Tema 5: Métodos directos

Métodos directos para sistema de ecuaciones lineales. Eliminación de Gauss. Sustitución hacia atrás. Estrategias de pivoteo. Conteo de operaciones. Matrices diagonal dominante y positiva definida. Escalamiento de una matriz. Cálculo de matriz inversa. Descomposición LU con y sin pivoteo. Sustitución hacia adelante. Descomposiciones de Doolittle, Crout y Cholesky. Algoritmos para matrices con estructuras especiales (banda, tridiagonales, Hessenberg). Refinamiento iterativo. Factorización QR usando transformada de Householder.

Sistemas de Ecuaciones Lineales - Métodos directos

Métodos directos

Por medio de las operaciones elementales anteriores, un sistema de ecuaciones lineales

$$Ax = b$$

se puede transformar a un sistema lineal (equivalente) más fácil de resolver y que tiene el mismo conjunto de soluciones.

El principio de los métodos directos que vamos a estudiar reside en determinar una matriz *M* invertible, tal que la matriz *MA* sea triangular superior. Tenemos que resolver entonces el sistema lineal

$$MAx = Mb$$

Este principio es la base del **método de Gauss** para la resolución de sistemas lineales con matrices A invertibles.

Sistemas de Ecuaciones Lineales

Método de Gauss.

El método de Gauss es un método general de resolución de un sistema lineal de la forma

$$Ax = b$$

donde *A* es una matriz invertible.

Esta basado en el siguiente hecho:

si tuviésemos una matriz triangular superior, la resolución numérica del sistema es inmediata.

Este se compone de las siguientes etapas:

- Procedimiento de eliminación, que equivale a determinar una matriz invertible *M* tal que la matriz *MA* sea una matriz triangular superior
- Cálculo del vector *Mb*
- Resolución del sistema lineal *MAx* = *Mb*, por el método de sustitución hacia atrás.

Sistemas de Ecuaciones Lineales

Matrices de permutación.

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \qquad P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

matriz de permutación

$$PA = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} 3 & 4 \\ 1 & 2 \end{pmatrix}$$

intercambia las filas 1 y 2

$$AP = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 4 & 3 \end{pmatrix}$$

intercambia las columnas 1 y 2

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb*

$$A = \begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 1 & 1 & -1 & 1 \\ 0 & 1 & 8 & 12 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 0 \\ 5 \\ 2 \end{pmatrix}$$

matriz ampliada $\left(\begin{array}{cccc|c} 0 & 1 & 2 & 1 & 1 \\ 1 & 2 & 1 & 3 & 0 \\ 1 & 1 & -1 & 1 & 5 \\ 0 & 1 & 8 & 12 & 2 \end{array}\right)$

$F_1 \leftrightarrow F_2$

$$\rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 1 & 3 & 0 \\ 0 & 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & 1 & 5 \\ 0 & 1 & 8 & 12 & 2 \end{array}\right)$$

$F_3 \leftarrow -1 \cdot F_1 + F_3$

$$\rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 1 & 3 & 0 \\ 0 & 1 & 2 & 1 & 1 \\ 0 & -1 & -2 & -2 & 5 \\ 0 & 1 & 8 & 12 & 2 \end{array}\right)$$

$F_3 \leftarrow 1 \cdot F_2 + F_3$
$F_4 \leftarrow -1 \cdot F_2 + F_4$

$$\rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 1 & 3 & 0 \\ 0 & 1 & 2 & 1 & 1 \\ 0 & 0 & 0 & -1 & 6 \\ 0 & 0 & 6 & 11 & 1 \end{array}\right)$$

$F_3 \leftrightarrow F_4$

$$\rightarrow \left(\begin{array}{cccc|c} 1 & 2 & 1 & 3 & 0 \\ 0 & 1 & 2 & 1 & 1 \\ 0 & 0 & 6 & 11 & 1 \\ 0 & 0 & 0 & -1 & 6 \end{array}\right)$$

matriz triangular superior

$$MA = \begin{pmatrix} 1 & 2 & 1 & 3 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 6 & 11 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad Mb = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 6 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb*

$$P_1 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad E_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad E_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix} \quad P_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$F_1 \leftrightarrow F_2$ $\qquad$ $F_3 \leftarrow -1 \cdot F_1 + F_3$ $\qquad$ $F_3 \leftarrow 1 \cdot F_2 + F_3$, $F_4 \leftarrow -1 \cdot F_2 + F_4$ $\qquad$ $F_3 \leftrightarrow F_4$

$$M = P_3 E_2 E_1 P_1 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & 1 & 0 \end{pmatrix}$$

En la práctica la matriz *M* no se calcula, sino que se obtiene directamente *MA* y *Mb*.

$$\det(M) = \begin{cases} 1 & \text{si } \Lambda \text{ es par} \\ -1 & \text{si } \Lambda \text{ es inpar} \end{cases}$$

Λ Es el número de matrices de permutación P distintas a la matriz identidad

$$Ax = b \iff MAx = Mb$$

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb* (otro ejemplo)

$$A = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 12 & -8 & 6 & 10 \\ 3 & -13 & 9 & 3 \\ -6 & 4 & 1 & -18 \end{pmatrix}, \quad b = \begin{pmatrix} 16 \\ 26 \\ -19 \\ -34 \end{pmatrix}$$

A = [6 -2 2 4; 12 -8 6 10; 3 -13 9 3; -6 4 1 -18]

b = [16; 26; -19; -34]

operaciones elementales sucesivas:

$$\left.\begin{array}{l} F_2 \leftarrow -2.F_1 + F_2 \\ F_3 \leftarrow -\dfrac{1}{2}.F_1 + F_3 \\ F_4 \leftarrow -(-1).F_1 + F_4 \end{array}\right\} \longrightarrow A = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & -12 & 8 & 1 \\ 0 & 2 & 3 & -14 \end{pmatrix}, \quad b = \begin{pmatrix} 16 \\ -6 \\ -27 \\ -18 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb* (otro ejemplo) (cont.)

$$\left.\begin{aligned} F_3 &\leftarrow -3.F_2 + F_3 \\ F_4 &\leftarrow -\left(-\frac{1}{2}\right).F_2 + F_4 \end{aligned}\right\} \longrightarrow \quad A = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 4 & -13 \end{pmatrix}, \quad b = \begin{pmatrix} 16 \\ -6 \\ -9 \\ -21 \end{pmatrix}$$

$$\left. F_4 \leftarrow -2.F_3 + F_4 \right\} \longrightarrow \quad A = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{pmatrix}, \quad b = \begin{pmatrix} 16 \\ -6 \\ -9 \\ -3 \end{pmatrix}$$

sistema triangular superior: $MA = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{pmatrix}, \quad Mb = \begin{pmatrix} 16 \\ -6 \\ -9 \\ -3 \end{pmatrix}$

U = [6 -2 2 4; 0 -4 2 2; 0 0 2 -5; 0 0 0 -3]
c = [16 -6 -9 -3]'

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb* (otro ejemplo) (cont.)

M es el producto de matrices elementales:

$$\left.\begin{aligned} F_2 &\leftarrow -2.F_1 + F_2 \\ F_3 &\leftarrow -\frac{1}{2}.F_1 + F_3 \\ F_4 &\leftarrow -(-1).F_1 + F_4 \end{aligned}\right\} \longrightarrow \qquad E_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -1/2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

E1 = [1 0 0 0; -2 1 0 0; -1/2 0 1 0; 1 0 0 1]

$$\left.\begin{aligned} F_3 &\leftarrow -3.F_2 + F_3 \\ F_4 &\leftarrow -\left(-\frac{1}{2}\right).F_2 + F_4 \end{aligned}\right\} \longrightarrow \qquad E_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -3 & 1 & 0 \\ 0 & 1/2 & 0 & 1 \end{pmatrix}$$

E2 = [1 0 0 0; 0 1 0 0; 0 -3 1 0; 0 1/2 0 1]

$$F_4 \leftarrow -2.F_3 + F_4 \quad \Big\} \longrightarrow \qquad E_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -2 & 1 \end{pmatrix}$$

E3 = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 -2 1]

Sistemas de Ecuaciones Lineales

Método de Gauss. Construcción de *MA* y *Mb* (otro ejemplo) (cont.)

M es el producto de matrices elementales:

$$M = E_3\, E_2\, E_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ 11/2 & -3 & 1 & 0 \\ -11 & 13/2 & -2 & 1 \end{pmatrix}$$

y se cumple que:

$$MA = \begin{pmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{pmatrix} \quad y \quad Mb = \begin{pmatrix} 16 \\ -6 \\ -9 \\ -3 \end{pmatrix}$$

Obs. Notar que la inversa de una matriz elemental E_1 y el producto de las matrices elementales E_1^{-1}, E_2^{-1} y E_3^{-1} es muy sencillo de construir

$$E_1^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix} \quad \text{y} \quad E_1^{-1} E_2^{-1} E_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1/2 & 3 & 1 & 0 \\ -1 & -1/2 & 2 & 1 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de Gauss. Ejemplo:

$$\left.\begin{aligned} x+2y+z&=\ \ 3\\ x+y+2z&=\ \ 9\\ 2x+y+z&=16\end{aligned}\right\}\Rightarrow \left.\begin{aligned} x+2y+z&=\ \ 3\\ -y+z&=\ \ 6\\ -3y-z&=10\end{aligned}\right\}\Rightarrow \left.\begin{aligned} x+2y+z&=\ \ 3\\ -y+z&=\ \ 6\\ -4z&=-8\end{aligned}\right\}\Rightarrow \begin{aligned} z&=\ \ 2\\ y&=-4\\ x&=\ \ 9\end{aligned}$$

$$F_2 \leftarrow -1\cdot F_1+F_2$$
$$F_3 \leftarrow -2\cdot F_1+F_3$$

$$F_3 \leftarrow -3\cdot F_2+F_3$$

$$\left.\begin{array}{c}\underbrace{\begin{bmatrix}1&&\\&1&\\&3&1\end{bmatrix}^{-1}\begin{bmatrix}1&&\\1&1&\\2&&1\end{bmatrix}^{-1}\begin{bmatrix}1&2&1\\1&1&2\\2&1&1\end{bmatrix}=\begin{bmatrix}1&2&1\\&-1&1\\&&-4\end{bmatrix}}\\ \Downarrow\\ \begin{bmatrix}1&2&1\\1&1&2\\2&1&1\end{bmatrix}=\begin{bmatrix}1&&\\1&1&\\2&3&1\end{bmatrix}\begin{bmatrix}1&2&1\\&-1&1\\&&-4\end{bmatrix}\end{array}\right\}\Rightarrow \begin{array}{c}\underbrace{\begin{bmatrix}1&2&1\\1&1&2\\2&1&1\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix}=\begin{bmatrix}3\\9\\6\end{bmatrix}}\\ \Downarrow\\ \begin{bmatrix}1&&\\1&1&\\2&3&1\end{bmatrix}\begin{bmatrix}1&2&1\\&-1&1\\&&-4\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix}=\begin{bmatrix}3\\9\\6\end{bmatrix}\end{array}$$

Sistemas de Ecuaciones Lineales

Algoritmo de eliminación en el método de Gauss.

Vamos a suponer que los elementos de la diagonal (pivotes) de las matrices sucesivas que surgen durante este método, son no nulos.

Leer *A* y *b*

Para *k* = 2 hasta *n* → k-1 fila pivote

  Para *i* = *k* hasta *n* → modifica filas *k* a la *n*

    $\alpha = a_{i,k-1} / a_{k-1,k-1}$

    Para *j* = *k* hasta *n* → modifica columnas *k* a la *n*

      $a_{i,j} = a_{i,j} - \alpha\, a_{k-1,j}$

    Fin para

    $b_i = b_i - \alpha\, b_{k-1}$ → modifica el lado derecho

  Fin para

Fin para

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

Obs. Al finalizar sabemos que U (matriz triangular superior *MA*) va a estar almacenada en la parte triangular superior de la matriz A, incluyendo la diagonal, y que el término de la derecha (es decir, *Mb*) está almacenado en el mismo vector *b*

Ejemplo: egs_sp.m

Sistemas de Ecuaciones Lineales

Algoritmo de eliminación en el método de Gauss.

En general algunos de los elementos de la diagonal de las matrices sucesivas, pueden ser cero. Esto se solventa permutando la filas en la matriz *A*, lo cual corresponde a multiplicar *A* por una matriz elemental de permutación *P*.

Leer *A* y *b*

Para *k* = 2 hasta *n*

Buscar un índice i_0 tal que $(k-1 \le i_0 \le n)$ **y** $(a_{i_0,k-1} \ne 0)$

Permutar la fila *k-1* con la fila i_0

estrategia de pivoteo

Para *i* = *k* hasta *n*

$$\alpha = a_{i,k-1} / a_{k-1,k-1}$$

Para *j* = *k* hasta *n*

$$a_{i,j} = a_{i,j} - \alpha\, a_{k-1,j}$$

Fin para

$$b_i = b_i - \alpha\, b_{k-1}$$

Fin para

Fin para

Obs. La mejor manera de escoger i_0 es tal que

$$\left|a_{i_0,k-1}\right| \ge \left|a_{i,k-1}\right| \quad \text{para } k-1 \le i \le n$$

Sistemas de Ecuaciones Lineales

Algoritmo de eliminación en el método de Gauss con pivoteo

$$C = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1n} & c_{1,n+1} \\ c_{21} & c_{22} & \cdots & c_{2n} & c_{2,n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} & c_{n,n+1} \end{pmatrix}$$

Leer *A* y *b* en una matriz *C* = (*A* | *b*)

Para *k* = 2 hasta *n*

    i_0 = k-1
    Para i=k hasta n
        Si $\left|c_{i_0,k-1}\right| < \left|c_{i,k-1}\right|$
            i_0 = i
        Fin si
    Fin para

escoger el mejor i_0

    Para j=1 hasta n+1
        $s = c_{i_0,j};\;\; c_{i_0,j} = c_{k-1,j};\;\; c_{k-1,j} = s$
    Fin para

permutar la fila *k-1* con i_0

    Para *i* = *k* hasta *n*
        $\alpha = c_{i,k-1} / c_{k-1,k-1}$
        Para *j* = *k* hasta *n+1*
            $c_{i,j} = c_{i,j} - \alpha c_{k-1,j}$
        Fin para
    Fin para

modificar elementos de las filas *k* a la *n* y columnas *k* a la *n*+1

Fin para

Sistemas de Ecuaciones Lineales

Método de sustitución hacia atrás.

$$MAx = Mb \qquad \text{donde} \qquad MA = \begin{pmatrix} 1 & 2 & 1 & 3 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 6 & 11 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad Mb = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 6 \end{pmatrix}$$

$$\det(MA) = 1 \cdot 1 \cdot 6 \cdot (-1) = -6 \neq 0$$

En forma de ecuaciones

$$\left\{ \begin{aligned} x_1 + 2x_2 + x_3 + 3x_4 &= 0 \\ x_2 + 2x_3 + x_4 &= 1 \\ 6x_3 + 11x_4 &= 1 \\ -x_4 &= 6 \end{aligned} \right. \Leftrightarrow \left\{ \begin{aligned} x_1 &= -2x_2 - x_3 - 3x_4 \\ x_2 &= 1 - 2x_3 - x_4 \\ x_3 &= (1 - 11x_4)/6 \\ x_4 &= -6 \end{aligned} \right. \Leftrightarrow x = \begin{pmatrix} 37.5 \\ -15.3333 \\ 11.1667 \\ -6 \end{pmatrix}$$

Algoritmo general para resolver el sistema *U x = d*

$$x_n = d_n / u_{nn}$$

$$x_i = (d_i - \sum_{j=i+1}^{n} u_{ij} x_j) / u_{ii}$$

$$\text{para } i = n-1 \text{ hasta } 1$$

Sistemas de Ecuaciones Lineales

Método de sustitución hacia atrás.

Resolución de un sistema de ecuaciones lineales triangular superior

D:\DATA_DISCO_D\saul\USB\Matlab\programas\triang_sup.m*

```
%Resuelve el sistema triangular superior Ux=b usando sustitucion hacia atras
%Entradas:  U = matriz cuadrada triangular superior,
%           b = vector de la misma dimension
%Salida:    x = vector (solucion)

U = input('matriz  ');
%disp(U);
b = input('vector  ');
%disp(b);
[n m]=size(U);
l = length(b);

if (n == m & n == l)
   for i = n:-1:1
      sum =0.;
      for j = i+1:n
         sum = sum + U(i,j)*x(j);
      end
      x(i) = (b(i) - sum)/U(i,i);
   end
   disp('vector solucion')
   disp(x);
else
   disp('matriz no es cuadrada o');
   disp('las dimensiones de la matriz y el vector columna no son compatibles');
end
```

Ready

U una matriz n×n triangular superior, *b* vector de longitud n. Resolver el sistema

$$U\,x = b$$

U = [1,2,1,3; 0,1,2,1; 0,0,6,11; 0,0,0,-1]
b = [0,1,1,6]

Algoritmo

$$x_n = b_n / u_{nn}$$

$$x_i = (b_i - \sum_{j=i+1}^{n} u_{ij} x_j) / u_{ii}$$

para $i = n\text{-}1$ hasta 1

Ejemplo: sust_atras.m

Sistemas de Ecuaciones Lineales

Estrategias de pivoteo - Ejemplo

$$\begin{pmatrix} 0.003 & 59.14 \\ 5.291 & -6.13 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 59.17 \\ 46.78 \end{pmatrix}$$ solución $$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10 \\ 1 \end{pmatrix}$$

Aplicamos el método de Gauss usando aritmética de 4 dígitos

$$\left(\begin{array}{cc|c} 0.003 & 59.14 & 59.17 \\ 5.291 & -6.13 & 46.78 \end{array}\right) \qquad \alpha = -\frac{5.291}{0.003} = -1763.66 \quad \rightarrow \quad \alpha = -1764$$

$$\left(\begin{array}{cc|c} 0.003 & 59.14 & 59.17 \\ 0 & -104300 & -104400 \end{array}\right)$$ usando sustitución hacia atrás

$$x_1 = \frac{59.17 - 59.14 \cdot 1.001}{0.003} = -10.0 \qquad \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -10 \\ 1.001 \end{pmatrix}$$

El error es grande en x_1, aunque es pequeño de 0.001 en x_2.

Este ejemplo ilustra las dificultades que pueden surgir en algunos casos cuando el elemento pivote, es pequeño en relación a los elementos de columna en la matriz.

Sistemas de Ecuaciones Lineales

Estrategias de pivoteo – Ejemplo (cont.)

$$\begin{pmatrix} 0.003 & 59.14 \\ 5.291 & -6.13 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 59.17 \\ 46.78 \end{pmatrix} \qquad \text{solución} \qquad \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10 \\ 1 \end{pmatrix}$$

La idea es seleccionar el elemento en la misma columna que está debajo de la diagonal y que tiene el mayor valor absoluto.

Permutamos las filas F_1 y F_2, ya que el pivote (0.003) es pequeño respecto a los otros elementos en la columna

$$\left(\begin{array}{cc|c} 5.291 & -6.13 & 46.78 \\ 0.003 & 59.14 & 59.17 \end{array}\right) \qquad \alpha = -\frac{0.003}{5.291} = -0.000567$$

$$\left(\begin{array}{cc|c} 5.291 & -6.13 & 46.78 \\ 0 & 59.14 & 59.14 \end{array}\right)$$

usando sustitución hacia atrás

$$x_2 = \frac{59.14}{59.14} = 1, \quad x_1 = \frac{46.78 + 6.13 \cdot 1}{5.291} = 10.0 \qquad \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10 \\ 1 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Conteo de operaciones de punto flotante (+ , - , * , /) para el método de Gauss

- Triangularización del sistema

Obs.

$$\sum_{k=1}^{m} k = \frac{m(m+1)}{2}$$

$$\sum_{k=1}^{m} k^2 = \frac{m(m+1)(2m+1)}{6}$$

Número de sumas y restas: nop1

$$\left.\begin{array}{l}\text{para } k=2,\ n\cdot(n-1)\\ \text{para } k=3,\ (n-1)\cdot(n-2)\\ \ldots\\ \text{para } k=n,\ 2\cdot 1\end{array}\right\} \quad \sum_{k=2}^{n}(n-k+2)(n-k+1)$$

$$nop1 = \sum_{k=2}^{n}(n-k+2)(n-k+1) = \sum_{k=1}^{n-1}(n-k+1)(n-k)$$

$$= \sum_{k=1}^{n-1} n^2 + (-2k+1)n + k^2 - k$$

$$= n^2(n-1) - 2\frac{(n-1)n}{2}n + n(n-1) + \frac{(n-1)n(2(n-1)+1)}{6} - \frac{(n-1)n}{2}$$

$$= n(n-1)[1 + \frac{1}{3}n - \frac{1}{6} - \frac{1}{2}] = \frac{1}{3}n(n-1)(n+1)$$

$$nop1 = \frac{n^3 - n}{3} \qquad (9)$$

Sistemas de Ecuaciones Lineales

Conteo de operaciones de punto flotante para el método de Gauss (cont.)

Número de multiplicaciones y divisiones: nop2

Número de multiplicaciones = nop1

Número de divisiones

$$\left.\begin{array}{l}\text{para k=2, (n-1)}\\ \text{para k=3, (n-2)}\\ \ldots\\ \text{para k=n, 1}\end{array}\right\} \sum_{k=2}^{n}(n-k+1)$$

$$nop2 = nop1 + \sum_{k=2}^{n}(n-k+1) = nop1 + \sum_{k=1}^{n-1}(n-k)$$

$$= nop1 + n(n-1) - \frac{n(n-1)}{2} = \frac{n^3-n}{3} + \frac{n^2-n}{2} = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6}$$

$$nop2 = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} \qquad (10)$$

De (9) y (10) el número total de operaciones: nop

$$nop = nop1 + nop2 = \frac{n^3-n}{3} + \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} = \boxed{\frac{2n^3}{3} + \frac{n^2}{2} - \frac{7n}{6}} \qquad (11)$$

Sistemas de Ecuaciones Lineales

Conteo de operaciones de punto flotante para el método de Gauss (cont.)

- Resolución del sistema triangular

Número de sumas y restas: nop3

$$nop3 = \sum_{i=n-1}^{1}(n-i+1) = \sum_{i=1}^{n-1}(i+1) = \frac{(n-1)n}{2} + n - 1 = \frac{n^2}{2} + \frac{n}{2} - 1 \qquad (12)$$

Número de multiplicaciones y divisiones: nop4

$$nop4 = [\sum_{i=n-1}^{1}(n-i+1)] + 1 = [\sum_{i=1}^{n-1}(i+1)] + 1 = \frac{n^2}{2} + \frac{n}{2} \qquad (13)$$

De (12) y (13) el número total de operaciones: nop

$$nop = \frac{n^2}{2} + \frac{n}{2} - 1 + \frac{n^2}{2} + \frac{n}{2} = \boxed{n^2 + n - 1} \qquad (14)$$

Finalmente, de (11) y (14), el método de Gauss involucra el número total de operaciones

$$\frac{2n^3}{3} + \frac{n^2}{2} - \frac{7n}{6} + n^2 + n - 1 = \boxed{\frac{2n^3}{3} + \frac{3n^2}{2} - \frac{n}{6} - 1} \quad \rightarrow \ O(n^3)$$

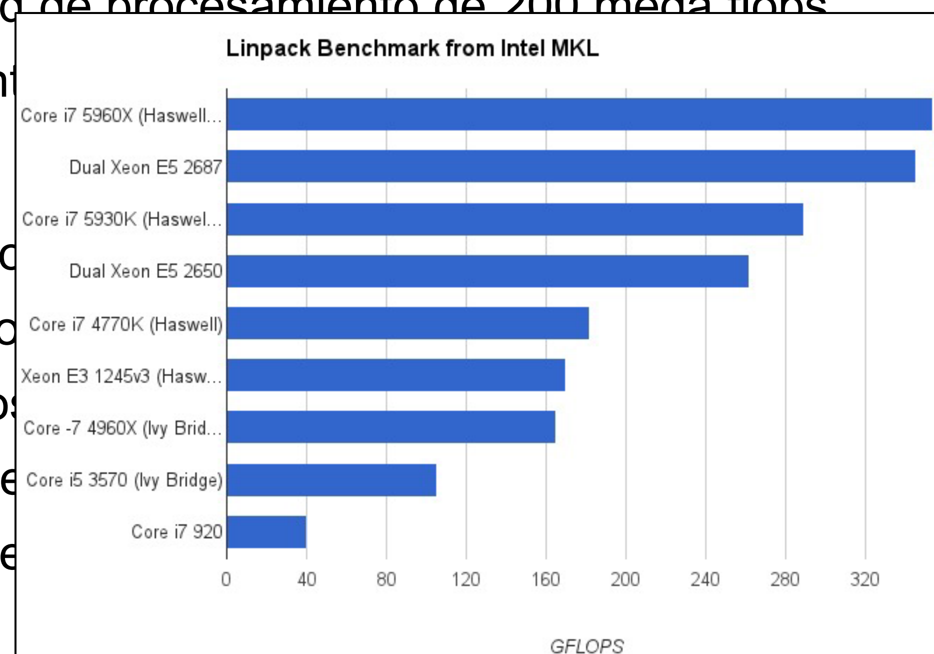
Sistemas de Ecuaciones Lineales

Conteo de operaciones de punto flotante (+ , - , * , /)

Observaciones:

A manera de ilustración, supongamos que el sistema lineal a ser resuelto tiene dimensión 5 mil, es decir *n* = 5000. Entonces, la cantidad de operaciones de punto flotante que se requerirán para obtener una aproximación numérica de la solución de dicho sistema, es aproximadamente $n^3 = 125 \times 10^9$. Si se dispone de un computador con una capacidad de procesamiento de 200 mega flops (200 millones de operaciones de punt requerirán

$125 \times 10^9 / 200 \times 10^6 = 625$ segundo equivale a un poco más de 10 minuto sobre todo si se deben resolver varios destacar que este valor de *n* no es de reales clásicos manejan sistemas line



¿Cálculo de los flops en matlab?
Ejemplo: ej_tictoc

clase 4

Sistemas de Ecuaciones Lineales

Método de descomposición LU.

Este consiste en encontrar una descomposición de la matriz cuadrada A invertible de la forma

$$A = LU$$

donde L es una matriz triangular inferior y U es una matriz triangular superior.

Supondremos que no es necesario realizar permutaciones de las filas de A, y procedemos como en el método de Gauss cuando pasamos del sistema lineal *Ax=b* al sistema triangular superior *Ux=d*, con

$$U = E_{n-1} \cdots E_1 A$$

donde E_i es una matriz elemental, para *i=1*,…,*n-1*.

$$E_i = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & e_{i+1,i} & 1 & & \\ & & \vdots & & \ddots & \\ & & e_{n,i} & & & 1 \end{pmatrix} \xrightarrow{\text{inversa}} E_i^{-1} = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -e_{i+1,i} & 1 & & \\ & & \vdots & & \ddots & \\ & & -e_{n,i} & & & 1 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

Producto de matrices elementales

$$E_i^{-1} = \begin{array}{c}\begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -e_{i+1,i} & 1 & & \\ & & \vdots & & \ddots & \\ & & -e_{n,i} & & & 1 \end{pmatrix} \\ \text{columna i}\end{array}$$

I≠J

$$E_j^{-1} = \begin{array}{c}\begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -e_{j+1,j} & 1 & & \\ & & \vdots & & \ddots & \\ & & -e_{n,j} & & & 1 \end{pmatrix} \\ \text{columna j}\end{array}$$

$$E_i^{-1}E_j^{-1} = \begin{array}{c}\begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & -e_{i+1,i} & \ddots & & & & \\ & & & & 1 & & & \\ & & \vdots & & -e_{j+1,j} & 1 & & \\ & & & & \vdots & & \ddots & \\ & & -e_{n,i} & & -e_{n,j} & & & 1 \end{pmatrix} \\ \text{columna i} \quad \text{columna j}\end{array}$$

$i < j$

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

Se tiene entonces $$E_1^{-1} \cdots E_{n-1}^{-1} U = A$$

Luego el producto de estas matrices elementales nos da la matriz *L* triangular inferior con 1 en la diagonal principal

$$L = E_1^{-1} \cdots E_{n-1}^{-1}$$

$$L = E_1^{-1} \cdots E_{n-1}^{-1} = \begin{pmatrix} 1 & & & & & & & \\ -e_{2,1} & \ddots & & & & & & \\ & & 1 & & & & & \\ & & -e_{i+1,i} & \ddots & & & & \\ \vdots & & & & 1 & & & \\ & & \vdots & & -e_{j+1,j} & \ddots & & \\ & & & & \vdots & & 1 & \\ -e_{n,1} & & -e_{n,i} & & -e_{n,j} & & -e_{n,n-1} & 1 \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

Algoritmo para calcular L y U simultáneamente.

Al final del proceso U está almacenada en la parte triangular superior de A y L en la parte triangular inferior estricta de A (no se incluye la diagonal)

Leer A

Para k = 2 hasta n

  Para i = k hasta n

    $a_{i,k-1} = a_{i,k-1} / a_{k-1,k-1}$

cálculo de L

    Para j = k hasta n

      $a_{i,j} = a_{i,j} - a_{i,k-1} a_{k-1,j}$

cálculo de U

    Fin para

  Fin para

Fin para

Ejemplo: egs_LU.m

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

Luego para resolver el sistema $Ax = b$ debemos resolver $LUx = b$ el cual lo operamos en 2 pasos (realizando el cambio $y = Ux$)

* $Ly = b$
* $Ux = y$

Algoritmo para resolver

$$LUx = b$$

$x_1 = b_1$
Para *i* = *2* hasta *n*

$$x_i = b_i - \sum_{j=1}^{i-1} a_{ij} x_j$$

Fin para

Resolver *Ly=b*

$x_n = x_n / a_{nn}$
Para i = n-1 hasta 1

$$x_i = \left(x_i - \sum_{j=i+1}^{n} a_{ij} x_j \right) / a_{ii}$$

Fin para

Resolver *Ux=y*

Obs. En este algoritmo se esta usando la matriz *A* =(a_{ij}) que sale del método de descomposición *LU* basado en Gauss, mostrado en la lamina anterior.

Ejercicio: Calcular el número de operaciones elementales para resolver un sistema lineal de ecuaciones usando el método de descomposición *LU*.

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

Condición necesaria y suficiente para que una matriz admita una descomposición *LU*.

Teorema:

Sea $A = (a_{ij})_{1 \le i, j \le n}$ una matriz invertible. Denotemos por A_{pp} la submatriz de A siguiente

$$A_{pp} = (a_{ij})_{1 \le i, j \le p}, \quad 1 \le p \le n$$

El $\det(A_{pp}) \neq 0, \quad 1 \le p \le n$ si y sólo si la matriz A admite una descomposición *LU*.

Ejercicio. Usando el teorema anterior, ¿qué puede decir de las matrices siguientes?

$$A = \begin{pmatrix} 0. & 2. \\ 3. & 1. \end{pmatrix} \qquad A = \begin{pmatrix} 0. & 1. \\ 0. & 2. \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Método de descomposición LU (cont.)

$$A = \begin{pmatrix} 0. & 2. \\ 3. & 1. \end{pmatrix}$$

A es invertible, det(*A*) = -6.

$$\det(A_{11}) = 0, \quad \det(A_{22}) = -6$$

descomposición LU para A

$$L = \begin{pmatrix} 1. & 0. \\ 0. & 1. \end{pmatrix}, \quad U = \begin{pmatrix} 3. & 1. \\ 0. & 2. \end{pmatrix}, \quad P = \begin{pmatrix} 0. & 1. \\ 1. & 0. \end{pmatrix}$$

$$A = PLU$$

$$A = \begin{pmatrix} 0. & 1. \\ 0. & 2. \end{pmatrix}$$

A no es invertible, det(*A*) = 0.

$$\det(A_{11}) = 0, \quad \det(A_{22}) = 0$$

descomposición LU para A

$$L = \begin{pmatrix} 1. & 0. \\ 0. & 1. \end{pmatrix}, \quad U = \begin{pmatrix} 0. & 1. \\ 0. & 2. \end{pmatrix}$$

$$A = LU$$

Sistemas de Ecuaciones Lineales

Definición:

Una matriz *A* de orden *n*×*n* es **diagonal dominante estricta** si

$$\left|a_{ii}\right| > \sum_{j=1, j\neq i}^{n} \left|a_{ij}\right|, \quad 1 \leq i \leq n$$

Consecuencia:

1. Si *A* es diagonal dominante estricta entonces todas la matrices equivalentes obtenidas en el método de Gauss, también son diagonal dominante estrictas. En este caso no se requiere pivoteo para el método de Gauss.
2. Si *A* es diagonal dominante estricta entonces *A* es invertible (+)
3. Si *A* es diagonal dominante estricta entonces *A* admite una descomposición *LU*.

(+) ver lema de Hadamar
(https://es.wikipedia.org/wiki/Matriz_de_diagonal_estrictamente_dominante)

Definición:

Una matriz A de orden *n*×*n* es **definida positiva** si

$$x^T A x > 0 \quad \text{para todo} \quad x \in R^n - \{0\}$$

Consecuencia:

Si *A* es simétrica y *A* es definida positiva , entonces $A = L\,L^T$ donde *L* es una matriz triangular inferior con los elementos de la diagonal positivos.

Sistemas de Ecuaciones Lineales

Ejemplo:

La matriz

$$A = \begin{pmatrix} 7 & 2 & 0 \\ 3 & 5 & -1 \\ 0 & 5 & -6 \end{pmatrix}$$

es diagonal dominante estricta, ya que $|a_{ii}| > \sum_{j=1, j \neq i}^{n} |a_{ij}|, \quad 1 \leq i \leq n$

$$|7| > |2| + |0|$$

$$|5| > |3| + |-1|$$

$$|-6| > |0| + |5|$$

Es interesante notar sin embargo que A^T no es diagonal dominante estricta

$$A^T = \begin{pmatrix} 7 & 3 & 0 \\ 2 & 5 & 5 \\ 0 & -1 & -6 \end{pmatrix}$$

$$|7| > |3| + |0|$$

$$\boxed{|5| < |2| + |5|}$$

$$|-6| > |0| + |-1|$$

Sistemas de Ecuaciones Lineales

Ejemplo:

La matriz

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

es positiva definida, ya que $x^T A x > 0$ para todo $x \in R^n - \{0\}$

$$x^T A x = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix} \begin{pmatrix} 2x_1 - x_2 \\ -x_1 + 2x_2 - x_3 \\ -x_2 + 2x_3 \end{pmatrix}$$

$$= 2x_1^2 - 2x_1x_2 + 2x_2^2 - 2x_2x_3 + 2x_3^2$$

$$= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2$$

Finalmente $x^T A x = x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 > 0$

a menos que $x_1 = x_2 = x_3 = 0$

Sistemas de Ecuaciones Lineales

Teorema:

Sea $A = (a_{ij})_{1 \le i, j \le n}$ una matriz simétrica. Denotemos por A_{pp} la submatriz de A siguiente

$$A_{pp} = (a_{ij})_{1 \le i, j \le p}, \quad 1 \le p \le n$$

El $\det(A_{pp}) > 0, \quad 1 \le p \le n$ si y sólo si la matriz A es definida positiva.

Ejemplo: En el ejemplo anterior se uso la definición para demostrar que la matriz simétrica A es definida positiva. Para confirmar esto usando el resultado de arriba notar que

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

$$\det(A_{11}) = 2 > 0 \qquad \det(A_{22}) = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} = 3 > 0$$

$$\det(A_{22}) = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} = 4 > 0$$

Sistemas de Ecuaciones Lineales

A matrices simétricas y definidas positivas

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

A=[2 -1 0; -1 2 -1; 0 -1 2]
Lt = chol(A)

$$= \begin{pmatrix} 1.4142 & 0 & 0 \\ -0.7071 & 1.2247 & 0 \\ 0 & -0.8165 & 1.1547 \end{pmatrix} \begin{pmatrix} 1.4142 & -0.7071 & 0 \\ 0 & 1.2247 & -0.8165 \\ 0 & 0 & 1.1547 \end{pmatrix}$$

$$A = \begin{pmatrix} 4 & 12 & -16 \\ 12 & 37 & -43 \\ -16 & -43 & 98 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 \\ 6 & 1 & 0 \\ -8 & 5 & 3 \end{pmatrix} \begin{pmatrix} 2 & 6 & -8 \\ 0 & 1 & 5 \\ 0 & 0 & 3 \end{pmatrix}$$

A=[4 12 -16; 12 37 -43; -16 -43 98]
Lt = chol(A)

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz

Supongamos que se quiere resolver el sistema lineal con matriz ampliada (usando aritmética de 4 dígitos)

$$A_1 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1\cdot 10^{-5} & 1\cdot 10^{-5} & 2\cdot 10^{-5} \end{array}\right) \quad \text{con solución} \quad x = \begin{pmatrix} 1.00002 \\ 0.99998 \end{pmatrix}$$

- Aplicando el método de Gauss sin y con pivoteo

$$\left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 0 & -0.5 & -0.5 \end{array}\right) \quad \Rightarrow \quad x = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{error del 100\%}$$

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz (cont.)

- Multiplicando la F_2 de A_1 por 10^5

$$A_1 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1\cdot 10^{-5} & 1\cdot 10^{-5} & 2\cdot 10^{-5} \end{array}\right)$$

$$A_2 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1 & 1 & 2 \end{array}\right)$$

aplicando el método de Gauss con y sin pivoteo

Gauss sin pivoteo

$$\left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 0 & -0.5\cdot 10^{5} & -0.5\cdot 10^{5} \end{array}\right) \Rightarrow x = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

error del 100%

Gauss con pivoteo

$$\left(\begin{array}{cc|c} 1 & 1 & 2 \\ 0 & 1 & 1 \end{array}\right) \Rightarrow x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

solución aceptable

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz (cont.)

- Multiplicando todo el sistema por 10^5 se obtiene

$$A_1 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1\cdot 10^{-5} & 1\cdot 10^{-5} & 2\cdot 10^{-5} \end{array}\right)$$

$$A_3 = \left(\begin{array}{cc|c} 2 & 1\cdot 10^{5} & 1\cdot 10^{5} \\ 1 & 1 & 2 \end{array}\right)$$

aplicando el método de Gauss con y sin pivoteo

$$\left(\begin{array}{cc|c} 2 & 10^{5} & 10^{5} \\ 0 & -0.5\cdot 10^{5} & -0.5\cdot 10^{5} \end{array}\right) \Rightarrow x = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

error del 100%

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz (cont.)

Obs. La diferencia esencial entre A_1, A_2 y A_3 es que

$$|\det(A_1)| \approx 10^{-5} \qquad A_1 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1\cdot 10^{-5} & 1\cdot 10^{-5} & 2\cdot 10^{-5} \end{array}\right)$$

$$|\det(A_2)| \approx 1 \qquad \text{donde} \qquad A_2 = \left(\begin{array}{cc|c} 2\cdot 10^{-5} & 1 & 1 \\ 1 & 1 & 2 \end{array}\right)$$

$$|\det(A_3)| \approx 10^{5} \qquad A_3 = \left(\begin{array}{cc|c} 2 & 1\cdot 10^{5} & 1\cdot 10^{5} \\ 1 & 1 & 2 \end{array}\right)$$

Para resolver el sistema $Ax = b$, el **método de escalamiento** consiste en reemplazar la matriz A por la matriz D_1AD_2, con D_1 y D_2 matrices diagonales invertibles para obtener el sistema equivalente

$$D_1AD_2(D_2^{-1}x) = D_1b \qquad (*)$$

con $|\det(D_1AD_2)| \approx 1$

El sistema (*) es equivalente al sistema $\begin{cases} D_1AD_2y = D_1b \\ x = (D_2y) \end{cases}$

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz (cont.)

Obs.

- La escogencia de D_1 y D_2 matrices diagonales corresponde únicamente a una razón de facilidad en los cálculos
- Usualmente se utilizan las cantidades

$$d_{1_i} = (\max_{1 \le j \le n} |a_{ij}|)^{-1}$$ i-ésimo elemento de la diagonal de la matriz D_1

(máximo sobre la fila i)

$$d_{2_i} = (\max_{1 \le j \le n} |a_{ji}|)^{-1}$$ i-ésimo elemento de la diagonal de la matriz D_2

(máximo sobre la columna i)

- Normalmente se utiliza escalamiento por filas, es decir se toma D_2 igual a la matriz identidad

Sistemas de Ecuaciones Lineales

Escalamiento de una matriz (cont.)

Ejemplo. Queremos resolver el sistema $Ax = b$ con

$$A = \begin{pmatrix} 2\cdot 10^{-5} & 1 \\ 1\cdot 10^{-5} & 1\cdot 10^{-5} \end{pmatrix} \qquad b = \begin{pmatrix} 1 \\ 2\cdot 10^{-5} \end{pmatrix}$$

$$D_1 = \begin{pmatrix} 1 & 0 \\ 0 & 10^5 \end{pmatrix}$$ calculada usando $$d_{1_i} = (\max_{1\le j\le n} |a_{ij}|)^{-1}$$

$$D_1 A = \begin{pmatrix} 2\cdot 10^{-5} & 1 \\ 1 & 1 \end{pmatrix}$$ corresponde con la matriz A_2 anterior donde $$|\det(D_1 A)| \approx 1$$

Finalmente resolvemos usando eliminación gausiana con pivoteo $$D_1 A x = D_1 b \quad \Rightarrow \quad x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Obs. Esta estrategia de escalamiento puede ser incorporada al algoritmo de Gauss con pivoteo, lo cual lo hace más robusto. En ese caso, se escala cada fila de la matriz de coeficientes usando los elementos de la diagonal de D_1.

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera)

$$\begin{bmatrix} 4 & 3 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} l_{11} & 0 \\ l_{21} & l_{22} \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{bmatrix}$$

$$\begin{aligned} l_{11} \cdot u_{11} + 0 \cdot 0 &= 4 \\ l_{11} \cdot u_{12} + 0 \cdot u_{22} &= 3 \\ l_{21} \cdot u_{11} + l_{22} \cdot 0 &= 6 \\ l_{21} \cdot u_{12} + l_{22} \cdot u_{22} &= 3. \end{aligned}$$

sistema sobrederminado, supongamos de $l_{11} = l_{22} = 1$ $\Rightarrow$

$$\begin{aligned} l_{21} &= 1.5 \\ u_{11} &= 4 \\ u_{12} &= 3 \\ u_{22} &= -1.5 \end{aligned}$$

$$\begin{bmatrix} 4 & 3 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1.5 & 1 \end{bmatrix} \begin{bmatrix} 4 & 3 \\ 0 & -1.5 \end{bmatrix}$$

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera)

$A = L\,U$ donde $\quad l_{is} = 0 \ \text{para} \ s > i \qquad u_{sj} = 0 \ \text{para} \ s > j$

$$A = \begin{array}{cc} & \begin{array}{ccccc} & j & & & \end{array} \\ \begin{array}{c} \\ \\ \\ i \\ \\ \end{array} & \begin{pmatrix} a_{11} & & \cdots & & a_{1n} \\ & \ddots & & & \\ \vdots & & a_{kk} & & \vdots \\ & a_{ij} & & \ddots & \\ a_{n1} & & \cdots & & a_{nn} \end{pmatrix} \end{array} \quad L = \begin{array}{cc} \begin{array}{c} \\ \\ \\ i \\ \\ \end{array} & \begin{pmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix} \end{array} \quad U = \begin{array}{c} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \cdots & u_{3n} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix} \\ \begin{array}{ccccc} & & & j & \end{array} \end{array}$$

$$a_{ij} = \begin{pmatrix} l_{i1} & l_{i2} & l_{i3} & \underbrace{\cdots \quad l_{in}}_{\substack{\text{ceros} \\ s > i}} \end{pmatrix} \begin{pmatrix} u_{1j} \\ u_{2j} \\ u_{3j} \\ \vdots \\ u_{nj} \end{pmatrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} \substack{\text{ceros} \\ s > j} = \sum_{s=1}^{n} l_{is} u_{sj} = \sum_{s=1}^{\min(i,j)} l_{is} u_{sj} \qquad (14)$$

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera) (cont.)

$$a_{ij} = \sum_{s=1}^{n} l_{is} u_{sj} = \sum_{s=1}^{\min(i,j)} l_{is} u_{sj} \qquad (14)$$

Fijamos *i=j=k* en la ecuación (14) (elemento en la diagonal)

$$a_{kk} = \sum_{s=1}^{k-1} l_{ks} u_{sk} + l_{kk} u_{kk} \qquad (15)$$

Supongamos que se han calculado los elementos de *U* hasta la fila *k-1* y los elementos de *L* hasta la columna *k-1*, de (15) se tiene la relación

$$U = \begin{pmatrix} u_{11} & u_{12} & \boxed{u_{13}} & \cdots & u_{1n} \\ & u_{22} & \boxed{u_{23}} & \cdots & u_{2n} \\ & & u_{33} & \cdots & u_{3n} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix} \quad L = \begin{pmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ \boxed{l_{31} \quad l_{32}} & & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix}$$

caso k=3

$$l_{kk} u_{kk} = a_{kk} - \sum_{s=1}^{k-1} l_{ks} u_{sk} \qquad (16)$$

La relación (16) permite calcular u_{kk} o l_{kk} a partir del otro.

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera) (cont.)

A continuación con l_{kk} y u_{kk} calculados, procedemos a ubicar

- la fila k de U ($i=k$)
- la columna k de L ($j=k$)

$$U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \boxed{\cdots \quad u_{3n}} & \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix}$$

caso k=3

$$L = \begin{pmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \boxed{\vdots} & \ddots & \\ l_{n1} & l_{n2} & \boxed{l_{n3}} & \cdots & l_{nn} \end{pmatrix}$$

Usando la ecuación (14) se tiene

$$\left.\begin{aligned} a_{kj} &= \sum_{s=1}^{k-1} l_{ks} u_{sj} + l_{kk} u_{kj} \quad \text{para } k+1 \le j \le n \\ a_{ik} &= \sum_{s=1}^{k-1} l_{is} u_{sk} + l_{ik} u_{kk} \quad \text{para } k+1 \le i \le n \end{aligned}\right\} \qquad (17)$$

es decir, si $\quad l_{kk} \neq 0 \quad$ y $\quad u_{kk} \neq 0$

$$\left.\begin{aligned} u_{kj} &= \left(a_{kj} - \sum_{s=1}^{k-1} l_{ks} u_{sj} \right) / l_{kk} \quad \text{para } k+1 \le j \le n \\ l_{ik} &= \left(a_{ik} - \sum_{s=1}^{k-1} l_{is} u_{sk} \right) / u_{kk} \quad \text{para } k+1 \le i \le n \end{aligned}\right\} \qquad (18)$$

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera) (cont.)

Obs. Es importante notar que los cálculos en (18) pueden realizarse en paralelo, lo cual puede representar un gran ahorro en tiempo de CPU.

Obs. El algoritmo basado en las fórmulas precedentes (16) y (18)

$$\left.\begin{aligned} l_{kk}u_{kk} &= a_{kk} - \sum_{s=1}^{k-1} l_{ks}u_{sk} \\ u_{kj} &= \left(a_{kj} - \sum_{s=1}^{k-1} l_{ks}u_{sj} + l_{kk} \right) / l_{kk} \quad \text{para } k+1 \le j \le n \\ l_{ik} &= \left(a_{ik} - \sum_{s=1}^{k-1} l_{is}u_{sk} + l_{kk} \right) / u_{kk} \quad \text{para } k+1 \le i \le n \end{aligned}\right\} \quad 1 \le k \le n$$

se conoce como

- factorización LU de Doolittle cuando *L* es una matriz triangular inferior con 1 en la diagonal principal
- factorización LU de Crout cuando *U* es una matriz triangular superior con 1 en la diagonal principal
- factorización LU de Cholesky cuando $U = L^t$ (de donde $u_{kk} = l_{kk}$)

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera) (cont.)

Algoritmo

Leer $A=(a_{ij}), n$

Para k = 1 hasta n

Especificar un valor no cero para l_{kk} o u_{kk}

y calcular el otro de

$$l_{kk}u_{kk} = a_{kk} - \sum_{s=1}^{k-1} l_{ks}u_{sk}$$

Para $j = k+1$ hasta n

$$u_{kj} = \left(a_{kj} - \sum_{s=1}^{k-1} l_{ks}u_{sj} \right) / l_{kk}$$

Fin para

Para $i = k+1$ hasta n

$$l_{ik} = \left(a_{ik} - \sum_{s=1}^{k-1} l_{is}u_{sk} \right) / u_{kk}$$

Fin para

Fin para

$$U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \boxed{\cdots \quad u_{3n}} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix}$$

caso k=3

$$L = \begin{pmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Factorización LU general (otra manera) (cont.)

Para resolver el sistema lineal $Ax = b$ usando la descomposición *LU* general de la matriz *A*, debemos resolver $LUx = b$, el cual lo operamos en 2 pasos (realizando el cambio $y = Ux$)

* $Ly = b$
* $Ux = y$

Algoritmo para resolver

$$LUx = b$$

$y_1 = b_1 / l_{11}$

Para *i* = 2 hasta *n*

$$y_i = \left(b_i - \sum_{j=1}^{i-1} l_{ij} y_j \right) / l_{ii}$$

Fin para

Resolver *Ly=b*

$x_n = y_n / u_{nn}$

Para i = n-1 hasta 1

$$x_i = \left(y_i - \sum_{j=i+1}^{n} u_{ij} x_j \right) / u_{ii}$$

Fin para

Resolver *Ux=y*

Ejercicio: Calcular el número de operaciones básicas para resolver un sistema lineal de ecuaciones usando el método de descomposición *LU* general.

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky

Obs. Cuando una matriz *A* es simétrica, sería razonable esperar que en la factorización *LU* (cuando esta existe) de A, se tenga $U = L^T$.

En general esto no es cierto.

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -0.5 & 1 & 0 \\ 0 & -0.6 & 1 \end{pmatrix} \begin{pmatrix} 2 & -1 & 0 \\ 0 & 1.5 & -1 \\ 0 & 0 & 1.3333 \end{pmatrix}$$

```
A=[2 -1 0; -1 2 -1; 0 -1 2]
[L U] = lu(A)
```

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky

Obs. Supongamos A es no singular, $A = LL^T$ y $x \neq 0$, entonces L es no singular, y si definimos $y = L^T x \neq 0$, entonces

$$x^T A x = x^T L L^T x = (L^T x)^T (L^T x) = y^T y = \sum_{i=1}^{n} y_i^2 > 0.$$

Entonces, una matriz A no singular que puede ser factorizada como LL^T tiene que ser definida positiva.

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky (cont.)

Obs. *Sea A una matriz definida positiva entonces A es no singular.*

Para ver esto, es suficiente demostrar que

$$Ax = 0 \Rightarrow x = 0$$

Supongamos lo contrario, es decir $Ax = 0$ pero $x \neq 0$.

Entonces, multiplicando por x^T por la izquierda

$$x^T A x = 0$$

lo cual contradice el hecho que A es definida positiva.

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky (cont.)

Obs. Sea A una matriz $n\times n$ definida positiva de la forma $A=\begin{pmatrix}\alpha & a^T\\ a & A_*\end{pmatrix}$, donde a es un vector de longitud n-1, entonces $\alpha > 0$ y A_* es definida positiva.

Veamos que $\alpha > 0$. Sea x^T=(1,0,…,0), entonces

$$0 < x^T A x = (1 \quad 0)\begin{pmatrix}\alpha & a^T\\ a & A_*\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = \alpha \quad \Rightarrow \quad \alpha > 0.$$

vector nulo de longitud n-1

Veamos que A_* es definida positiva.

Sea $y \neq 0$ vector de dimensión n-1 y $x^T = (0 \;\; y^T)$, entonces

$$0 < x^T A x = (0 \quad y^T)\begin{pmatrix}\alpha & a^T\\ a & A_*\end{pmatrix}\begin{pmatrix}0\\ y\end{pmatrix} = y^T A_* \, y \quad \Rightarrow \quad y^T A_* \, y > 0.$$

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky (cont.)

Hemos visto que una matriz A no singular, la cual puede ser factorizada como LL^T, cumple que A es definida positiva.

El inverso es también verdad. Esto se enuncia a continuación.

Teorema:

Si A es una matriz simétrica y definida positiva, entonces

$$A = L\,L^T,$$

donde L es una matriz triangular inferior con los elementos de la diagonal positivos. Además esta descomposición es única.

Obs. La descomposición $A = L\,L^T$ recibe el nombre de **descomposición de Cholesky**.

Obs. Si $A = L\,L^T$ se tiene que $A = (-L)\,(-L^T)$ lo cual indica que A tiene otra descomposición de Cholesky. Esto no contradice el teorema anterior. ¿Por qué?

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky (cont.)

Para obtener el algoritmo de la descomposición de Cholesky, basta observar, en el algoritmo de la descomposición *LU* general, que: lamina123

$$U = L^T$$

de donde, para cualquier índice *k* entre 1 y *n*, se cumple que:

$$L = \begin{pmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix}$$

$$U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \cdots & u_{3n} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix}$$

$$u_{kk} = l_{kk}$$

$$u_{kj} = l_{jk} \ , \ \text{para} \ \ k+1 \le j \le n$$

$$u_{ik} = l_{ki} \ , \ \text{para} \ \ k+1 \le i \le n$$

Sistemas de Ecuaciones Lineales

Método de descomposición Cholesky (cont.)

Algoritmo

Leer $A=(a_{ij}), n$

Para k = 1 hasta n

$$l_{kk} = \sqrt{a_{kk} - \sum_{s=1}^{k-1} l_{ks}^2}$$

Para $i = k+1$ hasta n

$$l_{ik} = \left(a_{ik} - \sum_{s=1}^{k-1} l_{is} l_{ks} \right) / \, l_{kk}$$

Fin para

Fin para

Ejercicio: Calcular el número de operaciones básicas para resolver un sistema lineal de ecuaciones usando el método de descomposición de Cholesky.

Sistemas de Ecuaciones Lineales

Cálculo de la matriz inversa

Sea A es una matriz invertible, $A = (a_{ij})_{1 \le i, j \le n}$

Sean e_1, e_2,…, e_n la base canónica de R^n. Para el cálculo de la matriz inversa procedemos así

Calcular B tal que $AB = I \quad \text{con} \quad B = \begin{pmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{pmatrix}$

Planteamos la secuencia de problemas siguientes

$$Ax_i = e_i \quad \text{con} \quad x_i = \begin{pmatrix} b_{i1} & \cdots & b_{in} \end{pmatrix}^T \quad \text{para} \quad 1 \le i \le n$$

(x_i es la columna i de la matriz B) por alguno de los procedimientos vistos.

Ejercicio: Calcular el número de operaciones básicas involucradas en la construcción de la inversa de A usando el método de descomposición LU.

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales

Una matriz $A=(a_{ij})_{1\le i,j\le n}$ se denomina matriz banda si existen enteros *p* y *q*, con la propiedad siguiente

$$a_{ij}=\begin{cases}0 & \text{si } \mathrm{i}+p\le j\\ 0 & \text{si } j+q\le i\end{cases} \quad \text{para} \quad 1<p,q<n$$

El ancho de banda es *w =p+q-1*

Caso *p* = *q* < *n*

$i+p \le j$

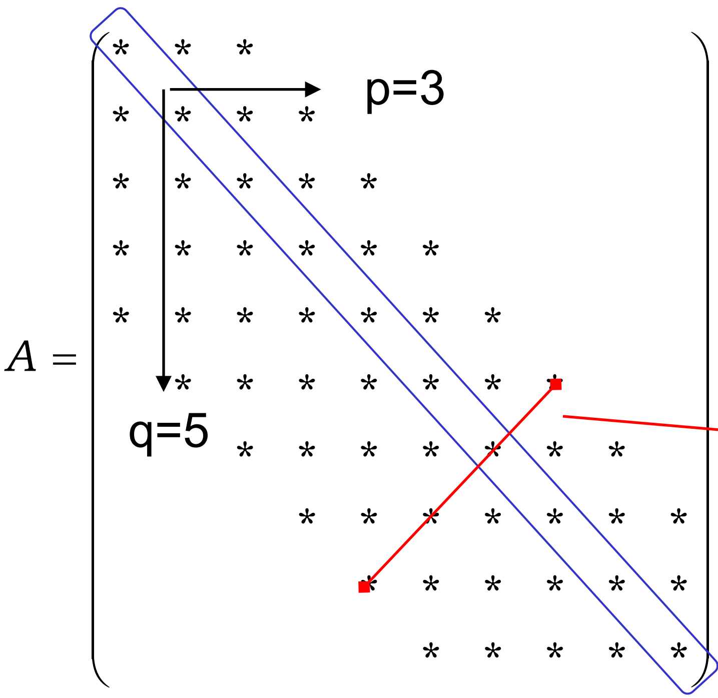
$$A=\begin{pmatrix} a_{11} & \cdots & a_{1p} & 0 & \cdots & 0\\ \vdots & \ddots & & & \ddots & \vdots\\ a_{p1} & & a_{pp} & & & 0\\ 0 & & & \ddots & & a_{pn}\\ \vdots & \ddots & & & & \vdots\\ 0 & \cdots & 0 & a_{np} & \cdots & a_{nn}\end{pmatrix}$$

$j+p \le i$

Un caso especial de estas matrices es cuando *p=q=2*, la cual da un ancho de banda *w=3*, y se denominan tridiagonales

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)



A es una matriz banda con ancho de banda

w = 3+5-1 = 7

$$a_{ij} = \begin{cases} 0 & \text{si } \mathrm{i} + p \leq j \\ 0 & \text{si } j + q \leq i \end{cases} \quad \text{para} \quad 1 < p, q < n$$

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)

Los algoritmos de factorización se pueden simplificar considerablemente en el caso de matrices banda, debido al gran número de ceros que aparecen en patrones regulares en estas matrices.

Caso *A* matriz tridiagonal

Supongamos que es posible encontrar matrices *L* y *U*, tal que *A* = *LU*.

$$L = \begin{pmatrix} l_{11} & 0 & \cdots & & 0 \\ l_{21} & l_{22} & \ddots & & \vdots \\ 0 & & \ddots & & \\ \vdots & \ddots & & & 0 \\ 0 & \cdots & 0 & l_{n,n-1} & l_{nn} \end{pmatrix} \qquad U = \begin{pmatrix} 1 & u_{12} & 0 & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ & \ddots & \ddots & & 0 \\ \vdots & & & & u_{n-1,n} \\ 0 & \cdots & & 0 & 1 \end{pmatrix}$$

La multiplicación *A* = *LU* da, sin contar los elementos cero, las ecuaciones

$$a_{11} = l_{11}$$

$$a_{i,i-1} = l_{i,i-1} \quad \text{para } i = 2, \ldots n$$

$$a_{ii} = l_{i,i-1} u_{i-1,i} + l_{ii} \quad \text{para } i = 2, \ldots n$$

$$a_{i,i+1} = l_{ii} u_{i,i+1} \quad \text{para } i = 1, \ldots n-1$$

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)

Algoritmo para matrices tridiagonales

Leer *A=(a_{ij}), n*

$$l_{11} = a_{11}; \quad u_{12} = a_{12} / l_{11}$$

Para *i* = 2 hasta *n-1*

$$\left.\begin{aligned} l_{i,i-1} &= a_{i,i-1} \\ l_{ii} &= a_{ii} - l_{i,i-1}u_{i-1,i} \end{aligned}\right\} \quad \text{i-ésima fila de } L$$

$$u_{i,i+1} = a_{i,i+1} / l_{ii} \quad \text{(i+1) columna de } U$$

Fin para

$$\left.\begin{aligned} l_{n,n-1} &= a_{n,n-1} \\ l_{nn} &= a_{nn} - l_{n,n-1}u_{n-1,n} \end{aligned}\right\} \quad \text{n-ésima fila de } L$$

Ejercicio:

- Calcular el número de operaciones elementales involucradas en el algoritmo.
- Modificar el algoritmo anterior usando un conjunto de vectores para las matrices A , L y U (no almacenar los ceros).

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)

Una matriz $A = (a_{ij})_{1 \le i, j \le n}$ se denomina de Hessenberg superior si para todo $i > j+1$ se tiene $a_{ij} = 0$

Para el caso que la dimensión de *A* sea 5×5

$$A = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}$$

Apliquemos Eliminación Gaussiana a esta matriz.

En el primer paso, sólo el elemento (2,1) requiere ser eliminado, el resto en la primera columna ya son ceros.

Para esto tenemos que restar un múltiplo de la primera fila de la segunda para obtener la matriz

$$\begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}$$

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)

En el segundo paso, restamos un múltiplo de la segunda fila de la tercera

$$\begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}$$

Dos pasos más del proceso llevan a las matrices

$$\begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix} \quad \text{y} \quad \begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix}$$

siendo la última triangular superior.

Sistemas de Ecuaciones Lineales

Algoritmos para factorizar matrices especiales (cont.)

El seudo-código para reducir este tipo de matrices a triangular superior, dejando de lado el pivoteo por simplicidad, es el siguiente:

Algoritmo de triangularización para matrices Hessenberg superior

```
Leer A=( a_ij ), n
Para k = 1 hasta n-1
    a(k+1,k) = a(k+1,k)/a(k,k)
    Para j = k+1 hasta n
        a(k+1,j) = a(k+1,j)-a(k+1,k)*a(k,j)
    Fin para
Fin para
```

Comparar con el método de eliminación gaussiana, p.90

hess2triangular.m para el caso de almacenamiento en 2 vectores.

Los factores de multiplicación sobrescriben los elementos debajo de la diagonal principal de *A*, y la matriz triangular final sobrescribe el triángulo superior de *A*.

Ejercicio: Calcular el número de operaciones elementales involucradas en este algoritmo y comparar con Gauss. Incluir pivoteo.

Sistemas de Ecuaciones Lineales

Técnicas de mejoramiento de la solución. Refinamiento Iterativo.

Sea *proc(A,b)* un procedimiento que calcula el vector $\bar{x}$ a partir de una matriz *A* y un vector *b*, es decir el vector $\bar{x}$ es solución aproximada para *Ax=b*,

$$\bar{x} \leftarrow proc(A,b).$$

Luego si *x* es la solución exacta de *Ax=b*, entonces podemos definir *Δb*

$$\Delta b = A\bar{x} - b = A\bar{x} - Ax = A\Delta x \text{ con } \Delta x = \bar{x} - x.$$

Podemos aplicar el procedimiento anterior para resolver $A\Delta x = \Delta b$

$$\overline{\Delta x} \leftarrow proc(A,\Delta b),$$

así, $\overline{\Delta x}$ es una aproximación del verdadero error Δx.

Podemos decir que la nueva solución será $\bar{\bar{x}} = \bar{x} + \overline{\Delta x},$
que es una mejora de la solución $\bar{x}$.

Este procedimiento lo podemos repetir varias veces!

Sistemas de Ecuaciones Lineales

Técnicas de mejoramiento de la solución. Refinamiento Iterativo (cont.)

Cuantas veces debemos repetir este procedimiento?

Para detenerlo procedemos así, dado ε muy pequeño

a) ver si $\left\| A\overline{\overline{x}} - b \right\| < \varepsilon$

b) ver si $\left\| \overline{\Delta x} \right\| < \varepsilon$

c) o una combinación de (a) y (b) al mismo tiempo

Ejercicio: Escribir un programa en MATLAB que lleve a cabo el mejoramiento iterativo de la solución aproximada *x* del sistema lineal de ecuaciones *Ax* = *b*.

Sistemas de Ecuaciones Lineales

De respuestas a las siguientes interrogantes:

(1) ¿Toda matriz tiene al menos una descomposición *LU*?

(2) ¿Toda matriz invertible tiene al menos una descomposición *LU*?

(3) ¿Una matriz que tiene uno o más ceros en su diagonal principal puede tener descomposición *LU*?

(4) Si una matriz A tiene descomposición de Cholesky, $A=LL^T$, donde L es una matriz triangular inferior, ¿se puede asegurar que A es definida positiva?

(5) ¿Toda matriz diagonal dominante estricta es definida positiva?

(6) ¿Toda matriz definida positiva es diagonal dominante estricta?

Sistemas de Ecuaciones Lineales

Respuestas a las interrogantes planteadas:

(1) (Falso). Si la matriz $A = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}$

tuviera una descomposición *LU*, se tendría algo así:

$$\begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} a & 0 \\ b & c \end{pmatrix} \begin{pmatrix} d & e \\ 0 & f \end{pmatrix}$$

$$A \quad = \quad L \quad \cdot \quad U$$

de donde $\begin{cases} ad = 0 \\ ae = 1 \\ bd = 1 \\ be + cf = 2 \end{cases}$. Así, a=0 ó d=0. Cualquiera de estas dos posibilidades genera una contradicción con las ecuaciones ae=1 y bd=1. Por lo tanto, la matriz en cuestión no puede tener una descomposición *LU*.

Sistemas de Ecuaciones Lineales

Respuestas a las interrogantes planteadas:

(2) (Falso). La matriz $A = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}$ es invertible,

ya que su determinantes es -1$\neq$ 0.

Sin embargo, ya se probó en la lámina anterior que dicha matriz no tiene ninguna descomposición *LU*.

Sistemas de Ecuaciones Lineales

Descomposición QR de una matriz

Sea L una recta en R^2 que pasa por el origen. El operador Q que refleja todo vector de R^2 a través de la recta L es una transformación lineal, la cual puede ser representada por una matriz.

$\vec{u}$ y $\vec{v} \in R^2$ con $\|\vec{u}\| = 1$,

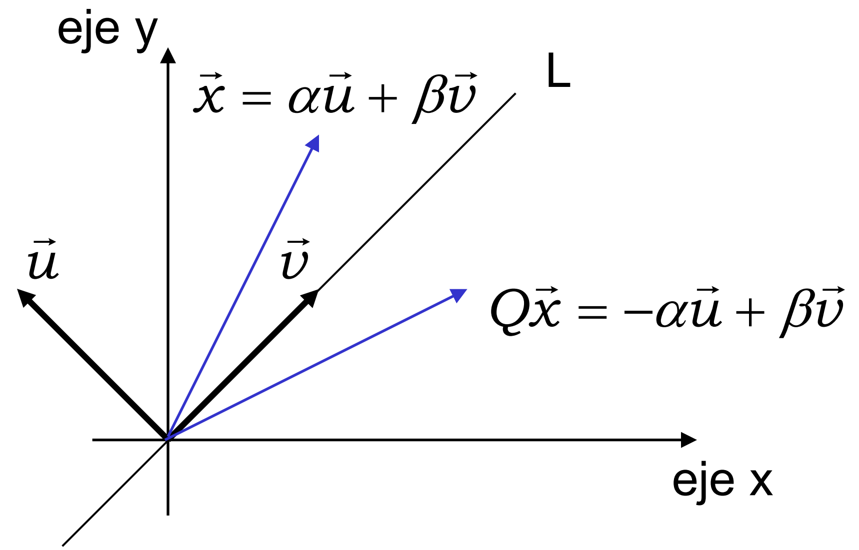
$\vec{v}$ el vector dirección de la recta L

y $\vec{u}$ y $\vec{v}$ perpendiculares.

$\vec{u}$ es perpendicular a $\vec{v} \Leftrightarrow u^t v = 0$

$\vec{u}$ y $\vec{v}$ forman un base de R^2

así, si $\vec{x} \in R^2 \Rightarrow \vec{x} = \alpha\vec{u} + \beta\vec{v}$



La reflexión de $\vec{x}$ a través de la recta L es $-\alpha\vec{u} + \beta\vec{v}$

La reflexión de $\vec{u}$ a través de la recta L es $-\vec{u}$

La reflexión de $\vec{v}$ a través de la recta L es $\vec{v}$

Sistemas de Ecuaciones Lineales

Descomposición QR de una matriz (cont.)

Podemos construir la matriz $P = \vec{u}\,\vec{u}^t \in R^{2x2}$

Se verifica:

$$P\vec{u} = \left(\vec{u}\,\vec{u}^t\right)\vec{u} = \vec{u}\left(\vec{u}^t\,\vec{u}\right) = \vec{u}\left\|\vec{u}\right\|^2 = \vec{u}$$

$$P\vec{v} = \left(\vec{u}\,\vec{u}^t\right)\vec{v} = \vec{u}\left(\vec{u}^t\,\vec{v}\right) = \vec{u}\,0 = \vec{0}$$

Construimos la matriz $Q = I - 2P = I - 2\vec{u}\,\vec{u}^t \in R^{2x2}$

Se verifica:

$$Q\vec{u} = \vec{u} - 2P\vec{u} = \vec{u} - 2\vec{u} = -\vec{u}$$

$$Q\vec{v} = \vec{v} - 2P\vec{v} = \vec{v}$$

Para $\vec{x} = \alpha\vec{u} + \beta\vec{v} \in R^2$ se tiene

$$Q\vec{x} = Q\left(\alpha\vec{u} + \beta\vec{v}\right) = \alpha Q\left(\vec{u}\right) + \beta Q\left(\vec{v}\right) = -\alpha\vec{u} + \beta\vec{v}$$

Q es la matriz que refleja vectores a través de la recta *L*.

Sistemas de Ecuaciones Lineales

Descomposición QR de una matriz (cont.)

En resumen para

$$\vec{u} \in R^n, \|\vec{u}\| = 1, P = \vec{u}\,\vec{u}^t \in R^{nxn}, Q = I - 2P \in R^{nxn}$$

se tiene

1) $P\vec{u} = \vec{u}, \quad P\vec{v} = \vec{0} \text{ si } \vec{u}^t\,\vec{v} = 0, \quad P^2 = P, \quad P^t = P \text{ (simétrica)}$

2) $Q\vec{u} = -\vec{u}, \quad Q\vec{v} = \vec{v} \text{ si } \vec{u}^t\,\vec{v} = 0, \quad Q^2 = I,$

$Q = Q^t \text{ (simétrica)}, \quad Q^{-1} = Q^t \text{ (ortogonal)}$

Definición. Si $\vec{u} \in R^n, \|\vec{u}\| = 1, Q = I - 2\vec{u}\,\vec{u}^t$
entonces *Q* se denomina un **reflector o transformación de Householder**.

Sistemas de Ecuaciones Lineales

Descomposición QR de una matriz (cont.)

Teorema 1. Si $\vec{u} \in R^n, \vec{u} \neq \vec{0}, \gamma = 2/\|\vec{u}\|_2^2, Q = I - \gamma\vec{u}\vec{u}^t$ entonces Q es un reflector o transformación de Householder.

Teorema 2. Si $\vec{x} \text{ e } \vec{y} \in R^n, \vec{x} \neq \vec{y}, \|\vec{x}\|_2 = \|\vec{y}\|_2,$ entonces existe un reflector o transformación de Householder Q, tal que $Q\vec{x} = \vec{y}$.

Teorema 3. Reflectores pueden ser usados para crear ceros en vectores y matrices, es decir, para

$$\vec{x} \in R^n, \vec{x} \neq \vec{0}, \sigma = \pm\|\vec{x}\|, \vec{y} = (-\sigma, 0, \ldots, 0)^t,$$

existe un reflector Q tal que $Q\vec{x} = \vec{y}$.

Teorema 4. Sea $A \in R^{n\times n}$, esta puede ser expresada como $A = QR$ donde *Q* es ortogonal y *R* triangular superior.

Sistemas de Ecuaciones Lineales

Descomposición QR de una matriz (cont.)

Ejemplo. Encontrar la descomposición *QR* para $A = \begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix}$

$$Q = \begin{pmatrix} -0.7071 & -0.7071 \\ -0.7071 & 0.7071 \end{pmatrix} \qquad R = \begin{pmatrix} -1.4142 & -3.5355 \\ 0 & 0.7071 \end{pmatrix}$$

$$Q * Q^T = Q^T * Q = I$$

R es triangular superior

En Matlab:
[Q,R] = qr(A)

Ejemplo. Si B=hilb(5), determinar det(B), cond(B) y las descomposiciones LU y QR de B, si esto es posible.

Obs.

- Toda matriz $n \times n$ tiene descomposición QR. Lo mismo no puede decirse para la existencia de su descomposición LU.
- Dada la descopmposición A=QR, resolver Ax=b se reduce a resolver Rx=Qb (probarlo). Esto se lleva a cabo usando sustitución hacia atrás.
- El costo de aplicar la descomposición QR es “alto” (este requiere el doble del número de operaciones elementales del método de descomposición LU).