Métodos Numéricos
Práctica 4: Resolución de sistemas de ecuaciones lineales. Métodos directos. Factorización de matrices

3. Suponga que queremos resolver los siguientes tres sistemas de ecuaciones:

$$\text{S1}\ \begin{cases} x_1 + 2x_2 + 3x_3 = 14 \\ 3x_1 - 2x_2 + x_3 = 2 \\ 4x_1 + 2x_2 - x_3 = 5 \end{cases}$$

$$\text{S2}\ \begin{cases} x_1 + 2x_2 + 3x_3 = 9 \\ 3x_1 - 2x_2 + x_3 = -5 \\ 4x_1 + 2x_2 - x_3 = 19 \end{cases}$$

$$\text{S3}\ \begin{cases} x_1 + 2x_2 + 3x_3 = -2 \\ 3x_1 - 2x_2 + x_3 = 2 \\ 4x_1 + 2x_2 - x_3 = 12 \end{cases}$$

Podemos escribir los tres sistemas de ecuaciones como una única ecuación matricial: $\mathsf{AX} = \mathsf{B}$, donde

$$\mathsf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & -2 & 1 \\ 4 & 2 & -1 \end{pmatrix}, \quad \mathsf{X} = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}, \quad \mathsf{B} = \begin{pmatrix} 14 & 9 & -2 \\ 2 & -5 & 2 \\ 5 & 19 & 12 \end{pmatrix}.$$

(a) Modificar el método de eliminación Gaussiana a fin de resolver múltiples sistemas de ecuaciones lineales. La matriz aumentada para este caso es $\mathsf{A}_{aum} = [\mathsf{A}\ \mathsf{B}]$, donde A es una matriz de $n \times n$, y B es una matriz de $n \times m$. La solución es una matriz de $n \times m$.

(b) Utilizar el método programado en el item anterior para resolver los tres sistemas lineales dados.

(c) Utilizar el método programado en el item (a) para calcular la inversa de la matriz A por eliminación Gaussiana.

<u>Solución:</u>

(a) Modificamos el método de eliminación Gaussiana de la siguiente manera

function [X,A] = GAUSSELIMSIM(A,B)
    $[nA, mA] = size(A);$
    $[nB, mB] = size(B);$
    if nA<>mA **then**

error('gausselim - La matriz A debe ser cuadrada');
abort;
else if nA<>nB **then**
error('gausselim - dimensiones incompatibles entre A y B');
abort;
end if;
$a = [AB]$; // Matriz aumentada
for i = 1:(nA-1) **do** // Eliminación progresiva
for j = (i+1):nA **do**
mji = a(j,i)/a(i,i); //multiplicador de fila
a(j,i)=0; //hacemos cero los elementos debajo de a_{ii}
a(j,(i+1):(mA+mB)) = a(j,(i+1):(mA+mB)) - mji*a(i,(i+1):(mA+mB));
//Cambiamos la fila j por la diferencia entre la fila j y la fila i multiplicada por mjk. $mA + mB$ es la cantidad de columnas de la matriz aumentada $[AB]$, y nA la cantidad de filas de la misma
end for
end for
// Sustitución regresiva
$X(nA, 1 : mB) = a(nA, (nA + 1) : (nA + mB))/a(nA, nA)$; //Calculamos la fila mB de X despejando de la ecuación mB-ésima de AX=B
for i = (nA-1):-1:1 **do**
$X(i, 1 : mB) = (a(i, (mA+1) : (mA+mB)) - (a(i, (i+1) : mA) * X((i+1) : mA, 1 : mB)))./a(i, i)$; //Calculamos la fila i de X despejando de la ecuación i-ésima de AX=B
end for
end function

(b) Utilizamos el método programado en el ítem anterior para resolver los tres sistemas lineales dados. En Scilab obtenemos lo siguiente:

```
--> A = [1 2 3; 3 -2 1; 4 2 -1]
 A  =

   1.   2.   3.
   3.  -2.   1.
   4.   2.  -1.
```

```
--> B = [14 9 -2; 2 -5 2; 5 19 12]
 B  =

   14.    9.    -2.
   2.    -5.     2.
   5.     19.    12.


--> [X,a] = gausselimsim(A,B)
 a  =

   1.    2.    3.    14.    9.    -2.
   0.   -8.   -8.   -40.   -32.    8.
   0.    0.   -7.   -21.    7.     14.

 X  =

   1.    2.    2.
   2.    5.    1.
   3.   -1.   -2.
```

Luego, las soluciones a los sistemas S1, S2 y S3 son

$$\text{S1: } x_1 = 1, x_2 = 2, x_3 = 3,$$

$$\text{S2: } x_1 = 2, x_2 = 5, x_3 = -1,$$

$$\text{S3: } x_1 = 2, x_2 = 1, x_3 = -2.$$

(c) Queremos calcular la inversa de A, A^{-1}, por eliminación Gaussiana. Por definición de matriz inversa, $AA^{-1} = A^{-1}A = I$, con I la matriz identidad. Luego, si en el algoritmo del apartado (a) consideramos $B = I$, la solución de $AX = B$ será $X = A^{-1}$. En Scilab obtenemos

```
--> I=eye(3,3)
 I  =

   1.    0.    0.
   0.    1.    0.
   0.    0.    1.
```

```
--> [X,a] = gausselimsim(A,I)
 a  =

   1.    2.    3.    1.      0.      0.
   0.   -8.   -8.   -3.      1.      0.
   0.    0.   -7.   -1.75   -0.75    1.

 X  =

   0.        0.1428571    0.1428571
   0.125   -0.2321429    0.1428571
   0.25     0.1071429   -0.1428571


--> A*X
 ans  =

   1.    0.           0.
   0.    1.           0.
   0.    1.804D-16    1.
```

6. Construir un algoritmo para resolver por el método de Gauss el problema $\mathsf{A}\mathbf{x} = \mathbf{b}$, siendo A una matriz tridiagonal:

$$\mathsf{A} = \begin{pmatrix} b_1 & c_1 & 0 & \cdots & \cdots & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & 0 \\ 0 & a_3 & \cdots & \cdots & & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & \cdots & & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & 0 & \cdots & 0 & a_n & b_n \end{pmatrix}$$

Contar la cantidad de operaciones que se realizan en este caso.

<u>Solución:</u>

Observemos que para realizar la eliminación Gaussiana, en el paso k tenemos 2 posibles pivotes, a_{kk} y $a_{(k+1)k}$, por lo que solo debemos comparar dichos valores para el pivoteo. Por otro lado, $a_{(k+1)k}$ es el único posible valor distinto de cero debajo de la diagonal en la columna k de la matriz, por lo que en el paso k tenemos una única operación de filas. Luego, para matrices tridiagonales, podemos modificar el algoritmo de eliminación Gaussiana con pivoteo parcial para obtener el siguiente algoritmo:

function [X,A] = GAUSSELIMPPTRI(A,b)
    [nA,mA] = size(A)
    [nb,mb] = size(b)

```
    if nA<>mA then then
        error('gausselim - La matriz A debe ser cuadrada');
        abort;
    else if mA<>nb then then
        error('gausselim - dimensiones incompatibles entre A y b');
        abort;
    end if;
    a = [A b]; // Matriz aumentada
    n = nA; // Tamaño de la matriz
    contador=0;
     // Eliminación progresiva con pivoteo parcial
    for k=1:n-1 do
        kpivot = k; amax = abs(a(k,k)); //pivoteo
        if abs(a(k+1,k))>amax then then
            temp = a(k+1,:); a(k+1,:) = a(k,:); a(k,:) = temp; //Intercambiamos las
filas k y k+1
        end if;
        for j=k+1:n+1 do
            a(k+1,j) = a(k+1,j) - a(k,j)*a(k+1,k)/a(k,k);
            contador=contador+3;
        end for;
    end for;
     // Sustitución regresiva
    x(n) = a(n,n+1)/a(n,n);
    x(n-1) = (a(n-1,n+1)-a(n-1,n)*x(n))/a(n-1,n-1);
    contador=contador+4;
    for i = n-2:-1:1 do
        x(i) = (a(i,n+1)-a(i,i+1)*x(i+1)-a(i,i+2)*x(i+2))/a(i,i);;
        contador=contador+5;
    end for
    disp(contador);
end function
```

Ejecutamos algunos casos de prueba:

```
--> A=[1 1 0 0;1 1 1 0;0 1 1 1;0 0 1 1]
 A  =

   1.   1.   0.   0.
   1.   1.   1.   0.
   0.   1.   1.   1.
   0.   0.   1.   1.


--> b=[2;3;3;2]
 b  =

   2.
   3.
   3.
   2.


--> x=gausselimPPTRI(A,b)

   37.
 x  =

   1.
   1.
   1.
   1.
```

```
--> A=[2 2 0 0;5 2 2 0;0 5 2 2;0 0 5 2]
 A  =

   2.   2.   0.   0.
   5.   2.   2.   0.
   0.   5.   2.   2.
   0.   0.   5.   2.


--> b=[6;15;24;23]
 b  =

   6.
   15.
   24.
   23.


--> x=gausselimPPTRI(A,b)

   37.
 x  =

   1.
   2.
   3.
   4.
```

Consideremos ahora una matriz A cuya eliminación Gaussiana no requiere pivoteo (por ejemplo para A estrictamente diagonal dominante o A simétrica y definida positiva). En este caso, el paso k de la eliminación Gaussiana consiste en restar a la fila k un múltiplo de la fila $k-1$ formando un cero en la posición $a_(k+1)k$. Debido a los ceros presentes en ambas filas, basta con calcular el nuevo valor para la posición $a_{(k+1)(k+1)}$. Luego, tenemos el siguiente algoritmo:

function [X,A] = GAUSSELIMTRI(A,b)
  [nA,mA] = size(A)
  [nb,mb] = size(b)
  if nA<>mA **then** then
    error('gausselim - La matriz A debe ser cuadrada');
    abort;
  else if mA<>nb **then** then
    error('gausselim - dimensiones incompatibles entre A y b');
    abort;
  end if;
  a = [A b]; // Matriz aumentada
  n = nA; // Tamaño de la matriz
  contador=0;
  // Eliminación progresiva sin pivoteo

```
    for k=1:n-1 do
        mkk=a(k+1,k)/a(k,k);
        a(k+1,k+1)=a(k+1,k+1)-mkk*a(k,k+1);
        a(k+1,n+1)=a(k+1,n+1)-mkk*a(k,n+1);
        contador=contador+5;
    end for;
     // Sustitución regresiva
    x(n)=a(n,n+1)/a(n,n);
    contador=contador+1;
    for i = n-1:-1:1 do
        x(k)=(a(k,n+1)-a(k,k+1)*x(k+1))/a(k,k);
        contador=contador+3;
    end for
    disp(contador);
end function
```

Ejecutamos algunos casos de prueba:

```
--> A=[2 1 0 0;1 2 1 0;0 1 2 1;0 0 1 2]
 A  =

   2.   1.   0.   0.
   1.   2.   1.   0.
   0.   1.   2.   1.
   0.   0.   1.   2.


--> b=[1;1;1;1]
 b  =

   1.
   1.
   1.
   1.


--> x=gausselimTRI(A,b)

   22.
 x  =

   0.4
   0.2
   0.2
   0.4
```

```
--> A=[1 2 0 0;5 1 2 0;0 5 1 2;0 0 5 1]
 A  =

   1.   2.   0.   0.
   5.   1.   2.   0.
   0.   5.   1.   2.
   0.   0.   5.   1.


--> b=[4;11;11;3]
 b  =

   4.
   11.
   11.
   3.


--> x=gausselimTRI(A,b)

   22.
 x  =

   2.
   1.
   0.
   3.
```

9. Considerar el sistema $\mathsf{A}\mathbf{x} = \mathbf{b}$ donde

$$\mathsf{A} = \begin{pmatrix} 1 & 2 & -2 & 1 \\ 4 & 5 & -7 & 6 \\ 5 & 25 & -15 & -3 \\ 6 & -12 & -6 & 22 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 1 \\ 2 \\ 0 \\ 1 \end{pmatrix}$$

(a) Resuelva el sistema mediante el método de eliminación de Gauss. En caso de realizar intercambios de ecuaciones, calcule la matriz de permutación P correspondiente.

(b) A partir de la información obtenida en la eliminación de Gauss, obtenga la factorización $\mathsf{PA} = \mathsf{LU}$, y utilize dicha factorización para resolver el sitema $\mathsf{A}\mathbf{x} = \tilde{\mathbf{b}}$, donde $\tilde{\mathbf{b}} = [\,2 \;\; 2 \;\; 1 \;\; 0\,]^T$.

Solución:

(a) Aplicando eliminación de Gauss a $[A\,b]$ obtenemos en el tercer paso la siguiente matriz aumentada:

$$\begin{bmatrix} 1 & 2 & -2 & 1 & \vdots & 1 \\ 0 & -3 & 1 & 2 & \vdots & -3 \\ 0 & 0 & 0 & 2 & \vdots & -19 \\ 0 & 0 & -2 & 0 & \vdots & 19 \end{bmatrix},$$

por lo que debemos intercambiar las filas 3 y 4. Luego, la matriz P de permutación es

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

Consideremos el sistema $PAx = Pb$, cuya solución coincide con la solución de $Ax = b$. Al aplicar eliminación de Gauss a $[PA\,Pb]$, obtenemos la descomposición LU donde:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 4 & 1 & 0 & 0 \\ 6 & 8 & 1 & 0 \\ 5 & -5 & 0 & 1 \end{bmatrix}, \qquad U = \begin{bmatrix} 1 & 2 & -2 & 1 \\ 0 & -3 & 1 & 2 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}.$$

Entonces:

$$PAx = Pb \Leftrightarrow LUx = Pb = \begin{bmatrix} 1 \\ 2 \\ 1 \\ 0 \end{bmatrix}.$$

Definiendo $y = Ux$ tenemos que resolver $LUx = Pb$ es equivalente a resolver $Ly = Pb$. Este último sistema puede resolverse por sustitución progresiva, obteniendo que

$$y = \begin{bmatrix} 1 \\ -2 \\ 11 \\ -15 \end{bmatrix}.$$

Luego, para calcular x basta con considerar el sistema $Ux = y$, el cual se resuelve por sustitución regresiva, obteniendo que

$$x = \begin{bmatrix} 59/6 \\ -37/6 \\ -11/2 \\ -15/2 \end{bmatrix}.$$

(b) De manera análoga a lo realizado en el apartado anterior, tenemos que $PAx = LUx = P\tilde{b}$, y definiendo $y = Ux$, resulta el sistema $Ly = P\tilde{b}$ con $P\tilde{b} = [2\,2\,0\,1]$. Resolviendo por sustitución progresiva:

$$y = \begin{bmatrix} 2 \\ -6 \\ 36 \\ -39 \end{bmatrix}.$$

Luego, resolviendo $Ux = y$ por sustitución regresiva:

$$x = \begin{bmatrix} 39/2 \\ -17 \\ -18 \\ -19/2 \end{bmatrix}.$$