V}}$  由两个分量组成:  $\mathbf{V}_S = \mathbf{E}_{S|H}\mathbf{V}$  是子空间  $S$  内的结构化噪声的新息矩阵,  $\mathbf{V}_{(H,S)^\perp}\mathbf{V}$  是在正交补子空间  $(H \oplus S)^\perp$  内的非结构化噪声的新息矩阵。若结构化噪声相比非结构化噪声可以忽略不计, 则新息矩阵  $\tilde{\mathbf{V}} = \mathbf{V}_{(H,S)^\perp}$ 。在这种情况下,  $\mathbf{P}_S^\perp\tilde{\mathbf{V}} \approx \mathbf{P}_S^\perp\mathbf{V}_{(H,S)^\perp} = \mathbf{V}_{(H,S)^\perp}$ , 并且

$$\mathbf{E}_{\tilde{\mathbf{V}}|S} = \tilde{\mathbf{V}}(\tilde{\mathbf{V}}^H \mathbf{P}_S^\perp \tilde{\mathbf{V}})^{-1} \mathbf{P}_S^\perp = \mathbf{V}_{(H,S)^\perp}(\mathbf{V}_{(H,S)^\perp}^H \mathbf{V}_{(H,S)^\perp})^{-1} \mathbf{V}_{(H,S)^\perp}^H$$

它正好就是  $\mathbf{V}_{(H,S)^\perp}$  的投影矩阵。

## 9.6 满行秩矩阵的斜投影算子

在前面几节关于正交投影算子和斜正交投影算子的讨论中, 均以满列秩的矩阵作为讨论对象。如果是满行秩矩阵, 则相对应的正交投影算子和斜投影算子具有不同的定义与表达形式。

### 9.6.1 满行秩矩阵的斜投影算子定义

若  $\mathbf{D} \in \mathbb{C}^{m \times k}$  具有满行秩, 即  $\text{rank}(\mathbf{D}) = m$  ( $m < k$ ), 则其投影矩阵  $\mathbf{P}_D$  是一个  $k \times k$  矩阵, 定义为

$$\mathbf{P}_D = \mathbf{D}^H (\mathbf{D}\mathbf{D}^H)^{-1} \mathbf{D} \quad (9.6.1)$$

考查矩阵  $\mathbf{B} \in \mathbb{C}^{m \times k}$  和  $\mathbf{C} \in \mathbb{C}^{n \times k}$ , 它们都是满行秩矩阵, 即  $\text{rank}(\mathbf{B}) = m, \text{rank}(\mathbf{C}) = n$ , 并且它们组合成的矩阵  $\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \in \mathbb{C}^{(m+n) \times k}$  也是满行秩的, 即其秩为  $m+n$ , 其中,  $m+n < k$ 。这意味着, 矩阵  $\mathbf{B}$  的行向量与  $\mathbf{C}$  的行向量线性无关。从而, 矩阵  $\mathbf{B}$  的行向量张成的值域空间(简称行空间)  $\mathcal{Z}_B$  和矩阵  $\mathbf{C}$  的行空间  $\mathcal{Z}_C$  是无交连的。

令  $\mathbf{D} = \begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix}$ , 并代入式 (9.6.1), 可求得正交投影算子

$$\mathbf{P}_D = [\mathbf{B}^H, \mathbf{C}^H] \begin{bmatrix} \mathbf{B}\mathbf{B}^H & \mathbf{B}\mathbf{C}^H \\ \mathbf{C}\mathbf{B}^H & \mathbf{C}\mathbf{C}^H \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \quad (9.6.2)$$

它可分解为

$$\mathbf{P}_D = \mathbf{E}_{\mathcal{Z}_B|\mathcal{Z}_C} + \mathbf{E}_{\mathcal{Z}_C|\mathcal{Z}_B} \quad (9.6.3)$$

式中

$$\mathbf{E}_{Z_B|Z_C} = [\mathbf{B}^H, \mathbf{C}^H] \begin{bmatrix} \mathbf{B}\mathbf{B}^H & \mathbf{B}\mathbf{C}^H \\ \mathbf{C}\mathbf{B}^H & \mathbf{C}\mathbf{C}^H \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \quad (9.6.4)$$

$$\mathbf{E}_{Z_C|Z_B} = [\mathbf{B}^H, \mathbf{C}^H] \begin{bmatrix} \mathbf{B}\mathbf{B}^H & \mathbf{B}\mathbf{C}^H \\ \mathbf{C}\mathbf{B}^H & \mathbf{C}\mathbf{C}^H \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{O} \\ \mathbf{C} \end{bmatrix} \quad (9.6.5)$$

用矩阵  $\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix}$  分别左乘以上两式，立即有

$$\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \mathbf{E}_{Z_B|Z_C} = \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \quad (9.6.6)$$

$$\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \mathbf{E}_{Z_C|Z_B} = \begin{bmatrix} \mathbf{O} \\ \mathbf{C} \end{bmatrix} \quad (9.6.7)$$

或等价为

$$\mathbf{B}\mathbf{E}_{Z_B|Z_C} = \mathbf{B}, \quad \mathbf{C}\mathbf{E}_{Z_B|Z_C} = \mathbf{O} \quad (9.6.8)$$

$$\mathbf{B}\mathbf{E}_{Z_C|Z_B} = \mathbf{O}, \quad \mathbf{C}\mathbf{E}_{Z_C|Z_B} = \mathbf{C} \quad (9.6.9)$$

这表明，算子  $\mathbf{E}_{Z_B|Z_C}$  与  $\mathbf{E}_{Z_C|Z_B}$  之间无交叉项，即

$$\mathbf{E}_{Z_B|Z_C} \mathbf{E}_{Z_C|Z_B} = \mathbf{O} \quad \text{和} \quad \mathbf{E}_{Z_C|Z_B} \mathbf{E}_{Z_B|Z_C} = \mathbf{O} \quad (9.6.10)$$

由正交投影算子  $\mathbf{P}_D$  的幂等性，得

$$\begin{aligned} \mathbf{P}_D^2 &= (\mathbf{E}_{Z_B|Z_C} + \mathbf{E}_{Z_C|Z_B})(\mathbf{E}_{Z_B|Z_C} + \mathbf{E}_{Z_C|Z_B}) \\ &= \mathbf{E}_{Z_B|Z_C}^2 + \mathbf{E}_{Z_C|Z_B}^2 \\ &= \mathbf{P}_D = \mathbf{E}_{Z_B|Z_C} + \mathbf{E}_{Z_C|Z_B} \end{aligned}$$

由此有

$$\mathbf{E}_{Z_B|Z_C}^2 = \mathbf{E}_{Z_B|Z_C}, \quad \mathbf{E}_{Z_C|Z_B}^2 = \mathbf{E}_{Z_C|Z_B} \quad (9.6.11)$$

由于具有幂等性，故  $\mathbf{E}_{Z_B|Z_C}$  和  $\mathbf{E}_{Z_C|Z_B}$  均为投影算子。又由式 (9.6.8) 和式 (9.6.9) 知， $\mathbf{E}_{Z_B|Z_C}$  和  $\mathbf{E}_{Z_C|Z_B}$  具有斜投影的几何意义。

满行秩矩阵的斜投影算子具有以下有用性质 [497]。

**性质 1** 若矩阵  $\mathbf{B}$  的行空间与  $\mathbf{C}$  的行空间正交，即  $\mathbf{B}\mathbf{C}^H = \mathbf{O}$ ,  $\mathbf{C}\mathbf{B}^H = \mathbf{O}$ ，则斜投影退化为正交投影

$$\mathbf{E}_{Z_B|Z_C} = \mathbf{B}^H(\mathbf{B}\mathbf{B}^H)^{-1}\mathbf{B} = \mathbf{P}_B \quad (9.6.12)$$

$$\mathbf{E}_{Z_C|Z_B} = \mathbf{C}^H(\mathbf{C}\mathbf{C}^H)^{-1}\mathbf{C} = \mathbf{P}_C \quad (9.6.13)$$

**证明** 将正交条件  $\mathbf{B}\mathbf{C}^H = \mathbf{O}$  和  $\mathbf{C}\mathbf{B}^H = \mathbf{O}$  代入式 (9.6.4), 立即有

$$\begin{aligned}\mathbf{E}_{\mathbf{Z}_B|\mathbf{Z}_C} &= [\mathbf{B}^H, \mathbf{C}^H] \begin{bmatrix} \mathbf{B}\mathbf{B}^H & \mathbf{O} \\ \mathbf{O} & \mathbf{C}\mathbf{C}^H \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \\ &= [\mathbf{B}^H, \mathbf{C}^H] \begin{bmatrix} (\mathbf{B}\mathbf{B}^H)^{-1} & \mathbf{O} \\ \mathbf{O} & (\mathbf{C}\mathbf{C}^H)^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \\ &= \mathbf{B}^H(\mathbf{B}\mathbf{B}^H)^{-1}\mathbf{B} \\ &= \mathbf{P}_B\end{aligned}$$

类似地, 可以证明  $\mathbf{E}_{\mathbf{Z}_C|\mathbf{Z}_B} = \mathbf{C}^H(\mathbf{C}\mathbf{C}^H)^{-1}\mathbf{C} = \mathbf{P}_C$ 。 ■

**性质 2** 若 (1)  $\mathbf{A} = \mathbf{MB} + \mathbf{NC}$ ; (2) 矩阵  $\mathbf{B}$  和  $\mathbf{C}$  的行空间无交连, 则

$$\mathbf{AE}_{\mathbf{Z}_B|\mathbf{Z}_C} = \mathbf{MB} \quad (9.6.14)$$

$$\mathbf{AE}_{\mathbf{Z}_C|\mathbf{Z}_B} = \mathbf{NC} \quad (9.6.15)$$

**证明** 由条件 (1) 知

$$\begin{aligned}\mathbf{AE}_{\mathbf{Z}_B|\mathbf{Z}_C} &= (\mathbf{MB} + \mathbf{NC})\mathbf{E}_{\mathbf{Z}_B|\mathbf{Z}_C} \\ &= \mathbf{MBE}_{\mathbf{Z}_B|\mathbf{Z}_C} + \mathbf{NCE}_{\mathbf{Z}_B|\mathbf{Z}_C}\end{aligned}$$

但是, 在条件 (2) 之下, 式 (9.6.8) 成立。将式 (9.6.8) 代入上式, 立即得  $\mathbf{AE}_{\mathbf{Z}_B|\mathbf{Z}_C} = \mathbf{MB}$ 。类似地, 可以证明  $\mathbf{AE}_{\mathbf{Z}_C|\mathbf{Z}_B} = \mathbf{NC}$ 。 ■

## 9.6.2 斜投影的计算

一个  $m \times n$  ( $m > n$ ) 实矩阵  $\mathbf{A}$  的 QR 分解为

$$\mathbf{Q}^T \mathbf{A} = \begin{bmatrix} \mathbf{R} \\ \mathbf{O} \end{bmatrix} \quad (9.6.16)$$

式中,  $\mathbf{Q}$  为  $m \times m$  正交矩阵,  $\mathbf{R}$  为上三角矩阵。

若  $\mathbf{B}$  是一个列数大于行数的实矩阵  $\mathbf{B}$ , 则只要令  $\mathbf{B} = \mathbf{A}^T$ , 并取 QR 分解式 (9.6.16) 的转置, 即可得到矩阵  $\mathbf{B}$  的 LQ 分解如下

$$\mathbf{BQ} = \mathbf{A}^T \mathbf{Q} = [\mathbf{L}, \mathbf{O}] \quad (9.6.17)$$

或

$$\mathbf{B} = [\mathbf{L}, \mathbf{O}] \mathbf{Q}^T \quad (9.6.18)$$

式中,  $\mathbf{L} = \mathbf{R}^T$  为下三角矩阵。

令  $\mathbf{B} \in \mathbb{R}^{m \times k}$ ,  $\mathbf{C} \in \mathbb{R}^{n \times k}$ ,  $\mathbf{A} \in \mathbb{R}^{p \times k}$ , 其中,  $(m+n+p) < k$ , 则矩阵  $[\mathbf{B}^T, \mathbf{C}^T, \mathbf{A}^T]^T$  的 LQ 分解为

$$\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \\ \mathbf{A} \end{bmatrix} [\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_3] = \begin{bmatrix} \mathbf{L}_{11} & & \\ \mathbf{L}_{21} & \mathbf{L}_{22} & \\ \mathbf{L}_{31} & \mathbf{L}_{32} & \mathbf{L}_{33} \end{bmatrix} \quad (9.6.19)$$

式中,  $\mathbf{Q}_1 \in \mathbb{R}^{k \times m}$ ,  $\mathbf{Q}_2 \in \mathbb{R}^{k \times n}$ ,  $\mathbf{Q}_3 \in \mathbb{R}^{k \times p}$  为正交矩阵, 即  $\mathbf{Q}_i^T \mathbf{Q}_i = \mathbf{I}$  和  $\mathbf{Q}_i^T \mathbf{Q}_j = \mathbf{O}, i \neq j$ ; 并且  $\mathbf{L}_{ij}$  为下三角矩阵。式 (9.6.19) 也可以等价写成

$$\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \\ \mathbf{A} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & & \\ \mathbf{L}_{21} & \mathbf{L}_{22} & \\ \mathbf{L}_{31} & \mathbf{L}_{32} & \mathbf{L}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \\ \mathbf{Q}_3^T \end{bmatrix} \quad (9.6.20)$$

根据斜投影定义式 (9.6.4), 并利用  $\mathbf{Q}_i^T \mathbf{Q}_i = \mathbf{I}$  和  $\mathbf{Q}_i^T \mathbf{Q}_j = \mathbf{O}, i \neq j$ , 易求得

$$\begin{aligned} \mathbf{E}_{\mathcal{Z}_B | \mathcal{Z}_C} &= [\mathbf{B}^T, \mathbf{C}^T] \begin{bmatrix} \mathbf{B}\mathbf{B}^T & \mathbf{B}\mathbf{C}^T \\ \mathbf{C}\mathbf{B}^T & \mathbf{C}\mathbf{C}^T \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \\ &= [\mathbf{Q}_1, \mathbf{Q}_2] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^T \left\{ \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^T \mathbf{Q}_1 & \mathbf{Q}_1^T \mathbf{Q}_2 \\ \mathbf{Q}_2^T \mathbf{Q}_1 & \mathbf{Q}_2^T \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^T \right\}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \\ &= [\mathbf{Q}_1, \mathbf{Q}_2] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \end{aligned} \quad (9.6.21)$$

类似地, 有

$$\mathbf{E}_{\mathcal{Z}_C | \mathcal{Z}_B} = [\mathbf{Q}_1, \mathbf{Q}_2] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{O} \\ \mathbf{C} \end{bmatrix} \quad (9.6.22)$$

注意到矩阵  $\mathbf{A}$  的 LQ 分解为

$$\mathbf{A} = [\mathbf{L}_{31}, \mathbf{L}_{32}, \mathbf{L}_{33}] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \\ \mathbf{Q}_3^T \end{bmatrix}$$

由式 (9.6.21) 可求得  $p \times k$  矩阵  $\mathbf{A}$  的行空间沿着与  $n \times k$  矩阵  $\mathbf{C}$  的行空间平行的方向, 到  $m \times k$  矩阵  $\mathbf{B}$  的行空间的斜投影等于

$$\begin{aligned} \mathbf{AE}_{\mathcal{Z}_B | \mathcal{Z}_C} &= [\mathbf{L}_{31}, \mathbf{L}_{32}, \mathbf{L}_{33}] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \\ \mathbf{Q}_3^T \end{bmatrix} [\mathbf{Q}_1, \mathbf{Q}_2] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \\ &= [\mathbf{L}_{31}, \mathbf{L}_{32}] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} \end{aligned}$$

若令

$$[\mathbf{L}_{31}, \mathbf{L}_{32}] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}^{-1} = [\mathbf{L}_B, \mathbf{L}_C] \quad (9.6.23)$$

则有<sup>[497]</sup>

$$\mathbf{AE}_{\mathcal{Z}_B | \mathcal{Z}_C} = [\mathbf{L}_B, \mathbf{L}_C] \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \end{bmatrix} = \mathbf{L}_B \mathbf{B} \quad (9.6.24)$$

类似地,  $p \times k$  矩阵  $\mathbf{A}$  的行空间沿着与  $m \times k$  矩阵  $\mathbf{B}$  的行空间平行的方向, 到  $n \times k$  矩阵  $\mathbf{C}$  的行空间的斜投影等于

$$\mathbf{AE}_{\mathcal{Z}_C | \mathcal{Z}_B} = [\mathbf{L}_B, \mathbf{L}_C] \begin{bmatrix} \mathbf{O} \\ \mathbf{C} \end{bmatrix} = \mathbf{L}_C \mathbf{C} \quad (9.6.25)$$

由式 (9.6.23) 得

$$[\mathbf{L}_{31}, \mathbf{L}_{32}] = [\mathbf{L}_B, \mathbf{L}_C] \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}$$

即有

$$\mathbf{L}_C = \mathbf{L}_{32} \mathbf{L}_{22}^{-1} \quad (9.6.26)$$

$$\mathbf{L}_B = (\mathbf{L}_{31} - \mathbf{L}_C \mathbf{L}_{21}) \mathbf{L}_{11}^{-1} = (\mathbf{L}_{31} - \mathbf{L}_{32} \mathbf{L}_{22}^{-1} \mathbf{L}_{21}) \mathbf{L}_{11}^{-1} \quad (9.6.27)$$

将以上两式分别代入式 (9.6.24) 和式 (9.6.25), 则有

$$\begin{aligned} \mathbf{A} \mathbf{E}_{Z_B|Z_C} &= (\mathbf{L}_{31} - \mathbf{L}_{32} \mathbf{L}_{22}^{-1} \mathbf{L}_{21}) \mathbf{L}_{11}^{-1} \mathbf{L}_{11} \mathbf{Q}_1^T \\ &= (\mathbf{L}_{31} - \mathbf{L}_{32} \mathbf{L}_{22}^{-1} \mathbf{L}_{21}) \mathbf{Q}_1^T \end{aligned} \quad (9.6.28)$$

$$\begin{aligned} \mathbf{A} \mathbf{E}_{Z_C|Z_B} &= \mathbf{L}_{32} \mathbf{L}_{22}^{-1} [\mathbf{L}_{21}, \mathbf{L}_{22}] \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \end{bmatrix} \\ &= \mathbf{L}_{32} \mathbf{L}_{22}^{-1} \mathbf{L}_{21} \mathbf{Q}_1^T + \mathbf{L}_{32} \mathbf{Q}_2^T \end{aligned} \quad (9.6.29)$$

### 9.6.3 斜投影算子的应用

斜投影算子已经陆续应用于广义图像恢复 [531]、求解大型非对称方程组 [433]、快速系统辨识 [436]、多变元分析 [470]、偏相关 (PARCOR) 估计 [273]、脉冲噪声对消 [516]、误码校正编码 [335]、猝发误码校正解码 [289]、模型简化 [249]、系统建模 [35]、无线信道的估计 [497]、信道与发射字符的联合估计 [533]。

在系统辨识、参数估计、信号检测等实际情况中, 除了感兴趣的信号 (简称期望信号) 外, 往往存在其他干扰信号或加性有色噪声。此外, 测量误差总是不可避免的, 它们通常表现为高斯白噪声。不妨令  $\boldsymbol{\theta}$  是期望信号待估计的参数向量, 它通过一线性系统  $\mathbf{H}$  后, 产生期望信号  $\mathbf{x} = \mathbf{H}\boldsymbol{\theta}$ 。假定其他干扰信号向量与 (或) 加性有色噪声向量  $\mathbf{i}$  由另外一个合成的线性系统  $\mathbf{S}$  所产生, 即  $\mathbf{i} = \mathbf{S}\boldsymbol{\phi}$ 。若观测数据向量为  $\mathbf{y}$ , 加性白色测量误差向量为  $\mathbf{e}$ , 则有

$$\mathbf{y} = \mathbf{H}\boldsymbol{\theta} + \mathbf{S}\boldsymbol{\phi} + \mathbf{e} \quad (9.6.30)$$

图 9.6.1 画出了这一观测模型的方框图。通常, 产生期望信号的线性系统  $\mathbf{H}$  的各个列向量不仅线性无关, 而且与产生干扰或者有色噪声的线性系统  $\mathbf{S}$  的各个列向量也线性无关。因此, 由线性系统  $\mathbf{H}$  的值域 Range( $\mathbf{H}$ ) 与线性系统  $\mathbf{S}$  的值域 Range( $\mathbf{S}$ ) 是无交连的, 但它们一般是不正交的。

由一线性系统产生的任何非期望信号常统称为结构化噪声 (structured noise)。假定结构化噪声与加性高斯白噪声  $\mathbf{e}$  正交, 则

$$\langle \mathbf{S}\boldsymbol{\phi}, \mathbf{e} \rangle = 0 \implies \boldsymbol{\phi}^H \mathbf{S}^H \mathbf{e} = 0, \forall \boldsymbol{\phi} \neq \mathbf{0} \implies \mathbf{S}^H \mathbf{e} = \mathbf{0} \quad (9.6.31)$$

类似地, 设期望信号也与加性高斯白噪声正交, 又有

$$\mathbf{H}^H \mathbf{e} = \mathbf{0} \quad (9.6.32)$$

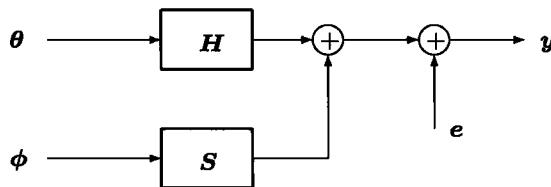


图 9.6.1 观测模型

给定矩阵  $H$  和  $S$ , 系统建模的目的是估计与期望信号有关的系统参数向量  $\theta$ 。为此, 用正交投影矩阵  $P_S^\perp$  左乘式 (9.6.30), 即得

$$P_S^\perp y = P_S^\perp H\theta + e \quad (9.6.33)$$

这里使用了  $P_S^\perp S = O$  和  $P_S^\perp e = [I - S(S^H S)^{-1} S^H]e = e$ 。

用矩阵  $H^H$  左乘式 (9.6.33) 两边, 并利用  $H^H e = 0$ , 易知

$$\theta = (H^H P_S^\perp H)^{-1} H^H P_S^\perp y \quad (9.6.34)$$

于是, 期望信号的估计为

$$\hat{x} = H\theta = H(H^H P_S^\perp H)^{-1} H^H P_S^\perp y = E_{H|S} y \quad (9.6.35)$$

即期望信号的估计是观测数据向量  $y$  沿着与矩阵  $S$  的列空间平行的方向, 到  $H$  的列空间上的斜投影。

### 本章小结

向量或矩阵到子空间的投影分为正交投影和斜投影两大类。正交投影是斜投影的特例。描述正交投影的矩阵分为投影矩阵和正交投影矩阵。本章从数学和信号处理的不同观点出发, 对投影矩阵进行了讨论与分析, 并得到了相同的结果。接着, 又介绍了投影矩阵与正交投影矩阵的递推计算及其在自适应滤波器设计中的应用。

斜投影刻画了另一类重要的科学与技术问题: 沿着一个子空间到另一个子空间的投影。本章围绕满列秩矩阵和满行秩矩阵的斜投影算子, 重点介绍了它们的性质、计算方法以及几种典型应用。

### 习题

**9.1** 证明唯一的非奇异幂等矩阵为单位矩阵。

**9.2 证明幂等矩阵的下列性质：**

- (1) 幂等矩阵的特征值只取 1 和 0 两个数值。
- (2) 所有的幂等矩阵(单位矩阵除外)  $A$  都是奇异矩阵。
- (3) 所有幂等矩阵的秩与迹相等, 即  $\text{rank}(A) = \text{tr}(A)$ 。
- (4) 若  $A$  为幂等矩阵, 则  $A^H$  也为幂等矩阵, 即有  $A^H A^H = A^H$ 。

**9.3 若  $A$  为幂等矩阵, 证明**

- (1) 矩阵  $A^k$  具有与  $A$  相同的特征值。
- (2)  $A^k$  与  $A$  具有相同的秩。

**9.4 假定  $A$  和  $B$  为对称矩阵, 并且  $B$  正定。若  $AB$  的所有特征值为 1 或者 0, 证明  $AB$  是幂等矩阵。**

**9.5 若  $A, B$  为  $n \times n$  幂等矩阵, 并且  $AB = BA$ , 证明  $AB$  也是幂等矩阵。**

**9.6 令  $X$  表示观测数据矩阵, 现在用它估计向量  $y$ 。已知两个观测数据向量  $x_1 = [2, 1, 2, 3]^T, x_2 = [1, 2, 1, 1]^T$ 。若使用它们估计  $y = [1, 2, 3, 2]^T$ , 求估计的误差平方和。(提示: 令最优滤波器为  $w_{\text{opt}}$ , 则有观测方程  $Xw_{\text{opt}} = y$ 。)**

**9.7 假定  $V_1, V_2$  分别由复向量  $C^n$  的子空间  $W$  的两组标准正交基组成, 证明**

$$V_1 V_1^H x = V_2 V_2^H x$$

对所有向量  $x$  成立。

**9.8 证明: 若投影算子  $P_1$  和  $P_2$  是可交换的, 即  $P_1 P_2 = P_2 P_1$ , 则它们的乘积  $P = P_1 P_2$  是一投影算子; 并求  $P$  的值域 Range( $P$ ) 和零空间 Null( $P$ )。**

**9.9 已知矩阵**

$$A = \begin{bmatrix} 6 & 2 \\ -7 & 6 \end{bmatrix}$$

和

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

分别求它们的 Moore-Penrose 逆矩阵  $A^\dagger$ , 并解释为什么  $AA^\dagger$  和  $A^\dagger A$  分别是到矩阵  $A$  的列空间和行空间的正交投影。

**9.10 假定两个基向量**

$$\begin{aligned} u_1 &= [-1, 2, -4, 3, 1]^T \\ u_2 &= [5, 6, 2, -2, -1]^T \end{aligned}$$

生成向量空间 Range( $U$ ) = Span{ $u_1, u_2$ }。试问向量

$$v = [-31, -18, -34, 28, 11]^T$$

是否在向量空间 Range( $U$ ) 内, 并加以证明。

**9.11** 证明下列关系为真

$$\mathbf{X}_{1,k}(n) = \begin{bmatrix} \mathbf{0}_k^T \\ \mathbf{X}_{0,k-1}(n-1) \end{bmatrix}$$

和

$$\mathbf{P}_{1,k}(n) = \begin{bmatrix} 0 & \mathbf{0}_{k-1}^T \\ \mathbf{0}_{k-1} & \mathbf{P}_{0,k-1}(n-1) \end{bmatrix}$$

式中,  $\mathbf{0}_k$  为  $k \times 1$  维零向量。

**9.12** 用逆矩阵

$$\langle \mathbf{X}_{1,p}(n-1), \mathbf{X}_{1,p}(n-1) \rangle^{-1}$$

表示逆矩阵

$$\langle \mathbf{X}_{1,p}(n), \mathbf{X}_{1,p}(n) \rangle^{-1}$$

**9.13** 已知

$$\gamma_m(n-1) = \langle \pi(n), \mathbf{P}_{1,m}^\perp(n) \pi(n) \rangle$$

其中  $\pi(n) = [0, \dots, 0, 1]^T$  的  $n$  维向量。

证明

$$\gamma_m(n) = \langle \pi(n), \mathbf{P}_{0,m-1}^\perp(n) \pi(n) \rangle$$

**9.14** 给定一时间信号  $\mathbf{v}(n) = [v(1), v(2), v(3), \dots, v(n)]^T = [4, 2, 4, \dots]^T$ 。计算:

- (1) 数据向量  $\mathbf{v}(2)$  和  $\mathbf{v}(3)$ 。
- (2) 向量  $z^{-1}\mathbf{v}(2)$  和  $z^{-2}\mathbf{v}(2)$ 。
- (3) 向量  $z^{-1}\mathbf{v}(3)$  和  $z^{-2}\mathbf{v}(3)$ 。
- (4) 若令  $\mathbf{u}(n) = z^{-1}\mathbf{v}(n)$ , 计算投影矩阵  $\mathbf{P}_u(2)$  和  $\mathbf{P}_u(3)$ 。
- (5) 利用  $\mathbf{u}(n)$  求  $\mathbf{v}(n)$  的最小二乘预测。这一预测称为  $\mathbf{v}(n)$  的一步前向预测。
- (6) 计算前向预测误差向量  $\mathbf{e}_1^f(2)$  和  $\mathbf{e}_1^f(3)$ 。

**9.15** 已知前向和后向预测残差分别为

$$\epsilon_m^f(n) = \langle \mathbf{x}(n), \mathbf{P}_{1,m}^\perp(n) \mathbf{x}(n) \rangle$$

$$\epsilon_m^b(n) = \langle z^{-m} \mathbf{x}(n), \mathbf{P}_{0,m-1}^\perp(n) z^{-m} \mathbf{x}(n) \rangle$$

和偏相关系数  $\Delta_{m+1}(n) = \langle \mathbf{e}_m^f(n), z^{-1} \mathbf{e}_m^b(n) \rangle$ 。证明

$$\epsilon_{m+1}^f(n) = \epsilon_m^f(n) - \frac{\Delta_{m+1}^2(n)}{\epsilon_m^b(n-1)}$$

$$\epsilon_{m+1}^b(n) = \epsilon_m^b(n-1) - \frac{\Delta_{m+1}^2(n)}{\epsilon_m^f(n)}$$

**9.16** 令  $\mathbf{U}$  是  $n \times N$  实矩阵, 并且满列秩, 则

$$\mathbf{K}_U = \langle \mathbf{U}, \mathbf{U} \rangle^{-1} \mathbf{U}^T$$

称为列空间  $U = \text{Span}(\mathbf{U})$  的横向滤波器算子。考虑新的列空间  $U\mathbf{u} = \text{Span}\{\mathbf{U}, \mathbf{u}\}$ , 试证明横向滤波器算子的下列递推公式

$$\mathbf{K}_{U\mathbf{u}} = \begin{bmatrix} \mathbf{K}_U \\ \mathbf{0}_N^T \end{bmatrix} + \left( \begin{bmatrix} \mathbf{0}_N \\ 1 \end{bmatrix} - \begin{bmatrix} \mathbf{K}_U \mathbf{u} \\ 0 \end{bmatrix} \right) \langle \mathbf{P}_U^\perp \mathbf{u}, \mathbf{P}_U^\perp \mathbf{u} \rangle^{-1} \mathbf{u}^T \mathbf{P}_U^\perp$$

### 9.17 记

$$\begin{aligned} \{\mathbf{U}, \mathbf{u}\} &= \{\mathbf{x}(n), z^{-1}\mathbf{x}(n), \dots, z^{-N+1}\mathbf{x}(n), \pi(n)\} \\ &= \{\mathbf{X}_{0,N-1}(n), \pi(n)\} \end{aligned}$$

证明

$$\mathbf{K}_{0,N-1,\pi}(n) = \begin{bmatrix} \mathbf{K}_{0,N-1}(n-1) & \mathbf{0}_N \\ \mathbf{y}^T(n-1) & 1 \end{bmatrix}$$

式中, “ $0, N-1, \pi$ ” 表示在  $\mathbf{X}_{0,N-1}(n)$  的最后一列之后追加  $\pi(n)$ ,  $\mathbf{y}(n-1)$  是一任意向量。 (提示: 使用  $\mathbf{P}_{0,N-1,\pi}(n)$  的递推公式。)

### 9.18 令

$$\begin{aligned} \mathbf{E}_{H|S} &= \mathbf{H}(\mathbf{H}^H \mathbf{P}_S^\perp \mathbf{H})^{-1} \mathbf{H}^H \mathbf{P}_S^\perp \\ \mathbf{E}_{S|H} &= \mathbf{S}(\mathbf{S}^H \mathbf{P}_H^\perp \mathbf{S})^{-1} \mathbf{S}^H \mathbf{P}_H^\perp \end{aligned}$$

试证明:

(1)  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  均为幂等算子, 即有

$$\mathbf{E}_{H|S}^2 = \mathbf{E}_{H|S} \quad \text{和} \quad \mathbf{E}_{S|H}^2 = \mathbf{E}_{S|H}$$

(2)  $[\mathbf{H}, \mathbf{S}]$  到  $\text{Range}([\mathbf{H}, \mathbf{S}])$  的斜投影

$$\begin{aligned} \mathbf{E}_{H|S}[\mathbf{H}, \mathbf{S}] &= [\mathbf{H}, \mathbf{O}] \\ \mathbf{E}_{S|H}[\mathbf{H}, \mathbf{S}] &= [\mathbf{O}, \mathbf{S}] \end{aligned}$$

或等价为

$$\begin{aligned} \mathbf{E}_{H|S} \mathbf{H} &= \mathbf{H} \quad \text{和} \quad \mathbf{E}_{H|S} \mathbf{S} = \mathbf{O} \\ \mathbf{E}_{S|H} \mathbf{H} &= \mathbf{O} \quad \text{和} \quad \mathbf{E}_{S|H} \mathbf{S} = \mathbf{S} \end{aligned}$$

### 9.19 证明广义逆矩阵与斜投影的乘积

$$\mathbf{H}^\dagger \mathbf{E}_{H|S} = (\mathbf{P}_S^\perp \mathbf{H})^\dagger, \quad \mathbf{S}^\dagger \mathbf{E}_{S|H} = (\mathbf{P}_H^\perp \mathbf{S})^\dagger \quad (9.6.36)$$

式中,  $\mathbf{B}^\dagger = (\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H$  表示矩阵  $\mathbf{B}$  的广义逆矩阵。

**9.20 证明斜投影算子  $\mathbf{E}_{H|S}$  和  $\mathbf{E}_{S|H}$  的交叉项为零, 即有**

$$\mathbf{E}_{H|S} \mathbf{E}_{S|H} = \mathbf{O}, \quad \mathbf{E}_{S|H} \mathbf{E}_{H|S} = \mathbf{O}$$

### 9.21 证明斜投影后，再正交投影，不会改变原斜投影

$$\mathbf{E}_{H|S} = \mathbf{P}_H \mathbf{E}_{H|S}, \quad \mathbf{E}_{S|H} = \mathbf{P}_S \mathbf{E}_{S|H}$$

9.22 假设在同步 CDMA 中有  $K$  个用户同时在通信，CDMA 的扩频增益为  $N$ ，在接收机处，第  $k$  个用户的接收功率为  $A_k$ ，第  $k$  个用户的扩频波形为  $s_k$ ，且  $\|s_k\| = 1$ ，则接收信号的等效基带信号可以表示为

$$\mathbf{r} = \mathbf{S}\mathbf{Ab} + \mathbf{n}$$

其中  $\mathbf{b} = [b_1, b_2, \dots, b_K]^T$  为  $K$  个用户传输的信息比特， $\mathbf{A} = \text{diag}(A_1, A_2, \dots, A_K)$ ， $\mathbf{S} = [s_1, s_2, \dots, s_K]$ ， $\mathbf{n}$  是高斯噪声向量。则解相关输出为

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{S}^\dagger \mathbf{r} = (\mathbf{S}^H \mathbf{S})^{-1} \mathbf{S}^H (\mathbf{S}\mathbf{Ab} + \mathbf{n}) \\ &= \mathbf{Ab} + (\mathbf{S}^H \mathbf{S})^{-1} \mathbf{S}^H \mathbf{n} = \mathbf{Ab} + \nu\end{aligned}$$

证明解相关检测器等价于斜投影。

# 第 10 章 张量分析

在多门学科中，越来越多的问题需要使用两个下标以上的数据来描述。两个下标以上的数据的多路排列称为多路数据，其表示形式为张量。

采用向量和矩阵的数据分析属线性数据分析，基于张量的数据分析称为张量分析 (tensor analysis)，属多重线性数据分析 (multilinear data analysis) 范畴。

线性数据分析是一种单因子 (single-factor) 分析方法，而多重线性数据分析则是一种多因子 (multi-factor) 分析方法。正如张量是矩阵的推广一样，多重线性数据分析是线性数据分析的自然扩展。

## 10.1 张量及其表示

数据沿一相同方向的排列称为一路阵列。标量是零路阵列的表示，行向量和列向量分别是数据沿水平和垂直方向排列的一路阵列，矩阵是数据沿水平和垂直两个方向排列的二路阵列。张量是数据的多路阵列表示，一个张量就是一个多路阵列或多维阵列，它是矩阵的一种扩展。数学中的张量专指多路阵列，因此不应与物理和工程中的张量 (例如应力张量) 混淆，后者在数学中常称为张量场 (tensor fields) [455]。

张量用花体符号表示，如  $T, A, \mathcal{X}$  等。 $n$  路阵列表示的张量称为  $n$  阶张量，是定义在  $n$  个向量空间的笛卡儿积上的多重线性函数，记为  $T \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_n}$ ，其中  $\mathbb{K}$  代表实数域  $\mathbb{R}$  或者复数域  $\mathbb{C}$ 。因此，标量为零阶张量 (zero-order tensor)，向量属一阶张量 (first-order tensor)，矩阵为二阶张量 (second-order tensor)；张量则是标量、向量和矩阵的高阶推广，是跨越  $n$  个向量空间的多线性映射 (multilinear mappings)，即有

$$T: \mathbb{K}^{I_1} \times \mathbb{K}^{I_2} \times \dots \times \mathbb{K}^{I_n} \rightarrow \mathbb{K}^{I_1 \times I_2 \times \dots \times I_n} \quad (10.1.1)$$

随着现代应用的发展，多路模型已涉及化学、医学与神经科学、文本挖掘、聚类、互联网流量、脑电图、计算机视觉、通信记录和大规模社会网络等。以下是多路模型在几门学科中的典型应用 [9]。

(1) 化学 在化学、医药和食品科学中常用的荧光激发发射数据 (fluorescence excitation-emission) 典型地含有不同浓度的几种化学成分。荧光光谱能够生成具有模式“样本  $\times$  激发  $\times$  发射”的三路数据集。对这种数据类型进行分析的主要目的是为了确定每一个样本中含有哪些化学成分以及这些成分的相对浓度。

(2) 医学与神经科学 多通道脑电图 (electroencephalogram, EEG) 数据通常表示为一个  $m \times n$  矩阵，该矩阵的元素是采自  $m$  个时间样本和  $n$  个电极的信号值。然而，为了

发现隐藏的脑动力结构，就需要考虑脑电信号的频率分量（例如  $p$  个特定频率的瞬时信号功率）。在这种情况下，脑电图数据即可排列成  $m \times n \times p$  三路数据集<sup>[348, 9]</sup>。三路数据阵列分别对应于通道即空间（不同位置的电极）、时间（数据样本）和频率分量。如果再增加主题和条件这两个模态（modelities），则得到的是“通道  $\times$  时间  $\times$  频率  $\times$  主题  $\times$  条件”的 5 路阵列或张量。多路模型早在 2001 年就已经用于研究新的药物对大脑活动的影响<sup>[159]</sup>。在这一研究中，EEG 数据和不同剂量的药物对几种病人在某些情况下的实验数据被排列成具有下列模式的六路阵列：EEG、病人、剂量、条件等。结果表明，通过多路模型而不是二路模型（如 PCA），从复杂的药物数据集中成功地提取到了重要的信息。

(3) 社会网络分析/Web 挖掘 多路数据分析也经常用于提取社会网络中的关系。社会网络分析的目的是研究和发现社会网络中的隐藏结构，例如，提取人与人之间或组织内的沟通模式。在文献 [6] 中，聊天室通信数据被排列成具有模式“用户  $\times$  关键词  $\times$  时间样本”的三路阵列，并且多路模型在捕捉潜在的用户群结构方面的性能与二路模型进行了比较。不仅聊天室，而且电子邮件（email）通信数据也可表示为“发送者  $\times$  接收者  $\times$  时间”的三路模型<sup>[27]</sup>。在网络链接分析的范围内，结合超链接和锚文本信息，将网页图形数据重新排列成具有模式“网页  $\times$  网页  $\times$  锚文本”的稀疏三路张量<sup>[275, 277]</sup>。在网页个性化中，点击数据被排列成具有模式“用户  $\times$  查询词  $\times$  网页”的三路阵列<sup>[468]</sup>。

(4) 计算机视觉 张量的逼近已被证明在计算机视觉中有着重要的应用，例如用于图像压缩和人脸识别。在计算机视觉和图形学中，数据本质上通常为多路。例如，一幅彩色图像像是三路阵列，其  $x$  和  $y$  坐标为两个模式，色彩为第三个模式。过去的图像编码技术将图像视为向量或矩阵。现已证明，当图像表示成张量时，迭代得到的张量的秩 1 逼近即可用于压缩这些图像<sup>[507]</sup>。此外，图像的这一张量构造既能够保留图像的二维特征，又可避免图像信息的损失<sup>[222]</sup>。因此，在进行图像压缩时，张量表示比矩阵表示更有效<sup>[447]</sup>。

(5) 癫痫张量 (epilepsy tensors) 通过识别癫痫发作与假象的空域、频谱与时域特征，确定癫痫发作的焦点，或者排除癫痫假象。因此，需要使用时间、频率和空间（电极）三路阵列对癫痫信号建模<sup>[8]</sup>。

(6) 高光谱图像 (hyperspectral image) 是一组二维图像，因此高光谱图像数据可以用三路阵列表示：两路空间数据和一路频谱数据<sup>[395, 96, 311]</sup>。其中，两路空间数据确定像素的位置，另外一路频谱数据与光谱波段有关。

(7) 人脸识别的多线性图像分析<sup>[501, 502]</sup> 其中，张量脸（tensor faces）建模为 5 阶张量  $\mathcal{D} \in \mathbb{R}^{28 \times 5 \times 3 \times 3 \times 7943}$ ，表示 28 个人在 5 种拍摄角度（viewpoint）、3 种光照（illumination）、3 种表情（expression）情况下拍摄的脸部图像，每幅图像有 7943 个像素（pixel）。

图 10.1.1 画出了三路阵列和张量脸的数据表示例子。

线性代数即有限维向量空间的矩阵的代数，定义了一个有限维向量空间的线性算子。由于矩阵属二阶张量，所以线性代数也可视为二阶张量的代数。多重线性代数即高阶张量代数，定义了一组有限维向量空间的多重线性算子。作为传统的线性分析的推广，张量分析提供了可处理一系列计算机视觉问题的一种统一的理论框架。

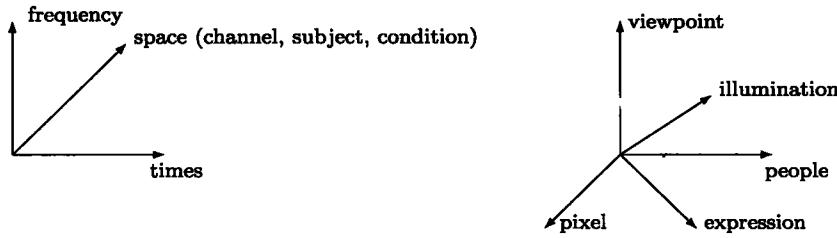


图 10.1.1 三路阵列 (左) 与张量脸 (右) 的数据表示

矩阵  $A \in \mathbb{K}^{m \times n}$  用其元素和矩阵符号  $[ \cdot ]$  表示为  $A = [a_{ij}]_{i,j=1}^{m,n}$ 。类似地,  $n$  阶张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_n}$  用双重矩阵符号  $[ \cdot ]$  表示为  $\mathcal{A} = [a_{i_1 \cdots i_n}]_{i_1, \dots, i_n=1}^{I_1, \dots, I_n}$ , 其中  $a_{i_1 i_2 \cdots i_n}$  是张量的第  $(i_1, \dots, i_n)$  元素。 $n$  阶张量有时也称  $n$  维超矩阵 ( $n$ -dimensional hypermatrix)<sup>[124]</sup>。所有  $I_1 \times I_2 \times \cdots \times I_n$  维张量的集合常记为  $\mathcal{T}(I_1, I_2, \dots, I_n)$ 。

最常用的张量为三阶张量 (third-order tensor)  $\mathcal{A} = [a_{ijk}]_{i,j,k=1}^{I,J,K} \in \mathbb{K}^{I \times J \times K}$ 。三阶张量有时也称三维矩阵<sup>[486]</sup>。维数相同的正方三阶张量  $\mathcal{X} \in \mathbb{K}^{I \times I \times I}$  称为立方体 (cubical)。特别地, 一个立方体是超对称张量 (supersymmetric tensor)<sup>[123, 278]</sup>, 若其元素具有下列对称性

$$x_{ijk} = x_{ikj} = x_{jik} = x_{jki} = x_{kij} = x_{kji}, \quad \forall i, j, k = 1, \dots, I$$

图 10.1.2 (a) 画出了三阶张量  $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ 。

正方三阶张量从  $i = j = k = 1$  到  $i = j = k = N$  相连接的直线称为超对角线 (superdiagonal)。超对角线上的元素全部为 1, 而其他所有元素皆等于零的三阶张量称为单位三阶张量, 即其元素为

$$I_{ijk} = \begin{cases} 1, & i = j = k \in \{1, \dots, N\} \\ 0, & \text{其他} \end{cases} \quad (10.1.2)$$

单位三阶张量用符号  $\mathcal{I} \in \mathbb{R}^{N \times N \times N}$  表示, 如图 10.1.2 (b) 所示。

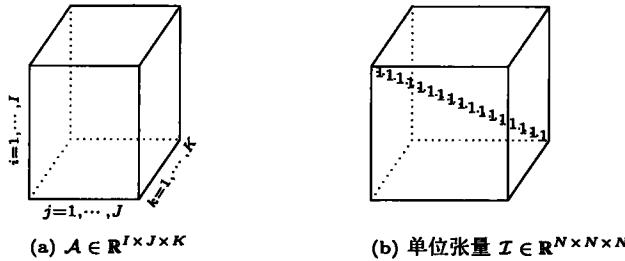


图 10.1.2 三阶张量

在张量分析中, 将三阶张量视为向量或者矩阵的集合, 往往会带来很大的方便。

向量  $\mathbf{a}$  的第  $i$  个元素记为  $a_i$ , 矩阵  $\mathbf{A} = [a_{ij}] \in \mathbb{K}^{I \times J}$  共有  $I$  个行向量  $\mathbf{a}_{i:}, i = 1, \dots, I$  和  $J$  个列向量  $\mathbf{a}_{:j}, j = 1, \dots, J$ 。第  $i$  行向量记为  $\mathbf{a}_{i:} = [a_{i1}, \dots, a_{iJ}]$ , 第  $j$  列记为  $\mathbf{a}_{:j} = [a_{1j}, \dots, a_{IJ}]^T$ 。行向量和列向量的概念对高阶张量不再适用。

三阶张量的三路阵列不以行向量、列向量等相称, 而改称张量纤维(tensor fiber)。纤维是只保留一个下标可变, 固定其他所有下标不变而得到的一路阵列。它们分别是三阶张量的水平纤维(horizontal fiber)、竖直纤维(vertical fiber)和纵深纤维(“depth” fiber)。三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  的竖直纤维又叫列纤维(column fiber), 用符号  $\mathbf{a}_{:jk}$  表示; 水平纤维也称行纤维(row fiber), 符号为  $\mathbf{a}_{i:k}$ ; 纵深纤维或叫管纤维(tube fibers), 用符号  $\mathbf{a}_{ij:}$  记之。图 10.1.3 (a)~(c) 的纤维图分别画出了三阶张量的列纤维、行纤维和管纤维。

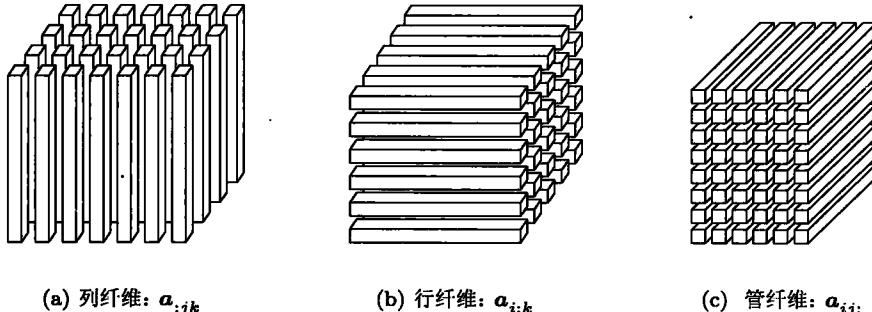


图 10.1.3 三阶张量的纤维图

显然, 三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  分别有  $J \cdot K = JK$  个列纤维、 $KI$  个行纤维和  $IJ$  个管纤维。 $N$  阶张量有  $N$  种不同的纤维, 称为模式- $n$  纤维或者模式- $n$  向量。

**定义 10.1.1** <sup>[302]</sup>  $N$  阶张量  $\mathcal{A} = [a_{i_1 i_2 \dots i_N}] \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_N}$  的模式- $n$  向量是一个以  $i_n$  为元素下标变量, 而其他下标  $\{i_1, \dots, i_N\} \setminus i_n$  全部被固定不变的  $I_n$  维向量, 用符号记作  $\mathbf{A}_{i_1 \dots i_{n-1} : i_{n+1} \dots i_N}$ 。

注意张量的阶数与维数的区别: 张量  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  中的  $N$  称为张量的阶数, 而  $I_n$  称为第  $n$  路阵列的维数。

矩阵中, 列向量称为模式-1 向量, 行向量称为模式-2 向量。在三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  中, 列纤维  $\mathbf{a}_{:jk}$ 、行纤维  $\mathbf{a}_{i:k}$  和管纤维  $\mathbf{a}_{ij:}$  分别是张量的模式-1、模式-2 和模式-3 向量。模式-1 向量共有  $J \cdot K = JK$  个, 用符号  $\mathbf{a}_{:jk}$  记之, 每一个模式-1 向量含有  $I$  个元素, 即  $\mathbf{a}_{:jk} = (a_{1jk}, \dots, a_{Ijk})$ 。类似地, 模式-2 向量共有  $KI$  个, 记为  $\mathbf{a}_{i:k}$ , 每一个模式-2 向量由  $J$  个元素组成, 即  $\mathbf{a}_{i:k} = (a_{i1k}, \dots, a_{iJk})$ ; 模式-3 向量有  $IJ$  个, 记为  $\mathbf{a}_{ij:} = (a_{ij1}, \dots, a_{ijk})$ 。模式- $n$  向量张成的子空间称为模式- $n$  空间。一般地,  $\mathbf{a}_{i_1 \dots i_N}, \mathbf{a}_{i_1 : i_3 \dots i_N}, \mathbf{a}_{i_1 \dots i_{n-1} : i_{n+1} \dots i_N}$  分别是  $N (> 3)$  阶张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_N}$  的模式-1、模式-2 和模式- $n$  向量。显然,  $N$  阶张量共有  $I_2 \dots I_N$  个模式-1 向量和  $I_1 \dots I_{n-1} I_{n+1} \dots I_N$  个模式- $n$  向量。

注意, 我们没有把模式- $n$  向量表示成列向量  $[\dots]^T$  或行向量  $[\dots]$  的形式, 而是刻意使用符号  $(\dots)$  表示, 乃是因为同一模式的向量有时为列向量, 有时为行向量, 取决于张量的切片矩阵的结构。

高阶张量也可以用矩阵的集合表示。这些矩阵形成了三阶张量的水平切片 (horizontal slice)、侧向切片 (lateral slice) 和正面切片 (frontal slice)，如图 10.1.4 (a)~(c) 所示。

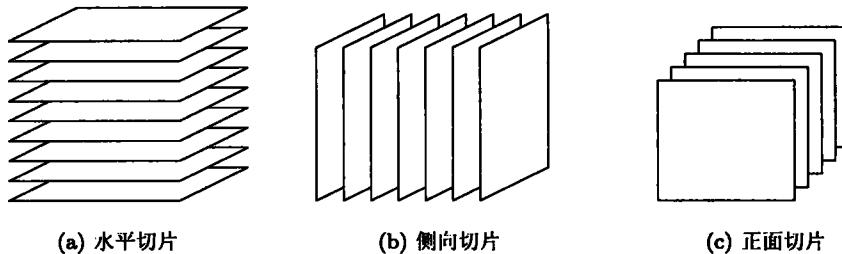


图 10.1.4 三阶张量的切片图

三阶张量的水平切片、侧向切片和正面切片分别使用矩阵符号  $A_{i::}$ ,  $A_{::j}$  和  $A_{::k}$  表示。图 10.1.5 示出了三种切片矩阵的标准表示。

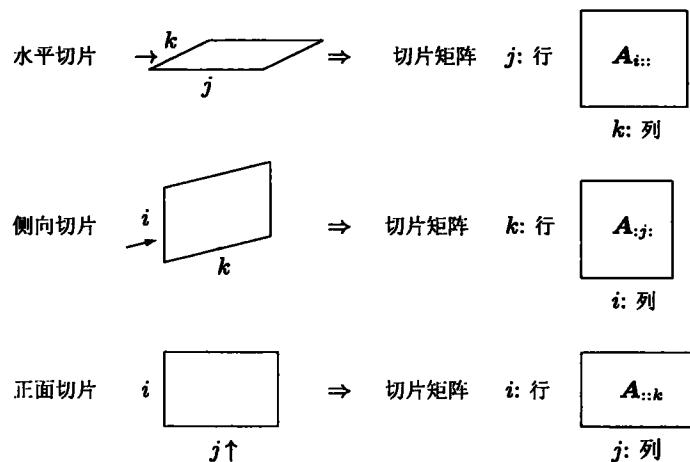


图 10.1.5 切片矩阵的标准表示

图中，箭头所指切口方向代表矩阵的列方向。事实上，三阶张量的标号  $i, j, k$  可按顺序组成标号集合 (index sets)  $(i, j), (j, k), (k, i)$ ，每组集合的第 1 个和第 2 个元素分别代表相应切片矩阵的行和列的标号。三阶张量的各个切片矩阵的行与列的关系如下：

正面切片矩阵  $A_{::k}$ :  $i$  为行的序号,  $j$  为列的序号;

水平切片矩阵  $A_{i::}$ :  $j$  为行的序号,  $k$  为列的序号;

侧向切片矩阵  $A_{::j}$ :  $k$  为行的序号,  $i$  为列的序号。

下面是三阶张量的三种切片矩阵的数学表示。

(1) 三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  有  $I$  个水平切片矩阵

$$\mathcal{A}_{i::} \stackrel{\text{def}}{=} \begin{bmatrix} a_{i11} & \cdots & a_{i1K} \\ \vdots & \ddots & \vdots \\ a_{iJ1} & \cdots & a_{iJK} \end{bmatrix} = [\mathbf{a}_{i:1}, \dots, \mathbf{a}_{i:K}] = \begin{bmatrix} \mathbf{a}_{i1:} \\ \vdots \\ \mathbf{a}_{iJ:} \end{bmatrix}, i = 1, \dots, I \quad (10.1.3)$$

(2) 三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  有  $J$  个侧向切片矩阵

$$\mathcal{A}_{:j} \stackrel{\text{def}}{=} \begin{bmatrix} a_{1j1} & \cdots & a_{IJj} \\ \vdots & \ddots & \vdots \\ a_{1jk} & \cdots & a_{Ijk} \end{bmatrix} = [\mathbf{a}_{1j:}, \dots, \mathbf{a}_{Ij:}] = \begin{bmatrix} \mathbf{a}_{:j1} \\ \vdots \\ \mathbf{a}_{:jK} \end{bmatrix}, j = 1, \dots, J \quad (10.1.4)$$

(3) 三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  有  $K$  个正面切片矩阵

$$\mathcal{A}_{::k} \stackrel{\text{def}}{=} \begin{bmatrix} a_{11k} & \cdots & a_{1Jk} \\ \vdots & \ddots & \vdots \\ a_{I1k} & \cdots & a_{IJk} \end{bmatrix} = [\mathbf{a}_{:1k}, \dots, \mathbf{a}_{:Jk}] = \begin{bmatrix} \mathbf{a}_{1:k} \\ \vdots \\ \mathbf{a}_{I:k} \end{bmatrix}, k = 1, \dots, K \quad (10.1.5)$$

从以上分析知, 同一种模式- $n$  向量在不同的切片矩阵中或作为列向量或作为行向量

模式-1 向量  $\mathbf{a}_{:jk}$   $\begin{cases} \mathbf{a}_{:1k}, \dots, \mathbf{a}_{:Jk} \text{ 表示 } \mathcal{A}_{::k} \text{ 的列向量} \\ \mathbf{a}_{:j1}, \dots, \mathbf{a}_{:jK} \text{ 表示 } \mathcal{A}_{:j} \text{ 的行向量} \end{cases}$

模式-2 向量  $\mathbf{a}_{i:k}$   $\begin{cases} \mathbf{a}_{i:1}, \dots, \mathbf{a}_{i:K} \text{ 表示 } \mathcal{A}_{i::} \text{ 的列向量} \\ \mathbf{a}_{1:k}, \dots, \mathbf{a}_{I:k} \text{ 表示 } \mathcal{A}_{::k} \text{ 的行向量} \end{cases}$

模式-3 向量  $\mathbf{a}_{ij:}$   $\begin{cases} \mathbf{a}_{1j:}, \dots, \mathbf{a}_{Ij:} \text{ 表示 } \mathcal{A}_{:j} \text{ 的列向量} \\ \mathbf{a}_{i1:}, \dots, \mathbf{a}_{iJ:} \text{ 表示 } \mathcal{A}_{i::} \text{ 的行向量} \end{cases}$

**例 10.1.1** 张量  $\mathcal{A} \in \mathbb{R}^{3 \times 4 \times 2}$  的两个正面切片分别为 [276]

$$\mathcal{A}_{::1} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix}, \quad \mathcal{A}_{::2} = \begin{bmatrix} 13 & 16 & 19 & 22 \\ 14 & 17 & 20 & 23 \\ 15 & 18 & 21 & 24 \end{bmatrix} \in \mathbb{R}^{3 \times 4} \quad (10.1.6)$$

模式-1 向量 (即列纤维) 共有  $JK = 4 \times 2 = 8$  个, 分别为

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}, \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix}, \begin{bmatrix} 10 \\ 11 \\ 12 \end{bmatrix}, \begin{bmatrix} 13 \\ 14 \\ 15 \end{bmatrix}, \begin{bmatrix} 16 \\ 17 \\ 18 \end{bmatrix}, \begin{bmatrix} 19 \\ 20 \\ 21 \end{bmatrix}, \begin{bmatrix} 22 \\ 23 \\ 24 \end{bmatrix}$$

它们张成为张量  $\mathcal{A} \in \mathbb{R}^{3 \times 4 \times 2}$  的模式-1 空间。模式-2 向量共有  $KI = 2 \times 3 = 6$  个, 分别为

$$[1, 4, 7, 10], [2, 5, 8, 11], [3, 6, 9, 12], [13, 16, 19, 22], [14, 17, 20, 23], [15, 18, 21, 24]$$

模式-3 向量共有  $IJ = 3 \times 4 = 12$  个, 分别是

$$[1, 13], [4, 16], [7, 19], [10, 22]; [2, 14], [5, 17], [8, 20], [11, 23]; [3, 15], [6, 18], [9, 21], [12, 24]$$

另外, 张量的三个水平切片矩阵为

$$\mathcal{A}_{1::} = \begin{bmatrix} 1 & 13 \\ 4 & 16 \\ 7 & 19 \\ 10 & 22 \end{bmatrix}, \quad \mathcal{A}_{2::} = \begin{bmatrix} 2 & 14 \\ 5 & 17 \\ 8 & 20 \\ 11 & 23 \end{bmatrix}, \quad \mathcal{A}_{3::} = \begin{bmatrix} 3 & 15 \\ 6 & 18 \\ 9 & 21 \\ 12 & 24 \end{bmatrix} \in \mathbb{R}^{4 \times 2} \quad (10.1.7)$$

四个侧向切片矩阵为

$$\left. \begin{aligned} \mathbf{A}_{::1} &= \begin{bmatrix} 1 & 2 & 3 \\ 13 & 14 & 15 \end{bmatrix}, & \mathbf{A}_{::2} &= \begin{bmatrix} 4 & 5 & 6 \\ 16 & 17 & 18 \end{bmatrix} \\ \mathbf{A}_{::3} &= \begin{bmatrix} 7 & 8 & 9 \\ 19 & 20 & 21 \end{bmatrix}, & \mathbf{A}_{::4} &= \begin{bmatrix} 10 & 11 & 12 \\ 22 & 23 & 24 \end{bmatrix} \end{aligned} \right\} \in \mathbb{R}^{2 \times 3} \quad (10.1.8)$$

将三阶张量的切片概念加以推广，我们可以将四阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K \times L}$  分为  $L$  个  $I \times J \times K$  维三阶张量切片  $\mathcal{A}_1, \dots, \mathcal{A}_L$ 。依次类推，五阶张量  $\mathcal{A}^{I \times J \times K \times L \times M}$  可以分为  $M$  个四阶张量切片，而每个四阶张量切片又可进一步分为  $L$  个三阶张量切片。从这个意义上讲，三阶张量是张量分析的基础。

## 10.2 张量的矩阵化与向量化

三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  有  $I$  个水平切片矩阵  $\mathbf{A}_{i::} \in \mathbb{K}^{J \times K}(i = 1, \dots, I)$ 、 $J$  个侧向切片矩阵  $\mathbf{A}_{::j} \in \mathbb{K}^{K \times I}(j = 1, \dots, J)$  和  $K$  个正面切片矩阵  $\mathbf{A}_{::k} \in \mathbb{K}^{I \times J}(k = 1, \dots, K)$ 。然而，在张量的分析与计算中，却经常希望用一个矩阵代表一个三阶张量。此时，就需要有一种运算，能够将一个三阶张量（三路阵列）经过重新组织或者排列，变成一个矩阵（二路阵列）。将一个三路或  $N$  路阵列重新组织成一个矩阵形式的变换称为张量的矩阵化（matricization 或 matricizing）<sup>[267]</sup>。张量的矩阵化有时也称张量的展开（unfolding）<sup>[66]</sup> 或扁平化（flattening）。注意，这里的展开和扁平化只是指将一个立体或三维的阵列展开为平面或二维的阵列。

除了高阶张量的唯一矩阵表示外，一个高阶张量的唯一向量表示也是很多场合感兴趣的。高阶张量的向量化（vectorization）是一种将张量排列成唯一一个向量的变换。

### 10.2.1 张量的水平展开与向量化

将三阶张量的同一种切片矩阵按照水平方向依次排列，称为三阶张量的水平展开（horizontal unfolding）。

依照切片矩阵的不同，可以得到三种不同的水平展开方法。为了避免切片矩阵符号的混淆，这里规定正面切片  $\mathbf{A}_{::k}$  为  $I \times J$  矩阵、水平切片  $\mathbf{A}_{i::}$  为  $J \times K$  矩阵、纵向切片  $\mathbf{A}_{::j}$  为  $K \times I$  矩阵。因此，如果正面切片是按照  $J \times I$  矩阵的形式水平展开，则用转置矩阵  $\mathbf{A}_{::k}^T$  表示，依次类推。

#### 1. Kiers 水平展开方法

这是 Kiers 于 2000 年提出的张量矩阵化方法。在这一方法里，三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$

分别矩阵化为以下三种水平展开矩阵<sup>[267]</sup>

$$\left. \begin{array}{l} a_{i,(k-1)J+j}^{(I \times JK)} = a_{ijk} \iff \mathbf{A}^{(I \times JK)} = \mathbf{A}_{(1)} = [\mathbf{A}_{::1}, \dots, \mathbf{A}_{::K}] \\ a_{j,(i-1)K+k}^{(J \times KI)} = a_{ijk} \iff \mathbf{A}^{(J \times KI)} = \mathbf{A}_{(2)} = [\mathbf{A}_{1::}, \dots, \mathbf{A}_{I::}] \\ a_{k,(j-1)I+i}^{(K \times IJ)} = a_{ijk} \iff \mathbf{A}^{(K \times IJ)} = \mathbf{A}_{(3)} = [\mathbf{A}_{1::}, \dots, \mathbf{A}_{J::}] \end{array} \right\} \quad (10.2.1)$$

图 10.2.1 画出了三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  的三种水平展开。

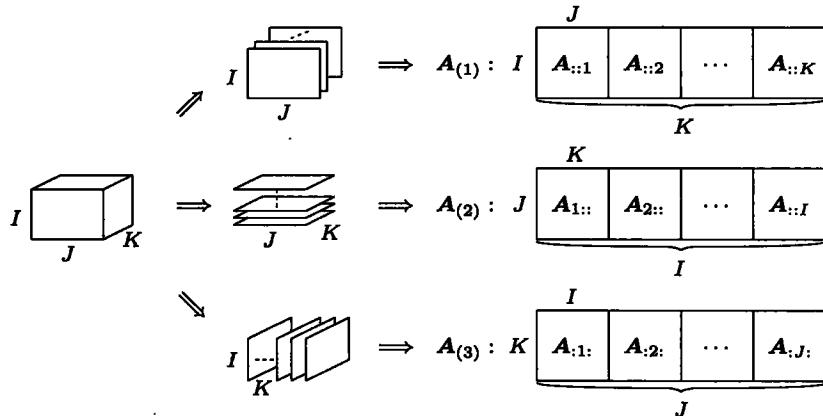


图 10.2.1 三阶张量的水平展开 (Kiers 方法)

推而广之,  $N$  阶张量的 Kiers 水平展开方法将张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_N}$  的元素  $a_{i_1 i_2 \dots i_N}$  映射为矩阵  $\mathbf{A}^{(I_n \times I_1 \dots I_{n-1} \dots I_{n+1} \dots I_N)}$  的第  $(i_n, j)$  个元素

$$[\mathbf{A}_{(n)}]_{i_n, j} = a_{i_n, j}^{(I_n \times I_1 \dots I_{n-1} \dots I_{n+1} \dots I_N)} = a_{i_1 i_2 \dots i_N} \quad (10.2.2)$$

其中  $i_n = 1, \dots, I_n$ , 且

$$j = \sum_{p=1}^{N-2} \left( (i_{N+n-p} - 1) \prod_{q=n+1}^{N+n-p-1} I_q \right) + i_{n+1}, \quad n = 1, \dots, N \quad (10.2.3)$$

以及  $I_{N+m} = I_m, i_{N+m} = i_m$  ( $m > 0$ )。

## 2. LMV 水平展开方法

Lathauwer, Moor 和 Vanderwalle 于 2000 年提出了三阶张量的以下水平展开 (简称 LMV 方法)<sup>[298]</sup>

$$\left. \begin{array}{l} a_{i,(j-1)K+k}^{(I \times JK)} = a_{ijk} \iff \mathbf{A}^{(I \times JK)} = \mathbf{A}_{(1)} = [\mathbf{A}_{1::}^T, \dots, \mathbf{A}_{J::}^T] \\ a_{j,(i-1)I+i}^{(J \times KI)} = a_{ijk} \iff \mathbf{A}^{(J \times KI)} = \mathbf{A}_{(2)} = [\mathbf{A}_{::1}^T, \dots, \mathbf{A}_{::K}^T] \\ a_{k,(j-1)I+j}^{(K \times IJ)} = a_{ijk} \iff \mathbf{A}^{(K \times IJ)} = \mathbf{A}_{(3)} = [\mathbf{A}_{1::}^T, \dots, \mathbf{A}_{I::}^T] \end{array} \right\} \quad (10.2.4)$$

图 10.2.2 示出了三阶张量的水平展开示意图。

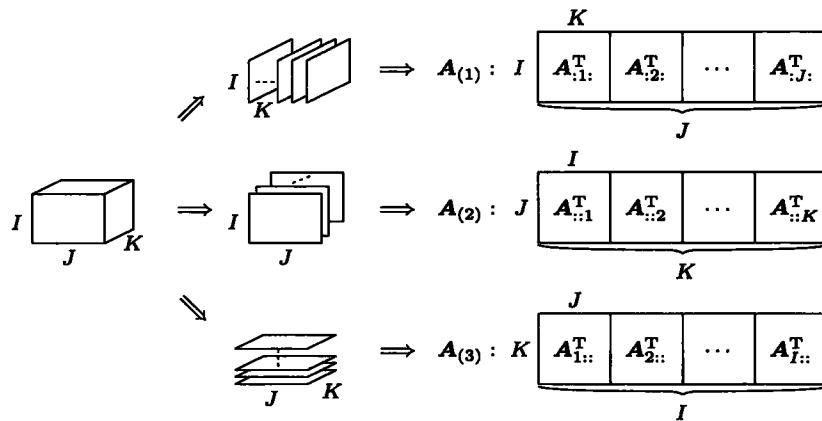


图 10.2.2 三阶张量的水平展开 (LMV 方法)

$N$  阶张量  $A$  的 LMV 水平展开将张量  $A$  的元素  $a_{i_1, i_2, \dots, i_N}$  映射为模式- $n$  矩阵  $A_{(n)}$  的元素  $a_{i_n, j}^{(I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)}$ , 其中

$$\begin{aligned} j = & (i_{n+1} - 1)I_{n+2}I_{n+3} \cdots I_N I_1 I_2 \cdots I_{n-1} + (i_{n+2} - 1)I_{n+3}I_{n+4} \cdots I_N I_1 I_2 \cdots I_{n-1} + \cdots \\ & + (i_N - 1)I_1 I_2 \cdots I_{n-1} + (i_1 - 1)I_2 I_3 \cdots I_{n-1} + (i_2 - 1)I_3 I_4 \cdots I_{n-1} + \cdots + i_{n-1} \end{aligned} \quad (10.2.5)$$

### 3. Kolda 水平展开方法

Kolda 于 2006 年提出的矩阵化方法 [276, 278] 将  $N$  阶张量元素  $a_{i_1, i_2, \dots, i_N}$  映射为模式- $n$  矩阵  $A_{(n)}$  的元素  $a_{i_n, j}^{(I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)}$ , 其中

$$j = 1 + \sum_{k=1, k \neq n}^N \left[ (i_k - 1) \prod_{m=1, m \neq n}^{k-1} I_m \right] \quad (10.2.6)$$

图 10.2.3 画出了三阶张量矩阵化的 Kolda 方法。

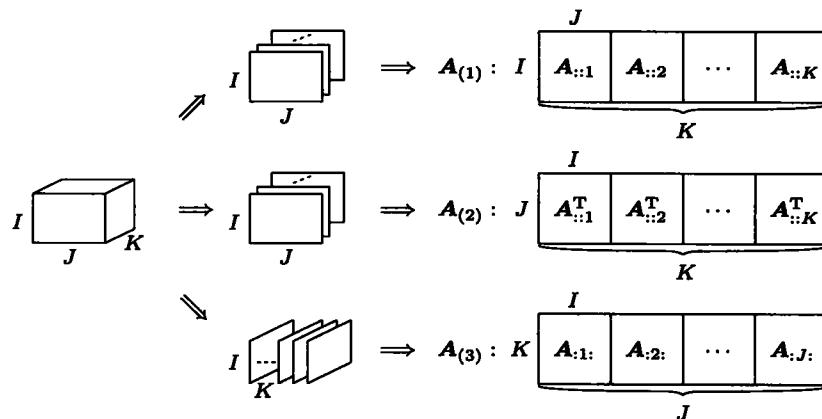


图 10.2.3 三阶张量的水平展开 (Kolda 方法)

三阶张量的 Kolda 矩阵化的元素表示形式为

$$\left. \begin{array}{l} a_{i,(k-1)J+j}^{(I \times JK)} = a_{ijk} \iff \mathbf{A}^{(I \times JK)} = \mathbf{A}_{(1)} = [\mathbf{A}_{::1}, \dots, \mathbf{A}_{::K}] \\ a_{j,(k-1)I+i}^{(J \times KI)} = a_{ijk} \iff \mathbf{A}^{(J \times KI)} = \mathbf{A}_{(2)} = [\mathbf{A}_{::1}^T, \dots, \mathbf{A}_{::K}^T] \\ a_{k,(j-1)I+j}^{(K \times IJ)} = a_{ijk} \iff \mathbf{A}^{(K \times IJ)} = \mathbf{A}_{(3)} = [\mathbf{A}_{::1}, \dots, \mathbf{A}_{::J}] \end{array} \right\} \quad (10.2.7)$$

比较以上三种水平展开方法，可以得出以下结论：

(1) Kolda 方法与 Kiers 方法的模式-1 和模式-3 水平展开矩阵分别相同，即

$$\mathbf{A}_{\text{Kolda}(1)} = \mathbf{A}_{\text{Kiers}(1)}, \quad \mathbf{A}_{\text{Kolda}(3)} = \mathbf{A}_{\text{Kiers}(3)}$$

(2) Kolda 方法与 LMV 方法的模式-2 水平展开矩阵相同

$$\mathbf{A}_{\text{Kolda}(2)} = \mathbf{A}_{\text{LMV}(2)}$$

三阶张量的(列)向量化是将张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  排列成一个  $(I \cdot J \cdot K) \times 1$  列向量  $\mathbf{a}$  的运算，记作  $\mathbf{a}^{(IJK \times 1)} = \text{vec}(\mathcal{A})$ 。三阶张量的行向量化则是将张量排列成行向量的运算，用符号  $\mathbf{a}^{(1 \times IJK)} = \text{rvec}(\mathcal{A})$  表示。

三阶张量的向量化通常定义为正面切片矩阵的列向量化的纵向排列

$$\mathbf{a}^{(IJK \times 1)} \stackrel{\text{def}}{=} \begin{bmatrix} \text{vec}(\mathbf{A}_{::1}) \\ \vdots \\ \text{vec}(\mathbf{A}_{::K}) \end{bmatrix} \quad (10.2.8)$$

其元素的定义公式为

$$a_{(k-1)IJ+(j-1)I+i}^{(IJK \times 1)} = a_{ijk} \iff \mathbf{a}^{(IJK \times 1)} = \text{vec}(\mathbf{A}^{(I \times JK)}) \quad (10.2.9)$$

三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  的行向量化定义为

$$\mathbf{a}^{(1 \times IJK)} \stackrel{\text{def}}{=} \text{rvec}(\mathcal{A}) = [\text{rvec}(\mathbf{A}_{::1}), \dots, \text{rvec}(\mathbf{A}_{::K})] \quad (10.2.10)$$

张量的列向量化与行向量化之间的关系为

$$\text{rvec}(\mathcal{A}) = [\text{vec}^T(\mathbf{A}_{::1}^T), \dots, \text{vec}^T(\mathbf{A}_{::K}^T)] \quad (10.2.11)$$

$$\text{vec}(\mathcal{A}) = [\text{rvec}(\mathbf{A}_{::1}^T), \dots, \text{rvec}(\mathbf{A}_{::K}^T)]^T \quad (10.2.12)$$

更一般地， $N$  阶张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_N}$  的列向量化和行向量化分别为

$$\begin{aligned} & \mathbf{a}_{(i_{N-1})I_1 \dots I_{N-1} + (i_{N-1}-1)I_1 \dots I_{N-2} + \dots + (i_3-1)I_1 I_2 + (i_2-1)I_1 + i_1}^{(I_1 I_2 \dots I_N \times 1)} = a_{i_1 i_2 \dots i_n} \\ & \iff \mathbf{a}^{(I_1 I_2 \dots I_N \times 1)} = \text{vec}(\mathbf{A}^{(I_1 \times I_2 I_3 \dots I_N)}) \end{aligned} \quad (10.2.13)$$

$$\begin{aligned} & \mathbf{a}_{(i_1-1)I_2 \dots I_N + (i_2-1)I_3 \dots I_N + \dots + (i_{N-2}-1)I_{N-1} + (i_{N-1}-1)I_N + i_N}^{(1 \times I_1 I_2 \dots I_N)} = a_{i_1 i_2 \dots i_N} \\ & \iff \mathbf{a}^{(1 \times I_1 I_2 \dots I_N)} = \text{rvec}(\mathbf{A}^{(I_3 \dots I_N I_1 \times I_2)}) \end{aligned} \quad (10.2.14)$$

### 10.2.2 张量的纵向展开

将张量的同一种切片矩阵按照纵向依次排列，称为张量的纵向展开 (longitudinal unfolding)。与水平展开类似，纵向展开也有三种不同的方法，它们分别与水平展开的三种方法相对应。 $N$  阶张量  $\mathbf{A} \in \mathbb{K}^{I_1 I_2 \cdots I_N}$  的模式- $n$  纵向展开用符号  $\mathbf{A}^{(n)}$  表示。

三阶张量  $\mathbf{A} \in \mathbb{K}^{I \times J \times K}$  可以分别矩阵化为  $(JK) \times I$  矩阵、 $(KI) \times J$  矩阵和  $(IJ) \times K$  矩阵，记作

$$\mathbf{A}^{(1)} = \mathbf{A}^{(JK \times I)}, \quad \mathbf{A}^{(2)} = \mathbf{A}^{(KI \times J)}, \quad \mathbf{A}^{(3)} = \mathbf{A}^{(IJ \times K)} \quad (10.2.15)$$

#### 1. 纵向展开的 Keirs 方法

三阶张量的 Keirs 纵向展开为

$$\left. \begin{aligned} a_{(k-1)J+j,i}^{(JK \times I)} = a_{ijk} &\iff \mathbf{A}^{(JK \times I)} = \mathbf{A}^{(1)} = \begin{bmatrix} \mathbf{A}_{::1}^T \\ \vdots \\ \mathbf{A}_{::K}^T \end{bmatrix} \\ a_{(i-1)K+k,j}^{(KI \times J)} = a_{ijk} &\iff \mathbf{A}^{(KI \times J)} = \mathbf{A}^{(2)} = \begin{bmatrix} \mathbf{A}_{1::}^T \\ \vdots \\ \mathbf{A}_{I::}^T \end{bmatrix} \\ a_{(j-1)I+i,k}^{(IJ \times K)} = a_{ijk} &\iff \mathbf{A}^{(IJ \times K)} = \mathbf{A}^{(3)} = \begin{bmatrix} \mathbf{A}_{::1}^T \\ \vdots \\ \mathbf{A}_{::J}^T \end{bmatrix} \end{aligned} \right\} \quad (10.2.16)$$

更一般地， $N$  阶张量  $\mathbf{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$  的模式  $n$ -纵向展开  $\mathbf{A}^{(I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times I_n)}$  的第  $j$  行、第  $i_n$  列元素定义为

$$a_{j,i_n}^{(I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times I_n)} = a_{i_1 i_2 \cdots i_N} \quad (10.2.17)$$

其中  $j$  由式 (10.2.3) 给定。

#### 2. 纵向展开的 LMV 方法

三阶张量的 LMV 纵向展开可以用元素的形式表示为 [303]

$$\left. \begin{aligned} a_{(j-1)K+k,i}^{(JK \times I)} = a_{ijk} &\iff \mathbf{A}^{(JK \times I)} = \mathbf{A}^{(1)} = \begin{bmatrix} \mathbf{A}_{::1} \\ \vdots \\ \mathbf{A}_{::J} \end{bmatrix} \\ a_{(k-1)I+i,j}^{(KI \times J)} = a_{ijk} &\iff \mathbf{A}^{(KI \times J)} = \mathbf{A}^{(2)} = \begin{bmatrix} \mathbf{A}_{1::} \\ \vdots \\ \mathbf{A}_{::K} \end{bmatrix} \\ a_{(i-1)J+j,k}^{(IJ \times K)} = a_{ijk} &\iff \mathbf{A}^{(IJ \times K)} = \mathbf{A}^{(3)} = \begin{bmatrix} \mathbf{A}_{1::} \\ \vdots \\ \mathbf{A}_{I::} \end{bmatrix} \end{aligned} \right\} \quad (10.2.18)$$

$N$  阶张量的 LMV 矩阵化的元素表达式与式 (10.2.17) 相同, 但其中的行下标  $j$  由式 (10.2.5) 确定。

### 3. 纵向展开的 Kolda 方法

与 Kolda 水平展开对应的纵向展开结果为

$$\left. \begin{aligned} a_{(k-1)J+j,i}^{(JK \times I)} = a_{ijk} &\iff \mathbf{A}^{(JK \times I)} = \mathbf{A}^{(1)} = \begin{bmatrix} \mathbf{A}_{::1}^T \\ \vdots \\ \mathbf{A}_{::K}^T \end{bmatrix} \\ a_{(k-1)I+i,j}^{(KI \times J)} = a_{ijk} &\iff \mathbf{A}^{(KI \times J)} = \mathbf{A}^{(2)} = \begin{bmatrix} \mathbf{A}_{::1} \\ \vdots \\ \mathbf{A}_{::K} \end{bmatrix} \\ a_{(j-1)I+i,k}^{(IJ \times K)} = a_{ijk} &\iff \mathbf{A}^{(IJ \times K)} = \mathbf{A}^{(3)} = \begin{bmatrix} \mathbf{A}_{1::}^T \\ \vdots \\ \mathbf{A}_{J::}^T \end{bmatrix} \end{aligned} \right\} \quad (10.2.19)$$

$N$  阶张量的 Kolda 矩阵化的元素表达式也取式 (10.2.17) 的形式, 但其中的行下标  $j$  由式 (10.2.6) 确定。

关于张量的水平展开与纵向展开, 存在以下关系。

#### (1) 三阶张量的纵向展开之间的关系

$$\mathbf{A}_{\text{Kiers}}^{(1)} = \mathbf{A}_{\text{Kolda}}^{(1)}, \quad \mathbf{A}_{\text{Kiers}}^{(3)} = \mathbf{A}_{\text{Kolda}}^{(3)}, \quad \mathbf{A}_{\text{LMV}}^{(2)} = \mathbf{A}_{\text{Kolda}}^{(2)}$$

#### (2) 三阶张量的纵向展开与水平展开之间的关系如下

$$\mathbf{A}_{\text{Kiers}}^{(n)} = (\mathbf{A}_{\text{Kiers}(n)})^T, \quad \mathbf{A}_{\text{LMV}}^{(n)} = (\mathbf{A}_{\text{LMV}(n)})^T, \quad \mathbf{A}_{\text{Kolda}}^{(n)} = (\mathbf{A}_{\text{Kolda}(n)})^T$$

(3) LMV 模式- $n$  纵向展开矩阵是 Kiers 模式- $n$  水平展开的切片矩阵的纵向排列。反之, Kiers 模式- $n$  纵向展开矩阵是 LMV 模式- $n$  水平展开的切片矩阵的纵向排列。

有些文献 (如 [66, 267, 8, 9]) 使用张量的水平展开, 另外一些文献 (如 [320, 303, 278]) 则采用纵向展开。

从下面的例子可以看出同一个张量的三种矩阵化结果之间的联系与不同。

**例 10.2.1** 例 10.1.1 中的三阶张量  $\mathcal{A} \in \mathbb{R}^{3 \times 4 \times 2}$  的三种水平展开结果如下。

#### (1) Keirs 水平展开

$$\mathbf{A}_{(1)} = \begin{bmatrix} 1 & 4 & 7 & 10 & 13 & 16 & 19 & 22 \\ 2 & 5 & 8 & 11 & 14 & 17 & 20 & 23 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 \end{bmatrix}$$

$$\mathbf{A}_{(2)} = \begin{bmatrix} 1 & 13 & 2 & 14 & 3 & 15 \\ 4 & 16 & 5 & 17 & 6 & 18 \\ 7 & 19 & 8 & 20 & 9 & 21 \\ 10 & 22 & 11 & 23 & 12 & 24 \end{bmatrix}$$

$$\mathbf{A}_{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \end{bmatrix}$$

## (2) LMV 水平展开

$$\begin{aligned}\mathbf{A}_{(1)} &= \begin{bmatrix} 1 & 13 & 4 & 16 & 7 & 19 & 10 & 22 \\ 2 & 14 & 5 & 17 & 8 & 20 & 11 & 23 \\ 3 & 15 & 6 & 18 & 9 & 21 & 12 & 24 \end{bmatrix} \\ \mathbf{A}_{(2)} &= \begin{bmatrix} 1 & 2 & 3 & 13 & 14 & 15 \\ 4 & 5 & 6 & 16 & 17 & 18 \\ 7 & 8 & 9 & 19 & 20 & 21 \\ 10 & 11 & 12 & 22 & 23 & 24 \end{bmatrix} \\ \mathbf{A}_{(3)} &= \begin{bmatrix} 1 & 4 & 7 & 10 & 2 & 5 & 8 & 11 & 3 & 6 & 9 & 12 \\ 13 & 16 & 19 & 22 & 14 & 17 & 20 & 23 & 15 & 18 & 21 & 24 \end{bmatrix}\end{aligned}$$

## (3) Kolda 水平展开

$$\begin{aligned}\mathbf{A}_{(1)} &= \begin{bmatrix} 1 & 4 & 7 & 10 & 13 & 16 & 19 & 22 \\ 2 & 5 & 8 & 11 & 14 & 17 & 20 & 23 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 \end{bmatrix} \\ \mathbf{A}_{(2)} &= \begin{bmatrix} 1 & 2 & 3 & 13 & 14 & 15 \\ 4 & 5 & 6 & 16 & 17 & 18 \\ 7 & 8 & 9 & 19 & 20 & 21 \\ 10 & 11 & 12 & 22 & 23 & 24 \end{bmatrix} \\ \mathbf{A}_{(3)} &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \end{bmatrix}\end{aligned}$$

相对应的三种纵向展开结果如下。

## (1) Kiers 纵向展开

$$\mathbf{A}^{(1)} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \\ 19 & 20 & 21 \\ 22 & 23 & 24 \end{bmatrix}, \quad \mathbf{A}^{(2)} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 13 & 16 & 19 & 22 \\ 2 & 5 & 8 & 11 \\ 14 & 17 & 20 & 23 \\ 3 & 6 & 9 & 12 \\ 15 & 18 & 21 & 24 \end{bmatrix}, \quad \mathbf{A}^{(3)} = \begin{bmatrix} 1 & 13 \\ 2 & 14 \\ 3 & 15 \\ \vdots & \vdots \\ 10 & 22 \\ 11 & 23 \\ 12 & 24 \end{bmatrix}$$

## (2) LMV 纵向展开

$$\begin{aligned}\mathbf{A}^{(1)} &= \begin{bmatrix} 1 & 2 & 3 \\ 13 & 14 & 15 \\ 4 & 5 & 6 \\ 16 & 17 & 18 \\ 7 & 8 & 9 \\ 19 & 20 & 21 \\ 10 & 11 & 12 \\ 22 & 23 & 24 \end{bmatrix}, \quad \mathbf{A}^{(2)} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \\ 13 & 16 & 19 & 22 \\ 14 & 17 & 20 & 23 \\ 15 & 18 & 21 & 24 \end{bmatrix} \\ \mathbf{A}^{(3)} &= \begin{bmatrix} 1 & 4 & 7 & 10 & 2 & 5 & 8 & 11 & 3 & 6 & 9 & 12 \\ 13 & 16 & 19 & 22 & 14 & 17 & 20 & 23 & 15 & 18 & 21 & 24 \end{bmatrix}^T\end{aligned}$$

## (3) Kolda 纵向展开

$$\mathbf{A}^{(1)} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \\ 19 & 20 & 21 \\ 22 & 23 & 24 \end{bmatrix}, \quad \mathbf{A}^{(2)} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \\ 13 & 16 & 19 & 22 \\ 14 & 17 & 20 & 23 \\ 15 & 18 & 21 & 24 \end{bmatrix}, \quad \mathbf{A}^{(3)} = \begin{bmatrix} 1 & 13 \\ 2 & 14 \\ 3 & 15 \\ \vdots & \vdots \\ 10 & 22 \\ 11 & 23 \\ 12 & 24 \end{bmatrix}$$

从以上结果, 可以看出:

- ① 不同方法的同一种模式- $n$  水平展开矩阵的差异主要在于列向量的排列有所不同。
- ② 不同方法的同一种模式- $n$  纵向展开矩阵的差异主要在于行向量的排列有所不同。

张量的矩阵化给张量的分析带来方便。但是, 需要注意<sup>[66]</sup>, 张量的矩阵化有可能导致得到的模型存在以下问题:

- ① 数值稳定性略差一点;
- ② 可解释性略差一点;
- ③ 可预测性略差一点;
- ④ 参数的个数可能比较多。

张量的矩阵化是进行张量分析的有效数学工具。然而, 张量分析的最终目的有时又要求能够由矩阵化还原原来的张量。

将一个张量的向量化或矩阵化结果扩展成一个张量的过程称为张量化 (tensorization)、张量的再生 (reshaping) 或重构 (reconstruction)。

水平展开  $\mathbf{A}^{(I \times JK)}$ ,  $\mathbf{A}^{(J \times KI)}$ ,  $\mathbf{A}^{(K \times IJ)}$  和纵向展开  $\mathbf{A}^{(KI \times J)}$ ,  $\mathbf{A}^{(IJ \times K)}$ ,  $\mathbf{A}^{(JK \times I)}$  中的任何一个都可以张量化为三阶张量  $\mathbf{A} \in \mathbb{K}^{I \times J \times K}$ 。例如, 对于 Kiers 矩阵化方法, 张量化的元素定义为

$$a_{ijk} = \mathbf{A}_{i,(k-1)J+j}^{(I \times JK)} = \mathbf{A}_{j,(i-1)K+k}^{(J \times KI)} = \mathbf{A}_{k,(j-1)I+i}^{(K \times IJ)} \quad (10.2.20)$$

$$= \mathbf{A}_{(k-1)J+j,i}^{(JK \times I)} = \mathbf{A}_{(i-1)K+k,j}^{(KI \times J)} = \mathbf{A}_{(j-1)I+i,k}^{(IJ \times K)} \quad (10.2.21)$$

而对于 LMV 方法, 则有

$$a_{ijk} = \mathbf{A}_{i,(j-1)K+k}^{(I \times JK)} = \mathbf{A}_{j,(k-1)I+i}^{(J \times KI)} = \mathbf{A}_{k,(i-1)J+j}^{(K \times IJ)} \quad (10.2.22)$$

$$= \mathbf{A}_{(j-1)K+k,i}^{(JK \times I)} = \mathbf{A}_{(k-1)I+i,j}^{(KI \times J)} = \mathbf{A}_{(i-1)J+j,k}^{(IJ \times K)} \quad (10.2.23)$$

列向量  $\mathbf{a}^{(IJK \times 1)}$  或行向量  $\mathbf{a}^{(1 \times IJK)}$  的张量化定义为

$$a_{ijk} = \mathbf{a}_{(k-1)I+(j-1)J+i}^{(IJK \times 1)} = \mathbf{a}_{(i-1)JK+(j-1)K+k}^{(1 \times IJK)} \quad (10.2.24)$$

其中,  $i = 1, \dots, I; j = 1, \dots, J; k = 1, \dots, K$ 。

更一般地,  $N$  阶张量则可以根据模式- $n$  水平展开公式 (10.2.3) 或公式 (10.2.5) 或公式 (10.2.6) 进行再生或重构。

### 10.3 张量的基本代数运算

张量的基本代数运算主要有张量的乘积、张量与矩阵的乘积以及张量的秩。

#### 10.3.1 张量的内积、范数与外积

张量的内积是向量内积的推广：首先将张量向量化，然后应用向量的内积，即可得到张量的内积。

**定义 10.3.1 (张量内积)** 若  $\mathcal{A}, \mathcal{B} \in \mathcal{T}(I_1, I_2, \dots, I_N)$ ，则  $\mathcal{A}$  和  $\mathcal{B}$  的内积为标量，定义为两个张量的列向量化之间的内积

$$\begin{aligned}\langle \mathcal{A}, \mathcal{B} \rangle &\stackrel{\text{def}}{=} \langle \text{vec}(\mathcal{A}), \text{vec}(\mathcal{B}) \rangle = (\text{vec}(\mathcal{A}))^H \text{vec}(\mathcal{B}) \\ &= \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_n=1}^{I_N} a_{i_1 i_2 \cdots i_n}^* b_{i_1 i_2 \cdots i_n}\end{aligned}\quad (10.3.1)$$

其中 \* 表示复数共轭。

有了张量内积的概念，又可直接引出张量范数的定义。

**定义 10.3.2 (张量的 Frobenius 范数)** 张量  $\mathcal{A}$  的 Frobenius 范数定义为

$$\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle} \stackrel{\text{def}}{=} \left( \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_n=1}^{I_N} |a_{i_1 i_2 \cdots i_n}|^2 \right)^{1/2} \quad (10.3.2)$$

张量的内积与范数具有以下性质 [276]。

**命题 10.3.1** 令  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ ，则 ①

(1) 张量的范数可以转换成该张量的矩阵化函数的范数

$$\|\mathcal{A}\| = \left\| \mathcal{A}^{(I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)} \right\| = \left\| \mathcal{A}^{(I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times I_n)} \right\|$$

(2) 张量的范数可以转换成该张量的向量化函数的范数

$$\|\mathcal{A}\| = \left\| \mathbf{a}^{(I_1 I_2 \cdots I_N \times 1)} \right\| = \left\| \mathbf{a}^{(1 \times I_1 I_2 \cdots I_N)} \right\|$$

(3) 两个张量之差的范数平方

$$\|\mathcal{A} - \mathcal{B}\|^2 = \|\mathcal{A}\|^2 - 2\langle \mathcal{A}, \mathcal{B} \rangle + \|\mathcal{B}\|^2$$

(4) 若  $\mathbf{Q} \in \mathbb{K}^{J \times I_n}$  为标准正交矩阵，即  $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}_{J \times J}$  或  $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}_{I_n \times I_n}$ ，则

$$\|\mathcal{A} \times_n \mathbf{Q}\| = \|\mathcal{A}\|$$

① 命题中出现的向量外积  $\diamond$  和张量的  $n$ -模式积  $\times_n$  将在稍后定义。

(5) 令  $\mathcal{A}, \mathcal{B} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ ,  $\mathbf{a}_n, \mathbf{b}_n \in \mathbb{K}^{J \times I_n}$ , 并且  $\mathcal{A} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \cdots \circ \mathbf{a}_N$  和  $\mathcal{B} = \mathbf{b}_1 \circ \mathbf{b}_2 \circ \cdots \circ \mathbf{b}_N$ , 则

$$\langle \mathcal{A}, \mathcal{B} \rangle = \prod_{n=1}^N \langle \mathbf{a}_n, \mathbf{b}_n \rangle$$

(6) 若  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N}$  和  $\mathcal{B} \in \mathbb{K}^{I_1 \times I_{n-1} \times K \times I_{n+1} \times \cdots \times I_N}$ , 且  $\mathbf{C} \in \mathbb{K}^{J \times K}$ , 则有

$$\langle \mathcal{A}, \mathcal{B} \times_n \mathbf{C} \rangle = \langle \mathcal{A} \times_n \mathbf{C}^H, \mathcal{B} \rangle$$

两个向量的外积 (output product) 为一矩阵, 即有  $\mathbf{X} = \mathbf{u}\mathbf{v}^T$ 。多个向量的外积给出一张量。此时, 就不方便使用向量的转置符号书写外积, 这里沿用大多数文献使用的符号  $\circ$  表示多个向量的外积。

**定义 10.3.3 (向量外积)**  $n$  个向量  $\mathbf{a}^{(i)} \in \mathbb{K}^{i \times 1}, i = 1, \dots, n$  的外积记为  $\mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \cdots \circ \mathbf{a}^{(n)}$ , 其结果为一  $n$  阶张量, 即有

$$\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \cdots \circ \mathbf{a}^{(n)} \quad (10.3.3)$$

或用元素形式定义为

$$a_{i_1 i_2 \cdots i_n} = a_{i_1}^{(1)} a_{i_2}^{(2)} \cdots a_{i_n}^{(n)} \quad (10.3.4)$$

式中  $a_j^{(i)}$  是模式- $i$  向量  $\mathbf{a}^{(i)}$  的第  $j$  个元素。

**例 10.3.1** 两个向量  $\mathbf{u} \in \mathbb{K}^{m \times 1}, \mathbf{v} \in \mathbb{K}^{n \times 1}$  的外积

$$\mathbf{X} = \mathbf{u} \circ \mathbf{v} = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} [v_1, \dots, v_n] = \begin{bmatrix} u_1 v_1 & \cdots & u_1 v_n \\ \vdots & \ddots & \vdots \\ u_m v_1 & \cdots & u_m v_n \end{bmatrix} \in \mathbb{K}^{m \times n}$$

三个向量  $\mathbf{u} \in \mathbb{K}^{I \times 1}, \mathbf{v} \in \mathbb{K}^{J \times 1}, \mathbf{w} \in \mathbb{K}^{K \times 1}$  的外积

$$\mathcal{A} = \mathbf{u} \circ \mathbf{v} \circ \mathbf{w} = \begin{bmatrix} u_1 v_1 & \cdots & u_1 v_J \\ \vdots & \ddots & \vdots \\ u_I v_1 & \cdots & u_I v_J \end{bmatrix} \circ [w_1, \dots, w_K] \in \mathbb{K}^{I \times J \times K}$$

其正面切片矩阵

$$\mathbf{A}_{::k} = \begin{bmatrix} u_1 v_1 w_k & \cdots & u_1 v_J w_k \\ \vdots & \ddots & \vdots \\ u_I v_1 w_k & \cdots & u_I v_J w_k \end{bmatrix}, \quad k = 1, \dots, K$$

即有  $a_{ijk} = u_i v_j w_k, i = 1, \dots, I; j = 1, \dots, J; k = 1, \dots, K$ 。

向量的外积容易推广为张量的外积。

**定义 10.3.4 (张量外积)** 两个张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_P}$  和  $\mathcal{B} \in \mathbb{K}^{J_1 \times J_2 \times \cdots \times J_Q}$  的外积仍然是张量, 记作  $\mathcal{A} \circ \mathcal{B} \in \mathbb{K}^{I_1 \times \cdots \times I_P \times J_1 \times \cdots \times J_Q}$ , 定义为

$$(\mathcal{A} \circ \mathcal{B})_{i_1 \cdots i_P j_1 \cdots j_Q} = a_{i_1 \cdots i_P} b_{j_1 \cdots j_Q} \quad \forall i_1, \dots, i_P; j_1, \dots, j_Q \quad (10.3.5)$$

### 10.3.2 张量的 $n$ -模式积

为了引出高阶张量与矩阵的乘积，先考虑三阶张量与矩阵的乘积。

三阶张量与矩阵的乘积是 Tucker 定义的<sup>[486, 487]</sup>，现在常称为 Tucker 积<sup>[303]</sup>。

**定义 10.3.5 (三阶张量的 Tucker 积)** 考虑三阶张量  $\mathcal{X} \in \mathbb{K}^{I_1 \times I_2 \times I_3}$  和矩阵  $\mathbf{A} \in \mathbb{K}^{J_1 \times I_1}, \mathbf{B} \in \mathbb{K}^{J_2 \times I_2}, \mathbf{C} \in \mathbb{K}^{J_3 \times I_3}$  的乘积。三阶张量的 Tucker 模式-1 积  $\mathcal{X} \times_1 \mathbf{A}$ ，模式-2 积  $\mathcal{X} \times_2 \mathbf{B}$  和模式-3 积  $\mathcal{X} \times_3 \mathbf{C}$  分别定义为<sup>[303]</sup>

$$(\mathcal{X} \times_1 \mathbf{A})_{j_1 i_2 i_3} = \sum_{i_1=1}^{I_1} x_{i_1 i_2 i_3} a_{j_1 i_1}, \forall j_1, i_2, i_3 \quad (10.3.6)$$

$$(\mathcal{X} \times_2 \mathbf{B})_{i_1 j_2 i_3} = \sum_{i_2=1}^{I_2} x_{i_1 i_2 i_3} b_{j_2 i_2}, \forall i_1, j_2, i_3 \quad (10.3.7)$$

$$(\mathcal{X} \times_3 \mathbf{C})_{i_1 i_2 j_3} = \sum_{i_3=1}^{I_3} x_{i_1 i_2 i_3} c_{j_3 i_3}, \forall i_1, i_2, j_3 \quad (10.3.8)$$

下面解读张量的 Tucker 模式- $n$  积。首先，张量的模式-1 积可以用张量符号表示

$$\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A} \quad (10.3.9)$$

其次，由三阶张量的模式-1 水平展开的元素定义公式 (10.2.7) 和纵向展开的元素定义公式 (10.2.18) 分别有

$$\begin{aligned} y_{j_1 i_2 i_3} &= \sum_{i_1=1}^{I_1} x_{i_1 i_2 i_3} a_{j_1 i_1} = \sum_{i_1=1}^{I_1} \mathbf{X}_{i_1, (i_3-1)I_2+i_2}^{(I_1 \times I_2 I_3)} a_{j_1 i_1} = (\mathbf{A} \mathbf{X}^{(I_1 \times I_2 I_3)})_{j_1, (i_3-1)I_2+i_2} \\ y_{j_1 i_2 i_3} &= \sum_{i_1=1}^{I_1} x_{i_1 i_2 i_3} a_{j_1 i_1} = \sum_{i_1=1}^{I_1} \mathbf{X}_{(i_2-1)I_3+i_3, i_1}^{(I_2 I_3 \times I_1)} a_{j_1 i_1} = (\mathbf{X}^{(I_2 I_3 \times I_1)} \mathbf{A}^T)_{(i_2-1)I_3+i_3, j_1} \end{aligned}$$

另由公式 (10.2.7) 和 (10.2.18) 分别知  $\mathbf{Y}_{j_1, (i_3-1)I_2+i_2}^{(J_1 \times I_2 I_3)} = y_{j_1 i_2 i_3}$  和  $\mathbf{Y}_{(i_2-1)I_3+i_3, j_1}^{(I_2 I_3 \times J_1)} = y_{j_1 i_2 i_3}$ 。于是，三阶张量的模式-1 积可以使用模式-1 扁平化矩阵表示为

$$\mathbf{Y}^{(J_1 \times I_2 I_3)} = (\mathcal{X} \times_1 \mathbf{A})^{(J_1 \times I_2 I_3)} = \mathbf{A} \mathbf{X}^{(I_1 \times I_2 I_3)}$$

$$\mathbf{Y}^{(I_2 I_3 \times J_1)} = (\mathcal{X} \times_1 \mathbf{A})^{(I_2 I_3 \times J_1)} = \mathbf{X}^{(I_2 I_3 \times I_1)} \mathbf{A}^T$$

仿此，可以得到三阶张量的模式-2 和模式-3 积的矩阵表示，现汇总于下

$$\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A} \iff \begin{cases} \mathbf{Y}^{(J_1 \times I_2 I_3)} = \mathbf{A} \mathbf{X}^{(I_1 \times I_2 I_3)} \\ \mathbf{Y}^{(I_2 I_3 \times J_1)} = \mathbf{X}^{(I_2 I_3 \times I_1)} \mathbf{A}^T \end{cases} \quad (10.3.10)$$

$$\mathcal{Y} = \mathcal{X} \times_2 \mathbf{B} \iff \begin{cases} \mathbf{Y}^{(J_2 \times I_3 I_1)} = \mathbf{B} \mathbf{X}^{(I_2 \times I_3 I_1)} \\ \mathbf{Y}^{(I_3 I_1 \times J_2)} = \mathbf{X}^{(I_3 I_1 \times I_2)} \mathbf{B}^T \end{cases} \quad (10.3.11)$$

$$\mathcal{Y} = \mathcal{X} \times_3 \mathbf{C} \iff \begin{cases} \mathbf{Y}^{(J_3 \times I_1 I_2)} = \mathbf{C} \mathbf{X}^{(I_3 \times I_1 I_2)} \\ \mathbf{Y}^{(I_1 I_2 \times J_3)} = \mathbf{X}^{(I_1 I_2 \times I_3)} \mathbf{C}^T \end{cases} \quad (10.3.12)$$

上述分析可以得出以下结论:

(1) 三阶张量  $\mathcal{X}$  的模式-1 矩阵积  $\mathcal{X} \times_1 \mathbf{A}$  相当于取矩阵  $\mathbf{A}$  与  $\mathcal{X}$  的模式-1 水平展开  $\mathbf{X}^{(I_1 \times I_2 I_3)}$  的乘法, 其乘积直接给出  $\mathcal{X} \times_1 \mathbf{A}$  的模式-1 水平展开, 或者等价于取  $\mathcal{X}$  的模式-1 纵向展开  $\mathbf{X}^{(I_2 I_3 \times I_1)}$  与矩阵转置  $\mathbf{A}^T$  之间的乘法, 其乘积为  $\mathcal{X} \times_1 \mathbf{A}$  的模式-1 纵向展开。

(2) 三阶张量  $\mathcal{X}$  的模式-2 矩阵积  $\mathcal{X} \times_2 \mathbf{B}$  相当于取矩阵  $\mathbf{B}$  与  $\mathcal{X}$  的模式-2 水平展开  $\mathbf{X}^{(I_2 \times I_3 I_1)}$  的乘法, 其乘积直接给出  $\mathcal{X} \times_2 \mathbf{B}$  的模式-2 水平展开, 或者等价于取  $\mathcal{X}$  的模式-2 纵向展开  $\mathbf{X}^{(I_3 I_1 \times I_2)}$  与矩阵转置  $\mathbf{B}^T$  的乘法, 其乘积直接是  $\mathcal{X} \times_2 \mathbf{B}$  的模式-2 纵向展开。

(3) 张量  $\mathcal{X}$  的模式-3 矩阵积  $\mathcal{X} \times_3 \mathbf{C}$  相当于取矩阵  $\mathbf{C}$  与  $\mathcal{X}$  的模式-3 水平展开  $\mathbf{X}^{(I_3 \times I_1 I_2)}$  的乘法, 其乘积直接给出  $\mathcal{X} \times_3 \mathbf{C}$  的模式-3 水平展开, 或者等价于取  $\mathcal{X}$  的模式-3 纵向展开  $\mathbf{X}^{(I_2 I_3 \times I_1)}$  与矩阵转置  $\mathbf{C}^T$  的乘法, 其乘积直接给出  $\mathcal{X} \times_3 \mathbf{C}$  的模式-3 纵向展开。

**例 10.3.2** 已知三阶张量  $\mathcal{X} \in \mathbb{R}^{3 \times 4 \times 2}$  的两个正面切片由

$$\mathbf{X}_{::1} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}, \quad \mathbf{X}_{::2} = \begin{bmatrix} 13 & 14 & 15 & 16 \\ 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 \end{bmatrix} \quad (10.3.13)$$

给出, 分别计算张量  $\mathcal{X}$  与矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

的乘积。由式 (10.3.6) 可求得  $\mathcal{X} \times_1 \mathbf{A} \in \mathbb{R}^{2 \times 4 \times 2}$  的两个正面切片分别为

$$\begin{aligned} (\mathcal{X} \times_1 \mathbf{A})_{::1} &= \begin{bmatrix} 38 & 44 & 50 & 56 \\ 83 & 98 & 113 & 138 \end{bmatrix} \\ (\mathcal{X} \times_1 \mathbf{A})_{::2} &= \begin{bmatrix} 110 & 116 & 122 & 128 \\ 263 & 278 & 293 & 308 \end{bmatrix} \end{aligned}$$

另由式 (10.3.8) 可求得  $\mathcal{X} \times_3 \mathbf{B} \in \mathbb{R}^{3 \times 4 \times 3}$  的三个正面切片分别为

$$\begin{aligned} (\mathcal{X} \times_3 \mathbf{B})_{::1} &= \begin{bmatrix} 27 & 30 & 33 & 36 \\ 39 & 42 & 45 & 48 \\ 51 & 54 & 57 & 60 \end{bmatrix} \\ (\mathcal{X} \times_3 \mathbf{B})_{::2} &= \begin{bmatrix} 55 & 62 & 69 & 76 \\ 83 & 90 & 97 & 104 \\ 111 & 118 & 125 & 132 \end{bmatrix} \\ (\mathcal{X} \times_3 \mathbf{B})_{::3} &= \begin{bmatrix} 83 & 94 & 105 & 116 \\ 127 & 138 & 149 & 160 \\ 171 & 182 & 193 & 204 \end{bmatrix} \end{aligned}$$

图 10.3.1 画出了三阶张量  $\mathcal{X} \in \mathbb{R}^{8 \times 7 \times 4}$  与向量  $\mathbf{u}_1 \in \mathbb{R}^{8 \times 1 \times 1}, \mathbf{u}_2 \in \mathbb{R}^{1 \times 7 \times 1}, \mathbf{u}_3 \in \mathbb{R}^{1 \times 1 \times 4}$  的 3-模式向量积  $\mathcal{X} \times_1 \mathbf{u}_1 \times_2 \mathbf{u}_2 \times_3 \mathbf{u}_3$  的运算原理图。

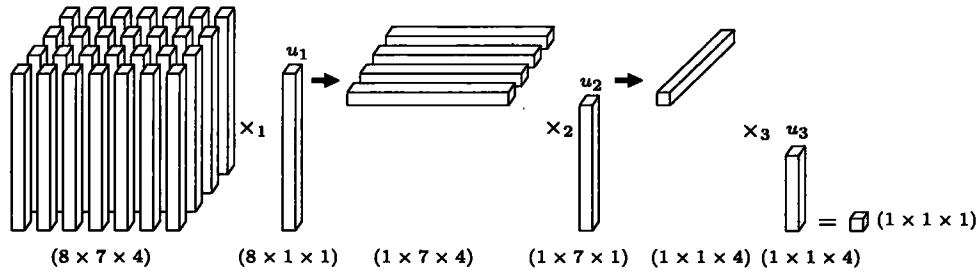


图 10.3.1 三阶张量的 3-模式向量积的原理图

由上图可以得出以下结论：

- (1) 三阶张量的每次模式向量积都使张量的阶数减一，最后变成零阶张量即标量。
- (2) 三阶张量的模式向量积的顺序可以交换。

Tucker 积可以推广为  $N$  阶张量的  $n$ -模式矩阵积。

**定义 10.3.6 ( $n$ -模式矩阵积)** 一个  $N$  阶张量  $\mathcal{X} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$  与一个  $J_n \times I_n$  矩阵  $U^{(n)}$  的  $n$ -模式(矩阵)积用符号  $\mathcal{X} \times_n U^{(n)}$  表示。这是一个  $I_1 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \cdots \times I_N$  张量，其元素定义为<sup>[298]</sup>

$$(\mathcal{X} \times_n U^{(n)})_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} \stackrel{\text{def}}{=} \sum_{i_n=1}^{J_n} x_{i_1 i_2 \cdots i_N} a_{j i_n} \quad (10.3.14)$$

其中  $j = 1, \dots, J_n; i_k = 1, \dots, I_k; k = 1, \dots, N$ 。

由上述定义易知，一个  $N$  阶张量  $\mathcal{X} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$  与单位矩阵  $I_{I_n \times I_n}$  的  $n$ -模式积等于原张量，即有

$$\mathcal{X} \times_n I_{I_n \times I_n} = \mathcal{X} \quad (10.3.15)$$

类似于三阶张量， $N$  阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$  与矩阵  $U^{(n)} \in \mathbb{R}^{J_n \times I_n}$  的  $n$ -模式积可以用张量的模式- $n$  矩阵化表示为

$$Y = \mathcal{X} \times_n U^{(n)} \iff Y_{(n)} = U^{(n)} \mathbf{X}_{(n)} \quad \text{或} \quad Y^{(n)} = \mathbf{X}^{(n)} U^{(n)\text{T}} \quad (10.3.16)$$

式中  $\mathbf{X}_{(n)} = \mathbf{X}^{(I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)}$  和  $\mathbf{X}^{(n)} = \mathbf{X}^{(I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times I_n)}$  分别是  $N$  阶张量  $\mathcal{X}$  的模式- $n$  水平展开和纵向展开。

张量的  $n$ -模式积具有以下性质<sup>[298]</sup>。

**命题 10.3.2** 令  $\mathcal{X} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$  是  $N$  阶张量。

(1) 给定矩阵  $A \in \mathbb{K}^{J_m \times I_m}, B \in \mathbb{K}^{J_n \times I_n}$ ，若  $m \neq n$ ，则

$$\mathcal{X} \times_m A \times_n B = (\mathcal{X} \times_m A) \times_n B = (\mathcal{X} \times_n B) \times_m A = \mathcal{X} \times_n B \times_m A$$

(2) 给定矩阵  $A \in \mathbb{K}^{J \times I_n}, B \in \mathbb{K}^{I_n \times J}$ ，则

$$\mathcal{X} \times_n A \times_n B = \mathcal{X} \times_n (BA)$$

(3) 若  $\mathbf{A} \in \mathbb{K}^{J \times I_n}$  具有满列秩, 则

$$\mathcal{Y} = \mathcal{X} \times_n \mathbf{A} \implies \mathcal{X} = \mathcal{Y} \times_n \mathbf{A}^\dagger$$

(4) 若  $\mathbf{A} \in \mathbb{K}^{J \times I_n}$  是标准半正交的, 即  $\mathbf{A}^H \mathbf{A} = \mathbf{I}_{I_n}$ , 则

$$\mathcal{Y} = \mathcal{X} \times_n \mathbf{A} \implies \mathcal{X} = \mathcal{Y} \times_n \mathbf{A}^H$$

性质 (3) 和 (4) 表明, 一个张量  $\mathcal{X}$  可以由张量的  $n$ -模式积  $\mathcal{Y} = \mathcal{X} \times_n \mathbf{A}$  通过  $\mathcal{X} = \mathcal{Y} \times_n \mathbf{A}^\dagger$  或者  $\mathcal{X} = \mathcal{Y} \times_n \mathbf{A}^H$  恢复或者重构。

**例 10.3.3** 一个三阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$  的两个正面切片分别为

$$\mathbf{X}_{::1} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, \quad \mathbf{X}_{::2} = \begin{bmatrix} 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix}$$

并且

$$\mathbf{A} = \begin{bmatrix} -0.7071 & 0.5774 \\ 0.0000 & 0.5774 \\ 0.7071 & 0.5774 \end{bmatrix} \in \mathbb{R}^{J \times I_1}$$

是标准半正交矩阵, 即  $\mathbf{A}^T \mathbf{A} = \mathbf{I}_2$ 。题中,  $I_1 = 2, I_2 = 3, I_3 = 2, J = 3$ 。于是,  $\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A} \in \mathbb{R}^{J \times I_2 \times I_3}$  的模式-1 水平展开为

$$\begin{aligned} \mathbf{Y}^{(J \times I_2 I_3)} &= \mathbf{A} \mathbf{X}^{(I_1 \times I_2 I_3)} = \begin{bmatrix} -0.70711 & 0.57735 \\ 0.00000 & 0.57735 \\ 0.70711 & 0.57735 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 7 & 8 & 9 \\ 4 & 5 & 6 & 10 & 11 & 12 \end{bmatrix} \\ &= \begin{bmatrix} 1.60229 & 1.47253 & 1.34277 & 0.82373 & 0.69397 & 0.56421 \\ 2.30940 & 2.88675 & 3.46410 & 5.77350 & 6.35085 & 6.92820 \\ 3.01651 & 4.30097 & 5.58543 & 10.72327 & 12.00773 & 13.29219 \end{bmatrix} \end{aligned}$$

如果已知矩阵  $\mathbf{A}$  和  $\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A}$  的模式-1 水平展开由上式给出, 则由张量的模式- $n$  积的性质 (4) 知, 张量  $\mathcal{X} = \mathcal{Y} \times_1 \mathbf{A}^T$  的模式-1 水平展开可以由  $\mathbf{X}^{(I_1 \times I_2 I_3)} = \mathbf{A}^T \mathbf{Y}^{(J \times I_2 I_3)}$  恢复

$$\begin{aligned} \hat{\mathbf{X}}^{(I_1 \times I_2 I_3)} &= \begin{bmatrix} -0.70711 & 0.00000 & 0.70711 \\ 0.57735 & 0.57735 & 0.57735 \end{bmatrix} \\ &\times \begin{bmatrix} 1.60229 & 1.47253 & 1.34277 & 0.82373 & 0.69397 & 0.56421 \\ 2.30940 & 2.88675 & 3.46410 & 5.77350 & 6.35085 & 6.92820 \\ 3.01651 & 4.30097 & 5.58543 & 10.72327 & 12.00773 & 13.29219 \end{bmatrix} \\ &= \begin{bmatrix} 1.00000 & 2.00002 & 3.00003 & 7.00006 & 8.00007 & 9.00008 \\ 4.00000 & 5.00000 & 5.99999 & 9.99999 & 10.99999 & 11.99999 \end{bmatrix} \end{aligned}$$

由此得张量的正面切片的估计

$$\hat{\mathbf{X}}_{::1} = \begin{bmatrix} 1.00000 & 2.00002 & 3.00003 \\ 4.00000 & 5.00000 & 5.99999 \end{bmatrix}$$

$$\hat{\mathbf{X}}_{::2} = \begin{bmatrix} 7.00006 & 8.00007 & 9.00008 \\ 9.99999 & 10.99999 & 11.99999 \end{bmatrix}$$

### 10.3.3 张量的秩

若张量  $\mathcal{A} \in \mathcal{T}(I_1, I_2, \dots, I_n)$  可以分解为

$$\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(n)} \quad (10.3.17)$$

则称为可分解张量 (decomposed tensor)。式中, 向量  $\mathbf{a}^{(i)} \in \mathbb{K}^{I_i}, i = 1, \dots, n$  称为可分解张量  $\mathcal{A}$  的分量或因子。由于各个因子为秩 1 向量, 故上述分解称为张量的秩 1 分解。

可分解张量的元素定义公式为

$$a_{i_1 i_2 \dots i_n} = a_{i_1}^{(1)} a_{i_2}^{(2)} \dots a_{i_n}^{(n)} \quad (10.3.18)$$

所有  $I_1 \times I_2 \times \dots \times I_n$  的可分解张量的集合称为可分解张量集, 用符号  $\mathcal{D}(I_1, I_2, \dots, I_n)$  表示, 或简记为  $\mathcal{D}$ 。

**引理 10.3.1** [274] 对于两个可分解张量  $\mathcal{A}, \mathcal{B} \in \mathcal{D}$ , 若

$$\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(n)}, \quad \mathcal{B} = \mathbf{b}^{(1)} \circ \mathbf{b}^{(2)} \circ \dots \circ \mathbf{b}^{(n)} \quad (10.3.19)$$

则下列结果为真:

$$(1) \langle \mathcal{A}, \mathcal{B} \rangle = \prod_{i=1}^n \langle \mathbf{a}^{(i)}, \mathbf{b}^{(i)} \rangle;$$

$$(2) \|\mathcal{A}\|_F = \sqrt{\sum_{i=1}^n \|\mathbf{a}^{(i)}\|_F^2} \text{ 和 } \|\mathcal{B}\|_F = \sqrt{\sum_{i=1}^n \|\mathbf{b}^{(i)}\|_F^2};$$

(3)  $\mathcal{A} + \mathcal{B} \in \mathcal{D}$ , 当且仅当  $\mathcal{A}$  和  $\mathcal{B}$  最多只有一个分量不同, 其他分量全部相同。

令  $\|\mathcal{A}\|_F = 1$  和  $\|\mathcal{B}\|_F = 1$ 。称两个可分解张量  $\mathcal{A}$  和  $\mathcal{B}$  正交, 并记作  $\mathcal{A} \perp \mathcal{B}$ , 若这两个张量的内积等于零, 即

$$\langle \mathcal{A}, \mathcal{B} \rangle = \prod_{i=1}^n \langle \mathbf{a}^{(i)}, \mathbf{b}^{(i)} \rangle = 0.$$

在矩阵代数中, 一个矩阵  $\mathbf{A}$  的线性独立的行 (或列) 向量的最大个数称为矩阵  $\mathbf{A}$  的行 (或列) 秩, 或者等价地,  $\mathbf{A}$  的列 (或行) 秩就是  $\mathbf{A}$  的列 (或行) 空间的维数。一个矩阵的列秩和行秩总是相等。因此, 一个矩阵的秩、列秩和行秩相同。然而, 矩阵秩的这一重要性质对高阶张量却不再成立。

相对于矩阵的列秩/行秩, 张量的模式- $n$  向量的秩称为张量的模式- $n$  秩。

**定义 10.3.7** (张量的模式- $n$  秩) [302]  $N$  阶张量  $\mathcal{A} \in \mathbb{K}^{I_1 \times \dots \times I_N}$  的  $I_1 \dots I_{n-1} I_{n+1} \dots I_N$  个  $I_n$  维模式- $n$  向量中, 相互线性无关的向量的最大个数称为张量  $\mathcal{A}$  的模式- $n$  秩 (mode- $n$  rank), 用符号  $r_n = \text{rank}_n(\mathcal{A})$  记之, 或者等价叙述为: 一个张量的模式- $n$  向量所张成的子空间的维数称为该张量的模式- $n$  秩。

例如, 三阶张量  $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$  的模式- $n$  秩用符号  $r_n(\mathcal{A})$  表示, 定义为

$$r_1(\mathcal{A}) \stackrel{\text{def}}{=} \dim(\text{span}_{\mathbb{R}}\{\mathbf{a}_{:,j,k} | j = 1, \dots, J, k = 1, \dots, K\})$$

$$r_2(\mathcal{A}) \stackrel{\text{def}}{=} \dim(\text{span}_{\mathbb{R}}\{\mathbf{a}_{i,:,k} | i = 1, \dots, I, k = 1, \dots, K\})$$

$$r_3(\mathcal{A}) \stackrel{\text{def}}{=} \dim(\text{span}_{\mathbb{R}}\{\mathbf{a}_{i,j,:} | i = 1, \dots, I, j = 1, \dots, J\})$$

其中,  $\mathbf{a}_{ijk}, \mathbf{a}_{ik}, \mathbf{a}_{ij}$  分别是张量的模式-1、模式-2 和模式-3 向量。

矩阵的列秩和行秩总是相等。但是, 若  $i \neq j$ , 则高阶张量的模式- $i$  秩和模式- $j$  秩一般不相同。

三阶张量  $\mathcal{A} \in \mathbb{K}^{I \times J \times K}$  是秩- $(r_1, r_2, r_3)$  的, 若它的模式-1 秩、模式-2 秩和模式-3 秩分别为  $r_1, r_2$  和  $r_3$ 。更一般地, 称张量  $\mathcal{A}$  是秩  $(r_1, r_2, \dots, r_N)$  的, 若其模式- $n$  秩等于  $r_n, n = 1, \dots, N$ 。特别地, 若每一个模式- $n$  矩阵化的秩都等于 1, 则称该张量是秩  $(1, 1, \dots, 1)$  的。

$N$  阶张量的模式- $n$  秩共有  $N$  个, 使用起来往往不很方便。为此, 有必要对一个张量只定义一个秩。这样的秩称为张量的秩。

考虑将张量  $\mathcal{A} \in \mathcal{T}$  分解成若干可分解张量的加权求和

$$\mathcal{A} = \sum_{i=1}^R \sigma_i \mathcal{U}_i \quad (10.3.20)$$

其中,  $\sigma_i > 0, i = 1, \dots, R; \mathcal{U}_i \in \mathcal{D}$ , 并且  $\|\mathcal{U}_i\|_F = 1, i = 1, \dots, R$ 。

张量  $\mathcal{A}$  的秩记为  $\text{rank}(\mathcal{A})$ , 定义为<sup>[286]</sup>: 使式 (10.3.20) 成立的最小  $R$ 。此时, 式 (10.3.20) 称为张量的秩分解 (rank decomposition)。

特别地, 若  $R = 1$ , 即  $\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(n)}$ , 则称  $\mathcal{A}$  为秩-1 张量。因此, 一个三阶张量  $\mathcal{A}$  是秩 1 张量, 若它可以表示为三个向量的外积, 即  $\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \mathbf{a}^{(3)}$ 。类似地, 三阶张量  $\mathcal{A}$  是秩 2 张量 (即张量的秩为 2), 若它可以表示为  $\mathcal{A} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \mathbf{a}^{(3)} + \mathbf{b}^{(1)} \circ \mathbf{b}^{(2)} \circ \mathbf{b}^{(3)}$ 。

张量的秩与矩阵的秩有很大的不同。一个不同点是同一个张量在实数域  $\mathbb{R}$  和复数域  $\mathbb{C}$  的秩可能不同, 如同下面的例子所示。

**例 10.3.4**<sup>[288]</sup> 张量  $\mathcal{A} \in \mathbb{R}^{2 \times 2 \times 2}$  的正面切片矩阵为

$$\mathbf{A}_{::1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{A}_{::2} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

则在实数域, 有

$$\mathcal{A} = \sum_{i=1}^3 \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i$$

式中  $\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i$  分别是矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 1 & 1 \end{bmatrix}$$

的第  $i$  列。因此, 张量在实数域的秩等于 3。然而, 在复数域, 张量的秩却等于 2, 因为

$$\mathcal{A} = \sum_{i=1}^2 \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i$$

式中  $\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i$  分别是下列复矩阵的第  $i$  列

$$\mathbf{A} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -j & j \end{bmatrix}, \quad \mathbf{B} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ j & -j \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 1 \\ -j & j \end{bmatrix}$$

对于一般的三阶张量  $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ , 只知道最大的张量秩存在一个弱的上界<sup>[287]</sup>

$$\text{rank}(\mathcal{A}) \leq \min\{IJ, IK, JK\} \quad (10.3.21)$$

若  $K = 2$  或者  $I = 2$ , 则<sup>[287]</sup>

$$\text{rank}(\mathcal{A}) \leq \min\{I, J\} + \min\{I, J, \lfloor \max\{I, J\}/2 \rfloor\} \quad (\text{若 } K = 2) \quad (10.3.22)$$

$$\text{rank}(\mathcal{A}) \leq \min\{J, K\} + \min\{J, K, \lfloor \max\{J, K\}/2 \rfloor\} \quad (\text{若 } I = 2) \quad (10.3.23)$$

其中  $\lfloor x \rfloor$  表示  $\leq x$  的最大整数。

文献 [278] 汇总了具有不同  $I, J, K$  的三阶张量的典型秩。

## 10.4 张量的 Tucker 分解

为了进行张量的信息挖掘, 需要对张量进行分解。张量分解的概念源自 Hitchcock 于 1927 年在数学和物理杂志上发表的两篇论文<sup>[234, 235]</sup>, 他提出一个张量可以表示为有限个秩 1 张量之和, 并称为典范多元分解 (canonical polyadic decomposition)。多路模型的概念则是由 Cattell 于 1944 年在心理测验学杂志提出的<sup>[94]</sup>。然而, 张量分解和多路模型这些概念只是到了 20 世纪 60 年代之后, 才引起人们的相继关注: Tucker<sup>[485, 486, 487]</sup> 相继发表了三篇关于张量因子分解方法的论文, Carroll 与 Chang<sup>[93]</sup> 以及 Harshman<sup>[216]</sup> 于 1970 年分别独立地提出了典范因子分解 (canonical factor decomposition, CANDECOMP) 和平行因子分解 (parallel factor decomposition, PARAFAC), 从而奠定了张量分解的两大类方法:

- (1) Tucker 分解, 又称高阶奇异值分解 (higher-order SVD);
- (2) 典范/平行因子分解 (CANDECOMP/PARAFAC), 常简称为 CP 分解。

本节介绍张量的 Tucker 分解, 它是 SVD 概念的多线性推广, 而张量的 CP 分解则留待 10.5 节专题讨论。

### 10.4.1 Tucker 分解 (高阶奇异值分解)

Tucker 分解与 Tucker 算子密切相关, 而 Tucker 算子是张量与矩阵的多模式乘法的一种有效表示。

**定义 10.4.1** 令  $\mathcal{G} \in \mathbb{K}^{J_1 \times J_2 \times \cdots \times J_N}$ , 矩阵  $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n \times J_n}$ , 其中  $n \in \{1, \dots, N\}$ , 则 Tucker 算子定义为<sup>[276]</sup>

$$[\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}] \stackrel{\text{def}}{=} \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} \quad (10.4.1)$$

其结果是一个  $N$  阶  $I_1 \times I_2 \times \cdots \times I_N$  张量。

给定  $N$  阶张量  $\mathcal{G} \in \mathbb{K}^{J_1 \times J_2 \times \cdots \times J_N}$  和标号集合  $\mathcal{N} = \{1, \dots, N\}$ , 则 Tucker 算子有以下性质<sup>[276]</sup>:

(1) 若  $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n \times J_n}, n \in \mathcal{N}$ , 则

$$[\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]; \mathbf{V}^{(1)}, \dots, \mathbf{V}^{(N)}] = [\mathcal{G}; \mathbf{V}^{(1)}\mathbf{U}^{(1)}, \dots, \mathbf{V}^{(N)}\mathbf{U}^{(N)}]$$

(2) 若  $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n \times J_n}, n \in \mathcal{N}$  具有满列秩, 则

$$\mathcal{X} = [\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}] \iff \mathcal{G} = [\mathcal{X}; \mathbf{U}^{(1)\dagger}, \dots, \mathbf{U}^{(N)\dagger}]$$

(3) 若  $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n \times J_n}$  (其中  $J_n \leq I_n$ , 且  $n \in \mathcal{N}$  为标准正交矩阵), 即  $\mathbf{U}^{(n)\mathrm{T}}\mathbf{U}^{(n)} = \mathbf{I}_{J_n}$ , 则

$$\mathcal{X} = [\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}] \iff \mathcal{G} = [\mathcal{X}; \mathbf{U}^{(1)\mathrm{T}}, \dots, \mathbf{U}^{(N)\mathrm{T}}]$$

**命题 10.4.1**<sup>[276]</sup> 考虑张量  $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$ , 令  $\mathcal{N} = \{1, \dots, N\}$ , 矩阵  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times J_n}, n \in \mathcal{N}$ . 若每个矩阵有 QR 分解  $\mathbf{U}^{(n)} = \mathbf{Q}_n \mathbf{R}_n, \forall n \in \mathcal{N}$ , 其中  $\mathbf{Q}_n$  是标准正交矩阵,  $\mathbf{R}_n$  为上三角矩阵, 则

$$\|[\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]\| = \|[\mathcal{X}; \mathbf{R}_1, \dots, \mathbf{R}_N]\|$$

这一命题表明, 若对张量  $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$ , 有 Tucker 算子

$$\mathcal{X} = [\mathcal{G}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}] \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$$

并且  $J_n \ll I_n$ , 则张量  $\mathcal{X}$  的范数与一维数小得多的张量

$$\mathcal{Z} = [\mathcal{G}; \mathbf{R}_1, \dots, \mathbf{R}_N] \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$$

的范数相同, 即有  $\|\mathcal{X}\| = \|\mathcal{Z}\|$ .

矩阵  $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2}$  是一个二模式的数学对象, 它有两个相伴的向量空间: 列空间和行空间。奇异值分解 (SVD) 将这两个向量空间正交化, 并将矩阵分解为三个矩阵的乘积  $\mathbf{A} = \mathbf{U}_1 \boldsymbol{\Sigma} \mathbf{U}_2^T$ , 其中, 左奇异矩阵  $\mathbf{U}_1 \in \mathbb{R}^{I_1 \times J_1}$  的  $J_1$  个左奇异向量张成  $\mathbf{A}$  的列空间, 中间的矩阵  $\boldsymbol{\Sigma}$  是一个  $J_1 \times J_2$  的对角奇异值矩阵, 而右奇异矩阵  $\mathbf{U}_2 \in \mathbb{R}^{I_2 \times J_2}$  的  $J_2$  个右奇异向量张成  $\mathbf{A}$  的行空间。

由于奇异值的作用往往比左和右奇异向量更加重要, 所以奇异值矩阵可视为矩阵  $\mathbf{A}$  的核心矩阵。若将对角奇异值矩阵  $\boldsymbol{\Sigma}$  看作一个二阶张量, 则奇异值矩阵很自然地是二阶张量  $\mathbf{A}$  的核心张量 (core tensor), 而矩阵  $\mathbf{A}$  的 SVD 的三个矩阵的乘积  $\mathbf{A} = \mathbf{U}_1 \boldsymbol{\Sigma} \mathbf{U}_2^T$  即可改写为张量的  $n$ -模式积  $\mathbf{A} = \boldsymbol{\Sigma} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2$ 。

矩阵的 SVD 的这一  $n$ -模式积很容易推广到  $N$  阶张量或  $N$  维超矩阵  $\mathcal{A} \in \mathbb{C}^{I_1 \times \cdots \times I_N}$  的奇异值分解。

**定理 10.4.1** ( $N$  阶奇异值分解) [298] 每一个  $I_1 \times I_2 \times \cdots \times I_N$  实张量  $\mathcal{X}$  均可以分解为  $n$ -模式积

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} = [\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}] \quad (10.4.2)$$

或

$$x_{i_1 i_2 \cdots i_N} = \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \cdots \sum_{j_N=1}^{J_N} g_{i_1 i_2 \cdots i_N} u_{i_1 j_1}^{(1)} u_{i_2 j_2}^{(2)} \cdots u_{i_N j_N}^{(N)} \quad (10.4.3)$$

其中

(1)  $\mathbf{U}^{(n)} = [\mathbf{u}_1^{(n)}, \dots, \mathbf{u}_{J_n}^{(n)}]$  是一个  $I_n \times J_n$  半正交矩阵, 即  $\mathbf{U}^{(n)\mathrm{T}} \mathbf{U}^{(n)} = \mathbf{I}_{J_n}$ , 且  $J_n \leqslant I_n$ 。

(2) 核心张量  $\mathcal{G}$  是一个  $J_1 \times J_2 \times \cdots \times J_N$  张量, 其子张量  $\mathcal{G}_{j_n=\alpha}$  是固定指标  $j_n = \alpha$  不变所得到的张量  $\mathcal{X}$ 。子张量具有以下两个性质:

① 全正交性 (all-orthogonality)  $\alpha \neq \beta$  的两个子核心张量  $\mathcal{G}_{j_n=\alpha}$  和  $\mathcal{G}_{j_n=\beta}$  正交

$$\langle \mathcal{G}_{j_n=\alpha}, \mathcal{G}_{j_n=\beta} \rangle = 0, \quad \forall \alpha \neq \beta, n = 1, \dots, N \quad (10.4.4)$$

② 排序

$$\|\mathcal{G}_{i_n=1}\|_{\mathrm{F}} \geq \|\mathcal{G}_{i_n=2}\|_{\mathrm{F}} \geq \cdots \geq \|\mathcal{G}_{i_n=N}\|_{\mathrm{F}} \quad (10.4.5)$$

注意, 与奇异值矩阵不同, 核心张量  $\mathcal{G}$  不取对角结构, 一般是一个满张量 (full tensor), 即其非对角元素通常也都不等于零 [274]。核心张量  $\mathcal{G} = [g_{j_1 \cdots j_N}]$  的元素  $g_{j_1 \cdots j_N}$  可以保证各个模式矩阵  $\mathbf{U}^{(n)}, n = 1, \dots, N$  之间的相互作用。

模式- $n$  矩阵  $\mathbf{U}^{(n)}$  要求具有与 SVD 的左奇异矩阵  $\mathbf{U}$  和右奇异矩阵  $\mathbf{V}$  类似的正交列结构, 即  $\mathbf{U}^{(n)}$  的任何两个列都是相互正交的。

由于式 (10.4.2) 是 SVD 在高阶张量下推广的分解形式, 所以很自然地称式 (10.4.2) 为张量的高阶 SVD。高阶 SVD 这一术语是 Lathauwer 等人于 2000 年提出的 [298]。

早在 20 世纪 60 年代, Tucker 就针对三阶张量提出了因子分解 [485, 486, 487], 现习惯称为 Tucker 分解。与 Tucker 分解相比较, 高阶 SVD 更紧密地反映了与矩阵 SVD 之间的联系与推广。不过, 现在人们往往对 Tucker 分解和高阶 SVD 不加区分, 混同使用。

**命题 10.4.2**  $N$  阶张量  $\mathcal{X} \in \mathbb{C}^{I_1 \times \cdots \times I_N}$  的 Tucker 分解或高阶 SVD 存在转换关系

$$\begin{aligned} \mathcal{X} &= \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} \\ &\Rightarrow \mathcal{G} = \mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \times_2 \mathbf{U}^{(2)\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\mathrm{T}} \end{aligned} \quad (10.4.6)$$

其中  $\mathcal{G} \in \mathbb{C}^{J_1 \times \cdots \times J_N}$  和  $\mathbf{U}^{(n)} \in \mathbb{C}^{I_n \times J_n}$ , 并且  $J_n \leqslant I_n$ 。

**证明** 连续使用张量与矩阵的  $n$ -模式积的性质  $\mathcal{X} \times_m \mathbf{A} \times_n \mathbf{B} = \mathcal{X} \times_n \mathbf{B} \times_m \mathbf{A}$  知, 式 (10.4.2) 可以等价写作  $\mathcal{X} = \mathcal{G} \times_N \mathbf{U}^{(N)} \times_{N-1} \mathbf{U}^{(N-1)} \cdots \times_1 \mathbf{U}^{(1)}$ 。利用  $n$ -模式积的性

质  $\mathcal{X} \times_n \mathbf{A} \times_n \mathbf{B} = \mathcal{X} \times_n (\mathbf{B}\mathbf{A})$ , 并注意到  $\mathbf{U}^{(1)\mathrm{T}}\mathbf{U}^{(1)} = \mathbf{I}_{J_1}$ , 有

$$\begin{aligned}\mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} &= \mathcal{G} \times_N \mathbf{U}^{(N)} \times_{N-1} \mathbf{U}^{(N-1)} \cdots \times_1 \mathbf{U}^{(1)} \times_1 \mathbf{U}^{(1)\mathrm{T}} \\ &= \mathcal{G} \times_N \mathbf{U}^{(N)} \times_{N-1} \mathbf{U}^{(N-1)} \cdots \times_2 \mathbf{U}^{(2)} \times_1 (\mathbf{U}^{(1)\mathrm{T}}\mathbf{U}^{(1)}) \\ &= \mathcal{G} \times_N \mathbf{U}^{(N)} \times_{N-1} \mathbf{U}^{(N-1)} \cdots \times_2 \mathbf{U}^{(2)}\end{aligned}$$

仿此, 又有  $\mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \times_2 \mathbf{U}^{(2)\mathrm{T}} = \mathcal{G} \times_N \mathbf{U}^{(N)} \times_{N-1} \mathbf{U}^{(N-1)} \cdots \times_3 \mathbf{U}^{(3)}$ 。以此类推, 易知  $\mathcal{G} = \mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \times_2 \mathbf{U}^{(2)\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\mathrm{T}}$ 。

高阶 SVD 式 (10.4.2) 的矩阵等价表示式与张量的矩阵化方法密切相关:

(1) 与 Kiers 矩阵化对应的高阶 SVD 的矩阵等价表示

$$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left( \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \right)^{\mathrm{T}} \quad (10.4.7)$$

$$\mathbf{X}^{(n)} = \left( \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)\mathrm{T}} \quad (10.4.8)$$

(2) 与 LMV 矩阵化对应的高阶 SVD 的矩阵等价表示

$$\mathbf{X}_{(n)} = \mathbf{U}_n \mathbf{G}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \cdots \otimes \mathbf{U}^{(n-1)} \right)^{\mathrm{T}} \quad (10.4.9)$$

$$\mathbf{X}^{(n)} = \left( \mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \cdots \otimes \mathbf{U}^{(n-1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)\mathrm{T}} \quad (10.4.10)$$

(3) 与 Kolda 矩阵化对应的高阶 SVD 的矩阵等价表示

$$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left( \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \right)^{\mathrm{T}} \quad (10.4.11)$$

$$\mathbf{X}^{(n)} = \left( \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)\mathrm{T}} \quad (10.4.12)$$

在运用上述矩阵等价表示公式时, 应该遵循两个基本原则: 不得出现下标等于零或小于零的因子矩阵  $\mathbf{U}_k, k \leq 0$ ; 相同下标的因子矩阵只取 1 次。

表 10.4.1 汇总了 Tucker 分解的各种数学表示形式。

#### 10.4.2 三阶奇异值分解

特别地, 对于三阶张量  $\mathcal{X} \in \mathbb{K}^{I \times J \times K}$  的 Tucker 分解 (三阶奇异值分解)

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \quad (10.4.13)$$

模式- $n$  矩阵和张量的纵向展开有下列性质 [303]:

(1) 模式-1 矩阵  $\mathbf{A} \in \mathbb{K}^{I \times P}$ 、模式-2 矩阵  $\mathbf{B} \in \mathbb{K}^{J \times Q}$  和模式-3 矩阵  $\mathbf{C} \in \mathbb{K}^{K \times R}$  全部是列向量形式标准正交的 (columnwise orthonormal), 即有

$$\mathbf{A}^{\mathrm{H}} \mathbf{A} = \mathbf{I}_P, \quad \mathbf{B}^{\mathrm{H}} \mathbf{B} = \mathbf{I}_Q, \quad \mathbf{C}^{\mathrm{H}} \mathbf{C} = \mathbf{I}_R$$

表 10.4.1 Tucker 分解的数学表示形式

表示形式	数学公式
算子形式	$\mathcal{X} = [\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}]$
模式-n 积	$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}$
元素形式	$x_{i_1 \cdots i_N} = \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \cdots \sum_{j_N=1}^{J_N} g_{j_1 j_2 \cdots j_N} u_{i_1, j_1}^{(1)} u_{i_2, j_2}^{(2)} \cdots u_{i_N, j_N}^{(N)}$
外积表示	$\mathcal{X} = \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \cdots \sum_{j_N=1}^{J_N} g_{j_1 j_2 \cdots j_N} u_{j_1}^{(1)} \circ u_{j_2}^{(2)} \circ \cdots \circ u_{j_N}^{(N)}$
Kiers 矩阵化	$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left( \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \right)^T$ $\mathbf{X}^{(n)} = \left( \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)T}$
LMV 矩阵化	$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left( \mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \cdots \otimes \mathbf{U}^{(n-1)} \right)^T$ $\mathbf{X}^{(n)} = \left( \mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \cdots \otimes \mathbf{U}^{(n-1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)T}$
Kolda 矩阵化	$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left( \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \right)^T$ $\mathbf{X}^{(n)} = \left( \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \right) \mathbf{G}^{(n)} \mathbf{U}^{(n)T}$

(2) 三阶核心张量  $\mathcal{G} \in \mathbb{K}^{P \times Q \times R}$  的纵向展开  $\mathbf{G}^{(QR \times P)}, \mathbf{G}^{(RP \times Q)}, \mathbf{G}^{(PQ \times R)}$  分别是列正交的

$$\langle \mathbf{G}_{:p_1}^{(QR \times P)}, \mathbf{G}_{:p_2}^{(QR \times P)} \rangle = \sigma_1^2(p_1) \delta_{p_1, p_2}, \quad 1 \leq p_1, p_2 \leq P$$

$$\langle \mathbf{G}_{:q_1}^{(RP \times Q)}, \mathbf{G}_{:q_2}^{(RP \times Q)} \rangle = \sigma_2^2(q_1) \delta_{q_1, q_2}, \quad 1 \leq q_1, q_2 \leq Q$$

$$\langle \mathbf{G}_{:r_1}^{(PQ \times R)}, \mathbf{G}_{:r_2}^{(PQ \times R)} \rangle = \sigma_3^2(r_1) \delta_{r_1, r_2}, \quad 1 \leq r_1, r_2 \leq R$$

其中，奇异值的排序如下

$$\begin{aligned} \sigma_1^2(1) &\geq \sigma_1^2(2) \geq \cdots \geq \sigma_1^2(P) \\ \sigma_2^2(1) &\geq \sigma_2^2(2) \geq \cdots \geq \sigma_2^2(Q) \\ \sigma_3^2(1) &\geq \sigma_3^2(2) \geq \cdots \geq \sigma_3^2(R) \end{aligned}$$

三阶张量  $\mathcal{X} = [x_{ijk}] \in \mathbb{R}^{I \times J \times K}$  的 SVD 共有三个下标变量  $i, j, k$ , 其全排列  $P_3 = 6$  种可能的水平展开矩阵表示

$$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A} \mathbf{G}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T \quad (10.4.14)$$

$$x_{j,(i-1)K+k}^{(J \times KI)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(J \times KI)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}] = \mathbf{B} \mathbf{G}^{(Q \times RP)} (\mathbf{A} \otimes \mathbf{C})^T \quad (10.4.15)$$

$$x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{:1::}, \dots, \mathbf{X}_{:J::}] = \mathbf{C} \mathbf{G}^{(R \times PQ)} (\mathbf{B} \otimes \mathbf{A})^T \quad (10.4.16)$$

和

$$x_{i,(j-1)K+k}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{J::}^T] = \mathbf{A}\mathbf{G}^{(P \times QR)}(\mathbf{B} \otimes \mathbf{C})^T \quad (10.4.17)$$

$$x_{j,(k-1)I+i}^{(J \times KI)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(J \times KI)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{K::}^T] = \mathbf{B}\mathbf{G}^{(Q \times RP)}(\mathbf{C} \otimes \mathbf{A})^T \quad (10.4.18)$$

$$x_{k,(i-1)J+j}^{(K \times IJ)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{I::}^T] = \mathbf{C}\mathbf{G}^{(R \times PQ)}(\mathbf{A} \otimes \mathbf{B})^T \quad (10.4.19)$$

以及下列 6 种可能的纵向展开矩阵表示

$$x_{(k-1)J+j,i}^{(JK \times I)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(JK \times I)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}]^T = (\mathbf{C} \otimes \mathbf{B})\mathbf{G}^{(QR \times P)}\mathbf{A}^T \quad (10.4.20)$$

$$x_{(i-1)K+k,j}^{(KI \times J)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(KI \times J)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{J::}]^T = (\mathbf{A} \otimes \mathbf{C})\mathbf{G}^{(RP \times Q)}\mathbf{B}^T \quad (10.4.21)$$

$$x_{(j-1)I+i,k}^{(IJ \times K)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(IJ \times K)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{J::}]^T = (\mathbf{B} \otimes \mathbf{A})\mathbf{G}^{(PQ \times R)}\mathbf{C}^T \quad (10.4.22)$$

$$x_{(j-1)K+k,i}^{(JK \times I)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(JK \times I)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{J::}^T]^T = (\mathbf{B} \otimes \mathbf{C})\mathbf{G}^{(QR \times P)}\mathbf{A}^T \quad (10.4.23)$$

$$x_{(k-1)I+i,j}^{(KI \times J)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(KI \times J)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{K::}^T]^T = (\mathbf{C} \otimes \mathbf{A})\mathbf{G}^{(RP \times Q)}\mathbf{B}^T \quad (10.4.24)$$

$$x_{(i-1)J+j,k}^{(IJ \times K)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(IJ \times K)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{I::}^T]^T = (\mathbf{A} \otimes \mathbf{B})\mathbf{G}^{(PQ \times R)}\mathbf{C}^T \quad (10.4.25)$$

下面是三阶 SVD 的矩阵化等价表示的三种方法。

(1) 与 Kiers 矩阵化对应的三阶 SVD 的矩阵化等价表示 [267, 501]

$$\begin{array}{ll} \text{水平展开} & \left\{ \begin{array}{l} x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(I \times JK)} = \mathbf{A}\mathbf{G}_{\text{Kiers}}^{(P \times QR)}(\mathbf{C} \otimes \mathbf{B})^T \\ x_{j,(i-1)K+k}^{(J \times KI)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(J \times KI)} = \mathbf{B}\mathbf{G}_{\text{Kiers}}^{(Q \times RP)}(\mathbf{A} \otimes \mathbf{C})^T \\ x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(K \times IJ)} = \mathbf{C}\mathbf{G}_{\text{Kiers}}^{(R \times PQ)}(\mathbf{B} \otimes \mathbf{A})^T \end{array} \right. \\ \text{纵向展开} & \left\{ \begin{array}{l} x_{(k-1)J+j,i}^{(JK \times I)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(JK \times I)} = (\mathbf{C} \otimes \mathbf{B})\mathbf{G}_{\text{Kiers}}^{(QR \times P)}\mathbf{A}^T \\ x_{(i-1)K+k,j}^{(KI \times J)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(KI \times J)} = (\mathbf{A} \otimes \mathbf{C})\mathbf{G}_{\text{Kiers}}^{(RP \times Q)}\mathbf{B}^T \\ x_{(j-1)I+i,k}^{(IJ \times K)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kiers}}^{(IJ \times K)} = (\mathbf{B} \otimes \mathbf{A})\mathbf{G}_{\text{Kiers}}^{(PQ \times R)}\mathbf{C}^T \end{array} \right. \end{array}$$

(2) 与 LMV 矩阵化对应的三阶 SVD 的矩阵化等价表示 [298, 303]

$$\begin{array}{ll} \text{水平展开} & \left\{ \begin{array}{l} x_{i,(j-1)K+k}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(I \times JK)} = \mathbf{A}\mathbf{G}_{\text{LMV}}^{(P \times QR)}(\mathbf{B} \otimes \mathbf{C})^T \\ x_{j,(k-1)I+i}^{(J \times KI)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(J \times KI)} = \mathbf{B}\mathbf{G}_{\text{LMV}}^{(Q \times RP)}(\mathbf{C} \otimes \mathbf{A})^T \\ x_{k,(i-1)J+j}^{(K \times IJ)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(K \times IJ)} = \mathbf{C}\mathbf{G}_{\text{LMV}}^{(R \times PQ)}(\mathbf{A} \otimes \mathbf{B})^T \end{array} \right. \\ \text{纵向展开} & \left\{ \begin{array}{l} x_{(j-1)K+k,i}^{(JK \times I)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(JK \times I)} = (\mathbf{B} \otimes \mathbf{C})\mathbf{G}_{\text{LMV}}^{(QR \times P)}\mathbf{A}^T \\ x_{(k-1)I+i,j}^{(KI \times J)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(KI \times J)} = (\mathbf{C} \otimes \mathbf{A})\mathbf{G}_{\text{LMV}}^{(RP \times Q)}\mathbf{B}^T \\ x_{(i-1)J+j,k}^{(IJ \times K)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{LMV}}^{(IJ \times K)} = (\mathbf{A} \otimes \mathbf{B})\mathbf{G}_{\text{LMV}}^{(PQ \times R)}\mathbf{C}^T \end{array} \right. \end{array}$$

(3) 与 Kolda 矩阵化对应的三阶 SVD 的矩阵化等价表示 [66, 267, 176, 276, 9]

$$\begin{array}{ll} \text{水平展开} & \left\{ \begin{array}{l} x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(I \times JK)} = \mathbf{A} \mathbf{G}_{\text{Kolda}}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T \\ x_{j,(k-1)I+i}^{(J \times KI)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(J \times KI)} = \mathbf{B} \mathbf{G}_{\text{Kolda}}^{(Q \times RP)} (\mathbf{C} \otimes \mathbf{A})^T \\ x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(K \times IJ)} = \mathbf{C} \mathbf{G}_{\text{Kolda}}^{(R \times PQ)} (\mathbf{B} \otimes \mathbf{A})^T \end{array} \right. \\ \text{纵向展开} & \left\{ \begin{array}{l} x_{(k-1)J+j,i}^{(JK \times I)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(JK \times I)} = (\mathbf{C} \otimes \mathbf{B}) \mathbf{G}_{\text{Kolda}}^{(QR \times P)} \mathbf{A}^T \\ x_{(k-1)I+i,j}^{(KI \times J)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(KI \times J)} = (\mathbf{C} \otimes \mathbf{A}) \mathbf{G}_{\text{Kolda}}^{(RP \times Q)} \mathbf{B}^T \\ x_{(j-1)I+i,k}^{(IJ \times K)} = x_{ijk} \Leftrightarrow \mathbf{X}_{\text{Kolda}}^{(IJ \times K)} = (\mathbf{B} \otimes \mathbf{A}) \mathbf{G}_{\text{Kolda}}^{(PQ \times R)} \mathbf{C}^T \end{array} \right. \end{array}$$

这里给出  $\mathbf{X}_{\text{Kiers}}^{(J \times KI)} = \mathbf{B} \mathbf{G}_{\text{Kiers}}^{(Q \times RP)} (\mathbf{A} \otimes \mathbf{C})^T$  的证明, 其他各种形式可类似证明。首先, Tucker 分解可写成水平展开形式

$$\begin{aligned} \mathbf{X}_{i::} &= \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} a_{ip} \mathbf{B}_{:q} \mathbf{C}_{:r}^T \\ &= \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R \mathbf{G}_{q,(p-1)R+r}^{(Q \times RP)} a_{ip} b_q c_r^T \\ &= \sum_{p=1}^P \sum_{r=1}^R [\mathbf{b}_1, \dots, \mathbf{b}_Q] \begin{bmatrix} \mathbf{G}_{1,(p-1)R+r}^{(Q \times RP)} \\ \vdots \\ \mathbf{G}_{Q,(p-1)R+r}^{(Q \times RP)} \end{bmatrix} a_{ip} \mathbf{c}_r^T \\ &= \sum_{p=1}^P \sum_{r=1}^R \mathbf{B} \mathbf{G}_{:(p-1)R+r}^{(Q \times RP)} a_{ip} \mathbf{c}_r^T \end{aligned}$$

展开后, 得

$$\mathbf{X}_{i::} = \mathbf{B} \left[ \mathbf{G}_{:1}^{(Q \times RP)}, \dots, \mathbf{G}_{:R}^{(Q \times RP)}, \dots, \mathbf{G}_{:(P-1)R+1}^{(Q \times RP)}, \dots, \mathbf{G}_{:RP}^{(Q \times RP)} \right] \begin{bmatrix} a_{i1} \mathbf{c}_1^T \\ \vdots \\ a_{i1} \mathbf{c}_R^T \\ \vdots \\ a_{iP} \mathbf{c}_1^T \\ \vdots \\ a_{iP} \mathbf{c}_R^T \end{bmatrix}$$

于是, 由  $\mathbf{X}_{\text{Kiers}}^{(J \times KI)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}]$  得

$$\begin{aligned} \mathbf{X}_{\text{Kiers}}^{(J \times KI)} &= \mathbf{B} \mathbf{G}_{\text{Kiers}}^{(Q \times RP)} \begin{bmatrix} a_{11} \mathbf{c}_1^T & \cdots & a_{11} \mathbf{c}_1^T \\ \vdots & \ddots & \vdots \\ a_{11} \mathbf{c}_R^T & \cdots & a_{11} \mathbf{c}_R^T \\ \vdots & \ddots & \vdots \\ a_{1P} \mathbf{c}_1^T & \cdots & a_{1P} \mathbf{c}_1^T \\ \vdots & \ddots & \vdots \\ a_{1P} \mathbf{c}_R^T & \cdots & a_{1P} \mathbf{c}_R^T \end{bmatrix} = \mathbf{B} \mathbf{G}_{\text{Kiers}}^{(Q \times RP)} \begin{bmatrix} \mathbf{a}_1^T \mathbf{c}_1^T \\ \vdots \\ \mathbf{a}_1^T \mathbf{c}_R^T \\ \vdots \\ \mathbf{a}_P^T \mathbf{c}_1^T \\ \vdots \\ \mathbf{a}_P^T \mathbf{c}_R^T \end{bmatrix} \\ &= \mathbf{B} \mathbf{G}_{\text{Kiers}}^{(Q \times RP)} (\mathbf{A} \otimes \mathbf{C})^T \end{aligned}$$

以上 Tucker 分解形式习惯称为 Tucker3 分解。

Tucker3 分解有以下两种简化形式

$$\text{Tucker2 } x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q g_{pqk} a_{ip} b_{jq} + e_{ijk} \quad (10.4.26)$$

$$\text{Tucker1 } x_{ijk} = \sum_{p=1}^P g_{pj} a_{ip} + e_{ijk} \quad (10.4.27)$$

与 Tucker3 分解相比, 在 Tucker2 分解中  $C = I_K$  和  $G \in \mathbb{R}^{P \times Q \times K}$ ; 在 Tucker1 分解中  $B = I_J$ ,  $C = I_K$  和  $G \in \mathbb{R}^{P \times J \times K}$ 。

### 10.4.3 高阶奇异值分解的交替最小二乘算法

$N$  阶张量的 Tucker 分解或者后面将介绍的典范/平行因子分解可以写成一个统一的数学模型

$$\mathcal{X} = f(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}) + \mathcal{E} \quad (10.4.28)$$

式中  $\mathbf{U}^{(n)}$ ,  $n = 1, \dots, N$  为分解的因子或分量矩阵,  $\mathcal{E}$  为  $N$  阶噪声或误差张量。因此, 因子矩阵可以通过下列优化问题求得

$$(\hat{\mathbf{U}}^{(1)}, \dots, \hat{\mathbf{U}}^{(N)}) = \arg \min_{\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}} \|\mathcal{X} - f(\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)})\|_2^2 \quad (10.4.29)$$

这是一个  $N$  个变元耦合在一起的优化问题。正如第 6 章指出的, 求解这类耦合优化问题的有效方法是交替最小二乘 (ALS) 算法。

Tucker 分解的交替最小二乘算法的基本思想是: 在第  $k+1$  次迭代中, 利用在  $k+1$  次迭代中已更新的因子矩阵  $\mathbf{U}_{k+1}^{(1)}, \dots, \mathbf{U}_{k+1}^{(i-1)}$  和在  $k$  次更新过的因子矩阵  $\mathbf{U}_k^{(i+1)}, \dots, \mathbf{U}_k^{(N)}$ , 求因子矩阵  $\mathbf{U}^{(1)}$  的最小二乘解

$$\hat{\mathbf{U}}_{k+1}^{(i)} = \arg \min_{\mathbf{U}^{(i)}} \|\mathcal{X} - f(\mathbf{U}_{k+1}^{(1)}, \dots, \mathbf{U}_{k+1}^{(i-1)}, \mathbf{U}_k^{(i)}, \mathbf{U}_k^{(i+1)}, \dots, \mathbf{U}_k^{(N)})\|_2^2 \quad (10.4.30)$$

其中  $i = 1, \dots, N$ 。对  $k = 1, 2, \dots$ , 交替使用最小二乘法, 直至所有因子矩阵收敛。

下面以张量的矩阵化的水平展开为对象, 讨论 Tucker3 分解的优化问题的求解

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{G}^{(P \times QR)}} \left\| \mathbf{X}^{(I \times JK)} - \mathbf{AG}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T \right\|_2^2 \quad (10.4.31)$$

根据交替最小二乘的原理, 假定模式-2 矩阵  $B$ 、模式-3 矩阵  $C$  和核心张量  $G$  的水平展开均固定, 则上述优化问题就解偶为仅含模式-1 矩阵  $A$  的优化问题

$$\min_{\mathbf{A}} \left\| \mathbf{X}^{(I \times JK)} - \mathbf{AG}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T \right\|_2^2$$

相当于求解矩阵方程  $\mathbf{X}^{(I \times JK)} = \mathbf{AG}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T$  的最小二乘解。在矩阵方程的两边右乘矩阵  $(\mathbf{C} \otimes \mathbf{B})$ , 得

$$\mathbf{X}^{(I \times JK)} (\mathbf{C} \otimes \mathbf{B}) = \mathbf{AG}^{(P \times QR)} (\mathbf{C} \otimes \mathbf{B})^T (\mathbf{C} \otimes \mathbf{B}) \quad (10.4.32)$$

若对上式左边的矩阵进行奇异值分解  $\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}) = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^T$ , 则可取前  $P$  个左奇异向量作为矩阵  $\mathbf{A}$  的估计结果  $\hat{\mathbf{A}} = \mathbf{U}_1(:, 1:P)$ 。这一运算可以简洁表示为  $[\mathbf{A}, \mathbf{S}, \mathbf{T}] = \text{SVD}[\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}), P]$ 。

类似地, 可以分别求出  $\mathbf{B}, \mathbf{C}$  的估计。然后, 固定已经求出的因子矩阵, 又可返回再次求解  $\mathbf{A}$ , 并且依次再计算  $\mathbf{B}, \mathbf{C}$ , 直至因子矩阵  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  全部收敛。

当因子矩阵全部收敛, 并且满足正交条件  $\mathbf{A}^T \mathbf{A} = \mathbf{I}_P, \mathbf{B}^T \mathbf{B} = \mathbf{I}_Q, \mathbf{C}^T \mathbf{C} = \mathbf{I}_R$  时, 由于  $(\mathbf{C}^T \otimes \mathbf{B}^T)(\mathbf{C} \otimes \mathbf{B}) = (\mathbf{C}^T \mathbf{C}) \otimes (\mathbf{B}^T \mathbf{B}) = \mathbf{I}_R \otimes \mathbf{I}_Q = \mathbf{I}_{QR}$ , 故式 (10.4.32) 两边左乘因子矩阵  $\mathbf{A}^T$  后, 立即得

$$\mathbf{G}^{(P \times QR)} = \mathbf{A}^T \mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}) \quad (10.4.33)$$

因为  $\mathbf{A}^T \mathbf{A} = \mathbf{I}_P$ 。类似地, 可以求出核心张量的其他两个水平展开矩阵  $\mathbf{G}^{(Q \times RP)}$  和  $\mathbf{G}^{(R \times PQ)}$ , 从而得到核心张量。

以上讨论可以总结得出下面的交替最小二乘算法。

#### 算法 10.4.1 Tucker 分解的交替最小二乘算法 [9]

输入 三阶张量  $\mathcal{X}$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I \times P}, \mathbf{B} \in \mathbb{R}^{J \times Q}, \mathbf{C} \in \mathbb{R}^{K \times R}$  和核心张量  $\mathcal{G}$ 。

初始化 矩阵  $\mathbf{B}$  和  $\mathbf{C}$ , 并令  $k = 0$ 。

步骤 1 令  $k = k + 1$ , 通过  $x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk}$  构造水平展开矩阵  $\mathbf{X}^{(I \times JK)}$ , 然后计算  $\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B})$  的奇异值分解, 并取前  $P$  个左奇异向量  $\mathbf{U}_1(:, j), j = 1, \dots, P$  构成因子矩阵  $\mathbf{A}$

$$[\mathbf{A}, \mathbf{S}_1, \mathbf{T}_1] = \text{SVD}[\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}), P]$$

步骤 2 计算因子矩阵

$$[\mathbf{B}, \mathbf{S}_2, \mathbf{T}_2] = \text{SVD}[\mathbf{X}^{(J \times KI)}(\mathbf{A} \otimes \mathbf{C}), Q]$$

$$[\mathbf{C}, \mathbf{S}_3, \mathbf{T}_3] = \text{SVD}[\mathbf{X}^{(K \times IJ)}(\mathbf{B} \otimes \mathbf{A}), R]$$

步骤 3 若收敛准则不满足, 则返回步骤 1, 并重复以上计算, 直至收敛准则满足。若收敛, 则计算核心张量  $\mathcal{G}$  的三个模式的水平展开矩阵

$$\mathbf{G}^{(P \times QR)} = \mathbf{A}^T \mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B})$$

$$\mathbf{G}^{(Q \times RP)} = \mathbf{B}^T \mathbf{X}^{(J \times KI)}(\mathbf{A} \otimes \mathbf{C})$$

$$\mathbf{G}^{(R \times PQ)} = \mathbf{C}^T \mathbf{X}^{(K \times IJ)}(\mathbf{B} \otimes \mathbf{A})$$

从而得到三阶核心张量  $\mathcal{G} \in \mathbb{R}^{(P \times Q \times R)}$ 。

注释: 以上算法适用于三阶张量的 Kiers 水平展开, 但很容易推广到其他矩阵化情况。例如, 只需要将步骤 1 和步骤 2 的 Kiers 水平展开分别替换为 Kolda 水平展开

$$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk}, \quad x_{j,(k-1)I+i}^{(J \times KI)} = x_{ijk}, \quad x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk}$$

和

$$\begin{aligned} [\mathbf{A}, \mathbf{S}, \mathbf{V}] &= \text{SVD}(\mathbf{X}^{(I \times JK)}(\mathbf{C} \otimes \mathbf{B}), P) \\ [\mathbf{B}, \mathbf{S}, \mathbf{V}] &= \text{SVD}(\mathbf{X}^{(J \times KI)}(\mathbf{C} \otimes \mathbf{A}), Q) \\ [\mathbf{C}, \mathbf{S}, \mathbf{V}] &= \text{SVD}(\mathbf{X}^{(K \times IJ)}(\mathbf{B} \otimes \mathbf{A}), R) \end{aligned}$$

则得到的是 Bro 于 1998 年提出的 Tucker 分解的交替最小二乘算法 [66]。因此，高阶奇异值分解的矩阵等价形式必须与张量的矩阵化方法严格对应。

考虑  $N$  阶张量  $\mathcal{X}$  的 Tucker3 分解即高阶奇异值分解

$$\min_{\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}, \mathbf{G}_{(n)}} \left\| \mathbf{X}_{(n)} - \mathbf{U}^{(n)} \mathbf{G}_{(n)} \mathbf{U}_{\otimes}^{(n)} \right\|_2^2 \quad (10.4.34)$$

式中

$$\mathbf{U}_{\otimes}^{(n)} = \begin{cases} \left( \mathbf{U}^{(n-1)} \otimes \dots \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(n+1)} \right)^T, & \mathbf{X}_{(n)} \text{ 由式 (10.2.3) 确定} \\ \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(n-1)} \right)^T, & \mathbf{X}_{(n)} \text{ 由式 (10.2.5) 确定} \\ \left( \mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \mathbf{U}^{(1)} \right)^T, & \mathbf{X}_{(n)} \text{ 由式 (10.2.6) 确定} \end{cases} \quad (10.4.35)$$

是除  $\mathbf{U}^{(n)}$  以外的其他  $N-1$  个因子矩阵的 Kronecker 积，这里视其为中间矩阵 (intermediate matrix)。

由于在高阶 SVD 中，因子矩阵  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times J_n}$  满足半正交条件  $\mathbf{U}^{(n)T} \mathbf{U}^{(n)} = \mathbf{I}_{J_n}$ ，故有

$$\mathbf{U}_{\otimes}^{(n)} \mathbf{U}_{\otimes}^{(n)T} = \mathbf{I}_{J_1 \dots J_{n-1} J_{n+1} \dots J_N} \quad (10.4.36)$$

以  $\mathbf{U}_{\otimes}^{(n)} = \left( \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(n-1)} \right)^T$  为例，证明如下：利用 Kronecker 积的性质  $(\mathbf{A} \otimes \mathbf{C})(\mathbf{B} \otimes \mathbf{D}) = (\mathbf{AB}) \otimes (\mathbf{CD})$  以及  $\mathbf{U}^{(k)T} \mathbf{U}^{(k)} = \mathbf{I}_{J_k}$ ,  $k = 1, \dots, N$ ，易知

$$\begin{aligned} \mathbf{U}_{\otimes}^{(n)} \mathbf{U}_{\otimes}^{(n)T} &= (\mathbf{U}^{(n+1)T} \otimes \dots \otimes \mathbf{U}^{(N)T} \otimes \mathbf{U}^{(1)T} \otimes \mathbf{U}^{(n-1)T}) \\ &\quad \times (\mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(n-1)}) \\ &= \mathbf{I}_{J_{n+1}} \otimes [(\mathbf{U}^{(n+2)T} \otimes \dots \otimes \mathbf{U}^{(N)T} \otimes \mathbf{U}^{(1)T} \otimes \mathbf{U}^{(n-1)T}) \\ &\quad \times (\mathbf{U}^{(n+2)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(n-1)})] \\ &= \mathbf{I}_{J_{n+1}} \otimes \mathbf{I}_{J_{n+2}} [(\mathbf{U}^{(n+3)T} \otimes \dots \otimes \mathbf{U}^{(N)T} \otimes \mathbf{U}^{(1)T} \otimes \mathbf{U}^{(n-1)T}) \\ &\quad \times (\mathbf{U}^{(n+3)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(n-1)})] \end{aligned}$$

持续以上类似过程，易知

$$\mathbf{U}_{\otimes}^{(n)} \mathbf{U}_{\otimes}^{(n)T} = \mathbf{I}_{J_{n+1}} \otimes \dots \otimes \mathbf{I}_{J_N} \otimes \mathbf{I}_{J_1} \otimes \dots \otimes \mathbf{I}_{J_{n-1}} = \mathbf{I}_{J_1 \dots J_{n-1} J_{n+1} \dots J_N}$$

为了求解矩阵方程  $\mathbf{X}_{(n)} \approx \mathbf{U}^{(n)} \mathbf{G}_{(n)} \mathbf{U}_{\otimes}^{(n)}$ ，令 SVD  $\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{S}^{(n)} \mathbf{V}^{(n)T}$ ，则  $\mathbf{U}^{(n)} \mathbf{S}^{(n)} \mathbf{V}^{(n)T} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \mathbf{U}_{\otimes}^{(n)}$ ，两边分别左乘  $\mathbf{U}^{(n)T}$  和右乘  $\mathbf{U}_{\otimes}^{(n)T}$ ，则由式 (10.4.36) 易

得

$$\mathbf{G}_{(n)} = \mathbf{S}^{(n)} \mathbf{V}^{(n)\mathrm{T}} \mathbf{U}_{\otimes}^{(n)\mathrm{T}} \quad (10.4.37)$$

**算法 10.4.2 HOSVD ( $\mathcal{X}, R_1, \dots, R_N$ )** [278]

输入  $N$  阶张量  $\mathcal{X}$ 。

输出 因子矩阵  $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}, \mathbf{U}^{(2)} \in \mathbb{R}^{I_2 \times R_2}, \dots, \mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$  和核心张量  $\mathcal{G}$ 。

步骤 1 计算  $N$  阶张量  $\mathcal{X}$  的模式- $n$  水平展开  $\mathbf{X}_{(n)}, n = 1, \dots, N$ 。令  $k = 0$ 。

步骤 2 令  $k = k + 1$ , 并对  $n = 1, \dots, N$ , 计算  $\mathbf{X}_{(n)} = \mathbf{U} \Sigma \mathbf{V}^{\mathrm{T}}$ , 确定其有效秩  $R_n$ , 并令  $\mathbf{U}^{(n)} \leftarrow \mathbf{U}(:, 1 : R_n)$ 。

步骤 3 计算  $\mathcal{G} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \times_2 \mathbf{U}^{(2)\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\mathrm{T}}$ 。

步骤 4 判断核心张量是否收敛

$$\|\mathcal{G}^{(k)} - \mathcal{G}^{(k-1)}\|_{\mathrm{F}} < \varepsilon$$

若收敛条件满足, 则执行下一步; 否则, 返回步骤 2, 继续迭代, 直至收敛准则满足为止。

步骤 5 输出核心张量  $\mathcal{G}$  和因子矩阵  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$ 。

下面是 Lathauwer 提出的高阶正交迭代 (higher-order orthogonal iteration, HOOI) 算法 [299], 这是计算因子矩阵和核心张量的一种比较有效的算法, 其特点是使用 SVD 而不是特征值分解, 只计算张量的水平展开  $\mathbf{X}_{(n)}$  的主要奇异向量。

**算法 10.4.3 HOOI ( $\mathcal{X}, R_1, \dots, R_N$ )** [299, 278]

输入  $N$  阶张量  $\mathcal{X}$ 。

输出 因子矩阵  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}, n = 1, \dots, N$  和核心张量  $\mathcal{G}$ 。

步骤 1 利用 HOSVD 算法计算因子矩阵  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}, n = 1, \dots, N$ 。令  $k = 0$ , 初始化核心张量  $\mathcal{G}^{(0)}$  为零张量 (全部元素等于零)。

步骤 2 令  $k = k + 1$ , 并对  $n = 1, \dots, N$ , 执行下列运算

$$\mathcal{B}^{(k)} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \cdots \times_{n-1} \mathbf{U}^{(n-1)\mathrm{T}} \times_{n+1} \mathbf{U}^{(n+1)\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\mathrm{T}}$$

并计算张量  $\mathcal{B}^{(k)}$  的模式- $n$  水平展开的 SVD:  $\mathbf{B}_{(n)} = \mathbf{U} \Sigma \mathbf{V}^{\mathrm{T}}$ , 确定其主要奇异值个数  $R_n$ , 然后令  $\mathbf{U}^{(n)} \leftarrow \mathbf{U}(:, 1 : R_n)$ 。

步骤 3 计算第  $k$  次迭代的核心张量

$$\mathcal{G}^{(k)} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)\mathrm{T}} \times_2 \mathbf{U}^{(2)\mathrm{T}} \cdots \times_N \mathbf{U}^{(N)\mathrm{T}}$$

判断其是否收敛

$$\|\mathcal{G}^{(k)} - \mathcal{G}^{(k-1)}\|_{\mathrm{F}} < \varepsilon$$

若收敛条件满足, 则执行下一步; 否则, 返回步骤 2, 继续迭代, 直至收敛准则满足为止。

步骤 4 输出因子矩阵  $\mathbf{U}^{(n)}, n = 1, \dots, N$  和核心张量  $\mathcal{G}$ 。

## 10.5 张量的平行因子分解

典范或平行因子分析 (canonical or parallel factor analysis, CANDECOMP/PARAFAC) 是由 Carroll 和 Chang<sup>[93]</sup> 以及 Harshman<sup>[216]</sup> 于 1970 年分别独立提出的数据分析方法，现在习惯合称为 CP 分析。CP 分析的基础是多路数据模型的典范或平行因子分解，简称 CP 分解。

### 10.5.1 双线性模型

张量分析本质上属于多线性分析。多线性分析是双线性分析的推广，而主分量分析与独立分量分析是两种典型的双线性分析方法。

使用双线性或者多线性模型，反映变量线性组合的因子（或称分量、载荷）被提取。这些被提取出来的因子随后用于解释数据的基本信息内容。

在数据分析中，给定二路数据矩阵  $\mathbf{X} \in \mathbb{K}^{I \times J}$ ，二路双线性分析采用模型

$$x_{ij} = \sum_{r=1}^R a_{ir} b_{jr} + e_{ij} \quad (10.5.1)$$

拟合二路数据矩阵的各个元素。式中，各参数的含义如下：

$x_{ij}$  为  $I \times J$  数据矩阵  $\mathbf{X}$  第  $i$  行、第  $j$  列的元素；

$R$  为因子的个数；

$a_{ir}$  为“因子载荷”(factor loadings)；

$b_{jr}$  为“因子得分”(factor score)；

$e_{ij}$  为数据  $x_{ij}$  的观测误差，是  $I \times J$  误差矩阵  $\mathbf{E}$  第  $i$  行、第  $j$  列的元素。

若固定  $b_{jr} = \beta_r$  为常数，则  $x_{ij} = \sum_{r=1}^R \beta_r a_{ir}$  是因子载荷  $a_{ir}$  的线性模型。反之，若固定  $a_{ir} = \alpha_r$  为常数，则  $x_{ij} = \sum_{r=1}^R \alpha_r b_{jr}$  是因子得分  $b_{jr}$  的线性模型。因此，二路数据模型 (10.5.1) 常称为二路双线性模型 (two-way bilinear model)。

定义因子载荷向量和因子得分向量分别为

$$\mathbf{a}_r = [a_{1r}, \dots, a_{Ir}]^T, \quad \mathbf{b}_r = [b_{1r}, \dots, b_{Jr}]^T$$

则二路双线性模型可以用矩阵形式改写为

$$\mathbf{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r = \sum_{r=1}^R \mathbf{a}_r \mathbf{b}_r^T \quad (10.5.2)$$

$$= \begin{bmatrix} a_{11}b_{11} + \dots + a_{1R}b_{1R} & \cdots & a_{11}b_{J1} + \dots + a_{1R}b_{JR} \\ \vdots & \ddots & \vdots \\ a_{I1}b_{11} + \dots + a_{IR}b_{1R} & \cdots & a_{I1}b_{J1} + \dots + a_{IR}b_{JR} \end{bmatrix} \quad (10.5.3)$$

上式又可等价写作

$$\mathbf{X} = \mathbf{AB}^T = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \quad (10.5.4)$$

式中

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1R} \\ \vdots & \ddots & \vdots \\ a_{I1} & \cdots & a_{IR} \end{bmatrix} \in \mathbb{K}^{I \times R}, \quad \mathbf{B} = \begin{bmatrix} b_{11} & \cdots & b_{1R} \\ \vdots & \ddots & \vdots \\ b_{J1} & \cdots & b_{JR} \end{bmatrix} \in \mathbb{K}^{J \times R} \quad (10.5.5)$$

分别是因子载荷矩阵和因子得分矩阵。

数据分析强调结构化模型的唯一性。“一个结构化模型是唯一的”意味着：该模型的辨识无须任何其他约束条件。无约束的二路双线性模型不具有唯一性：由式 (10.5.4) 易知，二路双线性模型存在大量的旋转自由度。这是因为，利用任何一个  $R \times R$  正交矩阵  $\mathbf{Q}$  分别对因子载荷矩阵和因子得分矩阵的转置进行旋转，都不会改变原二路数据矩阵，即

$$\mathbf{X} = \mathbf{AQ}(\mathbf{BQ})^T = \mathbf{AB}^T \quad \text{或} \quad \mathbf{X} = \mathbf{AQ}^T(\mathbf{BQ}^T)^T = \mathbf{AB}^T$$

就是说，若只给定二路数据矩阵  $\mathbf{X} \in \mathbb{R}^{I \times J}$ ，则存在无穷多组解  $(\mathbf{A}, \mathbf{B})$  满足二路双线性模型。因此，为了保证二路双线性模型拟合的唯一性，必须对因子矩阵施加约束条件。

主分量分析 (PCA) 就是对因子矩阵增加正交约束的一种二路双线性分析方法。假定数据矩阵  $\mathbf{X} \in \mathbb{R}^{I \times J}$  有  $R$  个主奇异值，则 PCA 使用截尾的奇异值分解

$$\mathbf{X} = \mathbf{U}_1 \boldsymbol{\Sigma}_1 \mathbf{V}_1^T$$

作为二路数据模型。式中， $\boldsymbol{\Sigma}_1$  是一个  $R \times R$  对角矩阵，其对角元素为  $R$  个主要的奇异值，而  $\mathbf{U}_1 \in \mathbb{R}^{I \times R}$  和  $\mathbf{V}_1 \in \mathbb{R}^{J \times R}$  是分别由与主奇异值对应的前  $R$  个左和右奇异向量组成的矩阵。若令

$$\mathbf{A} = \mathbf{U}_1 \boldsymbol{\Sigma}_1, \quad \mathbf{B} = \mathbf{V}_1 \quad (10.5.6)$$

则 PCA 数据模型可写成正交性约束的二路双线性模型

$$\mathbf{X} = \mathbf{AB}^T \text{ subject to } \mathbf{A}^T \mathbf{A} = \mathbf{D}, \quad \mathbf{B}^T \mathbf{B} = \mathbf{I} \quad (10.5.7)$$

式中， $\mathbf{D}$  为  $R \times R$  对角矩阵， $\mathbf{I}$  为  $R \times R$  单位矩阵。即是说，PCA 要求因子载荷矩阵  $\mathbf{A}$  的各个列向量相互正交，同时要求因子得分矩阵  $\mathbf{B}$  的各个列向量标准正交。

二路双线性分析不适用于多路数据集合的处理。

表 10.5.1 比较了 PCA 与无约束二路双线性分析之间的异同点。

表 10.5.1 PCA 与无约束二路双线性分析的比较

方法	无约束二路双线性分析	PCA 分析
结构化模型	$\mathbf{X} = \mathbf{AB}^T$	$\mathbf{X} = \mathbf{AB}^T$
约束条件	无	$\mathbf{A}^T \mathbf{A} = \mathbf{D}, \mathbf{B}^T \mathbf{B} = \mathbf{I}$
代价函数	$\ \mathbf{X} - \mathbf{AB}^T\ _2^2$	$\ \mathbf{X} - \mathbf{AB}^T\ _2^2$
优化问题	$\min_{\mathbf{A}, \mathbf{B}} \ \mathbf{X} - \mathbf{AB}^T\ _2^2$	$\min_{\mathbf{A}, \mathbf{B}} \ \mathbf{X} - \mathbf{AB}^T\ _2^2 \text{ s.t. } \mathbf{A}^T \mathbf{A} = \mathbf{D}, \mathbf{B}^T \mathbf{B} = \mathbf{I}$

### 10.5.2 平行因子分析

为了将无约束二路双线性分析推广到多路数据集合。需要解决以下两个问题：

- (1) 三个及更多个因子的名称问题；
- (2) 旋转自由度引起的模糊分解问题。

事实上，在某些现代应用（例如化学）中，很难区分哪一个是因子载荷，哪一个为因子得分。另一方面，如果我们要推广得到三路数据的分析方法，就需要增加一个新的因子，此时也很难给第3个因子再命名。一种简单的解决方法是用不同的模式来区分不同的因子。具体而言，称拟合数据矩阵  $\mathbf{X}$  的因子载荷向量  $a_r$  为模式-A 向量，因子得分向量  $b_r$  为模式-B 向量。如果再增加一个向量，则称为模式-C 向量。以此类推，还会有模式-D、模式-E 向量等。

为了克服由旋转自由度引起的模糊分解，Cattell 于 1944 年提出了平行比例配置剖面 (parallel proportional profiles) 原则 [94]：描述两个或者多个二路数据集的相同剖面或载荷向量，只需要配置不同的比例或者权系数，就可以得到无旋转自由度的模型。因此，二路双线性分析表达式 (10.5.1) 很自然地推广为三路张量的平行因子分解

$$x_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr} + e_{ijk} \quad (10.5.8)$$

式中， $x_{ijk}$  是三路阵列或张量  $\mathcal{X} = [x_{ijk}] \in \mathbb{K}^{I \times J \times K}$  的元素，而  $e_{ijk}$  则是加性误差张量  $\mathcal{E} = [e_{ijk}] \in \mathbb{K}^{I \times J \times K}$  的元素。

若定义模式-A、模式-B 和模式-C 向量分别为

$$\mathbf{a}_r = [a_{1r}, \dots, a_{Ir}]^T \in \mathbb{K}^{I \times 1} \quad (10.5.9)$$

$$\mathbf{b}_r = [b_{1r}, \dots, b_{Jr}]^T \in \mathbb{K}^{J \times 1} \quad (10.5.10)$$

$$\mathbf{c}_r = [c_{1r}, \dots, c_{Kr}]^T \in \mathbb{K}^{K \times 1} \quad (10.5.11)$$

则三路张量的因子分解的元素表达式 (10.5.8) 可以等价表示为

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r + \mathcal{E} \quad (10.5.12)$$

它是二路双线性分析的矩阵表达式 (10.5.2) 的三路推广。图 10.5.1 画出了因子分析和平行因子分析的比较。

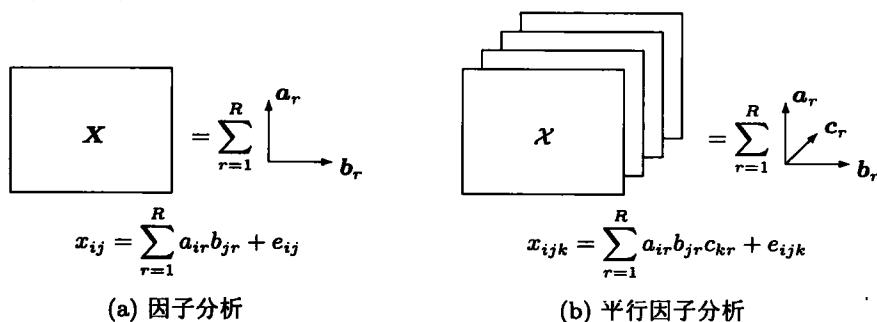


图 10.5.1 因子分析与平行因子分析的比较

图 10.5.2 画出了平行因子分解与 Tucker 分解的比较。

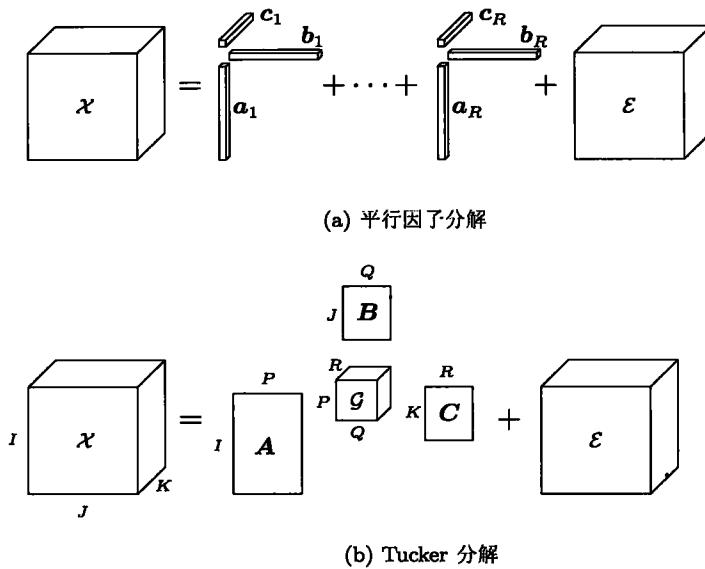


图 10.5.2 平行因子分解与 Tucker 分解的比较

显然,当我们只允许其中一个模式因子(例如模式-A)可以变化,而固定其他两个模式(例如模式-B 和模式-C)因子不变即  $b_{jr} = \alpha_r$  和  $c_{kr} = \beta_r$  时,三路数据模型便简化为模式-A 因子的线性组合。类似地,三路数据模型也可分别视为模式-B(固定模式-A 和模式-C 时)或模式-C(固定模式-A 和模式-B 时)的线性组合,所以三路数据模型式(10.5.8)为三线性因子模型(trilinear factor model)。

表 10.5.2 比较了 Tucker 分解、CP 分解及 SVD 之间的数学公式。

表 10.5.2 Tucker 分解、CP 分解与 SVD 的比较

分解方法	数学公式
Tucker 分解	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} a_{ip} b_{jq} c_{kr}$
平行因子(CP)分解	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R a_{ip} b_{jq} c_{kr}$
SVD	$x_{ij} = \sum_{r=1}^R g_{rr} a_{ir} b_{jr}$

从表中可以看出:

(1) 在 Tucker 分解中,核心张量  $G$  的元素  $g_{pqr}$  表示第 1 模式向量  $a_i = [a_{i1}, \dots, a_{iP}]^T$  的第  $p$  个元素  $a_{ip}$ 、第 2 模式向量  $b_j = [b_{j1}, \dots, b_{jQ}]^T$  的第  $q$  个元素  $b_{jq}$  与第 3 模式向量  $c_k = [c_{k1}, \dots, c_{kR}]^T$  的第  $r$  个元素  $c_{kr}$  之间的相互作用。

(2) 在 CP 分解中, 核心张量为单位张量, 即  $\mathcal{G} = \mathcal{I}$ 。由于只有超对角线  $p = q = r \in \{1, \dots, R\}$  的元素等于 1, 其他元素全部为零, 故第 1 模式向量  $\mathbf{a}_i$  的第  $r$  个因子  $a_{ir}$ 、第 2 模式向量  $\mathbf{b}_j$  的第  $r$  个因子  $b_{jr}$  与第 3 模式向量  $\mathbf{c}_k$  的第  $r$  个因子  $c_{kr}$  之间才存在相互作用。这意味着, 第 1 模式向量、第 2 模式向量和第 3 模式向量具有相同的因子数目  $R$ , 即它们都是  $R \times 1$  向量。也就是说, 在 CP 分解中, 每个模式应该抽取相同数目的因子。

水平展开的 CP 分解存在 6 种可能的形式

$$\mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \quad (10.5.13)$$

$$\mathbf{X}^{(J \times KI)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}] = \mathbf{B}(\mathbf{A} \odot \mathbf{C})^T \quad (10.5.14)$$

$$\mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}] = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T \quad (10.5.15)$$

$$\mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T] = \mathbf{A}(\mathbf{B} \odot \mathbf{C})^T \quad (10.5.16)$$

$$\mathbf{X}^{(J \times KI)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T] = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T \quad (10.5.17)$$

$$\mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::I}^T] = \mathbf{C}(\mathbf{A} \odot \mathbf{B})^T \quad (10.5.18)$$

式中,  $\mathbf{X} \odot \mathbf{Y}$  表示  $m \times n$  矩阵  $\mathbf{X}$  和  $l \times n$  矩阵  $\mathbf{Y}$  的 Khatri-Rao 积, 并且

$$\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_R] = \begin{bmatrix} a_{11} & \cdots & a_{1R} \\ \vdots & \ddots & \vdots \\ a_{I1} & \cdots & a_{IR} \end{bmatrix} \in \mathbb{R}^{I \times R} \quad (10.5.19)$$

$$\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_R] = \begin{bmatrix} b_{11} & \cdots & b_{1R} \\ \vdots & \ddots & \vdots \\ b_{J1} & \cdots & b_{JR} \end{bmatrix} \in \mathbb{R}^{J \times R} \quad (10.5.20)$$

$$\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R] = \begin{bmatrix} c_{11} & \cdots & c_{1R} \\ \vdots & \ddots & \vdots \\ c_{K1} & \cdots & c_{KR} \end{bmatrix} \in \mathbb{R}^{K \times R} \quad (10.5.21)$$

下面证明式 (10.5.13), 其他表达式可类似证明。

根据平行比例配置剖面原则, 考虑三阶张量的正面切片矩阵

$$\mathbf{X}_{::k} = \mathbf{a}_1 \mathbf{b}_1^T \mathbf{c}_{k1} + \cdots + \mathbf{a}_R \mathbf{b}_R^T \mathbf{c}_{kR} \quad (10.5.22)$$

则  $\mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}]$  可写作

$$\begin{aligned} \mathbf{X}^{(I \times JK)} &= [\mathbf{a}_1 \mathbf{b}_1^T \mathbf{c}_{11} + \cdots + \mathbf{a}_R \mathbf{b}_R^T \mathbf{c}_{1R}, \dots, \mathbf{a}_1 \mathbf{b}_1^T \mathbf{c}_{K1} + \cdots + \mathbf{a}_R \mathbf{b}_R^T \mathbf{c}_{KR}] \\ &= [\mathbf{a}_1, \dots, \mathbf{a}_R] \begin{bmatrix} \mathbf{c}_{11} \mathbf{b}_1^T & \cdots & \mathbf{c}_{K1} \mathbf{b}_1^T \\ \vdots & \ddots & \vdots \\ \mathbf{c}_{1R} \mathbf{b}_R^T & \cdots & \mathbf{c}_{KR} \mathbf{b}_R^T \end{bmatrix} \\ &= \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \end{aligned}$$

表 10.5.3 汇总了三阶张量的水平展开矩阵与 CP 分解的数学表示。

表 10.5.3 三阶张量的水平展开矩阵与 CP 分解表示

矩阵化方法	水平展开矩阵与 CP 分解表示	
Kiers 方法	$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \iff \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T$	
	$x_{j,(i-1)K+k}^{(J \times KI)} = x_{ijk} \iff \mathbf{X}^{(J \times KI)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{J::}] = \mathbf{B}(\mathbf{A} \odot \mathbf{C})^T$	
	$x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \iff \mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}] = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$	
LMV 方法	$x_{i,(k-1)K+j}^{(I \times JK)} = x_{ijk} \iff \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T] = \mathbf{A}(\mathbf{B} \odot \mathbf{C})^T$	
	$x_{j,(i-1)I+i}^{(J \times KI)} = x_{ijk} \iff \mathbf{X}^{(J \times KI)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T] = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T$	
	$x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \iff \mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T] = \mathbf{C}(\mathbf{A} \odot \mathbf{B})^T$	
Kolda 方法	$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \iff \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T$	
	$x_{j,(k-1)I+i}^{(J \times KI)} = x_{ijk} \iff \mathbf{X}^{(J \times KI)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T] = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T$	
	$x_{k,(j-1)I+i}^{(K \times IJ)} = x_{ijk} \iff \mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}] = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$	

如果对式 (10.5.13) 运用主分量分析 (PCA) 或独立分量分析 (ICA)，则需要使用  $R$  个主奇异值截尾的 SVD

$$\mathbf{A}^{(I \times JK)} = \sum_{r=1}^R \sigma_r \mathbf{u}_r \mathbf{v}_r^H \quad (10.5.23)$$

这将涉及  $R(I + JK + 1)$  个参数，因为  $\mathbf{u}_r \in \mathbb{K}^{I \times 1}$ ,  $\mathbf{v}_r \in \mathbb{K}^{JK \times 1}$ 。然而，基于式 (10.5.12) 的 CP 方法只需要  $R(I + J + K)$  个参数。因此，与 CP 分析方法相比较，PCA 和 ICA 方法需要大得多的自由参数个数，因为  $R(I + JK + 1) \gg R(I + J + K)$ 。自由参数少是 CP 方法的突出优点之一。

虽然大多数的多路分析技术能够保持数据的多路性质，不过有些简化的多路分析技术（例如 Tucker1），它们基于多路阵列的矩阵化，将三阶或者高阶阵列变换为一个二路数据集。一旦一个三路阵列被展平和排列成一个二路数据集，二路分析方法（例如 SVD, PCA, ICA）就可以应用于提取数据的结构。

然而，将多路阵列作为二路数据集进行重新排列，有可能导致信息的损失和错误解释。如果数据被噪声污染，则这一现象会更加严重。一个典型的例子为感官数据集，其中 8 名评委根据 11 类属性评价 10 种面包 [66]。当这一数据集利用 PARAFAC 模型建模时，该模型假定评委之间存在一个共同的评价指南，并且每个评委在不同程度上都遵守这一评价指南。另外，当感官数据展平为一个二路阵列，并使用二路因子模型建模时，就不再有共同的评价指南存在，每个评委可以完全自主地进行评判。在这样的情况下，为了解释数据的变化，二路因子模型需要抽取尽可能多的因子。相对于 PARAFAC 模型只解释服从基本假设的数据变化，由二路因子模型捕捉的额外的变化实际上只是反映噪声的作用，而不是某种内在的结构。因此，多路模型在解释性和精确度方面比二路模型更为优越。一个重要的事实是：多线性模型（如 PARAFAC、Tucker 分解以及它们的变型）能够

捕捉到数据中的多线性结构，而双线性模型（如 SVD, PCA 与 ICA 等）却不能。

与水平展开的 CP 分解存在 6 种形式类似，纵向展开的 CP 分解也存在 6 种形式

$$\mathbf{X}^{(JK \times I)} = \begin{bmatrix} \mathbf{X}_{::1} \\ \vdots \\ \mathbf{X}_{::J} \end{bmatrix} = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T, \quad \mathbf{X}^{(JK \times I)} = \begin{bmatrix} \mathbf{X}_{::1}^T \\ \vdots \\ \mathbf{X}_{::K}^T \end{bmatrix} = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T \quad (10.5.24)$$

$$\mathbf{X}^{(KI \times J)} = \begin{bmatrix} \mathbf{X}_{::1} \\ \vdots \\ \mathbf{X}_{::K} \end{bmatrix} = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T, \quad \mathbf{X}^{(KI \times J)} = \begin{bmatrix} \mathbf{X}_{1::}^T \\ \vdots \\ \mathbf{X}_{I::}^T \end{bmatrix} = (\mathbf{A} \odot \mathbf{C})\mathbf{B}^T \quad (10.5.25)$$

$$\mathbf{X}^{(IJ \times K)} = \begin{bmatrix} \mathbf{X}_{1::} \\ \vdots \\ \mathbf{X}_{I::} \end{bmatrix} = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T, \quad \mathbf{X}^{(IJ \times K)} = \begin{bmatrix} \mathbf{X}_{::1}^T \\ \vdots \\ \mathbf{X}_{::J}^T \end{bmatrix} = (\mathbf{B} \odot \mathbf{A})\mathbf{C}^T \quad (10.5.26)$$

这里证明式 (10.5.25) 中的第一个表达式

$$\begin{aligned} \mathbf{X}^{(KI \times J)} &= \begin{bmatrix} \mathbf{X}_{::1} \\ \vdots \\ \mathbf{X}_{::K} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1 \mathbf{b}_1^T c_{11} + \cdots + \mathbf{a}_R \mathbf{b}_R^T c_{1R} \\ \vdots \\ \mathbf{a}_1 \mathbf{b}_1^T c_{K1} + \cdots + \mathbf{a}_R \mathbf{b}_R^T c_{KR} \end{bmatrix} \\ &= \begin{bmatrix} c_{11} \mathbf{a}_1 & \cdots & c_{1R} \mathbf{a}_R \\ \vdots & \ddots & \vdots \\ c_{K1} \mathbf{a}_1 & \cdots & c_{KR} \mathbf{a}_R \end{bmatrix} \begin{bmatrix} \mathbf{b}_1^T \\ \vdots \\ \mathbf{b}_R^T \end{bmatrix} = [\mathbf{c}_1 \otimes \mathbf{a}_1, \dots, \mathbf{c}_R \otimes \mathbf{a}_R] \mathbf{B}^T \\ &= (\mathbf{C} \odot \mathbf{A}) \mathbf{B}^T \end{aligned}$$

式 (10.5.24) 至式 (10.5.26) 的其他各个表达式可类似证明。

表 10.5.4 汇总了三阶张量的纵向展开矩阵与 CP 分解的数学表示。

表 10.5.4 三阶张量的纵向展开矩阵与 CP 分解表示

矩阵化方法	纵向展开矩阵与 CP 分解表示
Kiers 方法	$x_{(k-1)J+j,i}^{(JK \times I)} = x_{ijk} \iff \mathbf{X}^{(JK \times I)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}]^T = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T$
	$x_{(i-1)K+k,j}^{(KI \times J)} = x_{ijk} \iff \mathbf{X}^{(KI \times J)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}]^T = (\mathbf{A} \odot \mathbf{C})\mathbf{B}^T$
	$x_{(j-1)I+i,k}^{(IJ \times K)} = x_{ijk} \iff \mathbf{X}^{(IJ \times K)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}]^T = (\mathbf{B} \odot \mathbf{A})\mathbf{C}^T$
LMV 方法	$x_{(j-1)K+k,i}^{(JK \times I)} = x_{ijk} \iff \mathbf{X}^{(JK \times I)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T]^T = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T$
	$x_{(k-1)I+i,j}^{(KI \times J)} = x_{ijk} \iff \mathbf{X}^{(KI \times J)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T]^T = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T$
	$x_{(i-1)J+j,k}^{(IJ \times K)} = x_{ijk} \iff \mathbf{X}^{(IJ \times K)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{I::}^T]^T = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T$
Kolda 方法	$x_{(k-1)J+j,i}^{(JK \times I)} = x_{ijk} \iff \mathbf{X}^{(JK \times I)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}]^T = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T$
	$x_{(k-1)I+i,j}^{(KI \times J)} = x_{ijk} \iff \mathbf{X}^{(KI \times J)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T]^T = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T$
	$x_{(j-1)I+i,k}^{(IJ \times K)} = x_{ijk} \iff \mathbf{X}^{(IJ \times K)} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{I::}^T]^T = (\mathbf{B} \odot \mathbf{A})\mathbf{C}^T$

由表 10.5.3 和表 10.5.4 易知

$$\mathbf{X}_{\text{Kiers}}^{(n)} = (\mathbf{X}_{\text{Kiers}(n)})^T, \quad \mathbf{X}_{\text{LMV}}^{(n)} = (\mathbf{X}_{\text{LMV}(n)})^T, \quad \mathbf{X}_{\text{Kolda}}^{(n)} = (\mathbf{X}_{\text{Kolda}(n)})^T$$

$N$  阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  的 CP 分解的元素表达式为

$$x_{i_1 \dots i_N} = \sum_{r=1}^R u_{i_1, r}^{(1)} u_{i_2, r}^{(2)} \dots u_{i_N, r}^{(N)} \quad (10.5.27)$$

或等价写作

$$\mathcal{X} = \sum_{r=1}^R \mathbf{u}_r^{(1)} \circ \mathbf{u}_r^{(2)} \circ \dots \circ \mathbf{u}_r^{(N)} \quad (10.5.28)$$

易知, 张量的 CP 分解是矩阵分解式 (10.5.4) 的直接推广。

CP 分解也可以用 Kruskal 算子写作 [276]

$$\mathcal{X} = [\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}] = [\mathcal{I}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}] \quad (10.5.29)$$

显然, CP 分解是核心张量  $\mathcal{G} \in \mathbb{K}^{J_1 \times \dots \times J_N}$  取  $N$  阶单位张量  $\mathcal{I} \in \mathbb{R}^{R \times \dots \times R}$  (其超对角元素为 1, 其他所有元素等于 0) 时 Tucker 分解的特例。

在运用上述模式- $n$  矩阵化公式时, 应该遵循两个基本原则: 不得出现下标等于零或小于零的因子矩阵  $\mathbf{U}_k, k \leq 0$ ; 相同下标的因子矩阵只取 1 次。

表 10.5.5 汇总了 CP 分解的各种数学表示形式。表中,  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R}, n = 1, \dots, N$  称为因子矩阵。

表 10.5.5 CP 分解的数学表示形式

表示方法	数 学 公 式
算子形式	$\mathcal{X} = [\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}] = [\mathcal{I}; \mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}]$
$n$ -模式积	$\mathcal{X} = \mathcal{I} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)}$
元素形式	$x_{i_1 \dots i_N} = \sum_{r=1}^R u_{i_1, r}^{(1)} u_{i_2, r}^{(2)} \dots u_{i_N, r}^{(N)}$
外积表示	$\mathcal{X} = \sum_{r=1}^R \mathbf{u}_r^{(1)} \circ \mathbf{u}_r^{(2)} \circ \dots \circ \mathbf{u}_r^{(N)}$
Kiers 矩阵化	$x_{i_n j}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = x_{i_1 \dots i_N} \quad (j \text{ 由式 (10.2.3) 确定})$ $\Leftrightarrow \mathbf{X}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = \mathbf{U}^{(n)} \left( \mathbf{U}^{(n-1)} \odot \dots \odot \mathbf{U}^{(1)} \odot \mathbf{U}^{(N)} \odot \dots \odot \mathbf{U}^{(n+1)} \right)^T$
LMV 矩阵化	$x_{i_n j}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = x_{i_1 \dots i_N} \quad (j \text{ 由式 (10.2.5) 确定})$ $\Leftrightarrow \mathbf{X}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = \mathbf{U}^{(n)} \left( \mathbf{U}^{(n+1)} \odot \dots \odot \mathbf{U}^{(N)} \odot \mathbf{U}^{(1)} \odot \dots \odot \mathbf{U}^{(n-1)} \right)^T$
Kolda 矩阵化	$x_{i_n j}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = x_{i_1 \dots i_N} \quad (j \text{ 由式 (10.2.6) 确定})$ $\Leftrightarrow \mathbf{X}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)} = \mathbf{U}^{(n)} \left( \mathbf{U}^{(N)} \odot \dots \odot \mathbf{U}^{(n+1)} \odot \mathbf{U}^{(n-1)} \odot \dots \odot \mathbf{U}^{(1)} \right)^T$

表 10.5.6 汇总了 CP 分解的多种变型: PARAFAC2 是 PARAFAC 模型的松弛形式, S-PARAFAC 为移位 PARAFAC, cPARAFAC 为卷积 PARAFAC, 而 PARALIND 则是线性相关的平行因子分析。

表 10.5.6 多路模型的比较<sup>[9]</sup>

模型名称	数学公式	处理秩亏缺	参考文献
PARAFAC	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{rrr} a_{ip} b_{jq} c_{kr}$	×	[93, 216]
PARAFAC2	$\mathbf{X}_{::k} = \mathbf{A}_k \mathbf{D}_k \mathbf{B}^T + \mathbf{E}_{::k}$	×	[217]
S-PARAFAC	$x_{ijk} = \sum_{r=1}^R a_{(i+s_{jr})r} b_{jr} c_{kr} + e_{ijk}$	×	[218]
PARALIND	$\mathbf{X}_{::k} = \mathbf{A} \mathbf{H} \mathbf{D}_k \mathbf{B}^T + \mathbf{E}_{::k}$	✓	[65]
cPARAFAC	$x_{ijk} = \sum_{r=1}^R a_{ir} b_{(j-\theta)r} c_{kr}^\theta + e_{ijk}$	×	[353]
Tucker3	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} a_{ip} b_{jq} c_{kr}$	✓	[487]
Tucker2	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q g_{pqk} a_{ip} b_{jq}$	✓	[487, 489]
Tucker1	$x_{ijk} = \sum_{p=1}^P g_{pjk} a_{ip}$	✓	[487, 489]
S-Tucker3	$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} a_{(i+s_p)p} b_{jq} c_{kr}$	✓	[218]

表中, 矩阵  $\mathbf{D}_k$  为对角矩阵, 其对角元素是模式-3 矩阵  $\mathbf{C}$  的第  $k$  行;  $\mathbf{H}$  是因子矩阵之间的依赖矩阵(相互作用矩阵); 而矩阵  $\mathbf{A}_k$  是与正面切片  $\mathbf{X}_{::k}$  对应的模式-1 矩阵, 并要求服从以下约束条件

$$\mathbf{A}_k^T \mathbf{A}_k = \Phi, \quad k = 1, \dots, K \quad (10.5.30)$$

其中  $\Phi$  是一个与所有切片保持不变的矩阵。

### 10.5.3 CP 分解的唯一性条件

CP 分解存在两种固有的不确定性, 即因子向量的排序不确定性 (permutation indeterminacy) 和尺度不确定性 (scaling indeterminacy)。

首先, 如果我们将因子向量  $(\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r)$  和  $(\mathbf{a}_p, \mathbf{b}_p, \mathbf{c}_p)$  互换, 显然

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

将保持不变。这一排序不确定性也可表述为

$$\mathcal{X} = [\mathbf{A}, \mathbf{B}, \mathbf{C}] = [\mathbf{AP}, \mathbf{BP}, \mathbf{CP}], \quad \forall R \times R \text{ 置换矩阵 } \mathbf{P}$$

式中  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_R]$ ,  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_R]$  和  $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R]$ 。

另外, 只要  $\alpha_r \beta_r \gamma_r = 1$  对所有  $r = 1, \dots, R$  满足, 则

$$\mathcal{X} = \sum_{r=1}^R (\alpha_r \mathbf{a}_r) \circ (\beta_r \mathbf{b}_r) \circ (\gamma_r \mathbf{c}_r) = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

换言之, 张量的 CP 分解对因子向量的尺度(即范数)是盲的。

CP 分解的排序不确定性和尺度不确定对于张量的多线性分析是没有影响的, 因此是允许的。排除这两种固有的不确定性, CP 分解的唯一性系指: 在分解项之和等于原张量的约束下, 秩 1 张量可能的组合是唯一的。

CP 分解在比较宽松的条件下具有唯一性。这一宽松条件与矩阵的 Kruskal 秩有关。

**定义 10.5.1** (Kruskal 秩)<sup>[286]</sup> 一个矩阵  $\mathbf{A} \in \mathbb{R}^{I \times J}$  的 Kruskal 秩(简称  $k$  秩)记作  $\text{rank}_k(\mathbf{A})$  或  $k_{\mathbf{A}}$ , 定义为使得  $\mathbf{A}$  的任意  $r$  个列向量都线性无关的最大整数  $r$ 。

由于矩阵的秩  $\text{rank}(\mathbf{A}) = r$  只要求  $r$  是满足一组列向量线性无关的最大列数, 而矩阵的 Kruskal 秩为  $r$  则要求  $r$  是每组  $r$  个列向量都线性无关的最大列数, 所以 Kruskal 秩总是小于或等于矩阵  $\mathbf{A}$  的秩, 即  $k_{\mathbf{A}} \leqslant r_{\mathbf{A}} \stackrel{\text{def}}{=} \text{rank}(\mathbf{A}) \leqslant \min\{I, J\}, \forall \mathbf{A}$ 。

CP 分解具有唯一性的充分条件是 Kruskal 于 1977 年提出的<sup>[286]</sup>

$$k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}} \geqslant 2R + 2 \quad (10.5.31)$$

其中  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  的列向量分别是  $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r$ 。

Berge 与 Sidiropoulos 证明了<sup>[42]</sup> 对于  $R = 2$  和  $R = 3$ , 上述 Kruskal 充分条件既是张量 CP 分解的充分条件, 也是必要条件; 但是这一结论对  $R > 3$  的情况不成立。

Sidiropoulos 与 Bro<sup>[453]</sup> 推广了 Kruskal 的上述充分条件, 证明了若  $N$  路张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_N}$  的秩为  $R$ , 并且张量的 CP 分解为

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)} \quad (10.5.32)$$

则上述分解为唯一分解的充分条件是

$$\sum_{n=1}^N k_{\mathbf{X}_{(n)}} \geqslant 2R + (N - 1) \quad (10.5.33)$$

式中  $k_{\mathbf{X}_{(n)}}$  是张量  $\mathcal{X}$  的模式- $n$  矩阵化  $\mathbf{X}_{(n)} = \mathbf{X}^{(I_n \times I_1 \dots I_{n-1} I_{n+1} \dots I_N)}$  的 Kruskal 秩。

由三阶张量的纵向展开  $\mathbf{X}^{(JK \times I)} = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T, \mathbf{X}^{(KI \times J)} = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T, \mathbf{X}^{(IJ \times K)} = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T$ , Liu 与 Sidiropoulos<sup>[320]</sup> 证明了三阶张量的 CP 分解唯一性的必要条件为

$$\min\{\text{rank}(\mathbf{A} \odot \mathbf{B}), \text{rank}(\mathbf{B} \odot \mathbf{C}), \text{rank}(\mathbf{C} \odot \mathbf{A})\} = R \quad (10.5.34)$$

$N$  阶张量的 CP 分解唯一性的必要条件为

$$\min_{n=1, \dots, N} \text{rank}(\mathbf{A}_1 \odot \dots \odot \mathbf{A}_{n-1} \odot \mathbf{A}_{n+1} \odot \dots \odot \mathbf{A}_N) = R \quad (10.5.35)$$

式中  $\mathbf{U}^{(n)}$ ,  $n = 1, \dots, N$  是满足纵向展开

$$\mathbf{X}^{(I_2 \cdots I_N \times I_1)} = (\mathbf{A}_N \odot \cdots \odot \mathbf{A}_3 \odot \mathbf{A}_2) \mathbf{A}_1^T \quad (10.5.36)$$

的分量矩阵。

由于  $\text{rank}(\mathbf{A} \odot \mathbf{B}) \leq \text{rank}(\mathbf{A} \otimes \mathbf{B}) \leq \text{rank}(\mathbf{A})\text{rank}(\mathbf{B})$ , 故更简单的必要条件为<sup>[320]</sup>

$$\min_{n=1, \dots, N} \left( \prod_{m=1, m \neq n}^N \text{rank}(\mathbf{A}_m) \right) \geq R \quad (10.5.37)$$

Lathauwer<sup>[300]</sup>证明了, 三阶张量  $\mathbf{A} \in \mathbb{R}^{I \times J \times K}$  的 CP 分解一般是唯一的, 若

$$R \leq K \text{ and } R(R-1) \leq I(I-1)J(J-1)/2 \quad (10.5.38)$$

类似地, 具有秩  $R$  的四阶张量  $\mathbf{A} \in \mathbb{R}^{I \times J \times K \times L}$  的 CP 分解一般是唯一的, 若<sup>[278]</sup>

$$R \leq L \text{ and } R(R-1) \leq IJK(3IJK - IJ - IK - JK - I - J - K + 3)/4 \quad (10.5.39)$$

#### 10.5.4 CP 分解的交替最小二乘算法

考虑三阶张量的 CP 分解的 Kiers 水平展开的分离优化问题

$$\mathbf{A} = \arg \min_{\mathbf{A}} \|\mathbf{X}^{(I \times JK)} - \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T\|_F^2 \quad (10.5.40)$$

$$\mathbf{B} = \arg \min_{\mathbf{B}} \|\mathbf{X}^{(J \times KI)} - \mathbf{B}(\mathbf{A} \odot \mathbf{C})^T\|_F^2 \quad (10.5.41)$$

$$\mathbf{C} = \arg \min_{\mathbf{C}} \|\mathbf{X}^{(K \times IJ)} - \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T\|_F^2 \quad (10.5.42)$$

它们的最小二乘解分别为

$$\mathbf{A} = \mathbf{X}^{(I \times JK)} ((\mathbf{C} \odot \mathbf{B})^T)^\dagger \quad (10.5.43)$$

$$\mathbf{B} = \mathbf{X}^{(J \times KI)} ((\mathbf{A} \odot \mathbf{C})^T)^\dagger \quad (10.5.44)$$

$$\mathbf{C} = \mathbf{X}^{(K \times IJ)} ((\mathbf{B} \odot \mathbf{A})^T)^\dagger \quad (10.5.45)$$

利用 Khatri-Rao 积的 Moore-Penrose 逆矩阵的性质

$$(\mathbf{C} \odot \mathbf{B})^\dagger = (\mathbf{C}^T \mathbf{C} * \mathbf{B}^T \mathbf{B})^\dagger (\mathbf{C} \odot \mathbf{B})^T \quad (10.5.46)$$

可求得因子矩阵  $\mathbf{A}$  的解为

$$\mathbf{A} = \mathbf{X}^{(I \times JK)} (\mathbf{C} \odot \mathbf{B}) (\mathbf{C}^T \mathbf{C} * \mathbf{B}^T \mathbf{B})^\dagger \quad (10.5.47)$$

仿此, 又可分别得到模式-B 矩阵和模式-C 矩阵的最小二乘解为

$$\mathbf{B} = \mathbf{X}^{(J \times KI)} (\mathbf{A} \odot \mathbf{C}) (\mathbf{A}^T \mathbf{A} * \mathbf{C}^T \mathbf{C})^\dagger \quad (10.5.48)$$

$$\mathbf{C} = \mathbf{X}^{(K \times IJ)} (\mathbf{B} \odot \mathbf{A}) (\mathbf{B}^T \mathbf{B} * \mathbf{A}^T \mathbf{A})^\dagger \quad (10.5.49)$$

**算法 10.5.1 CP 分解的交替最小二乘算法——Kiers 水平展开矩阵形式<sup>[66, 9]</sup>**

输入 张量  $\mathcal{X}$  的 Kiers 水平展开矩阵  $\mathbf{X}^{(I \times JK)}, \mathbf{X}^{(J \times KI)}, \mathbf{X}^{(K \times IJ)}$  及因子个数  $R$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I \times R}, \mathbf{B} \in \mathbb{R}^{J \times R}, \mathbf{C} \in \mathbb{R}^{K \times R}$ 。

初始化 矩阵  $\mathbf{B}_0$  和  $\mathbf{C}_0$ 。

步骤 1 对  $k = 1, 2, \dots$ , 执行以下更新

$$\begin{aligned}\mathbf{A}_{k+1} &\leftarrow \mathbf{X}^{(I \times JK)} (\mathbf{C}_k \odot \mathbf{B}_k) (\mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k)^\dagger \\ \mathbf{B}_{k+1} &\leftarrow \mathbf{X}^{(J \times KI)} (\mathbf{A}_{k+1} \odot \mathbf{C}_k) (\mathbf{A}_{k+1}^T \mathbf{A}_{k+1} * \mathbf{C}_k^T \mathbf{C}_k)^\dagger \\ \mathbf{C}_{k+1} &\leftarrow \mathbf{X}^{(K \times IJ)} (\mathbf{B}_{k+1} \odot \mathbf{A}_{k+1}) (\mathbf{B}_{k+1}^T \mathbf{B}_{k+1} * \mathbf{A}_{k+1}^T \mathbf{A}_{k+1})^\dagger\end{aligned}$$

步骤 2 收敛条件检验：若对某个误差常数  $\epsilon > 0$ , 收敛条件

$$\|\mathbf{X}^{(I \times JK)} - \mathbf{A}_{k+1} (\mathbf{C}_{k+1} \odot \mathbf{B}_{k+1})^T\|_2^2 < \epsilon$$

满足，则停止迭代，并输出因子矩阵  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ；否则，返回步骤 1，继续迭代，直至收敛。

类似地，又可得到针对三阶张量的 LMV 纵向展开矩阵的 CP 分解的交替最小二乘算法。

**算法 10.5.2 CP 分解的交替最小二乘算法——LMV 纵向展开矩阵形式<sup>[303]</sup>**

输入 三阶张量  $\mathcal{X}$  的 LMV 纵向展开矩阵  $\mathbf{X}^{(JK \times I)}, \mathbf{X}^{(KI \times J)}, \mathbf{X}^{(IJ \times K)}$ ，以及因子个数  $R$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I \times R}, \mathbf{B} \in \mathbb{R}^{J \times R}, \mathbf{C} \in \mathbb{R}^{K \times R}$ 。

初始化 矩阵  $\mathbf{B}_0$  和  $\mathbf{C}_0$ 。

步骤 1 对  $k = 1, 2, \dots$ , 执行以下更新

$$\begin{aligned}\mathbf{A}_{k+1} &\leftarrow [(\mathbf{B}_k \odot \mathbf{C}_k)^\dagger \mathbf{X}^{(JK \times I)}]^T \\ \tilde{\mathbf{B}} &\leftarrow [(\mathbf{C}_k \odot \mathbf{A}_{k+1})^\dagger \mathbf{X}^{(KI \times J)}]^T \\ [\mathbf{B}_{k+1}]_{:,r} &= \tilde{\mathbf{B}}_{:,r} / \|\tilde{\mathbf{B}}_{:,r}\|_2, \quad r = 1, \dots, R \\ \tilde{\mathbf{C}} &\leftarrow [(\mathbf{A}_{k+1} \odot \mathbf{B}_{k+1})^\dagger \mathbf{X}^{(IJ \times K)}]^T \\ [\mathbf{C}_{k+1}]_{:,r} &= \tilde{\mathbf{C}}_{:,r} / \|\tilde{\mathbf{C}}_{:,r}\|_2, \quad r = 1, \dots, R\end{aligned}$$

步骤 2 收敛条件检验：若对某个误差常数  $\epsilon > 0$ , 收敛条件

$$\|\mathbf{X}^{(JK \times I)} - (\mathbf{B} \odot \mathbf{C}) \mathbf{A}^T\|_2^2 < \epsilon$$

满足，则停止迭代，并输出因子矩阵  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ；否则，返回步骤 1，继续迭代，直至收敛。

下面是基于 Kolda 水平展开矩阵的  $N$  阶张量的 CP 分解的交替最小二乘算法<sup>[278]</sup>。

**算法 10.5.3 CP-ALS( $\mathcal{X}, R$ )**

输入 三阶张量  $\mathcal{X}$  以及因子个数  $R$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I \times R}, \mathbf{B} \in \mathbb{R}^{J \times R}, \mathbf{C} \in \mathbb{R}^{K \times R}$ 。

初始化  $\mathbf{A}_n \in \mathbb{R}^{I_n \times R}, n = 1, \dots, N$ 。

步骤 1 计算三阶张量  $\mathcal{X}$  的 Kolda 水平展开矩阵  $\mathbf{X}^{(I \times JK)}, \mathbf{X}^{(J \times KI)}, \mathbf{X}^{(K \times IJ)}$ 。

步骤 2 对  $n = 1, \dots, N$ , 计算

$$\begin{aligned}\mathbf{V} &\leftarrow \mathbf{A}_1^T \mathbf{A}_1 * \cdots * \mathbf{A}_{n-1}^T \mathbf{A}_{n-1} * \mathbf{A}_{n+1}^T \mathbf{A}_{n+1} * \cdots * \mathbf{A}_N^T \mathbf{A}_N \\ \mathbf{A}_n &\leftarrow \mathbf{X}_{(n)}(\mathbf{A}_N \odot \cdots \odot \mathbf{A}_{n+1} \odot \mathbf{A}_{n-1} \odot \cdots \odot \mathbf{A}_1) \mathbf{V}^\dagger \\ \lambda_n &\leftarrow \|\mathbf{A}_n\| \\ \mathbf{A}_n &\leftarrow \mathbf{A}_n / \lambda_n\end{aligned}$$

步骤 3 若收敛条件满足或达到最大迭代步数, 则输出  $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_N]^T$  和  $\mathbf{A}_1, \dots, \mathbf{A}_N$ ; 否则, 重复步骤 2 的运算, 直到收敛条件满足或者达到最大迭代步数。

交替最小二乘算法的主要优点是简单、容易实现, 主要缺点是有可能迭代过程徘徊不止, 不能收敛; 或者因为迭代过程陷入泥沼之中, 使得需要经过漫长迭代, 最终才能收敛。大多数的交替最小二乘算法收敛都比较慢。如果算法的结果检测到异常的数据值, 还需要重新拟合模型, 收敛慢的问题将更加突出。

为了避免交替最小二乘方法的发散或者缓慢的收敛, 需要对交替最小二乘的代价函数正则化。下面是三阶张量的 CP 分解的两种正则交替最小二乘迭代公式:

(1) Kolda 水平展开 [315]

$$\mathbf{A}_{k+1} = \arg \min_{\mathbf{A}} \|\mathbf{X}^{(I \times JK)} - \mathbf{A}(\mathbf{C}_k \odot \mathbf{B}_k)^T\|_F^2 + \tau_k \|\mathbf{A} - \mathbf{A}_k\|_F^2 \quad (10.5.50)$$

$$\mathbf{B}_{k+1} = \arg \min_{\mathbf{B}} \|\mathbf{X}^{(J \times KI)} - \mathbf{B}(\mathbf{C}_k \odot \mathbf{A}_{k+1})^T\|_F^2 + \tau_k \|\mathbf{B} - \mathbf{B}_k\|_F^2 \quad (10.5.51)$$

$$\mathbf{C}_{k+1} = \arg \min_{\mathbf{C}} \|\mathbf{X}^{(K \times IJ)} - \mathbf{C}(\mathbf{B}_{k+1} \odot \mathbf{A}_{k+1})^T\|_F^2 + \tau_k \|\mathbf{C} - \mathbf{C}_k\|_F^2 \quad (10.5.52)$$

(2) LMV 纵向展开 [356]

$$\mathbf{A}_{k+1}^T = \arg \min_{\mathbf{A}} \|\mathbf{X}^{(JK \times I)} - (\mathbf{B}_k \odot \mathbf{C}_k) \mathbf{A}^T\|_F^2 + \tau_k \|\mathbf{A}^T - \mathbf{A}_k^T\|_F^2 \quad (10.5.53)$$

$$\mathbf{B}_{k+1}^T = \arg \min_{\mathbf{B}} \|\mathbf{X}^{(KI \times J)} - (\mathbf{C}_k \odot \mathbf{A}_{k+1}) \mathbf{B}^T\|_F^2 + \tau_k \|\mathbf{B}^T - \mathbf{B}_k^T\|_F^2 \quad (10.5.54)$$

$$\mathbf{C}_{k+1}^T = \arg \min_{\mathbf{C}} \|\mathbf{X}^{(IJ \times K)} - (\mathbf{A}_{k+1} \odot \mathbf{B}_{k+1}) \mathbf{C}^T\|_F^2 + \tau_k \|\mathbf{C}^T - \mathbf{C}_k^T\|_F^2 \quad (10.5.55)$$

以 Kolda 水平展开中的因子矩阵  $\mathbf{A}$  的更新为例, 正则项  $\|\mathbf{A} - \mathbf{A}_k\|_F^2$  可以迫使更新之后的矩阵  $\mathbf{A}$  不会偏离  $\mathbf{A}_k$  太多, 从而避免迭代过程的发散。

求子优化问题式 (10.5.50) 中目标函数关于变元矩阵  $\mathbf{A}$  的梯度矩阵, 并令梯度矩阵等于零矩阵, 则有

$$((\mathbf{C}_k \odot \mathbf{B}_k)^T (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{I}) \mathbf{A}^T = (\mathbf{C}_k \odot \mathbf{B}_k)^T (\mathbf{X}^{(I \times JK)})^T + \tau_k (\mathbf{A}_k)^T$$

由此得子优化问题式 (10.5.50) 的正则化最小二乘解为

$$\begin{aligned}\mathbf{A} &= \left( \mathbf{X}^{I \times JK} (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{A}_k \right) \left( (\mathbf{C}_k \odot \mathbf{B}_k)^T (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{I} \right)^{-1} \\ &= \left( \mathbf{X}^{I \times JK} (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{A}_k \right) \left( \mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k + \tau_k \mathbf{I} \right)^{-1}\end{aligned}\quad (10.5.56)$$

类似地, 可得

$$\mathbf{B} = \left( \mathbf{X}^{(J \times KI)} (\mathbf{C}_k \odot \mathbf{A}_{k+1}) + \tau_k \mathbf{B}_k \right) \left( \mathbf{C}_k^T \mathbf{C}_k * \mathbf{A}_{k+1}^T \mathbf{A}_{k+1} + \tau_i \mathbf{I} \right)^{-1} \quad (10.5.57)$$

$$\mathbf{C} = \left( \mathbf{X}^{(K \times IJ)} (\mathbf{B}_{k+1} \odot \mathbf{A}_{k+1}) + \tau_k \mathbf{C}_k \right) \left( \mathbf{B}_{k+1}^T \mathbf{B}_{k+1} * \mathbf{A}_{k+1}^T \mathbf{A}_{k+1} + \tau_k \mathbf{I} \right)^{-1} \quad (10.5.58)$$

以下是三阶张量的 CP 分解的 Kolda 水平展开形式的正则交替最小二乘算法 [315]。

#### 算法 10.5.4 正则交替最小二乘算法 CP-RALS( $\mathcal{X}, R, N, \lambda$ )

输入 三阶张量  $\mathcal{X}$  的 Kolda 水平展开矩阵  $\mathbf{X}^{(I \times JK)}, \mathbf{X}^{(J \times KI)}, \mathbf{X}^{(K \times IJ)}$ , 以及因子个数  $R$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I \times R}, \mathbf{B} \in \mathbb{R}^{J \times R}, \mathbf{C} \in \mathbb{R}^{K \times R}$ 。

初始化  $\mathbf{A}_0 \in \mathbb{R}^{I \times R}, \mathbf{B}_0 \in \mathbb{R}^{J \times R}, \mathbf{C}_0 \in \mathbb{R}^{K \times R}, \tau_0$ 。

迭代  $k = 1, 2, \dots$

$$\begin{aligned}\mathbf{W} &\leftarrow \mathbf{X}^{I \times JK} (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{A}_k \\ \mathbf{S} &\leftarrow \mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k + \tau_k \mathbf{I} \\ \mathbf{A}_{k+1} &\leftarrow \mathbf{W} \mathbf{S}^{-1} \\ \mathbf{W} &\leftarrow \mathbf{X}^{(J \times KI)} (\mathbf{C}_k \odot \mathbf{A}_{k+1}) + \tau_k \mathbf{B}_k \\ \mathbf{S} &\leftarrow \mathbf{C}_k^T \mathbf{C}_k * \mathbf{A}_{k+1}^T \mathbf{A}_{k+1} + \tau_i \mathbf{I} \\ \mathbf{B}_{k+1} &\leftarrow \mathbf{W} \mathbf{S}^{-1} \\ \mathbf{W} &\leftarrow \mathbf{X}^{(K \times IJ)} (\mathbf{B}_{k+1} \odot \mathbf{A}_{k+1}) + \tau_k \mathbf{C}_k \\ \mathbf{S} &\leftarrow \mathbf{B}_{k+1}^T \mathbf{B}_{k+1} * \mathbf{A}_{k+1}^T \mathbf{A}_{k+1} + \tau_k \mathbf{I} \\ \mathbf{C}_{k+1} &\leftarrow \mathbf{W} \mathbf{S}^{-1} \\ \tau_{k+1} &\leftarrow \delta \cdot \tau_k\end{aligned}$$

若  $\tau_k \equiv 0, \forall k$ , 则算法 10.5.4 退化为普通的交替最小二乘算法。可以看出, 以因子矩阵  $\mathbf{A}$  的更新为例, 正则交替最小二乘算法对交替最小二乘算法的改进主要体现在以下两个方面:

(1) 用  $(\mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k + \tau_k \mathbf{I})^{-1}$  代替  $(\mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k)^\dagger$ , 可以避免  $\mathbf{C}_k^T \mathbf{C}_k * \mathbf{B}_k^T \mathbf{B}_k$  的可能奇异带来的数值稳定性问题。

(2) 用  $\mathbf{X}^{I \times JK} (\mathbf{C}_k \odot \mathbf{B}_k) + \tau_k \mathbf{A}_k$  代替  $\mathbf{X}^{I \times JK} (\mathbf{C}_k \odot \mathbf{B}_k)$ , 使得  $\mathbf{A}_{k+1}$  的更新通过  $\tau_k$  的加权作用, 与  $\mathbf{A}_k$  弱相关, 可以防止  $\mathbf{A}_{k+1}$  发生突跳。

需要指出的是, 如果完全与 Tikhonov 正则化一样, 正则项为  $\tau_k \|\mathbf{A}\|_F^2$ , 而不是  $\tau_k \|\mathbf{A} - \mathbf{A}_k\|_F^2$ , 则只能有上述优点 (1), 而不可能有优点 (2)。

## 10.6 多路数据分析的预处理与后处理

前面介绍了多路数据分析的理论与方法。和二路数据处理一样，多路数据处理也需要预处理和后处理 [219, 41, 63, 267]。

### 10.6.1 多路数据的中心化与比例化

在矩阵分析中，最常用的预处理是数据的零均值化。张量分析所需要的预处理比矩阵分析的预处理更为复杂，不仅需要中心化 (centering)，而且有必要对数据进行比例化或缩放 (scaling)。

零均值化或中心化的主要目的是剔除原始数据中的“直流”即固定的分量，只保留其中随机变化的分量。

张量数据的中心化必须指明是针对哪一个模式进行的。以三阶张量  $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$  为例，对张量元素的模式- $n$  中心化分别为

$$x_{ijk}^{\text{cent } 1} = x_{ijk} - \bar{x}_{\cdot jk}, \quad \bar{x}_{\cdot jk} = \frac{1}{I} \sum_{i=1}^I x_{ijk} \quad (10.6.1)$$

$$x_{ijk}^{\text{cent } 2} = x_{ijk} - \bar{x}_{i \cdot k}, \quad \bar{x}_{i \cdot k} = \frac{1}{J} \sum_{j=1}^J x_{ijk} \quad (10.6.2)$$

$$x_{ijk}^{\text{cent } 3} = x_{ijk} - \bar{x}_{ij \cdot}, \quad \bar{x}_{ij \cdot} = \frac{1}{K} \sum_{k=1}^K x_{ijk} \quad (10.6.3)$$

若中心化是针对两个模式同时进行的，则有

$$x_{ijk}^{\text{cent } (1,2)} = x_{ijk} - \bar{x}_{\cdot \cdot k}, \quad \bar{x}_{\cdot \cdot k} = \frac{1}{IJ} \sum_{i=1}^I \sum_{j=1}^J x_{ijk} \quad (10.6.4)$$

$$x_{ijk}^{\text{cent } (2,3)} = x_{ijk} - \bar{x}_{i \cdot \cdot}, \quad \bar{x}_{i \cdot \cdot} = \frac{1}{JK} \sum_{j=1}^J \sum_{k=1}^K x_{ijk} \quad (10.6.5)$$

$$x_{ijk}^{\text{cent } (3,1)} = x_{ijk} - \bar{x}_{\cdot \cdot j}, \quad \bar{x}_{\cdot \cdot j} = \frac{1}{IK} \sum_{i=1}^I \sum_{k=1}^K x_{ijk} \quad (10.6.6)$$

此外，张量数据还需要进行比例化或缩放，即对每一个数据除以某个固定因子

$$x_{ijk}^{\text{scal } 1} = \frac{x_{ijk}}{s_i}, \quad s_i = \sqrt{\sum_{j=1}^J \sum_{k=1}^K x_{ijk}^2} \quad (i = 1, \dots, I) \quad (10.6.7)$$

$$x_{ijk}^{\text{scal } 2} = \frac{x_{ijk}}{s_j}, \quad s_j = \sqrt{\sum_{i=1}^I \sum_{k=1}^K x_{ijk}^2} \quad (j = 1, \dots, J) \quad (10.6.8)$$

$$x_{ijk}^{\text{scal } 3} = \frac{x_{ijk}}{s_k}, \quad s_k = \sqrt{\sum_{i=1}^I \sum_{j=1}^J x_{ijk}^2} \quad (k = 1, \dots, K) \quad (10.6.9)$$

需要注意的是，通常只对三阶张量展开矩阵的列进行中心化，对行进行缩放<sup>[63]</sup>。因此，有

$$\begin{aligned} \text{Kiers 矩阵化} & \left\{ \begin{array}{l} \mathbf{X}^{I \times JK} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] \text{ 进行中心化 } x_{ijk}^{\text{cent } 2}, \text{ 缩放 } x_{ijk}^{\text{scal } 1} \\ \mathbf{X}^{J \times KI} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}] \text{ 进行中心化 } x_{ijk}^{\text{cent } 3}, \text{ 缩放 } x_{ijk}^{\text{scal } 2} \\ \mathbf{X}^{K \times IJ} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}] \text{ 进行中心化 } x_{ijk}^{\text{cent } 1}, \text{ 缩放 } x_{ijk}^{\text{scal } 3} \end{array} \right. \\ \text{LMV 矩阵化} & \left\{ \begin{array}{l} \mathbf{X}^{I \times JK} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T] \text{ 进行中心化 } x_{ijk}^{\text{cent } 3}, \text{ 缩放 } x_{ijk}^{\text{scal } 1} \\ \mathbf{X}^{J \times KI} = [\mathbf{X}_{1::}^T, \dots, \mathbf{X}_{::K}^T] \text{ 进行中心化 } x_{ijk}^{\text{cent } 1}, \text{ 缩放 } x_{ijk}^{\text{scal } 2} \\ \mathbf{X}^{K \times IJ} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{I::}^T] \text{ 进行中心化 } x_{ijk}^{\text{cent } 2}, \text{ 缩放 } x_{ijk}^{\text{scal } 3} \end{array} \right. \\ \text{Kolda 矩阵化} & \left\{ \begin{array}{l} \mathbf{X}^{I \times JK} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] \text{ 进行中心化 } x_{ijk}^{\text{cent } 2}, \text{ 缩放 } x_{ijk}^{\text{scal } 1} \\ \mathbf{X}^{J \times KI} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T] \text{ 进行中心化 } x_{ijk}^{\text{cent } 1}, \text{ 缩放 } x_{ijk}^{\text{scal } 2} \\ \mathbf{X}^{K \times IJ} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::J}^T] \text{ 进行中心化 } x_{ijk}^{\text{cent } 1}, \text{ 缩放 } x_{ijk}^{\text{scal } 3} \end{array} \right. \end{aligned}$$

### 10.6.2 正则化与数据阵列的压缩

在交替最小二乘算法 10.5.3 的迭代过程中，矩阵  $\mathbf{V}$  有可能出现病态。此时，需要采用 Tikhonov 正则化，即使用  $\mathbf{V}^\dagger = (\mathbf{V}^H \mathbf{V} + \lambda \mathbf{I})^{-1} \mathbf{V}^H$  或者  $\mathbf{V}^\dagger = \mathbf{V}^H (\mathbf{V} \mathbf{V}^H + \lambda \mathbf{I})^{-1}$ ，其中  $\lambda > 0$  是一个很小的常数。

当高阶阵列的某个维数很大（例如  $I_n$  数百或者数千）时，CP 分解的交替最小二乘算法的收敛慢问题往往会很严重。此时，需要对数据阵列进行压缩，将原  $N$  阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_N}$  压缩成一个维数更小的  $N$  阶核心张量  $\mathcal{G} \in \mathbb{R}^{J_1 \times J_N}$ ，然后对核心张量的 CP 分解运行交替最小二乘算法，最后由核心张量的 CP 分解得到的模式矩阵，重构原张量的 CP 分解的模式矩阵。

以三阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$  的 CP 分解

$$\mathbf{X}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T, \quad \mathbf{X}_{(2)} = \mathbf{B}(\mathbf{A} \odot \mathbf{C})^T, \quad \mathbf{X}_{(3)} = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$$

为例，假如某个维数（例如  $I_3$ ）很大，由于矩阵化  $\mathbf{X}_{(n)}, n = 1, 2, 3$  的维数很大，所以直接求因子矩阵  $\mathbf{A} \in \mathbb{R}^{I_1 \times R}, \mathbf{B} \in \mathbb{R}^{I_2 \times R}, \mathbf{C} \in \mathbb{R}^{I_3 \times R}$  的交替最小二乘算法将面临计算量大，收敛慢的困境。

选择三个正交矩阵  $\mathbf{U} \in \mathbb{R}^{I_1 \times J_1}, \mathbf{V} \in \mathbb{R}^{I_2 \times J_2}, \mathbf{W} \in \mathbb{R}^{I_3 \times J_3}$ ，使得

$$\mathbf{A} = \mathbf{U}\mathbf{P}, \quad \mathbf{B} = \mathbf{V}\mathbf{Q}, \quad \mathbf{C} = \mathbf{W}\mathbf{R}$$

于是，三阶张量 CP 分解的三个模式矩阵  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  的辨识分为正交矩阵三元组  $(\mathbf{U}, \mathbf{V}, \mathbf{W})$  和非正交矩阵三元组  $(\mathbf{P}, \mathbf{Q}, \mathbf{R})$  的两个子辨识问题。

利用 Khatri-Rao 积与 Kronecker 积之间的关系  $(\mathbf{W}\mathbf{R}) \odot (\mathbf{V}\mathbf{Q}) = (\mathbf{W} \otimes \mathbf{V})(\mathbf{R} \odot \mathbf{Q})$ ，易知

$$\mathbf{X}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T = \mathbf{U}\mathbf{P}[(\mathbf{W}\mathbf{R}) \odot (\mathbf{V}\mathbf{Q})]^T = \mathbf{U}\mathbf{P}(\mathbf{R} \odot \mathbf{Q})^T(\mathbf{W} \otimes \mathbf{V})^T$$

若令

$$\mathbf{G}_{(1)} = \mathbf{P}(\mathbf{R} \odot \mathbf{Q})^T$$

则  $\mathbf{X}_{(1)}$  可以改写为

$$\mathbf{X}_{(1)} = \mathbf{U}\mathbf{G}_{(1)}(\mathbf{W} \odot \mathbf{V})^T \quad (10.6.10)$$

类似地, 可以证明

$$\mathbf{X}_{(2)} = \mathbf{V}\mathbf{G}_{(2)}(\mathbf{U} \odot \mathbf{W})^T \quad (10.6.11)$$

$$\mathbf{X}_{(3)} = \mathbf{W}\mathbf{G}_{(3)}(\mathbf{V} \odot \mathbf{U})^T \quad (10.6.12)$$

式中

$$\mathbf{G}_{(2)} = \mathbf{Q}(\mathbf{P} \odot \mathbf{R})^T, \quad \mathbf{G}_{(3)} = \mathbf{R}(\mathbf{Q} \odot \mathbf{P})^T$$

式 (10.6.10) ~ 式 (10.6.12) 表明:

(1)  $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times J_3}$  是三阶张量  $\mathcal{X}$  的 Tucker 分解的核心张量, 而  $\mathbf{G}_{(n)}, n = 1, 2, 3$  是核心张量的模式- $n$  矩阵化。

(2) 矩阵三元组  $(\mathbf{P}, \mathbf{Q}, \mathbf{R})$  是核心张量  $\mathcal{G}$  的 CP 分解的因子矩阵。

以上讨论可以总结出数据阵列的压缩算法如下。

#### 算法 10.6.1 数据阵列压缩

输入 三阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ 。

输出 因子矩阵  $\mathbf{A} \in \mathbb{R}^{I_1 \times J_1}, \mathbf{B} \in \mathbb{R}^{I_2 \times J_2}, \mathbf{C} \in \mathbb{R}^{I_3 \times J_3}$ 。

步骤 1 利用张量的展开矩阵  $\mathbf{X}_{(n)}$  的奇异值分解, 分别求出正交的矩阵  $\mathbf{U}, \mathbf{V}, \mathbf{W}$

$$[\mathbf{U}, \mathbf{S}, \mathbf{T}] = \text{SVD}(\mathbf{X}_{(1)}, J_1) \quad (10.6.13)$$

$$[\mathbf{V}, \mathbf{S}, \mathbf{T}] = \text{SVD}(\mathbf{X}_{(2)}, J_2) \quad (10.6.14)$$

$$[\mathbf{W}, \mathbf{S}, \mathbf{T}] = \text{SVD}(\mathbf{X}_{(3)}, J_3) \quad (10.6.15)$$

其中  $\mathbf{U} \in \mathbb{R}^{I_1 \times J_1}, \mathbf{V} \in \mathbb{R}^{I_2 \times J_2}, \mathbf{W} \in \mathbb{R}^{I_3 \times J_3}$  分别是展开矩阵  $\mathbf{X}_{(1)}, \mathbf{X}_{(2)}, \mathbf{X}_{(3)}$  的奇异值分解中与  $J_1, J_2, J_3$  个主奇异值对应的左奇异向量矩阵。

步骤 2 计算核心张量的模式- $n$  矩阵

$$\mathbf{G}_{(1)} = \mathbf{U}^T \mathbf{X}_{(1)} (\mathbf{W} \otimes \mathbf{V}) \quad (10.6.16)$$

$$\mathbf{G}_{(2)} = \mathbf{V}^T \mathbf{X}_{(2)} (\mathbf{U} \otimes \mathbf{W}) \quad (10.6.17)$$

$$\mathbf{G}_{(3)} = \mathbf{W}^T \mathbf{X}_{(3)} (\mathbf{V} \otimes \mathbf{U}) \quad (10.6.18)$$

步骤 3 用交替最小二乘算法 10.5.2 求解核心张量  $\mathcal{G}$  的 CP 分解

$$\mathbf{G}_{(1)} = \mathbf{P}(\mathbf{R} \odot \mathbf{Q})^T, \quad \mathbf{G}_{(2)} = \mathbf{Q}(\mathbf{P} \odot \mathbf{R})^T, \quad \mathbf{G}_{(3)} = \mathbf{R}(\mathbf{Q} \odot \mathbf{P})^T$$

得到因子矩阵  $\mathbf{P} \in \mathbb{R}^{J_1 \times R}, \mathbf{Q} \in \mathbb{R}^{J_2 \times R}, \mathbf{R} \in \mathbb{R}^{J_3 \times R}$ 。

步骤 4 利用矩阵乘法, 求出原三阶张量的 CP 分解的因子矩阵

$$\mathbf{A} = \mathbf{U}\mathbf{P}, \quad \mathbf{B} = \mathbf{V}\mathbf{Q}, \quad \mathbf{C} = \mathbf{W}\mathbf{R} \quad (10.6.19)$$

**注释 1** 数据阵列的压缩本质是: 将大维数的原张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$  的 CP 分解“压缩为”小维数的核心张量  $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times J_3}$  的 CP 分解。

**注释 2** 压缩算法无须迭代, 计算简单, 不存在收敛问题。

**注释 3** 若将压缩算法求出的因子矩阵  $\mathbf{B}$  和  $\mathbf{C}$  作为初始化矩阵, 直接对原三阶张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$  运行 CP 分解的交替最小二乘算法 10.6.1, 则只需要少数几步迭代, 即可使交替最小二乘算法趋于收敛。因此, 数据阵列的压缩可有效加速大维数的张量的 CP 分解。

**注释 4** 压缩算法是针对张量的 Kiers 矩阵化设计的。只需要将算法中的展开矩阵和模式- $n$  矩阵作适当替换, 即可得适用于 LMV 矩阵化或者 Kolda 矩阵化的数据阵列压缩算法。

在使用交替最小二乘算法求出因子矩阵(载荷矩阵)  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  之后, 有必要对这些载荷矩阵进行质量评估——影响力分析<sup>[63]</sup>。

由于 CP 分解中的载荷矩阵不是正交矩阵, 所以需要计算一个载荷矩阵的影响力 (leverage) 向量。载荷矩阵  $\mathbf{U}$  的影响力向量定义为该矩阵的投影矩阵的对角元素组成的向量

$$\mathbf{v} = \text{diag}(\mathbf{U}(\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T) \quad (10.6.20)$$

依次令  $\mathbf{U} = \mathbf{A}, \mathbf{B}, \mathbf{C}$ , 即分别得到三个载荷矩阵的影响力向量。影响力向量的所有元素的值均介于 0 和 1 之间, 即  $0 \leq v_i \leq 1$ 。若某个影响力元素值越大, 则所使用的样本数值的影响力越大; 反之, 则影响力越小。如果某个影响力元素很大, 则说明样本数据中可能含有异常值 (outlier), 所得到的模型是不适当的, 需要剔除异常值, 重新启动交替最小二乘, 进行新的张量分析。

## 10.7 非负张量分解

第 6 章 6.6 节的讨论表明, 在很多现代的数据分析中, 非负矩阵分解比主分量分析、独立分量分析等方法更加有用。现在考虑非负矩阵分解对多路数据阵列(张量)的推广。

非负张量分解 (nonnegative tensor decomposition, NTF) 最早是化学计量学的研究人员以具有非负约束的 PARAFAC 的方式进行研究的<sup>[64, 67, 387, 388]</sup>。

一个全部元素为非负实数的张量称为非负张量。非负张量分解问题的提法是: 给定一个  $N$  阶非负张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ , 将其分解为

$$\text{NTD1: } \mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \cdots \times_N \mathbf{A}^{(N)}, \quad \mathcal{G}, \mathbf{A}^{(1)} \geq 0, \dots, \mathbf{A}^{(N)} \geq 0 \quad (10.7.1)$$

或者

$$\text{NTD 2: } \mathcal{X} \approx \mathcal{I} \times_1 \mathbf{A}^{(1)} \times_2 \cdots \times_N \mathbf{A}^{(N)}, \quad \mathbf{A}^{(1)} \geq 0, \dots, \mathbf{A}^{(N)} \geq 0 \quad (10.7.2)$$

非负张量分解可以用目标函数的最小化表示为

$$\text{NTD 1: } \min_{\mathcal{G}, \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}} \frac{1}{2} \|\mathcal{X} - \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \cdots \times_N \mathbf{A}^{(N)}\|_F^2 \quad (10.7.3)$$

或者

$$\text{NTD 2: } \min_{\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}} \frac{1}{2} \|\mathcal{X} - \mathcal{I} \times_1 \mathbf{A}^{(1)} \times_2 \cdots \times_N \mathbf{A}^{(N)}\|_F^2 \quad (10.7.4)$$

非负张量分解的基本思想是：将非负张量分解问题改写为非负矩阵分解的基本形式  $\mathbf{X} = \mathbf{AS}$ ,  $\mathbf{A} \succeq 0, \mathbf{S} \succeq 0$ 。非负张量分解有两类常用算法：Lee 和 Seung 的乘法更新算法和交替最小二乘更新算法。乘法更新算法的优点是实现简单，但收敛比较慢。此外，由于需要使用目标函数相对于各个因子矩阵的梯度矩阵，而非负张量分解的因子矩阵又比较多，所以各个梯度的计算比较麻烦。与乘法更新算法相比，交替最小二乘算法更适合非负张量分解的计算，因为只允许一个因子矩阵为优化问题的变元，而固定其他因子矩阵不变，最小二乘方法就可以交替进行，得到非负张量分解所需要的全部因子矩阵，更容易实现。因此，交替最小二乘算法成为非负张量分解的主流算法。

### 10.7.1 非负张量分解的乘法算法

设计非负张量分解的乘法算法的关键是如何将张量的 Tucker 分解或者 CP 分解写成非负矩阵分解的标准形式  $\mathbf{X} = \mathbf{AS}$ 。

#### 1. 非负张量 Tucker 分解的乘法算法

考虑非负张量  $\mathcal{X} = [x_{i_1 \dots i_N}] \in \mathbb{R}^{I_1 \times \dots \times I_N}$  的近似 Tucker 分解

$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \cdots \times_N \mathbf{A}^{(N)} \quad (10.7.5)$$

其中  $x_{i_1 \dots i_N} \geq 0$ ,  $\mathbf{A}^{(n)} = [a_{ij}^{(n)}] \in \mathbb{R}^{I_n \times J_n}$ ,  $n = 1, \dots, N$  为  $N$  个非负因子矩阵，而  $\mathcal{G} = [g_{j_1 \dots j_N}] \in \mathbb{R}^{J_1 \times \dots \times J_N}$  为非负核心张量，即  $a_{ij}^{(n)} \geq 0, \forall i = 1, \dots, I_n; j = 1, \dots, J_n; n = 1, \dots, N$  和  $g_{j_1 \dots j_N} \geq 0, \forall j_n = 1, \dots, J_n; n = 1, \dots, N$ 。

在一般张量的 Tucker 分解中，要求每一个因子矩阵的列向量之间正交，而在非负张量的 Tucker 分解中，这一要求不再合理。这一区别正是非负张量分解与高阶奇异值分解之间的本质不同。

根据非负张量  $\mathcal{X}$  的展开矩阵的不同，非负张量分解也可以使用矩阵化形式表示为

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_{\otimes}^{(n)T} \quad (10.7.6)$$

式中,  $\mathbf{X}_{(n)} \in \mathbb{R}_+^{I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N}$ ,  $\mathbf{G}_{(n)} \in \mathbb{R}_+^{J_n \times J_1 \cdots J_{n-1} J_{n+1} \cdots J_N}$ ,  $\mathbf{A}^{(n)} \in \mathbb{R}_+^{I_n \times J_n}$ ,  $\mathbf{A}_\otimes^{(n)} \in \mathbb{R}_+^{I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times J_1 \cdots J_{n-1} J_{n+1} \cdots J_N}$ , 并且

$$\mathbf{A}_\otimes^{(n)} = \begin{cases} \mathbf{A}^{(n-1)} \otimes \cdots \otimes \mathbf{A}^{(1)} \otimes \mathbf{A}^{(N)} \otimes \cdots \otimes \mathbf{A}^{(n+1)} & (\text{Kiers 矩阵化}) \\ \mathbf{A}^{(n+1)} \otimes \cdots \otimes \mathbf{A}^{(N)} \otimes \mathbf{A}^{(1)} \otimes \cdots \otimes \mathbf{A}^{(n-1)} & (\text{LMV 矩阵化}) \\ \mathbf{A}^{(N)} \otimes \cdots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \cdots \otimes \mathbf{A}^{(1)} & (\text{Kolda 矩阵化}) \end{cases} \quad (10.7.7)$$

表示除模式  $n$ -矩阵之外的其他  $n-1$  个因子矩阵的 Kronecker 积。

定义代价函数

$$J(\mathbf{A}^{(n)}, \mathbf{G}_{(n)}) = \frac{1}{2} \|\mathbf{X}_{(n)} - \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T}\|_F^2 \quad (10.7.8)$$

则有 [276]

$$\frac{\partial J(\mathbf{A}^{(n)}, \mathbf{G}_{(n)})}{\partial \mathbf{A}^{(n)}} = - \left( \mathbf{X}_{(n)} - \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \right) \left[ (\mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T) \otimes \mathbf{I} \right] \quad (10.7.9)$$

$$\frac{\partial J(\mathbf{A}^{(n)}, \mathbf{G}_{(n)})}{\partial \mathbf{G}_{(n)}} = - \left( \mathbf{X}_{(n)} - \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \right) \left( \mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)} \right) \quad (10.7.10)$$

其中利用了偏导公式

$$\frac{\partial \mathbf{WYZ}}{\partial \mathbf{Y}} = \mathbf{Z}^T \otimes \mathbf{W} \quad (10.7.11)$$

于是有以下的梯度算法

$$a^{(n)}(i, j) \leftarrow a^{(n)}(i, j) + \eta_A \left[ (\mathbf{X}_{(n)} - \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T}) [(\mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T) \otimes \mathbf{I}] \right]_{ij} \quad (10.7.12)$$

$$g_{(n)}(j, k) \leftarrow g_{(n)}(j, k) + \eta_G \left[ (\mathbf{X}_{(n)} - \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T}) (\mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)}) \right]_{kj} \quad (10.7.13)$$

若取步长分别为

$$\eta_A = \frac{a^{(n)}(i, j)}{\left[ \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T \otimes \mathbf{I} \right) \right]_{ij}}$$

$$\eta_G = \frac{g_{(n)}(j, k)}{\left[ \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)} \right) \right]_{kj}}$$

则梯度算法变为乘法更新算法 [276]

$$a^{(n)}(i, j) \leftarrow a^{(n)}(i, j) \frac{\left[ \mathbf{X}_{(n)} [(\mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T) \otimes \mathbf{I}] \right]_{ij}}{\left[ \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T \otimes \mathbf{I} \right) \right]_{ij}} \quad (10.7.14)$$

$$g_{(n)}(j, k) \leftarrow g_{(n)}(j, k) \frac{\left[ \mathbf{X}_{(n)} (\mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)}) \right]_{kj}}{\left[ \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)} \right) \right]_{kj}} \quad (10.7.15)$$

或用矩阵形式表示为

$$\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} * \left[ \left( \mathbf{X}_{(n)} [(\mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T) \otimes \mathbf{I}] \right) \oslash \left( \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \mathbf{G}_{(n)}^T \otimes \mathbf{I} \right) \right) \right] \quad (10.7.16)$$

$$\mathbf{G}_{(n)} \leftarrow \mathbf{G}_{(n)} * \left[ \left( \mathbf{X}_{(n)} (\mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)}) \right) \oslash \left( \mathbf{A}^{(n)} \mathbf{G}_{(n)} \mathbf{A}_\otimes^{(n)T} \left( \mathbf{A}_\otimes^{(n)} \otimes \mathbf{A}^{(n)} \right) \right) \right] \quad (10.7.17)$$

## 2. 非负张量 CP 分解的乘法算法

考虑  $N$  阶非负张量  $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$  的非负 CP 分解

$$x_{i_1 \cdots i_N} = \sum_{r=1}^R a_{i_1, r}^{(1)} a_{i_2, r}^{(2)} \cdots a_{i_N, r}^{(N)} \quad \text{subject to } a_{i_n, r}^{(n)} \geq 0, \forall i_n, r \quad (10.7.18)$$

并使得重构误差平方

$$\text{RE}(a_{i_1, r}^{(1)}, \dots, a_{i_N, r}^{(N)}) = \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \left( x_{i_1 \cdots i_N} - \sum_{r=1}^R a_{i_1, r}^{(1)} a_{i_2, r}^{(2)} \cdots a_{i_N, r}^{(N)} \right)^2 \quad (10.7.19)$$

最小化。

$N$  阶张量的非负 CP 分解也可等价写作

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{S}_{(n)}^T \in \mathbb{R}_+^{I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N} \quad (10.7.20)$$

式中,  $\mathbf{A}^{(n)} \in \mathbb{R}_+^{I_n \times R}$ ,  $\mathbf{S}_{(n)} \in \mathbb{R}_+^{I_1 \cdots I_{n-1} I_{n+1} \cdots I_N \times R}$ , 并且

$$\mathbf{S}_{(n)} = \begin{cases} \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)} \odot \mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)}, & \text{若 } \mathcal{X} \text{ 采用 Kiers 矩阵化} \\ \mathbf{A}^{(n+1)} \odot \cdots \odot \mathbf{A}^{(N)} \odot \mathbf{A}^{(1)} \odot \cdots \odot \mathbf{A}^{(n-1)}, & \text{若 } \mathcal{X} \text{ 采用 LMV 矩阵化} \\ \mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)}, & \text{若 } \mathcal{X} \text{ 采用 Kolda 矩阵化} \end{cases} \quad (10.7.21)$$

由于  $\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{S}_{(n)}^T$  写成了非负矩阵分解的标准形式  $\mathbf{X} = \mathbf{AS}$ , 所以非负矩阵的乘法算法可直接推广为  $N$  阶张量的非负 CP 分解的乘法算法

$$a^{(n)}(i_n, r) \leftarrow a^{(n)}(i_n, r) \frac{[\mathbf{X}_{(n)} \mathbf{S}_{(n)}]_{i_n, r}}{[\mathbf{A}^{(n)} \mathbf{S}_{(n)}^T \mathbf{S}_{(n)}]_{i_n, r}} \quad (10.7.22)$$

$$s_{(n)}^T(i, r) \leftarrow s_{(n)}^T(i, r) \frac{[\mathbf{A}^{(n)\top} \mathbf{X}_{(n)}]_{i, r}}{[\mathbf{A}^{(n)\top} \mathbf{A}^{(n)} \mathbf{S}_{(n)}^T]_{i, r}} \quad (10.7.23)$$

其中  $i_n = 1, \dots, I_n$ ;  $i = 1, \dots, I_1 \cdots I_{n-1} I_{n+1} \cdots I_N$  和  $r = 1, \dots, R$ 。上述元素形式的乘法算法也可以写成矩阵形式

$$\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} * \left[ (\mathbf{X}_{(n)} \mathbf{S}_{(n)}) \oslash (\mathbf{A}^{(n)} \mathbf{S}_{(n)}^T \mathbf{S}_{(n)}) \right] \quad (10.7.24)$$

$$\mathbf{S}_{(n)}^T \leftarrow \mathbf{S}_{(n)}^T * \left[ (\mathbf{A}^{(n)\top} \mathbf{X}_{(n)}) \oslash (\mathbf{A}^{(n)\top} \mathbf{A}^{(n)} \mathbf{S}_{(n)}^T) \right] \quad (10.7.25)$$

上述讨论可以总结为下列非负 CP 分解的乘法算法。

### 算法 10.7.1 $N$ 阶张量的非负 CP 分解的乘法算法

初始化 用非负随机变量初始化所有因子矩阵的元素  $a_{i_n, r}^{(n)}$ ,  $n = 1, \dots, N$ ;  $r = 1, \dots, R$ 。  
对所有  $i_1, \dots, i_N$ , 执行下列步骤:

步骤 1 针对下标  $i_n$ , 记  $I = \{i_1, \dots, i_{n-1}, i_{n+1}, \dots, i_N\}$ , 并定义  $y_{i_n, I} = x_{i_1 \cdots i_N}$  和  $m_{I, r} = a_{i_1, r}^{(1)} \cdots a_{i_{n-1}, r}^{(n-1)} a_{i_{n+1}, r}^{(n+1)} \cdots a_{i_N, r}^{(N)}$

步骤 2 利用下列规则更新矩阵  $\mathbf{A}^{(n)}$

$$\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} * [(\mathbf{Y}\mathbf{M}) \oslash (\mathbf{A}^{(n)}\mathbf{M}^T\mathbf{M})]$$

步骤 3 计算

$$K_r^{(n)} = \sqrt{\sum_{n=1}^N \left(a_{i_n, r}^{(n)}\right)^2}, \quad r = 1, \dots, R; n = 1, \dots, N$$

$$a_{i_n, r}^{(n)} \leftarrow a_{i_n, r}^{(n)} \frac{\left(\prod_{i=1}^N K_i^{(i)}\right)^{1/N}}{K_r^{(n)}}, \quad r = 1, \dots, R; n = 1, \dots, N$$

步骤 4 判断算法是否满足停止准则：若满足，则输出因子矩阵  $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R}$ ；否则，返回步骤 1，继续迭代，直至算法停止准则满足为止。

Welling 和 Weber 于 2001 年提出了正张量分解算法<sup>[512]</sup>。

### 10.7.2 非负张量分解的交替最小二乘算法

下面分别讨论非负张量的 Tucker 分解和 CP 分解的交替最小二乘方法。

#### 1. 非负张量 Tucker 分解的交替最小二乘方法

考虑非负张量分解<sup>[176]</sup>

$$\min_{\mathcal{G}, \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}} \frac{1}{2} \|\mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \cdots \times_N \mathbf{A}^{(N)} - \mathbf{X}\|_2^2 \quad (10.7.26)$$

或用矩阵形式写作

$$\min_{\mathcal{G}, \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}} \frac{1}{2} \sum_{n=1}^N \|\mathbf{A}^{(n)} \mathcal{G}_{(n)} \mathbf{A}_\otimes^n - \mathbf{X}_{(n)}\|_2^2 \quad (10.7.27)$$

式中  $\mathbf{A}_\otimes^n$  由式 (10.7.7) 给出。

当在第  $k+1$  步迭代，固定因子矩阵  $\mathbf{A}_{k+1}^{(1)}, \dots, \mathbf{A}_{k+1}^{(n-1)}, \mathbf{A}_k^{(n+1)}, \dots, \mathbf{A}_k^{(N-1)}$  和核心张量的水平展开矩阵  $\mathbf{G}_{(n)}^k$  为已知时，由式 (10.7.27) 易知， $\mathbf{A}_{k+1}^{(n)}$  的求解相当于求矩阵方程  $\mathbf{A}^{(n)} \mathcal{G}_{(n)} \mathbf{A}_\otimes^n \approx \mathbf{X}_{(n)}$  的最小二乘解，即有

$$\mathbf{A}_{k+1}^{(n)} = \mathcal{P}_+ \left( \mathbf{X}_{(n)} (\mathbf{G}_{(n)}^k \mathbf{S}_{k+1}^{(n)})^\dagger \right), \quad n = 1, \dots, N \quad (10.7.28)$$

式中， $\mathbf{B}^\dagger$  表示矩阵  $\mathbf{B}$  的 Moore-Penrose 广义逆矩阵， $[\mathcal{P}_+(\mathbf{C})]_{ij} = \max\{0, C_{ij}\}$  表示对矩阵元素的非负约束，并且

$$\mathbf{S}_{k+1}^{(n)} = \begin{cases} \mathbf{A}_{k+1}^{(n-1)} \otimes \cdots \otimes \mathbf{A}_{k+1}^{(1)} \otimes \mathbf{A}_k^{(N)} \otimes \cdots \otimes \mathbf{A}_k^{(n+1)} & (\text{Kiers 矩阵化}) \\ \mathbf{A}_k^{(n+1)} \otimes \cdots \otimes \mathbf{A}_k^{(N)} \otimes \mathbf{A}_{k+1}^{(1)} \otimes \cdots \otimes \mathbf{A}_{k+1}^{(n-1)} & (\text{LMV 矩阵化}) \\ \mathbf{A}_k^{(N)} \otimes \cdots \otimes \mathbf{A}_k^{(n+1)} \otimes \mathbf{A}_{k+1}^{(n-1)} \otimes \cdots \otimes \mathbf{A}_{k+1}^{(1)} & (\text{Kolda 矩阵化}) \end{cases} \quad (10.7.29)$$

为了更新  $\mathbf{G}_{(n)}^k$ ，对式 (10.7.27) 的等价矩阵方程  $\mathbf{A}^{(n)} \mathcal{G}_{(n)} \mathbf{A}_\otimes^n \approx \mathbf{X}_{(n)}$  应用向量化公式  $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B})$ ，得

$$\text{vec}(\mathbf{G}_{(n)}^{k+1}) = \mathcal{P}_+ \left( (\mathbf{S}_{k+1}^{(n)\top} \otimes \mathbf{A}_{k+1}^{(n)})^\dagger \text{vec}(\mathbf{X}_{(n)}) \right) \quad (10.7.30)$$

### 算法 10.7.2 $N$ 阶张量的非负 Tucker 分解的交替最小二乘算法

输入  $N$  阶张量  $\mathcal{X}$ 。

输出  $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}, n = 1, \dots, N$ 。

初始化  $\mathbf{A}_0^{(n)} \in \mathbb{R}^{I_n \times J_n}, n = 1, \dots, N$ , 并令  $k = 0$ 。

步骤 1 使用 Kiers 或 LMV 或 Kolda 方法将  $N$  阶张量水平展开为矩阵  $\mathbf{X}_{(n)}, n = 1, \dots, N$ 。

步骤 2 利用式 (10.7.29) 更新  $\mathbf{S}_{k+1}^{(n)}, n = 1, \dots, N$ 。

步骤 3 利用式 (10.7.28) 更新  $\mathbf{A}_{k+1}^{(n)}, n = 1, \dots, N$ 。

步骤 3 利用式 (10.7.30) 更新  $\text{vec}(\mathbf{G}_{(n)}^{k+1}), n = 1, \dots, N$ 。

步骤 4 若收敛条件满足或已达到某个预先规定的最大迭代次数, 则输出矩阵  $\mathbf{A}^{(n)}$  和向量  $\text{vec}(\mathbf{G}_{(n)}), n = 1, \dots, N$ , 并进而得到  $\mathbf{G}_{(n)}$  和核心张量  $\mathcal{G}$ ; 否则, 令  $k \leftarrow k + 1$ , 并返回步骤 2, 重复以上运算, 直到收敛条件满足或者达到最大迭代次数。

### 2. 非负张量 CP 分解的交替最小二乘算法

10.5 节介绍的张量的 CP 分解的交替最小二乘方法和正则交替最小二乘方法很容易分别推广为非负张量的 CP 分解的交替最小二乘方法和正则交替最小二乘方法。所增加的唯一运算就是对更新后的每一个因子矩阵  $\mathbf{A}_{k+1}^{(n)}$  加非负约束  $\mathcal{P}_+(\mathbf{A}_{k+1}^{(n)})$ , 其中  $[\mathcal{P}_+(\mathbf{A}_{k+1}^{(n)})]_{ij} = \max\{0, \mathbf{A}_{k+1}^{(n)}(i, j)\}$ 。

$N$  阶非负张量的 CP 分解可以写作非负矩阵分解的标准形式

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{S}^{(n)\top} \quad (10.7.31)$$

式中

$$\mathbf{S}^{(n)} = \begin{cases} \mathbf{A}^{(n-1)} \odot \dots \odot \mathbf{A}^{(1)} \odot \mathbf{A}^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} & (\text{Kiers 矩阵化}) \\ \mathbf{A}^{(n+1)} \odot \dots \odot \mathbf{A}_k^{(N)} \odot \mathbf{A}^{(1)} \odot \dots \odot \mathbf{A}^{(n-1)} & (\text{LMV 矩阵化}) \\ \mathbf{A}_k^{(N)} \odot \dots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \odot \dots \odot \mathbf{A}^{(1)} & (\text{Kolda 矩阵化}) \end{cases} \quad (10.7.32)$$

由式 (10.7.31) 直接得因子矩阵的最小二乘解为

$$\mathbf{A}^{(n)} = \mathbf{X}_{(n)} (\mathbf{S}^{(n)\top})^\dagger = \mathbf{X}_{(n)} \mathbf{S}^{(n)} (\mathbf{S}^{(n)} \mathbf{S}^{(n)\top})^\dagger \quad (10.7.33)$$

利用 Khatri-Rao 积的性质  $(\mathbf{A} \odot \mathbf{B})^\top = (\mathbf{A}^\top \mathbf{A} * \mathbf{B}^\top \mathbf{B})$  及 Hadamard 积的性质  $(\mathbf{C} * \mathbf{D})^\top = \mathbf{C}^\top * \mathbf{D}^\top$ , 可以将式 (10.7.33) 等价写作

$$\mathbf{A}^{(n)} = \mathbf{X}_{(n)} \mathbf{S}^{(n)} \mathbf{W}^\dagger \quad (10.7.34)$$

式中  $\mathbf{W} = \mathbf{S}^{(n)} \mathbf{S}^{(n)\top}$  可以表示为

$$\mathbf{W} = \begin{cases} \mathbf{A}^{(n-1)\top} \mathbf{A}^{(n-1)} * \dots * \mathbf{A}^{(1)\top} \mathbf{A}^{(1)} * \mathbf{A}^{(N)\top} \mathbf{A}^{(N)} * \dots * \mathbf{A}^{(n+1)\top} \mathbf{A}^{(n+1)} & (\text{Kiers 矩阵化}) \\ \mathbf{A}^{(n+1)\top} \mathbf{A}^{(n+1)} * \dots * \mathbf{A}^{(N)\top} \mathbf{A}^{(N)} * \mathbf{A}^{(1)\top} \mathbf{A}^{(1)} * \dots * \mathbf{A}^{(n-1)\top} \mathbf{A}^{(n-1)} & (\text{LMV 矩阵化}) \\ \mathbf{A}^{(N)\top} \mathbf{A}^{(N)} * \dots * \mathbf{A}^{(n+1)\top} \mathbf{A}^{(n+1)} * \mathbf{A}^{(n-1)\top} \mathbf{A}^{(n-1)} * \dots * \mathbf{A}^{(1)\top} \mathbf{A}^{(1)} & (\text{Kolda 矩阵化}) \end{cases} \quad (10.7.35)$$

### 算法 10.7.3 CP 分解的非负交替最小二乘算法 CP-NALS ( $\mathcal{X}, R$ )

输入  $N$  阶张量  $\mathcal{X}$  及因子个数  $R$ 。

输出 因子矩阵  $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R}, n = 1, \dots, N$ 。

初始化  $\mathbf{A}_0^{(n)} \in \mathbb{R}^{I_n \times R}, n = 1, \dots, N$ , 并令  $k = 0$ 。

步骤 1 使用 Kiers 或 LMV 或 Kolda 方法之一将  $N$  阶张量水平展开为矩阵  $\mathbf{X}_{(n)}, n = 1, \dots, N$ 。

步骤 2 利用式 (10.7.32) 计算  $\mathbf{S}_k^{(n)}$ , 其中  $\mathbf{A}^{(i)} = \mathbf{A}_{k+1}^{(i)}, i = 1, \dots, n-1$  和  $\mathbf{A}^{(i)} = \mathbf{A}_k^{(i)}, i = n+1, \dots, N$ 。

步骤 3 利用式 (10.7.35) 计算  $\mathbf{W}_k$ 。

步骤 4 更新因子矩阵  $\mathbf{A}_{k+1}^{(n)} = \mathbf{X}_{(n)} \mathbf{S}_k^{(n)} \mathbf{W}_k^\dagger$ 。

步骤 5 若收敛条件满足或已达到某个预先规定的最大迭代次数, 则输出  $\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}$ ; 否则, 令  $k \leftarrow k + 1$ , 并返回步骤 2, 重复以上运算, 直到收敛条件满足或者达到最大迭代步数。

利用 10.5 节所总结的 CP 分解的交替最小二乘和正则交替最小二乘算法之间的关系, 只要将算法 10.7.3 中步骤 4 的因子矩阵更新公式修正为

$$\mathbf{A}_{k+1}^{(n)} = (\mathbf{X}_{(n)} \mathbf{S}_k^{(n)} + \tau_k \mathbf{A}_k^{(n)}) (\mathbf{W}_k + \tau_k \mathbf{I})^{-1} \quad (10.7.36)$$

便可得到 CP 分解的正则非负交替最小二乘算法 CP-RNALS ( $\mathcal{X}, R$ )。其中,  $\tau_k$  为正则化参数。

## 本章小结

作为矩阵分析对多路阵列数据的推广, 本章讨论了张量分析的理论、方法与应用。首先, 介绍了张量的定义及其表示方法。其次, 讨论了张量的矩阵化与向量化, 建立了张量与矩阵、向量之间的直接关系。然后, 介绍了张量的基本代数运算, 主要包括张量的内积、范数、外积、 $n$ -模式积和秩。本章的重点是张量的信息挖掘的两种数学工具: Tucker 分解 (高阶奇异值分解) 和 CP 分解 (典范/平行因子分解)。本章还介绍了多路数据分析的预处理和后处理有关方法。作为非负矩阵的推广, 本章最后介绍了张量的非负 Tucker 分解和非负 CP 分解的交替最小二乘算法及其改进 (正则交替最小二乘算法)。

## 习题

### 10.1 已知三阶张量 $\mathcal{X} \in \mathbb{R}^{3 \times 4 \times 2}$ 的正面切片矩阵为

$$\mathbf{X}_1 = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix}, \quad \mathbf{X}_2 = \begin{bmatrix} 15 & 18 & 21 & 24 \\ 16 & 19 & 22 & 25 \\ 17 & 20 & 23 & 26 \end{bmatrix}$$

令  $U = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$ , 求  $\mathcal{Y} = \mathcal{X} \times_1 U$  的正面切片矩阵  $\mathbf{Y}_1$  和  $\mathbf{Y}_2$ 。

**10.2 证明:**

$$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A}\mathbf{G}^{(P \times QR)}(\mathbf{C} \otimes \mathbf{B})^T$$

**10.3 证明:**

$$x_{i,(k-1)J+j}^{(I \times JK)} = x_{ijk} \Leftrightarrow \mathbf{X}^{(I \times JK)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::K}] = \mathbf{A}\mathbf{G}^{(P \times QR)}(\mathbf{C} \otimes \mathbf{B})^T$$

**10.4 令**

$$\mathbf{X}_{i::} = \mathbf{b}_1 \mathbf{c}_1^T a_{i1} + \dots + \mathbf{b}_R \mathbf{c}_R^T a_{iR}$$

证明水平展开的 CP 分解

$$\mathbf{X}^{(J \times KI)} = [\mathbf{X}_{1::}, \dots, \mathbf{X}_{I::}] = \mathbf{B}(\mathbf{A} \odot \mathbf{C})^T$$

和垂直展开的 CP 分解

$$\mathbf{X}^{(IJ \times K)} = \begin{bmatrix} \mathbf{X}_{1::} \\ \vdots \\ \mathbf{X}_{I::} \end{bmatrix} = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T$$

**10.5 令**

$$\mathbf{X}_{::j} = \mathbf{a}_1 \mathbf{c}_1^T b_{j1} + \dots + \mathbf{a}_R \mathbf{c}_R^T b_{jR}$$

证明: 水平展开的 CP 分解为

$$\mathbf{X}^{(K \times IJ)} = [\mathbf{X}_{::1}, \dots, \mathbf{X}_{::J}] = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$$

垂直展开的 CP 分解为

$$\mathbf{X}^{(JK \times I)} = \begin{bmatrix} \mathbf{X}_{::1} \\ \vdots \\ \mathbf{X}_{::J} \end{bmatrix} = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T$$

**10.6 令**

$$\mathbf{X}_{::k} = \mathbf{a}_1 \mathbf{b}_1^T c_{k1} + \dots + \mathbf{a}_R \mathbf{b}_R^T c_{kR}$$

证明: 水平展开的 CP 分解为

$$\mathbf{X}^{(J \times KI)} = [\mathbf{X}_{::1}^T, \dots, \mathbf{X}_{::K}^T] = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T$$

垂直展开的 CP 分解为

$$\mathbf{X}^{(KI \times J)} = \begin{bmatrix} \mathbf{X}_{::1} \\ \vdots \\ \mathbf{X}_{::K} \end{bmatrix} = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T$$

## 参 考 文 献

- [1] Abatzoglos T J, Mendel J M, and Harada G A. The constrained total least squares technique and its applications to harmonic superresolution. *IEEE Trans. Signal Processing*, 1991, 39: 1070 ~ 1087.
- [2] Abbott D. *The Biographical Dictionary of Sciences: Mathematicians*. New York: P. Bedrick Books, 1986.
- [3] Abed-Meraim K, Chkeif A, Hua Y. Fast orthonormal PAST algorithm. *IEEE Signal Processing Letters*, 2000, 7(3): 60 ~ 62.
- [4] Abraham R, Marsden J E, Ratiu T. *Manifolds, Tensor Analysis, and Applications*. New York: Addison-Wesley, 1983.
- [5] Absil P A, Mahony R, Sepulchre R, Van Dooren P. Grassmann-Rayleigh quotient iteration for computing invariant subspace. *SIAM Review*, 2002, 44(1): 57 ~ 73.
- [6] Acar E, Camtepe S A, Krishnamoorthy M, Yener B. Modeling and multiway analysis of chatroom tensors. In Proc. of IEEE International Conference on Intelligence and Security Informatics. Springer, Germany, 2005, 256 ~ 268.
- [7] Acar E, Camtepe S A, and Yener B. Collective sampling and analysis of high order tensors for chatroom communications. In Proc. of IEEE International Conference on Intelligence and Security Informatics. Springer, Germany, 2006, 213 ~ 224.
- [8] Acar E, Aykut-Bingo C, Bingo H, Bro R, Yener B. Multiway analysis of epilepsy tensors. *Bioinformatics*, 2007, 23: i10 ~ i18.
- [9] Acar E, Yener B. Unsupervised multiway data analysis: A literature survey. *IEEE Transactions on Knowledge and Data Engineering*, 2009, 21(1): 6 ~ 20.
- [10] Acar R, Vogel C R. Analysis of bounded variation penalty methods for ill-posed problems. *Inverse Problems*, 1994, 10: 1217 ~ 1229.
- [11] Adamyan V M, Arov D Z. A general solution of a problem in linear prediction of stationary processes. *Theory Probab Appl*, 1968, 13: 294 ~ 407.
- [12] Adib A, Moreau E, Aboutajdine D. Source separation contrasts using a reference signal. *IEEE Signal Processing Letters*, 2004, 11(3): 312 ~ 315.
- [13] Afriat S N. Orthogonal and oblique projectors and the characteristics of pairs of vector spaces. *Math. PROC. Cambridge Philos. Soc.*, 1957, 53: 800 ~ 816.
- [14] Aitken A C. *Determinants and Matrices*. 4th ed. Edinburgh: Oliver and Boyd, 1946.
- [15] Alter O, Brown P O, Botstein D. Generalized singular value decomposition for comparative analysis of genome-scale expression data sets of two different organisms. *Proc. of the National Academy of Sciences of the United States of America*, 2003, 100(6): 3351 ~ 3356.
- [16] Amari S. Natural gradient works efficiently in learning. *Neural Computation*, 1998, 10: 251 ~ 276.
- [17] Amari S, Nagaoka H. *Methods of Information Geometry*. New York: Oxford University Press, 2000.

- [18] Ammar G S, Gragg W B. Superfast solution of real positive definite Toeplitz systems. In: P N Datta, et al eds. *Linear Algebra in Signals, Systems and Control*. SIAM, 1988, 107~125.
- [19] Anderson G W, Guionnet A, Zeitouni O. *An Introduction to Random Matrices*. Cambridge University Press, 2009.
- [20] Andrews H, Hunt B. *Digital Image Restoration*. Cliffside, NJ: Prentice-Hall, 1977.
- [21] Antman S. The influence of elasticity in analysis: Modern developments. *Bulletin of the American Mathematical Society*, 1983, 9(3): 267~291.
- [22] Anton H, Rorres C. *Elementary Linear Algebra*. 8th ed. New York: John Wiley & Sons, Inc, 2000.
- [23] Aronszajn N. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 1950, 68: 800~816.
- [24] Arrow K, Hurwicz L, Uzawa H. *Studies in Nonlinear Programming*. Stanford, CA: Stanford University Press, 1958.
- [25] Autonne L. Sur les groupes linéaires, réelles et orthogonaux. *Bull Soc. Math., France*, 1902, 30: 121~133.
- [26] Auslender A. Asymptotic properties of the Fenchel dual functional and applications to decomposition problems. *J. Optimization Theory and Applications*, 1992, 73(3): 427~449.
- [27] Bader B W, Harshman R A, Kolda, T G. Temporal analysis of social networks using three-way dedicom. Technical Report SAND2006-2161, Sandia National Laboratories, 2006.
- [28] Banachiewicz T. Zur Berechnung der Determinanten, wie auch der Inverse, und zur darauf basierten Auflösung der Systeme linearer Gleichungen. *Acta Astronomica, Sér C*, 1937, 3: 41~67.
- [29] Barabell A J. Improving the resolution performance of eigenstructure based direction-fading algorithms. *Proc. ICASSP-83*, 1983, Boston, 336~339.
- [30] Barbarossa S, Daddio E, Galati G. Comparison of optimum and linear prediction technique for clutter cancellation. *Proc IEE, Part F*, 1987, 134: 277~282.
- [31] Bapat R. *Nonnegative Matrices and Applications*. Cambridge University Press, 1997.
- [32] Barnett S. *Matrices: Methods and Applications*. Oxford: Clarendon Press, 1990.
- [33] Basri R, JACOBS D. Lambertian reflectance and linear subspaces. *IEEE Trans. Patt. Anal. Mach. Intel.*, 2003, 25(2): 218~233.
- [34] Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sciences*, 2009, 2(1): 183~202.
- [35] Behrens R T, Scharf L L. Signal processing applications of oblique projection operators. *IEEE Trans. Signal Processing*, 1994, 42(6): 1413~1424.
- [36] Bellman R. *Introduction to Matrix Analysis*. 2nd ed. New York: McGraw-Hill, 1970.
- [37] Belochrani A, Abed-Meraim K, Cardoso J F, Moulines E. A blind source separation technique using second-order statistics. *IEEE Trans. Signal Processing*, 1997, 45(2): 434~444.
- [38] Beltrami E. Sulle funzioni bilineari, Giornale di Mathematiche ad Uso Studenti Delle Universita. 1873, 11: 98~106. An English translation by D Boley is available as University of Minnesota, Department of Computer Science, Technical Report 90~37, 1990.
- [39] Ben-Israel H, Greville T N E. *Generalized Inverses: Theory and Applications*. New York: Wiley-Interscience, 1974.
- [40] Berberian S K. *Linear Algebra*. New York: Oxford University Press, 1992.

- [41] Berge J M F T. Convergence of PARAFAC preprocessing procedures and the Deming-Stephan method of iterative proportional fitting. In Multiway Data Analysis (Eds. Coppi R, Bolasco S). Amsterdam: Elsevier, 1989, 53~63.
- [42] Berge J M F T, Sidiropolous N D. On uniqueness in CANDECOMP/PARAFAC, Psychometrika, 2002, 67: 399~409.
- [43] Berry M W, Browne M, Langville A N, Pauca V P, Plemmons R J. Algorithms and applications for approximate nonnegative matrix factorization. Computational Statistics & Data Analysis, 2007, 52: 155~173.
- [44] Bertsekas D P. Multiplier methods: A survey. Automatica, 1976, 12: 133~145.
- [45] Bertsekas D P. Nonlinear Programming, 2nd ed., Belmont, MA: Athena Scientific, 1999.
- [46] Bertsekas D P, Nedic A, Ozdaglar A. Convex Analysis and Optimization. Belmont, MA: Athena Scientific, 2003.
- [47] Bertsekas D P, Tseng P. Partial proximal minimization algorithms for convex programming. SIAM J. Optimizations, 1994, 4(3): 551~572.
- [48] Biemvenu G, Kopp L. Principale la goniomgraveetre passive adaptive. Proc 7'eme Colloque GRESTIT, Nice Frace, 1979, 106/1~106/10.
- [49] Björck A, Bowie C. An iterative algorithm for computing the best estimate of an orthogonal matrix. SIAM J Num Anal, 1971, 8: 358~364.
- [50] Blumensath T, Davis M E. Gradient pursuits. IEEE Trans. Signal Processing, 2008, 56(6): 2370~2382.
- [51] Bodewig E. Matrix Calculus. 2nd ed. Amsterdam: North-Holland, 1959.
- [52] Boot J. Computation of the generalized inverse of singular or ractangular matrices. Amer Math Monthly, 1963, 70: 302~303.
- [53] Boyd S. EE364b, Stanford University, Spring quarter 2010~11, 2010.
- [54] Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning, 2010, 3(1): 1~122.
- [55] Boyd S, Vandenberghe L. Convex Optimization. Cambridge, UK: Cambridge Univ. Press, 2004.
- [56] Boyd S, Vandenberghe L. Subgradients. Notes for EE364b, Stanford University, Winter 2006-2007, April 13, 2008.
- [57] Bramble J, Pasciak J. A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems. Mathematics of Computation, 1988, 50(181): 1~17.
- [58] Brandwood D H. A complex gradient operator and its application in adaptive array theory. Proc Inst Elec Eng, 1983, 130: 11~16.
- [59] Branham R L. Total least squares in astronomy. In: Recent Advances in Total Least Squares Techniques and Error-in-Variables Modeling (Van Huffel S ed). Philadelphia, PA: SIAM, 1997.
- [60] Bregman L M. The method of successive projection for finding a common point of convex sets. Soviet Math. Dokl., 1965, 6: 688~692.

- [61] Brewer J W. Kronecker products and matrix calculus in system theory. *IEEE Trans. Circuits and Systems*, 1978, 25: 772 ~ 781.
- [62] Bridges T J, Morris P J. Differential eigenvalue problems in which the parameters appear nonlinearly. *J. Comput. Phys.*, 1984, 55: 437 ~ 460.
- [63] Bro R. PARAFAC: Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems*, 1997, 38: 149 ~ 171.
- [64] Bro R, de Jong S. A fast non-negativity constrained least squares algorithm. *J. Chemometrics* 1997, 11(5): 393 ~ 401.
- [65] Bro R, Harshman R A, Sidiropoulos N D. Modeling multi-way data with linearly dependent loadings. Technical Report 2005-176, KVL, 2005.
- [66] Bro R. Multiway analysis in the food industry: Models, algorithms and applications, Doctoral dissertation, University of Amsterdam, 1998.
- [67] Bro R, Sidiropoulos N. Least squares algorithms under unimodality and non-negativity constraints. *J. Chemometrics* 1998; 12 (4): 223 ~ 247.
- [68] Brockwell P J, Davis R A. *Time Series: Theory and Methods*. New York: Springer-Verlag, 1987.
- [69] Brookes M. Matrix Reference Manual 2004. Available at <http://www.ee.ic.ac.uk/hp/staff/dmb/matrix/intro.html>, 2005.
- [70] Bunch J. Stability of methods for solving Toeplitz systems of equations. *SIAM J Sci Stat Comput*, 1985, 6: 349 ~ 364.
- [71] Bunse-Gerstner A. An analysis of the HR algorithm for computing the eigenvalues of a matrix. *Linear Algebra and Its Applications*, 1981, 35: 155 ~ 173.
- [72] Byrd R H, Hribar M E, Nocedal J. An interior point algorithm for large scale nonlinear programming. *SIAM Journal on Optimization*, 1999, 9(4): 877 ~ 900.
- [73] Byrne C, Censor Y. Proximity function minimization using multiple Bregman projections, with applications to split feasibility and Kullback-Leibler distance minimization. *Annals of Operations Research*, 2001, 105: 77 ~ 98.
- [74] Cadzow J A. Spectral estimation: An overdetermined rational model equation approach. *Proc IEEE*, 1982, 70: 907 ~ 938.
- [75] Cai J -F, Candes E J, Shen Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 2010, 20(4): 1956 ~ 1982.
- [76] Cai J -F, Shen Z. Fast singular value thresholding without singular value decomposition, 2010, available at <ftp://ftp.math.ucla.edu/pub/camreport/cam10-24.pdf>.
- [77] Cai T T, Wang L, Xu G. New bounds for restricted isometry constants. *IEEE Trans. Information Theory*, 2010, 56(9): 4388 ~ 4394.
- [78] Candès E, Romberg J, Tao T. Stable signal recovery from incomplete and inaccurate information. *Commun. Pure Appl. Math.*, 2005, 59: 1207 ~ 1233.
- [79] Candès E J, Tao T. Near optimal signal recovery from random projections: Universal encoding strategies. *IEEE Trans. Inform. Theory*, 2006, 52(12): 5406 ~ 5425.
- [80] Candès E J, Romberg J, Tao T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, 2006, 52(2): 489 ~ 509.

- [81] Candès E J, Romberg J, Tao T. Stable signal recovery from incomplete and inaccurate measurements. *Comm. Pure Appl. Math.*, 2006, 59(8): 1207 ~ 1223.
- [82] Candès E J, Tao T. The Dantzig selector: Statistical estimation when  $p$  is much larger than  $n$ . *Ann. Statist.* 2007 [Online]. Available at <http://arxiv.org/abs/math.ST/0506081>.
- [83] Candès E J, Romberg J. Sparsity and incoherence in compressive sampling. *Inverse Prob.*, 2007, 23(3): 969 ~ 985.
- [84] Candès E J. The restricted isometry property and its implications for compressed sensing. *C. R. l' Academie des Sciences, Ser. I*, 2008, 346: 589 ~ 592.
- [85] Candès E J, Wakin M B. A introduction to compressive sampling. *IEEE Signal Processing Magazine*, 2008, 25(3): pp.21 ~ 30.
- [86] Candès E J, Recht B. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 2009, 9: 717 ~ 772.
- [87] Candès E J, Plan Y. Matrix completion with noise. *Proc. IEEE*, 2010, 98(6): 925 ~ 936.
- [88] Candès E J, Li X, Ma Y, Wright J. Robust principal component analysis? *J. the ACM*, 2011, 58(3): Article 11: 1-37.
- [89] Cardoso J F, Souloumiac A. Blind beamforming for non-Gaussian signals. *Proc IEE, F*, 1993, 40(6): 362 ~ 370.
- [90] Cordoso J F, Souloumiac A. Jacobi angles for simultaneous diagonalization. *SIAM J. Matrix Analysis Appl.*, 1996, 17(1): 161 ~ 164.
- [91] Carroll C W. The created response surface technique for optimizing nonlinear restrained systems. *Oper. Res.*, 1961, 9: 169 ~ 184.
- [92] Carroll J D, Arabie P. Multidimensional scaling. *Annual Review of Psychology*, 1980, 31: 438 ~ 457.
- [93] Carroll, J D, Chang J. Analysis of individual differences in multidimensional scaling via an N way generalization of "Eckart-Young" decomposition. *Psychometrika*, 1970, 35: 283 ~ 319.
- [94] Cattell R B. Parallel proportional profiles and other principles for determining the choice of factors by rotation. *Psychometrika*, 1944, 9: 267 ~ 283.
- [95] Champagne B. Adaptive eigendecomposition of data covariance matrices based on first-order perturbations. *IEEE Trans. Signal Processing*, 1994, 42: 2758 ~ 2770.
- [96] Chang C -I, Du Q. Estimation of number of spectrally distinct signal sources in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.*, 2004, 42(3): 608 ~ 619.
- [97] Chan T F. An improved algorithm for computing the singular value decomposition. *ACM Trans. Math. Software*, 1982, 8: 72 ~ 83.
- [98] Chan Y T, Wood J C. A new order determination technique for ARMA processes. *IEEE Trans. Acoust, Speech, Signal Processing*, 1984, 32: 517 ~ 521.
- [99] Chan R H, Ng M K. Conjugate gradient methods for Toeplitz systems. *SIAM Review*, 1996, 38(3): 427 ~ 482.
- [100] Chandrasekaran V, Sanghavi S, Parrilo P A, Wilsky A S. Rank-sparsity incoherence for matrix decomposition. *SIAM J. Optim.* 2011, 21(2): 572 ~ 596.
- [101] Chatelin F. *Eigenvalues of Matrices*. New York: Wiley, 1993
- [102] Chen B, Petropulu A P. Frequency domain blind MIMO system identification based on second- and higher order statistics. *IEEE Trans. Signal Processing*, 2001, 49(8): 1677 ~ 1688.

- [103] Chen H, Sarkar T K, Brule J, Dianat S A. Adaptive spectral estimation by the conjugate gradient method. *IEEE Trans Acoust, Speech, Signal Processing*, 1986, 34(2): 272~284.
- [104] Chen S S, Donoho D L, Saunders M A. Atomic decomposition by basis pursuit. *SIAM J. Science Computations*, 1998, 20(1): 33~61.
- [105] Chen S S, Donoho D L, Saunders M A. Atomic decomposition by basis pursuit. *SIAM Rev.*, 2001, 43(1): 129~159.
- [106] Chen W, Chen M, Zhou J. Adaptively regularized constrained total least-squares image restoration. *IEEE Trans. Image Processing*, 2000, 9(4): 588~596.
- [107] Chen X, Pan W, Kwok J T, Carbonell J G. Accelerated gradient method for multi-task sparse learning problem. In Proc. Ninth IEEE International Conference on Data Mining, 2009, 746~751.
- [108] Chua L O. Dynamic nonlinear networks: State-of-the-art. *IEEE Trans. Circuits and Systems*, 1980, 27: 1024~1044.
- [109] Cichocki A, Amari S -I. Families of alpha- beta- and gamma-divergences: Flexible and robust measures of similarities. *Entropy*, 2010, 12(6): 1532~1568.
- [110] Cichocki A, Cruces S, Amari S -I. Generalized Alpha-Beta divergences and their application to robust nonnegative matrix factorization. *Entropy*, 2011, 13: 134~170.
- [111] Cichocki A, Lee H, Kim Y -D, Choi S. Non-negative matrix factorization with  $\alpha$ -divergence. *Pattern Recognition Letters*, 2008, 29: 1433~1440.
- [112] Cichocki A, Zdunek R, Amari S -I. Nonnegative matrix and tensor factorization. *IEEE Signal Processing Magazine*, 2008, 25(1): 142~145.
- [113] Cichocki A, Zdunek R, Amari S -I. Csiszar's divergences for nonnegative matrix factorization: Family of new algorithms. In *Lecture Notes in Computer Science*; Springer: Charleston, SC, USA, 2006, 3889: 32~39.
- [114] Cichocki A, Zdunek R, Amari S -I. Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization. *Springer LNCS*, 2007, 4666: 169~176.
- [115] Cichocki A, Zdunek R, Choi S, Plemmons R, Amari S -I. Novel multi-layer nonnegative tensor factorization with sparsity constraints. *Springer LNCS*, 2007, 4432: 271~280.
- [116] Cichocki A, Zdunek R, Phan A -H, Amari S -I. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation*. Chichester, UK: Wiley, 2009.
- [117] Cirrincione G, Cirrincione M, Herault J, et al. The MCA EXIN neuron for the minor component analysis. *IEEE Trans. Neural Networks*, 2002, 13(1): 160~187.
- [118] Clark J V, Zhou N, Pister K S J. Modified nodal analysis for MEMS with multi-energy domains. In: *International Conference on Modeling and Simulation of Microsystems, Semiconductors, Sensors and Actuators*. San Diego, USA, 2000; also available at <http://www-bsac.EECS.Berkely.EDU/-cfm/publication.html>
- [119] Cline R E. Note on the generalized inverse of the product of matrices. *SIAM Review*, 1964, 6: 57~58.
- [120] Cline A K, Moler C B, Stewart G W, Wilkinson J H. An estimate for the condition number of a matrix. *SIAM J Numer Anal*, 1979, 16: 368~375.

- [121] Coifman R, Geshwind F, Meyer Y. Noiselet. *Applied and Computational Harmonic Analysis*, 2001, 10(1): 27 ~ 44.
- [122] Combettes P L, Pesquet J.-C. Proximal splitting methods in signal processing. In: *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, New York: Springer, 2011, 185 ~ 212.
- [123] Comon P, Golub G, Lim L -H, Mourrain B. Symmetric tensors and symmetric tensor rank. SM Technical Report 06-02, Stanford University, 2006.
- [124] Comon P, Golub G, Lim L.-H, Mourrain B. Symmetric tensors and symmetric tensor rank. *SIAM J. Matrix Anal. Appl.*, 2008, 30(3): 1254 ~ 1279.
- [125] Comon, P, Moreau, E. Blind MIMO equalization and joint-diagonalization criteria. Proc 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01), 2001, 5: 2749 ~ 2752.
- [126] Dai W, Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inform. Theory*, 2009, 55(5): 2230 ~ 2249.
- [127] Davis P. *Circular Matrices*. New York: John Wiley, 1979.
- [128] Davis G. A fast algorithm for inversion of block Toeplitz Signal Processing, 1995, 43: 3022 ~ 3025.
- [129] Davis G, Mallat S, Avellaneda M. Adaptive greedy approximation. *J. Constr. Approx.*, 1997, 13(1): 57 ~ 98.
- [130] Davila C E. A subspace approach to estimation of autoregressive parameters from noisy measurements, *IEEE Trans. Signal Processing*, 1998, 46: 531 ~ 534.
- [131] Decell Jr. H P. An application of the Cayley-Hamilton theorem to generalized matrix inversion. *SIAM Review*, 1965, 7(4): 526 ~ 528.
- [132] Delsarte P, Genin Y. The split Levinson algorithm. *IEEE Trans. Acoust, Speech, Signal Processing*, 1986, 34: 471 ~ 478.
- [133] Delsarte P, Genin Y. On the splitting of classical algorithms in linear prediction theory. *IEEE Trans. Acoust, Speech, Signal Processing*, 1987, 35: 645 ~ 653.
- [134] Dembo R S, Steihaug T. Truncated-Newton algorithms for large-scale unconstrained optimization. *Math Programming*, 1983, 26: 190 ~ 212.
- [135] Demoment G. Image reconstruction and restoration: Overview of common estimation problems. *IEEE Trans. Acoust, Speech, Signal Processing*, 1989, 37(12): 2024 ~ 2036.
- [136] Dewester S, Dumains S, Landauer T, Furnas G, Harshman R. Indexing by latent semantic analysis. *J. Soc. Inf. Sci.*, 1990, 41(6): 391 ~ 407.
- [137] Doclo S, Moonen M. GSVD-based optimal filtering for single and multimicrophone speech enhancement. *IEEE Trans. Signal Processing*, 2002, 50(9): 2230 ~ 2244.
- [138] Donoho D L, Huo X. Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inform. Theory*, 2001, 47(7): 2845 ~ 2862.
- [139] Donoho D L, Elad M. Optimally sparse representations in general (non-orthogonal) dictionaries via  $\ell^1$  minimization. *Proc. Nat. Acad. Sci.* 2003, 100(5): 2197 ~ 2202.
- [140] Donoho D L. Compressed sensing, *IEEE Trans. Information Theory*, 2006, 52(4): 1289 ~ 1306.

- [141] Donoho D L. For most large underdetermined systems of linear equations, the minimal  $\ell^1$  solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*. 2006, vol.LIX: 797~829.
- [142] Donoho D L, Elad M, Temlyakov V N. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inform. Theory*, 2006, 52(1): 6~18.
- [143] Donoho D L, Tsaig T, Drori T, Starck J -L. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit (StOMP). Stanford Univ., Palo Alto, CA, Stat. Dept. Tech. Rep. 2006~02, Mar. 2006.
- [144] Donoho D L, Tsaig Y. Fast solution of  $l_1$ -norm minimization problems when the solution may be sparse. *IEEE Trans. Inform. Theory*, 2008, 54(11): 4789~4812.
- [145] Drmac Z. Accurate computation of the product-induced singular value decomposition with applications. *SIAM J Numer Anal*, 1998, 35(5): 1969~1994.
- [146] Drmac Z. A tangent algorithm for computing the generalized singular value decomposition. *SIAM J Numer Anal*, 1998, 35(5): 1804~1832.
- [147] Drmac Z. New accurate algorithms for singular value decomposition of matrix triplets. *SIAM J Matrix Anal Appl*, 2000, 21(3): 1026~1050.
- [148] Duchi, J C, Agarwal A, Wainwright M J. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Trans. on Automatic Control*, 2012, 57(3): 592~606.
- [149] Duda R O, Hart P E. Pattern Classification and Scene Analysis. New York: Wiley, 1973.
- [150] Duncan W J. Some devices for the solution of large sets of simultaneous linear equations. *The London, Edinburgh, and Dublin Philosophical Magazine and J. Science*, Seventh Series, 1944, 35: 660~670.
- [151] Eckart C, Young G. The approximation of one matrix by another of lower rank. *Psychometrika*, 1936, 1: 211~218.
- [152] Eckart C, Young G. A Principal axis transformation for non-Hermitian matrices. *Null Amer. Math. Soc.*, 1939, 45: 118~121.
- [153] Edelman A, Arias T A, Smith S T. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Analysis, Applications*, 1998, 20(2): 303~353.
- [154] Efron B, Hastie T, Johnstone I, Tibshirani R. Least angle regression. *Ann. Statist.*, 2004, 32: 407~499.
- [155] Efroymson G, Steger A, Steeinberg S. A matrix eigenvalue problem. *SIAM Review*, 1980, 22(1): 99~100.
- [156] Elad M, Matalon B, Zibulevsky M. Image denoising with shrinkage and redundant representations, in Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition—CVPR' 2006, New York, 2006.
- [157] Eldar Y C, Oppenheim A V. MMSE whitening and subspace whitening. *IEEE Trans. Inform. Theory*, 2003, 49(7): 1846~1851.
- [158] Eldar Y C, Opeenheim A V. Orthogonal and projected orthogonal matched filter detection. *Signal Processing*, 2004, 84: 677~693.
- [159] Estienne F, Matthijs N, Massart D L, Ricoux, P, Leibovici D. Multi-way modelling of

- high-dimensionality electroencephalographic data. *Chemometrics Intell. Lab. Systems*, 2001, 58(1): 59~72.
- [160] Facchinei F, Pang J -S. *Finite-Dimensional Variational Inequalities and Complementarity Problem*. New York: Springer, 2003.
- [161] Faddeev D K, Faddeeva V N. *Computational Methods of Linear Algebra*. San Francisco: W H Freedman Co, 1963.
- [162] Farina A, Golino G, Timmoneri L, Comparison between LS and TLS in adaptive processing for radar systems. *IEE P-Radar Sonar Nav*, 2003, 150(1): 2~6.
- [163] Fernando K V, Hammarling S J. A product induced singular value decomposition (PSVD) for two matrices and balanced relation. In: Proc Conference on Linear Algebra in Signals, Systems and Controls, Society for Industrial and Applied Mathematics (SIAM). PA: Philadelphia, 1988, 128~140.
- [164] Fiacco A V, McCormick G P. *Nonlinear Programming: Sequential Unconstrained minimization Techniques*. New York: Wiley, 1968; or Classics Appl. Math. 4, SIAM, Philadelphia, PA, 1990. Reprint of the 1968 original.
- [165] Field D J. Relation between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Amer. A*, 1984, 4: 2370~2393.
- [166] Figueiredo M A T, Nowak R D. An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Processing*, 2003, 12: 906~916.
- [167] Flanigan F. *Complex Variables: Harmonic and Analytic Functions* (2nd Edition). New York: Dover Publications, 1983.
- [168] Fletcher R. Conjugate gradient methods for indefinite systems. In: Watson G A. ed. *Proc Dundee Conf on Num Anal*. New York: Springer-Verlag, 1975, 73~89.
- [169] Fletcher R. *Practical Methods of Optimization*, 2nd ed., New York: John Wiley & Sons, 1987.
- [170] Flury B N. Common principal components in k groups. *J. Amer. Statist. Assoc.*, 1984, 79: 892~897.
- [171] Forsgren A, Gill P E, Wright M H. Interior methods for nonlinear optimization. *SIAM Review*, 2002, 44: 525~597.
- [172] Forsgren A. Inertia-controlling factorization for optimization algorithms. *Appl. Numer. Math.*, 2002, 43: 91~107.
- [173] Foucart S, Lai M -J. Sparsest solutions of underdetermined linear systems via  $l_q$ -minimization for  $0 < q \leq 1$ . *Appl. Comput. Harmonic Anal.*, 2009, 26(3): 395~407.
- [174] Foygel R, Srebro N. Concentration-based guarantees for low-rank matrix reconstruction. Available at [http://olt2011.sztaki.hu/colt2011\\_submission\\_90.pdf](http://olt2011.sztaki.hu/colt2011_submission_90.pdf).
- [175] Frankel T. *The Geometry of Physics: An Introduction* (with corrections and additions), Cambridge University Press, 2001.
- [176] Friedlander M P, Hatz K. Computing nonnegative tensor factorizations. Available at <http://www.optimization-online.org/DBHTML/2006/10/1494.html>.
- [177] Fuhrmann D R. An algorithm for subspace computation with applications in signal processing. *SIAM J. Matrix Anal. Appl.*, 1988, 9: 213~220.
- [178] Fukunaga K. *Statistical Pattern Recognition*. 2nd ed. New York: Academic Press, 1990.

- [179] Gabay D, Mercier B. A dual algorithm for the solution of nonlinear variational problems via finite element approximations. *Computers and Mathematics with Applications*. 1976, 2: 17~40.
- [180] Galatsanou N P, Katsaggelos A K. Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation. *IEEE Trans. Image Processing*, 1992, 1(3): 322~336.
- [181] Gander W, Golub G H, Von Matt U. A constrained eigenvalue problem. *Linear Algebra Appl.*, 1989, 114-115: 815~839.
- [182] Gilbert A C, Muthukrishnan M, Strauss M J. Approximation of functions over redundant dictionaries using coherence. *Proc. 14th Annu. ACM-SIAM Symp. Discrete Algorithms*, Jan. 2003.
- [183] Glowinski R, Marrocco A. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par penalisation-dualité, d'une classe de problèmes de Dirichlet non linéaires. *Revue Française d'Automatique, Informatique, et Recherche Opérationnelle*, 1975, 9: 41~76.
- [184] Glowinski R, Tallec P Le. *Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics*. Philadelphia, PA: SIAM Studies in Applied Mathematics, 1989.
- [185] Gantmacher F R. *Applications of the Theory of Matrices*. New York: Interscience, 1959.
- [186] Gantmacher F R. *The Theory of Matrices*. Chelsea Publishing, 1977.
- [187] Gersch W. Estimation of the autoregressive parameters of a mixed autoregressive moving-averaging time series. *IEEE Trans. Automatic Control*, 1970, 15: 583~585.
- [188] Gersho A, Gray R M. *Vector Quantization and Signal Compression*. Kluwer Acad. Press, 1992.
- [189] Gillies A W. On the classification of matrix generalized inverse. *SIAM Review*, 1970, 12(4): 573~576.
- [190] Gleser L J. Estimation in a multivariate “errors in variables” regression model: large sample results. *Ann. Statist.*, 1981, 9: 24~44.
- [191] Goldfarb D, Ma S, Scheinberg K. Fast alternating linearization methods for minimizing the sum of two convex functions. Available at <http://arxiv.org/abs/0912.4571> (2010)
- [192] Goldstein T, Osher S. The split Bregman method for L1-regularized problems. *SIAM J. Imaging Sciences*, 2009, 2(2): 323~343.
- [193] Golub G H, Reinsch C. Singular Value Decomposition and Least Squares Solutions. *Numer Math*, 1970, 14: 403~420.
- [194] Golub G H. Some modified matrix eigenvalue problems. *SIAM Review*, 1973, 15: 318~334.
- [195] Golub G H, Pereyra V. The differentiation of pseudoinverses and nonlinear least squares problems whose variables separate. *SIAM J. Numer Anal*, 1973, 10: 413~432.
- [196] Golub G H, Van Loan C F. An analysis of the total least squares problem. *SIAM J. Numer Anal*, 1980, 17: 883~893.
- [197] Golub G H, Klema V, Stewart G W. Rank degeneracy and least squares problems. Technical Report TR-456, Dept Computer Science, University of Maryland, College Park, MD, 1986.
- [198] Golub G H, Van Loan C F. *Matrix Computation*. 2nd ed. Baltimore: The John Hopkins University Press, 1989.

- [199] Gonzales E F, Zhang Y. Accelerating the Lee-Seung algorithm for non-negative matrix factorization. Technical report. Department of Computational and Applied Mathematics, Rice University, 2005.
- [200] Grassmann H G. Die Ausdehnungslehre. Berlin: Enslin, 1862.
- [201] Gray R M. On the asymptotic eigenvalue distribution of Toeplitz matrices. IEEE Trans Information Theory, 1972, 18(6): 267~271.
- [202] Graybill F A, Meyer C D, Painter R J. Note on the computation of the generalized inverse of a matrix. SIAM Review, 1966, 8(4): 522~524.
- [203] Graybill F A. Matrices with Applications in Statistics. Belmont CA: Wadsworth International Group, 1983.
- [204] Green B. The orthogonal approximation of an oblique structure in factor analysis. Psychometrika, 1952, 17: 429~440.
- [205] Greville T N E. Some applications of the pseudoinverse of a matrix. SIAM Review, 1960, 2: 15~22.
- [206] Greville T N E. Note on the generalized inverse of a matrix product. SIAM Review, 1966, 8(4): 518~521.
- [207] Gribonval R, Nielsen M. Sparse representations in unions of bases, IEEE Trans. Inform. Theory, 2003, 49: 3320~3325.
- [208] Griffiths J W. Adaptive array processing: A tutorial. Proc IEE, Part F, 1983, 130: 137~142.
- [209] Grippo L, Sciandrone M. On the convergence of the block nonlinear Gauss-Seidel method under convex constraints. Operations Research Letter, 1999, 26: 127~136.
- [210] Guan N, Tao D, Lou Z, Yu B. NeNMF: An optimal gradient method for non-negative matrix factorization. IEEE Trans. Signal Processing, 2012, 60(6): 2082~2098.
- [211] Guttman L. Enlargement methods for computing the inverse matrix. Ann Math Statist, 1946, 17: 336~343.
- [212] Hager W W. Updating the inverse of a matrix. SIAM Review, 1989, 31(2): 221~239.
- [213] Hale E T, Yin W, ZHANG Y. Fixed-point continuation for  $\ell_1$ -minimization: Methodology and convergence. SIAM J. Optim., 2008, 19(3): 1107~1130.
- [214] Halmos P R. Finite Dimensional Vector Spaces. New York: Springer-Verlag, 1974.
- [215] Hanchez Y, Dooren P V. Elliptic and hyperbolic quadratic eigenvalue problems and associated distance problems. Linear Algebra and Its Applications, 2003, 371: 31~44.
- [216] Harshman R A. Foundation of the PARAFAC procedure: models and conditions for an “explanatory” multi-modal factor analysis. UCLA Work. Pap. Phon. 1970, 16: 1~84.
- [217] Harshman R A. Parafac2: Mathematical and technical notes. UCLA working papers in phonetics 1972, 22: 30~44.
- [218] Harshman R A, Hong S, Lundy M E. Shifted factor analysis - Part i: Models and properties. J. of Chemometrics, 2003, 17(7): 363~378.
- [219] Harshman R A, Lundy M E. Data preprocessing and the extended PARAFAC model. In Research Methods for Multimode Data Analysis (Eds. Law H G, Snyder C W, Hattie J A, McDonald R P.). New York: Praeger, 1984 (pp.216~284).
- [220] Harshman R A, Lundy M E. PARAFAC: Parallel factor analysis. Computational Statistics of Data Analysis, 1994, 18: 39~72.

- [221] Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001, Springer Series in Statistics.
- [222] Hazan T, Polak T, Shashua A. Sparse image coding using a 3d nonnegative tensor factorization. Technical report, The Hebrew University, 2005.
- [223] Heeg R S, Geurts B J. Spatial instabilities of the incompressible attachment-line flow using sparse matrix Jacobi-Davidson techniques. *Appl Sci Res*, 1998, 59: 315 ~ 329.
- [224] Helmke U, Moore J B. *Optimization and Dynamical Systems*. London, UK: Springer-Verlag, 1994.
- [225] Henderson H V, Searle S R. On deriving the inverse of a sum of matrices. *SIAM Review*, 1981, 23: 53 ~ 60.
- [226] Henderson H V, Searle S R. The vec-permutation matrix, the vec operator and Kronecker products: A review. *Linear and Multilinear Algebra*, 1981, 9: 271 ~ 288.
- [227] Herzog R, Sachs E. Preconditioned conjugate gradient method for optimal control problems with control and state constraints. *SIAM. J. Matrix Anal. and Appl.*, 2010, 31(5): 2291 ~ 2317.
- [228] Hestenes M R, Stiefel E. Methods of conjugate gradients for solving linear systems. *J. Res National Bureau of Standards*, 1952, 49: 409 ~ 436.
- [229] Hestenes M R. Multiplier and gradient methods, *J. Optimization Theory and Applications*, 1969, 4: 303 ~ 320.
- [230] Higham N J. Computing the polar decomposition – with applications. *SIAM J. Sci. Stat. Comp.*, 1986, 7(4): 1160 ~ 1974.
- [231] Higham N, Schreiber R. Fast polar decomposition of an arbitrary matrix. *SIAM J. Sci. Stat. Comput.*, 1990, 11(4): 648 ~ 655.
- [232] Hindi H. A tutorial on convex optimization, in: Proceeding of the 2004 American Control Conference, Boston, Massachusetts June 30 ~ July 2, 2004, pp.3252 ~ 3265.
- [233] Hindi H. A tutorial on convex optimization II: Duality and Interior Point Methods, In: Proc. of the 2006 American Control Conference Minneapolis, Minnesota, USA, June 14 ~ 16, 2006, pp.868 ~ 696.
- [234] Hitchcock F L. The expression of a tensor or a polyadic as a sum of products. *J. Mathematics and Physics*, 1927, 6: 164 ~ 189.
- [235] Hitchcock F L. Multiple invariants and generalized rank of a p-way matrix or tensor, *J. Mathematics and Physics*, 1927 7: 39 ~ 79.
- [236] Hochstenbach M E. A Jacobi-Davidson type SVD method. *SIAM J. Sci. Comput.*, 2001, 23(2): 606 ~ 628.
- [237] Honig M L, Madhow U, Verdu S. Blind adaptive multiuser detection. *IEEE Trans. Inform Theory*, 1995, 41: 944 ~ 960.
- [238] Horn R A, Johnson C R. *Matrix Analysis*. Cambridge: Cambridge University Press, 1985.
- [239] Horn R A, Johnson C R. *Topics in Matrix Analysis*. Cambridge: Cambridge University Press, 1991.
- [240] Hotelling H. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol*, 1933, 24: 417 ~ 441.
- [241] Hotelling H. Some new methods in matrix calculation. *Ann Math Statist*, 1943, 14: 1 ~ 34.

- [242] Hotelling H. Further points on matrix calculation and simultaneous equations. *Ann Math Statist*, 1943, 14: 440 ~ 441.
- [243] Howland P, Jeon M, Park H. Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition. *SIAM J. Matrix Anal. Appl*, 2003, 25(1): 165 ~ 179.
- [244] Howland P, Park H. Generalizing discriminant analysis using the generalized singular value decomposition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2004, 26(8): 995 ~ 1006.
- [245] Hoyer P O. Non-negative matrix factorization with sparseness constraints. *J. Machine Learning Research*, 2004, 5: 1457 ~ 1469.
- [246] Huang B. Detection of abrupt changes of total least squares models and application in fault detection. *IEEE Trans. Control Systems Technology*, 2001, 9(2): 357 ~ 367.
- [247] Huber, P J. Robust estimation of a location parameter. *Annals of Statistics*, 1964, 53: 73 ~ 101.
- [248] Huffel S V, Vandewalle J. *The Total Least Squares Problems: Computational Aspects and Analysis*. Frontiers Appl Math 9, Philadelphia: SIAM, 1991.
- [249] Hyland D C, Bernstein D S. The optimal projection equations for model reduction and the relationships among the methods of Wilson, Skelton and Moore. *IEEE Trans. Automatic Control*, 1985, 30: 1201 ~ 1211.
- [250] Jain P K, Ahmad K. *Functional analysis* (2nd ed.). New Age International, 1995.
- [251] Jain S K, Gunawardena A D. *Linear Algebra: An Interactive Approach*. Thomson Learning, 2003.
- [252] Jennings A, McKeown J J. *Matrix Computations*. New York: John Wiley & Sons, 1992.
- [253] Johnson C. *Matrix Theory and Applications*. American Mathematical Society, 1990.
- [254] Johnson D H, Dudgeon D E. *Array Signal Processing: Concepts and Techniques*. Englewood Cliffs, NJ: PTR Prentice Hall, 1993.
- [255] Johnson L W, Riess R D, Arnold J T. *Introduction to Linear Algebra*. 5th ed. New York: Prentice ~ Hall, 2000.
- [256] Jolliffe I. *Principal Component Analysis*. Springer-Verlag, 1986.
- [257] Jordan C. Memoire sur les formes bilineaires. *J. Math Pures Appl*, Deuxieme Serie, 1874, 19: 35 ~ 54.
- [258] Kantorovich L V. Function analysis and applied mathematics. *Uspekhi Matematicheskikh Nauk*, 1948, 3: 89 ~ 185. Translated from Russian by C D Benster, National Bureau of Standards, Report 1509, 7 March 1952.
- [259] Karmarkar N. A new polynomial-time algorithm for linear programming. *Combinatorica*, 1984, 4(4): 373 ~ 395.
- [260] Kato T. *A Short Introduction to Perturbation Theory for Linear Operators*. New York: Springer-Verlag, 1982.
- [261] Kay S M. *Modern Spectral Estimation: Theory and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
- [262] Kayalar S, Weinert H L. Oblique projections: Formulas, algorithms, and error bounds. *Math of Control, Signals, and Systems*, 1989, 2(1): 33 ~ 45.

- [263] Kelley C T. Iterative methods for linear and nonlinear equations. *Frontiers in Applied Mathematics*, vol.16, 1995, SIAM: Philadelphia, PA.
- [264] Keshavan R, Montanari A, Oh S. Matrix Completion from a Few Entries. *IEEE Trans. Information Theory*, 2010, 56(6): 2980 ~ 2998.
- [265] Khatri C G. Some results for the singular multivariate regression models. *Sankya, Series A*, 1968, 30: 267 ~ 280.
- [266] Khatri C G, Rao C R. Solutions to some functional equations and their applications to characterization of probability distributions. *Sankhya: The Indian J. Stat., Series A*, 1968, 30: 167 ~ 180.
- [267] Kiers H A L. Towards a standardized notation and terminology in multiway analysis. *J. Chemometrics*, 2000, 14: 105 ~ 122.
- [268] Kim J, Park H. Fast nonnegative matrix factorization: An active-set-like method and comparisons. *SIAM Journal on Scientific Computing*, 2011, 33(6): 3261 ~ 3281.
- [269] Kim H, Park H. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 2007, 23(12): 1495 ~ 1502.
- [270] Kim S J, Koh K, Lustig M, Boyd S, Gorinevsky D. An interior-point method for large-scale  $\ell_1$ -regularized least squares. *IEEE Journal Of Selected Topics in Signal Processing*, 2007, 1(4): 606 ~ 617.
- [271] Klema V C, Laub A J. The singular value decomposition: Its computation and some applications. *IEEE Trans. Automatic Control*, 1980, 25: 164 ~ 176.
- [272] Klemm R. Adaptive airborne MTI: An auxiliary channel approach. *Proc IEE, Part F*, 1987, 134: 269 ~ 276.
- [273] Klein J D, Dickinson B W. A normalized ladder form of residual energy ratio algorithm for PARCOR estimation via projections. *IEEE Trans. Automatic Control*, 1983, 28: 943 ~ 952.
- [274] Kolda T G. Orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.*, 2001, 23(1): 243 ~ 255.
- [275] Kolda T G, Bader B W, Kenny J P. Higher-order web link analysis using multilinear algebra. In *Proc. of The 5th IEEE International Conference on Data Mining*, 2005, 242 ~ 249.
- [276] Kolda T G. Multilinear operators for higher-order decompositions. Sandia Report SAND2006-2081, Sandia National Laboratories, Albuquerque, New Mexico and Livermore, California, Apr. 2006.
- [277] Kolda T G, Bader B W. The tophits model for higher-order web link analysis. In *Workshop on Link Analysis, Counterterrorism and Security*, 2006.
- [278] Kolda T G, Bader B W. Tensor decompositions and applications. *SIAM Review*, 2009, 51(3): 455 ~ 500.
- [279] Komzsik L. Implicit computational solution of generalized quadratic eigenvalue problems. *Finite Elements in Analysis and Design*, 2001, 37: 799 ~ 810.
- [280] Krabill D M. On extension of Wronskian matrices. *Bell Amer. Math. Soc.*, 1943, 49: 593 ~ 601.
- [281] Kreutz-Delgado K. Real vector derivatives and gradients. *Dept. Elect. Comput. Eng. UC*

- San Diego, Tech. Rep. Course Lecture Suppl. No.ECE275A, Dec.5, 2005 [Online]. Available at <http://dsp.ucsd.edu/kreutz/PEI05.html>.
- [282] Kreutz-Delgado K. Finite Dimensional Hilbert Spaces and Linear Inverse Problems. Dept. Elect. Comput. Eng. UC San Diego. Report Number ECE174LSHS-S2009V1.0, 2009.
- [283] Kreutz-Delgado K. The complex gradient operator and the calculus. Dept. Elect. Comput. Eng., Univ. California, San Diego, Tech. Rep. Course Lecture Suppl. No. ECE275A, Sep.-Dec. 2005 [Online]. Available at <http://dsp.ucsd.edu/kreutz/PEI05.html>
- [284] Kreyszig E. Advanced Engineering Mathematics, 7th ed. New York: John Wiley & Sons, Inc., 1993.
- [285] Krishna H, Morgera S D. The Levinson recurrence and fast algorithms for solving Toeplitz systems of linear equations. IEEE Trans. Acoust, Speech, Signal Processing, 1987, 35: 839 ~ 847.
- [286] Kruskal J B. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. Linear Algebra Appl., 1977, 18: 95 ~ 138.
- [287] Kruskal J B. Rank, decomposition, and uniqueness for 3-way and N-way arrays, in Multiway Data Analysis (Eds. Coppi and Bolasco), North-Holland: Elsevier Science Publishers B.V. 1989, 7 ~ 18.
- [288] Kruskal J B. Statement of some current results about three-way arrays. Unpublished manuscript, AT&T Bell Laboratories, Murray Hill, NJ. Available at <http://three-mode.leidenuniv.nl/pdf/k/kruskal1983.pdf>, 1983.
- [289] Kumaresan R. Rank reduction techniques and burst error-correction decoding in real/complex fields. In: Proc Nineteenth Asilomar Conf Circuits Syst Comput CA: Pacific Grove, 1985.
- [290] Kumar R. A fast algorithm for solving a Toeplitz system of equations. IEEE Trans Acoust, Speech, Signal Processing, 1985, 33: 254 ~ 267.
- [291] Kumaresan R. Estimating the parameters of exponentially damped or undamped sinusoidal signals in noise: [Ph.D. dissertation]. RI: University of Rhode Island, 1982.
- [292] Kumaresan R, Tufts D W. Estimating the angle of arrival of multiple plane waves. IEEE Trans. Aerospace Electron Syst, 1983, 19: 134 ~ 139.
- [293] Lancaster P. Lambda-Matrices and Vibrating Systems. Oxford: Pergamon Press, 1966.
- [294] Lancaster P, Tismenetsky M. The Theory of Matrices with Applications. 2nd ed. New York: Academic, 1985.
- [295] Lancaster P. Quadratic eigenvalue problems. Linear Algebra Appl., 1991, 150: 499 ~ 506.
- [296] Langville A N, Meyer C D, Albright R, Cox J, Duling D. Algorithms, initializations, and convergence for the nonnegative matrix factorization, 2006. Available at <http://langvillea.people.cofc.edu/NMFInitAlgConv.pdf>.
- [297] Lasdon L. Optimization Theory for Large Systems. New York: Macmillan, 1970.
- [298] Lathauwer L D, Moor B D, Vandewalle J. A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl., 2000, 21: 1253 ~ 1278.
- [299] Lathauwer L D, Moor B D, Vandewalle J. On the best rank-1 and rank- $(R_1, R_2, \dots, R_N)$  approximation of higher-order tensors. SIAM J. Matrix Anal. Appl., 2000, 21: 1324 ~ 1342.

- [300] Lathauwer L D. A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 2006, 28: 642~666.
- [301] Lathauwer L D. Decompositions of a higher-order tensor in block terms — PART I: Lemmas for partitioned matrices. *SIAM J. Matrix Anal. Appl.*, 2008, 30(3): 1022~1032.
- [302] Lathauwer L D. Decompositions of a higher-order tensor in block terms — PART I: Definitions and Uniqueness. *SIAM J. Matrix Anal. Appl.*, 2008, 30(3): 1033~1066.
- [303] Lathauwer L D, Nion D. Decompositions of a higher-order tensor in block terms — PART III: Alternating least squares algorithms. *SIAM J. Matrix Anal. Appl.*, 2008, 30(3): 1067~1083.
- [304] Laub A J, Heath M T, Paige C C, Ward R C. Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms. *IEEE Trans. Automatic Control*, 1987, 32: 115~122.
- [305] S. Lauritzen L. Graphical Models. London: Oxford University Press, 1996.
- [306] Lay D C. Linear Algebra and Its Applications, 2nd Edition. New York: Addison-Wesley, 2000.
- [307] Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 1999, 401: 788~791.
- [308] Lee D D, Seung H S. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing 13 (Proc. of NIPS 2000)*, MIT Press, 2001, 13: 556~562.
- [309] Lee H, Battle A, Raina R, Ng. A Y. Efficient sparse coding algorithms. *Advances in Neural Information Processing Systems (NIPS)*, 19, 2007.
- [310] Leonard I E. The matrix exponential. *SIAM Review*, 1996, 38(3): 507~512.
- [311] Letexier D, Bourennane S, Blanc-Talon J. Nonorthogonal tensor matricization for hyper spectral image filtering. *IEEE Geosci. Remote Sens. Letters*, 2008, 5(1): 3~7.
- [312] Levinson N. The Wiener RMS (root-mean-square) error criterion in filter design and prediction, *J. Math Phys*, 1947, 25: 261~278.
- [313] Lewicki M S, Sejnowski T J. Learning overcomplete representations. *Neural Comp.*, 2000, 12(2): 337~365.
- [314] Lewis A S. The mathematics of eigenvalue optimization. *Math. Program.*, 2003, 97(1-2): 155~176.
- [315] Li N, Kindermann S, Navasca C. Some convergence results on the regularized alternating least-squares method for tensor decomposition. *Linear Algebra and its Applications*, 2013, 438(2): 796~812.
- [316] Li X L, Zhang X D. Non-orthogonal approximate joint diagonalization free of degenerate solution. *IEEE Trans. Signal Processing*, 2007, 55(5): 1803~1814.
- [317] Lin C J. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 2007, 19(10): 2756~2779.
- [318] Lin Z, Chen M, Ma Y. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. Available at <http://arxiv.org/pdf/1009.5055.pdf>.
- [319] Liu S, Trenklerz G. Hadamard, Khatri-Rao, Kronecker and other matrix products. *International J. Information and Systems*, 2008, 4(1): 160~177.

- [320] Liu X, Sidiropoulos N D. Cramer-Rao lower bounds for low-rank decomposition of multidimensional arrays. *IEEE Transactions on Signal Processing*, 2001, 49(9): 2074~2086.
- [321] Lorch E R. On a calculus of operators in reflexive vector spaces. *Trans. Amer Math Soc*, 1939, 45: 217~234.
- [322] Lueberger D. *An Introduction to Linear and Nonlinear Programming*, 2nd ed., MA: Addison-Wesley, 1989.
- [323] Luenberger D D. *Linear and Nonlinear Programming*. 2nd ed. London: Addison-Wesley, 1984.
- [324] Lütkepohl H. *Handbook of Matrices*. New York: John Wiley & Sons, 1996.
- [325] Lyantse V E. Some properties of idempotent operators. *Trotet i Prikl Mat*, 1958, 1: 16~22.
- [326] MacDuffee C C. *The Theory of Matrices*. Berlin: Springer-Verlag, 1933.
- [327] Magnus J R, Neudecker H. The commutation matrix: Some properties and applications. *Ann Ststist*, 1979, 7: 381~394.
- [328] Magnus J R, Neudecker H. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Revised ed. Chichester: Wiley 1999.
- [329] Mahalanobis P C. On the generalised distance in statistics. *Proc. of the National Institute of Sciences of India*, 1936, 2(1): 49~55.
- [330] Makhoul J. Toeplitz determinants and positive semidefiniteness. *IEEE Trans. Signal Processing*, 1991, 39: 743~746.
- [331] Mallat S G, Zhang Z. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Processing*, 1993, 41(12): 3397~3415.
- [332] Manolakis D G, Ingle V K, Kogon S M. *Statistical and Adaptive Signal Processing*. Boston: McGraw-Hill, 2000.
- [333] Marcus M, Minc H. *A survey of Matrix Theory and Matrix Inequalities*. Boston: Allyn and Bacon, 1964.
- [334] Mardia K V, Kent J T, Bibby J M. *Multivariate Analysis*. London: Academic, 1979.
- [335] Marshall Jr. T G. Coding of real-number sequences for error correction: A digital signal processing problem. *IEEE J. Select Areas Commun*, 1984, 2(2): 381~392.
- [336] Martin R, Friedrich S. A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields. *J. Computational Neuroscience*, 2007, 22: 135~146.
- [337] Mathew G, Reddy V. Development and analysis of a neural network approach to Pisarenko's harmonic retrieval method. *IEEE Trans. Signal Processing*, 1994, 42: 663~667.
- [338] Mathew G, Reddy V. Orthogonal eigensubspaces estimation using neural networks. *IEEE Trans. Signal Processing*, 1994, 42: 1803~1811.
- [339] Meerbergen K. Locking and restarting quadratic eigenvalue solvers. *SIAM J Sci. Comput.*, 2001, 22(5): 1814~1839.
- [340] Megginson R E. *An Introduction to Banach Space Theory*. Graduate Texts in Mathematics 183. Springer-Verlag. Retrieved 1998.
- [341] Mesarovic V Z, Galatsanos N P, Katsaggelos K. Regularized constrained total least squares image restoration. *IEEE Trans. Image Processing*, 1995, 4(8): 1096~1108.

- [342] Michalewicz Z, Dasgupta D, Le Riche R, Schoenauer M. Evolutionary algorithms for constrained engineering problems, *Computers & Industrial Engineering Journal*, 1996, 30: 851 ~ 870.
- [343] Micka O J, Weiss A J. Estimating frequencies of exponentials in noise using joint diagonalization. *IEEE Trans. Signal Processing*, 1999, 47(2): 341 ~ 348.
- [344] Milliken G A, Akdeniz F. A theorem on the difference of the generalized inverses of two nonnegative matrices. *Communications in Statistics*, 1977, A6: 73 ~ 79.
- [345] Million E. The Hadamard product. Available at <http://buzzard.ups.edu/courses/2007spring/projects/million-paper.pdf>.
- [346] Minka T P. Old and new matrix algebra useful for statistics, December 2000. Notes.
- [347] Mirsky L. Symmetric gauge functions and unitarily invariant norms. *Quart J Math Oxford*, 1960, 11: 50 ~ 59.
- [348] Miwakeichi, F, Martnez-Montes E, Valds-Sosa, P, Nishiyama, N, Mizuhara, H, Yamaguchi, Y. Decomposing EEG data into space-time-frequency components using parallel factor analysis. *NeuroImage*, 2004, 22(3): 1035 ~ 1045.
- [349] Mohanty N. Random Signal Estimation and Identification. Van Nostrand Reinhold, 1986.
- [350] Moonen M, De Moor B, Vandenberghe L, Vandewalle J. On- and off-line identification of linear state space models. *International J Control*, 1989, 49(1): 219 ~ 232.
- [351] Moore E H. General analysis, Part 1. *Mem Amer Philos Sic*, 1935, 1: 1.
- [352] Moreau E. A generalization of joint-diagonalization criteria for source separation. *IEEE Trans. Signal Processing*, 2001, 49(3): 530 ~ 541.
- [353] Morup M, Hansen L K, Herrmann C S, Parnas J, Arnfred S M. Parallel Factor Analysis as an exploratory tool for wavelet transformed event-related EEG. *NeuroImage*, 2006, 29: 938 ~ 947.
- [354] Murray F J. On complementary manifolds and projections in  $L_p$  and  $l_p$ . *Trans. Amer Math Soc*, 1937, 43: 138 ~ 152.
- [355] Natarajan B K. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 1995, 24: 227 ~ 234.
- [356] Navasca C, Lathauwer L D, Kindermann S. Swamp reducing technique for tensor decomposition. In the 16th Proceedings of the European Signal Processing Conference, Lausanne, Switzerland, August 25-29, 2008.
- [357] Neagoe V E. Inversion of the Van der Monde Matrix. *IEEE Signal Processing Letters*, 1996, 3: 119 ~ 120.
- [358] Needell D, Vershynin R. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.*, 2009, 9(3): 317 ~ 334.
- [359] Needell D, Vershynin R. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *IEEE J. Sel. Topics Signal Process.*, 2009, 4(2): 310 ~ 316.
- [360] Needell D, Tropp J A. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmonic Anal.*, 2009, 26(3): 301 ~ 321.
- [361] Nesterov Y. A method for solving a convex programming problem with rate of convergence  $O(\frac{1}{k^2})$ . *Soviet Math. Doklady*, 1983, 269(3): 543 ~ 547.

- [362] Nesterov Y, Nemirovsky A. A general approach to polynomial-time algorithms design for convex programming. Report, Central Economical and Mathematical Institute, USSR Academy of Sciences, Moscow, 1988.
- [363] Nesterov Y. Introductory Lectures on Convex Optimization: A Basic Course. Boston, MA: Kluwer Academic, 2004.
- [364] Nesterov Y. Smooth minimization of nonsmooth functions (CORE Discussion Paper #2003/12, CORE 2003). *Math. Program.* 2005, 103(1): 127~152.
- [365] Nesterov Y. Gradient methods for minimizing composite objective function. CORE Discussion Paper #2007/96, 2007.
- [366] Nesterov Y. Primal-dual subgradient methods for convex problems. *Math. Program., Ser. B*, 2009, 120: 221~259.
- [367] Neumaier A. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review*, 1998, 40(3): 636~666.
- [368] Nevelson M, Hasminskii R. Stochastic Approximation and Recursive Estimation. American Mathematical Society, 1973.
- [369] Ng L, Solo V. Error-in-variables modeling in optical flow estimation. *IEEE Trans. Image Processing*, 2001, 10(10): 1528~1540.
- [370] Nievergelt Y. Total least squares: State-of-the-art regression in numerical analysis. *SIAM Review*, 1994, 36(2): 258~264.
- [371] Noble B, Dattorro J W. Applied Linear Algebra. 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1988.
- [372] Nocedal J, Wright S J. Numerical Optimization. New York: Springer-Verlag, 1999.
- [373] Nour-Omid B, Parlett B N, Ericsson T, Jensen P S. How to implement the spectral transformation. *Math Comput*, 1987, 48: 663~673.
- [374] Ohmann M. Fast cosine transform of Toeplitz matrices, algorithm and applications. *IEEE Trans. Signal Processing*, 1993, 41: 3057~3061.
- [375] Oja E. A simplified neuron model as a principal component analyzer. *J. Math. Bio.*, 1982, 15: 267~273.
- [376] Oja E, Karhunen J. On stochastic Approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *J Math Anal Appl*, 1985, 106: 69~84.
- [377] Oja E. The nonlinear PCA learning rule in independent component analysis. *Neurocomputing*, 1997, 17: 25~45.
- [378] Olshausen B A. Sparse coding of time-varying natural images. *J. Vision*, 2002, 2(7): 130~135.
- [379] Olshausen B A, Field D J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 1996, 381: 607~609.
- [380] Olshausen B A, Field D J. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 1997, 37(23): 3311~3325.
- [381] Olson L, Vandini T. Eigenproblems from finite element analysis of fluid-structure interactions. *Comput. Struct*, 1989, 33: 679~687.
- [382] Ortega J M, Rheinboldt W C. Iterative Solution of Nonlinear Equations in Several Variables. New York/London: Academic Press, 1970.

- [383] Osborne M, Presnell B, Turlach B. A new approach to variable selection in least squares problems. *IMA J. Numer. Anal.*, 2000, 20: 389 ~ 403.
- [384] Osher S, Burger M, Goldfarb D, Xu J, Yin W. An iterative regularization method for total variation-based image restoration. *Multiscale Model. Simul.*, 2005, 4(2): 460 ~ 489.
- [385] Ottersten B, Asztele D, Kristensson M, Parkvall S. A statistical approach to subspace based estimation with applications in telecommunications. In: Van Huffel S ed. *Recent Advances in Total Least Squares Techniques and Error-in-Variables Modeling*, Philadelphia, PA: SIAM, 1997.
- [386] Paatero P, Tapper U. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 1994, 5: 111 ~ 126.
- [387] Paatero P, Tapper U. Least squares formulation of robust non-negative factor analysis. *Chemometrics Intell. Lab.*, 1997, 37: 23 ~ 35.
- [388] Paatero P. A weighted non-negative least squares algorithm for three-way PARAFAC factor analysis. *Chemometrics Intell. Lab. Syst.* 1997; 38(2): 223 ~ 242.
- [389] Paige C C, Saunders N A. Towards a generalized singular value decomposition. *SIAM J. Numer. Anal.*, 1981, 18: 269 ~ 284.
- [390] Paige C C. Computing the generalized singular value decomposition. *SIAM J. Sci. Stat. Comput.*, 1986, 7: 1126 ~ 1146.
- [391] Pajunnen P, Karhunen J. Least-Squares methods for blind source Separation based on Non-linear PCA. *Int. J. of Neural Systems*, 1998, 8: 601 ~ 612.
- [392] Papoulis A. *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 1991.
- [393] Parlett B N. The Rayleigh quotient iteration and some generalizations for nonnormal matrices. *Mathematics of Computation*, 1974, 28(127): 679 ~ 693.
- [394] Parlett B N. *The Symmetric Eigenvalue Problem*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [395] Parra L, Spence C, Sajda P, Ziehe, Muller K. Unmixing hyperspectral data. In: *Advances in Neural Information Processing Systems*, vol.12. Cambridge, MA: MIT Press, 2000, 942 ~ 948.
- [396] Pati Y C, Rezaifar R, Krishnaprasad P S. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. *Proc. 27th Annu. Asilomar Conf. Signals Syst. Comput.*, Nov. 1993, vol.1, 40 ~ 44.
- [397] Pauca V P V, Piper J, Plemmons R. Nonnegative matrix factorization for spectral data analysis. *Linear Algebra and Applications*. 2006, 416(1): 29 ~ 47.
- [398] Pavon M. New results on the interpolation problem for continuous-time stationary-increments processes. *SIAM J. Control Optim.*, 1984, 22: 133 ~ 142.
- [399] Pearson K. On lines and planes of closest fit to points in space. *Phil Mag*, 1901, 559 ~ 572.
- [400] Pease M C. *Methods of Matrix Algebra*. New York: Academic Press, 1965.
- [401] Peng C Y, Zhang X D. On recursive oblique projectors. *IEEE Signal Processing Letters*, 2005, 12(6): 433 ~ 436.
- [402] Penrose R A. A generalized inverse for matrices. *Proc. Cambridge Philos. Soc.*, 1955, 51: 406 ~ 413.

- [403] Petersen K B, Petersen M S. The Matrix Cookbook. 2008 [Online]. Available at <http://matrixcookbook.com>.
- [404] D. T. Pham, Joint approximate diagonalization of positive definite matrices. SIAM J. Matrix Anal. Appl., 2001, 22(4): 1136 ~ 1152.
- [405] Phillip A, Regalia P A, Mitra S. Kronecker products, unitary matrices and signal processing applications. SIAM Review, 1989, 31(4): 586 ~ 613.
- [406] Piegorsch W W, Casella G. The early use of matrix diagonal increments in statistical problems. SIAM Review, 1989, 31: 428 ~ 434.
- [407] Piegorsch W W, Casella G. Erratum: Inverting a sum of matrices. SIAM Review, 1990, 32: 470.
- [408] Pintelon R, Guillaume P, Vandersteen G, Rolain Y. Analyzes, development and applications of TLS algorithms in frequency domain system identification. In: Van Huffel S ed. Recent Advances in Total Least Squares Techniques and Error-in-Variable Modeling, Philadelphia, PA: SIAM, 1997.
- [409] Pisarenko V F. The retrieval of harmonics from a covariance function. Geophysics, J Roy Astron Soc, 1973, 33: 347 ~ 366.
- [410] Piziak R, Odell P L. Full rank factorization of matrices. Mathematics Magazine, 1999, 72(3): 193 ~ 202.
- [411] Polyak B T. Introduction to Optimization. Optimization Software Inc., 1987.
- [412] Ponnappalli S P, Saunders M A, Van Loan C F, Alter O. A higher-order generalized singular value decomposition for comparison of global mRNA expression from multiple organisms. PLOS ONE, 2011. Available at <http://www.plosone.org/article/info>
- [413] Pouliarikas A D. The Handbook of Formulas and Tables for Signal Processing. New York: CRC Press, Springer, IEEE Press, 1999.
- [414] Powell M J D. A method for nonlinear constraints in minimization problems. In Optimization (ed. by R. Fletcher), New York: Academic Press, 1969, 283 ~ 298.
- [415] Powell M J D. On search directions for minimization algorithms. Math. Programming, 1973, 4: 193 ~ 201.
- [416] Powell M J D. Convergence properties of algorithms for nonlinear optimization. SIAM Review, 1986, 28: 487 ~ 500.
- [417] Price C. The matrix pseudoinverse and minimal variance estimates. SIAM Review, 1964, 6: 115 ~ 120.
- [418] Pringle R M, Rayner A A. Expressions for generalized inverses of a bordered matrix with application to the theory of constrained linear models. SIAM Review, 1970, 12: 107 ~ 115.
- [419] Pringle R M, Rayner A A. Generalized Inverse of Matrices with Applications to Statistics. London: Griffin 1971.
- [420] Prugovecki E. Quantum Mechanics in Hilbert Space (2nd ed.). Academic Press, 1981.
- [421] Quattoni A, Collins M, Darrell T. Transfer learning for image classification with sparse prototype representation. Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., 2008. DOI: 10.1109/CVPR.2008.4587637.
- [422] Rado R. Note on generalized inverse of matrices. Proc. Cambridge Philos Soc, 1956, 52: 600 ~ 601.

- [423] Rao C R. Estimation of heteroscedastic variances in linear models. *J. Amer Statist Assoc*, 1970, 65: 161 ~ 172.
- [424] Rao C R, Mitra S K. *Generalized Inverse of Matrices*. New York: John Wiley & Sons, 1971.
- [425] Rayleigh L. *The Theory of Sound*. 2nd ed. New York: Macmillian, 1937.
- [426] Regalia P A, Mitra S K. Kronecker products, unitary matrices and signal processing applications. *SIAM Review*, 1989, 31(4): 586 ~ 613.
- [427] Riba J, Goldberg J, Vazquez G. Robust beamforming for interference rejection in mobile communications. *IEEE Trans. Signal Processing*, 1997, 45(1): 271 ~ 275.
- [428] Rockafellar R, Wets R. *Variational analysis*. Springer- Verlag, 1998.
- [429] Roos C. A full-Newton step  $O(n)$  infeasible interior-point algorithm for linear optimization. *SIAM J. Optimization* 2006, 16(1): 1110 ~ 1136.
- [430] Roos C, Terlaky T, Vial J.-Ph. *Theory and Algorithms for Linear Optimization: An Interior-Point Approach*. Chichester, UK: John Wiley & Sons, 1997.
- [431] Roy R, Kailath T. ESPRIT — Estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust, Speech, Signal Processing*, 1989, 37: 297 ~ 301.
- [432] Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Physica D*, 1992, 60: 259 ~ 268.
- [433] Saad Y. The Lanczos biorthogonalization algorithm and other oblique projection methods for solving large unsymmetric systems. *SIAM J. Numer. Anal.*, 1982, 19: 485 ~ 506.
- [434] Saad Y. *Numerical Methods for Large Eigenvalue Problems*. New York: Manchester University Press, 1992.
- [435] Salehi H. On the alternating projections theorem and bivariate stationary stochastic processes. *Trans. Amer. Math. Soc.*, 1967, 128: 121 ~ 134.
- [436] Samson C. A unified treatment of fast algorithms for identification. *International J Control*, 1982, 35: 909 ~ 934.
- [437] Scales L E. *Introduction to Non-linear Optimization*. London: Macmillan, 1985.
- [438] Schmidt R O. Multiple emitter location and signal parameter estimation. Proc RADC Spectral Estimation Workshop, NY: Rome, 1979, 243 ~ 258.
- [439] Schmidt O R. Multiple emitter location and signal parameter estimation. *IEEE Trans. Antenna Propagat.*, 1986, 34: 276 ~ 280.
- [440] Schott J R. *Matrix Analysis for Statistics*. Wiley: New York, 1997.
- [441] Schoukens J, Pintelon R, Vandersteen G, Guillaume P. Frequency-domain system identification using nonparametric noise models estimated from a small number of data sets. *Automatica*, 1997, 33(6): 1073 ~ 1086.
- [442] Schutz B. *Geometrical Methods of Mathematical Physics*. Cambridge University Press, 1980.
- [443] Scutari G, Palomar D P, Facchini F, Pang J S. Convex optimization, game theory, and variational inequality theory. *IEEE Signal Processing Magazine*, 2010, 27(3): 35 ~ 49.
- [444] Searle S R. *Matrix Algebra Useful for Statistics*. New York: John Wiley & Sons, 1982.
- [445] Selby S M. *Standard Mathematical Tables*. CRC Press, 1974.
- [446] Sharman K, Durrani T S. A comparative study of modern eigenstructure methods for bearing estimation — A new high performance approach. Proc IEEE ICASSP-87, Greece, Athens, 1987, 1737 ~ 1742.

- [447] Shashua A, Levin A. Linear image coding for regression and classification using the tensor-rank principle. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, 2001.
- [448] Shashua A, Zass R, Hazan T. Multi-way clustering using super-symmetric nonnegative tensor factorization. In European Conference on Computer Vision (EV), Graz, Austria, May 2006.
- [449] Shavitt I, Bender C F, Pipano A, Hosteney R P. The iterative calculation of several of the lowest or highest eigenvalues and corresponding eigenvectors of very large symmetric matrices. *J Comput Phys*, 1973, 11: 90~108.
- [450] Sherman J, Morrison W J. Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix (abstract). *Ann Math Statist*, 1949, 20: 621.
- [451] Sherman J, Morrison W J. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *Ann Math Statist*, 1950, 21: 124~127.
- [452] Shewchuk J R. An introduction to the conjugate gradient method without the agonizing pain. Available at <http://quake-papers/painless-conjugate-gradient-pics.ps>.
- [453] Sidiropoulos N D, Bro R. On the uniqueness of multilinear decomposition of N-way arrays, *J. Chemometrics*, 2000, 14: 229~239.
- [454] Sidiropoulos D, Budampati R S. Khatri-Rao space-time Codes. *IEEE Trans. Signal Processing*, 2002, 50(10): 2396~2407.
- [455] Silva V, Lim L -H. Tensor rank and the ill-posedness of the best low-rank approximation problem, *SIAM J. Matrix Analysis and Applications*, 2008, 30(3): 1084~1127.
- [456] Silvery S D. Statistical Inference. Penguin books, 1970.
- [457] Simon J C. Patterns and Operators: The Foundations and Data Representation. North Oxford Academic Publishers Ltd, 1986.
- [458] Speiser J, and Van Loan C. Signal processing computations using the generalized singular value decomposition. In: Proc of SPIE, Vol495, SPIE International Symposium, San Diego, 1984.
- [459] Spivak M. A Comprehensive Introduction to Differential Geometry (2nd Edition) (5 Volumes). Publish or Perish Press, 1979.
- [460] Stallings W T, Boullion T L. Computation of pseudoinverse matrices using residue arithmetic. *SIAM Review*, 1972, 14(1): 152~163.
- [461] Stewart G W. On the sensitivity of the eigenvalue problem  $Ax = \lambda Bx$ . *SIAM J. Num. Anal.*, 1972, 9: 669~686.
- [462] Stewart G W. An Introduction to Matrix Computations. New York: Academic Press, 1973.
- [463] Stewart G W, Sun J G. Matrix Perturbation Theory. New York: Academic Press, 1990.
- [464] Stewart G W. An updating algorithm for subspace tracking. *IEEE Trans. Signal Processing*, 1992, 40: 1535~1541.
- [465] Stewart G W. On the early history of the singular value decomposition. *SIAM Review*, 1993, 35(4): 551~566.
- [466] Stiefel E. Richtungsfelder und ferparallelismus in  $n$ -dimensionalem mannig faltigkeiten. *Commentarii Math Helvetici*, 1935-1936, 8: 305~353.

- [467] Stoica P, Sorelius J, Cedervall M, Söderström T. Error-in-variables modeling: An instrumental variable approach. In: Van Huffel S ed. *Recent Advances in Total Least Squares Techniques and Error-in-Variables Modeling*, Philadelphia, PA: SIAM, 1997.
- [468] Sun J, Zeng H, Liu H, Lu Y, Chen Z. Cubesvd: a novel approach to personalized web search. In: *Proceedings of the 14th international conference on World Wide Web*, 2005, 652~662.
- [469] Syau Y R. A note on convex functions. *Internat. J. Math. & Math. Sci.* 1999, 22(3): 525~534.
- [470] Takeuchi K, Yanai H, Mukherjee B N. *The Foundations of Multivariate Analysis*. New York: Wiley, 1982.
- [471] Tibshirani R. Regression shrinkage and selection via the lasso. *J. R. Statist. Soc. B*, 1996, 58: 267~288.
- [472] Tikhonov A. Solution of incorrectly formulated problems and the regularization method, *Soviet Math. Dokl.*, 1963, 4: 1035~1038.
- [473] Tikhonov A, Arsenin V. *Solution of Ill-Posed Problems*. New York: Wiley, 1977.
- [474] Tisseur F, Meerbergen K. Quadratic eigenvalue problem. *SIAM Review*, 2001, 43(2): 235~286.
- [475] Toeplitz O. Zur Theorie der quadratischen und bilinearen Formen von unendlichvielen Veränderlichen. I Teil: Theorie der L-Formen, *Math Annal*, 1911, 70: 351~376.
- [476] Todd R. Seminorm. From MathWorld—A Wolfram Web Resource, created by Eric W. Weisstein. <http://mathworld.wolfram.com/Seminorm.html>
- [477] Tou J T, Gonzalez R C. *Pattern Recognition Principles*. London: Addison-Wesley Publishing Comp, 1974.
- [478] Tropp J A. Greed is good: algorithmic results for sparse approximation. *IEEE Trans. Information Theory*, 2004, 50(10): 2231~2242.
- [479] Tropp J A. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Trans. Information Theory*, 2006, 52(3): 1030~1051.
- [480] Tropp J A, Wright S J. Computational methods for sparse solution of linear inverse problems. *Proc. IEEE*, 2010, 98(6): 948~958.
- [481] Tsallis C. Possible generalization of Boltzmann-Gibbs statistics. *J. Statistical Physics*, 1988, 52: 479~487.
- [482] Tsallis C. The nonadditive entropy  $S_q$  and its applications in physics and elsewhere: Some remarks. *Entropy*, 2011, 13: 1765~1804.
- [483] Tsatsanis M K, Z. Xu. Performance analysis of minimum variance CDMA receivers. *IEEE Trans. Signal Processing*, 1998, 46: 3014~3022.
- [484] Tseng P. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optimization Theory and Applications*, 2001, 109(3): 475~494.
- [485] Tucker L R. Implications of factor analysis of three-way matrices for measurement of change. In *Problems in Measuring Change* (C W. Harris ed.), University of Wisconsin Press, 1963, 122~137.
- [486] Tucker L R. The extension of factor analysis to three-dimensional matrices. In *Contributions to Mathematical Psychology* (H. Gulliksen and N. Frederiksen, eds.), New York: Holt, Rinehart & Winston, 1964, 109~127.

- [487] Tucker L R. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 1966, 31: 279~311.
- [488] Utschick W. Tracking of signal subspace projectors. *IEEE Trans. Signal Processing*, 2002, 50(4): 769~778.
- [489] van der Kloot W A, Kroonenberg P M. External analysis with three-mode principal component models, *Psychometrika*, 1985, 50: 479~494.
- [490] van der Veen A J. Joint diagonalization via subspace fitting techniques. Proc 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01), 2001, 5: 2773~2776.
- [491] Van Huffel S (Ed). Recent Advances in Total Least Squares Techniques and Error-in-Variables Modeling. Philadelphia, PA: SIAM, 1997.
- [492] Van Huffel S. TLS applications in biomedical signal processing. In: Van Huffel S ed. Recent Advances in Total Least Squares Techniques and Error-in-Variables Modeling, Philadelphia, PA: SIAM, 1997.
- [493] Van Loan C F. Generalizing the singular value decomposition. *SIAM J. Numer. Anal.*, 1976, 13: 76~83.
- [494] Van Loan C F. Matrix computations and signal processing. In: Haykin S ed. Selected Topics in Signal Processing, Englewood Cliffs: Prentice-Hall, 1989.
- [495] van Overschee P, De Moor B. Subspace Identification for Linear Systems. Boston, MA: Kluwer, 1996.
- [496] Van Huffel S, Vandewalle J. Analysis and properties of the generalized total least squares problem  $Ax = b$  when some or all columns in  $A$  are subject to error. *SIAM J. Matrix Anal. Appl.*, 1989, 10: 294~315.
- [497] Vandaele P, Moonen M. Two deterministic blind channel estimation algorithms Based on Oblique Projections. *Signal Processing*, 2000, 80: 481~495.
- [498] Vandenberghe L. Lecture Notes for EE236C (Spring 2011-12), UCLA.
- [499] Vanderbei R J. An interior-point algorithm for nonconvex nonlinear programming. Available at <http://orfe.princeton.edu/rvdb/pdf/talks/level3/nl.pdf>
- [500] Vanderbei R J, Shanno D F. An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 1999, 13(1-3): 231~252.
- [501] Vasilescu M A O, Terzopoulos D. Multilinear analysis of image ensembles: TensorFaces. In Proc. of the European Conf. on Computer Vision (EV' 02), Copenhagen, Denmark, May, 2002, 447~460.
- [502] Vasilescu M A O, Terzopoulos D. Multilinear image analysis for facial recognition. In Proc. of the International Conference on Pattern Recognition (ICPR' 02), Quebec City, Canada, August, 2002.
- [503] Veen A V D. Algebraic methods for deterministic blind beamforming. Proc IEEE, 1998, 86: 1987~2008.
- [504] Viberg M, Ottersten B. Sensor array processing based on subspace fitting. *IEEE Trans. Signal Processing*, 1991, 39: 1110~1121.
- [505] von Neumann J. Some matrix inequalities and metrization of matric-space. Tomsk University

- Review, 1937, 1: 286~300. In: Collected Works, Oxford: Pergamon, 1962, Volume IV, 205~218.
- [506] Wächter A, Biegler L T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical Programming, Ser. A, 2006, 106(1): 25~57.
- [507] Wang H, Ahuja N. Compact representation of multidimensional data using tensor rank-one decomposition. In Proc. of International Conference on Pattern Recognition, 2004, Vol.1, 44~47.
- [508] Watkins D S. Understanding the QR algorithm. SIAM Review, 1982, 24(4): 427~440.
- [509] Watson G A. Characterization of the subdifferential of some matrix norms, Linear Algebra Appl., 1992, 170: 33~45.
- [510] Wax M, Sheinvald J. A least squares approach to joint diagonalization. IEEE Signal Processing Letters, 1997, 4(2): 52~53.
- [511] Weiss A J, Friedlander B. Array processing using joint diagonalization. Signal Processing, 1996, 1996, 50(3): 205~222.
- [512] Welling M, Weber M. Positive tensor factorization. Pattern Recognition Letters, 2001, 22: 1255~1261.
- [513] Wilkinson J H. The Algebraic Eigenvalue Problem. Oxford, UK: Clarendon Press, 1965.
- [514] Wikipedia. Variational inequality. [http://en.wikipedia.org/wiki/Variational\\_inequality](http://en.wikipedia.org/wiki/Variational_inequality).
- [515] Wirtinger W. Zur formalen theorie der funktionen von mehr komplexen veränderlichen. Mathematische Annalen, 1927, 97: 357~375.
- [516] Wolf J K. Redundancy, the discrete Fourier transform, and impulse noise cancellation. IEEE Trans. Commun, 1983, 31: 458~461.
- [517] Woodbury M A. Inverting modified matrices. Memorandum Report 42, Statistical Research Group, NJ: Princeton, 1950.
- [518] Wright J, Ma Y. Dense error correction via  $l^1$ -minimization. IEEE Trans. Information Theory, 2010, 56(7): 3540~3560.
- [519] Xu G, Cho Y, Kailath T. Application of fast subspace decomposition to signal processing and communication problems. IEEE Trans. Signal Processing, 1994, 42: 1453~1461.
- [520] Xu G, Kailath T. Fast subspace decomposition. IEEE Trans. Signal Processing, 1994, 42: 539~551.
- [521] Xu L, Oja E, Suen C. Modified Hebbian learning for curve and surface fitting. Neural Networks, 1992, 5: 441~457.
- [522] Yang B. Projection approximation subspace tracking. IEEE Trans. Signal Processing, 1995, 43: 95~107.
- [523] Yang B. An extension of the PASTd algorithm to both rank and subspace tracking. IEEE Signal Processing Letters, 1995, 2(9): 179~182.
- [524] Yang J F, Kaveh M. Adaptive eigensubspace algorithms for direction or frequency estimation and tracking. IEEE Trans. Acoust, Speech, Signal Processing, 1988, 36: 241~251.
- [525] Yang X, Sarkar T K, Arvas E. A survey of conjugate gradient algorithms for solution of extreme eigen-problems of a symmetric matrix. IEEE Trans Acoust, Speech, Signal Processing, 1989, 37: 1550~1556.

- [526] Yeniay O, Ankara B. Penalty function methods for constrained optimization with genetic algorithms. *Mathematical and Computational Applications*, 2005, 10(1): 45~56.
- [527] Yeredor A. Non-orthogonal joint diagonalization in the least squares sense with application in blind source separation. *IEEE Trans. Signal Processing*, 2002, 50(7): 1545~1553.
- [528] Yeredor A. Time-delay estimation in mixtures. Proc. 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03), 2003, 5: 237~240.
- [529] Yin W. Sparse Optimization: Lecture Outlines. Department of Computational and Applied Mathematics, Rice University.
- [530] Yin W, Osher S, Goldfarb D, Darbon J. Bregman iterative algorithms for  $l_1$ -minimization with applications to compressed sensing. *SIAM J. Imaging Sciences*, 2008, 1(1): 143~168.
- [531] Youla D C. Generalized image restoration by the method of alternating projections. *IEEE Trans. Circuits and Systems*, 1978, 25: 694~702.
- [532] Yu K B. Recursive updating the eigenvalue decomposition of a covariance matrix. *IEEE Trans. Signal Processing*, 1991, 39: 1136~1145.
- [533] Yu X, L Tong. Joint channel and symbol estimation by oblique projections. *IEEE Trans. Signal Processing*, 2001, 49(12): 3074~3083.
- [534] Yu Y L. Nesterov's Optimal Gradient Method. 2009, available at <http://www.webdocs.cs.ualberta.ca/yaoliang/mytalks/NS.pdf>.
- [535] Zass R, Shashua A. Nonnegative sparse PCA. In Neural Information Processing Systems (NIPS), Vancouver, Canada, Dec. 2006.
- [536] Zdunek R, Cichocki A. Nonnegative matrix factorization with constrained second-order optimization. *Signal Processing*, 2007, 87(8): 1904~1916.
- [537] Zha H. The restricted singular value decomposition of matrix triplets. *SIAM J. Matrix Anal Appl*, 1991, 12: 172~194.
- [538] Zhang X D, Liang Y C. Prefiltering-based ESPRIT for estimating parameters of sinusoids in non-Gaussian ARMA noise. *IEEE Trans. Signal Processing*, 1995, 43: 349~353.
- [539] Zhang X D, Zhang Y S. Determination of the MA order of an ARMA process using sample correlations. *IEEE Trans. Signal Processing*, 1993, 41: 2277~2280.
- [540] 黄琳. 系统与控制理论中的线性代数, 北京: 科学出版社, 1984.
- [541] 冯俊文.  $(H, \Omega)$  共轭函数理论. 系统科学与数学 (J. Sys. Sci. & Math. Sci.), 1990, 10(1): 71~77.
- [542] 《数学手册》编写组. 数学手册. 北京: 高等教育出版社, 1979.
- [543] 依理正夫, 児玉慎三, 須田信英. 特異値分解とそのシステム制御への応用. 計測と制御, 1982, 21: 763~772.
- [544] 张贤达. 左、右伪逆矩阵的数值计算. 科学通报, 1982, 27(2): 126.
- [545] 张贤达, 保铮. 通信信号处理. 北京: 国防工业出版社, 2000.
- [546] 张贤达. 现代信号处理(第二版). 北京: 清华大学出版社, 2002.

# 索引

## A

鞍点, 196  
按列堆栈, 74  
 $A$  不变, 487  
AB-散度, 361  
Alpha-散度, 361  
Armijo 条件, 267  
Aronszajn 综合公式, 546

## B

半负定矩阵, 45  
半空间, 226  
半正定矩阵, 45  
半正交矩阵, 109  
半正定锥, 226  
伴随矩阵, 67  
伴随算子, 25  
  自伴随算子, 25  
包容性, 487  
本质相等矩阵, 463  
闭合性, 17  
闭球体, 212  
逼近, 193  
逼近解序列, 238  
比例化, 610  
边界点, 196  
变分不等式, 211  
变形对数, 364  
变形指数, 364  
编码矩阵, 80  
编码区, 357  
表示矩阵, 83  
标量乘法单位律, 18  
标量乘法分配律, 18

标量乘法结合律, 18  
并向量(奇异值)分解, 289  
病态矩阵, 287  
薄奇异值分解, 见“截尾奇异值分解”  
不可跟踪, 377  
不变子空间, 487, 508  
不定矩阵, 45  
不精确直线搜索, 267  
不可辨识的, 327  
不可行点, 210  
补角, 491  
Banach 空间, 25, 36  
Beta-散度, 361  
Boltzmann-Gibbs 熵, 362  
Bregman 迭代, 391  
Bregman 迭代算法, 392  
Bregman 距离, 360, 391  
Broyden-Fletcher-Goldfarb-Shanno 算法, 267

## C

测度, 30  
侧向切片, 566  
侧向切片矩阵, 567  
常数向量, 2  
超定方程, 52  
超对角线, 565  
超平面, 225  
超平面拟合, 347  
超正规矩阵, 见“ $J$  正交矩阵”  
乘法结合律, 5  
乘法右分配律, 5  
乘法左分配律, 5  
乘积奇异值分解, 298  
乘幂法, 445

- 惩罚函数, 257  
二次罚函数, 257  
非平滑罚函数, 257  
内罚函数, 258  
外罚函数, 258  
惩罚函数法, 259  
混合内罚函数法, 260  
混合外罚函数法, 260  
内罚函数法, 259  
外罚函数法, 259  
惩罚因子, 258  
尺度不确定性, 604  
初等列变换, 14  
初等行变换, 9  
次梯度算法, 246  
次梯度向量, 241  
次分量分析, 313  
次微分, 241  
次最优点, 210  
 $\epsilon$ -次最优对偶点, 219  
 $\epsilon$ -次最优原始点, 219  
Cauchy-Riemann 方程, 171  
Cauchy-Riemann 条件, 171  
Cauchy-Schwartz 不等式, 27, 35, 47  
Cayley-Hamilton 定理, 415  
CP 分解, 585, 599  
交替最小二乘算法, 607  
数学表示形式, 603  
正则交替最小二乘算法, 609  
CS 分解, 309
- D**
- 代价函数, 193  
代数多重度, 402  
代数向量, 2  
带型矩阵, 112  
单调, 214  
强单调, 214
- 严格单调, 214  
单元素集, 241  
单位矩阵, 3  
单位球体, 212  
单位上三角矩阵, 113  
单位下三角矩阵, 113  
单元素集, 241  
等价矩阵, 508  
等价矩阵束, 435  
等价张成集, 484  
等价子空间类, 508  
第 1 友型, 455  
第 2 友型, 456  
低秩与稀疏矩阵分解, 314  
低秩总体最小二乘解, 341  
笛卡儿积, 17  
典范内积, 26  
典范/平行因子分解, 见“CP 分解”  
迭代 Tikhonov 正则化, 331  
迭代矩阵, 227  
叠加原理, 21  
动量, 237  
独立法则, 177  
独立同分布, 41  
独立性基本假设, 148  
对称矩阵, 4  
对合矩阵, 6  
对合性, 105  
对角矩阵, 3  
对角线法, 46  
对偶变量, 217  
对偶范数, 245  
对偶目标函数, 218  
对偶残差, 265  
对偶间隙, 219  
对偶可行性, 264  
对偶平均算法, 247

- 对偶上升法, 256  
 对偶问题, 217  
 对偶向量空间, 244  
 对偶正规矩阵, 279  
 对偶最优点, 219  
 对应列 Kronecker 积, 见 “Khatri-Rao 积”  
 多重度, 400  
 Davidon-Fletcher-Powell 算法, 267  
 Duncan-Guttman 求逆公式, 57
- E**
- 二次矩阵多项式, 454  
 二次矩阵方程, 455  
 二次特征值问题, 452  
 二次特征值问题求解, 454  
 分解法, 454  
 线性化方法, 455  
 二次型, 44  
 二阶辨识表, 167  
 二阶锥, 226  
 Euclidean 空间, 25  
 Euclidean 球, 226  
 ESPRIT, 437  
 基本 ESPRIT 算法 1, 439  
 基本 ESPRIT 算法 2, 442  
 TLS-ESPRIT, 439
- F**
- 法向量, 344  
 反 Hermitian 矩阵, 101  
 反 Tikhonov 正则化, 331  
 范数, 24  
 非负性, 24  
 齐次性, 24  
 三角不等式, 24  
 正性, 24  
 范数公理, 24  
 仿射函数, 213  
 仿射象, 212  
 仿射组合, 213  
 仿酉矩阵, 109  
 非负矩阵, 355  
 非负矩阵分解, 359  
 乘法算法, 364  
 交替非负最小二乘算法, 371  
 Nesterov 最优梯度法, 370  
 投影梯度算法, 369  
 非负象限, 212  
 非负张量, 613  
 非负张量分解, 613  
 非负张量 CP 分解, 616  
 乘法算法, 616  
 交替最小二乘算法, 618  
 非负张量 Tucker 分解, 614  
 乘法算法, 614  
 交替最小二乘算法, 618  
 非负性, 23  
 非负性约束, 355  
 非固定迭代法, 227  
 非积极约束, 389  
 非扩张熵, 363  
 非平凡解, 13  
 非平滑凸优化问题, 216  
 非奇异矩阵, 46  
 非相干的, 84  
 非相干性, 84  
 非一致方程, 51  
 非正则  $\lambda$  矩阵, 453  
 分段正交匹配追踪, 385  
 分割 Bregman 迭代算法, 395  
 分割优化问题, 252  
 分块非线性 Gauss-Seidel 法, 见 “GS 法”  
 分块矩阵, 3  
 分块协同下降法, 333  
 分散算法, 264

- 符号差, 407  
符号矩阵, 111  
复 Hessian 矩阵, 179  
部分 Hessian 矩阵, 181  
全 Hessian 矩阵, 180  
主 Hessian 矩阵, 181  
复 Hessian 矩阵辨识, 189  
复解析函数, 171  
复矩阵方程求解, 13  
复矩阵微分, 175  
复随机向量, 37  
复随机向量的边缘概率密度, 37  
复随机向量的累积分布函数, 37  
复梯度矩阵, 178  
负定矩阵, 45  
负曲率方向, 208  
赋范向量空间, 24  
Fejer 定理, 69  
Fenchel 不等式, 245  
Fischer 不等式, 47, 102  
FISTA 算法, 255  
Fourier 矩阵, 129  
Fourier 矩阵性质, 124
- G**
- 概率向量, 142  
感知基, 82  
感知矩阵, 82  
高阶奇异值分解, 585  
高阶正交迭代 (HOOI) 算法, 595  
HOSVD 算法, 595  
高斯随机向量, 41  
特征函数, 42  
性质, 42  
高斯消去法, 11  
共轭对称性, 23  
共轭 Jacobian 矩阵, 178  
共轭 Jacobian 矩阵辨识, 182, 186
- 共轭梯度, 206  
共轭梯度矩阵, 178  
共轭梯度矩阵辨识, 182, 186  
共轭梯度 (CG) 算法, 227  
共轭函数, 244  
共轭梯度算子, 176  
共轭梯度向量, 176  
共轭协梯度算子, 176  
共轭性, 227  
共形矩阵, 84  
固定迭代法, 227  
估计序列, 237  
孤立局部极值点, 195  
惯性, 407  
广义 Bezout 定理, 454  
广义逆矩阵, 62  
弱广义逆矩阵, 63  
性质, 63  
正规化广义逆矩阵, 63  
自反广义逆矩阵, 63  
广义奇异值分解, 305  
GSVD 算法1, 308  
GSVD 算法2, 308  
GSVD 算法3, 309  
广义特征对, 433  
广义特征多项式, 433  
广义特征方程, 433  
广义特征值分解, 432  
广义特征向量, 433  
广义特征值, 433  
广义特征值问题, 433  
广义置换矩阵, 107  
广义 Rayleigh 商, 447  
 $g$  矩阵, 见“广义置换矩阵”  
Gauss-Markov 定理, 327  
Grassmann 流形, 510  
GS 法, 333

**H**

函数向量, 2  
 黑盒优化, 216  
 行列式, 45  
 后向预测误差向量, 542  
 后向预测值向量, 542  
 互补松弛性, 219  
 互换矩阵, 104  
 互相干参数, 84  
 互相关矩阵, 39  
 互协方差矩阵, 39  
 回溯直线搜索, 268  
 Hadamard 不等式, 47, 102  
 Hadamard 积, 68  
 Hadamard 积定理, 68  
 Hadamard 矩阵, 129  
     规范化 Hadamard 矩阵, 129  
 Hankel 矩阵, 136  
 Hermite 标准型, 9  
 Hellinger 距离, 362  
 Helmert 矩阵, 139  
 Hermitian 矩阵, 4  
 Hermitian Toeplitz 矩阵, 133  
 Hessian 矩阵, 197, 164  
 Hessian 矩阵辨识, 167  
     辨识定理, 169  
 Hilbert 矩阵, 142  
 Hilbert 空间, 25, 36  
 Hotelling 变换, 428  
 Householder 矩阵, 112  
 Huber 函数, 250

**I**

Itakura-Saito 散度, 362

**J**

基本向量, 3  
 基数, 82

基追踪去噪, 390  
 几何多重度, 402  
 几何向量, 2  
 积极约束, 389  
 集合, 16  
     并集, 16  
     补集, 17  
     差集, 17  
     超集, 16  
     交集, 16  
     空集, 16  
 迹, 49  
 极大-极小化问题, 218  
 极化恒等式, 27, 35  
 极式分解, 492  
 极式分解算法, 322  
 极限点, 333  
 极小-极大化问题, 217  
 极值, 194  
 极值点, 194  
     一阶必要条件, 200  
 加法交换律, 5, 17  
 加法结合律, 5, 17  
 加权对偶平均算法, 248  
 加权最小二乘, 281  
 价值向量, 244  
 减次矩阵, 402  
 降维, 431  
 交叉对角线, 3  
 交换矩阵, 75  
 交替方向乘子法, 264  
 交替正则化非负最小二乘, 372  
 交替最小二乘法,  
     解释变量, 79  
 阶梯型矩阵, 9  
     简约阶梯型矩阵, 9  
 结构优化方法, 216

- 截尾 Newton 法, 266  
截尾奇异值分解, 289  
近邻, 31  
近邻分类法, 31  
近似联合对角化, 463  
紧致框架, 84  
矩形集, 226  
矩阵, 1  
    导数, 7  
    复共轭转置, 4  
    宽矩阵, 2  
    高矩阵, 2  
    高阶导数, 7  
    积分, 7  
    正定性判据, 102  
    转置, 4  
矩阵变换, 21  
矩阵等式, 46~46  
矩阵的性能指标, 54  
矩阵对, 433  
矩阵范数, 34  
    核范数, 316  
    行和范数, 34  
    列和范数, 34  
    谱范数, 34  
    三角不等式, 24, 33  
    诱导范数, 33  
    酉不变范数, 316  
    元素形式范数, 34  
    最大范数, 34  
    Frobenius 范数, 34  
    Mahalanobis 范数, 35  
    Schatten 范数, 316  
     $l_p$  范数, 33  
矩阵范数-对偶矩阵范数对, 246  
矩阵函数, 7  
    对数函数, 7  
三角函数, 7  
指数函数, 7  
矩阵化函数, 76  
矩阵恢复, 314  
矩阵幂, 418  
矩阵求逆, 14  
    高斯消去法, 14  
    矩阵求逆引理, 56  
    分块矩阵求逆引理, 57  
    增广矩阵求逆引理, 56  
    Sherman-Morrison 公式, 56  
    Woodbury 公式, 56  
矩阵束, 433  
矩阵完备, 313  
矩阵微分, 152  
    计算公式, 152  
矩阵 Rayleigh 商, 512  
均值向量, 38  
局部极小点, 194  
    二阶必要条件, 201  
局部极大值, 194  
局部极小值, 194  
局部最优点, 210  
绝对极小点, 194  
 $J$  正交矩阵, 111  
Jacobian 矩阵, 155  
Jacobian 矩阵辨识, 153  
    命题, 421, 162  
Jacobian 行列式, 155  
Jacobian 算子, 146  
Jensen 不等式, 214  
Jordan-Wielandt 定理, 411
- K**
- 开球体, 212  
可辨识的, 326  
可对角化, 413  
可对角化定理, 414

- 可加性, 362  
 可交换矩阵, 142  
 可解释性, 356  
 可行点, 210  
 可行点启动原始-对偶内点法, 276  
 可行点算法, 276  
 可行集, 210  
 快速子空间分解算法, 521  
 扩张熵, 362  
 Kantorovich 不等式, 482  
 Karhunen-Loeve 变换, 469  
 Karhunen-Loeve 展开, 469  
 Karush-Kuhn-Tucker (KKT) 条件, 219  
 Khatri-Rao 积, 74  
 KKT 点, 219  
 KL 散度, 361  
 Kronecker 积, 71
  - 右 Kronecker 积, 71
  - 左 Kronecker 积, 71
  - 性质, 71~72
 Krylov 矩阵, 518  
 Krylov 子空间, 227  
 Krylov 子空间方法, 227  
 Kruskal 秩, 605  
 Kullback-Leibler 散度, 见“KL”散度
- L**
- 离散 Karhunen-Loeve 变换, 428  
 联合对角化, 463  
 联合对角化器, 464  
 链式法则, 148  
 良好定义的, 335  
 良态矩阵, 287  
 列等价矩阵, 15  
 列阶梯型, 15  
 临界点, 334  
 邻域, 194  
 零化多项式, 415
- 零矩阵, 3  
 零空间, 494  
 零维, 494  
 $l_0$  拟范数最小化, 377  
 $l_1$  范数球, 226  
 $L_2$  空间, 36  
 $L_2$  理论, 36  
 Lanczos 算法, 435  
 LARS 算法, 388  
 Laplace 方程, 171  
 LASSO 算法, 386  
 LQ 分解, 555  
 Lagrangian 乘子法, 217
  - 增广 Lagrangian 乘子法, 261
  - Lagrangian 乘子向量, 217
  - Lagrangian 对偶法, 219
 Lipschitz 常数, 233  
 Lipschitz 连续, 233  
 Lipschitz 连续函数类, 233  
 $\lambda$  矩阵, 453
- M**
- 码矢, 356  
 码书, 357  
 满行秩, 53  
 满列秩, 53  
 满秩, 53  
 满秩分解, 65  
 盲矩阵方程, 353  
 迷向分布, 425  
 迷向圆变换, 424  
 幂等矩阵, 6, 531  
 幂零矩阵, 532  
 幂单矩阵, 6  
 密码文本, 80  
 明码文本, 80  
 目标函数, 193  
 模式向量, 30

- 模式- $n$  向量, 566
- Mahalanobis 距离, 31
- Mangasarian-Fromovitz 约束规定, 221
- Markov 矩阵, 142
- Minkowski 不等式, 47, 103
- Moore-Penrose 逆矩阵, 62
- 递推算法, 66
  - KL 分解法, 65
- Moreau 分解, 253
- $\mu$  中心, 276
- N**
- 泥沼, 335
  - 拟 Newton 法, 267
  - 逆仿射象, 212
  - 逆矩阵, 6
  - 逆问题, 22
  - 拟凸函数, 334
  - 内点, 196
  - 内点条件, 276
  - 内罚函数法, 259
  - 内集, 196
  - 内积, 23
  - 内积向量空间, 23
  - Nesterov 最优梯度法, 237
    - Nesterov 第 1 最优梯度法, 238
    - Nesterov 第 2 最优梯度法, 240
    - Nesterov 第 3 最优梯度法, 240
  - Newton 法, 224
    - 不可行点启动 Newton 法, 271
    - 可行点启动复 Newton 算法, 273
    - 可行点启动 Newton 算法, 270
  - Newton-Raphson 算法, 224
  - Neyman  $\chi^2$  距离, 362
  - $N$  阶奇异值分解, 587
  - $n$ -模式矩阵积, 581
- O**
- off 函数, 464
  - Oppenheim 不等式, 102
  - Ostrowski-Taussky 定理, 103
- P**
- 徘徊现象, 335
  - 排序不确定性, 604
  - 偏导算子, 144
  - 偏导向量, 144
  - 膨胀映射, 21
  - 匹配追踪,
  - 平凡解, 13
  - 平凡子空间, 483
  - 平滑凸优化问题, 216
  - 平稳点, 195
  - 平行四边形法则, 27
  - 评分与协同筛选, 315
  - 评价函数, 279
  - 逼近点版本, 335
  - 逼近函数, 243
    - 归一化逼近函数, 243
  - 逼近梯度算法, 254
  - 逼近映射, 252
  - 逼近中心, 243
  - 谱半径, 406
  - Pearson  $\chi^2$  距离, 362
  - Pisarenko 谐波分解, 426
  - Pythagorean 定理, 29, 35
- Q**
- 齐次线性方程组, 13
  - 齐次性约束, 508
  - 奇异, 8
  - 奇异值分解, 288
  - 奇异值阈值化, 255
  - 前向预测误差向量, 541
  - 前向预测值向量, 541

- 欠定方程, 52  
 潜在语义检索, 315  
 强单调, 214  
 强对偶性, 219  
 强局部极小点, 见“严格局部极小点”  
 切空间, 318  
 求和向量, 115  
 全变分, 391  
 全纯函数, 171  
 全局极小点, 194  
 全局极小值, 194  
 全奇异值分解, 289  
 全息字典, 83  
 全微分, 154  
 曲率, 208  
 q 对数, 363  
 q 分布, 363  
 q 指数, 363  
 Q 收敛速率, 231  
 QR 分解, 309
- R**
- 人脸识别, 80, 314  
 容量矩阵, 56  
 弱对偶性, 219  
 弱局部极小点, 见“局部极小点”  
**Rayleigh 商, 443**  
 推广的 Rayleigh 商, 512  
 梯度, 444  
**Rayleigh 商的重要性质, 443**  
 平移不变性, 443  
 齐次性, 443  
 有界性, 443  
 正交性, 443  
 最小残差, 443  
**Rayleigh商迭代, 445**  
**Rayleigh商问题求解, 445**  
 共轭梯度算法, 446
- 梯度算法, 446  
**Rayleigh 序列, 445**  
 尺度不变性, 445  
 平移不变性, 445  
 西相似性, 445  
**Rayleigh-Ritz 定理, 443**  
**Rayleigh-Ritz 比, 443**  
**Rayleigh-Ritz 逼近, 518**  
**Rayleigh-Ritz (RR) 向量, 518**  
**Rayleigh-Ritz (RR) 值, 518**  
 Richardson 迭代, 227  
 Rudin-Osher-Fatemi (ROF) 去噪模型, 391
- S**
- 三 Lanczos 迭代, 519  
 三阶奇异值分解, 588  
 散度, 360  
 商奇异值分解, 306  
 上确界, 217  
 上三角矩阵, 113  
 上 Hessenberg 矩阵, 113  
 生成元, 483  
 矢量量化, 356  
 视频监控, 314  
 适定方程, 52  
 胜者赢得一切, 357  
 搜索点序列, 238  
 搜索方向, 223  
 实随机向量, 2, 36  
 边缘概率密度函数, 37  
 概率密度函数, 36  
 收敛速率, 231  
 极限收敛速率, 231  
 超线性收敛速率, 231  
 二次收敛速率, 232  
 三次收敛速率, 232  
 次线性收敛速率, 231  
 线性收敛速率, 231

- 局部收敛速率, 232  
次线性速率, 232  
二次速率, 232  
线性速率, 232  
首项元素, 9  
首一多项式, 415  
首一元素, 9  
数据阵列压缩, 612  
数据最小二乘, 330  
数值稳定性, 286  
输入重生, 399  
双共轭梯度法, 228  
双 Lanczos 迭代, 521  
双曲对称性, 112  
双线性分析, 596  
双线性模型, 596  
水平切片, 566  
水平切片矩阵, 567  
“死亡”惩罚, 260  
松弛变量, 263  
松弛法, 331  
松弛序列, 193  
缩放对偶向量, 265  
Schur 不等式, 50  
Shannon 熵, 362  
Sherman-Morrison 公式, 56  
Slater 定理, 221  
Slater 条件, 221  
Stiefel 流形, 511  
Sturmian 分离定理, 481  
SVD-TLS 算法, 341
- T  
贪婪算法, 384  
特解, 13  
特征对, 400  
性质, 410~411  
特征多项式, 401
- 特征方程, 401  
特征根, 401  
特征空间, 487  
特征系统, 433  
特征向量, 48, 400  
性质, 410~411  
特征值, 48, 401  
半单特征值, 402  
单特征值, 402  
多重特征值, 402  
条件数, 405  
性质, 408~410  
特征值分解, 400  
特征值-特征向量方程式, 400  
梯度, 196  
梯度矩阵, 146  
梯度流, 146  
梯度算子, 145, 176  
梯度投影法, 225  
收敛速率上界, 235  
调和函数, 172  
条件数, 286  
同构, 23  
同构映射, 23  
同伦算法, 389  
通解, 12  
统计保真度, 356  
统计不相关, 40  
投影, 225  
投影次梯度法, 248  
投影定理, 528  
几何解释, 529  
投影共轭梯度算法, 231  
投影矩阵, 530  
二阶偏导数, 538  
更新公式, 545  
一阶偏导数, 537

- 投影梯度, 540  
 投影梯度法, 见“梯度投影法”  
 投影算法, 322  
 凸包, 212  
 凸函数, 213  
     强凸函数, 214  
     严格凸函数, 213  
 凸集, 211  
 凸性参数, 214  
 凸优化问题, 216  
 凸组合, 213  
 凸锥, 213  
 图形化建模, 314  
 退化特征值, 402  
 Taylor 级数展开, 195  
 Tanimoto 测度, 32  
 Tikhonov 正则化, 331  
 Tikhonov 正则化解, 332  
     性质, 332  
 TLS 算法, 338  
 Toeplitz 矩阵, 132  
     Hermitian Toeplitz 矩阵, 133  
     快速离散余弦变换, 134  
     离散余弦变换, 134  
 斜 Hermitian Toeplitz 矩阵, 133  
 斜 Hermitian 型 Toeplitz 矩阵, 133  
 Tsallis 对数, 见“q 对数”  
 Tsallis 熵, 362  
 Tsallis 数理统计, 363  
 Tucker 分解, 585  
     Tucker1 分解, 592  
     Tucker2 分解, 592  
     Tucker3 分解, 592  
 乘法算法, 615  
 交替最小二乘算法, 593  
 Tucker 积, 579  
 Tucker 算子, 585
- V
- Vandermonde 矩阵, 120~122  
 求逆公式, 122
- W
- 外罚函数法, 259  
 外积, 38  
 伪可加性, 363  
 违法约束, 389  
 唯一表示定理, 470  
 稳健主分量分析, 317  
 无交连, 485  
 无限维向量子空间, 485  
 无约束最小化问题, 206  
     极值点一阶必要条件, 200  
     局部极小点二阶必要条件, 201  
     局部极小点二阶充分条件, 201  
 物理向量, 2  
 完备性, 25  
 完备向量空间, 25  
 完备正交基, 78  
 Weyl 定理, 412  
 Wirtinger 偏导, 172  
 Woodbury 公式, 56
- X
- 稀疏逼近, 80  
 稀疏编码, 81  
     过完备编码, 81  
     临界完备编码, 81  
 稀疏表示, 80  
 稀疏分解, 79  
     最稀疏表示, 80  
 稀疏化, 81  
 稀疏矩阵, 78  
 稀疏向量, 78  
 稀疏性, 83  
 稀疏性约束, 355

- 下 Hessenberg 矩阵, 113  
下确界, 218  
下三角矩阵, 113  
下无界, 218  
线性化 Bregman 迭代算法, 394  
线性流形, 530  
线性无关, 8  
线性无关约束限制, 221  
线性相关, 8  
线性变换, 21  
线性映射, 21  
线性主部, 154  
线性组合, 8  
现代内点法, 277  
显式约束, 210  
相对可行内点集, 260  
相对内点, 221  
相对内域, 221  
相干, 40  
相关系数, 40  
相合变换, 119  
相合规范型, 119  
相合矩阵, 119  
    传递性, 119  
    对称性, 119  
    规范相合矩阵, 119  
    自反性, 119  
相似变换, 117  
相似不变量, 50  
相似度, 30  
相似范式, 413  
相似矩阵, 117  
    传递性, 117  
    对称性, 117  
    自反性, 117  
相异度, 30  
响应变量, 79  
向量, 1  
行向量, 1  
列向量, 1  
夹角, 29  
向量范数, 27  
极大范数, 28  
Euclidean 范数, 27  
Hölder 范数, 28  
 $l_0$  范数, 27  
 $l_1$  范数, 27  
向量范数-对偶向量范数对, 246  
向量化函数, 74  
向量外积, 578  
协变算子, 146  
协梯度矩阵, 146  
协梯度算子, 176  
协梯度向量, 146  
斜对称矩阵, 见“交叉对称矩阵”  
斜率参数, 344  
斜投影算子, 545  
几何解释, 551  
满行秩矩阵的斜投影算子, 553~557  
满列秩矩阵的斜投影算子, 545~552  
物理含义, 551  
修正 Newton 法, 267  
形式偏导, 172  
选择矩阵, 108  
旋转算符, 438  
学习算法, 209
- Y
- 压缩采样匹配追踪, 385  
压缩感知, 82  
压缩映射, 21  
严格单调, 214  
严格可行集, 见“可行内集”  
严格拟凸函数, 334  
严格平方可积分函数, 486

- 严格局部极小点, 194, 197  
 严格绝对极小点, 194  
 一阶黑盒优化, 222  
 一致方程, 51  
 移位矩阵, 106  
 因子得分, 596  
 因子得分矩阵, 597  
 因子载荷, 596  
 因子载荷矩阵, 597  
 映射, 20  
 单射, 21  
 满射, 21  
 逆映射, 21  
 始集, 20  
 上域, 20  
 像, 见“值域”  
 像点, 20  
 一对一映射, 21  
 域, 见“始集”  
 值, 见“像点”  
 值域, 20  
 终集, 20  
 有界变差范数, 391  
 有界变差函数, 390  
 有序对, 17  
 有序  $n$  元组, 17  
 优化向量, 193  
 友矩阵, 482  
 右解, 455  
 右逆矩阵, 60  
 右奇异向量, 289  
 右奇异向量矩阵, 289  
 右伪逆矩阵, 60  
 阶数递推, 61  
 西变换, 109  
 西不变, 28  
 西等价, 110  
 西矩阵, 109  
 余子式, 45  
 预处理共轭梯度 (PCG) 算法, 230  
 原始变量, 217  
 原始残差, 265  
 原始代价函数, 218  
 原始-对偶内点法, 276  
 二阶原始-对偶内点法, 277  
 一阶原始-对偶内点法, 276  
 原始问题, 217  
 原始向量空间, 244  
 原始最优点, 219  
 原子, 78  
 约束非负矩阵分解, 372  
 约束最优化问题, 209
- Z**
- 增广矩阵, 11  
 增广乘子法, 261  
 张成集, 483  
 张成集定理, 483  
 张量, 563  
 $n$  阶张量, 563  
 超对称张量, 565  
 单位三阶张量, 565  
 二阶张量, 563  
 癫痫张量, 564  
 可分解张量, 583  
 核心张量, 586  
 零阶张量, 563  
 矩阵化, 569  
 模式- $n$  秩, 583  
 三阶张量, 565  
 水平展开, 569  
 Kiers 水平展开, 569  
 Kolda 水平展开, 571  
 LMV 水平展开, 570  
 一阶张量, 563

- 张量范数, 577  
张量化, 576  
张量脸, 564  
张量内积, 577  
张量外积, 578  
秩, 584  
秩分解, 584  
纵向展开, 573  
Kiers 纵向展开, 573  
Kolda 纵向展开, 574  
LMV 纵向展开, 573  
张量分析, 563  
张量积, 见“Kronecker积”  
张量纤维, 566  
管纤维, 566  
行纤维, 566  
列纤维, 566  
竖直纤维, 566  
水平纤维, 566  
纵深纤维, 566  
障碍(函数)法, 259  
障碍函数, 259  
对数障碍函数, 259  
幂函数障碍函数, 259  
逆障碍函数, 259  
指数障碍函数, 259  
指示函数, 245  
支撑函数, 247  
正定矩阵, 45  
正矩阵分解, 359  
阵列, 563  
二路阵列, 563  
多路阵列, 563  
一路阵列, 563  
正规矩阵, 111  
正规锥, 246  
正交, 40  
常数向量正交, 28  
几何解释, 30  
函数向量正交, 29  
随机向量正交, 29  
物理意义, 30  
正交补空间, 486  
正交等价, 110  
正交多分辨分析, 487  
正交分解, 531  
正交极因子, 492  
正交矩阵, 109  
正交匹配追踪, 655  
正交强迫一致问题, 491  
正交群, 511  
正交投影矩阵, 537  
更新公式, 545  
正交投影算子, 488, 533  
正交约束, 508  
正交子空间, 486  
正面切片, 566  
正面切片矩阵, 567  
正切算法, 435  
正态随机向量, 见“高斯随机向量”  
正向问题, 22  
正则化参数, 331  
正则化方法, 331  
正则化路径, 332  
正则化约束总体最小二乘图像恢复, 351  
正则条件, 300  
正则矩阵三元组, 300  
正则  $\lambda$  矩阵, 453  
正则正交匹配追踪, 655  
直和, 67  
直和分解, 533, 547  
直和性质, 67  
直积, 见“Kronecker积”  
值域, 20

- 置换矩阵, 103  
 秩, 495  
 秩不等式, 53  
 秩等式, 53  
 秩定理, 498  
 秩亏缺, 53  
 秩-稀疏非相干性, 318  
 秩 1 更新算法, 267  
 中心化, 610  
 中心化矩阵, 116  
 中心路径, 276  
 中央对称矩阵, 105  
 中央复共轭对称矩阵, 101  
 中央 Hermitian 矩阵, 101  
 重球法, 237  
 主对角线, 3  
 主分量分析, 431, 597  
 主角, 490  
 主元列, 10  
 主元位置, 10  
 主子式, 45  
 祖母细胞编码, 357  
 逐点极大函数, 242  
 子空间, 18  
 子空间的维, 485  
 子空间套, 487  
 子空间追踪算法, 386  
 字典, 79  
 自反广义逆矩阵, 97  
 自相关矩阵, 38  
 自协方差矩阵, 39  
 字典式排序, 74  
 总体最小二乘, 336  
 总体最小二乘拟合, 344  
 总体最小二乘解, 337  
 组合系统辨识, 314  
 组合优化问题, 252
- 锥包, 212  
 锥组合, 213  
 最大化问题, 193  
 最大似然解, 329  
 最陡下降法, 209, 223  
 最小多项式, 415  
 最小二乘解, 67  
 最小范数解, 67, 339  
 最小  $l_0$  范数解, 79  
 最小  $l_1$  范数解, 382  
 最小  $l_2$  范数解, 79  
 最小范数解的 TLS 算法, 339  
 最小范数最小二乘解, 67  
 最小化问题, 193  
 最小角度, 490  
 最小输出能量准则, 539  
 最小值原理, 211  
 最优对偶值, 219  
 最优原始值, 218  
 最优值, 210  
 最优最小二乘近似解, 339  
 左解, 455  
 左逆矩阵, 60  
 左奇异向量, 289  
 左奇异向量矩阵, 289  
 左伪逆矩阵, 60  
 阶数递推, 60  
 坐标, 470  
 坐标系, 470, 485  
 坐标向量, 471  
 作用集, 389

