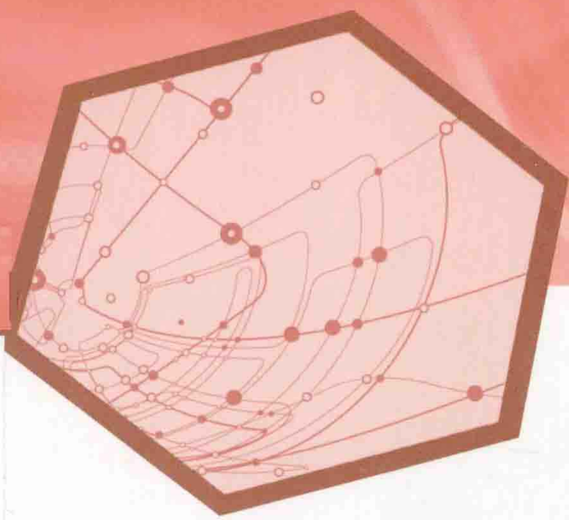
Matrix Analysis and Computation

矩阵分析与计算

李继根 张新发 编著



WUHAN UNIVERSITY PRESS
武汉大学出版社

矩阵分析与计算

李继根 张新发 编著

武汉大学出版社

图书在版编目(CIP)数据

矩阵分析与计算/李继根,张新发编著. —武汉:武汉大学出版社,2013.10
ISBN 978-7-307-11546-0

Ⅰ.矩… Ⅱ.①李… ②张… Ⅲ.①矩阵分析—高等学校—教材 ②矩阵—计算方法—高等学校—教材 Ⅳ.①O151.21 ②O241.6

中国版本图书馆 CIP 数据核字(2013)第 210341 号

出版:**武汉大学出版社** (430072 武昌 珞珈山)
(电子邮件:cbs22@whu.edu.cn 网址:www.wdp.com.cn)
印刷:安徽新华印刷股份有限公司
开本:787×1092 1/16 印张:23.75 字数:546 千字
版次:2013 年 10 月第 1 版 2013 年 10 月第 1 次印刷
ISBN 978-7-307-11546-0 定价:60.00 元

前 言

在著名的博文"理解矩阵"中,作者孟岩一上来就提出了令人困惑的问题:矩阵究竟是什么东西? 这看似一个简单的问题,但其实很难回答,因为众所周知,越是简单的问题越难以说清楚.

最直观的说法就是"矩阵是一堆数,只是在其上规定了一些运算",这有点类似于面向对象编程中的"类(class)"的概念.究其原因,乃是矩阵从逻辑上可视为是从"自然数→实数→复数→超复数→向量→矩阵"推广而来,这也与其历史发展轨迹大致相当.大家知道,人类对于数的研究已经贯穿了几千年的文明史,更遑论内涵更大、应用更广的"向量"乃至"矩阵". 自上世纪末,G. W. 斯图尔特就拟以 5 册之巨著,戮力于阐释"矩阵算法".其实依我们愚见,关于矩阵"这堆数",即使来个"百集连续剧"也难以穷尽.因为"凡有井台处皆能歌柳词",矩阵的"芳踪"正如柳永的词一般,"凡有多元处必有矩阵". 如今关于矩阵的知识早已成为工程技术人员必备的数学基础知识,有人甚至认为它就是"研究生的线性代数 + 高等数学".这也说明线性代数和高等数学是学习矩阵分析课程的必备基础.

基于自身的工科背景,孟岩指出"矩阵是线性空间里的变换的描述",并结合计算机图形学知识对之进行了详细阐释.事实上,从更高的层面看,矩阵是泛函中的一种特殊的线性算子,可谓"矩阵即变换".了解泛函分析的读者都知道,这已经是用泛函的眼光来看待矩阵了.一般可将矩阵的理论知识大致划分为"空间与变换"、"矩阵分解论"以及"矩阵分析论",其中"空间与变换"依托的就是泛函分析与抽象代数,因此以变换的眼光看待矩阵,自然就具有高屋建瓴之势.这就意味着我们在教材编写和实际教学中,既要通过大量实例丰富读者的感性认识,又要极力渗透泛函分析的初步知识,以使读者逐步养成"一览众山小"的视野.当然,不同学科背景的读者自然会收获不同的"感性+理性".

在名著《古今数学思想》中,M. 克莱因写道:

(行列式与矩阵)在数学上并不是大的改革……尽管行列式和矩阵用做紧凑的表达式,尽管矩阵在领悟群论的一般定理方面具有作为具体的群的启发作用,但它们都没有深刻地影响数学的进程.然而已经证明这两个概念是高度有用的工具,现在是数学器具的一部分.

在该书出版的 1972 年,他的观点无可厚非.但计算机的飞速发展和信息社会的如火如荼,使得矩阵世界在这几十年里发生了翻天覆地的变化.

首先是矩阵计算(又称数值线性代数)的异军突起.计算机的横空出世给线性代数研究带来了新的机遇和挑战,极大地促进了矩阵计算乃至科学计算的兴起和发展,使得原本被许多人认为已经"寿终正寝"的线性代数"枯木逢春".反过来,矩阵计算的研究以及 Matlab 等计算软件的不断改进,更是成为大规模、高速、并行、移动、网络计算中的得力工

具,这进一步促进了诸如“模型降阶”等化解“维数之咒”的新兴课题的蓬勃发展. 世界顶尖的数值分析学家 L. N. Trefethen 早在 1997 年就深刻地指出:

如果除了微积分与微分方程之外,还有什么领域是数学科学的基础的话,那就是数值线性代数.

世界上最大的专业学术组织 IEEE(电器与电子工程师学会)主办的《科学与工程计算》杂志,在 2000 年评选出了“20 世纪十大算法”,其中就有 3 个与矩阵计算直接相关,它们是 1950 年提出的 Krylov 子空间迭代法、1951 年提出的矩阵计算的分解方法以及 1959 年至 1961 年间提出的计算矩阵特征值的 QR 算法.

此外,吴文俊先生的数学史研究,校正了人们对数学世界的看法. 受西方中心论的束缚,之前人们一直认为“中国古代数学著作都是应用问题集”,“在古代中国的数学思想中,最大的缺点是缺少严格求证的思想”. 吴文俊先生通过大量分析,明确推出“近代数学之所以能够发展到今天,主要是靠中国(式)的数学,而非希腊(式)的数学,决定数学历史发展进程的主要是靠中国(式)的数学,而非希腊(式)的数学”,并将中世纪数学发展过程概括为:

$$\left.\begin{array}{l}\text{中国}\xrightarrow{5c}\text{印度}\\ \text{希腊}\xrightarrow{9c}\text{阿拉伯}\end{array}\right\}\xrightarrow{10c}\text{欧洲}$$

其中,c 表示世纪. 他深刻地指出:

从数学有史料为依据的几千年发展过程来看,以公理化思想为主的演绎倾向以及以机械化思想为主的算法倾向互为消长.

……

我国传统数学有其自身的发展途径与独到的思想体系,而以机械化为其特色;方程求解尤其是贯穿两千多年发展中的一条主线. 这与遵循古希腊传统的西方数学的公理化演绎体系大相径庭,旨趣迥异. 在历史长河中,数学机械化算法体系与数学公理化演绎体系曾多次反复互为消长交替成为数学发展中的主流.

这就从理论上回答了什么是世界数学发展的主流问题. 当然,与吴文俊先生的研究遥相辉映的是,当时科学哲学领域也兴起了以库恩为代表的新历史主义学派.

吴文俊先生的研究更使我们明确意识到:对待演绎体系与算法体系,合理的态度应该是取两者之长,兼收并蓄,而不能厚此薄彼,褒一贬一. 比如矩阵计算领域的奠基性大师豪斯霍尔德,最先研究的是泛函分析,之后转向生物数学,最终建功立业于矩阵计算领域. 事实上,深厚的数学基础和丰富的计算机知识是从事矩阵计算的必要条件.

另外,布尔巴基学派的“结构数学”也给人们带来了极大的冲击. 自 20 世纪 30 年代兴起的结构数学,在系统地整理了数学知识的同时,更给数学教育带来了“新数学运动”. 但由于“新数学运动”一味注重形式上的严格性,忽视乃至抹杀数学的直觉性,使得学生的数学学习跌跌撞撞,步履踉跄,满满的都是泪水,学生也恍如杂技中表演钻火圈游戏的小白鼠,在考试的皮鞭挥舞之下拼命奔跑,完全变成了枯燥的规则的奴隶.

正是基于上述认识,我们认为,在矩阵分析课程的教学中,我们更需要的是“返璞归真改变思维”,是“抛弃各种有形和无形的思想枷锁”(钱旭红语). 因此在本教材的体系、选材和编写中,我们力求突出以下特点:

1. 重新整合内容体系，兼顾矩阵计算

从实践角度看，矩阵分析处理的是线性化以后的非线性问题，它已经深入到了数值求解的每个领域，其后续是数值分析（包括矩阵计算、微分方程数值解等分支学科）的理论、算法及其语言实现. 因此我们认为，应该将矩阵分析类课程看成是实践性很强的理论性课程. 然而由于方方面面的制约，目前已有的教材都很少涉及实践性方面，而这正是我们希望通过本书加以弥补并大力宣扬的目标之一. 为此，我们精心挑选了矩阵计算中处理三大核心问题（线性方程组求解、最小二乘问题和特征值问题）的一些最重要的数值方法，尽可能详细地阐明它们的设计思想和理论依据，并重新整合矩阵分析的内容，尽可能使两者无缝连接，以避免给人以突兀之感. 例如，在回顾了线性方程组的知识之后，我们通过高斯消元法，引入了 LU 分解，进而自然地导出了线性方程组的数值求解问题.

在本书中，我们还通过分析 Matlab 软件的内置函数和帮助文档，将其中已公开的实现算法中所涉及的矩阵计算理论细节与课程教学紧密联系起来，以便读者理解计算结果，例如 Jordan 标准型与内置函数 jordan、矩阵指数与内置函数 expm、LU 分解与内置函数 lu、QR 分解与内置函数 qr、奇异值分解与内置函数 svd，等等. 对于本书中所使用的 Matlab 代码，需要的读者请来邮索取.

2. 注重启发式教学，力争将“冰冷的美丽”转变成“火热的思考”

以定义开头，继之以定理和公式，再辅以应用，这似乎成了数学类教材的典型模式. 这种形式化的编书方式，经常让学生觉得数学概念就像孙悟空一样，是从石头缝里蹦出来的. 对这种“空降部队”，学生从心理上难以接受，也使得“从问题出发抽象出数学知识再回到问题”这种丰富多彩的数学思维活动被“掐了两头，只剩中间”，变成了纯粹的逻辑推理，从定理到定理，从结论到结论. 作为面向非数学类工科学生的教材，如果也采取这种模式，对学生而言，无异于梦魇.

事实上，作为启发式教学的重要辅助工具，教材必须充分反映学生的思维过程，要通过一系列启发性的问题和各种各样的尝试和想法，让学生在观察、比较和推理中形成结论. 我们在本书中充分注意学生已有的基础和经验，注重采用多种方式自然地引入数学基本概念和基本方法. 例如，从齐次方程组的求解引入向量空间，从向量空间和微积分引入线性空间，从 Gram-Schmidt 方法引入 QR 分解，利用几何直观引入 SVD，从对角化问题引入正规变换及其矩阵，从向量的长度引入向量范数及矩阵范数，从高斯消元法引入 LU 分解，等等.

3. 适当增加矩阵的各类应用，淡化部分结论的理论证明

为了加强本课程对修读学生的吸引力，以利于学生快速进入实践环节，同时为了充分展示矩阵工具的强大，教材中还加入了一些具体应用. 当然，我们也希望通过它们，提供解决实际问题的理论框架和思想方法. 这些应用，既包括数学内部的应用，比如矩阵的谱半径、线性方程组的扰动分析、线性方程组迭代法的敛散性分析、微分方程数值解，也包括跨学科的各种应用，比如线性系统中的状态空间理论、运动分析及能控性和能观性、模式识别中的模式分类问题、主成分分析法（PCA）、图像压缩，等等.

考虑到教学中的实际情况及篇幅，编者适当略去了部分定理的证明，并对部分内容做了简化处理. 需要深入研究的读者，可参阅书后所附的参考文献.

4. 渗入数学史及数学文化,增强教材的趣味性和可读性

为了活跃课堂气氛,在课程讲授中,我们经常提及数学史及数学文化,但因为时间关系,师生都觉得不过瘾. 在撰写本书的过程中,与学生多有互动,他们也希望能在书中多讲点这些知识,以增加课程的趣味性. 我们的想法写进书以后,课堂上就不必多费口舌了. 当然,讲什么和讲多少是个问题.

首先,我们以较大的笔墨(2. 1. 3 和 2. 1. 4 共两小节),比较详细地阐述了向量空间的历史,以期让读者初步领略公理化思想所带来的革命性变化,从而在感叹"向量空间"这个最重要的观念来之不易的同时,能够更深刻地领悟到矩阵计算对公理化思想的反拨,进而建立起更加理性的数学态度.

其次,基于我们对矩阵分析与计算发展历史的理解,对于与重要思想有关的重要人物,我们也特辟专节加以详细阐述. 比如提出 Jordan 标准型的艾米尔 · 约当、提出 Householder 变换的豪斯霍尔德、提出 Hermite 变换的埃尔米特、发现 SVD 的 G. H. 戈卢布、提出雅可比迭代和雅可比矩阵的雅可比、提出 Rayleigh 商的瑞利勋爵,以及提出 Lanczos 过程的兰乔斯,等等. 当然,限于篇幅,我们只能"故意遗漏"一些重要思想和人物,比如向后扰动分析法的创始人和代数特征值问题上的大师威尔金森、制定浮点数标准的卡亨,等等. 至于与相对次要的思想有关的人物,我们也尽量给予一定笔墨,比如发现 LU 分解的图灵、发现施密特正交化的施密特、提出 Schur 引理的舒尔、提出 Galerkin 方法的伽辽金、提出 Krylov 子空间的克雷洛夫,等等.

再次,我们还刻意凸显了课程中的"中国元素". 例如高斯消元法与《九章算术》、嵌套乘法与秦九韶算法、范数与樊畿先生,等等. 这在西方人占据主导的矩阵知识中,是弥足珍贵的.

感谢何志庆教授在出国前夕的百忙之中审阅了大部分书稿. 感谢鲁习文教授、刘朝晖教授、李建奎教授和张先梅教授等院系领导对本教材给与的大力支持,还要感谢周国标教授给予笔者的点金之语,以及刘剑平教授对本教材的热忱关注. 另外,要特别感谢华东理工大学教务处相关领导及校教材建设委员会的诸位专家,蒙他们垂青,本书得以忝列 2012 年度教材立项项目.

我们深知,不仅矩阵所涉方方面面皆博大精深,而且对知识和思想真正的吸纳与融会也是任重而道远. 在戴维 · 洛奇的《小世界》中,年轻的柏斯苦苦追寻着心中的"莎士比亚". 对于矩阵这个"心中的圣殿",不年轻的我们也心有戚戚焉. 夫学术者,天下之公器也. 我们衷心期望的是,当我们怯怯地"放下第二步"的时候,听到的不是"第一步空寥的回声"(何其芳《预言》). 书中不当乃至谬误之处,诚盼各位方家高手来函批评指正. 来信请邮至:jgli@ecust. edu. cn

编　者

于华东理工大学数学系

目　录

第1章 线性方程组

微积分中对非线性问题采用线性化处理之后，就产生了大量线性方程组求解问题. 在线性代数中，借助于“秩”这个核心概念，线性方程组解的三种状态得到完美的阐述，因此线性代数作为研究课题一度被认为已“寿终正寝”. 然而计算机的横空出世，使得线性代数“枯木逢春”，催生出发展最快、思想也空前活跃的数值线性代数(又称矩阵计算). 如今，几乎所有的数值计算软件都具有求解线性方程组的功能. 事实上，它们大都以线性代数软件包 LAPACK (Linear Algebra PACKage) 为基础，其中涉及的基本代数子程序 BLAS(Basic Linear Algebra Subprograms)在高性能计算领域更是被广泛应用.

1.1 线性方程组的解法回顾

1.1.1 从高斯消元法谈起

让我们先来求解一个线性方程组.

例 1.1.1 解线性方程组

$$(\mathrm{I}):\begin{cases} x_1+2x_2-x_3=0 & ① \\ 3x_1+x_2=-1 & ② \\ -x_1-x_2-2x_3=1 & ③ \end{cases} \tag{1.1.1}$$

众所周知，我们可以采用被高斯(C. F. Gauss，1777－1855)重新发现的消元策略来求解这个线性方程组，这样就得到了如下的**初等行变换法**(elementary row transformation of matrix，也称**高斯消元法**).

解法一：初等行变换法.

首先是**消元过程**(elimination process).

$$(\mathrm{I}) \xrightarrow[1\times①+③]{(-3)\times①+②} \begin{cases} x_1+2x_2-x_3=0 & ① \\ -5x_2+3x_3=-1 & ② \\ x_2-3x_3=1 & ③ \end{cases} \xrightarrow{\frac{1}{5}\times②+③} \begin{cases} x_1+2x_2-x_3=0 & ① \\ -5x_2+3x_3=-1 & ② \\ -\frac{12}{5}x_3=\frac{4}{5} & ③ \end{cases}$$

$$\xrightarrow{(-\frac{5}{12})\times③}(\text{II}):\begin{cases}x_1+2x_2-x_3=0 & ①\\ -5x_2+3x_3=-1 & ②\\ x_3=-\dfrac{1}{3} & ③\end{cases}$$

接下来是**回代过程**(back substitution process).

$$(\text{II})\xrightarrow[1\times③+①]{(-3)\times③+②}\begin{cases}x_1+2x_2=-\dfrac{1}{3} & ①\\ -5x_2=0 & ②\\ x_3=-\dfrac{1}{3} & ③\end{cases}\xrightarrow{(-\frac{1}{5})\times②}\begin{cases}x_1+2x_2=-\dfrac{1}{3} & ①\\ x_2=0 & ②\\ x_3=-\dfrac{1}{3} & ③\end{cases}$$

$$\xrightarrow{(-2)\times②+①}(\text{III}):\begin{cases}x_1=-\dfrac{1}{3} & ①\\ x_2=0 & ②\\ x_3=-\dfrac{1}{3} & ③\end{cases}$$

上述过程用矩阵形式可简洁地表示如下：

$$\bar{\boldsymbol{A}}=\left(\begin{array}{ccc:c}1&2&-1&0\\3&1&0&-1\\-1&-1&-2&1\end{array}\right)\underset{r_{13}(1)}{\overset{r_{12}(-3)}{\sim}}\left(\begin{array}{ccc:c}1&2&-1&0\\0&-5&3&-1\\0&1&-3&1\end{array}\right)$$

$$\overset{r_{23}(1/5)}{\sim}\left(\begin{array}{ccc:c}1&2&-1&0\\0&-5&3&-1\\0&0&-12/5&4/5\end{array}\right)\overset{r_3(-5/12)}{\sim}\left(\begin{array}{ccc:c}1&2&-1&0\\0&-5&3&-1\\0&0&1&-1/3\end{array}\right)$$

$$\underset{r_{31}(1)}{\overset{r_{32}(-3)}{\sim}}\left(\begin{array}{ccc:c}1&2&0&-1/3\\0&-5&0&0\\0&0&1&-1/3\end{array}\right)\overset{r_2(-1/5)}{\sim}\left(\begin{array}{ccc:c}1&2&0&-1/3\\0&1&0&0\\0&0&1&-1/3\end{array}\right)\overset{r_{21}(-2)}{\sim}\left(\begin{array}{ccc:c}1&0&0&-1/3\\0&1&0&0\\0&0&1&-1/3\end{array}\right)$$

所以 $\boldsymbol{x}=(x_1,x_2,x_3)^{\mathrm{T}}=\left(-\dfrac{1}{3},0,-\dfrac{1}{3}\right)^{\mathrm{T}}$.

利用矩阵的乘法及分块矩阵的知识,可以证明以下定理：

定理 1.1.1　(初等变换与矩阵乘法)对 $m\times n$ 阶矩阵 $\boldsymbol{A}$ 施行一次初等行变换(一次初等列变换),相当于在 $\boldsymbol{A}$ 的左边(右边)乘以相应的 m 阶行初等矩阵 $\boldsymbol{R}$ (n 阶列初等矩阵 $\boldsymbol{C}$),即

$$\boldsymbol{A}\xrightarrow{\text{一次行初等变换}}\boldsymbol{B}\Leftrightarrow\boldsymbol{B}=\boldsymbol{RA},\ \boldsymbol{A}\xrightarrow{\text{一次列初等变换}}\boldsymbol{B}\Leftrightarrow\boldsymbol{B}=\boldsymbol{AC}\qquad(1.1.2)$$

由于可逆矩阵可以分解成有限个初等矩阵的乘积,因此定理 1.1.1 的结论可推广到可逆矩阵.

定理 1.1.2　(初等变换与可逆矩阵)对 $m\times n$ 阶矩阵 $\boldsymbol{A}$ 施行一系列初等行变换 $r_1,r_2,\cdots,r_s$ (初等列变换 $c_1,c_2,\cdots,c_t$),相当于在 $\boldsymbol{A}$ 的左边(右边)乘以相应的 m 阶可逆矩阵 $\boldsymbol{R}=\boldsymbol{R}_s\cdots\boldsymbol{R}_2\boldsymbol{R}_1$ (n 阶可逆矩阵 $\boldsymbol{C}=\boldsymbol{C}_1\boldsymbol{C}_2\cdots\boldsymbol{C}_t$),即

$$\boldsymbol{A}\xrightarrow{r_1,r_2,\cdots,r_s}\boldsymbol{B}\Leftrightarrow\boldsymbol{B}=\boldsymbol{RA},\quad\boldsymbol{A}\xrightarrow{c_1,c_2,\cdots,c_t}\boldsymbol{B}\Leftrightarrow\boldsymbol{B}=\boldsymbol{AC}\qquad(1.1.3)$$

其中,行初等矩阵 $\boldsymbol{R}_i$ 对应 r_i ,列初等矩阵 $\boldsymbol{C}_j$ 对应 c_j ($i=1,2,\cdots,s;j=1,2,\cdots,t$).

按定理 1.1.2，如果 $m\times n$ 阶矩阵 $\boldsymbol{A}$ 经过一系列初等变换化为 $m\times n$ 阶矩阵 $\boldsymbol{B}$，则意味着存在 m 阶可逆矩阵 $\boldsymbol{R}$ 及 n 阶可逆矩阵 $\boldsymbol{C}$，使得

$$\boldsymbol{RAC}=\boldsymbol{B} \tag{1.1.4}$$

此时，我们称矩阵 $\boldsymbol{A}$ 与 $\boldsymbol{B}$ **等价**(equivalence)，记为 $\boldsymbol{A}\sim\boldsymbol{B}$.

当系数矩阵 $\boldsymbol{A}$ 可逆时，线性方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 的解为 $\boldsymbol{x}=\boldsymbol{A}^{-1}\boldsymbol{b}$. 若能求出 $\boldsymbol{A}^{-1}$，即可求得线性方程组的解. 在式 (1.1.3) 中，设 $\boldsymbol{RA}=\boldsymbol{I}$，也就是 $\boldsymbol{R}=\boldsymbol{A}^{-1}$，此时，式 (1.1.3) 显然可理解成 $\boldsymbol{A}=\boldsymbol{A}^1$ 被一系列初等行变换 $r_1,r_2,\cdots,r_s$ 变换成了 $\boldsymbol{RA}=\boldsymbol{I}=\boldsymbol{A}^0$，即矩阵 $\boldsymbol{A}$ 的幂由 1 降为 0，因此，若想得到 $\boldsymbol{A}^{-1}$，显然只需将同样的这一系列初等行变换 $r_1,r_2,\cdots,r_s$ 作用到 $\boldsymbol{I}=\boldsymbol{A}^0$ 上. 这样就产生了例 1.1.1 的第二种解法.

解法二：矩阵求逆法.

先求系数矩阵 $\boldsymbol{A}$ 的逆矩阵.

$$(\boldsymbol{A}\,\vdots\,\boldsymbol{I})=\left(\begin{array}{ccc:ccc}1&2&-1&1&0&0\\3&1&0&0&1&0\\-1&-1&-2&0&0&1\end{array}\right)\sim\left(\begin{array}{ccc:ccc}1&2&-1&1&0&0\\0&-5&3&-3&1&0\\0&1&-3&1&0&1\end{array}\right)$$

$$\sim\left(\begin{array}{ccc:ccc}1&2&-1&1&0&0\\0&-5&3&-3&1&0\\0&0&-12/5&2/5&1/5&1\end{array}\right)\sim\left(\begin{array}{ccc:ccc}1&2&-1&1&0&0\\0&-5&3&-3&1&0\\0&0&1&-1/6&-1/12&-5/12\end{array}\right)$$

$$\sim\left(\begin{array}{ccc:ccc}1&2&0&5/6&-1/12&-5/12\\0&-5&0&-5/2&5/4&5/4\\0&0&1&-1/6&-1/12&-5/12\end{array}\right)\sim\left(\begin{array}{ccc:ccc}1&2&0&5/6&-1/12&-5/12\\0&1&0&1/2&-1/4&-1/4\\0&0&1&-1/6&-1/12&-5/12\end{array}\right)$$

$$\sim\left(\begin{array}{ccc:ccc}1&0&0&-1/6&5/12&1/12\\0&1&0&1/2&-1/4&-1/4\\0&0&1&-1/6&-1/12&-5/12\end{array}\right),$$

所以

$$\boldsymbol{A}^{-1}=\begin{pmatrix}-1/6&5/12&1/12\\1/2&-1/4&-1/4\\-1/6&-1/12&-5/12\end{pmatrix}=\frac{1}{12}\begin{pmatrix}-2&5&1\\6&-3&-3\\-2&-1&-5\end{pmatrix}.$$

注意到线性方程组 (I) 等价于 $\boldsymbol{Ax}=\boldsymbol{b}$，这里 $\boldsymbol{b}=(0,-1,1)^{\mathrm{T}}$，因此 $\boldsymbol{x}=\boldsymbol{A}^{-1}\boldsymbol{b}=\left(-\dfrac{1}{3},0,-\dfrac{1}{3}\right)^{\mathrm{T}}$.

在线性代数中，我们还学习了求解线性方程组的伴随矩阵法和克莱姆法则. 克莱姆法则来源自克莱姆(Gabriel Cramer，1704－1752)于 1750 年出版的著作，可事实上，它最早由麦克劳林(Colin Maclaurin，1698－1746，如今以麦克劳林级数闻名于后世)创立于

1729 年,并写入其 1748 年出版的遗作《代数学》之中. 数学史上充满了这样的"误称定律"(the law of misonomy),即当代科学史专家斯蒂芬·施蒂格勒(Stephen Stigler)提出的一个富有戏谑性的说法:"Nothing in mathematics is ever named after the person who discovered it."

解法三:伴随矩阵法.

由于 $|\boldsymbol{A}|=12$,且 $A_{11}=-2, A_{12}=6, A_{13}=-2, A_{21}=5, A_{22}=-3, A_{23}=-1, A_{31}=1, A_{32}=-3, A_{33}=-5$.

所以 $\boldsymbol{A}^{-1}=\dfrac{1}{|\boldsymbol{A}|}\boldsymbol{A}^{*}=\dfrac{1}{12}\begin{pmatrix}-2 & 5 & 1\\ 6 & -3 & -3\\ -2 & -1 & -5\end{pmatrix}$,其中 $\boldsymbol{A}^{*}$ 表示 $\boldsymbol{A}$ 的伴随矩阵. 从而

$$\boldsymbol{x}=\boldsymbol{A}^{-1}\boldsymbol{b}=\left(-\frac{1}{3},0,-\frac{1}{3}\right)^{\mathrm{T}}$$

解法四:克莱姆法则.

由于 $D=|\boldsymbol{A}|=12\neq 0$,且

$$D_1=\begin{vmatrix}0 & 2 & -1\\ -1 & 1 & 0\\ 1 & -1 & -2\end{vmatrix}=-4,\ D_2=\begin{vmatrix}1 & 0 & -1\\ 3 & -1 & 0\\ -1 & 1 & -2\end{vmatrix}=0,\ D_3=\begin{vmatrix}1 & 2 & 0\\ 3 & 1 & -1\\ -1 & -1 & 1\end{vmatrix}=-4$$

因此 $\boldsymbol{x}=\left(\dfrac{D_1}{D},\dfrac{D_2}{D},\dfrac{D_3}{D}\right)^{\mathrm{T}}=\left(-\dfrac{1}{3},0,-\dfrac{1}{3}\right)^{\mathrm{T}}$.

思考:在如此诱人的"四朵金花"中,哪一朵才是最娇艳欲滴的呢?

这个问题读者可能一时难以回答. 不过只要查阅一下 Matlab 软件的内置函数,我们就能给出初步解答. 事实上,Matlab 提供了内置函数 rref ,可将矩阵化成最简型,同时,Matlab 也提供了内置函数 inv,可用于求矩阵的逆. 当然也可以借助幂运算符(本质上也是内置函数)"^"或左除运算符(本质上也是内置函数)"\"来求解线性方程组,具体代码如下:

```
A= [1 2 - 1;3 1 0;- 1 - 1 - 2];b= [0 - 1 1]'; Ab= [A b];
%方法一:初等行变换法
[U,ip]= rref(Ab); U(:,ip)= [ ]; x= U;
%方法二:矩阵求逆法
x= inv(A)* b  % x= A^(- 1)* b
```

遗憾的是,Matlab 中没有为伴随矩阵法和克莱姆法则提供专门的内置函数(我们当然可以自己编程实现,需要的读者可来邮索取配套程序),这是否已暗示了哪一朵"金花"是最娇艳欲滴的了呢?

1.1.2　计算复杂性分析

关于"金花"问题,前面基于 Matlab 的分析显然只涉及问题的表象,我们需要深入探讨问题的本质,以给出更有说服力的解答. 特别要注意的是,我们要从各个角度尽可能地将问题一般化,这意味着在"金花"问题中,具体的系数矩阵要被一般化为任意的可逆矩阵 $\boldsymbol{A}$,矩阵的阶数要从 3 阶一般化为 n 阶,因此线性方程组 (I) 将被一般化为线性方程组

$Ax=b$，其中 A 是 n 阶可逆方阵.

判断一个算法的优劣，有很多标准，其中涉及算法效率的计算复杂性分析，主要指算法执行时需要消耗的计算机资源的"时"、"空"分析，包括运行时间的开销（**时间复杂度**）和存储的开销（**空间复杂度**）. 显然，时间复杂度越小，说明该算法效率越高，因此该算法越有价值；而空间复杂度越小，则说明算法越好. 我们当然希望算法"又快又省"，但鱼与熊掌往往不可兼得. 考虑到人生苦短，我们一般采取"空间换时间"的策略，即主要考虑降低时间复杂度，同时兼顾甚至牺牲空间复杂度.

由于计算机性能的差异，显然不能用时间的绝对量来"测度"算法的时间复杂度，更加合理的应该是采用算法的基本运算次数，也就是算法的所有运算次数的总和. 具体到矩阵算法问题，正如"维数灾难"所揭示的那样，矩阵的大小显然影响矩阵的运算量，以此观之，时间复杂度就是矩阵维数的函数. 对于 n 阶方阵而言，可记时间复杂度为 $T(n)$. 这样，我们就可以引入算法复杂度的阶数的概念.

定义 1.1.1　(算法复杂度的阶数) 如果存在正常数 C 和 N，使得当 $n>N$ 时，对于某矩阵算法的时间复杂度 $T(n)$，总成立函数 $f(n)$，使得 $T(n)\leqslant Cf(n)$，则称该算法的时间复杂度 $T(n)$ 是 $f(n)$ 阶的，并借鉴高等数学中表示同阶无穷小的大 O 表示法，记作 $T(n)=O(T(n))$.

对于一些常用的 $f(n)$，成立下列重要关系：

$$O(1)<O(\log n)<O(n)<O(n\log n)<O(n^2)<O(n^3)<$$
$$O(2^n)<O(3^n)<O(n!)<O(n^n) \tag{1.1.5}$$

值得一提的是，$O(n^3)$ 与 $O(2^n)$ 是一个重要的分水岭，前者说明时间复杂度 $T(n)$ 是 n 的多项式函数，当 n 增长时，$T(n)$ 的增长是可以忍受的，因此相应的算法被视为"好"的算法；后者则说明 $T(n)$ 是 n 的指数函数，当 n 增长时，$T(n)$ 的增长让人难以忍受，因此相应的算法被视为"坏"的算法. 如果某问题存在多项式算法，即"好"的算法，则该问题为多项式时间可解问题，简称**多项式**(polynomial)**问题**. 与之相关的 P/NP 问题(NP, non-deterministic polynomial)是计算复杂性理论(computational complexity theory)中至今尚未解决的问题.

下面我们从计算复杂性的视角来深入分析"金花"问题.

我们知道，拉普拉斯展开定理将一个 n 阶行列式的计算转化为 n 个 $n-1$ 阶行列式的计算. 若按此定理来计算行列式，并用 $T(n)$ 表示 n 阶行列式的时间复杂度，那么显然可有 $T(n)=nT(n-1)$，从而 $T(n)=nT(n-1)=n(n-1)T(n-2)=\cdots=n!T(1)$，因此这种算法的时间复杂度 $T(n)$ 是 $n!$ 级的，即 $T(n)=O(n!)$，这显然是比指数增长 $O(2^n)$ 还要"坏"的算法. 所以，尽管拉普拉斯展开定理在理论上非常重要，但在计算上一般仅用于低阶或特殊的行列式.

克莱姆法则要计算 $n+1$ 个 n 阶行列式，按拉普拉斯展开定理折算，这相当于计算一个 $n+1$ 行列式，即时间复杂度 $T(n)$ 是 $(n+1)!$ 级的；伴随矩阵法要计算 1 个 n 阶行列式以及 n^2 个 $n-1$ 阶的代数余子式，后者相当于 n 个 n 阶行列式，因此总体计算量相当于一个 $n+1$ 行列式，时间复杂度 $T(n)$ 也是 $(n+1)!$ 级的，与克莱姆法则持平. 这两朵"金花"的计算复杂度都是 $(n+1)!$ 级的，根据前面的分析，当维数达到一定规模后，这显然是