# Homework 2 Foundations of Computational Math 2 Spring 2019

## Problem 2.1

(2.1.a) Suppose $(v_1, \lambda_1)$ and $(v_2, \lambda_2)$ are eigenpairs for a matrix $A \in \mathbb{C}^{n\times n}$. Show that if $\lambda_1 \neq \lambda_2$ then $v_1$ and $v_2$ are linearly independent.

(2.1.b) Suppose that $(v_i, \lambda_i)$ for $i = 1, \ldots, n$ are eigenpairs for a matrix $A \in \mathbb{C}^{n\times n}$ where $\lambda_i \neq \lambda_j$ for any $i \neq j$. Show that $\{v_i\}$ for $i = 1, \ldots, n$ is a set of linear independent vectors.

(2.1.c) What decomposition can be written for the matrix $A$ in part (2.1.b)?

## Problem 2.2

(2.2.a) Suppose the matrix $A \in \mathbb{C}^{m\times n}$ with $m \geq n$. Show that $M = A^H A$ is an Hermitian positive semidefinite matrix.

(2.2.b) If $B \in \mathbb{C}^{n\times n}$ then there exists $Q \in \mathbb{C}^{n\times n}$ and $R \in \mathbb{C}^{n\times n}$ where $Q^H Q = I \in \mathbb{C}^{n\times n}$ and $R$ is an upper triangular matrix where the diagonal elements of $R$ are the eigenvalues of $B$ including algebraic multiplicity such that $B = QRQ^H$. This is the Schur Decomposition of a matrix. Show that if $B$ is an Hermitian matrix then it has real eigenvalues and a set of orthonomal eigenvectors.

(2.2.c) Show that if $B \in \mathbb{C}^{n\times n}$ is an Hermitian positive semidefinite matrix then its eigenvalues are real and nonnegative. What form of decomposition of $B$ does this statement imply?

(2.2.d) Suppose the matrix $A \in \mathbb{C}^{m\times n}$ with $m \geq n$. Show that the SVD of $A$ exists, i.e., show there exists $U \in \mathbb{C}^{m\times m}$, $V \in \mathbb{C}^{n\times n}$, $\Sigma \in {}^{n\times n}$, and $S \in \mathbb{C}^{m\times n}$ such that $U^H U = I_m$, $V^H V = I_n$,

$$S = \begin{pmatrix} \Sigma \\ 0 \end{pmatrix}, \quad \text{and} \quad A = USV^H$$

with $\Sigma$ a diagonal matrix with nonnegative diagonal elements.

## Problem 2.3

Suppose $A \in \mathbb{R}^{m\times n}$, $m \geq n$ and $rank(A) = p \leq n$. Show that there exists $X \in \mathbb{R}^{m\times p}$ and $Y \in \mathbb{R}^{\times p}$ such that $rank(X) = rank(Y) = p$ and

$$A = XY^T$$

# Problem 2.4

# Definitions

Let $F_n \in \mathbb{C}^{n\times n}$ be the unitary matrix representing the discrete Fourier transform of length $n$ and so $F_n^H \in \mathbb{C}^{n\times n}$ is the inverse DFT of length $n$. For example, for $n = 4$

$$F_4 = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \mu & \mu^2 & \mu^3 \\ 1 & \mu^2 & \mu^4 & \mu^6 \\ 1 & \mu^3 & \mu^6 & \mu^9 \end{pmatrix} \quad \text{and} \quad F_4^H = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 \\ 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega^9 \end{pmatrix}$$

where $\theta = 2\pi/n$, $\omega = e^{i\theta}$ and $\mu = e^{-i\theta}$.

Let $Z_n \in \mathbb{C}^{n\times n}$ be the permutation matrix of order $n$ such that $Zv$ represents the circulant "upshift" of the elements of the vector $v$, i.e.,

$$Z_n = \begin{pmatrix} e_n & e_1 & e_2 & \dots & e_{n-1} \end{pmatrix}.$$

For example, for $n = 4$

$$Z_4 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Let $C_n \in \mathbb{C}^{n\times n}$ be a circulant matrix of order $n$. The circulant matrix $C_n$ has $n$ parameters (either the first row or first column can be viewed as these parameters). It is a Toeplitz matrix (all diagonals are constant) with the additional constraint that each row (column) is a circulant shift of the previous row (column).

For example, for $n = 4$ and using the first row as the parameters we have

$$C_4 = \begin{pmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_3 & \alpha_0 & \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_3 & \alpha_0 & \alpha_1 \\ \alpha_1 & \alpha_2 & \alpha_3 & \alpha_0 \end{pmatrix}$$

Given a polynomial of degree $d$, a matrix polynomial is defined as follows

$$P_d(\xi) = \delta_0 + \delta_1\xi + \delta_2\xi^2 + \cdots + \delta_d\xi^d$$

$$P_d(A) = \delta_0 I + \delta_1 A + \delta_2 A^2 + \cdots + \delta_d A^d$$

$$\xi, \in \mathbb{C}, \quad \delta_i \in \mathbb{C}, \quad P_d(A), \; A \in \mathbb{C}^{n\times n}.$$

Hint: For the problems below it might be useful to consider a small dimension, e.g., $n = 4$ and then generalize the proofs and results to any $n$.

(2.4.a) Determine a diagonal matrix $\Lambda_n \in \mathbb{C}^{n\times n}$ i.e., nonzero elements may only appear on the main diagonal, that satisfies $Z_n = F_n^H \Lambda_n F_n$. This says that the columns of $F_n^H$ are the eigenvectors of $Z_n$ and the associated eigenvalues are the elements on the diagonal of $\Lambda_n$.

(2.4.b) Recall, that the set of $n \times n$ matrices is a vector space with dimension $n^2$. Show that the set of $n \times n$ circulant matrices, $C_n$, is a subspace of that vector space with dimension $n$. Hint: find a basis for the subspace using the results and definitions above.

(2.4.c) Show that any circulant matrix can be written

$$C_n = F_n^H \Gamma_n F_n$$

where $\Gamma_n \in \mathbb{C}^{n\times n}$ is a diagonal matrix. This says that the columns of $F_n^H$ are the eigenvectors of $C_n$ and the associated eigenvalues are the elements on the diagonal of $\Gamma_n$. Your proof should develop a formula for $\Gamma_n$ that allows its diagonal elements to be easily evaluated and understood.

(2.4.d) Describe how you determine if $C_n$ is a nonsingular matrix.

(2.4.e) How does this factorization of $C_n$ result in a fast method of solving a linear system $C_n x = b$, where $x,\ b \in \mathbb{C}^n$. (Here a fast method is one that has complexity less than the $O(n^3)$ computations associated with standard factorization methods.)

## Problem 2.5

Recall that a unit lower triangular matrix $L \in \mathbb{R}^{n\times n}$ is a lower triangular matrix with diagonal elements $e_i^T L e_i = \lambda_{ii} = 1$. An elementary unit lower triangular column form matrix, $L_i$, is an elementary unit lower triangular matrix in which all of the nonzero subdiagonal elements are contained in a single column. For example, for $n = 4$

$$L_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \lambda_{21} & 1 & 0 & 0 \\ \lambda_{31} & 0 & 1 & 0 \\ \lambda_{41} & 0 & 0 & 1 \end{pmatrix} \quad L_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \lambda_{32} & 1 & 0 \\ 0 & \lambda_{42} & 0 & 1 \end{pmatrix} \quad L_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \lambda_{43} & 1 \end{pmatrix}$$

**2.5.a**. Show that any elementary unit lower triangular column form matrix , $L_i \in \mathbb{R}^{n\times n}$, can be written as the identity matrix plus an outer product of two vectors, i.e., $L_i = I + v_i w_i^T$ where $v_i \in \mathbb{R}^n$ and $w_i \in \mathbb{R}^n$. (This is often called a rank-1 update of a matrix.) Make sure the structure required in $v_i$ and $w_i$ is clearly stated.

**2.5.b**. Show that $L_i$ has an inverse and it is an elementary unit lower triangular column form matrix.

**2.5.c**. Consider the matrix vector product $y = L_i x$ where $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$, and $L_i \in \mathbb{R}^{n\times n}$ is an elementary unit lower triangular column form matrix. Determine an efficient algorithm to compute the product and its computational/storage complexity.

**2.5.d**. Suppose $L_i \in \mathbb{R}^{n\times n}$ and $L_j \in \mathbb{R}^{n\times n}$ are elementary unit lower triangular column form matrices with $1 \leq i < j \leq n-1$. Consider the matrix product $B = L_i L_j$. Determine an efficient algorithm to compute the product and its computational/storage complexity.

**2.5.e**. Suppose $L_i \in \mathbb{R}^{n\times n}$ and $L_j \in \mathbb{R}^{n\times n}$ are elementary unit lower triangular column form matrices with $1 \leq j \leq i \leq n-1$. Consider the matrix product $B = L_i L_j$. Determine an efficient algorithm to compute the product and its computational/storage complexity.

**2.5.f**. Let $L \in \mathbb{R}^{n\times n}$ be a unit lower triangular matrix. Show that $L = L_1 L_2 \cdots L_{n-1}$ where $L_i$ is an elementary unit lower triangular column form matrix for $1 \leq i \leq n-1$.

**2.5.g**. Express the column-oriented algorithm for solving $Lx = b$ where $L$ is a unit lower triangular matrix in terms of operations involving unit lower triangular column form matrices.

# Problem 2.6

## 2.6.a

An **elementary unit upper triangular column form matrix** $U_i \in \mathbb{R}^{n\times n}$ is of the form

$$I + u_i e_i^T$$

where $u_i^T e_j = 0$ for $i \leq j \leq n$. This matrix has 1 on the diagonal and the nonzero elements of $u_i$ appear in the $i$-th column above the diagonal.

For example, if $n = 3$ then

$$\begin{aligned} U_3 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} \mu_{13} \\ \mu_{23} \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & \mu_{13} \\ 0 & 1 & \mu_{23} \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

Let $U \in \mathbb{R}^{n\times n}$ be a unit upper triangular matrix. Show that the factorization

$$U = U_n U_{n-1} \cdots U_2,$$

where $U_i = I + u_i e_i^T$ and the nonzeros of $u_i$ are the nonzeros in the $i$-th column of $U$ above the diagonal, can be formed without any computations.

### 2.6.b

Now suppose that $U \in \mathbb{R}^{n\times n}$ is a upper triangular with nonzero diagonal elements $\mu_{ii}$. Let $S_i \in \mathbb{R}^{n\times n}$ be a diagonal matrix with its $i$-th diagonal element $e_i^T S_i e_i = \mu_{ii}$ and all of the other diagonal elements $e_j^T S_i e_j = 1$ for $i \neq j$.

For example, if $n = 3$ then

$$\begin{aligned}
S_1 &= \begin{pmatrix} \mu_{11} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
S_2 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \mu_{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
S_3 &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \mu_{33} \end{pmatrix}
\end{aligned}$$

Let $U_i = I + u_i e_i^T$ and the nonzeros of $u_i$ be the nonzeros in the $i$-th column of $U$ above the diagonal. (This implies that $U_1 = I$) Show that

$$U = (S_n U_n)(S_{n-1} U_{n-1}) \cdots (S_2 U_2)(S_1 U_1).$$

### 2.6.c

From the factorization of the previous part of the problem, derive an algorithm to solve $Ux = b$ given $U$ is an $n \times n$ nonsingular upper triangular matrix. Describe the basic computational primitives required.

## Problem 2.7

Recall that any unit lower triangular matrix $L \in \mathbb{R}^{n\times n}$ can be written in factored form as

$$L = M_1 M_2 \cdots M_{n-1} \qquad (1)$$

where $M_i = I + l_i e_i^T$ is an elementary unit lower triangular matrix (column form). Given the ordering of the elementary matrices, this factorization did not require any computation.

Consider a simpler elementary unit lower triangular matrix (element form) that differs from the identity in **one off-diagonal element** in the strict lower triangular part, i.e.,

$$E_{ij} = I + \lambda_{ij} e_i e_j^T$$

where $i \neq j$.

**2.7.a**. Show that computing the product of two element form elementary matrices is simply superposition of the elements into the product given by

$$E_{ij} E_{rs} = I + \lambda_{ij} e_i e_j^T + \lambda_{rs} e_r e_s^T$$

whenever $j \neq r$.

**2.7.b**. Show that if $j \neq r$ and $i \neq s$ then computing $E_{ij} E_{rs}$ with requires no computation and

$$E_{ij} E_{rs} = E_{rs} E_{ij}$$

i.e., the matrices commute.

**2.7.c**. Write a column form elementary matrix $M_i$ in terms of element form elementary matrices. Does the order of the $E_{ji}$ matter in this product?

**2.7.d**. Show how it follows that the factorization of (1) is easily expressed in terms of element form elementary matrices.

**2.7.e**. Show that the expression from part (2.7.d) can be rearranged to form $L = R_2 \dots R_n$ where $R_i = I + e_i r_i^T$ is an elementary unit lower triangular matrix in row form.

# Problem 2.8

Consider the matrix vector product $x = Lb$ where $L$ is an $n \times n$ unit lower triangular matrix with **all** of its nonzero elements equal to 1. For example, if $n = 4$ then

$$x = Lb$$
$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \end{pmatrix}$$

The vector $x$ is called the scan of $b$. Show that $x$ can be computed in $O(n)$ computations rather than the $O(n^2)$ typically required by a matrix vector product. Express your solution in terms of matrices and vectors.

## Problem 2.9

Suppose that $A \in \mathbb{R}^{n\times n}$ is nonsingular and that $A = LU$ is its $LU$ factorization. Give an algorithm that can compute, $e_i^T A^{-1} e_j$, i.e., the $(i,j)$ element of $A^{-1}$ in approximately $(n-j)^2 + (n-i)^2$ operations.

## Problem 2.10

Consider an $n \times n$ real matrix where

- $\alpha_{ij} = e_i^T A e_j = -1$ when $i > j$, i.e., all elements strictly below the diagonal are $-1$;
- $\alpha_{ii} = e_i^T A e_i = 1$, i.e., all elements on the diagonal are 1;
- $\alpha_{in} = e_i^T A e_n = 1$, i.e., all elements in the last column of the matrix are 1;
- all other elements are 0

For $n = 4$ we have

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{pmatrix}$$

**2.10.a**. Compute the factorization $A = LU$ for $n = 4$ where $L$ is unit lower triangular and $U$ is upper triangular.

**2.10.b**. What is the pattern of element values in $L$ and $U$ for any $n$?

## Problem 2.11

Let $A \in \mathbb{R}^{n\times n}$ and its inverse be partitioned

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

$$A^{-1} = \begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{pmatrix}$$

where $A_{11} \in \mathbb{R}^{k\times k}$ and $\tilde{A}_{11} \in \mathbb{R}^{k\times k}$.

**2.11.a**. Show that if, $S = A_{22} - A_{21}A_{11}^{-1}A_{12}$, the Schur complement of $A$ with respect to $A_{11}$ exists then $A$ is nonsingular if and only if $S$ is nonsingular.

**2.11.b**. Show that $S^{-1} = \tilde{A}_{22}$.

## Problem 2.12

Define the elementary matrix $N(y,k) = I - ye_k^T \in \mathbb{R}^{n\times n}$, where $1 \le k \le n$ is an integer, $y \in \mathbb{R}^n$ and $e_k \in \mathbb{R}^n$ is the $k$-th standard basis vector. $N(y,k)$ is a Gauss-Jordan transform if it is defined by requiring $N(y,k)v = e_k$ for a given vector $v \in \mathbb{R}^n$.

Perform all of the basic analyses on the Gauss-Jordan transform that were performed on the Gauss transform and elementary unit lower triangular matrices, i.e., existence, inverse, etc., and use the results to show that the Gauss-Jordan algorithm that computes $A^{-1}$ and $x$ from a nonsingular matrix $A \in \mathbb{R}^{n\times n}$ and vector $b \in \mathbb{R}^n$ can be expressed in terms of Gauss-Jordan transforms. Identify the condition for each Gauss-Jordan transformation to exist on each step of the elimination, i.e., determine when the algorithm fails in terms of the transformation's construction.

## Problem 2.13

Suppose you have the LU factorization of an $i \times i$ matrix $A_i = L_i U_i$ and suppose the matrix $A_{i+1}$ is an $i+1 \times i+1$ matrix formed by adding a row and column to $A_i$, i.e.,

$$A_{i+1} = \begin{pmatrix} A_i & a_{i+1} \\ b_{i+1}^T & \alpha_{i+1,i+1} \end{pmatrix}$$

where $a_{i+1}$ and $b_{i+1}$ are vectors in $\mathbb{R}^i$ and $\alpha_{i+1,i+1}$ is a scalar.

**2.13.a**. Derive an algorithm that, given $L_i$, $U_i$ and the new row and column information, computes the LU factorization of $A_{i+1}$ and identify the conditions under which the step will fail.

**2.13.b**. What computational primitives are involved?

**2.13.c**. Show how this basic step could be used to form an algorithm that computes the LU factorization of an $n \times n$ matrix $A$.

## Problem 2.14

Consider a symmetric matrix $A$, i.e., $A = A^T$.

**2.14.a**. Consider the use of Gauss transforms to factor $A = LU$ where $L$ is unit lower triangular and $U$ is upper triangular. **You may assume that the factorization does not fail.** Show that $A = LDL^T$ where $L$ is unit lower triangular and $D$ is a matrix with nonzeros on the main diagonal. i.e., elements in positions $(i,i)$, and zero everywhere else, by demonstrating that $L$ and $D$ can be computed by applying Gauss transforms appropriately to the matrix $A$.

**2.14.b**. For an arbitrary symmetric matrix the $LDL^T$ factorization will not always exist due to the possibility of 0 in the $(i, i)$ position of the transformed matrix that defines the $i$-th Gauss transform. Suppose, however, that $A$ is a **positive definite** symmetric matrix, i.e., $x^T Ax > 0$ for any vector $x \neq 0$. Show that the diagonal element of the transformed matrix $A$ that is used to define the vector $l_i$ that determines the Gauss transform on step $i$, $M_i^{-1} = I - l_i e_i^T$, is always positive and therefore the factorization will not fail. Combine this with the existence of the $LDL^T$ factorization to show that, in this case, the nonzero elements of $D$ are in fact positive.

# Problem 2.15

Suppose $A \in \mathbb{R}^{n \times n}$ is a nonsymmetric nonsingular diagonally dominant matrix with the following nonzero pattern (shown for $n = 6$)

$$\begin{pmatrix} * & * & * & * & * & * \\ * & * & 0 & 0 & 0 & 0 \\ * & 0 & * & 0 & 0 & 0 \\ * & 0 & 0 & * & 0 & 0 \\ * & 0 & 0 & 0 & * & 0 \\ * & 0 & 0 & 0 & 0 & * \end{pmatrix}$$

It is known that a diagonally dominant (row or column dominant) matrix has an $LU$ factorization and that pivoting is not required for numerical reliability.

**2.15.a**. Describe an algorithm that solves $Ax = b$ as efficiently as possible.

**2.15.b**. Given that the number of operations in the algorithm is of the form $Cn^k + O(n^{k-1})$, where $C$ is a constant independent of $n$ and $k > 0$, what are $C$ and $k$?

# Problem 2.16

Suppose you are computing a factorization of the $A \in \mathbb{C}^{n \times n}$ with partial pivoting and at the beginning of step $i$ of the algorithm you encounter the the transformed matrix with the form

$$TA = A^{(i-1)} = \begin{pmatrix} U_{11} & U_{12} \\ 0 & A_{i-1} \end{pmatrix}$$

where $U_{11} \in \mathbb{R}^{i-1 \times i-1}$ and nonsingular, and $U_{12} \in \mathbb{R}^{i-1 \times n-i+1}$ contain the first $i - 1$ rows of $U$. Show that if the first column of $A_{i-1}$ is all 0 then $A$ must be a singular matrix.

## Problem 2.17

It is known that if partial or complete pivoting is used to compute $PA = LU$ or $PAQ = LU$ of a nonsingular matrix then the elements of $L$ are less than 1 in magnitude, i.e., $|\lambda_{ij}| \leq 1$. Now suppose $A \in \mathbb{R}^{n \times n}$ is a symmetric positive definite matrix, i.e., $A = A^T$ and $x \neq 0 \rightarrow x^T Ax > 0$. It is known that $A$ has a factorization $A = LL^T$ where $L$ is lower triangular with positive elements on the main diagonal (the Cholesky factorization). Does this imply that $|\lambda_{ij}| \leq 1$? If so prove it and if not give an $n \times n$ symmetric positive definite matrix with $n > 3$ that is a counterexample and justify that it is indeed a counterexample.

## Problem 2.18

Suppose $PAQ = LU$ is computed via Gaussian elimination with complete pivoting. Show that there is no element in $e_i^T U$, i.e., row $i$ of $U$, whose magnitude is larger than $|\mu_{ii}| = |e_i^T U e_i|$, i.e., the magnitude of the $(i, i)$ diagonal element of $U$.