César Isaí García Cornejo
cesar.cornejo@cimat.mx
Computo Científico

Tarea 1

Semestre 3

CIMAT

Ejercicio 1
Implementar los algoritmos de *Backward* y *Forward substitution*.

Solución:

Para poder obtener un algoritmo que ejecute una **substitución *forward*** es necesario un sistema de ecuaciones lineales cuya matriz es triangular inferior. Es decir, el sistema se puede expresar de la forma

$$Ly = b \tag{1}$$

donde L es una matriz triangular inferior. Otra forma de verlo es por medio de la expansión

$$\begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{m1} & l_{m2} & l_{m3} & \cdots & l_{mm} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_m \end{pmatrix}$$

Así tenemos,

$$\begin{aligned} y_1 &= \frac{b_1}{l_{11}}, \\ y_2 &= \frac{b_2 - l_{21}y_2}{l_{22}}, \\ &\vdots \\ y_j &= \frac{(b_j - l_{j1}y_1 - l_{j2}y_2 - \cdots - l_{j,j-1}y_{j-1})}{l_{jj}}, \qquad 2 \leq j \leq m. \end{aligned} \tag{2}$$

Para la implementación de dicho algoritmo es necesario anidar dos ciclos for. Se crea una variable auxiliar llamada aux, la cual el ciclo for interno suma los terminos $l_{j1}y_1 + l_{j2}y_2 + \cdots + l_{j,j-1}y_{j-1}$ para posteriormente usar (2)

```
for i in range(len(b)):
    aux = 0
    for j in range(i):
        aux += L[i,j]*y[j]    #Auxiliar variable
    y[i] = (b[i] - aux)/L[i,i]    # 'y' is the output and solution to the system of
        equations.
```

Por otro lado, la implementación de la **substitución *backward*** es análogo al algoritmo previo. Consideremos el sistema de ecuaciones lineales

$$Ux = b \tag{3}$$

donde U es una matriz triangular superior. La expansión es de la forma

$$\begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & u_{m-2,m-2} & u_{m-2,m-1} & u_{m-2,m} \\ 0 & 0 & 0 & u_{m-1,m-1} & u_{m-1,m} \\ 0 & 0 & 0 & 0 & u_{mm} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{m-2} \\ x_{m-1} \\ x_m \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_{m-2} \\ b_{m-1} \\ b_m \end{pmatrix}$$

De esta forma

$$x_{mm} = \frac{b_m}{u_{mm}}, \tag{4}$$

$$x_{m-1,m-1} = \frac{b_{m-1} - u_{m-1,m}x_m}{u_{m-1,m-1}}, \tag{5}$$

$$\tag{6}$$

Ejercicio 2
Implementar el algoritmo de eliminación Gaussiana con pivoteo parcial LUP, 21.1 del Trefethen (p. 160).

Ejercicio 3
Dar la descomposición LUP para una matriz aleatoria de entradas U(0,1) de tamaño 5×5, y para la matriz

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 & 1 \\ -1 & -1 & 1 & 0 & 1 \\ -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 \end{pmatrix} \tag{7}$$

Ejercicio 4
Usando la descomposición LUP anterior, resolver el sistema de la forma

$$Dx = b \tag{8}$$

donde D son las matrices del problema 3, para 5 diferentes b aleatorios con entradas $U(0,1)$. Verificando si es posible o no resolver el sistema.

Ejercicio 5
Implementar el algoritmo de descomposición de Cholesky 23.1 del Trefethen (p. 175).

Ejercicio 6
Comparar la complejidad de su implementación de los algoritmos de factorización de Cholesky y LUP mediante la medición de los tiempos que tardan con respecto a la descomposición de una matriz aleatoria hermitiana definida positiva. Graficar la comparación.