Métodos Numéricos
1^o Cuatrimestre 2018
Trabajo Práctico 1: PageRank

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Sistemas de ecuaciones lineales

Introducción

El motor del buscador de Google, desde hace más de 20 años, utiliza el denominado *ranking de Page* o *PageRank* [1] como uno de los criterios para ponderar la importancia de los resultados de cada búsqueda. Calcular este ranking requiere simplemente resolver un sistema de ecuaciones lineales donde la cantidad de ecuaciones e incógnitas del sistema es igual al número de páginas consideradas. ¿Simplemente?

Modelado del problema

Para un determinado conjunto de n páginas web definamos la *matriz de conectividad* $\mathbf{W}$ poniendo $w_{ij} = 1$ si la página j tiene un link a la página i y $w_{ij} = 0$ si no. Además $w_{ii} = 0$ pues ignoramos los *autolinks*. De esta forma, la matriz $\mathbf{W}$ puede resultar extremadamente rala y muy grande de acuerdo al tamaño del conjunto.

Para cada página $j = 1 \ldots n$, definimos su *grado* como:

$$c_j = \sum_{i=1}^{n} w_{ij} \tag{1}$$

Es decir, la cantidad de links *salientes* de j. Típicamente c_j es un número mucho menor que n.

Se busca que el ranking sea mayor en las páginas *importantes*. Heurísticamente, una página es importante cuando recibe muchos “votos” de otras páginas, es decir, links. Pero no todos los links pesan igual: los links de páginas más importantes valen más. Pocos links de páginas importantes pueden valer más que muchos links de páginas poco importantes. Y los links de páginas con muchos links valen poco. Por lo tanto, una forma de calcular la importancia o *puntaje* x_i de la página i es:

$$x_i = \sum_{j=1}^{n} \frac{x_j}{c_j} w_{ij} \tag{2}$$

Es decir, la página j le aporta a i su puntaje ponderado por cuántos links salientes tiene. También, se puede definir la matriz de puntajes $\mathbf{R} = \mathbf{WD}$, donde $\mathbf{D}$ es una matriz diagonal con elementos d_{jj} de la forma:

$$d_{jj} = \begin{cases} 1/c_j & \text{si} \quad c_j \neq 0 \\ 0 & \text{si} \quad c_j = 0 \end{cases},$$

Lo cual nos permite calcular el ranking de todas las páginas como:

$$\mathbf{R}\,\mathbf{x} = \mathbf{x} \tag{3}$$

donde $\mathbf{x} = (x_1, \ldots, x_i, \ldots, x_n)^T$. Luego, la ecuación 2 corresponde al elemento i-ésimo $(\mathbf{R}\,\mathbf{x})_i$.

Este modelo tiene un problema y es que no logra capturar el comportamiento errático del usuario mientras surfea la red redes.

[1] Por Larry Page, uno de los fundadores de Google, otrora joven científico actualmente devenido multimillonario. Ver artículo original del 1998 con más de 16500 citas [1]

Modelo del navegante aleatorio

Un enfoque alternativo es considerar el modelo del *navegante aleatorio*. El navegante aleatorio empieza en una página cualquiera del conjunto, y luego en cada página j que visita elige con probabilidad $p \in (0,1)$ si va a seguir uno de sus links, o con probabilidad $1-p$, si va a pasar a otra página cualquiera del conjunto. Una vez tomada esa decisión, si decidió seguir un link de la página j elige uno al azar con probabilidad $1/c_j$, mientras que si decidió pasar a otra página cualquiera entonces elige una al azar con probabilidad $1/n$). Cuando la página j no tiene links salientes, es decir $c_j = 0$, elige al azar una página cualquiera del conjunto. Por lo tanto, se espera que luego de mucho surfear el navegante aleatorio va a estar en páginas importantes con mayor probabilidad.

Formalmente, la probabilidad de pasar de la página j a la página i es:

$$a_{ij} = \begin{cases} (1-p)/n + (p\,w_{ij})/c_j & \text{si} \quad c_j \neq 0 \\ 1/n & \text{si} \quad c_j = 0 \end{cases}, \tag{4}$$

y sea $\mathbf{A} \in \mathbb{R}^{n\times n}$ a la matriz de elementos a_{ij}. Entonces el *ranking de Page* es la solución del sistema:

$$\mathbf{A}\,\mathbf{x} = \mathbf{x} \tag{5}$$

que cumple $x_i \geq 0$ y $\sum_i x_i = 1$.

Por lo tanto, el elemento i-ésimo $(\mathbf{Ax})_i$ es la probabilidad de encontrar al navegante aleatorio en la página i sabiendo que x_j es la probabilidad de encontrarlo en la página j, para $j = 1 \ldots n$;

Luego, la matriz $\mathbf{A}$ puede reescribirse como:

$$\mathbf{A} = p\,\mathbf{W}\,\mathbf{D} + \mathbf{e}\,\mathbf{z}^T,$$

donde $\mathbf{D}$ es una matriz diagonal de la forma

$$d_{jj} = \begin{cases} 1/c_j & \text{si} \quad c_j \neq 0 \\ 0 & \text{si} \quad c_j = 0 \end{cases},$$

$\mathbf{e}$ es un vector columna de unos de dimensión n y $\mathbf{z}$ es un vector columna cuyos componentes son:

$$z_j = \begin{cases} (1-p)/n & \text{si} \quad c_j \neq 0 \\ 1/n & \text{si} \quad c_j = 0 \end{cases}.$$

Así, la ecuación (5) puede reescribirse como

$$(\mathbf{I} - p\,\mathbf{W}\,\mathbf{D})\,\mathbf{x} = \gamma\,\mathbf{e}, \tag{6}$$

donde $\gamma = \mathbf{z}^T\,\mathbf{x}$ funciona como un factor de escala.

Cálculo del ranking de Page

De esta manera, un procedimiento para calcular el ranking de Page consiste en:

1. Suponer $\gamma = 1$.
2. Resolver el sistema lineal de la ecuación (6).
3. Normalizar el vector $\mathbf{x}$ de manera que $\sum_i x_i = 1$.

Enunciado

Se debe implementar un programa en `C` o `C++` que realice el cálculo del ranking de Page según el procedimiento descripto anteriormente.

Como parte **obligatoria** se pide implementar lo siguiente:

1. El método de Eliminación Gaussiana (EG)
2. Una estructura de matrices ralas que sea eficiente en espacio y en tiempo para la tarea que se busca realizar.
3. Herramientas de entrada/salida para la lectura de archivos con los datos de los conjuntos de páginas y escritura del ranking de Page. Se debe respetar el formato que se describe más abajo.

Previamente, **deberán** estudiar las características de la matriz involucrada y responder a lo siguiente:

1. ¿Por qué la matriz $\mathbf{A}$ definida en (4) es equivalente a $p\,\mathbf{W}\,\mathbf{D} + \mathbf{e}\,\mathbf{z}^T$? Justificar adecuadamente.
2. ¿Cómo se garantiza la aplicabilidad de EG? ¿ La matriz $(\mathbf{I} - p\,\mathbf{W}\,\mathbf{D})$ está bien condicionada? ¿Cómo influye el valor de p ?

Además, **se pide** realizar un informe utilizando como guía las pautas de laboratorio de la materia conteniendo la experimentación pedida en la siguiente sección.

Es importante incluir en el informe del trabajo práctico, en la sección desarrollo, aquellas decisiones tomadas en función de las estructuras de datos utilizadas y las alternativas consideradas y descartadas para los métodos utilizados.

Experimentación

Se deberá realizar tanto un análisis cualitativo como cuantitativo de los métodos vistos en el trabajo.

Para el análisis cuantitativo, se pide, como mínimo, estudiar los tiempos de procesamiento en función del tamaño del grafo de páginas y de la densidad del mismo.

Para el análisis cualitativo se deberán estudiar los rankings obtenidos, en función de la estructura del grafo, y del valor de p.

Para esto, el grupo **deberá** proponer al menos 3 instancias de prueba no triviales, que crean pertinentes para analizar el método (entregando además los archivos correspondientes).

Para el análisis, guiarse y responder las siguientes preguntas:

1. ¿Cómo es el ranking obtenido en cada caso de acuerdo a la estructura del grafo páginas?
2. ¿Qué conclusiones pueden sacar de la interpretación de los resultados?
3. Respecto del ranking de Page: ¿Funciona cómo era esperado? ¿Hubo sorpresas? ¿Qué pueden concluir sobre su significado? ¿Dirían que es un buen ranking?
4. ¿Cómo es la calidad de los rankings?

Datos de entrada/salida

Los archivos de entrada y salida serán de texto plano, y deberán respetar el siguiente formato:

archivo entrada: En la primera línea un entero N, la cantidad total de páginas. En la segunda línea un entero M, la cantidad total de links. Luego siguen M líneas, cada una con dos enteros i j separados por un espacio ($1 \leq i, j \leq N$), indicando que hay un link de la página i a la página j.

archivo de salida: La primera línea deberá contener el valor de p utilizado. Luego, deberán seguir N líneas, donde la línea i contiene el valor del ranking de Page para la página i.

El programa ejecutable deberá cumplir con el siguiente formato de uso:

```
./tp1 archivo p
```

Donde `archivo` es la ubicación del archivo de entrada a leer, y `p` es el valor de p a usar. El archivo de salida deberá ser escrito en la ubicación `archivo.out`.

La cátedra provee además un conjunto de tests con archivos de entrada y salida esperada, los cuales deberán funcionar correctamente en sus implementaciones para aprobar el trabajo.

Fechas de entrega

- *Formato Electrónico:* hasta el Domingo 15 de Abril de 2018, a las 23:59 hs, enviando el trabajo (informe + código) a la dirección `metnum.lab@gmail.com`. El subject del email debe comenzar con el texto `[TP1]` seguido de la lista de apellidos de los integrantes del grupo separados por punto y coma `;`.
- Formato físico: Lunes 16 de Abril de 2018, 17 hs. en la clase de laboratorio.
- Pautas de laboratorio: `http://www-2.dc.uba.ar/materias/metnum/homepage.html`

Importante: El horario es estricto. Los correos recibidos después de la hora indicada serán considerados re-entrega.

Referencias

[1] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.