EL ÁLGEBRA LINEAL DETRÁS DE LOS BUSCADORES DE INTERNET

CARLOS D'ANDREA

La vérité est trop compliqué.
Les mathématiques sont simples.
Cédric Villani

1. Álgebra Lineal para Informáticos

Los alumnos que se matriculan en el grado de Ingeniería en Informática que ofrece esta facultad, en el primer semestre del primer año de estudios se encontrarán con la asignatura "Álgebra", que entre sus bloques temáticos ofrece el siguiente menú:

- Sistemas de ecuaciones lineales
- Matrices y determinantes
- Espacios vectoriales. Subespacios
- Transformaciones lineales. Núcleo, imagen, isomorfismos,...
- Polinomios
- Números complejos
- Diagonalización

Alguien con un mínimo entendimiento en estos temas se dará cuenta rápidamente que la diagonalización es un proceso que involucra *todos* los temas anteriores; y concluirá -con bastante certitud- que éste es un curso donde se aprende a (decidir cuándo se puede) diagonalizar matrices.

No hay nada de trivial ni de sarcástico en esta conclusión. Es indudable que el álgebra lineal en general -y el problema del cálculo de vectores y valores propios (que necesitamos conocer para decidir si una matriz es diagonalizable) en particular- son muy importantes en la informática, ya que están presentes en varios procesos centrales en esta disciplina. Podemos mencionar como ejemplos los siguientes:

- Agrupamiento y clasificación de datos
- Programación gráfica
- Redes sociales
- Descomposición en valores singulares para sistemas de recomendación
- Reconocimiento de formas (canciones, huellas digitales, fotografías)
- Inteligencia artificial

En el grado de Ingeniería en Informática de esta facultad, varios de estos temas serán cubiertos a lo largo de la carrera. Naturalmente, los alumnos "lo verán después" de haber acabado el curso de álgebra. Es entendible que no sea muy motivador para el alumnado aprender a utilizar unas herramientas que

serán indudablemente importantes, pero que todavía no podemos explicarles en qué lo serán y cómo se utilizarán estas herramientas.

Es por ello que he elegido presentar en esta clase, para motivar a los alumnos que comienzan a estudiar el álgebra que les estamos ofreciendo en esta Casa y también para mostrar a los más avanzados en ambas carreras (matemática e informática), un problema de valores y vectores propios (diagonalización) sencillo de enunciar, que ha sido utilizado recientemente y con mucho éxito en el mundo de la informática para resolver un problema de los mencionados más arriba, el problema de recomendación que tienen por delante los "motores de búsqueda" (o buscadores) de internet a la hora de sugerir al usuario qué páginas visitar como respuesta a unas ciertas palabras clave previamente introducidas por el mismo usuario en su ordenador.

Para ello nos concentraremos en un buscador específico, que es el más exitoso de todos, y en el algoritmo que inicialmente utilizaba y ha venido utilizando hasta hace muy poco. Este algoritmo produjo una verdadera revolución en el mundo del tráfico de información en línea. Y todo gracias al álgebra lineal.

2. Un buscador de internet muy rápido y eficiente

En el año 1996, dos jóvenes alumnos de doctorado de la Universidad de Stanford (EEUU), Sergei Brin y Lawrence Page comenzaron a trabajar en el diseño de un buscador de internet. Ambos tenían 23 años en ese momento. Brin se había graduado en matemáticas y Page, en informática.

Figura 1. Sergei Brin (izquierda) y Larry Page (derecha)

El algoritmo que iba a utilizar este buscador de internet fue denominado "PageRank", dado que Page ya había comenzado inicialmente con el proyecto al que luego se incorporó Brin (cf. [BP98]), y acabó siendo implementado por **Google**. En efecto, en 1998 el algoritmo es patentado, y al mismo tiempo aparece en internet el buscador Google que fue también realizado por Brin y Page. Desde sus inicios, Google va a utilizar este algoritmo exitosamente para posicionarse desde muy temprano (y hasta nuestros días) como líder en el mercado de los buscadores de internet.

La palabra "google" es una variación fonética del término **googol** con el que se denomina (en inglés) al número 10^{100}. Sus fundadores pretendían ofrecer un buscador que fuera rápido y eficiente. De hecho, el objetivo inicial de Brin y Page era que al menos *una* de las diez primeras páginas que mostrara el buscador contenga información útil para la persona que la consulta.

El éxito que ha tenido Google desde sus inicios hasta el día de hoy no necesita ser explicado aquí; sin lugar a dudas se trata del buscador de internet más utilizado en todo el mundo, batiendo records de popularidad impensables. Por citar un ejemplo, en mayo de 2011 consiguió superar los mil millones de visitantes al mes por primera vez en la historia. De más está decir que ningún otro buscador de internet se ha siquiera acercado a esta cifra.

Este suceso también se traduce obviamente en las finanzas, ya que cuando salió a cotizar en el mercado de valores en 2004, la compañía estaba valorada en aproximadamente $ 25.000.000.000, cifra que ha ido creciendo a lo largo del tiempo, alcanzando los $ 37.905.000.000 enel último reporte de 2011. Y todo por diagonalizar unas matrices...

Para intentar explicar brevemente el algoritmo PageRank y ver cómo aparecen naturalmente los vectores y valores propios en este tema, primero tenemos que ver cómo se modela matemáticamente un buscador de internet, ya que este algoritmo forma parte fundamental de la arquitectura del mismo.

3. Los buscadores de internet

Uno podría comparar el trabajo de un buscador con el de un bibliotecario. Para hacerlo más explícito, digamos que se trata de un bibliotecario de las épocas en las que no había ordenadores. Si uno acudía a la biblioteca en aquellos cada vez más lejanos tiempos intentando encontrar información sobre algún tema en particular, se iba a encontrar con un gran fichero o catálogo enorme, impreso, conteniendo toda la información existente en esa biblioteca hasta la última actualización. Con un poco de suerte además había también alguna especie de catálogo-diccionario, relacionando libros con algunas palabras clave.

Supongamos ahora que yo me acercara a una de esas bibliotecas antiguas porque me han enviado a investigar sobre el tema "jirafa". No me han dado ninguna referencia bibliográfica, y sé que la información que pudiera proporcionarme un diccionario y/o enciclopedia no me será suficiente. ¿Qué había de hacer? La respuesta más simple en esos tiempos era: preguntar al bibliotecario, y consultar las referencias recomendadas por él. Si no quedara satisfecho con su/s recomendación/es, habría que o bien preguntarle con más precisión sobre lo que estoy buscando, o buscarse otra biblioteca.

Toda esta interacción con el bibliotecario que estoy contando parece casi trivial y uno podría preguntarse por qué os estoy haciendo perder tiempo contando esta historia tan aburrida. Pero supongamos ahora que mi biblioteca contiene más de mil millones de libros, y que bajo la palabra clave "jirafa" hay cuatro millones de textos que tienen algo que decir al respecto, y que para enumerarme uno por uno todos estos textos -a razón de un texto cada 10 segundos- el bibliotecario demoraría casi 463 días. Yo claramente no necesito leer los cuatro millones de libros para hacer el trabajo que me toca, quizás con 10 de ellos ya me alcance. Pero entonces... ¿cuáles 10? El algoritmo PageRank es justamente quien va a ayudarme (o más bien, ayudar al bibliotecario) a decidir sobre cómo ordenar la lista de salida, cuáles son los libros que tiene que recomendarme

de tal manera que pueda encontrar yo información útil dentro de las primeras referencias que me vaya dando.

Un buscador de internet esencialmente es una especie de "catálogo de biblioteca" junto con un "bibliotecario" que te recomienda qué libros leer. El éxito de este buscador depende justamente de tener una buena base de datos, ordenada de acuerdo a palabras clave de una manera razonable, y también un buen "recomendador", ya que uno quiere acceder a la información de manera rápida y eficiente.

La tarea de "censar" las páginas webs es hecha por unos robots que circulan por la red continuamente. Notar que éste es un procedimiento dinámico ya que hay miles de páginas nuevas que aparecen en la red minuto a minuto, y varias (pocas respecto de las nuevas) que desaparecen. Y uno quiere que la información esté siempre actualizada, así que este trabajo es muy importante. Otro elemento también a tener en cuenta es que esta base de datos es *enorme*, y crece exponencialmente. En 1998 cuando fue lanzada Google, tenía 26 millones de páginas. En 2008 (cf. [Goo08]) alcanzó el billón (1.000.000.000.000) de entradas.

El trabajo de "catalogar", es decir indexar los datos censados en función de ciertas palabras clave también es hecho por programas informáticos, que estudian distribuciones estadísticas de palabras, frecuencias de aparición y enlaces a esa página, para hacer este trabajo.

O sea que todo buscador de internet tiene que tener tanto un buen catálogo de páginas indexadas, así como un buen índice en lo que respecta a las palabras clave. Bien... ¿Cómo se hace el trabajo de bibliotecario? Es decir, ¿cómo decido qué páginas mostrar primero cuando alguien pone en el buscador la palabra "jirafa"?

Hay miles de algoritmos y programas dedicados a responder esta pregunta, entre ellos el algoritmo PageRank, que es el que catapultó a Google al éxito entre los buscadores de internet. En la sección siguiente nos dedicaremos a explicarlo.

4. El modelo PageRank. Vectores y valores propios

Tal como hemos explicado hasta ahora, lo que faltaría para completar el trabajo del buscador es asignarle una "importancia" a cada página web de las que tengo censadas. Para ello, la *teoría de grafos* nos ayudará a modelar nuestra situación.

En el modelo PageRank, el universo de las páginas web indexadas es un gran *grafo dirigido,* donde cada página web censada es un *nodo*, y habrá una "flecha" (*arista orientada*) desde la página p_i hacia la página p_j si hay un enlace desde la primera página hacia la segunda. Por ejemplo, si el gráfico de la figura 2 fuera lo que vamos a llamar a partir de ahora *el grafo de internet*, entonces podríamos concluir de este dibujo que -por ejemplo- la primera página es la más popular ya que hay enlaces desde todas las otras hacia ésta, y es la única que cumple con esta propiedad.

En este gran grafo dirigido, uno tiene ahora que asignar una "importancia" a cada página. Una manera razonable de asignar importancias podría ser que

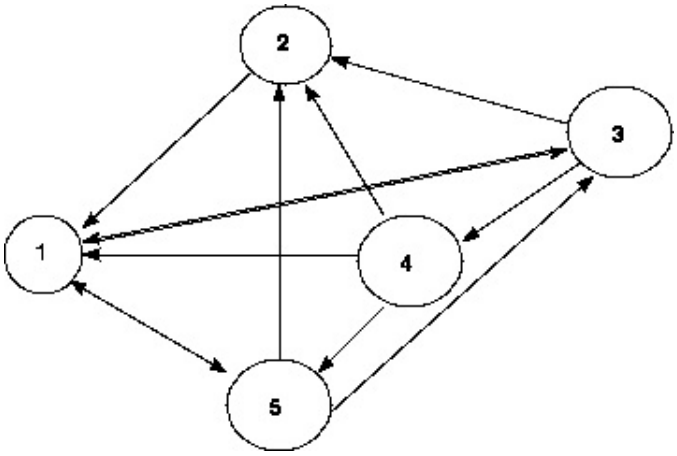

FIGURA 2. Un grafo dirigido

cuantos más enlaces recibe una página, más importante será. Notar la analogía aquí con el famoso y no siempre tan apreciado *índice de citas* que intenta regular el currículo de los investigadores.

El problema que tiene este modelo aparentemente sensato es que uno podría "inflar" rápidamente la importancia de una página web determinada simplemente creando varias páginas que tengan enlaces con la misma, y este procedimiento es muy fácil de implementar en pocos minutos, lo cual haría que todo el sistema fuera muy fácil de influir.

Para evitar esto, cambiaremos la función "número de citas" por "importancia de las citas". Es decir, no solo vamos a darle importancia a la cantidad de citas que tiene una página dada, sino que también tendremos en cuenta si la citan páginas importantes. Digamos que -por ejemplo- si obtengo enlaces desde `Amazon.com` o `Microsoft.com`, mi importancia debería ser mayor. En ese sentido, el grafo de las páginas web sería algo más bien parecido a lo que aparece en la figura 3, donde se ve una distribución de importancias relativas a las importancias de las páginas dadas. Aquí se entiende por qué la página "C" es más importante que la "F" dado que ambas reciben un enlace cada una, pero la primera es enlazada por una página mucho más importante que la segunda.

Dicho en palabras, el "postulado" del modelo PageRank dice lo siguiente:
La importancia x_i de la página p_i es directamente proporcional a la suma de las importancias de las páginas que enlazan con ella.

Veamos cómo se traduce esto matemáticamente, y cómo aparece el álgebra lineal naturalmente en este contexto. El dibujo de un grafo (dirigido o no) es ilustrativo e interesante si se trata de pocos nodos, pero el grafo de internet tiene más de un billón de páginas así que no vamos a ganar mucho intentando dibujarlo (y perderemos mucho tiempo y tinta). En lugar de ello, consideraremos la *matriz de incidencia* del grafo, que se define como la matriz cuadrada de tamaño igual a la cantidad de nodos del grafo. En esta matriz, pondremos un 1 en el lugar (i, j) si hay un enlace desde p_i hasta p_j. Si no lo hay, pondremos un cero. Por

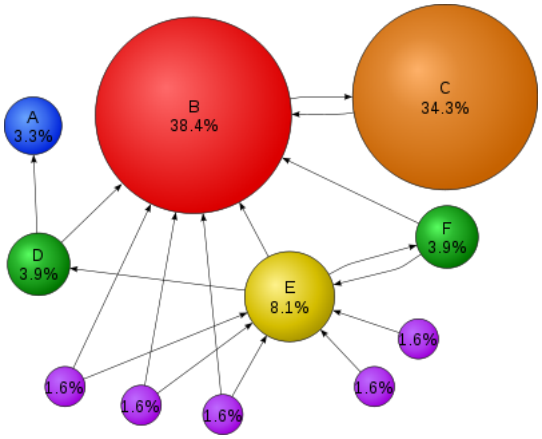

FIGURA 3. El grafo de "importancias"

ejemplo, la matriz de incidencia del grafo de la figura 2 es la siguiente

$$\mathbb{M}_0 = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Y aquí es donde aparece el álgebra lineal junto con los vectores y valores propios.

Teorema 4.1. *Si $\mathbb{M}_I$ es la matriz de incidencia del grafo de internet, y $\mathbf{x} = (x_1, \ldots, x_N) \in \mathbb{R}_{\geq 0}{}^n$ el vector de importancias, entonces se cumple*

$$\mathbb{M}_I^t \mathbf{x}^t = \lambda \cdot \mathbf{x}^t$$

donde $\lambda \in \mathbb{R}_{>0}$ es la constante de proporcionalidad.

Y aquí viene bien recordar algunas definiciones clásicas del álgebra lineal.

Definición 4.2. Dados una matriz cuadrada $\mathbb{M}$ de tamaño $n \times n$, un vector no nulo $\mathbf{x} \in \mathbb{R}^n$ (o $\mathbb{C}^n$) y un número $\lambda \in \mathbb{R}$ (o $\mathbb{C}$), el vector $\mathbf{x}$ se dice *vector propio* de $\mathbb{M}$ con *valor propio* asociado λ si y solo si se verifica

$$\mathbb{M}\mathbf{x}^t = \lambda \cdot \mathbf{x}^t.$$

Corolario 4.3. *El vector de importancias de las páginas web es un vector propio (positivo) de la matriz $\mathbb{M}_I^t$, y la constante de proporcionalidad λ es el valor propio asociado a este vector.*

Ejemplo 4.4. *Veamos cómo efectivamente lo que dice el postulado PageRank y lo que enunciamos en el Teorema 4.1 coinciden. Digamos que λ es la constante*

de proporcionalidad de la que habla el postulado. Entonces, de acuerdo con esa afirmación, tenemos las siguientes ecuaciones:

$$\begin{cases} 0\cdot x_1 & +1\cdot x_2 & +1\cdot x_3 & +1\cdot x_4 & +1\cdot x_5 & = & \lambda x_1 \\ 0\cdot x_1 & +0\cdot x_2 & +1\cdot x_3 & +1\cdot x_4 & +1\cdot x_5 & = & \lambda x_2 \\ 1\cdot x_1 & +0\cdot x_2 & +0\cdot x_3 & +1\cdot x_4 & +1\cdot x_5 & = & \lambda x_3 \\ 0\cdot x_1 & +0\cdot x_2 & +1\cdot x_3 & +0\cdot x_4 & +0\cdot x_5 & = & \lambda x_4 \\ 1\cdot x_1 & +0\cdot x_2 & +0\cdot x_3 & +1\cdot x_4 & +0\cdot x_5 & = & \lambda x_5, \end{cases}$$

que en notación matricial es precisamente

$$\mathbb{M}_0^t \mathbf{x}^t = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}^t \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_7 \end{pmatrix} = \lambda \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_7 \end{pmatrix} = \lambda \cdot \mathbf{x}^t.$$

Notar que lo que hemos visto en este ejemplo no es casualidad, ya que si en las filas de la matriz de incidencia del grafo de internet uno puede leer cuántos enlaces salen de una página dada, justamente en las columnas aparecerán tantos unos como enlaces haya hacia la página indexada por esa columna. Es por ello que se necesita trasponer la matriz para utilizarla en el problema del PageRank.

Una vez establecido el problema... uno quisiera encontrar la solución. ¿Cómo lo haríamos para nuestro grafo de la figura 2? Utilizando alguna de las herramientas computacionales que tenemos a disposición (por ejemplo, el programa `Mathematica` que es el que se utiliza en las prácticas de laboratorio de la asignatura de Álgebra), nos encontramos con lo siguiente:

- *Posibles valores (aproximados) para λ (valores propios de $\mathbb{M}_0^t$)*:

$$2{,}27;\ -0{,}500-0{,}866i;\ -0{,}500+0{,}866i;\ -0{,}635+0{,}692i;\ -0{,}635-0{,}692i$$

- *Posibles valores (aproximados) para los vectores propios (ordenados con respecto a los valores propios enumerados arriba):*

$(1{,}74; 1{,}21; 1{,}21; 0{,}532; 1{,}00)$,
$(0,0; -0{,}500-0{,}866i; -0{,}500+0{,}866i; 1{,}00)$,
$(0,0; -0{,}500+0{,}866i; -0{,}500-0{,}866i; 1{,}00)$,
$(-0{,}469+0{,}101i; -0{,}303-0{,}490i; -0{,}303-0{,}490i; -0{,}166+0{,}591i; 1{,}00)$,
$(-0{,}469-0{,}101i; -0{,}303+0{,}490i; -0{,}303+0{,}490i; -0{,}166-0{,}591i; 1{,}00)$

En este caso en particular, el del grafo asociado a la figura 2, dado que la constante de proporcionalidad tiene que ser real y positiva, parecería ser que hay una única solución al problema que sería la siguiente:

- $\lambda = 2{,}27$
- $x_1 = 1{,}74$, $x_2 = 1{,}21$, $x_3 = 1{,}21$, $x_4 = 0{,}532$, $x_5 = 1$,

lo cual parece ser una respuesta razonable ya que la primera página es votada por todas las otras, y es la única con esta situación. O sea que merece ganar la

contienda, y si bien la segunda tiene un voto más que la tercera, esta última es votada por la más importante (la primera) mientras que la otra no.

Uno podría suponer que lo que ocurre en este ejemplo es un hecho general, que de cualquier matriz cuadrada con ceros y unos habrá un único valor propio positivo, y asociado al mismo un solo vector propio positivo que será la solución a nuestro problema. Lamentablemente la respuesta a esta pregunta no es cierta, ya que -por ejemplo- una matriz tan sencilla como $\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$ solamente tiene como único valor propio a $\lambda = 0$. También es fácil construirse matrices de tamaño más grande con más de un valor propio positivo. Entonces... ¿cómo hacemos para resolver esta ambigüedad?

Antes de responder esta pregunta, hagamos una modificación pequeña pero de vital importancia al modelo. Tal como lo hemos explicado hasta aquí, este modelo fue el que originalmente se utilizó en el buscador Google durante sus primeros años. Con el transcurrir del tiempo y el advenimiento de las redes sociales (muy propensas a producir hechos "en cadena", en varios lugares y al mismo tiempo) se encontró una falla en el modelo previsible desde un primer momento: si una página tiene un solo enlace, este enlace vale lo mismo que cualquier otro enlace de otra página que produzca un millón de enlaces. Es como -si bien producir enlaces desde mi propia página no aumenta mi importancia- cuantos más enlaces produce mi página, más afecta a toda la red.

Para evitar ese exceso autoridad, el modelo se modificó sencillamente de la manera siguiente: si hubiera un enlace desde p_i hacia p_j, en el lugar (i, j) de la matriz de incidencia se coloca el número $\frac{1}{\#\text{enlaces desde}\, p_i}$. De esta manera, cada página tiene "poder de voto" igual a 1, y esta unidad se va distribuyendo de acuerdo a los enlaces.

Por ejemplo, la matriz modificada asociada al grafo de la figura 2 , que llamaremos $\mathbb{M}_{0,E}$, es la siguiente:

$$\mathbb{M}_{0,E} = \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 \end{pmatrix}.$$

Notar que ahora en cada fila hay una distribución de números no negativos que suman 1, como si fuera una distribución de probabilidades sobre los nodos del grafo de internet. Este tipo de matrices se conoce como *estocástica por filas*, y aparece con frecuencia en el modelado de procesos diversos que mencionaremos luego.

Calculemos ahora los valores y vectores propios de la matriz estocástica.

- *Posibles valores (aproximados) para λ (valores propios de $\mathbb{M}_{0,E}^t$)*:

$$1{,}00; -0{,}333 + 0{,}471i; -0{,}333 - 0{,}471i; -0{,}167 + 0{,}289i; -0{,}167 - 0{,}289i$$

- *Posibles valores (aproximados) para los vectores propios (en orden con respecto a los valores propios enumerados arriba):*

  $(1{,}73; 0{,}867; 1{,}20; 0{,}400; 1{,}00)$,
  $(-0{,}333 + 0{,}943i; -0{,}667 - 0{,}236i; 0{,}500 - 0{,}707i; -0{,}500; 1{,}00)$,
  $(-0{,}333 - 0{,}943i; -0{,}667 + 0{,}236i; 0{,}500 + 0{,}707i; -0{,}500; 1{,}00)$,
  $(0; 0; -0{,}500 - 0{,}866i; -0{,}500 + 0{,}866i; 1{,}00)$,
  $(0; 0; -0{,}500 + 0{,}866i; -0{,}500 - 0{,}866i; 1{,}00)$.

Aquí también parece haber una única solución que consiste en $\lambda = 1$ y

$$x_1 = 1{,}73;\ x_2 = 0{,}867;\ x_3 = 1{,}20;\ x_4 = 0{,}400;\ x_5 = 1{,}00.$$

Notar la diferencia sutil que hay entre las dos distribuciones de importancia. Mientras que en el primer modelo las páginas 2 y 3 reciben la misma importancia, en el segundo la tercer página "le gana" a la segunda, siendo que la tercera recibe solo dos votos y la segunda tres. El motivo de esta diferencia puede ser explicado por el hecho de que la página 3 no solo es votada por la primera página que es la más importante, sino que además la primera página solo emite dos votos, mientras que todas las que votan a la página 3 emiten 3 votos. Es decir, estos votos cuentan por menos que los de la primera página.

Notar también que el hecho de que 1 sea un valor propio de la matriz estocástica no es casualidad, ya que es fácil comprobar que el vector $(1, 1, \ldots, 1)$ es siempre un vector propio de toda matriz estocástica por columnas, asociado al valor propio $\lambda = 1$.

5. Solución del Problema... ¿Unicidad?

La respuesta al problema de la unicidad viene de la mano de otra rama de la matemática que es el *Análisis Funcional*. El primer resultado en esta dirección fue dado por Oskar Perron a principios del siglo pasado.

Teorema 5.1 (Perron, 1907)**.**
Sea $\mathbb{M}$ *una matriz cuadrada con todos sus coeficientes* **positivos**. *Entonces*

- *existe un valor propio simple* $\lambda > 0$ *tal que* $\mathbb{M} \cdot \mathbf{v}^t = \lambda \cdot \mathbf{v}^t$, *donde* $\mathbf{v}$ *es el vector correspondiente, y tiene todas sus coordenadas* **positivas**
- *este valor propio es mayor, en módulo, que todos los demás valores propios de* $\mathbb{M}$
- *Cualquier otro vector propio positivo de* $\mathbb{M}$ *es un múltiplo de* $\mathbf{v}$

Este teorema nos trae una cierta unicidad que consistiría en quedarnos con el único vector propio positivo de la matriz, el asociado al valor propio más grande que todos los otros (en módulo). Lamentablemente, nuestras matrices $\mathbb{M}_I$ están asociadas a los grafos de páginas de internet, y tienen muchos ceros. Están muy lejos de ser positivas. ¡Para que ello ocurra necesitaríamos que se enlazaran todas las páginas con todas, incluso con ellas mismas!

Un resultado un poco más general es imposible como nos enseña el ejemplo de la matriz $\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$. Sin embargo, Frobenius consiguió una versión del Teorema

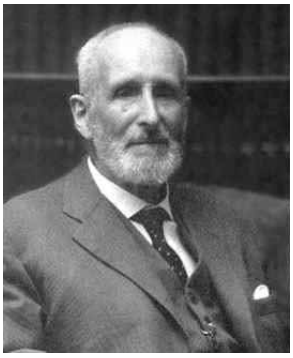

FIGURA 4. Oskar Perron (1880–1975)

de Perron para matrices no negativas, bajo cierta condición adicional sobre la matriz de entrada. Enunciemos primero el resultado y veamos luego las restricciones que se nos impone, quizás con la esperanza de que el grafo de internet sí que las cumpla.

Teorema 5.2 (Frobenius, 1908–1912)**.**
Sea $\mathbb{M}$ *una matriz cuadrada con entradas* **no negativas**. *Si* $\mathbb{M}$ *is* **irreducible**, *entonces*

- *existe un valor propio simple* $\lambda > 0$ *tal que* $\mathbb{M} \cdot \mathbf{v}^t = \lambda \cdot \mathbf{v}^t$, *donde* $\mathbf{v}$ *es el vector correspondiente, y tiene todas sus coordenadas* **positivas***;*
- *este valor propio es* **mayor o igual,** *en módulo, que todos los demás valores propios de* $\mathbb{M}$;
- *cualquier otro vector propio con entradas* **no negativas** *de* $\mathbb{M}$ *es un múltiplo de* $\mathbf{v}$.

FIGURA 5. Georg Frobenius (1849–1917)

Definición 5.3. Una matriz $\mathbb{M}$ se dice **irreducible** si no existe ninguna permutación de sus filas y columnas que la transforme en otra matriz del tipo

$$\begin{pmatrix} \mathbb{M}_{11} & \mathbb{M}_{12} \\ \mathbf{0} & \mathbb{M}_{22} \end{pmatrix},$$

donde $\mathbb{M}_{11}$ y $\mathbb{M}_{22}$ son matrices cuadradas.

Proposición 5.4. *Si* $\mathbb{M}$ *es la matriz de incidencia de un grafo dirigido, entonces* $\mathbb{M}$ *irreducible es equivalente a que el grafo sea "fuertemente conexo", es decir que dados dos nodos cualesquiera del mismo, es posible encontrar una sucesión de aristas que lleven de uno a otro.*

Esta proposición es de hecho interesante ya que no es trivial calcular una descomposición que valide la irreducibilidad de una matriz, pero sin embargo -para un número de nodos relativamente pequeño- es inmediato verificar si un grafo es fuertemente conexo o no. Como ejercicio para el lector dejamos el de mostrar que el grafo de la figura 2 es fuertemente conexo.

De todos modos, es altamente improbable que el grafo de internet sea fuertemente conexo. De hecho, un estudio hecho en 1999 ([Bro99]) mostraba una distribución del grafo de internet como se ve en la figura 6. De 203 millones de páginas que había censadas en ese momento, un 90 % estaba en una gigantesca componente conexa, y solo 56 millones estaban conectados "fuertemente".

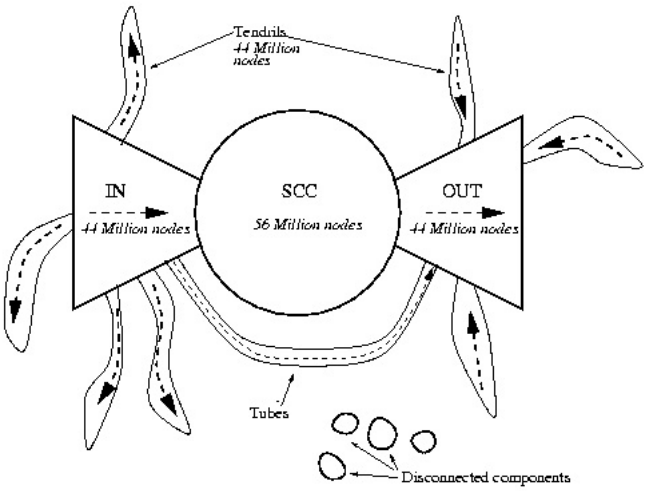

FIGURA 6. Componentes conexas del grafo de internet en 1999

O sea que ni el Teorema de Perron ni el de Frobenius se pueden aplicar directamente a nuestro problema. ¿Qué hacemos, entonces? ¿Qué hace Google?

La salida a este aparente callejón sin salida viene de la mano de una "perturbación", algo muy frecuente en la *matemática computacional* y el *álgebra lineal numérica*, donde es habitual trabajar con datos aproximados. Aquí lo que haremos será algo muy ingenuo pero eficiente, reemplazaremos nuestra matriz estocástica (que denotaremos con $\mathbb{M}_{I,E}$) por una matriz a la que haremos positivos todos sus coeficientes sumándole una matriz conveniente. El principio subyacente a esta idea es que la función "importancia" es continua, y si puedo calcularla "cerca" de la situación donde me encuentro, ya me alcanza para lo que quiero que es ordenar las importancias y no realmente calcularlas.

En símbolos, dado $\varepsilon > 0$, muy pequeño, definimos

$$\mathbb{M}^{\varepsilon}_{I,E} := (1-\varepsilon)\cdot \mathbb{M}_{I,E} + \frac{\varepsilon}{n}\cdot \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \dots & 1 \end{pmatrix}.$$

Notar que esta operación nos vuelve a dar una matriz estocástica por filas, pero que ahora tiene todas sus entradas positivas (mayores o iguales que $\frac{\varepsilon}{n}$)! La matriz $\mathbb{M}_{I,E}$ cumple con las hipótesis del Teorema de Perron, y entonces "declaramos" que "la" solución al problema es la que se obtiene según ese enunciado para un ε prefijado (en la práctica, Google utiliza $\varepsilon = 0{,}15$).

Ejemplo 5.5. *Calculemos la matriz "perturbada" de nuestro grafo 2:*

$$\mathbb{M}^{\varepsilon}_{0,E} = \begin{pmatrix} 0{,}03 & 0{,}03 & 0{,}455 & 0{,}03 & 0{,}455 \\ 0{,}88 & 0{,}03 & 0{,}03 & 0{,}03 & 0{,}03 \\ 0{,}313 & 0{,}313 & 0{,}03 & 0{,}313 & 0{,}03 \\ 0{,}313 & 0{,}313 & 0{,}03 & 0{,}03 & 0{,}313 \\ 0{,}313 & 0{,}313 & 0{,}313 & 0{,}03 & 0{,}03 \end{pmatrix}.$$

Si ahora pedimos a `Mathematica` *que nos calcule el vector propio positivo de esta matriz (asociado al valor propio $\lambda = 1$) obtenemos*

$$(0{,}67259; 0{,}363478; 0{,}463318; 0{,}194141; 0{,}403921).$$

Este vector de importancias induce el mismo orden que el que produce la matriz sin perturbar. Es decir, que antes y después de la perturbación teníamos este orden entre las páginas:

$$x_1 > x_3 > x_5 > x_2 > x_4.$$

6. La solución computacional

Todo parece muy bonito y agradable con una matriz de 5×5, pero si quisiéramos calcular la verdadera solución al problema de Google, con la matriz de más un billón de entradas, ¿Cómo se hace? ¿Cómo lo hace Google?

Desde ya que desaconsejamos a cualquier optimista intentar utilizar las técnicas aprendidas (o por aprender) en el curso de álgebra que involucrarían calcular un polinomio característico de grado mayor que un billón, encontrar todas sus raíces, y detectar entre todas ellas la que tiene módulo máximo, y luego resolver un sistema lineal enorme para calcular el vector de importancias. Incluso si supiéramos el valor de λ (que en los casos estocásticos es 1), resolver un sistema lineal del orden de un billón es una tarea dantesca que no puede ser realizada en poco tiempo ni siquiera por los ordenadores más rápidos que hay disponibles en este momento.

Google utiliza lo que se llama el *método de las potencias*, en apariencia bastante ingenuo de enunciar pero computacionalmente muy efectivo. Se basa en el siguiente hecho bastante simple: si una matriz cuadrada $\mathbb{M}$ es diagonalizable y

tiene todos sus vectores propios $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ numerados de tal manera que los autovalores correspondientes cumplan lo siguiente

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

partiendo de $\mathbf{v}_0 \geq 0$ tal que

$$\mathbf{v}_0 = \alpha_1 \mathbf{v}_1 + \ldots + \alpha_n \mathbf{v}_n,$$

con $\alpha_1 \neq 0$, entonces se tendrá

$$\mathbb{M}^k \mathbf{v}_0 = \alpha_1 \lambda_1^k \mathbf{v}_1 + \ldots + \alpha_n \lambda_n^k \mathbf{v}_n.$$

Luego

$$\lim_{k\to\infty} \frac{\mathbb{M}^k \mathbf{v}_0}{\lambda_1^k} = \alpha_1 \, \mathbf{v}_1,$$

un múltiplo no trivial del vector propio buscado.

Este es el método que utiliza Google para ordenar sus páginas de internet, y con resultados bastante razonables. Un análisis con más detalle de la velocidad de convergencia y aspectos relacionados se puede encontrar en [LSW09, S-C05, Wil07].

7. Googlepílogo

El algoritmo PageRank es ahora marca registrada de Google, y está patentado en los Estados Unidos. Debido a temas legales, la patente está asignada a la Universidad de Stanford y no a Google. Sin embargo, la compañía de internet tiene derechos exclusivos sobre esa patente, y Stanford recibió 1.800.000 acciones de Google por permitirle el uso exclusivo de esa patente.

Una versión modificada del PageRank fue propuesta recientemente (cf. [BRSR06]) como alternativa al polémico factor de impacto elaborado por el ISI. Una implementación del mismo se puede encontrar en `http://www.eigenfactor.org`. También se ha aplicado para predecir concentraciones humanas en calles o plazas (cf. [Jia06]), para modelos de evolución en ecosistemas (cf. [AP09]), en otros tipos de búsquedas de internet, e incluso en análisis de redes de proteínas (cf. [IG11])

PageRank fue utilizado por Google hasta muy recientemente. En febrero de 2011 la compañía comenzó a hacer pruebas de un nuevo algoritmo de búsqueda bautizado "Google Panda", que esencialmente tiene la capacidad de modificar la importancia de secciones enteras de la lista, y no solamente páginas individuales. Google Panda reemplazó definitivamente a PageRank en abril de 2011, y continúa liderando el mercado de buscadores de internet hasta el día de hoy.

8. ¿Qué hemos aprendido hoy?

Uno podría sentirse un poco engañado luego de esta presentación del algoritmo PageRank, dado que hemos prometido al principio ver una aplicación sencilla de los vectores y valores propios, que nos ha sido de utilidad para modelar el problema del PageRank. Pero luego, para resolver este problema, nuestro camino se ha convertido en un verdadero "tour de force" por varias ramas de la

matemática: Teoría de Grafos, Análisis Funcional, Cálculo Numérico, Matrices Estocásticas, Matemática Computacional...

Según la Wikipedia [Wiki], un ingeniero es alguien que resuelve *problemas que afectan la actividad cotidiana de la sociedad.* Y más adelante agrega *la ingeniería es la actividad de trasformar el conocimiento en algo práctico.*

El trabajo de Brin y Page es un buen ejemplo de ello, ya que cualquier persona que quiera trabajar resolviendo problemas necesita de recursos, de herramientas. Y cada una de estas áreas de la matemática y de la informática tiene que ser para el ingeniero precisamente eso, una herramienta. Y cuanto más herramientas tengamos, mejor.

8.1. Para saber más... y profundizar las ideas y resultados matemáticos y computacionales que se encuentran alrededor del algoritmo PageRank, sugerimos los trabajos [BL06, Fer04, Gim11, LM06, MGF06, Wil06], y muchos más que se encuentran en las referencias bibliográficas de estas obras. Y seguramente si uno "googlea" alguna de estas palabras clave... ¡encontrará mucho más para leer!

Agradecimientos: Agradezco a David Cox por haberme introducido en este fascinante mundo del álgebra lineal de Google, y también las sugerencias y comentarios de Teresa Cortadellas, Emiliano Gómez, Gabriela Jerónimo, Pablo Mislej, Adrian Paenza y Juan Pablo Pinasco sobre una versión preliminar de estas notas.

Referencias

[AP09] Allesina, Stefano; Pascual, Mercedes. *Googling Food Webs: Can an Eigenvector Measure Species' Importance for Coextinctions?* PLOS Computational Biology, Issue 9, Volume 5, September 2009. http://dx.plos.org/10.1371/journal.pcbi.1000494

[BRSR06] Johan Bollen, Marko A. Rodriguez, and Herbert Van de Sompel.; Rodriguez; Van De Sompel. *Journal Status* . Scientometrics 69 (3): 1030, December 2006). http://arxiv.org/abs/cs/0601030

[BP98] Brin, S. and Page, L. *The Anatomy of a Large-Scale Hypertextual Web Search Engine.* In: Seventh International World-Wide Web Conference (WWW 1998), April 14–18, 1998, Brisbane, Australia.

[Bro99] Broder, A. et all. *Graph structure in the web.* http://www9.org/w9cdrom/160/160.html

[BL06] Bryan, Kurt; Leise, Tanya. *The $ 25,000,000,000 eigenvector: the linear algebra behind Google.* SIAM Rev. 48 (2006), no. 3, 569–581

[CS10] Cicone, Antonio; Serra-Capizzano, Stefano. *Google PageRanking problem: the model and the analysis.* J. Comput. Appl. Math. 234 (2010), no. 11, 3140–3169.

[Fer04] Fernández, Pablo. *El secreto de Google y el Álgebra lineal.* Bol. Soc. Esp. Mat. Apl. Nro. 30 (2004), 115–141.

[Gim11] Gimbert, Joan. *The mathematics of Google: the PageRank algorithm.* Butl. Soc. Catalana Mat. 26 (2011), no. 1, 29–55, 97.

[Goo08] Google official Blog. *We knew the web was big...* http://googleblog.blogspot.com.es/2008/07/we-knew-web-was-big.html

[IG11] Ivan, G; Grolmusz,V. *When the Web meets the cell: using personalized PageRank for analyzing protein interaction networks.* Bioinformatics (Vol. 27, No. 3. pp. 405-407) 27 (3): 405–7, 2011.

[IW06] Ipsen, Ilse C. F.; Wills, Rebecca S. *Mathematical properties and analysis of Google's PageRank.* Bol. Soc. Esp. Mat. Apl. No. 34 (2006), 191–196.

[Jia06] Jiang, B. *Ranking spaces for predicting human movement in an urban environment.* International Journal of Geographical Information Science 23 (7), 2006, 823–837.

[LM06] Langville, Amy N.; Meyer, Carl D. *Google's PageRank and beyond: the science of search engine rankings.* Princeton University Press, Princeton, NJ, 2006. x+224 pp. ISBN: 978-0-691-12202-1; 0-691-12202-4

[LSW09] Lin, Yiqin; Shi, Xinghua; Wei, Yimin. *On computing PageRank via lumping the Google matrix.* J. Comput. Appl. Math. 224 (2009), no. 2, 702–708.

[MGF06] Madrid de la Vega, Humberto; Guerra Ones, Valia; Flores Garrido, Marisol. *The numerical linear algebra of Google's PageRank.* Papers of the Mexican Mathematical Society (Spanish), 33–52, Aportaciones Mat. Comun., 36, Soc. Mat. Mexicana, México, 2006.

[S-C05] Serra-Capizzano, Stefano. *Jordan canonical form of the Google matrix: a potential contribution to the PageRank computation.* SIAM J. Matrix Anal. Appl. 27 (2005), no. 2, 305–312.

[Wiki] `http://es.wikipedia.org/wiki/Ingenier%C3%ADa`

[Wil06] Wills, Rebecca S. *Google's PageRank: the math behind the search engine.* Math. Intelligencer 28 (2006), no. 4, 6–11.

[Wil07] Wills, Rebecca S. *When rank trumps precision: Using the power method to compute Google's PageRank.* Thesis (Ph.D.) – North Carolina State University. 2007. 110 pp. ISBN: 978-0549-19626-6

Universitat de Barcelona, Departament d'Àlgebra i Geometria. Gran Via 585, 08007 Barcelona, Spain

E-mail address: `cdandrea@ub.edu`

URL: `http://atlas.mat.ub.es/personals/dandrea/`