Section exam 2 Numerical Analysis: Linear Algebra
9:30am – 10:45am, October 24, 2024.

Instructions:

- ***This is a closed books exam.*** No books or calculators are allowed.
  However, you are allowed to bring one hand written, single sided formula sheet.
- Try to answer the questions within the space given. If you must use more space, you *can* attach extra pages, but try to avoid it. If you do need extra space, then please mark this very clearly on the exam.
- If a question does not specifically ask for a motivation, then the correct answer alone will give full points. You are welcome to enter a brief explanation of how you arrived at the answer — this might yield partial credit in case the given answer is incorrect.
- Use your time strategically. Note in particular that Question 6 is only worth 10 points.
- In this exam, the default is that for a vector $\mathbf{x}$, the notation $\|\mathbf{x}\|$ denotes the Euclidean norm. For a matrix $\mathbf{A}$, the notation $\|\mathbf{A}\|$ refers to the operator norm with respect to the Euclidean vector norm, and $\|\mathbf{A}\|_F$ refers to the Frobenius norm. If $\mathbf{A}$ is a matrix, then $\mathbf{A}^*$ refers to the complex conjugate of the transpose of $\mathbf{A}$ (or merely the transpose when $\mathbf{A}$ is real).

Name: ____________________________________

Section: ________________

Question	Max points	Scored points
1	20	
2	20	
3	15	
4	15	
5	20	
6	10	
Total:		

Question 1: (20p) Set $\mathbf{A} = \begin{bmatrix} 1 & 2 & -1 & 1 \\ 2 & 6 & -1 & 5 \\ -1 & 0 & 3 & 1 \\ -3 & -12 & -1 & -9 \end{bmatrix}$. In the first two steps of the LU factorization of $\mathbf{A}$, we form $\mathbf{A}_1 = \mathbf{L}_1\mathbf{A} = \begin{bmatrix} 1 & 2 & -1 & 1 \\ 0 & 2 & 1 & 3 \\ 0 & 2 & 2 & 2 \\ 0 & -6 & -4 & -6 \end{bmatrix}$ and $\mathbf{A}_2 = \mathbf{L}_2\mathbf{A}_1 = \begin{bmatrix} 1 & 2 & -1 & 1 \\ 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 3 \end{bmatrix}$.

Please provide only answers to these questions. (Motivations will not be graded.)

(a) (7p) Specify the following matrices:

$\mathbf{L}_1 =$ $\qquad$ $\mathbf{L}_2 =$ $\qquad$ $\mathbf{L}_2^{-1} =$ $\qquad$ $\mathbf{L}_1^{-1}\mathbf{L}_2^{-1} =$

(b) (7p) Complete the LU factorization of $\mathbf{A}$ so that $\mathbf{A} = \mathbf{LU}$. Specify:

$\mathbf{L} =$ $\qquad$ $\mathbf{U} =$

(c) (6p) Specify the determinants of the matrices involved:

$\det(\mathbf{L}) =$ $\qquad$ $\det(\mathbf{U}) =$ $\qquad$ $\det(\mathbf{A}) =$

Question 2: (20p) For a given matrix $\mathbf{X}$, let $\sigma_{\max}(\mathbf{X})$ and $\sigma_{\min}(\mathbf{X})$ denote the largest and the smallest singular values of $\mathbf{X}$, respectively. Let m and n be positive integers such that $m > n$. You know that the $m \times n$ matrix $\mathbf{A}$ has the largest and smallest singular values $\sigma_{\max}(\mathbf{A}) = 6$ and $\sigma_{\min}(\mathbf{A}) = 2$. In this question, let $\mathbf{A}^\dagger$ denote the pseudo inverse of $\mathbf{A}$, let $\kappa(\mathbf{A})$ denote the condition number of $\mathbf{A}$, and let $\mathbf{Q} \in \mathbb{R}^{m\times n}$ and $\mathbf{R} \in \mathbb{R}^{n\times n}$ denote the factors in the QR factorization $\mathbf{A} = \mathbf{QR}$. Specify the following quantities without motivation:

$\kappa(\mathbf{A}) =$ $\qquad$ $\sigma_{\max}(\mathbf{R}) =$ $\qquad$ $\sigma_{\max}(\mathbf{A}^*\mathbf{A}) =$ $\qquad$ $\sigma_{\max}(\mathbf{AA}^*) =$ $\qquad$ $\sigma_{\max}(\mathbf{A}^\dagger) =$

$\kappa(\mathbf{A}^*\mathbf{A}) =$ $\qquad$ $\sigma_{\min}(\mathbf{R}) =$ $\qquad$ $\sigma_{\min}(\mathbf{A}^*\mathbf{A}) =$ $\qquad$ $\sigma_{\min}(\mathbf{AA}^*) =$ $\qquad$ $\sigma_{\min}(\mathbf{A}^\dagger) =$

Question 3: (15p) Let m be a positive integer, and let $\{x_i\}_{i=1}^m$ and $\{y_i\}_{i=1}^m$ be two sets of real numbers. We seek to fit the pairs $\{x_i,\, y_i\}_{i=1}^m$ to a function

$$y = f(x) = c_1 + c_2 \sin(x) + c_3\, x^2.$$

To be precise, we seek to determine real numbers $c_1,\, c_2,\, c_3$ such that the mean square error

$$E = \sum_{i=1}^{m} \big|\big(c_1 + c_2 \sin(x_i) + c_3\, x_i^2\big) - y_i\big|^2$$

is minimized. As we saw in class, minimizing E is equivalent to solving a linear system $\mathbf{Ac} = \mathbf{b}$ for the vector $\mathbf{c} = \big[c_1,\, c_2,\, c_3\big]^*$ in a least squares sense. Specify (without motivation) $\mathbf{A}$ and $\mathbf{b}$:

$\mathbf{A} =$ $\qquad$ $\mathbf{b} =$

Question 4: (15p) Let $\mathbf{B}$ denote the 10×10 "Hilbert matrix". In other words, $\mathbf{B}(i,j) = \frac{1}{i+j-1}$. Set

$$\mathbf{A} = \frac{1}{\|\mathbf{B}\|}\mathbf{B}.$$

The matrix $\mathbf{A}$ has 10 distinct positive eigenvalues $\{\lambda_j\}_{j=1}^{10}$. Suppose that these are ordered so that

$$\lambda_1 > \lambda_2 > \cdots > \lambda_{10} > 0.$$

(a) (6p) Let k be a positive integer. Specify the eigenvalues of $\mathbf{A}^k$. (Express the eigenvalues of $\mathbf{A}^k$ in terms of the quantities introduced. It is not recommended to try to compute them explicitly!) What is the (exact mathematical) rank of $\mathbf{A}^k$?

(b) (9p) Consider the code snippet:

```
A=(1/norm(hilb(10)))*hilb(10); C=A; for i=1:10; disp(rank(C)); C=A*C; end
```

(The function $r =$ `rank`$(\mathbf{A})$ outputs an estimate for the *numerical* rank r of $\mathbf{A}$. When answering this problem, you may take r to be the largest number r such that $\sigma_r/\sigma_1 > \epsilon_{\text{mach}}$ where $\{\sigma_j\}_{j=1}^{10}$ are the singular values of $\mathbf{A}$.) When executed in Matlab, the output is:

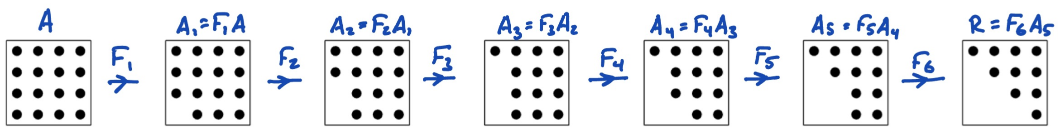
```
10
 7
 5
 4
 4
 3
 3
 3
 2
 2
```

Based on this, provide estimated bounds for λ_2, λ_5, and λ_{10}, expressed in terms of ϵ_{mach}.

Question 5: (20p) Given any vector $\mathbf{v} = [a,\, b]^* \in \mathbb{R}^2$, there exist real numbers c and s such that the matrix $\mathbf{G} = \begin{bmatrix} c & -s \\ s & c \end{bmatrix}$ is unitary, and satisfies

(1) $$\mathbf{G}\,\mathbf{v} = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sqrt{a^2+b^2} \\ 0 \end{bmatrix}.$$

Such a matrix is called a "Givens rotation" and can be used to compute the QR factorization of a given $m \times m$ matrix $\mathbf{A}$. The idea is similar to Householder QR, in that we apply transformations from the left that successively drive $\mathbf{A}$ to upper triangular form. The first step of the process is to apply a Givens rotation to the bottom two rows of $\mathbf{A}$ to create a zero in the $(m, 1)$ slot. Then continue to work upwards in the first column to successively introduce additional zeros. Then proceed with the second column, and so on. For a 4×4 matrix, the process involves six steps:



(a) (6p) Given a and b, provide formulas for c and s that result in a Givens rotation satisfying (1):

$c =$ $\qquad\qquad\qquad\qquad$ $s =$

(b) (6p) Specify a matrix $\mathbf{F}_1$ that contains a Givens rotation as a submatrix such that

$$\mathbf{F}_1 \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ a & \times & \times & \times \\ b & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \sqrt{a^2+b^2} & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix} \qquad \text{where} \qquad \mathbf{F}_1 =$$

(c) (8p) The figure shows Matlab code that drives a given matrix $\mathbf{A}$ to the factor $\mathbf{R}$ in a QR factorization $\mathbf{A} = \mathbf{QR}$, with $\mathbf{Q}$ unitary. Fill in the missing lines so that the code builds the required Givens matrix $\mathbf{G}$ using `mygivens`, and then applies it to the two relevant rows of $\mathbf{R}$:

```
function R = computeR(A)
m = size(A,1);
R = A;
for j = 1:(m-1)
  for i = (m-1):(-1):j
                          <- CODE MISSING HERE!



  end
end
return
```

```
function G = mygivens(v)
c =                       <- INSERT MISSING CODE
s =
G = [c,-s;s,c];
return
```

Question 6: (10p) Let $\mathbf{A}$ be a matrix of size $m \times n$. Suppose that $m < n$ and that $\text{rank}(\mathbf{A}) = m$. Given a vector $\mathbf{b} \in \mathbb{R}^m$, we seek to solve

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

for $\mathbf{x}$ in a least squares sense. We covered in class how to do this using the SVD. Suppose that you are working in a software environment where you do not have access to a routine for computing the SVD, but you do have access to a routine for computing QR factorizations. Could you use this to construct the solution $\mathbf{x}$? If yes, then write down a formula for $\mathbf{x}$, and prove that it is the unique least squares solution. If no, then motivate why the problem cannot be solved using QR factorization.