Math 104B HW 5

Qingze Lan 3046380

In [1]:
```
from numpy import array, zeros, dot, tril, triu
```

Problem 1

(a) Implementing Gaussian elimination with partial pivoting

In [5]:
```
# Echelon form and multipliers
# INPUT: pa (nxn matrix), b (n-vector)
# OUTPUT: U (echelon form of PA)
# PA (A with swapped rows)
# Pb (b with swapped rows)
# Qingze Lan
# 02/22/2021


eps = 2**(-52)

def LU_form(A,b):
    Ab = zeros((A[0].size, A[0].size + 1))
    l_index = []
    Ab[:,:-1] = A
    Ab[:,-1] = b
    Ab_bis = Ab.copy()
    for i in range(0, len(A) - 1):
        a = max(abs(A[i:,i]))
        if abs(a) < eps:
            print('ERROR: Singular matrix')
            return
        ix = list(abs(A[:,i])).index(a)
        l_index.append((i,ix))
        Ab[i], Ab[ix] = Ab[ix].copy(), Ab[i].copy()
        for j in range(i + 1, len(A)):
            lm = Ab[j,i]/Ab[i,i]
            Ab[j] -= Ab[i]*lm
            Ab[j,i] = lm # storing multipliers same matrix
    if Ab[-1,-2] == 0: # cannot be zero!
            print('ERROR: Singular matrix')
            return
    for k in l_index: # obtaining PA
        Ab_bis[k[0]], Ab_bis[k[1]] = Ab_bis[k[1]].copy(), Ab_bis[k[
0]].copy()
    return Ab, Ab_bis

# Solving upper diagonal system Ux = y
```

```python
# INPUT: U (linear system matrix, upper diagonal)
# y (column vector)
# OUTPUT: x (unknowns column vector)
# Qingze Lan
# 02/22/2021

def linear_u_solver(U,y):
    x = zeros( len(y) )
    for i in range(len(y) - 1,-1,-1):
        coef = [U[i,j]*x[j] for j in range(len(y) - 1,i,-1)]
        x[i] = ( y[i] - sum(coef) )/U[i,i]
    return x

# Solving linear system, Gaussian elimination, LU
# INPUT: A (nxn matrix), b (n-vector)
# OUTPUT: PA (system matrix, not expanded, swapped rows)
# x (solution to PAx = b)
# L (lower triagonal matrix)
# U (upper triagonal matrix)
# Qingze Lan
# 02/22/2021

def gaussian_LU(A,b):
    LU, PA = LU_form(A,b)
    L = tril(LU[:,:-1])
    U = triu(LU[:,:-1])
    for i in range(0,len(L)):
        L[i,i] = 1
    return PA[:,:-1], linear_u_solver(U,LU[:,-1]), L, U
```

(b) Solving $Ax = b$ for

$$A = \begin{bmatrix} 5 & 1 & 0 & 2 & 1 \\ 0 & 4 & 0 & 1 & 2 \\ 1 & 1 & 4 & 1 & 1 \\ 0 & 1 & 2 & 6 & 0 \\ 0 & 0 & 1 & 2 & 4 \end{bmatrix} \qquad b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$$

In [6]:

```
a_b = array([[5.,1.,0.,2.,1.],
             [0.,4.,0.,1.,2.],
             [1.,1.,4.,1.,1.],
             [0.,1.,2.,6.,0.],
             [0.,0.,1.,2.,4.]])

b_b = array([1.,2.,3.,4.,5.])

pa_b, x_b, L_b, U_b = gaussian_LU(a_b,b_b)

print('Solving system Ax = b with A:')
print(a_b)

print('\nMatrix with swapped rows')
print(pa_b)

print('\nSolution to the system PAx = Pb')
print(x_b)
```

```
Solving system Ax = b with A:
[[5. 1. 0. 2. 1.]
 [0. 4. 0. 1. 2.]
 [1. 1. 4. 1. 1.]
 [0. 1. 2. 6. 0.]
 [0. 0. 1. 2. 4.]]

Matrix with swapped rows
[[5. 1. 0. 2. 1.]
 [0. 4. 0. 1. 2.]
 [1. 1. 4. 1. 1.]
 [0. 1. 2. 6. 0.]
 [0. 0. 1. 2. 4.]]

Solution to the system PAx = Pb
[-0.17083787 -0.06746464  0.46028292  0.52448313  0.8726877 ]
```

(c) Testing code for

$$A = \begin{bmatrix} 5 & 1 & 0 & 2 \\ 0 & 4 & 0 & 8 \\ 1 & 1 & 4 & 2 \\ 0 & 1 & 2 & 2 \end{bmatrix} \qquad b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

In [7]:

```
a_c = array([[5.,1.,0.,2.],
             [0.,4.,0.,8.],
             [1.,1.,4.,2.],
             [0.,1.,2.,2.]])

b_c = array([1.,2.,3.,4.])

LU_form(a_c,b_c)
```

```
ERROR: Singular matrix
```

Problem 2

(a) Let A be an $n \times n$ upper of lower triagonal matrix. If we denote by $A_1, \ldots, A_n$ the principal submatrices of A we can expand the determinant, starting from the last row/column (depending on whether is upper/lower triangular) so that:

$$\det(A) = a_{nn} \det(A_{n-1}) = a_{nn} a_{nn-1} \det(A_{n-2}) = \ldots = a_{nn} a_{nn-1} \ldots a_{22} \det(A_1) = a_{11} a_{22} \ldots a_{nn}$$

(b) We'll prove that the product of pivots in the Gaussian Elimination for $Ax = b$ is equal to the determinant of A up to a sign.

After the Gaussian Elimination process we obtain a factorization of the matrix A according: $PA = LU$,where the matrix P is a $n \times n$ identity matrix that might have some swapped rows.

Hence $\det(PA) = \pm \det(A)$.

L is a lower triagonal matrix with 1's in the diagonal. By (a) we have $\det(L) = 1$.

On the other hand U is the matrix we obtain when transforming A into an upper triangular matrix. This will have all the pivots in the diagonal and by (a) again:

$$\det(A) = \pm \det(P) = \pm p_1 \ldots p_n$$

where $p_1, \ldots, p_n$ are all the pivots.

(c) We'll prove that the product of two $n \times n$ upper (lower) triangular matrix is also an upper (lower) triangular matrix.

Let's assume that A, B are two $n \times n$ upper triangular matrices. That means that $a_{ij}, b_{ij} = 0$ for $i > j$, this is for the elements under the diagonal.

If $AB = C = (c_{ij})$ then assuming $l > m$:

$$c_{lm} = \sum_{k=1}^{n} a_{ik} b_{kj} = \sum_{k=1}^{m} a_{lk} b_{km} + \sum_{k=m+1}^{n} a_{lk} b_{km} = 0$$

because in the first sum the terms $a_{lk} = 0$ and in the second sum the terms $b_{km} = 0$. Then C is also upper triangular.

If both A, B are lower triangular then $a_{ij}, b_{ij} = 0$ for $i < j$. Assuming $l < m$ one finds:

$$c_{lm} = \sum_{k=1}^{n} a_{ik} b_{kj} = \sum_{k=1}^{m-1} a_{lk} b_{km} + \sum_{k=m}^{n} a_{lk} b_{km} = 0$$

because in the first sum the terms $b_{lk} = 0$ and in the second sum the terms $a_{km} = 0$. Then C is lower triangular.

(d) If L_i is a lower triangular matrix with the $n-i$ multipliers -negative sign- produced in the ith step of the Gaussian Elimination stored in its ith column, we'll prove that L_i^{-1} is obtained by changing the sign of the multipliers. We'll call this matrix A and we'll compute its inverse using Gaussian elimination.

$$A = \begin{bmatrix} 1 & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & 1 & \dots & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & & & & \vdots \\ \vdots & \vdots & & 1 & & & \vdots \\ \vdots & \vdots & & -m_{i+1,i} & \ddots & & \vdots \\ \vdots & \vdots & & -m_{i+2,i} & & & \vdots \\ 0 & 0 & \dots & \vdots & \dots & \ddots & 0 \\ 0 & 0 & \dots & -m_{n,i} & \dots & & 1 \end{bmatrix}$$

Now expanding the matrix and writing as blocks $Ab = (A|I_n)$ where I_n is a $n \times n$ identity matrix, we can apply the row transformations:

$$R_k \rightarrow R_k - a_{ki} R_i, \quad \forall k > i$$
$$Ab \longrightarrow (I_n|B)$$

The elements a_{ki} are precisely the multpliers with a negative sign and hence the matrix B will be:

$$B = \begin{bmatrix} 1 & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & 1 & \dots & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & & & & \vdots \\ \vdots & \vdots & & 1 & & & \vdots \\ \vdots & \vdots & & m_{i+1,i} & \ddots & & \vdots \\ \vdots & \vdots & & m_{i+2,i} & & & \vdots \\ 0 & 0 & \dots & \vdots & \dots & \ddots & 0 \\ 0 & 0 & \dots & m_{n,i} & \dots & & 1 \end{bmatrix}$$

Obviously this will be the inverse since all row transformations can be written as a product of matrices:

$$Q(A|I_n) = (I_n|B) \implies QA = I_n, \quad Q = B$$

And hence $B = A^{-1}$.

Problem 3

$$A = \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 1 & 1 & 4 \end{bmatrix}$$

The factorization exists because the principal submatrices are nonsingular:

$$\det(5) = 5, \qquad \det \begin{bmatrix} 5 & 1 \\ 0 & 4 \end{bmatrix} = 20 \qquad \det \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 1 & 1 & 4 \end{bmatrix} = 4 \cdot 20$$

Using row transformations:

- First $R_3 \to R_3 - R_1/5$
- Second $R_3 \to R_3 - R_2/5$

$$\begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 1 & 1 & 4 \end{bmatrix} \to \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 0 & 4/5 & 4 \end{bmatrix} \to \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} = U$$

And we can find L by storing the multipliers in a lower triangular matrix:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1/5 & 1/5 & 1 \end{bmatrix}$$

Hence,

$$LU = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1/5 & 1/5 & 1 \end{bmatrix} \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} = \begin{bmatrix} 5 & 1 & 0 \\ 0 & 4 & 0 \\ 1 & 1 & 4 \end{bmatrix}$$

Problem 4

$$A = \begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$

The factorization exists since A is symmetric and positive definite. This follows from:

$$\det(3) > 0, \qquad \det\begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} > 0, \qquad \det\begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix} > 0$$

Now writing

$$L = \begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

with positive diagonal entries and imposing $LL^t = A$,

$$\begin{aligned} a_{11} &= \sqrt{3} \\ a_{21} &= -\frac{1}{\sqrt{3}} \\ a_{31} &= 0 \\ a_{22} &= \sqrt{3 - a_{11}^{-2}} = 2\sqrt{\frac{2}{3}} \\ a_{23} &= -\sqrt{\frac{3}{8}} \\ a_{33} &= \sqrt{\frac{21}{8}} \end{aligned}$$

and hence:

$$L = \begin{bmatrix} \sqrt{3} & 0 & 0 \\ -\frac{1}{\sqrt{3}} & 2\sqrt{\frac{2}{3}} & 0 \\ 0 & -\frac{1}{2}\sqrt{\frac{3}{2}} & \frac{1}{2}\sqrt{\frac{21}{2}} \end{bmatrix}$$

We can verify that $A = LL^t$:

$$\begin{bmatrix} \sqrt{3} & 0 & 0 \\ \frac{-1}{\sqrt{3}} & 2\sqrt{\frac{2}{3}} & 0 \\ 0 & -\frac{1}{2}\sqrt{\frac{3}{2}} & \frac{1}{2}\sqrt{\frac{21}{2}} \end{bmatrix} \begin{bmatrix} \sqrt{3} & \frac{-1}{\sqrt{3}} & 0 \\ 0 & 2\sqrt{\frac{2}{3}} & -\frac{1}{2}\sqrt{\frac{3}{2}} \\ 0 & 0 & \frac{1}{2}\sqrt{\frac{21}{2}} \end{bmatrix} = \begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$