HW8

Saturday, April 13, 2024 8:09 PM

1. **[Q1]** ($25 = 10 + 10 + 5$ pts) **(a)** Let H be the initial input matrix for the shifted QR iteration. Show that $(H-\mu^{(k)}I)\cdots(H-\mu^{(2)}I)(H-\mu^{(1)}I) = \underline{Q}^{(k)}\underline{R}^{(k)}$ (in practice, H is upper Hessenberg, but we do not need this assumption here)

   (b) Other than the single real Wilkinson shift, we may also let $\mu^{(k)} = h_{nn}^{(k-1)}$ if $h_{n(n-1)}^{(k-1)}$ is small. Assume, for example, that $H^{(k-1)} = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \delta & h_{nn}^{(k-1)} \end{bmatrix}$. After the application of $n-2$ Givens rotations to $H^{(k-1)} - h_{nn}^{(k-1)}I$, we have the intermediate matrix $H_{tmp}^{(k-1)} = \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & a & b \\ 0 & 0 & 0 & \delta & 0 \end{bmatrix}$ (make sure you understand why it is of this form), and the last Givens rotation is needed on the left before we compute $H^{(k)}$ by transposed Givens rotations. Show that the new matrix $H^{(k)} = R^{(k)}Q^{(k)} + h_{nn}^{(k-1)}I$ satisfies $h_{n(n-1)}^{(k)} = -\frac{b\delta^2}{a^2+\delta^2}$. What does this observation suggest, if $\left|h_{n(n-1)}^{(k-1)}\right| = |\delta| \ll 1$, and either $|b| < 2|a|$ (δ can be arbitrary) or if $|\delta| < \frac{a^2}{|b|}$?

   (c) What can we say about $h_{n(n-1)}^{(k)}$ if A is real symmetric, such that $H^{(k-1)}$ is also real symmetric (hence tridiagonal)? In particular, does this entry decrease more slowly or more rapidly in the symmetric case than in the nonsymmetric case?

a) $\underline{Q}^{(k)}\underline{R}^{(k)} = Q^{(1)} \cdots Q^{(k)}R^{(k)} \cdots R^{(1)}$

$= Q^{(1)} \cdots Q^{(k-1)}(A^{(k-1)} - \mu^{(k)}I)R^{(k-1)} \cdots R^{(1)}$

$= Q^{(1)} \cdots Q^{(k-1)}Q^{(k-1)T}(A^{(k-2)} - \mu^{(k)}I)Q^{(k-1)}R^{(k-1)} \cdots R^{(1)}$

$= Q^{(1)} \cdots Q^{(k-2)}(A^{(k-2)} - \mu^{(k)}I)(A^{(k-2)} - \mu^{(k-1)}I)R^{(k-2)} \cdots R^{(1)}$

$= Q^{(1)} \cdots Q^{(k-2)}Q^{(k-2)T}(A^{(k-3)} - \mu^{(k)}I)Q^{(k-2)}Q^{(k-2)T}(A^{(k-3)} - \mu^{(k)}I)Q^{(k-2)}R^{(k-2)} \cdots R^{(1)}$

$= Q^{(1)} \cdots Q^{(k-3)}(A^{(k-3)} - \mu^{(k)}I)(A^{(k-3)} - \mu^{(k-1)}I)(A^{(k-3)} - \mu^{(k-2)}I)R^{(k-3)} \cdots R^{(1)}$

$\vdots$

$= \prod_{j=k}^{1}(A - \mu^{(j)}I)$

b i) Since $h_{n(n-1)}^{(k)}$ is only updated by the last Given's rotation, consider,

$$\left[\begin{array}{c|cc} I_3 & \multicolumn{2}{c}{0} \\ \hline 0 & c & -s \\ & s & c \end{array}\right]\left[\begin{array}{c|cc} \nabla & \multicolumn{2}{c}{X} \\ \hline 0 & a & b \\ & \delta & 0 \end{array}\right]\left[\begin{array}{c|cc} I_3 & \multicolumn{2}{c}{0} \\ \hline 0 & c & s \\ & -s & c \end{array}\right]$$

where $\begin{bmatrix} c & -s \\ s & c \end{bmatrix}$ is the Given's rotation based on $\begin{bmatrix} a \\ \delta \end{bmatrix}$

$$= \left[\begin{array}{c|cc} \nabla & \multicolumn{2}{c}{X} \\ \hline 0 & \frac{a^2-b^2}{\sqrt{a^2+\delta^2}} & \frac{ba}{\sqrt{a^2+\delta^2}} \\ & 0 & \frac{\delta b}{\sqrt{a^2+\delta^2}} \end{array}\right]\left[\begin{array}{c|cc} I_3 & \multicolumn{2}{c}{0} \\ \hline 0 & c & s \\ & -s & c \end{array}\right]$$

$$\left[\begin{array}{c|cc} & 0 & \frac{\delta b}{\sqrt{a^2+\delta^2}} \end{array}\right]\left[\begin{array}{c|cc} 0 & -s & c \end{array}\right]$$

$$= \left[\begin{array}{c|cc} \nabla & X & \\ \hline 0 & X & X \\ & \frac{-b\delta^2}{a^2+\delta^2} & X \end{array}\right] \quad \text{Thus, } h^{(k)}_{n(n-1)} = \frac{-b\delta^2}{a^2+\delta^2}$$

bii) $\frac{b\delta^2}{a^2+\delta^2} < \frac{4b\delta^2}{b^2+4\delta^2} < \frac{4b\delta^2}{b^2+4\delta^2} < \frac{4\delta^2}{b} < 4\delta^2 << 4$

biii) $\frac{b\delta^2}{a^2+\delta^2} < \frac{a^2}{b(a^2+\delta^2)} < \frac{\delta}{a^2+\delta^2} < \delta << 1$

This suggest $h^{(k)}_{n(n-1)} \to 0$

c) $$\left[\begin{array}{c|cc} I_3 & 0 & \\ \hline 0 & c & -s \\ & s & c \end{array}\right]\left[\begin{array}{ccc|cc} \times & \times & 0 & 0 & \\ 0 & \times & \times & & \\ 0 & \delta & \times & & \\ \hline & 0 & & a & \delta \\ & & & \delta & 0 \end{array}\right]\left[\begin{array}{c|cc} I_3 & 0 & \\ \hline 0 & c & s \\ & -s & c \end{array}\right]$$

where $\begin{bmatrix} c & -s \\ s & c \end{bmatrix}$ is the Given's rotation based on $\begin{bmatrix} a \\ \delta \end{bmatrix}$

$$= \left[\begin{array}{c|cc} \nabla & X & \\ \hline 0 & \frac{a^2-\delta^2}{\sqrt{a^2+\delta^2}} & \frac{\delta a}{\sqrt{a^2+\delta^2}} \\ & 0 & \frac{\delta^2}{\sqrt{a^2+\delta^2}} \end{array}\right]\left[\begin{array}{c|cc} I_3 & 0 & \\ \hline 0 & c & s \\ & -s & c \end{array}\right]$$

$$= \left[\begin{array}{c|cc} \nabla & X & \\ \hline 0 & X & X \\ & -\frac{\delta^3}{a^2+\delta^2} & X \end{array}\right] \quad \text{Thus, } h^{(k)}_{n(n-1)} = -\frac{\delta^3}{a^2+\delta^2}$$

Which will converge faster to zero.

2. **[Q2]** (15 pts) Implement the single-shift QR step in MATLAB; that is, given an upper Hessenberg $H^{(k-1)}$ and shift $\mu^{(k)}$, we compute $Q^{(k)}R^{(k)} = H^{(k-1)} - \mu^{(k)}I$ by Givens rotations and then use these Givens rotations to compute $H^{(k)} = R^{(k)}Q^{(k)} + \mu^{(k)}I$. Make simple changes in my code to enforce the use of single (Wilkinson) shift only, even if complex arithmetic is needed. Assemble your single shift code with the uploaded subroutines. Test it with the matrix obtained by

```
load west0479;
A = full(west0479);
```

2. **[Q2]** (15 pts) Implement the single-shift QR step in MATLAB; that is, given an upper Hessenberg $H^{(k-1)}$ and shift $\mu^{(k)}$, we compute $Q^{(k)}R^{(k)} = H^{(k-1)} - \mu^{(k)}I$ by Givens rotations and then use these Givens rotations to compute $H^{(k)} = R^{(k)}Q^{(k)} + \mu^{(k)}I$. Make simple changes in my code to enforce the use of single (Wilkinson) shift only, even if complex arithmetic is needed. Assemble your single shift code with the uploaded subroutines. Test it with the matrix obtained by

```
load west0479;
A = full(west0479);
```

Compare the eigenvalues of your final $H^{(k)}$ (use `ordeig`) with those of A. Be aware that the ordering of eigenvalues must be consistent to make a meaningful comparison.

Calculating $\frac{\|\Lambda(A)-\Lambda(H)\|}{\|\Lambda(A)\|} \sim O(10^{-11})$ which about the same as I got testing with Matlab's QR.

3. **[Q3]** (10 pts) Implement the Arnoldi's method without and with reorthogonalization, and test the orthogonality of the column vectors in U_{50} for the matrix A generated by `u = cos((0:2048)/2048*pi); A = vander(u);` Is the reorthogonalization effective for generating an orthonormal basis?

Use Arnoldi with reorthogonalization to compute the 11 dominant eigenvalues and eigenvectors of `aerofoil_new`, using $m = 30, 60, 100$, and 150 dimensional Krylov subspaces. For each m, plot all eigenvalues $\{\lambda_i\}_{i=1}^n$ of A together with the eigenvalues $\{\mu_i\}_{i=1}^m$ of H_m on the complex plane. Intuitively, how do $\{\mu_i\}_{i=1}^m$ approximate $\{\lambda_i\}_{i=1}^n$ as m increases? Give the relative eigenresidual norm $\frac{\|AU_m w_i - \mu_i U_m w_i\|_2}{\|AU_m w_i\|_2}$ $(1 \le i \le 11)$ of the desired eigenpairs for each m in a table.

a) Reorthogonalization was effective $\|V'V - I\| \sim O(\varepsilon_m)$

The eigvals seem to cluster at the beginning where the most real eigvals are, as m increases, there is more spread to cover each part of the spectrum.

Pictures and tabel in code