

# La Matematica dentro Google

F. Iavernaro, A. Pugliese

Dipartimento di Matematica, Università degli Studi di Bari Aldo Moro

`felice.iavernaro@uniba.it`    `alessandro.pugliese@uniba.it`

# Funzionamento di un motore di ricerca

- ▶ Come funziona un motore di ricerca?
  - ▶ scansione (web crawling, eg. Googlebot)
  - ▶ indicizzazione (inserimento nel database del motore di ricerca)
  - ▶ classificazione (posizionamento di una pagina web in termini della sua importanza)

Quando un'utente effettua una ricerca, viene selezionato un sottoinsieme di pagine web nel database che contengono l'informazione desiderata. I risultati vengono ordinati e presentati secondo la loro importanza (i primi risultati sono quelli più rilevanti).

- ▶ Come ordinare i risultati di un motore di ricerca dal più rilevante al meno rilevante? Come misurare l'importanza di una pagina web?

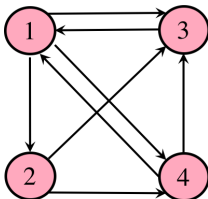
Un modo classico (ma non funzionante) di procedere è quello di ordinare le pagine in base alla frequenza con cui compaiono i termini della query.

## Il PageRank di Google (1/3)

- ▶ Sergey Brin e Larry Page creano Google nel 1998 a Stanford (California) e ideano un nuovo modo di classificare i documenti web in termini della loro rilevanza e popolarità: il PageRank.
- ▶ Il loro obiettivo era quello di aiutare gli utenti di internet a trovare più facilmente le informazioni.
- ▶ Al fine di identificare le pagine pertinenti con la query, Page e Brin ebbero l'idea di considerare non solo le ripetizioni delle parole ma anche i link che provenivano da altri siti e che puntavano ad una determinata pagina.
- ▶ Il loro ragionamento era semplice: se un certo sito viene citato e consigliato da molti altri significa che ha dei contenuti interessanti e quindi è giusto farlo vedere prima di altri.

## Il PageRank di Google (2/3)

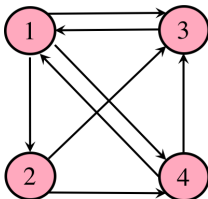
Supponiamo che, a seguito di una query, siano state identificate le quattro pagine visualizzate in figura. Le frecce indicano i collegamenti (link) da una pagina all'altra.



Quale sarà la pagina più importante?

## Il PageRank di Google (2/3)

Supponiamo che, a seguito di una query, siano state identificate le quattro pagine visualizzate in figura. Le frecce indicano i collegamenti (link) da una pagina all'altra.



Quale sarà la pagina più importante?

Non è la pagina 3

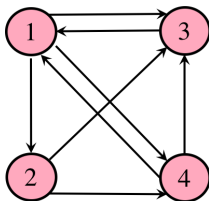
## Il PageRank di Google (3/3)

- ▶ Per il calcolo delPageRank di una pagina web  $P$  si tiene conto:
  - ▶ del numero di link che  $P$  riceve dalle altre pagine del web;
  - ▶ della qualità di ciascun link:
    - ▶ un link da una pagina importante darà un contributo maggiore rispetto a un link da una pagina più sconosciuta;
    - ▶ un link da una pagina da cui partono molti link dovrà pesare meno di un link da una pagina che ne contiene pochi.

### Idea fondamentale:

Un link alla pagina  $P$  dalla pagina  $Q$  aumenta il PageRank della pagina  $P$  in misura direttamente proporzionale al PageRank della pagina  $Q$  ed inversamente proporzionale al numero di link che dalla pagina  $Q$  rimandano ad altre pagine

## Esempio (1/3)



Consideriamo il microweb rappresentato dal grafo orientato in figura e costituito dalle quattro pagine  $P_1$ ,  $P_2$ ,  $P_3$ , e  $P_4$ .

Denotiamo con  $x_1$ ,  $x_2$ ,  $x_3$  e  $x_4$  il loro Pagerank.

- ▶ Alla pagina  $P_1$  giungono due link: uno dalla pagina  $P_3$  (che contiene un solo link) e l'altro dalla pagina  $P_4$  (che contiene due link). Pertanto:  $x_1 = x_3 + \frac{1}{2}x_4$ .
- ▶ Ragionando analogamente, per la pagina  $P_2$  risulta:  $x_2 = \frac{1}{3}x_1$ .
- ▶ Per la pagina  $P_3$  risulta:  $x_3 = \frac{1}{3}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_4$ .
- ▶ Infine, per la pagina  $P_4$  risulta:  $x_4 = \frac{1}{3}x_1 + \frac{1}{2}x_2$ .

## Esempio (2/3)

Abbiamo ottenuto il sistema lineare

$$\begin{cases} x_3 + \frac{1}{2}x_4 = x_1 \\ \frac{1}{3}x_1 = x_2 \\ \frac{1}{3}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_4 = x_3 \\ \frac{1}{3}x_1 + \frac{1}{2}x_2 = x_4 \end{cases}$$

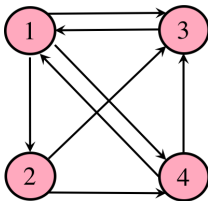
- ▶ Si vede facilmente che, se  $(x_1, x_2, x_3, x_4)$  è una sua soluzione, lo sono anche  $(\alpha x_1, \alpha x_2, \alpha x_3, \alpha x_4)$  al variare di  $\alpha$  parametro reale (non sorprende: la classifica è invariante rispetto alla moltiplicazione di tutti i punteggi per uno stesso fattore)
- ▶ Per convenzione si sceglie la quadrupla che soddisfa

$$x_1 + x_2 + x_3 + x_4 = 1$$

(si chiama normalizzazione). In questa maniera, il punteggio di ogni pagina pu essere interpretato come la sua importanza in termini percentuali rispetto alle altre pagine.

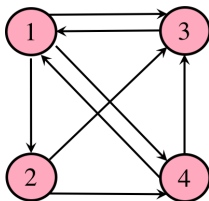


## Esempio (3/3)



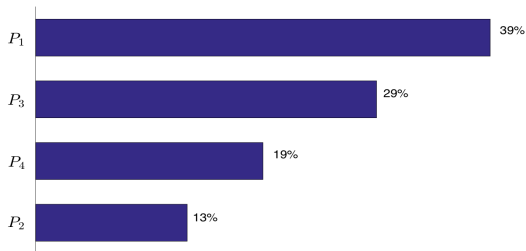
- Qual è la pagina web più importante?

## Esempio (3/3)



- Qual è la pagina web più importante?
- La soluzione (normalizzata) del sistema è:

$$x_1 = \frac{12}{31} \approx 0.39, \quad x_2 = \frac{4}{31} \approx 0.13, \quad x_3 = \frac{9}{31} \approx 0.29, \quad x_4 = \frac{6}{31} \approx 0.19.$$



# Definizione del PageRank

- ▶ Consideriamo un insieme di  $n$  pagine web:  $P_1, P_2, \dots, P_n$
- ▶ Indichiamo con  $l_j$  il numero di link che dalla pagina  $P_j$  puntano verso altre pagine
- ▶ Indichiamo con  $B_i$  l'insieme delle pagine che contengono un link alla pagina  $P_i$
- ▶ L'importanza  $x_i$  della pagina  $P_i$  sarà la somma dei contributi di tutte le pagine che hanno un link ad essa

Formalmente:

$$x_i = \sum_{P_j \in B_i} \frac{x_j}{l_j}, \quad i = 1, \dots, n. \quad (1)$$

La (1) equivale a un sistema lineare di  $n$  equazioni in  $n$  incognite.

## Matrice di Google

Riprendiamo il sistema dell'esempio precedente:

$$\begin{cases} x_3 + \frac{1}{2}x_4 = x_1 \\ \frac{1}{3}x_1 = x_2 \\ \frac{1}{3}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_4 = x_3 \\ \frac{1}{3}x_1 + \frac{1}{2}x_2 = x_4 \end{cases}$$

Il sistema lineare può essere scritto utilizzando il *formalismo delle matrici*. I coefficienti che compaiono ai membri sinistri delle 4 equazioni possono essere rappresentati mediante la matrice di 4 righe e 4 colonne:

$$G = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Il sistema lineare diventa  $Gx = x$ .  $G$  è detta *matrice di Google*.

## Come costruire $G$ (1/2)

Si parte dalla *matrice di adiacenza*  $A = (a_{ij})$ :

$$a_{ij} = \begin{cases} 1, & \text{se c'è un link alla pagina } P_i \text{ dalla pagina } P_j \\ 0, & \text{altrimenti} \end{cases}$$

Esempio precedente:

$$A = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

- ▶ Ogni riga ed ogni colonna corrispondono ad una pagina web
- ▶ Somma lungo la riga  $i$ : numero di link alla pagina  $P_i$  da altre pagine
- ▶ Somma lungo la colonna  $j$ : numero di link dalla pagina  $P_j$  verso altre pagine
- ▶  $l_j$  è la somma degli elementi lungo la colonna  $j$

## Come costruire $G$ (2/2)

$$A = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \longrightarrow G = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Formalmente:

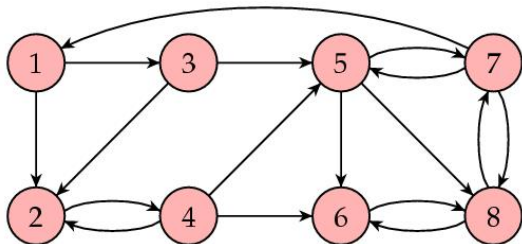
$$g_{ij} = \frac{a_{ij}}{l_j}, \text{ per ogni } i \text{ e } j$$

- ▶ Gli elementi di  $A$  valgono 0 oppure 1
- ▶ Gli elementi di  $G$  sono numeri positivi compresi tra 0 e 1
- ▶ La somma degli elementi di ciascuna colonna di  $G$  è uguale a 1



$G$  ammette autovalore  $\lambda = 1$

## Esercizio



Costruire la matrice di adiacenza  $A$  e la matrice di Google  $G$  per questo insieme di pagine web.

## Considerazioni sul costo computazionale

- ▶ Al momento Google indicizza circa  $30 \times 10^{12}$  pagine web. La matrice  $G$  ha quindi dimensioni enormi.
- ▶ Per risolvere  $Gx = x$  potremmo usare un computer. Algoritmi classici per risolvere i sistemi lineari, quali il metodo di eliminazione di Gauss, diventano molto onerosi al crescere delle dimensioni del sistema:

circa  $n^3$  operazioni algebriche elementari  
se la matrice  $G$  ha dimensione  $n$

- ▶ Il più veloce supercomputer al mondo può eseguire circa  $93 \times 10^{15}$  operazioni al secondo
- ▶ Quanto tempo occorrerebbe per risolvere il sistema? Troppo!

Per fortuna c'è un metodo più efficiente per ottenere il PageRank. Si sfrutta la sparsità della matrice di Google (la maggior parte degli elementi di  $G$  sono nulli).



## Il metodo delle potenze

- ▶ Si considera un vettore  $x_0$  arbitrario con l'unica condizione che la somma dei suoi elementi sia 1.
- ▶ Si genera la successione dei vettori

$$x_1 = Gx_0$$

$$x_2 = Gx_1$$

$$\vdots$$

$$x_{k+1} = Gx_k$$

$$\vdots$$

**N.B.** Così come  $x_0$ , ogni  $x_k$  ha elementi che sommano a 1.

## Il metodo delle potenze

- ▶ Si considera un vettore  $x_0$  arbitrario con l'unica condizione che la somma dei suoi elementi sia 1.
- ▶ Si genera la successione dei vettori

$$x_1 = Gx_0$$

$$x_2 = Gx_1$$

$$\vdots$$

$$x_{k+1} = Gx_k$$

$$\vdots$$

**N.B.** Così come  $x_0$ , ogni  $x_k$  ha elementi che sommano a 1.

- ▶ Assumiamo che
  - (a)  $G$  non possiede colonne nulle (non ci sono pagine che non contengono alcun link);
  - (b) il grafo che rappresenta il web è fortemente connesso (si può raggiungere un qualsiasi nodo a partire da un qualsiasi altro nodo);

Allora,  $\lambda = 1$  autovalore dominante di  $G$  e, di conseguenza, la successione  $\{x_k\}$  converge alla soluzione del sistema lineare  $Gx = x$  cioè al vettore dei PageRank.

## Modifica della matrice di Google

Vediamo come ovviare ai casi (frequenti nel web) in cui una delle due condizioni non è soddisfatta:

- (a) **Dangling nodes** (pagine che non contengono link)



Ogni colonna nulla di  $G$  viene sostituita dalla colonna di valori  $\frac{1}{n}$ . È come se la pagina contenesse i link verso tutte le altre pagine del web.

- (b) Denotiamo con  $E$  la matrice  $n \times n$  i cui elementi sono tutti uguali ad 1, e modifichiamo la matrice di Google come segue:

$$G \leftarrow (1 - m)G + m\frac{1}{n}E, \quad (\text{solitamente si pone } m = 0.15)$$

Osserviamo che le colonne della nuova matrice  $G$  così ottenuta, continuano ad avere somma pari a 1.

# Programma Python per il calcolo del PageRank

```
def pagerank(A,m=0.15):
    """
    calcola il PageRank di un web
    Dati di input
        A: matrice di adiacenza
        m: parametro di compensazione della matrice di Google
    Dati di output
        x: vettore dei PageRank
    """
    A=double(copy(A))
    [r,n]=shape(A)
    print(A)
    for j in range(0,n):
        #controllo la presenza di colonne nulle
        if sum(A[:,j])==0:
            A[:,j]=ones(shape=(n,1)); # A[:,j] indica la j-esima colonna di A
            A[:,j]=A[:,j]/sum(A[:,j]);
    print(A)
    E=ones(shape=(n,n)); # matrice di dimensione n con elementi uguali a 1
    G=(1-m)*A+m/n*E; # compensazione della matrice di Google
    #vettore iniziale della successione generata dal metodo delle potenze
    x0=ones(shape=(n,1))/n
    x=potenze(G,x0) #metodo delle potenze implementato a parte
    return x
```

## Risoluzione del primo esempio in Python

```
>>> A=array([[0,0,1,1],[1,0,0,0],[1,1,0,1],[1,1,0,0]])
```

```
>>> print(A)
```

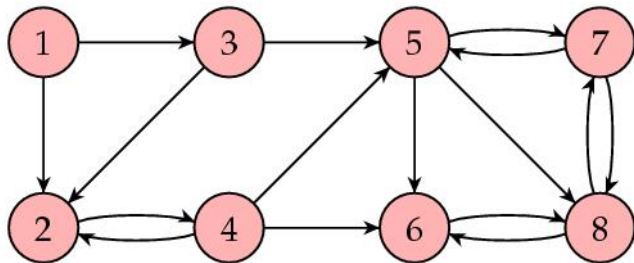
```
[[0 0 1 1]
 [1 0 0 0]
 [1 1 0 1]
 [1 1 0 0]]
```

```
>>> pagerank(A,0)
```

```
array([[ 0.38709677],
       [ 0.12903226],
       [ 0.29032258],
       [ 0.19354839]])
```

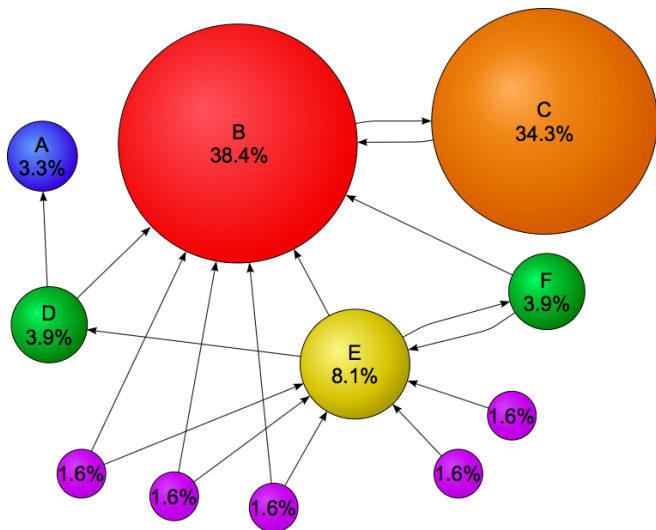
## Esercizio

Calcolare il vettore dei PageRank, prima e dopo l'ultima modifica sulla matrice di Google:



## Esercizio (da Wikipedia)

Calcolare il vettore dei PageRank della seguente sezione del web.



## Curiosità

- ▶ Poco dopo aver fondato l'azienda, per mancanza dei fondi necessari, Larry page e Sergey Brin cercarono di venderla per un milione di dollari a diverse società finanziarie, oltre che a diretti concorrenti come Altavista e Yahoo, ma senza successo.
- ▶ Ora è una delle più grandi aziende a livello globale con capitalizzazione azionaria superiore ai 500 miliardi di dollari.
- ▶ Il nuovo verbo “to google” significa cercare sulla rete.
- ▶ Il motto dei fondatori di google è “don't Be Evil” (non compiere mai il male).
- ▶ Il termine Google deriva da googol con cui è indicato il numero  $10^{100}$  (10 seguito da 100 zeri).
- ▶ Il quartier generale di Google in California si chiama googleplex il cui termine identifica (oltre che lo stabile di Google) il numero  $10^{\text{googl}}$ .



- ▶ AMS Feature Article: How Google Finds Your Needle in the Web's Haystack
- ▶ Wikipedia: PageRank
- ▶ Wikipedia: Google Matrix
- ▶ SIAM Review Article: The \$25,000,000,000 Eigenvector, the Linear Algebra Behind Google