



Métodos Numéricos | Laboratorio 4

Ricardo Largaespada

10 de abril de 2025

1. Ecuaciones Algebraicas Lineales

1.1. Antecedentes Matemáticos

En la resolución de sistemas de ecuaciones lineales es fundamental comprender la notación y las propiedades de matrices y vectores. A continuación, se indican algunos puntos clave:

1. Notación matricial.

Se utiliza comúnmente A para denotar una matriz de dimensiones $m \times n$. Cada uno de sus elementos se puede denotar como a_{ij} , donde i indica la fila y j la columna.

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

2. Reglas de operaciones con matrices.

a) **Suma y resta:** Dos matrices se pueden sumar o restar sólo si tienen las mismas dimensiones. La suma se define elemento a elemento:

$$(A + B)_{ij} = a_{ij} + b_{ij}.$$

b) **Multiplicación por un escalar:** Si c es un escalar,

$$(c \cdot A)_{ij} = c a_{ij}.$$

c) **Multiplicación de matrices:** Si A es de dimensiones $m \times n$ y B es de dimensiones $n \times p$, la matriz $C = AB$ será de dimensiones $m \times p$ y se calcula como

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}.$$

3. Representación de ecuaciones algebraicas lineales en forma matricial.

Un sistema de ecuaciones lineales

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \end{cases} \quad (1.1)$$

se puede escribir en forma matricial como:

$$A \mathbf{x} = \mathbf{b}, \quad (1.2)$$

donde A es la matriz de coeficientes (a_{ij}) , \mathbf{x} el vector de incógnitas $(x_1, x_2, \dots, x_n)^T$ y \mathbf{b} el vector de términos independientes $(b_1, b_2, \dots, b_m)^T$.

Métodos de Solución

Existen dos clases de métodos para resolver sistemas de ecuaciones lineales algebraicas: métodos **directos** y métodos **iterativos**. La característica común de los métodos directos es que transforman las ecuaciones originales en ecuaciones equivalentes (ecuaciones que tienen la misma solución) que pueden resolverse más fácilmente. La transformación se realiza aplicando las tres operaciones que se listan a continuación. Estas llamadas *operaciones elementales* no cambian la solución, pero pueden afectar el determinante de la matriz de coeficientes, como se indica entre paréntesis.

- Intercambiar dos ecuaciones (cambia el signo de $|A|$).
- Multiplicar una ecuación por una constante distinta de cero (multiplica $|A|$ por esa misma constante).
- Multiplicar una ecuación por una constante distinta de cero y luego restarla de otra ecuación (deja $|A|$ sin cambios).

Resumen de los Métodos Directos

La Tabla (1) muestra tres métodos directos populares, cada uno de los cuales utiliza operaciones elementales para producir su propia forma final de ecuaciones fáciles de resolver.

Método	Forma inicial	Forma final
Eliminación de Gauss	$Ax = b$	$Ux = c$
Descomposición LU	$Ax = b$	$LUx = b$
Eliminación de Gauss-Jordan	$Ax = b$	$Ix = c$

Cuadro 1: Resumen de métodos directos



En la tabla anterior, U representa una matriz triangular superior, L es una matriz triangular inferior e I denota la matriz identidad. Una matriz cuadrada se llama *triangular* si contiene solo elementos cero a un lado de la diagonal principal. Así, una matriz triangular superior de 3×3 tiene la forma

$$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

y una matriz triangular inferior de 3×3 tiene la forma

$$L = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix}$$

Las matrices triangulares juegan un papel importante en el álgebra lineal, ya que simplifican muchos cálculos. Por ejemplo, considere las ecuaciones $Lx = c$, o

$$\begin{aligned} l_{11}x_1 &= c_1 \\ l_{21}x_1 + l_{22}x_2 &= c_2 \\ l_{31}x_1 + l_{32}x_2 + l_{33}x_3 &= c_3 \\ &\vdots \end{aligned}$$

Si resolvemos las ecuaciones hacia adelante, comenzando con la primera ecuación, los cálculos son muy sencillos, ya que cada ecuación contiene solo una incógnita a la vez. La solución procedería de la siguiente manera:

$$\begin{aligned} x_1 &= \frac{c_1}{l_{11}} \\ x_2 &= \frac{c_2 - l_{21}x_1}{l_{22}} \\ x_3 &= \frac{c_3 - l_{31}x_1 - l_{32}x_2}{l_{33}} \\ &\vdots \end{aligned}$$

Este procedimiento se conoce como *sustitución hacia adelante*. De manera similar, $Ux = c$, que se encuentra en la eliminación de Gauss, puede resolverse fácilmente mediante *sustitución hacia atrás*, que comienza con la última ecuación y procede hacia atrás a través de las ecuaciones.

Las ecuaciones $LUx = b$, que están asociadas con la descomposición LU, también pueden resolverse rápidamente si las reemplazamos por dos conjuntos de ecuaciones equivalentes: $Ly = b$ y $Ux = y$. Ahora, $Ly = b$ puede resolverse para y mediante sustitución hacia adelante, seguido de la solución de $Ux = y$ mediante sustitución hacia atrás.

Las ecuaciones $Ix = c$, que se obtienen mediante la eliminación de Gauss-Jordan, son equivalentes a $x = c$ (recordar que $Ix = x$), por lo que c ya es la solución.



2. Eliminación de Gauss

La eliminación de Gauss es un procedimiento sistemático para resolver sistemas de ecuaciones lineales. Se basa en la transformación escalonada de la matriz de coeficientes (y el vector de términos independientes) mediante operaciones elementales de fila.

2.1. Solución de pequeños sistemas de ecuaciones

Para sistemas de ecuaciones de pocas variables, se puede proceder con métodos más simples, incluso manuales:

1. **Método gráfico.**

Para sistemas de dos ecuaciones con dos incógnitas, se pueden representar ambas ecuaciones como rectas en el plano cartesiano y encontrar su intersección.

2. **Determinante y Regla de Cramer.**

Para sistemas de n ecuaciones con n incógnitas, el determinante de la matriz de coeficientes (si es distinto de cero) puede utilizarse para encontrar las soluciones mediante la Regla de Cramer:

$$x_i = \frac{\det(A_i)}{\det(A)},$$

donde A_i es la matriz que se obtiene reemplazando la columna i de A por el vector de términos independientes \mathbf{b} .

3. **La eliminación de incógnitas.**

Procedimiento clásico que consiste en combinar las ecuaciones para ir reduciendo el número de incógnitas, hasta llegar a ecuaciones muy simples (por ejemplo, de una sola variable).

2.2. Eliminación de Gauss Simple

El algoritmo de Eliminación de Gauss simple procede transformando la matriz A en una matriz triangular superior. Este proceso se logra restando múltiplos adecuados de cada fila a las filas de abajo para eliminar las incógnitas secuencialmente.

1. **Conteo de operaciones.**

Para un sistema de n ecuaciones con n incógnitas, cada paso de eliminación involucra multiplicaciones y sumas. Aproximadamente, el conteo de operaciones para la eliminación de Gauss simple es del orden de $O(n^3)$, considerando que la operación principal es el número de multiplicaciones.

2.3. Dificultades en los métodos de eliminación

Aunque la Eliminación de Gauss es un método robusto, existen algunas dificultades:



1. **División entre cero.**

Puede ocurrir que en algún paso del proceso el elemento pivote sea cero, lo que impide la operación de división necesaria.

2. **Sistemas mal condicionados.**

Se dice que un sistema está mal condicionado cuando pequeñas variaciones en los datos de entrada provocan grandes variaciones en el resultado. Esto se hace evidente si la matriz de coeficientes es casi singular o si los números involucrados son muy grandes o muy pequeños.

3. **Sistemas singulares.**

Un sistema es singular si la matriz de coeficientes no es invertible, es decir, su determinante es cero. En tal caso, o bien el sistema no tiene solución o tiene infinitas soluciones.

2.4. Técnicas para mejorar soluciones

1. **Uso de más cifras significativas.**

Emplear una mayor precisión en la representación numérica puede reducir los errores de redondeo.

2. **Pivoteo.**

El pivoteo consiste en intercambiar filas (y eventualmente columnas) para situar en la posición pivote un elemento no nulo y, preferiblemente, de gran magnitud, con el fin de minimizar los errores.

a) **Pivoteo Parcial:** Se intercambian filas para asegurar que el pivote sea el valor mayor posible (en valor absoluto) dentro de su columna.

b) **Pivoteo Completo:** Además de intercambiar filas, permite también el intercambio de columnas.

3. **Escalamiento.**

Consiste en dividir cada ecuación (o cada fila) por un factor que facilite los cálculos y reduzca la propagación de error.

2.5. Gauss-Jordan

La Eliminación de Gauss-Jordan extiende el proceso hasta conseguir, no sólo una matriz triangular superior, sino una *matriz identidad* en el lugar de la matriz de coeficientes. De este modo, no es necesaria una sustitución hacia atrás; el método deja las soluciones explícitamente en la parte derecha (el vector **b** transformado).

3. Método de Eliminación de Gauss

Introducción

La eliminación de Gauss es el método más conocido para resolver ecuaciones simultáneas. Consta de dos partes: la fase de eliminación y la fase de solución. Como se indica en la Tabla 2.1, la función de la fase de eliminación es transformar las ecuaciones en la forma $Ux = c$. Luego, las ecuaciones se resuelven mediante sustitución hacia atrás.

Para ilustrar el procedimiento, resolvamos el siguiente sistema de ecuaciones:

$$\begin{aligned}4x_1 - 2x_2 + x_3 &= 11 & (a) \\ -2x_1 + 4x_2 - 2x_3 &= -16 & (b) \\ x_1 - 2x_2 + 4x_3 &= 17 & (c)\end{aligned}$$

Fase de eliminación

La fase de eliminación utiliza una de las operaciones elementales: multiplicar una ecuación (digamos la ecuación j) por una constante λ y restarla de otra ecuación (ecuación i). Esto se representa simbólicamente como:

$$\text{Ecuación } (i) \leftarrow \text{Ecuación } (i) - \lambda \times \text{Ecuación } (j) \quad (1.3)$$

La ecuación que se resta (Ecuación j) se llama *ecuación pivote*.

Tomamos la ecuación (a) como pivote y eliminamos x_1 de las ecuaciones (b) y (c):

$$\begin{aligned}4x_1 - 2x_2 + x_3 &= 11 & (a) \\ 3x_2 - 1.5x_3 &= -10.5 & (b) \\ -1.5x_2 + 3.75x_3 &= 14.25 & (c)\end{aligned}$$

Ahora usamos la ecuación (b) como pivote para eliminar x_2 de (c):

$$\begin{aligned}4x_1 - 2x_2 + x_3 &= 11 \\ 3x_2 - 1.5x_3 &= -10.5 \\ 3x_3 &= 9\end{aligned}$$

Forma matricial

Las ecuaciones originales se escriben como:

$$\left[\begin{array}{ccc|c} 4 & -2 & 1 & 11 \\ -2 & 4 & -2 & -16 \\ 1 & -2 & 4 & 17 \end{array} \right]$$

Después de la primera y segunda transformación:

$$\left[\begin{array}{ccc|c} 4 & -2 & 1 & 11.00 \\ 0 & 3 & -1.5 & -10.50 \\ 0 & -1.5 & 3.75 & 14.25 \end{array} \right] \Rightarrow \left[\begin{array}{ccc|c} 4 & -2 & 1 & 11.0 \\ 0 & 3 & -1.5 & -10.5 \\ 0 & 0 & 3 & 9.0 \end{array} \right]$$



Determinante

La operación elemental no cambia el determinante. Dado que la matriz triangular es fácil de manejar, su determinante se calcula como el producto de sus elementos diagonales:

$$|A| = |\mathbf{U}| = U_{11} \times U_{22} \times \cdots \times U_{nn} \quad (1.4)$$

Fase de sustitución hacia atrás

Ahora podemos resolver por sustitución hacia atrás:

$$\begin{aligned} x_3 &= 9/3 = 3 \\ x_2 &= (-10.5 + 1.5 \cdot x_3)/3 = [-10.5 + 1.5(3)]/3 = -2 \\ x_1 &= (11 + 2x_2 - x_3)/4 = [11 + 2(-2) - 3]/4 = 1 \end{aligned}$$

Algoritmo del Método de Eliminación de Gauss

Fase de eliminación: Observemos el sistema de ecuaciones en un instante durante la fase de eliminación. Supongamos que las primeras k filas de A ya han sido transformadas a forma triangular superior. Por lo tanto, la ecuación pivote actual es la ecuación k -ésima, y todas las ecuaciones por debajo de esta aún deben ser transformadas. Nótese que los componentes de A no son los coeficientes originales (excepto para la primera fila), ya que han sido alterados por el procedimiento de eliminación. Lo mismo para a los componentes del vector constante \mathbf{b} .

$$\left[\begin{array}{cccccccc|c} A_{11} & A_{12} & A_{13} & \cdots & A_{1k} & \cdots & A_{1j} & \cdots & A_{1n} & b_1 \\ 0 & A_{22} & A_{23} & \cdots & A_{2k} & \cdots & A_{2j} & \cdots & A_{2n} & b_2 \\ 0 & 0 & A_{33} & \cdots & A_{3k} & \cdots & A_{3j} & \cdots & A_{3n} & b_3 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{kk} & \cdots & A_{kj} & \cdots & A_{kn} & b_k \\ \hline \vdots & \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{ik} & \cdots & A_{ij} & \cdots & A_{in} & b_i \\ \vdots & \vdots & \vdots & & \vdots & & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{nk} & \cdots & A_{nj} & \cdots & A_{nn} & b_n \end{array} \right] \quad \begin{array}{l} \leftarrow \text{fila pivote} \\ \\ \leftarrow \text{fila en} \\ \text{transformación} \end{array}$$

Sea la fila i una fila típica por debajo de la ecuación pivote que se va a transformar, lo que significa que el elemento A_{ik} debe ser eliminado. Podemos lograr esto multiplicando la fila pivote por $\lambda = A_{ik}/A_{kk}$ y restándola de la fila i . Los cambios correspondientes en la fila i son:

$$A_{ij} \leftarrow A_{ij} - \lambda A_{kj}, \quad j = k, k+1, \dots, n \quad (1.5a)$$

$$b_i \leftarrow b_i - \lambda b_k \quad (1.5b)$$

Para transformar toda la matriz de coeficientes a forma triangular superior, los índices k e i en las Ecs. (1.5a) y (1.5b) deben recorrer los siguientes rangos: $k = 1, 2, \dots, n-1$ (selecciona

la fila pivote), $i = k + 1, k + 2, \dots, n$ (selecciona la fila a transformar). El algoritmo para la fase de eliminación casi se escribe por sí solo:

```
for k = 1:n-1
    for i = k+1:n
        if A(i,k) ~= 0
            lambda = A(i,k)/A(k,k);
            A(i,k+1:n) = A(i,k+1:n) - lambda*A(k,k+1:n);
            b(i) = b(i) - lambda*b(k);
        end
    end
end
```

Para evitar operaciones innecesarias, el algoritmo anterior se desvía ligeramente de las ecuaciones (1.5) de las siguientes maneras:

- Si A_{ik} resulta ser cero, la transformación de la fila i se omite.
- El índice j en la Ecuación (1.5a) comienza en $k + 1$ en lugar de k . Por lo tanto, A_{ik} no se reemplaza por cero, sino que conserva su valor original. Como la fase de solución nunca accede a la parte triangular inferior de la matriz de coeficientes, su contenido es irrelevante.

Fase de sustitución hacia atrás

Después de la eliminación, la matriz aumentada toma la forma:

$$\left[\begin{array}{cccc|c} A_{11} & A_{12} & \cdots & A_{1n} & b_1 \\ 0 & A_{22} & \cdots & A_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A_{nn} & b_n \end{array} \right]$$

La última ecuación $A_{nn}x_n = b_n$ se resuelve como: $x_n = \frac{b_n}{A_{nn}}$.

Para las anteriores:

$$x_k = \left(b_k - \sum_{j=k+1}^n A_{kj}x_j \right) \frac{1}{A_{kk}}$$

Algoritmo:

```
for k = n:-1:1
    b(k) = (b(k) - A(k,k+1:n)*b(k+1:n)) / A(k,k);
end
```




3.1. gauss

La función `gauss` combina la fase de eliminación y la fase de sustitución hacia atrás. Durante la sustitución hacia atrás, el vector **b** es sobrescrito por el vector solución **x**, de modo que **b** contiene la solución al finalizar la ejecución.

```
1 function [x, determinante] = gauss(A, b)
2 % Resuelve  $A*x = b$  usando eliminación de Gauss y calcula  $\det(A)$ 
3 % USO: [x, det] = gauss(A, b)
4
5 % Asegurar que b sea un vector columna
6 if size(b, 2) > 1
7     b = b';
8 end
9
10 n = length(b);
11
12 % -----
13 % Fase de eliminación
14 % -----
15 for k = 1:n-1
16     for i = k+1:n
17         if A(i,k) ~= 0
18             lambda = A(i,k) / A(k,k);
19             A(i,k+1:n) = A(i,k+1:n) - lambda * A(k,k+1:n);
20             b(i) = b(i) - lambda * b(k);
21         end
22     end
23 end
24
25 % Cálculo del determinante si se solicita
26 if nargin == 2
27     determinante = prod(diag(A));
28 end
29
30 % -----
31 % Fase de sustitución hacia atrás
32 % -----
33 for k = n:-1:1
34     b(k) = (b(k) - A(k,k+1:n) * b(k+1:n)) / A(k,k);
35 end
36
37 x = b;
38 end
```

4. Métodos de Descomposición LU

Introducción

Es posible demostrar que cualquier matriz cuadrada \mathbf{A} puede expresarse como el producto de una matriz triangular inferior \mathbf{L} y una matriz triangular superior \mathbf{U} :

$$\mathbf{A} = \mathbf{LU} \quad (1.6)$$

El proceso de calcular \mathbf{L} y \mathbf{U} para una matriz dada \mathbf{A} se conoce como *descomposición LU* o *factorización LU*. La descomposición LU no es única (las combinaciones posibles de \mathbf{L} y \mathbf{U} para una \mathbf{A} dada son infinitas), a menos que se impongan ciertas restricciones sobre \mathbf{L} o \mathbf{U} . Estas restricciones distinguen un tipo de descomposición de otro. Tres descomposiciones comúnmente usadas se listan en la Tabla 2.

Nombre	Restricciones
Descomposición de Doolittle	$L_{ii} = 1, \quad i = 1, 2, \dots, n$
Descomposición de Crout	$U_{ii} = 1, \quad i = 1, 2, \dots, n$
Descomposición de Choleski	$\mathbf{L} = \mathbf{U}^T$

Cuadro 2: Tipos comunes de descomposición LU

Después de descomponer \mathbf{A} , es fácil resolver las ecuaciones $\mathbf{Ax} = \mathbf{b}$, como se mencionó en el Art. 2.1. Primero reescribimos las ecuaciones como $\mathbf{LUx} = \mathbf{b}$. Usando la notación $\mathbf{Ux} = \mathbf{y}$, las ecuaciones se convierten en

$$\mathbf{Ly} = \mathbf{b}$$

que puede resolverse para \mathbf{y} mediante sustitución hacia adelante. Luego,

$$\mathbf{Ux} = \mathbf{y}$$

proporcionará \mathbf{x} mediante el proceso de sustitución hacia atrás.

La ventaja de la descomposición LU sobre el método de eliminación de Gauss es que, una vez que \mathbf{A} ha sido descompuesta, podemos resolver $\mathbf{Ax} = \mathbf{b}$ para tantos vectores constantes \mathbf{b} como deseemos. El costo de cada solución adicional es relativamente pequeño, ya que las operaciones de sustitución hacia adelante y hacia atrás consumen mucho menos tiempo que el proceso de descomposición.

4.1. Método de Descomposición de Doolittle

Fase de descomposición La descomposición de Doolittle está estrechamente relacionada con la eliminación de Gauss. Para ilustrar esta relación, consideremos una matriz \mathbf{A} de 3×3 y asumamos que existen matrices triangulares

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ L_{21} & 1 & 0 \\ L_{31} & L_{32} & 1 \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & 0 & U_{33} \end{bmatrix}$$

tal que $\mathbf{A} = \mathbf{L}\mathbf{U}$. Al completar la multiplicación en el lado derecho, obtenemos:

$$\mathbf{A} = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ U_{11}L_{21} & U_{12}L_{21} + U_{22} & U_{13}L_{21} + U_{23} \\ U_{11}L_{31} & U_{12}L_{31} + U_{22}L_{32} & U_{13}L_{31} + U_{23}L_{32} + U_{33} \end{bmatrix} \quad (1.7)$$

Apliquemos ahora la eliminación de Gauss a la Ec. (1.7). El primer paso del procedimiento consiste en elegir la primera fila como pivote y aplicar las operaciones elementales:

$$\text{fila 2} \leftarrow \text{fila 2} - L_{21} \times \text{fila 1} \quad (\text{elimina } A_{21})$$

$$\text{fila 3} \leftarrow \text{fila 3} - L_{31} \times \text{fila 1} \quad (\text{elimina } A_{31})$$

El resultado es:

$$\mathbf{A}' = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & U_{22}L_{32} & U_{23}L_{32} + U_{33} \end{bmatrix}$$

En el siguiente paso tomamos la segunda fila como pivote y utilizamos la operación:

$$\text{fila 3} \leftarrow \text{fila 3} - L_{32} \times \text{fila 2} \quad (\text{elimina } A_{32})$$

obteniendo finalmente:

$$\mathbf{A}'' = \mathbf{U} = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & 0 & U_{33} \end{bmatrix}$$

La ilustración anterior revela dos características importantes de la descomposición de Doolittle:

- La matriz \mathbf{U} es idéntica a la matriz triangular superior que resulta de la eliminación de Gauss.
- Los elementos fuera de la diagonal de \mathbf{L} son los multiplicadores de la ecuación pivote usados durante la eliminación de Gauss; es decir, L_{ij} es el multiplicador que eliminó A_{ij} .

Es práctica común almacenar los multiplicadores en la parte triangular inferior de la matriz de coeficientes, reemplazando los coeficientes a medida que son eliminados (L_{ij} reemplazando a A_{ij}). Los elementos diagonales de \mathbf{L} no necesitan almacenarse, ya que se entiende que todos ellos son iguales a uno. La forma final de la matriz de coeficientes sería entonces la siguiente mezcla de \mathbf{L} y \mathbf{U} :

$$[\mathbf{L} \setminus \mathbf{U}] = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ L_{21} & U_{22} & U_{23} \\ L_{31} & L_{32} & U_{33} \end{bmatrix} \quad (1.8)$$

El algoritmo para la descomposición de Doolittle es, por lo tanto, idéntico al procedimiento de eliminación de Gauss en **gauss**, salvo que ahora cada multiplicador λ se almacena en la parte triangular inferior de \mathbf{A} .

4.1.1. LUdec

En esta versión de la descomposición LU, la matriz original \mathbf{A} es destruida y reemplazada por su forma descompuesta $[\mathbf{L} \setminus \mathbf{U}]$.

```

1 function A = LUdec(A)
2 % LU decomposition of matrix A; returns A = [L\U]
3 % USO: A = LUdec(A)
4
5 n = size(A, 1);
6
7 for k = 1:n-1
8     for i = k+1:n
9         if A(i,k) ~= 0.0
10             lambda = A(i,k) / A(k,k);
11             A(i,k+1:n) = A(i,k+1:n) - lambda * A(k,k+1:n);
12             A(i,k) = lambda;
13         end
14     end
15 end

```

Fase de solución

Consideremos ahora el procedimiento para resolver $\mathbf{Ly} = \mathbf{b}$ mediante sustitución hacia adelante. La forma escalar de las ecuaciones es (recordando que $L_{ii} = 1$):

$$\begin{aligned}
 y_1 &= b_1 \\
 L_{21}y_1 + y_2 &= b_2 \\
 &\vdots \\
 L_{k1}y_1 + L_{k2}y_2 + \cdots + L_{k,k-1}y_{k-1} + y_k &= b_k \\
 &\vdots
 \end{aligned}$$

Resolviendo la ecuación k -ésima para y_k se obtiene:

$$y_k = b_k - \sum_{j=1}^{k-1} L_{kj}y_j, \quad k = 2, 3, \dots, n \quad (1.9)$$

Haciendo que \mathbf{y} sobrescriba a \mathbf{b} , obtenemos el algoritmo de sustitución hacia adelante:

```

for k = 2:n
    y(k) = b(k) - A(k,1:k-1)*y(1:k-1);
end

```

La fase de sustitución hacia atrás para resolver $\mathbf{Ux} = \mathbf{y}$ es idéntica a la utilizada en el método de eliminación de Gauss.

4.1.2. LUso1

Esta función lleva a cabo la fase de solución (sustitución hacia adelante y hacia atrás). Se asume que la matriz de coeficientes original ha sido descompuesta, de modo que la entrada es $\mathbf{A} = [\mathbf{L} \setminus \mathbf{U}]$. El contenido de \mathbf{b} es reemplazado por \mathbf{y} durante la sustitución hacia adelante. De manera similar, la sustitución hacia atrás sobrescribe \mathbf{y} con la solución \mathbf{x} .

```

1 function x = LUso1(A, b)
2 % Resuelve  $\mathbf{L} \cdot \mathbf{U} \cdot \mathbf{x} = \mathbf{b}$ , donde A contiene tanto L como U;
3 % es decir, A tiene la forma  $[\mathbf{L} \setminus \mathbf{U}]$ .
4 % USO:  $\mathbf{x} = \text{LUso1}(\mathbf{A}, \mathbf{b})$ 
5 % Asegurar que b sea un vector columna
6 if size(b, 2) > 1
7     b = b';
8 end
9 n = length(b);
10 % Sustitución hacia adelante ( $\mathbf{L}\mathbf{y} = \mathbf{b}$ )
11 for k = 2:n
12     b(k) = b(k) - A(k, 1:k-1) * b(1:k-1);
13 end
14 % Sustitución hacia atrás ( $\mathbf{U}\mathbf{x} = \mathbf{y}$ )
15 for k = n:-1:1
16     b(k) = (b(k) - A(k, k+1:n) * b(k+1:n)) / A(k, k);
17 end
18 x = b;
19 end

```

4.2. Descomposición de Choleski

La descomposición de Choleski $\mathbf{A} = \mathbf{L}\mathbf{L}^T$ tiene dos limitaciones:

- Como el producto matricial $\mathbf{L}\mathbf{L}^T$ es simétrico, la descomposición de Choleski requiere que \mathbf{A} sea *simétrica*.
- El proceso de descomposición implica extraer raíces cuadradas de ciertas combinaciones de los elementos de \mathbf{A} . Puede demostrarse que las raíces cuadradas de números negativos solo pueden evitarse si \mathbf{A} es *definida positiva*.

Aunque el número de operaciones largas en todos los métodos de descomposición es aproximadamente el mismo, la descomposición de Choleski no es un método particularmente popular para resolver ecuaciones simultáneas, principalmente debido a las restricciones mencionadas. La estudiamos aquí porque es de gran valor en ciertas otras aplicaciones (por ejemplo, en la transformación de problemas de autovalores).

Comencemos analizando la descomposición de Choleski:

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T \quad (1.10)$$

para una matriz de 3×3 :

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{bmatrix}$$

Después de realizar la multiplicación matricial en el lado derecho, se obtiene:

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} L_{11}^2 & L_{11}L_{21} & L_{11}L_{31} \\ L_{11}L_{21} & L_{21}^2 + L_{22}^2 & L_{21}L_{31} + L_{22}L_{32} \\ L_{11}L_{31} & L_{21}L_{31} + L_{22}L_{32} & L_{31}^2 + L_{32}^2 + L_{33}^2 \end{bmatrix} \quad (1.11)$$

Nótese que la matriz del lado derecho es simétrica, como se mencionó antes. Igualando las matrices \mathbf{A} y \mathbf{LL}^T elemento a elemento, obtenemos seis ecuaciones (debido a la simetría). Solo es necesario considerar los elementos triangulares inferiores (o superiores) en las seis incógnitas de los componentes de \mathbf{L} . Resolviendo estas ecuaciones en cierto orden, es posible que cada una tenga solo una incógnita.

Consideremos la parte triangular inferior de cada matriz en la Ec. (1.11) (la parte superior también serviría). Igualando los elementos en la primera columna, y empezando desde la primera fila hacia abajo, podemos calcular L_{11}, L_{21}, L_{31} en ese orden:

$$\begin{aligned} A_{11} &= L_{11}^2 & \Rightarrow & L_{11} = \sqrt{A_{11}} \\ A_{21} &= L_{11}L_{21} & \Rightarrow & L_{21} = A_{21}/L_{11} \\ A_{31} &= L_{11}L_{31} & \Rightarrow & L_{31} = A_{31}/L_{11} \end{aligned}$$

La segunda columna, comenzando con la segunda fila, nos da L_{22} y L_{32} :

$$\begin{aligned} A_{22} &= L_{21}^2 + L_{22}^2 & \Rightarrow & L_{22} = \sqrt{A_{22} - L_{21}^2} \\ A_{32} &= L_{21}L_{31} + L_{22}L_{32} & \Rightarrow & L_{32} = (A_{32} - L_{21}L_{31})/L_{22} \end{aligned}$$

Finalmente, la tercera columna y tercera fila nos da L_{33} :

$$A_{33} = L_{31}^2 + L_{32}^2 + L_{33}^2 \quad \Rightarrow \quad L_{33} = \sqrt{A_{33} - L_{31}^2 - L_{32}^2}$$

Podemos ahora extrapolar los resultados para una matriz $n \times n$. Observamos que un elemento típico en la parte triangular inferior de \mathbf{LL}^T tiene la forma:

$$(\mathbf{LL}^T)_{ij} = L_{i1}L_{j1} + L_{i2}L_{j2} + \cdots + L_{ij}L_{jj} = \sum_{k=1}^j L_{ik}L_{jk}, \quad i \geq j$$

Igualando este término con el elemento correspondiente de \mathbf{A} , obtenemos:

$$A_{ij} = \sum_{k=1}^j L_{ik}L_{jk}, \quad i = j, j+1, \dots, n; \quad j = 1, 2, \dots, n \quad (1.12)$$

El rango de los índices mostrados limita los elementos a la parte triangular inferior. Para la primera columna ($j = 1$), obtenemos de la Ec.(1.12):

$$L_{11} = \sqrt{A_{11}} \quad \text{y} \quad L_{i1} = \frac{A_{i1}}{L_{11}}, \quad i = 2, 3, \dots, n \quad (1.13)$$

Al continuar con otras columnas, observamos que la incógnita en la Ec. (1.12) es L_{ij} (los otros elementos de L en la ecuación ya han sido calculados). Separando el término L_{ij} fuera de la suma en la Ec. (1.12), obtenemos:

$$A_{ij} = \sum_{k=1}^{j-1} L_{ik}L_{jk} + L_{ij}L_{jj}$$

Si $i = j$ (un término diagonal), la solución es

$$L_{jj} = \sqrt{A_{jj} - \sum_{k=1}^{j-1} L_{jk}^2}, \quad j = 2, 3, \dots, n \quad (1.14)$$

Para un término no diagonal se obtiene:

$$L_{ij} = \left(A_{ij} - \sum_{k=1}^{j-1} L_{ik}L_{jk} \right) / L_{jj}, \quad j = 2, 3, \dots, n-1, \quad i = j+1, j+2, \dots, n \quad (1.15)$$

4.2.1. choleski

Nótese que en las Ecs.(1.14) y (1.15), A_{ij} aparece solo en la fórmula para L_{ij} . Por lo tanto, una vez que L_{ij} ha sido calculado, A_{ij} ya no es necesario. Esto permite escribir los elementos de \mathbf{L} directamente sobre la parte triangular inferior de \mathbf{A} a medida que se calculan. Los elementos por encima de la diagonal principal de \mathbf{A} permanecerán sin cambios.

```

1 function L = choleski(A)
2 % Computes L in Choleski's decomposition A = L*L'.
3 % USO: L = choleski(A)
4 n = size(A,1);
5 for j = 1:n
6     temp = A(j,j) - dot(A(j,1:j-1), A(j,1:j-1));
7     if temp < 0.0
8         error('Matrix is not positive definite')
9     end
10    A(j,j) = sqrt(temp);
11    for i = j+1:n
12        A(i,j) = (A(i,j) - dot(A(i,1:j-1), A(j,1:j-1))) / A(j,j);
13    end
14 end
15 L = tril(A);

```



Al finalizar la descomposición, \mathbf{L} se puede extraer utilizando el comando de MATLAB `tril(A)`. Si durante la descomposición se encuentra un valor negativo para L_{jj}^2 , se muestra un mensaje de error y el programa termina.

5. Problemas Resueltos

Ejemplo 5.1. Una matriz de Vandermonde $n \times n$ está definida por

$$A_{ij} = v_i^{n-j}, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, n$$

donde \mathbf{v} es un vector. En MATLAB, una matriz de Vandermonde puede generarse mediante el comando `vander(v)`. Se usa la función `gauss` para calcular la solución de $\mathbf{Ax} = \mathbf{b}$, donde \mathbf{A} es la matriz de Vandermonde 6×6 generada a partir del vector

$$\mathbf{v} = [1.0 \ 1.2 \ 1.4 \ 1.6 \ 1.8 \ 2.0]^T \quad \text{y} \quad \mathbf{b} = [0 \ 1 \ 0 \ 1 \ 0 \ 1]^T$$

También se debe evaluar la precisión de la solución (las matrices de Vandermonde tienden a estar mal condicionadas).

Solución Utilizamos el programa que se muestra a continuación. Después de construir \mathbf{A} y \mathbf{b} , el formato de salida se cambió a `long` para que la solución se imprimiera con 14 cifras decimales.

```
1 % Example 5.1 (Gauss elimination)
2 A = vander(1:0.2:2);
3 b = [0 1 0 1 0 1]';
4 format long
5 [x, determinante] = gauss(A, b);
6 x
7 determinante
```

A continuación se presentan los resultados:

```
1 x =
2
3     1.0e+04 *
4
5     0.0416666666667010
6    -0.3125000000002465
7     0.9250000000006972
8    -1.3500000000009722
9     0.9709333333340017
10    -0.2751000000001813
11
12 det =
13    -1.132462079991823e-06
```


Como el determinante es bastante pequeño en relación con los elementos de \mathbf{A} (puede imprimir \mathbf{A} para verificarlo), se espera un error de redondeo detectable. La inspección de \mathbf{x} nos lleva a sospechar que la solución exacta es

$$\mathbf{x} = [1250/3 \quad -3125 \quad 9250 \quad -13500 \quad 29128/3 \quad -2751]^T$$

en cuyo caso la solución numérica sería precisa hasta 9 cifras decimales.

Ejemplo 5.2. Resolver $\mathbf{AX} = \mathbf{B}$ usando la descomposición de Doolittle y calcular $|\mathbf{A}|$, donde

$$\mathbf{A} = \begin{bmatrix} 3 & -1 & 4 \\ -2 & 0 & 5 \\ 7 & 2 & -2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 6 & -4 \\ 3 & 2 \\ 7 & -5 \end{bmatrix}$$

Solución En el siguiente programa, la matriz de coeficientes \mathbf{A} es primero descompuesta mediante la llamada a `LUdec`. Luego, `LUsol` se usa para calcular la solución, un vector a la vez.

```
1 % Example 5.2 (Doolittle's decomposition)
2 A = [3 -1 4; -2 0 5; 7 2 -2];
3 B = [6 -4; 3 2; 7 -5];
4 % Descomposición LU de A
5 A = LUdec(A);
6 % Cálculo del determinante de A
7 determinate = prod(diag(A))
8 % Resolución de AX = B, columna por columna
9 for i = 1:size(B, 2)
10     X(:, i) = LUsol(A, B(:, i));
11 end
12 % Mostrar la solución
13 X
```

Aquí los resultados:

```
1 determinate =
2     -77
3
4 X =
5     1.0000    -1.0000
6     1.0000     1.0000
7     1.0000     0.0000
```

Ejemplo 5.3. Probar la función `choleski` descomponiendo

$$\mathbf{A} = \begin{bmatrix} 1.44 & -0.36 & 5.52 & 0.00 \\ -0.36 & 10.33 & -7.78 & 0.00 \\ 5.52 & -7.78 & 28.40 & 9.00 \\ 0.00 & 0.00 & 9.00 & 61.00 \end{bmatrix}$$



```

1 % Example 5.3 (Choleski decomposition)
2 A = [ 1.44 -0.36 5.52 0.00;
3      -0.36 10.33 -7.78 0.00;
4      5.52 -7.78 28.40 9.00;
5      0.00 0.00 9.00 61.00 ];
6 % Ejecutar la descomposición de Choleski
7 L = choleski(A);
8 % Verificar el resultado
9 Check = L * L'

1 >> L
2
3      1.2000      0      0      0
4     -0.3000     3.2000      0      0
5     4.6000    -2.0000     1.8000      0
6      0      0     5.0000     6.0000
7
8 >> Check
9
10     1.4400    -0.3600     5.5200      0
11    -0.3600    10.3300    -7.7800      0
12     5.5200    -7.7800    28.4000     9.0000
13      0      0     9.0000    61.0000

```

6. Problemas Propuestos

Problema 6.1. Modifique la función `gauss` para que funcione con m vectores constantes. Pruebe el programa resolviendo $\mathbf{AX} = \mathbf{B}$, donde

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

Problema 6.2. Un ejemplo conocido de una matriz mal condicionada es la *matriz de Hilbert*

$$\mathbf{A} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \dots \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \dots \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Escriba un programa que resuelva $\mathbf{Ax} = \mathbf{b}$ mediante el método de descomposición de Doolittle, donde \mathbf{A} es una matriz de Hilbert de tamaño arbitrario $n \times n$ y

$$b_i = \sum_{j=1}^n A_{ij}$$



El programa no debe tener entrada aparte de n . Al ejecutar el programa, determine el valor máximo de n para el cual la solución esté dentro de 6 cifras significativas de la solución exacta

$$\mathbf{x} = [1 \ 1 \ 1 \ \dots]^T$$

Problema 6.3. Escriba una función para la fase de solución del método de descomposición de Choleski. Pruebe la función resolviendo $\mathbf{Ax} = \mathbf{b}$, donde

$$\mathbf{A} = \begin{bmatrix} 4 & -2 & 2 \\ -2 & 2 & -4 \\ 2 & -4 & 11 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 6 \\ -10 \\ 27 \end{bmatrix}$$

Use la función `choleski` para la fase de descomposición.

Problema 6.4. Resuelva el sistema $\mathbf{Ax} = \mathbf{b}$, donde

$$\mathbf{A} = \begin{bmatrix} 3.50 & 2.77 & -0.76 & 1.80 \\ -1.80 & 2.68 & 3.44 & -0.09 \\ 0.27 & 5.07 & 6.90 & 1.61 \\ 1.71 & 5.45 & 2.68 & 1.71 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 7.31 \\ 4.23 \\ 13.85 \\ 11.55 \end{bmatrix}$$

Calcule $|\mathbf{A}|$ y \mathbf{Ax} y comente sobre la precisión de la solución.