111-2 Numerical Analysis
Homework 4

Due Time: 22:00 , Tuesday, 3/21, 2023.

Instructor: Min-Hui Lo

- **Regulation**

  1. **NO PLAGIARISM and NO LATE ASSIGNMENTS.**

- **Submission**

  1. Please write down your answers (including discussions and figures) in the same order as the problem sheet in the word/pdf file.
  2. You should upload zip file,including code and pdf (or word) file via NTU COOL.
  3. zip file name: "*hw{hw number}_g{group id}.zip" (e.g. hw01_g01.zip)*

---

1. **Backward substitution**

    1. Please write a function **BackwardSub(A, AX)** regarding linear equations to solve X, given A and AX. Try to find the RULE from the below code.

        Hint

        Try to write the function as general form, i.e., you should get sizes of matrix in the first step.

        ```
        # backward substitution
        X[3] = AX[3]/A[3,3]
        X[2] = (AX[2]-A[2,3]*X[3])/A[2,2]
        X[1] = (AX[1]-A[1,3]*X[3]-A[1,2]*X[2])/A[1,1]
        X[0] = (AX[0]-A[0,3]*X[3]-A[0,2]*X[2]-A[0,1]*X[1])/A[0,0]
        ```

    2. Use BackwardSub(A, AX) to solve the following equations.

$$\begin{bmatrix} 1 & 2 & 2 & 1 \\ 0 & -8 & -3 & -7 \\ 0 & 0 & -4.8750 & -3.3750 \\ 0 & 0 & 0 & -5.5385 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 17 \\ -26 \\ -14.25 \\ 16.6155 \end{bmatrix}$$

    3. Use BackwardSub(A, AX) to solve the following equations.

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & -7 & -2 & -10 & 9 \\ 0 & 0 & 1 & 5 & 4 \\ 0 & 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 23 \\ 63 \\ 6 \\ 10 \\ 8 \end{bmatrix}$$

2. **Gauss elimination method and backward substitution**

   1. Please write a function **Gauss_elim(A, AX)** to solve X, given A and AX. Try to find the RULE from the below code.

      Hint

      The equation below is just an example. Please write a function in a general form.

```
import numpy as np
A = np.array([[1, 2, 2, 1], [2, -4, 1, -5], [2, 1, -2, -4], [-1, 2, 1, -2]]
, dtype = float)
AX = np.array([17, 8, 10, 17], dtype = float)

# forward elimination
for i in range(1,4):
AX[i] = AX[i] - AX[0]*(A[i,0]/A[0,0])
A[i,0:4] = A[i,0:4] - A[0,0:4]*(A[i,0]/A[0,0])

for i in range(2,4):
AX[i] = AX[i] - AX[1]*(A[i,1]/A[1,1])
A[i,1:4] = A[i,1:4] - A[1,1:4]*(A[i,1]/A[1,1])

for i in range(3,4):
AX[i] = AX[i] - AX[2]*(A[i,2]/A[2,2])
A[i,2:4] = A[i,2:4] - A[2,2:4]*(A[i,2]/A[2,2])

# backward substitution
X = np.zeros(4)
X[3] = AX[3]/A[3,3]
X[2] = (AX[2]-A[2,3]*X[3])/A[2,2]
X[1] = (AX[1]-A[1,3]*X[3]-A[1,2]*X[2])/A[1,1]
X[0] = (AX[0]-A[0,3]*X[3]-A[0,2]*X[2]-A[0,1]*X[1])/A[0,0]
```

   2. Use Gauss_elim(A, AX) to solve the following equations.

$$\begin{cases} \mathrm{x}_1 + 2x_2 + 2x_3 + x_4 + x_5 = 17 \\ 2\mathrm{x}_1 - 4x_2 + x_3 - 5x_4 = 8 \\ 2\mathrm{x}_1 + x_2 - 2x_3 - 4x_4 = 10 \\ \text{-}\mathrm{x}_1 + 2x_2 + x_3 - 2x_4 = 17 \end{cases}$$

3. Use Gauss_elim(A, AX) to solve the following equations.

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 = 23 \\ 2x_1 - 3x_2 + 4x_3 - 2x_4 + x_5 = 37 \\ 3x_1 - 4x_2 + x_3 - x_4 + 3x_5 = 30 \\ 4x_1 + x_2 + 2x_3 + 2x_4 + 3x_5 = 23 \\ 5x_1 + 5x_2 - 3x_3 + x_4 + 4x_5 = 3 \end{cases}$$

3. **Modify your function Gauss_elim(A, AX) to solve the "0" in below question**

1. Solve

$$\begin{cases} 0x_1 + 2x_2 + 3x_3 = 46 \\ 4x_1 - 3x_2 + 2x_3 = 16 \\ 2x_1 + 4x_2 - 3x_3 = 12 \end{cases}$$

2. Solve

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 = 23 \\ 2x_1 + 4x_2 + 4x_3 - 2x_4 + x_5 = 30 \\ 3x_1 - 4x_2 + x_3 - x_4 + 3x_5 = 30 \\ 4x_1 + x_2 + 2x_3 + 2x_4 + 3x_5 = 23 \\ 5x_1 + 5x_2 - 3x_3 + x_4 + 4x_5 = 3 \end{cases}$$