FACTORIZACIÓN MATRICIAL. LU II (18.06 Lecture 04)

Inverse of A, AB, A^T

Product of elimination matrices

$A = LU$ (no row exchanges)

Recordemos la definición de inversa $AA^{-1} = I = A^{-1}A$

¿Cual es la inversa de AB?

Asumamos que A y B son invertibles

$$AB\,B^{-1}A^{-1} \longrightarrow A\underbrace{I}A^{-1} = AA^{-1} = I$$

Ej. zapatos y calcetines

(Tb funciona al revés $B^{-1}A^{-1}AB = I$). Luego $(AB)^{-1} = B^{-1}A^{-1}$

Veamos ahora que ocurre con la traspuesta de una matriz.

$$AA^{-1} = I \longrightarrow (AA^{-1})^T = I^T = I$$

$$(A^{-1})^T A^T = I$$

Luego esta es la inversa de $A^T \rightarrow (A^T)^{-1}$

$$\boxed{(A^T)^{-1} = (A^{-1})^T}$$

Descomposición $A = LU$

Partimos de la eliminación $EA = U$ que usamos en clases anteriores

Ejemplo

$$\begin{bmatrix} 2 & 1 \\ 8 & 7 \end{bmatrix} \longrightarrow \underset{E_{21}}{\begin{bmatrix} 1 & 0 \\ -4 & 1 \end{bmatrix}} \underset{A}{\begin{bmatrix} 2 & 1 \\ 8 & 7 \end{bmatrix}} = \underset{U}{\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix}}$$

No 4! (Singular)

¿Cómo paso de aquí a $A = LU$?

$$\underset{A}{\begin{bmatrix} 2 & 1 \\ 8 & 7 \end{bmatrix}} = \underset{L}{\begin{bmatrix} 1 & 0 \\ 4 & 1 \end{bmatrix}} \underset{U}{\begin{bmatrix} 2 & 1 \\ 0 & 3 \end{bmatrix}} \longrightarrow \text{Comprobar que funciona}$$

$L = (E_{21})^{-1} \longrightarrow$ Recordemos, la que deshace la operación original

Se llama L pq es la triangular inferior (lower).

A veces, se separa en tres partes LDU

$$\begin{bmatrix} 1 & 0 \\ 4 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1/2 \\ 0 & 1 \end{bmatrix}$$

$\longrightarrow$ La calculamos para que $DU = \hat{U}$

Esta es la antigua U

Contiene únicamente los pivotes

Como vemos, cuesta lo mismo obtener $EA = U$ que $A = LU$

¿Que ocurre en el caso 3×3?

$$E_{32}E_{31}E_{21}A = U$$ (asumimos que no tenemos que hacer intercambios de filas)

Queremos tener la L, así que necesitamos mover $E_{32}E_{31}E_{21}$ a la derecha.

$$A = (E_{32}E_{31}E_{21})^{-1} U$$

Sabemos que invertir cada E_{ij} por separado es sencillo. ¿Cómo de fácil es invertir el pack completo?

Es mejor idea invertir cada E_{ij} por separado

$$A = E_{21}^{-1}E_{31}^{-1}E_{32}^{-1}U \longrightarrow A = LU$$

$$L = \underline{E_{21}^{-1}E_{31}^{-1}E_{32}^{-1}}$$

Este producto es más amigable que $E_{32}E_{31}E_{21}$. Veamos pq.

Ejemplo: en este ejemplo, consideramos que la eliminación se consigue con estas dos matrices:

(E_{31} no es necesario)

$$\underset{E_{32}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -5 & 1 \end{bmatrix}} \underset{E_{21}}{\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}} =$$

Esto tiene que quedar así pues el método de Gauss no cambia nada en esta región

$$= \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 10 & -5 & 1 \end{bmatrix} = E \longrightarrow EA = U$$

No me gusta este 10, ¿de dónde sale? Es para tener en cuenta el efecto de la eliminación de F_1 en F_3

Veamos que ocurre si utilizo las inversas

$$\left(\overset{E_{32}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -5 & 1 \end{bmatrix}} \overset{E_{21}}{\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}} \right)^{-1}$$

$$\overset{E_{21}^{-1}}{\begin{bmatrix} 1 & 0 & 0 \\ +2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{E_{32}^{-1}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & +5 & 1 \end{bmatrix}} = \overset{L}{\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 5 & 1 \end{bmatrix}} = L$$

Aquí no tengo el 10!

$$\underline{\underline{A = LU}}$$

A = LU (sin intercambio de filas)

Si quiero obtener L solo tengo que hacer Gauss y colocar los multiplicadores en la posición correcta dentro de la matriz L.

De hecho, se puede almacenar LU en el espacio utilizado para guardar A (muy útil para ordenador)

Tb desde el pto de vista computacional es muy importante el número de operaciones que es necesario realizar. ¿Si el tamaño de la matriz es 10^6, cuánto tarda en resolverlo?

¿Cuántas operaciones para una matriz A $n \times n$?

$(n, n^2, n^3, n \log n, n!\ ?)$ → Afecta de forma directa al tiempo de cálculo.

Asumimos $\underline{n = 100}$

$$\begin{bmatrix} - & - & - & - \\ - & - & - & - \\ - & - & - & - \\ - & - & - & - \end{bmatrix} \longrightarrow \begin{bmatrix} \square & - & - & - & - \\ 0 & - & - & - & - \\ 0 & - & - & - & - \\ 0 & - & - & - & - \end{bmatrix} *$$

* Para hacer este 0, tengo que multiplicar toda la fila 1 por algún número y luego restársela a la fila 2.

Para simplificar operación = mult. + resta

El coste de hacer esto es $\sim n^2 = 100^2$

$$\begin{bmatrix} \square & - & - & - & - \\ 0 & \square & - & - & - \\ 0 & 0 & - & - & - \\ 0 & 0 & - & - & - \end{bmatrix}$$

→ Y el coste de esto $99^2 = (n-1)^2$

→ nuevo problema tiene tamaño 99

Coste total del problema

$$n^2 + (n-1)^2 + \dots + 3^2 + 2^2 + 1 \overset{*}{=}$$

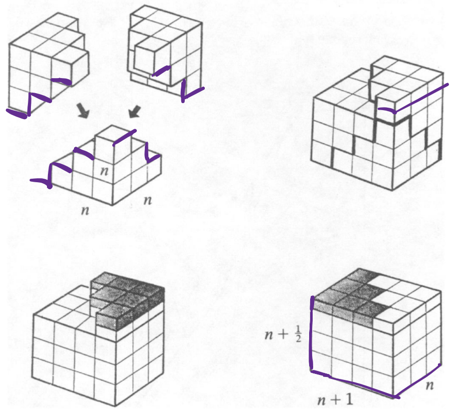
$$= \sum_{i=1}^{n} i^2 = \frac{1}{3} n (n+1) \left(n + \frac{1}{2}\right) =$$

$$= \frac{1}{3} n \left(n^2 + \frac{3}{2} n + \frac{1}{2}\right) =$$

$$= \underline{\underline{\frac{1}{3} n^3}} + \frac{1}{2} n^2 + \frac{1}{6}$$

Proof without words:
Sum of squares *

$$1^2 + 2^2 + \cdots + n^2 = \tfrac{1}{3} n (n+1)\left(n + \tfrac{1}{2}\right)$$



—Man-Keung Siu
University of Hong Kong

03/1984 - Mathematics magazine

→ Este es el término que nos importa. El coste computacional del problema crece con n^3

¿Qué hay del coste de b para resolver $Ax=b$? El coste de b es mucho más pequeño, escala con n^2. Esto significa que una vez tenga la descomposición LU, puedo resolver para muchos b de forma eficiente. Lo veremos con un ejemplo.

Ejemplo des composición LU

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 10 \end{bmatrix} \xrightarrow{F_2 - 4F_1} \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 7 & 8 & 10 \end{bmatrix} \xrightarrow{F_3 - 7F_1}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -11 \end{bmatrix} \xrightarrow{F_3 - 2F_2} \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 1 \end{bmatrix}$$

$$LU = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 10 \end{bmatrix}$$

¿Y si esto es un problema de verdad cómo se resuelve? → Sustitución hacia arriba y hacia abajo

$$Ax = b \rightarrow LUx = b \rightarrow \boxed{Ly = b}$$

$$Ux = y$$

$$\boxed{Ly = b}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$A \quad x = b$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 2 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \longrightarrow \begin{array}{l} y_1 = 1 \\ y_2 = -3 \\ y_3 = 0 \end{array}$$

$$L \quad y \quad b$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -3 \\ 0 \end{bmatrix} \qquad \begin{array}{l} x_1 = -1 \\ x_2 = 1 \\ x_3 = 0 \end{array}$$

$$U \quad x \quad y$$

Comprobamos el resultado

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \longrightarrow -\begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix} + \begin{bmatrix} 2 \\ 5 \\ 8 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \checkmark$$

Traspuestas y permutaciones

Veamos como funciona el algoritmo si tengo que hacer intercambio de filas.

Necesito matrices de permutación.

Matrices permutación 3x3

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}, \begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}$$

No hace nada // Intercambian dos filas // Intercambian 3 filas

¿Que pasa si multiplico dos del grupo? → Obtengo otra del grupo (son intercambios de filas)

¿Que pasa con la inversa? → Tb es otra del grupo. (es deshacer intercambios de filas).

Además $P^{-1} = P^T$ (no justificamos esto de momento)

¿Cuantas matrices de Permutación 4x4 hay? 24.