

El secreto de Google™ y el Álgebra lineal

Pablo Fernández Gallardo

Departamento de Matemáticas (Universidad Autónoma de Madrid)

Los editoriales de los principales periódicos del mundo se ocupan estos días de la noticia de la salida a Bolsa de Google, no sólo por el volumen de negocio que supone¹, sino por lo que de simbólico tiene el ser la primera salida a Bolsa de una empresa de esta envergadura desde la “exuberancia irracional”² de las llamadas *punto.com* de los años 90.

Pero hay algo más que explica este destacado interés, y tiene que ver con las características propias de la empresa. Ya han pasado varios lustros desde que se produjo una revolución en el campo de la tecnología y de la información —y, quizás (¿o sin duda?), una revolución cultural, sociológica, etc.—, como fue la generalización del acceso, uso y participación en la red de redes, Internet. La aparición del buscador Google ha supuesto una revolución equiparable a la anterior, al convertirse en una herramienta que ha sido capaz de poner orden en todo ese (antes inabarcable) universo de información.

El diseño de un buscador en la red es un problema de *ingeniería matemática*. Nótese el adjetivo. Se necesita primero un buen conocimiento del contexto, que permita traducirlo a modelos, a Matemáticas. Pero tras ese proceso de abstracción, de matematización, y una vez extraídas las conclusiones pertinentes, se requiere también una cuidadosa y eficiente implementación, un detallado diseño de las cuestiones computacionales inherentes al problema.

El objetivo de este artículo es mostrar algunos de los ingredientes matemáticos que están en la base del funcionamiento del algoritmo (*PageRank*) con el que Google ordena los resultados de las búsquedas. Es un sabroso cóctel de Álgebra lineal, Teoría de Grafos y Probabilidad que, sin duda, nos facilita la vida.

1. El buscador Google

El origen del buscador Google es ya bien conocido. Fue diseñado en 1998 por Sergei Brin y Lawrence Page, dos estudiantes de doctorado en Informática de la Universidad de Stanford: Brin se había graduado en Matemáticas, y Page en Informática. Los vemos en las fotografías de la derecha³. Dos jóvenes que hoy, en la frontera de la treintena, se han convertido en multimillonarios. El curioso nombre es una variación sobre el término *googol*, que alguien⁴ inventó para referirse al



¹Se pondrán a la venta acciones por valor de 2700 millones de dólares.

²Parece ser que, para intentar evitar los excesos y escándalos que rodearon la burbuja financiera de las empresas tecnológicas en años pasados, la oferta se articulará en forma de subasta *on line*, de manera que muchos inversores tengan similares oportunidades de adquirir acciones, y con objeto de evitar grandes movimientos especulativos. Pero nunca se sabe...

³Podemos jurar que no tenemos nada que ver con esa suerte de bolitas navideñas que adornan las fotografías. Están extraídas, tal cual, de la página web de Google.

⁴Se dice que un sobrino del matemático Edward Kasner. Kasner se animó y también definió el googolplex, que es 10^{googol} . ¡Cielos!

apabullante número 10^{100} . Uno de esos números que los matemáticos manejamos con comodidad pero que, quizás, sea mayor que el número de partículas del Universo.

Aunque sin llegar a esos extremos, las escalas de la cuestión que nos interesa son también gigantescas. En 1997, cuando Brin y Page empezaban a trabajar en el diseño de Google, había censadas en torno a los 100 millones de páginas web. Altavista, el buscador más popular por entonces, atendía 20 millones de consultas diarias. Hoy, esas cifras se han multiplicado: el propio buscador Google atiende 200 millones de consultas diarias e indexa varios miles de millones de páginas web.

Así que el diseño de un buscador ha de resolver con eficacia ciertas cuestiones computacionales, como la manera en que se almacena toda esa información, cómo se actualiza, cómo se pueden gestionar las peticiones, cómo buscar en las gigantescas bases de datos, etc., problemas en los que, sin duda, las Matemáticas tienen mucho que decir.

Pero sobre ninguna de estas cuestiones, con ser interesantes, reflexionaremos aquí. El asunto que nos interesa se puede formular de manera sencilla: supongamos que, a resultas de una consulta, tras el proceso de búsqueda en las bases de datos, hemos determinado que, digamos, unas cien páginas contienen información que, de una manera u otra, puede resultar relevante para el usuario. Ahora,

¿en qué orden mostramos esos resultados?

El objetivo, como de manera explícita se plantearon⁵ Brin y Page (véase [BrPa]), es que, en un número suficientemente grande de los casos, al menos una de, digamos, las diez primeras páginas que se muestren contenga información útil para el que realiza la consulta.

Pedimos ahora al lector (que en un alto porcentaje será sin duda Google-adicto) que decida, a partir de su experiencia personal, si Google cumple este objetivo. Estamos seguros de que la respuesta general es que sí, y de manera asombrosa. Parece magia⁶... pero son Matemáticas.

Matemáticas que, además, no requieren conocimientos que vayan más allá de cursos básicos de licenciatura, como veremos a lo largo de estas páginas.

Para la tarea que queremos abordar, necesitamos un **criterio** de ordenación. Obsérvese (y aquí ya nos hemos puesto en modo matemático) que si etiquetamos con los símbolos P_1, \dots, P_n cada una de las páginas de la red, todo lo que nos hace falta es asignar a cada P_j un número x_j , que llamaremos su **importancia**. Estos números podrían ser, por ejemplo, números entre 0 y 1.

Supongamos entonces que, tras un censo de los sitios de la red, hemos conseguido construir la lista de páginas web, y que le hemos asignado a cada una de ellas, de la manera que sea, una importancia. Esta lista queda a nuestra disposición para ser utilizada cada vez que contestemos a una determinada consulta: las páginas seleccionadas se mostrarán en el orden que indique dicha lista.

Queda, por supuesto, explicar cómo se construye esa lista. Vamos con ello.

NOTA. Para completar la descripción, y aunque no entraremos en los detalles, deberíamos decir que hay un par de ingredientes que, en cada consulta particular, se combinan en el buscador Google con el criterio general que veremos aquí:

- por un lado, y como resulta razonable, Google no “puntuá” por igual el que un cierto término aparezca en el título de la página, en negrita, en un tipo de letra pequeña, etc.
- Para búsquedas combinadas, tampoco es lo mismo que los términos buscados aparezcan, dentro del documento, “cerca” o “lejos” unos de otros.

⁵No era el único objetivo que se propusieron. También pretendían que el buscador fuera “resistente” a intentos de manipulación, a estrategias que, por ejemplo por intereses comerciales, pretendieran situar ciertas páginas en los primeros puestos de la lista. Curiosamente, hoy en día se ha hecho popular un “deporte”, el *Google bombing*, que consiste precisamente en eso, en conseguir colocar una página en esas primeras posiciones, aunque generalmente con fines más lúdicos. Pruebe por ejemplo el lector a buscar en Google los términos “miserable fracasado”. El resultado le sorprenderá (y quizás regocijará).

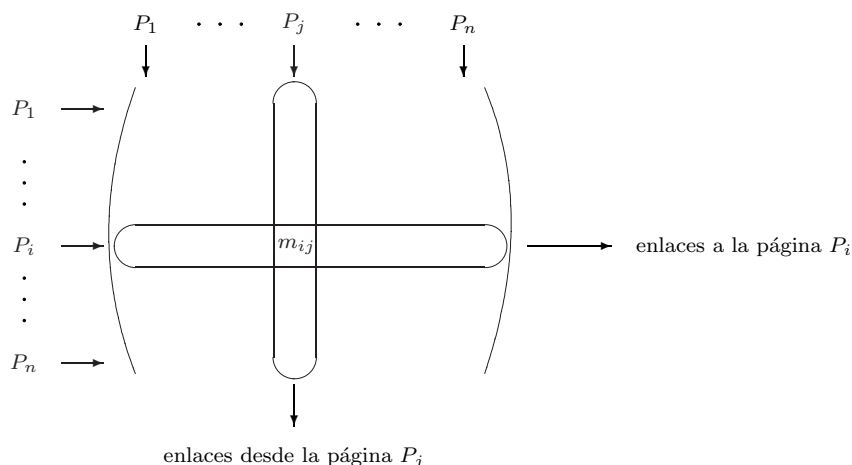
⁶Por no hablar de la capacidad de la que hace gala el buscador para “corregir” los términos de la búsqueda y sugerir, ¡sí!, la palabra que uno en realidad estaba pensando teclear. Lo que le lleva a uno en ocasiones a imaginar intervenciones sobrenaturales... En fin, dejémoslo estar.

2. El modelo

El primer paso consiste en describir, en forma manejable, los elementos relevantes para la cuestión que interesa, la asignación de importancias. Digamos que hemos recogido toda la información sobre la red (sitios, contenidos, enlaces de unas páginas a otras, etc.). De toda esta información, y en este primer paso de la modelización, vamos a rescatar únicamente la referida a cada una de las páginas, a la que asignamos etiquetas P_1, \dots, P_n , y los enlaces entre ellas.

En estos términos, la red puede ser descrita mediante un **grafo** (dirigido) G . Cada página P_j de la red es un vértice del grafo, y hay una arista (dirigida) entre los vértices P_i y P_j si desde la página P_i hay un enlace a la página P_j . Es un grafo gigantesco, abrumador, sobre cuya estructura real reflexionaremos más adelante (véase la sección 7).

Cuando tratamos con grafos, nos gusta siempre recurrir a los dibujos en el papel, en los que los vértices son puntos del plano, mientras que las aristas son flechas que unen esos puntos. Pero, a los efectos que aquí perseguimos, conviene considerar una interpretación alternativa, matricial. Vamos, pues, a formar una matriz \mathbf{M} , de dimensiones $n \times n$, cuyas filas y columnas van etiquetados con los P_1, \dots, P_n , y cuyas entradas son ceros y unos. La entrada m_{ij} de la matriz será un *uno* si es que hay un enlace de la página P_j a la página P_i ; y un *cero* en caso contrario:



La matriz \mathbf{M} es, salvo una trasposición, la matriz de adyacencia (o de vecindades) del grafo. Nótese que no tiene por qué ser simétrica, pues nuestro grafo es dirigido. Obsérvese también que, tal como se señala en el dibujo, la suma de las entradas correspondientes a la columna de P_j es el número de enlaces que salen de la página P_j , mientras que la suma de los registros de una fila coincide con el número de enlaces entrantes.

Vamos a postular que la importancia de una cierta página de la red P_j “tiene que ver” con las páginas desde las que hay enlaces a ella. Esto es muy razonable: si muchas páginas enlazan con P_j , será porque la información que ésta contiene ha sido considerada por muchos participantes de la red como “recomendable”.

Ese “tiene que ver” queda por ahora algo difuso. Un **primer intento**, quizás algo ingenuo, consiste en suponer que la importancia x_j de cada P_j es *proporcional al número de páginas desde las que hay enlaces a P_j* . Observemos que, una vez que disponemos de la matriz \mathbf{M} , calcular cada uno de los x_j sería sencillo en este caso, pues bastaría sumar las entradas correspondientes a la fila de P_j .

Pero este modelo no recoge adecuadamente una situación que nos gustaría tener en cuenta, y es aquella en la que una cierta página es citada desde unas pocas páginas, pero muy relevantes. Digamos, por ejemplo, que desde www.microsoft.com, o desde www.amazon.com, etc. El procedimiento anterior le asignaría baja importancia, algo que no parece razonable.

Así que necesitamos afinar nuestro modelo de manera que asignemos alta importancia

- tanto a páginas muy citadas;
- como a páginas poco citadas, pero desde sitios “importantes”.

El **segundo intento**, ya en esta línea, consiste en decidir que la importancia x_j de cada página P_j es proporcional a la suma de las importancias de las páginas que enlazan con P_j . Nótese que hemos cambiado, con respecto a nuestro primer intento, “número de páginas que enlazan con P_j ” por “suma de las importancias de las páginas que enlazan con P_j ”. Esta ligera variación cambia por completo, como veremos, las características del problema.

Supongamos, por ejemplo, que la página P_1 es citada desde las páginas P_2 , P_{25} y P_{256} , que P_2 sólo se cita desde P_1 y P_{256} , etc., mientras que, digamos, hay enlaces a la última página, P_n , desde P_1, P_2, P_3, P_{25} y P_{n-1} . En nuestra asignación anterior, x_1 debería ser proporcional a 3, x_2 lo sería a 2, etc., mientras que x_n habría de ser proporcional a 5.

Pero ahora nuestra asignación x_1, \dots, x_n debe cumplir que

$$\begin{aligned} x_1 &= K(x_2 + x_{25} + x_{256}), \\ x_2 &= K(x_1 + x_{256}), \\ &\vdots \\ x_n &= K(x_1 + x_2 + x_3 + x_{25} + x_{n-1}), \end{aligned}$$

donde K es una constante de proporcionalidad. Nos encontramos así con un enorme sistema de ecuaciones, cuyas soluciones son las posibles asignaciones x_1, \dots, x_n .

Escribamos el sistema anterior en términos más manejables, matriciales:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = K \begin{pmatrix} \begin{matrix} P_1 & P_2 & & & P_{25} & & & P_{256} & & & P_{n-1} \\ \downarrow & \downarrow & & & \downarrow & & & \downarrow & & & \downarrow \end{matrix} \\ \begin{matrix} 0 & 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 \end{matrix} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Hagamos ahora un alarde simbólico al que tan aficionados somos los matemáticos: llamemos \mathbf{x} al vector de importancias. La matriz de dimensiones $n \times n$ del sistema es, justamente, la \mathbf{M} asociada al grafo. Así que podremos escribir que la asignación de importancias que perseguimos es una solución de

$$\mathbf{M}\mathbf{x} = \lambda\mathbf{x}.$$

Ya hemos escrito la constante de proporcionalidad con el símbolo λ . Y es que, como cualquier estudiante de un curso de Álgebra lineal reconoce al instante, la cuestión se ha transformado en un problema de **autovalores** y **autovectores**: nuestro anhelado vector de importancias \mathbf{x} no es sino un autovector de la matriz \mathbf{M} que, recordemos, recogía la información sobre la estructura de la red (vértices y relaciones de adyacencia).

Salvo por la cuestión, casi sentimental, de comprobar que uno de los ingredientes básicos de los cursos de Álgebra lineal permite describir una cuestión tan “aplicada”, es posible que esto todavía no haya despertado el entusiasmo del lector. Sí, de acuerdo, un autovector, ¿pero cuál? Hay tantos... Y además, ¿cómo podemos calcularlo?, la matriz es inimaginablemente grande: recordemos, varios miles de millones de filas (o columnas).

Paciencia.

Por ahora, parece razonable exigir que las entradas del vector (las importancias de las páginas de la red) sean todas no negativas (o, al menos, del mismo signo). Esto lo escribiremos, y perdónenos el lector el abuso de notación, como $\mathbf{x} \geq 0$.

Pero además, si pretendemos que el método sea útil, necesitaríamos que ese hipotético autovector de entradas no negativas fuera **único** (si hubiera más de uno, ¿con cuál nos quedaríamos?).

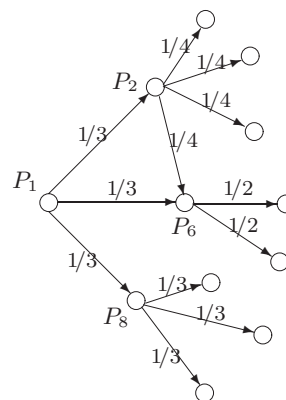
3. El surfista aleatorio

A modo de interludio, vamos a enfocar el problema desde un nuevo punto de vista. Imaginemos que un surfista se dedica a navegar por la red. En un cierto instante de tiempo está en una página, por ejemplo en P_1 . En el instante siguiente, algo aburrido de los contenidos de P_1 , va a saltar a una de las páginas a las que enlaza P_1 (digamos que hay N_1 de ellas). Pero, ¿exactamente a cuál?

Nuestro intrépido navegante es un surfista (además de rubio y bronceado) *aleatorio*. Así que, para decidir a cuál de ellas va, efectúa un sorteo. Y lo hace de la manera más simple posible: utilizando un dado regular (y virtual, suponemos), que tenga tantas caras como enlaces salgan de la página P_1 . En términos más técnicos, la elección de la página de destino sigue una distribución de probabilidad *uniforme* (discreta) en $[1, N_1]$.

Nuestro modelo no es *determinista*, sino *probabilístico*: no sabemos dónde estará un instante de tiempo después, pero sí *con qué probabilidad* estará en cada uno de los posibles destinos. Y además es un modelo *dinámico*, porque el mismo argumento se podría aplicar al segundo movimiento, y al tercero, etc. Nuestro surfista sigue lo que se denomina un **paseo aleatorio** por el grafo.

En el dibujo de la derecha mostramos una posible situación: de P_1 parten tres aristas, a los vértices P_2 , P_6 y P_8 . Así que el navegante sortea entre estas tres páginas, con probabilidad $1/3$ para cada una de ellas. Si, por ejemplo, el resultado del sorteo hubiera sido la página P_2 , entonces volvería a sortear, pero esta vez asignando probabilidad $1/4$ a cada una de los cuatro posibles destinos desde P_2 .

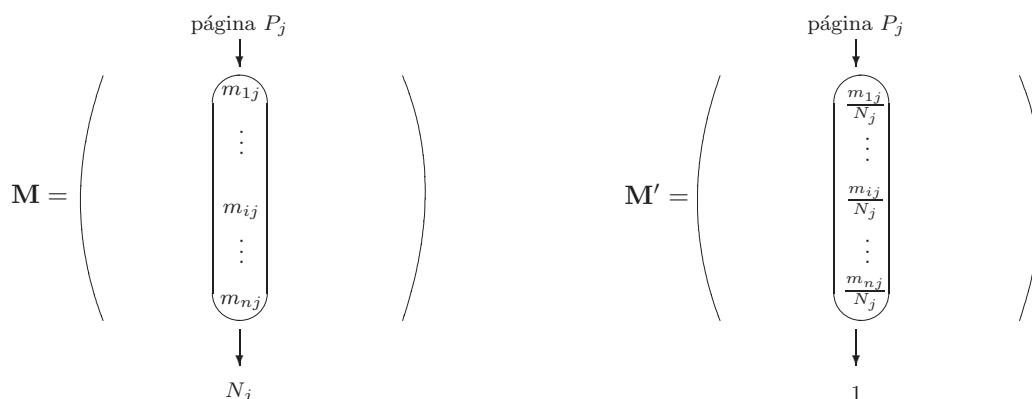


La formulación es muy sugerente, pero no está claro cómo podemos formalizarla: cómo podemos calcular, por ejemplo, la probabilidad de que, partiendo de la página P_1 , el navegante esté en la página P_{17} tras cinco instantes de tiempo. Más aún, todavía ni sospechamos qué puede tener que ver con nuestro problema de asignación de importancias.

Vamos poco a poco. Recordemos que \mathbf{M} es la matriz del grafo de la red, y que sus entradas m_{ij} son ceros o unos. Llamemos N_j al número de enlaces desde la página P_j (esto es, la suma de las entradas de cada columna de \mathbf{M}). Vamos ahora a construir una nueva matriz \mathbf{M}' a partir de \mathbf{M} original sustituyendo cada entrada m_{ij} por

$$m'_{ij} = \frac{m_{ij}}{N_j}$$

Los registros de la nueva matriz \mathbf{M}' son números no negativos (entre 0 y 1) tales que la suma de los registros de *cada columna* vale 1. En la siguiente figura se muestra, para una cierta columna, el resultado de la transformación:



La matriz \mathbf{M}' así construida es lo que se llama una **matriz estocástica** (o de Markov).

Supongamos que el surfista se encuentra, en el primer instante de tiempo, en la página (o vértice) P_k . Para poner en términos probabilísticos esta situación tan de por sí determinística, digamos que está en la página P_k con una probabilidad del 100%. Vamos a representar esta situación inicial con el vector cuyas entradas son todas cero, excepto la de la posición k , que lleva un uno. El navegante sorteando entre los N_k destinos, asignando probabilidad $1/N_k$ a cada uno de ellos.

Pero si multiplicamos la matriz \mathbf{M}' por este vector inicial obtenemos

$$\begin{pmatrix} \cdots & \cdots & m'_{1k} & \cdots \\ \vdots & \ddots & \vdots & \vdots \\ \cdots & \cdots & m'_{kk} & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ \cdots & \cdots & m'_{nk} & \cdots \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} m'_{1k} \\ \vdots \\ m'_{kk} \\ \vdots \\ m'_{nk} \end{pmatrix}.$$

El resultado sigue siendo un vector cuyas entradas (números entre 0 y 1, pues los m'_{jk} son, o bien 0, o bien $1/N_k$) suman 1, porque hay justamente N_k entradas no nulas. Pero más aún: el vector que obtenemos describe exactamente con qué probabilidad estará el surfista, tras una unidad de tiempo, en cada una de las páginas de la red, si es que ha partido de P_k .

Lo que hace especialmente manejable este modelo es que, para conocer con qué probabilidad estará en cada una de las páginas de la red tras *dos* instantes de tiempo, basta repetir la operación. Esto es, multiplicar $(\mathbf{M}')^2$ por el vector inicial. Y el análogo para el transcurrir de tres instantes de tiempo, cuatro, etc.

Siguiendo la nomenclatura habitual, vamos a considerar una serie de **estados**, que en este caso son los vértices del grafo G . La matriz \mathbf{M}' recibe el (apropiado) nombre de **matriz de transición** del sistema: cada entrada m'_{ij} es la probabilidad de pasar del estado (vértice) P_j al estado (vértice) P_i . Y los registros de las sucesivas potencias de la matriz, las probabilidades de pasar de P_i a P_j tras varios instantes de tiempo.

Esto resuelve la primera de las cuestiones: cómo calcular las probabilidades de transición entre vértices y para el transcurso de varias unidades de tiempo. Pero todavía no hemos descubierto la relación con el problema de ordenación de Google. Esto lo dejamos para más adelante (véase la subsección 8.1). Aunque avanzamos, para aquel lector especialmente impaciente y versado en estas cuestiones, que el vector de entradas no negativas que llevamos persiguiendo ya un buen rato resulta ser, justamente, el estado estacionario de la cadena de Markov.

NOTA. Podría ocurrir que alguna página no tuviera enlaces salientes (su columna tuviera sólo ceros). No sería una matriz estocástica. La solución de Google: la sustituimos por una columna con valores $1/n$. Así que si el surfista llega a una página de la que (antes) “no podía salir”, ahora sorteando (uniformemente) entre *todas* las de la red. Más sobre esto, en la sección 7.

4. La clasificación para las eliminatorias por el título

Con tanta reinterpretación y reformulación, nos estamos olvidando de la cuestión que nos interesa: el sistema de ordenación de Google. Disciplinémonos y volvamos a ella. Conviene precisar que las ideas que utiliza Google para ordenar los resultados provienen de unos algoritmos desarrollados por Kendall y Wei en los años 50 (uups, ¡del siglo pasado!, que siempre nos olvidamos de recalcarlo), véanse [Ke] y [We].

Vamos a plantear una cuestión muy similar a la que nos ha ocupado hasta ahora, en un contexto algo más lúdico, que ilustraremos con un ejemplo sencillo que podremos resolver sin ayuda de grandes herramientas computacionales.

La estructura de los campeonatos deportivos, por ejemplo de baloncesto, en Estados Unidos es algo diferente a la que suele seguirse en los europeos. En las ligas europeas, cada equipo juega un

cierto número de partidos (dos, generalmente) contra cada uno de los demás equipos de la competición. Al final de lo que se conoce con el (curioso) nombre de temporada regular, se disputan unas eliminatorias por el título, en las que se intervienen los que han conseguido mejores clasificaciones (por ejemplo, los ocho primeros).

Pero en Estados Unidos, debido a las grandes distancias que separan las distintas ciudades, y para evitar un número excesivo de viajes, la estructura no es tan homogénea.

Lo vamos a ilustrar con la liga profesional de baloncesto, aunque la cuestión que vamos a explicar podría aplicarse también a la Liga universitaria. En la NBA, los 29 equipos están divididos en dos conferencias, cada una de las cuales está a su vez formada por dos divisiones:

- Conferencia Este:
 - División Atlántico;
 - División Central;
- Conferencia Oeste:
 - División Medio Oeste⁷;
 - División Pacífico,



siguiendo (más o menos) criterios geográficos, como se muestra en el mapa de la derecha. Todos los equipos juegan el mismo número de partidos, 82, pero no disputan *el mismo número* contra cada equipo; lo habitual es que se jueguen más partidos contra los de la propia conferencia.

Se plantea así la siguiente cuestión: acabada la temporada regular, ¿qué 16 equipos han de clasificarse para las eliminatorias? El sistema actual computa, simplemente, el número de victorias obtenidas por cada equipo, que determinan las posiciones finales. Pero cabría preguntarse si éste es un sistema “justo”. Al fin y al cabo, podría ocurrir que un equipo hubiera acumulado muchas victorias precisamente por estar encuadrado en una conferencia muy “débil”. ¿Qué debe valer más, el número de victorias, o la “calidad” de las victorias obtenidas? De nuevo, la dicotomía ante la que nos encontrábamos en el diseño de Google.

Digamos que en la competición hay n equipos, E_1, \dots, E_n . Los resultados de la temporada regular los registramos en una matriz \mathbf{A} cuyas entradas a_{ij} son las victorias obtenidas por cada equipo:

$$a_{ij} = \frac{\# \text{victorias de } E_i \text{ sobre } E_j}{\# \text{partidos de } E_i}.$$

Como medida de normalización, dividimos por ejemplo por el número total de partidos que juega cada equipo.

Nuestro objetivo es asignar a cada equipo E_j un número x_j , que por analogía denominaremos importancia, de manera que la lista x_1, \dots, x_n determine la clasificación final.

El primer modelo, en el que x_j es proporcional al número total de victorias conseguido por el equipo E_j , no recoge adecuadamente nuestro interés por valorar la calidad de las victorias.

En el segundo modelo, decidimos que la importancia x_j sea proporcional al número de victorias obtenidas, pero *ponderadas* con la importancia que asignemos a los demás equipos:

$$x_j = K \sum_{k=1}^n a_{jk} x_k.$$

⁷Aquí andan los Memphis Grizzlies y los Utah Jazz, donde juegan Pau Gasol y Raúl López, respectivamente.

Lo que nos conduce, otra vez, a

$$\mathbf{Ax} = \lambda \mathbf{x}.$$

El interés, de nuevo, radica en encontrar un autovector de \mathbf{A} con entradas no negativas (y único, a poder ser). Veamos un ejemplo concreto.

4.1. Para cálculos serios: el ordenador

Tenemos seis equipos, E_1, \dots, E_6 : los tres primeros forman una conferencia, los siguientes la otra. Cada equipo juega 21 partidos en total: 6 contra los de su propia conferencia, 3 contra los de la otra.

La información sobre las victorias conseguidas está contenida en la siguiente tabla:

	E_1	E_2	E_3	E_4	E_5	E_6	
E_1	—	3/21	0/21	0/21	1/21	2/21	→ 6/21
E_2	3/21	—	2/21	2/21	2/21	1/21	→ 10/21
E_3	6/21	4/21	—	2/21	1/21	1/21	→ 14/21
E_4	3/21	1/21	1/21	—	2/21	2/21	→ 9/21
E_5	2/21	1/21	2/21	4/21	—	2/21	→ 11/21
E_6	1/21	2/21	2/21	4/21	4/21	—	→ 13/21

A la derecha de la tabla hemos sumado el número de victorias que ha conseguido cada escuadra. Ese balance de victorias y derrotas sugiere que la ordenación adecuada es

$$E_3 \rightarrow E_6 \rightarrow E_5 \rightarrow E_2 \rightarrow E_4 \rightarrow E_1.$$

Pero observemos que, por ejemplo, el líder de la competición, E_3 , ha acumulado muchas victorias contra E_1 , que es el peor equipo.

¿Qué ocurre con el segundo enfoque? Tenemos que calcular, y en serio. Incluso en un ejemplo tan sencillo como éste, necesitamos la ayuda del ordenador.

Así que recurrimos a un paquete matemático de cálculo, como por ejemplo⁸ **Maple**, para realizar los cálculos. Al usuario experto, los comandos que siguen le resultarán familiares. Al novicio tampoco le costará mucho trabajo entenderlos. Primero, cargamos un paquete necesario:

```
> restart:with(linalg):
```

```
Warning, the protected names norm and trace have been redefined and unprotected
```

Tras esta concisa (y quizás profunda) respuesta del programa, lo ponemos a trabajar en serio. ¡Maple en acción!:

```
> A:=matrix(6,6,[0,3/21,0,0,1/21,2/21,3/21,0,2/21,2/21,2/21,1/21,6/21,4/21,0,2/21,1/21,1/21,3/21,1/21,1/21,0,2/21,2/21,2/21,1/21,2/21,4/21,0,2/21,1/21,2/21,2/21,4/21,4/21,0]);
```

$$A := \begin{bmatrix} 0 & \frac{1}{7} & 0 & 0 & \frac{1}{21} & \frac{2}{21} \\ \frac{1}{7} & 0 & \frac{2}{21} & \frac{2}{21} & \frac{2}{21} & \frac{1}{21} \\ \frac{2}{7} & \frac{4}{21} & 0 & \frac{2}{21} & \frac{1}{21} & \frac{1}{21} \\ \frac{1}{7} & \frac{1}{21} & \frac{1}{21} & 0 & \frac{2}{21} & \frac{2}{21} \\ \frac{2}{21} & \frac{1}{21} & \frac{2}{21} & \frac{4}{21} & 0 & \frac{2}{21} \\ \frac{1}{21} & \frac{2}{21} & \frac{2}{21} & \frac{4}{21} & \frac{4}{21} & 0 \end{bmatrix}$$

⁸Sin ningún afán propagandístico. Matlab, Mathematica, Derive o incluso Excel también valdrían.


```

> autovects:=[evalf(eigenvectors(A))];
autovects := [[0.012, 1., {[−0.062, −0.916, −2.131, 0.873, 0.731, 1.]}],
[0.475, 1., {[0.509, 0.746, 0.928, 0.690, 0.840, 1.]}],
[−0.111 + 0.117 I, 1., {[−0.151 − 0.901 I, 0.123 + 0.451 I, −0.727 + 0.728 I, −0.435 − 0.128 I, 0.192 + 0.377 I, 1.]}],
[−0.126, 1., {[0.008, −0.685, 1.434, −1.071, 0.032, 1.]}],
[−0.139, 1., {[1.785, −3.880, 3.478, −5.392, 4.415, 1.]}],
[−0.111 − 0.117 I, 1., {[−0.151 + 0.901 I, 0.123 − 0.451 I, −0.727 − 0.728 I, −0.435 + 0.128 I, 0.192 − 0.377 I, 1.}]]

```

Una vez acostumbrados a la manera en que **Maple** muestra los resultados (en cada línea, primero el autovalor, luego la multiplicidad, y luego los autovectores asociados), comprobamos que la matriz **A** tiene seis autovalores distintos, dos complejos (conjugados) y cuatro reales. Si calculamos sus módulos,

```

> autovals:=[evalf(eigenvalues(A))]: modulos:=[seq(abs(autovals[j]),j=1..6)];
modulos := [0.012, 0.475, 0.161, 0.126, 0.139, 0.161]

```

observamos que el mayor autovalor (en módulo) es $\lambda = 0.475$, cuyo autovector asociado,

$$\mathbf{x} = (0.509, 0.746, 0.928, 0.690, 0.840, 1),$$

es el **único** (véase arriba) cuyas entradas son todas números reales y no negativos.

Ya tenemos la respuesta que buscábamos: el orden que sugiere este cálculo es

$$E_6 \rightarrow E_3 \rightarrow E_5 \rightarrow E_2 \rightarrow E_4 \rightarrow E_1,$$

que difiere del anterior en los dos primeros (ahora E_6 es el mejor equipo).

Recapitemos. En esta matriz particular (que bien pudiera servir de versión a pequeña escala de la de Internet), cuyas entradas son todas no negativas, se ha dado la mejor de las situaciones posibles: hay un único autovector cuyas entradas son todas no negativas. Este autovector, que además está asociado al autovalor de módulo máximo, nos sirve como solución al problema de ordenación que nos habíamos planteado.

¿Ha sido casualidad? ¿Ha sido simplemente un ardid de quien esto escribe, una artera elección de matriz que busca convencer a los incautos de que las cosas funcionan realmente?

El lector, que seguramente de incauto no tiene nada, y sí mucho de curioso, sentirá la necesidad de dar contundente respuesta a esta cuestión. Y sabrá, sin duda, que ha llegado el momento de dar entrada a las Matemáticas.

5. Las Matemáticas entran en escena

Es la hora de las Matemáticas, esa Ciencia que se ocupa de realidades abstractas, virtuales, para, a través de ellas, entender realidades concretas. Pongámonos, pues, matemáticos, y destilemos la esencia común a todas las cuestiones de las que nos hemos ocupado hasta ahora.

Si lo hacemos, descubriremos que la única propiedad que comparten todas las matrices que nos conciernen (sean estocásticas o no) es que sus entradas son **no negativos**. No parece mucha información. Las matrices no tienen por qué ser simétricas, ni definidas positivas, ni...

Sin embargo, como mostraría Perron⁹ a principios del siglo XX, es suficiente para empezar a extraer buenos réditos:

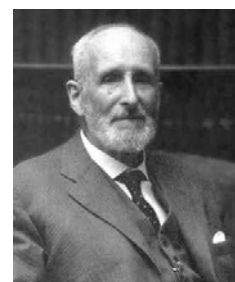


Figura 1: Oskar Perron

⁹El alemán Oskar Perron (1880-1975), todo un ejemplo de longevidad matemática, trabajó en campos tan diversos como el Análisis, las Ecuaciones Diferenciales, el Álgebra, la Geometría, la Teoría de Números, etc., en los que publicó algunos textos que se convertirían en clásicos.

Teorema 1 (Perron, 1907) Sea \mathbf{A} una matriz (cuadrada) con entradas positivas, $\mathbf{A} > 0$. Entonces,

- (a) existe un autovalor (simple) $\lambda > 0$ tal que $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, donde el autovector es $\mathbf{v} > 0$;
- (b) este autovalor es mayor, en módulo, que todos los demás autovalores;
- (c) cualquier otro autovector positivo de \mathbf{A} es un múltiplo de \mathbf{v} .

Paremos un instante y reflexionemos: *ahora* tenemos un *teorema*. Maravillémonos: si usted logra comprobar que las hipótesis se satisfacen (y aquí no admitiremos medias tintas, han de satisfacerse, sin excusas, sin excepciones), entonces la conclusión es inapelable. No albergue dudas, sea cual sea la matriz, mientras que tenga entradas no negativas, existirá un autovalor simple tal que... y un autovector con... ¡Qué cosas!



Figura 2: Georg Frobenius

Tras este raptó casi místico, que el lector sabrá disculpar, volvamos a la realidad y analicemos el resultado de Perron. Apunta en la dirección que buscamos, pero no es suficiente, porque las matrices que manejamos pueden tener ceros en ciertas posiciones. Así que necesitamos algo más. El siguiente acto de esta obra lo escribe Frobenius¹⁰, unos años después, cuando se ocupa del caso más general de las matrices no negativas. Frobenius observa que si únicamente tenemos que $\mathbf{A} \geq 0$ entonces, aunque sigue habiendo un autovalor $\lambda > 0$ *dominante* (de valor absoluto máximo) asociado a un autovector $\mathbf{v} \geq 0$, puede haber otros autovalores del mismo “tamaño”. Ahí va su resultado:

Teorema 2 (Frobenius, 1908-1912) Sea \mathbf{A} una matriz (cuadrada) con entradas no negativas, $\mathbf{A} \geq 0$. Si \mathbf{A} es *irreducible*, entonces

- (a) existe un autovalor (simple) $\lambda > 0$ tal que $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, donde el autovector es $\mathbf{v} > 0$. Además, $\lambda \geq |\mu|$, para cualquier otro autovalor μ de \mathbf{A} .
- (b) Cualquier autovector ≥ 0 es un múltiplo de \mathbf{v} .
- (c) Si hay k autovalores de módulo máximo, entonces son las soluciones de $x^k - \lambda^k = 0$.

NOTA. ¿Qué quiere decir que una matriz \mathbf{A} de dimensiones $n \times n$ sea *irreducible*? Hay varias maneras de entenderlo.

1. No existe ninguna permutación (de filas y columnas) que transforma \mathbf{A} en una matriz del tipo

$$\left(\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{0} & \mathbf{A}_{22} \end{array} \right),$$

donde \mathbf{A}_{11} y \mathbf{A}_{22} son matrices cuadradas.

2. La matriz $(\mathbf{I} + \mathbf{A})^{n-1}$, donde \mathbf{I} es la identidad $n \times n$, tiene todas sus entradas positivas.
3. Si \mathbf{A} es la matriz de adyacencia de un grafo, entonces el grafo está *fuertemente* conectado (véase la sección 7).

Observemos primero que el teorema de Frobenius generaliza realmente el de Perron, pues si $\mathbf{A} > 0$, entonces es $\mathbf{A} \geq 0$ e irreducible.

Segundo, si la matriz \mathbf{A} es irreducible (y sobre si éste es el caso o no de la matriz de la red reflexionaremos más adelante), entonces la cuestión queda completamente resuelta: existe un único autovector con entradas no negativas, que además está asociado al autovalor positivo de módulo máximo (en un momento entenderemos las ventajas que supone esto último).

¹⁰Ferdinand Georg Frobenius (1849-1917) es uno de los más destacados integrantes de la Escuela de Berlín, que lideró el mundo de las matemáticas durante el final del siglo XIX y principios del XX, junto con personajes de la talla de Kronecker, Kummer o Weierstrass (quien fue su director de tesis). Estricta escuela prusiana: rigor, tradición, un matemático “puro”, sin concesiones a la Matemática aplicada. El caprichoso curso de la Historia haría, sin embargo, que muchas de sus ideas sobre representaciones de grupos finitos acabaran sirviendo como base de la Mecánica Cuántica. Es sobre todo conocido por sus aportaciones a la Teoría de Grupos, y sus trabajos sobre las matrices no negativas pertenecen a la última etapa de su vida.

Estos resultados, a los que a partir de ahora nos referiremos conjuntamente como teorema de Perron-Frobenius, tienen una amplia utilidad en diversos contextos (véase la sección 8). Tanto es así que hay quien habla de la “teoría de Perron-Frobenius”, del que este teorema es el resultado fundamental.

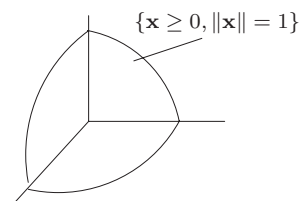
Su demostración es algo complicada, y aquí nos limitaremos a pergeñar un argumento que contiene los ingredientes fundamentales de la prueba.

Una “demostración” del teorema de Perron-Frobenius (ilustrada en el caso 3×3)

Como la matriz \mathbf{A} tiene entradas no negativas, si partimos de un vector $\mathbf{x} \geq 0$, entonces el vector \mathbf{Ax} sigue teniendo entradas no negativas. En términos geométricos, la matriz \mathbf{A} aplica el octante positivo en sí mismo.

Consideramos ahora la aplicación α dada por

$$\alpha(\mathbf{x}) = \frac{\mathbf{Ax}}{\|\mathbf{Ax}\|}.$$



Obsérvese que $\alpha(\mathbf{x})$ es siempre un vector de longitud 1. Por la observación anterior, α envía el conjunto $\{\mathbf{x} \in \mathbb{R}^3 : \mathbf{x} \geq 0, \|\mathbf{x}\| = 1\}$, esto es, la porción de la esfera unidad que dibujamos a la derecha, en sí mismo.

Ahora, aplicando el teorema del punto fijo de Brouwer, existe un cierto $\tilde{\mathbf{x}}$ tal que $\alpha(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}$. Así que

$$\alpha(\tilde{\mathbf{x}}) = \frac{\mathbf{A}\tilde{\mathbf{x}}}{\|\mathbf{A}\tilde{\mathbf{x}}\|} = \tilde{\mathbf{x}} \implies \mathbf{A}\tilde{\mathbf{x}} = \|\mathbf{A}\tilde{\mathbf{x}}\| \tilde{\mathbf{x}}.$$

En resumen, $\tilde{\mathbf{x}}$ es un autovector de \mathbf{A} con entradas no negativas asociado a un autovalor > 0 .

Quedan algunos detalles, como el de demostrar que ese autovector es esencialmente único, y los demás apartados del teorema. Remitimos, al lector interesado en estas cuestiones, a las referencias [BaRa], [BePl], [Ma] y [Mi].

6. ¿Y la cuestión computacional?

El lector crítico estará planteando ya una objeción (sería): el teorema de Perron-Frobenius garantiza la existencia del autovector que es la respuesta a nuestro problema de ordenación, pero *nada nos dice* sobre cómo calcularlo. Obsérvese que la demostración que hemos exhibido no es constructiva. Así que cabe todavía la posibilidad de que estos resultados no constituyan una respuesta satisfactoria. Recordemos que la matriz que maneja Google es gigantesca. El cálculo de nuestro anhelado autovector, ¿es realmente una cuestión abordable desde el punto de vista computacional?

Vamos a suponer que estamos en el mejor escenario posible, en las condiciones que aseguran la existencia de un autovalor λ_1 positivo *estrictamente mayor* (en módulo) que todos los demás autovalores. Llamemos \mathbf{v}_1 al autovector (positivo) asociado.

NOTA. Una matriz $\mathbf{A} \geq 0$ es **primitiva** si tiene un único autovalor dominante (mayor, en módulo, que todos los demás). Esto ocurre cuando, por ejemplo, si, para cierto entero positivo k , la matriz \mathbf{A}^k tiene todas sus entradas positivas.

Podemos, por supuesto, calcular *todos* los autovectores y quedarnos con el que nos interesa. Pero incluso aunque utilizáramos métodos eficaces para calcular autovalores (no mediante determinantes, desde luego), la tarea sería excesiva.

Pero de nuevo la propia estructura del problema sale en nuestra ayuda para facilitarnos el cálculo. Una feliz coincidencia. Todo arranca de la observación, hasta ahora inocente, de que el autovector que buscamos está asociado al autovalor de módulo máximo.

Supongamos, por simplificar, que \mathbf{A} es diagonalizable. Los autovectores $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$, numerados de manera que los autovalores correspondientes vayan en orden decreciente de tamaños,

$$\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|,$$

son una base de \mathbb{R}^n . Partimos, por ejemplo, de un $\mathbf{v}_0 \geq 0$, que escribimos como

$$\mathbf{v}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_n \mathbf{v}_n,$$

donde los números c_1, \dots, c_n son las coordenadas de \mathbf{v}_0 en la base considerada. Para el argumento que sigue, *no va a ser necesario* calcularlos explícitamente, y de hecho no lo haremos. Nos basta con saber que tales números existen. Ahora multiplicamos el vector \mathbf{v}_0 por la matriz \mathbf{A} , para obtener

$$\mathbf{A}\mathbf{v}_0 = c_1 \lambda_1 \mathbf{v}_1 + c_2 \lambda_2 \mathbf{v}_2 + \dots + c_n \lambda_n \mathbf{v}_n,$$

puesto que los vectores $\mathbf{v}_1, \dots, \mathbf{v}_n$ son autovectores de \mathbf{A} . Sólo *calculamos* el producto de la izquierda; de lo de la derecha nos basta, para nuestros argumentos, con *saber* que ocurre. Perdona el lector la insistencia.

Cada vez más animados, repetimos la operación:

$$\mathbf{A}^2 \mathbf{v}_0 = c_1 \lambda_1^2 \mathbf{v}_1 + c_2 \lambda_2^2 \mathbf{v}_2 + \dots + c_n \lambda_n^2 \mathbf{v}_n.$$

Y lo hacemos muchas más veces, digamos k de ellas:

$$\mathbf{A}^k \mathbf{v}_0 = c_1 \lambda_1^k \mathbf{v}_1 + c_2 \lambda_2^k \mathbf{v}_2 + \dots + c_n \lambda_n^k \mathbf{v}_n$$

Supongamos que $c_1 \neq 0$. Entonces,

$$\frac{1}{\lambda_1^k} \mathbf{A}^k \mathbf{v}_0 = c_1 \mathbf{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k \mathbf{v}_2 + \dots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^k \mathbf{v}_n.$$

Pero como $|\lambda_j/\lambda_1| < 1$ para cada $j = 2, \dots, n$ (recordemos que λ_1 era el autovalor dominante),

$$\frac{1}{\lambda_1^k} \mathbf{A}^k \mathbf{v}_0 \xrightarrow{k \rightarrow \infty} c_1 \mathbf{v}_1.$$

De manera que, al multiplicar reiteradamente el vector inicial por la matriz \mathbf{A} , vamos determinando, cada vez con mayor precisión, la *dirección* que nos interesa, la que determina el vector \mathbf{v}_1 . Este método numérico es conocido como el **método de las potencias**, y su velocidad de convergencia depende del cociente entre el primer y el segundo autovalor (véase en [HaKa] una estimación para el caso de la matriz de Google).

Con esta observación, nuestro problema queda resuelto, al menos si estamos en las mejores condiciones posibles (una matriz de entradas no negativas e irreducible). La respuesta que buscábamos existe, es única, y además disponemos de un método eficaz¹¹ para calcularla. Pero...

7. ¿Estamos realmente en una situación ideal?

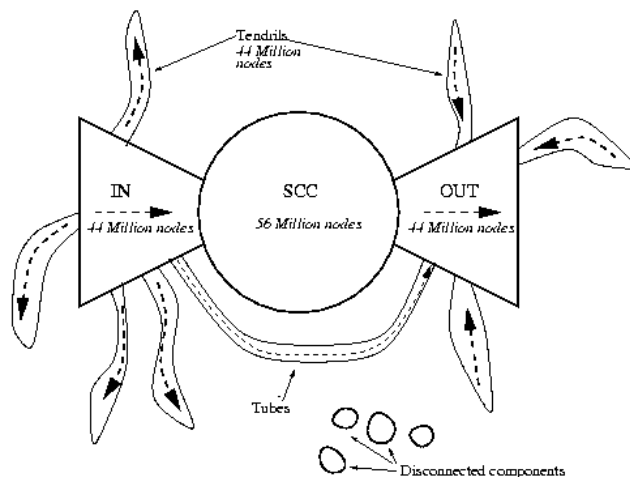
Para que todo funcione, necesitamos que la matriz \mathbf{M} (o quizás \mathbf{M}') asociada al grafo G de la red sea irreducible. En otra palabras, que G sea fuertemente conexo.

NOTA. Consideremos un grafo dirigido G (un conjunto de vértices y un conjunto de aristas a las que asociamos un sentido). Si dados cualesquiera dos vértices u y v del mismo podemos encontrar una sucesión de aristas que llevan de uno a otro, entonces el grafo es *fuertemente conexo*.

Si la misma conclusión se obtiene cuando “borramos” los sentidos de las aristas, entonces se dice que G es *débilmente conexo*. Por supuesto, un grafo dirigido fuertemente conexo es débilmente conexo, pero no (necesariamente) al revés.

¹¹Según información extraída de la propia página de Google, es un cálculo que no lleva más que unas horas de trabajo computacional.

Como el lector podrá sospechar, no es éste el caso. Un estudio realizado en 1999 (véase [Br]) concluía que, de las 203 millones de páginas censadas, el 90% de ellas estaba en una gigantesca componente (débilmente) conexas, que a su vez tenía una estructura interna compleja (más, quizás, de lo que uno podría sospechar), como se muestra en el siguiente dibujo, extraído del citado [Br]:



Es una estructura peculiar, que nos recuerda a un organismo biológico, una suerte de gigantesca ameba. La pieza central (SCC, *strongly connected component*) es, como el nombre indica, una componente fuertemente conexas. Junto a ella aparecen otras dos piezas¹²: la IN está formada por las páginas que tienen enlaces hacia las de SCC, y la OUT está constituida por las páginas a las que apuntan los de SCC. Además aparecen una especie de dendritas, que contienen páginas desde las que no se puede llegar a los nodos de SCC, ni tampoco son accesibles desde ellos (que, en ocasiones, y para aumentar la complejidad, pueden llegar a formar tubos).

Obsérvese, de todas formas, que la organización de la red es dinámica, y cambia con el tiempo. Y no está claro si se habrá mantenido, en líneas generales, la estructura¹³ que aquí exhibimos.

¿Qué podemos hacer (o más bien, ¿qué hace Google?) ante esto? Un truco muy habitual: se trata de conseguir (de una manera razonable) estar en la mejor situación posible. Por ejemplo, *añadiendo* toda una serie de probabilidades de transición (de salida) a todos los vértices. Esto es, considerando la matriz

$$\mathbf{M}'' = c\mathbf{M}' + (1 - c) \begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix} (1, \dots, 1),$$

¹²Los autores del estudio aventuran algunas explicaciones sobre estas dos partes. Así, el conjunto IN podría estar formado por páginas de nueva creación, tanto, que no habría dado tiempo todavía a que los sitios del núcleo central hubieran establecido vínculos hacia ellas. Las de la componente OUT podrían ser páginas de webs corporativas, que sólo contienen vínculos internos.

¹³Hay muchas otras preguntas interesantes que podemos hacernos sobre la estructura del grafo de la red. Por ejemplo, cuál es el número medio de enlaces que tiene una página, cuál es la distancia media entre dos páginas, o cuál es la probabilidad $\mathbf{P}(k)$ de que una página elegida al azar tenga exactamente k enlaces (por ejemplo, entrantes). Si el grafo de la red fuera *aleatorio* (en el sentido preciso que establecieron Erdős y Rényi), entonces esperaríamos que fuera una distribución binomial (o, en el límite, una distribución de Poisson). Y que, en media, la mayor parte de las páginas tuviera un número similar de enlaces. Sin embargo, estudios empíricos (véase, por ejemplo, [Ba1]) sugieren que el decaimiento de la distribución de probabilidad no es exponencial, sino como el de una potencia (*power law*), $k^{-\beta}$, donde β es del orden de 2 y pico. Lo que implicaría, por ejemplo, que hubiera una gran mayoría de páginas con muy pocos enlaces, al tiempo que una pequeña minoría (pero muy significativa) tendría un gran número de enlaces. Más aún, si consideramos la red como un sistema en evolución, al que sucesivamente se van añadiendo nuevas páginas, la conclusión es que esta tendencia se reforzaría. En términos coloquiales, “los ricos se vuelven cada vez más ricos”. Conclusión habitual en sistemas competitivos (y, no nos engañemos, en la vida real). Recomendamos aquí la lectura de la referencia [Ba2].

donde p_1, \dots, p_n es una distribución de probabilidad ($p_j \geq 0, \sum_j p_j = 1$) y c es un cierto parámetro entre 0 y 1 (en los cálculos de Google, del orden de 0.85).

Por ejemplo, podríamos tomar una distribución uniforme, $p_j = 1/n$ para cada $j = 1, \dots, n$ (con lo que la matriz ya tendría todas sus entradas positivas). Pero podríamos elegir otras, y este nuevo grado de libertad permitiría hacer búsquedas “personalizadas”.

En términos del surfista aleatorio, estamos añadiendo la posibilidad de que (con probabilidad $1 - c$) se “aburra” de seguir los enlaces y opte por saltar a otras páginas (con arreglo a cierta distribución de probabilidad).

8. Matrices no negativas en otros contextos

Las matrices no negativas, y la teoría de Perron-Frobenius asociada, tienen aplicaciones en muchos otros campos, además del problema de ordenación que nos ha ocupado hasta aquí. Para situarlas en la posición que merecen, vamos a revisar brevemente algunos modelos en los que resultan imprescindibles (el lector podrá encontrar otras aplicaciones en [Ma]).

Esta ubicuidad se explica por las siguientes dos observaciones:

- En las situaciones “reales”, las interacciones que se miden son, muy frecuentemente, positivas, o al menos no negativas. Y los objetos con que codificamos estas medidas son matrices cuyas entradas son no negativas.
- Por otro lado, muchos modelos son procesos iterativos simples: de un estado inicial \mathbf{x}_0 pasamos a uno general dado por $\mathbf{x}_k = \mathbf{A}^k \mathbf{x}_0$. La convergencia del método depende del tamaño de los autovalores de \mathbf{A} , o más bien de la razón entre los tamaños de los autovalores (en particular, del más grande a los demás). Y aquí, por supuesto, el teorema de Perron-Frobenius tiene mucho que decir, si es que la matriz es no negativa.

8.1. Modelos de evolución probabilística

Recordemos que una matriz \mathbf{A} es estocástica (o de Markov) si $\mathbf{A} \geq 0$ y, para cada columna, la suma de las entradas es 1. Para este tipo de matrices, $\lambda = 1$ es siempre un autovalor.

NOTA. Razón: las columnas de $\mathbf{A} - \mathbf{I}$ suman 0. Luego al sumar todas las filas de $\mathbf{A} - \mathbf{I}$ obtenemos el vector cero, así que las filas son linealmente dependientes. Esto quiere decir que $\mathbf{A} - \mathbf{I}$ es singular. Por tanto, $\det(\mathbf{A} - \mathbf{I}) = 0$. Lo que supone, finalmente, que $\lambda = 1$ sea un autovalor.
Además, no puede haber autovalores de módulo > 1 , pues \mathbf{A} transforma vectores cuyas entradas sumen 1 (vectores de probabilidad) en vectores del mismo tipo.

Las cadenas de Markov son un modelo probabilístico bien conocido, aplicable a situaciones muy diversas: la que hemos visto de Google es un buen ejemplo, pero también se utiliza como modelo de migraciones, de propagación de epidemias, etc.

Consideremos otra situación, que tal vez no le resulte tan familiar al lector. Supongamos que tenemos algo de dinero y que hemos decidido invertirlo para obtener un cierto rendimiento. Quizás debido a alguna experiencia reciente algo desagradable, optamos por no invertirlo en Bolsa. En este plan decididamente conservador, buscamos una inversión “segura”: le prestaremos dinero al Estado. Podría ser al Estado español, quizá a la Reserva Federal estadounidense, pero también, ¿por qué no?, al Corte Inglés, a Repsol, etc. Hay multitud de formas de hacerlo: Bonos del Tesoro, Obligaciones de empresas, emisiones de deuda, etc., aunque alguna de ellas no está al alcance del inversor particular. Es posible que obtengamos un rendimiento medio menor que en la Bolsa, pero a cambio tenemos la seguridad de que no perderemos, en ningún caso, el valor de nuestra inversión. Y aunque no tengamos tanto pánico a la Bolsa, hemos oído por ahí que la regla básica del inversor es la de “diversificar”, y ésta parece una buena opción.

Pero no, las cosas no son tan de color de rosa. Háblenles de la seguridad de este tipo de inversiones a aquéllos que en su momento prestaron dinero al Estado argentino, al ruso o, en casos más recientes, a los que aventuraron su dinero en lo que parecían sólidas¹⁴ compañías como Enron, Parmalat o Swiss Air (si nos fallan hasta los suizos, ¡a dónde iremos a parar!).

No existen inversiones “seguras”, aunque algunas lo son más que otras. Así que el mercado se ocupa de determinar, con la máxima precisión posible, la **solvencia** de cada una de las empresas (entendidas con la mayor generalidad posible, también valen Estados) en las que uno puede invertir. Si una empresa es muy segura, en el sentido de que con altísima probabilidad devolverá el dinero (junto con los intereses) que los inversores le han prestado, entonces podrá permitirse ofrecer una baja rentabilidad. Pero si otra está catalogada como “dudosa” y pretende captar inversiones, entonces deberá necesariamente ofrecer intereses más altos que la hagan atractiva. Queda luego al arbitrio del poseedor del dinero elegir una u otra, en función de sus características (y de la vena especuladora que tenga, quizá oculta).

Éste es todo un mundo, el llamado *riesgo de crédito*, en el que los modelos matemáticos, algunos muy sofisticados, desempeñan un papel fundamental. Describamos aquí uno de ellos, el modelo (de evolución probabilística) de la solvencia de las empresas.

Tenemos n estados de solvencia, digamos S_1, \dots, S_n . En la jerga que utilizan las empresas que se dedican a estos menesteres clasificatorios (las llamadas agencias de calificación, Standard & Poor’s, Moody’s, etc.), estos estados son AAA, BBB+, CC, etc. (con un cierto soniquete de calificaciones escolares anglosajonas).

Tras un estudio detallado, se determina que la probabilidad de pasar del estado S_i al estado S_j es el número a_{ij} . Toda esta información se recoge en una matriz \mathbf{A} , cuyas entradas, una vez más, son no negativas. Además, es una matriz estocástica (por filas). Suele incluirse un estado especial, D (de *default*), que en la jerga de las cadenas de Markov se dice que es *absorbente* (todos ceros, menos el elemento de la diagonal). La siguiente matriz \mathbf{A} es un ejemplo¹⁵:

	AAA	AA	A	BBB	BB	B	CCC	D
AAA	90,58	8,36	0,84	0,09	0,13	0,00	0,00	0,00
AA	0,84	90,04	8,11	0,71	0,12	0,15	0,03	0,00
A	0,14	2,69	89,94	5,92	0,85	0,32	0,04	0,10
BBB	0,03	0,32	5,94	87,41	4,92	1,04	0,13	0,21
BB	0,02	0,12	0,62	7,49	81,93	7,88	0,92	1,02
B	0,00	0,09	0,29	0,65	6,96	81,60	4,43	5,98
CCC	0,17	0,00	0,36	1,09	2,35	10,37	64,06	21,60
D	0,00	0,00	0,00	0,00	0,00	0,00	0,00	100,0

Que nos dice, por ejemplo, que el 90.58% de las empresas con la máxima calificación siguen en la misma categoría al año siguiente.

Miramos la situación hoy, la proporción de empresas que están en cada uno de los estados de solvencia, para formar el vector inicial $\mathbf{z}^{(0)} = (z_1^{(0)}, \dots, z_n^{(0)})$. Según este modelo, el vector de proporciones vendrá dado, tras k años, por $\mathbf{z}^{(k)} = \mathbf{z}^{(0)} \mathbf{A}^k$.

Puede interesarnos estudiar, por ejemplo si queremos hacer o valorar inversiones a muy largo plazo (muchos de los bonos y obligaciones tienen horizontes temporales de 20 años o más), el *comportamiento asintótico* del sistema. Esto es, calcular el vector de proporciones cuando $k \rightarrow \infty$, lo que llamaríamos $\mathbf{z}^{(\infty)}$.

Lo sorprendente del caso es que, si $\lambda = 1$ es el único autovalor dominante (es decir, la matriz es primitiva), entonces, *sean cuales sean las proporciones iniciales*, el estado estacionario $\mathbf{z}^{(\infty)}$ es el autovector correspondiente a $\lambda = 1$. Autovector, por cierto, que es el que la teoría de Perron-Frobenius nos ofrecía.

¹⁴Sólidas, pero con productos líquidos (Enron y Parmalat) ¡o gaseosos! (Swiss Air).

¹⁵Es la matriz de Standard & Poor’s correspondiente a las probabilidades de transición, en tantos por ciento, entre estados de *rating* con los datos del periodo 1981-96.

Por aquello de que estas cosas más vale verlas antes de creerlas, recurrimos de nuevo a Maple para los cálculos, con la matriz estocástica siguiente:

```
> A:=matrix(4,4,[.4,.1,.1,.2,.4,.6,.3,.1,.15,.2,.4,0,.05,.1,.2,.7]);
```

$$A := \begin{bmatrix} 0.4 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.6 & 0.3 & 0.1 \\ 0.15 & 0.2 & 0.4 & 0 \\ 0.05 & 0.1 & 0.2 & 0.7 \end{bmatrix}$$

Ahora, amablemente, solicitamos a Maple que nos calcule los autovalores y los autovectores:

```
> eigenvectors(A);
```

```
[1.000000002, 1, {[-0.3782256600, -0.7865374545, -0.3567355661, -0.5630404726]}],
[0.4999999982, 1, {[-0.2962451280, 0.4443676920, 0.4443676932, -0.5924902550]}],
[0.2292893218, 1, {[.2060703913, -0.7035683205, .6432117000, -1.1457137708]}],
[0.3707106784, 1, {[-1.587596362, 0.9299924275, 1.780204107, -1.122600167]}]
```

A estas alturas, el resultado ya no nos sorprende: el autovector asociado al autovalor $\lambda = 1$ es el único que tiene todas las entradas con el mismo signo. Maple nos ofrece una versión del autovector cuyas entradas son todas negativas, pero esto no nos importa mucho, porque basta cambiarlas de signo (y seguiremos teniendo un autovector).

Para la comparación que haremos en un momento, conviene reescribir el autovector de manera que todas sus entradas sean positivas y que además sumen 1. El resultado es el siguiente:

$$\mathbf{v} = (0.1814432986, 0.3773195881, 0.1711340205, 0.2701030929).$$

Vamos ahora con la evolución probabilística. Partimos de un cierto vector \mathbf{v}_0 (que incluso podríamos sortear) y dejamos que la matriz haga el resto:

```
> v0:=matrix(4,1,[0.2,0.2,0.4,0.2]);
```

$$v_0 := \begin{bmatrix} 0.2 \\ 0.2 \\ 0.4 \\ 0.2 \end{bmatrix}$$

```
> seq(evalm(A^n &* v0), n=1..20);
```

```
[ 0.18 ] [ 0.1790 ] [ 0.18010 ] [ 0.1808450 ] [ 0.18119850 ] [ 0.1813472750 ]
[ 0.34 ] [ 0.3700 ] [ 0.37610 ] [ 0.3772100 ] [ 0.37735750 ] [ 0.3773516500 ]
[ 0.230 ] [ 0.1870 ] [ 0.175650 ] [ 0.1724950 ] [ 0.171566750 ] [ 0.1712779750 ]
[ 0.250 ] [ 0.2640 ] [ 0.268150 ] [ 0.2694500 ] [ 0.269877250 ] [ 0.2700231000 ]

[ 0.1814064925 ] [ 0.1814293771 ] [ 0.1814380744 ] [ 0.1814413475 ] [ 0.1814425722 ]
[ 0.3773356025 ] [ 0.3773264713 ] [ 0.3773223566 ] [ 0.3773206638 ] [ 0.3773199980 ]
[ 0.1711836113 ] [ 0.1711515388 ] [ 0.1711403163 ] [ 0.1711363090 ] [ 0.1711348585 ]
[ 0.2700742937 ] [ 0.2700926128 ] [ 0.2700992527 ] [ 0.2701016795 ] [ 0.2701025713 ]

[ 0.1814430288 ] [ 0.1814431987 ] [ 0.1814432617 ] [ 0.1814432852 ] [ 0.1814432937 ]
[ 0.3773197423 ] [ 0.3773196455 ] [ 0.3773196092 ] [ 0.3773195957 ] [ 0.3773195905 ]
[ 0.1711343288 ] [ 0.1711341344 ] [ 0.1711340626 ] [ 0.1711340361 ] [ 0.1711340263 ]
[ 0.2701029001 ] [ 0.2701030214 ] [ 0.2701030663 ] [ 0.2701030830 ] [ 0.2701030891 ]

[ 0.1814432970 ] [ 0.1814432982 ] [ 0.1814432987 ] [ 0.1814432988 ]
[ 0.3773195887 ] [ 0.3773195881 ] [ 0.3773195878 ] [ 0.3773195877 ]
[ 0.1711340226 ] [ 0.1711340214 ] [ 0.1711340209 ] [ 0.1711340207 ]
[ 0.2701030913 ] [ 0.2701030923 ] [ 0.2701030926 ] [ 0.2701030927 ]
```

Sólo hemos incluido 20 pasos porque, como el lector puede comprobar, la convergencia al vector \mathbf{v} es vertiginosa. Una última prueba, por si alguien sigue desconfiando. Partimos ahora de otro vector inicial (muy poco “probabilístico”, en verdad):

```
> v0:=matrix(4,1,[1,0,0,0]);
```

$$v0 := \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

```
> seq(evalm(A^n &* v0), n=1..20);
```

$$\begin{bmatrix} 0.4 \\ 0.4 \\ 0.15 \\ 0.05 \end{bmatrix}, \begin{bmatrix} 0.225 \\ 0.450 \\ 0.200 \\ 0.125 \end{bmatrix}, \begin{bmatrix} 0.1800 \\ 0.4325 \\ 0.20375 \\ 0.18375 \end{bmatrix}, \begin{bmatrix} 0.172375 \\ 0.411000 \\ 0.195000 \\ 0.221625 \end{bmatrix}, \begin{bmatrix} 0.1738750 \\ 0.3962125 \\ 0.18605625 \\ 0.24385625 \end{bmatrix}, \begin{bmatrix} 0.176548125 \\ 0.387480000 \\ 0.179746250 \\ 0.256225625 \end{bmatrix}, \\ \begin{bmatrix} 0.1785870000 \\ 0.3826536875 \\ 0.1758767188 \\ 0.2628825937 \end{bmatrix}, \begin{bmatrix} 0.1798643594 \\ 0.3800782875 \\ 0.1736694750 \\ 0.2663878782 \end{bmatrix}, \begin{bmatrix} 0.1805980957 \\ 0.3787323466 \\ 0.1724631013 \\ 0.2682064564 \end{bmatrix}, \begin{bmatrix} 0.1810000743 \\ 0.3780382222 \\ 0.1718214243 \\ 0.2691402792 \end{bmatrix}, \begin{bmatrix} 0.1812140502 \\ 0.3776834182 \\ 0.1714862253 \\ 0.2696163063 \end{bmatrix}, \\ \begin{bmatrix} 0.1813258456 \\ 0.3775031692 \\ 0.1713132813 \\ 0.2698577038 \end{bmatrix}, \begin{bmatrix} 0.1813835241 \\ 0.3774119945 \\ 0.1712248232 \\ 0.2699796580 \end{bmatrix}, \begin{bmatrix} 0.1814130230 \\ 0.3773660192 \\ 0.1711798568 \\ 0.2700411010 \end{bmatrix}, \begin{bmatrix} 0.1814280170 \\ 0.3773428878 \\ 0.1711571000 \\ 0.2700719950 \end{bmatrix}, \begin{bmatrix} 0.1814356046 \\ 0.3773312691 \\ 0.1711456201 \\ 0.2700875062 \end{bmatrix}, \\ \begin{bmatrix} 0.1814394320 \\ 0.3773254398 \\ 0.1711398425 \\ 0.2700952855 \end{bmatrix}, \begin{bmatrix} 0.1814413581 \\ 0.3773225181 \\ 0.1711369398 \\ 0.2700991840 \end{bmatrix}, \begin{bmatrix} 0.1814423258 \\ 0.3773210544 \\ 0.1711354833 \\ 0.2701011364 \end{bmatrix}, \begin{bmatrix} 0.1814428114 \\ 0.3773203216 \\ 0.1711347530 \\ 0.2701021139 \end{bmatrix}.$$

No hay trampas, ¡sale siempre el mismo resultado! Magia matemática.

8.2. Modelos de dinámica de poblaciones

En una cierta especie, los individuos se agrupan por grupos de edad C_1, \dots, C_n . La población inicial es $\mathbf{z}^{(0)} = (z_1^{(0)}, \dots, z_n^{(0)})$. Planteamos las siguientes hipótesis:

- cada individuo pasa al siguiente grupo en cada unidad de tiempo.
- en la etapa i , cada individuo da lugar a m_i descendientes.
- s_i es la fracción que sobrevive de la edad $i - 1$ a la edad i .

Este modelo, muy utilizado en Biología, es una generalización del modelo de cunicultura (vamos, la cría de conejos) que se planteó en su día Fibonacci (y lo que han dado de sí los dichos conejos¹⁶). La dinámica de la población viene determinada por el siguiente sistema matricial:

$$\begin{pmatrix} s_1 m_1 & s_1 m_2 & \cdots & s_1 m_{n-1} & s_1 m_n \\ s_2 & 0 & \cdots & 0 & 0 \\ 0 & s_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & s_n & 0 \end{pmatrix}^k \begin{pmatrix} z_1^{(0)} \\ \vdots \\ z_n^{(0)} \end{pmatrix} = \begin{pmatrix} z_1^{(k)} \\ \vdots \\ z_n^{(k)} \end{pmatrix}.$$

Si la matriz anterior (conocida como **matriz de Leslie**) es “buena”, entonces

$$\mathbf{z}^{(k)} \approx K \lambda_1^k \mathbf{v}_1 \quad \text{cuando } k \rightarrow \infty,$$

donde λ_1 es “el” autovalor dominante y \mathbf{v}_1 es su autovector asociado. De donde se deduce que el comportamiento asintótico (crecimiento, extinción o quizá oscilación) de la población depende de si λ_1 es > 1 , < 1 ó $= 1$.

¹⁶No los conejos en sí, sino la sucesión de números de Fibonacci que surge del análisis de la cuestión. Tanto es así que, quizá como caso único en las Matemáticas, existe una revista, *Fibonacci Quarterly*, íntegramente dedicada a la sucesión y sus conexiones.

8.3. Modelos económicos

Una economía (muy simplificada) consta de tres sectores, digamos agricultura, industria y servicios, cuyas producciones respectivas son x_1 , x_2 y x_3 (en las unidades que sean).

La hipótesis fundamental es que el consumo que de la producción x_i hace el sector j es proporcional a x_j (la producción de j). Éste es un modelo bien conocido en Economía, el **modelo input-output de Leontief**¹⁷.

Podríamos estar, por ejemplo, en una situación como la que se describe en la siguiente tabla:

	Agricultura	Industria	Servicios	Consumidor	total producido
Agricultura	$0.3 x_1$	$0.2 x_2$	$0.3 x_3$	4	x_1
Industria	$0.2 x_1$	$0.4 x_2$	$0.3 x_3$	5	x_2
Servicios	$0.2 x_1$	$0.5 x_2$	$0.1 x_3$	12	x_3

De las x_1 unidades producidas por el sector agrario, el 30% son “autoconsumidas”, $0.2 x_2$ utilizadas por la industria, $0.3 x_3$ por el sector de servicios, mientras que 4 unidades lo son por los consumidores finales.

Ahora, la cuestión, escrita en términos matriciales, adopta la forma

$$\mathbf{Ax} + \mathbf{b} = \mathbf{x}.$$

Y la pregunta es: dado un vector de consumo $\mathbf{b} \geq 0$, ¿tiene el sistema anterior una solución $\mathbf{x} \geq 0$?

Desde luego, será el caso si la matriz $\mathbf{I} - \mathbf{A}$ tiene una inversa no negativa, pues entonces

$$\mathbf{x} = (\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} \geq 0.$$

Una condición suficiente para garantizar la existencia de esta inversa no negativa es que el autovalor dominante de \mathbf{A} sea¹⁸ < 1 .

Coda

De un reto tecnológico formidable, como es el diseño de un buscador en la red, descubrimos que, al final, la clave son las Matemáticas: la astuta aplicación de teoremas y el cuidadoso análisis de la convergencia de los algoritmos. Una nueva comprobación de la *unreasonable effectiveness* de las Matemáticas en las Ciencias Naturales, que decía Wigner, y en tantos campos, que podríamos añadir nosotros.

Confiamos en que estas páginas, además de haber servido como entretenimiento para el lector, le animen a indagar por su cuenta en los numerosos problemas que aquí apenas hemos esbozado.

Y nos despedimos, con especial cariño, del teorema de Perron y Frobenius, que desempeña un papel tan destacado en tan variadas cuestiones. Lo hacemos con una coplilla, manriqueña y jocosa:

Un hermoso resultado
que además se nos revela
indiscreto;
y un tanto desvergonzado,
porque de Google desvela
... su secreto.

¹⁷En la exposición de motivos del Premio Nobel en Economía concedido a Leontief en 1973 se lee: “[...] por el desarrollo del método *input-output* y por sus aplicaciones a importantes problemas económicos”. El modelo que Leontief utilizó para analizar la economía de Estados Unidos en 1958 constaba de 81 sectores.

¹⁸La clave es que, en este caso, $\mathbf{I} + \mathbf{A} + \mathbf{A}^2 + \dots = (\mathbf{I} - \mathbf{A})^{-1}$.

Para saber más

- [BaRa] BAPAT, R. B.; RAGHAVAN, T. E. S.: *Nonnegative matrices and Applications*. Encyclopedia of Mathematics and its applications **64**. Cambridge University Press, 1997.
- [Ba1] BARABÁSI, A.-L.: The Physics of the web. *Physics world* (julio 2001). Se puede obtener en la página web del autor, <http://www.nd.edu/~alb>.
- [Ba2] BARABÁSI, A.-L.: *Linked: The New Science of Networks*. Plume, 2003.
- [BePl] BERMAN, A.; PLEMMONS, R. J.: *Nonnegative matrices in the Mathematical Sciences*. Classics in applied Mathematics **9**. Academic Press, 1979.
- [BrPa] BRIN, S.; PAGE, L.: *The anatomy of a large-scale hypertextual web search engine*. www-db.stanford.edu/~sergey/
- [Br] BRODER, A. ET AL.: *Graph structure in the web*. www9.org/w9cdrom/160/160.html
- [HaKa] HAWELIWALA, T.S.; KAMVAR, S.D.: The second eigenvalue of the Google matrix. www.stanford.edu/~taherh/papers/secondeigenvalue.pdf
- [Ke] KEENER, J. P.: The Perron-Frobenius Theorem and the ranking of football teams. *SIAM Review* **35** (1993), no. 1, 80-93.
- [Ken] KENDALL, M. G.: Further contributions to the theory of paired comparisons. *Biometrics* **11** (1955), 43-62.
- [Ma] MACLAUER, C. R.: The Many Proofs and Applications of Perron's Theorem. *SIAM Review* **42** (2000), no. 3, 487-498.
- [Mi] MINC, H.: *Nonnegative matrices*. Wiley-Interscience. New York, 1988.
- [Mo] MOLER, C.: *The world's largest matrix computation*. www.mathworks.com/company/newsletter/clevescorner/oct02_cleve.shtml
- [We] WEI, T. H.: *The algebraic foundations of ranking theory*. Cambridge University Press, London, 1952.
- [Wi] WILF, H. S.: *Searching the web with eigenvectors*. www.cis.upenn.edu/~wilf/