Math/CS 466/666 Programming Project 2

QR Factorization

Your work should be presented in the form of a typed report using clear and properly punctuated English. Where appropriate include full program listings and output. If you choose to work in a group of two, please turn in independently prepared reports.

1. A matrix $Q \in \mathbf{R}^{n\times n}$ is called an orthogonal matrix if $Q^tQ = I$. An orthogonal matrix Q and an upper triangular matrix R is called a QR factorization of A if $A = QR$. The Frobenius norm of a matrix A, denoted by $\|A\|_F$, is defined as

$$\|A\|_F = \Big(\sum_{i=1}^{n}\sum_{j=1}^{n}|A_{i,j}|^2\Big)^{1/2}.$$

The $n \times n$ Hilbert H matrix is defined as the matrix with entries

$$H_{ij} = 1/(i+j-1) \qquad \text{for} \qquad i,j = 1,2,\ldots,n.$$

(i) Suppose A is a non-singular $n \times n$ matrix with columns denoted by u_i. Apply Gram–Schmidt orthogonalization to the u_i's to obtain an orthonormal basis v_i. Let Q be the matrix with columns consisting of the vectors v_i and let $R = Q^tA$. Prove the matrix Q is orthogonal and that R is upper triangular.

(ii) Suppose Q is an orthogonal matrix. Show that $QQ^t = I$.

(iii) [Extra Credit and Math/CS 666] Show that H is non-singular for any n.

2. Let A be a matrix with columns given by u_i for $i = 1,\ldots,n$. Consider the following modification of the Gram–Schmidt orthogonalization algorithm:

$$\begin{aligned}
&t_{1,1} = u_1 && v_1 = t_{1,1}/\|t_{1,1}\|\\[4pt]
&t_{1,2} = u_2 &&\\
&t_{2,2} = t_{1,2} - (v_1, t_{1,2})v_1 && v_2 = t_{2,2}/\|t_{2,2}\|\\[4pt]
&t_{1,3} = u_3 &&\\
&t_{2,3} = t_{1,3} - (v_1, t_{1,3})v_1 &&\\
&t_{3,3} = t_{2,3} - (v_2, t_{2,3})v_2 && v_3 = t_{3,3}/\|t_{3,3}\|\\[4pt]
&t_{1,4} = u_4 &&\\
&t_{2,4} = t_{1,4} - (v_1, t_{1,4})v_1 &&\\
&t_{3,4} = t_{2,4} - (v_2, t_{2,4})v_2 &&\\
&t_{4,4} = t_{3,4} - (v_3, t_{3,4})v_3 && v_4 = t_{4,4}/\|t_{4,4}\|\\
&\qquad\vdots &&\\
&t_{1,n} = u_n &&\\
&t_{2,n} = t_{1,n} - (v_1, t_{1,n})v_1 &&\\
&t_{3,n} = t_{2,n} - (v_2, t_{2,n})v_2 &&\\
&\qquad\cdots &&\\
&t_{n,n} = t_{n-1,n} - (v_{n-1}, t_{n-1,n})v_{n-1} && v_n = t_{n,n}/\|t_{n,n}\|.
\end{aligned}$$

Let Q be the matrix with columns v_i and let R be the matrix with entries

$$R_{i,j} = \begin{cases} (v_i, t_{i,j}) & \text{for } i < j \\ \|t_{i,i}\| & \text{for } i = j \\ 0 & \text{for } i > j. \end{cases}$$

(i) Show that $\|t_{i,i}\| = (v_i, t_{i,i})$ and $(v_i, t_{i,j}) = (v_i, u_j)$ for $i \le j$.

(ii) Prove the matrix Q defined above is orthogonal and that $A = QR$.

(iii) Let H be the Hilbert matrix and write a program that computes Q and R such that $H = QR$ using the modified Gram–Schmidt algorithm described above.

(iv) Use the above program to find an orthogonal matrix Q and an upper triangular matrix R such that $H = QR$ when $n = 4$.

(v) For the matrices Q and R found using the above program compute

$$\|H - QR\|_F, \qquad \|Q^t Q - I\|_F \qquad \text{and} \qquad \|QQ^t - I\|_F$$

when $n = 4$, 6, 8, 10, 12 and 20.

3. The LAPACK routine DGEQRFP computes the QR factorization of a matrix. After factorization, the upper triangular part of the matrix A is overwritten with R. The matrix Q may be obtained from DORGQR routine.

(i) Use these subroutines or equivalent ones to find an orthogonal matrix Q and an upper triangular matrix R such that $H = QR$ when $n = 4$.

(ii) Is the QR factorization found using LAPACK the same as the factorization found using the modified Gram–Schmidt algorithm in the previous question?

(iii) For the matrices Q and R found using LAPACK compute

$$\|H - QR\|_F, \qquad \|Q^t Q - I\|_F \qquad \text{and} \qquad \|QQ^t - I\|_F$$

when $n = 4$, 6, 8, 10, 12 and 20.

(iv) Compare the accuracy of the results obtained using LAPACK with the accuracy of the results obtained in using the modified Gram–Schmidt algorithm.

(v) [Extra Credit and Math/CS 666] Look up the the algorithm used by LAPACK and explain how it works. What is special about the Hilbert matrix? Compare the speed and accuracy of the two methods when finding the QR factorization for random $n \times n$ matrices where $n = 10$, 50, 100, 500, 1000 and 2000.