



Universidad de Valladolid

Facultad de Ciencias

TRABAJO FIN DE GRADO

Grado en Matemáticas

Matrices no negativas: algunos resultados y aplicaciones.

Autora: Yaiza Fernández de la Fuente

Tutora: María Paz Calvo Cabrero

Índice

1. Matrices no negativas. Resultados teóricos.	5
1.1. Normas y radio espectral de una matriz	5
1.2. Matrices irreducibles	11
1.3. Matrices diagonalmente dominantes	15
1.4. Matrices no negativas. El teorema de Perron-Frobenius	17
2. Análisis de convergencia de los métodos iterativos clásicos	26
2.1. Descripción de los métodos	26
2.1.1. Método de Jacobi	26
2.1.2. Método de Gauss-Seidel	27
2.1.3. Método SOR (<i>Método de sobrerrelajación sucesiva</i>)	27
2.2. El teorema de Stein-Rosenberg	29
2.3. Determinación del factor de relajación óptimo de SOR	32
3. Aplicaciones	39
3.1. El cálculo del PageRank de Google	39
3.2. Análisis de resultados de competiciones deportivas	45
3.3. Discretización del laplaciano en un dominio con forma de L	48
Bibliografía	55
Códigos de MATLAB	57
A. Cálculo del PageRank de Google	57
B. Cálculo del ranking de la Champions League	59
C. Métodos iterativos clásicos	61

Introducción

El Trabajo Fin de Grado que presentamos se dedica al estudio de algunos resultados para matrices no negativas que son de aplicación tanto en el estudio de las matrices estocásticas, como en el análisis de la convergencia de los métodos iterativos clásicos. En la Sección 1 se introducen las definiciones y resultados parciales necesarios para concluir con la demostración del Teorema de Perron-Frobenius para matrices irreducibles.

La Sección 2 tiene un doble objetivo. Por un lado, nos proponemos comparar la velocidad de convergencia de los diferentes métodos iterativos clásicos que se utilizan para resolver sistemas lineales que surgen tras la discretización espacial de ecuaciones elípticas y, por otro, establoceremos algún resultado que permite estimar el parámetro óptimo del método SOR. Las matrices asociadas a este tipo de sistemas tienen la propiedad de ser dispersas (o *sparse*), es decir, solo hay un pequeño porcentaje de elementos no nulos que además suelen seguir un patrón determinado y con frecuencia tienen otras propiedades importantes como la irreducibilidad y el carácter no negativo.

En la Sección 3 se recogen tres aplicaciones que ilustran el interés de los resultados teóricos presentados en las Secciones 1 y 2. El teorema de Perron-Frobenius es un resultado que puede tener aplicaciones a diferentes ámbitos de la ciencia y la tecnología. Algunas de estas aplicaciones son el análisis de las características demográficas de una población, la teoría de juegos, la distribución del poder en una red social, el desarrollo de buscadores como Google y, algo que nunca me habría imaginado, el análisis de resultados de ciertas competiciones deportivas. Como aplicaciones directas del teorema de Perron-Frobenius hemos decidido estudiar el algoritmo que calcula el PageRank de Google, en el cual aparecen las matrices estocásticas. También, utilizando un algoritmo similar al de PageRank estableceremos clasificaciones (o *rankings*) de la última Champions League y compararemos los resultados obtenidos con los resultados reales.

Para ilustrar los resultados de la Sección 2, consideraremos el problema de calcular el estado estacionario de temperaturas de una placa con forma de L, sometida a una fuente de calor. Para ello discretizaremos el Laplaciano mediante diferencias finitas de segundo orden y resolveremos los sistemas lineales resultantes con los métodos de Jacobi, Gauss-Seidel y SOR.

En el Apéndice se incluyen los programas en Matlab que hemos implementado para obtener los resultados de las aplicaciones de la Sección 3.

1. Matrices no negativas. Resultados teóricos.

En esta sección vamos a recordar algunos conceptos básicos como son la norma espectral, el radio espectral, las matrices diagonalmente dominantes e introduciremos otros conceptos nuevos como son la noción de irreducibilidad y convergencia de una matriz, las matrices no negativas y relacionaremos algunas de estas propiedades con los grafos. Además, veremos algunas propiedades interesantes y daremos cotas para el radio espectral para diferentes tipos de matrices basándonos principalmente en el ya conocido Teorema de Gerschgorin y en el Teorema de Perron-Frobenius.

Durante todo el trabajo, salvo que indiquemos otra cosa, consideraremos matrices complejas de tamaño $n \times n$.

1.1. Normas y radio espectral de una matriz

En primer lugar, comenzamos recordando la definición de norma euclídea y sus propiedades, que recogeremos en el [Teorema 1.1](#).

Definición 1.1. Sea $\mathbf{x} \in \mathbb{C}^n$. La norma euclídea de \mathbf{x} viene dada por

$$\|\mathbf{x}\| = \sqrt{\mathbf{x}^* \mathbf{x}} = \sqrt{\sum_{i=1}^n |x_i|^2}.$$

Teorema 1.1. Sean $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ se tiene que

- 1) $\|\mathbf{x}\| > 0$ para todo $\mathbf{x} \neq \mathbf{0}$,
- 2) $\|\mathbf{x}\| = 0$ si y solo si $\mathbf{x} = \mathbf{0}$,
- 3) Si $\alpha \in \mathbb{C}$ entonces $\|\alpha \mathbf{x}\| = |\alpha| \|\mathbf{x}\|$,
- 4) $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ para cualquier \mathbf{x} e \mathbf{y} . (**Desigualdad triangular**)

A continuación definimos el concepto de radio espectral de una matriz, que desde el punto de vista geométrico es el radio del menor disco centrado en el origen que contiene a todos los autovalores de la matriz.

Definición 1.2. Sea A una matriz con autovalores λ_i , $1 \leq i \leq n$. Definimos el radio espectral de la matriz A como

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i|.$$

En algunos casos, por simplicidad, denotaremos por $\sigma(A)$ al conjunto de autovalores de la matriz A . Lo llamaremos espectro de A .

De manera análoga a los vectores, recordamos la definición de norma espectral o euclídea de una matriz y sus propiedades.

Definición 1.3. Sea A una matriz. Definimos la norma espectral de A de la siguiente manera

$$\|A\| = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|},$$

donde $\|\cdot\|$ en la parte derecha de la igualdad denota la norma euclídea de \mathbb{C}^n .

Teorema 1.2. Sean A y B dos matrices. Si denotamos por O a la matriz nula, se verifica que

- 1) $\|A\| > 0$ para toda $A \neq O$,
- 2) $\|A\| = 0$ si y solo si $A = O$,
- 3) Si $\alpha \in \mathbb{C}$ entonces $\|\alpha A\| = |\alpha| \|A\|$,
- 4) $\|A + B\| \leq \|A\| + \|B\|$,
- 5) $\|I\| = 1$,
- 6) $\|AB\| \leq \|A\| \|B\|$.
- 7) Para cualquier $\mathbf{x} \in \mathbb{C}^n$ tenemos que $\|A\mathbf{x}\| \leq \|A\| \|\mathbf{x}\|$, y existe algún vector $\mathbf{x} \neq \mathbf{0}$ para el que se da la igualdad.

En general, el cálculo preciso del radio espectral de una matriz es complicado por lo que vamos a dar algunas cotas. Como corolario de este teorema, vamos a relacionar la norma espectral y el radio espectral de una matriz.

Corolario 1.1. Sea A una matriz. Se verifica que

$$\rho(A) \leq \|A\|.$$

Demostración. Sea λ un autovalor cualquiera de A y $\mathbf{x} \neq \mathbf{0}$ un autovector asociado, entonces $A\mathbf{x} = \lambda\mathbf{x}$. Tomando normas y aplicando el [Teorema 1.2](#) y el tercer apartado del [Teorema 1.1](#) tenemos que

$$\|A\| \|\mathbf{x}\| \geq \|A\mathbf{x}\| = \|\lambda\mathbf{x}\| = |\lambda| \|\mathbf{x}\|$$

y, por tanto,

$$\|A\| \geq |\lambda|.$$

Como esto es cierto para todos los autovalores de A y $\rho(A) = \max_{1 \leq i \leq n} |\lambda_i|$, es claro que

$$\|A\| \geq \rho(A).$$

□

De ahora en adelante denotaremos $A^* = \overline{A^T}$. Vamos a definir el concepto de matriz hermítica y, al igual que antes, relacionaremos el radio espectral y la norma espectral pero ahora teniendo en cuenta que la matriz es hermítica.

Definición 1.4. Sea A una matriz. Se dice que es hermítica si es igual a su propia traspuesta conjugada, es decir, si $A^* = A$.

Teorema 1.3. Sea A una matriz. Entonces

$$\|A\| = \sqrt{\rho(A^*A)}.$$

Demostración. Es claro que la matriz A^*A es hermítica y semidefinida positiva, es decir,

$$\begin{aligned} (A^*A)^* &= A^*A \\ \mathbf{x}^* A^* A \mathbf{x} &= \|A\mathbf{x}\|^2 \geq 0 \quad \text{para cualquier } \mathbf{x}. \end{aligned}$$

Por ser A^*A hermítica, tiene una base ortonormal de autovectores $\{\mathbf{x}_i\}_{i=1}^n$, es decir,

$$A^*A\mathbf{x}_i = \lambda_i\mathbf{x}_i \quad \text{con} \quad 0 \leq \lambda_1 \leq \dots \leq \lambda_n = \rho(A^*A).$$

Como el conjunto es ortonormal se tiene que

$$\mathbf{x}_i^* \mathbf{x}_j = \begin{cases} 0 & \text{si } i \neq j, \\ 1 & \text{si } i = j. \end{cases}$$

Consideramos el vector no nulo $\mathbf{x} = \sum_{i=1}^n c_i \mathbf{x}_i$, entonces

$$\left(\frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|} \right)^2 = \frac{\mathbf{x}^* A^* A \mathbf{x}}{\mathbf{x}^* \mathbf{x}} = \frac{\sum_{i,j=1}^n c_i^* c_j \mathbf{x}_i^* A^* A \mathbf{x}_j}{\sum_{j=1}^n |c_j|^2} = \frac{\sum_{i,j=1}^n c_i^* c_j \mathbf{x}_i^* \lambda_j \mathbf{x}_j}{\sum_{j=1}^n |c_j|^2} = \frac{\sum_{i=1}^n |c_i|^2 \lambda_i}{\sum_{j=1}^n |c_j|^2},$$

luego

$$0 \leq \left(\frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|} \right)^2 = \frac{\sum_{i=1}^n |c_i|^2 \lambda_i}{\sum_{j=1}^n |c_j|^2} \leq \frac{\left(\sum_{i=1}^n |c_i|^2 \right) \lambda_n}{\sum_{j=1}^n |c_j|^2} = \lambda_n$$

y, por tanto, se tiene que $\|A\|^2 \leq \lambda_n$. En particular, si tomamos $c_1 = \dots = c_{n-1} = 0$ y $c_n = 1$, es decir, $\mathbf{x} = \mathbf{x}_n$

$$\|A\|^2 = \sup_{\mathbf{x} \neq 0} \left(\frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|} \right)^2 \geq \lambda_n = \rho(A^*A)$$

y, por tanto,

$$\|A\| = \sqrt{\rho(A^*A)}.$$

□

Corolario 1.2. Si A es una matriz hermítica, entonces

$$\|A\| = \rho(A).$$

Además, si $q_m(x)$ es un polinomio real en la variable x de grado m , entonces

$$\|q_m(A)\| = \rho(q_m(A)).$$

Demostración. Según el [Teorema 1.3](#) y como A es hermítica, es decir, $A^* = A$, tenemos que

$$\|A\|^2 = \rho(A^*A) = \rho(A^2) = [\rho(A)]^2.$$

Tomando raíces, como $\rho(A)$ debe ser un valor positivo tenemos la primera parte del resultado

$$\|A\| = \rho(A).$$

Por otra parte, dado $q_m(x) = \alpha_m x^m + \dots + \alpha_1 x + \alpha_0$ un polinomio real de grado m en x , mediante una simple comprobación, utilizando propiedades de la conjugación y la trasposición de matrices y el hecho de que $\alpha_m, \dots, \alpha_0 \in \mathbb{R}$ es fácil ver que la matriz $q_m(A)$ es hermítica

$$\begin{aligned} q_m(A)^* &= (\alpha_m A^m + \dots + \alpha_1 A + \alpha_0 I)^* = \bar{\alpha}_m (A^m)^* + \dots + \bar{\alpha}_1 A^* + \bar{\alpha}_0 I^* \\ &= \alpha_m A^m + \dots + \alpha_1 A + \alpha_0 I = q_m(A), \end{aligned}$$

y, por tanto,

$$\|q_m(A)\| = \rho(q_m(A)).$$

□

A continuación recordaremos el teorema de Gerschgorin, el cual da cotas para los autovalores de una matriz y permite deducir cotas para su radio espectral.

Teorema 1.4 (Gerschgorin). *Sea $A = [a_{i,j}]$ una matriz. Definimos*

$$\Lambda_i = \sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}|, \quad 1 \leq i \leq n.$$

Si λ es un autovalor de A , existe un entero r con $1 \leq r \leq n$ tal que

$$|\lambda - a_{r,r}| \leq \Lambda_r.$$

Por tanto, todos los autovalores de la matriz A están en la unión de los discos

$$|z - a_{i,i}| \leq \Lambda_i, \quad 1 \leq i \leq n.$$

A estos discos los llamaremos discos de Gerschgorin.

Basándonos en este resultado, vamos a dar nuevas cotas para el radio espectral de una matriz.

Corolario 1.3. *Sea $A = [a_{i,j}]$ una matriz. Entonces se verifica*

1) $\rho(A) \leq \|A\|_\infty,$

2) $\rho(A) \leq \|A\|_1.$

Demostración.

1) Cada uno de los discos de Gerschgorin $|z - a_{i,i}| \leq \Lambda_i$ está contenido en el disco $|z| \leq |a_{i,i}| + \Lambda_i \leq \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{i,j}| = \|A\|_\infty$. En consecuencia, $\rho(A) \leq \|A\|_\infty$.

2) Puesto que A y A^T tienen los mismos autovalores, aplicando el [Teorema 1.4](#) a la matriz A^T tenemos que los autovalores están en la unión de los discos

$$|z - a_{j,j}| \leq \Lambda_j, \quad 1 \leq j \leq n,$$

siendo

$$\Lambda_j = \sum_{\substack{i=1 \\ i \neq j}}^n |a_{i,j}|.$$

Al igual que antes, estos discos están contenidos en el disco $|z| \leq |a_{j,j}| + \Lambda_j \leq \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{i,j}| = \|A\|_1$ y, por tanto, $\rho(A) \leq \|A\|_1$.

□

Corolario 1.4. Sea $A = [a_{i,j}]$ una matriz y sean x_1, \dots, x_n números reales positivos. Definimos

$$v = \max_{1 \leq i \leq n} \left(\frac{\sum_{j=1}^n |a_{i,j}| x_j}{x_i} \right) \quad y \quad v' = \max_{1 \leq j \leq n} \left(x_j \sum_{i=1}^n \frac{|a_{i,j}|}{x_i} \right).$$

Entonces, $\rho(A) \leq \min(v, v')$.

Demostración. Basta aplicar el [Corolario 1.3](#) a la matriz $D^{-1}AD$ siendo D la matriz diagonal

$$D = \begin{bmatrix} x_1 & & \\ & \ddots & \\ & & x_n \end{bmatrix}.$$

□

Definición 1.5. Sea A una matriz. Se dice que es convergente si la sucesión de matrices $\{A^m\}_{m=1}^{\infty}$ converge a la matriz nula cuando $m \rightarrow \infty$. Diremos que A es divergente en cualquier otro caso.

Teorema 1.5. Sea A una matriz $n \times n$. A es convergente si y solo si $\rho(A) < 1$.

Demostración. Vamos a utilizar la forma canónica de Jordan de la matriz A . Dada la matriz A , sabemos que existe una matriz regular $S \in M_{n \times n}(\mathbb{C})$ tal que

$$\tilde{A} = SAS^{-1} = \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_r \end{bmatrix}$$

donde cada $J_s \in M_{n_s \times n_s}(\mathbb{C})$ es de la forma

$$J_s = \begin{bmatrix} \lambda_s & 1 & & \\ & \ddots & \ddots & \\ & & \lambda_s & 1 \\ & & & \lambda_s \end{bmatrix}$$

y los $\{\lambda_s\}_{s=1}^r$ son los autovalores de A y también los de \tilde{A} puesto que son los mismos. Dado que $\tilde{A}^m = SA^mS^{-1}$, vamos a calcular los elementos de \tilde{A}^m

$$\tilde{A}^m = \begin{bmatrix} J_1^m & & \\ & \ddots & \\ & & J_r^m \end{bmatrix}.$$

Los bloques de Jordan podemos escribirlos como

$$J_s = \lambda_s I + N_s,$$

donde I es la matriz identidad de tamaño $n_s \times n_s$ y N_s una matriz nilpotente de orden n_s . Inductivamente podemos dar una fórmula general para los elementos de J_s^m

$$\begin{aligned} J_s^m &= (\lambda_s I + N_s)^m \\ &= \binom{m}{0} \lambda_s^m + \binom{m}{1} \lambda_s^{m-1} N_s + \cdots + \binom{m}{n_s-1} \lambda_s^{m-(n_s-1)} N_s^{n_s-1}. \end{aligned}$$

Podemos escribir de forma general los elementos de $J_s^m = [d_{i,j}^{(m)}]$ de la siguiente forma

$$d_{i,j}^{(m)} = \begin{cases} 0 & \text{si } i > j \\ \binom{m}{j-i} \lambda_s^{m-j+i} & \text{si } i \leq j \leq \min\{n_s, m+i\} \\ 0 & \text{si } m+i < j < n_s \end{cases}$$

Supongamos que A es convergente, es decir, $A^m \xrightarrow{m \rightarrow \infty} O$. Como $\tilde{A} = SAS^{-1}$ tenemos que $\tilde{A}^m \xrightarrow{m \rightarrow \infty} O$ y por tanto $J_s^m \xrightarrow{m \rightarrow \infty} O$ lo que implica que $|\lambda_s| < 1$ para $1 \leq s \leq r$, es decir, $\rho(\tilde{A}) < 1$ y como A y \tilde{A} tienen los mismos autovalores concluimos que $\rho(A) < 1$.

Recíprocamente si $\rho(A) = \rho(\tilde{A}) < 1$ tenemos que $d_{i,j}^{(m)} \xrightarrow{m \rightarrow \infty} 0$, es decir, $J_s^m \xrightarrow{m \rightarrow \infty} O$, lo que implica que $\tilde{A} \xrightarrow{m \rightarrow \infty} O$ y por tanto $A \xrightarrow{m \rightarrow \infty} O$, es decir, A es convergente. \square

1.2. Matrices irreducibles

Definición 1.6. Para $n \geq 2$, decimos que la matriz A es irreducible si no existe ninguna matriz de permutación P tal que

$$PAP^T = \begin{bmatrix} A_{1,1} & A_{1,2} \\ O & A_{2,2} \end{bmatrix},$$

donde $A_{1,1}$ y $A_{2,2}$ son matrices cuadradas de tamaños $k \times k$ y $(n-k) \times (n-k)$ respectivamente con $k \neq 0, n$. Si existe tal permutación diremos que la matriz es reducible.

Supongamos que queremos resolver $A\mathbf{x} = \mathbf{b}$ donde

$$A = P\tilde{A}P^T = \begin{bmatrix} A_{1,1} & A_{1,2} \\ O & A_{2,2} \end{bmatrix}.$$

Entonces podemos escribirlo de la siguiente manera

$$\begin{aligned} A_{1,1}\mathbf{x}_1 + A_{1,2}\mathbf{x}_2 &= \mathbf{b}_1 \\ A_{2,2}\mathbf{x}_2 &= \mathbf{b}_2, \end{aligned}$$

es decir, resolver el sistema lineal original se reduce a resolver dos sistemas lineales de menor tamaño, de ahí el nombre de matrices reducibles.

Si reducimos la matriz A todo lo que podamos llegaremos a una matriz que tiene la siguiente forma

$$\begin{bmatrix} R_{1,1} & R_{1,2} & \dots & R_{1,m} \\ O & R_{2,2} & \dots & R_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \dots & R_{m,m} \end{bmatrix},$$

siendo $R_{i,i}$, $1 \leq i \leq m$, una matriz irreducible o una matriz nula de tamaño 1×1 . Diremos que esta matriz es la forma normal de una matriz reducible.

Definición 1.7. Un grafo dirigido es un par $(\mathcal{V}, \mathcal{A})$ donde $\mathcal{V} = \{1, 2, \dots, n\}$ es el conjunto de vértices y $\mathcal{A} \subset \mathcal{V} \times \mathcal{V}$ es el conjunto de aristas dirigidas, de modo que el par ordenado (i, j) está en \mathcal{A} si existe una arista que parte del nodo i y llega al nodo j . Dada una matriz $A = [a_{i,j}]$, el grafo $G(A)$ asociado a la matriz A es un grafo dirigido de tal forma que si $a_{i,j} \neq 0$, existe una arista que parte del nodo i y llega al nodo j , es decir, $\mathcal{A} = \{(i, j), 1 \leq i, j \leq n : a_{i,j} \neq 0\}$.

Definición 1.8. Un grafo dirigido de n nodos es fuertemente conexo si para cualquier par de nodos i y j con $1 \leq i, j \leq n$, existe un camino dirigido que conecta i con j , es decir, existe un número finito de pares $(i_1, i_2), (i_2, i_3), \dots, (i_{k-1}, i_k)$ en \mathcal{A} con $i_1 = i$ e $i_k = j$.

Teorema 1.6. Una matriz A es irreducible si y solo si su grafo dirigido $G(A)$ es fuertemente conexo.

Demostración. Para la implicación directa veamos que si $G(A)$ no es fuertemente conexo entonces A es reducible. Si $G(A)$ no es fuertemente conexo, entonces existen dos índices $i \neq j$ tales que no es posible encontrar un camino dirigido que parta del nodo i y llegue al nodo j . Si $G(A) = (\mathcal{V}, \mathcal{A})$, consideramos los siguientes conjuntos disjuntos

$$I = \{r \in \mathcal{V} : \text{existe un camino dirigido que va de } i \text{ a } r\},$$

$$J = \mathcal{V} \setminus I.$$

Tomando la permutación P , que ordena los nodos de forma que aparecen en primer lugar los que están en I y después los que están en J ,

$$P : \mathcal{V} \mapsto \{I, J\},$$

entonces

$$PAP^T = \begin{bmatrix} A_{1,1} & A_{1,2} \\ O & A_{2,2} \end{bmatrix},$$

lo que prueba que A es reducible como queríamos ver.

Para el recíproco veamos que si A es reducible entonces el grafo asociado $G(A)$ no es fuertemente conexo. Si A es reducible existe una matriz de permutación P tal que

$$PAP^T = \begin{bmatrix} A_{1,1} & A_{1,2} \\ O & A_{2,2} \end{bmatrix}$$

Considerando esta matriz por bloques, no es posible encontrar un camino dirigido que parta de un nodo i del bloque 2 y llegue a un nodo j del bloque 1, luego $G(A)$ no es fuertemente conexo. \square

Veamos algunos ejemplos de aplicación de este teorema. Consideramos las matrices

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad \text{y} \quad B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

En cuanto a la matriz A podemos observar que es reducible ya que el grafo asociado $G(A)$ no es fuertemente conexo porque no podemos ir del nodo 2 al 1. La matriz B es irreducible porque como podemos ver el grafo $G(B)$ es fuertemente conexo.

Ahora vamos a extender el teorema de Gerschgorin a matrices irreducibles.

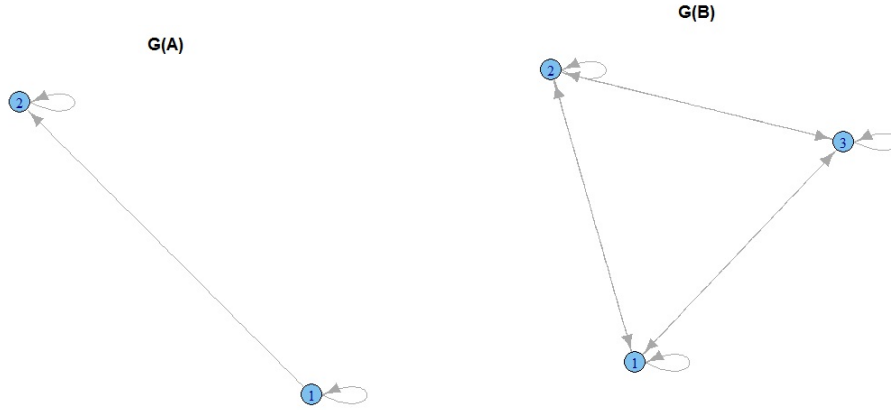


Figura 1: Grafos dirigidos asociados a las matrices A y B

Teorema 1.7. *Sea A una matriz irreducible y supongamos que λ es un autovalor de A que está en la frontera de la unión de los discos de Gerschgorin $|z - a_{i,i}| \leq \Lambda_i$. Entonces, λ está en las n circunferencias $|z - a_{i,i}| = \Lambda_i$, es decir,*

$$|\lambda - a_{i,i}| = \Lambda_i, \quad 1 \leq i \leq n.$$

Demostración. Sea λ un autovalor de la matriz A y $\mathbf{x} \neq 0$ el autovector asociado, normalizado para que la componente de mayor módulo sea la unidad. Supongamos que dicha componente es la r -ésima, es decir, $|x_r| = 1$ y $|x_i| \leq 1$ si $i \neq r$. Dado que $A\mathbf{x} = \lambda\mathbf{x}$, se tiene que

$$\sum_{j=1}^n a_{i,j}x_j = \lambda x_i, \quad \text{para cada } 1 \leq i \leq n,$$

o equivalentemente, podemos escribir

$$\sum_{\substack{j=1 \\ j \neq i}}^n a_{i,j}x_j = (\lambda - a_{i,i})x_i, \quad \text{para cada } 1 \leq i \leq n.$$

En particular, para $i = r$ tomamos módulos, aplicamos la desigualdad triangular, y tenemos que

$$|\lambda - a_{r,r}| \leq \sum_{\substack{j=1 \\ j \neq r}}^n |a_{r,j}||x_j| \leq \sum_{\substack{j=1 \\ j \neq r}}^n |a_{r,j}| = \Lambda_r. \quad (1)$$

Como λ tiene que estar en la frontera de los discos a los que pertenece, debe verificarse la igualdad en (1), es decir,

$$|\lambda - a_{r,r}| = \sum_{\substack{j=1 \\ j \neq r}}^n |a_{r,j}| |x_j| = \Lambda_r.$$

En consecuencia, para todos $a_{r,p} \neq 0$ con $p \neq r$, tiene que cumplirse que $|x_p| = 1$. Dado que A es irreducible, existe al menos un p cumpliendo $a_{r,p} \neq 0$, luego $|x_p| = 1$ y repetimos anterior cambiando r por p . Finalmente, por el Teorema 1.6 como A es irreducible su grafo asociado, $G(A)$, es fuertemente conexo. Por lo tanto podemos encontrar un camino $a_{rp}, a_{pp_1}, \dots, a_{p_m j}$ que va de r a j para cualquier $1 \leq j \leq n$. Repitiendo el proceso anterior con los índices p, p_1, \dots, p_m, j queda demostrado que todos los nodos tienen la propiedad de que

$$|\lambda - a_{j,j}| = \Lambda_j \text{ para } 1 \leq j \leq n,$$

como queríamos ver. □

1.3. Matrices diagonalmente dominantes

Definición 1.9. Sea A una matriz $n \times n$. Decimos que es diagonalmente dominante si

$$\sum_{\substack{i=1 \\ i \neq j}}^n |a_{i,j}| \leq |a_{i,i}|, \quad 1 \leq i \leq n.$$

Si la desigualdad es estricta diremos que A es estrictamente diagonalmente dominante. Si es diagonalmente dominante, se da la desigualdad estricta para al menos un índice y es irreducible, diremos que A es irreduciblemente diagonalmente dominante.

Teorema 1.8. Sea $A = [a_{i,j}]$ una matriz estrictamente o irreduciblemente diagonalmente dominante, entonces la matriz A es regular. Además, si todos los elementos diagonales de A son números reales positivos entonces los autovalores λ_i cumplen que

$$\operatorname{Re}(\lambda_i) > 0, \quad 1 \leq i \leq n.$$

Demostración. Para ver que A es regular basta con ver que $\lambda = 0$ no es autovalor. El caso más sencillo es que A sea estrictamente diagonalmente

dominante, es decir,

$$\sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}| < |a_{i,i}|, \quad 1 \leq i \leq n.$$

Por el [Teorema 1.4](#) sabemos que si λ es autovalor tiene que estar en la unión de los discos de Gerschgorin, es decir, debe cumplirse que

$$|\lambda - a_{i,i}| \leq \Lambda_i, \quad \text{para algún } i.$$

Por reducción al absurdo, supongamos que $\lambda = 0$ es autovalor. Entonces tendríamos que

$$|-a_{i,i}| = |a_{i,i}| \leq \Lambda_i$$

en contra de que A es estrictamente diagonalmente dominante. Por tanto $\lambda = 0$ no es autovalor y en consecuencia A es regular. El hecho de que $\operatorname{Re}(\lambda_i) > 0$ es bastante intuitivo: puesto que A es estrictamente diagonalmente dominante es claro que si consideramos los discos de Gerschgorin, los radios Λ_i van a ser menores estrictamente que $|a_{i,i}|$ y como los $a_{i,i}$ son números reales positivos claramente los discos estarán contenidos en la parte del plano complejo con parte real estrictamente positiva y por tanto los autovalores λ_i también.

En el caso de que A sea irreduciblemente diagonalmente dominante, por el [Teorema 1.7](#) si λ es autovalor y no está en el interior de los discos debe ocurrir que

$$|\lambda - a_{i,i}| = \Lambda_i, \quad 1 \leq i \leq n.$$

Si $\lambda = 0$ fuera autovalor tendríamos que

$$|-a_{i,i}| = |a_{i,i}| = \Lambda_i, \quad 1 \leq i \leq n,$$

lo cual es imposible porque según la [Definición 1.9](#) para algún índice i se tiene

$$\Lambda_i = \sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}| < |a_{i,i}|.$$

Al igual que antes y una vez descartado $\lambda = 0$ como autovalor, es claro que $\operatorname{Re}(\lambda_i) > 0$. \square

Corolario 1.5. *Sea $A = [a_{i,j}]$ una matriz hermítica, estrictamente diagonalmente dominante o irreduciblemente diagonalmente dominante cuyos elementos diagonales son números reales positivos, entonces la matriz A es definida positiva.*

Demostración. Para ver que A es definida positiva basta comprobar que tiene todos sus autovalores positivos. Podemos aplicar el [Teorema 1.8](#) pues A es estrictamente o irreduciblemente diagonalmente dominante y los elementos diagonales de A son reales, entonces $Re(\lambda_i) > 0$ para $1 \leq i \leq n$. Como A es hermítica, sus autovalores λ_i son reales y en conclusión, como queríamos ver, $\lambda_i > 0$ para todo $1 \leq i \leq n$. \square

1.4. Matrices no negativas. El teorema de Perron-Frobenius

Definición 1.10. Sean $A = [a_{i,j}], B = [b_{i,j}] \in M_{n \times m}(\mathbb{R})$. Diremos que $A \geq B$ si $a_{i,j} \geq b_{i,j}$ para todo $1 \leq i \leq n$ y $1 \leq j \leq m$. Si O es la matriz nula, diremos que la matriz A es no negativa (resp. positiva) si $A \geq O$ (resp. $A > O$). Si $A = [a_{i,j}] \in M_{n \times m}(\mathbb{C})$ denotaremos por $|A|$ a la matriz cuyos elementos son $|a_{i,j}|$.

También utilizaremos esta notación con vectores.

Lema 1.1. Si $A \geq O$ es irreducible, entonces

$$(I + A)^{n-1} > O$$

Demostración. Basta ver que $(I + A)^{n-1} \mathbf{x} > \mathbf{0}$ para cualquier $\mathbf{x} \geq \mathbf{0}, \mathbf{x} \neq \mathbf{0}$. Recurrentemente, definimos la siguiente sucesión de vectores no negativos

$$\mathbf{x}_{k+1} = (I + A)\mathbf{x}_k, \quad k = 0, 1, \dots, n-2,$$

donde $\mathbf{x}_0 = \mathbf{x}$. Puesto que $I + A$ y \mathbf{x} son no negativos, todos los \mathbf{x}_k serán no negativos. De hecho vamos a ver que el vector \mathbf{x}_{k+1} tiene menos componentes nulas que \mathbf{x}_k . Por reducción al absurdo, supongamos que \mathbf{x}_{k+1} y \mathbf{x}_k tienen las últimas r componentes nulas, (si no fuera así existiría un permutación P tal que

$$P\mathbf{x}_{k+1} = \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{0} \end{bmatrix} \text{ y } P\mathbf{x}_k = \begin{bmatrix} \boldsymbol{\beta} \\ \mathbf{0} \end{bmatrix}$$

donde $\boldsymbol{\alpha}, \boldsymbol{\beta} > \mathbf{0}$ son vectores de tamaño $n - r$).

Según la ecuación de recurrencia tenemos que

$$\mathbf{x}_{k+1} = (I + A)\mathbf{x}_k = \mathbf{x}_k + A\mathbf{x}_k = \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\beta} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta} \\ \mathbf{0} \end{bmatrix}$$

siendo $A_{1,1}$ y $A_{2,2}$ matrices cuadradas de tamaño $n - r$ y r respectivamente. Realizando las operaciones matriciales por bloques debe cumplirse

que $A_{2,1}\boldsymbol{\beta} = \mathbf{0}$, lo cual solo es posible si $A_{2,1} = O$ porque $\boldsymbol{\beta} > \mathbf{0}$, pero esto entra en contradicción con el supuesto de que A sea irreducible. Puesto que $\mathbf{x} \neq \mathbf{0}$, a lo sumo puede tener $n - 1$ componentes nulas, luego el vector \mathbf{x}_k tendrá a lo sumo $n - k - 1$ componentes nulas y por tanto

$$\mathbf{x}_{n-1} = (I + A)\mathbf{x}_{n-2} = (I + A)^2\mathbf{x}_{n-3} = \cdots = (I + A)^{n-1}\mathbf{x}_0 > \mathbf{0}.$$

□

Definición 1.11. Sean $A = [a_{i,j}] \geq O$ irreducible y $\mathbf{x} \geq \mathbf{0}$. Definimos

$$r_{\mathbf{x}} = \min_{x_i > 0} \left[\frac{\sum_{j=1}^n a_{i,j}x_j}{x_i} \right] \geq 0, \quad r = \sup_{\substack{\mathbf{x} \geq \mathbf{0} \\ \mathbf{x} \neq \mathbf{0}}} r_{\mathbf{x}} \geq 0.$$

Puesto que $r_{\mathbf{x}}$ y $r_{\alpha\mathbf{x}}$ coinciden para cualquier $\alpha > 0$ nos basta considerar el siguiente conjunto compacto

$$P = \{\mathbf{x} \geq \mathbf{0} : \|\mathbf{x}\| = 1\}$$

al cual vamos a asociar otro conjunto de vectores positivos

$$Q = \{\mathbf{y} = (I + A)^{n-1}\mathbf{x} : \mathbf{x} \in P\}$$

que también es compacto. Además $r_{\mathbf{y}}$ es una función continua en Q por lo que existe $\mathbf{z} > \mathbf{0}$ tal que

$$A\mathbf{z} \geq r\mathbf{z}.$$

A este vector \mathbf{z} lo llamaremos vector extremal.

Además, no existe ningún vector $\mathbf{w} \geq \mathbf{0}$ tal que $A\mathbf{w} > r\mathbf{w}$.

Lema 1.2. Si $A \geq O$ es irreducible, la cantidad r de la [Definición 1.11](#) es positiva. Además, cada vector extremal \mathbf{z} es un autovector positivo de A asociado al autovalor r , es decir, $A\mathbf{z} = r\mathbf{z}$ y \mathbf{z} es positivo.

Demostración. Puesto que A es irreducible no puede tener toda una fila nula. Entonces si $\boldsymbol{\epsilon} = (1, \dots, 1)^T$, el producto $A\boldsymbol{\epsilon}$ tampoco puede tener ninguna componente nula. En este caso y puesto que $A \geq O$ tenemos que

$$r_{\boldsymbol{\epsilon}} = \sum_{j=1}^n a_{i,j} > 0$$

y por tanto $r \geq r_{\boldsymbol{\epsilon}} > 0$.

Sea ahora \mathbf{z} un vector extremal de A , es decir, tal que

$$A\mathbf{z} \geq r\mathbf{z}.$$

Para que r sea autovalor asociado al autovector \mathbf{z} tendría que darse la igualdad, es decir, si suponemos que

$$A\mathbf{z} - r\mathbf{z} = \boldsymbol{\eta} \geq \mathbf{0}, \quad (2)$$

hay que ver que $\boldsymbol{\eta} = \mathbf{0}$. Por reducción al absurdo, supongamos que $\boldsymbol{\eta} \neq \mathbf{0}$. Entonces alguna de sus componentes es positiva. Multiplicando en (2) por $(I + A)^{n-1}$ tenemos que

$$A(I + A)^{n-1}\mathbf{z} - r(I + A)^{n-1}\mathbf{z} = (I + A)^{n-1}\boldsymbol{\eta} > \mathbf{0}.$$

Para simplificar, llamamos $\mathbf{w} = (I + A)^{n-1}\mathbf{z} > \mathbf{0}$ y la desigualdad anterior puede expresarse como $A\mathbf{w} > r\mathbf{w}$. Si analizamos dicha desigualdad componente a componente tenemos que

$$\sum_{j=1}^n a_{i,j}w_j > rw_i,$$

lo cual implica que

$$\frac{\sum_{j=1}^n a_{i,j}w_j}{w_i} > r$$

y, por tanto,

$$r_{\mathbf{w}} = \min_{1 \leq i \leq n} \left(\frac{\sum_{j=1}^n a_{i,j}w_j}{w_i} \right) > r,$$

llegando a contradicción por ser, por definición, $r \geq r_{\mathbf{w}}$. En conclusión, tenemos que $\boldsymbol{\eta} = \mathbf{0}$ y, por tanto, \mathbf{z} es autovector de A .

Finalmente, por ser r autovalor de A tenemos que

$$\mathbf{w} = (I + A)^{n-1}\mathbf{z} = (1 + r)^{n-1}\mathbf{z},$$

y como \mathbf{w} y r son positivos, debe ser $\mathbf{z} > \mathbf{0}$.

□

Lema 1.3. Sean $A = [a_{i,j}] \geq O$ irreducible y $B = [b_{i,j}] \in M_{n \times n}(\mathbb{C})$ con $|B| \leq A$. Si β es un autovalor cualquiera de B entonces,

$$|\beta| \leq r$$

siendo r el valor de la [Definición 1.11](#). Además, se puede dar la igualdad, es decir, $\beta = re^{i\phi}$ para algún número real ϕ si y solo si $|B| = A$ y $B = e^{i\phi}DAD^{-1}$ siendo D una matriz diagonal con elementos unitarios.

Demostración. Dado que β es autovalor de B , sabemos que $B\mathbf{x} = \beta\mathbf{x}$ con $\mathbf{x} \neq \mathbf{0}$. Escrito componente a componente

$$\sum_{j=1}^n b_{i,j}x_j = \beta x_i \text{ para } 1 \leq i \leq n.$$

Si tomamos módulos

$$|\beta||x_i| = |\beta x_i| = \left| \sum_{j=1}^n b_{i,j}x_j \right| \leq \sum_{j=1}^n |b_{i,j}||x_j|,$$

luego

$$|\beta||\mathbf{x}| \leq |B||\mathbf{x}| \leq A|\mathbf{x}|,$$

donde la segunda desigualdad se da por hipótesis y de donde deducimos que

$$|\beta| \leq r_{|\mathbf{x}|} \leq r.$$

Ahora si $\beta = r$ se tiene que $|\mathbf{x}|$ es un vector extremal de A y por el [Lema 1.2](#) sabemos que $|\mathbf{x}| > \mathbf{0}$ es autovector asociado al autovalor positivo r . Además, debe ocurrir que

$$r|\mathbf{x}| = |B||\mathbf{x}| = A|\mathbf{x}|, \tag{3}$$

puesto que $|\mathbf{x}| > \mathbf{0}$, y junto con la hipótesis de que $|B| \leq A$ debe ocurrir que $|B| = A$.

Consideramos la matriz

$$D = \begin{bmatrix} \frac{x_1}{|x_1|} & & \\ & \ddots & \\ & & \frac{x_n}{|x_n|} \end{bmatrix}$$

cuyos elementos no nulos tienen módulo 1 y es claro que

$$\mathbf{x} = D|\mathbf{x}|. \tag{4}$$

Escribiendo $\beta = re^{i\phi}$ para algún $\phi \in \mathbb{R}$, entonces podemos reescribir $B\mathbf{x} = \beta\mathbf{x}$ como

$$B\mathbf{x} = re^{i\phi}\mathbf{x}$$

y utilizando (4)

$$BD|\mathbf{x}| = re^{i\phi}D|\mathbf{x}|.$$

Denotando $C = e^{-i\phi}D^{-1}BD$, entonces $B\mathbf{x} = \beta\mathbf{x}$ se reescribe como

$$C|\mathbf{x}| = r|\mathbf{x}|.$$

Llevando esto a (3) se tiene que

$$r|\mathbf{x}| = C|\mathbf{x}| = |B||\mathbf{x}| = A|\mathbf{x}|.$$

Por otro lado, de la definición de C y de la igualdad $|B| = A$ se deduce

$$|C| = |B| = A.$$

En consecuencia,

$$C|\mathbf{x}| = |C||\mathbf{x}| = A|\mathbf{x}|,$$

y como $|\mathbf{x}| > 0$ deducimos que

$$C = |C| = A.$$

Finalmente, como $C = e^{-i\phi}D^{-1}BD = A$ tenemos

$$B = e^{i\phi}DAD^{-1}.$$

Para el recíproco, es claro que si $B = e^{i\phi}DAD^{-1}$ entonces $|B| = A$ por ser $A \geq O$ y B tiene un autovalor β tal que $|\beta| = r$. \square

Corolario 1.6. *Sea $A \geq O$ irreducible. Entonces el autovalor r del [Lema 1.2](#) coincide con $\rho(A)$.*

Demostración. Según el [Lema 1.2](#), el valor r es autovalor de A y por tanto $r \leq \rho(A)$. Además, aplicando el [Lema 1.3](#) tomando $B = A$ tenemos que si β es autovalor de A ,

$$|\beta| \leq r$$

siendo β el autovalor de A . Entonces,

$$\rho(A) = \max_{\beta \in \sigma(A)} |\beta| \leq r \leq \rho(A)$$

y por tanto debe cumplirse que $\rho(A) = r$. \square

Lema 1.4. Sea $A \geq O$ irreducible. Si B es una submatriz cuadrada principal de A entonces $\rho(B) < \rho(A)$

Demostración. Como B es una submatriz principal de A , existe una matriz de permutación P tal que

$$PAP^T = \begin{bmatrix} B & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix},$$

donde B y $A_{2,2}$ son matrices cuadradas de tamaño $m \times m$ y $(n-m) \times (n-m)$, respectivamente.

Definimos

$$C = \begin{bmatrix} B & O \\ O & O \end{bmatrix}.$$

Es claro que $\rho(B) = \rho(C)$ y $O \leq C \leq PAP^T$ con $C \neq PAP^T$. Por el [Lema 1.3](#) y el [Corolario 1.6](#) sabemos que si β es autovalor de B ,

$$|\beta| < r = \rho(A),$$

luego

$$\rho(B) = \max_{\beta \in \sigma(B)} |\beta| < \rho(A).$$

□

Lema 1.5. Sea A una matriz y $\Phi(\lambda)$ su polinomio característico. Entonces $\Phi'(\lambda)$ coincide con la suma de todos los determinantes de las submatrices principales de tamaño $(n-1) \times (n-1)$ de la matriz $\lambda I - A$.

Demostración. Para simplificar, llamamos $B = \lambda I - A$. Por definición de determinante tenemos que

$$\Phi(\lambda) = \det(B) = \sum_{\sigma \in S_n} (-1)^{\pi(\sigma)} \prod_{i=1}^n b_{i,\sigma(i)},$$

siendo $\pi(\sigma)$ la signatura de la permutación y

$$b_{i,j} = \begin{cases} -a_{i,j} & \text{si } i \neq j, \\ \lambda - a_{i,j} & \text{si } i = j. \end{cases}$$

Ahora, derivamos respecto de λ

$$\Phi'(\lambda) = \sum_{\sigma \in S_n} (-1)^{\pi(\sigma)} \left[\sum_{k=1}^n b'_{k,\sigma(k)} \prod_{j \neq k} b_{j,\sigma(j)} \right]. \quad (5)$$

Por como están definidos los $b_{i,j}$, sabemos que

$$b'_{k,\sigma(k)} = \begin{cases} 1 & \text{si } k = \sigma(k), \\ 0 & \text{en otro caso.} \end{cases}$$

Entonces, podemos reescribir (5) como

$$\Phi'(\lambda) = \sum_{k=1}^n \left[\sum_{\substack{\sigma \in S_n \\ \sigma(k)=k}} (-1)^{\pi(\sigma)} \prod_{j \neq k} b_{j,\sigma(j)} \right].$$

Consideramos la aplicación

$$R: \begin{array}{ccc} \{\sigma \in S_n : \sigma(k) = k\} & \longrightarrow & S_{n-1} \\ \sigma & \longmapsto & \tilde{\sigma} \end{array}$$

dada por $\tilde{\sigma}(i) = \sigma(i)$ para $i \neq k$. Puesto que σ deja fijo el índice k , ambas permutaciones tienen la misma signatura y, por tanto,

$$\Phi'(\lambda) = \sum_{k=1}^n \left[\sum_{\tilde{\sigma} \in S_{n-1}} (-1)^{\pi(\tilde{\sigma})} \prod_{\substack{i=1 \\ i \neq k}}^n b_{i,\tilde{\sigma}(i)} \right] = \sum_{k=1}^n \det(\lambda I - A_{k,k}),$$

siendo $A_{k,k}$ el menor de tamaño $(n-1) \times (n-1)$ que resulta de eliminar de A la fila y la columna k . \square

Teorema 1.9 (Perron-Frobenius). *Sea $A \geq O$ irreducible. Se verifica que*

- 1) *La matriz A tiene un autovalor positivo que coincide con su radio espectral.*
- 2) *El autovector asociado al autovalor $\rho(A)$ es positivo.*
- 3) *El radio espectral de A aumenta al aumentar cualquier elemento de A .*
- 4) *$\rho(A)$ es un autovalor simple de A .*

Demostración.

- 1) Según el [Lema 1.2](#) el autovalor r es positivo y por el [Corolario 1.6](#) sabemos que coincide con $\rho(A)$.

- 2) Se sigue del [Lema 1.2](#).
- 3) Si aumentamos algunos elementos de A , conseguiremos otra matriz B que también es irreducible. Según el [Lema 1.3](#) y el [Corolario 1.6](#), si α es un autovalor cualquiera de A se tiene que

$$|\alpha| \leq r = \rho(B).$$

Como esto ocurre para cualquier autovalor de A , se tiene que $\rho(A) \leq \rho(B)$. Además, la igualdad no puede darse, por la segunda parte del [Lema 1.3](#) y puesto que $A \neq B$.

- 4) Sea $\Phi(\lambda)$ el polinomio característico de A . Como $\rho(A)$ es raíz de $\Phi(\lambda)$, basta comprobar que es una raíz simple, o equivalentemente, tenemos que ver que $\Phi'(\rho(A)) > 0$. Por el [Lema 1.5](#) sabemos que

$$\Phi'(\rho(A)) = \sum_{i=1}^n \det(\rho(A)I - A_{i,i}) \text{ para } 1 \leq i \leq n,$$

siendo $A_{i,i}$ el menor de tamaño $(n-1) \times (n-1)$ que resulta de eliminar de A la fila y la columna i . Para cada i la función $\det(\lambda I - A_{i,i})$ tiende hacia infinito cuando λ tiende a infinito y no puede anularse para ningún $\lambda > \rho(A_{i,i})$. Por el [Lema 1.4](#) sabemos que $\rho(A) > \rho(A_{i,i})$, entonces $\det(\rho(A)I - A_{i,i}) > 0$ para todo i y, por tanto, $\Phi'(\rho(A)) > 0$.

□

También podemos enunciar una generalización del Teorema de Perron-Frobenius para el caso de matrices reducibles.

Teorema 1.10. *Sea $A \geq O$. Se verifica que*

- 1) *La matriz A tiene un autovalor real no negativo que coincide con su radio espectral. De hecho es positivo, salvo que A sea reducible y su forma normal sea estrictamente triangular superior.*
- 2) *El autovector asociado al autovalor $\rho(A)$ es no negativo.*
- 3) *El radio espectral de A no decrece al aumentar cualquier elemento de A .*

Demostración. Supongamos que A está en forma normal. Basta analizar lo que ocurre con cada bloque diagonal.

1) Si $R_{i,i}$ es una matriz nula de tamaño 1×1 , tiene un autovalor simple nulo. Si $R_{i,i}$ es irreducible, según el [Teorema 1.9](#) tiene un autovalor positivo que coincide con su radio espectral. Como el radio espectral de A será el mayor de los radios espectrales de los bloques $R_{i,i}$ diagonales irreducibles, coincidirá con el correspondiente autovalor positivo del bloque de mayor radio espectral. Si la forma normal de A es estrictamente triangular superior es porque todos los $R_{i,i}$ son matrices nulas, y en consecuencia $\rho(A) = 0$.

2) Si $\rho(A) = 0$, como la forma normal de la matriz será estrictamente triangular superior, basta tomar el vector $\mathbf{e}_1 = (1, 0, \dots, 0)^T$, que es no negativo, como autovector asociado al 0.

Si $\rho(A) > 0$, existe un i tal que $\rho(A)$ es el autovalor de mayor módulo de la matriz irreducible $R_{i,i}$ y, por el [Teorema 1.9](#) tiene un autovector asociado $\mathbf{w}^{(i)} > \mathbf{0}$. El vector \mathbf{w} con todas las componentes nulas excepto las correspondientes al bloque i que se toman iguales al vector $\mathbf{w}^{(i)}$, es no negativo y es autovector de la matriz A asociado al autovalor positivo $\rho(A)$.

3) Si el elemento que se aumenta corresponde a un bloque irreducible $R_{i,i}$ para el cual $\rho(A) = \rho(R_{i,i})$ entonces, por el [Teorema 1.9](#) aumentará $\rho(R_{i,i})$ y, en consecuencia, aumentará $\rho(A)$.

Si el elemento que se aumenta corresponde a un bloque irreducible $R_{i,i}$ para el cual $\rho(A) > \rho(R_{i,i})$ aunque, por el [Teorema 1.9](#) aumentará $\rho(R_{i,i})$ es posible que $\rho(A)$ se mantenga igual.

Por último, si se aumenta un elemento diagonal correspondiente a un bloque 1×1 nulo, entonces claramente el radio espectral de dicho bloque aumentará, pero es posible que no supere el valor de $\rho(A)$.

También podría suceder que al aumentar algún elemento de A , los bloques diagonales de su forma normal no cambiasen, en cuyo caso $\rho(A)$ tampoco cambia.

Lo que se puede afirmar en todos los casos es que $\rho(A)$ no va a disminuir al aumentar algún elemento de A .

□

2. Análisis de convergencia de los métodos iterativos clásicos

El objetivo en esta sección es analizar la convergencia de los métodos iterativos clásicos: Jacobi, Gauss-Seidel y SOR. Hablaremos de que los métodos iterativos son convergentes cuando sean convergentes las correspondientes matrices de iteración.

2.1. Descripción de los métodos

Nuestro problema consiste en resolver un sistema lineal de n ecuaciones con n incógnitas

$$A\mathbf{x} = \mathbf{b}$$

donde $A \in M_{n \times n}(\mathbb{C})$ es una matriz regular con elementos diagonales no nulos y $\mathbf{b} \in \mathbb{C}^n$ es un vector columna. Partiendo de una aproximación inicial a la solución, $\mathbf{x}^{(0)}$, los métodos iterativos clásicos generan una sucesión de aproximaciones, $\mathbf{x}^{(m)}$, con la esperanza de que converjan a la solución exacta del sistema, \mathbf{x} , que será única por la regularidad de A . Además, estos métodos tratan de aprovechar la estructura y la dispersión de las matrices.

Los tres métodos que vamos a considerar parten de escribir A como $A = D - E - F$ donde D es una matriz diagonal formada por los elementos diagonales de A , E es una matriz formada por la parte estrictamente triangular inferior de A con signo opuesto y F es una matriz formada por la parte estrictamente triangular superior de A también con signo opuesto. Entonces podemos escribir el sistema lineal original como

$$(D - E - F)\mathbf{x} = \mathbf{b}. \quad (6)$$

2.1.1. Método de Jacobi

Reescribiendo (6) de forma equivalente

$$D\mathbf{x} = (E + F)\mathbf{x} + \mathbf{b},$$

podemos llevar a cabo el siguiente método iterativo

$$D\mathbf{x}^{(m+1)} = (E + F)\mathbf{x}^{(m)} + \mathbf{b}, \quad m \geq 0.$$

Como los elementos diagonales de A son no nulos, D es invertible y tenemos que

$$\mathbf{x}^{(m+1)} = D^{-1}(E + F)\mathbf{x}^{(m)} + D^{-1}\mathbf{b}, \quad m \geq 0,$$

a lo que llamaremos método de Jacobi. A la matriz

$$J = D^{-1}(E + F)$$

la llamaremos matriz de iteración del método de Jacobi asociada a la matriz A . Si definimos las matrices $L = D^{-1}E$ y $U = D^{-1}F$, se tiene que

$$J = L + U$$

y, teniendo en cuenta que $E + F = D - A$, se puede escribir también

$$J = I - D^{-1}A.$$

De estas últimas desigualdades, se deduce fácilmente que J tiene los elementos diagonales nulos.

2.1.2. Método de Gauss-Seidel

Analizando el método de Jacobi podemos observar que al hacer el cálculo de la componente i -ésima del iterante $\mathbf{x}^{(m+1)}$ podríamos utilizar las componentes $x_1^{(m+1)}, \dots, x_{i-1}^{(m+1)}$ que ya tenemos calculadas de este y, si el método está convergiendo, cabe esperar que se cometa menor error. Esta es la idea básica del método que vamos a describir a continuación en forma matricial

$$(D - E)\mathbf{x}^{(m+1)} = F\mathbf{x}^{(m)} + \mathbf{b}, \quad m \geq 0.$$

Como la matriz $D - E$ es regular, por ser triangular con elementos diagonales no nulos, tenemos que

$$\mathbf{x}^{(m+1)} = (D - E)^{-1}F\mathbf{x}^{(m)} + (D - E)^{-1}\mathbf{b}, \quad m \geq 0.$$

A la matriz

$$\mathcal{L}_1 = (D - E)^{-1}F$$

la llamamos matriz de iteración del método de Gauss-Seidel asociada a la matriz A .

2.1.3. Método SOR (*Método de sobrerrelajación sucesiva*)

El tercer método está relacionado con el de Gauss-Seidel en el sentido de que para calcular la componente i -ésima de un iterante utiliza una combinación lineal de la componente i -ésima del iterante anterior y de la componente i -ésima que se obtendría con el método de Gauss-Seidel. En notación matricial el método se escribe como

$$(D - \omega E)\mathbf{x}^{(m+1)} = [(1 - \omega)D - \omega F]\mathbf{x}^{(m)} + \omega\mathbf{b}, \quad m \geq 0,$$

donde ω es un parámetro real no nulo.

Como los elementos diagonales de A son no nulos, la matriz $D - \omega E$ es regular para cualquier valor de ω y, por tanto, se tiene que

$$\mathbf{x}^{(m+1)} = (D - \omega E)^{-1}[(1 - \omega)D - \omega F]\mathbf{x}^{(m)} + \omega(D - \omega E)^{-1}\mathbf{b}, \quad m \geq 0. \quad (7)$$

Considerando de nuevo las matrices $L = D^{-1}E$ y $U = D^{-1}F$, reescribimos (7) como

$$\mathbf{x}^{(m+1)} = (I - \omega L)^{-1}[(1 - \omega)I - \omega U]\mathbf{x}^{(m)} + \omega(I - \omega L)^{-1}D^{-1}\mathbf{b}, \quad m \geq 0.$$

A la matriz

$$\mathcal{L}_\omega = (I - \omega L)^{-1}[(1 - \omega)I + \omega U]$$

la llamamos matriz de iteración del método SOR asociada a la matriz A y el parámetro ω se llama factor de relajación.

Los tres métodos anteriormente descritos se pueden escribir de una forma más general como

$$M\mathbf{x}^{(m+1)} = N\mathbf{x}^{(m)} + \mathbf{b} \quad (8)$$

o, equivalentemente,

$$\mathbf{x}^{(m+1)} = M^{-1}N\mathbf{x}^{(m)} + M^{-1}\mathbf{b}, \quad (9)$$

donde $A = M - N$ es una escisión de la matriz A . La matriz de iteración del método será $M^{-1}N$. La siguiente tabla recoge cómo están definidas las matrices M y N para cada uno de los tres métodos considerados.

Método	M	N
Jacobi	D	$E + F$
Gauss-Seidel	$D - E$	F
SOR	$\frac{1}{\omega}(D - \omega E)$	$\frac{1}{\omega}(\omega F + (1 - \omega)D)$

Si sustituimos estos valores de M y N en (9) obtendremos el método correspondiente.

Aunque el análisis de convergencia de los métodos se realizará utilizando (9), la implementación práctica de los mismos se basa en (8), que requiere resolver en cada iteración un sistema lineal con matriz M . El hecho de que en los métodos considerados la matriz M sea diagonal o triangular se traduce en que el costo operativo de resolver un sistema lineal con matriz M es mucho menor que el de resolver el sistema original con matriz A .

A cada uno de estos métodos iterativos podemos asociarle el vector de error que definimos para cada iterante como

$$\mathbf{e}^{(m)} = \mathbf{x}^{(m)} - \mathbf{x}, \text{ para } m \geq 0,$$

donde \mathbf{x} es la solución exacta del sistema $A\mathbf{x} = \mathbf{b}$ y $\mathbf{x}^{(m)}$ es la aproximación a la solución tras m iteraciones. Se satisface la recurrencia

$$\mathbf{e}^{(m)} = B\mathbf{e}^{(m-1)},$$

donde $B = M^{-1}N$ es la matriz de iteración del método. Podemos escribir el error tras m iteraciones como

$$\mathbf{e}^{(m)} = B\mathbf{e}^{(m-1)} = \dots = B^m\mathbf{e}^{(0)}, \text{ para } m \geq 0.$$

Además, por el [Teorema 1.5](#) sabemos que $\mathbf{e}^{(m)} \rightarrow \mathbf{0}$ para cualquier elección de $\mathbf{e}^{(0)}$ si y solo si $\rho(B) < 1$.

Definición 2.1. *Sea A una matriz convergente. Definimos la razón asintótica de convergencia como*

$$R_\infty(A) = -\ln(\rho(A)).$$

Si B es la matriz de iteración del método, asintóticamente, por cada iteración del método la norma del error se divide por $e^{R_\infty(B)}$. Entonces para dividir el error por e^n , asintóticamente, necesitaremos $\frac{n}{R_\infty(B)}$ iteraciones.

2.2. El teorema de Stein-Rosenberg

En esta sección presentaremos el teorema de Stein-Rosenberg, que va a comparar la velocidad de convergencia de los métodos de Jacobi y Gauss-Seidel, basándose en la comparación de los radios espectrales de las matrices de iteración de dichos métodos.

En general, no existe ningún resultado que asegure que un método sea mejor que el otro. Sin embargo, añadiendo la condición de no negatividad a la matriz de Jacobi, el teorema de Stein-Rosenberg demuestra que estos dos métodos son comparables y, en caso de convergencia, podremos decir cuál de los dos es mejor.

Definición 2.2. *Sea $J \geq O$ una matriz cuyos elementos diagonales son nulos. Podemos expresar esta matriz como $J = L + U$, donde L y U son matrices estrictamente triangular inferior y superior, respectivamente. A este tipo de matrices las llamaremos matrices de Jacobi.*

Definición 2.3. Sean L y U matrices estrictamente triangular inferior y superior respectivamente. Para $\sigma \geq 0$ definimos

$$m(\sigma) = \rho(\sigma L + U),$$

$$n(\sigma) = \rho(L + \sigma U).$$

En particular, $m(1) = n(1) = \rho(J)$.

Lema 2.1. Sea $J = L + U \geq O$ la matriz de Jacobi. Si $\rho(J) > 0$ entonces las funciones n y m que acabamos de definir son ambas estrictamente crecientes y si $\rho(J) = 0$ entonces $n(\sigma) = m(\sigma) = 0$ para cualquier $\sigma \geq 0$.

Demostración. Supongamos, en primer lugar, que J es reducible y su forma normal es estrictamente triangular superior. Por el [Teorema 1.10](#) sabemos que $\rho(J) = 0$. Ahora, sea $M(\sigma) = \sigma L + U$. La matriz $M(\sigma)$ también es una matriz reducible para cualquier $\sigma \geq 0$, ya que su grafo dirigido va a coincidir con el de J o, en caso de que $\sigma = 0$ el grafo aún tendrá menos aristas que $G(J)$. La forma normal de $M(\sigma)$ también debe ser estrictamente triangular superior y, por tanto, $\rho(M(\sigma)) = 0$, es decir, $m(\sigma) = 0$.

De forma análoga se prueba que $n(\sigma) = 0$.

Ahora, supongamos que J es irreducible. Por el [Teorema 1.9](#) sabemos que $\rho(J) > 0$. Para $\sigma > 0$, la matriz $M(\sigma)$ que definimos antes es irreducible. Entonces por el [Teorema 1.9](#), $\rho(M(\sigma))$ aumenta si aumentamos algún elemento de $M(\sigma)$, es decir, si aumentamos σ . Por tanto, la función $m(\sigma)$ es estrictamente creciente.

De igual manera se prueba que $n(\sigma)$ es estrictamente creciente. \square

Teorema 2.1 (Stein-Rosenberg). Sea $J=L+U$ la matriz de Jacobi de tamaño $n \times n$ que suponemos no negativa y con todos los elementos diagonales nulos, y sea \mathcal{L}_1 la matriz de iteración de Gauss-Seidel. Entonces una y solamente una de las siguientes relaciones es válida

1) $\rho(J) = \rho(\mathcal{L}_1) = 0$,

2) $0 < \rho(\mathcal{L}_1) < \rho(J) < 1$,

3) $1 < \rho(J) < \rho(\mathcal{L}_1)$,

4) $\rho(J) = \rho(\mathcal{L}_1) = 1$.

En conclusión, la matriz J de Jacobi y la matriz \mathcal{L}_1 convergen o divergen simultáneamente.

Incluimos en este memoria únicamente la demostración del teorema en el caso en que la matriz de iteración del método de Jacobi es, además, irreducible.

Demostración. Sean \mathcal{L}_1 la matriz de Gauss-Seidel y $J = L + U$ la matriz de Jacobi que suponemos no negativa e irreducible. Sabemos que $I - L$ es regular y

$$(I - L)^{-1} = \sum_{r=1}^{\infty} L^r.$$

Como L es estrictamente triangular inferior esta suma es finita,

$$(I - L)^{-1} = I + L + \dots + L^{n-1} \geq O.$$

Por tanto $\mathcal{L}_1 = (I - L)^{-1}U \geq O$. Por el [Teorema 1.10](#) sabemos que existe $\mathbf{x} \geq \mathbf{0}$ de tal forma que

$$\mathcal{L}_1 \mathbf{x} = \lambda \mathbf{x}, \tag{10}$$

siendo $\lambda = \rho(\mathcal{L}_1) \geq 0$. Dado que J es irreducible, sabemos que las matrices $\lambda L + U$ y $L + \frac{1}{\lambda}U$ son también no negativas e irreducibles. Entonces podemos escribir (10) de las dos siguientes formas

$$(\lambda L + U)\mathbf{x} = \lambda \mathbf{x},$$

$$(L + \frac{1}{\lambda}U)\mathbf{x} = \mathbf{x}.$$

De aquí deducimos que $m(\lambda) = \lambda$ y $n(\frac{1}{\lambda}) = 1$. Además, $\lambda > 0$ y $\mathbf{x} > \mathbf{0}$, dado que las matrices $\lambda L + U$ y $L + \frac{1}{\lambda}U$ son irreducibles.

1) Si $\rho(J) = 0$ sabemos que $m(\sigma) = n(\sigma) = 0$ para cualquier $\sigma \geq 0$. Supongamos que existe $\lambda > 0$ tal que

$$\mathcal{L}_1 \mathbf{x} = \lambda \mathbf{x},$$

o equivalentemente,

$$(\lambda L + U)\mathbf{x} = \lambda \mathbf{x}.$$

Entonces λ es autovalor de $\lambda L + U$, es decir,

$$\lambda \leq \rho(\lambda L + U) = m(\lambda) = 0.$$

Lo cual entra en contradicción con el supuesto de que λ sea positivo, luego $\rho(J) = \rho(\mathcal{L}_1) = 0$.

- 2) Supongamos que $0 < \rho(J) < 1$. Sabemos que $n(1) = \rho(J) < 1$, $n(\frac{1}{\lambda}) = 1$ y que n es estrictamente creciente. Luego debe ocurrir que $1 < \frac{1}{\lambda}$ o, equivalentemente, $0 < \lambda < 1$. Como $m(1) = \rho(J)$ y m es estrictamente creciente tenemos que

$$\lambda = m(\lambda) < m(1) = \rho(J).$$

En conclusión, $0 < \rho(\mathcal{L}_1) < \rho(J) < 1$ como queríamos ver.

- 3) Supongamos ahora que $\rho(J) > 1$. Análogamente al caso anterior se tiene que $1 < \rho(J) < \rho(\mathcal{L}_1)$.
- 4) Si $\rho(J) = 1$ sabemos que $m(\sigma) = n(\sigma) = 1$. Supongamos que existe $\lambda > 1$ tal que

$$\mathcal{L}_1 \mathbf{x} = \lambda \mathbf{x}$$

o, equivalentemente,

$$(L + \frac{1}{\lambda}U)\mathbf{x} = \mathbf{x}.$$

Entonces 1 es autovalor de $L + \frac{1}{\lambda}U$, es decir, $1 \leq n(\frac{1}{\lambda})$. Por el carácter estrictamente creciente de n debe ocurrir que $1 < \frac{1}{\lambda}$. Pero esto es imposible si $\lambda > 1$, por tanto $\lambda = 1$ y $\rho(J) = \rho(\mathcal{L}_1) = 1$.

□

Corolario 2.1. *Sea J la matriz de Jacobi. Bajo las hipótesis del Teorema 2.1, si $0 < \rho(J) < 1$, entonces $R_\infty(\mathcal{L}_1) > R_\infty(J)$.*

Demostración. Por el Teorema 2.1 sabemos que se tiene que cumplir que $0 < \rho(\mathcal{L}_1) < \rho(J) < 1$. Teniendo en cuenta la definición de la razón asintótica de convergencia, es claro que $R_\infty(\mathcal{L}_1) > R_\infty(J)$. □

De aquí podemos concluir que, cuando los dos métodos convergen el método de Gauss-Seidel es asintóticamente más rápido que el de Jacobi.

2.3. Determinación del factor de relajación óptimo de SOR

El problema de calcular para el método SOR el factor de relajación que lleva a una mayor velocidad de convergencia, en general, es complicado. De hecho no existen resultados generales para la determinación de este parámetro. En esta sección vamos a deducir cual es el factor de relajación óptimo en el caso particular de matrices 2-cíclicas. Vamos a medir la velocidad de convergencia mediante la razón asintótica de convergencia. El método será mejor cuanto mayor sea su razón asintótica de convergencia, o equivalentemente, cuanto menor sea el radio espectral de la matriz de iteración del método.

Teorema 2.2. Sean $J = L + U$ la matriz de Jacobi y \mathcal{L}_ω la matriz del SOR. Entonces

$$\rho(\mathcal{L}_\omega) \geq |\omega - 1| \text{ para cualquier } \omega,$$

y la igualdad se da solo si todos los autovalores de \mathcal{L}_ω tienen modulo $|\omega - 1|$.

Demostración. Dado que L y U son matrices estrictamente triangular inferior y superior respectivamente, sabemos que $\det(L) = \det(U) = 0$. También es claro que $\det(I - \omega L) = 1$ y, por tanto, $\det((I - \omega L)^{-1}) = 1$. Entonces

$$\det(\mathcal{L}_\omega) = \det((I - \omega L)^{-1}) \det((1 - \omega)I + \omega U) = (1 - \omega)^n.$$

Dado que $\det(\mathcal{L}_\omega) = \prod_{i=1}^n \lambda_i$, siendo λ_i los autovalores de \mathcal{L}_ω , tomando módulos se tiene que

$$|\omega - 1|^n = \prod_{i=1}^n |\lambda_i|.$$

Si existe un índice i de tal forma que $|\lambda_i| < |\omega - 1|$, entonces existirá otro índice j tal que $|\lambda_j| > |\omega - 1|$ y, por tanto, $\rho(\mathcal{L}_\omega) > |\omega - 1|$.

Si $|\lambda_i| = |\omega - 1|$ para todo $1 \leq i \leq n$, es claro que $\rho(\mathcal{L}_\omega) = |\omega - 1|$. \square

Definición 2.4. Sea $A \in M_{n \times n}(\mathbb{C})$. Diremos que A es débilmente 2-cíclica si existe una matriz de permutación P tal que

$$PAP^T = \begin{bmatrix} O & A_{1,2} \\ A_{2,1} & O \end{bmatrix},$$

donde las submatrices diagonales nulas son cuadradas de tamaños $r \times r$ y $(n - r) \times (n - r)$ con $1 \leq r < n$. A esta matriz la llamaremos forma normal de una matriz débilmente 2-cíclica.

Teorema 2.3. Sea $A \geq O$, irreducible y débilmente 2-cíclica. Entonces su polinomio característico es de la forma

$$\Phi(t) = t^m \prod_{i=1}^r (t^2 - \lambda_i^2)$$

con $m + 2r = n$. En consecuencia, A tiene dos autovalores de módulo $\rho(A)$.

Demostración. Basta con ver que si $\lambda \neq 0$ es autovalor de A entonces $-\lambda$ también lo es y con la misma multiplicidad. Supongamos que A está en forma normal. Entonces

$$\begin{aligned} \Phi(z) &= \det(zI - A) = \det \begin{bmatrix} zI_r & -A_{1,2} \\ -A_{2,1} & zI_{n-r} \end{bmatrix} = (-1)^r \det \begin{bmatrix} -zI_r & A_{1,2} \\ -A_{2,1} & zI_{n-r} \end{bmatrix} \\ &= (-1)^r (-1)^{n-r} \det \begin{bmatrix} -zI_r & -A_{1,2} \\ -A_{2,1} & -zI_{n-r} \end{bmatrix} = (-1)^n \det \begin{bmatrix} -zI_r & -A_{1,2} \\ -A_{2,1} & -zI_{n-r} \end{bmatrix} \\ &= (-1)^n \det(-zI - A) = (-1)^n \Phi(-z). \end{aligned}$$

Derivando j veces la relación

$$\Phi(z) = (-1)^n \Phi(-z)$$

tenemos que

$$\Phi^{(j)}(z) = (-1)^{n+j} \Phi^{(j)}(-z).$$

Ahora, λ es autovalor de A de multiplicidad k si y solo si

$$\begin{aligned} \Phi(\lambda) = \Phi'(\lambda) = \dots = \Phi^{(k-1)}(\lambda) = 0, \\ \Phi^{(k)}(\lambda) \neq 0, \end{aligned}$$

o equivalentemente,

$$\begin{aligned} (-1)^n \Phi(-\lambda) = (-1)^{n+1} \Phi'(-\lambda) = \dots = (-1)^{n+k-1} \Phi^{(k-1)}(-\lambda) = 0, \\ (-1)^{n+k} \Phi^{(k)}(-\lambda) \neq 0. \end{aligned}$$

Por tanto, debe ocurrir que

$$\Phi(-\lambda) = \Phi'(-\lambda) = \dots = \Phi^{(k-1)}(-\lambda) = 0$$

y que

$$\Phi^{(k)}(-\lambda) \neq 0,$$

es decir, $-\lambda$ es autovalor de A de multiplicidad k , como queríamos ver. \square

Definición 2.5. Diremos que una matriz A es 2-cíclica si la matriz de Jacobi asociada

$$J = I - D^{-1}A,$$

es débilmente 2-cíclica. Como viene siendo habitual en este capítulo, D es una matriz diagonal cuyos elementos son los elementos diagonales de A .

Definición 2.6. Sea A una matriz 2-cíclica. Decimos que A está consistentemente ordenada si los autovalores de la matriz

$$J(\alpha) = \alpha L + \frac{1}{\alpha} U$$

son independientes de α , para $\alpha \neq 0$. Además, también podemos decir que la matriz de Jacobi, $J = L + U$, es consistentemente ordenada. En caso contrario, diremos que A y J están inconsistentemente ordenadas.

Teorema 2.4. Sea $J = L + U$ una matriz consistentemente ordenada y débilmente 2-cíclica. Para cualquier $\alpha, \beta, \gamma \in \mathbb{C}$ se tiene que

$$\det[\gamma I - (\alpha L + \beta U)] = \det[\gamma I - (\alpha\beta)^{\frac{1}{2}}(L + U)].$$

Demostración. Dado que el determinante de una matriz es el producto de todos sus autovalores, basta ver que $\gamma I - (\alpha L + \beta U)$ y $\gamma I - (\alpha\beta)^{\frac{1}{2}}(L + U)$ tienen los mismos autovalores, o equivalentemente, que los autovalores de $\alpha L + \beta U$ y $(\alpha\beta)^{\frac{1}{2}}(L + U)$ coinciden.

Si $\alpha = 0$ ó $\beta = 0$ se tiene que la matriz $\alpha L + \beta U$ es estrictamente triangular y, por tanto, sus autovalores son nulos. Por otro lado la matriz $(\alpha\beta)^{\frac{1}{2}}(L + U)$ es la matriz nula, cuyos autovalores también son nulos.

Ahora, supongamos que $\alpha \neq 0$ y $\beta \neq 0$ y llamamos $\nu = \sqrt{\alpha/\beta}$. Entonces

$$\sqrt{\alpha\beta} \left[\nu L + \frac{1}{\nu} U \right] = \alpha L + \beta U$$

y, por tanto, los autovalores de estas dos matrices deben coincidir. Como J está consistentemente ordenada sabemos que los autovalores de $\nu L + \frac{1}{\nu} U$ son independientes de ν . En particular, si tomamos $\nu = 1$ tenemos que los autovalores de $\alpha L + \beta U$ y $\sqrt{\alpha\beta}(L + U)$ son iguales, como queríamos ver. \square

El problema que planteamos en esta sección consiste en minimizar $\rho(\mathcal{L}_\omega)$ cuando $\rho(\mathcal{L}_\omega)$ se ve como función de ω . Al valor de ω para el que se alcanza dicho mínimo lo llamaremos ω óptimo y lo denotaremos por ω_{op} .

Teorema 2.5. *Sea A una matriz 2-cíclica de tamaño $n \times n$, consistentemente ordenada y cuyos elementos diagonales son no nulos. Para $\omega \neq 0$, si $\lambda \neq 0$ es un autovalor de \mathcal{L}_ω y μ cumple que*

$$(\lambda + \omega - 1)^2 = \lambda\omega^2\mu^2, \quad (11)$$

entonces μ es autovalor de la matriz de Jacobi. Recíprocamente, si μ es un autovalor de la matriz de Jacobi y λ satisface (11), entonces λ es un autovalor de \mathcal{L}_ω .

Demostración. Sabemos que $I - \omega L$ es regular y $\det(I - \omega L) = 1$ debido al carácter estrictamente triangular de L . Por tanto,

$$\begin{aligned} \Phi(\lambda) = \det(\lambda I - \mathcal{L}_\omega) &= \det(I - \omega L) \det(\lambda I - \mathcal{L}_\omega) \\ &= \det((I - \omega L)(\lambda I - \mathcal{L}_\omega)) \\ &= \det((I - \omega L)\lambda I - \omega U - (1 - \omega)I) \\ &= \det((\lambda + \omega - 1)I - \omega U - \omega\lambda L). \end{aligned}$$

Por el [Teorema 2.4](#) sabemos que

$$\Phi(\lambda) = \det((\lambda + \omega - 1)I - \omega\lambda^{\frac{1}{2}}J).$$

Dado que A es 2-cíclica, la matriz J es débilmente 2-cíclica y, por tanto, $\omega\lambda^{\frac{1}{2}}J$ también. Si μ_i , $1 \leq i \leq n$, es autovalor de J , según el [Teorema 2.3](#), tenemos que

$$\Phi(\lambda) = (\lambda + \omega - 1)^m \prod_{i=1}^r [(\lambda + \omega - 1)^2 - \lambda\omega^2\mu_i^2],$$

con $m + 2r = n$.

Supongamos que $\omega \neq 0$. Sea $\lambda \neq 0$ un autovalor de \mathcal{L}_ω , entonces alguno de los factores de $\Phi(\lambda)$ debe ser nulo. Sea $\mu \neq 0$ cumpliendo (11). Es claro que $\lambda + \omega - 1 \neq 0$ y, por tanto, debe ocurrir que $(\lambda + \omega - 1)^2 - \lambda\omega^2\mu_i^2 = 0$ para algún $1 \leq i \leq r$. Deben cumplirse las ecuaciones

$$\begin{aligned} (\lambda + \omega - 1)^2 &= \lambda\omega^2\mu_i^2, \\ (\lambda + \omega - 1)^2 &= \lambda\omega^2\mu^2, \end{aligned}$$

luego $\lambda\omega^2(\mu_i^2 - \mu^2) = 0$. Entonces debe cumplirse que

$$\mu = \pm\mu_i$$

y, por tanto, μ es autovalor de J por el [Teorema 2.3](#).

En caso de que $\mu = 0$ y cumpla (11) hay que ver que $\mu = 0$ es autovalor de J . Como λ es autovalor de \mathcal{L}_ω tiene que cumplirse que $\Phi(\lambda) = 0$, es decir,

$$\Phi(\lambda) = \det((\lambda + \omega - 1)I - \omega\lambda^{\frac{1}{2}}J) = 0. \quad (12)$$

Dado que $\mu = 0$ sabemos que $\lambda + \omega - 1 = 0$ y podemos reescribir (12) de la siguiente manera

$$\Phi(\lambda) = \det(-\omega\lambda^{\frac{1}{2}}J) = 0.$$

De donde deducimos que $\mu = 0$ tiene que ser autovalor de J porque tanto λ como ω son no nulos.

Recíprocamente, si μ es un autovalor de J y λ cumple (11), es claro que alguno de los factores $(\lambda - \omega - 1)^2 - \lambda\omega^2\mu_i^2$ es nulo. Se tiene que $\Phi(\lambda) = 0$ y, en conclusión, λ es autovalor de \mathcal{L}_ω . \square

Teorema 2.6. *Sea A una matriz 2-cíclica, consistentemente ordenada y cuyos elementos diagonales son no nulos. Si la matriz de Jacobi asociada a la matriz A converge y tiene todos sus autovalores reales, entonces*

$$\omega_{op} = \frac{2}{1 + \sqrt{1 - \rho^2(B)}} \quad y \quad \rho(\mathcal{L}_{\omega_{op}}) = \omega_{op} - 1.$$

Demostración. Según el [Teorema 2.5](#) los autovalores de las matrices de Jacobi y del SOR se relacionan mediante la igualdad

$$(\lambda + \omega - 1)^2 = \lambda\omega^2\mu^2,$$

siendo μ un autovalor de J con $0 < \mu \leq \rho(J) < 1$. Tomando raíces tenemos

$$(\lambda + \omega - 1) = \pm\lambda^{\frac{1}{2}}\omega\mu.$$

Definimos la funciones

$$g_\omega(\lambda) = \frac{\lambda + \omega - 1}{\omega} \text{ para } \omega \neq 0,$$

$$f_\mu(\lambda) = \pm\lambda^{\frac{1}{2}}\mu.$$

Es claro que g_ω es una recta como función de λ y, además pasa por el punto $(1,1)$. Su pendiente $(1/\omega)$ es positiva y decrece cuando ω aumenta. Al igual que la pendiente, la mayor abscisa de los dos puntos de intersección de g_ω y f_μ también decrece al aumentar ω hasta el momento en que g_ω se vuelve tangente a la rama positiva de f_μ . Sea $(\tilde{\lambda}, \tilde{\omega})$ el punto de tangencia. En ese punto debe ocurrir que las funciones y sus derivadas coinciden, es decir,

$$\frac{\tilde{\lambda} + \tilde{\omega} - 1}{\tilde{\omega}} = \tilde{\lambda}^{\frac{1}{2}}\mu, \quad (13)$$

$$\frac{1}{\tilde{\omega}} = \frac{1}{2}\tilde{\lambda}^{-\frac{1}{2}}\mu. \quad (14)$$

Despejando $\tilde{\lambda}$ de (14) y sustituyendo en (13) llegamos a la siguiente ecuación de segundo grado en $\tilde{\omega}$

$$\mu^2\tilde{\omega}^2 - 4\tilde{\omega} + 4 = 0.$$

Resolviendo obtenemos que

$$\tilde{\omega} = \frac{2}{1 + \sqrt{1 - \mu^2}}.$$

Despejando μ de (14) y sustituyendo en (13) obtenemos la abscisa del punto de tangencia, $\tilde{\lambda} = \tilde{\omega} - 1$. Si $\omega > \tilde{\omega}$ ya no tendríamos punto de corte con la rama positiva de f_μ . Entonces las raíces de (13) serían números complejos conjugados de módulo $\omega - 1$. Vamos a considerar las curvas $f_{\rho(J)}(\lambda)$ y $f_\mu(\lambda)$ siendo $0 \leq \mu < \rho(J)$, que están representadas en la [Figura 2](#) por las parábolas azul y roja, respectivamente. Es claro que la parábola correspondiente a $\rho(J)$

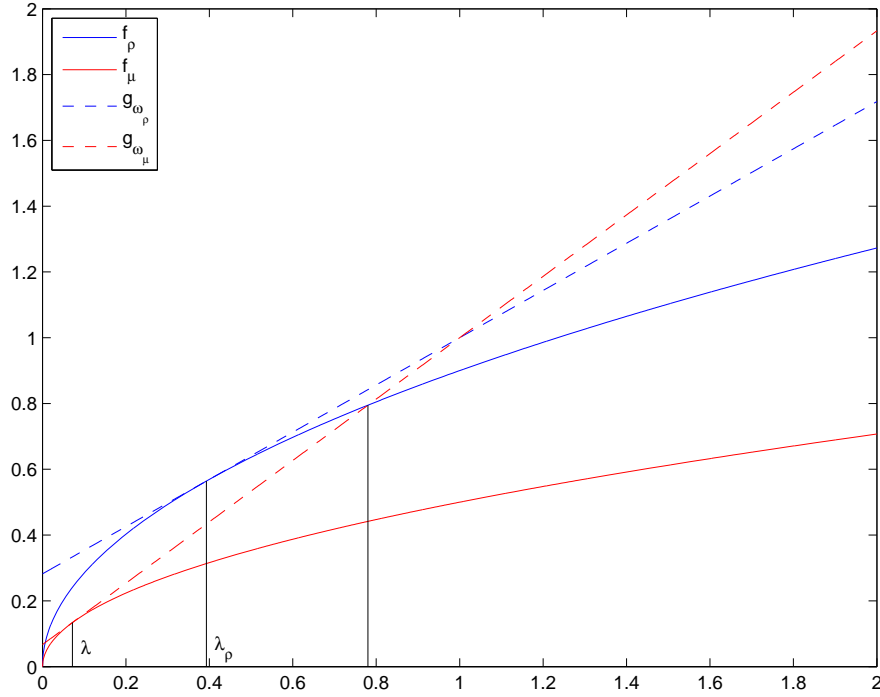


Figura 2: Gráficas de las funciones f_μ y g_ω

es más abierta que la correspondiente a otro autovalor de menor módulo μ . Sean $\tilde{\lambda}_{\rho(J)}$ y $\tilde{\lambda}$ las abscisas de los puntos de corte de $f_{\rho(J)}$ con $g_{\omega_{\rho(J)}}$ y de f_μ con g_{ω_μ} , respectivamente. Como podemos observar en la [Figura 2](#), $\tilde{\lambda}_\rho > \tilde{\lambda}$. Por otro lado, la recta $g_{\omega_\mu}(\lambda)$ corta también a la curva $f_{\omega_{\rho(J)}}$ pero lo hace en un punto cuya abscisa es mayor que $\tilde{\lambda}_\rho$. Entonces debe ser

$$\omega_{op} = \frac{2}{1 + \sqrt{1 - \rho^2(B)}}$$

y

$$\rho(\mathcal{L}_{\omega_{op}}) = \omega_{op} - 1.$$

□

3. Aplicaciones

3.1. El cálculo del PageRank de Google

Como bien conocemos, Google es el buscador mas popular y mas utilizado del planeta. Tiene en su base de datos miles de millones de páginas web y atiende mas de 3000 millones de consultas diarias por parte de los usuarios de internet [10]. En el diseño del buscador hay muchos aspectos a tener en cuenta como el almacenamiento de toda la información, la gestión de las consultas de los usuarios o la búsqueda dentro de las bases de datos, y todas estas acciones deben realizarse de forma eficiente. Nosotros en esta subsección vamos a tratar otra cuestión. Supongamos que tras la búsqueda realizada se han encontrado n páginas que contienen información que puede resultar útil al usuario. El problema es,

¿En qué orden se muestran los resultados?

La solución: el algoritmo de PageRank. La única información que necesita dicho algoritmo son las paginas web, las cuales visualizaremos como nodos de un grafo dirigido ponderado, y los enlaces que hay de unas a otras que serán las aristas del grafo.

Asociado al grafo tendremos la matriz de incidencia $A = [a_{i,j}]$ definida por

$$a_{i,j} = \begin{cases} 1 & \text{si la página } i \text{ enlaza a la página } j, \\ 0 & \text{en otro caso.} \end{cases}$$

Supongamos que el usuario navega aleatoriamente por la red. En el instante inicial el usuario se sitúa aleatoriamente en una de las n páginas y va moviéndose a páginas a las que enlaza la página que actualmente está visitando. Se supone que cada uno de los enlaces tiene la misma probabilidad de ser elegido por el usuario. Construimos en primer lugar la matriz $P = [p_{i,j}]$ donde $p_{i,j}$ es la probabilidad de que en un *click* el usuario vaya de la página i a la página j . Es decir, si desde la página i hay enlaces a n_i páginas, entonces $p_{i,j} = \frac{1}{n_i}$ para cada j con $a_{i,j} = 1$, y si la página i no tiene enlaces a ninguna otra página entonces $p_{i,j} = 0$ para $1 \leq j \leq n$. Esta matriz es dispersa, se construye fácilmente a partir de la matriz de incidencia A pero, en general, no es irreducible dado que en la práctica hay páginas que no contienen enlaces a ninguna otra.

Para intentar conseguir una matriz irreducible definimos a continuación

$$S = P^T + \frac{1}{n} \mathbf{e} \mathbf{w}^T,$$

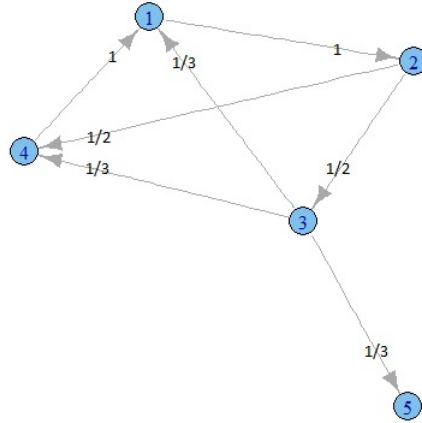


Figura 3: Ejemplo de una red con 5 páginas web. En las aristas aparece la probabilidad que hay de pasar de cada página a las que tienen enlace desde ella.

donde \mathbf{e} es el vector columna de R^n con todos los elementos iguales a uno y $\mathbf{w} = (w_1, \dots, w_n)^T$ está definido por

$$w_i = \begin{cases} 1 & \text{si la fila } i\text{-ésima de } P \text{ es nula,} \\ 0 & \text{en otro caso.} \end{cases}$$

Esta matriz S sigue siendo dispersa aunque no tendrá tantos elementos nulos como P . El término que se añade a la matriz P^T para construir la matriz S refleja el hecho de que si un usuario está en una página que no tiene enlaces a ninguna otra, se moverá a cualquiera de las n páginas de la red con la misma probabilidad $1/n$. Además, de este modo, se conseguirá que la matriz S sea una matriz estocástica, es decir, $0 \leq s_{i,j} \leq 1$ para $1 \leq i, j \leq n$ y $S\mathbf{e} = \mathbf{e}$.

Veamos que interpretación podemos darle a esta matriz S . Si denotamos por \mathbf{q} al vector cuya componente i -ésima es la probabilidad de que el usuario esté visitando la página i en el momento actual, la componente i -ésima del producto $S\mathbf{q}$ representa la probabilidad de que el usuario esté visitando la página i después de un *click*.

En un tercer paso, se define la matriz de Google como

$$G = \alpha S + (1 - \alpha) \frac{1}{n} \mathbf{e}\mathbf{e}^T$$

donde α es un parámetro real que toma un valor entre 0 y 1. El parámetro α controla la prioridad que se le da a la estructura de enlaces de la web frente a

la idea de que desde cualquier página se puede ir en un *click* a cualquier otra con probabilidad igual a $\frac{1}{n}$. La matriz G es no negativa, ya no es dispersa, sino llena, sigue siendo estocástica y, además, es irreducible. Por el carácter no negativo e irreducible de G los [Teoremas 1.7](#) y [1.9](#) garantizan que 1 es el autovalor dominante, que es simple y, por tanto, que el método de la potencia aplicado a G converge hacia él. Además, el autovector asociado también es no negativo, y normalizado adecuadamente, es un vector de probabilidad con las probabilidades de que un usuario esté visitando cada una de las n páginas de la red cuando el número de *clicks* se hace grande. Las páginas con mayor probabilidad de ser visitadas deben considerarse como más importantes en la ordenación de los resultados de una búsqueda. El vector de PageRank es, por tanto, el autovector asociado al autovalor 1 y, por tanto, satisface que

$$G\boldsymbol{\pi} = \boldsymbol{\pi}. \quad (15)$$

En cuanto a la computación práctica de dicho vector $\boldsymbol{\pi}$, utilizaremos el comando `sparse` para realizar los cálculos eficientemente y no vamos a calcular expresamente la matriz G porque perderíamos las ventajas del carácter disperso de la matriz de incidencia del grafo de la red y sería mucho más costoso aplicar el método de la potencia. Para la ilustración numérica que incluimos en esta subsección, hemos generado un código en Matlab, `copyright.m`, para construir la matriz P de probabilidades y el vector \mathbf{w} a partir de la matriz de adyacencia del grafo de una red.

Hemos tomado datos reales reducidos encontrados en [\[7\]](#):

<http://www.cs.toronto.edu/~tsap/experiments/datasets/>

Trataremos aproximadamente con 2600 páginas web. Sin más que utilizar el comando `spy` de Matlab, podemos ver la estructura de ceros de la matriz P con la que hemos trabajado. Solo los puntos azules representan elementos no nulos de la matriz. Poco más del 1 % de los más de seis millones de elementos de la matriz es no nulo.

En la función `pagerank.m` hemos programado el método de la potencia de una manera eficiente, aprovechando el carácter disperso de la matriz P y las herramientas con las que cuenta Matlab para trabajar con este tipo de matrices [\[5\]](#). Nuestro problema consiste en encontrar el autovector asociado al autovalor dominante, es decir, encontrar $\boldsymbol{\pi}$ de tal forma que se cumpla [\(15\)](#). Al aplicar el método de la potencia es necesario calcular productos $G\boldsymbol{\pi}^{(k)}$, donde $\boldsymbol{\pi}^{(k)}$ será la aproximación al vector $\boldsymbol{\pi}$ tras k iteraciones. Para

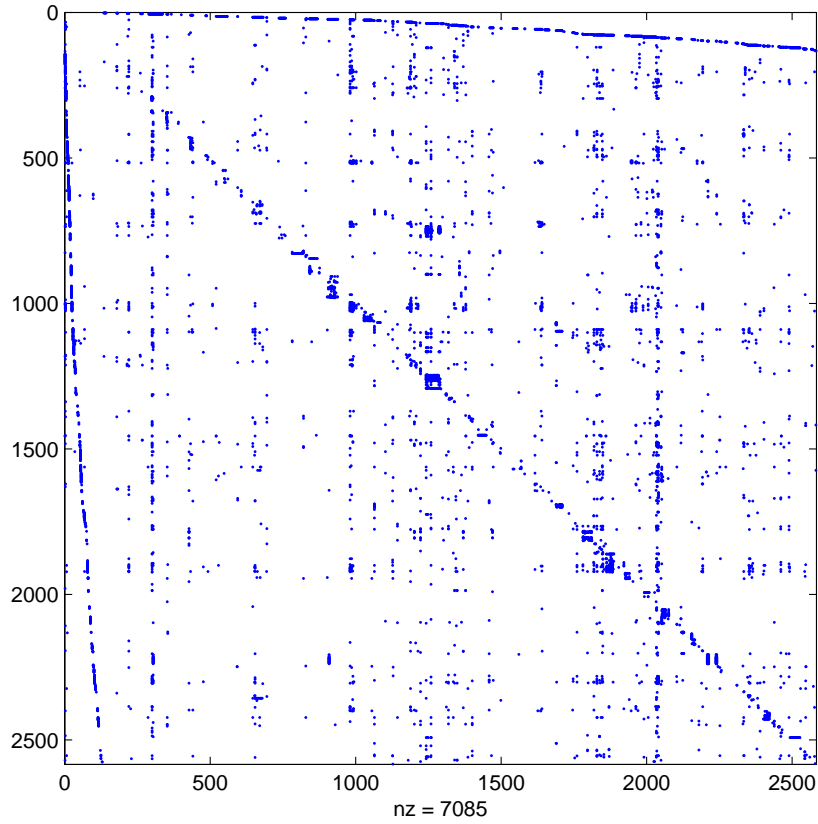


Figura 4: Estructura de dispersión de la matriz P

ello escribimos $G\boldsymbol{\pi}^{(k)}$ como

$$\begin{aligned}
 G\boldsymbol{\pi}^{(k)} &= \alpha P^T \boldsymbol{\pi}^{(k)} + \alpha \frac{1}{n} \mathbf{e} \mathbf{w}^T \boldsymbol{\pi}^{(k)} + (1 - \alpha) \frac{1}{n} \mathbf{e} \mathbf{e}^T \boldsymbol{\pi}^{(k)} \\
 &= \alpha P^T \boldsymbol{\pi}^{(k)} + \alpha \frac{1}{n} \mathbf{e} (\mathbf{w}^T \boldsymbol{\pi}^{(k)}) + (1 - \alpha) \frac{1}{n} \mathbf{e} (\mathbf{e}^T \boldsymbol{\pi}^{(k)}) \\
 &= \alpha P^T \boldsymbol{\pi}^{(k)} + \frac{1}{n} [(\alpha \mathbf{w}^T + (1 - \alpha) \mathbf{e}^T) \boldsymbol{\pi}^{(k)}] \mathbf{e}
 \end{aligned}$$

y así las correspondientes operaciones se harán de la forma mas eficiente posible y aprovechando el carácter disperso de las matrices.

Una vez calculada una aproximación al autovector dominante, cuya componente i -ésima indica la importancia de la página web i , el programa **pagerank.m** da como salida un vector numérico con los índices de las páginas web ordenados por orden de importancia decreciente de las mismas, es decir, la página cuyo índice aparece en la primera posición del vector es la mas impor-

tante y será la primera que aparezca en el resultado de la búsqueda realizada con Google.

Tomando como parámetros de entrada $\alpha = 0.85$ (el valor sugerido en [7]), $tol = 10^{-6}$ (para parar la iteración en el método de la potencia) y $N = 100$ (número máximo de iteraciones, por si la convergencia fuese demasiado lenta), mostramos a continuación la salida, reducida a las 20 páginas web más importantes.

niter =

49

page =

6
1248
300
1242
1064
1
2041
54
990
1246
1191
13
981
1223
2353
2035
220
350
2352
15

El parámetro α , además de controlar la prioridad que se le da a la estructura de enlaces de la web, también influye en la razón asintótica de convergencia del método de la potencia. Considerando el valor 0.85 el algoritmo será eficiente y efectivo. Nos gustaría que el algoritmo tratara con una estructura de la web lo más parecida a la realidad, pero cuando $\alpha \rightarrow 1$ el método

de la potencia necesita muchas mas iteraciones para encontrar la solución. Tomando como tolerancia 10^{-6} , en la siguiente tabla podemos ver cómo va aumentando el numero de iteraciones necesarias cuando α se aproxima a 1.

α	Numero de iteraciones
0.5	12
0.7	23
0.85	49
0.95	152
0.99	770
0.999	7726

La velocidad de convergencia del método de la potencia depende del cociente $|\frac{\lambda_2}{\lambda_1}|$ entre el módulo del segundo y del primer autovalor dominante de la matriz. Cuanto menor sea este cociente la convergencia será más rápida (Langville, 2006, p. 42).

Si variamos ligeramente el parámetro α , por ejemplo, $\alpha = 0.7$, y ejecutamos de nuevo nuestro programa, vemos que el orden de importancia de las páginas web no varía mucho. De hecho de las primeras 20 páginas obtenidas como más importantes para los dos valores de α considerados, 17 son comunes a ambos listados. La siguiente salida es la que se obtiene cuando $\alpha = 0.7$

niter =

23

page =

6
300
1248
1064
1242
54
1
2041
13
981
15

2155
 1223
 2
 2035
 1191
 23
 350
 990
 220

El algoritmo que utiliza Google actualmente para calcular el vector PageRank incorpora al algoritmo básico que acabamos de describir diversas mejoras que están fuera del alcance del trabajo que presentamos.

3.2. Análisis de resultados de competiciones deportivas

En los deportes hay muchos factores que se pueden tener en cuenta a la hora de decidir un ranking de los equipos participantes en una competición.

En esta subsección vamos a utilizar la teoría de Perron-Frobenius y, de nuevo, el método de la potencia, para elaborar un ranking de los n equipos participantes en una competición de fútbol en la cual no se enfrentan todos contra todos. Ideas similares han aparecido en [6] para la liga de fútbol universitario americana y en [4] para el tenis. La idea que hay detrás del ranking que vamos a elaborar es asignar a cada equipo una puntuación que tenga en cuenta tanto el resultado de sus enfrentamientos con los otros equipos, como la fortaleza de los rivales a los que se ha enfrentado. Si suponemos que $\mathbf{r} = [r_j]$, $1 \leq j \leq n$, es el vector de ranking de los distintos equipos, podemos definir la puntuación del equipo i -ésimo como

$$p_i = \sum_{j=1}^n a_{i,j} r_j \text{ para } 1 \leq i \leq n,$$

siendo $a_{i,j}$ valores no negativos dependientes del resultado del enfrentamiento entre el equipo i y el equipo j .

Siguiendo [6], para definir los coeficientes a_{ij} vamos a tener en cuenta el número de goles que marca cada equipo a cada uno de los rivales, puesto que en caso contrario, un equipo que gane su partido 7-0 sumaría a su puntuación lo mismo que un equipo que ha ganado 1-0 al mismo rival. Si denotamos por $S_{i,j}$ el número de goles que marca el equipo i al equipo j podemos elegir [6]

$$a_{i,j} = \frac{S_{i,j} + 1}{S_{i,j} + S_{j,i} + 2}. \tag{16}$$

Esta elección puede tener una debilidad, dado que un equipo puede aprovechar un partido fácil para marcar muchos goles y así aumentar mucho su puntuación. Para paliar este efecto tomaremos, siguiendo de nuevo [6],

$$a_{i,j} = h\left(\frac{S_{i,j} + 1}{S_{i,j} + S_{j,i} + 2}\right), \quad (17)$$

siendo

$$h(x) = \frac{1}{2} + \frac{1}{2} \operatorname{sgn}\left(x - \frac{1}{2}\right) \sqrt{|2x - 1|}.$$

En el caso de que dos equipos i y j no se hayan enfrentado en [6] se sugiere mantener los coeficientes $a_{i,j} = a_{j,i} = h\left(\frac{1}{2}\right) = \frac{1}{2}$ (como si se hubieran enfrentado y se hubiera producido un empate a 0). Nosotros sugerimos incorporar un parámetro α (análogo al parámetro utilizado en la determinación del PageRank de Google) que tome valores entre 0 y 0.5. El papel del parámetro α es asignar a los equipos una pequeña puntuación en caso de no enfrentamiento entre ellos, que no se confunda con el $\frac{1}{2}$ que se asignaría en caso de empate real. En los experimentos numéricos hemos utilizado distintos valores de α y veremos en la [Tabla 1](#) qué repercusión tiene en la ordenación final. En el caso de que $i = j$ consideramos el valor $a_{i,i} = 0$ porque un equipo no se enfrenta a si mismo. Como podemos observar, hay muchos detalles que se pueden tener en cuenta, que van mejorando poco a poco el algoritmo y cabría esperar que al añadir esta mejoras los resultados fueran cada vez más cercanos a los reales.

Una vez establecidas las puntuaciones de cada enfrentamiento, si consideramos que el ranking final debería ser proporcional a la puntuación, tendríamos que encontrar un vector \mathbf{r} tal que

$$A\mathbf{r} = \lambda\mathbf{r},$$

es decir, \mathbf{r} tendrá que ser autovector de la matriz $A = [a_{i,j}]$ no negativa e irreducible que se ha definido anteriormente y que tiene en cuenta los resultados de los enfrentamientos. El teorema de Perron-Frobenius se puede aplicar y asegura que el autovector asociado al autovalor dominante satisface $\mathbf{r} \geq \mathbf{0}$. Para calcular el vector de ranking \mathbf{r} hemos implementado la función **champions.m** que utiliza el método de la potencia para calcular el autovector asociado al autovalor dominante.

Nosotros hemos trabajado con los resultados de la Champions League de la temporada 2014-2015 en la que participaron 32 equipos.

Clasificación real	$\alpha = 0.1$ y (16)	$\alpha = 0.1$ y (17)	$\alpha = 0.5$ y (16)	$\alpha = 0.5$ y (17)
Barcelona	Barcelona	Barcelona	Barcelona	Barcelona
Juventus	Juventus	Juventus	Bayern	Real Madrid
Bayern	Bayern	Real Madrid	Real Madrid	Bayern
Real Madrid	Real Madrid	Bayern	Chelsea	Chelsea
Oporto	At.Madrid	At.Madrid	Oporto	At.Madrid
PSG	Oporto	Chelsea	At.Madrid	Juventus
At.Madrid	Monaco	Oporto	Juventus	Oporto
Monaco	Chelsea	Monaco	Arsenal	Arsenal
Chelsea	PSG	Arsenal	Br.Dortmund	Ath.Bilbao
Arsenal	Arsenal	Leverkusen	Ath.Bilbao	Br.Dortmund
Schalke	Br.Dortmund	PSG	PSG	PSG
Br.Dortmund	Leverkusen	Br.Dortmund	Monaco	Leverkusen
Man.City	Man.City	Man.City	Leverkusen	Monaco
Basilea	Schalke	Ath.Bilbao	Man.City	Sp.Lisboa
Leverkusen	Ath.Bilbao	Basilea	Sp.Lisboa	Man.City
Shakhtar	Shakhtar	Shakhtar	Anderlecht	Anderlecht
Sp.Lisboa	Basilea	Schalke	Olympiacos	Basilea
Olympiacos	Olympiacos	Olympiacos	Ajax	Olympiacos
Ath.Bilbao	Sp.Lisboa	Sp.Lisboa	Zenit	Zenit
Anderlecht	Anderlecht	Benfica	Shakhtar	Ajax
Ajax	Ajax	Anderlecht	Basilea	Shakhtar
Roma	Zenit	Zenit	Schalke	Benfica
Ludogorets	Benfica	Ajax	Benfica	Roma
Maribor	Roma	Roma	Roma	Schalke
Galatasaray	CSK Moscu	CSK Moscu	Liverpool	Liverpool
CSK Moscu	Liverpool	Liverpool	CSK Moscu	CSK Moscu
Liverpool	Maribor	Maribor	Maribor	Maribor
Zenit	Ludogorets	Malmoe	Ludogorets	Malmoe
Benfica	Malmoe	Ludogorets	Galatasaray	Ludogorets
Malmoe	Galatasaray	Galatasaray	Malmoe	Galatasaray
Bate Borisov	Apoel	Bate Borisov	Apoel	Bate Borisov
Apoel	Bate Borisov	Apoel	Bate Borisov	Apoel

Tabla 1: Clasificaciones de la Champions League obtenidas con diferentes valores de los parámetros

A la hora de comparar las clasificaciones, más que tener en cuenta la posición de cada equipo (pues la propia competición no los clasifica) nos

hemos fijado en la fase de la competición hasta la que ha llegado cada equipo y hemos identificado cada una de estas fases con colores según indica la [Tabla 2](#).

Ganador	1º
Finalista	2º
Semifinales	3º-4º
Cuartos	5º-8º
Octavos	9º-16º
Grupos	17º-32º

Tabla 2: Asignación de colores por posiciones

La función **champions.m** tiene como parámetros de entrada el parámetro α , un indicador que permite elegir si se realizan los cálculos con (16) o (17), la tolerancia para parar la iteración del método de la potencia y el número máximo de iteraciones permitidas. En primer lugar, a partir de los resultados de los partidos jugados, se construye la matriz a la que vamos a aplicar el método de la potencia, siguiendo los pasos descritos anteriormente y utilizando (16) o (17) según indique el correspondiente parámetro de entrada. Después con la función **potencia.m** calculamos el autovalor dominante y el autovector asociado, que es el vector de *ranking*. Ordenado de forma decreciente dicho vector da la clasificación para los equipos.

En los experimentos numéricos hemos utilizado como número máximo de iteraciones permitidas 100, como tolerancia para parar la iteración del método de la potencia 10^{-6} y distintos valores del parámetro α entre 0 y 0.5. Para definir la matriz A hemos utilizado tanto (16) como (17). La [Tabla 1](#) muestra los resultados obtenidos para $\alpha = 0.1$ y $\alpha = 0.5$ (valor sugerido en [6]) con las dos opciones (16) y (17). En la primera columna se ha incluido la clasificación real de la competición. Se puede observar que los resultados que mejor se ajustan a la realidad son los obtenidos con $\alpha = 0.1$ y, dentro de estos, no parece necesario utilizar (17).

3.3. Discretización del laplaciano en un dominio con forma de L

En esta subsección presentamos un ejemplo en el que son de aplicación los resultados teóricos presentados en la Sección 2. Nuestro objetivo es calcular el estado estacionario de temperaturas de una placa en forma de L. Para ello debemos resolver una ecuación en derivadas parciales, que vamos a aproximar

numéricamente utilizando diferencias divididas de segundo orden sobre una malla uniforme de puntos $(x_i, y_j) = (ih, jh)$, con los índices i, j variando adecuadamente para recorrer todos los nodos interiores del dominio con forma de L y siendo $h = \frac{1}{n}$ el diámetro de la malla. Estas aproximaciones las calcularemos resolviendo con los diferentes métodos iterativos estudiados un sistema lineal cuya dimensión crece enormemente cuando h tiende a 0.

El dominio Ω donde queremos resolver la ecuación de Poisson es el cuadrado $[0,2] \times [0,2]$ al que le quitamos el cuadrado de vértices $(1,1), (1,2), (2,1), (2,2)$. Vamos a resolver

$$u_{xx} + u_{yy} = f(x, y), \quad (x, y) \in \mathring{\Omega} \quad (18)$$

con

$$u(x, y) = g(x, y), \quad (x, y) \in \partial\Omega, \quad (19)$$

donde f y g están definidas de la siguiente forma

$$f(x, y) = \frac{\alpha}{2} \left(\frac{\alpha}{2} - 1 \right) (x^2 + y^2)(xy)^{\frac{\alpha}{2}-2},$$

$$g(x, y) = (xy)^{\frac{\alpha}{2}}.$$

De este modo la solución exacta es conocida $u(x, y) = (xy)^{\frac{\alpha}{2}}$. El parámetro α da cuenta de la regularidad de la solución exacta de (18)–(19) y, por tanto, del orden de convergencia de las aproximaciones numéricas calculadas. En los experimentos numéricos se ha tomado α igual a 9.

Si denotamos por $U_{i,j}$ la aproximación a la solución de (18) en el nodo (x_i, y_j) y escribimos $F_{i,j} = f(x_i, y_j)$, el sistema lineal que tenemos que resolver es el siguiente:

$$\frac{U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1} - 4U_{i,j}}{h^2} = F_{i,j}, \quad (20)$$

con los índices i, j variando adecuadamente para recorrer todos los nodos interiores del dominio con forma de L. El orden utilizado tanto para los nodos como para resolver las ecuaciones en los métodos iterativos es el orden natural (de izquierda a derecha y de abajo hacia arriba). Los valores $U_{i,j}$ correspondientes a nodos en la frontera del dominio no son incógnitas puesto que deben satisfacer la condición frontera (19). Dichos valores pasarían a formar parte del término independiente del sistema lineal si se escribiese en el formato general $\mathbf{Ax} = \mathbf{b}$.

Hemos programado en Matlab las funciones **Jacobi.m**, **GaussSeidel.m** y **SOR.m** que implementan los métodos de Jacobi, Gauss-Seidel y SOR, respectivamente, para resolver este sistema lineal sin construir explícitamente la matriz del sistema, sino utilizando (20).

Vamos a ilustrar en primer lugar sobre este ejemplo el teorema de Stein-Rosenberg. Para ello vamos a considerar diferentes valores del diámetro de la red $\frac{1}{n}$, vamos a fijar una tolerancia para detener la iteración y compararemos el número de iteraciones necesarias para obtener aproximaciones con errores inferiores a dicha tolerancia con los métodos de Jacobi y Gauss-Seidel. Los métodos de Jacobi y Gauss-Seidel aplicados a (20) con la ordenación natural de los nodos se escriben como

$$U_{i,j}^{(m+1)} = \frac{U_{i+1,j}^{(m)} + U_{i-1,j}^{(m)} + U_{i,j+1}^{(m)} + U_{i,j-1}^{(m)} - h^2 F_{i,j}}{4} \quad \text{para } m \geq 0,$$

$$U_{i,j}^{(m+1)} = \frac{U_{i+1,j}^{(m)} + U_{i-1,j}^{(m+1)} + U_{i,j+1}^{(m)} + U_{i,j-1}^{(m+1)} - h^2 F_{i,j}}{4} \quad \text{para } m \geq 0,$$

respectivamente, donde los superíndices (m) y $(m+1)$ se refieren al número de iteraciones realizadas. Como iterante inicial (para $m=0$) se toma el que tiene nulos todos los elementos correspondientes a nodos en el interior del dominio.

En la siguiente tabla observamos el número de iteraciones necesarias para cada uno de los métodos si tomamos la tolerancia proporcional al error de discretización $\frac{h^2}{100}$.

n	Nº iteraciones Jacobi	Nº iteraciones Gauss-Seidel
4	34	21
8	130	81
16	510	311
32	2060	1212

En todos los casos, el número de iteraciones que necesita el método de Jacobi es superior al número de iteraciones requeridas por el método de Gauss-Seidel. Este hecho está de acuerdo con el apartado 2 del Teorema 2.1 que establece que cuando los dos métodos convergen, el radio espectral de la matriz de iteración de Gauss-Seidel es menor que el radio espectral de la matriz de iteración de Jacobi. El teorema de Stein-Rosenberg es de aplicación en este ejemplo porque la matriz de Jacobi es no negativa (ver (20)) y con elementos diagonales nulos.

En segundo lugar, vamos a estimar numéricamente el valor del factor de sobrerelajación óptimo del método SOR. El método SOR aplicado a (20) con la ordenación natural de los nodos se escribe como

$$U_{i,j}^{(m+1)} = \omega \frac{U_{i+1,j}^{(m)} + U_{i-1,j}^{(m+1)} + U_{i,j+1}^{(m)} + U_{i,j-1}^{(m+1)} - h^2 F_{i,j}}{4} + (1 - \omega)U_{i,j}^{(m)}.$$

Para estimar el parámetro óptimo ω_{op} hemos ejecutado **SOR.m** con valores de ω equiespaciados entre 0 y 2 (el intervalo en el que, según la [Teorema 2.2](#), tiene que estar ω para que el método sea convergente) y hemos dibujado en una gráfica el número de iteraciones frente al valor de ω utilizado en cada caso.

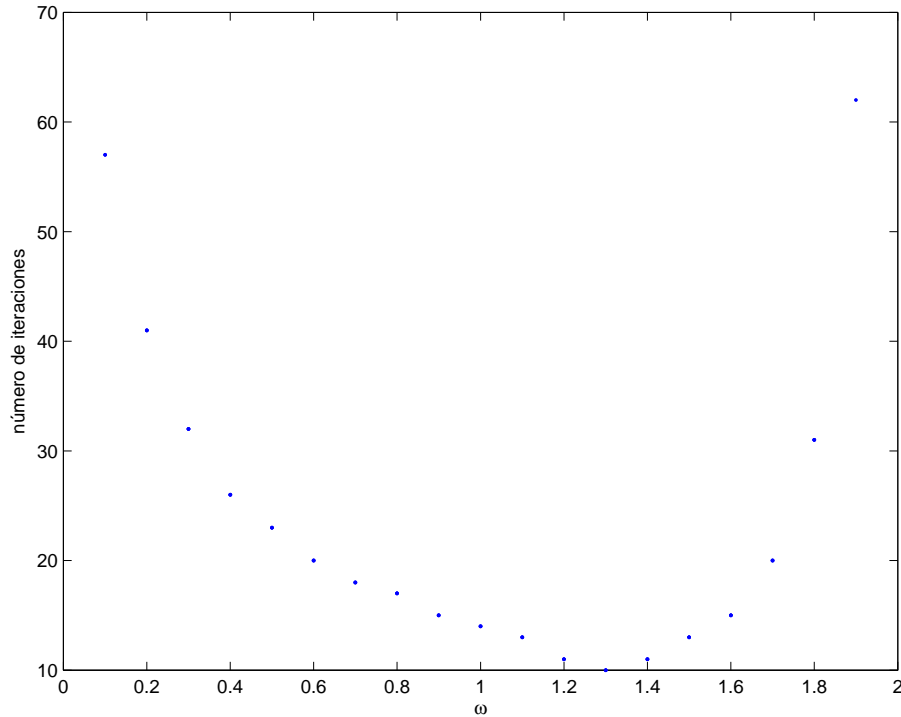


Figura 5: Número de iteraciones en función de ω para $n = 4$

En la [Figura 5](#) se han representado los datos obtenidos para $n = 4$ y $\omega = 0.1, 0.2, \dots, 1.9$. Podemos observar que el valor de ω , de entre los utilizados, para el que se realizan menos iteraciones es 1.3.

Por otro lado, hemos implementado en **MatrizJacobi.m**, la construcción explícita de la matriz de Jacobi para el problema que nos ocupa, en función de n . Hemos calculado numéricamente el radio espectral de dicha matriz con el comando `eig` de Matlab y hemos aplicado el [Teorema 2.6](#) de la Sección 2.3 que permite calcular el valor del parámetro óptimo ω_{op} en función del radio espectral de la matriz de iteración del método de Jacobi.

Como salida de **MatrizJacobi.m** cuando $n = 4$ obtenemos

$w =$

1.3090

Este valor coincide, dentro de la precisión utilizada en la estimación numérica, con el que habíamos estimado numéricamente, probando con distintos valores de ω y quedándonos con aquel que requiere un menor número de iteraciones para satisfacer un mismo criterio de precisión.

Se ha repetido el mismo experimento cuando se aumenta el número de nodos en los que se obtiene aproximación a la solución de la ecuación de Poisson. Más precisamente, se ha trabajado con $n = 4, 8, 16$ y 32 .

En la siguiente tabla se pueden ver los valores de ω_{op} calculados con la función **MatrizJacobi.m** (segunda columna) y los estimados siguiendo el procedimiento que se ha descrito para $n = 4$ pero tomando ahora $\omega = 0.01, 0.02, \dots, 1.99$ (tercera columna).

n	ω_{op} MatrizJacobi.m	ω_{op} iterativos.m
4	1.3090	1.30
8	1.5716	1.57
16	1.7588	1.76
32	1.8716	1.87

Observamos, en primer lugar, que para cada n ambos valores distan menos de una centésima, que es el espaciado que se ha utilizado para el parámetro ω cuando se ha estimado ω_{op} numéricamente. También se observa que al aumentar n el valor del parámetro óptimo ω_{op} se va acercando a 2.

Para finalizar esta sección, mostramos en la [Figura 6](#) la distribución de los elementos no nulos de la matriz de iteración de Jacobi para $n = 4$ cuando se utiliza el ordena natural (arriba) y el orden blanco-negro (abajo). En la parte inferior de la [Figura 6](#) se observa el carácter 2-cíclico de dicha matriz, una de las propiedades necesarias para poder aplicar el [Teorema 2.6](#) que nos ha permitido calcular ω_{op} en función del radio espectral de la matriz de iteración de Jacobi.

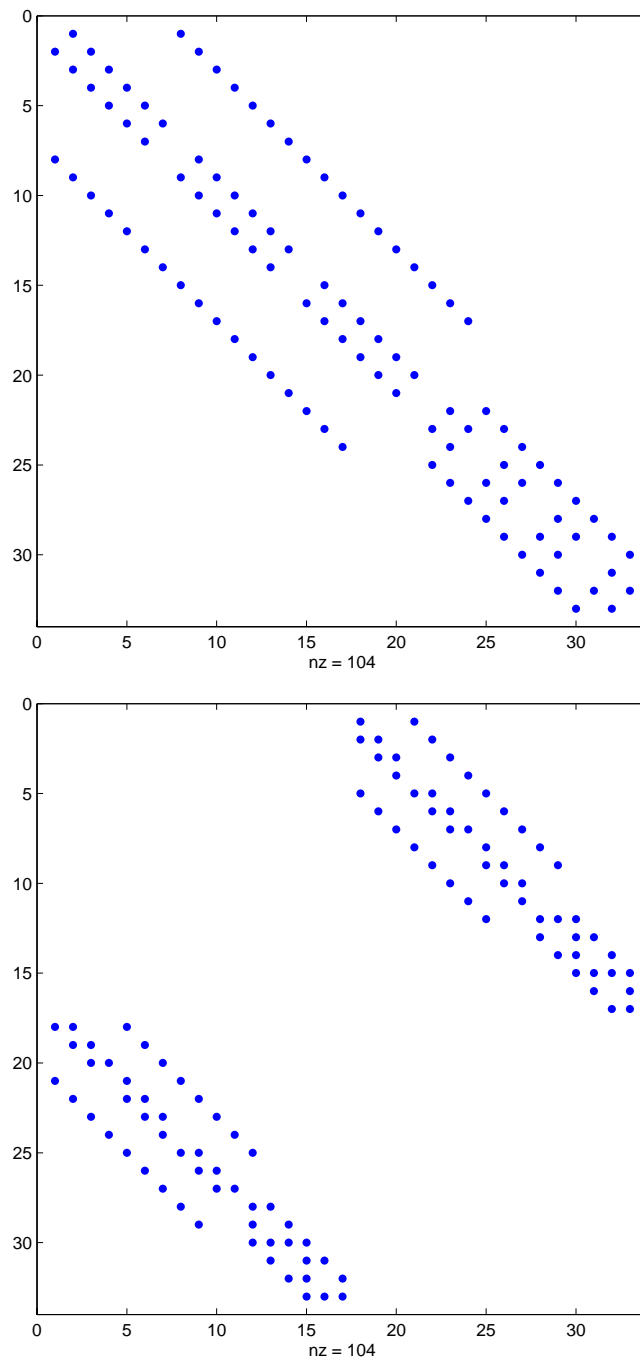


Figura 6: Estructura de elementos no nulos de la matriz de Jacobi para $n = 4$. Arriba: ordenación natural. Abajo: ordenación blanco-negro.

Referencias

- [1] G. BIRKHOFF & S. MACLANE, *Álgebra Moderna*. VICENS-VIVES 1970.
- [2] S. BRIN & L. PAGE, *The Anatomy of a Large-Scale Hypertextual Web Search Engine*, Computer Networks 30 Vol. 1–7 (1998), pp. 107–117.
- [3] R. CRIADO, M. ROMANCE & L. E. SOLÁ, *Teoría de Perron-Frobenius: importancia, poder y centralidad*, La Gaceta de la RSME. Vol.17 No. 3 (2014), pp. 485–514.
- [4] N. DINGLE, W. KNOTTENBELT & D. SPANIAS, *On the (Page) Ranking of Professional Tennis Players*, en *Computer performance engineering*, SPRINGER-VERLAG 2012, pp. 237–247.
- [5] J. R. GILBERT, C. MOLER & R. SCHREIBER, *Sparse matrices in Matlab: Design and implementation*, SIAM J. Matrix Anal. Appl. Vol.13 No. 1 (1992), pp. 333–356.
- [6] J. P. KEENER, *The Perron-Frobenius theorem and the ranking of football teams*, SIAM Review Vol.35 No. 1 (1993), pp. 80–93.
- [7] A. N. LANGVILLE & C. D. MEYER, *Google's PageRank and Beyond: The science of search engine rankings*. PRINCETON UNIVERSITY PRESS 2006.
- [8] J. M. ORTEGA, *Numerical Analysis. A second course*. ACADEMIC PRESS 1972.
- [9] R. S. VARGA, *Matrix Iterative Analysis*. SPRINGER-VERLAG 2000.
- [10] <http://www.muycomputer.com/2012/09/20/cuantas-peticiones-de-busqueda-recibe-google-por-segundo>.

Códigos de MATLAB

En este apéndice vamos a incluir los programas de Matlab que hemos implementado para realizar los experimentos en la Sección 3.

A. Cálculo del PageRank de Google

sensorship.m

```
function [H,a] = sensorship;
% Creamos y guardamos en un fichero la matriz con
% la que vamos a trabajar y el vector indicador de
% las filas nulas de dicha matriz

H=sparse(dlmread('adj_matrix_sensorship.txt'));

ss=sum(H,2);
Inz=ss~=0; %indices de las filas no nulas de H
H(Inz,:)=diag(1./ss(Inz))*H(Inz,:);
a=(~Inz)';
H=H';
% La matriz a la que tenemos que aplicar el metodo
% de la potencia es la que tiene normalizadas las
% columnas, por eso guardamos la traspuesta

save matriz_sensorship_ej H a

end
```

pagerank.m

```
function [niter,b] = pagerank(alfa, tol, N);
% alfa: parámetro alfa de Google
% tol: tolerancia para parar la iteración
% N: numero maximo de iteraciones permitidas

load matriz_censorship_ej
% Cargamos la matriz H y el vector a generados
% por censorship.m
% H: matriz (sparse) que tiene normalizadas las
% columnas que no son idénticamente nulas
% a: vector (sparse) fila indicador de las
% columnas nulas

n=length(H); %dimension de la matriz
y=(1/n*ones(1,n))'; %vector columna,vector inicial
C=(alfa*(a*y)+(1-alfa))*sum(y)./n;
y1=alfa*(H*y)+C;
y1=y1/sum(y1);
niter=1;

while (niter<N & norm(y1-y)>tol)
    y=y1;
    C=(alfa*(a*y)+(1-alfa))*sum(y)./n;
    y1=alfa*(H*y)+C;
    y1=y1/sum(y1);
    niter=niter+1;
end

if niter==N
    disp('Se ha alcanzado el numero maximo de
        iteraciones')
end

v=y1/sum(y1);
[vord,ind]=sort(v,'descend');
b=ind(1:20);

end
```

B. Cálculo del ranking de la Champions League

champions.m

```
function clas = champions(alfa,indh,tol,N);
% alfa: parametro utilizado en partidos no jugados
% indh: indicador para decidir si se usa la func. h
% tol: tolerancia para para la iteracion
% N: numero maximo de iteraciones permitidas

A=sparse(dlmread('Champions_final.txt'));
n=length(A);
Aux=sparse(zeros(n,n));
for i=1:n
    for j=1:n
        if i==j
            A(i,i)=0;
        elseif A(i,j)+A(j,i)==0
            Aux(i,j)=alfa;
            Aux(j,i)=alfa;
        elseif indh==1 % con h
            Aux(i,j)=h((A(i,j)+1)/(A(i,j)+A(j,i)+2));
        else % sin h
            Aux(i,j)=(A(i,j)+1)/(A(i,j)+A(j,i)+2);
        end
    end
end
end
champ=char('Ajax','Anderlecht','Apoel','Arsenal','At
.Madrid','Ath.Bilbao','Barcelona','Basilea','Bate
Borisov','Bayern','Benfica','Br.Dortmund','
Chelsea','CSK Moscu','Galatasaray','Juventus','
Leverkusen','Liverpool','Ludogorets','Malmoe','
Man.City','Maribor','Monaco','Olympiacos','Oporto
','PSG','Real Madrid','Roma','Schalke','Shakhtar'
,'Sp.Lisboa','Zenit');
[niter,lambda,v]=potencia(Aux,tol,N);
[vord,ind]=sort(v,'descend');
clas=champ(ind,:);
end
```

h.m

```
function h = h(x);  
% Funcion h de [6]  
  
h=1/2+1/2*sign(x-1/2)*sqrt(abs(2*x-1));  
end
```

potencia.m

```
function [niter, lambda, v] = potencia(Aux, tol, N)  
% Funcion que implementa el metodo de la potencia  
  
[n, n]=size(Aux);  
x0=(1/n)*ones(n,1);  
  
[M,i]=max(abs(x0));  
y=(1/x0(i)).*x0; %y0  
z=Aux*y; %z1  
  
niter=1;  
  
while (niter<N & abs(x0(i)-z(i))>tol)  
    x0=z;  
    [M,i]=max(abs(z));  
    y=(1/z(i)).*z;  
    z=Aux*y;  
    niter=niter+1;  
end  
  
if niter==N  
    disp('Se ha alcanzado el numero maximo de  
        iteraciones')  
end  
  
lambda=z(i);  
v=y;  
  
end
```

C. Métodos iterativos clásicos

iterativos.m

```
function []=iterativos(alfa, n, omega)
% Estado estacionario de temperaturas de una placa
% en forma de L
% alfa: parametro de las funciones f y g
% n: parametro que define el diametro de la malla
% omega: vector con valores de omega en (0,2)

h = 1/n;
N=2*n; % cuadrado [0,2]*[0,2]
tol=h*h/100; % tolerancia prop. al error de
    discretizacion

% condiciones frontera (L)
x=[0:N]'/n;
y=zeros(N+1,1);
U(:,1) = (x.*y).^(alfa/2); % cf en y=0

x=[0:n]'/n;
y=2*ones(n+1,1);
U(1:n+1,N+1) = (x.*y).^(alfa/2); % cf en y=2

x=[n:N]'/n;
y=ones(n+1,1);
U(n+1:N+1,n+1)=(x.*y).^(alfa/2); % cf en y=1

x=zeros(1,N+1);
y=[0:N]/n;
U(1,:) = (x.*y).^(alfa/2); % cf en x=0

x=2*ones(1,n+1);
y=[0:n]/n;
U(N+1,1:n+1) = (x.*y).^(alfa/2); % cf en x=2

x=ones(1,n+1);
y=[n:N]/n;
U(n+1,n+1:N+1)=(x.*y).^(alfa/2); % cf en x=1
```

```

F=zeros(N+1,N+1);
x=[0:N]/n;
y=[0:N]'/n;
for i=2:N
    for j=2:N
        F(i,j)=(alfa/2)*(alfa/2-1)*(x(i)*y(j)).^...
            (alfa/2-2)*(x(i)^2+y(j)^2);
    end
end
F(n+2:N+1,n+2:N+1)=zeros(N-n);
F=F';

disp('Jacobi')
[niterj, V] = Jacobi(U,F,tol,h);
niterj
disp('Gauss-Seidel')
[niterg, V] = GaussSeidel(U,F,tol,h);
niterg

disp('SOR')
for i=1:length(omega)
[niters, V] = SOR(U,F,tol,h,omega(i));
niters
plot(omega(i),niters,'b. ');
hold on
end
xlabel('omega');
ylabel('número de iteraciones');
end

```


Jacobi.m

```
function [niter,U]=Jacobi(U,F,tol,h);
% Funcion que implementa el metodo de Jacobi
% para el problema de la placa en forma de L

N=length(U);
d=tol+1;
n=floor(N/2);
niter=0;

while max(max(abs(d)))>tol
    x0=U;
    for i=2:n
        for j=2:N-1
            U(i,j)=(x0(i+1,j)+x0(i,j+1)+...
                x0(i-1,j)+ x0(i,j-1)-h*h*F(i,j))/4;
        end
    end

    for i=n+1:N-1
        for j=2:n
            U(i,j)=(x0(i+1,j)+x0(i,j+1)+...
                x0(i-1,j)+x0(i,j-1)-h*h*F(i,j))/4;
        end
    end

    d=x0(2:N-1,2:N-1)-U(2:N-1,2:N-1);

    niter=niter+1;
end

end
```

GaussSeidel.m

```
function [niter,U]=GaussSeidel(U,F,tol,h);
% Funcion que implementa el metodo de Gauss-Seidel
% para el problema de la placa en forma de L

N=length(U);
d=tol+1;
n=floor(N/2);
niter=0;

while max(max(abs(d)))>tol

    x0=U;
    for i=2:n
        for j=2:N-1
            U(i,j)=(x0(i+1,j)+x0(i,j+1)+U(i-1,j)+...
                U(i,j-1)-h*h*F(i,j))/4;
        end
    end

    for i=n+1:N-1
        for j=2:n
            U(i,j)=(x0(i+1,j)+x0(i,j+1)+U(i-1,j)+...
                U(i,j-1)-h*h*F(i,j))/4;
        end
    end

    d=x0(2:N-1,2:N-1)-U(2:N-1,2:N-1);

    niter=niter+1;
end

end
```

SOR.m

```
function [niter,U]=SOR(U,F,tol,h,omega);
% Funcion que implementa el metodo SOR
% para el problema de la placa en forma de L

N=length(U);
d=tol+1;
n=floor(N/2);
niter=0;

while max(max(abs(d)))>tol

    x0=U;
    for i=2:n
        for j=2:N-1
            U(i,j)=omega*(x0(i+1,j)+x0(i,j+1)+...
                U(i-1,j)+U(i,j-1)-h*h*F(i,j))/4 +...
                (1-omega)*x0(i,j);
        end
    end

    for i=n+1:N-1
        for j=2:n
            U(i,j)=omega*(x0(i+1,j)+x0(i,j+1)+...
                U(i-1,j)+U(i,j-1)-h*h*F(i,j))/4 +...
                (1-omega)*x0(i,j);
        end
    end

    d=x0(2:N-1,2:N-1)-U(2:N-1,2:N-1);

    niter=niter+1;
end

end
```

MatrizJacobi.m

```
function w = MatrizJacobi(n)
% Construccion por bloques de la matriz de Jacobi
% con el orden natural
% n: parametro que define el diametro de la malla

h = 1/n;
N=2*n; %cuadrado [0,2]*[0,2]

ong=ones((n-1)*(N-1)-1,1);
ong(N-1:N-1:(n-1)*(N-1)-1)=0;
onp=ones(n*(n-1)-1,1);
onp(n-1:n-1:n*(n-1)-1)=0;

%Bloques grandes
bloqg=diag(-4*ones((n-1)*(N-1),1),0)+...
    diag(ong,1)+diag(ong,-1)+...
    diag(ones((n-1)*(N-1)-(N-1),1),N-1)+...
    diag(ones((n-1)*(N-1)-(N-1),1),-(N-1));
%Bloques pequeños
bloqp=diag(-4*ones(n*(n-1),1),0)+diag(onp,1)+...
    diag(onp,-1)+diag(ones(n*(n-1)-(n-1),1),n-1)+...
    diag(ones(n*(n-1)-(n-1),1),-(n-1));

L=zeros((N-1)*(n-1),n*(n-1));
L((N-1)*(n-2)+1:(N-1)*(n-2)+(n-1),1:n-1)=eye(n-1);

A=[bloqg,L;L',bloqp];
A1=(1/(h^2))*A;
d=diag(A1);
D=diag(d,0);
m=size(A);
J=eye(m)-inv(D)*A1;
figure(1)
spy(J)

%Reordenacion blanco-negro de la matriz de Jacobi
C=zeros(m,m);
J1(1:2:m)=[1:(m+1)/2];
J1(2:2:m)=[(m+1)/2+1:m];
```

```
C(J1, J1)=J;  
figure(2)  
spy(C)  
  
au=eig(J);  
  
ro=max(abs(au));  
w=2/(1+sqrt(1-ro^2));  
  
end
```