MAT 423: Homework 4 (09/19)

Ex 1. [4pts]

a) We introduce the tableau: $[A \,|\, \mathrm{Id}]$.

$$\left[\begin{array}{rrr|rrr} 2 & -1 & 1 & 1 & 0 & 0 \\ -4 & 3 & -3 & 0 & 1 & 0 \\ 2 & 0 & 3 & 0 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 2 & -1 & 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 2 & 1 & 0 \\ 0 & 1 & 2 & -1 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 2 & -1 & 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 2 & 1 & 0 \\ 0 & 0 & 3 & -3 & -1 & 1 \end{array}\right]$$

Thus, $MA = U$ with

$$U = \begin{bmatrix} 2 & -1 & 1 \\ 0 & 1 & -1 \\ 0 & 0 & 3 \end{bmatrix} \quad , \quad M = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & -1 & 1 \end{bmatrix}$$

2pt

To obtain L, we need to inverse M:

$$\left[\begin{array}{rrr|rrr} 1 & 0 & 0 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 \\ -3 & -1 & 1 & 0 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -2 & 1 & 0 \\ 0 & -1 & 1 & 3 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -2 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{array}\right]$$

giving $L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$.

1pt

b) Performing similarly operation on B gives:

$$\left[\begin{array}{rrr|rrr} 1 & 2 & -2 & 1 & 0 & 0 \\ -2 & -4 & 6 & 0 & 1 & 0 \\ 0 & 4 & 2 & 0 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 1 & 2 & -2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 2 & 1 & 0 \\ 0 & 4 & 2 & 0 & 0 & 1 \end{array}\right]$$

Unfortunately the pivot becomes zero, thus one need to perform a permutation P. Rather than exchanging row (left multiplication), we choose to exchange column (right multiplication) by multiplying by:

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

.5pt

leading to:

$$\left[\begin{array}{rrr|rrr} 1 & -2 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 2 & 1 & 0 \\ 0 & 2 & 4 & 0 & 0 & 1 \end{array}\right] \Rightarrow \left[\begin{array}{rrr|rrr} 1 & -2 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 2 & 1 & 0 \\ 0 & 0 & 4 & -2 & -1 & 1 \end{array}\right]$$

Thus we get: $MAP = U$ with

$$U = \begin{bmatrix} 1 & -2 & 2 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \quad \text{and} \quad M = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -2 & -1 & 1 \end{bmatrix}.$$

.5pt

Computing the inverse of M gives:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

Ex 2. [2pts]

a) See script below. We use that: $M_k \dots M_1 A = U$. Thus, U is the last matrix of the Gauss elimination and L would be: 1pt

$$L = (M_k \dots M_1)^{-1} = \text{Id} M_1^{-1} \dots M_K^{-1}$$

thus to get L we need to follow the same operation but by operating on the column rather than the rows.

b) The decomposition gives:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2/3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3/4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -4/5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -5/6 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -6/7 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -7/8 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -8/9 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9/10 & 1 \end{bmatrix}$$

1pt

$$U = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3/2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4/3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5/4 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6/5 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 7/6 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8/7 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9/8 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10/9 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 11/10 \end{bmatrix}$$

Ex 3. [2pts]

Consider the matrix $A = \begin{bmatrix} 2 & -3 \\ -3 & 5 \end{bmatrix}$. It is symmetric ($A^T = A$) definite positive ($a_{11} > 0$ and $\det(A) > 0$) but not diagonal dominant (first raw).

Ex 4. [2pts]

Let's first show that the matrix is symmetric (we use that $A^T = A$)

$$(MAM^T)^T = (M^T)^T A^T M^T = MAM^T. \qquad \boxed{1\text{pt}}$$

To show that it is also definite positive, we consider x a non-zero vector:

$$\langle MAM^T x, x\rangle = \langle AM^T x, M^T x\rangle = \langle Ay, y\rangle \qquad \boxed{1\text{pt}}$$

with $y = M^T x$. Since $x \neq 0$ and M non-singular, we also have $y \neq 0$. Since A is symmetric definite positive, we deduce: $\langle Ay, y\rangle > 0$. Therefore $\langle MAM^T x, x\rangle > 0$.

Ex 5. [+1pt1]

a) We first compute $M_1 A$ leading to:

$$M_1 A = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ 0 & & & \\ \vdots & & \hat{B} & \\ 0 & & & \end{bmatrix}$$

Thus, taking the transpose, we obtain:

$$AM_1^T = (M_1 A)^T = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & & & \\ \vdots & & \hat{B}^T & \\ a_{n1} & & & \end{bmatrix}$$

Applying M_1 again leads:

$$M_1 A M_1^T = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & & & \\ \vdots & & \tilde{A} & \\ 0 & & & \end{bmatrix}.$$

To show that $\tilde{A}$ is a symmetric definite positive matrix, we use **Ex. 4** since A is sym. def. pos. and M_1 non-singular.

b) By iteration, we obtain:

$$M_k \dots M_1 A M_1^T \dots M_k^T = D$$

with D diagonal matrix and the matrices M_i non-singular and lower triangular. Denote: $M = M_k \dots M_1$ and $L = M^{-1}$, we deduce:

$$MAM^T = D \quad \Rightarrow \quad A = M^{-1}D(M^T)^{-1} \quad \Rightarrow \quad A = LD(M^{-1})^T = LDL^T.$$

Since D is a diagonal matrix (i.e. $D = diag([d_{11}, \dots, d_{nn}])$), we can construct its 'square root' as follow:

$$\sqrt{D} = \begin{bmatrix} \sqrt{d_{11}} & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & & 0 & \sqrt{d_{nn}} \end{bmatrix} \qquad \text{satisfying} \quad \sqrt{D} \cdot \sqrt{D} = D.$$

Denoting $C = L\sqrt{D}$, we find:

$$CC^T = L\sqrt{D}(L\sqrt{D})^T = L\sqrt{D}\sqrt{D}^T L^T = L\sqrt{D}\sqrt{D}L^T = LDL^T = A.$$

Extra The implementation of the Cholesky decomposition is given below. The algorithm gives the following matrix: +.5pt

+.5pt

$$C = \begin{bmatrix} 1.414 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.707 & 1.225 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.816 & 1.155 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.866 & 1.118 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.894 & 1.095 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.913 & 1.08 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.926 & 1.069 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.935 & 1.061 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.943 & 1.054 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.949 & 1.049 \end{bmatrix}$$

We can also find an explicit expression:

$$C = \begin{bmatrix} \sqrt{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \sqrt{1/2} & \sqrt{3/2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{2/3} & \sqrt{4/3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{3/4} & \sqrt{5/4} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{4/5} & \sqrt{6/5} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{5/6} & \sqrt{7/6} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{6/7} & \sqrt{8/7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{7/8} & \sqrt{9/8} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{8/9} & \sqrt{10/9} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{9/10} & \sqrt{11/10} \end{bmatrix}.$$

```julia
####################################################
##        LU decomposition     (Julia)            ##
####################################################
using LinearAlgebra

function LU_decomp(A)
    #= Find the LU decomposition of the matrix A using Gauss elimination
    =#

    # init
    n = size(A)[1]
    U = copy(A)
    L = diagm(0=>ones(n))
    #  Gauss elimination    #
    #-----------------------#
    for j=1:(n-1)
        # pivot
        a_jj = U[j,j]
        for i=(j+1):n
            c = U[i,j]/a_jj
            U[i,:] .-= c*U[j,:]
            L[:,j] .+= c*L[:,i]
        end
    end

    return L,U
end
# testing
#--------
# A = randn(5,5)
# L,U = LU_decomp(A)
# A - L*U
```

```julia
#####################################################
##        Cholesky decomposition     (Julia)       ##
#####################################################
using LinearAlgebra

function Cholesky_decomp(A)
    #= Find the Cholesky decomposition of the matrix A using Gauss elimination
    =#

    # init
    n = size(A)[1]
    D = copy(A)
    L = diagm(0=>ones(n))
    #--------------------------#
    #   A) Gauss elimination   #
    #--------------------------#
    for j=1:(n-1)
        # pivot
        a_jj = D[j,j]
        for i=(j+1):n
            c = D[i,j]/a_jj
            D[i,:] .-= c*D[j,:]
            D[:,i] .-= c*D[:,j]
            L[:,j] .+= c*L[:,i]
        end
    end

    return L*sqrt.(D)
end
# testing
#--------
# n = 10
# A = diagm(0=>2*ones(n)) - diagm(1=>1*ones(n-1))- diagm(-1=>1*ones(n-1))
# C = Cholesky_decomp(A)
# A - C*C'
```