1ª Practica: Sistemas Lineales (Métodos directos)

Ejercicio 1º.

$$2.1754\,x1 + 4.0231\,x2 - 2.1732\,x3 + 5.1967\,x4 == 17.108$$
$$-4.0231\,x1 + 6\,x2 + 1.1973\,x4 == -6.1593$$
$$-x1 - 5.2107\,x2 + 1.1111\,x3 == 3.0004$$
$$6.0235\,x1 + 7\,x2 - 4.1561\,x4 == 0$$

Resolver el sistema A.x=b utilizando los procedimientos de *Mathematica* :

- Solve

- LinearSolve

- Mediante las funciones preparadas de eliminación gaussiana (GaussElim []) y de pivotaje parcial escalado (PivParEsc[])

- LUDecomposition

1º.-

```
A = {{2.1754, 4.0231, -2.1732, 5.1967}, {-4.0231, 6, 0, 1.1973},
   {-1, -5.2107, 1.1111, 0}, {6.0235, 7, 0, -4.1561}};
```

```
b = {17.108, -6.1593, 3.0004, 0};
```

```
x = {x1, x2, x3, x4};
```

```
A.x == b
```

```
{2.1754 x1 + 4.0231 x2 - 2.1732 x3 + 5.1967 x4,
  -4.0231 x1 + 6 x2 + 1.1973 x4, -x1 - 5.2107 x2 + 1.1111 x3,
  6.0235 x1 + 7 x2 - 4.1561 x4} == {17.108, -6.1593, 3.0004, 0}
```

```
Solve[A.x == b]
```

```
{{x1 → 2.94063, x2 → 0.0708992, x3 → 5.67947, x4 → 4.38131}}
```

2º.- El comando `LinearSolve` **sirve para sistemas en que la matriz de coeficientes es tanto numérica como simbólica. La matriz puede ser cuadrada o rectangular. en el caso de sistemas indeterminados dara como resultado una de las posibles soluciones mientras que solve da la solución general.**
La salida que se obtiene también es distinta.
Cuando es necesario resolver el mismo sistema con distintos vectores de términos independientes es mucho más eficiente utilizar el comandos LUDecomposition.

```
sol = LinearSolve[A, b]
```

```
{2.94063, 0.0708992, 5.67947, 4.38131}
```

```
A.sol - b
```

$\{0., 2.66454 \times 10^{-15}, 1.77636 \times 10^{-15}, 2.68882 \times 10^{-16}\}$

3º.- Se utilizan dos funciones creadas por nosotros para ilustrar el procedimiento de resolución de istemas de ecuaciones lineales mediante la eliminación gaussiana y mediante el pivotaje parcial escalado.
Se ejecutan las funciones para que mathematica las identifique y sepa como operar y a continuación se aplican a las matrices de coeficientes y termino independiente que se desee.

```
GaussElim [AA_, bb_] := Module[{x, c, m, n, p, i, j, k, A, NoUniqueSoln},
  A = Transpose[Append[Transpose[AA], b]];
  n = Length[A[[All, 1]]]; m = A; x = A[[All, 1]];
  p = 1 ; NoUniqueSoln = 0; Print["Sistema Inicial", MatrixForm[A]];
   For
    [i = 1, i < n, i++,
    Print["Etapa", i];
        If[A[[i, i]] ≠ 0, p = i,
         p = i; While[(p ≤ n) && (p ≥ i) && (Abs[A[[p, i]]] < 10^-15), p = p + 1]];
        If[p > n, NoUniqueSoln = 1; Abort,
     If[p ≠ i, For[j = 1, j ≤ n + 1, j++, c = A[[i, j]]; A[[i, j]] = A[[p, j]];
        A[[p, j]] = c]; Print["F", p, "<->F", i, MatrixForm[A]], Continue]
      For[j = i + 1, j ≤ n, j++, m[[j, i]] = A[[j, i]] / A[[i, i]];
       A = ReplacePart[A, A[[j]] - m[[j, i]] * A[[i]] , j];
       Print["F", j, "-(", m[[j, i]], ")F", i, MatrixForm[Chop[A]]]
      ] ] ] ;
  If
   [(A[[n, n]] == 0) || ( NoUniqueSoln == 1), Abort,
           x[[n]] = A[[n, n + 1]] / A[[n, n]] ;
   For[i = n - 1, i ≥ 1, i--,
    x[[i]] = (A[[i, n + 1]] - Sum[A[[i, j]] * x[[j]], {j, i + 1, n}]) / A[[i, i]] ]
  ];
  Print["Sistema Triangularizado", MatrixForm[Chop[A]]];
  If[NoUniqueSoln == 1 ,
   Print["No existe solución única"], Print["Solución=", x]]]
```

$$A = \begin{pmatrix} 2 & -7 & 4 \\ 2 & -8 & -6 \\ -4 & 8 & 5 \end{pmatrix}$$

```
{{2, -7, 4}, {2, -8, -6}, {-4, 8, 5}}
```

```
b = {9, 1, 7}
```

```
{9, 1, 7}
```

```
GaussElim [A, b]
```

Sistema Inicial $\begin{pmatrix} 2 & -7 & 4 & 9 \\ 2 & -8 & -6 & 1 \\ -4 & 8 & 5 & 7 \end{pmatrix}$

Etapa1

F2-(1)F1 $\begin{pmatrix} 2 & -7 & 4 & 9 \\ 0 & -1 & -10 & -8 \\ -4 & 8 & 5 & 7 \end{pmatrix}$

F3-(-2)F1 $\begin{pmatrix} 2 & -7 & 4 & 9 \\ 0 & -1 & -10 & -8 \\ 0 & -6 & 13 & 25 \end{pmatrix}$

Etapa2

F3-(6)F2 $\begin{pmatrix} 2 & -7 & 4 & 9 \\ 0 & -1 & -10 & -8 \\ 0 & 0 & 73 & 73 \end{pmatrix}$

Sistema Triangularizado $\begin{pmatrix} 2 & -7 & 4 & 9 \\ 0 & -1 & -10 & -8 \\ 0 & 0 & 73 & 73 \end{pmatrix}$

Solución $=\left\{-\frac{9}{2}, -2, 1\right\}$

```
PivParEsc[AA_, B_] :=
 Module[{s, per, i, j, k, calpiv, pospiv, el, m, a, b, sol}, a = AA; b = B;
  s = Table[Abs[Max[Abs[a[[i]]]]], {i, 1, Length[a]}];
  per = Table[i, {i, 1, Length[a]}];
  For[i = 1, i < Length[a],
   calpiv =
    Table[Abs[a[[per[[j]], i]] / s[[per[[j]]]]], {j, i, Length[per]}];
   pospiv = Flatten[Position[calpiv, Max[calpiv]]];
   pospiv = pospiv[[1]] + (i - 1);
   el = per[[pospiv]];
   per = Delete[per, pospiv];
   per = Insert[per, el, i];
   Print["Etapa ", i, "   Pivote ",
    a[[el, i]], " situado en la posición (", el, ",", i, ")"];
   Print["Vector de permutación:", per];

   For[j = i + 1, j ≤ Length[a], m = a[[per[[j]], i]] / a[[el, i]];

    For[k = i + 1, k ≤ Length[a],
     a[[per[[j]], k]] = a[[per[[j]], k]] - m * a[[el, k]]; k++];


    a[[per[[j]], i]] = 0; b[[per[[j]]]] = b[[per[[j]]]] - m * b[[el]]; j++];
   Print[MatrixForm[a]];
   Print["Término independiente=", b];
   i++];

  Print["Por sustitución regresiva se obtiene: "];

  sol = Table[0, {i, 1, Length[a]}];
  sol[[Length[a]]] = b[[per[[Length[a]]]]] / a[[per[[Length[a]]], Length[a]]];
  For[i = Length[a] - 1, i ≥ 1,
   sol[[i]] = (b[[per[[i]]]] - Sum[a[[per[[i]], j]] * sol[[j]], {j, i + 1, Length[a]}]) / a[[per[[i]], i]];
   i--;];
  Print["Solucion= ", sol];
  Return[{a, b, sol}]]
```

```
PivParEsc[A, b]
```

Etapa 1 Pivote −4 situado en la posición (3,1)

Vector de permutación:{3, 1, 2}

$$\begin{pmatrix} 0 & -3 & \frac{13}{2} \\ 0 & -4 & -\frac{7}{2} \\ -4 & 8 & 5 \end{pmatrix}$$

Término independiente$=\left\{\frac{25}{2}, \frac{9}{2}, 7\right\}$

Etapa 2 Pivote −4 situado en la posición (2,2)

Vector de permutación:{3, 2, 1}

$$\begin{pmatrix} 0 & 0 & \frac{73}{8} \\ 0 & -4 & -\frac{7}{2} \\ -4 & 8 & 5 \end{pmatrix}$$

Término independiente$=\left\{\frac{73}{8}, \frac{9}{2}, 7\right\}$

Por sustitución regresiva se obtiene:

Solucion$=\left\{-\frac{9}{2}, -2, 1\right\}$

$$\left\{\left\{\left\{0, 0, \frac{73}{8}\right\}, \left\{0, -4, -\frac{7}{2}\right\}, \{-4, 8, 5\}\right\}, \left\{\frac{73}{8}, \frac{9}{2}, 7\right\}, \left\{-\frac{9}{2}, -2, 1\right\}\right\}$$

4º.- El comando LUDecomposition realiza una descomposicion de A en producto de una triangular inferior por una triangular superior con intercambio de filas. LUDecomposition, **devuelve una lista tres elementos. El primero es una combinacion de las matrices triangular inferior y superior, el segundo es un vector que especifica las filas que se han usado en el pivoteo, y el tercero es una estimación del nº de condición de la matriz. El comando LUMatrices[ ] transforma el resultado de la matriz conjunta que devuelve el comando LUDecomposition, en dos matrices L y U, de modo que ya se puede realizar el proceso de resolucion del sistema en dos etapas. Se debe tener en cuenta que si el comando LUDecomposition ha realizado intercambio de filas, tambien se debera aplicar dicho intercambio a el vector b , termino independiente del sistema que se pretende resolver.**

```
{lu, p, cn} = LUDecomposition[A]
```

```
{{{2, -7, 4}, {1, -1, -10}, {-2, 6, 73}}, {1, 2, 3}, 1}
```

```
l = Table[If[i > j, lu[[i, j]], If[i == j, 1, 0]],
  {i, 1, Length[lu]}, {j, 1, Length[lu]}]
```

```
{{1, 0, 0}, {1, 1, 0}, {-2, 6, 1}}
```

```
l // MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ -2 & 6 & 1 \end{pmatrix}$$

```
u = Table[If[i <= j, lu[[i, j]], 0], {i, 1, Length[lu]}, {j, 1, Length[lu]}]
```

```
{{2, -7, 4}, {0, -1, -10}, {0, 0, 73}}
```

```
u // MatrixForm
```

$$\begin{pmatrix} 2 & -7 & 4 \\ 0 & -1 & -10 \\ 0 & 0 & 73 \end{pmatrix}$$

```
u // MatrixForm
```

$$\begin{pmatrix} 2 & -7 & 4 \\ 0 & -1 & -10 \\ 0 & 0 & 73 \end{pmatrix}$$

```
b
```

```
{9, 1, 7}
```

```
b[[p]]
```

```
{9, 1, 7}
```

```
LinearSolve[u, LinearSolve[l, b[[p]]]]
```

$$\left\{-\frac{9}{2}, -2, 1\right\}$$

```
LinearSolve[A, b]
```

$$\left\{-\frac{9}{2}, -2, 1\right\}$$

Ejercicio 2º

Resolver el siguiente sistema usando la factorización de

Choleski:

$$6x_1+2x_2+x_3-x_4=0$$
$$2x_1+4x_2+x_3 \quad =7$$
$$x_1+x_2+4x_3-x_4 \quad =-1$$
$$-x_1-x_3+3x_4 \quad =-2$$

La descomposición de Cholesky se aplica a una matriz simétrica definida positiva y da como resultado el producto de una matriz triangular superior por una triangular inferior que es la traspuesta de la anterior. Para aplicar el comando CholeskiDecomposition es necesario cargar previamente el paquete LinearAlgebra`Cholesky`.

```
A = {{6, 2, 1, -1}, {2, 4, 1, 0}, {1, 1, 4, -1}, {-1, 0, -1, 3}};
```

```
trisup = CholeskyDecomposition[A]
```

$$\left\{\left\{\sqrt{6}, \sqrt{\frac{2}{3}}, \frac{1}{\sqrt{6}}, -\frac{1}{\sqrt{6}}\right\}, \left\{0, \sqrt{\frac{10}{3}}, \sqrt{\frac{2}{15}}, \frac{1}{\sqrt{30}}\right\}, \left\{0, 0, \sqrt{\frac{37}{10}}, -\frac{9}{\sqrt{370}}\right\}, \left\{0, 0, 0, \sqrt{\frac{191}{74}}\right\}\right\}$$

Comprobamos que el producto de las dos matrices es igual a la matriz A

```
A - Transpose[trisup].trisup
```

```
{{0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0, 0}}
```

```
triinf = Transpose[trisup];
```

```
b = {0, 7, -1, -2};
```

```
y = Table[0, {4}];
```

Sustitución progresiva para resolver el sistema triinf.y=b

```
y[[1]] = b[[1]] / triinf[[1, 1]];
```

$$\text{Do}\left[y[[i]] = \left(b[[i]] - \sum_{k=1}^{i-1} \text{triinf}[[i, k]] * y[[k]]\right) \Big/ \text{triinf}[[i, i]], \{i, 2, 4\}\right]$$

```
x = Table[0, {4}];
```

Sustitución regresiva para resolver el sistema trisup.x=y

```
x[[4]] = y[[4]] / trisup[[4, 4]];
```

$$\text{Do}\left[x[[i]] = \left(y[[i]] - \sum_{k=i+1}^{4} \text{trisup}[[i, k]] * x[[k]]\right) \Big/ \text{trisup}[[i, i]], \{i, 3, 1, -1\}\right]$$

```
x // N
```

```
{-0.858639, 2.41885, -0.958115, -1.27225}
```

```
LinearSolve[trisup, LinearSolve[triinf, b]] // N
```

```
{-0.858639, 2.41885, -0.958115, -1.27225}
```

```
LinearSolve[A, b] // N
```

```
{-0.858639, 2.41885, -0.958115, -1.27225}
```