

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

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

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 [5], [4].

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.

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

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

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

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 `a1j`, 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.

1.2. Solución general

Una vez que conocemos las cuentas que hay que hacer para resolver un sistema de ecuaciones lineales, vamos a estudiar la estructura del conjunto de soluciones. No hay grandes ideas, solo ordenar las conclusiones que sacaríamos tras practicar con unos cuantos ejemplos de eliminación de Gauss. ¿Lo has hecho ya?

En el estudio de esta estructura desempeña un papel importante el número de pivotes (escalones) al aplicar eliminación de Gauss a una matriz A . Llamaremos a este número *rango* de A , y escribiremos $\text{rg}(A)$. Aparte de la queja de “así no es como me lo explicaron a mí”, que más adelante solventaremos mostrando la equivalencia, alguien muy perspicaz podría protestar diciendo que no hay una única forma de aplicar la eliminación de Gauss y por tanto no está claro que a mí y a mi compañero nos salga el mismo rango sin habernos equivocado ninguno. A sabiendas de que esto es solo alimento para filomatemáticos, haremos de ello el primer resultado teórico del curso.

Lema 1.2.1. *Al reducir una matriz a su forma escalonada, los pivotes aparecen siempre en las mismas posiciones independientemente de como se lleve a cabo la eliminación de Gauss. En particular el rango está bien definido.*

Demostración. Sean E y \tilde{E} dos matrices escalonadas obtenidas por eliminación de Gauss sobre $A \in \mathcal{M}_{m \times n}$. Si la situación de algún pivote no coincidiera, digamos que $1 \leq N \leq n$ es tal que los $N - 1$ primeros pivotes tienen la misma localización pero E tiene el N -ésimo pivote en la columna J y \tilde{E} en la columna $J' > J$. Consideremos

el sistema $N \times J$, $(a_{ij})_{i,j=1}^{N,J} \vec{x} = \vec{0}$. Al resolverlo con la forma escalonada $(e_{ij})_{i,j=1}^{N,J}$ se sigue $x_J = 0$, ya que el último elemento e_{NJ} es un pivote, mientras que con $(\tilde{e}_{ij})_{i,j=1}^{N,J}$ la incógnita x_J resulta arbitraria. Esto contradice que en la eliminación de Gauss no se pierden ni ganan soluciones. \square

Destacaremos los *sistemas homogéneos* $A\vec{x} = \vec{0}$, que ya han aparecido en la demostración anterior. Lo que tienen de peculiar es que siempre admiten al menos la solución trivial $\vec{x} = \vec{0}$ y que la última columna de A^+ siempre va a conservar los ceros a lo largo del proceso de eliminación de Gauss, por ello a menudo no se escribe.

La estructura de las soluciones de un sistema homogéneo se describe en el siguiente resultado fundamental en la teoría del curso:

Teorema 1.2.2. *Consideremos el sistema homogéneo $A\vec{x} = \vec{0}$ con $A \in \mathcal{M}_{m \times n}$ y sea $r = \text{rg}(A)$. Si $r = n$ la única solución es $\vec{x} = \vec{0}$ y si $r < n$ todas las soluciones son*

$$\vec{x} = \lambda_1 \vec{v}_1 + \lambda_2 \vec{v}_2 + \cdots + \lambda_{n-r} \vec{v}_{n-r}$$

con λ_j arbitrarios y \vec{v}_j ciertos vectores. Además elecciones distintas de los λ_j dan lugar a soluciones distintas.

La estructura de las soluciones de un sistema general se relaciona con el caso homogéneo.

Teorema 1.2.3. *Un sistema de ecuaciones lineales $A\vec{x} = \vec{b}$ tiene solución si y solo si $\text{rg}(A) = \text{rg}(A^+)$. En ese caso todas las soluciones son $\vec{x} = \vec{x}_0 + \vec{y}$ con \vec{y} la solución general del sistema homogéneo $A\vec{y} = \vec{0}$.*

La siguiente consecuencia se llama en el mundo hispano *teorema de Rouché-Frobenius* aunque el matemático F.G. Frobenius no tenga tanto que ver con este resultado². Por alguna razón, ha sido tradicionalmente material ineludible en los exámenes de matemáticas de acceso a la universidad española.

Corolario 1.2.4. *Dado un sistema de ecuaciones lineales $A\vec{x} = \vec{b}$ con $A \in \mathcal{M}_{m \times n}$ se cumple:*

- 1) *Tiene solución única, y se dice que es compatible determinado, si y solo si $n = \text{rg}(A) = \text{rg}(A^+)$.*
- 2) *Tiene infinitas soluciones, y se dice que es compatible indeterminado, si y solo si $n > \text{rg}(A) = \text{rg}(A^+)$.*
- 3) *No tiene solución, y se dice que es incompatible, si y solo si $\text{rg}(A) \neq \text{rg}(A^+)$.*

Antes de ir a las demostraciones de los teoremas, veamos unos ejemplos.

²Parece que el nombre fue introducido por el matemático hispanoargentino J. Rey Pastor. En el mundo anglosajón se llama *teorema de Rouché-Capelli*, lo que es mucho más lógico porque tanto E. Rouché como A. Capelli lo enunciaron y demostraron, aunque no fueran los primeros en hacerlo.

En primer lugar, hallemos vectores \vec{v}_j que dan la solución en el Teorema 1.2.2 para el sistema homogéneo $A\vec{x} = \vec{0}$ con

$$(1.2) \quad A = \begin{pmatrix} 1 & 1 & -1 & 2 & -3 \\ -3 & -3 & 4 & -6 & 8 \\ 2 & 2 & 1 & 4 & -7 \end{pmatrix}.$$

La forma escalonada se consigue con los siguientes pasos:

$$A \xrightarrow[\substack{f_2 \mapsto f_2 + 3f_1 \\ f_3 \mapsto f_3 - 2f_1}]{\begin{pmatrix} 1 & 1 & -1 & 2 & -3 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 3 & 0 & -1 \end{pmatrix}} \xrightarrow{f_3 \mapsto f_3 - 3f_2} \begin{pmatrix} 1 & 1 & -1 & 2 & -3 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 2 \end{pmatrix}.$$

Las columnas que no contienen pivotes son la segunda y la cuarta y corresponden a parámetros libres, así $x_2 = \lambda_1$, $x_4 = \lambda_2$, mientras que despejando x_1 , x_3 y x_5 utilizando las ecuaciones de abajo a arriba, $x_5 = 0$, $x_3 = 0$, $x_1 = -\lambda_1 - 2\lambda_2$. Con ello

$$\vec{x} = \begin{pmatrix} -\lambda_1 - 2\lambda_2 \\ \lambda_1 \\ 0 \\ \lambda_2 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 & -2 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \lambda_1 \begin{pmatrix} -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} -2 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}.$$

El paso intermedio es solo para practicar con el producto de matrices, está claro que las coordenadas de \vec{v}_1 y \vec{v}_2 vienen dadas por los coeficientes de λ_1 y λ_2 en la expresión inicial para \vec{x} .

Supongamos ahora que deseamos hallar todas las soluciones de $A\vec{x} = \vec{b}$ con A como antes y $\vec{b} = (-1, 2, -3)^t$. Para reducir $A^+ = (A|\vec{b})$ a una matriz escalonada solo tenemos que aplicar los pasos anteriores a la última columna, el resto será igual. Con esas breves cuentas se obtiene:

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

Si no tuviéramos el ejemplo anterior, despejaríamos aquí para hallar la solución general pero podemos aprovechar un poco lo hecho tomando las incógnitas libres x_2 y x_4 como nulas llegando al sistema escalonado más simple $2x_5 = 2$, $x_3 - x_5 = -1$, $x_1 - x_3 - 3x_5 = 1$ que conduce a la solución $\vec{x}_0 = (2, 0, 0, 0, 1)^t$ y según el Teorema 1.2.3 todas las soluciones son $\vec{x} = \vec{x}_0 + \lambda_1 \vec{v}_1 + \lambda_2 \vec{v}_2$ con \vec{v}_1 y \vec{v}_2 como antes.

Como último ejemplo resolveremos con eliminación de Gauss el primer problema del modelo de la opción A de la EVAU 2011, ¡seguramente nadie lo hizo así! El enunciado pide discutir el siguiente sistema en función del parámetro λ y hallar la solución para $\lambda = 1$.

$$\begin{array}{rcl} \lambda x & + \lambda z & = 2, \\ x & + \lambda y & - z = 1, \\ x & + 3y & + z = 2\lambda. \end{array}$$

Para estar seguros de tener un pivote en a_{11} , ordenamos las ecuaciones en orden contrario al presentado ($f_1 \leftrightarrow f_3$). En esta situación conseguimos los ceros de la primera columna con $f_2 \mapsto f_2 - f_1$ y $f_3 \mapsto f_3 - \lambda f_1$

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

¿Hemos acabado ya? Solo tenemos una matriz escalonada si $\lambda = 0$ y en ese caso el sistema resulta incompatible por el Corolario 1.2.4 (A con dos pivotes y A^+ con tres). Si $\lambda \neq 0$ podríamos intercambiar f_2 y f_3 (con ello tenemos un pivote bajo el 3) y después aplicar $f_3 \mapsto f_3 + (\lambda - 3)f_2/(3\lambda)$. Da igual lo que salgan exactamente las cuentas, está claro que hay tres pivotes y por tanto solución única: el sistema es compatible determinado para $\lambda \neq 0$. Para la segunda parte del problema sustituyendo $\lambda = 1$ en la última matriz y aplicando $f_3 \mapsto f_3 - 3f_2/2$ tenemos

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

que da $z = 1/2$, $y = 0$, $x = 3/2$.

Ahora vamos con las demostraciones de los teoremas. Aunque la primera parezca un poco farragosa, solo pone por escrito conclusiones naturales al aplicar eliminación de Gauss.

Demostración del Teorema 1.2.2. Si $r = n$ entonces todas las columnas contienen pivotes, es decir, ocupan toda la diagonal de la forma escalonada E . La última ecuación es $e_{nn}x_n = 0$ que implica $x_n = 0$, la anterior es $e_{n-1,n-1}x_{n-1} + e_{n-1,n}x_n = 0$ que implica $x_{n-1} = 0$ y así sucesivamente.

Supongamos ahora $r < n$. Según hemos visto, en la eliminación de Gauss despejamos x_j si en la columna j de la matriz escalonada hay un pivote, el resto de las incógnitas son libres. Digamos que los pivotes están en las columnas numeradas con $\{p_1, \dots, p_r\}$ y el resto de números de columnas son $\{q_1, \dots, q_{n-r}\}$, entonces tomamos $x_{q_l} = \lambda_l$ con λ_l arbitrario $1 \leq l \leq n - r$ y al despejar las incógnitas correspondientes a las columnas pivote se tendrá algo del tipo $x_{p_k} = c_{k1}\lambda_1 + \dots + c_{kn-r}\lambda_{n-r}$ con $1 \leq k \leq r$. En forma matricial, $\vec{x} = F\vec{\lambda}$ donde $f_{p_k j} = c_{kj}$ mientras que $f_{q_l l} = 1$ y $f_{q_l j} = 0$ si $j \neq l$. Si $\vec{v}_1, \dots, \vec{v}_{n-r}$ son las columnas de F esto es lo mismo que $\vec{x} = \lambda_1 \vec{v}_1 + \dots + \lambda_{n-r} \vec{v}_{n-r}$ por la regla para multiplicar matrices. Finalmente, si dos vectores de parámetros $\vec{\lambda} = \vec{\mu}$ y $\vec{\lambda} = \vec{\nu}$ dieran la misma solución \vec{x} entonces como todos los vectores tienen la q_l -ésima coordenada nula excepto \vec{v}_l que la tiene igual a 1, se deduce $\mu_l = \nu_l$ y esto es válido para cualquier $1 \leq l \leq n - r$. \square

Demostración del Teorema 1.2.3. Para la primera parte pensemos en que hallamos una forma escalonada de A^+ , que es una de A extendida con una columna. Si la forma escalonada de A^+ tiene un pivote p más que la de A , la ecuación correspondiente a

la última fila no nula será $0 = p$ lo que indica que no hay solución. Si tuvieran los mismos pivotes, esta última ecuación sería de la forma $e_{ij}x_j + \dots + e_{in}x_n = c$ con $e_{ij} \neq 0$ el pivote y se puede llevar a cabo la sustitución regresiva de la eliminación de Gauss.

Para la segunda parte basta observar que $A\vec{x} = \vec{b}$ equivale a $A\vec{y} = \vec{0}$ con $\vec{y} = \vec{x} - \vec{x}_0$ porque $A\vec{x}_0 = \vec{b}$. \square

Exprimiendo el silicio [opcional]. Sin cargar paquetes especiales, `matlab/octave` no hace manipulación simbólica por ello hay que tener mucha precaución con cantidades como el rango que no varían continuamente. Si pensamos en la matriz 2×2 de rango 1 que tiene como primera fila $(\sqrt{3}, 3)$ y como segunda fila $(1, \sqrt{3})$, cualquier aproximación interna por la cual $\sqrt{3}$ no sea un número que al cuadrado dé tres producirá el valor erróneo 2 para el rango. Las calculadoras y ordenadores tienen trucos para cambiar un dos coma muchos nueves por 3 basándose en que los humanos tratamos la mayor parte del tiempo con números redondos pero eso no siempre funciona. Aquí hay un ejemplo basado en la documentación de `matlab`. Incluye el comando `diag` que permite teclear menos al introducir una matriz diagonal, por otro lado, `rank` es el comando para hallar el rango.

```
1 % Variante la documentación de matlab
2 A = diag([20,21,2021,1e-15])
3 rank(A)
```

El resultado que da `matlab/octave` es 3 cuando está claro que el rango correcto es 4. Hay una variante de `rank` que permite especificar una tolerancia y tener algún control sobre estos errores.

La instrucción `A\b` da una solución particular de $Ax=b$ (como curiosidad, el caso incompatible no produce un error sino que devuelve una solución aproximada). Si el sistema es compatible indeterminado, por supuesto, es raro que coincida con la que nosotros hayamos imaginado. Lo mismo ocurre con los vectores del Teorema 1.2.2, que forman una matriz que se calcula con `null(A)` y que tampoco están únicamente determinada. En `matlab/octave` es posible definir matrices por bloques y por ello la ampliada se obtiene simplemente con `[A b]`.

Las siguientes líneas de código hallan primero los rangos de A y A^+ para el sistema del segundo ejemplo y después calcula \vec{x}_0 y la matriz F formada por los vectores \vec{v}_j .

```
1 % Matriz A
2 A = [1,1,-1,2,-3; -3,-3,4,-6,8; 2,2,1,4,-7];
3 % vector b
4 b = [-1;2;-3];
5 % Matriz A ampliada con b
6 Aplus = [A b];
7 % Calcula los rangos
8 rank(A)
9 rank(Aplus)
10
11 % Solución del sistema Ax=b
12 % Calcula una solución particular
13 x_0 = A\b
14 % Los vectores v_j forman la matriz
15 F = null(A)
```

El rango de A y A^+ se obtiene correctamente como 3. La solución particular y la matriz formada por los vectores \vec{v}_j que ofrece el programa es

$$\mathbf{x}_0 \rightarrow \begin{pmatrix} 3.3333\text{e-}01 \\ 3.3333\text{e-}01 \\ -5.8634\text{e-}16 \\ 6.6667\text{e-}01 \\ 1.0000\text{e+}00 \end{pmatrix}, \quad \mathbf{F} \rightarrow \begin{pmatrix} -5.9545\text{e-}01 & 6.9193\text{e-}01 \\ -5.5886\text{e-}01 & -7.2181\text{e-}01 \\ 1.7347\text{e-}15 & -1.8041\text{e-}15 \\ 5.7716\text{e-}01 & 1.4940\text{e-}02 \\ 3.8303\text{e-}15 & -4.3854\text{e-}15 \end{pmatrix}$$

Aparte de que no coincida con lo que hemos obtenido, vemos el efecto de la computación no simbólica con algunos números infinitesimales. Por ejemplo, en \vec{x}_0 es de suponer que $-5.8634\text{e-}16$ lo podríamos cambiar por cero y el resto por las fracciones de denominador tres a las que se parecen. Concretamente, es fácil comprobar que $\frac{1}{3}(1, 1, 0, 2, 3)^t$ es solución.

Quizá te intrigue que tanto en el ejemplo del rango como aquí se confunda algo del orden de 10^{-15} con cero. En las calculadoras y el *software* numérico existe un $\epsilon > 0$, llamado *épsilon máquina*, tal que 1 y $1 + \delta$ son indistinguibles para $0 < \delta < \epsilon$. Con los 64 bits empleados en doble precisión se suelen dedicar 1 bit al signo, 52 bits a la mantisa y 11 bits al exponente (si no estás al tanto de la terminología, para $-0.8125 = -13 \cdot 2^{-4}$ el signo es $-$, el exponente -4 y la mantisa 13) por tanto $\epsilon = 2^{-52} \approx 2.2 \cdot 10^{-16}$ es un valor habitual del *épsilon máquina*, en particular el de *matlab/octave*.

La computación simbólica de serie en *sagemath* elimina estos problemas. Obviamente hay variaciones en el nombre de los comandos. El análogo del programa anterior es:

```

1 # Matriz A
2 A = matrix(3,5,[1,1,-1,2,-3, -3,-3,4,-6,8, 2,2,1,4,-7])
3 # vector b, tambien b = matrix(3,1,[-1,2,-3])
4 b = vector([-1,2,-3])
5 # Matriz A ampliada con b
6 Aplus = A.augment(b)
7 # Calcula los rangos
8 print(rank(A))
9 print(rank(Aplus))
10
11 # Solución del sistema Ax=b
12 # Calcula una solución particular
13 print(A.solve_right(b))
14 # Los vectores v_j son
15 print(A.right_kernel())

```

Para ilustrar el constructor de vectores, se ha usado para \mathbf{b} en vez del de matrices, que también funcionaría. La solución particular dada por el programa es $(2, 0, 0, 0, 1)$, la misma que habíamos calculado nosotros (aunque no se ha usado, $\mathbf{A} \backslash \mathbf{b}$ también funciona en *sagemath*), y los vectores son $\vec{v}_1 = (1, 1, 0, -1, 0)$ y $\vec{v}_2 = (0, 2, 0, -1, 0)$ que no coinciden con los nuestros pero no es difícil relacionarlos con ellos.

1.3. Matriz inversa

Antes de entrar en el tema del título de esta sección, vamos a dejar de refunfuñar por la falta de unicidad de la eliminación de Gauss. Para ello, una vez que la hayamos

aplicado a nuestro gusto utilizaremos la segunda transformación elemental forzando que todos los pivotes sean unos y la primera transformación elemental para que todos los números encima de cada pivote sean ceros. De esta forma los unos que conforman los pivotes son los únicos elementos no nulos en su columna. Se dice que esta forma extendida de la eliminación de Gauss es la *eliminación de Gauss-Jordan*³ o la *reducción de Gauss-Jordan*. La matriz resultante se dice que es la *forma escalonada reducida* de la matriz de partida.

Veamos un ejemplo. Al final de la primera sección al resolver un sistema habíamos aplicado eliminación de Gauss en la forma $f_1 \leftrightarrow f_2$, $f_3 \mapsto f_3 - f_1$, $f_3 \mapsto f_3 - f_2$ para conseguir

$$A^+ = \left(\begin{array}{ccc|c} 0 & 2 & 1 & 3 \\ 1 & 1 & -3 & -1 \\ 1 & 3 & -2 & 2 \end{array} \right) \longrightarrow E = \left(\begin{array}{ccc|c} 1 & 1 & -3 & -1 \\ 0 & 2 & 1 & 3 \\ 0 & 0 & 0 & 0 \end{array} \right).$$

Esta es una matriz escalonada pero no la forma escalonada reducida de A^+ . En primer lugar tenemos que dividir la segunda fila por 2 para que e_{22} pase a ser 1 y después cancelar el $e_{12} = 1$ que tiene encima. Los cálculos son:

$$E \xrightarrow{f_2 \mapsto f_2/2} \left(\begin{array}{ccc|c} 1 & 1 & -3 & -1 \\ 0 & 1 & 1/2 & 3/2 \\ 0 & 0 & 0 & 0 \end{array} \right) \xrightarrow{f_1 \mapsto f_1 - f_2} \left(\begin{array}{ccc|c} 1 & 0 & -7/2 & -5/2 \\ 0 & 1 & 1/2 & 3/2 \\ 0 & 0 & 0 & 0 \end{array} \right).$$

Esta última ya es la forma escalonada reducida.

Seguro que te estás preguntando qué tiene esto de particular. En primer lugar, da las soluciones de un sistema directamente, sin necesidad de sustituir, en segundo lugar permite resolver varios sistemas al tiempo si comparten la misma matriz y finalmente, es útil para el cálculo de la matriz inversa de la que habla el título.

Veamos la primera ventaja sobre el ejemplo anterior. Allí x_3 es un parámetro arbitrario, $x_3 = \lambda$, y la primera fila da directamente $x_1 = (7\lambda - 5)/2$, sin utilizar las otras ecuaciones. De la misma forma la segunda fila da $x_2 = (3 - \lambda)/2$. Nótese que la última columna es una solución particular, correspondiente a $\lambda = 0$, para las variables de las columnas pivote.

En general, en un sistema compatible $A\vec{x} = \vec{b}$, para $i \leq \text{rg}(A)$ la i -ésima fila de la forma escalonada reducida de A^+ nos permite hallar la variable correspondiente a la columna del i -ésimo pivote. Pensándolo con cuidado esto se traduce en que solo hay una forma escalonada reducida aunque haya muchos modos de llegar a ella.

Proposición 1.3.1. *Cada matriz A tiene solo una forma escalonada reducida.*

Demostración. Si $A \in \mathcal{M}_{m \times n}$ y $\text{rg}(A) = n$ todas las columnas tienen pivotes y la única matriz escalonada posible E es la que tiene $e_{ii} = 1$ para $1 \leq i \leq n$ y el resto

³Este Jordan es W. Jordan que trabajaba en geodesia, no tiene que ver con el matemático famoso C. Jordan cuyo nombre aparecerá en el último capítulo [1].

de los elementos cero. Si $\text{rg}(A) < n$, consideremos el sistema homogéneo $A\vec{x} = \vec{0}$ y digamos que E y E' son dos formas escalonadas reducidas de A . Por el Lema 1.2.1, si $e_{ij} = 1$ es un pivote también lo es $e'_{ij} = 1$ y las columnas con pivote coinciden. Al usar E y E' para resolver el sistema, la i -ésima fila implica la relación

$$x_j = - \sum_l e_{iql} x_{ql} = - \sum_l e'_{iql} x_{ql} \quad \text{con } q_l \text{ las columnas sin pivote.}$$

Ahora bien, como x_{q_l} son parámetros arbitrarios, podemos elegirlos todos nulos excepto uno, con lo que se deduce $e_{iql} = e'_{iql}$ cualquiera que sea $1 \leq l \leq n - r$. \square

La segunda ventaja, acerca de la resolución simultánea de varios sistemas, está relacionada con la primera. Como en la eliminación de Gauss-Jordan no hay que sustituir, al aplicarla a una matriz $(A|\vec{b}_1\vec{b}_2 \dots \vec{b}_k)$, ampliada con el segundo miembro de varios sistemas, podemos leer en las k últimas columnas de la forma escalonada reducida soluciones particulares para las variables de las columnas pivote.

Supongamos que extendemos el ejemplo anterior a resolver los sistemas $A\vec{x} = \vec{b}_1$, $A\vec{x} = \vec{b}_2$ y $A\vec{x} = \vec{b}_3$ con

$$A = \begin{pmatrix} 0 & 2 & 1 \\ 1 & 1 & -3 \\ 1 & 3 & -2 \end{pmatrix}, \quad \vec{b}_1 = \begin{pmatrix} 3 \\ -1 \\ 2 \end{pmatrix}, \quad \vec{b}_2 = \begin{pmatrix} -2 \\ 0 \\ -2 \end{pmatrix}, \quad \text{y} \quad \vec{b}_3 = \begin{pmatrix} 2 \\ 2 \\ 4 \end{pmatrix}.$$

Ya habíamos mencionado que se puede llevar a cabo la reducción de Gauss sobre A con $f_1 \leftrightarrow f_2$, $f_3 \mapsto f_3 - f_1$, $f_3 \mapsto f_3 - f_2$. Haciendo los cálculos (aprovechando lo anterior, basta hacerlos para \vec{b}_2 y \vec{b}_3), se obtiene

$$(A|\vec{b}_1\vec{b}_2\vec{b}_3) \quad \longrightarrow \quad E = \left(\begin{array}{ccc|ccc} 1 & 1 & -3 & -1 & 0 & 2 \\ 0 & 2 & 1 & 3 & -2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right).$$

Siguiendo con la reducción de Gauss-Jordan, como antes, con $f_2 \mapsto f_2/2$ y $f_1 \mapsto f_1 - f_2$ se llega a la forma escalonada reducida. De nuevo, la única diferencia es que tenemos dos columnas nuevas y basta trabajar sobre ellas.

$$E \xrightarrow{f_2 \mapsto f_2/2} \left(\begin{array}{ccc|ccc} 1 & 1 & -3 & -1 & 0 & 2 \\ 0 & 1 & 1/2 & 3/2 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right) \xrightarrow{f_1 \mapsto f_1 - f_2} \left(\begin{array}{ccc|ccc} 1 & 0 & -7/2 & -5/2 & 1 & 1 \\ 0 & 1 & 1/2 & 3/2 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right).$$

Las tres últimas columnas dan soluciones particulares para x_1 y x_2 (correspondientes a los pivotes) de los sistemas con $x_3 = 0$. Tomando $x_3 = \lambda$ se deduce que las soluciones generales de los tres sistemas son, respectivamente,

$$\begin{cases} \vec{x} = (-5/2, 3/2, 0)^t + \lambda\vec{v}, \\ \vec{x} = (1, -1, 0)^t + \lambda\vec{v}, \\ \vec{x} = (1, 1, 0)^t + \lambda\vec{v}, \end{cases} \quad \text{con} \quad \vec{v} = \begin{pmatrix} 7/2 \\ -1/2 \\ 1 \end{pmatrix}.$$

Pasamos por fin al tema principal de esta sección. A estas alturas todos sabemos que solo se calculan inversas de matrices cuadradas pero es el momento de echarle un poco de literatura y entender por qué. Si no eres *filomatemático*, todo lo que necesitas saber está en el enunciado de la Proposición 1.3.2 y tienes permiso para saltarte lo que media hasta allí y su demostración si te resulta aburrido.

El inverso de un número x es otro número y tal que $xy = 1$, como la multiplicación es *conmutativa*⁴, no importa dónde poner la y pero en el caso de matrices el producto puede depender del orden e incluso solo tener sentido en uno de los órdenes posibles. Como unas veces necesitamos un orden y otras veces otro, lo lógico es decir que una matriz $A \in \mathcal{M}_{m \times n}$ es *invertible* si existen matrices B y C , llamadas provisionalmente *inversa por la izquierda* e *inversa por la derecha* tales que BA y AC son matrices identidad. Basta pensar un momento en las dimensiones de las matrices para deducir que esto requiere $B, C \in \mathcal{M}_{n \times m}$. A pesar de que la definición no da ningún indicio de ello, se prueba que la única posibilidad es $B = C$. La demostración es simple pero ingeniosa, difícil que a uno se le ocurra. Se reduce a la siguiente línea⁵

$$(1.3) \quad B = BI_n = B(AC) = (BA)C = I_m C = C.$$

Con esto, la distinción de inversa por la derecha o por la izquierda se revela innecesaria. En definitiva, tenemos la definición equivalente de que A es invertible si existe una matriz C tal que $CA = AC = I$. Todavía no sabemos que tal matriz solo pueda existir para matrices cuadradas, ni si para una A puede haber varias posibles C . En fin, somos un mar de dudas, que quedan aclaradas en el siguiente resultado que da una caracterización sencilla de las matrices invertibles.

Proposición 1.3.2. *Una matriz A es invertible si y solo si $A \in \mathcal{M}_n$ y $\text{rg}(A) = n$. En ese caso, existe una sola matriz que llamaremos matriz inversa de A y denotaremos con A^{-1} , tal que $A^{-1}A = AA^{-1} = I$.*

Demostración. Si $A \in \mathcal{M}_n$ y $\text{rg}(A) = n$, el Corolario 1.2.4 asegura que los sistemas que resultan al comparar columnas en la ecuación matricial $AX = I_n$ con $X \in \mathcal{M}_n$ son compatibles determinados. Con ello existe una inversa a la derecha C , esto es, $AC = I$. Necesariamente $\text{rg}(C) = n$ porque en otro caso, de nuevo por el Corolario 1.2.4, existiría $\vec{x} \neq \vec{0}$ con $C\vec{x} = \vec{0}$, lo que contradiría $AC\vec{x} = I\vec{x}$. Entonces C también tiene inversa a la derecha $CC' = I$ y con un razonamiento que recuerda a (1.3),

$$CA = CA(CC') = C(AC)C' = CC' = I.$$

⁴Recuerda, una operación $*$ entre los elementos de un conjunto se dice que tiene la propiedad conmutativa si $A * B = B * A$. Para números reales o en general complejos, la suma y el producto son conmutativos pero para matrices solo lo es la suma (de matrices de las mismas dimensiones).

⁵Aquí se está usando la propiedad *asociativa* del producto de matrices, es decir, que $B(AC) = (BA)C$. Esto nos lo creemos todos sin dudarlo. Si nos dejaran sin comer hasta que no diéramos una prueba matemática, romperíamos el ayuno utilizando que el elemento ij en estos productos es $\sum_k b_{ik} \sum_l a_{kl} c_{lj}$ y $\sum_k b_{ik} a_{kl} \sum_l c_{lj}$, por tanto la asociativa del producto de matrices se sigue de la asociativa de los números reales o complejos que ya nadie pedirá que demostremos [10].

Por tanto $CA = AC = I$ y A es invertible. Además como los sistemas eran compatibles determinados, esta C es única.

Falta por probar que si A es invertible entonces es cuadrada de rango máximo. Si $A \in \mathcal{M}_{m \times n}$ y $\text{rg}(A) < n$ entonces, como antes, $A\vec{x} = \vec{0}$ tiene una solución $\vec{x} \neq \vec{0}$ por el Corolario 1.2.4 y esto contradice $CA\vec{x} = I\vec{x}$. De la misma forma, $C^t A^t \vec{x} = I\vec{x}$, que se deduce trasponiendo $CA = I$, lleva a contradicción si $\text{rg}(A^t) < m$. Entonces si A es invertible $\text{rg}(A) = n$ y $\text{rg}(A^t) = m$. Obviamente ambos rangos son como mucho el mínimo de m y n (a lo más un pivote por fila o columna), por tanto $n = m$. \square

Para practicar, calculemos la inversa de la siguiente matriz utilizando la eliminación de Gauss-Jordan:

$$A = \begin{pmatrix} 1 & 2 & -1 \\ 2 & 5 & 1 \\ -1 & 1 & 11 \end{pmatrix}.$$

Para ello debemos aplicar la reducción de Gauss a $(A|I_3)$,

$$(A|I_3) \xrightarrow[\substack{f_2 \mapsto f_2 - 2f_1 \\ f_3 \mapsto f_3 + f_1}]{} \left(\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ 0 & 1 & 3 & -2 & 1 & 0 \\ 0 & 3 & 10 & 1 & 0 & 1 \end{array} \right) \xrightarrow{f_3 \mapsto f_3 - 3f_2} \left(\begin{array}{ccc|ccc} 1 & 2 & -1 & 1 & 0 & 0 \\ 0 & 1 & 3 & -2 & 1 & 0 \\ 0 & 0 & 1 & 7 & -3 & 1 \end{array} \right).$$

Los pivotes ya son unos y solo resta crear los ceros encima de ellos:

$$\xrightarrow{f_1 \mapsto f_1 - 2f_2} \left(\begin{array}{ccc|ccc} 1 & 0 & -7 & 5 & -2 & 0 \\ 0 & 1 & 3 & -2 & 1 & 0 \\ 0 & 0 & 1 & 7 & -3 & 1 \end{array} \right) \xrightarrow[\substack{f_1 \mapsto f_1 + 7f_3 \\ f_2 \mapsto f_2 - 3f_3}]{} \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 54 & -23 & 7 \\ 0 & 1 & 0 & -23 & 10 & -3 \\ 0 & 0 & 1 & 7 & -3 & 1 \end{array} \right).$$

En definitiva,

$$A^{-1} = \begin{pmatrix} 54 & -23 & 7 \\ -23 & 10 & -3 \\ 7 & -3 & 1 \end{pmatrix}$$

Quizá a alguien no se le haya pasado por alto que la matriz de partida es invariante por trasposición $A = A^t$, es simétrica, y la inversa también tiene esa propiedad. ¿Es una casualidad? No, porque que una matriz A sea invertible es equivalente a A^t sea invertible y en ese caso

$$(1.4) \quad (A^{-1})^t = (A^t)^{-1}.$$

Esto se sigue notando que al trasponer $A^{-1}A = I$ y $AA^{-1} = I$ se obtienen $A^t(A^{-1})^t = I$ y $(A^{-1})^t A^t = I$ y esto dice que la inversa de A^t es $(A^{-1})^t$. Con respecto a los productos, la inversa se comporta como la trasposición invirtiendo el orden. Concretamente, es fácil probar que si A_1, \dots, A_k son matrices invertibles entonces $A_1 A_2 \cdots A_k$ también lo es y

$$(A_1 A_2 \cdots A_k)^{-1} = A_k^{-1} A_{k-1}^{-1} \cdots A_1^{-1}.$$

Seguro que tras el último ejemplo del cálculo de una inversa con reducción de Gauss-Jordan más de uno está murmurando “eso lo sé hacer yo más rápido con determinantes”. No lo niego pero estoy seguro de que en un ejemplo típico 5×5 , Gauss-Jordan te ganaría por goleada. En breve, cuando la dimensión es grande los determinantes se vuelven impracticables. Volveremos sobre ello en un capítulo posterior.

Lo que sí es cierto es que si queremos escribir la fórmula general que da la inversa de una matriz 2×2 y si hemos memorizado la fórmula con determinantes no hay que hacer prácticamente nada. Para el que tenga flaca la memoria y para el que quiera practicar, veamos cómo se haría con Gauss-Jordan.

Partimos de

$$(A|I) = \left(\begin{array}{cc|cc} a & b & 1 & 0 \\ c & d & 0 & 1 \end{array} \right).$$

Supongamos primero $a \neq 0$ de modo que a_{11} es un pivote y llevamos a cabo la eliminación de Gauss con

$$(A|I) \xrightarrow{f_1 \mapsto f_1/a} \left(\begin{array}{cc|cc} 1 & b/a & 1/a & 0 \\ c & d & 0 & 1 \end{array} \right) \xrightarrow{f_2 \mapsto f_2 - cf_1} \left(\begin{array}{cc|cc} 1 & b/a & 1/a & 0 \\ 0 & \Delta/a & -c/a & 1 \end{array} \right)$$

donde se ha abreviado $\Delta = ad - bc$. Necesitamos $\Delta \neq 0$ para que $\text{rg}(A) = 2$ y por tanto que sea invertible. Terminamos la reducción de Gauss-Jordan con

$$\xrightarrow{f_2 \mapsto af_2/\Delta} \left(\begin{array}{cc|cc} 1 & b/a & 1/a & 0 \\ 0 & 1 & -c/\Delta & a/\Delta \end{array} \right) \xrightarrow{f_1 \mapsto f_1 - bf_2/a} \left(\begin{array}{cc|cc} 1 & 0 & d/\Delta & -b/\Delta \\ 0 & 1 & -c/\Delta & a/\Delta \end{array} \right)$$

Entonces cuando $a \neq 0$ la matriz A es invertible si y solo si $\Delta \neq 0$ y se tiene

$$A^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.$$

¿Y qué ocurre si $a = 0$? En ese caso si A es invertible $c \neq 0$ porque $\text{rg}(A) = 2$ y se podría repetir el razonamiento intercambiando las dos primeras filas en el primer paso de la eliminación de Gauss. Más profundo y más rápido es notar que hemos probado que

$$\frac{1}{\Delta} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} A = A \frac{1}{\Delta} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} = I_2 \quad \text{para } a \neq 0.$$

Nada nos impide tomar límites $a \rightarrow 0$ siempre que $\Delta \neq 0$ y la continuidad de la expresión prueba que también es válida para $a = 0$. Un poco más pedestre es ver que a mano que esta identidad es cierta para $a = 0$ operando los productos.

La fórmula por supuesto funciona igual con reales y complejos. Un ejemplo es:

$$A = \frac{1}{3} \begin{pmatrix} 2i & 1+2i \\ -1+2i & -2i \end{pmatrix} \implies A^{-1} = \frac{1}{3} \begin{pmatrix} -2i & -1-2i \\ 1-2i & 2i \end{pmatrix}.$$

Por razones que es fácil que justifiques, es correcto invertir $1/3$ a 3 y multiplicar por la inversa del resto. Una particularidad de este ejemplo es que $A^{-1} = A^\dagger$. En un capítulo posterior daremos un nombre a esta situación. Por ahora, para captar tu atención solo diré que el caso 2×2 , como este, corresponde a puertas lógicas actuando sobre un solo *qubit* en ordenadores cuánticos.

Exprimiendo el silicio [opcional]. En `matlab/octave` el comando `rref` (abreviatura de *reduced row echelon form*) produce la forma escalonada reducida de una matriz. Ya sabemos que la inversa se podría obtener con ello, ampliando con la matriz identidad pero hay un comando directo `inv` y también reconoce A^{-1} .

El siguiente código comprueba la forma escalonada reducida usada para el triple sistema en un ejemplo anterior y el cálculo de la inversa 3×3 que habíamos hecho, tanto a través de `rref` como con el comando directo `inv`.

```

1 % Matriz y vectores
2 A = [0,2,1; 1, 1, -3; 1,3,-2 ];
3 b1 = [3;-1;2]
4 b2 = [-2;0;-2]
5 b3 = [2;2;4]
6 % Forma escalonada reducida
7 rref( [A, b1, b2, b3] )
8
9 % Matriz
10 A = [1,2,-1; 2, 5, 1; -1,1,11 ];
11 % Inversa con Gauss-Jordan sobre [A I]
12 E = rref( [A, eye(3)] );
13 % Esto muestra las últimas tres columnas
14 E(:,4:6)
15
16 % Inversa con un comando directo (también A^-1)
17 inv(A)

```

Si se usan dos variables para capturar la salida de `rref`, la segunda contendrá las columnas pivote. Esto es, si cambiamos la línea 7 por `[X, p] = rref([A, b1, b2, b3])` entonces `X` será la forma escalonada y `p` será la matriz $[1 \ 2]$ que indica que los pivotes están en las dos primeras columnas.

Se puede demostrar que el número de operaciones para calcular la inversa de una matriz $n \times n$ con eliminación de Gauss-Jordan es similar a $2n^3/3$ en el sentido de que el cociente entre ambas cantidades tiende a 1 cuando $n \rightarrow \infty$ [11, §2.1]. Despreciando el tiempo de acceso a la memoria, esto implica que a la larga el tiempo de computación se multiplica por 8 cuando el tamaño de la matriz se duplica. Para comprobarlo, he ejecutado `rref` para matrices de dimensiones muy grandes con elementos aleatorios en $[0, 1]$.

En mi portátil (viejo y lento) los tiempos presentaban cierta variabilidad por ello consideré el siguiente programa que incluye un bucle `for` que repite el cálculo 20 veces y toma la mediana, `median`, de los tiempos de computación. Esta es menos sensible que la media a valores extremos. Los comandos `tic` y `toc` permiten medir intervalos de tiempo real (no de la CPU).

```

1 % Dimensión
2 N = 500
3 time_val = zeros(1,20);
4 for k = 1:20
5     % Random matrix N x N

```

```

6         A = [rand(N) eye(N)];
7         t0 = tic;
8         E = rref(A);
9         time_val(k) = toc(t0);
10    end
11    median(time_val)

```

Al ejecutarlo tres veces el valor medio fue 2.80 mientras que al cambiar en la segunda línea 500 por 1000, pasó a ser 21.96. El cociente es alrededor de 7.84 que se parece lo suficiente a 8 como para no extrañarse del resultado.

El análogo para `sagemath` del primer programa, sería:

```

1  # Matriz A
2  A = matrix(3,3,[0,2,1, 1, 1, -3, 1,3,-2])
3  # Ampliada con los tres vectores
4  A = A.augment(vector([3,-1,2]))
5  A = A.augment(vector([-2,0,-2]))
6  A = A.augment(vector([2,2,4]))
7  # Forma escalonada reducida
8  print( A.echelon_form() )
9
10 # Matriz
11 A = matrix(3,3, [1,2,-1, 2, 5, 1, -1,1,11 ])
12 B = (A.augment( identity_matrix(3) )).echelon_form()
13 print( B[:,3:] )
14
15 # Inversa con un comando directo (también A^-1)
16 print( A.inverse() )

```

Para los no iniciados, `B[:,3:]` es lo mismo que `B[0:3,3:6]`.