6 Una introducción a los métodos gaussianos para sistemas lineales de ecuaciones

Objetivos

- Presentar y clasificar los métodos de resolución numérica de sistemas lineales de ecuaciones.
- Estudiar detalladamente los métodos directos: de solución inmediata, de eliminación y de descomposición.
- Desarrollar el análisis matricial del método de Gauss.

6.1 Consideraciones generales

6.1.1 Introducción

El objetivo de este tema es introducir al lector en la resolución de sistemas lineales de ecuaciones por métodos gaussianos. Parece conveniente, en primer lugar, establecer la notación y algunas de las bases de álgebra lineal necesarias para alcanzar el objetivo planteado.

Siguiendo la notación introducida por Householder en 1964, en general se emplean

mayúsculas en negrita	$\boldsymbol{A}, \boldsymbol{L}, \boldsymbol{U}, \boldsymbol{\Delta}, \boldsymbol{\Lambda}$	para las matrices,
minúsculas con subíndices	$a_{ij}, l_{ij}, u_{ij}, \delta_{ij}, \lambda_{ij}$	para los coeficientes de matrices,
minúsculas en negrita	$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}$	para los vectores,
letras griegas en minúscula	$\alpha, \beta, \gamma, \delta, \theta, \mu$	para los escalares.

El espacio vectorial de las matrices reales $m \times n$ se denota por $\mathbb{R}^{m \times n}$; un elemento cualquiera de ese espacio, $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, es una matriz rectangular de m filas y n columnas que puede escribirse

como

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

Si $m = n$ entonces $\boldsymbol{A}$ es cuadrada y se dice que tiene orden n. De la misma forma, $\mathbb{C}^{m\times n}$ es el espacio vectorial de las matrices de coeficientes complejos.

Los vectores, que pueden ser interpretados como un caso particular del anterior con $\mathbb{R}^{n\times 1}$ (equivalente a $\mathbb{R}^n$), siempre se asumen como *vectores columna*, es decir $\boldsymbol{x} \in \mathbb{R}^n$ es

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix}$$

donde las componentes x_i son números reales. Puesto que por convención se toman los vectores como columna, objetos del tipo $(x_1\ x_2\ \cdots\ x_{n-1}\ x_n)$ o bien $(x_1, \ldots, x_n)$ son *vectores fila* y se denotan por $\boldsymbol{x}^T$ (^{T} indica matriz o vector traspuesto).

Además de las operaciones inherentes al espacio vectorial (suma interna y producto exterior por reales) conviene resaltar por su importancia el producto escalar de vectores. Si $\boldsymbol{x}$ e $\boldsymbol{y}$ son dos vectores de $\mathbb{R}^n$, entonces $\boldsymbol{x}^T\boldsymbol{y} = x_1y_1 + x_2y_2 + \cdots + x_ny_n \in \mathbb{R}$. Nótese, que de forma equivalente, si $\boldsymbol{A}$ y $\boldsymbol{B}$ son dos matrices de $\mathbb{R}^{m\times n}$, entonces, $\boldsymbol{A}^T\boldsymbol{B} \in \mathbb{R}^{n\times n}$. El producto escalar de vectores permite definir una métrica: la norma euclídea de $\boldsymbol{x}$, que es simplemente $\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^T\boldsymbol{x}}$.

Del mismo modo que se ha definido $\boldsymbol{x}^T\boldsymbol{y}$, también se puede definir $\boldsymbol{x}\boldsymbol{y}^T$. Sin embargo, el significado de este último producto entre vectores es radicalmente distinto. Sean $\boldsymbol{x}$ e $\boldsymbol{y}$ dos vectores no nulos; la matriz $\boldsymbol{x}\boldsymbol{y}^T$ es de $\mathbb{R}^{n\times n}$, se escribe como

$$\boldsymbol{x}\boldsymbol{y}^T = \begin{pmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_{n-1} & x_1y_n \\ x_2y_1 & x_2y_2 & \cdots & x_2y_{n-1} & x_2y_n \\ \vdots & \vdots & & \vdots & \vdots \\ x_{n-1}y_1 & x_{n-1}y_2 & \cdots & x_{n-1}y_{n-1} & x_{n-1}y_n \\ x_ny_1 & x_ny_2 & \cdots & x_ny_{n-1} & x_ny_n \end{pmatrix}$$

y todas sus columnas (y filas) definen vectores paralelos, es decir, elementos de un espacio vectorial de dimensión uno. Por consiguiente, esta matriz es de rango uno. En realidad, toda matriz de rango uno puede expresarse como el producto de dos vectores de la forma $\boldsymbol{x}\boldsymbol{y}^T$. Estas matrices son comunes en métodos numéricos y conviene saber trabajar con ellas. Por ejemplo, su almacenamiento no se hace guardando todos los coeficientes de la matriz, lo que implicaría almacenar n^2 números reales; estas matrices se almacenan guardando únicamente las componentes de los vectores $\boldsymbol{x}$ e $\boldsymbol{y}$, es decir $2n$ números reales. Para tener una idea del ahorro computacional que esto representa basta suponer que $n = 1000$: mientras almacenar $\boldsymbol{x}$ e $\boldsymbol{y}$ sólo necesita de 2000 variables reales, $\boldsymbol{x}\boldsymbol{y}^T$ requiere un millón (es decir, 500 veces más memoria).

6.1.2 Planteamiento general

A lo largo de este tema se plantea la resolución de sistemas lineales de ecuaciones

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{6.1}$$

donde $\boldsymbol{A}$ es una matriz de $n \times n$ coeficientes reales a_{ij} con $i = 1,\ldots,n$, $j = 1,\ldots,n$; $\boldsymbol{b}$ es el término independiente, también de coeficientes reales, $\boldsymbol{b}^T = (b_1,\ldots,b_n)$; y finalmente $\boldsymbol{x}^T = (x_1,\ldots,x_n)$ es el vector solución del sistema.

La existencia y unicidad de soluciones del sistema definido en 6.1 es por fortuna un tema ampliamente estudiado en el álgebra lineal. Precisamente, el álgebra lineal proporciona una serie de condiciones que permiten verificar si 6.1 tiene solución:

Si $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, entonces las siguientes afirmaciones son equivalentes:

1. Para cualquier $\boldsymbol{b} \in \mathbb{R}^n$, el sistema $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ tiene solución.
2. Si $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ tiene solución, ésta es única.
3. Para cualquier $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \implies \boldsymbol{x} = \boldsymbol{0}$.
4. Las columnas (filas) de $\boldsymbol{A}$ son linealmente independientes.
5. Existe $\boldsymbol{A}^{-1}$ matriz inversa de $\boldsymbol{A}$ tal que $\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{I}$ ($\boldsymbol{I}$ matriz identidad de orden n).
6. $\det(\boldsymbol{A}) = |\boldsymbol{A}| \neq 0$.

A pesar de la indudable importancia de todas estas condiciones, en el ámbito de la resolución numérica de sistemas de ecuaciones deben ser empleadas con sumo cuidado.

6.1.3 Resolución algebraica: método de Cramer

A continuación se plantea la resolución analítica de problemas muy pequeños siguiendo un posible enfoque algebraico clásico. El sistema de ecuaciones planteado en 6.1 tiene solución única si y sólo si $\det(\boldsymbol{A}) = |\boldsymbol{A}| \neq 0$. En este caso, existe la matriz inversa de $\boldsymbol{A}$, $\boldsymbol{A}^{-1}$, que permite escribir la solución del sistema de ecuaciones como

$$\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{6.2}$$

La ecuación anterior no es sólo una expresión formal de la solución sino que describe un posible algoritmo que permitiría obtenerla:

1. `Calcular` $|\boldsymbol{A}|$.
2. `Si` $|\boldsymbol{A}| = 0$ `indicar que la matriz es singular y FIN.`
3. `Calcular la inversa` $\boldsymbol{C} = \boldsymbol{A}^{-1}$.
4. `Calcular la solución` $\boldsymbol{x} = \boldsymbol{C}\boldsymbol{b}$.
5. `Escribir la solución y FIN.`

A pesar de tener todo el fundamento analítico necesario, este algoritmo para obtener la solución de 6.1 es el peor método posible desde un punto de vista numérico. De hecho, salvo en contadas excepciones, este algoritmo está condenado al más absoluto fracaso. Para darse cuenta de ello, basta observar sólo dos de los problemas que plantea.

En primer lugar, el cálculo del determinante puede ser bastante tedioso puesto que el determinante puede variar bruscamente con pequeños escalados de la matriz. Obsérvese que si $\boldsymbol{A}$ es de orden n, entonces $\det(\gamma\boldsymbol{A}) = \gamma^n \det(\boldsymbol{A})$. Para ver las implicaciones que esta igualdad impone basta tomar el caso particular de $n = 100$ (número de ecuaciones pequeño hoy en día), entonces: $\det(0.1\ \boldsymbol{A}) = 10^{-100} \det(\boldsymbol{A})$. Es decir, dividiendo los coeficientes de $\boldsymbol{A}$ por diez, se reduce el determinante de $\boldsymbol{A}$ en un factor de 10^{-100}. Por consiguiente, es muy difícil determinar numéricamente si el determinante de una matriz es realmente nulo. El uso del determinante se centra básicamente en estudios teóricos.

En segundo lugar, el cálculo de la inversa de $\boldsymbol{A}$ (que presenta serios problemas asociados al almacenamiento de la matriz y a la precisión con la que obtengan los resultados), no se emplearía ni en el caso escalar ($n = 1$). Por ejemplo, para resolver $15x = 3$ no se evaluaría primero $c = 1/15$ para después calcular x como $x = 3c$. Lo más lógico sería dividir directamente 3 por 15, $x = 3/15$, lo que permitiría ahorrarse una operación y un error de almacenamiento. Esta situación puede extrapolarse al caso de orden n donde la diferencia en número de operaciones es muy considerable y además los errores de redondeo pueden dar lugar a inestabilidades numéricas.

A continuación se estudia el *método de Cramer*. Este método es una mejora del algoritmo anterior puesto que permite realizar los pasos 3 y 4 de una sola vez. A pesar de ello no es un método adecuado desde un punto de vista numérico.

La expresión general de la solución por el método de Cramer es:

$$x_i = \frac{\begin{vmatrix} a_{11} & \dots & a_{1,i-1} & b_1 & a_{1,i+1} & \dots & a_{1n} \\ a_{21} & \dots & a_{2,i-1} & b_2 & a_{2,i+1} & \dots & a_{2n} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & \dots & a_{n,i-1} & b_n & a_{n,i+1} & \dots & a_{nn} \end{vmatrix}}{|\boldsymbol{A}|} \qquad i = 1, \dots, n \tag{6.3}$$

Es interesante evaluar el número de operaciones elementales (sumas, productos y divisiones) necesarias para obtener la solución del sistema. En primer lugar hay que evaluar $n + 1$ determinantes y luego realizar n divisiones. Para el cálculo de los determinantes, una de las posibles técnicas necesita de $n!$ n multiplicaciones y $n!-1$ sumas. Por consiguiente, el método de Cramer necesita de

$$\begin{cases} (n+1)\ (n!-1) & \text{sumas} \\ (n+1)\ n!\ n & \text{productos} \\ n & \text{divisiones} \end{cases}$$

Cada operación elemental puede tener un coste computacional distinto (por ejemplo, muchos ordenadores dividen empleando el método de Newton para ceros de funciones). A pesar de ello, aquí se les asignará el mismo coste computacional a todas las operaciones elementales

puesto que ello ya permite realizar las comparaciones pertinentes. El número de operaciones elementales con el método de Cramer es $T_C = (n+1)^2\, n! - 1$. La tabla 6.1 muestra los valores de T_C para diferentes tamaños del sistema de ecuaciones.

Tabla 6.1 Operaciones elementales del método de Cramer según el tamaño de la matriz (n)

n	T_C
5	4 319
10	4×10^8
100	10^{158}

Los números presentados en la tabla 6.1 adquieren un mayor relieve cuando se asocian al tiempo necesario para efectuarlos. Si se dispusiera de un superordenador capaz de realizar 100 millones de operaciones en coma flotante por segundo (100 Mflops), el sistema de $n = 100$ necesitaría aproximadamente $3 \cdot 10^{142}$ *años* para ser resuelto. Es evidente que el número de operaciones asociado a este método hace prohibitivo su uso, aún para sistemas pequeños. Si además se tiene en cuenta que el *ENIAC* (primer ordenador digital, fabricado en 1940) realizaba sólo 300 operaciones por segundo y tenía un tiempo medio entre averías de 12 horas, se comprenderá por qué la resolución de sistemas lineales de ecuaciones está en el origen del desarrollo de los métodos numéricos.

6.1.4 Resolución numérica: un enfoque global

La estrategia y metodología que se aplica a la resolución numérica de sistemas lineales de ecuaciones parte de una filosofía distinta a la expuesta anteriormente. La regularidad de la matriz $\boldsymbol{A}$ no se determina por un cálculo previo de su determinante. Sin embargo, en algunos problemas se puede estudiar la regularidad de $\boldsymbol{A}$ en función de su origen (por ejemplo cuando proviene de la discretización de ecuaciones diferenciales) o a partir de propiedades fácilmente computables como la dominancia diagonal. Por lo general se aplica alguno de los métodos de resolución que se verán seguidamente sin evaluar previamente el determinante; en muchas ocasiones el determinante y la inversa de la matriz son un subproducto de los cálculos efectuados.

En realidad los algoritmos eficaces para la resolución de sistemas lineales plantean procesos con un enfoque radicalmente distinto, sobretodo desde una perspectiva que contempla el hecho de que los cálculos se realizan en ordenadores digitales. Por consiguiente, es lógico que los algoritmos se evalúen en función de su eficacia y siguiendo criterios directamente relacionados con su implementación en ordenadores digitales. Existen tres criterios fundamentales para analizar los algoritmos:

1. *Número de operaciones necesarias*, íntimamente ligado al tiempo de CPU. Se tendrán en cuenta las operaciones elementales entre números en coma flotante (*flop*): `+`, `-`, `/` ó `*`, todas a un mismo coste computacional aunque no sea exactamente cierto. El número de operaciones es obviamente un excelente indicador del coste computacional pero no debe tomarse

en un sentido estricto. De hecho, multiplicar el tiempo necesario para una operación por el número de operaciones siempre infravalora el tiempo necesario del algoritmo. Además del tiempo invertido en efectuar las operaciones hay una sobrecarga, debido a la gestión de la memoria, al manejo de los índices enteros, a las instrucciones lógicas en los propios bucles, etc. A pesar de ello, y por fortuna, el número de operaciones es un buen indicador del tiempo de CPU porque esta sobrecarga es generalmente proporcional al número de operaciones, de forma que, aunque no se pueda predecir exactamente el tiempo de CPU, se puede saber cómo varía (linealmente, cuadráticamente, etc.) al modificar, por ejemplo, el orden n de la matriz.

2. *Necesidades de almacenamiento*, que inciden clara y directamente en las limitaciones de la memoria de los diversos ordenadores; los diferentes métodos de resolución requieren almacenar las matrices de distinta forma en el ordenador y esto varía considerablemente las necesidades de memoria.

3. *Rango de aplicabilidad*: no todos los métodos sirven para cualquier matriz no singular; además, en función del método y de las propiedades de la matriz, la precisión de los resultados puede verse afectada dramáticamente. Como se verá más adelante, pequeños errores de redondeo pueden producir errores en la *solución numérica* completamente desproporcionados. No se debe olvidar que debido al enorme número de operaciones necesarias para la resolución de un sistema de ecuaciones de tamaño medio–grande, el análisis estándar de propagación de errores de redondeo no es en absoluto trivial.

Conviene resaltar que cada uno de estos criterios puede ser determinante para rechazar un algoritmo. Por ejemplo, para un tipo de ordenador dado, métodos que impliquen exceder la memoria disponible son inutilizables por muy rápidos y precisos que resulten. Por lo tanto, el desarrollo de los algoritmos que se plantean a continuación debe tener presentes estos tres criterios simultáneamente.

Desde un punto de vista general las matrices más usuales en las ciencias aplicadas y en ingeniería pueden englobarse en dos grandes categorías:

1. Matrices llenas pero no muy grandes. Por *llenas* se entiende que poseen pocos elementos nulos y por no muy grandes que el número de ecuaciones es de unos pocos miles a lo sumo. Estas matrices aparecen en problemas estadísticos, matemáticos, físicos e ingenieriles.

2. Matrices vacías y muy grandes. En oposición al caso anterior, *vacías* indica que hay pocos elementos no nulos y además están situados con una cierta regularidad. En la mayoría de estos casos el número de ecuaciones supera los miles y puede llegar en ocasiones a los millones. Estas matrices son comunes en la resolución de ecuaciones diferenciales de problemas de ingeniería.

Parece lógico que los métodos para resolver sistemas lineales de ecuaciones se adecuen a las categorías de matrices anteriormente expuestas. En general los *métodos directos* se aplican al primer tipo de matrices, mientras que los *métodos iterativos* se emplean con el segundo

grupo. Es importante observar que no existen reglas absolutas y que todavía en la actualidad existe cierta controversia sobre los métodos óptimos a aplicar en cada caso. En particular, la distinción establecida entre matrices llenas y vacías depende en gran medida del ordenador disponible (fundamentalmente de la memoria). De hecho, los límites han ido evolucionando a lo largo de los años a medida que también evolucionaban los ordenadores (cada vez más rápidos y con más memoria, o al menos más barata). A pesar de ello, casi nadie recomendaría métodos iterativos para matrices llenas con pocas ecuaciones; en cambio, algunos autores trabajan con métodos directos altamente sofisticados y particularizados al entorno informático disponible para resolver sistemas con varios millones de ecuaciones.

Observación: En todo lo que sigue se supone que $\boldsymbol{A}$ y $\boldsymbol{b}$ son de coeficientes reales. Si los elementos de $\boldsymbol{A}$ o $\boldsymbol{b}$ son complejos, aparte de las generalizaciones de los métodos que aquí se estudian o de los algoritmos específicos para este caso, se puede replantear el problema como un sistema lineal, con matriz y término independiente reales, de $2n$ ecuaciones e incógnitas. Para ello se escriben la matriz y los vectores de la siguiente manera:

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{C} + i\boldsymbol{D} \\ \boldsymbol{b} &= \boldsymbol{c} + i\boldsymbol{d} \\ \boldsymbol{x} &= \boldsymbol{y} + i\boldsymbol{z} \end{aligned}$$

donde $\boldsymbol{C}$ y $\boldsymbol{D}$ son matrices reales $n \times n$, y $\boldsymbol{c}$ $\boldsymbol{d}$, $\boldsymbol{y}$ y $\boldsymbol{z}$ son de $\mathbb{R}^n$. El sistema lineal de ecuaciones original se escribe ahora como:

$$\begin{pmatrix} \boldsymbol{C} & -\boldsymbol{D} \\ \boldsymbol{D} & \boldsymbol{C} \end{pmatrix} \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{z} \end{pmatrix} = \begin{pmatrix} \boldsymbol{c} \\ \boldsymbol{d} \end{pmatrix}$$

que es el resultado deseado. □

6.2 Métodos directos

6.2.1 Introducción

Los *métodos directos* de resolución de sistemas lineales de ecuaciones son aquellos que permiten obtener la solución después de un número finito de operaciones aritméticas. Este número de operaciones es función del tamaño de la matriz.

Si los ordenadores pudieran almacenar y operar con todas las cifras de los números reales, es decir, si emplearan una aritmética exacta, con los métodos directos se obtendría la solución exacta del sistema en un número finito de pasos. Puesto que los ordenadores tienen una precisión finita, los errores de redondeo se propagan y la solución numérica obtenida siempre difiere de la solución exacta. La cota del error, para una matriz y término independiente dados, se asocia por lo general al número de operaciones de cada método. Se pretende, por lo tanto, obtener métodos con el mínimo número de operaciones posible.

- **Sistemas con solución inmediata**
  - ▷ Matriz diagonal, $\boldsymbol{A} = \boldsymbol{D}$
  - ▷ Matriz triangular superior, $\boldsymbol{A} = \boldsymbol{U}$
  - ▷ Matriz triangular inferior, $\boldsymbol{A} = \boldsymbol{L}$
- **Métodos de eliminación**
  - ▷ Método de Gauss
  - ▷ Método de Gauss-Jordan
- **Métodos de descomposición**
  - ▷ Método de Doolittle, $\boldsymbol{A} = \boldsymbol{LU}$
  - ▷ Método de Crout, $\boldsymbol{A} = \boldsymbol{LU}$
  - ▷ Método de Cholesky, $\boldsymbol{A} = \boldsymbol{LL}^T$
  - ▷ Descomposición generalizada, $\boldsymbol{A} = \boldsymbol{LDL}^T$
  - ▷ Método de Thomas ($\boldsymbol{A}$ tridiagonal)
- **Métodos de ortogonalización,** $\boldsymbol{A} = \boldsymbol{QR}$

Fig. 6.1 Clasificación de los métodos directos

Otra particularidad de los métodos directos es que siempre conducen, después de ciertas operaciones, a la resolución de uno o varios sistemas con solución inmediata. Es decir, sistemas donde la matriz es diagonal o triangular. Los métodos para sistemas de resolución inmediata son de hecho métodos directos. Además de éstos, los métodos directos se dividen en *métodos*

de eliminación y *métodos de descomposición*. En la figura 6.1 se presenta un esquema con la clasificación de los métodos directos más característicos.

6.2.2 Sistemas con solución inmediata

MATRIZ DIAGONAL

En este caso la matriz $\boldsymbol{A}$ se escribe como:

$$\boldsymbol{A} = \boldsymbol{D} = \begin{pmatrix} d_{11} & 0 & \cdots & \cdots & 0 \\ 0 & d_{22} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & d_{n-1,n-1} & 0 \\ 0 & \cdots & \cdots & 0 & d_{nn} \end{pmatrix} \tag{6.4}$$

y la solución se obtiene directamente

$$x_i = \frac{b_i}{d_{ii}} \qquad i = 1, \ldots, n \tag{6.5}$$

Existe solución si todos los términos de diagonal son no nulos. Además, si se desea evaluar el determinante de $\boldsymbol{A}$ sólo es necesario calcular el producto de todos los términos de la diagonal. Por último, el número de operaciones necesario es de n divisiones, es decir $T_D = n$ operaciones elementales.

MATRIZ TRIANGULAR SUPERIOR

$$\boldsymbol{A} = \boldsymbol{U} = \begin{pmatrix} u_{11} & u_{12} & \cdots & \cdots & u_{1n} \\ 0 & u_{22} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & u_{n-1,n-1} & u_{n-1,n} \\ 0 & \cdots & \cdots & 0 & u_{nn} \end{pmatrix} \tag{6.6}$$

En este caso la solución de la última ecuación es trivial $x_n = b_n \;/\; u_{nn}$. Una vez conocido x_n, la penúltima ecuación (la $n-1$) sólo tiene una incógnita que se deduce de forma sencilla. Conocidos ahora x_n y x_{n-1}, se pasa a la ecuación anterior (la $n-2$) y se resuelve para su única incógnita, x_{n-2}. Retrocediendo progresivamente se obtiene el algoritmo de *sustitución hacia atrás* que se escribe de la siguiente forma

$$\begin{aligned} x_n &= b_n \;/\; u_{nn} \\ x_i &= \left(b_i - \sum_{j=i+1}^{n} u_{ij} x_j \right) \;/\; u_{ii} \qquad i = n-1, n-2, \ldots, 1 \end{aligned} \tag{6.7}$$

De nuevo la solución existe si todos los términos de la diagonal de $\boldsymbol{U}$ son no nulos. El determinante se evalúa multiplicando los términos de la diagonal. El número de operaciones es ahora:

$$\begin{cases} 1+2+\cdots+(n-1)=\dfrac{n(n-1)}{2} & \text{sumas} \\ 1+2+\cdots+(n-1)=\dfrac{n(n-1)}{2} & \text{productos} \\ n & \text{divisiones} \end{cases}$$

por consiguiente $T_{\triangle}=n^2$ operaciones elementales.

MATRIZ TRIANGULAR INFERIOR

$$\boldsymbol{A}=\boldsymbol{L}=\begin{pmatrix} l_{11} & 0 & \cdots & \cdots & 0 \\ l_{21} & l_{22} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & l_{n-1,n-1} & 0 \\ l_{n1} & \cdots & \cdots & l_{n,n-1} & l_{nn} \end{pmatrix} \tag{6.8}$$

Se aplica un algoritmo similar al anterior que se denomina de *sustitución hacia adelante*:

$$\begin{aligned} x_1 &= b_1 \,/\, l_{11} \\ x_i &= \left(b_i - \sum_{j=1}^{i-1} l_{ij} x_j \right) \,/\, l_{ii} \qquad i=2,\ldots,n \end{aligned} \tag{6.9}$$

La existencia de solución, el determinante y el número de operaciones se evalúan exactamente como en el caso anterior y se llega a los mismos resultados.

6.2.3 Métodos de eliminación

MÉTODO DE GAUSS

En el método de eliminación de Gauss el problema original, $\boldsymbol{Ax}=\boldsymbol{b}$, se transforma mediante permutaciones adecuadas y combinaciones lineales de las ecuaciones en un sistema de la forma $\boldsymbol{Ux}=\boldsymbol{c}$ donde $\boldsymbol{U}$ es una matriz triangular superior. Este nuevo sistema equivalente al original es de resolución inmediata: sólo es necesario aplicar el algoritmo de sustitución hacia atrás presentado en el subapartado anterior.

Durante la transformación del sistema original al equivalente con matriz triangular, las operaciones (que *sólo* dependen de la matriz $\boldsymbol{A}$) se realizan sobre la matriz y *al mismo tiempo* sobre el término independiente. Esto constituye una de las grandes ventajas y a la vez inconvenientes de los métodos de eliminación. Si se dispone de una serie de términos independientes,

$\boldsymbol{b}_j$ $j = 1, \ldots, m$, conocidos de antemano, se efectúan sobre *todos* ellos, y al mismo tiempo, las operaciones necesarias para reducir el sistema y obtener una serie de $\boldsymbol{c}_j$ $j = 1, \ldots, m$. Por consiguiente, se almacenan y se manipulan todos los términos independientes a la vez. Posteriormente se resuelve un sistema con matriz triangular $\boldsymbol{U}$ para cada uno de los $\boldsymbol{c}_j$. Si, por el contrario, no se conocen todos los términos independientes al iniciar los cálculos, es necesario *recordar* todas las transformaciones necesarias para obtener $\boldsymbol{c}_1$ partiendo de $\boldsymbol{b}_1$; seguidamente se *repiten* todas estas operaciones sobre los demás términos independientes hasta obtener todos los $\boldsymbol{c}_j$ deseados. Hoy en día, en la mayoría de los problemas con matrices de tamaño pequeño o medio, esta propiedad de los métodos de eliminación es la determinante para su elección frente a los métodos de descomposición.

Otro punto importante que conviene valorar en el método de Gauss es su importante valor pedagógico. Muchos autores denominan de manera genérica *métodos gaussianos* al resto de los métodos de eliminación y de descomposición, puesto que la mayoría derivan del trabajo original de Gauss escrito en 1823 (que como otros métodos fundamentales del cálculo numérico fue desarrollado mucho antes de la aparición del primer ordenador). Su implementación en un ordenador sigue siendo la más simple, y con pocas modificaciones, como ya se verá, se obtiene el método más general que existe para la resolución de sistemas lineales de ecuaciones.

El algoritmo que se presenta a continuación parte de la ecuación 6.1 que se escribirá como:

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ a_{21}^{(0)} & a_{22}^{(0)} & a_{23}^{(0)} & \cdots & a_{2n}^{(0)} \\ a_{31}^{(0)} & a_{32}^{(0)} & a_{33}^{(0)} & \cdots & a_{3n}^{(0)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1}^{(0)} & a_{n2}^{(0)} & a_{n3}^{(0)} & \cdots & a_{nn}^{(0)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(0)} \\ b_3^{(0)} \\ \vdots \\ b_n^{(0)} \end{pmatrix} \tag{6.10}$$

donde el superíndice $^{(0)}$ indica coeficiente de la matriz o del término independiente originales. Si $a_{11}^{(0)} \neq 0$, se sustrae de todas la ecuaciones, a partir de la segunda fila, la primera ecuación multiplicada por $m_{i1} = \frac{a_{i1}^{(0)}}{a_{11}^{(0)}}$ con $i = 2, \ldots, n$. Esto induce el primer sistema equivalente derivado del original donde la primera columna tiene todos los coeficientes nulos exceptuando el primer coeficiente, y el resto de los coeficientes de la matriz y del término independiente se han visto modificados a partir de la segunda fila.

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \cdots & a_{nn}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(1)} \\ \vdots \\ b_n^{(1)} \end{pmatrix} \tag{6.11}$$

con

$$\begin{aligned} a_{ij}^{(1)} &= a_{ij}^{(0)} - m_{i1} a_{1j}^{(0)} = a_{ij}^{(0)} - \frac{a_{i1}^{(0)}}{a_{11}^{(0)}} a_{1j}^{(0)} \\ b_i^{(1)} &= b_i^{(0)} - m_{i1} b_1^{(0)} = b_i^{(0)} - \frac{a_{i1}^{(0)}}{a_{11}^{(0)}} b_1^{(0)} \end{aligned} \qquad \begin{cases} i = 2, \ldots, n \\ j = 2, \ldots, n \end{cases} \tag{6.12}$$

Ahora, si $a_{22}^{(1)}$ (que ya no coincide con el coeficiente que originalmente se encontraba en su posición, $a_{22}^{(0)}$) es distinto de cero, se sustrae de todas la ecuaciones siguientes la segunda ecuación multiplicada por $m_{i2} = \frac{a_{i2}^{(1)}}{a_{22}^{(1)}}$ con $i = 3, \ldots, n$. Después de realizadas estas operaciones sobre el sistema 6.11 se obtiene el segundo sistema equivalente al original

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(2)} \end{pmatrix} \tag{6.13}$$

donde la segunda columna a partir de la tercera fila sólo contiene términos nulos y los nuevos coeficientes de la matriz y término independiente se obtienen con las siguientes ecuaciones

$$\begin{aligned} a_{ij}^{(2)} &= a_{ij}^{(1)} - m_{i2} a_{2j}^{(1)} = a_{ij}^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} a_{2j}^{(1)} \\ b_i^{(2)} &= b_i^{(1)} - m_{i2} b_2^{(1)} = b_i^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} b_2^{(1)} \end{aligned} \qquad \begin{cases} i = 3, \ldots, n \\ j = 3, \ldots, n \end{cases} \tag{6.14}$$

Cada paso conduce a un nuevo sistema equivalente al original (ecuación 6.1) con la particularidad de que la k-ésima matriz es triangular superior si sólo se miran las primeras k ecuaciones y k incógnitas. En general, se escribe como

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots & a_{1k}^{(0)} & a_{1,k+1}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & \cdots & a_{2k}^{(1)} & a_{2,k+1}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \cdots & \vdots \\ \vdots & & \ddots & a_{kk}^{(k-1)} & a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\ 0 & \cdots & \cdots & 0 & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ \vdots & & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ \vdots \\ b_k^{(k-1)} \\ b_{k+1}^{(k)} \\ \vdots \\ b_n^{(k)} \end{pmatrix} \tag{6.15}$$

que se ha obtenido a partir de las siguientes ecuaciones

$$\begin{aligned} a_{ij}^{(k)} &= a_{ij}^{(k-1)} - m_{ik} a_{kj}^{(k-1)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)} \\ b_i^{(k)} &= b_i^{(k-1)} - m_{ik} b_k^{(k-1)} = b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)} \end{aligned} \qquad \begin{cases} i = k+1, \ldots, n \\ j = k+1, \ldots, n \end{cases} \tag{6.16}$$

Obsérvese que al pasar del $(k-1)$-ésimo al k-ésimo sistema es necesario realizar las siguientes operaciones

$$\begin{cases} (n-k)(n-k+1) & \text{sumas} \\ (n-k)(n-k+1) & \text{productos} \\ n-k & \text{divisiones} \end{cases}$$

Finalmente, al deducir el $(n-1)$-ésimo sistema se obtiene una matriz triangular superior:

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & a_{n-1,n}^{(n-2)} \\ 0 & 0 & \cdots & \cdots & a_{nn}^{(n-1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ \vdots \\ b_{n-1}^{(n-2)} \\ b_n^{(n-1)} \end{pmatrix} \tag{6.17}$$

A cada uno de los términos que aparecen en la diagonal de la matriz anterior se le denomina *pivote*. Conviene resaltar que los pivotes no coinciden con los términos originales de la diagonal de $\boldsymbol{A}$; es decir, $a_{kk}^{(k-1)} \neq a_{kk}^{(0)}$ para $k = 1, \ldots, n$.

Para resumir todos los pasos realizados hasta obtener el sistema 6.17, es necesario suponer que todos los pivotes son no nulos. Es decir, $a_{ii}^{(i-1)} \neq 0 \quad i = 1, \ldots, n$. A continuación se presenta el algoritmo que permite obtener la matriz y el término independiente del sistema 6.17,

$$\begin{aligned} a_{ij}^{(k)} &= a_{ij}^{(k-1)} - m_{ik} a_{kj}^{(k-1)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)} \\ b_i^{(k)} &= b_i^{(k-1)} - m_{ik} b_k^{(k-1)} = b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)} \end{aligned} \qquad \begin{cases} k = 1, \ldots, n-1 \\ i = k+1, \ldots, n \\ j = k+1, \ldots, n \end{cases} \tag{6.18}$$

donde los términos de superíndice (0) son iguales a los originales del sistema de ecuaciones. El sistema triangular obtenido en 6.17 es de resolución inmediata (véase el subapartado 6.2.2). El número de operaciones necesarias para realizar esta primera fase de eliminación ha sido de

$$\begin{cases} \displaystyle\sum_{k=1}^{n-1} (n-k)(n-k+1) = \frac{n(n^2-1)}{3} & \text{sumas} \\ \displaystyle\sum_{k=1}^{n-1} (n-k)(n-k+1) = \frac{n(n^2-1)}{3} & \text{productos} \\ \displaystyle\sum_{k=1}^{n-1} n-k = \frac{n(n-1)}{2} & \text{divisiones} \end{cases}$$

Si se tienen en cuenta las operaciones correspondientes a la segunda fase de sustitución hacia atrás, el número total de operaciones elementales necesarias para el método de Gauss es $T_G = \frac{4n^3+9n^2-7n}{6}$. La tabla 6.2 muestra el número de operaciones elementales para distintos tamaños

del sistema de ecuaciones. Obviamente, se ha obtenido una importante reducción al disponer ahora de un método que crece con n^3, en vez de $n!\,n^2$ (Cramer).

Tabla 6.2 Operaciones elementales del método de Gauss sin pivotamiento según el tamaño de la matriz (n)

n	T_G
5	115
10	805
100	681 550
1000	6.68×10^8

Como ya se ha comentado, se ha supuesto a lo largo de toda esta deducción que los pivotes eran distintos de cero. Si durante el proceso de eliminación se obtiene un pivote nulo, por ejemplo el $a_{kk}^{(k-1)}$, se debe buscar en la parte inferior de la columna k-ésima un coeficiente no nulo, es decir de entre los $a_{ik}^{(k-1)} \quad i = k+1, \ldots, n$ se toma uno que sea distinto de cero. Se sustituye entonces la fila k (y su término independiente) por la fila i (y su término independiente) que se haya escogido. Si dicho coeficiente no nulo no existiera, *la matriz sería singular*. Más adelante se verá una justificación teórica de este proceder.

Esta permutación de filas no sólo tiene interés cuando el pivote es exactamente cero. Es obvio que valores pequeños del pivote pueden producir grandes errores de redondeo, ya que siempre se divide por el valor del pivote. Por consiguiente, para reducir los errores de redondeo conviene escoger el pivote máximo en valor absoluto. Para ello, hay dos técnicas posibles:

1. En el k-ésimo sistema (véanse las ecuaciones 6.15 y 6.16) se toma como pivote el coeficiente mayor en valor absoluto de la columna k situado por debajo de la fila k inclusive. Para ello es necesario permutar las filas k y la correspondiente al pivote escogido en la matriz y su término independiente. Esta técnica se denomina *método de Gauss con pivotamiento parcial*.

2. En el k-ésimo sistema, se toma como pivote el coeficiente mayor en valor absoluto de la submatriz de orden $n-k$ definida por los coeficientes que quedan por debajo de la fila k y a la derecha de la columna k. Para ello, se permuta la fila k (y el término independiente asociado) y la columna k con las correspondientes al coeficiente que cumple la condición citada. Al final del proceso deben ponerse en el orden inicial las componentes del vector solución, puesto que su posición ha sido modificada al realizar las permutaciones de columnas. Esta técnica es el *método de Gauss con pivotamiento total*.

Estas dos últimas estrategias producen métodos numéricamente estables. El método de Gauss sin pivotamiento no es necesariamente estable. El estudio detallado de la estabilidad y propagación de errores de redondeo del método de Gauss no es trivial (véase Wilkinson (1965)).

Desde el punto de vista de la implementación práctica de los métodos de Gauss con pivotamiento, conviene señalar que las permutaciones no se realizan físicamente en el ordenador, sino que se emplean vectores de redireccionamiento de memoria similares a los empleados en los esquemas de almacenamiento específicos para matrices con estructuras simples.

Observación: Si la matriz $\boldsymbol{A}$ es simétrica, las matrices llenas de orden $n-k$ sobre las que se aplica sucesivamente el algoritmo sólo permanecen simétricas si no se realiza ninguna permutación de filas o columnas (véase el problema 6.1). La misma observación es válida si la matriz $\boldsymbol{A}$ tiene una estructura que permite el almacenamiento en banda o en perfil (apartado 7.4). □

Problema 6.1:
Sea $\boldsymbol{A}$ una matriz regular *simétrica*. Se desea resolver el sistema lineal $\boldsymbol{Ax} = \boldsymbol{b}$ mediante el método de Gauss *sin pivotamiento*.

a) Adaptar el algoritmo de Gauss al caso de matrices simétricas, aprovechando la simetría para eliminar las operaciones innecesarias. Sugerencia: nótese que en cada paso del proceso de eliminación, cuando se anulan los términos de la columna k-ésima por debajo del pivote, $a_{kk}^{(k-1)}$, sólo se modifica la submatriz llena de orden $n-k$:

$$\begin{pmatrix} a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ \vdots & \ddots & \vdots \\ a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix}$$

Emplear la propiedad de que en $k=0$ la submatriz correspondiente (que es la matriz original $\boldsymbol{A}$) es simétrica.

b) Calcular el número de operaciones necesarias, y compararlo con el caso general (matrices no simétricas).

c) ¿Puede emplearse el algoritmo desarrollado en el apartado *a* si es necesario pivotar? ¿Por qué? ●

Método de Gauss-Jordan

A continuación se presenta una variante del método de Gauss que conviene considerar. En este método, además de sustraer la fila k multiplicada por $m_{ik} = a_{ik}^{(k-1)}/a_{kk}^{(k-1)}$ a las filas posteriores, también se sustrae a las anteriores. Es práctica común, en este caso, dividir la fila k por su pivote para que el término de la diagonal quede unitario. De esta forma, el k-ésimo sistema así obtenido se escribe como:

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & a_{1,k+1}^{(k)} & \cdots & a_{1n}^{(k)} \\ 0 & 1 & 0 & \cdots & 0 & a_{2,k+1}^{(k)} & \cdots & a_{2n}^{(k)} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & & \vdots \\ \vdots & & \ddots & \ddots & 0 & a_{k-1,k+1}^{(k)} & \cdots & a_{k-1,n}^{(k)} \\ \vdots & & & \ddots & 1 & a_{k,k+1}^{(k)} & \cdots & a_{kn}^{(k)} \\ 0 & \cdots & \cdots & \cdots & 0 & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ \vdots & & & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{k-1} \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(k)} \\ b_2^{(k)} \\ \vdots \\ b_{k-1}^{(k)} \\ b_k^{(k)} \\ b_{k+1}^{(k)} \\ \vdots \\ b_n^{(k)} \end{pmatrix} \tag{6.19}$$

Como se puede observar, al anular todos los coeficientes de la columa k, excepto el diagonal, se va transformando la matriz original en la identidad. Al final, la $(n-1)$-ésima matriz obtenida por operaciones simples de fila es la identidad, y por lo tanto, el $(n-1)$-ésimo término independiente, $(b_1^{(n-1)}, \ldots, b_n^{(n-1)})^T$ es la solución del sistema de ecuaciones original.

El algoritmo necesario para la transformación de la matriz es el siguiente

$$\begin{aligned} a_{kj}^{(k)} &= \frac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}} \\ a_{ij}^{(k)} &= a_{ij}^{(k-1)} - a_{ik}^{(k-1)} a_{kj}^{(k-1)} \\ b_k^{(k)} &= \frac{b_k^{(k-1)}}{a_{kk}^{(k-1)}} \\ b_i^{(k)} &= b_i^{(k-1)} - a_{ik}^{(k-1)} b_k^{(k-1)} \end{aligned} \qquad \begin{cases} k = 1, \ldots, n-1 \\ i = 1, \ldots, k-1, k+1, \ldots, n \\ j = k+1, \ldots, n \end{cases} \tag{6.20}$$

El número de operaciones que se deben realizar es

$$\begin{cases} \displaystyle\sum_{k=1}^{n-1} (n-1)(n-k+1) = \frac{(n-1)^2(n-2)}{2} & \text{sumas} \\ \displaystyle\sum_{k=1}^{n-1} (n-1)(n-k+1) = \frac{(n-1)^2(n-2)}{2} & \text{productos} \\ \displaystyle\sum_{k=1}^{n-1} (n-k+1) = \frac{(n-1)(n-2)}{2} & \text{divisiones} \end{cases}$$

Por consiguiente el número total de operaciones elementales del método de Gauss-Jordan, tal como se ha presentado aquí, es de $T_{GJ} = n^3 + \frac{1}{2}n^2 - \frac{5}{2}n + 1$