

ANÁLISIS NUMÉRICO II (Curso 2009-2010)

Tercer Curso de Ingeniería Técnica Informática

Tema1: Sistemas de ecuaciones lineales

ELIMINACIÓN GAUSSIANA.

Introducción. Los sistemas de ecuaciones lineales aparecen en una gran variedad de áreas científicas, pero también indirectamente en la solución numérica de diferentes problemas matemáticos. Por ejemplo, estos sistemas aparecen en la resolución de problemas numéricos en optimización, aproximación de funciones, ecuaciones diferenciales ordinarias, ecuaciones en derivadas parciales, ecuaciones integrales, etc. Por esta razón, el estudio de métodos numéricos para la resolución de los sistemas de ecuaciones lineales es uno de los pilares del análisis numérico.

Nos centraremos primero en el caso en que la matriz de coeficientes del sistema es cuadrada e invertible, en cuyo caso, como bien sabemos la solución del sistema es única. Posteriormente se tratarán los casos de matrices rectangulares y de matrices cuadradas no invertibles.

Método de eliminación gaussiana. Nos centramos en esta sección en sistemas de ecuaciones lineales con n ecuaciones y n incógnitas. Un ejemplo es el siguiente:

$$\left\{ \begin{array}{rcl} x_1 & +2x_3 & = 3, \\ -2x_1 & +2x_2 & -3x_3 = -2, \\ -3x_1 & +2x_2 & -4x_3 = -3. \end{array} \right\}.$$

Recordemos que el proceso de eliminación gaussiana para su resolución consiste en restar múltiplos de la primera a las demás para hacer desaparecer (eliminar) la primera incógnita de las restantes ecuaciones, y, una vez hecho esto, se repite el proceso con la segunda incógnita, así hasta tener la última incógnita despejada. En el caso que nos ocupa,

$$\left\{ \begin{array}{rcl} x_1 & +2x_3 & = 3, \\ -2x_1 & +2x_2 & -3x_3 = -2, \\ -3x_1 & +2x_2 & -4x_3 = -3. \end{array} \right\} \xrightarrow{\substack{e_2 \leftarrow e_2 + 2e_1 \\ e_3 \leftarrow e_3 + 3e_1}} \left\{ \begin{array}{rcl} x_1 & +2x_3 & = 3, \\ 2x_2 & +x_3 & = 4, \\ 2x_2 & +2x_3 & = 6. \end{array} \right\}$$

$$\xrightarrow{e_3 \leftarrow e_3 - e_2} \left\{ \begin{array}{rcl} x_1 & +2x_3 & = 3, \\ 2x_2 & +x_3 & = 4, \\ x_3 & = 2. \end{array} \right\}.$$

Este último sistema se resuelve fácilmente por sustitución regresiva:

$$\left\{ \begin{array}{rcl} x_1 & +2x_3 & = 3, \\ 2x_2 & +x_3 & = 4, \\ x_3 & = 2. \end{array} \right\} \Rightarrow x_3 = 2, \Rightarrow x_2 = \frac{4 - x_3}{2} = 1, \Rightarrow x_1 = 3 - 2x_3 = -1.$$

En la asignatura *Álgebra*, del primer curso, vimos que este proceso tenía una escritura matricial. El sistema se escribe como $Ax = b$, donde A es la matriz de coeficientes, x el vector de incógnitas y b el vector dato:

$$A = \begin{bmatrix} 1 & 0 & 2 \\ -2 & 2 & -3 \\ -3 & 2 & -4 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ -2 \\ -3 \end{bmatrix}.$$

Recuerde asimismo que, si A es invertible, entonces la única solución de $Ax = b$ es $x = A^{-1}b$. En la práctica, sin embargo, nunca se calcula A^{-1} y después se efectúa $x = A^{-1}b$ por dos razones:

- Calcular A^{-1} es más costoso que resolver el sistema $Ax = b$ por eliminación gaussiana.
- Calcular A^{-1} y multiplicar $A^{-1}b$ es un proceso más sensible a errores de redondeo que resolver $Ax = b$ por eliminación gaussiana.

De hecho, en la práctica para multiplicar por A^{-1} se resuelve un sistema con matriz de coeficientes A .

El proceso de eliminación gaussiana se puede realizar matricialmente, trabajando con la matriz ampliada del sistema $[A|b]$ como sigue:

$$[A|b] = \left[\begin{array}{ccc|c} 1 & 0 & 2 & 3 \\ -2 & 2 & -3 & -2 \\ -3 & 2 & -4 & -3 \end{array} \right] \xrightarrow{\substack{f_2 \leftarrow f_2 + 2f_1 \\ f_3 \leftarrow f_3 + 3f_1}} \left[\begin{array}{ccc|c} 1 & 0 & 2 & 3 \\ 0 & 2 & 1 & 4 \\ 0 & 2 & 2 & 6 \end{array} \right] \xrightarrow{f_3 \leftarrow f_3 - f_2} \left[\begin{array}{ccc|c} 1 & 0 & 2 & 3 \\ 0 & 2 & 1 & 4 \\ 0 & 0 & 1 & 2 \end{array} \right].$$

La última matriz representa a un sistema $Ux = b'$, en el que la matriz de coeficientes es triangular superior.

Las operaciones por filas efectuadas tienen una interpretación matricial en términos de multiplicaciones por matrices elementales. Así por ejemplo, el primer paso del proceso de eliminación gaussiana consiste en multiplicar $Ax = b$ por M_1 para obtener $M_1Ax = M_1b$, donde M_1 es la matriz que al multiplicar por ella resta múltiplos de la primera fila a las demás, dejando ceros en la primera columna por debajo de la diagonal. En nuestro ejemplo,

$$M_1 = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix}.$$

Observe también que

$$M_1[A|b] = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix} \left[\begin{array}{ccc|c} 1 & 0 & 2 & 3 \\ -2 & 2 & -3 & -2 \\ -3 & 2 & -4 & -3 \end{array} \right] = \left[\begin{array}{ccc|c} 1 & 0 & 2 & 3 \\ 0 & 2 & 1 & 4 \\ 0 & 2 & 2 & 6 \end{array} \right].$$

Abreviadamente escribiremos

$$M_1A = A^{(2)}, \quad M_1b = b^{(2)} \quad \text{o también} \quad M_1[A|b] = [A^{(2)}|b^{(2)}].$$

La matriz M_1 se obtiene realizando sobre la matriz identidad las correspondientes operaciones elementales. De esta forma, M_1 es como la identidad, excepto que en la primera columna aparecen los opuestos de los *multiplicadores elementales* $l_{j1} = a_{j1}/a_{11}$:

$$M_1 = \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix}, \quad \text{siendo } l_{21} = a_{21}/a_{11} = -2/1 = -2, \quad l_{31} = a_{31}/a_{11} = -3/1 = -3.$$

El segundo paso de la eliminación consiste en multiplicar el sistema obtenido anteriormente $A^{(2)}x = b^{(2)}$ (o bien su matriz ampliada: $[A^{(2)}|b^{(2)}]$) por la matriz M_2 , que es como la identidad excepto que en la segunda columna aparecen los opuestos de los multiplicadores elementales $l_{j2} = a_{j2}^{(2)}/a_{22}^{(2)}$:

$$M_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -l_{32} & 1 \end{bmatrix}, \quad l_{32} = a_{32}^{(2)}/a_{22}^{(2)} = 2/2 = 1.$$

Se tiene entonces:

$$M_2 [A^{(2)}|b^{(2)}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 2 & | & 3 \\ 0 & 2 & 1 & | & 4 \\ 0 & 2 & 2 & | & 6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 2 & | & 3 \\ 0 & 2 & 1 & | & 4 \\ 0 & 0 & 1 & | & 2 \end{bmatrix}.$$

Resumiendo, el proceso de eliminación conduce a la matriz ampliada $M_2M_1 [A|b] = [A^{(3)}|b^{(3)}]$, de donde deducimos que :

$$M_2M_1 A = A^{(3)} = U \text{ es triangular superior, y } M_2M_1 b = b^{(3)} = b'. \quad (1)$$

Deshaciendo los cambios, obtenemos

$$A = M_1^{-1}M_2^{-1}U, \quad b = M_1^{-1}M_2^{-1}b'.$$

Puesto que las matrices de transformación elemental M_1 y M_2 efectúan cambios por filas cuando se multiplican por la izquierda, sus inversas consistirán en deshacer dichos cambios:

$$M_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix},$$

$$M_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -l_{32} & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & l_{32} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

Debido a que el efecto de multiplicar por la izquierda por la matriz M_1^{-1} consiste en sumar a las filas segunda y tercera múltiplos de la primera, se puede comprobar que la matriz $M_1^{-1}M_2^{-1}$ contiene en sus columnas por debajo de la diagonal los multiplicadores correspondientes:

$$M_1^{-1}M_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & l_{32} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} = L.$$

$(A^{(k)})$ tiene nulos sus elementos por debajo de la diagonal en las $k - 1$ primeras columnas) y M_k es la identidad pero con los multiplicadores por debajo de la diagonal en su columna k -ésima:

$$M_k = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & -l_{k+1k} & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & -l_{k+2k} & 0 & 1 & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -l_{nk} & 0 & \dots & 0 & 1 \end{bmatrix},$$

donde

$$l_{k+1k} = a_{k+1k}^{(k)} / a_{kk}^{(k)}, \quad l_{k+2k} = a_{k+2k}^{(k)} / a_{kk}^{(k)}, \quad \dots, \quad l_{nk} = a_{nk}^{(k)} / a_{kk}^{(k)}.$$

Llegamos pues a que

$$U = M_{n-1} \dots M_1 A, \quad b' = M_{n-1} \dots M_1 b,$$

y por tanto,

$$A = M_1^{-1} \dots M_{n-1}^{-1} U, \quad M_1^{-1} \dots M_{n-1}^{-1} b' = b. \quad (3)$$

Es posible comprobar que la inversa de las matrices de transformación elemental M_k es:

$$M_k^{-1} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots & \vdots & \vdots & & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & l_{k+1k} & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & l_{k+2k} & 0 & 1 & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & l_{nk} & 0 & \dots & 0 & 1 \end{bmatrix},$$

y también que

$$M_1^{-1} \dots M_{n-1}^{-1} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ l_{n1} & \dots & l_{nn-1} & 1 \end{bmatrix} = L,$$

es una matriz *triangular inferior con unos en la diagonal*, y con los multiplicadores elementales calculados en el proceso de eliminación gaussiana en sus lugares correspondientes. Por tanto, de (3) deducimos que

$$A = LU, \quad Lb' = b.$$

Para poder llevar a cabo el proceso de eliminación gaussiana descrito es necesario que los elementos diagonales $a_{kk}^{(k)}$, llamados *pivotes*, que vayamos obteniendo sean todos distintos de cero:

$$a_{kk}^{(k)} \neq 0, \text{ para cada } k = 1, \dots, n.$$

(de hecho, sólo hace falta que sea cierta la afirmación anterior hasta $k = n - 1$). Si en algún paso de la eliminación se obtiene algún $a_{kk}^{(k)}$ nulo, será preciso permutar la ecuación k -ésima con alguna posterior. En tal caso, el producto LU no es igual a la matriz A , sino a una matriz PA obtenida permutando las filas de A , esto es:

$$PA = LU, \quad Lb' = Pb. \quad (4)$$

La matriz P es una matriz de permutación, que se obtiene haciendo sobre la identidad los mismos cambios de filas que se hacen en el proceso de eliminación gaussiana. Observe que en P , los elementos de cada fila (o de cada columna) son todos cero salvo uno de ellos que es igual a uno. Por tanto, al hacer el producto $P^T P$, dichos unos sólo coincidirán unos con otros cuando una fila de P^T coincida con ella misma como columna de P , y por tanto,

$$P^T P = I, \text{ o } P^{-1} = P^T.$$

Por otro lado, si algún elemento diagonal $a_{kk}^{(k)}$ es nulo, y también son nulos todos los elementos que quedan por debajo de él mismo (es decir: $a_{k+1k}^{(k)} = \dots = a_{nk}^{(k)} = 0$), entonces que la matriz A no es invertible, pues todas las operaciones que hemos hecho hasta ahora no alteran el valor del determinante de A ($\det(A) = \dots = \det(A^{(k)})$) y $A^{(k)}$ tiene determinante nulo.

Podemos resumir todo lo anterior en el siguiente resultado:

Teorema 1. *Sea A una matriz $n \times n$ invertible y $b \in \mathbb{R}^n$. Entonces:*

- *El proceso de eliminación gaussiana con pivotaje reduce el sistema $Ax = b$ a otro equivalente $Ux = b'$, donde U es una matriz triangular superior.*
- *Existe una matriz P de permutación de modo que $PA = LU$, donde L es una matriz triangular inferior con unos en la diagonal.*
- *El vector b' se puede obtener resolviendo el sistema $Lb' = Pb$.*
- *El costo de obtener la factorización $PA = LU$ es de*

$$\frac{n^3 - n}{3} = O\left(\frac{n^3}{3}\right) \text{ flops,}$$

y el costo total de resolver el sistema $Ax = b$ es de

$$\frac{n^3 + 3n^2 - n}{3} = O\left(\frac{n^3}{3}\right) \text{ flops,}$$

entendiéndose por flop multiplicación o división.

Conviene aclarar el último enunciado. Observe que una vez obtenida la factorización $A = LU$ (o $PA = LU$) de la matriz A , para resolver el sistema $Ax = b$, nos basta con resolver $Ux = b'$. En vista de (4), para obtener b' basta resolver el sistema

$$Lb' = b,$$

(en caso de que se precisen permutaciones, este sistema quedará $Lb' = Pb$, pero el producto Pb no conlleva multiplicaciones, pues se trata de reordenar las componentes del vector b). Dado que L es triangular inferior con diagonal de unos, la resolución de $Lb' = b$ se realiza con el siguiente algoritmo, conocido como de *sustitución progresiva*:

$$\begin{aligned} b'_1 &= b_1, \\ b'_j &= b_j - (l_{j1}b'_1 + \cdots + l_{j,j-1}b'_{j-1}), \quad \text{para } j = 2, \dots, n. \end{aligned}$$

Su costo en *flops*, es como vemos,

$$0 + 1 + 2 + \cdots + (n - 1) = \frac{n(n - 1)}{2} \text{ flops.}$$

Una vez obtenido b' , el algoritmo de sustitución regresiva (2) requiere una división más en cada paso (dividir por el elemento diagonal u_{jj}). Por tanto, su costo es

$$1 + 2 + 3 + \cdots + n = \frac{n(n + 1)}{2} \text{ flops.}$$

Luego, una vez obtenida la factorización LU de A el costo de obtener x es n^2 *flops*.

El costo de obtener la factorización LU de A también es fácil de deducir. Basta contar el número de *flops* que se hacen en el primer paso de la eliminación gaussiana:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix}.$$

Vemos que se deben calcular los $n - 1$ multiplicadores

$$l_{21} = a_{21}/a_{11}, \dots, l_{n1} = a_{n1}/a_{11},$$

(que son $n - 1$ *flops*, y a continuación, para cada $j = 2, \dots, n$, debemos escalar:

$$[a_{12}, \dots, a_{1n}] \rightarrow [l_{j1} \times a_{12}, \dots, l_{j1} \times a_{1n}],$$

(que son $n - 1$ *flops*), para restar el resultado a la fila j -ésima de A , contando desde la segunda columna hasta n . Como este proceso lo debemos hacer para $n - 1$ filas, el resultado es que el primer paso de la eliminación gaussiana requiere

$$(n - 1) + (n - 1)^2 \text{ flops.}$$

El segundo paso será repetir el primero con la submatriz de orden $n - 1$

$$\begin{bmatrix} a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \ddots & \vdots \\ a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{bmatrix}.$$

Luego costará $(n - 2) + (n - 2)^2$ flops. El tercer paso, el cuarto, etc, consisten en repetir este proceso con matrices de orden más pequeño, con lo que vemos que el coste de reducir A a matriz triangular superior es

$$\sum_{j=1}^{n-1} (j^2 + j) = \sum_{j=1}^{n-1} j^2 + \sum_{j=1}^{n-1} j = \frac{(2n - 1)n(n - 1)}{6} + \frac{n(n - 1)}{2} = \frac{n^3 - n}{3} = O\left(\frac{n^3}{3}\right).$$

Debemos añadir que el proceso de intercambio de filas es también significativo desde el punto de la estabilidad numérica. En general, la eliminación gaussiana es inestable numéricamente si los pivotes son muy pequeños o equivalentemente, los multiplicadores son muy grandes. Para evitar esta situación se introducen las estrategias de pivotaje parcial y de pivotaje total:

- Pivotaje parcial: En el paso k -ésimo de la eliminación gaussiana, se determina

$$c_k = \max\{|a_{ik}^{(k)}| : k \leq i \leq n\}, \quad (5)$$

y se elige el menor índice i_k donde se alcanza el máximo. Posteriormente, se intercambian las filas k e i_k y se realiza la eliminación. Conviene observar que en este caso los multiplicadores verifican $|l_{ik}| \leq 1$, para todo $i = k + 1, \dots, n$.

- Pivotaje total: En el paso k -ésimo de la eliminación gaussiana, se determina

$$c_k = \max\{|a_{ij}^{(k)}| : k \leq i, j \leq n\},$$

y se elige un par de índices (i_k, j_k) donde se alcanza el máximo. Posteriormente, se intercambian filas y columnas para que el par (i_k, j_k) ocupe la posición diagonal (k, k) y se realiza la eliminación.

Los resultados teóricos indican que el pivotaje total es mejor que el parcial desde el punto de vista de la propagación de errores de redondeo. Sin embargo, para la mayoría de las matrices que aparecen en la práctica, el comportamiento del pivotaje parcial es más que aceptable. Puesto que el coste computacional del pivotaje total es muy superior al parcial, es este último sobre el que suelen diseñarse los algoritmos de eliminación gaussiana.

En resumen, un algoritmo de amplia aplicabilidad y estable para la eliminación gaussiana debe incorporar estrategias de pivotaje.

Comandos de Matlab. Para resolver el sistema $Ax = b$ en MATLAB, basta ejecutar $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$. Si la matriz A no tiene características especiales, el operador \backslash resuelve el sistema $Ax = b$ mediante eliminación gaussiana con pivotaje parcial de filas. El comando `lu`, proporciona la descomposición LU de una matriz, llevada a cabo por eliminación gaussiana con pivotaje parcial de filas, esto es $PA = LU$. Conviene señalar lo siguiente:

- El comando `[L,U]=lu(A)`, de la descomposición $PA = LU$ devuelve en `U` la matriz U , y en `L` la matriz $P^T L$ (teniendo en cuenta que $P^{-1} = P^T$, de $PA = LU$ se obtiene $A = P^T LU$). Luego, en general, `[L,U]=lu(A)` no devuelve en `L` una matriz triangular inferior.
- El comando `[L,U,P]=lu(A)` devuelve las tres matrices de la descomposición $PA = LU$.

Otros aspectos a destacar. En general, las rutinas o procedimientos para resolver un sistema lineal $Ax = b$, actúan como hemos señalado anteriormente:

- Se factoriza $PA = LU$.
- Se resuelve $Lb' = Pb$
- Se resuelve $Ux = b'$.

En otros lenguajes de programación (FORTRAN, C, etc) las rutinas para resolver sistemas lineales sobrescriben A con $LU = PA$, de la forma

$$\begin{bmatrix} u_{11} & \dots & u_{1n-1} & u_{1n} \\ l_{21} & \ddots & \vdots & \vdots \\ \vdots & \ddots & u_{n-1n-1} & u_{n-1n} \\ l_{n1} & \dots & l_{n-1n} & u_{nn} \end{bmatrix}.$$

(los unos de la diagonal de L no se guardan), pues en general (a diferencia de MATLAB) los argumentos A , b y x se pasan por referencia y no por valor. Asimismo, proporcionan un vector entero (cuyos valores son enteros) con los cambios de filas que se han llevado a cabo. El propósito es que si se debe volver a resolver otro sistema $Ay = c$ con la misma matriz de coeficientes, no sea necesario factorizar de nuevo la matriz (como hemos visto, éste es el proceso más costoso: $O(n^3/3)$ flops, frente a $O(n^2/3)$ flops que cuesta resolver $Lb' = b$, $Ux = b$).

Si en MATLAB se nos presentase esta necesidad de resolver varios sistemas con la misma matriz de coeficientes, debemos ejecutar primero `[L,U]=lu(A)`, y, posteriormente, para cada segundo miembro b , ejecutar `x=U\(L\b)`.

Normas matriciales. Para el análisis matemático de los sistemas de ecuaciones lineales, es útil el manejo de normas matriciales (las cuales juegan un papel similar al concepto de módulo en los números reales). La principal norma matricial es, sin duda, la norma euclídea: para una matriz A $m \times n$ se define como

$$\|A\| := \max \left\{ \frac{\|Ax\|}{\|x\|} : x \in \mathbb{R}^n, x \neq 0 \right\} = \max \{ \|Ax\| : \|x\| = 1 \},$$

donde para $x = [x_1, \dots, x_n]^T \in \mathbb{R}^n$, $\|x\| = (x_1^2 + \dots + x_n^2)^{1/2}$ es la norma euclídea.

Utilizando resultados sobre diagonalización de matrices, se puede mostrar una expresión alternativa de la norma euclídea matricial: puesto que la matriz $A^T A$ es simétrica y semidefinida positiva, es ortogonalmente semejante a una matriz diagonal, siendo todos sus autovalores reales no negativos. Entonces, se tiene que:

$$\|A\| = \rho(A^T A)^{1/2},$$

donde $\rho(\cdot)$ denota el radio espectral de la matriz.

A continuación, enumeramos ciertas propiedades de esta norma:

- $\|I_n\| = 1$, siendo I_n la matriz identidad de orden n ,
- $\|Ax\| \leq \|A\| \|x\|$, para toda matriz A y todo vector x ,
- $\|AB\| \leq \|A\| \|B\|$, para cualquier par de matrices A y B .

Condicionamiento de un sistema de ecuaciones lineales. El condicionamiento de un sistema de ecuaciones lineales $Ax = b$ puede enfocarse respecto a perturbaciones en el vector b o bien respecto a perturbaciones en la matriz A . En ambos casos, la misma expresión de la norma matricial de A sirve eficazmente como número de condición.

Consideremos una perturbación $b + \delta b$ en el segundo miembro de $Ax = b$, donde A es invertible y x es la solución del sistema ($Ax = b$). Sea $x + \delta x$ la solución del sistema perturbado, esto es, $A(x + \delta x) = b + \delta b$. Si restamos, obtenemos $A\delta x = \delta b$. Tomando normas, deducimos que

$$b = Ax \Rightarrow \|b\| \leq \|A\| \|x\| \Rightarrow \frac{1}{\|x\|} \leq \frac{\|A\|}{\|b\|}, \text{ y}$$

$$A\delta x = \delta b \Rightarrow \delta x = A^{-1}\delta b \Rightarrow \|\delta x\| \leq \|A^{-1}\| \|\delta b\|.$$

Por tanto,

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}.$$

A tenor de esta acotación de los errores relativos, es lógico definir *el número de condición de una matriz invertible A* como

$$\text{cond}(A) = \|A^{-1}\| \|A\|.$$

Puesto que $1 = \|I_n\| = \|A^{-1}A\| \leq \|A^{-1}\| \|A\|$, se tiene que $\text{cond}(A) \geq 1$.

Cuando se consideran perturbaciones (invertibles) de la matriz A en el sistema $Ax = b$, el número $\text{cond}(A)$ sigue siendo una excelente herramienta para medir el condicionamiento del problema, y se tiene una fórmula similar para los correspondientes errores relativos.

PROBLEMAS SOBREDETERMINADOS: SOLUCIÓN DE MÍNIMOS CUADRADOS.

Introducción. Hay muchas situaciones donde se plantea la obtención de un cierto modelo matemático lineal que ajuste a un conjunto de datos dados. Esto conduce usualmente a la resolución de un sistema de ecuaciones lineales con más ecuaciones que incógnitas, o *problemas sobredeterminados*, que casi siempre resultan ser incompatibles. Para dichos sistemas se introduce un concepto nuevo de solución¹, denominado *solución en el sentido de los mínimos cuadrados*, buscando vectores que minimicen la norma euclídea del correspondiente vector residual.

¹Este concepto coincide con el usual cuando el sistema es compatible.

Problemas sobredeterminados. Cuando un sistema lineal tiene más ecuaciones que incógnitas es fácil que sea incompatible, esto es, que no posea solución. Por ejemplo, si aplicamos el método de eliminación en el sistema

$$\left\{ \begin{array}{l} x_1 + 2x_2 = 3, \\ 2x_1 + 3x_2 = 5, \\ x_1 + 3x_2 = 2, \end{array} \right\} \xrightarrow{\begin{array}{l} e_2 \leftarrow e_2 - 2e_1 \\ e_3 \leftarrow e_3 - e_1 \end{array}} \left\{ \begin{array}{l} x_1 + 2x_2 = 3, \\ -x_2 = -1, \\ x_2 = -1, \end{array} \right\}$$

vemos que las dos últimas ecuaciones no pueden ser ambas ciertas, luego no hay solución posible. No obstante, si en MATLAB introducimos la matriz A y el vector b :

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 1 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 5 \\ 2 \end{bmatrix}, \quad \text{y ejecutamos } \mathbf{x}=\mathbf{A}\backslash\mathbf{b}, \text{ obtenemos:}$$

$$\mathbf{x} = \begin{array}{l} 3.0000 \\ -0.2727 \end{array}$$

En esta sección veremos qué tipo de resultado es éste, cómo se obtiene y cuánto cuesta.

Dada una matriz A real de orden $m \times n$ y un vector $b \in \mathbb{R}^m$, si $m \geq n$ se dice que el sistema $Ax = b$ es sobredeterminado. En la práctica es improbable que este sistema sea compatible. Por ello, introducimos un nuevo concepto de solución: se dice que $\tilde{x} \in \mathbb{R}^n$ es una *solución en el sentido de los mínimos cuadrados* del sistema $Ax = b$ si se verifica que

$$\|b - A\tilde{x}\| \leq \|b - Ax\|, \quad \text{para todo } x \in \mathbb{R}^n,$$

o, equivalentemente, si \tilde{x} es un mínimo de la función real de n variables,

$$x \in \mathbb{R}^n \mapsto f(x) = \sum_{j=1}^n (b_j - (Ax)_j)^2. \quad (6)$$

De hecho, la solución proporcionada por MATLAB en el ejemplo mostrado es la solución de mínimos cuadrados del sistema incompatible $Ax = b$.

Desde un punto de vista geométrico, estamos buscando la mejor aproximación en norma euclídea del vector b al subespacio vectorial $\text{col}(A)$ generado por las columnas de A . El teorema de la mejor aproximación establece que el vector de $\text{col}(A)$ que dista menos de b es la proyección ortogonal de b sobre $\text{col}(A)$: $P_{\text{col}(A)}b$.

El sistema $Ax = P_{\text{col}(A)}b$ siempre es compatible², y por lo tanto siempre existe la solución de mínimos cuadrados.

Recordemos que si restamos a un vector b su proyección ortogonal $P_V b$ sobre un subespacio V , el resultado $b - P_V b$ es ortogonal al subespacio V : $b - P_V b \perp V$. En particular, Si \tilde{x} es solución de mínimos cuadrados, entonces tenemos que

$$A\tilde{x} = P_{\text{col}(A)}b \Leftrightarrow b - A\tilde{x} \perp \text{col}(A) \Leftrightarrow A^T(b - A\tilde{x}) = 0 \Leftrightarrow A^T A\tilde{x} = A^T b.$$

²Esto es debido a que $P_{\text{col}(A)}b$ siempre está en $\text{col}(A)$.

Por último, comentemos que, si A no tiene rango máximo siempre existen vectores $x \in \mathbb{R}^n$ no nulos tales que $Ax = 0$. En este caso, si \tilde{x} es solución de mínimos cuadrados también lo son $\tilde{x} + x$, pues³ $A(\tilde{x} + x) = A\tilde{x} = P_{\text{col}(A)}b$. En cambio, si A es una matriz $m \times n$, con $m > n$ y $\text{rg}(A) = n$, la solución de mínimos cuadrados sí es única.

Resumimos todo lo anterior en el siguiente resultado:

Teorema 2. (*Ecuaciones normales de Gauss*). Sea A una matriz real $m \times n$ y $b \in \mathbb{R}^m$. Las siguientes afirmaciones son equivalentes:

- \tilde{x} es una solución en el sentido de los mínimos cuadrados del sistema $Ax = b$.
- \tilde{x} es solución del sistema $A^T Ax = A^T b$ (ecuaciones normales de Gauss).
- $b - A\tilde{x}$ es ortogonal a $\text{col}(A)$.

Además, si el rango de A es máximo, $\text{rg}(A) = n$, entonces la solución de mínimos cuadrados es única.

Nota 1. A las ecuaciones normales de Gauss también se llega sin necesidad de argumentos geométricos. De hecho, la solución de mínimos cuadrados \tilde{x} es un mínimo de la función f definida en (6) y por tanto, el gradiente de esta función debe anularse en \tilde{x} :

$$f(x) = \|b - Ax\|^2 = (b - Ax)^T (b - Ax) = x^T A^T Ax - 2x^T A^T b + b^T b \Rightarrow \nabla f(x) = 2(A^T Ax - A^T b).$$

Las ecuaciones normales de Gauss están peor condicionadas que otros sistemas que también permiten encontrar la solución de mínimos cuadrados, por lo que no conviene usarlas en los problemas de mínimos cuadrados. En realidad, las técnicas eficientes para la resolución de los problemas de mínimos cuadrados suelen basarse en transformar las ecuaciones normales mediante ciertas factorizaciones matriciales que recordamos a continuación.

Descomposición QR de una matriz. Recordemos el método de ortogonalización de Gram-Schmidt:

Teorema 3. (*Método de ortogonalización de Gram-Schmidt*). Sea $\{a_1, a_2, \dots, a_n\}$ un conjunto de vectores linealmente independiente de \mathbb{R}^m . Consideremos los vectores $\{v_1, v_2, \dots, v_n\}$ obtenidos con el siguiente procedimiento:

- Definimos $v_1 = a_1$.
- Para $j = 2, 3, \dots, n$, construimos

$$v_j = a_j - \alpha_{1j}v_1 - \alpha_{2j}v_2 - \dots - \alpha_{j-1,j}v_{j-1}, \text{ donde } \alpha_{ij} = \frac{v_i^T a_j}{v_i^T v_i}, \quad i = 1, \dots, j-1. \quad (7)$$

Entonces, se cumple que $v_i^T v_j = 0$ si $i \neq j$, y además $\text{lin}\{a_1, a_2, \dots, a_j\} = \text{lin}\{v_1, v_2, \dots, v_j\}$, para $j = 1, 2, \dots, n$.

³Observe que el sistema $Ax = 0$ es compatible indeterminado y por tanto existirán infinitas soluciones de mínimos cuadrados.

De la ecuación (7) podemos despejar a_j :

$$a_j = v_j + \alpha_{1j}v_1 + \alpha_{2j}v_2 + \cdots + \alpha_{j-1j}v_{j-1}.$$

Si $A = [a_1, \dots, a_n]$ es la matriz $m \times n$ cuyas columnas son los vectores a_j , y $V = [v_1, \dots, v_n]$ es la matriz cuyas columnas son los vectores v_j , tenemos $A = VU$, donde

$$U = \begin{bmatrix} 1 & \alpha_{12} & \cdots & \alpha_{1n-1} & \alpha_{1n} \\ & \ddots & \ddots & & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & 1 & \alpha_{n-1n} \\ & & & & 1 \end{bmatrix} \text{ es una matriz triangular superior,}$$

y las columnas de V son ortogonales dos a dos. Si llamamos $D = \text{diag}(\|v_1\|, \dots, \|v_n\|)$, tenemos que

$$A = VU = VD^{-1}DU = \underbrace{VD^{-1}}_{Q_1} \underbrace{DU}_{R_1} = Q_1R_1.$$

Esta expresión de A como el producto Q_1R_1 se conoce como descomposición QR *reducida* de la matriz. Notemos que R_1 es una matriz $n \times n$ triangular superior y Q_1 es una matriz $m \times n$ que cumple⁴ $Q_1^T Q_1 = I_n$.

En la práctica, (ya veremos por qué en la siguiente sección cuando estudiemos las transformaciones de Householder) siempre se completa Q_1 hasta obtener $Q = [Q_1|Q_2]$, donde Q_2 es una matriz $m \times (m - n)$ cuyas columnas son una base ortonormal del ortogonal de $\text{col}(Q_1)$, de modo que $Q = [Q_1|Q_2]$ es matriz $m \times m$ ortogonal⁵; y a su vez se amplía R_1 hasta completar una matriz R añadiéndole $m - n$ filas de ceros, de manera que tenemos:

$$QR = [Q_1|Q_2] \begin{bmatrix} R_1 \\ O \end{bmatrix} = Q_1R_1 + Q_2O = A.$$

Resumimos lo expuesto en el siguiente resultado:

Teorema 4. (*Descomposición QR*) Sea A una matriz real $m \times n$, de rango $n \leq m$. Entonces, podemos factorizar la matriz en la forma:

$$A = QR,$$

donde Q es una matriz $m \times m$ ortogonal (esto es, $Q^{-1} = Q^T$ o equivalentemente, las columnas de Q son ortonormales), y R es una matriz $m \times n$, de rango n , cuyas $m - n$ últimas filas son nulas, y sus n primeras filas forman una matriz cuadrada triangular superior.

Si se conoce una descomposición QR de la matriz A , entonces las soluciones en el sentido de los mínimos cuadrados de $Ax = b$ se pueden calcular resolviendo (también en el sentido de los mínimos cuadrados) el sistema⁶

$$Rx = Q^T b. \tag{8}$$

⁴Esto es consecuencia de ser las columnas de Q_1 ortogonales y unitarias.

⁵Es decir, se cumple $Q^{-1} = Q^T$ puesto que $Q^T Q = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} [Q_1|Q_2] = \begin{bmatrix} Q_1^T Q_1 & Q_1^T Q_2 \\ Q_2^T Q_1 & Q_2^T Q_2 \end{bmatrix} = \begin{bmatrix} I & O \\ O & I \end{bmatrix}$.

⁶Ello es debido a que $A^T A \tilde{x} = A^T b \Rightarrow R^T Q^T Q R \tilde{x} = R^T Q^T b \Rightarrow R^T R \tilde{x} = R^T Q^T b$.

Si usamos que $Q = [Q_1|Q_2]$, $R = \begin{bmatrix} R_1 \\ O \end{bmatrix}$, entonces el sistema (8) se escribe como

$$\begin{bmatrix} R_1 \\ O \end{bmatrix} x = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} b = \begin{bmatrix} Q_1^T b \\ Q_2^T b \end{bmatrix}.$$

La solución de mínimos cuadrados de este sistema es

$$\tilde{x} = R_1^{-1} Q_1^T b, \text{ pues } Q^T b - R\tilde{x} = \begin{bmatrix} 0 \\ Q_2^T b \end{bmatrix} \perp \text{col} \begin{bmatrix} R_1 \\ O \end{bmatrix} = \text{col}(R).$$

El uso de la descomposición QR para resolver, como hemos mencionado, por mínimos cuadrados el sistema $Ax = b$ presenta ventajas desde el punto de vista numérico: observe que el procedimiento llevado a cabo involucra el producto por matrices ortogonales y este proceso conserva la norma matricial y el número de condición.

Método de Householder para la descomposición QR . Una simetría especular es una transformación de \mathbb{R}^m en sí mismo que a cada punto le hace corresponder su simétrico respecto de un hiperplano V (subespacio de \mathbb{R}^m de dimensión $m - 1$). Si una ecuación implícita de dicho hiperplano es $v^T x = 0$ (para un cierto vector no nulo v de \mathbb{R}^m), la simetría especular en torno a V es⁷

$$x \mapsto \left(I - 2 \frac{vv^T}{v^T v} \right) x.$$

La matriz $I - 2 \frac{vv^T}{v^T v}$ de una simetría especular es simétrica y ortogonal, pues

$$\begin{aligned} \left(I - 2 \frac{vv^T}{v^T v} \right)^T \left(I - 2 \frac{vv^T}{v^T v} \right) &= \left(I - 2 \frac{vv^T}{v^T v} \right) \left(I - 2 \frac{vv^T}{v^T v} \right) = \\ &= I - 4 \frac{vv^T}{v^T v} + 4 \frac{(vv^T)(vv^T)}{(v^T v)^2} = I - 4 \frac{vv^T}{v^T v} + 4 \frac{v(v^T v)v^T}{(v^T v)^2} = I. \end{aligned}$$

Las transformaciones de Householder son un caso particular de simetrías especulares, que sirven para transformar un vector cualquiera $x \in \mathbb{R}^m$, en uno proporcional al primer vector coordenado $e_1 = [1, 0, \dots, 0]^T$, que debe ser por tanto $\pm \|x\| e_1$. La correspondiente matriz H la presentamos en el siguiente resultado:

Teorema 5. Sea $x = [x_1, \dots, x_m]^T \in \mathbb{R}^m$ ($m > 1$), $x \neq 0$. Sea $\delta = \begin{cases} \text{sign}(x_1), & \text{si } x_1 \neq 0 \\ 1, & \text{si } x_1 = 0 \end{cases}$, y definamos $v = x + \delta \|x\| e_1$. Entonces, la matriz

$$H = I - 2 \frac{vv^T}{v^T v},$$

verifica:

$$Hx = \begin{bmatrix} -\delta \|x\| \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

⁷Recuerde que $V^\perp = \text{lin}(v)$ y que $\text{lin}(v)^\perp = V$.

Pasemos a describir cómo se puede obtener la factorización QR de una matriz A $m \times n$ ($m \geq n > 1$) con rango $\text{rg}(A) = n$.

Consideramos la primera columna de la matriz A :

$$b_1 = \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix},$$

Aplicando el teorema anterior, podemos determinar una matriz de Householder H_1 (de orden m) tal que la primera columna de $H_1 A$ tenga las componentes nulas por debajo de la diagonal:

$$H_1 A = A^{(2)} = \begin{bmatrix} a_{11}^{(2)} & a_{12}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(2)} & \cdots & a_{mn}^{(2)} \end{bmatrix}.$$

Denotemos $Q^{(1)} = H_1$. Ahora nos fijamos en la segunda columna de la matriz $A^{(2)}$ a partir de la diagonal, en concreto en el vector

$$b_2 = \begin{bmatrix} a_{22}^{(2)} \\ \vdots \\ a_{m2}^{(2)} \end{bmatrix}.$$

Volvemos a aplicar el teorema anterior y obtenemos una matriz de Householder \widetilde{H}_2 (de orden $m - 1$) de modo que $\widetilde{H}_2 b_2$ tenga las componentes nulas por debajo de la primera. Por tanto si definimos

$$H_2 = \left[\begin{array}{c|c} 1 & 0^T \\ \hline 0 & \widetilde{H}_2 \end{array} \right], \quad \text{tenemos que } H_2 A^{(2)} = A^{(3)} = \begin{bmatrix} a_{11}^{(2)} & a_{12}^{(2)} & a_{13}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(3)} & a_{23}^{(3)} & \cdots & a_{2n}^{(3)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{m3}^{(3)} & \cdots & a_{mn}^{(3)} \end{bmatrix},$$

y denotamos

$$Q^{(2)} = H_2.$$

Si continuamos el proceso sucesivamente, obtenemos el siguiente resultado:

Teorema 6. Si A es una matriz $m \times n$ con $1 < \text{rg}(A) = n \leq m$, el método anterior genera una matriz $m \times m$ ortogonal $Q = \begin{cases} H_1 \dots H_{m-1}, & \text{si } m = n \\ H_1 \dots H_n, & \text{si } m > n \end{cases}$, y una matriz $m \times n$, $R = A^{(m)}$ cuyas $m - n$ últimas filas son nulas y cuyas n primeras forman una matriz triangular superior de rango n , y tales que

$$A = QR.$$

El coste computacional del método de Householder descrito anteriormente requiere

$$2m^2n + \frac{n^3}{3} \text{ flops}, \quad (9)$$

es decir, para matrices cuadradas es aproximadamente el doble que el de efectuar la factorización LU de A con el método de eliminación gaussiana.

MATRICES DE RANGO DEFICIENTE.

Introducción. Anteriormente hemos tratado el caso de sistemas lineales $Ax = b$ donde A tiene más filas que columnas, pero de rango máximo. Tratamos ahora el caso en que el rango no es máximo. En este caso, la matriz A de coeficientes del sistema se dice que es *de rango deficiente*.

Un ejemplo de dicho sistema es por ejemplo,

$$\begin{bmatrix} 1 & 2 & 1 \\ 1 & 3 & 4 \\ 2 & 5 & 5 \\ 3 & 8 & 9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ -2 \\ 0 \end{bmatrix}.$$

Si procedemos a resolverlo mediante el método de eliminación de Gauss:

$$\left[\begin{array}{ccc|c} 1 & 2 & 1 & 1 \\ 1 & 3 & 4 & 3 \\ 2 & 5 & 5 & -2 \\ 3 & 8 & 9 & 0 \end{array} \right] \xrightarrow{\substack{f_2 \leftarrow f_2 - f_1 \\ f_3 \leftarrow f_3 - 2f_1 \\ f_4 \leftarrow f_4 - 3f_1}} \left[\begin{array}{ccc|c} 1 & 2 & 1 & 1 \\ 0 & 1 & 3 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & -3 \end{array} \right] \xrightarrow{\substack{f_3 \leftarrow f_3 - f_2 \\ f_4 \leftarrow f_4 - 2f_2}} \left[\begin{array}{ccc|c} 1 & 2 & 1 & 1 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 0 & -6 \\ 0 & 0 & 0 & -7 \end{array} \right],$$

vemos que el sistema es incompatible (las dos últimas ecuaciones son $0x_3 = -6$ y $0x_3 = -7$). Observamos también que el rango de la matriz es 2: la tercera columna es tres veces la segunda menos cinco veces la primera. Tenemos pues que el vector

$$c = \begin{bmatrix} -5 \\ 3 \\ -1 \end{bmatrix}, \text{ satisface } Ac = 0.$$

Si introducimos en MATLAB la matriz A y el vector b , y ejecutamos $\mathbf{x} = A \setminus \mathbf{b}$, obtenemos:

```

Warning: Rank deficient, rank = 2 tol = 9.8504e-15.
x =
    0
-0.9333
 0.8667

```

Observamos en primer lugar que MATLAB nos avisa de la matriz es de rango 2 (siendo más preciso, que una matriz \tilde{A} con $\|\tilde{A} - A\| / \|A\| \leq \text{tol} = 9.8504 \times 10^{-15}$ tiene rango 2). El vector x obtenido es una aproximación a una de las soluciones de mínimos cuadrados, ya que se

verifica que $A^T(b - Ax) = 0$ con un cierto error de redondeo. En efecto, ejecutando `A'*(b-A*x)`, MATLAB responde

```
ans =
      1.0e-13 *
      -0.0533
      -0.1421
      -0.1243
```

Notemos además que la solución x proporcionada por MATLAB es una de las muchas posibles, ya que cualquier múltiplo del vector c también cumple $A(\lambda c) = 0$.

En el caso matrices de rango deficiente, es estándar aceptar como solución la que se conoce como *solución óptima*, que se define como la solución x^* en el sentido de mínimos cuadrados de norma mínima, es decir, $\|x^*\| \leq \|\tilde{x}\|$ para toda solución de mínimos cuadrados \tilde{x} de $Ax = b$. En el ejemplo anterior, la solución óptima, aproximada a 5 dígitos decimales, es:

$$x^* = \begin{bmatrix} -0.5238 \\ -0.6190 \\ 0.7619 \end{bmatrix}.$$

Observemos que MATLAB no proporciona necesariamente la solución óptima en un problema de mínimos cuadrados.

El cálculo efectivo de la solución óptima pasa por la descomposición en valores singulares de la matriz A . Antes de ver cómo se calcula la solución óptima x^* , terminamos esta introducción con un resultado de caracterización de la misma:

Teorema 7. *Sea A matriz $m \times n$ con $\text{rg}(A) = r < n \leq m$ y $b \in \mathbb{R}^n$.*

- *Si \tilde{x} es una solución de mínimos cuadrados del problema $Ax = b$, entonces el conjunto de soluciones de mínimos cuadrados es*

$$\{y = \tilde{x} + z \mid z \in \text{Nul}(A)\}, \text{ donde } \text{Nul}(A) = \{z \in \mathbb{R}^n \mid Az = 0\}.$$

- *La solución óptima x^* es la única que satisface que $z^T x^* = 0$, para todo $z \in \text{Nul}(A)$.*

Descomposición en valores singulares (SVD). Si A es una matriz $m \times n$ ($m \geq n$) de rango r , la matriz $A^T A$ (que es simétrica y semidefinida positiva) tiene sus autovalores reales y no negativos:

$$\lambda_1 \geq \dots \geq \lambda_r > 0 = \lambda_{r+1} = \dots = \lambda_n.$$

Consideremos la correspondiente base de autovectores asociados de la matriz $A^T A$:

$$\{v_1, \dots, v_n\}, \text{ (es decir: } A^T A v_j = \lambda_j v_j, j = 1, \dots, n).$$

Esta base puede elegirse ortonormal (esto es: $v_j^T v_k = 0$ si $j \neq k$, y $\|v_j\| = 1$, $j = 1, \dots, n$).

- Los *valores singulares* de la matriz A se definen como:

$$\sigma_j = \sqrt{\lambda_j}, j = 1, \dots, r.$$

- Los *vectores singulares derechos* (o por la derecha) son v_1, \dots, v_n .
- Los *vectores singulares izquierdos* o por la izquierda son

$$u_1 = \frac{1}{\sigma_1} Av_1, \dots, u_r = \frac{1}{\sigma_r} Av_r.$$

(note que sólo incluimos los correspondientes a los autovalores no nulos). Puede comprobarse que $\{u_1, \dots, u_r\}$ es un sistema ortonormal en \mathbb{R}^m . Dicho sistema puede ampliarse hasta una base ortonormal de \mathbb{R}^m : $\{u_1, \dots, u_r, u_{r+1}, \dots, u_m\}$.

Definimos ahora las matrices:

$$U = [u_1, \dots, u_m]_{m \times m}, \quad V = [v_1, \dots, v_n]_{n \times n},$$

y la matriz:

$$\Sigma = \left[\begin{array}{cccc|ccc} \sigma_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & \ddots & \vdots & \vdots & & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \sigma_r & 0 & \dots & 0 \\ \hline 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{array} \right] = \left[\begin{array}{cc} \Sigma_1 & O \\ O & O \end{array} \right]_{m \times n}.$$

Entonces, se tiene que $AV = U\Sigma$, y por tanto, obtenemos la siguiente factorización de la matriz A (conocida como descomposición en valores singulares, abreviada SVD del inglés *singular value decomposition*):

$$A = U\Sigma V^T.$$

El siguiente teorema recoge la existencia y unicidad de la factorización SVD:

Teorema 8. *Sea A una matriz $m \times n$ con $m \geq n$, y de rango $r \leq n$. Entonces, existen dos matrices ortogonales U $m \times m$ y V $n \times n$, y otra matriz Σ $m \times n$ tales que*

$$A = U\Sigma V^T = U \left[\begin{array}{cccc|c} \Sigma_1 & O \\ O & O \end{array} \right] V^T, \quad \text{donde } \Sigma_1 = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \sigma_r \end{bmatrix},$$

con $\sigma_1 \geq \dots \geq \sigma_r > 0$. La matriz Σ está determinada de forma única. Los números σ_i son necesariamente los valores singulares de A (las raíces cuadradas de los autovalores no nulos de la matriz $A^T A$).

Si tenemos la SVD para una matriz A $m \times n$ de rango r :

$$A = U \left[\begin{array}{cccc|c} \Sigma_1 & O \\ O & O \end{array} \right] V^T, \quad \Sigma_1 = \text{diag}(\sigma_1, \dots, \sigma_r),$$

se denomina matriz *inversa generalizada de Moore-Penrose* o *pseudoinversa*) de A , a la matriz A^+ $n \times m$ dada por

$$A^+ = V \left[\begin{array}{c|c} \Sigma^+ & O \\ \hline O & O \end{array} \right] U^T, \text{ donde } \Sigma^+ = \Sigma_1^{-1} = \text{diag}(1/\sigma_1, \dots, 1/\sigma_r).$$

Si la matriz A es cuadrada y no singular, se verifica que $A^+ = A^{-1}$, lo cual justifica el nombre de pseudoinversa. Por otro lado si A es $m \times n$ con $\text{rg}(A) = n \leq m$, entonces $A^+ = (A^T A)^{-1} A^T$. Es decir, la pseudoinversa permite resolver las ecuaciones normales de Gauss, $A^T A \tilde{x} = A^T b$, cuando éstas tienen solución única. En el caso de que la solución no sea única se tiene el siguiente resultado:

Teorema 9. Sean A matriz $m \times n$ con ($m \geq n$), y $b \in \mathbb{R}^m$. Entonces el vector $x^* \in \mathbb{R}^n$ es la solución óptima del problema de mínimos cuadrados asociado al sistema $Ax = b$ si y sólo si $x^* = A^+ b$.

El método con el que hemos obtenido aquí la SVD no se debe emplear para su cálculo efectivo en el ordenador. Hay otros procedimientos más eficaces (véase por ejemplo Golub & Van Loan, *Matrix Computations (2nd Ed)*, John Hopkins University Press, London, 1989) que no serán tratados en este curso. Nos bastará con saber que el costo de computacional es de

$$4m^2n + 8mn^2 + 9n^3 \text{ flops.}$$

y que, en MATLAB, se puede calcular con el comando `svd`.

CASOS PARTICULARES RELEVANTES.

Matrices simétricas y definidas positivas. Recuerde que una matriz cuadrada A es simétrica si $A^T = A$, y que es definida positiva si

$$x^T Ax > 0 \text{ para todo } x \neq 0.$$

Toda matriz simétrica y definida positiva es invertible, pues si $x \neq 0$, el vector $Ax \neq 0$ al ser $x^T Ax > 0$.

Las matrices simétricas y definidas positivas son, con diferencia, las que más se presentan en las aplicaciones prácticas.

El primer resultado que veremos sobre estas matrices es que es posible efectuar en ellas la eliminación gaussiana sin pivotaje (esto es, que jamás en dicho proceso obtendremos un elemento diagonal nulo). Empecemos observando que una matriz A de orden n simétrica y definida positiva, tiene *todos* sus elementos diagonales *positivos* pues

$$a_{jj} = e_j^T A e_j > 0, \text{ donde } e_j \text{ es el } j\text{-ésimo vector coordenado de } \mathbb{R}^n.$$

Para ver que efectivamente podemos realizar la eliminación gaussiana sin pivotaje, escribamos A como

$$A = \left[\begin{array}{c|c} \alpha & b^T \\ \hline b & B \end{array} \right], \alpha \in \mathbb{R}, b \in \mathbb{R}^{n-1}, B \text{ matriz de orden } n-1.$$

Puesto que el elemento diagonal $\alpha > 0$, el vector de multiplicadores será

$$l = \frac{1}{\alpha}b, \text{ y tomando } M_1 = \left[\begin{array}{c|c} 1 & 0^T \\ \hline -l & I \end{array} \right],$$

tras efectuar el primer paso de la eliminación obtenemos

$$A^{(2)} = M_1 A = \left[\begin{array}{c|c} 1 & 0^T \\ \hline -\frac{1}{\alpha}b & I \end{array} \right] \left[\begin{array}{c|c} \alpha & b^T \\ \hline b & B \end{array} \right] = \left[\begin{array}{c|c} \alpha & b^T \\ \hline 0 & B - \frac{1}{\alpha}bb^T \end{array} \right]. \quad (10)$$

La submatriz $B - \frac{1}{\alpha}bb^T$ de $A^{(2)}$ es simétrica pues B lo es. Por otra parte, la matriz

$$M_1 A M_1^T = A^{(2)} M_1^T = \left[\begin{array}{c|c} \alpha & b^T \\ \hline 0 & B - \frac{1}{\alpha}bb^T \end{array} \right] \left[\begin{array}{c|c} 1 & -\frac{1}{\alpha}b^T \\ \hline 0 & I \end{array} \right] = \left[\begin{array}{c|c} \alpha & 0^T \\ \hline 0 & B - \frac{1}{\alpha}bb^T \end{array} \right],$$

es también simétrica, y además es definida positiva pues⁸:

$$\text{Si } x \neq 0 \Rightarrow y = M_1^T x \neq 0 \Rightarrow x^T M_1 A M_1^T x = y^T A^{(2)} y > 0.$$

En particular, podemos deducir que $B - \frac{1}{\alpha}bb^T$ es positiva:

$$\text{Si } x \in \mathbb{R}^{n-1}, x \neq 0 \Rightarrow x^T (B - \frac{1}{\alpha}bb^T)x = [0, x^T] \left[\begin{array}{c|c} \alpha & 0^T \\ \hline 0 & B - \frac{1}{\alpha}bb^T \end{array} \right] \begin{bmatrix} 0 \\ x \end{bmatrix} > 0.$$

Si repetimos el razonamiento que hemos hecho con A pero con la matriz $B - \frac{1}{\alpha}bb^T$ (que es asimismo simétrica y definida positiva), obtenemos el siguiente resultado:

Teorema 10. En una matriz simétrica y definida positiva es posible realizar la eliminación gaussiana sin pivotaje, siendo todos los pivotes (elementos diagonales) positivos.

Nota 2. Hemos visto que, la eliminación gaussiana sin pivotaje, cuando se lleva a cabo sobre matrices de tipo general, puede dar lugar a fenómenos de inestabilidad numérica, es decir de acumulación excesiva del error de redondeo. También hemos visto que esta inestabilidad numérica se corrige (salvo en excepciones patológicas) en el momento en que se hace pivotaje parcial de filas.

En el caso de matrices simétricas y definidas positivas, la eliminación gaussiana sin pivotaje es un algoritmo numéricamente estable (salvo excepciones patológicas) y por tanto, la misma confianza que tenemos en la eliminación gaussiana con pivotaje parcial de filas para matrices generales, la podemos depositar, en el caso de matrices simétricas y definidas positivas, en la eliminación gaussiana sin pivotaje. Véase por ejemplo, el Teorema 23.2 de N. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

Otro aspecto importante del caso de matrices simétricas y definidas positivas es que, en la práctica, no se lleva a cabo la factorización LU sobre ellas. En su lugar, se usa la de Cholesky, que pasamos a comentar.

⁸Note que M_1^T es invertible al ser triangular superior con unos en la diagonal.

Observemos primero que si $A = LU$, sacando factor común en U a sus elementos diagonales:

$$U = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & u_{nn} \end{bmatrix} = \begin{bmatrix} u_{11} & 0 & \dots & 0 \\ 0 & u_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & u_{nn} \end{bmatrix} \begin{bmatrix} 1 & \frac{u_{12}}{u_{11}} & \dots & \frac{u_{1n}}{u_{11}} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{u_{n-1 n}}{u_{n-1 n-1}} \\ 0 & \dots & 0 & 1 \end{bmatrix} = D\tilde{U},$$

podemos escribir $A = LD\tilde{U}$. Dado que A es simétrica, tenemos que

$$\left. \begin{array}{l} A = LD\tilde{U} \\ A^T = \tilde{U}^T D L^T \\ A = A^T \end{array} \right\} \Rightarrow LD\tilde{U} = \tilde{U}^T D L^T \Rightarrow \tilde{U} = L^T.$$

En definitiva:

$$A = LDL^T. \quad (11)$$

Puesto que $u_{jj} > 0$, para $j = 1, \dots, n$,

$$D = \begin{bmatrix} u_{11} & 0 & \dots & 0 \\ 0 & u_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & u_{nn} \end{bmatrix} \Rightarrow D^{1/2} = \begin{bmatrix} \sqrt{u_{11}} & 0 & \dots & 0 \\ 0 & \sqrt{u_{22}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \sqrt{u_{nn}} \end{bmatrix}.$$

Por último, si denotamos $R = D^{1/2}L^T$, de (11) se obtiene la denominada *descomposición de Cholesky* de A :

$$A = R^T R,$$

Además, tal y como argumentamos en (10), tanto A como las sucesivas submatrices que se obtienen en el proceso de eliminación gaussiana son simétricas. Esto hace que sea posible hacer operaciones solamente sobre la parte triangular superior de A , con lo que el costo de la eliminación gaussiana se reduce a la mitad.

Resumimos estas propiedades en el siguiente resultado:

Teorema 11. (Descomposición de Cholesky) Si A es una matriz $n \times n$ simétrica y definida positiva, entonces:

- Existe una matriz R , triangular superior y con diagonal de elementos positivos tal que $A = R^T R$.
- La matriz R se puede obtener con un costo de $O(n^3/6)$ flops.

Nota 3. El comando de MATLAB `chol` devuelve la descomposición de Cholesky de A .

ÓRDENES DE MATLAB

En general, el comando `\` de MATLAB suele ser suficiente para la resolución de un sistema lineal. Suele distinguir que una matriz es simétrica y definida positiva y resolver el sistema en menor tiempo. En problemas de mínimos cuadrados, mientras la matriz de coeficientes tenga rango máximo, el comando `\` de MATLAB resuelve el problema (utilizando una descomposición QR de la matriz). Cuando la matriz de coeficientes tiene rango deficiente, se hace imprescindible resolver mediante la descomposición en valores singulares.

El comando `\` de MATLAB no siempre aprecia el carácter disperso de una matriz, a no ser que esté ya almacenada en formato triada.

Por último, resumimos algunos comandos útiles:

<code>\</code>	para resolver sistemas
<code>lu</code>	descomposición LU
<code>qr</code>	descomposición QR
<code>svd</code>	descomposición en valores singulares
<code>chol</code>	descomposición de Cholesky
<code>tril</code>	parte triangular inferior de una matriz
<code>triu</code>	parte triangular superior de una matriz
<code>cond</code>	número de condición
<code>sparse</code>	almacena en formato triada
<code>full</code>	transforma de formato triada a estándar
<code>spdiags</code>	crea matrices banda
<code>eye</code>	matriz identidad
<code>speye</code>	matriz identidad en formato triada

CUESTIONES

Ejercicio 1. Obtener la factorización LU de la matriz

$$A = \begin{bmatrix} 2 & 4 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 2 & 3 & 2 & 1 \\ 4 & 6 & 1 & 2 \end{bmatrix}.$$

Una vez obtenida la factorización, resuelva el sistema $Ax = b$ donde $b = [4, 2, 5, 7]^T$.

Ejercicio 2. Calcular la factorización LU de la matriz

$$A = \begin{bmatrix} 1 & 0 & \alpha \\ -1 & 0 & 0 \\ -1 & 1 & 0 \end{bmatrix}.$$

¿Podría estimar el rango de A para todos los valores de α ? Resuelva el sistema $Ax = b$ donde $b = [-1, 0, 1]^T$.

Ejercicio 3. Deducir que el costo en *flops* de calcular la inversa de una matriz A de orden n es n^3 *flops*. Recuerde que $\sum_{j=1}^n j^2 = \frac{(2n+1)(n+1)n}{6}$.

Ejercicio 4. Pruebe que si A es una matriz $m \times n$, entonces $\|A\| = \sqrt{\rho(A^T A)}$.

Ejercicio 5. Determinar la norma euclídea de la matriz

$$\begin{bmatrix} 2 & -1 \\ 1 & 1 \end{bmatrix}.$$

Ejercicio 6. Probar que si A es una matriz cuadrada de orden n invertible, entonces

$$\text{cond}(A) = \sqrt{\frac{\lambda_n(A^T A)}{\lambda_1(A^T A)}},$$

donde $\lambda_1(A^T A)$ y $\lambda_n(A^T A)$ son, respectivamente, el menor y el mayor de los autovalores de la matriz $A^T A$.

Ejercicio 7. Calcular el número de condición de una matriz invertible cuyas columnas sean ortogonales entre sí.

Ejercicio 8. Determinar la solución de mínimos cuadrados, vía las ecuaciones normales, de los sistemas sobredeterminados

$$\left\{ \begin{array}{l} x_1 + x_2 = 0 \\ -x_1 + x_2 = 1 \\ x_1 + x_3 = 1 \\ x_1 + x_2 = 1 \end{array} \right\}, \quad \left\{ \begin{array}{l} 3x_1 - x_2 = 0 \\ 4x_1 + 2x_2 = 2 \\ x_2 = 1 \end{array} \right\}.$$

Ejercicio 9. Aplicando el método de Gram-Schmidt, obtener una factorización QR de las matrices

$$A = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 1 & 4 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 1 & 2 & 2 \\ 2 & 0 & -1 \end{bmatrix}.$$

Ejercicio 10. Probar que los autovalores de toda matriz ortogonal son de módulo unidad. Demostrar que $\lambda = -1$ es siempre un autovalor de cualquier matriz de Householder. Interpretar geoméricamente este hecho, para las matrices de orden dos.

Ejercicio 11. Utilizando transformaciones de Householder, obtener una factorización QR de las matrices

$$A = \begin{bmatrix} 1 & -1 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 2 & 1 \end{bmatrix}.$$

Ejercicio 12. Si A es una matriz $m \times n$, obtener el costo en *flops* de su factorización QR reducida por el método de Gram-Schmidt. Comprobar que el costo de la factorización QR por el método de Householder es el que se indica en (9).

Ejercicio 13. Sea A una matriz de orden n invertible. Demuestre las siguientes propiedades del número de condición.

1. $\text{cond}(A) \geq 1$.
2. $\text{cond}(A^{-1}) = \text{cond}(A)$.
3. $\text{cond}(\alpha A) = \text{cond}(A)$, $\forall \alpha \in \mathbb{R}$.
4. $\text{cond}(A) = \frac{\sigma_1}{\sigma_n}$, donde σ_1 es el máximo valor singular de A y σ_n el mínimo.
5. $\|QA\| = \|AQ\| = \|A\|$, para cualquier Q ortogonal. Deduzca que $\text{cond}(QA) = \text{cond}(AQ) = \text{cond}(A)$.

Ejercicio 14. Obtener la descomposición en valores singulares de las matrices:

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ -1 & 1 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0.0 & -1.6 & 0.6 \\ 0.0 & 1.2 & 0.8 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \end{bmatrix}.$$

Ejercicio 15. Aplicar el resultado del ejercicio anterior para encontrar la solución óptima del problema de mínimos cuadrados $Ax = b$ con $b = [1, 2, 3, 4]^T$.

Ejercicio 16. Repetir los dos ejercicios anteriores con

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}.$$

Ejercicio 17. Considere la matriz

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

- Calcule su descomposición en valores singulares.
- Calcule la solución óptima del sistema $Ax = b$ con $b = [1 \ 2 \ 3]^T$.
- Sin realizar ningún cálculo adicional, ¿cuál es el rango y la norma de la matriz A ?

Ejercicio 18. Probar que efectivamente A^+b es la solución óptima del problema de mínimos cuadrados $Ax = b$.

Ejercicio 19. Mostrar que la pseudoinversa generalizada de una matriz A $n \times n$ verifica que

$$(AA^+)^T = AA^+, \quad (AA^+)^2 = AA^+.$$

Interpretar el significado de las igualdades anteriores desde el punto de vista de la teoría de aplicaciones lineales.

Ejercicio 20. Calcular la factorización de Cholesky de las matrices

$$A = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 3 & 1 & 1 \\ 0 & 1 & 3 & 1 \\ 1 & 1 & 1 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 4 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 4 \end{bmatrix}.$$

Ejercicio 21. Determine los valores de α para los que la matriz

$$\begin{bmatrix} 2 & \alpha & 0 \\ \alpha & 2 & \alpha \\ 0 & \alpha & 2 \end{bmatrix}$$

admite factorización de Cholesky y calcúlela en su caso.

Ejercicio 22. Probar que si una matriz cuadrada tiene factorización de Cholesky, entonces es definida positiva.

PROBLEMAS

Problema 1. Sea A una matriz $n \times n$. El algoritmo de eliminación gaussiana requiere, para cada k , encontrar los multiplicadores $l_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$, ($i = k + 1, \dots, n$), y a continuación restar a cada fila i posterior a k , la fila k -ésima, multiplicada por l_{ik} .

1. Elabore una función de MATLAB que, para la matriz A , devuelva las matrices L y U , de su factorización (sin pivotaje) $A = LU$. Con una matriz cualquiera (por ejemplo, una aleatoria obtenida con el comando `rand(4)`), pruebe su función viendo si el resultado de efectuar `A-L*U` es una matriz de ceros o una matriz cuyos elementos son del orden de la precisión de la máquina.
2. Elabore una tabla con los errores relativos que se cometen al resolver un sistema con la factorización LU programada en el punto anterior y con la orden de MATLAB `A\b`, para matrices A de orden 10, 20, 40, 80, 160. Puede elegir A aleatoria, así como la solución exacta `xe=rand(n,1)`, creando entonces el vector b como `b=A*xe`. ¿Qué observa?
3. Modifique su programa para que lleve a cabo la eliminación gaussiana con pivotaje parcial de filas, tal y como se indica en (5), y repita la tabla del punto anterior (no olvide probarlo antes con una matriz aleatoria de orden 4). ¿Qué conclusiones podemos sacar?
4. Sabiendo que el comando de MATLAB `cpitime` es un reloj, y que por tanto, para saber el tiempo de ejecución de un comando podemos realizar `t=cpitime;comando;t=cpitime-t`, repita el punto anterior, pero cambiando errores por tiempos de ejecución de la factorización LU (con pivotaje parcial) hecha con su programa y con el comando de MATLAB `lu`. Cada vez que dobla el orden n , ¿cómo cambia el tiempo de ejecución de su comando? ¿Es razonable esto? ¿Qué comando es más rápido, el suyo o el de MATLAB?

5. Observando que si $l \in \mathbb{R}^p$ y $a \in \mathbb{R}^q$,

$$l = \begin{bmatrix} l_1 \\ \vdots \\ l_p \end{bmatrix}, \quad a^T = [a_1, \dots, a_q], \quad \text{entonces } la^T = \begin{bmatrix} l_1 a_1 & \dots & l_1 a_q \\ \vdots & & \vdots \\ l_p a_1 & \dots & l_p a_q \end{bmatrix},$$

es matriz de rango 1 cuyas filas son todas múltiplos de a^T , modifique su rutina de eliminación gaussiana con pivotaje para que sólo efectúe un ciclo (`for k=1:n-1 ...`). Pruebe la nueva función con una matriz de orden 4. Repita lo hecho en el punto anterior, comparando los tiempos de ejecución de su rutina de eliminación gaussiana del punto anterior y de este. ¿Qué conclusión se puede sacar? Comparando con los tiempos de ejecución del comando `lu` de MATLAB, ¿Qué otra conclusión deducimos? ¿Tiene idea del por qué de estos tiempos tan diferentes para un mismo algoritmo?

Problema 2. Considere la matriz

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}.$$

1. ¿Cuál es el rango de A ?

¿Es posible realizar la descomposición LU de A sin intercambio de filas?

Calcule numéricamente la descomposición $PA=LU$ de A con pivoteo parcial.

2. Considere el vector $b = [1, 1, 1, 1]^T$. Resuelva el sistema $Ax = b$

- Directamente con la orden `\` de MATLAB.
- Con la descomposición LU calculada en el apartado 1.

Compruebe que ambas soluciones lo son en el sentido de los mínimos cuadrados. Calcule la norma de las dos soluciones y explique el resultado.

3. Construya una función de MATLAB que devuelva la **solución óptima** en el sentido de los mínimos cuadrados. Los argumentos de entrada deben ser la matriz A y el vector b y la salida debe ser la solución. Puede emplear la orden `svd` de MATLAB para calcular la descomposición en valores singulares.

Calcule la norma de la solución y compárela con las del apartado 2. Explique el resultado.

4. Genere matrices aleatorias A_n de orden n y vectores aleatorios $b_n \in \mathbb{R}^n$ con $n = 40, 80, 160, 320, 640$ y calcule los tiempos de ejecución para resolver los problemas $A_n x = b_n$,

- con la orden `\` de MATLAB, y
- con la descomposición en valores singulares.

Escriba una tabla con los valores de n y los tiempos correspondientes, dibuje (de manera aproximada) una gráfica en escala logarítmica y estime (también de manera aproximada) el orden de los dos métodos. ¿Le parece razonable que el comando `\` de MATLAB no calcule la solución óptima por defecto? Justifique su respuesta.

Problema 3. Se desea ajustar a un conjunto de datos bidimensionales, $Z = \{(x_i, y_i), i = 1, 2, \dots, n\}$, curvas polinómicas y trigonométricas mediante el método de los mínimos cuadrados.

1. Diseñe una función en MATLAB que ajuste la función $y = a_1 \text{sen}(x) + a_2 \text{cos}(x) + a_3 \text{sen}(2x) + a_4 \text{cos}(2x)$, en el sentido de los mínimos cuadrados, al conjunto de puntos Z , es decir, que encuentre los valores de los parámetros a_1, a_2, a_3, a_4 que resuelven el sistema sobredeterminado:

$$\begin{cases} a_1 \text{sen}(x_1) + a_2 \text{cos}(x_1) + a_3 \text{sen}(2x_1) + a_4 \text{cos}(2x_1) = y_1 \\ a_1 \text{sen}(x_2) + a_2 \text{cos}(x_2) + a_3 \text{sen}(2x_2) + a_4 \text{cos}(2x_2) = y_2 \\ \dots \\ a_1 \text{sen}(x_n) + a_2 \text{cos}(x_n) + a_3 \text{sen}(2x_n) + a_4 \text{cos}(2x_n) = y_n \end{cases}$$

Los argumentos de entrada deben ser los vectores de abscisas, $X = (x_1, x_2, \dots, x_n)$, y de ordenadas, $Y = (y_1, y_2, \dots, y_n)$, y los argumentos de salida deben ser el vector solución a y la matriz A del sistema sobredeterminado.

2. Considere ahora el problema de ajustar un polinomio de grado N al mismo conjunto de datos Z , es decir, se trata de encontrar un polinomio $p_N(x) = c_0 + c_1 x + c_2 x^2 + \dots + c_N x^N$, cuyo vector de coeficientes $c_N = (c_0, c_1, c_2, \dots, c_N)^t$ sea solución en el sentido de los mínimos cuadrados del sistema $A_N c_N = Y$:

$$\begin{cases} c_0 + c_1 x_1 + \dots + c_N x_1^N = y_1 \\ c_0 + c_1 x_2 + \dots + c_N x_2^N = y_2 \\ \dots \\ c_0 + c_1 x_n + \dots + c_N x_n^N = y_n \end{cases}$$

Los argumentos de entrada deben ser los vectores de abscisas, $X = (x_1, x_2, \dots, x_n)$, y de ordenadas, $Y = (y_1, y_2, \dots, y_n)$, y el grado N del polinomio, y los argumentos de salida deben ser la solución c_N y la matriz A_N de coeficientes del sistema sobredeterminado.

3. Aplique las funciones de los apartados (1) y (2) (para $N = 1, 2, 3, 4, 5$) al siguiente conjunto de datos: $\{(0, -6), (\frac{\pi}{4}, 2), (\frac{\pi}{2}, 5), (\frac{3\pi}{4}, -1), (\pi, -2), (\frac{5\pi}{4}, 1), (\frac{3\pi}{2}, 3), (\frac{7\pi}{4}, -5), (2\pi, -6)\}$.
4. Compare los errores, $\|A a^* - Y^t\|$ en el caso del ajuste trigonométrico y $\|A_N c_N^* - Y^t\|$, $N = 1, 2, 3, 4, 5$ en el polinómico, que se cometen cuando se utilizan las funciones de los apartados (1) y (2), respectivamente (a^* y c_N^* son las correspondientes soluciones en el sentido de los mínimos cuadrados).

Problema 4. La matriz de Hilbert de orden n es

$$H_n = \left(\frac{1}{i+j-1} \right)_{1 \leq i, j \leq n}.$$

1. Construya una función de MATLAB que, dado n , devuelva la matriz de Hilbert H_n , sin que ejecute ningún ciclo. (Sugerencia: la última línea puede ser $H=1./H$;) Utilice el comando `hilb` de MATLAB para saber si su función es correcta.
2. Utilizando el comando `hilb`, y el comando `cond`, calcule los números de condición de las 15 primeras matrices de Hilbert.
3. Para $n = 5, 10, 15$ calcule los vectores w_n formados por la suma de las columnas de H_n . ¿Cuál es la solución de los sistemas $H_n x_n = w_n$? Calcule dichas soluciones con el comando `\` de MATLAB. Comente y justifique las diferencias en precisión observadas en las tres soluciones numéricas.
4. Para los valores de n del apartado anterior, calcule los residuos $w_n - H_n x_n$. ¿Qué norma tienen? ¿Es o no es fiable considerar como solución aceptable de $Ax = b$ aquella para la que su residuo $r = b - Ax$ sea pequeño? ¿Cuándo podemos fiarnos de este criterio?
5. Obtenga la inversa de las matrices H_n para $n = 5, 10, 15$ a través de sus factorizaciones LU y mediante el comando `invhilb` de MATLAB. Multiplique por las correspondientes matrices de Hilbert y vea si el resultado del producto se parece a la identidad. ¿Por qué un resultado es significativamente peor que el otro, si se están manejando básicamente la mismas matrices?

Problema 5. Considere el siguiente sistema de ecuaciones lineales

$$\left. \begin{array}{rcccc} x_1 + & x_2 + & x_3 + & x_4 = & 1 \\ \epsilon x_1 & & & & = & 0 \\ & \epsilon x_2 & & & = & 0 \\ & & \epsilon x_3 & & = & 0 \\ & & & \epsilon x_4 & = & 0 \end{array} \right\}, \epsilon \geq 0.$$

1. ¿Para qué valores de ϵ tiene el sistema solución única en el sentido de mínimos cuadrados? Obtenga la solución exacta calculando con lápiz y papel. Puede ayudarle el hecho de que, si $v \in \mathbb{R}^n$, la inversa de la matriz $I + \lambda v v^T$ es

$$(I + \lambda v v^T)^{-1} = I - \frac{\lambda}{1 + \lambda v^T v} v v^T.$$

2. Para los valores de $\epsilon = 1, 10^{-2}, 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}$, resuelva las ecuaciones normales de Gauss y obtenga el error relativo. ¿Qué observa?
3. Elabore una función de MATLAB que, dada una matriz A devuelva su factorización QR reducida obtenida por el método de Gram-Schmidt. ¿Sería capaz de programarla con un sólo ciclo? Aplíquela a resolver los problemas de mínimos cuadrados del apartado 1. ¿Qué observa? ¿Son mejores estos errores que los del apartado anterior? ¿Le parecen aceptables los errores obtenidos? ¿Qué número de condición tienen las matrices $A^T A$? ¿Y las matrices A ?

- Resuelva los problemas de mínimos cuadrados del Apartado 1 mediante la factorización QR obtenida con el comando `qr` de MATLAB. Compare los resultados con los de apartados anteriores, y describa y justifique el fenómeno numérico observado.
- Resuelva los problemas de mínimos cuadrados que venimos considerando mediante la orden `A\b` de MATLAB. ¿Las soluciones obtenidas son las mismas que las del Apartado 3?

Problema 6. Para evitar la inestabilidad numérica en la eliminación gaussiana se introducen las estrategias de pivotaje parcial y total. La primera se estudió en el Problema 1, y obtuvimos como resultado tres matrices P , L y U a partir de la matriz A , de forma que $PA = LU$. En la segunda, en el paso k -ésimo de la eliminación gaussiana se determina $c_k = \max\{|a_{ij}^{(k)}| : k \leq i, j \leq n\}$ y se elige un par de índices (i_k, j_k) donde se alcanza el máximo. Posteriormente, se intercambian filas y columnas para que el par (i_k, j_k) ocupe la posición (k, k) y se realiza la eliminación. Como resultado obtenemos cuatro matrices L , U , P y Q de manera que $PAQ = LU$. La matriz Q recoge la información de los intercambios de columnas efectuados.

- Diseñe una función de MATLAB que calcule la descomposición LU de una matriz A con pivotaje total (la orden `[ym im] = max(A)` de MATLAB devuelve dos vectores; `ym` contiene los elementos máximos de cada una de las columnas de A , e `im` las filas donde aparecen).
- Considere la matriz

$$A = \begin{bmatrix} 4 & -4 & -4 & 5 \\ 1 & 4 & 4 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & -3 & -6 & 0 \end{bmatrix}.$$

Emplee la función anterior para calcular la descomposición LU de A con pivotaje total y compruebe que $PAQ = LU$.

- Elabore una tabla con los tiempos de ejecución y los errores relativos que se comenten al resolver un sistema con la factorización LU programada en el apartado 1 y con la orden que desarrollamos en la práctica con pivotaje parcial (Problema 1), para matrices aleatorias de orden $n = 2n_0, 4n_0, 8n_0, 16n_0$ y $32n_0$. Escoja n_0 de acuerdo con la velocidad de su procesador y la memoria de la máquina. Elija la solución exacta de manera aleatoria con `xe=rand(n,1)`, creando entonces el vector b como `b=A*xe`.

Escriba los errores y los tiempos de ejecución en forma de tabla y comente los resultados.

- Conocida la descomposición LU de una matriz se pide resolver el problema $B^3x = b$ de dos formas distintas. En primer lugar, calculando explícitamente B^3 y resolviendo el sistema con la orden `\` de MATLAB y en segundo lugar, aprovechando la descomposición LU sin intercambio de B y transformando la ecuación $B^3x = b$ en 6 sistemas triangulares encadenados. Diseñe un fichero de comandos de MATLAB que resuelva y estime el tiempo de ejecución y el error relativo cometido en este problema de dos formas distintas. Elija B como una matriz aleatoria de orden 300 y la solución exacta y el vector b como en el apartado 3.

Escriba los errores relativos, los tiempos de ejecución y los ficheros de comandos. Para que no sea preciso realizar intercambios de filas en la matriz B puede generar la matriz con las siguientes órdenes:

```
B=rand(300); [L,U,P]=lu(B); B=P*B; [L,U]=lu(B);
```