Análisis Numérico II para Ingeniería Matemática

Michael Karkulik
michael.karkulik@usm.cl

Índice general

Capítulo 1

Álgebra lineal numérica

El objetivo de álgebra lineal numérica es realizar las operaciones de álgebra lineal con algoritmos numéricos. Vamos a trabajar sobre el campo de escalares $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$. Matrices $A \in \mathbb{K}^{m\times n}$ tienen m filas y n columnas, y $\mathbb{K}^n = \mathbb{K}^{n,1}$, es decir vectores $x \in \mathbb{K}^n$ siempre son columnas:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}.$$

Si $A \in \mathbb{K}^{m\times n}$, usamos la notación $A^{\mathsf{H}} := \overline{A}^{\top} \in \mathbb{K}^{n\times m}$, donde $\overline{x}$ representa el conjugado de $x \in \mathbb{C}$ y $A^{\top}$ es la matriz traspuesta[1]. Notamos que si $x \in \mathbb{K}^n$ un vector columna, entonces x^{H} es un vector fila, y el producto interno Euclidiano en $\mathbb{K}^n$ es $\langle x\,, y\rangle := x^{\mathsf{H}}y$. Recordamos que $(AB)^{\mathsf{H}} = B^{\mathsf{H}}A^{\mathsf{H}}$. La matriz de identidad en $\mathbb{K}^{n\times n}$ la anotamos como I_n. Recordamos que una matriz $Q \in \mathbb{K}^{n\times n}$ se llama *unitaria*, si $Q^{\mathsf{H}} = Q^{-1}$ (también se dice *ortogonal* si $\mathbb{K} = \mathbb{R}$, es decir $Q^{\top} = Q^{-1}$).

1.1. Repaso/Profundización de Análisis Numérico I

En Análisis Numérico I hemos presentado métodos para resolver un sistema lineal $Ax = b$ con $A \in \mathbb{R}^{n\times n}$. En problemas reales, el tamaño n del sistema suele ser muy grande, es decir $n \approx 10^9$. Si queremos implementar algoritmos para resolver sistemas lineales, tenemos que tener en cuenta lo siguiente.

(1) **Costo de almacenamiento:** Una matriz $A \in \mathbb{R}^{n\times n}$ en doble precisión necesita $8\cdot n^2$ bytes de memoria. Por ejemplo, para $n = 10^6$ necesitamos aproximadamente 8000 Gigabyte de memoria. Destacamos que todos los datos que se quieren usar en una calculación deberian estar en la *memoria de acceso aleatorio* (RAM) de la maquina.

[1] Si $\mathbb{K} = \mathbb{R}$, entonces $A^{\mathsf{H}} = A^{\top}$

(2) **Costo operacional:** El tiempo de cálculo necesario debe ser lo menor posible. Una medida standard del costo operacional es la cantidad de operaciones aritméticas $(+, -, \cdot, /)$ que requiere un algoritmo. La unidad de una operacion aritmética se llama **flop** (floating point operation). Usualmente, la cantidad de flops es un polinomio en n. Por ejemplo, para calcular el producto Ax con $A \in \mathbb{R}^{n\times n}$ se necesita $n(2n-1) = 2n^2-n$ flops. Para n grande, el termino n es despreciable comparado con el termino $2n^2$, y se dice que el costo operacional es **asintóticamente** n^2. Visualizaremos la importancia del costo computacional con un ejemplo. La regla de Cramer dice que la solución x del sistema $Ax = b$ es dada por $x_j = \det(A_j)/\det(A)$, donde A_j es la matriz que se obtiene reemplazando en A su columna j-ésima por b. Si los determinantes se calculan mediante la fórmula recursiva usual, el costo operacional de calcular x es asintóticamente $(n+1)!$. La eliminación de Gauss, por el otro lado, tiene un costo asintótico de n^3. En un computador moderno con 1 Gflop por segundo, se obtiene para $n = 20$ un tiempo computacional de 10^{-5} segundos para la eliminación de Gauss, pero 1500 años para la regla de Cramer.

1.1.1. Normas y condicionamiento

Ya sabemos que si X es un espacio vectorial sobre el campo de escalares $\mathbb{K}$, entonces una función $\|\cdot\| : X \to [0,\infty)$ se llama *norma* sobre X y $(X, \|\cdot\|)$ *espacio normado*, si

(i) $\|x\| = 0$ si y solo si $x = 0$,

(ii) $\|\lambda x\| = |\lambda| \cdot \|x\|$ para todo $x \in X$ y $\lambda \in \mathbb{K}$,

(iii) $\|x + y\| \le \|x\| + \|y\|$ para todo $x, y \in X$.

Las normas mas importantes sobre $X = \mathbb{K}^n$ son

- la *norma euclidiana* inducida por el producto interno $\|x\|_2 := \left(x^{\mathsf{H}}x\right)^{1/2} = \left(\sum_{j=1}^n |x_j|^2\right)^{1/2}$,
- las *normas p* $\|x\|_p := \left(\sum_{j=1}^n |x_j|^p\right)^{1/p}$ para $1 \le p < \infty$,
- la *norma maxima* $\|x\|_\infty := \max_{1\le j\le n} |x_j|$.

Recordamos el siguiente resultado de analisis funcional.

Lema 1. *Sean X un espacio vectorial* ***con dimension finita*** *y $\|\cdot\|_a, \|\cdot\|_b$ dos normas sobre X. Entonces existen dos constantes $C_1, C_2 > 0$ tal que*

$$C_1\|x\|_a \le \|x\|_b \le C_2\|x\|_a \quad \textit{para todo } x \in X.$$

□

Las constantes del último lema dependen de la dimension del espacio. Por ejemplo, para las normas sobre $\mathbb{K}^n$ tenemos

$$\|x\|_\infty \leq \|x\|_p \leq n^{1/p}\|x\|_\infty,$$
$$\|x\|_2 \leq \|x\|_1 \leq n^{1/2}\|x\|_2.$$

Las normas sobre $\mathbb{K}^n$ inducen normas sobre el espacio de las matrices.

Lema 2. *Sean $\|\cdot\|_{\mathbb{K}^n}$ y $\|\cdot\|_{\mathbb{K}^m}$ normas vectoriales sobre $\mathbb{K}^n$ y $\mathbb{K}^m$ y define para $A \in \mathbb{K}^{m\times n}$*

$$\|A\| := \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|_{\mathbb{K}^m}}{\|x\|_{\mathbb{K}^n}}.$$

Entonces,

(i) $\|\cdot\| : \mathbb{K}^{m\times n} \to \mathbb{R}$ *es una norma, y la llamamos norma matricial inducida.*

(ii) Tenemos

$$\|A\| = \sup_{\|x\|_{\mathbb{K}^n}=1} \|Ax\|_{\mathbb{K}^m} = \sup_{\|x\|_{\mathbb{K}^n}\leq 1} \|Ax\|_{\mathbb{K}^m} = \text{ínf}\left\{C > 0 \mid \forall x \in \mathbb{K}^n : \|Ax\|_{\mathbb{K}^m} \leq C\|x\|_{\mathbb{K}^n}\right\}.$$

(iii) Todos los supremos/infimos son máximos/mínimos.

(iv) Para matrices $A \in \mathbb{K}^{m\times p}$ y $B \in \mathbb{K}^{p\times n}$ tenemos $\|A \cdot B\| \leq \|A\| \cdot \|B\|$.

(v) Si $I_n \in \mathbb{K}^{n\times n}$ es la identidad, entonces $\|I_n\| = 1$.

Demostración. Para mostrar el punto (i), escribimos $x = \sum_{j=1}^n x_j e_j$ e usamos Lemma 1 para ver que existe $C > 0$ tal que

$$\|Ax\|_{\mathbb{K}^m} \leq \sum_{j=1}^n |x_j|\|Ae_j\|_{\mathbb{K}^m} \leq \|x\|_\infty \sum_{j=1}^n \|Ae_j\|_{\mathbb{K}^m} \leq \|x\|_{\mathbb{K}^n} C \sum_{j=1}^n \|Ae_j\|_{\mathbb{K}^m}.$$

Por lo tanto, $\|A\| \in [0,\infty)$, y el resto de las propiedades para concluir (i) son consecuencias de que $\|\cdot\|_{\mathbb{K}^n}$ y $\|\cdot\|_{\mathbb{K}^m}$ son normas y A lineal. Para mostrar (ii), notamos que $\|Ax\|_{\mathbb{K}^m}/\|x\|_{\mathbb{K}^n} = \|A(x/\|x\|_{\mathbb{K}^n})\|_{\mathbb{K}^m}$, y por lo tanto

$$\|A\| = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|_{\mathbb{K}^m}}{\|x\|_{\mathbb{K}^n}} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \|A(x/\|x\|_{\mathbb{K}^n})\|_{\mathbb{K}^m} = \sup_{\|x\|_{\mathbb{K}^n}=1} \|Ax\|_{\mathbb{K}^m}.$$

Además,

$$\sup_{\|x\|_{\mathbb{K}^n}=1} \|Ax\|_{\mathbb{K}^m} \leq \sup_{\|x\|_{\mathbb{K}^n}\leq 1} \|Ax\|_{\mathbb{K}^m} \leq \sup_{\|x\|_{\mathbb{K}^n}\leq 1} \frac{\|Ax\|_{\mathbb{K}^m}}{\|x\|_{\mathbb{K}^n}} \leq \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|_{\mathbb{K}^m}}{\|x\|_{\mathbb{K}^n}} = \|A\|.$$

Sea $M := \{C > 0 \mid \forall x \in \mathbb{K}^n : \|Ax\|_{\mathbb{K}^m} \leq C\|x\|_{\mathbb{K}^n}\}$. Notamos que para cada $C' \in M$ y para cada $x \in \mathbb{K}^n$, $\|Ax\|_{\mathbb{K}^m}/\|x\|_{\mathbb{K}^n} \leq C'$. Por definición de supremo obtenemos $\|A\| \leq C'$, y por definición de infimo obtenemos $\|A\| \leq \text{ínf}\, M$. Por otro lado, vemos que $\|A\| \in M$, y asi concluimos que $\|A\| = \text{ínf}\, M$. Para mostrar (iii), notamos que $\{x \in \mathbb{K}^n \mid \|x\|_{\mathbb{K}^n} = 1\}$ es un conjunto compacto, y $x \mapsto Ax$ es continua (incluso continuamente): $\|x - y\| \leq \varepsilon/\|A\|$, entonces $\|Ax - Ay\|_{\mathbb{K}^m} \leq \|A\|\|x - y\|_{\mathbb{K}^n} \leq \varepsilon$. Por lo tanto, $x \mapsto \|Ax\|_{\mathbb{K}^m}$ es continua por la desigualdad triangular inversa, y funciones continuas sobre conjuntos compactos alcanzan sus máximos/mínimos. Los puntos (iv) y (v) son claros. □

En general es dificil calcular explicitamente una norma matricial, pero en algunos casos especiales se puede obtener otra representación mas útil para calculos numéricos. Recordamos que el *radio espectral* de una matriz A se define como

$$\rho(A) := \text{máx}\,\{|\lambda| \mid \lambda \in \mathbb{C} \text{ es un autovalor de } A\}\,.$$

Lema 3. *Si usamos sobre $\mathbb{K}^n$ y $\mathbb{K}^m$ las mismas normas vectoriales y anotamos las normas matriciales inducidas con la misma notación, entonces*

$$\|A\|_1 = \max_{k=1,\ldots,n} \sum_{j=1}^{m} |a_{j,k}|, \quad y \quad \|A\|_\infty = \max_{j=1,\ldots,m} \sum_{k=1}^{n} |a_{j,k}|, \quad y \quad \|A\|_2 = \rho(\overline{A}^\top A)^{1/2}.$$

□

Notamos que $\mathbb{R} \subset \mathbb{C}$. Es decir, para matrices $A \in \mathbb{R}^{m\times n}$ podemos definir

$$\|A\|_{\mathbb{R}} := \sup_{x\in\mathbb{R}^n\setminus\{0\}} \frac{\|Ax\|_{\mathbb{R}^m}}{\|x\|_{\mathbb{R}^n}}, \quad \text{o tambien } \|A\|_{\mathbb{C}} := \sup_{x\in\mathbb{C}^n\setminus\{0\}} \frac{\|Ax\|_{\mathbb{C}^m}}{\|x\|_{\mathbb{C}^n}}.$$

En general, las dos normas son distintas. Para ciertos casos, son iguales. Por ejemplo, según el lema 3 no importa si usamos la version real o compleja de la definición de las normas matriciales con indices $1, 2, \infty$.

Lema 4. *Sea $A \in \mathbb{K}^{n\times n}$. Entonces*

(i) $\rho(A) \leq \|A\|$ para cada norma matricial inducida,

(ii) para cada $\varepsilon > 0$ existe una norma matrical inducida $\|\cdot\|_\varepsilon$ tal que

$$\rho(A) \leq \|A\|_\varepsilon \leq \rho(A) + \varepsilon.$$

Demostración. Mostraremos solo (i). Primero sea $\mathbb{K} = \mathbb{C}$. Sea $\lambda \in \mathbb{C}$ con $|\lambda| = \rho(A)$. Sea $0 \neq x \in \mathbb{C}^n$ el vector propio asociado, $Ax = \lambda x$. Entonces $\rho(A) = |\lambda| = \|Ax\|/\|x\| \leq \|A\|$. Ahora sea $\mathbb{K} = \mathbb{R}$ y $\|\cdot\|$ una norma en $\mathbb{R}^n$ que induce la norma matricial. El problema es que los valores y/o vectores propios pueden ser complejos. Por lo tanto, definimos sobre $\mathbb{C}^n$ la norma

$\|x+iy\|_*^2 = \|x\|^2+\|y\|^2$. Notamos que $\|x\|_* = \|x\|$ para $x \in \mathbb{R}^n$, y de la primera parte obtenemos $\rho(A) \leq \|A\|_*$. Falta mostrar que $\|A\| = \|A\|_*$. Primero,

$$\|A\| = \sup_{x\in\mathbb{R}^n} \frac{\|Ax\|}{\|x\|} = \sup_{x\in\mathbb{R}^n} \frac{\|Ax\|_*}{\|x\|_*} \leq \sup_{x\in\mathbb{C}^n} \frac{\|Ax\|_*}{\|x\|_*} = \|A\|_*.$$

Por otro lado,

$$\|A\|_*^2 = \sup_{x\in\mathbb{C}^n} \frac{\|Ax\|_*^2}{\|x\|_*^2} = \sup_{x,y\in\mathbb{R}^n} \frac{\|A(x+iy)\|_*^2}{\|x+iy\|_*^2} = \sup_{x,y\in\mathbb{R}^n} \frac{\|Ax\|^2+\|Ay\|^2}{\|x\|^2+\|y\|^2} \leq \sup_{x,y\in\mathbb{R}^n} \frac{\|A\|\|x\|^2+\|A\|\|y\|^2}{\|x\|^2+\|y\|^2} = \|A\|.$$

□

Existen normas matriciales que no son inducidas. Por ejemplo, la *norma de Frobenius*

$$\|A\|_F := \left(\sum_{j,k=1}^{n} |a_{j,k}|^2\right)^{1/2}$$

es una norma sobre $\mathbb{K}^{n\times n}$, pero no es inducida para $n \geq 2$, porque $\|I_n\|_F = \sqrt{n} \neq 1$. Tampoco es inducida la norma

$$\|A\|_{\widehat{F}} := \frac{\|A\|_F}{\sqrt{n}},$$

pues la matriz A definida por $a_{jk} = \delta_{jk}\delta_{j1}$, es decir

$$A = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix},$$

satisface $\rho(A) = 1$ y $\|A\|_F = 1/\sqrt{n}$, lo que se contradice con Lema 4. Recordamos el siguiente resultado de Análisis Numérico I.

Lema 5. *Sea $\|\cdot\|$ una norma vectorial y $\|\cdot\|$ la norma matricial inducida. Sea $A \in \mathbb{K}^{n\times n}$ una matriz invertible y $x, \widetilde{x}, b, \widetilde{b} \in \mathbb{K}^n \setminus \{0\}$ con $Ax = b$ y $A\widetilde{x} = \widetilde{b}$. Entonces*

$$\frac{\|x-\widetilde{x}\|}{\|x\|} \leq \|A\|\|A^{-1}\| \frac{\|b-\widetilde{b}\|}{\|b\|}$$

Demostración. El resultado se concluye de $\|x-\widetilde{x}\| = \|A^{-1}(b-\widetilde{b})\| \leq \|A^{-1}\|\|b-\widetilde{b}\|$ y de $\|b\| = \|Ax\| \leq \|A\|\|x\|$. □

El número $\mathrm{cond}(A) := \|A\|\|A^{-1}\|$ se llama *factor de condicionamiento* de A. Usaremos cond_p para indicar el factor de condicionamiento calculado en base a la norma p.

Lema 6. *Sea $Q \in \mathbb{K}^{n\times n}$ una matriz unitaria. Entonces*

(i) $\|Qx\|_2 = \|x\|_2$ *para todo* $x \in \mathbb{K}^n$,

(ii) $\|QA\|_2 = \|AQ\|_2 = \|A\|_2$ *para todo* $A \in \mathbb{K}^{n\times n}$,

(iii) $\|Q\|_2 = 1$ *y* $\operatorname{cond}_2(Q) = 1$.

(iv) La matriz

$$\begin{pmatrix} I_k & 0 \\ 0 & Q \end{pmatrix} \in \mathbb{K}^{(k+n)\times(k+n)}$$

es unitaria.

□

Demostración. Todos los resultados son consecuencia de

$$\|Qx\|_2^2 = (Qx)^{\mathsf{H}} Qx = x^{\mathsf{H}} Q^{\mathsf{H}} Qx = x^{\mathsf{H}} x = \|x\|_2^2.$$

□

1.1.2. Las descomposiciones LU y Cholesky

Definición 7. *Una matriz $L \in \mathbb{K}^{n\times n}$ se llama **triangular inferior**, si $\ell_{jk} = 0$ para $k > j$. Es decir, todos los elementos arriba de la diagonal son zero. Una matriz $U \in \mathbb{K}^{n\times n}$ se llama **triangular superior**, si $u_{jk} = 0$ para $j > k$. Es decir, todos los elementos abajo de la diagonal son zero*[2]*. Las matrices tienen entonces la forma*

$$L = \begin{pmatrix} \ell_{11} & 0 & \dots & \dots & 0 \\ \ell_{21} & \ell_{22} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ \ell_{n1} & \ell_{n2} & \dots & \dots & \ell_{nn} \end{pmatrix}, \quad y \quad U = \begin{pmatrix} u_{11} & u_{12} & \dots & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & u_{nn} \end{pmatrix}.$$

Para resolver un sistema $Ux = b$ con una matriz triangular superior, notamos que en terminos de ecuaciones se lee

$$\begin{aligned} u_{11}x_1 + u_{12}x_2 + \cdots + u_{1,n-1}x_{n-1} + u_{1n}x_n &= b_1 \\ u_{22}x_2 + \cdots + u_{2,n-1}x_{n-1} + u_{2n}x_n &= b_2 \\ &\vdots \\ u_{n-1,n-1}x_{n-1} + u_{n-1,n}x_n &= b_{n-1} \\ u_{nn}x_n &= b_n. \end{aligned}$$

[2] L por *lower* y U por *upper* en ingles.

Empezando con la última ecuacion calculamos primero

$$x_n = \frac{b_n}{u_{nn}}.$$

Ahora conocemos x_n, y usando la penúltima ecuación podemos calcular

$$x_{n-1} = \frac{b_{n-1} - u_{n-1,n}x_n}{u_{n-1,n-1}}.$$

Ahora conocemos x_n y x_{n-1}, y podemos calcular

$$x_{n-2} = \frac{b_{n-2} - u_{n-2,n}x_n - u_{n-2,n-1}x_{n-1}}{u_{n-2,n-2}}.$$

Una vez que conocemos $x_n, x_{n-1}, \ldots, x_{j+1}$, podemos caluclar x_j usando la j-ésima ecuación

$$x_j = \frac{b_j - \sum_{k=j+1}^{n} u_{jk}x_k}{u_{jj}}.$$

Este procedimiento se llama *sustitución ascendente.*

Corolario 8. *Sea $U \in \mathbb{K}^{n\times n}$ una matriz triangular superior invertible. Entonces la sustitucion ascendente calcula la solución x del sistema $Ux = b$ en asintóticamente n^2 flops.*

Demostración. Ya sabemos que si U es invertible, entonces los elementos de su diagonal no son zeros. Por lo tanto, no se produce una division por zero y el algoritmo está bien definido. En el paso j se calcula $n - j$ productos y restas y 1 division. El número total de operaciones es

$$\sum_{j=1}^{n}(1 + 2(n-j)) = n + 2\sum_{k=1}^{n-1} k = n + \frac{(n-1)n}{2} = n^2.$$

□

El algoritmo correspondiente para matrizes triangulares inferiores se llama *sustitución descendente.* Para resolver el sistema $Ax = b$, el método mas comun es la **eliminación de Gauss**. Usando operaciones elementales fila, se transforma el sistema $Ax = b$ a un sistema equivalente (es decir, un sistema que tiene exactamente la misma solución) con una matriz triangular superior $Ux = \widetilde{b}$, lo cuál se puede resolver usando sustitución ascendente. Recordamos que hay tres operaciones elementales fila.

Definición 9. *Las tres operaciones elementales fila son*

(i) Multiplicar fila k con el número α, $E_k(\alpha)$,

(ii) Multiplicar fila k con el número α y sumar a la fila j, $E_{jk}(\alpha)$.

(iii) Intercambiar filas j y k, E_{jk},

En este contexto es comodo usar la notación de la *matriz aumentada* $(A|b)$, a la cual se aplican las operaciones elementales. Por ejemplo, para resolver el sistema $Ax = b$ con

$$A = \begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix}, \qquad b = \begin{pmatrix} 1 \\ 4 \\ 8 \end{pmatrix},$$

generamos la matriz aumentada y aplicamos operaciones elementales fila para generar zeros abajo de la diagonal,

$$\begin{aligned} (A|b) = \left(\begin{array}{ccc|c} 1 & -3 & 2 & 1 \\ -2 & 8 & -1 & 4 \\ 4 & -6 & 5 & 8 \end{array}\right) &\xrightarrow[E_{3,1}(-4)]{E_{2,1}(2)} \left(\begin{array}{ccc|c} 1 & -3 & 2 & 1 \\ 0 & 2 & 3 & 6 \\ 0 & 6 & -3 & 4 \end{array}\right) \\ &\xrightarrow{E_{3,2}(-3)} \left(\begin{array}{ccc|c} 1 & -3 & 2 & 1 \\ 0 & 2 & 3 & 6 \\ 0 & 0 & -12 & -14 \end{array}\right) = (U|\widetilde{b}). \end{aligned} \tag{1.1}$$

El sistema final $Ux = \widetilde{b}$ lo podemos resolver con sustitución ascendente.

En lo que sigue, vamos a representar este proceso en forma matricial. Cada operacion elemental fila de la definición 9 puede ser representada por multiplicación por la izquierda con su *matriz elemental* asociada.

Lema 10. *Sea $A \in \mathbb{R}^{n\times n}$ y $I_n \in \mathbb{R}^{n\times n}$ la matriz de identidad.*

*(a) Sea E la **matriz elemental** de una de las operaciones elementales de la definición 9, es decir, la matriz que se obtiene aplicando la operación elemental a la matriz de identidad I_n. Entonces, el resultado de aplicar la operación elemental a una matriz A es igual a $E \cdot A$.*

*(b) Sea P una matriz que se obtiene aplicando una seria de cambios de filas a la matriz de identidad I_n. Entonces P se llama **matriz de permutación**. Se tiene $P^{-1} = P^\top$.*

Para una matriz elemental vamos a usar la misma notación de la operación elemental. Por ejemplo, la operación $E_{2,1}(\alpha)$ en $\mathbb{R}^3$ se representa por la matriz elemental

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow{E_{2,1}(\alpha)} \begin{pmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = E_{2,1(\alpha)}.$$

Es decir, la operación elemental

$$\begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} \xrightarrow{E_{2,1}(2)} \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 4 & -6 & 5 \end{pmatrix} \tag{1.2}$$

la podemos representar en su forma matricial

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} = \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 4 & -6 & 5 \end{pmatrix}.$$

Por otro lado, todos las cambios de fila en $\mathbb{R}^3$ son

$$\underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{=E_{12}}, \underbrace{\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}}_{=E_{13}}, \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}}_{=E_{23}}.$$

Es decir, la operación elemental

$$\begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} \xrightarrow{E_{1,3}} \begin{pmatrix} 4 & -6 & 5 \\ -2 & 8 & -1 \\ 1 & -3 & 2 \end{pmatrix} \tag{1.3}$$

la podemos representar en su forma matricial

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} = \begin{pmatrix} 4 & -6 & 5 \\ -2 & 8 & -1 \\ 1 & -3 & 2 \end{pmatrix}.$$

Ahora podemos obtener una representación matricial de la eliminación de Gauss (1.1). Primero notamos que la operación elemental fila $E_{2,1}(2)$ es invertible, y su inversa es obviamente $E_{2,1}(-2)$. Es decir, (1.2) se puede escribir como

$$\begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} \xleftarrow{E_{2,1}(-2)} \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 4 & -6 & 5 \end{pmatrix},$$

y según Lema 10 eso se lee en forma matricial

$$\begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 4 & -6 & 5 \end{pmatrix}.$$

La segunda operación elemental $E_{3,1}(-4)$ se explicita entonces como

$$\begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 4 & -6 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 4 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 0 & 6 & -3 \end{pmatrix},$$

y la tercera $E_{3,2}(-3)$ como

$$\begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 0 & 6 & -3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 0 & 0 & -12 \end{pmatrix},$$

Finalmente, podemos representar (1.1) como

$$\begin{pmatrix} 1 & -3 & 2 \\ -2 & 8 & -1 \\ 4 & -6 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 4 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 3 \\ 0 & 0 & -12 \end{pmatrix}.$$

Lema 11. *Sean $A, B \in \mathbb{K}^{n\times n}$ triangular superior (inferior). Entonces el producto AB es triangular superior (inferior).*

Demostración. Demostramos solamente el caso superior. Sabemos $a_{ij} = 0$ para $i > j$ y $b_{jk} = 0$ para $j > k$. Los elementos del producto $D := AB$ satisfacen

$$d_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk} = \sum_{j=i}^{k} a_{ij} b_{jk},$$

o sea $d_{ik} = 0$ para $i > k$. □

Volviendo al ejemplo de arriba, concluimos que $A = LU$ con matriz triangular inferior L y matriz triangular superior U. La matriz U es el resultado de la eliminación de Gauss, y la matriz L contiene todos los factores que usamos en el proceso. De hecho, es facil calcular L, pues

$$\begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 4 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 4 & 3 & 1 \end{pmatrix}.$$

Este procedimiento se extiende obviamente a matrices $A \in \mathbb{R}^{n\times n}$. Sea

$$A := A^{(1)} := \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

(1) **Paso 1:** Multiplicar $m_{j1} := a_{j1}/a_{11}$ con fila 1 y restar de fila j, $j > 1$:

$$A^{(1)} = \underbrace{\begin{pmatrix} 1 & & & & \\ m_{21} & 1 & & & \\ m_{31} & & 1 & & \\ \vdots & & & \ddots & \\ m_{n1} & & & & 1 \end{pmatrix}}_{:=L^{(1)}} \cdot \underbrace{\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ & \vdots & \ddots & \vdots \\ & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix}}_{:=A^{(2)}}$$

(2) **Paso 2:** Multiplicar $m_{j2} := a_{j2}^{(2)}/a_{22}^{(2)}$ con fila 2 y restar de fila j, $j > 2$:

$$A^{(2)} = \underbrace{\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & m_{32} & 1 & & \\ & \vdots & & \ddots & \\ & m_{n2} & & & 1 \end{pmatrix}}_{:=L^{(2)}} \cdot \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ & & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ & & \vdots & \ddots & \vdots \\ & & a_{n3}^{(3)} & \dots & a_{nn}^{(3)} \end{pmatrix}}_{:=A^{(3)}}$$

(3) **Paso k:** Tenemos

$$A^{(k)} = \begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & a_{2n}^{(2)} \\ & & \ddots & & & \vdots \\ & & & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \dots & a_{nn}^{(k)} \end{pmatrix}$$

Multiplicamos $m_{jk} := a_{jk}^{(k)}/a_{kk}^{(k)}$ con fila k y restar de fila j, $j > k$:

$$A^{(k)} = \underbrace{\begin{pmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & m_{(k+1)k} & 1 & & \\ & & & \vdots & & \ddots & \\ & & & m_{nk} & & & 1 \end{pmatrix}}_{:=L^{(k)}} \cdot \underbrace{\begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & & a_{2n}^{(2)} \\ & & \ddots & & & & \vdots \\ & & & a_{kk}^{(k)} & a_{k(k+1)}^{(k)} & \dots & a_{kn}^{(k)} \\ & & & & a_{(k+1)(k+1)}^{(k+1)} & \dots & a_{(k+1)n}^{(k+1)} \\ & & & & \vdots & \ddots & \vdots \\ & & & & a_{n(k+1)}^{(k+1)} & \dots & a_{nn}^{(k)} \end{pmatrix}}_{:=A^{(k+1)}}$$

Despues de $(n-1)$ pasos obtenemos

$$A = L^{(1)} \cdot L^{(2)} \cdot \dots \cdot L^{(n-1)} \cdot A^{(n)},$$

donde $A^{(n)}$ es triangular superior y los $L^{(k)}$ son triangulares inferiores. Como producto de matrices

triangulares, la matriz $L := L^{(1)} \cdot L^{(2)} \cdot \dots \cdot L^{(n-1)}$ es triangular inferior. Mas aún, es fácil calcular

$$L = \begin{pmatrix} 1 & & & & & & \\ m_{21} & 1 & & & & & \\ m_{31} & m_{32} & \ddots & & & & \\ \vdots & \vdots & \vdots & 1 & & & \\ \vdots & \vdots & \vdots & m_{(r+1)r} & 1 & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \\ m_{n1} & m_{n2} & \dots & m_{nr} & \dots & \dots & 1 \end{pmatrix}.$$

Es decir, podemos expresar A como un producto de una matriz triangular inferior con una matriz triangular superior.

Teorema 12 (Factorización LU sin pivote)**.** *Para una matriz $A \in \mathbb{K}^{n\times n}$ son equivalentes los siguientes dos enunciados.*

(i) Cada submatriz $A_k := (a_{ij})_{i,j=1}^{k} \in \mathbb{K}^{k\times k}$ es invertible.

(ii) A tiene una factorización LU, es decir, $A = LU$, con L triangular inferior e invertible, U triangular superior e invertible. Además, existe única descomposición LU con $\ell_{jj} = 1$ para todo $j = 1, \dots, n$.

(iii) La eliminación de Gauss se puede llevar a cabo sin intercambiar filas.

En este caso, la descomposición LU se puede calcular en asintóticamente n^3 flops.

Demostración. Demostramos solo la equivalencia $(i) \Leftrightarrow (ii)$, la relación con la eliminación de Gauss fue abordada arriba.
$\underline{(ii) \Rightarrow (i)}$: Según hipótesis son invertibles L y U, y por lo tanto todos los submatrices L_k y U_k. Dado que $A_k = L_k U_k$, obtenemos (i).
$\underline{(i) \Rightarrow (ii)}$: Procedemos por inducción en n: Para $n = 1$ sabemos $a_{11} \neq 0$, y por lo tanto podemos escribir $a_{11} = l_{11}u_{11}$ con $l_{11}, u_{11} \neq 0$. En el paso inductivo, supongamos $(i) \Rightarrow (ii)$ para $n-1$. Supongamos (i) para n. Escribimos

$$A = \begin{pmatrix} A_{n-1} & b \\ c^\top & a_{nn} \end{pmatrix}$$

con $b, c \in \mathbb{K}^{n-1}$ y $a_{nn} \in \mathbb{K}$. Falta demostrar que existen únicos $\ell, u \in \mathbb{K}^{n-1}$ y $\rho \neq 0 \in \mathbb{K}$ tal que

$$\begin{pmatrix} A_{n-1} & b \\ c^\top & a_{nn} \end{pmatrix} = \begin{pmatrix} L_{n-1} & 0 \\ \ell^\top & 1 \end{pmatrix} \begin{pmatrix} U_{n-1} & u \\ 0 & \rho \end{pmatrix}.$$

Notamos que el último sistema es equivalente a los tres sistemas

$$b = L_{n-1}u, \qquad c = U_{n-1}^\top \ell, \qquad \text{y } a_{nn} = \ell^\top u + \rho.$$

Los primeros dos sistemas tienen únicas soluciones u, ℓ, pues L_{n-1}, U_{n-1} son invertibles según hipótesis. Falta demostrar que $\rho \neq 0$. □

El problema obvio con la la descomposición LU sin pivote (es decir, con la eliminación de Gauss sin intercambiar filas) es que no se puede llevar a cabo si nos encontramos con un elemento en la diagonal que es zero. Por ejemplo, como caso extremo consideramos

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

La matriz A es invertible, pero no podemos hacer ni el primer paso en eliminación de Gauss por $a_{11} = 0$. No obstante, observamos que con un cambio de filas

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Leftrightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_2 \\ b_1 \end{pmatrix}$$

obtenemos una matriz donde la eliminación de Gauss se puede llevar a cabo. En general, en el paso k de la descomposición LU tenemos

$$A^{(k)} = \begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & a_{2n}^{(2)} \\ & & \ddots & & & \vdots \\ & & & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \dots & a_{nn}^{(k)}. \end{pmatrix}$$

Si $a_{kk}^{(k)} = 0$, entonces no podemos seguir con la eliminación de Gauss, pues, los m_{jk} no están bien definidos por una división por zero. En este caso es necesario aplicar un cambio de filas con el fin de obtener un elemento no zero en la diagonal. Aun si $a_{kk}^{(k)} \neq 0$, es aconsejable aplicar un cambio de fila con el fin de obtener un elemento en la diagonal que sea mayor en valor absoluto. Este procedimiento resulta en un algoritmo más estable, que se llama **factorización LU con pivote parcial**. Si la matriz A es invertible, entonces se puede demostrar que existe por lo menos un elemento en la columna restante k de $A^{(k)}$

$$\begin{pmatrix} a_{kk}^{(k)} \\ a_{k+1,k}^{(k)} \\ a_{k+2,k}^{(k)} \\ \vdots \\ a_{n,k}^{(k)} \end{pmatrix}$$

que no es zero. Como ya mencionado, buscamos una fila $\ell \in \{k, \dots, n\}$ con elemento $a_{\ell k}^{(k)}$ mas grande en modulo,

$$|a_{\ell k}^{(k)}| \geq \max_{i=k,\dots,n} |a_{ik}^{(k)}|.$$

Notamos que $a_{\ell k}^{(k)} \neq 0$ si A es invertible, y aplicamos la operacion elemental de cambio de filas $E_{\ell k}$,

$$A^{(k)} = \begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & a_{2n}^{(2)} \\ & & \ddots & & & \vdots \\ & & & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{\ell k}^{(k)} & \dots & a_{\ell n}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \dots & a_{nn}^{(k)} \end{pmatrix} = E_{\ell k} \cdot \begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & a_{2n}^{(2)} \\ & & \ddots & & & \vdots \\ & & & a_{\ell k}^{(k)} & \dots & a_{\ell n}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \dots & a_{nn}^{(k)} \end{pmatrix} = E_{\ell k} \cdot \widetilde{A}^{(k)}.$$

A la matriz $\widetilde{A}^{(k)}$ podemos aplicar el próximo paso en la eliminación de Gauss, pues $a_{\ell k}^{(k)} \neq 0$. Multiplicamos $m_{jk} := a_{jk}^{(k)} / a_{\ell k}^{(k)}$ con fila k y restar de fila j, $j > k$:

$$A^{(k)} = E_{\ell k} \cdot \underbrace{\begin{pmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & 1 & & \\ & & & m_{(k+1)k} & 1 & \\ & & & \vdots & & \ddots \\ & & & m_{nk} & & 1 \end{pmatrix}}_{:=L^{(k)}} \cdot \underbrace{\begin{pmatrix} a_{11} & \dots & \dots & \dots & \dots & \dots & a_{nn} \\ & a_{22}^{(2)} & & & & & a_{2n}^{(2)} \\ & & \ddots & & & & \vdots \\ & & & a_{jk}^{(k)} & a_{j(k+1)}^{(k)} & \dots & a_{jn}^{(k)} \\ & & & & a_{(k+1)(k+1)}^{(k+1)} & \dots & a_{(k+1)n}^{(k+1)} \\ & & & & \vdots & \ddots & \vdots \\ & & & & a_{n(k+1)}^{(k+1)} & \dots & a_{nn}^{(k)} \end{pmatrix}}_{:=A^{(k+1)}}.$$

Al final obtendremos

$$A = P^{(1)} \cdot L^{(1)} \cdot P^{(2)} \cdot L^{(2)} \cdot \dots \cdot P^{(n-1)} \cdot L^{(n-1)} A^{(n)},$$

donde las $L^{(k)}$ representan las eliminaciones de Gauss, y las $P^{(k)}$ son las matrices de permutación de los cambios de filas (si no hay cambio de filas, entonces $P^{(k)} = I_n$ la identidad). Todos los cambios de filas que se producen durante el algoritmo corresponden a una permutación de filas de la matriz original A.

Teorema 13 (Factorización LU con pivote parcial)**.** *Sea $A \in \mathbb{R}^{\times n}$ una matriz invertible. Entonces existe una matriz de permutación P tal que $P \cdot A$ tiene una descomposición LU, es decir $P \cdot A = L \cdot U$, con L triangular inferior e invertible, U triangular superior e invertible. La eliminación de Gauss con pivoteo parcial calcula la descomposicón de LU con pivote parcial en asintóticamente n^3 flops.* □

Tambien se puede hacer un **pivote total**, donde se aplican tambien cambios de columnas. En este caso, la descomposición será $P \cdot A \cdot Q = L \cdot U$, donde P es una matriz de permutación reflejando los cambios de filas, y Q es una matriz de permutación reflejando los cambios de columnas.

Digamos que nuestro objetivo es

$$\text{hallar } x \in \mathbb{R}^n \text{ tal que } Ax = b. \tag{1.4}$$

Si tenemos una factorización LU como $PA = LU$, entonces podemos proceder de la siguiente manera:

(i) usando sustitución descendente, calcular $y \in \mathbb{R}^n$ tal que

$$Ly = Pb.$$

(ii) usando sustitución ascendente, calcular $x \in \mathbb{R}^n$ tal que

$$Ux = y.$$

Obviamente, si x es la solución calculada en (a)–(b), entonces

$$PAx = LUx = Ly = Pb,$$

lo que implica $Ax = b$ y asi x también es solución del sistema original.

Notamos que la solución de (1.4) con eliminación de Gauss necesita asintóticamente n^3 flops, mientras la solución de los sistemas en (i) y (ii) necesita asintóticamente n^2 flops. Digamos que ahora nuestro objetivo es resolver n sistemas

$$Ax_j = b_j, \quad j = 1, \ldots, n, \tag{1.5}$$

con la misma matriz pero diferentes lados derechos. Si resolvemos cada sistema (1.5) con eliminación de Gauss, necesitamos asintóticamente $n \cdot n^3 = n^4$ flops. Por el otro lado, si calculamos primero una factorización LU como $PA = LU$ y resolvemos despues cada sistema (1.5) con el procedimiento (i)-(ii) de arriba, entonces necesitamos $n^3 + n \cdot n^2 = 2n^3$ flops. Notamos que la condición de L y/o U puede ser *mas grande* que la condición de A. En este caso, resolver un sistema lineal con el procedimiento de arriba resulta en un *algoritmo inestable*. Este efecto es peor para factorización LU sin pivote.

Si la matriz A bajo consideración tiene cierta estructura o propiedad, entonces el costo de almacenamiento y el costo operacional para resolver un sistema pueden optimizarse.

Definición 14. *Una matriz $A \in \mathbb{R}^{n \times n}$ se llama*

(1) ***simétrica**, si $a_{jk} = a_{kj}$ para todo j, k. En otras palabras la matriz y su transpuesta son iguales, $A = A^\top$.*

(2) ***definida positiva****, si* $x^\top \cdot A \cdot x > 0$ *para todo* $0 \neq x \in \mathbb{R}^n$*.*

□

Si una matriz A es simétrica, entonces será deseable obtener una descomposición LU con la simetría $U = L^\top$, es decir,

$$A = L \cdot L^\top$$

con L triangular inferior. Para desarollar un algoritmo, simplemente igualamos A y $L \cdot L^\top$ y calculamos los elementos de L en el orden correcto. En el caso de matrices en $\mathbb{R}^{3\times 3}$, obtenemos

$$\begin{pmatrix} a_{11} & a_{21} & a_{31} \\ a_{21} & a_{22} & a_{32} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} \ell_{11} & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} \end{pmatrix} \cdot \begin{pmatrix} \ell_{11} & \ell_{21} & \ell_{31} \\ 0 & \ell_{22} & \ell_{32} \\ 0 & 0 & \ell_{33} \end{pmatrix}.$$

Entonces, recorremos la parte triangular inferior de A y obtenemos

1. columna 1
   a) fila 1: $\ell_{11} = \sqrt{a_{11}}$,
   b) fila 2: $\ell_{21} = a_{21}/\ell_{11}$,
   c) fila 3: $\ell_{31} = a_{31}/\ell_{11}$,
2. columna 2
   a) fila 2: $\ell_{22} = \sqrt{a_{22} - \ell_{21}^2}$,
   b) fila 3: $\ell_{32} = (a_{32} - \ell_{31}\ell_{21})/\ell_{22}$,
3. columna 3
   a) fila 3: $\ell_{33} = \sqrt{a_{33} - \ell_{31}^2 - \ell_{32}^2}$.

Las operaciones críticas que se producen durante el algoritmo (raizes, divisiones) son bien definidas si la matriz A, aparte de ser simétrica, es definida positiva.

Teorema 15 (Cholesky)**.** *La matriz* $A \in \mathbb{R}^{n\times n}$ *es simétrica y definida positiva si y solo si existe* $L \in \mathbb{R}^{n\times n}$ *triangular inferior e invertible, tal que*

$$A = L \cdot L^\top.$$

Adicionalmente, la matriz L *es única bajo la condición* $\ell_{jj} > 0$*. El algoritmo de Cholesky calcula la matriz* L *en asintóticamente* n^3 *flops.*

1.1.3. Matrices ralas

En la siguiente tabla recordamos costo de memoria y computacional para matrices generales:

operación	costo asintótico
almanecenamiento de $A \in \mathbb{R}^{n\times n}$	n^2
$A \cdot x$ para $A \in \mathbb{R}^{n\times n}, x \in \mathbb{R}^n$	n^2
$A + B$ para $A, B \in \mathbb{R}^{n\times n}$	n^2
$A \cdot B$ para $A, B \in \mathbb{R}^{n\times n}$	n^3
resolver $Ax = b$, $A \in \mathbb{R}^{n\times n}$	n^3

En la práctica aparecen matrices que son muy grandes, pero contienen muchos zeros. Estas matrices se llaman *ralas*[3]. Aunque no existe una definición rigurosa, vamos a llamar una matrix $A \in \mathbb{R}^{n\times n}$ *rala* si el número N de elementos no zeros de A es proporcional al tamaño n. El almacenamiento de matrices ralas y operaciones con ellas pueden optimizarse, omitiendo los elementos zeros. Para almacenar una matriz rala es suficiente almacenar los elementos $a_{jk} \neq 0$ y las coordenadas j, k. Es decir, necesitamos tres vectores $\mathbf{j}, \mathbf{k}, \mathbf{a} \in \mathbb{R}^N$, donde N es el número de elementos no zeros de A. Para cada elemento $a_{jk} \neq 0$, existe único inidice ℓ tal que $\mathbf{j}_\ell = j$, $\mathbf{k}_\ell = k$, y $\mathbf{a}_\ell = a_{jk}$. Por ejemplo, la matriz

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 4 & -2 & 0 & 0 \\ 0 & -2 & 4 & -3 & 0 \\ 0 & 0 & -3 & 6 & -2 \\ 0 & 0 & 0 & -2 & 4 \end{pmatrix}$$

la podemos representar en este formato como

$$\begin{aligned} \mathbf{j} &= \big(1,1,2,2,2,3,3,3,4,4,4,5,5\big) \\ \mathbf{k} &= \big(1,2,1,2,3,2,3,4,3,4,5,4,5\big) \\ \mathbf{a} &= \big(2,-1,-1,4,-2,-2,4,-3,-3,6,-2,-2,4\big) \end{aligned}$$

Mencionamos que existen otros formatos, el más eficiente para muchas operaciones es el formato CSR (compressed sparse row).

[3] *sparse* en inglés

1.1.4. Métodos iterativos

En lo siguiente, vamos a mostrar dos matrices y sus factores de Cholesky.

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 & 0 & 0 & 2 \\ 0 & 0 & 1 & 3 & 0 & 0 & 3 \\ 1 & 2 & 3 & 39 & 0 & 0 & 104 \\ 0 & 0 & 0 & 0 & 1 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 & 1 & 6 \\ 1 & 2 & 3 & 104 & 5 & 6 & 8863 \end{pmatrix}, \quad L_A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 2 & 3 & 18 & 5 & 6 & 92 \end{pmatrix}$$

$$B = \begin{pmatrix} 8863 & 6 & 5 & 104 & 3 & 2 & 1 \\ 6 & 1 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 1 & 0 & 0 & 0 & 0 \\ 104 & 0 & 0 & 39 & 3 & 2 & 1 \\ 3 & 0 & 0 & 3 & 1 & 0 & 0 \\ 2 & 0 & 0 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad L_B = \begin{pmatrix} 94{,}1435 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 \\ 0{,}0637 & 0{,}9979 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 \\ 0{,}0531 & -0{,}0033 & 0{,}9985 & 0{,}0000 & 0{,}0000 & 0{,}0000 & 0{,}0000 \\ 1{,}1047 & -0{,}0705 & -0{,}0589 & 6{,}1458 & 0{,}0000 & 0{,}0000 & 0{,}0000 \\ 0{,}0318 & -0{,}0020 & -0{,}0017 & 0{,}4823 & 0{,}8753 & 0{,}0000 & 0{,}0000 \\ 0{,}0212 & -0{,}0013 & -0{,}0011 & 0{,}3215 & -0{,}1779 & 0{,}9297 & 0{,}0000 \\ 0{,}0106 & -0{,}0006 & -0{,}0005 & 0{,}1607 & -0{,}0889 & -0{,}0728 & 0{,}9802 \end{pmatrix}$$

Observamos que en el caso del la matriz A, el factor de Cholesky L_A reproduce la estructura rala de A. Sin embargo, en el caso de la matriz B ocurre un *fill in*, es decir, aunque B es rala, se llena todo el perfil con elementos no zeros en el factor de Cholesky. En el caso extremo, podemos almacenar una matriz A rala, pero no su factor de Cholesky. El efecto se produce en muchos algoritmos. En el caso de resolver sistemas lineales $Ax = b$, se prefiere entonces *métodos iterativos*, que solo requieren la múltiplicación con la matriz A. Métodos iterativos calculan una sucesión $(x_k)_{k\in\mathbb{N}}$, $x_k \in \mathbb{R}^n$, que converge a la solución exacta x del sistema $Ax = b$.

Definición 16 (Método iterativo simple)**.** *Sea $A \in \mathbb{R}^{n\times n}$ invertible y $b \in \mathbb{R}^n$. Sean $M, N \in \mathbb{R}^{n\times n}$ tal que $A = M - N$. Para un vector inicial $x_0 \in \mathbb{R}^n$ se define la iteración*

$$Mx_{k+1} = Nx_k + b, \qquad \text{para todo } k \geq 0.$$

□

Para calcular la sucesión $(x_k)_{k\in\mathbb{N}}$, tenemos que resolver entonces en cada paso un sistema de la forma $Mx_{k+1} = \widetilde{b}$. Es decir, la solución de este último sistema tiene que ser muy económiceconómicoomparación con el sistema $Ax = b$ para que el método iterativo tenga sentido.

Definición 17. *El método iterativo de la Definición 16 se llama convergente si para cada punto inicial $x_0 \in \mathbb{R}^n$ la sucesión $(x_k)_{k\in\mathbb{N}}$ converge.* □

En la última definición no pedimos convergencia de $(x_k)_{k\in\mathbb{N}}$ a la solución exacta x. Eso no es necesario, pues, si la sucesión $(x_k)_{k\in\mathbb{N}}$ converge a un vector x, entonces $Mx = Nx + b$, es decir, $Ax = b$. Concluimos que el límite de $(x_k)_{k\in\mathbb{N}}$, si existe, es solución de nuestro problema.

Teorema 18. *El método iterativo de la Definición 16 converge si y solo si*

$$\rho(M^{-1}N) < 1.$$

La matriz $M^{-1}N$ se llama matriz de iteración del método.

Demostración. Por el Lema 4 existe una norma matricial $\|\cdot\|$ inducida por una norma vectorial que también vamos a anotar por $\|\cdot\|$, tal que $\|M^{-1}N\| < 1$. Sea x la solución de $Ax = b$, entonces $x = M^{-1}(Nx + b)$. Por lo tanto

$$x_k - x = M^{-1}Nx_{k-1} + M^{-1}b - M^{-1}(Nx + b) = M^{-1}N(x_{k-1} - x).$$

Concluimos que $x_k - x = (M^{-1}N)(x_0 - x)$, y por el Lema 2,

$$\|x_k - x\| \leq \|M^{-1}N\|^k \|x_0 - x\| \tag{1.6}$$

Dado que $\|M^{-1}N\| < 1$, concluimos que $\|x_k - x\| \to 0$. Dado que todas las normas sobre un espacio vectorial de dimension finita son equivalentes, concluimos el resultado. □

En lo que sigue vamos a descomponer la matriz $A \in \mathbb{R}^{n\times n}$ en los tres componentes

$$D = \begin{pmatrix} a_{11} & & \cdots & 0 \\ & \ddots & & \\ & & \ddots & \\ 0 & \cdots & & a_{nn} \end{pmatrix}, L = \begin{pmatrix} 0 & & \cdots & 0 \\ a_{21} & \ddots & & \\ \vdots & \ddots & \ddots & \\ a_{n1} & \cdots & a_{n,n-1} & 0 \end{pmatrix}, U = \begin{pmatrix} 0 & a_{12} & \cdots & a_{1n} \\ & \ddots & \ddots & \vdots \\ & & \ddots & a_{n-1,n} \\ 0 & \cdots & & 0 \end{pmatrix},$$

es decir $A = D + L + U$. Dos métodos iterativos básicos son

1. **Método de Jacobi:** En este método, $M = D$ y $N = -(L + U)$, es decir

   $$Dx_{k+1} = b - (L + U)x_k.$$

   La matriz de iteración es $-D^{-1}(L + U)$. El último sistema se resuleve con costo computacional n, pues, D es una matriz diagonal. De hecho,

   $$x_{k+1,\ell} = \frac{1}{a_{\ell\ell}}\Big(b_\ell - \sum_{\substack{j=1 \\ j\neq\ell}}^{n} a_{\ell j}x_{k,j}\Big).$$

2. **Método de Gauss-Seidel:** En este método, $M = (D + L)$, $N = -U$, es decir

   $$(D + L)x_{k+1} = b - Ux_k.$$

   La matriz de iteración es $-(D + L)^{-1}U$. En el último sistema se puede resuelve con costo computacional n^2, pues, la matriz $D + L$ es una matriz triangular inferior. De hecho, si reducimos la sustitución descendente, obtenemos

   $$x_{k+1,\ell} = \frac{1}{a_{\ell\ell}}\Big(b_\ell - \sum_{j=1}^{\ell-1} a_{\ell j}x_{k+1,j} - \sum_{j=\ell+1}^{n} a_{\ell j}x_{k,j}\Big). \tag{1.7}$$

1.2. Proyecciones sobre subespacios y mínimos cuadrados

Muchos métodos en ingeniería producen el problema de concordar ciertos parametros de un módelo con algunas mediciones. Eso produce el problema de *resolver* en algún sentido un sistema *sobredeterminado* $Ax = b$ con $A \in \mathbb{K}^{m\times n}$, $m \geq n$. Si $b \notin \operatorname{Im}(A)$ entonces no hay solución x clásica. La idea entonces es encontrar un $x \in \mathbb{K}^n$ tal que $b - Ax$ es *pequeño* en la norma euclidiana, es decir

$$\|b - Ax\|_2 = \min_{y\in\mathbb{K}^n} \|b - Ay\|_2. \tag{1.8}$$

Si $m = n$ y A es regular, entonces (1.8) tiene única solución $x = A^{-1}b$. Obviamente, una solución de (1.8) depende de la norma usada. Vamos a usar la norma euclidiana $\|\cdot\|_2$, pues en este caso es fácil resolver (1.8): la norma euclidiana es inducida por el producto interno $x^{\mathsf{H}}y$, y por lo tanto Ax es la proyección ortogonal de b en el subespacio $\operatorname{Im}(A)$, es decir $z^{\mathsf{H}}(b - Ax) = 0$ para todo $z \in \operatorname{Im}(A)$. Dado que $\operatorname{Im}(A)$ es el espacio generado por las columnas de A, obtenemos las *ecuaciones normales*

$$A^{\mathsf{H}}(b - Ax) = 0, \quad \text{o bien} \quad A^{\mathsf{H}}Ax = A^{\mathsf{H}}b. \tag{1.9}$$

Usar la norma euclidiana $\|\cdot\|_2$ en (1.8) es equivalente a minimizar una suma de cuadrados $\sum_{j=1}^{n} |b_j - (Ay)_j|^2$, y eso explica el nombre *mínimos cuadrados*.

Lema 19. *Una solución $x \in \mathbb{K}^n$ de (1.8) es solución de (1.9) y vice versa.*

Demostración. Primero demostramos que (1.8) implica (1.9). Si x es solución de (1.8), entonces para cada $z \in \mathbb{K}^n$ la función $p : \mathbb{R} \to \mathbb{R}$ dada por $p(t) = \|b - A(x + tz)\|_2^2$ tiene un mínimo local en $t = 0$. Calculamos

$$\begin{aligned} p(t) &= (b - A(x + tz))^{\mathsf{H}} (b - A(x + tz)) \\ &= b^{\mathsf{H}}b - 2\operatorname{Re} b^{\mathsf{H}}Ax + (Ax)^{\mathsf{H}}Ax + 2t \operatorname{Re}\Big((Az)^{\mathsf{H}}Ax - (Az)^{\mathsf{H}}b\Big) + t^2\Big((Az)^{\mathsf{H}}Az\Big) \end{aligned}$$

y concluimos que p es un polinomio de grado 2 y por lo tanto diferenciable. Dado que p tiene mínimo local en $t = 0$, concluimos que $p'(0) = 0$. De la formula arriba se concluye

$$\operatorname{Re}\Big(z^{\mathsf{H}}A^{\mathsf{H}}Ax - z^{\mathsf{H}}A^{\mathsf{H}}b\Big) = 0 \quad \text{para todo } z \in \mathbb{K}^n,$$

lo que implica (1.9). Por el otro lado, si x es solución de (1.9), entonces

$$(Az)^{\mathsf{H}}(b - Ax) = 0 \quad \text{para todo } z \in \mathbb{K}^n$$

y para cada $y \in \mathbb{K}^n$ podemos concluir

$$\|b - Ax\|_2^2 = (b - Ax)^{\mathsf{H}}(b - Ax) = b^{\mathsf{H}}(b - Ax) = (b - Ay)^{\mathsf{H}}(b - Ax) \leq \|b - Ay\|_2\|b - Ax\|_2,$$

lo que implica (1.8). □

Lema 20. *Problema* (1.9) *siempre tiene solución* $x \in \mathbb{K}^n$. *Si* $\ker(A) = \{0\}$, *entonces la solución es única.*

Demostración. Para demostrar la existencia, recordamos primero la identidad $\mathrm{Im}(A)^\perp = \ker(A^{\mathsf{H}})$,

$$\mathrm{Im}(A)^\perp = \left\{ y \in \mathbb{K}^m \mid \forall x \in \mathbb{K}^n \quad (Ax)^{\mathsf{H}} y \right\} = \left\{ y \in \mathbb{K}^m \mid \forall x \in \mathbb{K}^n \quad xA^{\mathsf{H}} y = 0 \right\} = \ker(A^{\mathsf{H}}),$$

y la descomposición ortogonal $\mathbb{K}^m = \mathrm{Im}(A) \oplus \mathrm{Im}(A)^\perp = \mathrm{Im}(A) \oplus \ker(A^{\mathsf{H}})$. Es decir, cada $b \in \mathbb{K}^m$ lo podemos escribir únicamente como $b = v + w$ von $v \in \mathrm{Im}(A)$ y $w \in \ker(A^{\mathsf{H}})$. Sea $x \in \mathbb{K}^n$ con $v = Ax$, entonces $A^{\mathsf{H}} b = A^{\mathsf{H}}(Ax + w) = A^{\mathsf{H}} Ax$.

Para ver la unicidad, supongamos que $A^{\mathsf{H}} Ax = 0$. Concluimos que

$$0 = x^{\mathsf{H}} A^{\mathsf{H}} Ax = (Ax)^{\mathsf{H}} Ax = \|Ax\|_2^2,$$

y por lo tanto $Ax = 0$. □

Como ya mencionamos, el problema (1.8) está relacionado con el cálculo de proyecciones ortogonales: Sea $\langle \mathcal{A} \rangle$ un subespacio de $\mathbb{R}^m$ y $\Pi_{\langle \mathcal{A} \rangle} : \mathbb{R}^m \to \langle \mathcal{A} \rangle$ la proyección ortogonal, es decir

$$\|b - \Pi_{\langle \mathcal{A} \rangle} b\|_2 = \min_{a \in \langle \mathcal{A} \rangle} \|b - a\|_2, \tag{1.10}$$

o bien $b - \Pi_{\langle \mathcal{A} \rangle} b \in \langle \mathcal{A} \rangle^\perp$. Si $\mathcal{A} = \{a_1, \ldots, a_n\}$ es una base de $\langle \mathcal{A} \rangle$ y

$$A := \begin{pmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{pmatrix},$$

entonces las soluciones x de (1.8) y $\Pi_{\langle \mathcal{A} \rangle} b$ de (1.10) están relacionados por $Ax = \Pi_{\langle \mathcal{A} \rangle} b$. Usando (1.9) obtenemos de inmediato lo siguiente.

Corolario 21. *Sea* $\mathcal{A} = \{a_1, \ldots, a_n\}$ *una base de un subespacio* $\langle \mathcal{A} \rangle$ *de* $\mathbb{K}^m$ *y*

$$A := \begin{pmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{pmatrix} \in \mathbb{K}^{m \times n}.$$

Si $\Pi_{\langle \mathcal{A} \rangle} : \mathbb{R}^m \to \langle \mathcal{A} \rangle$ *la proyección ortogonal sobre* $\langle \mathcal{A} \rangle$, *entonces*

$$\Pi_{\langle \mathcal{A} \rangle} b = A \left(A^{\mathsf{H}} A \right)^{-1} A^{\mathsf{H}} b.$$

Si $\mathcal{A}$ *es una base ortonormal, entonces*

$$\Pi_{\langle \mathcal{A} \rangle} b = A A^{\mathsf{H}} b = \sum_{\ell=1}^{n} \langle b, a_\ell \rangle a_\ell,$$

y además

$$I_m - \Pi_{\langle \mathcal{A} \rangle} = (I - \Pi_{\langle a_n \rangle}) \cdots (I - \Pi_{\langle a_1 \rangle}).$$

□

1.3. Ortogonalización de vectores y la factorización QR

Para trabajar con un subespacio de $\mathbb{R}^m$, lo tenemos que representar usando una base, $\mathcal{A} = \{a_1, \ldots, a_n\}$. La matriz que tiene los vectores de la base como columnas,

$$A := \begin{pmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{pmatrix} \in \mathbb{R}^{m\times n},$$

representa entonces el espacio $\langle \mathcal{A} \rangle$. Supongamos que $\mathcal{Q} = \{q_1, \ldots, q_n\}$ es una base ortonormal de $\langle \mathcal{A} \rangle$, es decir $q_j^\top q_k = \delta_{j,k}$, entonces $Q^\top Q = I_n$. En este caso, las coordenadas de $a \in \langle \mathcal{A} \rangle$ con respecto a la base $\mathcal{Q}$ están dadas simplemente por $Q^\top a$ (la inversa de una matriz ortogonal es su transpuesta). Concluimos que si queremos hacer ,$\omega linealamano, entoncesesmuytiltenerunabaseorthogonal, pue$ 1 por Lema 6 (en general $\mathrm{cond}(Q)_2 \approx 1$ por errores de redondeo). Nuestro primer objetivo será entonces transformar una base cualquiera $\mathcal{A}$ a una base ortonormal $\mathcal{Q}$. Aún mas: en muchas aplicaciones tenemos que trabajar sucesivamente con los espacios

$$\langle a_1 \rangle \subset \langle a_1, a_2 \rangle \subset \langle a_1, a_2, a_3 \rangle \subset \cdots \subset \langle a_1, \ldots, a_n \rangle.$$

Si ya tenemos una base ortonormal de $\langle a_1, \ldots a_k \rangle$, entonces será ideal reutilizarla para determinar una base orthonormal de $\langle a_1, \ldots a_{k+1} \rangle$, es decir

$$\langle a_1, \ldots a_k \rangle = \langle q_1, \ldots q_k \rangle, \qquad \text{para todo } 1 \le k \le n.$$

En terminos de matrices, esta última condición se lee

$$\begin{pmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{pmatrix} = \begin{pmatrix} | & & | \\ q_1 & \cdots & q_n \\ | & & | \end{pmatrix} \cdot \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix},$$

es decir $A = QR$ con $Q \in \mathbb{R}^{m\times n}$ con columnas ortonormales y $R \in \mathbb{R}^{n\times n}$ una matriz triangular superior.

Teorema 22 (Factorización QR reducida)**.** *Sea $A \in \mathbb{K}^{m\times n}$ con $m \ge n$ y $\mathrm{rango}(A) = n$. Entonces existe única factorización $A = QR$ con $Q \in \mathbb{K}^{m\times n}$, $Q^{\mathsf{H}}Q = I_n$, y $R \in \mathbb{K}^{n\times n}$ una matriz triangular superior, es decir $r_{j,k} = 0$ para todo $j > k$ y $r_{j,j} \in \mathbb{R}$ y $r_{j,j} > 0$.*

Demostración. Primero demostramos la existencia. La matriz $A^{\mathsf{H}}A \in \mathbb{K}^{n\times n}$ es hermitiana, y $x^{\mathsf{H}}A^{\mathsf{H}}Ax = (Ax)^{\mathsf{H}}Ax = \|Ax\|_2^2 \ge 0$. Además, dado que A tiene rango maximo, eso muestra que $x^{\mathsf{H}}A^{\mathsf{H}}Ax = 0$ si y solo si $x = 0$. Es decir, $A^{\mathsf{H}}A \in \mathbb{K}^{n\times n}$ es hermitiana y definida positiva. Por lo tanto, existe única factorización de Cholesky $A^{\mathsf{H}}A = LL^{\mathsf{H}}$ con L triangular inferior y elementos reales y positivos en la diagonal. La matrix $R := L^{\mathsf{H}}$ cumple con las condiciones del teorema y $A^{\mathsf{H}}A = R^{\mathsf{H}}R$. Dado que R es invertible, podemos definir $Q := AR^{-1} \in \mathbb{K}^{m\times n}$. Calculamos

$$Q^{\mathsf{H}}Q = (AR^{-1})^{\mathsf{H}}AR^{-1} = R^{-\mathsf{H}}A^{\mathsf{H}}AR^{-1} = R^{-\mathsf{H}}R^{\mathsf{H}}RR^{-1} = I_n.$$

Es fácil ver la unicidad de la factorización: Sea $A = Q_1R_1 = Q_2R_2$, entonces $A^{\mathsf{H}}A = R_1^{\mathsf{H}}R_1 = R_2^{\mathsf{H}}R_2$ son dos factorizaciones de Cholesky y por lo tanto $R_1 = R_2$. Eso implica $Q_1 = Q_2$. □

Si tenemos la factorización $A = QR$ para $A \in \mathbb{K}^{n\times n}$ invertible, entonces podemos resolver el sistema $Ax = b$ en dos pasos,

$$Qy = b,$$
$$Rx = y.$$

El primer paso es económico ($y = Q^{\mathsf{H}}b$) y tiene costo $\mathcal{O}(n^2)$. El segundo paso corresponde a un sistema con matriz triangular y también tiene costo $\mathcal{O}(n^2)$. Además, las condiciones de los dos pasos no son peores que la condición de A, pues según Lema 6 tenemos $\mathrm{cond}_2(Q) = 1$ y $\mathrm{cond}_2(R) = \mathrm{cond}_2(Q^{-1}A) = \mathrm{cond}_2(A)$. En otras palabras, resolver un sistema lineal con la factorización QR es estable como algoritmo. Comparamos eso con la factorización LU: Si $A = LU$, entonces podemos resolver el sistema $Ax = b$ en dos pasos

$$Ly = b,$$
$$Ux = y.$$

En general, las condiciones de L y/o U son mas altas que la condición de A. En otras palabras, resolver sistemas con la factorización LU es un algoritmo no estable. Para calcular una factorización QR (es decir, ortogonalizar una base), tenemos dos opciones que presentaremos en los dos capítulos siguientes.

1.3.1. Ortogonalización triangular

La primer opción es aplicar unas operaciones "triangulares" a las columnas de A para transformarla sucesivamente a una matriz ortogonal,

$$A \to AR_1 \to AR_1R_2 \to \cdots \to AR_1 \cdots R_n = Q,$$

donde los R_j son triangulares superiores y Q es ortogonal. En este caso, $A\widetilde{R} = Q$, donde $\widetilde{R} = R_1 \cdots R_n$ también es triangular superior como producto de matrices triangulares superiores. Finalmente, $R = \widetilde{R}^{-1}$ también es triangular superior. Eso corresponde entonces a una *ortogonalización triangular*, y ya conocemos un método para realizarla: la ortogonalización de Gram Schmidt: Empezamos con $q_1 = a_1/\|a_1\|_2$, y si tenemos $\{q_1, \ldots, q_{k-1}\}$, entonces q_k se calcula ortogonalizando a_k con respecto a $\langle q_1, \ldots, q_{k-1}\rangle$.

Lema 23 (Gram-Schmidt)**.** *Sea $\{a_1, \ldots, a_n\} \subset \mathbb{R}^m$ un conjunto de vectores linealmente independientes. Entonces existe un conjunto de vectores ortonormales $\{q_1, \ldots, q_n\} \subset \mathbb{R}^m$ tal que*

$$\langle a_1, \ldots a_k\rangle = \langle q_1, \ldots q_k\rangle, \qquad \textit{para todo } 1 \le k \le n.$$

Los q_k están dados por

$$q_k = \frac{\widetilde{q}_k}{\|\widetilde{q}_k\|_2}, \qquad \textit{donde} \qquad \widetilde{q}_k = \left(I - \Pi_{\langle q_1,\ldots,q_{k-1}\rangle}\right) a_k, \quad k = 1,\ldots,n. \tag{1.11}$$

□

Notamos que según Corolario 21,

$$\Pi_{\langle q_1,\ldots,q_{k-1}\rangle} a_k = \sum_{\ell=1}^{k-1} \langle a_k\,, q_\ell\rangle q_\ell, \tag{1.12}$$

asi que si usamos el método de Gram-Schmidt para calcular la factorizacón QR de una matriz A, entonces la matriz R está dada por

$$r_{kk} = \|\widetilde{q}_k\|_2, \qquad r_{\ell k} = \langle a_k\,, q_\ell\rangle,$$

y la matriz Q por $Q = (q_1,\ldots,q_n) \in \mathbb{R}^{m\times n}$. Para una primera implementación, usamos simplemente la formula (1.12).

Algorithm 1: Gram-Schmidt clásico

Input: $A = (a_1,\ldots,a_n) \in \mathbb{R}^{m\times n}$
for $k = 1$ **to** n **do**
  $q_k = a_k$
  for $\ell = 1$ **to** $k-1$ **do**
    $r_{\ell k} = \langle a_k\,, q_\ell\rangle$
    $q_k = q_k - r_{\ell k} q_\ell$
  end
  $r_{kk} = \|q_k\|_2$
  $q_k = q_k / r_{kk}$
end
Output: $Q = (q_1,\ldots,q_m) \in \mathbb{R}^{n\times m}$ columnas ortogonales,
$R \in \mathbb{R}^{m\times m}$ triangular superior

En el algoritmo de Gram-Schmidt clásico usamos (1.12). Los vectores q_ℓ, $\ell = 1,\ldots,k-1$ en esta formula ya son resultados de calculos anteriores. En aritmética punto flotante llevarán errores de redondeo y ya no serán exactamente ortonormales. Eso implica efectos de cancelación que implican que el algoritmo de Gram-Schmidt clásico es inestable. Como remedio, usamos en vez de (1.12) la siguiente identidad de Corolario 21

$$I_m - \Pi_{\langle q_1,\ldots,q_{k-1}\rangle} = (I_m - \Pi_{\langle q_{k-1}\rangle}) \cdots (I_m - \Pi_{\langle q_1\rangle}),$$

y asi llegamos al siguiente algoritmo.

Algorithm 2: Gram-Schmidt modificado

Input: $A = (a_1, \ldots, a_n) \in \mathbb{R}^{m\times n}$

for $k = 1$ **to** n **do**
  $q_k = a_k$
  for $\ell = 1$ **to** $k-1$ **do**
    $r_{\ell k} = \langle q_k \, , q_\ell \rangle$
    $q_k = q_k - r_{\ell k} q_\ell$
  end
  $r_{kk} = \|q_k\|_2$
  $q_k = q_k / r_{kk}$
end

Output: $Q = (q_1, \ldots, q_m) \in \mathbb{R}^{n\times m}$ columnas ortogonales,
$R \in \mathbb{R}^{m\times m}$ triangular superior

Notamos que en aritmetica exacta, los algoritmos 1 y 2 son equivalentes. Sin embargo, el algoritmo de Gram-Schmidt modificado es numericamente más estable. Con respecto al costo obtenemos lo siguiente.

Lema 24. *Algoritmos 1 y 2 calculan la factorización QR de una matriz $A \in \mathbb{K}^{m\times n}$ (ortogonalizan una base de un subespacio de $\mathbb{K}^m$ de dimension n) en $\mathcal{O}(mn^2)$ flops.*

1.3.2. Triangularización ortogonal

La segunda opción que tenemos para ortogonalizar una base, respectivamente calcular una factorización QR, es aplicar operaciones "ortogonales" a las filas de A para transformarla sucesivamente a una matriz triangular,

$$A \to Q_1 A \to Q_2 Q_1 A \to \cdots \to Q_n \cdots Q_1 A = R,$$

donde R es triangular superior y los Q_j son ortogonales. En este caso, $\widetilde{Q}A = R$, donde $\widetilde{Q} = Q_n \cdots Q_1$ también es ortogonal como producto de matrices ortogonales. Finalmente, $Q = Q^{\mathsf{H}}$ es ortogonal. Consideramos el paso k, es decir

$$A^{(k-1)} =: Q_{k-1} Q_{k-2} \cdots Q_1 A \to Q_k A^{(k-1)}.$$

La idea es que cada operación ortogonal Q_k genere zeros abajo de la diagonal en la k-ésima columna de A_{k-1}, mientras no afecte los zeros ya generados en los pasos anteriores. Concluimos que la Q_k tendrá que tener la forma

$$Q^{(k)} := \begin{pmatrix} I_{k-1} & 0 \\ 0 & \widetilde{Q}^{(k)} \end{pmatrix} \in \mathbb{K}^{m \times m},$$

donde $I_{k-1} \in \mathbb{K}^{(k-1)\times(k-1)}$ es la identidad y $\widetilde{Q}^{(k)} \in \mathbb{K}^{(m-k-1)\times(m-k-1)}$ es ortogonal y satisface

$$\widetilde{Q}^{(k)} a_k^{(k-1)} = \begin{pmatrix} \lambda \| a_k^{(k-1)} \|_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{1.13}$$

donde $a_k^{(k-1)}$ es la k-ésima columna de $A^{(k-1)}$, y $|\lambda| = 1$. Hay dos posibilidades de elegir $\widetilde{Q}^{(k)}$, o como una reflexión (reflexiones de Householder), o como una rotación (rotaciones de Givens). Solo presentaremos la primera variante.

Lema 25 (Reflexión de Householder)**.** *Sea $0 \neq w \in \mathbb{K}^n$. La reflexión de Householder $H_w \in \mathbb{K}^{n\times n}$, definida por*

$$H_w := I_n - 2\frac{ww^{\mathsf{H}}}{w^{\mathsf{H}}w} = I_n - 2\frac{ww^{\mathsf{H}}}{\|w\|_2^2},$$

tiene las siguientes propiedades:

(i) H_w es hermitiana $H_w^{\mathsf{H}} = H_w$,

(ii) H_w es involutiva, $H_w^2 = I_n$,

(iii) H_w es unitaria $H_w^{-1} = H_w^{\mathsf{H}}$.

Demostración. Propiedad (*iii*) es obviamente consecuencia de las primeras dos propiedades. Calculamos

$$H_w^{\mathsf{H}} = I_n - 2\frac{(ww^{\mathsf{H}})^{\mathsf{H}}}{\|w\|_2^2} = I_n - 2\frac{ww^{\mathsf{H}}}{\|w\|_2^2} = H_w.$$

Además, dado que $(ww^{\mathsf{H}})(ww^{\mathsf{H}}) = ww^{\mathsf{H}}\|w\|_2^2$, obtenemos

$$H_w^2 = H_w H_w = I_n - 4\frac{ww^{\mathsf{H}}}{\|w\|_2^2} + 4\frac{(ww^{\mathsf{H}})(ww^{\mathsf{H}})}{\|w\|_2^2\|w\|_2^2} = I_n.$$

□

La transformación H_w realiza una reflexión por el hiperplano $P_w = \{x \in \mathbb{K}^n \mid x^{\mathsf{H}}w = 0\}$: si $x \in P_w$, entonces $H_w x = x$. Si $x = \langle w \rangle$, es decir, $x = cw$ para algún $c \in \mathbb{K}$, entonces

$$H_w x = cw - 2\frac{ww^{\mathsf{H}}w}{\|w\|_2^2}c = cw - 2cw = -cw.$$

Concluimos entonces que bajo ciertas condiciones, dos vectores x, y pueden transformarse el uno al otro usando una reflexión de Householder, usando $w = x - y$.

Lema 26. *Sean $x \neq y \in \mathbb{K}^n$ con $\|x\|_2 = \|y\|_2$ y $y^{\mathsf{H}}x \in \mathbb{R}$. Sea $w = x - y$. Entonces $H_w x = y$.*

Demostración. Notamos $x^{\mathsf{H}}x = y^{\mathsf{H}}y$ y $\operatorname{Re}(y^{\mathsf{H}}x) = y^{\mathsf{H}}x$. Calculamos

$$w^{\mathsf{H}}w = (x-y)^{\mathsf{H}}(x-y) = 2x^{\mathsf{H}}x - 2\operatorname{Re}(y^{\mathsf{H}}x) = 2w^{\mathsf{H}}x,$$

y concluimos

$$H_w x = x - 2\frac{w^{\mathsf{H}}x}{w^{\mathsf{H}}w}w = x - w = y.$$

□

En el paso k del método de Householder hay que realizar un producto de la forma $H_w B$ con $H_w \in \mathbb{K}^{(m-k-1)\times(m-k-1)}$, $B \in \mathbb{K}^{(n-k-1)\times(n-k-1)}$. En vez de calcular H_w explicitamente, es suficiente almacenar w y usar la identidad $H_w B = B - \frac{2}{\|w\|_2^2}ww^{\mathsf{H}}B$.

Para obtener la propiedad (1.13) tenemos que seleccionar en Lema 26

$$y = \begin{pmatrix} \lambda\|x\|_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad \text{es decir} \qquad w = \begin{pmatrix} x_1 - \lambda\|x\|_2 \\ x_2 \\ \vdots \\ x_n \end{pmatrix},$$

con $|\lambda| = 1$. En $\mathbb{R}^n$ hay dos posibilidades $\lambda = \pm 1$. Para evitar efectos de cancelación en el caso $x_2, \ldots, x_n \approx 0$, o sea $|x_1| \approx \|x\|_2$, se elige $\lambda = -\operatorname{signo}(x_1)$. En $\mathbb{C}^n$ hay una infinita cantidad de posibilidades, y el siguiente resultado indica la mejor opción.

Lema 27. *Sea $x \in \mathbb{K}^n \setminus \langle e_1 \rangle$, $\lambda \in \mathbb{K}$ con $|\lambda| = 1$ y $\lambda\overline{x_1} = |x_1|$. Definimos*

$$w := \frac{x + \sigma e_1}{\|x + \sigma e_1\|_2}, \quad \sigma := \lambda \|x\|_2,$$

entonces $\|w\|_2 = 1$ y $H_w x = -\sigma e_1$.

Demostración. Notamos que $\sigma x^{\mathsf{H}} e_1 = \sigma\overline{x_1} = \lambda\overline{x_1}\|x\|_2 = |x_1|\|x\|_2 \in \mathbb{R}$. Calculamos

$$\|x + \sigma e_1\|_2^2 = x^{\mathsf{H}}x + 2\,\mathrm{Re}(\sigma x^{\mathsf{H}} e_1) + |\sigma|^2 = x^{\mathsf{H}}x + 2\sigma x^{\mathsf{H}} e_1 + |\lambda|^2 x^{\mathsf{H}} x = 2x^{\mathsf{H}}\left(x + \sigma e_1\right).$$

Concluimos que $2x^{\mathsf{H}} w = \|x + \sigma e_1\|_2 \in \mathbb{R}$, es decir $2x^{\mathsf{H}} w = 2w^{\mathsf{H}} x$. Finalmente,

$$2ww^{\mathsf{H}} x = \|x + \sigma e_1\|_2 w = x + \sigma e_1,$$

lo que significa $H_w x = -\sigma e_1$. □

Ahora tenemos todas las herramientas para construir la factorización QR usando el método de Householder. Sea $A \in \mathbb{K}^{m\times n}$, $m \geq n$.

Paso 1: Si la primera columna $a_1 \in \mathbb{K}^m$ de A es un multiple de $e_1 \in \mathbb{K}^m$, entonces definimos $H := I_m$. Si no, elegimos una reflexión de Householder $H \in \mathbb{K}^{m\times m}$ según Lema 27 tal que $Ha_1 \in \langle e_1 \rangle$. Si definimos $Q_1 := H \in \mathbb{K}^{m\times m}$, entonces

$$A^{(1)} := Q_1 A = \begin{pmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & \dots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(1)} & \dots & a_{2,n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m,2}^{(1)} & \dots & a_{m,n}^{(1)} \end{pmatrix}.$$

Paso 2: Consideramos la matiz $B \in \mathbb{K}^{(m-1)\times(n-1)}$ dada por

$$B := \begin{pmatrix} a_{2,2}^{(1)} & \dots & a_{2,n}^{(1)} \\ \vdots & & \vdots \\ a_{m,2}^{(1)} & \dots & a_{m,n}^{(1)} \end{pmatrix}.$$

Si la primera columna b_1 de B es un multiple de $e_1 \in \mathbb{K}^{m-1}$, entonces definimos $H := I_{m-1}$. Si no, elegimos una reflexión de Householder $H \in \mathbb{K}^{(m-1)\times(m-1)}$ según Lema 27 tal que $Hb_1 \in \langle e_1 \rangle$. Definimos la matriz

$$Q_2 := \begin{pmatrix} 1 & 0 \\ 0 & H \end{pmatrix} \in \mathbb{K}^{m\times m},$$

entonces

$$A^{(2)} := Q_2 A^{(1)} = \begin{pmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & a_{1,3}^{(1)} & \dots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & a_{2,3}^{(2)} & \dots & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(2)} & \dots & a_{3,n}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{m,3}^{(2)} & \dots & a_{m,n}^{(2)} \end{pmatrix}.$$

Paso k: Consideramos la matriz $B \in K^{(m-k+1)\times(n-k+1)}$ dada por

$$B = \begin{pmatrix} a_{k,k}^{(k-1)} & \dots & a_{k,n}^{(k-1)} \\ \vdots & & \vdots \\ a_{m,k}^{(k-1)} & \dots & a_{m,n}^{(k-1)} \end{pmatrix}.$$

Si la primera columna b_1 de B es un multiple de $e_1 \in \mathbb{K}^{m-k+1}$, entonces definimos $H := I_{m-k+1}$. Si no, elegimos una reflexión de Householder $H \in \mathbb{K}^{(m-k+1)\times(m-k+1)}$ según Lema 27 tal que $Hb_1 \in \langle e_1 \rangle$. Definimos la matriz

$$Q_k := \begin{pmatrix} I_{k-1} & 0 \\ 0 & H \end{pmatrix} \in \mathbb{K}^{m\times m},$$

y $A^{(k)} := Q_k A^{(k-1)} \in \mathbb{K}^{m\times n}$.

Después de n pasos obtenemos

$$R := A^{(n)} = Q_n \cdots Q_1 A,$$

donde $R \in \mathbb{K}^{m\times n}$ es una matriz triangular superior extendida y $Q^{\mathsf{H}} := Q_n \cdots Q_1 \in \mathbb{K}^{m\times m}$, como producto de matrices unitarias, es unitaria. Obtenemos $A = QR$.

Comentario 28. *Podemos hacer un par de observaciones.*

1. *En el caso de una matriz cuadrada $A \in \mathbb{K}^{n\times n}$, el método de Housholder necesita solamente $n-1$ pasos.*
2. *En el paso k del método de Householder hay que realizar un producto de la forma $HB = B - 2ww^{\mathsf{H}}B$. No es necesario calcular $H \in \mathbb{K}^{(m-k+1)\times(m-k+1)}$ explicitamente, los productos se calculan como $HB = B - \frac{2}{\|w\|_2^2} ww^{\mathsf{H}}B$.*
3. *La factorización QR con el método de Householder se puede calcular in situ, es decir, usando solamente la memoria de A. Se sobreescribe la matriz A de la siguiente manera: En la parte triangular superior de A se guarda R. Quedan espacio $\mathbb{K}^{m-k}$ en cada columna k para guardar información sobre H_w. Como explicamos en el punto 2., la única información que se necesita para aplicar H_w es el vector $w \in \mathbb{K}^{m-k+1}$. Observamos que no importa la longitud de w, entonces podemos exigir $w_1 = 1$.*

TBC: Costo de QR con Householder.

1.4. Problemas de equilibrio - mínimos cuadrados

Usando la representación (1.9) para calcular la solución de (1.8) no es muy recomendable, porque el factor de condición se eleva al cuadrado, como demostrarémos en el próximo resultado.

Lema 29. *Sea $A \in \mathbb{K}^{n\times n}$ invertible. Entonces*

$$\operatorname{cond}(A^{\mathsf{H}}A) = \operatorname{cond}(A)^2.$$

Demostración. Notamos que

$$\|A\|_2^2 = \max_{\|x\|_2=1} \|Ax\|_2^2 = \max_{\|x\|_2=1} x^{\mathsf{H}}A^{\mathsf{H}}Ax = \lambda_{\text{máx}}(A^{\mathsf{H}}A),$$

donde $\lambda_{\text{máx}}$ denota el autovalor mas grande (esta última igualdad la vamos a demostrar en el laboratorio). De manera análoga,

$$\|A^{-1}\|_2^2 = \frac{1}{\text{mín}_{\|x\|_2=1} \|Ax\|_2^2} = \frac{1}{\text{mín}_{\|x\|_2=1} x^{\mathsf{H}}A^{\mathsf{H}}Ax} = \frac{1}{\lambda_{\text{mín}}(A^{\mathsf{H}}A)}$$

Es decir,

$$\operatorname{cond}(A)^2 = \frac{\lambda_{\text{máx}}(A^{\mathsf{H}}A)}{\lambda_{\text{mín}}(A^{\mathsf{H}}A)}.$$

Notamos que si $\lambda \in \mathbb{R}$ es autovalor de una matriz hermitiana B, entonces λ^2 es autovalor de $B^{\mathsf{H}}B$. La matriz $B := A^{\mathsf{H}}A$ es hermitiana y definida positiva, y podemos concluir

$$\lambda_{\text{máx}}(A^{\mathsf{H}}A) = \lambda_{\text{máx}}(B) = \lambda_{\text{máx}}(B^{\mathsf{H}}B)^{1/2} = \max_{\|x\|_2=1} \|Bx\|_2$$

y

$$\lambda_{\text{mín}}(A^{\mathsf{H}}A) = \lambda_{\text{mín}}(B) = \lambda_{\text{mín}}(B^{\mathsf{H}}B)^{1/2} = \min_{\|x\|_2=1} \|Bx\|_2.$$

□

Lema 30. *Sea $m \geq n$ y $A \in \mathbb{K}^{m\times n}$ con rango máximo, y $b \in \mathbb{K}^m$. Sea $Q \in \mathbb{K}^{m\times m}$ unitaria y $R \in \mathbb{K}^{n\times n}$ triangular superior, y*

$$QA = \begin{pmatrix} R \\ 0 \end{pmatrix}.$$

Sea

$$Qb = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

con $b_1 \in \mathbb{K}^n$, $b_2 \in \mathbb{K}^{m-n}$. Entonces, la única solución $x \in \mathbb{K}^n$ de los problemas (1.8)*, respectivamente* (1.9)*, es dada por*

$$Rx = b_1.$$

Demostración. Por las condiciones de A sabemos que R es invertible. Usando $Q^{\mathsf{H}}Q = I_m$ calculamos

$$\|b - Ax\|_2 = \|Q^{\mathsf{H}}(Qb - QAx)\|_2 = \|Qb - QAx\|_2.$$

Observamos

$$Qb - QAx = \begin{pmatrix} b_1 - Rx \\ b_2 \end{pmatrix},$$

y concluimos el resultado. □

1.5. Descomposición en valores singulares

Un resultado fundamental de álgebra lineal dice que si $B \in \mathbb{K}^{n\times n}$ es hermitiana, entonces sus autovalores son reales y existe una base ortogonal de $\mathbb{K}^n$ de autovectores de B. En otras palabras, existe una matriz unitaria $Q \in \mathbb{K}^n$ tal que $B = QDQ^{\mathsf{H}}$, donde la matriz diagonal D contiene los autovalores de B. Se dice que B es *unitariamente diagonalizable*. Una interpretación geometrica de este concepto es que la imagen de la es esfera unitaria de $\mathbb{K}^n$ bajo la transformación B es es una *hiperelipse*. De hecho, la transformación unitaria Q^{H} mantiene la esfera unitaria, la transformación D estira la esfera unitaria a una hiperelipse alineada con la base canonica de $\mathbb{K}^n$, y la tranformación unitaria Q mantiene la hiperelipse. La descomposición en valores singulares (SVD por sus siglas *singular value decomposition* en inglés) es una extension de esta idea.

Teorema 31 (Full SVD)**.** *Sea $A \in \mathbb{K}^{m\times n}$. Entonces existen matrices unitarias $U \in \mathbb{K}^{m\times m}$, $V \in \mathbb{K}^{n\times n}$, y una matriz diagonal generalizada $\Sigma \in \mathbb{R}^{m\times n}$, es decir $\Sigma_{j,k} = \sigma_j \delta_{j,k}$ tal que $A = U\Sigma V^{\mathsf{H}}$, y $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\text{mín}(m,n)} \geq 0$. Los números σ_j se llaman valores singulares de A.*

(1) La matriz Σ es única y σ_j^2 es autovalor de $A^{\mathsf{H}}A$.

(2) $\|A\|_2 = \sigma_1$.

(3) Si $\text{rango}(A) = r$*, entonces $\sigma_1 \geq \cdots \geq \sigma_r > 0 = \sigma_{r+1} = \cdots = \sigma_{\text{mín}(m,n)}$.*

Demostración. La matriz $A^{\mathsf{H}}A \in \mathbb{K}^{n\times n}$ es hermitiana, por lo tanto existe una base ortogonal de $\mathbb{K}^n$ de autovectores $v_1, \ldots, v_n$ de $A^{\mathsf{H}}A$, y los autovalores μ_j (asociado a v_j) son reales y no negativos. Sin pérdida de generalidad acordamos $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_r > 0 = \mu_{r+1} = \cdots = \mu_n$, donde $r = \text{rango}(A^{\mathsf{H}}A) = \text{rango}(A)$. Definimos $\sigma_j := \sqrt{\mu_j}$ y

$$S := \begin{pmatrix} \sigma_1 & 0 & \ldots & 0 \\ 0 & \sigma_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \sigma_r \end{pmatrix}.$$

Define $V_1 = (v_1, \dots v_r) \in \mathbb{K}^{n\times r}$ y $V_2 = (v_{r+1}, \dots, v_n) \in \mathbb{K}^{n\times(n-r)}$, y $V = (V_1, V_2) \in \mathbb{K}^{n\times n}$. Entonces S es regular y $A^{\mathsf{H}}AV_1 = V_1S^2$. Si definimos $U_1 = AV_1S^{-1} \in \mathbb{K}^{m\times r}$, entonces

$$U_1^{\mathsf{H}}U_1 = S^{-1}V_1^{\mathsf{H}}A^{\mathsf{H}}AV_1S^{-1} = S^{-1}\underbrace{V_1^{\mathsf{H}}V_1}_{I_r}S = I_r.$$

Concluimos que las columnas de U_1 son ortonormales, y las podemos completar a matriz ortonormal $U = (U_1, U_2) \in \mathbb{K}^{m\times m}$. Falta demostrar que $\Sigma = U^{\mathsf{H}}AV$, o, en terminos de sus bloques,

$$\Sigma = \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix} = U^{\mathsf{H}}AV = \begin{pmatrix} U_1^{\mathsf{H}} \\ U_2^{\mathsf{H}} \end{pmatrix} A \begin{pmatrix} V_1 & V_2 \end{pmatrix} = \begin{pmatrix} U_1^{\mathsf{H}}AV_1 & U_1^{\mathsf{H}}AV_2 \\ U_2^{\mathsf{H}}AV_1 & U_2^{\mathsf{H}}AV_2 \end{pmatrix}.$$

Por la definición de U_1 tenemos $U_1^{\mathsf{H}}AV_1 = S$. Ádemas, las columnas de U son ortonormales y por lo tanto $U_2^{\mathsf{H}}AV_1 = U_2^{\mathsf{H}}U_1S = 0$. Las columnas de V_2 son autovectores de $A^{\mathsf{H}}A$ con autovalor 0, es decir $A^{\mathsf{H}}AV_2 = 0$. Eso implica $V_2^{\mathsf{H}}A^{\mathsf{H}}AV_2 = 0$, y por lo tanto $AV_2 = 0$.

Además, notamos que $\|A\|_2 = \rho(A^{\mathsf{H}}A)^{1/2} = \sigma_1$.

Falta demostrar que la matriz Σ es única. Dado que $A^{\mathsf{H}}A = V\Sigma^{\top}\Sigma V^{-1}$, concluimos que $A^{\mathsf{H}}A$ y $\Sigma^{\top}\Sigma$ tienen los mismos autovalores. La matriz $\Sigma^{\top}\Sigma$ es diagonal con elementos $\sigma_1^2, \dots, \sigma_n^2$ con $\sigma_j = 0$ para $j > \text{mín}(m, n)$. Por la condición $\sigma_1 \geq \cdots \geq \sigma_{\text{mín}(m,n)}$ determina Σ. □

Según el última resultado, la imagen de la esfera unitaria en $\mathbb{K}^n$ bajo *cualquier* transformación lineal $A \in \mathbb{K}^{m\times n}$ es una hiperelipse. Dado que $AV = U\Sigma$, observamos que $\sigma_1u_1, \dots, \sigma_ru_r$ (u_j las columnas de U) son los ejes principales de la hiperelipse, y $Av_j = \sigma_ju_j$.

La factorización SVD nos permite definir una *pseudoinversa* para cualquier matriz $A \in \mathbb{K}^{m\times n}$.

Teorema 32 (Pseudoinversa de *Moore-Penrose*)**.** *(1) Para cualquier matriz $A \in \mathbb{K}^{m\times n}$ existe única matriz $A^+ \in \mathbb{K}^{n\times m}$ con las propiedades siguientes:*

a) $A^+A = (A^+A)^{\mathsf{H}}$,
b) $AA^+ = (AA^+)^{\mathsf{H}}$,
c) $AA^+A = A$,
d) $A^+AA^+ = A^+$.

La matriz A^+ se llama pseudoinversa o inversa de Moore-Penrose de A.

(2) Si $A \in \mathbb{K}^{n\times n}$ regular, entonces $A^+ = A^{-1}$.

(3) Si $\Sigma \in \mathbb{K}^{m\times n}$ es una matriz diagonal generalizada $\Sigma_{j,k} = \sigma_j\delta_{j,k}$, entonces Σ^+ es diagonal generalizada y

$$\Sigma^+_{j,k} = \tau_j\delta_{j,k} \quad con \quad \tau_j := \begin{cases} \sigma_j^{-1} & si\ \sigma_j \neq 0, \\ 0 & si\ \sigma_j = 0. \end{cases}$$

(4) Si $A = U\Sigma V^{\mathsf{H}}$ una descomposición SVD, entonces $A^+ = V\Sigma^+ U^{\mathsf{H}}$.

Demostración. Si $A = \Sigma$ es una matriz diagonal generalizada y definimos Σ^+ a traves de (3), entonces se cumplen (a)–(b). Dado que U y V son unitarias, concluimos que para A una matriz general y A^+ definida por (4) se cumplen (a)–(b). Eso muestra la existencia. Para demostrar la unicidad, sean B, C dos matrices que satisfacen (a)–(b). Entonces

$$\begin{aligned} B &= BAB = BACAB = BACACACAB = (BA)^{\mathsf{H}}(CA)^{\mathsf{H}}C(AC)^{\mathsf{H}}(AB)^{\mathsf{H}} \\ &= (ABA)^{\mathsf{H}}C^{\mathsf{H}}CC^{\mathsf{H}}(ABA)^{\mathsf{H}} = A^{\mathsf{H}}C^{\mathsf{H}}CC^{\mathsf{H}}A^{\mathsf{H}} = (CA)^{\mathsf{H}}C(AC)^{\mathsf{H}} \\ &= CACAC = CAC = C. \end{aligned}$$

Si A es regular, obviamente $A^+ = A^{-1}$ porque A^{-1} ya satisface (a)–(b). □

Para una matriz $A \in \mathbb{K}^{n\times n}$ invertible tenemos $A^{-1} = (A^{\mathsf{H}}A)^{-1}A^{\mathsf{H}}$. Para la pseudoinversa, tenemos lo siguiente.

Lema 33. *Sea $A \in \mathbb{K}^{m\times n}$ con $m \geq n$ y rango máximo. Entonces $A^+ = \left(A^{\mathsf{H}}A\right)^{-1} A^{\mathsf{H}}$.*

Demostración. Dado que A tiene rango máximo, la matriz $\left(A^{\mathsf{H}}A\right)^{-1} A^{\mathsf{H}}$ está bien definida. Es fácil verificar los puntos (a)–(b) del Teorema 32. □

Si comparamos el último lema con las ecuaciones normales (1.9), entonces concluimos que A^+b es la solución única del problema de equilibrio. En el caso en que A no tiene rango máximo, no hay única solución, pero A^+b tiene un rol destacado.

Lema 34. *Para $A \in \mathbb{K}^{m\times n}$ y $b \in \mathbb{K}^m$ sea*

$$\mathcal{A} = \left\{ x \in \mathbb{K}^n \mid \|b - Ax\|_2 = \min_{y\in\mathbb{K}^n} \|b - Ay\|_2 \right\}$$

el conjunto de todas las soluciones del problema de equilibrio (1.8)*. Entonces, existe única solución $x \in \mathcal{A}$ con*

$$\|x\|_2 = \min_{y\in\mathcal{A}} \|y\|_2,$$

la solución con norma minima. Se tiene $x = A^+b$.

Demostración. Sea $A = U\Sigma V^{\mathsf{H}}$ una descomposición en valores singulares, entonces

$$\|b - Ax\|_2 = \|U^{\mathsf{H}}b - \Sigma V^{\mathsf{H}}x\|_2 = \|U^{\mathsf{H}}b - \Sigma z\|_2$$

El vector x minimiza el lado izquierdo si y solo si $z = V^{\mathsf{H}}x$ minimiza el lado derecho. Si x tiene norma minima dentro de todas las minimizadores del lado izquierdo, entonces z tiene norma minima dentro de todas las minimizadores del lado derecho, dado que V^{H} es unitaria.

Concluimos que es suficiente demostrar que el lado derecho tiene único minimizador z con norma minima. Escribimos con $r = \text{rango}(A)$ y u_j las columnas de U

$$\|U^{\mathsf{H}}b - \Sigma z\|_2^2 = \sum_{j=1}^{r} |u_j^{\mathsf{H}}b - \sigma_j z_j|^2 + \sum_{j=r+1}^{m} |u_j^{\mathsf{H}}b|^2.$$

Obviamente, un minimizador $z \in \mathbb{K}^n$ satisface $z_j = \sigma_j^{-1} u_j^{\mathsf{H}} b$ para $0 \leq j \leq r$. El único minimizador con norma minima es obviamente

$$z_j = \begin{cases} \sigma_j^{-1} u_j^{\mathsf{H}} b & 0 \leq j \leq r, \\ 0 & r < j. \end{cases}.$$

Finalemente calculamos

$$x = Vz = \sum_{j=1}^{r} \sigma_j^{-1} v_j u_j^{\mathsf{H}} b = V\Sigma^{+}U^{\mathsf{H}}b = A^{+}b.$$

□

Teorema 35 (Rayleigh)**.** *Sea $B \in \mathbb{K}^{n\times n}$ hermitiana con autovalores $\lambda_1 \geq \cdots \geq \lambda_n$ y autovectores ortonormales $v_1, \ldots, v_n$. Entonces*

$$\begin{aligned} \lambda_j &= \min_{\substack{x\in\langle v_1,\ldots,v_j\rangle \\ \|x\|_2=1}} x^{\mathsf{H}}Bx = \min_{\substack{x\perp\langle v_{j+1},\ldots,v_n\rangle \\ \|x\|_2=1}} x^{\mathsf{H}}Bx \\ &= \max_{\substack{x\in\langle v_j,\ldots,v_n\rangle \\ \|x\|_2=1}} x^{\mathsf{H}}Bx = \max_{\substack{x\perp\langle v_1,\ldots,v_{j-1}\rangle \\ \|x\|_2=1}} x^{\mathsf{H}}Bx. \end{aligned}$$

Demostración. Vamos a demostrar la desigualdad con los dos minimos, el caso de los maximos sigue el mismo argumento. Dado que $v_1, \ldots, v_n$ es una base ortonormal de $\mathbb{K}^n$, concluimos que los dos minimos son iguales. Un vector $x \in \mathbb{K}^n$ tiene la representación $x = \sum_{i=1}^{n} x_i w_i$, y $\|x\|_2^2 = x^{\mathsf{H}}x = \sum_{i=1}^{n} |x_i|^2$. Si $x \in \langle v_1, \ldots, v_j\rangle$, entonces $x = \sum_{i=1}^{j} x_i v_i$ y

$$x^{\mathsf{H}}Bx = \sum_{i=1}^{j}\sum_{k=1}^{j} \overline{x_i} x_k v_i^{\mathsf{H}} B v_k = \sum_{i=1}^{j} |x_i|^2 \lambda_i \geq \lambda_j \sum_{i=1}^{j} |x_i|^2.$$

Bdemás, $v_j^{\mathsf{H}} B v_j = \lambda_j$. □

En el último resultado, encontramos λ_j minimizando sobre el subespacio $\langle v_1, \ldots, v_j\rangle$. De hecho tenemos lo siguiente.

Teorema 36 (Courant-Fischer)**.** *Sea $B \in \mathbb{K}^{n\times n}$ hermitiana con autovalores $\lambda_1 \geq \cdots \geq \lambda_n$. Entonces*

$$\lambda_j = \max_{\substack{U\subset\mathbb{K}^n \text{ subespacio} \\ \dim(U)\geq j}} \min_{\substack{x\in U \\ \|x\|_2=1}} x^{\mathsf{H}}Bx = \min_{\substack{U\subset\mathbb{K}^n \text{ subespacio} \\ \dim(U)\leq n-j+1}} \max_{\substack{x\in U \\ \|x\|_2=1}} x^{\mathsf{H}}Bx.$$

Demostración. Vamos a demostrar la primer igualdad, la segunda sigue el mismo argumento. Primero, usando el Teorema 35, tenemos

$$\lambda_j = \min_{\substack{x\in\langle v_1,\dots,v_j\rangle \\ \|x\|_2=1}} x^{\mathsf{H}}Bx \leq \max_{\substack{U\subset\mathbb{K}^n \text{ subespacio} \\ \dim(U)\geq j}} \min_{\substack{x\in U \\ \|x\|_2=1}} x^{\mathsf{H}}Bx.$$

Segundo, el espacio $\langle v_j,\dots,v_n\rangle$ tiene dimension $n-j+1$, mientras cada subespacio U que consideramos en nuestro máximo tiene dimension mayor o igual que j. Entonces,

$$\begin{aligned}\dim\left(\langle v_j,\dots,v_n\rangle \cap U\right) &= \dim\left(\langle v_j,\dots,v_n\rangle\right) + \dim(U) - \dim\left(\langle v_j,\dots,v_n\rangle + U\right) \\ &\geq n-j+1+j-n = 1.\end{aligned}$$

Es decir, para cada $U\subset\mathbb{K}^n$ un subespacio de dimension j existe un $x\in\langle v_j,\dots,v_n\rangle\cap U$ con $x\neq 0$. Tal x tiene la representación $x=\sum_{i=j}^n x_j v_j$, y como antes obtenemos

$$x^{\mathsf{H}}Bx = \sum_{i=j}^n\sum_{k=j}^n \overline{x_i}x_k v_i^{\mathsf{H}}Bv_k = \sum_{i=j}^n |x_i|^2\lambda_i \leq \lambda_j\sum_{i=j}^n |x_i|^2.$$

Por lo tanto, para cada subespacio $U\subset\mathbb{K}^n$ de dimension j,

$$\min_{\substack{x\in U \\ \|x\|_2=1}} x^{\mathsf{H}}Bx \leq \lambda_j.$$

□

Si $A=U\Sigma V^{\mathsf{H}}$ es una descomposición en valores singulares y u_j y v_j son las columnas de U y V, entonces

$$A = U\Sigma V^{\mathsf{H}} = \begin{pmatrix} | & & | \\ u_1 & \dots & u_m \\ | & & | \end{pmatrix}\begin{pmatrix} \sigma_1 & 0 & \dots \\ 0 & \sigma_2 & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}\begin{pmatrix} - & v_1^{\mathsf{H}} & - \\ & \vdots & \\ - & v_n^{\mathsf{H}} & - \end{pmatrix} = \sum_{j=1}^{\text{rango}(A)} \sigma_j u_j v_j^{\mathsf{H}}.$$

Es decir, A es una suma de rango(A) matrices de rango 1. Si cortamos la suma despues de r terminos, obtenemos entonces una aproximación a la matriz A por una matriz de rango r. El próximo resultado nos permite controlar el error.

Lema 37 (Aproximación de rango bajo)**.** *Sea $A\in\mathbb{K}^{m\times n}$, $r\in\mathbb{N}_0$, y $\sigma_1\geq\cdots\geq\sigma_{\min(m,n)}$ los valores singulares de A.*

(1) Si $R\in\mathbb{K}^{m\times n}$ con rango$(R)\leq r$*, entonces*

$$\|A-R\|_2 \geq \sigma_{r+1}.$$

(2) Si $A = U\Sigma V^{\mathsf{H}}$ es una descomposición en valores singulares y $\Sigma^{(r)} \in \mathbb{K}^{m\times n}$ se define por

$$\Sigma^{(r)}_{j,k} := \begin{cases} \sigma_j & j = k \leq \text{mín}(r, m, n), \\ 0 & \textit{de no ser así}, \end{cases}$$

entonces la matriz $R := U\Sigma^{(r)}V^{\mathsf{H}}$ satisface

$$\|A - R\|_2 = \sigma_{r+1}.$$

Demostración. (1) Sean $\tilde{\lambda}_1 \geq \cdots \geq \tilde{\lambda}_n \geq 0$ los autovalores de $A^{\mathsf{H}}A$, es decir, $\sigma_j^2 = \tilde{\lambda}_j$. Sea $r' := \text{rango}(R) \leq r$, entonces $\dim \ker(R) = \dim \mathbb{K}^n - \text{rango}(R) = n - r'$, y el Teorema 36 implica

$$\begin{aligned} \tilde{\lambda}_{r'+1} &\leq \max_{\substack{x\in\ker(R)\\ \|x\|_2=1}} x^{\mathsf{H}}A^{\mathsf{H}}Ax = \max_{\substack{x\in\ker(R)\\ \|x\|_2=1}} (Ax)^{\mathsf{H}}Ax = \max_{\substack{x\in\ker(R)\\ \|x\|_2=1}} ((A-R)x)^{\mathsf{H}}(A-R)x \\ &\leq \max_{\substack{x\in\mathbb{K}^n\\ \|x\|_2=1}} x^{\mathsf{H}}(A-R)^{\mathsf{H}}(A-R)x = \|A-R\|_2^2 \end{aligned}$$

Concluimos que

$$\|A - R\|_2 \geq \sigma_{r'+1} \geq \sigma_{r+1}.$$

(2) Notamos que $A - R = U(\Sigma - \Sigma^{(r)})V^{\mathsf{H}}$, y por lo tanto el valor singular mas grande de $A - R$ es σ_{r+1}.

□

1.6. Métodos Krylov

Para resolver el sistema lineal $Ax = b$ con $A \in \mathbb{R}^{n\times n}$ hemos visto dos tipos de métodos: métodos directos como eliminación de Gauss, y métodos iterativos como Jacobi, Gauss-Seidel y SOR. Métodos directos determinan la solución exacta en una finita cantidad de operaciones (por lo menos en aritmética exacta), pero los resultados intermedios no llevan ninguna información adicional. Por otro lado, métodos iterativos en general nunca terminan, pero cada resultado intermedio es una aproximación a la solución exacta. Nos podemos preguntar si existen métodos que combinan los ventajas de los dos tipos de métodos, es decir: (i) el método termina en una finita cantidad de pasos, (ii) los resultados intermedios son aproximaciones a la solución exacta. Estos métodos existen, y se llaman *métodos Krylov*.

Según el teorema de Cayley-Hamilton, la matriz A anula su propio polinomio característico

$$p(\lambda) = \det(\lambda I_n - A) = \lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_1\lambda + p_0,$$

es decir $A^n + p_{n-1}A^{n-1} + \cdots + p_1 A + p_0 I_n = 0$. Dado que A es invertible tenemos $p_0 = \det(A) \neq 0$, y podemos calcular

$$A^{-1} = -\frac{1}{p_0}A^{n-1} - \frac{p_{n-1}}{p_0}A^{n-2} - \cdots - \frac{p_1}{p_0}I_n.$$

Concluimos que $A^{-1}b \in \langle b, Ab, A^2b, \ldots, A^{n-1}b\rangle$. Si definimos el espacio Krylov $\mathcal{K}_m$ como

$$\mathcal{K}_m := \mathcal{K}_m(A, b) := \langle b, Ab, A^2b, \ldots, A^{m-1}b\rangle,$$

entonces $\mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq \cdots \subseteq \mathcal{K}_n$. Ya sabemos $x = A^{-1}b \in \mathcal{K}_n$. Los métodos Krylov consisten en recorrer sucesivamente los espacios $\mathcal{K}_1, \mathcal{K}_2, \ldots$ y determinar (de una forma que vamos a especificar mas adelante) una aproximación $x_m \in \mathcal{K}_m(A, b)$ a la solución exacta x.

Lema 38. *Para $m \in \mathbb{N}$ son equivalentes*

(1) $\dim \mathcal{K}_{m+1} \leq m$,

(2) $\mathcal{K}_m = \mathcal{K}_{m+1}$,

(3) $A(\mathcal{K}_m) \subset \mathcal{K}_m$,

(4) $A^{-1}b \in \mathcal{K}_m$.

Demostración. tbc □

Definimos $m^*(A, b) := \min\{m \in \mathbb{N} \mid A^{-1}b \in \mathcal{K}_m(A, b)\}$, entonces el último resultado implica $\mathcal{K}_\ell(A, b) = \mathcal{K}_{m^*(A,b)}(A, b)$ para $\ell \geq m^*(A, b)$. Podemos formular el siguiente algoritmo.

Algorithm 3: Algoritmo general de Krylov

Input: $A, b, \texttt{tol}$

```
m=1;
repeat
    Encontrar aproximación x_m ∈ K_m(A,b) a la solución exacta x;
    m = m+1;
until ‖x − x_m‖_2 ≤ tol ∨ m = m*(A,b);
```

Output: x_m

El criterio de cancelación $\|x - x_m\|_2 \leq \texttt{tol}$ es no es práctico, pues no conocemos x, y lo vamos a tener que cambiar por algo computable. Notamos que $\dim \mathcal{K}_m(A,b) \leq m$, y por lo tanto es económico determinar x_m. La suposición de métodos Krylov es que aún con $m \ll n$ es posible encontrar una buena aproximación $x_m \in \mathcal{K}_m(A,b)$ a la solución exacta x. Obviamente, lo mas deseable es $m^*(A,b) \ll n$, y este caso se llama *breakdown*. Para concluir los algoritmos de Krylov, tenemos que

1. determinar como comprobar un breakdown,
2. calcular bases adecuadas de los espacios $\mathcal{K}_m(A,b)$,
3. y encontrar x_m.

1.6.1. Los algoritmos de Arnoldi y Lanczos

Las bases triviales $\left\{b, Ab, A^2b, \ldots, A^{m-1}b\right\}$ no son adecuadas, pues las matrices asociadas K_m son mal condicionadas. Eso se debe al efecto que las columnas $A^m b$ se vuelven casi paralelas, pues convergen a un vector propio asociado al valor propio de A mas grande en modulo.

Ejemplo 39. *Para la matriz $A = diag(1,2)$ y el vector $x^{(0)} = (1,1)^\top$ calculamos la sucesión*

$$x^{(k)} := A^k x^{(0)} = \begin{pmatrix} 1 \\ 2^k \end{pmatrix}.$$

Observamos que $x^{(k)}$ no converge, pero $x^{(k)}/\|x^{(k)}\|_2 \to (0,1)^\top$, y $(0,1)^\top$ es el vector propio a $\lambda_{\text{máx}} = 2$.

Aplicaremos Gram-Schmidt para ortogonalizar sucesivamente las bases de $\mathcal{K}_1(A,b) \subset \cdots \subset \mathcal{K}_{m^*(A,b)}(A,b)$: Si $\{q_1, \ldots, q_m\}$ es base ortogonal de $\mathcal{K}_m(A,b)$, entonces el elemento q_{m+1} se calcula según Lema 23

$$q_{m+1} = \frac{\widetilde{q}_{m+1}}{\|\widetilde{q}_{m+1}\|_2}, \qquad \text{donde} \qquad \widetilde{q}_{m+1} = \left(I - \Pi_{\mathcal{K}_m(A,b)}\right) A^m b,$$

y un *breakdown* se evidencia por

$$\widetilde{q}_{m+1} = 0.$$

Ortogonalizar las bases de de los espacios de Krylov es equivalente a determinar una factorizaciño QR de las matrices correspondientes, $K_m = Q_m R_m$. Sin embargo, no queremos determinar K_m y R_m de manera explicita, pues eso implicaría inestabilidades numéricas. Observamos lo siguiente. Si $\{q_1, \ldots, q_m\}$ es una base de $\mathcal{K}_m(A, b)$, entonces $q_m = c_1 b + c_2 Ab + \cdots + c_m A^{m-1} b$ con $c_m \neq 0$, o sea

$$Aq_m = c_1 Ab + c_2 A^2 b + \cdots + c_m A^m b = z + c_m A^m b$$

con $z \in \mathcal{K}_m(A, b)$. Vemos que

$$\widetilde{q}_{m+1} = \left(I - \Pi_{\mathcal{K}_m(A,b)}\right) A^m b = c_m^{-1} \left(I - \Pi_{\mathcal{K}_m(A,b)}\right) (Aq_m - z) = c_m^{-1} \left(I - \Pi_{\mathcal{K}_m(A,b)}\right) Aq_m,$$

y concluimos

$$q_{m+1} = \frac{\widehat{q}_{m+1}}{\|\widehat{q}_{m+1}\|_2}, \qquad \text{donde} \qquad \widehat{q}_{m+1} = \left(I - \Pi_{\mathcal{K}_m(A,b)}\right) Aq_m = Aq_m - \sum_{\ell=1}^{m} \langle Aq_m \, , q_\ell \rangle q_\ell.$$

La última formula para calcular una base ortonormal de $\mathcal{K}_m(A, b)$ se conoce como *método de Arnoldi*. Otra vez, el caso de un *breakdown* se evidencia por $\widehat{q}_{m+1} = 0$. Como en el caso de Gram-Schmidt, no se implementa la última suma directamente.

Algorithm 4: Algoritmo de Arnoldi

Input: A, b

$m = 1, q_1 = b/\|b\|_2$;
repeat
  $v = Aq_m$;
  for $\ell = 1$ **to** m **do**
    $h_{\ell,m} = \langle v \, , q_\ell \rangle$;
    $v = v - h_{\ell,m} q_\ell$;
  end
  $h_{m+1,m} = \|v\|_2$;
  if $h_{m+1,m} \neq 0$ **then**
    $q_{m+1} = v/h_{m+1,m}$;
  end
  $m = m + 1$;
until $h_{m+1,m} = 0$;

Haremos una observación con respecto a los números $h_{\ell,k}$ que usamos como variables intermedios en el Algoritmo de Arnoldi. Notamos que $h_{m+1,m} = \|\widehat{q}_{m+1}\|_2 = \langle \widehat{q}_{m+1} \, , q_{m+1} \rangle = \langle Aq_m \, , q_{m+1} \rangle$, y por lo tanto $\widehat{q}_{m+1} = \langle Aq_m \, , q_{m+1} \rangle q_{m+1}$. Obtenemos

$$Aq_m = \sum_{\ell=1}^{m+1} \langle Aq_m \, , q_\ell \rangle q_\ell = \sum_{\ell=1}^{m+1} h_{\ell,m} q_\ell. \tag{1.14}$$

Definición 40 (Matriz Hessenberg)**.** *Una matriz A se llama* ***matriz Hessenberg****, si $A_{j,k} = 0$ para $j > k + 1$.* □

Definimos la matriz $H \in \mathbb{K}^{n\times n}$ por $H_{j,k} := h_{j,k} = \langle Aq_k\,, q_j\rangle$, entonces H es una matriz Hessenberg, pues $Aq_k \in \mathcal{K}_{k+1}(A, b)$ y si $j > k + 1$ entonces q_j es ortogonal al $\mathcal{K}_{k+1}(A, b)$. La formula (1.14) se lee

$$A\underbrace{\begin{pmatrix} | & & | \\ q_1 & \cdots & q_m \\ | & & | \end{pmatrix}}_{=:Q_m} = \begin{pmatrix} | & & | \\ q_1 & \cdots & q_{m+1} \\ | & & | \end{pmatrix} \underbrace{\begin{pmatrix} h_{11} & \cdots & \cdots & h_{1m} \\ h_{21} & & & \\ & \ddots & & \vdots \\ & & h_{m,m-1} & h_{m,m} \\ & & & h_{m+1,m} \end{pmatrix}}_{=:\widetilde{H}_m}$$

y un *breakdown* se evidencia por $h_{m+1,m} = 0$. Además vemos que

$$\begin{pmatrix} - & q_1^{\mathsf{H}} & - \\ & \vdots & \\ - & q_m^{\mathsf{H}} & - \end{pmatrix} A \begin{pmatrix} | & & | \\ q_1 & \cdots & q_m \\ | & & | \end{pmatrix} = \underbrace{\begin{pmatrix} h_{11} & \cdots & \cdots & h_{1m} \\ h_{21} & & & \\ & \ddots & \ddots & \vdots \\ & & h_{m,m-1} & h_{m,m} \end{pmatrix}}_{=:H_m}.$$

Si la matriz A bajo consideración es hermitiana, se puede accelerar el Algoritmo de Arnoldi, y el resultado se llama el Algoritmo de Lanzcos. Por comodidad supongamos que $A \in \mathbb{R}^{n\times n}$ es simétrica. Entonces, H_m también es simétrica, y por ser Hessenberg, es tridiagonal. Eso significa que el loop interior de Algoritmo 4 corre solamente de $m - 1$ a m. Cambiaremos la notación y escribiremos

$$H_m = \begin{pmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_{m-1} \\ & & & \beta_{m-1} & \alpha_m \end{pmatrix}$$

Entonces podemos modificar el Algoritmo de Arnoldi, aprovechando la estructura tridiagonal de H_m.

Algorithm 5: Algoritmo de Lanczos

Input: A, b

$m = 1, \beta_0 = 0, q_0 = 0, q_1 = b/\|b\|_2$;
repeat
  $v = Aq_m$;
  $\alpha_m = \langle v\,, q_m \rangle$;
  $v = v - \beta_{m-1} q_{m-1} - \alpha_m q_m$;
  $\beta_m = \|v\|_2$;
  if $\beta_m \neq 0$ **then**
    $q_{m+1} = v/\beta_m$;
  end
  $m = m + 1$;
until $\beta_m = 0$;

1.6.2. Solución de sistemas lineales con métodos Krylov

Ya que tenemos bases de los espacios de Krylov $\mathcal{K}_m$, podemos considerar el paso 3 del Algoritmo general de Krylov,

encontrar aproximación $x_m \in \mathcal{K}_m(A, b)$ a la solución exacta x.

El método GMRES

En el método GRMES (*Generalized Minimal Residual Method*), se determina x_m como

$$x_m = \underset{y \in \mathcal{K}_m}{\arg\min} \|b - Ay\|_2.$$

Para determinar x_m, formulamos el problema anterior usando la matriz Q_m que contiene la base de $\mathcal{K}_m$ como columnas: $x_m = Q_m z_m$, donde

$$\begin{aligned} z_m = \underset{y \in \mathbb{R}^m}{\arg\min} \|b - AQ_m y\|_2 &= \underset{y \in \mathbb{R}^m}{\arg\min} \|b - Q_{m+1} \widetilde{H}_m y\|_2 \\ &= \underset{y \in \mathbb{R}^m}{\arg\min} \|Q_{m+1}^\top b - \widetilde{H}_m y\|_2 \\ &= \underset{y \in \mathbb{R}^m}{\arg\min} \|\|b\|_2 e_1 - \widetilde{H}_m y\|_2. \end{aligned}$$

En la penúltima identidad usamos el hecho de que $b - Q_{m+1}\widetilde{H}_m y$ es elemento de la imagen de Q_{m+1}. Notamos que el primer problema de minimización de arriba es del tamaño $n \times m$, mientras el último es de tamaño $(m+1) \times m$, y aprovechando la estructura Hessenberg de $\widetilde{H}_m$ puede ser resulto con una descomposición QR con costo asintótico m^2.

El método CG

En el método CG (*Conjugated Gradients*), se asume que la matriz A es simétrica y definida positiva. Recordamos que si una matriz $A \in \mathbb{R}^{n\times n}$ es simetrica y definida positiva, entonces $\langle x\,, y\rangle_A := y^\top Ax$ es un producto interno en $\mathbb{R}^n$ con norma inducida $\|x\|_A^2 := x^\top Ax$. En el método CG se determina x_m como $x_m = \arg\min_{y\in\mathcal{K}_m} \|x-y\|_A$, o sea

$$\|x - x_m\|_A = \min_{y\in\mathcal{K}_m(A,b)} \|x-y\|_A. \tag{1.15}$$

Como en el caso de mínimos cuadrados, podemos formular las ecuaciones normales del problema anterior: $x_m \in \mathcal{K}_m$ es la proyección ortogonal de x en el producto interno $\langle\cdot\,,\cdot\rangle_A$, es decir $x - x_m$ es ortogonal a $\mathcal{K}_m$,

$$0 = v^\top A(x - x_m) = v^\top(b - Ax_m) \quad \text{para todo } v \in \mathcal{K}_m(A,b), \tag{1.16}$$

Podemos reformular (1.16): $x_m = Q_m y_m$, donde $y_m \in \mathbb{R}^m$ es solución de

$$Q_m^\top A Q_m y_m = Q_m^\top b = \|b\|_2 e_1.$$

Recordamos que $Q_m^\top A Q_m = Q_m^\top Q_{m+1}\widetilde{H}_m = H_m$. Es decir, aparte de tridiagonal y simétrica, H_m es definida positiva. Por lo tanto, tiene una descomposición Cholesky de la forma $H_m = L_m L_m^\top$ con

$$L_m = \begin{pmatrix} \ell_{1,1} & & & & \\ \ell_{2,1} & \ell_{2,2} & & & \\ & \ddots & \ddots & & \\ & & \ell_{m-1,m-2} & \ell_{m-1,m-1} & \\ & & & \ell_{m,m-1} & \ell_{m,m} \end{pmatrix}.$$

Observamos que L_m, y por lo tanto L_m^{-1} tienen la forma

$$L_m = \begin{pmatrix} L_{m-1} & 0 \\ \ell_m^\top & \ell_{m,m} \end{pmatrix} \quad \text{y} \quad L_m^{-1} = \begin{pmatrix} L_{m-1}^{-1} & 0 \\ \ell_{m,m}^{-1}\ell_m^\top L_{m-1}^{-1} & \ell_{m,m}^{-1} \end{pmatrix}.$$

Escribimos $x_m = Q_m L_m^{-\top} L_m^{-1}\|b\|_2 e_1 = P_m a_m$, con $P_m = Q_m L_m^{-\top}$ y $a_m = L_m^{-1}\|b\|_2 e_1$. Vemos

$$a_m = L_m^{-1}\|b\|_2 e_1 = \begin{pmatrix} a_{m-1} \\ \alpha_{m-1} \end{pmatrix},$$

donde $\alpha_{m-1} = \ell_{m,m}^{-1}\ell_m^\top L_{m-1}^\top \|b\|_2 e_1$, y también

$$P_m = Q_m L_m^{-\top} = (Q_{m-1}, q_m)\begin{pmatrix} L_{m-1}^{-\top} & \ell_{m,m}^{-1} L_{m-1}^{-\top}\ell_m \\ 0 & \ell_{m,m}^{-1} \end{pmatrix} = \begin{pmatrix} P_{m-1} & p_{m-1} \end{pmatrix},$$

donde

$$p_{m-1} = \ell_{m,m}^{-1} Q_{m-1} L_{m-1}^{-\top} \ell_m + \ell_{m,m}^{-1} q_m.$$

Eso implica que x_m y los residuos $r_m = b - Ax_m$ son "updates" de la forma

$$\begin{aligned} x_m &= x_{m-1} + \alpha_{m-1} p_{m-1}, \\ r_m &= r_{m-1} - \alpha_{m-1} A p_{m-1}. \end{aligned} \tag{1.17}$$

Aunque arriba encontramos formulas para α_{m-1} y p_{m-1}, existe una forma mas eficiente realizar los updates en (1.17). Primero, notamos que $P_m L_m = Q_m$, es decir

$$q_m = \ell_{m,m-1} p_{m-2} + \ell_{m,m} p_{m-1}. \tag{1.18}$$

Además,

$$\begin{aligned} r_m &= b - Ax_m = \|b\|_2 q_1 - AQ_m y_m = \|b\|_2 v_1 - Q_{m+1} \widetilde{H}_m y_m \\ &= Q_m \|b\|_2 e_1 - Q_m \widetilde{H}_m y_m - q_{m+1} h_{m+1,m} e_m^\top y_m = -q_{m+1} h_{m+1,m} e_m^\top y_m \end{aligned} \tag{1.19}$$

De (1.18) y (1.19) podemos concluir que

$$p_m \in \operatorname{span}(p_{m-1}, q_{m+1}) = \operatorname{span}(p_{m-1}, r_m). \tag{1.20}$$

Además, $P_m^\top A P_m = L_m^{-1} V_m^\top A V_m L_m^{-\top} = L_m^{-1} \widetilde{H}_m L_m^{-\top} = I_m$, es decir,

$$p_m^\top A p_{m-1} = 0, \tag{1.21}$$

y de (1.19) vemos que

$$r_m^\top r_{m-1} = 0. \tag{1.22}$$

Ahora vamos a cambiar las normas de los p_{m-1}, y en función de este cambio también los α_{m-1}, tal que los updates en (1.17) no cambian. Escribimos

$$p_m = r_m + \beta_{m-1} p_{m-1}. \tag{1.23}$$

Falta calcular los α_{m-1} y β_{m-1} en (1.17) y (1.23). De (1.23) y (1.21) obtenemos

$$\beta_{m-1} = -\frac{r_m^\top A p_{m-1}}{p_{m-1}^\top A p_{m-1}}.$$

De (1.22) y (1.17) obtenemos

$$\alpha_{m-1} = \frac{r_{m-1}^\top r_{m-1}}{r_{m-1}^\top A p_{m-1}}.$$

Usando (1.23) notamos que $p_{m-1}^\top A p_{m-1} = r_{m-1}^\top A p_{m-1}$, y de (1.17) obtenemos $A p_{m-1} = \alpha_{m-1}^{-1}(r_{m-1} - r_m)$, y asi podemos escribir

$$\beta_{m-1} = \frac{r_m^\top r_m}{r_{m-1}^\top r_{m-1}}, \quad \text{y} \quad \alpha_{m-1} = \frac{r_{m-1}^\top r_{m-1}}{p_{m-1}^\top A p_{m-1}}.$$

Además, $x_1 = \alpha_0 b$, y

$$b^\top A x_1 = b^\top b \Rightarrow \alpha_0 = \frac{b^\top b}{b^\top A b}.$$

Eso nos lleva al siguiente algoritmo.

Algorithm 6: Algoritmo de gradientes conjugados (CG)

Input: A, b, tol

$x = 0$, $r = b - Ax$, $p = r$, $\gamma = \|r\|_2^2$, $m = 0$;
while $(\|r\|_2 > tol)$ **do**
  $\gamma = r^\top r$;
  $y = Ap$;
  $\alpha = \gamma / p^\top y$;
  $x = x + \alpha p$;
  $r = r - \alpha y$;
  $\beta = r^\top r / \gamma$;
  $p = r + \beta p$;
  $m = m + 1$;
end

Output: x

Lema 41. *Sea $A \in \mathbb{R}^{n\times n}$ simétrica y definida positiva con autovalores $\lambda_1 \geq \cdots \geq \lambda_n > 0$. Si $Ax = b$ y x_m es la solución en el paso m del algoritmo CG, entonces*

$$\|x - x_m\|_A \leq \min_{\substack{p_m \in \mathbb{P}_m \\ p_m(0)=1}} \max_{j=1,\ldots,n} |p_m(\lambda_j)| \|x\|_A.$$

Demostración. Primero notamos que $z_m \in \mathcal{K}_m(A, b)$ si y solo si existe $p_m \in \mathbb{P}_m$ con $p_m(0) = 1$ y $x - z_m = p_m(A)x$. Efectivamente, $z_m \in \mathcal{K}_m(A, b)$ por definición significa $z_m = q_{m-1}(A)b$ para algún $q_{m-1} \in \mathbb{P}_{m-1}$, y con $p_m(t) = 1 - tq_{m-1}(t)$ tenemos

$$x - z_m = x - q_{m-1}(A)b = x - q_{m-1}(A)Ax = p_m(A)x.$$

Por el otro lado, sea $p_m \in \mathbb{P}_m$ con $p_m(0) = 1$ y $x - z_m = p_m(A)x$. Entonces $1 - p_m(t) = tq_{m-1}(t)$ con q_{m-1} un polinomio de grado menor o igual que $m - 1$, y

$$z_m = (I - p_m(A))x = q_{m-1}(A)Ax = q_{m-1}(A)b \in \mathcal{K}_m(A, b).$$

De la propiedad (1.15) concluimos entonces

$$\|x - x_m\|_A = \min_{\substack{p_m \in \mathbb{P}_m \\ p_m(0)=1}} \|p_m(A)x\|_A.$$

Escribiremos $x = \sum_{j=1}^n \alpha_j v_j$ donde v_j es una base ortonormal de autovectores de A con autovalores λ_j. Calculamos $\|x\|_A^2 = \sum_{j=1}^n \alpha_j^2 \lambda_j$, y

$$\|p_m(A)x\|_A^2 = \sum_{j=1}^n \alpha_j^2 \lambda_j p_m(\lambda_j)^2 \leq \max_{j=1,\ldots,n} |p_m(\lambda_j)|^2 \|x\|_A^2.$$

□

Para estimar el lado derecho en de la desigualdad del último lema, vamos a usar los polinomios de *Chebyshev*.

Lema 42. *Definimos de manera recursiva los polinomios de Chebyshev*

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{m+1}(x) = 2xT_m(x) - T_{m-1}(x).$$

Entonces $T_m \in \mathbb{P}_m \setminus \mathbb{P}_{m-1}$, *y*

(a) $|T_m(x)| \leq 1$ *para* $|x| \leq 1$,

(b) en $[-1, 1]$, *los extremos de* T_m *son* ± 1 *y los alcanza en los* $m + 1$ *puntos* $x_j = \cos(\pi j/m)$, $x_j = 0, \ldots, m$. *Tenemos* $T_m(x_j) = (-1)^j$.

(c) Para $t \in [0, 1)$ *se tiene*

$$T_m\left(\frac{1+t}{1-t}\right) \geq \frac{1}{2}\left(\frac{1+\sqrt{t}}{1-\sqrt{t}}\right)^m.$$

Demostración. Usando identidades trigonometricas, es fácil ver

$$T_m(x) = \cos(m \arc\cos(x)), \quad |x| \leq 1.$$

Eso implica (a) y (b). Para demostrar (c), por inducción es fácil ver que

$$T_m(x) = \frac{1}{2}\left((x + \sqrt{x^2-1})^m + (x - \sqrt{x^2-1})^m\right), \quad |x| \geq 1.$$

Para $t \in [0, 1)$ tenemos $x = (1 + t)/(1 - t) \geq 1$, y

$$T_m\left(\frac{1+t}{1-t}\right) \geq \frac{1}{2}\left(\frac{1+t}{1-t} + \sqrt{\frac{(1+t)^2}{(1-t)^2} - 1}\right)^m = \frac{1}{2}\left(\frac{1+\sqrt{t}}{1-\sqrt{t}}\right)^m.$$

□

Lema 43. *Sea $0 < \alpha < \beta$. Entonces la única solución del problema de minimización*

$$\min_{\substack{p_m \in \mathbb{P}_m \\ p_m(0)=1}} \max_{x\in[\alpha,\beta]} |p_m(x)| \quad es \quad p_m^\star(x) = \frac{T_m\left(\frac{\beta+\alpha-2x}{\beta-\alpha}\right)}{T_m\left(\frac{\beta+\alpha}{\beta-\alpha}\right)}.$$

Demostración. Obviamente, $p_m^\star(0) = 1$, y

$$\max_{x\in[\alpha,\beta]} |p_m^\star(x)| = \frac{1}{|T_m\left(\frac{\beta+\alpha}{\beta-\alpha}\right)|}.$$

Además, dado que la aplicación

$$\begin{cases} [\alpha,\beta] \to [-1,1] \\ x \mapsto \frac{\beta+\alpha-2x}{\beta-\alpha} \end{cases}$$

es biyectiva, Lema 42 implica que hay $m+1$ puntos x_k donde $|p_m^\star(x_k)| = \max_{x\in[\alpha,\beta]} |p_m^\star(x)|$. Supongamos que $p_m^\star$ no es solución del problema de minimización o que hay otro, en ambos casos será $q_m^\star$ tal mínimo. Entonces $p_m^\star - q_m^\star$ tendrá $m+1$ zeros, y por lo tanto $p_m^\star - q_m^\star = 0$. □

Teorema 44. *Sea $A \in \mathbb{R}^{n\times n}$ simétrica y definida positiva con autovalores $\lambda_1 \geq \cdots \geq \lambda_n > 0$. Si $Ax = b$ y x_m es la solución en el paso m del algoritmo CG, entonces*

$$\|x - x_m\|_A \leq 2\left(\frac{\sqrt{\mathrm{cond}_2(A)}-1}{\sqrt{\mathrm{cond}_2(A)}+1}\right)^m \|x\|_A.$$

Demostración. Recordamos que $\mathrm{cond}_2(A) = \frac{\lambda_1}{\lambda_n}$. Usando Lemas 41 y 43, y al final Lema 42, obtenemos

$$\|x - x_m\|_A \leq \frac{1}{|T_m\left(\frac{\lambda_1+\lambda_n}{\lambda_1-\lambda_n}\right)|} = \frac{1}{|T_m\left(\frac{1+\mathrm{cond}_2(A)^{-1}}{1-\mathrm{cond}_2(A)^{-1}}\right)|} \leq 2\left(\frac{1-\mathrm{cond}_2(A)^{-1/2}}{1+\mathrm{cond}_2(A)^{-1/2}}\right)^m$$

□

1.7. Métodos para calcular valores propios

Recordamos que $\lambda \in \mathbb{K}$ se llama valor propio de $A \in \mathbb{K}^{n\times n}$ si existe $0 \neq v \in \mathbb{K}^n$ tal que $Av = \lambda v$. Para desarollar métodos para calcular valores propios y/o vectores propios tenemos que tomar en cuenta las siguientes dificultades:

(1) los autovalores de una matriz son las raices del polinomio característico, y por lo tanto no puede existir un *método directo* para calcular valores propios, es decir, un método que calcula valores propios con una finita cantidad de pasos. Tenemos que usar métodos iterativos.

(2) Una matriz real no necesariamente tiene valores propios reales. En general, los resultados de un algoritmo para calcular valores propios son números *complejos*.

Nuestro primer objetivo será calcular un vector propio asociado al valor propio mas grande en modulo.

- Para la matriz diagonal $A = \operatorname{diag}(\frac{1}{2}, 1)$ y el vector $x^{(0)} = (1,1)^\top$ calculamos la sucesión

$$x^{(k)} := A^k x^{(0)} = \begin{pmatrix} 2^{-k} \\ 1 \end{pmatrix}.$$

  Observamos que $x^{(k)} \to (0,1)^\top$, y $(0,1)^\top$ es el vector propio a $\lambda_{\text{máx}} = 1$.

- Para la matriz $A = \operatorname{diag}(1, 2)$ y el vector $x^{(0)} = (1,1)^\top$ calculamos la sucesión

$$x^{(k)} := A^k x^{(0)} = \begin{pmatrix} 1 \\ 2^k \end{pmatrix}.$$

  Observamos que $x^{(k)}$ no converge, pero $x^{(k)}/\|x^{(k)}\|_2 \to (0,1)^\top$, y $(0,1)^\top$ es el vector propio a $\lambda_{\text{máx}} = 2$.

Lema 45. *Sea $A \in \mathbb{R}^{n\times n}$ diagonalizable con autovalores reales $|\lambda_1| > |\lambda_2| \geq \ldots |\lambda_n|$ y autovectores $v_1, \ldots, v_n$. Sea $x^{(0)} \in \mathbb{R}^n$ con $v_1^\top x^{(0)} \neq 0$, entonces la sucesión $w^{(k)} := \frac{A^k x^{(0)}}{\|A^k x^{(0)}\|_2}$ satisface*

$$\|w^{(k)} - \mathit{signo}(\lambda_1)^k \widetilde{v}_1\|_2 = \mathcal{O}\left(|\lambda_2/\lambda_1|^k\right),$$

donde $\widetilde{v}_1$ es un autovector normalizado a λ_1. □

Demostración. Dado que A es diagonalizable existe una base $v_1, \ldots, v_n$ de $\mathbb{R}^n$ de vectores propios de A. Con $x^{(0)} = \sum_{j=1}^n x_j^{(0)} v_j$ calculamos

$$A^k x^{(0)} = \sum_{j=1}^n x_j^{(0)} \lambda_j^k v_j = \lambda_1^k \left(x_1^{(0)} v_1 + \sum_{j=2}^n x_j^{(0)} \left(\frac{\lambda_j}{\lambda_1}\right)^k v_j \right) =: \lambda_1^k (z + y^{(k)}).$$

Tenemos $\|y^{(k)}\|_2 = \mathcal{O}\left(|\lambda_2/\lambda_1|^k\right)$ y por la desigualdad triangular inversa también $\|z\|_2 - \|z + y^{(k)}\|_2 = \mathcal{O}\left(|\lambda_2/\lambda_1|^k\right)$. Por lo tanto

$$\frac{z + y^{(k)}}{\|z + y^{(k)}\|_2} - \frac{z}{\|z\|_2} = z\left(\frac{\|z\|_2 - \|z + y^{(k)}\|_2}{\|z\|_2\|z + y^{(k)}\|_2}\right) + \frac{y^{(k)}}{\|z + y^{(k)}\|_2}. \tag{1.24}$$

Multiplicando la última identitad por el signo de λ_1^k y tomando la norma $\|\cdot\|_2$ muestra el resultado. □

La sucesión de vectores $w^{(k)}$ converge a un autovector normalizado a λ_1. Para generar una aproximación a λ_1, vamos a usar el *cociente de Rayleigh*,

$$R_A(x) := \frac{x^\top Ax}{x^\top x}.$$

Lema 46. *Sea $A \in \mathbb{R}^{n\times n}$ una matriz, y λ valor propio con vector propio v. Para $y \in \mathbb{R}^n$ tenemos*

$$|R_A(y) - \lambda| \leq \|A - \lambda I_n\|_2 \frac{\|y - \alpha v\|_2}{\|y\|_2} \quad \text{para todo } \alpha \in \mathbb{R}.$$

Si A es simétrica[4]*, tenemos*

$$|R_A(y) - \lambda| \leq \|A - \lambda I_n\|_2 \left(\frac{\|y - \alpha v\|_2}{\|y\|_2}\right)^2 \quad \text{para todo } \alpha \in \mathbb{R}.$$

Demostración. Usando $A(\alpha v) = \lambda\alpha v$ calculamos

$$R_A(y) - \lambda = \frac{y^\top Ay - y^\top \lambda I_n y}{y^\top y} = \frac{y^\top (A - \lambda I_n)y}{y^\top y} = \frac{y^\top (A - \lambda I_n)(y - \alpha v)}{y^\top y}.$$

Si A es simétrica, podemos seguir calculando

$$R_A(y) - \lambda = \frac{y^\top (A - \lambda I_n)(y - \alpha v)}{y^\top y} = \frac{(y - \alpha v)^\top (A - \lambda I_n)(y - \alpha v)}{y^\top y}.$$

□

Corolario 47. *Sea $A \in \mathbb{R}^{n\times n}$ diagonalizable con autovalores reales $|\lambda_1| > |\lambda_2| \geq \ldots |\lambda_n|$ y autovectores $v_1, \ldots, v_n$. Sea $x^{(0)} \in \mathbb{R}^n$ con $v_1^\top x^{(0)} \neq 0$, y $w^{(k)}$ calculados según Lema 45. Entonces*

$$|R_A(w^{(k)}) - \lambda_1| = \mathcal{O}\left((\lambda_2/\lambda_1)^k\right).$$

Si A es simétrica, tenemos

$$|R_A(w^{(k)}) - \lambda_1| = \mathcal{O}\left((\lambda_2/\lambda_1)^{2k}\right).$$

□

[4]Es suficiente que A sea normal, es decir $A^\top A = AA^\top$.

Un algoritmo basado en los últimos resultados se llama *iteración de potencia (power iteration).*

Algorithm 7: power iteration

Input: $A, x^{(0)}, maxiter$, tol

$w = x^{(0)}/\|x^{(0)}\|_2$;
$\lambda = w^\top Aw$;
while $(\|Aw - \lambda w\|_2 > tol|\lambda|) \wedge (m < maxiter)$ **do**
  $w = Aw$;
  $w = w/\|w\|_2$;
  $\lambda = w^\top Aw$;
end

Output: w, λ

Comentario 48. *1. Notamos $Av_1 - \lambda_1 v_1 = 0$. El último algoritmo termina si las aproximaciones w y λ satisfacen eso numéricamente, es decir $\|Aw - \lambda w\|_2 \leq tol|\lambda|$.*

2. Si la condición $v_1^\top x^{(0)} \neq 0$ no se cumple al principio, se cumple despues de la primera iteración por errores de redondeo.

La iteración de potencia calcula el valor propio mas grande en modulo y vector propio asociado. Si $\mu \in \mathbb{R}$ no es valor propio de A, entonces $A - \mu I_n$ es invertible y los autovalores de $(A - \mu I_n)^{-1}$ son $(\lambda_j - \mu)^{-1}$, donde los λ_j son los autovalores de A, y los vectores propios son iguales.

Lema 49. *Sea $A \in \mathbb{R}^{n\times n}$ diagonalizable con valores propios reales λ_j y vectores propios v_j. Sea $\mu \in \mathbb{R}$ y $|\mu - \lambda_J| < |\mu - \lambda_K| \leq |\mu - \lambda_j|$, $j \neq J$. Sea $x^{(0)} \in \mathbb{R}^n$ con $v_J^\top x^{(0)} \neq 0$. Entonces la sucesión $w^{(k)} := \frac{(A-\mu I_n)^{-k}x^{(0)}}{\|(A-\mu I_n)^{-k}x^{(0)}\|_2}$ satisface*

$$\|w^{(k)} - signo(\lambda_J)^k \widetilde{v}_J\|_2 = \mathcal{O}\left(|(\mu - \lambda_J)/(\mu - \lambda_K)|^k\right),$$

donde $\widetilde{v}_J$ es un autovector normalizado a λ_J. Además, $|R_A(w^{(k)}) - \lambda_J| = \mathcal{O}\left(|(\mu - \lambda_J)/(\mu - \lambda_K)|^k\right)$, y si A es simétrica $|R_A(w^{(k)}) - \lambda_J| = \mathcal{O}\left(|(\mu - \lambda_J)/(\mu - \lambda_K)|^{2k}\right)$. □

Un algoritmo basado en el último resultados se llama *iteración inversa (inverse iteration).*

Algorithm 8: inverse iteration

Input: $A, x^{(0)}, maxiter$, μ, tol

$w = x^{(0)}/\|x^{(0)}\|_2$;
$\lambda = w^\top Aw$;
while $(\|Aw - \lambda w\|_2 > tol|\lambda|) \wedge (m < maxiter)$ **do**
  Resuelve sistema $(A + \mu I_n)v = w$ para v;
  $w = v/\|v\|_2$;
  $\lambda = w^\top Aw$;
end

Output: w, λ

Comentario 50. *1. El parametro μ se llama shift.*

2. *Si A es invertible y $\mu = 0$, entonces la iteración inversa aproxima el valor propio mas pequeño en modulo.*

3. *En cada iteración hay que resolver un sistema con matriz $A + \mu I_n$. Es la práctica, se hace una descomposición LU antes del loop principal, y en la línea 4 se resuelve con sustitución ascendende/descendente.*

Como ejemplo consideramos la matriz

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1+\varepsilon \end{pmatrix}$$

con $\varepsilon > 0$. Los valores propios de A son $\lambda_1 = 1, \lambda_2 = 1 + \varepsilon$, y la iteración de potencia converge con $\mathcal{O}(1/(1+\varepsilon)^k)$. Si ε es pequeño, la convergencia será muy lenta. Si aplicamos la iteración inversa con shift $\mu = 1 + \varepsilon/\delta$ para $\delta > 0$ grande, entonces obtenemos convegencia con

$$\mathcal{O}\left(\left|\frac{1-\mu}{1+\varepsilon-\mu}\right|^k\right) = \mathcal{O}\left(\left|\frac{-\varepsilon/\delta}{(1-1/\delta)\varepsilon}\right|^k\right) = \mathcal{O}\left((\delta-1)^{-k}\right).$$

Usando la iteración inversa con el shift correcto podemos logramos buena convergencia aún para problemas con autovalores mal separados. Concluimos que la elección del shift es importante para la convergencia de la iteración inversa, y la convergencia será mas rapida si el shift μ está cerca al valor propio que estamos buscando. Eso nos lleva a la idea de cambiar el shift en cada iteración: Si $w^{(0)}$ ya es una buena aproximación a un vector propio v_1 con valor propio λ_1, entonces $\mu^{(0)} = R_A(w^{(0)})$ es una buena aproximación a λ_1 según Lema 46. Podemos usar $\mu^{(0)}$ como shift y resolver el sistema $(A + \mu^{(0)} I_n)w^{(1)} = w^{(0)}$ para calcular una nueva aproximación $w^{(1)}$ a v_1. Asignamos $\mu^{(1)} := R_A(w^{(1)})$ y calculamos una nueva aproximación $w^{(2)}$ a v_1 resolviendo el sistema $(A + \mu^{(1)} I_n)w^{(2)} = w^{(1)}$. En iteración k, eso será

$$(A + R_A(w^{(k-1)})I_n)w^{(k)} = w^{(k-1)}.$$

Lema 51. *Sea $A \in \mathbb{R}^{n\times n}$ diagonalizable con valores propios reales λ_j y vectores propios v_j. Sea $x^{(0)} \in \mathbb{R}^n$ y $w^{(0)} := x^{(0)}/\|x^{(0)}\|_2$ y define la iteración*

$$w^{(k)} := \frac{u^{(k)}}{\|u^{(k)}\|_2}, \qquad \text{donde } u^{(k)} := (A + R_A(w^{(k-1)})I_n)^{-1} w^{(k-1)} \tag{1.25}$$

Entonces, para casi todos $x^{(0)}$, la iteración $w^{(k)}$ converge a un vector propio v_J. Es decir, si $x^{(0)}$ es suficientemente cerca de v_J, entonces la convergencia es cuadratica: existe $C > 0$ tal que

$$\|w^{(k)} - signo(\lambda_J)^k \widetilde{v}_J\|_2 \le C\|w^{(k-1)} - signo(\lambda_J)^{k-1}\widetilde{v}_J\|_2^2,$$
$$|R_A(w^{(k)}) - \lambda_J| \le C|R_A(w^{(k-1)}) - \lambda_J|^2.$$

Si la matriz A es simétrica[5]*, entonces la convergencia es cubica*

$$\|w^{(k)} - signo(\lambda_J)^k \widetilde{v}_J\|_2 \leq C\|w^{(k-1)} - signo(\lambda_J)^{k-1}\widetilde{v}_J\|_2^3,$$
$$|R_A(w^{(k)}) - \lambda_J| \leq C|R_A(w^{(k-1)}) - \lambda_J|^3.$$

Demostración. Demostramos solamente el orden de convergencia. Por ello, regresamos un momento al Lema 45 de la iteración de potencia. La convergencia en este caso es lineal, porque

$$\|y^{(k)}\|_2 \leq \left|\frac{\lambda_2}{\lambda_1}\right| \|z - A^{k-1}x^{(0)}\|_2,$$

y con (1.24) concluimos que existe una constante $C > 0$ tal que

$$\|w^{(k)} - \text{signo}(\lambda_1)^k \widetilde{v}_1\|_2 \leq C\left|\frac{\lambda_2}{\lambda_1}\right| \|w^{(k-1)} - \text{signo}(\lambda_1)^{k-1}\widetilde{v}_1\|_2$$

En el contexto del lema 49 la iteración inversa con shift μ, la última desigualdad se lee

$$\|w^{(k)} - \text{signo}(\lambda_J)^k \widetilde{v}_J\|_2 \leq C\left|\frac{\lambda_J - \mu}{\lambda_K - \mu}\right| \|w^{(k-1)} - \text{signo}(\lambda_J)^{k-1}\widetilde{v}_J\|_2$$

En la iteración de Rayleigh, $w^{(k)}$ se calcula por (1.25), es decir con shift $\mu = R_A(w^{(k-1)})$. Con el Lema 46 obtenemos

$$\begin{aligned}\|w^{(k)} - \text{signo}(\lambda_J)^k \widetilde{v}_J\|_2 &\leq C|\lambda_J - R_A(w^{(k-1)})|\|w^{(k-1)} - \text{signo}(\lambda_J)^{k-1}\widetilde{v}_J\|_2 \\ &\leq C\|w^{(k-1)} - \text{signo}(\lambda_J)^{k-1}\widetilde{v}_J\|_2^m,\end{aligned}$$

con $m = 2$ y, si A es simétrica, $m = 3$. □

El algoritmo basado en el último resultado se llama *iteración de Rayleigh (Rayleigh iteration).*

Algorithm 9: Rayleigh iteration

Input: $A, x^{(0)}, maxiter$, μ, tol
1 $w = x^{(0)}/\|x^{(0)}\|_2$;
2 $\lambda = w^\top Aw$;
3 **while** $(\|Aw - \lambda w\|_2 > tol|\lambda|) \wedge (m < maxiter)$ **do**
4 Resuelve sistema $(A + \lambda I_n)v = w$ para v;
5 $w = v/\|v\|_2$;
6 $\lambda = w^\top Aw$;
7 **end**
Output: w, λ

[5]Es suficiente que A sea normal, es decir $A^\top A = AA^\top$.

Comentario 52. *Observamos que en cada iteración tenemos que resolver un sistema con matriz $A + \lambda I_n$, y λ cambia en cada iteración. Es decir, el método es mas caro que la iteración inversa porque no podemos calcular una descomposición LU antes de entrar en el loop principal. Sin embargo, la convergencia de la iteración de Rayleigh es mayor que la converencia de la iteración inversa.*

Supongamos ahora que $A \in \mathbb{R}^{n\times n}$ es diagonalizable y

$$|\lambda_1| \geq \cdots \geq |\lambda_m| > |\lambda_{m+1}| \geq \cdots \geq |\lambda_n|,$$

y nuestro objetivo es calcular el subespacio generado por los primeros m vectores propios, $\mathcal{V} := \langle v_1, \ldots, v_m\rangle$, es decir, una base. Un caso especial será $\lambda_1 = \cdots = \lambda_m = \lambda$, donde $\mathcal{V} = \ker(A - \lambda I_n)$.

Generalizamos la iteración de potencia: Usamos m vectores de partida,

$$X^{(0)} := \left(x_1^{(0)}, \ldots, x_m^{(0)}\right) \in \mathbb{R}^{n\times m},$$

tal que las proyecciones ortogonales de las columnas de $X^{(0)}$ a $\mathcal{V}$ son linealmente independientes. Entonces esperamos que la iteración $A^k X^{(0)}$ converja en algún sentido a $\mathcal{V}$ con orden de convergencia $\mathcal{O}(|\lambda_{m+1}/\lambda_m|^k)$. Como en el caso de la iteración de potencia, esperamos overflow/underflow en el caso de valores propios mayor/menor que 1 en modulo. Además, si λ_1 es dominante en modulo, entonces todas las columnas de $A^k X^{(0)}$ convergerán a v_1 en dirección. Es decir, teoricamente siguen linealmente independientes las columnas de $A^k X^{(0)}$, pero en la practica son *numericamente linealmente dependientes*. Para evitar ambos problemas en la practica, tenemos que hacer una normalización. Dado que nos interesa solamente un base del subespacio $\mathcal{V}$, podemos calcular una base ortogonal de las columnas de $A^k X^{(0)}$. Por ejemplo, podemos calcular una factorización QR reducida

$$A^k X^{(0)} = Q^{(k)} R^{(k)}$$

con $Q^{(k)} \in \mathbb{R}^{n\times m}$ y $R^{(k)} \in \mathbb{R}^{m\times m}$ y usar $Q^{(k)}$ como aproximación a una base de $\mathcal{V}$. ¿Como vamos a medir el error? Observamos que $\mathcal{V}$ es invariante, es decir $A\mathcal{V} \subset \mathcal{V}$. Si $X \in \mathbb{R}^{n\times m}$ es una base de $\mathcal{V}$, entonces existe única matriz $V \in \mathbb{R}^{m\times m}$ con $AX = XV$, es decir $\|AX - XV\|_2 = 0$.

Lema 53. *Sea $A \in \mathbb{R}^{n\times n}$ unitariamente diagonalizable con valores propios $|\lambda_1| \geq \cdots \geq |\lambda_n|$, es decir*

$$A = Q \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} Q^{\mathsf{H}}$$

con $Q \in \mathbb{C}^{n\times n} = (v_1, \cdots, v_n)$ unitaria. Sea $1 \leq m \leq n-1$ con $|\lambda_m| > |\lambda_{m+1}|$, y P_m la proyección ortogonal al $\langle v_1, \ldots, v_m\rangle$. Sea $X^{(0)} \in \mathbb{R}^{n\times m}$ con $P_m X^{(0)}$ rango ráximo. Sea $A^k X^{(0)} = Q^{(k)} R^{(k)}$

una factorización QR reducida. Entonces existe una sucesión de matrizes $V^{(k)} \in \mathbb{R}^{m\times m}$ *tal que*

$$\|AQ^{(k)} - Q^{(k)}V^{(k)}\|_2 = \mathcal{O}\left(|\lambda_{m+1}/\lambda_m|^k\right).$$

□

Para calcular $Q^{(k)}$ hay que hacer una factorización QR de la matriz $A^kX^{(0)}$. Ya mencionamos que $A^kX^{(0)}$ es numericamente linealmente dependiente, es decir, su número de condicionamiento es muy grande. Para evitar de calcular $A^kX^{(0)}$, escribiremos

$$A^{k+1}X^{(0)} = AA^kX^{(0)} = AQ^{(k)}R^{(k)}$$

y calculamos una factorización $AQ^{(k)} = Q^{(k+1)}\widetilde{R}^{(k+1)}$. Con $R^{(k+1)} := \widetilde{R}^{(k+1)} \cdot R^{(k)}$ obtenemos

$$A^{k+1}X^{(0)} = Q^{(k+1)}R^{(k+1)}.$$

Para obtener un criterio de cancelación, usaremos la siguiente observación: si $Q^{(k)}$ fuera una base exacta del espacio propio que buscamos, entonces existe una matriz $\widetilde{V}^{(k)} \in \mathbb{R}^{m\times m}$ tal que $AQ^{(k)} - Q^{(k)}\widetilde{V}^{(k)} = 0$. Multiplicamos la última identidad con $(Q^{(k)})^{\mathsf{H}}$ y obtenemos

$$(Q^{(k)})^{\mathsf{H}}AQ^{(k)} = \widetilde{V}^{(k)}.$$

Lema 54. *Sea* $\widetilde{V}^{(k)} := (Q^{(k)})^{\mathsf{H}}AQ^{(k)}$. *Entonces*

$$\|(AQ^{(k)} - Q^{(k)}V)y\|_2^2 = \|(AQ^{(k)} - Q^{(k)}\widetilde{V}^{(k)})y\|_2^2 + \|(\widetilde{V}^{(k)} - V)y\|_2^2$$

para todo $k \in \mathbb{N}$, $y \in \mathbb{R}^m$, *y* $V \in \mathbb{R}^{m\times m}$.

El algoritmo basado en el último resultado se llama *iteración ortogonal*.

Algorithm 10: orthogonal iteration

Input: $A, X^{(0)}$,tol

$k = 0$;

$Q^{(k)}R^{(k)} = X^{(k)}$;

while $\|AQ^{(k)} - Q^{(k)}(Q^{(k)})^{\mathsf{H}}AQ^{(k)}\|_2 \leq tol$ **do**

  $Q^{(k+1)}R^{(k+1)} = AQ^{(k)}$;

  $k = k + 1$;

end

Capítulo 2

La transformación rápida de Fourier

Si queremos interpolar una función 2π-periodica, entonces tendrá sentido usar un interpolante 2π-periodico. Una función compleja 2π-periodica f pueder ser representada por su serie de Fourier,

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos(kx) + b_k \sin(kx).$$

Usamos el símbolo $\approx$, porque la convergencia de la suma y sus valores dependen de las propiedades de la función f. Usando la identidad de Euler $e^{i\theta} = \cos(\theta) + i\sin(\theta)$ podemos escribir

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos(kx) + b_k \sin(kx) = \sum_{k=-\infty}^{\infty} c_k e^{ikx},$$

donde $c_0 = a_0/2$, $c_k = (a_k - ib_k)/2$, $c_{-k} = (a_k + ib_k)/2$. Podemos cortar la suma del lado derecho e interpolar f por una suma de la forma

$$\sum_{k=-N}^{N} c_k e^{ikx} = e^{-iNx} \sum_{k=0}^{2N} c_{k-N} e^{ikx}.$$

2.1. La transformación discreta de Fourier (DFT)

Sea $\mathbb{T}_{N-1} := \left\{ p : [0, 2\pi] \to \mathbb{C} \mid p(x) = \sum_{k=0}^{N-1} c_k e^{ikx} \right\}$ el espacio vectorial de los polinomios trigonometricos sobre $\mathbb{C}$.

Lema 55. *Se tiene* $\dim \mathbb{T}_{N-1} = N$*, y para nodos* $x_0, \ldots, x_{N-1} \in [0, 2\pi)$ *distintos entre si y valores* $y_j \in \mathbb{C}$ *existe único polinomio trigonometrico interpolante* $p \in \mathbb{T}_{N-1}$ *con* $p(x_j) = y_j$ *para todo* $j = 0, \ldots, N-1$*.*

Demostración. Sea $p(x) = \sum_{k=0}^{N-1} c_k e^{ikx}$ un elemento de $\mathbb{T}_{N-1}$ con $p(x_j) = 0$ para todo $j = 0, \ldots, N-1$. Definimos $z_j = e^{ix_j}$ y observamos

$$0 = p(x_j) = \sum_{k=0}^{N-1} c_k e^{ikx_j} = \sum_{k=0}^{N-1} c_k z_j^k.$$

Es decir, el polinomio $\widetilde{p}(z) := \sum_{k=0}^{N-1} c_k z^k$ de grado $N-1$ tiene N raices x_j, $j = 0, \ldots, N-1$, y por lo tanto $\widetilde{p} = 0$, o bien $c_k = 0$, $k = 0, \ldots, N-1$. En particular obtenemos que las funciones e^{ikx}, $k = 0, \ldots, N-1$ son linealmente independientes, y concluimos $\dim \mathbb{T}_{N-1} = N$. Notamos que hemos demostrado la inyectividad de la aplicación lineal

$$\begin{cases} \mathbb{T}_{N-1} \to \mathbb{C}^N \\ p \mapsto (p(x_0), \ldots, p(x_{N-1})) \end{cases}.$$

Una aplicación lineal inyectiva tambien es sobreyectiva. Eso muestra la existencia y unicidad del polinomio trigonometrico interpolante. □

A partir de ahora consideramos nodos *equiespaciados*, es decir

$$x_j = 2\pi \frac{j}{N}, \quad j = 0, \ldots N-1. \tag{2.1}$$

En este caso podemos encontrar una formula explicita para calcular los coeficientes del polinomio trigonometrico interpolante. Introducimos la N-ésima raiz unitaria

$$\omega_N = e^{2\pi i/N}.$$

Lema 56. *Para $0 \leq k, \ell \leq N-1$ se tiene*

$$\frac{1}{N} \sum_{j=0}^{N-1} \left(\omega_N^{k-\ell}\right)^j = \delta_{k,\ell}.$$

Demostración. El caso $k = \ell$ es fácil, dado que $\omega_N^0 = 1$. Para $k \neq \ell$ notamos $\omega_N^{k-\ell} \neq 1$, y la suma es una suma geometrica. Calculamos

$$\sum_{j=0}^{N-1} \left(\omega_N^{k-\ell}\right)^j = \frac{1 - \omega_N^{(k-\ell)N}}{1 - \omega_N^{k-\ell}} = 0,$$

donde usamos $\omega_N^{(k-\ell)N} = e^{2\pi i(k-\ell)} = 1$. □

Lema 57. *Sean los nodos $x_0, \ldots, x_{N-1}$ dados por (2.1), y sean $y_j \in \mathbb{C}$. Sea $p \in \mathbb{T}_{N-1}$ el único polinomio trigonometrico interpolante, es decir $p(x_j) = y_j$ para todo $j = 0, \ldots, N-1$. Entonces*

(1) El polinomio p está dado por

$$p(x) = \sum_{k=0}^{N-1} c_k e^{ikx}, \qquad \textit{donde } c_k = \frac{1}{N}\sum_{\ell=0}^{N-1} \omega_N^{-k\ell} y_\ell.$$

(2) Con la matriz $V_N \in \mathbb{C}^{N\times N}$ dada por $(V_N)_{k,j} := \omega_N^{-kj}$, $k, j = 0, \ldots, N-1$ y los vectores $y = (y_0, \ldots, y_{N-1})^\top \in \mathbb{C}^N$, $c = (c_0, \ldots, c_{N-1})^\top \in \mathbb{C}^N$ se tiene

$$c = \frac{1}{N} V_n y.$$

(3) La matriz $\frac{1}{\sqrt{N}} V_N$ es simétrica y unitaria, es decir

$$\left(\frac{1}{\sqrt{N}} V_N\right)^{-1} = \frac{1}{\sqrt{N}} V_N^{\mathsf{H}}.$$

En otras palabras, la transformación inversa discreta de Fourier está dada por

$$y_k = \sum_{\ell=0}^{N-1} \omega_N^{k\ell} c_\ell.$$

Demostración. Para demostrar (1) es suficiente verificar $p(x_j) = y_j$. Notamos $e^{ikx_j} = e^{2\pi ikj/N} = \omega_N^{kj}$, y con Lemma 56 concluimos

$$p(x_j) = \sum_{k=0}^{N-1} \frac{1}{N} \sum_{\ell=0}^{N-1} \omega_N^{-k\ell} y_\ell e^{ikx_j} = \sum_{\ell=0}^{N-1} y_\ell \frac{1}{N} \sum_{k=0}^{N-1} \omega_N^{(j-\ell)k} = \sum_{\ell=0}^{N-1} y_\ell \delta_{j,\ell} = y_j.$$

Para demostrar (3), notamos que la matriz V_N es simétrica por definición, y por el Lema (56) se concluye que la matriz indicada es unitaria. □

La aplicación lineal

$$\mathcal{F}_N : \begin{cases} \mathbb{C}^N \to \mathbb{C}^N \\ y \mapsto \frac{1}{N} V_N y \end{cases}$$

se llama *transformación discreta de Fourier*[1]. Notamos que V_N contiene solamente los N elementos diferentes $\omega_N^{-\ell}$, $\ell = 0, \ldots, N-1$. Es decir, la computación $\mathcal{F}_N(y)$ a traves de la multiplicación matriz-vector necesita $\mathcal{O}(N)$ operaciones para ensamblar la matriz V_N y $\mathcal{O}(N^2)$ operaciones para la multiplicación matriz-vector, en total $\mathcal{O}(N^2)$.

[1] *Discrete Fourier transform, DFT.*

2.2. La transformación rapida de Fourier (FFT)

Una transformación rapida de Fourier es un algoritmo que realiza la operación $V_N y$ con un costo computacional $\mathcal{O}(N \log N)$. Hay varios algoritmos, el mas famoso es el algoritmo de *Cooley-Tukey* (1965). Se basa en la siguiente observación.

Lema 58. *Sea $N = 2M$ y $\omega = e^{\pm 2\pi i/N}$. Entonces, las sumas*

$$c_k = \sum_{j=0}^{N-1} \omega^{kj} y_j, \quad k = 0, \ldots, N-1,$$

pueden ser calculadas como

$$c_{2\ell} = \sum_{j=0}^{M-1} \xi^{\ell j} g_j, \quad c_{2\ell+1} = \sum_{j=0}^{M-1} \xi^{\ell j} h_j, \quad \ell = 0, \ldots, M-1$$

donde $\xi = \omega^2$ y $g_j = y_j + y_{j+M}$, $h_j = (y_j - y_{j+M})\omega^j$.

Demostración. Si $j = 2\ell$, entonces por $\omega^{2M\ell} = \omega^{N\ell} = 1$,

$$c_{2\ell} = \sum_{j=0}^{N-1} \omega^{2j\ell} y_j = \sum_{j=0}^{M-1} \left(\omega^{2j\ell} y_j + \omega^{2(j+M)\ell} y_{j+M} \right) = \sum_{j=0}^{M-1} (y_j + y_{j+M}) \xi^{j\ell}.$$

Si $j = 2\ell + 1$, entonces por $\omega^{M(2\ell+1)} = -1$,

$$c_{2\ell+1} = \sum_{j=0}^{N-1} \omega^{j(2\ell+1)} y_j = \sum_{j=0}^{M-1} \left(\omega^{j(2\ell+1)} y_j + \omega^{(j+M)(2\ell+1)} y_{j+M} \right) = \sum_{j=0}^{M-1} (y_j - y_{j+M}) \omega^j \xi^{j\ell}.$$

□

El Lema 58 muestra que la transformación discreta de un vector en $\mathbb{C}^N$ puede ser realizada usando dos transformaciones discretas de vectores en $\mathbb{C}^{N/2}$: Para $y \in \mathbb{C}^N$ sea $c = \mathcal{F}_N(y)$. Se define $g_k = y_k + y_{k+N/2}$, $h_k = (y_k - y_{k+N/2})\omega_N^k$, y según Lema 58 tenemos

$$(c_0, c_2, \ldots, c_{2M-2}) = \frac{1}{2} \mathcal{F}_{N/2}(g_0, \ldots, g_{N/2-1})$$
$$(c_1, c_3, \ldots, c_{2M-1}) = \frac{1}{2} \mathcal{F}_{N/2}(h_0, \ldots, g_{N/2-1}).$$

Si aplicamos esta idea de manera recursiva, llegamos al siguiente algoritmo. Por motivos de una visión completa, presentaremos el algoritmo rapido para la transformación inversa.

Algorithm 11: c = FFT(y) %Cooley-Tukey radix-2 FFT

Input: $N = 2^p$, $p \in \mathbb{N}_0$, $y = (y_0, \ldots, y_{N-1}) \in \mathbb{C}^N$

if $N = 1$ **then**
  $c_0 = y_0$;
else
  $\omega = e^{-2\pi i/N}$;
  $M = N/2$;
  Calcula $g = (g_0, \ldots, g_{M-1})$ con $g_k = y_k + y_{k+M}$;
  Calcula $h = (h_0, \ldots, h_{M-1})$ con $h_k = (y_k - y_{k+M})\omega^k$;
  $(c_0, c_2, \ldots, c_{N-2}) = \frac{1}{2}FFT(g)$;
  $(c_1, c_3, \ldots, c_{N-1}) = \frac{1}{2}FFT(h)$;
end

Output: $c = (c_0, \ldots, c_{N-1}) \in \mathbb{C}^N$

Corolario 59. *Algoritmo 11 calcula la tranformación discreta de Fourier de $y \in \mathbb{C}^N$ con un costo computacional de $\mathcal{O}(N \log_2 N)$.*

Demostración. Sea $N = 2^p$ y a_p el número de sumas/restas y m_p el número de productos para calcular $\mathcal{F}_N$. Entonces, aplicamos inducción con respecto a p para demostrar que $a_p = p2^p$ y $m_p = p2^{p-1}$. Para $p = 0$ tenemos $a_0 = 0$ y $m_0 = 0$. Luego,

$$a_{p+1} = 2a_p + 2 \cdot 2^p = 2p2^p + 2 \cdot 2^p = (p+1) \cdot 2 \cdot 2^p = (p+1) \cdot 2^{p+1},$$
$$m_{p+1} = 2m_p + 2^p = p2^p + 2^p = (p+1)2^p.$$

En total,

$$a_p + m_p = p2^p + p2^{p-1} = \frac{3}{2}p2^p = \frac{3}{2}N \log_2(N).$$

□

Algorithm 12: c = inverseFFT(y) %Cooley-Tukey radix-2 inverse FFT

Input: $N = 2^p$, $p \in \mathbb{N}_0$, $c = (c_0, \ldots, c_{N-1}) \in \mathbb{C}^N$

if $N = 1$ **then**
  $y_0 = c_0$;
else
  $\omega = e^{2\pi i/N}$;
  $M = N/2$;
  Calcula $g = (g_0, \ldots, g_{M-1})$ con $g_k = c_k + c_{k+M}$;
  Calcula $h = (h_0, \ldots, h_{M-1})$ con $h_k = (c_k - c_{k+M})\omega^k$;
  $(y_0, y_2, \ldots, y_{N-2}) = inverseFFT(g)$;
  $(y_1, y_3, \ldots, y_{N-1}) = inverseFFT(h)$;
end

Output: $y = (y_0, \ldots, y_{N-1}) \in \mathbb{C}^N$

2.3. Una aplicación de FFT: Convolución discreta rapida

Recordamos el *teorema de convolución*, es decir,

$$\mathcal{F}(f \star g) = \mathcal{F}(f) \cdot \mathcal{F}(g),$$

donde la convolución de dos funciones se define por

$$f \star g(x) = \int f(y)g(x-y)\,dy.$$

Hay una version discreta de este resultado. La convolución discreta aparece por ejemplo si uno quiere calcular el producto de dos polinomios $p(x) = \sum_{k=0}^{n} p_k x^k$ y $q(x) = \sum_{k=0}^{n} q_k x^k$, es decir

$$(pq)(x) = \sum_{k=0}^{m+n} c_k x^k, \qquad \text{con } c_k = \sum_{j=0}^{k} p_j q_{k-j}.$$

Definición 60. *Una sucesión $(f_j)_{j\in\mathbb{Z}} \in \mathbb{C}^{\mathbb{Z}}$ se llama N-periodica, si $f_j = f_{j+N}$ para todo $j \in \mathbb{Z}$. Sea $\mathbb{C}^{\mathbb{Z}}_N$ el espacio vectorial de todas las sucesiones N-periodicas.*

(1) Definimos la transformación de Fourier $\mathcal{F}_N : \mathbb{C}^{\mathbb{Z}}_N \to \mathbb{C}^{\mathbb{Z}}_N$ por

$$\mathcal{F}_N(f)_j = \frac{1}{N} \sum_{k=0}^{N-1} \omega_N^{-jk} f_k.$$

(2) Definimos la convolución $\star : \mathbb{C}^{\mathbb{Z}}_N \times \mathbb{C}^{\mathbb{Z}}_N \to \mathbb{C}^{\mathbb{Z}}_N$ por

$$(f \star g)_j = \frac{1}{N} \sum_{k=0}^{N-1} f_k g_{j-k}$$

□

Entonces tenemos el siguiente resultado.

Teorema 61. *Sean $f, g \in \mathbb{C}^{\mathbb{Z}}_N$. Entonces*

$$\mathcal{F}_N(f \star g) = \mathcal{F}_N(f) \cdot \mathcal{F}_N(g),$$

donde $\cdot$ representa el producto por componente. Además, $\mathcal{F}_N : \mathbb{C}^{\mathbb{Z}}_N \to \mathbb{C}^{\mathbb{Z}}_N$ es biyectiva, y $\mathcal{F}_N^{-1}(f)_j = \sum_{k=0}^{N-1} \omega_N^{kj} f_j$. □

Por el último teorema, podemos calcular $f \star g$ por

$$f \star g = \mathcal{F}_N^{-1}\left(\mathcal{F}_N(f) \cdot \mathcal{F}_N(g)\right),$$

y si usamos las versiones rapidas de la transformación de Fourier y su inversa, el costo computacional será $\mathcal{O}(N \log_2 N)$ en vez de $\mathcal{O}(N^2)$.

Capítulo 3

Diferencias finitas para ecuaciones diferenciales

Problemas de valor inicial para ecuaciones diferencial como

$$\begin{aligned} y'(x) &= f(x, y(x)), \quad \text{para todo } x > 0, \\ y(0) &= y_0, \end{aligned}$$

o tambien

$$\begin{aligned} y''(x) + y'(x) &= f(x, y(x), y'(x)), \quad \text{para todo } x > 0, \\ y(0) &= y_0, \\ y'(0) &= y_1 \end{aligned}$$

pueden ser aproximados numéricamente con métodos que *marchan en el tiempo*, como el método de Euler, métodos Runge-Kutta, o Adams-Bashforth. Estos métodos son, estructuralmente, nada mas que métodos de integración numérica.

Por otro lado, si consideramos problemas de condición de frontera como

$$\begin{aligned} -u''(x) &= f(x) \quad \text{para todo } x \in (a, b) \\ u(a) &= u_a, \\ u(b) &= u_b, \end{aligned}$$

o tambien para ecuaciones diferenciales parciales, los métodos mencionados antes ya no se aplican. En este caso, se usa el método de diferencias finitas (MDF). Es conceptualmente sencillo, y su implementación es relativamente fácil. Para aproximar la solución $u : \Omega \to \mathbb{R}$ a una ecuacion diferencial con condiciones de frontera dada en un dominio Ω, la idea del MDF es la siguiente:

1. Generar una *malla* Ω_h del dominio, es decir, un conjunto finito de puntos. El parametro $h > 0$ indica la *resolución*. El objetivo será calcular una función *discreta* $u_h : \Omega_h \to \mathbb{R}$,

dada solamente en los puntos de Ω_h. Se espera que u_h converge a u en algún sentido si h converge a 0.

2. Para calcular la función u_h, tenemos que transformar la ecuación diferencial en un sistema de dimension finita, reemplazando las derivadas por cocientes de la diferencia con parametro h. Por ejemplo, si x y $x+h$ ambos son elementos de la malla Ω_h, podemos aproximar

$$u'(x) \sim \frac{u(x+h)-u(x)}{h}.$$

De esa manera reemplazamos una ecuación diferencial lineal por un sistema lineal $A_h u_h = f_h$, donde interpretamos nuestra función discreta u_h ya como un vector. La matriz A_h representa el operador diferencial *aproximado* de la EDP, y el vector f_h el lado derecho.

3.1. Diferencias finitas para problemas ellipticos

Sea $\Omega \subset \mathbb{R}^d$ con frontera $\partial\Omega$, y $f \in C(\Omega)$. Consideramos el problema de buscar la solución u de la ecuación de Poisson

$$\begin{aligned} -\Delta u(x) &= f(x) \text{ para todo } x \in \Omega, \\ u(x) &= g(x) \text{ para todo } x \in \partial\Omega. \end{aligned} \tag{3.1}$$

La última ecuación es un modelo para la distribución de la temperatura en un cuerpo Ω con fuente de calor f y temperatura fija g en el borde del cuerpo. Si la fuente es negativa, entonces esperamos que el máximo de tempertura se alcanza en el borde del cuerpo. El próximo resultado formaliza esta intuición. Se demuestra en el curso de EDP.

Teorema 62 (Principio débil del máximo)**.** *Sea $u \in C^2(\Omega) \cap C(\overline{\Omega})$.*

(1) Si $-\Delta u \leq 0$ en Ω, entonces $\max_{x\in\Omega} u(x) \leq \max_{x\in\partial\Omega} u(x)$.

(2) Si $-\Delta u \geq 0$ en Ω, entonces $\min_{x\in\Omega} u(x) \geq \min_{x\in\partial\Omega} u(x)$.

□

El principio de máximo implica el siguiente resultado de estabilidad.

Corolario 63. *Una solución $u \in C^2(\Omega) \cap C(\overline{\Omega})$ de (3.1) es única.*

Demostración. Si u y v son dos soluciones, entonces $-\Delta(u-v) = 0$ en Ω y $u - v = 0$ sobre $\partial\Omega$. El principio de máximo implica

$$0 = \min_{x\in\partial\Omega}(u-v)(x) \leq \min_{x\in\overline{\Omega}}(u-v)(x) \leq \max_{x\in\partial\Omega}(u-v)(x) \leq \max_{x\in\overline{\Omega}}(u-v)(x) = 0.$$

□

3.1.1. Diferencias finitas en una dimension

Dado $h > 0$, la primera derivada $u'(x)$ de una función pueder ser aproximada por las diferencias

$$\partial^{h,+}u(x) := \frac{u(x+h)-u(x)}{h}, \quad \partial^{h,-}u(x) := \frac{u(x)-u(x-h)}{h},$$

o también por la diferencia simétrica

$$\partial^{h,0}u(x) := \frac{u(x+h)-u(x-h)}{2h}.$$

La segunda derivada $u''(x)$ puede ser aproximada por la diferencia

$$\partial^{h,-}\partial^{h,+}u(x) = \frac{\frac{u(x+h)-u(x)}{h} - \frac{u(x)-u(x-h)}{h}}{h} = \frac{u(x+h)-2u(x)+u(x-h)}{h^2}.$$

Con respecto al error que se introduce al aproximar derivadas por diferencias, podemos decir lo siguiente.

Lema 64. *Sea $\Omega \subset \mathbb{R}$ un intervalo abierto y sea $[x-h, x+h] \subset \overline{\Omega}$. Entonces*

$$|\partial^{h,\pm}u(x) - u'(x)| \leq \frac{h}{2}\|u\|_{C^2(\overline{\Omega})} \text{ si } u \in C^2(\overline{\Omega}),$$
$$|\partial^{h,0}u(x) - u'(x)| \leq \frac{h^2}{6}\|u\|_{C^3(\overline{\Omega})} \text{ si } u \in C^3(\overline{\Omega}),$$
$$|\partial^{h,-}\partial^{h,+}u(x) - u''(x)| \leq \frac{h^2}{12}\|u\|_{C^4(\overline{\Omega})} \text{ si } u \in C^4(\overline{\Omega}).$$

Demostración. Demostramos solamente la primera desigualdad. Por el Teorema de Taylor existe $\xi \in (x, x+h)$ tal que

$$u(x+h) = u(x) + hu'(x) + h^2\frac{u''(\xi)}{2},$$

o bien

$$\frac{u(x+h)-u(x)}{h} - u'(x) = \frac{h}{2}u''(\xi).$$

Tomando el valor absoluto y usando

$$|u''(\xi)| \leq \|u\|_{C^2(\overline{\Omega})}$$

muestra el resultado. □

Dada $f : (0,1) \to \mathbb{R}$ y $u_0, u_1 \in \mathbb{R}$, consideramos ahora el problema

$$-u''(x) = f(x) \text{ para todo } x \in \Omega := (0,1), \tag{3.2a}$$
$$u(0) = u_0, u(1) = u_1. \tag{3.2b}$$

Por el Lema 64 podemos escribir

$$-\partial^{h,-}\partial^{h,+}u(x) = f(x) + \mathcal{O}(h^2) \text{ para todo } x \in \Omega.$$

Primero, podemos abandonar el termino $\mathcal{O}(h^2)$. Segundo, definimos las mallas

$$\begin{aligned}\Omega_h &:= \{h, 2h, \ldots, (n-1)h = 1-h\},\\ \overline{\Omega}_h &:= \{0, h, 2h, \ldots, 1-h, 1\},\\ \partial\Omega_h &:= \overline{\Omega}_h \setminus \Omega_h = \{0,1\},\end{aligned}$$

donde $n = 1/h$. Notamos que si $x \in \Omega_h$, entonces la diferencia $\partial^{h,-}\partial^{h,+}u(x)$ se calcula usando solamente los valores de u en $x-h, x, x+h \in \overline{\Omega}_h$. Es decir, si $v_h : \overline{\Omega}_h \to \mathbb{R}$ es una función definida solamente en los puntos de la malla $\overline{\Omega}_h$, entonces queda bien definida la diferencia $\partial^{h,-}\partial^{h,+}v_h(x)$ para cada punto $x \in \Omega_h$. El problema de hallar $u : \Omega \to \mathbb{R}$ solución de (3.2) lo podemos aproximar por el problema de hallar $u_h : \overline{\Omega}_h \to \mathbb{R}$ dada por

$$-\partial^{h,-}\partial^{h,+}u_h(x) = f(x) \text{ para todo } x \in \Omega_h, \tag{3.3a}$$
$$u_h(0) = u_0, u_h(1) = u_1. \tag{3.3b}$$

Para calcular u_h tenemos que fijar sus valores en los $n+1$ puntos de $\overline{\Omega}_h$. Pues, (3.3a) representan $n-1$ ecuaciones lineales, una para cada punto en Ω_h. Las condiciones (3.3b) son 2 ecuaciones lineales. En total, (3.3) es un sistema lineal de $n+1$ ecuaciones para los $n+1$ valores de u_h. Explicitando la ecuaciones llegamos a

$$\begin{aligned} u_h(0) &= u_0\\ h^{-2}[-u_h(0) + 2u_h(h) - u_h(2h)] &= f(h)\\ h^{-2}[-u_h(h) + 2u_h(2h) - u_h(3h)] &= f(2h)\\ h^{-2}[-u_h(2h) + 2u_h(3h) - u_h(4h)] &= f(3h)\\ &\vdots\\ h^{-2}[-u_h(1-2h) + 2u_h(1-h) - u_h(1)] &= f(1-h)\\ u_h(1) &= u_1. \end{aligned} \tag{3.4}$$

La primera y última ecuación en (3.4) corresponden a las condiciones de frontera (3.3b), y las

podemos eliminar desde el principio, asi llegamos a

$$\begin{aligned} h^{-2}[2u_h(h) - u_h(2h)] &= f(h) + h^{-2}u_0 \\ h^{-2}[-u_h(h) + 2u_h(2h) - u_h(3h)] &= f(2h) \\ h^{-2}[-u_h(2h) + 2u_h(3h) - u_h(4h)] &= f(3h) \\ &\vdots \\ h^{-2}[-u_h(1-2h) + 2u_h(1-h)] &= f(1-h) + h^{-2}u_1. \end{aligned} \tag{3.5}$$

En forma matricial, eso es

$$\frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} u_h(h) \\ u_h(2h) \\ \vdots \\ u_h(1-h) \end{pmatrix} = \begin{pmatrix} f(h) + h^{-2}u_0 \\ f(2h) \\ \vdots \\ f(1-h) + h^{-2}u_1 \end{pmatrix}. \tag{3.6}$$

Por ahora hemos considerado u_h como una función. Su dominio es un conjunto finito de $n+1$ puntos, y por lo tanto podemos considerar u_h como un vector $\underline{u}_h \in \mathbb{R}^{n+1}$, usando la relacion

$$\underline{u}_{h,j} = u_h(jh), \quad \text{para } j = 1, \ldots, n-1.$$

El lado derecho de (3.6) lo escribiremos también como un vector

$$\begin{aligned} \underline{f}_1 &= f(h) + h^{-2}u_0, \\ \underline{f}_j &= f(jh), \quad \text{para } j = 2, \ldots, n-2, \\ \underline{f}_{n-1} &= f(1-h) + h^{-2}u_1, \end{aligned}$$

y al final tenemos llegamos al sistema

$$A_h \underline{u}_h = \underline{f}_h. \tag{3.7}$$

3.1.2. Diferencias finitas en dos dimensiones

Ahora sea $\Omega = (0,1) \times (0,1)$. Para $h > 0$ usamos las diferencias

$$\begin{aligned} \partial_x^{h,+}u(x,y) &:= \frac{u(x+h,y) - u(x,y)}{h}, \quad \partial_x^{h,-}u(x,y) := \frac{u(x,y) - u(x-h,y)}{h}, \\ \partial_y^{h,+}u(x,y) &:= \frac{u(x,y+h) - u(x,y)}{h}, \quad \partial_y^{h,-}u(x,y) := \frac{u(x,y) - u(x,y-h)}{h}. \end{aligned}$$

El Laplaciano $-\Delta u$ puede ser aproximado por la diferencia

$$\begin{aligned} -\Delta_h u(x,y) :&= -\partial_x^{h,-}\partial_x^{h,+}u(x,y) - \partial_y^{h,-}\partial_y^{h,+}u(x,y) \\ &= \frac{4u(x,y) - u(x+h,y) - u(x-h,y) - u(x,y+h) - u(x,y-h)}{h^2}. \end{aligned}$$

Por el Lema 64 podemos escribir

$$-\Delta_h u(x,y) = f(x,y) + \mathcal{O}(h^2) \text{ para todo } (x,y) \in \Omega.$$

Para generar una malla, usamos los puntos $x_j = jh$, $j = 0, \ldots, n$, donde $n = 1/h$, y $x_{ij} = (x_i, x_j)$. Definimos las mallas

$$\begin{aligned} \Omega_h &:= \{x_{ij} \mid 1 \le i,j \le n-1\}, \\ \overline{\Omega}_h &:= \{x_{ij} \mid 0 \le i,j \le n\}, \\ \partial\Omega_h &:= \overline{\Omega}_h \setminus \Omega_h. \end{aligned}$$

El problema de hallar la solución u de (3.1) (por comodidad supongamos que $g = 0$) lo podemos aproximar por el problema de hallar $u_h : \overline{\Omega}_h \to \mathbb{R}$ dada por

$$\begin{aligned} -\Delta_h u_h(x_{ij}) &= f(x_{ij}) \text{ para todo } x_{ij} \in \Omega_h, \\ u_h(x_{ij}) &= 0 \text{ para todo } x_{ij} \in \partial\Omega_h. \end{aligned} \tag{3.8}$$

Notamos que para $x_{ij} \in \Omega_h$, se tiene

$$-\Delta_h u_h(x_{ij}) = \frac{4u_h(x_{ij}) - u(x_{i+1,j}) - u(x_{i-1,j}) - u(x_{i,j+1}) - u(x_{i,j-1})}{h^2},$$

y por lo tanto Δ_h se llama *patrón de 5 puntos*. Notamos que (3.8) es un sistema lineal de $(n+1)^2$ ecuaciones lineales para las $(n+1)^2$ valores $u_h(x_{ij})$, $x_{ij} \in \overline{\Omega}_h$. Si lo queremos escribir en forma matricial, tenemos que numerar las desconocidas. Como en el caso de una dimensión, vamos a eliminar las desconocidas que corresponden a valores de u_h en la frontera $\partial\Omega_h$ e incorporarlas en el lado derecho. Asi nos quedamos con $(n-1)^2$ desconocidas, que son los valores de $u_h(x_{ij})$, $x_{ij} \in \Omega_h$. Para representar las desconocidas como un vector $\underline{u}_h \in \mathbb{R}^{(n-1)^2}$ vamos a numerar los puntos $x_{ij} \in \Omega_h$ de manera lexicográfica

$$\underline{u}_{h,i(n-1)+j} = u_h(x_{ij}). \tag{3.9}$$

Con esta numeración, llegamos al sistema $A_h\underline{u}_h = \underline{f}_h$, donde

$$A_h = \frac{1}{h^2}\begin{pmatrix} \widetilde{A} & -I & & & & \\ -I & \widetilde{A} & -I & & & \\ & -I & \widetilde{A} & -I & & \\ & & \ddots & \ddots & \ddots & \\ & & & -I & \widetilde{A} & -I \\ & & & & -I & \widetilde{A} \end{pmatrix} \in \mathbb{R}^{(n-1)^2 \times (n-1)^2}, \tag{3.10}$$

con

$$\widetilde{A} = \begin{pmatrix} 4 & -1 & & & & \\ -1 & 4 & -1 & & & \\ & -1 & 4 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 4 & -1 \\ & & & & -1 & 4 \end{pmatrix} \in \mathbb{R}^{(n-1)\times(n-1)}$$

e $I \in \mathbb{R}^{(n-1)\times(n-1)}$ la matriz de identitad.

3.1.3. Teoria de convergencia a priori

Supongamos que $d = 2$, aunque todos los resultados de esta sección se aplican en cualquier dimension. Ya concluimos que la discretizacion de la ecuación de Poisson (3.1) por diferencias finitas puede escribirse en forma matricial $A_h \underline{u}_h = f_h$. Ahora presentamos la teoria a priori. Nos preguntamos

(1) ¿Existe única solución u_h de (3.8)? En otras palabras, ¿Es invertible la matriz A_h?

(2) ¿En que sentido converge u_h a la solución exacta u?

Notamos que A_h es nada mas que la representación de la aplicación lineal $-\Delta_h$ definida sobre espacios de funciones discretas. Para formalizar esto, definimos los espacios vectoriales de funciones discretas

$$U_h := \{u_h : \Omega_h \to \mathbb{R}\}, \quad \text{y} \quad \overline{U}_h := \left\{u_h : \overline{\Omega}_h \to \mathbb{R}\right\}.$$

Ambos espacios son de dimension finita, $\dim U_h = (n-1)^2$ y $\dim \overline{U}_h = (n+1)^2$. Además, vamos a necesitar el espacio con condicion de frontera

$$U_h^0 = \left\{u_h \in \overline{U}_h \mid u_h(x_{ij}) = 0 \text{ para todo } x_{ij} \in \partial\Omega_h\right\} \subset U_h.$$

Notamos que $\dim U_h^0 = \dim U_h$. Definimos las normas

$$\|v_h\|_{\infty,\Omega_h} = \max_{x_{ij}\in\Omega_h} |v_h(x_{ij})|, \quad \text{y} \quad \|v_h\|_{\infty,\overline{\Omega}_h} = \max_{x_{ij}\in\overline{\Omega}_h} |v_h(x_{ij})|,$$

y las operadores de restricción

$$[\cdot]_{\overline{\Omega}_h} : \begin{cases} C(\overline{\Omega}) & \to \overline{U}_h \\ u & \mapsto ([u]_{\overline{\Omega}_h})(x_{ij}) = u(x_{ij}), \text{ para todo } x_{ij} \in \overline{\Omega}_h, \end{cases}$$

$$[\cdot]_{\Omega_h} : \begin{cases} C(\Omega) & \to U_h \\ u & \mapsto ([u]_{\Omega_h})(x_{ij}) = u(x_{ij}), \text{ para todo } x_{ij} \in \Omega_h. \end{cases}$$

Vamos a formular una teoria general de convergencia a priori, usando patrones generales L_h que aproximan el operador diferencial bajo consideración.

Definición 65 (Consistencia y Estabilidad)**.** *Sea $h > 0$ y $L_h : \overline{U}_h \to U_h$ una aplicación lineal.*

*(1) Si $u \in C^2(\Omega) \cap C(\overline{\Omega})$, se dice que L_h es **consistente** con Δ de orden $k \in \mathbb{N}$ para u, si existe una constante $C_c > 0$ independiente de h, tal que*

$$\|L_h[u]_{\overline{\Omega}_h} - [\Delta u]_{\Omega_h}\|_{\infty,\Omega_h} \leq C_c h^k.$$

*(2) Se dice que L_h es **estable**, si existe una constante $C_e > 0$ independiente de h, tal que*

$$\|u_h\|_{\infty,\overline{\Omega}_h} \leq C_e \|L_h u_h\|_{\infty,\Omega_h} \quad \text{para todo } u_h \in U_h^0.$$

□

Teorema 66 (Consistencia + estabilidad $\Longrightarrow$ convergencia)**.** *Sea $u \in C^2(\Omega) \cap C(\overline{\Omega})$ solución de* (3.1)*, es decir*

$$\begin{aligned} -\Delta u(x,y) &= f(x,y) \text{ para todo } (x,y) \in \Omega, \\ u(x,y) &= 0 \text{ para todo } (x,y) \in \partial\Omega. \end{aligned}$$

Sea L_h estable y consistente con Δ de orden k para u. Entonces, $L_h : U_h^0 \to U_h$ es invertible, y la solución discreta $u_h := -L_h^{-1}[f]_{\Omega_h}$ método converge con orden k, es decir,

$$\|u_h - [u]_{\overline{\Omega}_h}\|_{\infty,\overline{\Omega}_h} \leq C h^k.$$

Demostración. Notamos que L_h sea estable significa nada mas que $L_h : U_h^0 \to U_h$ es inyectiva. Dada que $\dim U_h^0 = \dim U_h$, concluimos que L_h es invertible. Por la estabilidad y consistencia,

$$\begin{aligned} \|u_h - [u]_{\overline{\Omega}_h}\|_{\infty,\overline{\Omega}_h} &= \| - L_h^{-1}[f]_{\Omega_h} - [u]_{\overline{\Omega}_h}\|_{\infty,\overline{\Omega}_h} \\ &\leq C_e \|[f]_{\Omega_h} + L_h[u]_{\overline{\Omega}_h}\|_{\infty,\Omega_h} = \|L_h[u]_{\overline{\Omega}_h} - [\Delta u]_{\Omega_h}\|_{\infty,\Omega_h} \leq C_e C_c h^k. \end{aligned}$$

□

Notamos que

$$\Delta_h : \overline{U}_h \to U_h$$

es una aplicación lineal. El Lema 64 implica la consistencia de Δ_h.

Corolario 67. *Si $u \in C^4(\overline{\Omega})$, entonces Δ_h es es consistente con Δ para u de orden* 2. □

Para obtener la estabilidad de Δ_h, tenemos que trabajar un poco.

Lema 68. *Sea $u_h \in \overline{U}_h$.*

(1) Si $-\Delta_h u_h(x_{ij}) \leq 0$ para todo $x_{ij} \in \Omega_h$, entonces $\max_{x\in\Omega_h} u_h(x) \leq \max_{x\in\partial\Omega_h} u_h(x)$.

(2) Si $-\Delta_h u_h(x_{ij}) \geq 0$ *para todo* $x_{ij} \in \Omega_h$*, entonces* $\min_{x\in\Omega_h} u_h(x) \geq \min_{x\in\partial\Omega_h} u_h(x)$.

Demostración. Vamos a llevar la condición

$$\max_{x_{ij}\in\Omega_h} u_h(x_{ij}) > \max_{x_{ij}\in\partial\Omega_h} u_h(x_{ij}) \tag{3.11}$$

a una contradicción, pues hay dos casos:

(i) La función $u_h|_{\Omega_h}$ es constante. En este caso, para cualquier punto $x^\star_{ij} \in \Omega_h$ tal que existe un punto

$$x \in \left\{x^\star_{i-1,j}, x^\star_{i+1,j}, x^\star_{i,j-1}, x^\star_{i,j+1}\right\}$$

con $x \in \partial\Omega_h$, se tiene

$$0 \geq -h^2\Delta_h u_h(x^\star_{ij}) = 4u_h(x^\star_{ij}) - u_h(x^\star_{i-1,j}) - u_h(x^\star_{i+1,j}) - u_h(x^\star_{i,j-1}) - u_h(x^\star_{i,j+1}) > 0,$$

(ii) La función $u_h|_{\Omega_h}$ no es constante. En este caso, existe un punto $x^\star_{ij} \in \Omega_h$ donde $u_h|_{\Omega_h}$ alcanza su máximo, y en particular un punto

$$x \in \left\{x^\star_{i-1,j}, x^\star_{i+1,j}, x^\star_{i,j-1}, x^\star_{i,j+1}\right\}$$

con

$$u_h(x) < u_h(x^\star_{ij}). \tag{3.12}$$

En este caso también vemos

$$0 \geq -h^2\Delta_h u_h(x^\star_{ij}) = 4u_h(x^\star_{ij}) - u_h(x^\star_{i-1,j}) - u_h(x^\star_{i+1,j}) - u_h(x^\star_{i,j-1}) - u_h(x^\star_{i,j+1}) > 0,$$

□

Si aplicamos el último Lema a la diferencia $u_h - v_h$, obtenemos el siguiento resultado.

Corolario 69. *Sean* $u_h, v_h : \overline{\Omega}_h \to \mathbb{R}$ *dos funciones con* $-\Delta_h u_h(x_{ij}) \leq -\Delta_h v_h(x_{ij})$ *para todo* $x_{ij} \in \Omega_h$ *y* $u_h(x_{ij}) \leq v_h(x_{ij})$ *para todo* $x_{ij} \in \partial\Omega_h$*. Entonces* $u_h(x_{ij}) \leq v_h(x_{ij})$ *para todo* $x_{ij} \in \overline{\Omega}_h$. □

Lema 70. *Existe una constante* $C_e > 0$ *independiente de* h*, tal que para todo* $v_h \in U_h^0$ *se tiene*

$$\|v_h\|_{\infty,\overline{\Omega}_h} \leq C_e\|\Delta_h v_h\|_{\infty,\Omega_h}.$$

Demostración. Sea

$$w(x,y) := \|\Delta_h v_h\|_\infty \frac{4 - x^2 - y^2}{4}$$

y definimos $w_h := [w]_{\overline{\Omega}_h}$. Calculamos

$$\begin{aligned}
-\Delta_h w_h(x_{ij}) &= \frac{\|\Delta_h v_h\|_\infty}{4}\Big(4(4-x_i^2-y_j^2)-(4-x_{i+1}^2-y_j^2)-(4-x_{i-1}^2-y_j^2)\\
&\qquad -(4-x_i^2-y_{i+1}^2)-(4-x_i^2-y_{i-1}^2)\Big)\\
&= \frac{\|\Delta_h v_h\|_\infty}{4}\Big(4(4-x_i^2-y_j^2)-(4-(x_i+h)^2-y_j^2)-(4-(x_i-h)^2-y_j)^2\\
&\qquad -(4-x_i^2-(y_j+h)^2)-(4-x_i^2-(y_j-h)^2)\Big)\\
&= \|\Delta_h v_h\|_\infty.
\end{aligned}$$

Eso implica $-\Delta_h v_h(x_{ij}) \le -\Delta_h w_h(x_{ij})$ para todo $x_{ij} \in \Omega_h$. Además, $v_h(x_{ij}) = 0 \le w_h(x_{ij})$ para $x_{ij} \in \partial\Omega_h$. Con el Corolario 69 concluimos $v_h(x_{ij}) \le w_h(x_{ij})$. Aplicando el mismo argumento con la función $-w_h$ muestra que $-w_h(x_{ij}) \le v_h(x_{ij})$ para todo $x_{ij} \in \overline{\Omega}_h$. Concluimos que

$$\|v_h\|_{\infty,\overline{\Omega}_h} \le \|w_h\|_{\infty,\overline{\Omega}_h} = C_e\|\Delta_h v_h\|_{\infty,\Omega_h}.$$

□

Concluimos entonces que el operador $L_h : U_h^0 \to U_h$, respectivamente la matriz A_h, son invertible, y que

$$\|A_h^{-1}\|_\infty \le C_e,$$

uniforme en h. Para demostrar este resultado, hemos usado el siguiente *principio del máximo discreto*,

$$A_h u \le 0 \Rightarrow u \le 0.$$

Arriba, y a partir de ahora, la desigualdad $\le$ para vectores y matrices se entiende por componentes.

Definición 71. *Una matriz $A \in \mathbb{R}^{n\times n}$ se llama*

(i) matriz L_0, si $a_{ij} \le 0$ para todo $i \ne j$,

(ii) monotona inversa, si $Ax \le Ay \Rightarrow x \le y$.

(iii) matriz M, si A es L_0, invertible, y $A^{-1} \ge 0$.

Lema 72. *Una matriz $A \in \mathbb{R}^{n\times n}$ es monotona inversa si y solo si es invertible y $A^{-1} \ge 0$. En particular, las matrices M son las matrices L_0 monotonas inversas.*

Demostración. Si A es monotona inversa, entonces $Ax = 0$ implica $x = 0$. Por lo tanto, A es inyectiva, y por se cuadrada, invertible. También notamos que $0 \le e_j$, y por lo tanto $0 \le A^{-1}e_j$ para todo $j = 1,\dots,n$. Por otro lado, si $A^{-1} \ge 0$, entonces $0 \le A(y-x)$ implica $0 \le A^{-1}A(y-x) = y - x$. □

En la demostración de Lema 70 hemos usado la función $\psi(x,y) = \frac{4-x^2-y^2}{4}$, y $\psi := [\psi]_{\overline{\Omega}_h}$. Notamos $\psi|_{\Omega_h} > 0$, $\psi|_{\partial\Omega_h} > 0$, y hemos visto que $-\Delta_h\psi = 1$. Si anotamos con $\varphi_h \in U_h^0$ la función con $\varphi_h|_{\Omega_h} = \psi_h|_{\Omega_h}$, entonces vemos que $\varphi_h|_{\Omega_h} > 0$ y $-\Delta_h\varphi_h \geq 1$.

Lema 73. *Sea $A \in \mathbb{R}^{n\times n}$ una matriz L_0. Entonces A es monotona inversa si y solo si existe un vector $x \in \mathbb{R}^n$, $x > 0$, tal que $Ax > 0$. En este caso,*

$$\|A^{-1}\|_\infty \leq \frac{\|x\|_\infty}{\min_{j=1,\dots,n}(Ax)_j}.$$

Demostración. Si A es monotona inversa, entonces por Lema 72 A es invertible y $A^{-1} \geq 0$. En particular, existe un elemento de $A^{-1} > 0$, pues en el caso contrario A^{-1} no sería invertible. Por lo tanto, $x := A^{-1}(1,\dots,1)^\top$ cumple con lo requerido.

Por otro lado, sea x un vector que cumple con el enunciado. Mostraremos que A^{-1} existe y que $A^{-1} \geq 0$, pues Lema 72 implica entonces que A es monotona inversa. Vemos

$$A_{jj} = \frac{A_{jj}x_j}{x_j} = \frac{1}{x_j}\Big(\underbrace{\sum_{k=1}^{n} A_{jk}x_k}_{>0} - \sum_{\substack{k=1\\k\neq j}}^{n} \underbrace{A_{jk}}_{\leq 0}\underbrace{x_k}_{>0}\Big) > 0.$$

Si anotamos la parte diagonal de A con D, entonces vemos que D es invertible con $D^{-1} \geq 0$. Sea $P := D^{-1}(D-A)$. Dado que A es L_0, $D - A \geq 0$. Concluimos que

(i) $P \geq 0$,

(ii) $(I-P)x = D^{-1}Ax > 0$, o sea $Px < x$.

Ahora mostraremos que $(I-P)$ es invertible. Recordamos que si la serie de Neumann $\sum_{k=1}^{\infty} P^k$ converge, entonces su límite es $(I-P)^{-1}$. Para convergencia de la serie, será suficiente encontrar una norma $\|\cdot\|_*$ tal que $\|P\|_* < 1$. Definimos

$$\|y\|_* := \max_{i=1,\dots,n} \frac{|y_i|}{x_i}.$$

Notamos $\|P\|_* = \max_{\|y\|_*=1} \|Py\|_*$. Si $\|y\|_* = 1$ entonces $-x \leq y \leq x$, y con $P \geq 0$ concluimos $-Px \leq Py \leq Px$. Entonces,

$$\|Py\|_* = \max_{i=1,\dots,n} \frac{|(Py)_i|}{x_i} \leq \max_{i=1,\dots,n} \frac{|(Px)_i|}{x_i} = \max_{i=1,\dots,n} \frac{(Px)_i}{x_i} < \max_{i=1,\dots,n} \frac{x_i}{x_i} = 1.$$

Ahora mostraremos que $A^{-1} \geq 0$. Primero notamos que la última desigualdad es equivalente a $A^{-1}D \geq 0$, pues $D \geq 0$. Calculamos

$$A^{-1}D = (D^{-1}A)^{-1} = (I-P)^{-1} = \sum_{k=1}^{\infty} P^k \geq 0,$$

pues $P \geq 0$. Finalmente mostramos la desigualdad. Notamos que

$$Ax \geq \min_{i=1,\dots,n} (Ax)_i \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top,$$

y dado que $A^{-1} \geq 0$ y $Ax > 0$ obtenemos

$$\frac{x}{\min_{i=1,\dots,n}(Ax)_i} \geq A^{-1} \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top,$$

o sea

$$\|A^{-1} \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top\|_\infty \leq \frac{\|x\|_\infty}{\min_{i=1,\dots,n}(Ax)_i}.$$

Para $x \in \mathbb{R}^n$ se tiene

$$-\|x\|_\infty \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top \leq x \leq \|x\|_\infty \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top,$$

y dado que $A^{-1} \geq 0$ se obtiene

$$-\|x\|_\infty A^{-1} \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top \leq A^{-1}x \leq \|x\|_\infty A^{-1} \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top,$$

de donde obtenemos

$$\|A^{-1}x\|_\infty \leq \|x\|_\infty \|A^{-1} \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix}^\top\|_\infty.$$

□

Por ejemplo, la matriz A_h de (3.10) es una matriz L_0 y es monotona inversa por Corolario 69, y en Lema 70 hemos mostrdado $\|A_h^{-1}\|_\infty \leq C_e$ con C_e independiente de h. La misma conclusion sigue con Lema 73, usando $x = \varphi_h$.

3.1.4. Condiciones de frontera de Neumann

Consideramos el problema

$$-u''(x) = f(x) \text{ para todo } x \in \Omega := (0,1), \tag{3.13a}$$
$$u(0) = u_0, \tag{3.13b}$$
$$u'(1) = u_1. \tag{3.13c}$$

Usamos las mismas mallas que antes, $h = 1/n$ y

$$\begin{aligned} \Omega_h &:= \{h, 2h, \dots, (n-1)h = 1-h\}, \\ \overline{\Omega}_h &:= \{0, h, 2h, \dots, 1-h, 1\}, \\ \partial\Omega_h &:= \overline{\Omega}_h \setminus \Omega_h = \{0, 1\}. \end{aligned}$$

La condición (3.13c) la podriamos discretizar por

$$u'(1) = \frac{u(1) - u(1-h)}{h} + \mathcal{O}(h).$$

Sin embargo, es una discretización de primer orden, mientras la discretización $-\Delta_h$ en el interior es de segundo orden. Si finalmente queremos un método de segundo orden, necesitaremos una discretización de la condición de Neumann de segundo orden. Usaremos la discretización

$$u'(1) = \frac{u(1+h) - u(1-h)}{2h} + \mathcal{O}(h^2)$$

para expresar

$$u(1+h) = 2hu_1 + u(1-h) + \mathcal{O}(h^3)$$

3.2. Diferencias finitas para problemas parabólicos

Sean $\Omega = (0,1)$ y $J = (0,T)$ intervalos. Consideramos el problema de hallar una función $u(x,t)$ tal que

$$\frac{\partial}{\partial t}u(x,t) - \frac{\partial^2}{\partial x^2}u(x,t) = f(x,t) \quad \text{para todo } (x,t) \in \Omega \times J,$$

donde f es una función dada. Para obtener única solución, tenemos que proveer condiciones de frontera en el espacio,

$$u(0,t) = u(1,t) = 0 \quad \text{para todo } t \in (0,T),$$

y una condición inicial en el tiempo

$$u(x,0) = u_0(x) \quad \text{para todo } x \in \Omega.$$

Este problema es un modelo para la evolución de la temperatura $u(x,t)$ de un cuerpo Ω en el tiempo $(0,T)$, donde g es la temperatura inicial y f es una fuente de calor. Es por eso que consideramos x como la variable espacial y t la variable temporal. Si $f = 0$, entonces la solución exacta es

$$u(x,t) = 2\sum_{j=1}^{\infty} e^{-(j\pi)^2 t} \sin(j\pi x) \int_0^1 u_0(y) \sin(j\pi y)\, dy. \tag{3.14}$$

Notamos entonces que la ecuación diferencial contiene un problema de condición de frontera *en el espacio*, y un problema de valor inicial *en el tiempo*. Para aproximar la derivada espacial $\partial^2/\partial x^2$, vamos a usar las diferencias finitas Δ_h con $h = 1/n > 0$ sobre la malla espacial

$$\Omega_h := \{x_1, x_2, \ldots, x_{n-1}\} = \{h, 2h, \ldots, (n-1)h = 1-h\}.$$

Eso nos lleva a la semi-discretización

$$\frac{\partial}{\partial t}u(x_i,t) - \Delta_h u(x_i,t) = f(x_i,t) \quad \text{para todo } x_i \in \Omega_h.$$

Para la discretización en el tiempo, eligimos un paso temporal $k = T/m$ y puntos $t_\ell = \ell k$, $\ell = 0,\ldots,m$, y

$$\frac{\partial}{\partial t}u(x_i,t_\ell) \approx \frac{u(x_i,t_{\ell+1}) - u(x_i,t_\ell)}{k}.$$

Anotamos $u(x_i,t_\ell) = u_i^\ell$. Dependiente de donde evaluamos $\Delta_h u(x_i,t)$, obtenemos dos métodos diferentes.

(1) **Euler explícito:**

$$\frac{u_i^{\ell+1} - u_i^\ell}{k} - \frac{u_{i-1}^\ell - 2u_i^\ell + u_{i+1}^\ell}{h^2} = f_i^\ell.$$

Este método es explicito en el tiempo, pues

$$u_i^{\ell+1} = u_i^\ell + kf_i^\ell + k\frac{u_{i-1}^\ell - 2u_i^\ell + u_{i+1}^\ell}{h^2}.$$

(2) **Euler implícito:**

$$\frac{u_i^{\ell+1} - u_i^\ell}{k} - \frac{u_{i-1}^{\ell+1} - 2u_i^{\ell+1} + u_{i+1}^{\ell+1}}{h^2} = f_i^{\ell+1}.$$

Notamos que si usamos la matriz de diferencias finitas para u'' en una dimension

$$A_h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & 0 & \ldots & 0 \\ -1 & 2 & -1 & 0 & 0 \\ & & \ddots & & \\ 0 & \ldots & -1 & 2 & -1 \\ 0 & \ldots & 0 & -1 & 2 \end{pmatrix} \in \mathbb{R}^{(n-1)\times(n-1)},$$

podemos escribir el método de Euler explícito como

$$\underline{u}_h^{\ell+1} = \underline{u}_h^\ell + k(\underline{f}_h^\ell - A_h\underline{u}_h^\ell), \quad \text{para } \ell \geq 0, \tag{3.15}$$

y el método de Euler implícito como

$$(I + kA_h)\underline{u}_h^{\ell+1} = \underline{u}_h^\ell + k\underline{f}_h^{\ell+1}, \quad \text{para } \ell \geq 0.$$

Recordamos brevemente la necesidad de tener métodos implícitos y consideramos la EDO

$$\begin{aligned} y'(t) &= \lambda y(t), \\ y(0) &= 1 \end{aligned}$$

con $\lambda < 0$. La solución exacta es $y(t) = e^{\lambda t}$, y por lo tanto $\lim_{t\to\infty} y(t) = 0$. El método de Euler explícito en este caso es $y_{\ell+1} = y_\ell + k\lambda y_\ell$, y conlcuimos $|y_\ell| \leq |y_0||1+k\lambda|^\ell$. Para obtener $|y_\ell| \to 0$ es suficiente tener $-2 < k\lambda < 0$, es decir,

$$k < -\frac{2}{\lambda}.$$

Si λ es grande en valor absoluto, k tiene que ser muy pequeño. El mismo efecto podemos observar con el método (3.15). La solución exacta (3.14) satisface $\lim_{t\to\infty} u(x,t) = 0$. Usamos la base ortonormal de $\mathbb{R}^{n-1}$ de vectores propios de A_h con autovalores λ_j, y notamos

$$\|\underline{u}_h^\ell\|_2^2 = \|(I - kA_h)^\ell \underline{u}_h^0\|_2^2 = \sum_{j=1}^{n-1} (1 - k\lambda_j)^{2\ell} \alpha_j^2.$$

Notamos que para $\|\underline{u}_h^\ell\|_2 \to 0$, es necesario $|1 - k\lambda_j| < 1$ para todos los autovalores de A_h.

Lema 74. *Los autovalores de $A_h \in \mathbb{R}^{(n-1)\times(n-1)}$ son*

$$\lambda_j = \frac{2}{h^2}\left(1 - \cos(j\pi h)\right), \quad j = 1, \dots n-1$$

Demostración. Sea $\underline{u}_j \in \mathbb{R}^{(n-1)}$ dado por $\underline{u}_{j,k} = \sin(j\pi hk)$. Entonces, para $2 \leq k \leq n-2$,

$$\begin{aligned} \left(A_h \underline{u}_j\right)_k &= \frac{1}{h^2}\left[-\underline{u}_{j,k-1} + 2\underline{u}_{j,k} - \underline{u}_{j,k+1}\right] \\ &= \frac{1}{h^2}\left[-\sin\left(j\pi hk - j\pi h\right) + 2\sin(j\pi hk) - \sin(j\pi hk + j\pi h)\right] \\ &= \frac{1}{h^2}\left[-2\sin(j\pi hk)\cos(j\pi h) + 2\sin(j\pi hk)\right]. \end{aligned}$$

Los casos $k = 1$ y $k = n-1$ siguen de la misma manera. □

La condición $|1 - k\lambda_j| < 1$ se transforma entonces en

$$\frac{k}{h^2} < \frac{2}{2[1 - \cos\left(\pi(n-1)/n\right)]} \approx \frac{1}{2}.$$

En la practica, eso significa $k < h^2/2$, es decir, estabilidad de método explícito exige que la resolución temporal tiene que ser el cuadrado de la resolución espacial.
Para establecer una teoría de convergencia, consideramos esquemas de la forma

$$\begin{aligned} u_h^0 &= [u_0]_{\Omega_h}, \\ u_h^\ell &= Q_{h,k} u_h^{\ell-1} + g^{\ell-1}, \quad \text{para } \ell \geq 1. \end{aligned} \tag{3.16}$$

Definición 75 (Consistencia y Estabilidad)**.** *Sean $h, k > 0$ y $Q_{h,k} : U_h \to U_h$ una aplicación lineal.*

(1) Se dice que el método es ***consistente*** *con $\partial_t - \Delta$ de orden $(p_1, p_2) \in \mathbb{N}^2$ para u, si existe una constante $C_c > 0$ tal que para el error de truncamiento*

$$\tau^\ell := [u(t_\ell)]_{\Omega_h} - Q_{h,k}[u(t_{\ell-1})]_{\Omega_h} - g^{\ell-1}, \quad 1 \leq \ell \leq m,$$

se tiene

$$\max_{\ell=1,\dots,m} \|\tau^\ell\|_{\Omega_h} \leq C_c k(h^{p_1} + k^{p_2}).$$

(2) Se dice que el método es ***estable****, si existe una constante $M > 0$ tal que*

$$\max_{\ell=1,\dots,m} \|Q_{h,k}^\ell\|_\infty \leq M,$$

con la norma inducida

$$\|A\|_\infty := \sup_{u_h \in U_h} \frac{\|Au_h\|_{\infty,\Omega_h}}{\|u_h\|_{\infty,\Omega_h}}.$$

□

Teorema 76 (Lax-Richtmyer)**.** *Sea el método* (3.16) *consistente con $\partial_t - \Delta$ de orden (p_1, p_2) para u y estable. Entonces, el método es convergente de orden (p_1, p_2), es decir*

$$\max_{\ell=0,\dots,m} \|u_h^\ell - [u(t_\ell)]_{\Omega_h}\|_{\infty,\Omega_h} \leq C(h^{p_1} + k^{p_2}).$$

Demostración. Definimos $e^\ell := [u(t_\ell)]_{\Omega_h} - u_h^\ell$ el error al tiempo t_ℓ. Entonces $e^0 = 0$ y para $\ell \geq 1$

$$\begin{aligned} e^\ell &= [u(t_\ell)]_{\Omega_h} - Q_{h,k}u_h^{\ell-1} - g^{\ell-1} \\ &= [u(t_\ell)]_{\Omega_h} + Q_{h,k}\left([u(t_{\ell-1})]_{\Omega_h} - u_h^{\ell-1}\right) - Q_{h,k}[u(t_{\ell-1})]_{\Omega_h} - g^{\ell-1} \\ &= Q_{h,k}e^{\ell-1} + \tau^\ell. \end{aligned}$$

Concluimos

$$e^\ell = \sum_{j=0}^{\ell-1} Q_{h,k}^j \tau^{\ell-j},$$

y tomando normas

$$\|e^\ell\|_{\infty,\Omega_h} \leq \sum_{j=0}^{\ell-1} \|Q_{h,k}^j\|_\infty \cdot \|\tau^{\ell-j}\|_{\infty,\Omega_h} \leq MmC_c k(h^{p_1} + k^{p_2}) = MTC_c(h^{p_1} + k^{p_2}).$$

□

Vamos a analizar los dos métodos de Euler.

(1) **Euler explícito:** Dado que

$$u_h^\ell = u_h^{\ell-1} + k(f_h^{\ell-1} + \Delta_h u_h^{\ell-1}) = (1 + k\Delta_h)u_h^{\ell-1} + kf_h^{\ell-1},$$

el Euler explícito es de la forma (3.16) con $Q_{h,k} = 1 + k\Delta_h$ y $g^{\ell-1} = kf_h^{\ell-1}$. Para analizar el error de truncamiento notamos que si $u_{xxxx} \in C(\overline{\Omega \times J})$, entonces

$$\|\Delta_h[u(t_{\ell-1})]_{\Omega_h} - [\Delta u(t_{\ell-1})]_{\Omega_h}\|_{\infty,\Omega_h} \leq C_1 h^2,$$

y si $u_{tt} \in C(\overline{\Omega \times J})$, entonces

$$\|([u(t_\ell)]_{\Omega_h} - [u(t_{\ell-1})]_{\Omega_h})/k - [\partial_t u(t_{\ell-1})]_{\Omega_h}\|_{\infty,\Omega_h} \leq C_2 k.$$

Concluimos

$$\begin{aligned}
\|\tau^\ell\|_{\infty,\Omega_h} &= \|[u(t_\ell)]_{\Omega_h} - (1 + k\Delta_h)[u(t_{\ell-1})]_{\Omega_h} - kf_h^{\ell-1}\|_{\infty,\Omega_h} \\
&\leq \|[u(t_\ell)]_{\Omega_h} - [u(t_{\ell-1})]_{\Omega_h} - k[\Delta u(t_{\ell-1})]_{\Omega_h} - kf_h^{\ell-1}\|_{\infty,\Omega_h} + C_1 kh^2 \\
&= \|[u(t_\ell)]_{\Omega_h} - [u(t_{\ell-1})]_{\Omega_h} - k[\partial_t u(t_{\ell-1})]_{\Omega_h}\|_{\infty,\Omega_h} + C_1 kh^2 \\
&\leq C_2 k^2 + C_1 kh^2 \leq \text{máx}(C_1, C_2)k(h^2 + k),
\end{aligned}$$

es decir, el método es consistente de orden $(2, 1)$. Para analizar la estabilidad, notamos que si $v_h \in U_h$, entonces

$$Q_{h,k}v_h = v_h + \frac{k}{h^2}\begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & & \ddots & & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} v_h = \begin{pmatrix} 1-2r & r & & & \\ r & 1-2r & r & & \\ & & \ddots & & \\ & & r & 1-2r & r \\ & & & r & 1-2r \end{pmatrix} v_h,$$

donde $r = \frac{k}{h^2}$. Notamos que para $A \in \mathbb{R}^{(n-1)\times(n-1)}$ se tiene

$$\|A\|_\infty = \max_{j=1,\ldots,n-1} \sum_{k=0}^{n-1} |A_{jk}|.$$

Concluimos que $\|Q_{h,k}\|_\infty = |1 - 2r| + 2r$, y $0 < r < 1/2$ implica $\|Q_{h,k}\|_\infty \leq 1$, es decir,

$$\|Q_{h,k}^\ell\|_\infty \leq \|Q_{h,k}\|_\infty^\ell \leq 1^\ell = 1.$$

Corolario 77. *Si $u_{xxxx}, u_{tt} \in C(\overline{\Omega \times J})$ y $\frac{k}{h^2} \leq \frac{1}{2}$, entonces el método de Euler explícito es convergente de orden $(2, 1)$.* □

(2) **Euler implícito:** Este método se define por

$$(I - k\Delta_h)u_h^\ell = u_h^{\ell-1} + kf_h^\ell, \quad \ell = 1, \ldots, m.$$

Notamos que

$$I - k\Delta_h = \begin{pmatrix} 1+2r & -r & & & \\ -r & 1+2r & -r & & \\ & & \ddots & & \\ & & -r & 1+2r & -r \\ & & & -r & 1+2r \end{pmatrix},$$

donde $r = \frac{k}{h^2}$. Concluimos que $I - k\Delta_h$ es diagonal dominante y por lo tanto invertible. Concluimos que el método se puede escribir como

$$u_h^\ell = (I - k\Delta_h)^{-1}u_h^{\ell-1} + k(I - k\Delta_h)^{-1}f_h^\ell, \quad \ell = 1, \ldots, m,$$

es decir, en la forma (3.16) con $Q_{h,k} = (I - k\Delta_h)^{-1}$ y $g^{\ell-1} = k(I - k\Delta_h)^{-1}f_h^\ell$. Analizemos primero la estabilidad. Si $w_h = Q_{h,k}v_h$, entonces $v_h = (I - k\Delta_h)w_h$. Supongamos que $\|w_h\|_\infty = w_{h,i}$ con $1 < i < n-1$, entonces $(1+2r)w_{h,i} - r(w_{h,i-1} - w_{h,i+1}) = v_{h,i}$, y concluimos

$$\|w_h\|_\infty = w_{h,i} = \frac{r}{1+2r}(w_{h,i-1} + w_{h,i+1}) + \frac{v_{h,i}}{1+2r} \leq \frac{2r}{1+2r}\|w_h\|_\infty + \frac{1}{1+2r}\|v_h\|_\infty.$$

Es decir,

$$\|w_h\|_\infty \leq \|v_h\|_\infty, \quad \text{o bien} \quad \|Q_{h,k}\|_\infty \leq 1.$$

Como antes, concluimos que el método es estable con constante $M = 1$. Para analizar la consistencia, consideramos

$$\widehat{\tau}^\ell := (I - k\Delta_h)\tau^\ell = (I - k\Delta_h)[u(t_\ell)]_{\Omega_h} - [u(t_{\ell-1})]_{\Omega_h} - k[f(t_\ell)]_{\Omega_h},$$

y como en el caso del Euler explícito se demuestra, usando Taylor, que $\|\widehat{\tau}^\ell\|_\infty \leq Ck(h^2+k)$. Usando la estabilidad, concluimos

$$\|\tau^\ell\|_\infty = \|Q_{h,k}\widehat{\tau}^\ell\|_\infty \leq \|\widehat{\tau}^\ell\|_\infty \leq Ck(h^2+k),$$

es decir, el método es consistente de orden $(2,1)$.

Corolario 78. *Si $u_{xxxx}, u_{tt} \in C(\overline{\Omega \times J})$, entonces el método de Euler implícito es convergente de orden $(2,1)$.* □

Para obtener un método que es de orden $(2,2)$, necesitaremos aproximar la derivada temporal por una diferencia de segundo orden. La primera idea será

$$\frac{\partial}{\partial t}u(x,t_\ell) \approx \frac{u(x,t_{\ell+1}) - u(x,t_{\ell-1})}{2k}.$$

Esta aproximación tiene dos desventajas: primero, tenemos que tener en memoria los valores de u_h en **dos** tiempos, es decir, el costo de almacenamiento es el doble comparado con los métodos de Euler. Segundo, no es tan claro como empezar el método. Para evitar eso, introduciremos los tiempos $t_{\ell-1/2} := t_\ell - k/2$ y aproximaremos

$$\frac{\partial}{\partial t}u(x,t_{\ell-1/2}) \approx \frac{u(x,t_\ell) - u(x,t_{\ell-1})}{k}.$$

Para la parte espacial de la EDP escribiremos

$$\Delta u(x_j,t_{\ell-1/2}) \approx \frac{u(x_{j-1},t_{\ell-1/2}) - 2u(x_j,t_{\ell-1/2}) + u(x_{j+1},t_{\ell-1/2})}{h^2}.$$

Para evitar de tener que calcular valores de u_h en $t_{\ell-1/2}$, usaremos otra aproximación

$$u(x_j,t_{\ell-1/2}) \approx \frac{u(x_j,t_{\ell+1}) + u(x_j,t_{\ell-1})}{2},$$

y asi llegamos al método de **Crank-Nicolson**,

$$\frac{u_j^\ell - u_j^{\ell-1}}{k} - \frac{\left(u_{j-1}^\ell - 2u_j^\ell + u_{j+1}^\ell\right) + \left(u_{j-1}^{\ell-1} - 2u_j^{\ell-1} + u_{j+1}^{\ell-1}\right)}{2h^2} = f_j^{\ell-1/2}.$$

3.3. Diferencias finitas para problemas hiperbólicos

En preparación.

Capítulo 4

Métodos multigrid

Dada $f : (0,1) \to \mathbb{R}$, consideramos el problema

$$\begin{aligned} &- u''(x) = f(x) \text{ para todo } x \in \Omega := (0,1), \\ &u(0) = u(1) = 0. \end{aligned}$$

En el capítulo anterior hemos desarollado métodos de diferencias finitas sobre mallas

$$\Omega_h := \{h, 2h, \ldots, (n-1)h = 1 - h\}$$

con resolución $h = n^{-1}$, que consisten en la resolución del sistema lineal

$$A_h u_h = f_h, \tag{4.1}$$

o sea

$$\frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} u_{h,1} \\ u_{h,2} \\ \vdots \\ u_{h,n-1} \end{pmatrix} = \begin{pmatrix} f_{h,1} \\ f_{h,2} \\ \vdots \\ f_{h,n-1} \end{pmatrix},$$

donde $f_{h,j} = f(x_j) = f(jh)$. Vamos a resolver el sistema (4.1) con el método amortiguado de Jacobi. El método amortiguado de Jacobi es un método iterativo: dado una amortiguación ω, a partir de un vector inicial $u_h^{(0)} \in \mathbb{R}^{n-1}$ calculamos una sucesión $u_h^{(1)}, u_h^{(2)}, \ldots$ a través de

$$\omega^{-1} D_h u_h^{(k+1)} = (\omega^{-1} D_h - A_h) u_h^{(k)} + f_h,$$

donde D_h es la parte diagonal de A_h. Notamos

$$u_h^{(k+1)} = u_h^{(k)} - \omega D_h^{-1}(A_h u_h^{(k)} - f_h).$$

Sabemos que el método de Jacobi converge si y solo si $\omega D_h^{-1}(\omega^{-1}D_h - A_h) = I_h - \omega D_h^{-1}A_h$ cumple con

$$\rho(I_h - \omega D_h^{-1}A_h) < 1.$$

Lema 79. *Los valores propios de $I_h - \omega D_h^{-1}A_h$ son*

$$\lambda_{h,k} = 1 - 2\omega \sin^2(k\pi h/2), \quad k = 1, \ldots, n-1,$$

y los vectores propios asociados son

$$w_{h,k} = \begin{pmatrix} \sin(k\pi h) \\ \sin(2k\pi h) \\ \vdots \\ \sin((n-1)k\pi h) \end{pmatrix}.$$

En particular, el método de Jacobi converge. □

Se puede mostrar que el parametro de amortiguación $\omega > 0$ optimal es

$$\omega_{\text{opt}} = \arg\min_{\omega>0} \rho(I_h - \omega D_h^{-1}A_h) = 1,$$

y el orden de convergencia en este caso es

$$\rho(I_h - D_h^{-1}A_h) = 1 - 2\sin^2(\pi h/2).$$

Sin embargo, nuestro objetivo no es optimizar el orden de convergencia, asi que sigamos usando un parametro de amortiguación ω. Los vectores propios $w_{h,k}$ con $\frac{n}{2} \leq k \leq n-1$ se llaman **frecuencias altas**, mientras los con $1 \leq k < \frac{n}{2}$ se llaman **frecuencias bajas**.

Sea $G_h = I_h - \omega D_h^{-1}A_h$. La solución exacta u_h de (4.1) satisface

$$G_h u_h = u_h - \omega D_h^{-1} f_h,$$

y por lo tanto vemos

$$G_h(u_h^{(k)} - u_h) = G_h u_h^{(k)} - u_h + \omega D_h^{-1} f_h = u_h^{(k+1)} - u_h,$$

o bien

$$u_h^{(k)} - u_h = G_h^k(u_h^{(0)} - u_h).$$

Consideramos dos casos extremos.

(i) El error inicial es de frecuencia muy baja, $u_h^0 - u_h = w_{h,1}$. Entonces $u_h^{(k)} - u_h = \lambda_{h,1}^k w_{h,1}$, y concluimos que

$$\|u_h^{(k)} - u_h\| = \mathcal{O}(\lambda_{h,1}^k) = \big(1 - 2\omega \sin^2(\pi h/2)\big)^k$$

(ii) El error inicial es de frecuencia muy alta, $u_h^0 - u_h = w_{h,n-1}$. Entonces $u_h^{(k)} - u_h = \lambda_{h,n-1}^k w_{h,n-1}$, y concluimos que

$$\|u_h^{(k)} - u_h\| = \mathcal{O}(\lambda_{h,n-1}^k) = \left(1 - 2\omega \sin^2(\pi(n-1)h/2)\right)^k.$$

Vemos que si usamos $\omega = 1/2$, entonces un error inicial de frecuencia muy baja se reduce por un factor ≈ 1 en cada paso (y eso define el orden bajo de convergencia asintótica), mientras un error inicial de frecuencia muy alta se reduce por un factor ≈ 0 en cada paso.

Según Lema 79, la matriz $G_h \in \mathbb{R}^{(n-1)\times(n-1)}$ tiene $n-1$ valores propios distintos, por lo tanto es diagonalizable y los vectores propios $w_{h,1}, \dots w_{h,n-1}$ son una base de $\mathbb{R}^{n-1}$. Si expresamos el error inicial en esta base, $u_h^{(0)} - u_h = \sum_{j=1}^{n-1} \alpha_j w_{h,j}$, entonces vemos

$$u_h^{(k)} - u_h = \sum_{j=1}^{n-1} \alpha_j \lambda_{h,j}^k w_{h,j}.$$

Generalizando los casos extremos de arriba, vemos que $\lambda_{h,j} < 1/2$ para las frecuencias altas, mientras $\lambda_{h,j} > 1/2$ para las frecuencias bajas. En otras palabras, aunque el error $u_h^{(k)} - u_h$ **no es mas pequeño** que $u_h^{(0)} - u_h$, seguramente es **mas suave**. Por lo tanto, el método de Jacobi amortiguado en este contexto se llama *iteración suavizante*. Despues de un par de pasos de una iteración suavizante, el error no se ha reducido mucho, pero es mas suave, y deberiamos seguir con otra iteración que tenga el efecto contrario, es decir, que reduzca rapidamente errores suaves. Obviamente, no tenemos acceso al error suave $u_h^{(k)} - u_h$, pero notamos que

$$A_h(u_h^{(k)} - u_h) = A_h u_h^{(k)} - f_h. \tag{4.2}$$

Es decir, el error suave $u_h^{(k)} - u_h$ resuelve el mismo sistema lineal con un lado derecho conocido. Sin embargo, al ser suave, el error suave puede representarse también sobre una malla con una resolución menor. Eso es la idea del método multigrid.

4.1. El método two-grid

La idea del método two-grid es reducir (4.2) a una malla de resolución $2h$. Por lo tanto, supongamos que $n_0 \in \mathbb{N}$, $n_1 = 2n_0$, y $h_\ell = n_\ell^{-1}$, y

$$\Omega_\ell := \{h_\ell, 2h_\ell, \dots, (n_\ell - 1)h_\ell = 1 - h_\ell\}$$

dos mallas. Sea $u_1^{(0)} \in \mathbb{R}^{n_1 - 1}$ un vector inicial, y

$$u_1^{(k+1)} = u_1^{(k)} - \frac{1}{2} D_1^{-1}(A_1 u_1^{(k)} - f_1)$$

la iteración de Jacobi amortiguado sobre la malla fina Ω_1. Sea $u_1^{(k)}$ la iteración despues de k pasos. Para reducir el sistema (4.2) a la malla gruesa Ω_0, introduciremos una transformación lineal, el **operador de restricción**, $I_{1,0} : \mathbb{R}^{n_1-1} \to \mathbb{R}^{n_0-1}$. La opción mas simple es

$$(I_{1,0}u_1)(x_j) = u_1(x_j) \qquad \text{para } x_j \in \Omega_0,$$

o sea

$$I_{1,0} = \begin{pmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & & & & & \end{pmatrix}$$

Sin embargo, esta restricción pierde información. Una restricciño mas adecuada será

$$(I_{1,0}u_1)(x_j) = \frac{u_1(x_j - h_1) + 2u_1(x_j) + u_1(x + h_1)}{4}.$$