WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Deutsch
Español
日本語
Русский
Suomi
Edit links

QR algorithm

From Wikipedia, the free encyclopedia

In numerical linear algebra, the **QR algorithm** is an eigenvalue algorithm: that is, a procedure to calculate the eigenvalues and eigenvectors of a matrix. The QR transformation was developed in the late 1950s by John G.F. Francis (England) and by Vera N. Kublanovskaya (USSR), working independently.[1] The basic idea is to perform a QR decomposition, writing the matrix as a product of an orthogonal matrix and an upper triangular matrix, multiply the factors in the reverse order, and iterate.

Contents [hide]

The practical QR algorithm [edit]

Formally, let *A* be a real matrix of which we want to compute the eigenvalues, and let A_0:=*A*. At the *k*-th step (starting with *k* = 0), we compute the QR decomposition $A_k=Q_kR_k$ where Q_k is an orthogonal matrix (i.e., $Q^T = Q^{-1}$) and R_k is an upper triangular matrix. We then form $A_{k+1} = R_kQ_k$. Note that

$$A_{k+1} = R_kQ_k = Q_k^{-1}Q_kR_kQ_k = Q_k^{-1}A_kQ_k = Q_k^TA_kQ_k,$$

so all the A_k are similar and hence they have the same eigenvalues. The algorithm is numerically stable because it proceeds by *orthogonal* similarity transforms.

Under certain conditions,[2] the matrices A_k converge to a triangular matrix, the Schur form of *A*. The eigenvalues of a triangular matrix are listed on the diagonal, and the eigenvalue problem is solved. In testing for convergence it is impractical to require exact zeros, but the Gershgorin circle theorem provides a bound on the error.

In this crude form the iterations are relatively expensive. This can be mitigated by first bringing the matrix *A* to upper Hessenberg form (which costs $\tfrac{10}{3}n^3 + O(n^2)$ arithmetic operations using a technique based on Householder reduction), with a finite sequence of orthogonal similarity transforms, somewhat like a two-sided QR decomposition.[3][4] (For QR decomposition, the Householder reflectors are multiplied only on the left, but for the Hessenberg case they are multiplied on both left and right.) Determining the QR decomposition of an upper Hessenberg matrix costs $6n^2 + O(n)$ arithmetic operations. Moreover, because the Hessenberg form is already nearly upper-triangular (it has just one nonzero entry below each diagonal), using it as a starting point reduces the number of steps required for convergence of the QR algorithm.

If the original matrix is symmetric, then the upper Hessenberg matrix is also symmetric and thus tridiagonal, and so are all the A_k. This procedure costs $\tfrac{4}{3}n^3 + O(n^2)$ arithmetic operations using a technique based on Householder reduction.[3][4] Determining the QR decomposition of a symmetric tridiagonal matrix costs $O(n)$ operations.[5]

The rate of convergence depends on the separation between eigenvalues, so a practical algorithm will use shifts, either explicit or implicit, to increase separation and accelerate convergence. A typical symmetric QR algorithm isolates each eigenvalue (then reduces the size of the matrix) with only one or two iterations, making it efficient as well as robust.[*clarification needed*]

The implicit QR algorithm [edit]

In modern computational practice,[6] the QR algorithm is performed in an implicit version which makes the use of multiple shifts easier to introduce. The matrix is first brought to upper Hessenberg form $A_0 = QAQ^T$ as in the explicit version; then, at each step, the first column of A_k is transformed via a small-size Householder similarity transformation to the first column of $p(A_k)$ (or $p(A_k)e_1$), where $p(A_k)$, of degree r, is the polynomial that defines the shifting strategy (often $p(x) = (x - \lambda)(x - \bar{\lambda})$, where λ and $\bar{\lambda}$ are the two eigenvalues of the trailing 2×2 principal submatrix of A_k, the so-called *implicit double-shift*). Then successive Householder transformation of size $r + 1$ are performed in order to return the working matrix A_k to upper Hessenberg form. This operation is known as *bulge chasing*, due to the peculiar shape of the non-zero entries of the matrix along the steps of the algorithm. As in the first version, deflation is performed as soon as one of the sub-diagonal entries of A_k is sufficiently small.

Renaming proposal [edit]

Since in the modern implicit version of the procedure no QR decompositions are explicitly performed, some authors, for instance Watkins,[7] suggested changing its name to *Francis algorithm*. Golub and Van Loan use the term *Francis QR step*.

Interpretation and convergence [edit]

The QR algorithm can be seen as a more sophisticated variation of the basic "power" eigenvalue algorithm. Recall that the power algorithm repeatedly multiplies *A* times a single vector, normalizing after each iteration. The vector converges to an eigenvector of the largest eigenvalue. Instead, the QR algorithm works with a complete basis of vectors, using QR decomposition to renormalize (and orthogonalize). For a symmetric matrix *A*, upon convergence, *AQ* = *QΛ*, where *Λ* is the diagonal matrix of eigenvalues to which *A* converged, and where *Q* is a composite of all the orthogonal similarity transforms required to get there. Thus the columns of *Q* are the eigenvectors.

History [edit]

The QR algorithm was preceded by the *LR algorithm*, which uses the LU decomposition instead of the QR decomposition. The QR algorithm is more stable, so the LR algorithm is rarely used nowadays. However, it represents an important step in the development of the QR algorithm.

The LR algorithm was developed in the early 1950s by Heinz Rutishauser, who worked at that time as a research assistant of Eduard Stiefel at ETH Zurich. Stiefel suggested that Rutishauer use the sequence of moments $y_0^T A^k x_0$, $k = 0, 1, \ldots$ (where x_0 and y_0 are arbitrary vectors) to find the eigenvalues of *A*. Rutishauer took an algorithm of Alexander Aitken for this task and developed it into the *quotient–difference algorithm* or *qd algorithm*. After arranging the computation in a suitable shape, he discovered that the qd algorithm is in fact the iteration $A_k = L_k U_k$ (LU decomposition), $A_{k+1} = U_k L_k$, applied on a tridiagonal matrix, from which the LR algorithm follows.[8]

Other variants [edit]

One variant of the *QR algorithm*, *the Golub-Kahan-Reinsch* algorithm starts with reducing a general matrix into a bidiagonal one.[9] This variant of the *QR algorithm* for the computation of singular values was first described by Golub & Kahan (1965). The LAPACK subroutine DBDSQR implements this iterative method, with some modifications to cover the case where the singular values are very small (Demmel & Kahan 1990). Together with a first step using Householder reflections and, if appropriate, QR decomposition, this forms the DGESVD routine for the computation of the singular value decomposition.

References [edit]

1. **^** J.G.F. Francis, "The QR Transformation, I", *The Computer Journal*, vol. 4, no. 3, pages 265-271 (1961, received Oct 1959) online at oxfordjournals.org;
J.G.F. Francis, "The QR Transformation, II" *The Computer Journal*, vol. 4, no. 4, pages 332-345 (1962) online at oxfordjournals.org.
Vera N. Kublanovskaya, "On some algorithms for the solution of the complete eigenvalue problem," *USSR Computational Mathematics and Mathematical Physics*, vol. 1, no. 3, pages 637–657 (1963, received Feb 1961). Also published in: *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, vol.1, no. 4, pages 555–570 (1961).
2. **^** Golub, G. H. and Van Loan, C. F.: Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, 1996, ISBN 0-8018-5414-8.
3. ^ ***a*** ***b*** James W. Demmel, *Applied Numerical Linear Algebra* (SIAM, 1997).
4. ^ ***a*** ***b*** Lloyd N. Trefethen and David Bau, *Numerical Linear Algebra* (SIAM, 1997).
5. **^** James M. Ortega and Henry F. Kaiser, "The LL^T and *QR* methods for symmetric tridiagonal matrices," *The Computer Journal* **6** (1), 99–101 (1963).

6. ^ Golub and Van Loan, chapter 7
7. ^ Watkins, David S.: The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods, SIAM, Philadelphia, PA, USA, 2007, ISBN 0-89871-641-1, ISBN 978-0-89871-641-2
8. ^ Parlett, Beresford N.; Gutknecht, Martin H. (2011), "From qd to LR, or, how were the qd and LR algorithms discovered?", *IMA Journal of Numerical Analysis* **31**: 741–754, doi:10.1093/imanum/drq003, ISSN 0272-4979
9. ^ Bochkanov Sergey Anatolyevich. ALGLIB User Guide - General Matrix operations - Singular value decomposition . ALGLIB Project. 2010-12-11. URL:http://www.alglib.net/matrixops/general/svd.php. Accessed: 2010-12-11. (Archived by WebCite at http://www.webcitation.org/5utO4iSnR)

External links [edit]

- Eigenvalue problem at PlanetMath.org.
- Prof. Peter Olver's notes on orthogonal bases and the workings of the QR algorithm
- Module for the QR Method

v · t · e	Numerical linear algebra [hide]
Key concepts	Floating point · Numerical stability
Problems	Matrix multiplication (algorithms) · Matrix decompositions · Linear equations · Sparse problems
Hardware	CPU cache · TLB · Cache-oblivious algorithm · SIMD · Multiprocessing
Software	BLAS · Specialized libraries · General purpose software

Categories: Numerical linear algebra