LAB4: THE QR ALGORITHM TO FIND EIGENVALUES AND INVERSE ITERATION

One of the many applications of the QR-factorisation of a matrix is finding its eigenvalues.

We start by recalling some basic facts from linear algebra. A number $\lambda \in \mathbb{C}$ is an eigenvalue of $A \in \mathbb{R}^{n\times n}$ if there is a nonzero vector $\boldsymbol{v} \in \mathbb{C}^n$ such that $A\boldsymbol{v} = \lambda\boldsymbol{v}$. If A is symmetric then all the eigenvalues of A are real. If Q is an orthogonal matrix then the eigenvalues of the matrix Q^TAQ are the same as the eigenvalues of A.

1. The QR algorithm for finding eigenvalues

To keep matters simple, in what follows we suppose that $A \in \mathbb{R}^{n\times n}$ is a symmetric tridiagonal matrix.[1] The QR algorithm for finding eigenvalues of A defines a sequence of symmetric tridiagonal matrices $A^{(k)}$, $k = 0, 1, 2, \ldots$. It starts with $A^{(0)} = A$. Having constructed the matrix $A^{(k)}$ we then form the QR-factorization of $A^{(k)} - \mu_k I$, with a suitable number $\mu_k \in \mathbb{R}$ (an eigenvalue estimate):

$$A^{(k)} - \mu_k I = Q^{(k)} R^{(k)}. \tag{1}$$

We then construct the next matrix

$$A^{(k+1)} = R^{(k)} Q^{(k)} + \mu_k I.$$

As $Q^{(k)}$ is an orthogonal matrix, it follows from (1) that

$$Q^{(k)\top} A^{(k)} Q^{(k)} = R^{(k)} Q^{(k)} + \mu_k I = A^{(k+1)}.$$

Therefore, for all $k = 0, 1, \ldots$, the matrices $A^{(k)}$ have the same eigenvalues; in particular, the matrices $A^{(k)}$, $k = 1, 2, \ldots$ have the same eigenvalues as $A^{(0)} = A$. It is not hard to see that the matrices $A^{(k)}$, $k = 1, 2, \ldots$ are symmetric and tridiagonal. A usual choice for the shift is $\mu_k = a^{(k)}_{n,n}$, the last diagonal element of $A^{(k)}$. Whenever an offdiagonal element $a^{(k)}_{j,j+1}$ (usually $a^{(k)}_{n-1,n}$) is sufficiently close to 0 (smaller in absolute value than a given tolerance), then we consider the element $a^{(k)}_{j,j}$ a sufficiently good approximation to an eigenvalue of $A^{(k)}$ and hence of A. We remove the corresponding row and column and continue the iteration with the reduced matrix. The convergence analysis of this method is long and technical and it is beyond the scope of this course.

2. Inverse iteration for finding eigenvectors

The QR algorithm for finding eigenvalues does not provide the corresponding eigenvectors. However, a simple iteration can be set up as follows. As above suppose that $A \in \mathbb{R}^{n\times n}$ is a symmetric tridiagonal matrix (again, as above, being tridiagonal is not essential). Suppose that we have obtained a good approximation $\vartheta \in \mathbb{R}$ to

[1] When A is symmetric but not tridiagonal, then one can always find an orthogonal matrix Q (product of Householder triangularization matrices) such that the matrix Q^TAQ is tridiagonal.

an eigenvalue $\lambda \in \mathbb{R}$, which we assume to be simple, of A and we wish to find the normalized eigenvector $\boldsymbol{v} \in \mathbb{R}^n$ corresponding to λ; that is $A\boldsymbol{v} = \lambda\boldsymbol{v}$ and $\|\boldsymbol{v}\|_2 = 1$. Suppose that $\vartheta \neq \lambda$. First start with an initial approximation $\boldsymbol{v}^{(0)} \in \mathbb{R}^n$ with $\|\boldsymbol{v}^{(0)}\|_2 = 1$. Having found $\boldsymbol{v}^{(k)}$ we first solve

$$(A - \vartheta I)\boldsymbol{w}^{(k)} = \boldsymbol{v}^{(k)}$$

then set

$$\boldsymbol{v}^{(k+1)} = \frac{1}{\|\boldsymbol{w}^{(k)}\|_2}\boldsymbol{w}^{(k)}.$$

If λ is a simple eigenvalue of A, closest to ϑ, with normalized eigenvector $\boldsymbol{v}$ and $\boldsymbol{v}^{(0)}$ is not orthogonal to $\boldsymbol{v}$, then it is possible to prove that $\|\boldsymbol{v}^{(k)} - \boldsymbol{v}\|_2 \to 0$ as $k \to \infty$. The convergence is usually rapid if ϑ is within rounding error of λ and the eigenvalues of A are well-spaced.

Exercise 1. Let $n \geq 2$ and $A \in \mathbb{R}^{n\times n}$ be the matrix

$$A = (n+1)^2 \begin{bmatrix} 2 & -1 & 0 & \dots & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & \dots & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \\ 0 & 0 & 0 & \dots & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & \dots & 0 & 0 & -1 & 2 \end{bmatrix}.$$

(a) Write a code that implements the QR algorithm for finding the eigenvalues of A. For $n = 10$, find all eigenvalues of A using your code. Compare the results with the exact values $\lambda_k = 4(n+1)^2 \sin^2(\frac{k\pi}{2(n+1)})$, $k = 1, 2, ..., n$.

(b) Use inverse iteration to find the eigenvector corresponding to the smallest eigenvalue of A. Use your approximation from part (a) for the smallest eigenvalue of A to set up the inverse iteration. Compare your results with the exact eigenvector given by $(\boldsymbol{v})_i = \sin(\frac{\pi i}{n+1})$ (normalize the latter if necessary).

(c) For those who are interested (that is, this part is not compulsory): calculate the limit

$$\lim_{n\to\infty} \lambda_k = \lim_{n\to\infty} 4(n+1)^2 \sin^2\left(\frac{k\pi}{2(n+1)}\right).$$