Práctica 2

2.1 Sistemas de ecuaciones lineales: Métodos directos

En la práctica anterior ya hemos visto algunas órdenes para resolver directamente sistemas de ecuaciones lineales. Vamos a estudiar además otros dos métodos directos que consisten en factorizar la matriz del sistema: factorización LU y factorización de Cholesky.

2.1.1 Factorización LU

Esta factorización se basa en la eliminación gaussiana. Agrupando convenientemente las operaciones elementales necesarias, se trata de escribir la matriz de coeficientes del sistema $A \in M_{m\times n}(\mathbb{R})$ en la forma

$$A = PLU,$$

con $P \in M_m(\mathbb{R})$ una matriz permutación, $L \in M_m(\mathbb{R})$ triangular inferior con unos en la diagonal principal y $U \in M_{m\times n}(\mathbb{R})$ una matriz "triangular superior".
Con esta factorización de A, para resolver el sistema $AX = B$, es decir, $P\,(\underbrace{L\,(\underbrace{UX})}) = B$, se procede resolviendo los sistemas

$$PZ = B,$$
$$LY = Z,$$
$$UX = Y.$$

La matriz P reúne las operaciones de intercambio de filas, lo que equivale a una reordenación en la escritura de las ecuaciones. Supondremos, en primer lugar, que **<u>no</u>** ha sido necesario realizar ninguna **<u>permutación de filas</u>** por lo que P es la matriz identidad y $Z = B$ sin tener que resolver el primer sistema (que no es sino una permutación). Así, se obtendría la factorización

$$A = LU,$$

conocida como factorización de Doolittle. Para la obtención de L y U se sigue el siguiente proceso: efectuamos en A operaciones elementales por filas (únicamente sumar a algunas filas múltiplos adecuados de otras) hasta obtener una matriz escalonada por filas: U, esto es

$$(A|I_m)\ldots(P_k\ldots P_1 A = U|P_k\ldots P_1 I_m)$$

Como las matrices P_i son elementales, son inversibles y su producto también es inversible, por lo que podemos escribir $L^{-1} = P_k \dots P_1$ de donde resulta $L^{-1}A = U$ y, por ello, $A = LU$. Se observa que con el escalonamiento anterior lo que hemos obtenido es

$$(A|I_m)\dots(U|L^{-1}).$$

Vamos a obtener la factorización de $A = \begin{pmatrix} 1 & 0 & 2 & 4 \\ -1 & 1 & 0 & 3 \\ -2 & 1 & 1 & 1 \end{pmatrix}$

Introducimos la matriz y le añadimos la identidad:

```
-->     a:matrix([1,0,2,4],[-1,1,0,3],[-2,1,1,1])$
```

```
-->     a1:transpose(append(transpose(a),ident(3)))$
```

escalonamos por filas, sin hacer 1 los elementos iniciales:

```
-->     a2:triangularize(a1);
```

prescindiendo de las 3 últimas columnas obtenemos U

```
-->     u:submatrix(a2,5,6,7);
```

prescindiendo de las 4 primeras columnas obtenemos L^{-1}

```
-->     l:invert(submatrix(a2,1,2,3,4));
```

por último, comprobamos que $LU = A$

```
-->     is(l.u=a);
```

Utilizando la factorización anterior, vamos a resolver el sistema $A(x_1\ x_2\ x_3\ x_4)^T = (1\ 2\ 3)^T$.

```
-->     b:transpose([1,2,3]);
```

```
-->     y:echelon(addcol(l,b));
```

```
-->     y:submatrix(y,1,2,3);
```

```
-->     x:echelon(addcol(u,y));
```

```
-->     x3:x[3,5]-x[3,4]*x4;
        x2:x[2,5]-x[2,4]*x4-x[2,3]*x3;
        x1:x[1,5]-x[1,4]*x4-x[1,3]*x3-x[1,2]*x2;
```

La variable x_4 actúa como parámetro. Si para comprobar que hemos obtenido las soluciones del sistema utilizamos la orden

```
-->     is(a.transpose([x1,x2,x3,x4])=b);
```

se observa que la respuesta es que no. Esto es debido a que no simplifica los cálculos, por lo que lo adecuado en este caso es utilizar la orden

```
-->     is(ratsimp(a.transpose([x1,x2,x3,x4]))=b);
```

Si la **matriz** de coeficientes A es **cuadrada**. La función:
`get_lu_factors (lu_factor(A))`
devuelve una lista de la forma $[P, L, U]$. Por lo que, aunque sean necesarias permutaciones de filas, el problema lo tenemos resuelto en este caso. Vamos a utilizar este procedimiento con el sistema $\begin{pmatrix} 0 & 0 & 2 \\ -1 & 1 & 0 \\ -2 & 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$

Introducimos las matrices, obtenemos las matrices de la factorización y lo comprobamos:

```
-->     a:matrix([0,0,2],[-1,1,0],[-2,1,1])$

-->     b:transpose([1,2,3])$

-->     plu:get_lu_factors(lu_factor(a));

-->     is(plu[1].plu[2].plu[3]=a);
```

resolvemos el sistema asociado a P, obteniendo Z:

```
-->     z:echelon(addcol(plu[1],b));

-->     z:submatrix(z,1,2,3);
```

conocida Z, resolvemos el sistema asociado a L, obteniendo Y:

```
-->     y:echelon(addcol(plu[2],z));

-->     y:submatrix(y,1,2,3);
```

finalmente, con Y, resolvemos el sistema asociado a U, obteniendo X:

```
-->     x:echelon(addcol(plu[3],y));

-->     x3:x[3,4];
        x2:x[2,4]-x[2,3]*x3;
        x1:x[1,4]-x[1,3]*x3-x[1,2]*x2;
```

y comprobamos que es la solución del sistema propuesto:

```
-->     is(a.transpose([x1,x2,x3])=b);
```

Ejercicio 1. Empleando la factorización LU resolver el sistema

$$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 4 \\ 2 & -1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

Por último, consideremos el **caso general**: la matriz A no es necesariamente cuadrada y puede ser necesaria la permutación de filas.

La permutación de filas, no es otra cosa que la alteración del orden de escritura de las ecuaciones. En ocasiones, es necesario alterar también el orden de escritura de las variables, el sistema de ecuaciones es idéntico pero no así la matriz que se va a factorizar.

La permutación de filas se efectúa para pasar a una matriz escalonada por filas. La permutación de columnas se efectúa para que las h variables principales del sistema asociado se correspondan con las h primeras columnas de U. Está claro que la factorización obtenida no es única.

El algoritmo que se sigue en este caso tiene 3 pasos:

I Tomar $U = A \in M_{m\times n}(\mathbb{R}), L = 0 \in M_m(\mathbb{R})$ y $P = I_m$.

II Efectuamos en U todas las operaciones elementales necesarias para escalonar por filas, si es necesario, se reordenan previamente las columnas. Para cada operación de filas,

a) Si es del tipo $P_{ij}(\lambda)$ entonces hacemos $l_{ij} = -\lambda$,

b) Si es del tipo P_{ij}, hacemos la misma operación con P y L.

III Con U ya escalonada, tomar $L = L + I_m$ y $P = P^T$.

Estas matrices verifican $PLU = \widetilde{A}$ donde $\widetilde{A}$ es la obtenida mediante la reordenación de columnas de A. Por ello, apuntaremos qué orden tienen ahora esas columnas. Esto lo vamos a controlar con la lista `var`.

En el fichero **factPLU.mac** están escitas las órdenes necesarias para obtener la factorización buscada. Una vez cargado ese archivo, para obtener la factorización de la matriz `a` hay que escribir `plu(a)`. La salida es una lista con cuatro términos que corresponden a `[p,l,u,var]`.

Así, por ejemplo, si introducimos la matriz "a"

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 2 & 3 \\ 1 & 0 & 0 & 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 & -1 & 1 & 0 & 3 \\ 1 & 0 & 0 & 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 & -1 & 1 & 0 & 3 \\ 0 & 1 & 2 & 0 & -1 & 1 & 0 & 5 \\ 0 & 0 & 0 & 1 & -2 & 1 & 1 & 3 \end{pmatrix}$$

```
-->     a:matrix([1,0,0,1,0,0, 2,3], [1,0,0,1,0,0,2,3],
        [0,0,0,1,-1,1,0,3],[1,0,0,1,0,0, 2,3],
        [0,0,0,1,-1,1,0,3],[0,1,2,0,-1,1,0,5],[0,0,0,1,-2,1,1,3]);
```

Con la orden

```
-->     plu(a);
```

lo que obtenemos es la factorización PLU de $\widetilde{A}$

```
-->     p.l.u;
```

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 2 & 3 & 0 \\ 1 & 0 & 1 & 0 & 0 & 2 & 3 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & 3 & 0 \\ 1 & 0 & 1 & 0 & 0 & 2 & 3 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & 3 & 0 \\ 0 & 1 & 0 & -1 & 1 & 0 & 5 & 2 \\ 0 & 0 & 1 & -2 & 1 & 1 & 3 & 0 \end{pmatrix}$$

en la que se observa que la tercera columna de la matriz A ha pasado a ser la última de ésta. Ahora bien, las variables también han cambiado también de orden,

```
-->     var;
```

$[1, 2, 4, 5, 6, 7, 8, 3]$

por lo que en el sistema asociado, el orden de las variables debería ser el que indica `var`, es decir: si

$$\widetilde{X} = (x_1, x_2, x_4, x_5, x_6, x_7, x_8, x_3)^T,$$

$AX = B = \widetilde{A}\widetilde{X}$ y el sistema factorizado

$$PLU\widetilde{X} = B$$

nos da la solución del propuesto.

Según hemos visto, cuando son necesarias permutaciones, bien sea sólo de filas o de filas y columnas, lo que se obtiene no es la factorización de la matriz A sino una factorización de la matriz $\widetilde{A}$ con las filas/columnas permutadas. A pesar de ello, abusando del término, hablaremos aquí de factorización de A.

Ejercicio 2. Introducir una matriz a de dimensiones 4×6 y hallar la factorización PLU correspondiente indicando, si es necesario, el nuevo orden de las variables.

2.1.2 Factorización de Cholesky

Si la matriz A es simétrica definida positiva, se puede obtener una factorización del tipo

$$A = L^T L$$

que se llama factorización de Cholesky. La orden `cholesky(A)` nos proporciona la matriz L asociada. El método de resolución es análogo al de la factorización anterior.

Vamos a utilizar esta factorización para resolver el sistema:

$$\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

```
-->     a:matrix([2,-1,0],[-1,2,-1],[0,-1,2])$ c:cholesky (a);

-->     is(c.transpose(c)=a);

-->     b:transpose([1,2,3]);

-->     y:echelon(addcol(c,b));

-->     y:submatrix(y,1,2,3);

-->     x:echelon(addcol(transpose(c),y));

-->     x3:x[3,4];
        x2:x[2,4]-x[2,3]*x3;
        x1:x[1,4]-x[1,3]*x3-x[1,2]*x2;

-->     is(a.transpose([x1,x2,x3])=b);
```