Calcular el pagerank de matrices de gran tamaño:

- **Stanford Web Matrix**
- **Google Web Matrix**
- **Berkely-Stanford Matrix**

Codificar una rutina para calcular el pagerank de la matriz G del modelo, a partir de la matriz A.

Calcular e interpretar el pagerank de la Standford Web Matrix (Web graph). 281903 nodos y 2.3 millones de links.

Calcular e interpretar el pagerank de la Google Web Matrix (Web graph). 916428 nodos y 5.1 millones de links).

Calcular e interpretar el pagerank de la Berkely-Standord Web (Web graph). 685230 nodos y 7600595 de links).

1. Codificar la rutina para calcular el pagerank de la matriz de pagerank G a partir de la matriz A

El objetivo es calcular el pagerank de la una matriz G de gran tamaño.
La memoria del ordenador no tiene capacidad para almacenar la matriz G.
Calculamos el pagerank de G utilizando como variable de entrada la matriz A.

Codificar la rutina:

[*pagerank*,ordenpagerank,precision,tiempo]=calculo_PR(A,alfa,niter)

Variables de entrada:

La matriz dispersa *A*, el parámetro alfa y el número de iteraciones *niter*.

Variables de salida:

pagerank es el pagerank de la matriz G, **ordenpagerank** el orden de las páginas según su pagerank, la **precisión** obtenida (precisión = $\|Gx - x\|_2$) y el **tiempo** utilizado.

Características:

A partir de la matriz A de entrada, el código calcula el pagerank de G, pero NO dispone de la matriz G explícitamente.

Se debe utilizar indexación lógica.

El código debe ser lo más eficiente y eficaz posible.

Proceso:

Ver la presentación de clase y el siguiente esquema.
A partir de la matriz A, calculamos los vectores Nj, dj y v.
Utilizar el código del método de la potencia.
Utilizar el parámetro $\alpha = 0.85$.

[pagerank,ordenpagerank,precision,tiempo]=calculo_PR(A,alfa,niter)

$$
\begin{cases}
e = ones(1,N); v = [\alpha\ dj + (1-\alpha)\ e] \\
k = 1 : niter \\
x = x / norm(x) \\
x = \alpha Ax + \dfrac{1}{N} e^T (vx) \\
end \\
x = x / sum(x); \\
\text{precision } = \|Gx - x\|_2 \quad \%\text{Ojo! ¿Como calcular Gx 'sin utilizar' G?}
\end{cases}
$$

Probar la rutina:

- Elegir un N suficientemente grande (N=1e+4, 5e+4, 7.5e+4, 1e+5,…), dependiente de la capacidad de vuestros ordenadores.
- Generar una matriz aleatoria C de dimensión N. Calcular las matrices A y G del modelo.
- Calcular el pagerank de G a partir de la matriz A con la rutina calculo_PR.
- Comprobar que el pagerank es el correcto.

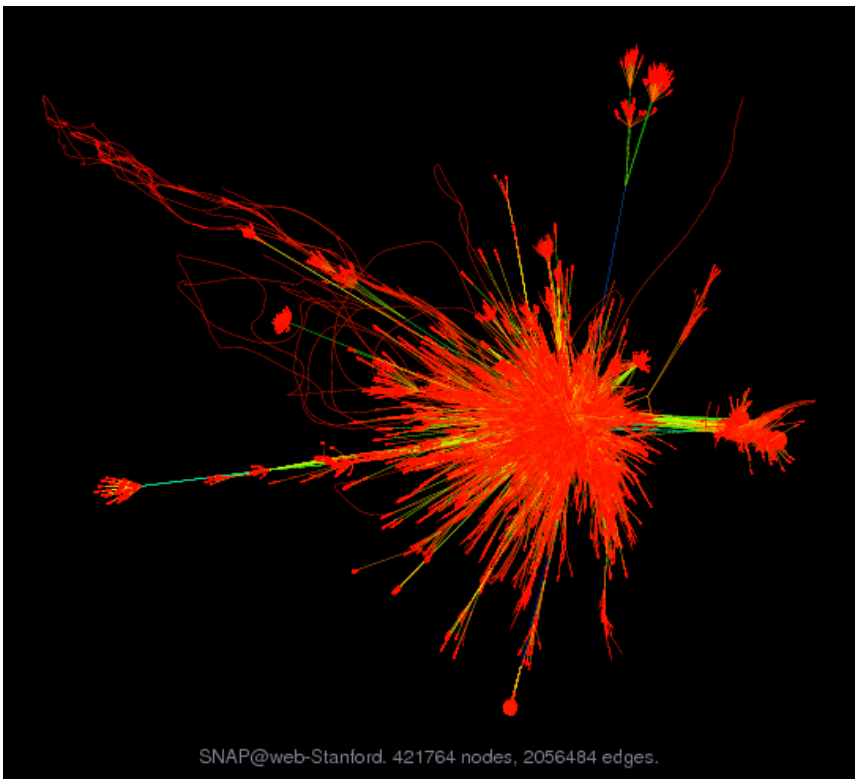
2. Calcular e interpretar el pagerank de Stanford Web Matrix

Calcular el **pagerank de Stanford Web Matrix** (281.903 nodos y 2.312.497 links). Visualizar y ordenar los nodos en función de su pagerank.

1. Buscar en Google: 'stanford web matrix'. Acceder a la página:

https://www.cise.ufl.edu/research/sparse/matrices/SNAP/web-Stanford.html

Descargar download as a MATLAB mat-file, file size: 6 MB.



También podéis acceder a las a las Web graphs en la siguiente dirección:

https://snap.stanford.edu/data/

Ejecutar los comandos (tarda unos minutos, que no se os olvide el ; para que no muestre el contenido de A por pantalla).

```
load web-Stanford.mat;
Problem
A=Problem.A;
whos
spy(A);title('Gráfica de la dispersión de la matriz A')
```

Hacer zoom en la figura para ver mejor el contenido.

a) Dar los comandos necesarios para conocer las **características de la matriz A**.

- ¿Es cuadrada?. ¿Cuántos nodos tiene la red?.
- Tipo de matriz (dispersa, completa).

- Tamaño en memoria de A.
- Número de elementos no nulos de A.
- Dar el comando:>> B=A-1; ¿Qué sucede?. ¿Por qué?.
- ¿Cuántos elementos de A son 1's? ¿Cuántos son 0's?. ¿Cuántos son distintos de 1's y de 0's?.
- Mostrar el contenido de A de las filas 1:1000 y las columnas 1:1000. ¿Cuántos elementos no nulos hay en esa submatriz 1000x1000?.
- ¿Qué nodo tiene el mayor número de links de salida? ¿Cuántos links de salida tiene?.
- La matriz A ¿qué tipo de matriz es de las descritas en las diapositivas (C, A, S o G) según el modelo?.
  ¿Es una matriz de conectividad C? ¿Por qué?.
  ¿Es una matriz de transición A? ¿Por qué?.
  ¿Es una matriz de transición modificada S? ¿Por qué?.
  ¿Es una matriz de Google G? ¿Por qué?.
- Calcular el índice de dispersión de A (número de elementos no nulos/número total de elementos).
- Calcular el número medio de links de salida: ¿cada página cuantos línks de salida de media tiene?.
- **Sin utilizar el número total de nodos**: ¿Cuántos nodos sin salida tiene la red?, ¿cuántos nodos con salida tiene la red?.
  Comprobar si se verifica la siguiente relación:

  número de nodos totales = número de nodos sin salida + número de nodos con salida.

- **Necesitáis la matriz A del modelo para utilizar como variable de entrada en el siguiente apartado.**

b) Calcular el pagerank de Stanford Web Matrix.

Utilizar la función [pagerarank,ordenpagerank,precision,tiempo]=calculo_PR(A,alfa,niter)*;*
para calcular el pagerank, de la Stanford Web Matrix. Utilizar un niter suficiente para obtener una precisión menor de 1e-12. Completar la siguiente tabla:

	Tiempo [seg]	Memoria [MB]	Nº iteraciones	Precisión $\|Gx - x\|_2$
N=281903				

c) Visualizar y analizar los resultados.

- Visualizar el pagerank obtenido con el comando *bar(pagerank)*.
  Dar el comando *fprintf* para extraer de la tabla los 20 nodos con los mayores pagerank.
  El resultado debe ser algo similar a:

```
Orden   1  Nodo  xxxx  Pagerank  0.yyyy
Orden   2  Nodo  xxxx  Pagerank  0.yyyy
Orden   3  Nodo  xxxx  Pagerank  0.yyyy
Orden   4  Nodo  xxxx  Pagerank  0.yyyy
Orden   5  Nodo  xxxx  Pagerank  0.yyyy
```

- Repetir el comando anterior para extraer de la tabla los 20 nodos con los **menores pagerank**.
  - ¿Tienen el mismo pagerank esos nodos? ¿Porqué?
  - ¿Cuántos nodos tienen el menor pagerank? ¿Cuáles son esos nodos?. Justificar.
    Iidentificar estos nodos y visualizarlos con el comando *fprintf*.
- ¿Hay varios nodos que tienen el mayor pagerank?.

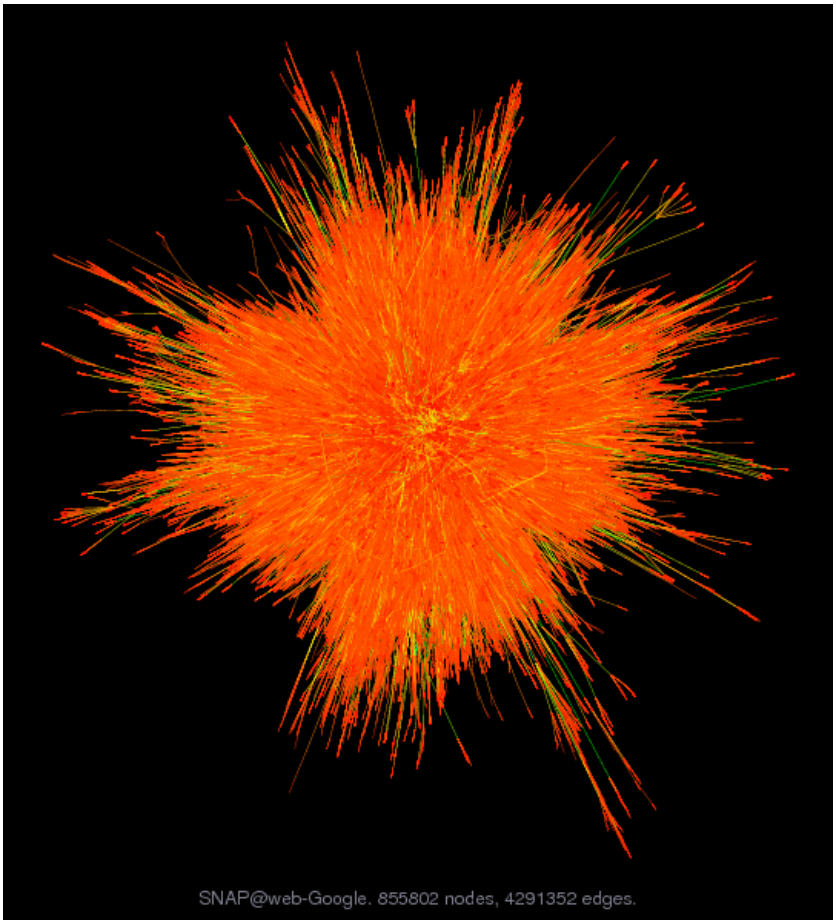
3. Calcular e interpretar el pagerank de Google **Web Matrix**

Calcular el **pagerank de Google Web Matrix** (875.713 nodos y 5105039 links). Visualizar y ordenar los nodos en función de su pagerank.

Acceder a las a las Web graphs en la siguiente dirección:

https://snap.stanford.edu/data/

Descargar download as a MATLAB mat-file, file size: 16 MB.



a) Dar todas las **características de la matriz A**.
b) Calcular el pagerank de Google Web Matrix.
c) Visualizar y analizar los resultados.

4. Calcular e interpretar el pagerank de Berkely-Stanford web grahp

Calcular el **pagerank de Berkely-Stanford web grahp** (685.230 nodos y 7.600.595 links). Visualizar y ordenar los nodos en función de su pagerank.

Acceder a las a las Web graphs en la siguiente dirección:

https://snap.stanford.edu/data/

Berkeley-Stanford web graph

Dataset information

Nodes represent pages from berkely.edu and stanford.edu domains and directed edges represent hyperlinks between them. The data was collected in 2002.

Dataset statistics	
Nodes	685230
Edges	7600595
Nodes in largest WCC	654782 (0.956)
Edges in largest WCC	7499425 (0.987)
Nodes in largest SCC	334857 (0.489)
Edges in largest SCC	4523232 (0.595)
Average clustering coefficient	0.5967
Number of triangles	64690980
Fraction of closed triangles	0.002746
Diameter (longest shortest path)	514
90-percentile effective diameter	9.9

```
Es un fichero de texto que contiene los índices de la matriz de
conectividad cuyo elemento es 1.

Los primeros elementos de la matriz C son los siguientes:

C(1,[2,5,7,8,9,11,17,254913,438238])=1,
C(254913,[ 255378, 255379, 255383, 255384])=1,

Etc.

# Directed graph (each unordered pair of nodes is saved once): web-
BerkStan.txt
# Berkely-Stanford web graph from 2002
# Nodes: 685230 Edges: 7600595
# FromNodeId      ToNodeId
1     2
1     5
1     7
1     8
1     9
1     11
1     17
1     254913
1     438238
```

```
254913      255378
254913      255379
254913      255383
254913      255384
254913      255392
254913      255393
254913      255394
254913      255396
```

a) Definir la matriz C de conectividad.
b) Calcular el pagerank de **Berkely-Stanford**.
c) Visualizar y analizar los resultados.