Práctica 5
Cálculo de inversas y descomposición LU

27 de enero de 2014

Índice

1. Cálculo de la matriz inversa

1.1. La función inv

Para calcular la matriz inversa de una matriz cuadrada A, utilizamos la función **inv** aplicada sobre A. Si A es invertible, entonces el comando

```
-->inv(A)
```

nos da la matriz inversa A^{-1}. En otro caso, o bien aparece un error indicando que la matriz es singular (es decir, no invertible), o aparece un mensaje de aviso diciendo que la matriz es próxima a ser singular o está mal condicionada. En el último caso no es claro si la matriz es singular o no, podemos utilizar, en esta situación, la función **rank** para calcular el rango de A y comprobar si A tiene rango máximo.

Ejemplo 1. Consideramos la matriz

$$\mathsf{A} = \begin{bmatrix} 2 & 5 & -1 \\ 0 & 0 & 9 \\ 0 & 5 & 4 \end{bmatrix}.$$

```
-->A=[2 5 -1;0 0 9;0 -5 4];

-->inv(A)
 ans  =
    0.5  - 0.1666667    0.5
    0.     0.0888889  - 0.2
    0.     0.1111111    0.
```

Podemos comprobar que la inversa proporcionada es correcta:

```
-->A*inv(A)
 ans  =
    1.    5.551D-17    0.
    0.    1.           0.
    0.  - 5.551D-17    1.
```

Las entradas $(1,2)$ y $(3,2)$ (es decir, $5{,}551 \cdot 10^{-17}$ y $-5{,}551 \cdot 10^{-17}$) habrían, probablemente, de ser cero (no son cero, probablemente, debido a errores de redondeo en los cálculos). Utilizando la función **clean**, podemos sustituir a estos números por 0:

```
-->clean(ans)
 ans  =
    1.    0.    0.
    0.    1.    0.
    0.    0.    1.
```

Esta es la matriz identidad y, por lo tanto, el resultado es correcto.

Ejemplo 2. Consideramos la matriz

$$\mathsf{B} = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{bmatrix}.$$

```
-->B=[1 1 1; 2 2 2; 3 3 3]
 B  =
    1.    1.    1.
    2.    2.    2.
    3.    3.    3.

-->inv(B)
       !--error 19
Problem is singular.
```

Calculando el rango de B podemos ver que esta matriz no tiene rango máximo:

```
->rank(B)
 ans  =
    1.
```

Así, la matriz no es invertible en este caso.

Ejemplo 3. Consideramos ahora la matriz

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}.$$

```
-->C=[1 2 3;4 5 6;7 8 9]
 C  =
    1.    2.    3.
    4.    5.    6.
    7.    8.    9.

-->inv(C)
warning :
matrix is close to singular or badly scaled. rcond =    0.0000D+00

 ans  =
   1.0D+15 *

  - 4.5035996    9.0071993  - 4.5035996
    9.0071993  - 18.014399    9.0071993
  - 4.5035996    9.0071993  - 4.5035996

-->rank(C)
 ans  =
    2.
```

Concluimos que C no es invertible.

1.2. Cálculo de matrices inversas con el algoritmo de Gauss-Jordan

Dada una matriz cuadrada A, podemos también encontrar la inversa de A aplicando el algoritmo de Gauss Jordan a la matriz A1 = [A I], donde I es la matriz identidad:

Ejemplo 4. Sea A la matriz del ejemplo 1.

```
-->A1=[A,eye(3,3)]
 A1  =
    2.    5.  - 1.    1.    0.    0.
    0.    0.    9.    0.    1.    0.
    0.  - 5.    4.    0.    0.    1.

-->rref(A1)
 ans  =
    1.    0.    0.    0.5  - 0.1666667    0.5
    0.    1.    0.    0.     0.0888889  - 0.2
    0.    0.    1.    0.     0.1111111    0.
```

Entonces A es invertible y su inversa es

```
-->ans(:,4:6)
 ans  =
    0.5  - 0.1666667    0.5
    0.     0.0888889  - 0.2
    0.     0.1111111    0.
```

Ejemplo 5. Haciendo lo mismo con la matriz C del ejemplo 3:

```
-->C1=[C, eye(3,3)]
 C1  =
    1.    2.    3.    1.    0.    0.
    4.    5.    6.    0.    1.    0.
    7.    8.    9.    0.    0.    1.

-->rref(C1)
 ans  =
    1.    0.  - 1.    0.  - 2.6666667    1.6666667
    0.    1.    2.    0.    2.3333333  - 1.3333333
    0.    0.    0.    1.  - 2.           1.
```

Como la matriz de la izquierda no es la matriz identidad, C no es invertible.

2. Descomposición LU de una matriz cuadrada

2.1. Cálculo

Una descomposición LU (también llamada factorización LU) es una descomposición de una matriz que escribe una matriz como el producto de una matriz triangular inferior y una triangular superior. Esta descomposición tiene muchas aplicaciones. Presentaremos aquí tres de ellas: resolución de sistemas de ecuaciones lineales, cálculo del determinante de una matriz cuadrada y cálculo de matrices inversas.

Antes que nada, explicaremos el proceso de obtención (a mano) de una descomposición LU mediante un par de ejemplos. Consideramos la matriz

$$A = \begin{bmatrix} 2 & 5 & -3 \\ 4 & 7 & -4 \\ -6 & -3 & 1 \end{bmatrix}$$

Efectuando operaciones elementales por filas (es decir, multiplicando A por matrices elementales convenientes) podemos encontrar, en este caso, una matriz triangular superior que es equivalente (por filas) a A:

$$E_{32}(4)E_{31}(3)E_{21}(-2)A = \begin{bmatrix} 2 & 5 & -3 \\ 0 & -3 & 2 \\ 0 & 0 & 0 \end{bmatrix}.$$

Tomando $B = E_{32}(4)E_{31}(3)E_{21}(-2)$ y

$$U = \begin{bmatrix} 2 & 5 & -3 \\ 0 & -3 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

tenemos que

$$BA = U,$$

donde B es una matriz triangular inferior y U es una matriz triangular superior. Como B es producto de matrices elementales, es invertible. Multiplicando la igualdad anterior por B^{-1} (por la izquierda), tenemos:

$$A = B^{-1}U.$$

No es difícil probar que la inversa de una matriz triangular inferior invertible es también una matriz triangular inferior. Por tanto, tomando $L = B^{-1}$, obtenemos una descomposición LU de A:

$$A = LU.$$

Es muy fácil calcular L a mano porque conocemos las inversas de las matrices elementales:

$$\begin{aligned} L = B^{-1} &= (E_{32}(4)E_{31}(3)E_{21}(-2))^{-1} = E_{21}(-2)^{-1}E_{31}(3)^{-1}E_{32}(4)^{-1} \\ &= E_{21}(2)E_{31}(-3)E_{32}(-4) = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & -4 & 1 \end{bmatrix}. \end{aligned}$$

En resumen, hemos obtenido la siguiente descomposición LU de A:

$$A = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & -4 & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} 2 & 5 & -3 \\ 0 & -3 & 2 \\ 0 & 0 & 0 \end{bmatrix}}_{U}.$$

Veamos otro ejemplo. Consideramos la matriz

$$\mathsf{C} = \begin{bmatrix} 0 & -2 & 1 \\ 3 & 0 & 2 \\ 0 & 2 & 4 \end{bmatrix}.$$

Si aplicamos el mismo proceso de antes a la matriz C obtenemos:

$$\mathsf{E}_{32}(1)\mathsf{E}_{12}\mathsf{C} = \underbrace{\begin{bmatrix} 3 & 0 & 2 \\ 0 & -2 & 1 \\ 0 & 0 & 5 \end{bmatrix}}_{\mathsf{U}}.$$

Entonces,

$$\mathsf{L} = (\mathsf{E}_{32}(1)\mathsf{E}_{12})^{-1} = \mathsf{E}_{12}^{-1}\mathsf{E}_{32}(1)^{-1} = \mathsf{E}_{12}\mathsf{E}_{32}(-1) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 1 \end{bmatrix}.$$

Pero... esta matriz L **no es triangular inferior!** La razón es que, en el proceso de obtención, hemos intercambiado filas (es decir, hemos efectuado operaciones elementales de tipo 1). Ésta es la diferencia con el ejemplo previo. No obstante, en este caso, mantendremos el nombre de «descomposició LU» para referirnos a la factorización obtenida. Es decir: **una descomposición** LU **de una matriz** A **será una factorización** $\mathsf{A} = \mathsf{LU}$ **donde** U **es triangular superior y** L **es triangular inferior (*excepto reordenación de filas*).**

Las descomposiciones LU pueden obtenerse fácilmente con Scilab utilizando la función **lu**. Pero... atención! Esta función utiliza la eliminación gaussiana *con pivotación parcial* para obtener la descomposición. Esto quiere decir que, probablemente, Scilab utilizará intercambios de filas en el proceso y que la matriz obtenida L no será triangular inferior (aunque, naturalmente, lo será si reordenamos las filas). Por ejemplo, si calculamos, utilizando Scilab, una descomposición LU de la matriz anterior A:

```
-->A=[2 5 -3; 4 7 -4; -6 -3 1];

-->[L,U]=lu(A)
 U  =
  - 6.   - 3.     1.
    0.     5.   - 3.3333333
    0.     0.     4.441D-16
 L  =
  - 0.3333333    0.8    1.
  - 0.6666667    1.     0.
    1.           0.     0.
```

«Anulamos» los elementos «cercanos a cero» del factor U:

```
->clean(U)
 ans  =
  - 6.  - 3.    1.
    0.    5.  - 3.3333333
    0.    0.    0.
```

Observamos que el factor L que Scilab ha calculado no es triangular inferior.

También se puede utilizar el comando de Scilab $[\mathrm{L}, \mathrm{U}, \mathrm{P}] = \mathbf{lu}(\mathrm{A})$ (añadiendo otro parámetro a la salida):

```
-->[L1,U1,P]=lu(A)
 P  =
    0.    0.    1.
    0.    1.    0.
    1.    0.    0.
 U1  =
  - 6.  - 3.    1.
    0.    5.  - 3.3333333
    0.    0.    4.441D-16
 L1  =
    1.           0.     0.
  - 0.6666667    1.     0.
  - 0.3333333    0.8    1.

 -->P*L1*U1
 ans  =
    2.    5.  - 3.
    4.    7.  - 4.
  - 6.  - 3.    1.

-->U-U1
 ans  =
    0.    0.    0.
    0.    0.    0.
    0.    0.    0.

-->L-P*L1
 ans  =
    0.    0.    0.
    0.    0.    0.
    0.    0.    0.
```

Es decir, cuando escribimos tres parámetros como salida, $[\mathrm{L}, \mathrm{U}, \mathrm{P}] = \mathrm{lu}(\mathrm{A})$, nos devuelve la matriz de permutación P tal que $\mathrm{PA} = \mathrm{LU}$, donde U es triangular superior y L es triangular

inferior con todos sus elementos diagonales iguales a 1; si escribimos sólo dos parámetros, $[\mathsf{L},\mathsf{U}] = \mathrm{lu}(\mathsf{A})$, la matriz U es la misma pero la matriz L nos la da ya permutada y, por tanto, esta L verificará la igualdad $\mathsf{A} = \mathsf{LU}$.

2.2. Aplicaciones

2.2.1. Resolución de sistemas de ecuaciones lineales

La descomposición LU puede aplicarse a la resolución de sistemas de ecuaciones lineales. Veamos un ejemplo.

Consideramos el sistema

$$\left.\begin{array}{rcl} -2y + z &=& 1 \\ 3x + 2y &=& 2 \\ 2y + 4z &=& 3 \end{array}\right\},$$

cuya matriz de coeficientes es la matriz C del ejemplo anterior y cuya espresión matricial es:

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

Sustituyendo la matriz de coeficientes por su descomposición LU tenemos

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 1 \end{bmatrix}}_{\mathsf{L}} \underbrace{\begin{bmatrix} 3 & 0 & 2 \\ 0 & -2 & 1 \\ 0 & 0 & 5 \end{bmatrix}}_{\mathsf{U}} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_{\vec{x}} = \underbrace{\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}}_{\vec{b}}. \tag{1}$$

El producto $\mathsf{U}\vec{x}$ es un vector columna de tres componentes que denotaremos por $\vec{y}$, es decir:

$$\mathsf{U}\vec{x} = \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}.$$

Por consiguiente, la igualdad (1) se puede escribir:

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 1 \end{bmatrix}}_{\mathsf{L}} \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}}_{\vec{y}} = \underbrace{\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}}_{\vec{b}}.$$

Esta es la expresión matricial del siguiente sistema:

$$\left.\begin{array}{rcl} y_2 &=& 1 \\ y_1 &=& 2 \\ -y_2 + y_3 &=& 3 \end{array}\right\} \tag{2}$$

Este sistema tiene una matriz de coeficientes «casi triangular» y puede ser resuelto directamente por una sustitución progresiva, dando lugar a esta solución:

$$\vec{y} = \begin{bmatrix} 2 \\ 1 \\ 4 \end{bmatrix}.$$

Ahora, como $\mathrm{U}\vec{x} = \vec{y}$ y conocemos $\vec{y}$, podemos calcular $\vec{x}$ resolviendo el sistema de ecuaciones lineales cuya expresión matricial es $\mathrm{U}\vec{x} = \vec{y}$:

$$\underbrace{\begin{bmatrix} 3 & 0 & 2 \\ 0 & -2 & 1 \\ 0 & 0 & 5 \end{bmatrix}}_{\mathrm{U}} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_{\vec{x}} = \underbrace{\begin{bmatrix} 2 \\ 1 \\ 4 \end{bmatrix}}_{\vec{y}}.$$

Este es el «sistema triangular»

$$\left.\begin{array}{rcl} 3x + 2z & = & 2 \\ -2y + z & = & 1 \\ -3z & = & 4 \end{array}\right\}. \tag{3}$$

Resolviéndolo por sustitución regresiva obtenemos la solución:

$$\vec{x} = \begin{bmatrix} 2/15 \\ -1/10 \\ 4/5 \end{bmatrix}.$$

En resumen, la descomposición LU de la matriz de coeficientes nos ha permitido transformar el sistema de ecuaciones en dos sistemas que pueden resolverse directamente por sustitución progresiva y regresiva, respectivamente: (2) y (3).

Notamos que este método de resolver sistemas de ecuaciones es especialmente útil cuando se deben resolver muchos sistemas de ecuaciones lineales con la misma matriz de coeficientes. La razón es que la descomposición LU se aplica sólo sobre la matriz de coeficientes; esto significa que la misma descomposición LU es válida para todos los sistemas lineales.

Ejemplo 6. Si quisiéramos resolver el sistema de ecuaciones anterior utilizando la descomposición LU y con Scilab podríamos introducir lo siguiente:

```
-->C=[0 -2 1;3 0 2;0 2 4];[L,U]=lu(C)
 U  =
    3.    0.    2.
    0.  - 2.    1.
    0.    0.    5.
 L  =
    0.    1.    0.
    1.    0.    0.
    0.  - 1.    1.
```

```
-->y=L\[1;2;3]
 y  =
    2.
    1.
    4.

-->x=U\y
 x  =
    0.1333333
  - 0.1
    0.8
```

2.2.2. Aplicación al cálculo de determinantes

El determinante de una matriz cuadrada se puede calcular con Scilab utilizando la función **det**. Por ejemplo:

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

-->det(D)
 ans  =
  - 2.
```

El procedimiento utilizado internamente por Scilab cuando se aplica esta función a una matriz A es el siguiente:

1. Se obtiene la descomposición LU de A.
2. Teniendo en cuenta que A $=$ LU, el determinante de A se calcula como $\det \mathrm{L} \det \mathrm{U}$. Notamos que ambos determinantes involucrados aquí son muy fáciles de calcular:
   - Como U es triangular superior, su determinante es el producto de las entradas diagonales.
   - El determinante de L es ± 1, donde el signo será $+$ cuando el número de operaciones elementales de intercambio sea par, y $-$ cuando este número sea impar[1].

```
-->A=[0 2 3; -4 6 0; 2 -5 5];

-->[L,U]=lu(A)
 U  =
  - 4.    6.    0.
```

[1]Scilab calcula L de manera que, después de permutar convenientemente las filas para obtener una matriz triangular inferior, en la diagonal sólo hay unos.

```
    0.    2.    3.
    0.    0.    8.
 L  =
    0.     1.    0.
    1.     0.    0.
  - 0.5  - 1.    1.

-->det(L)
 ans  =
  - 1.

-->det(U)
 ans  =
  - 64.

-->det(L)*det(U)
 ans  =
    64.

-->det(A)
 ans  =
    64.
```

2.2.3. Aplicación al cálculo de matrices inversas

Cuando utilizamos la función **inv** para calcular la inversa de una matriz cuadrada A de orden n, Scilab efectúa, automáticamente, las siguientes operaciones:

1. Obtén una descomposición LU de A.
2. Calcula $\det \mathrm{U}$.
3. Si $\det \mathrm{U} = 0$, entonces Scilab indica que la matriz es singular (no invertible).
4. Si $\det \mathrm{U} \neq 0$, entonces calcula U^{-1} de la siguiente manera: Si $\vec{u}_j$ denota la columna j de U^{-1} e I es la matriz identidad, es evidente que la igualdad $\mathrm{U}\mathrm{U}^{-1} = \mathrm{I}$ es equivalente al conjunto de igualdades
$$\mathrm{U}\vec{u}_j = \vec{e}_j \quad j = 1, 2, \ldots n,$$
donde $\vec{e}_j$ es el vector columna j de la matriz identidad; por consiguiente, resolviendo todos los sistemas de ecuaciones lineales $\mathrm{U}\vec{x} = \vec{e}_j$ se obtienen todas las columnas de U^{-1}, es decir, se obtiene la matriz inversa de U. La inversa de la matriz L, L^{-1}, se calcula con el mismo algoritmo. Finalmente, A^{-1} se obtiene como el producto $\mathrm{U}^{-1}\mathrm{L}^{-1}$.

Notamos que es posible obtener una matriz singular tal que todas las entradas diagonales de U sean diferentes de cero (debido a errores de redondeo en la eliminación gaussiana). En este

caso, la matriz inversa es calculada por Scilab y, en algunos casos, aparece un mensaje de aviso. Por esta razón, recomendamos comprobar si o bien la matriz obtenida cuando introducimos el comando inv(A) es de verdad la matriz inversa de A o A tiene rango máximo (por medio de la función **rank**).

Ejemplo 7. Consideramos la matriz:

$$A = \begin{bmatrix} 1 & 2 & -5 \\ -4 & 1 & 0 \\ 1 & 2 & 7 \end{bmatrix}.$$

Calculamos su inversa a partir de su descomposición LU y utilizando la función **inv**:

```
-->A=[1 2 -5; -4 1 0; 1 2 7]; [L,U]=lu(A)
 U  =
  - 4.    1.      0.
    0.    2.25  - 5.
    0.    0.      12.
 L  =
  - 0.25    1.    0.
    1.      0.    0.
  - 0.25    1.    1.

-->inv(U)*inv(L)
 ans  =
    0.0648148  - 0.2222222    0.0462963
    0.2592593    0.1111111    0.1851852
  - 0.0833333    0.           0.0833333

-->inv(A)
 ans  =
    0.0648148  - 0.2222222    0.0462963
    0.2592593    0.1111111    0.1851852
  - 0.0833333    0.           0.0833333
```

Comprobamos que la matriz obtenida es la inversa de A:

```
-->clean(A*ans)
 ans  =
    1.    0.    0.
    0.    1.    0.
    0.    0.    1.
```