Journal of Statistical Software

November 2018, Volume 87, Book Review 2. *doi: 10.18637/jss.v087.b02*

Reviewer: Abdolvahab Khademi
University of Massachusetts

Computational Methods for Numerical Analysis with R

James P. Howard, II
Chapman & Hall/CRC, Boca Raton, 2017.
ISBN 9781498723633. xx+257 pp. USD 99.95 (H).
https://www.crcpress.com/9781498723633

Numerical approximation algorithms have traditionally been implemented in generic and specialized programming languages, such as C++, Fortran, and MATLAB. However, newer programming languages such as Python and R are becoming more popular among students and researchers. What makes the latter languages distinct from the traditional ones is their tuning towards data analysis (structurally and through libraries), their free cost and accessibility to everyone, and faster updates due to community-based development. These amenities are the main drivers behind the rise and adoption of such modern computational languages. *Computational Methods for Numerical Analysis with R* reflects this change and a future trend in the use of modern specialized programming languages, such as R.

This book is structured in seven chapters, essentially covering the topics in an undergraduate course in numerical analysis. In each chapter, the author presents the concepts clearly, provides R code for the different algorithms used for computations, presents insights, and ends the chapter with a good number of exercises for the reader. The exercises provide practice in both coding and conceptual understanding. The author's website provides all the R code in a package and an errata sheet.

Numerical analysis is defined and compared with symbolic computation in Chapter 1, *Introduction to Numerical Analysis*. After postulating the three goals desirable in numerical analysis (efficiency, accuracy, precision), the author focuses on the R language (data types, data structures, functions) and presents the reasons for selecting R for numerical analysis. Elementary algorithms for summation, polynomial evaluation, and root finding are introduced to the reader with several examples and comparison of the algorithms.

Because numerical analysis solutions are approximations, Chapter 2, *Error Analysis*, is dedicated to the discussion of closeness of solutions to true values in terms of accuracy and precision. The author then discusses types of variables in programming languages (including integers, floating-points, doubles) and how they are handled in R. Other topics include error propagation, round-off error and machine ϵ, and overflow/underflow. Treatment of errors is demonstrated through examples, code, diagrams, and applications such as integer division and binary long division.

Increased computation power and reformulation of mathematical problems in matrix form has increased the popularity of linear algebra among researchers, in particular among statisticians. In Chapter 3, *Linear Algebra*, the author introduces basic concepts of matrix algebra and matrix operations along with R code to implement them. Gaussian elimination, matrix decomposition (LU and Cholesky methods), iterative methods (Jacobi iteration, Gauss-Seidel iteration), and applications in least-squares estimation are presented.

Approximation of univariate and multivariate missing or nonpresent data is treated in Chapter 4, *Interpolation and Extrapolation.* These concepts are clarified and demonstrated through code using simple but effective problems, such as finding the coordinates of expected points on a line or on a curve using linear and polynomial interpolation. For functions that may be of much higher order, noncontinuous, and nondifferentiable, piecewise and (cubic) spline interpolation techniques are presented. Bézier curves receive a special treatment as alternative techniques with application in computer graphics. Bilinear interpolation and nearest neighbor interpolations are presented when data are multidimensional. Applications in time series data and computer graphics conclude the chapter. Extrapolation, though, is not comprehensively treated due to its similarity to interpolation. However, because this is an applied textbook, it would be much more beneficial to the reader if the author had presented more materials about extrapolation in this chapter.

Core concepts in calculus and how they are solved numerically are presented in Chapter 5, *Differentiation and Integration.* Differentiation is briefly explained and shown using different variants of finite difference approach. Numerical integration techniques are demonstrated through Newton-Cotes approach (using rectangles and trapezoids), Gaussian method, adaptive integration, Romberg's method, and Monte Carlo methods. Applications of numerical integration are demonstrated for the computation of solid volumes and the Gini coefficient.

Chapter 6, *Root Finding and Optimization*, introduces algorithms and R code for finding the root, the minimum, and the maximum of functions. The section on root finding covers the bisection method, the Newton-Raphson iterative method, and the secant method. Advantages and disadvantages of these methods are discussed and illustrated through examples and code. In the section on optimization techniques, the author covers the golden section search method, gradient descent, hill climbing, and simulated annealing. These methods are contrasted with each other to point out advantages and applications. The reader can find the application of these algorithms in least squares (using gradient descent), and in the well-known problem of the traveling salesman (using simulated annealing).

Ordinary (ODE) and partial (PDE) differential equations with approximation methods, examples, and applications are presented in Chapter 7, *Differential Equations.* This chapter begins with the challenge of finding initial values in solving an ODE and the algorithms used to solve the initial value problem. The author begins with the elementary Euler method, the approximation errors involved in this method, and the particular case of stiff ODE's to the more comprehensive Runge-Kutta method. Linear multistep methods where earlier function evaluations are reused for a next step evaluation (such as Adams-Bashforth method) are introduced next. Systems of ODE's, boundary value problems, PDE's, and the heat and wave equations as instances of PDE's are also covered in this chapter. The author concludes this chapter and the book by demonstrating applications of differential equations in carbon dating and Lotka-Volterra equations in population dynamics. R implementation code is provided for all algorithms discussed in this chapter.

Numerical analysis relies on three main components: algorithms, computing hardware, and computing language. Algorithms usually do not affect the choice of a programming language as they can be implemented in any language. Hardware today plays a role in the selection of a programming language only if they support a particular language in computing, as in the case of GPU's. The choice of a programming language has always been a matter of long-term investment by both researchers and institutions. The open source feature of the R language has encouraged vast adoption by researchers and research institutions (in addition to businesses) and wide and steady contributions from independent package developers. It cannot be understated what impact the existence of learning resources can have on the adoption and growth of a programming language. *Computational Methods for Numerical Analysis with R* can be considered one of the few resources on numerical analysis algorithms in R language currently on the market.

The book starts with simple numerical algorithms and mathematical operations that become more engaging than intimidating to the reader who has little exposure to numerical computing. The transition to more complicated topics remains smooth through clear description of the algorithms, comparisons, and examples. The overall clarity of the book makes it very accessible to self-learners. The book assumes that the reader has knowledge at the level of undergraduate mathematics and the ability to write code in R. Therefore, the explanations of mathematical concepts are brief and more exposition is dedicated to the algorithms and their implementation in R.

A prominent feature of the book is the inclusion of exercises at the end of each chapter. The exercises are enough in number and rich enough in scope to help the readers reinforce their understanding of the concepts and further extend them to more advanced and complicated problems. The exercises provide an opportunity for the readers to improve their algorithmic thinking needed for scientific computing. At the time of writing this review, there is no solution for the exercises.

In summary, *Computational Methods for Numerical Analysis with R* provides an excellent introduction to numerical analysis using R for undergraduate students and researchers who prefer to invest in or switch to R for their academic and research endeavors and also for those students and researchers who need access to a free and rich programming environment. This introductory text clearly shows the potential of R in applied mathematical fields such as optimization, scientific computing, and engineering.

Reviewer:

Abdolvahab Khademi
University of Massachusetts
Department of Mathematics and Statistics
Amherst MA 01002, United States of America
E-mail: khademi@math.umass.edu

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
November 2018, Volume 87, Book Review 2 *Published:* 2018-11-03
doi:10.18637/jss.v087.b02