

数值分析

Numerical Analysis

潘建瑜

华东师范大学数学科学学院

数值分析是什么

Numerical analysis: the study of **quantitative approximations** to the solutions of mathematical problems including consideration of and bounds to the errors involved.

—— The dictionary by Merriam-Webster (韦氏英语词典), 2021

Numerical analysis the study of methods of **approximation** and their accuracy, etc.

—— The Chambers Dictionary (钱伯斯英语词典), 2021

Numerical analysis is the study of **algorithms** for solving the problems of continuous mathematics, by which we mean problems involving real or complex variables.

—— Lloyd N. Trefethen, *Princeton Companion to Mathematics*, 2008

- ▶ 1947 年 Von Neumann 和 Goldstine 在《美国数学会通报》发表了题为“高阶矩阵的数值求逆”的著名论文, 开启了现代计算数学的研究.
- ▶ 一般来说, 计算数学主要研究如何 (利用计算机) 计算数学问题的近似解 (数值解), 包括算法的设计、分析与计算机实现.

- ▶ 1947 年 Von Neumann 和 Goldstine 在《美国数学会通报》发表了题为“高阶矩阵的数值求逆”的著名论文, 开启了现代计算数学的研究.
- ▶ 一般来说, 计算数学主要研究如何 (利用计算机) 计算数学问题的近似解 (数值解), 包括算法的设计、分析与计算机实现.

计算数学主要研究内容

数值代数 (线性、非线性), 数值逼近, 数值微积分, 微分方程数值解 (常微分方程、偏微分方程), 数值优化, 反问题计算, 等等

国家自然科学基金委员会关于计算数学的分类 (2019)

➤ 计算数学与科学工程计算 (A0117)

- 偏微分方程数值解 (A011701)
- 流体力学中的数值计算 (A011702)
- 一般反问题的计算方法 (A011703)
- 常微分方程数值计算 (A011704)
- 数值代数 (A011705)
- 数值逼近与计算几何 (A011706)
- 谱方法及高精度数值方法 (A011707)
- 有限元和边界元方法 (A011708)
- 多重网格技术与区域分解 (A011709)
- 自适应方法 (A011720)
- 并行计算 (A011711)

➤ 运筹学 (数值最优化, 非线性方程, ...)

国家自然科学基金委员会关于计算数学的分类 (2021)

➤ A05 计算数学

- A0501 算法基础理论与构造方法
- A0502 数值代数
- A0503 数值逼近与计算几何
- A0504 微分方程数值计算
- A0505 反问题建模与计算
- A0506 复杂问题的可计算建模与数值模拟
- A0507 新型计算方法

➤ A04 统计与运筹: 连续优化, 离散优化, 随机优化与统计优化, ...

➤ A06 数学与其他学科的交叉: 符号计算, 人工智能, 数据科学

- A0813 计算固体力学
- A0910 计算流体力学

➤ F02 计算机科学: 信息安全, 数据科学与大数据计算

➤ F06 人工智能

为什么计算数学

计算科学是 21 世纪确保国家核心竞争能力的战略技术之一.

—— 计算科学: 确保美国竞争力, 2005 年总统信息技术咨询委员会报告

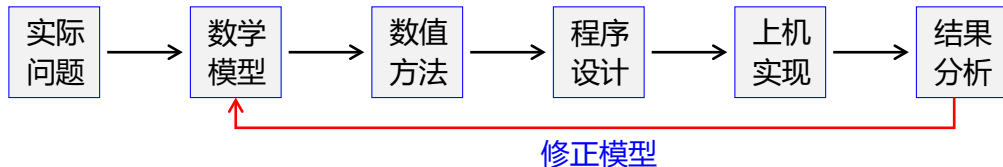
科学计算是 20 世纪重要科学技术进步之一, 已与理论研究 and 实验研究相并列成为科学研究的第三种方法. 现今科学计算已是体现国家科学技术核心竞争力的重要标志, 是国家科学技术创新发展的关键要素

—— 国家自然科学基金·重大项目指南, 2014

科学计算的核心/数学基础: 计算数学.

数值分析/计算方法/数值计算方法

 借助计算机的高速计算能力, 解决现代科学、工程、经济、人文等领域中的各类复杂 (数学) 问题, 是数学与计算机的有机结合.



 数值分析的任务主要包括其中的数值方法、程序设计和上机实现

应用举例: 线性方程组

一个古老的数学问题

今有

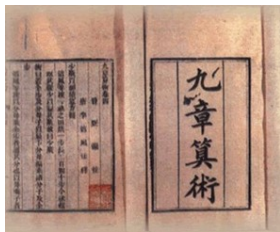
上禾三秉, 中禾二秉, 下禾一秉, 实三十九斗;

上禾二秉, 中禾三秉, 下禾一秉, 实三十四斗;

上禾一秉, 中禾二秉, 下禾三秉, 实二十六斗.

问上、中、下禾实一秉各几何?

— 方程术 (消元思想) 《九章算术》(公元初)



$$3x + 2y + z = 39$$

$$2x + 3y + z = 34$$

$$x + 2y + 3z = 26$$



$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

$$Ax = b$$

应用举例: 非线性方程

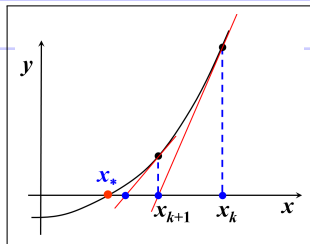
平方根

➤ 怎样计算平方根? 如 $\sqrt{2}$, $\sqrt{21}$, $\sqrt{201}$, $\sqrt{n}$

➤ 等价于非线性方程: $f(x) = x^2 - 2 = 0$

➤ Newton 迭代法: $x_{k+1} = \frac{1}{2} \left(x_k + \frac{2}{x_k} \right)$, $k = 0, 1, 2, \dots$

➤ 得到迭代序列: $x_0 = 1.0, x_1 = 1.5, x_2 = 1.416667, x_3 = 1.414216, x_4 = 1.414214, \dots$



🔍 思考: 怎么计算 n 次多项式的零点?

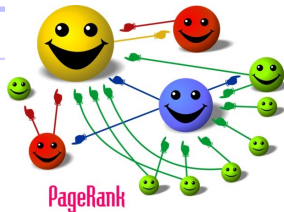
应用举例: 矩阵特征值与特征向量

Google PageRank

➤ Google, 1998 年创立, 现今市值约 1.76 万亿美元 (2024.02)

$$Gx = x, \quad e^T x = 1$$

- G : Google Matrix, “the world’s largest matrix computation”
- x : PageRank vector, “The \$25,000,000,000 Eigenvector” (SIAM Review, 2006)



SIAM REVIEW
Vol. 48, No. 3, pp. 569-581

© 2006 Society for Industrial and Applied Mathematics

The \$25,000,000,000 Eigenvector: The Linear Algebra behind Google*

Kurt Bryan[†]
Tanya Leise[‡]

Abstract. Google's success derives in large part from its PageRank algorithm, which ranks the importance of web pages according to an eigenvector of a weighted link matrix. Analysis of the PageRank formula provides a wonderful applied topic for a linear algebra course. Instructors may assign this article as a project to more advanced students or spend one or two lectures presenting the material with assigned homework from the exercises. This material also complements the discussion of Markov chains in matrix algebra. Maple and Mathematica files supporting this material can be found at www.rose-hulman.edu/~bryan.

计算数学的主要任务

设计求解各种实际问题的高效可靠的数值方法

- **算法设计**: 构造求解各种数学问题的数值方法
- **算法分析**: 收敛性、稳定性、复杂性、计算精度等
- **算法实现**: 编程实现、软件开发
- **数学模型分析**: 数学问题的结构特点, 数学模型的评估 (合理性、准确性)

计算数学的主要任务

设计求解各种实际问题的高效可靠的数值方法

- **算法设计**: 构造求解各种数学问题的数值方法
- **算法分析**: 收敛性、稳定性、复杂性、计算精度等
- **算法实现**: 编程实现、软件开发
- **数学模型分析**: 数学问题的结构特点, 数学模型的评估 (合理性、准确性)

对于同一个问题, 不同的算法在计算性能上可能相差百万倍或者更多!

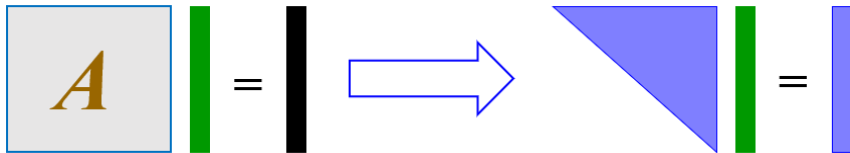
应用举例：线性方程组求解

线性方程组求解

求解一个 n 阶线性方程组, 若使用**克莱姆法则**, 需要计算 $n + 1$ 个 n 阶行列式, 大约需要 $n!(n^2 - 1)$ 次乘除运算. 而使用**高斯消去法**, 只需约 $2n^3/3$ 次加减乘除运算.

➤ 当 $n = 20$ 时, $n!(n^2 - 1) \approx 9.7^{20}$, $2n^3/3 \approx 5.3 \times 10^3$

如果用每秒运算 30 亿次 (主频 3.0G) 的计算机求解, 克莱姆法则大约需要 10000 年!
但高斯消去法一秒不到就能解决!



应用举例：矩阵乘积 (Deepmind 团队)



nature

Explore content ▾

About the journal ▾

Publish with us ▾

[nature](#) > [articles](#) > article

Article | [Open access](#) | [Published: 05 October 2022](#)

Discovering faster matrix multiplication algorithms with reinforcement learning

[Alhussein Fawzi](#) , [Matej Balog](#), [Aja Huang](#), [Thomas Hubert](#), [Bernardino Romera-Paredes](#), [Mohammadamin Barekatain](#), [Alexander Novikov](#), [Francisco J. R. Ruiz](#), [Julian Schrittwieser](#), [Grzegorz Swirszcz](#), [David Silver](#), [Demis Hassabis](#) & [Pushmeet Kohli](#)

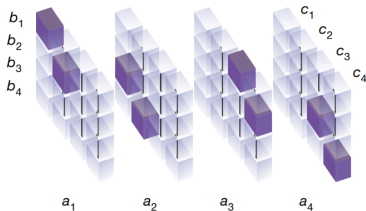
[Nature](#) **610**, 47–53 (2022) | [Cite this article](#)

563k Accesses | **131** Citations | **3658** Altmetric | [Metrics](#)

应用举例：矩阵乘积

a

$$\begin{pmatrix} c_1 & c_2 \\ c_3 & c_4 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \cdot \begin{pmatrix} b_1 & b_2 \\ b_3 & b_4 \end{pmatrix}$$



b

$$m_1 = (a_1 + a_4)(b_1 + b_4)$$

$$m_2 = (a_3 + a_4)b_1$$

$$m_3 = a_1(b_2 - b_4)$$

$$m_4 = a_4(b_3 - b_1)$$

$$m_5 = (a_1 + a_2)b_4$$

$$m_6 = (a_3 - a_1)(b_1 + b_2)$$

$$m_7 = (a_2 - a_4)(b_3 + b_4)$$

$$c_1 = m_1 + m_4 - m_5 + m_7$$

$$c_2 = m_3 + m_5$$

$$c_3 = m_2 + m_4$$

$$c_4 = m_1 - m_2 + m_3 + m_6$$

c

$$\mathbf{U} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{V} = \begin{pmatrix} 1 & 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

$$\mathbf{W} = \begin{pmatrix} 1 & 0 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

事实上, 除了计算复杂度外, 数据在内存/缓存中的移动/访问方式也会影响计算效率.

(GEMM_Performance.m)

数值方法基本概念

- 解析解、精确解、真解、真值, 数值解、近似解
- 数值方法的特点
 - 求的是近似解, 即求出的解是有误差的
 - 与计算机紧密结合, 易于上机实现
- 算法的评价
 - 时间复杂度 (计算机运行所需的时间)
 - 空间复杂度 (所占用的计算机存储空间)
 - 逻辑复杂度 (影响程序开发的周期以及后续维护的难易程度)

好的数值方法

- 有可靠的理论分析, 即收敛性、稳定性等有数学理论保证
- 有良好的计算复杂性 (时间和空间)
- 易于在计算机上实现
- 要有具体的数值试验来证明是行之有效的

二十世纪十大优秀算法 (SIAM News, 2000)

1. Monte Carlo method (1946)
2. Simplex Method for Linear Programming (1947)
3. Krylov Subspace Iteration Methods (1950)
4. The Decompositional Approach to Matrix Computations (1951)
5. The Fortran Optimizing Compiler (1957)
6. QR Algorithm for Computing Eigenvalues (1959-61)
7. Quicksort Algorithm for Sorting (1962)
8. Fast Fourier Transform (1965)
9. Integer Relation Detection Algorithm (1977)
10. Fast Multipole Method (1987)

 The Best of the 20th Century: Editors Name Top 10 Algorithms, B. A. Cipra, SIAM News, 2000.

课程主要内容

➤ 数值代数

- 线性方程组的求解: 直接法和迭代法
- 线性最小二乘问题求解
- 非线性方程的求解
- 特征值计算: 非对称情形和对称情形, 奇异值分解 *

➤ 数值逼近

- 函数插值
- 函数逼近
- 数值积分与微分

➤ 常微分方程数值解 *

预备知识和学习建议

预备知识

- 数学分析
- 高等代数（线性代数）
- 常微分方程
- 计算机编程（推荐 MATLAB）

预备知识和学习建议

预备知识

- 数学分析
- 高等代数（线性代数）
- 常微分方程
- 计算机编程（推荐 MATLAB）

学习建议

- 掌握数值方法的基本思想和原理 (不能仅仅靠记忆);
- 掌握数值方法的设计和分析的一些常用技术和技巧;
- 要重视误差分析、收敛性和稳定性分析等基本理论;
- 适量的数值计算训练 (包括编程实践和手工推导, 有助于加深对算法的理解).

成绩评定

 平时成绩（小测 + 作业 + 上机 + 课堂表现）50%，期末成绩（笔试）50%

成绩评定

 平时成绩（小测 + 作业 + 上机 + 课堂表现）50%，期末成绩（笔试）50%

主要参考资料

-  李庆扬等, **数值分析**, 第五版, 清华大学出版社, 2008
-  Sauer, **Numerical Analysis**, 3rd Edition, 2018
-  Burden et.al, **Numerical Analysis**, 10th Edition, 2016
-  Heath,, **Scientific Computing: An Introductory Survey**, Revised 2nd Edition, 2018
-  Stoer and Bulirsch, **Introduction to Numerical Analysis** , 3rd Edition, 2002
-  本课程需要编程, 请自学编程语言, 如北太天元, MATLAB, Python 等
-  课程主页: <http://math.ecnu.edu.cn/~jypan/Teaching/NA>

谢谢
THANK YOU

