

Capítulo 1

Sistemas de ecuaciones lineales

1.1. Eliminación de Gauss

Un *sistema de ecuaciones lineales* es una ecuación del tipo $A\vec{x} = \vec{b}$ donde A y \vec{b} son una matriz y un vector fijos y $\vec{x} = (x_1, \dots, x_n)^t$ es un vector de incógnitas. De una manera más explícita, tiene la forma:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1, \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2, \\ \dots &= \dots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m. \end{aligned}$$

Se dice que A es la *matriz del sistema* o la *matriz de coeficientes* y la matriz obtenida al añadir \vec{b} tras la última columna se dice que es la *matriz ampliada*. A veces se denota con A^+ (esto se parece tipográficamente a A^t y A^\dagger pero es difícil que haya confusión) y con frecuencia se separa \vec{b} mediante una barra vertical $A^+ = (A|\vec{b})$. Suponemos sin decirlo cada vez que $a_{ij}, b_i \in \mathbb{C}$ y en casi todos los ejemplos serán de hecho reales.

Resolver un sistema de ecuaciones lineales manualmente es algo que empieza a dar bastante pereza para $n > 2$ sin embargo hay casos especiales sencillos. Por ejemplo, si A es una matriz diagonal, la solución es directamente $x_i = b_i/a_{ii}$ suponiendo que todos los a_{ii} son no nulos. Si $a_{ii} = 0$ para algún i y $b_i \neq 0$, claramente no habrá solución y si $a_{ii} = b_i = 0$ podríamos elegir ese x_i arbitrariamente. Ya este caso diagonal tan sencillo muestra que los sistemas de ecuaciones lineales pueden tener o no solución y en caso de que la tengan, no es necesariamente única. Más adelante daremos nombres, que seguro que ya conoces, a estas situaciones.

Nuestro objetivo en esta sección es diseñar un algoritmo que transforme cualquier sistema de ecuaciones lineales en uno muy sencillo. La aspiración de reducir al caso diagonal es solo realista si A es una matriz cuadrada. Esto nos lleva a considerar sistemas sencillos que sean más generales. Se dice que A es una *matriz escalonada* si sus filas nulas, si las hubiera, están al final y cada fila no nula tiene siempre más ceros a la izquierda que la que está encima. Por ejemplo, una matriz diagonal es

escalonada. Algunos ejemplos no diagonales son

$$\begin{pmatrix} 2 & 1 & 0 & 7 \\ 0 & 0 & 1 & 3 \end{pmatrix}, \quad \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{y} \quad \begin{pmatrix} 7 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Para clarificar más la situación e ilustrar el nombre de “escalonada”, aparte de la matriz nula, estas son todas las plantillas posibles para matrices escalonadas 2×3

$$\begin{pmatrix} p_1 & * & * \\ 0 & p_2 & * \end{pmatrix}, \begin{pmatrix} p_1 & * & * \\ 0 & 0 & p_2 \end{pmatrix}, \begin{pmatrix} p_1 & * & * \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & p_1 & * \\ 0 & 0 & p_2 \end{pmatrix}, \begin{pmatrix} 0 & p_1 & * \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & p_1 \\ 0 & 0 & 0 \end{pmatrix}$$

donde $*$ indican números cualesquiera y los p_i son no nulos. Estos números que marcan el comienzo de un “escalón” se llaman *pivotes*. Esto es, los pivotes son los primeros valores no nulos de cada fila.

Un sistema de ecuaciones lineales con matriz es escalonada es sencillo de resolver (dando por hecho que esto sea posible) porque la solución de la ecuación correspondiente al último pivote se puede sustituir en las anteriores para obtener un sistema con una incógnita menos y repetir el proceso tantas veces como pivotes haya. Por ejemplo, consideremos el siguiente sistema con matriz escalonada

$$(1.1) \quad \begin{array}{rcl} 2x_1 - x_2 - 2x_3 & = & 6 \\ x_2 - x_3 & = & -1 \\ 3x_3 & = & -3 \end{array} \quad A = \begin{pmatrix} 2 & -1 & -2 \\ 0 & 1 & -1 \\ 0 & 0 & 3 \end{pmatrix}.$$

La última ecuación da $x_3 = -1$ y al sustituir en las otras se obtiene

$$\begin{array}{rcl} 2x_1 - x_2 & = & 4 \\ x_2 & = & -2 \end{array} \quad \text{con matriz} \quad \begin{pmatrix} 2 & -1 \\ 0 & 1 \end{pmatrix}.$$

Sustituyendo $x_2 = -2$ se sigue $x_1 = 1$ con lo que la solución es $x_1 = 1$, $x_2 = -2$, $x_3 = -1$.

Si en el sistema original $a_{33} = 0$ entonces no habría solución, ya que la última ecuación sería $0 = -3$. Si ponemos también $b_3 = 0$ para no que exista este problema, se tendría el sistema

$$\begin{array}{rcl} 2x_1 - x_2 - 2x_3 & = & 6 \\ x_2 - x_3 & = & -1 \end{array} \quad \text{con matriz} \quad \begin{pmatrix} 2 & -1 & -2 \\ 0 & 1 & -1 \end{pmatrix}.$$

La última ecuación dice $x_2 = x_3 - 1$ y aquí x_3 es arbitrario, lo cual viene heredado de que la fila nula que no hemos escrito daría la trivialidad $0 \cdot x_3 = 0$ que se cumple para toda x_3 . Escribamos $x_3 = \lambda$. Al sustituir en la primera ecuación, se sigue $x_1 = (3\lambda + 5)/2$ así que todas las soluciones del sistema vienen descritas por

$$x_1 = \frac{3\lambda + 5}{2}, \quad x_2 = \lambda - 1, \quad x_3 = \lambda \quad \text{con } \lambda \text{ arbitrario.}$$

La *eliminación de Gauss* o *reducción de Gauss*¹ es un sencillo algoritmo para transformar un sistema de ecuaciones lineales en otro equivalente con matriz de coeficientes escalonada, y por tanto fácil de resolver. Consiste en ir generando ordenadamente en las columnas los ceros necesarios empleando las siguientes *transformaciones elementales* en la matriz ampliada:

1. Sumar a una fila un múltiplo de otra.
2. Multiplicar una fila por un número distinto de cero.
3. Intercambiar dos filas.

En realidad la segunda no es estrictamente necesaria. Para un sistema típico basta con la primera y algunos especiales requieren también la tercera.

Está claro que una solución del sistema lo seguirá siendo después de aplicar cualquiera de estas transformaciones, por tanto no perdemos soluciones. También está claro que se pueden revertir, por tanto tampoco introducimos soluciones extrañas.

Veamos cómo funciona sobre el sistema

$$\begin{array}{rcl} 2x_1 - x_2 - 2x_3 & = & 6 \\ 2x_1 + x_2 - 4x_3 & = & 4 \\ -2x_1 + 3x_2 + 3x_3 & = & -11 \\ 10x_1 - 2x_2 - 10x_3 & = & 24 \end{array} \quad \text{con} \quad A^+ = \left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 2 & 1 & -4 & 4 \\ -2 & 3 & 3 & -11 \\ 10 & -2 & -10 & 24 \end{array} \right).$$

Cuando en un sistema hay más condiciones (ecuaciones) que incógnitas que hallar, lo natural es que no tenga solución pero no siempre es así, como veremos en este ejemplo. Conservando la primera fila f_1 , queremos crear ceros bajo $a_{11} = 2$, el primer pivote. Para ello aplicamos la primera transformación elemental en la forma $f_2 \mapsto f_2 - f_1$, $f_3 \mapsto f_3 + f_1$, $f_4 \mapsto f_4 - 5f_1$. Así pues

$$\left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 2 & 1 & -4 & 4 \\ -2 & 3 & 3 & -11 \\ 10 & -2 & -10 & 24 \end{array} \right) \longrightarrow \left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 0 & 2 & -2 & -2 \\ 0 & 2 & 1 & -5 \\ 0 & 3 & 0 & -6 \end{array} \right).$$

Aunque no sea necesario, para tratar con números enteros pequeños, apelemos a la segunda transformación elemental efectuando $f_2 \mapsto f_2/2$ y $f_4 \mapsto f_4/3$. Tras ello, para crear ceros bajo el segundo pivote $a_{22} = 1$ aplicamos $f_3 \mapsto f_3 - 2f_2$, $f_4 \mapsto f_4 - f_2$:

$$\left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 0 & 2 & -2 & -2 \\ 0 & 2 & 1 & -5 \\ 0 & 3 & 0 & -6 \end{array} \right) \longrightarrow \left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 0 & 1 & -1 & -1 \\ 0 & 2 & 1 & -5 \\ 0 & 1 & 0 & -2 \end{array} \right) \longrightarrow \left(\begin{array}{ccc|c} 2 & -1 & -2 & 6 \\ 0 & 1 & -1 & -2 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 1 & -1 \end{array} \right).$$

¹El nombre proviene de C.F. Gauss quien lo sistematizó para cierta aplicación aunque no con la forma actual de matrices. Dada la relevancia de este matemático, para algunos el mejor de todos los tiempos y conocido en su época como *princeps mathematicorum* (el rey de los matemáticos), es un poco injusto que sea tan recordado por una contribución así de simple y en la que su protagonismo es discutible [4], [3].

Por último, si $f_4 \mapsto f_4 - f_3/3$, la última fila se anula y llegamos a una forma escalonada como la de (1.1), así que la solución es como allí $x_1 = 1$, $x_2 = -2$, $x_3 = -1$.

Los números complejos solo añaden que los cálculos se nos puedan hacer más cuesta arriba dependiendo de nuestra experiencia. Veamos un ejemplo con dos incógnitas.

$$\begin{aligned} (2-i)x_1 - x_2 &= 2 \\ ix_1 + (1+i)x_2 &= 1 \end{aligned} \quad \text{con} \quad A^+ = \left(\begin{array}{cc|c} 2-i & -1 & 2 \\ i & 1+i & 1 \end{array} \right).$$

En principio lo natural es proceder con $f_2 \mapsto f_2 - if_1/(2-i)$ pero dividir por $2-i$ no hace mucha gracia mientras que todos deberíamos saber la tabla de dividir por i que se reduce a $1/i = -i$. Así que para hacer los cálculos a mano es más conveniente intercambiar las dos filas apelando a la tercera transformación elemental. Con ello, una posibilidad para la reducción de Gauss es

$$A^+ \xrightarrow{f_1 \leftrightarrow f_2} \left(\begin{array}{cc|c} i & 1+i & 1 \\ 2-i & -1 & 2 \end{array} \right) \xrightarrow{f_2 \mapsto f_2 + (1+2i)f_1} \left(\begin{array}{cc|c} i & 1+i & 1 \\ 0 & -2+3i & 3+2i \end{array} \right).$$

El $1+2i$ proviene de $-(2-i)/i = i(2-i)$. De la segunda fila se deduce a simple vista $x_2 = -i$ y si tu simple vista no funciona aquí, haz el cálculo que necesites. Sustituyendo se sigue $x_1 = 1$.

La tercera transformación elemental, aparte de para simplificar cálculos, se usa cuando en el lugar en que debiera haber un pivote hay un cero. Por ejemplo, consideremos

$$\begin{aligned} 2x_2 + x_3 &= 3 \\ x_1 + x_2 - 3x_3 &= -1 \\ x_1 + 3x_2 - 2x_3 &= 2 \end{aligned} \quad \text{con} \quad A^+ = \left(\begin{array}{ccc|c} 0 & 2 & 1 & 3 \\ 1 & 1 & -3 & -1 \\ 1 & 3 & -2 & 2 \end{array} \right).$$

No podemos tomar a_{11} como pivote porque es nulo. Para remediarlo podemos intercambiar las dos primeras filas y después aplicar, como antes, la eliminación de Gauss con la primera transformación elemental:

$$\left(\begin{array}{ccc|c} 1 & 1 & -3 & -1 \\ 0 & 2 & 1 & 3 \\ 1 & 3 & -2 & 2 \end{array} \right) \xrightarrow{f_3 \mapsto f_3 - f_1} \left(\begin{array}{ccc|c} 1 & 1 & -3 & -1 \\ 0 & 2 & 1 & 3 \\ 0 & 2 & 1 & 3 \end{array} \right) \xrightarrow{f_3 \mapsto f_3 - f_2} \left(\begin{array}{ccc|c} 1 & 1 & -3 & -1 \\ 0 & 2 & 1 & 3 \\ 0 & 0 & 0 & 0 \end{array} \right).$$

Con ello se tiene que x_3 es arbitrario, $x_3 = \lambda$ y $x_2 = (3-\lambda)/2$. Sustituyendo en la primera ecuación se deduce que la solución general es

$$x_1 = \frac{7\lambda - 5}{2}, \quad x_2 = \frac{3 - \lambda}{2}, \quad x_3 = \lambda.$$

Dando a λ valores enteros impares obtenemos soluciones enteras sencillas. Por ejemplo, $\lambda = 1, 3, 5$ conducen a las soluciones $(1, 1, 1)^t$, $(8, 0, 3)^t$, $(15, -1, 5)^t$. Asignar algún valor sencillo a los parámetros y comprobar que el resultado satisface la ecuación, apenas requiere esfuerzo y es una buena garantía de que no nos hemos equivocado en los cálculos de la eliminación de Gauss.

Un apunte final es que quizá uno tenga la idea de que siempre que haya infinitas soluciones las últimas variables siempre se pueden elegir como parámetros. Esto no es así en general. Pensemos en casos en que las formas escalonadas de la matriz de coeficientes sean

$$\begin{pmatrix} p_1 & * & * & * \\ 0 & 0 & p_2 & * \\ 0 & 0 & 0 & p_3 \end{pmatrix} \quad \text{o} \quad \begin{pmatrix} 0 & p_1 & * & * \\ 0 & 0 & p_2 & * \\ 0 & 0 & 0 & p_3 \end{pmatrix}.$$

En la primera situación x_4 queda determinada con la tercera fila y x_3 sustituyendo con la segunda, sin embargo la primera ecuación solo establece una relación entre x_1 y x_2 así que una de estas variables es un parámetro arbitrario. En la segunda situación, ninguna de las ecuaciones involucra x_1 , así pues podemos asignarle cualquier valor, mientras que x_2 , x_3 y x_4 están determinadas.

La idea para tener en mente es que las incógnitas que corresponden a columnas que no contienen pivotes son las que elegimos como parámetros arbitrarios.

Exprimiendo el silicio [opcional]. A pesar de que tanto `matlab/octave` como `sagemath` tienen comandos que transforman una matriz a forma escalonada, no son muy útiles para que comprobemos cuantitativamente nuestros cálculos porque no existe “la” forma escalonada, sino “una” forma escalonada. En una sección posterior introduciremos una forma escalonada especial que es la que dan `matlab/octave` y `sagemath`. Ahora lo único que vamos a ver es cómo llevar a cabo las transformaciones elementales. Si tienes habilidad programando con ello podrías escribir código que a partir de una matriz te diera todos los pasos intermedios en una aplicación de la eliminación de Gauss. Para ser realista, seguramente tardarás más en escribir el código que te resuelva automáticamente los problemas propuestos de eliminación de Gauss que en hacerlos directamente.

En `matlab/octave` se indica con dos puntos todos los posibles valores de índices (como en `python`). Así para una matriz `A` con `A(1,:)` tenemos la primera fila, los elementos a_{1j} , y con `A(:,1)` la primera columna.

Sin usar ningún atajo, podríamos hacer las transformaciones indicadas en el último ejemplo y finalmente multiplicar la segunda fila con 2 (para practicar con la segunda transformación elemental) con el siguiente código:

```

1  % Matriz ampliada de un ejemplo anterior
2  A = [0,2,1,3; 1,1,-3,-1; 1,3,-2,2];
3
4  % Intercambia las dos primeras filas (tr. el. 3)
5  f1 = A(1,:); % guarda la 1ª fila;
6  A(1,:) = A(2,:); % copia la 2ª fila en la 1ª
7  A(2,:) = f1 %copia la fila guardada en la 2ª
8
9  % f_3 -> f_3-f_1 (tr. el. 1)
10 A(3,:) = A(3,:) - A(1,:)
11 % f_3 -> f_3-f_2 (tr. el. 1)
12 A(3,:) = A(3,:) - A(2,:)
13
14 % Multiplica la segunda fila por 2 (tr. el. 2)
15 A(2,:) = 2*A(2,:);

```

Los punto y coma tras las operaciones están situados de manera que se vean las tres matrices del ejemplo y la matriz final. Así la ausencia de punto y coma en la línea 7 muestra toda la matriz a pesar de que solo estamos asignado la segunda fila.

Como contrapunto, veamos cómo proceder con `sagemath` usando comandos compactos. Este *software* usa las convenciones informáticas en vez de las de la gente de la calle: las filas y columnas empiezan con el índice cero y la matriz es una lista de listas (las filas). Así `A[0][:]` es la primera fila de `A` y `A[1][2]` indica el elemento a_{23} .

```

1 # Matriz ampliada de un ejemplo anterior
2 A = matrix(3,4, [0,2,1,3, 1,1,-3,-1, 1,3,-2,2])
3
4 # Intercambia las dos primeras filas (tr. el. 3)
5 A.swap_rows(0,1)
6 print(A)
7
8 # f_3 -> f_3-f_1 (tr. el. 1)
9 A.add_multiple_of_row (2,0,-1)
10 print(A)
11 # f_3 -> f_3-f_2 (tr. el. 1)
12 A.add_multiple_of_row (2,1,-1)
13 print(A)
14
15 # Multiplica la segunda fila por 2 (tr. el. 2)
16 A.rescale_row(1, 2)
17 print(A)

```

Los comandos `print(A)` se han situado de forma que, de nuevo, se vean las matrices mostradas en el ejemplo y el resultado final.