Calcul des valeurs propres par l'algorithme itératif QR

Analyse exploratoire: **« Approximating Eigenvalues with the QR Algorithm »**

Pages 398-399 du livre de référence D. Poole.

Approximating Eigenvalues with the QR Algorithm

See G. H. Golub and C. F. Van Loan, *Matrix Computations* (Baltimore: Johns Hopkins University Press, 1983).

One of the best (and most widely used) methods for numerically approximating the eigenvalues of a matrix makes use of the QR factorization. The purpose of this exploration is to introduce this method, the ***QR* algorithm,** and to show it at work in a few examples. For a more complete treatment of this topic, consult any good text on numerical linear algebra. (You will find it helpful to use a CAS to perform the calculations in the problems below.)

Given a square matrix A, the first step is to factor it as $A = QR$ (using whichever method is appropriate). Then we define $A_1 = RQ$.

1. First prove that A_1 is similar to A. Then prove that A_1 has the same eigenvalues as A.

2. If $A = \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix}$, find A_1 and verify that it has the same eigenvalues as A.

Continuing the algorithm, we factor A_1 as $A_1 = Q_1R_1$ and set $A_2 = R_1Q_1$. Then we factor $A_2 = Q_2R_2$ and set $A_3 = R_2Q_2$, and so on. That is, for $k \geq 1$, we compute $A_k = Q_kR_k$ and then set $A_{k+1} = R_kQ_k$.

3. Prove that A_k is similar to A for all $k \geq 1$.

4. Continuing Problem 2, compute A_2, A_3, A_4, and A_5, using two-decimal-place accuracy. What do you notice?

It can be shown that if the eigenvalues of A are all real and have distinct absolute values, then the matrices A_k approach an upper triangular matrix U.

5. What will be true of the diagonal entries of this matrix U?

6. Approximate the eigenvalues of the following matrices by applying the QR algorithm. Use two-decimal-place accuracy and perform at least five iterations.

(a) $\begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}$ (b) $\begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}$

(c) $\begin{bmatrix} 1 & 0 & -1 \\ 1 & 2 & 1 \\ -4 & 0 & 1 \end{bmatrix}$ (d) $\begin{bmatrix} 1 & 1 & -1 \\ 0 & 2 & 0 \\ -2 & 4 & 2 \end{bmatrix}$

7. Apply the QR algorithm to the matrix $A = \begin{bmatrix} 2 & 3 \\ -1 & -2 \end{bmatrix}$. What happens? Why?

8. Shift the eigenvalues of the matrix in Problem 7 by replacing A with $B = A + 0.9I$. Apply the QR algorithm to B and then shift back by subtracting 0.9 from the (approximate) eigenvalues of B. Verify that this method approximates the eigenvalues of A.

i) Algorithme QR itératif de base. Soit A_0 une matrice carrée d'ordre n dont on cherche les valeurs propres. On calcule $A_0 = Q_0 R_0$, on calcule finalement $A_1 = R_0 Q_0$. On recommence le cycle pour la matrice A_1 en calculant $A_1 = Q_1 R_1$ et ainsi de suite jusqu'à ce que $A_k = R_{k-1} Q_{k-1}$ converge vers une matrice triangulaire supérieure. Les éléments sur sa diagonale sont les valeurs propres de A_0.

ii) Algorithme QR itératif décalé. Soit A_0 une matrice carrée d'ordre n dont on cherche les valeurs propres. On construit $B_0 = A_0 - \sigma_0 I$, on calcule $B_0 = Q_0 R_0$, on calcule finalement $A_1 = R_0 Q_0 + \sigma_0 I$. On recommence le cycle pour la matrice A_1 en calculant $B_1 = A_1 - \sigma_1 I$ et ainsi de suite jusqu'à ce que $A_k = R_{k-1} Q_{k-1} + \sigma_{\kappa-1} I$ converge vers une matrice triangulaire supérieure. Les éléments sur sa diagonale sont les valeurs propres de A_0. À chaque cycle, on choisit $\sigma_\kappa = A_k(n,n)$.