# Prestaciones numéricas del método de la potencia

Los **objetivos de esta práctica** son:
1. Determinar las prestaciones numéricas del método de la potencia: tiempo de cpu y tamaño de memoria necesaria para calcular el autovalor/autovector dominantes de una matriz A, con N=dim(A).
2. Calcular la capacidad de nuestro equipo: tamaño máximo de la matriz A, para la cual podemos resolver el problema del autovalor/autovector. Extrapolar los datos obtenidos a valores mayores de N.

Para medir las prestaciones numéricas y la capacidad del equipo utilizamos 3 tipos de matrices:
- Matrices **completas y densas**.
- Matrices **completas y dispersas**.
- Matrices **dispersas** (definidas con el comando *sparse* de Matlab, que almacena en memoria únicamente los elementos no nulos).

Una matriz es dispersa o rala cuando es de gran tamaño y la mayor parte de sus elementos son cero. Por el contrario, una matriz es densa cuando casi todos sus elementos son no nulos.

Para cada tipo de matriz, vamos a obtener las siguientes tablas, en función de la dimensión de la matriz N=dim(A).

1. Con el método de la potencia, para N=10^1, 10^2, 10^3,… completamos la siguiente tabla, hasta la capacidad de nuestro equipo:

| | Tiempo [seg] | Memoria [MB/GB] | Nº iteraciones | Precisión $\|Ax-x\|$ | Precisión $\lvert\lambda-1\rvert$ |
|---|---|---|---|---|---|
| N=10^1 | | | | | |
| N=10^2 | | | | | |
| N=10^3 | | | | | |
| N=10^4 | | | | | |
| …. | | | | | |

**Tabla 1: Prestaciones del método de la potencia**

2. Extrapolamos los datos obtenidos para completar la siguiente tabla:

| | Tiempo estimado [horas/años/siglos] | Memoria estimada [GB/TB] |
|---|---|---|
| N=10^5 | | |
| N=10^6 | | |
| … | | |
| N=10^10 | | |

**Tabla 2: Estimación de tiempo y memoria**

## A. Rutinas auxiliares

### 1. Método de la potencia (potencia)

Utilizamos la rutina de la práctica anterior ***[lambda,x,iter]=potencia(A,tol,n_iter,x0)*** que calcula el autovalor/auvector dominantes, verificando que la suma de todos los elementos del vector x es 1.

### 2. Matrices completas y densas (mcden)

A partir de la rutina de la práctica anterior (reescalar una matriz) codificar la rutina

***S=mcden(N)***

Tiene como variable de entrada la dimensión N, y como variable de salida una matriz S de tamaño NxN, definida de forma aleatoria, rand(N), y reescalada por columnas, de tal manera que sea estocástica por columnas. Comprobar que S verifica el teorema de Perron-Frobenius.

### 3. Matrices completas y dispersas (mcdis)

Codificar la rutina

***S=mcdis(N)***

Tiene como variable de entrada la dimensión N, y como variable de salida una matriz S de tamaño NxN.
Para calcular la matriz S realizar los siguientes pasos:

i) Calcular una matriz C de dimensión N=500 dispersa. Ejecutar los siguientes comandos:

```
clear
N=500; % N=dim(A)
R=5;  % R=número de links entrada/salida de cada nodo
p=randi(N,1,5*N);q=randi(N,1,5*N);  % Numeros enteros aleatorios
Cs=sparse(p,q,1,N,N);
C=full(Cs);
Nj=sum(C);
i=find(Nj==0)
nnz(C)
```

Si el vector de índices i que obtenéis es nulo, ejecutar de nuevo los comandos hasta que el vector de índices i tenga algún elemento.

Calcular el porcentaje de elementos no nulos de C: ¿la matriz C es dispersa?.
Calcular el espacio de memoria ocupado, con el comando *whos*.

ii) Consideramos a la matriz C una matriz de conectividad.
Con el procedimiento de la práctica anterior, calcular la matriz S de pagerank.
Comprobar que S verifica el teorema de Perron-Frobenius.

### 4. Matrices dispersas (mdis)

Codificar la rutina

***S=mdis(N)***

Tiene como variable de entrada la dimensión N, y como variable de salida una matriz dispersa S de tamaño NxN. Para calcular la matriz S realizar los siguientes pasos:

i) Calcular una matriz C de dimensión N=500 dispersa. Ejecutar los siguientes comandos:

```
clear
N=500; % N=dim(A)
R=5;  % R=número de links entrada/salida de cada nodo
p=randi(N,1,5*N);q=randi(N,1,5*N);
C=sparse(p,q,1,N,N);
Nj=sum(C);
i=find(Nj==0)
```

Si el vector de índices i que obtenéis es nulo, ejecutar de nuevo los comandos hasta que el vector de índices i tenga algún elemento.

Calcular el espacio ocupado en memoria por la matriz C.

ii) Consideramos a la matriz C una matriz de conectividad.
Con el procedimiento de la práctica anterior, calcular la matriz S de pagerank.
Comprobar que S verifica el teorema de Perron-Frobenius.

## B. Prestaciones numéricas del método de la potencia

Codificar un script con el siguiente contenido:

```
clear
N=   ; % N=dim(A)
[S]=mcden(N)
tic    % Arranca el contador de tiempo
tol=1e-13   % Tolerancia
[lambda,x,iter]=potencia(A,tol,n_iter,x0)
t1=toc    % Para el contador de tiempo y lo almcena en t1
precision1=  % Calcular la precisión 1 obtenida
precision1=  % Calcular la precisión 1 obtenida
whos  % Calcular la memoria utilizada
```

Ejecutar el script anterior para N=10^1, 10^2, 10^3,… hasta la capacidad de vuestro sistema. También podeis utilizar valores intermedios como N=5*10^4.

- **Completar la Tabla 1 para matrices completas y densas.**

- "El tiempo de ejecución (t) y la dimensión de la matriz N=dim(A) verifican la relación

$$t = aN^2$$

con a una constante."

Ajustar los datos de la Tabla 1 a la relación $t = aN^2$ (ver transparencias, pagina 45).
**Completar la Tabla 2 para matrices completas y densas.**

Repetir el procedimiento anterior para obtener:

- **Tablas 1 y 2 para matrices completas y dispersas.**
- **Tablas 1 y 2 para matrices dispersas.**

Nota: En las Tablas 1 y 2 indicar las unidades de tiempo (seg/horas/…) y de tamaño de memoria (MB/GB/…)

## C. Conclusiones finales

1. Si queremos calcular el pagerank de una matriz muy grande, ¿cuál es la mayor limitación o dificultad numérica?.
2. Cuantificar la mejora computacional de utilizar matrices dispersas frente a matrices completas y dispersas.
3. ¿Es adecuada la siguiente afirmación?

"El tiempo de ejecución (t) y la dimensión de la matriz N=dim(A) verifican la relación $t = aN^2$."

Ajustar los datos con otras relaciones del tipo $t = a + bN + cN^2 + dN^3$ ¿Qué relación ajusta mejor los datos?. Justificar.

4. Estimar el coste computacional del método de la potencia: número de operaciones realizadas por iteración. ¿El resultado obtenido tiene alguna relación con el apartado anterior?.
5. Hacer cualquier observación que tenga relevancia numérica.