

## BOOK REVIEWS

JOHAN HOFFMAN, *Methods in Computational Science*, SIAM, Philadelphia, 2021, xvi + 396 pp., ISBN 978-1-61197-671-7 (paperback), ISBN 978-1-61197-672-4 (ebook). Part of the Computational Science and Engineering series.

*Methods in Computational Science* has been developed based on lecture notes used in a course taught at the KTH Royal Institute of Technology in Stockholm. The material has been refined over the years, ensuring its clarity and pedagogical effectiveness. Throughout the book, readers will find numerous exercises that deepen understanding and build practical skills. Each chapter concludes with an outlook section, offering readers avenues for further exploration and referencing additional resources. The inclusion of pseudocode for algorithms ensures clarity and flexibility, allowing readers to implement the methods in their preferred programming language. Supplementary materials, including Python implementations of key algorithms, are available online, enhancing the learning experience.

The book begins by establishing the fundamental principles in mathematics and computer science, setting the stage for the subsequent chapters. We provide a brief description of the content covered in each of the eight parts of the book. Part 1 covers the core concepts of mathematical principles. Chapter 1 introduces vector spaces, their algebraic and geometric properties, and their applications in areas like linear algebra and optimization. Then in Chapter 2 it is explored the linear transformations, represented by matrices, and their properties, such as invertibility, diagonalization, and the fundamental theorem of linear algebra. These mathematical concepts provide a solid foundation for understanding computational science and its applications.

Part 2 covers the fundamental concepts for computer science. Chapter 3 introduces algorithms, discussing their formulation and implementation as computer programs. It explores algorithm efficiency, complexity analysis, and essential data structures like arrays, linked lists, graphs, grids, and networks. Additionally, it touches on image representation and processing techniques in computer graphics. Chapter 4 focuses on high-performance computing, used for data processing and simulations. It examines the architecture of supercomputing systems with millions of processors and complex memory structures. The chapter highlights designing efficient algorithms, parallelism, and hardware accelerators like GPUs and quantum processors.

Part 3 explores matrix factorization techniques. Chapter 5 focuses on solving systems of linear equations using matrix factorization techniques. It covers efficient algorithms based on diagonal, orthogonal, and triangular matrices. The chapter also showcases applications in differential and integral equations, such as heat conduction and air pollution dispersion. Chapter 6 discusses eigenvalue and singular value decompositions. It explains how diagonalizable matrices reveal stretch transformations and eigenvalues, while non-diagonalizable matrices involve shear transformations. The chapter highlights stability analysis, iterative algorithms, and the significance of singular value decomposition in various disciplines, including continuum mechanics and data analysis.

Part 4 delves into iterative methods for solving linear and nonlinear equations. Chapter 7 focuses on solving large sparse systems of linear equations efficiently using iterative methods. These methods exploit matrix sparsity, reducing memory usage while gradually improving the approximation. Stationary methods and Krylov methods are introduced, offering different approaches for achieving convergence. Chapter 8 shifts to solving systems of nonlinear equations. Fixed point iteration is the fundamental method, with convergence dependent

on the continuity of the residual. The Newton method, relying on the Jacobian matrix, achieves quadratic convergence under certain conditions. Quasi-Newton methods provide approximations of the Jacobian to enhance computational efficiency.

In Part 5 Approximation methods are introduced. Chapter 9 covers various approximation problems, from interpolation to regression, using models like polynomials and neural networks. The chapter emphasizes the use of Hilbert spaces and orthogonal projection for constructing accurate approximations. Chapter 10 focuses on function approximation over multidimensional domains. It explores structured grids and advanced methods like implicit level set and explicit unstructured meshes to represent domain geometry. Piecewise polynomials, including splines, are discussed for their local support and smoothness properties.

In the following Part 6 Integration and Stochastic methods are discussed. Chapter 11 introduces quadrature methods for approximating integrals, while Chapter 12 focuses on stochastic methods for probabilistic analysis and statistical inference. These chapters provide valuable techniques for approximating integrals, handling uncertainty, and drawing conclusions from data.

Part 7 is about Differential Equations. Chapter 13 explores the solution methods for scalar initial value problems, while Chapter 14 focuses on systems of initial value problems. Both chapters discuss the historical development of calculus and its application to solving practical problems represented by differential equations. They cover various topics such as time stepping algorithms, stability properties, and efficient algorithms for solving initial value problems. These chapters provide insights into the wide range of applications of differential equations in fields like physics, biology, and engineering.

Part 8 is about Optimization and Learning from data. Chapter 15 focuses on optimization, addressing the minimization and maximization of objective functions. It covers unconstrained and constrained minimization, global minimization techniques, and the characteristics of linear objective functions. Chapter 16 explores learning from data, distinguishing between unsupervised and supervised learning. It discusses the construction of models for classification and approximation, particularly through neural networks. It also touches on parallelization strategies for machine learning.

In summary, *Methods in Computational Science* is a comprehensive textbook for graduate-level students in computer science and data science. It offers a thorough understanding of computational methods, from theoretical foundations to practical applications, across a wide range of disciplines. With its emphasis on integration, practical implementation, and real-world examples, this book is an invaluable resource for anyone looking to expand the knowledge and proficiency in computational science.

*Imre Boros*

GUOJUN CAN, CHAOQUN MA, JIANHONG WU, *Data Clustering: Theory, Algorithms, and Applications*, 2nd ed., SIAM, Philadelphia, 2021, xxiii + 406 pp., ISBN 978-1-61197-632-8 (paperback), ISBN 978-1-61197-633-5 (ebook). Part of the Mathematics in Industry series.

The first edition of this monograph has grown and evolved from some collaborative projects for industrial applications undertaken by the Laboratory for Industrial and Applied Mathematics at York University, some of which were in collaboration with Generation 5 Mathematical Technologies, Inc. The motivation behind the creation of the first edition (2007) of the book was that there have been many clustering algorithms scattered in publications in very diversified areas such as pattern recognition, artificial intelligence, information technology, image processing, biology, psychology, and marketing. So the authors gathered and categorized the most important clustering algorithms from their point of view, stating that they made no attempt to provide a comprehensive coverage of the subject area.

Since the first edition, the advancement of technology in all the data related subject areas exploded. The second edition was published in 2020 to update the first edition with the latest developments. Chapter 19 was replaced because MATLAB lost the race compared to other programming languages as is not efficient for developing clustering algorithms; the new chapter now presents some Open Source Clustering Software from Python, R, Java, C++. Since the topic of Chapter 20 in the first edition is discussed now in a whole book, Chapter 20 was replaced by a new chapter, about Lightweight Java Clustering Framework. Chapters 21 and 22 were added presenting two applications of clustering algorithms.

The book is divided into four parts: basic concepts (clustering, data, and similarity measures), algorithms, programming languages and applications. We briefly describe the content of each part of the book.

The first part of the book introduces all the concepts related to data clustering. The first chapter defines what data clustering is, describes the clustering process and list some resources related to clustering. In the second chapter the most important data types are discussed which are a major factor in choosing the appropriate clustering algorithm. The following two chapters deal with data such as the scale conversion and the standardization and transformation techniques. In Chapter 5 different data visualization techniques are presented like Sammon's Mapping, Tree Maps,  $t$ -SNE, etc. In the ending chapter of this part similarity and dissimilarity measures are introduced which are very important in data clustering because they are used to quantitatively describe the similarity/dissimilarity between two data points or two clusters.

The second part of the book presents some popular clustering algorithms based on some specific methodologies, such as hierarchical, center-based, search-based methods being the largest number of algorithms. It also includes chapters for fuzzy, graph-based, grid-based, density-based, model-based, subspace-based and scalable algorithms. There is also a chapter for some miscellaneous algorithms which did not fit into any other category and the last chapter of this part is discussing evaluation methods for clustering algorithms.

The third part consists of two chapters. The first one describes the most popular software related to clustering in the most widely used programming languages, such as R, Python, Java and C++. The second chapter of the third part presents a lightweight Java clustering framework which was written by the authors of this book.

The final, fourth part of the book presents two applications for clustering: one about gene expression and the other about variable annuity policies.

Anyone familiar with elementary linear algebra, calculus, and basic statistical concepts can easily read and understand the presented concepts. The monograph is intended not only for statistics, applied mathematics, and computer science senior undergraduates and graduates, but also for research scientists who need cluster analysis to deal with data. It can be a great resource for any data scientist or machine learning engineer who is dealing with unsupervised learning problems.

*Imre Boros*

JOSEF MÁLEK, ZDENĚK STRAKOŠ, *Preconditioning and the Conjugate Gradient Method in the Context of Solving PDEs*, SIAM, Philadelphia, 2015, IX + 104 pp., ISBN 978-1-61197-383-9 (paperback), ISBN 978-1-61197-384-6 (ebook). Part of the Spotlights series.

This is the first book that appeared in the SIAM Spotlights series, which aims to publish brief and enlightening books in applied and computational mathematics and scientific computing. The authors are renowned experts in the field of numerical analysis; they herein provide a masterful example of how to *write* clear and rigorous mathematics in short chapters, and how to *read* science in general – always looking for the primary sources with great care in tracing the historical contributions, bringing to the surface important ideas from

older works (some often cited, but rarely read); and at the same time covering in breadth the progress made in the most recent literature. The key take-away ideas in each chapter are emphasized inside boxes and written in a clear manner, while several deep and illuminating quotes are also given.

The book discusses as a prototypical example linear second order elliptic PDEs, aiming to present the interplay between functional analysis, discretization (Galerkin methods) and iterative methods for linear problems, arguing *against* the (common) workflow schematically described by problem  $\rightarrow$  model  $\rightarrow$  discretization  $\rightarrow$  computation.

After an introduction that summarizes the essential ideas of the book, boundary value problems for elliptic PDEs are discussed together with the appropriate function spaces and norms, and solutions given by minimizing the energy functional. Then standard linear functional analysis follows with the Riesz representation theorem and Lax-Milgram's lemma for bilinear forms that are bounded and elliptic (coercive). Writing the operator equation using the Riesz map and choosing a certain scalar product, one obtains a new problem: the so-called *operator preconditioning*.

This problem is approximated over Krylov subspaces defined through the Riesz map by the conjugated gradient method (CG) in Hilbert spaces. The CG algorithm can be obtained by minimizing the energy functional over Krylov subspaces. The method can be seen as *model reduction* through the Vorobyev method of moments. An essential idea is that, through the spectral decomposition written using the Riemann-Stieltjes integral, CG can be formulated as a Gauss-Christoffel quadrature for this integral. It follows that the rate of converge for CG is determined by the approximation of the spectral distribution function, while bounds using only the operator condition number cannot be generally explanatory. The authors emphasize that descriptions of Krylov subspace methods based only on condition numbers are, in general, not complete (apart from special cases and cases where the use of Krylov subspace methods could actually be questioned). The matrix formulation of CG is obtained by minimizing the quadratic energy functional over finite-dimensional Krylov subspaces; the discretized Hilbert space formulation of CG gives the algebraically preconditioned matrix formulation of CG. The next short chapters (mostly 3-4 pages) concern Galerkin discretizations, discretization errors and computational/algebraic errors, and the fundamental principles of consistency, stability, convergence. Preconditioning for finite element methods (FEM) can be seen as a way of obtaining a global transfer of information, given that sparse FEM matrices provide locality of information due to the bases with local support. Controlling the spatial distribution of the algebraic error and the (possibly different) spatial distribution of the discretization error are indicated as key ingredients for a posteriori error estimates.

In contrast to the common approach of separating discretization of PDEs from the construction of preconditioners, this book focuses on the coupling between the two. Subsequent works since the publication of the book have been analyzing, for example, the spectrum of preconditioned second order differential operators, estimating and localizing the algebraic and total numerical errors in finite element methods. To quote the motto at the end of the book, the authors strongly advocate for travelling *from PDEs through functional analysis to iterative methods, there and back again*.

Mihai Nechita