QR Iterations

Consider the iteration
\[
\begin{align*}
Q_i R_i & \leftarrow A_i \\
A_{i+1} & \leftarrow R_i Q_i
\end{align*}
\]

Here we have first computed the QR factorization of \( A_i \), and then reversed their product to form \( A_{i+1} \). From \( A_i = Q_i R_i \) we have \( R_i = Q_i^t A_i \), and substituting that into \( A_{i+1} = R_i Q_i \) gives
\[
A_{i+1} = Q_i^t A_i Q_i
\]

and thus the QR step is a similarity transformation!

If the eigenvalues of \( A \) are all real, then this iteration almost always converges to an upper triangular matrix \( T \). In this limit, the eigenvalues of \( T \) (and hence \( A \), right?) are \( t_{11}, t_{22}, \ldots t_{nn} \). \( T \) is called a Schur form for \( A \) and the eigenvectors of \( T \) are Schur vectors of \( A \). Every matrix is unitarily similar to an upper triangular matrix, and \( T = Q^* A Q \) is called a Schur decomposition of \( A \).

As it stands, this QR iteration requires \( O(n^3) \) flops per iteration. We can reduce this by an order of magnitude by first reducing \( A \) to Hessenberg form \( H_0 = Q^t A Q \). The following iteration preserves the Hessenberg form, and if we use a Householder (or Givens) QR factorization it requires only \( O(n^2) \) flops:
\[
\begin{align*}
Q_i R_i & \leftarrow H_i \\
H_{i+1} & \leftarrow R_i Q_i
\end{align*}
\]

Notice that if a Hessenberg matrix \( H \) has \( h_{k+1,k} = 0 \), then the eigenproblem decouples: it is a block triangular matrix, and the eigenvalues of \( H \) are the union of the eigenvalues of the diagonal blocks (which are Hessenberg). A Hessenberg matrix for which none of the subdiagonal elements are zero is called unreduced.

Now if \( \lambda \) is an eigenvalue of an unreduced Hessenberg matrix \( H \), then the QR factorization \( QR = H - \lambda I \) will have \( r_{nn} = 0 \) (right?), and thus \( H_{new} = RQ + \lambda I \) will have last row \( \lambda e_n^t \). So what? \( H_{new} = Q^t HQ \) is a reduced Hessenberg matrix: we just decoupled \( \lambda \)!

Now we don’t usually know \( \lambda \) a priori, but we can speed convergence of the QR iterations by shifting \( H_i \) at each step by an approximate eigenvalue:
\[
\begin{align*}
Q_i R_i & \leftarrow H_i - s_i I \\
H_{i+1} & \leftarrow R_i Q_i + s_i I
\end{align*}
\]

This iteration is one of the most used methods to compute the eigenvalues and eigenvectors of symmetric (or Hermitian) matrices. In this case, \( H \) is both upper and lower Hessenberg, (called tridiagonal) and has only real eigenvalues. Furthermore, the QR iteration in this case requires only \( O(n) \) flops.

For nonsymmetric matrices we still have to address complex eigenvalues and the added cost complex arithmetic...