Università degli studi di Cagliari
FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

Corso di Laurea Magistrale in Matematica

L'ALGORITMO DI PAGERANK

Relatore:

Dott. Lucio Cadeddu

Tesi di laurea di:

Veronica Cugusi

Anno accademico 2011/2012

Indice

Introduzione

Capitolo 1

Matematica alla base dell'algoritmo di PageRank

1.1 Cenni alla teoria dei grafi e il grafo del web

1.2 Il metodo delle potenze (The power method)

1.3 Concetti fondamentali sulle catene di Markov

Capitolo 2

Nascita dell'algoritmo di PageRank

2.1 Storia dell'algoritmo PageRank

2.2 Information Retrieval e gli algoritmi di Link Analysis Ranking (Algoritmi di analisi dei link)

Capitolo 3

Trattazione matematica di PageRank

3.1 Formula originale di PageRank

3.2 Le prime modifiche apportate al modello

3.3 Calcolo del vettore PageRank

3.4 Analisi di sensibilità dell'algoritmo

3.5 Codice Matlab per il calcolo del PageRank di un sito web

Introduzione

In questa tesi tratteremo l'algoritmo di PageRank, un metodo per il calcolo di un ranking (punteggio) delle pagine web, basato sul grafo del web, proposto da Larry Page e Sergey Brin, con l'obiettivo di misurare la loro importanza relativa. La ricerca di informazioni sul web è diventata un operazione quotidiana per gran parte delle persone, per quasi ogni esigenza. Con l'utilizzo di un motore di ricerca è possibile ottenere una lunga lista di documenti contenenti i dati inseriti nella ricerca. Molto spesso il numero dei dati da trattare risulta essere molto elevato. È quindi particolarmente importante cercare di ordinare tali risultati secondo un qualche criterio di importanza. Il noto motore di ricerca Google, ad esempio, attraverso l'utilizzo dell'algoritmo di PageRank riesce oggi a fornire risultati altamente pertinenti riguardo la misurazione dell'oggettiva importanza di ogni singola pagina web presente nel suo database, grazie alla formula

$$\boldsymbol{\pi}^T = \boldsymbol{\pi}^T(\alpha \mathrm{S} + (I - \alpha)\mathrm{E})$$

Nel primo capitolo sono introdotte la teoria dei grafi e le catene di Markov, che costituiscono la base matematica dell'algoritmo di PageRank, studiato approfonditamente nel terzo capitolo. Viene anche sviluppato un metodo iterativo di analisi numerica, il cosiddetto power method, metodo delle potenze, che permette di calcolare il vettore PageRank. Il secondo capitolo è focalizzato sulla storia del PageRank e l'introduzione degli algoritmi di link analysis ranking nella branca dell'informatica chiamata Information retrieval (recupero dell'informazione). Sono citati i precursori del metodo PageRank, cioè gli algoritmi Indegree e HITS. Il terzo capitolo, infine, è interamente dedicato alla trattazione matematica dell'algoritmo. Dopo una breve introduzione, viene affrontato il processo di determinazione della matrice delle probabilità di transizione ed il conseguente calcolo del vettore PageRank, effettuato adottando il metodo delle potenze (power method). Il capitolo si conclude prendendo in considerazione il sito web della Regione Sardegna e dell'Università di Cagliari e calcolando, tramite codice Matlab, il vettore PageRank della porzione di rete scaricata.

Capitolo 1

Matematica alla base dell'Algoritmo di Pagerank

1.1 Cenni alla teoria dei grafi e il grafo del web

I grafi sono strutture matematiche discrete che rivestono interesse sia per la matematica, sia per un ampia gamma di campi applicativi, in quanto permettono di schematizzare una grande varietà di situazioni e di processi. Un grafo è un insieme di elementi, detti nodi, collegati fra loro da archi. L'origine storica della teoria è generalmente fatta risalire ad un lavoro sviluppato da Eulero nel 1736, in cui veniva data una risposta ad un famoso quesito matematico noto come "problema dei ponti di Königsberg". Königsberg era una città della Prussia attraversata dal fiume Pregel. Un quartiere sorgeva su di un isola oltre la quale il fiume si spezzava in due rami, i quali erano muniti di sette ponti. Si chiedeva se fosse possibile costruire un percorso, in modo da transitare attraverso ciascun ponte una e una sola volta. La risposta negativa la fornì Eulero, dimostrando che tale problema non ammetteva soluzione sostituendo ogni riva del fiume e ogni isola con un nodo ed ogni ponte con un arco, trasformando il problema nello studio del grafo così costruito.

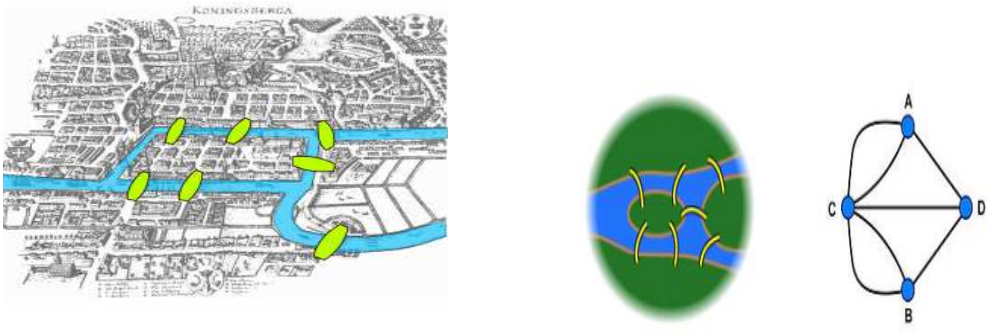

Figura 1.1 Problema di Königsberg e sua relativa schematizzazione tramite un grafo

Definizione 1.1.1. *Si dice grafo una coppia* $G = (V, E)$ *di insiemi, dove* $V = \{v_1, \dots, v_n\}$ *è un insieme finito di elementi detti nodi o vertici, mentre* $E = \{e_1, \dots, e_m\} \subseteq V \times V$ *è un sottoinsieme di coppie di nodi detti archi o spigoli.*

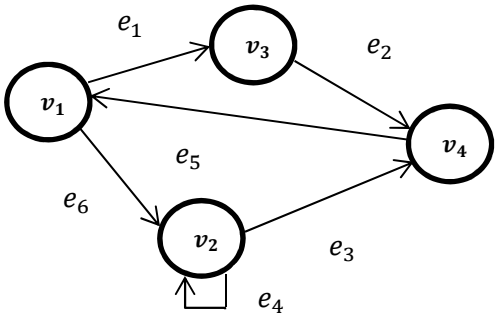

Figura 1.2 Grafo con $V = \{v_1, v_2, v_3, v_4\}$ e $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$

I nodi sono rappresentati tramite cerchi, mentre gli archi sono frecce che partono dal primo nodo della coppia e terminano nel secondo nodo. Due spigoli sono adiacenti se hanno un nodo in comune, mentre due nodi v_i e v_k, $i \neq k$, sono adiacenti se esiste lo spigolo (v_i, v_k).

Definizione 1.1.2. *Se ogni arco è una coppia ordinata di vertici, $(v_i, v_k) \neq (v_k, v_i)$, il grafo si dice orientato o diretto, cioè l'arco (v_i, v_k) stabilisce un collegamento tra il nodo v_i e il nodo v_k, ma non viceversa. I grafi privi di loop (archi di tipo (v_i, v_k) , con $i = k$) e di archi paralleli (coppie di archi uguali) si dicono grafi semplici.*

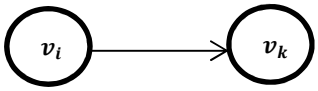

Figura 1.3 Arco orientato

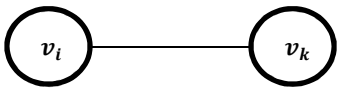

Figura 1.4 Arco non orientato

Definizione 1.1.3. *Un grafo si dice connesso se esiste una sequenza di archi che collega ciascuna coppia di nodi.*

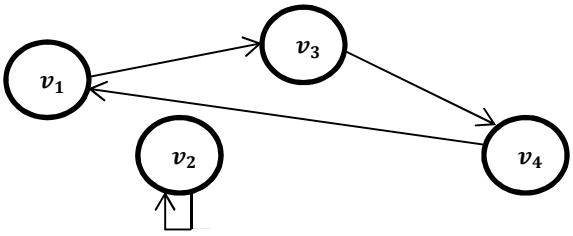

Figura 1.5. Grafo non connesso

Definizione 1.1.4. *Sia $G = (V, E)$ un grafo orientato. Si definisce cammino diretto la sequenza di vertici $(v_0, v_1, \dots, v_k)$ tali che $(v_i, v_{i+1}) \in E, \forall i = 0,1, \dots, k-1$. Il valore k si dice lunghezza del cammino. Se nessun vertice, esclusi v_0 e v_k , è ripetuto all'interno del cammino, il cammino si dice semplice. Se $v_0 = v_k$ il cammino si dice chiuso. Un cammino semplice e chiuso si dice ciclo.*

Definizione 1.1.5. *Si definisce cammino non diretto la sequenza di vertici $(v_0, v_1, \dots, v_k)$ tali che $(v_i, v_{i+1}) \in E$ oppure $(v_{i+1}, v_i) \in E$, $\forall i = 0, 1, \dots, k-1$.*

Per esempio, il grafo orientato della figura 1.2. è connesso e si ha:

$$V = \{v_1, v_2, v_3, v_4\} \quad \text{e} \quad E = \{(1,3); (1,2); (2,2); (2,4); (3,4); (4,1)\} \subseteq V \times V$$

Di seguito si riporta la seguente tabella per alcuni cammini del grafo.

Cammino diretto	Semplice	Chiuso	Ciclo	Lunghezza
v_1, v_3, v_4	si	no	no	2
v_1, v_2, v_4, v_1	si	si	si	3
v_1, v_2, v_2, v_4, v_1	no	si	no	4
$v_1, v_2, v_2, v_4, v_1, v_2$	no	no	no	5

Notiamo che il cammino v_4, v_3, v_1 è un esempio di cammino non diretto.

Definizione 1.1.6. *Dato un grafo non orientato* $G = (V, E)$ *si definisce catena una sequenza di vertici* $(v_0, v_1, \dots, v_k)$ *tali che* $(v_i, v_{i+1}) \in E, \ \forall i = 0,1, \dots, k-1$ *. Il valore k rappresenta la lunghezza della catena. Se nessun vertice è ripetuto la catena si dice semplice, se* $v_0 = v_k$ *la catena si dice chiusa. Infine una catena semplice e chiusa è detta circuito.*

Definizione 1.1.7. *Un grafo non orientato si dice connesso se* $\forall v_i, v_j \in V$*, dove* $i \neq j$*, esiste almeno una catena* $(v_i, \dots, v_j)$.

Definizione 1.1.8. *Un grafo orientato si dice fortemente connesso se, per ogni coppia di nodi* (v_i, v_j)*, esiste un cammino diretto da* v_i *a* v_j*. In altre parole è fortemente connesso se è possibile transitare per tutti i nodi percorrendo un cammino di archi orientati.*

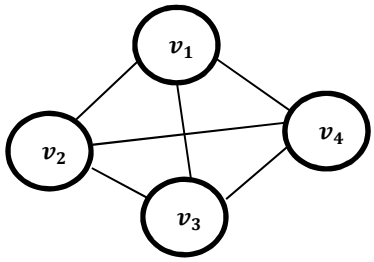

Figura 1.6 Grafo non orientato e connesso

Definizione 1.1.9. *Sia* $G = (V, E)$ *un grafo. La matrice di adiacenza del grafo è una matrice quadrata* A*, di ordine* n*, dove* $n = |V|$*, tale che:*

$$\mathrm{A}_{i,j} = 0 \; se \; (v_i, v_j) \notin E$$

$$\mathrm{A}_{i,j} = 1 \; se \; (v_i, v_j) \in E$$

La matrice delle adiacenze costituisce una particolare struttura dati, comunemente utilizzata nella rappresentazione dei grafi. In particolare è ampiamente utilizzata nella stesura di algoritmi che operano su grafi e in generale nella loro rappresentazione informatica. Vediamo ora due esempi di calcolo della matrice di adiacenza di un grafo.

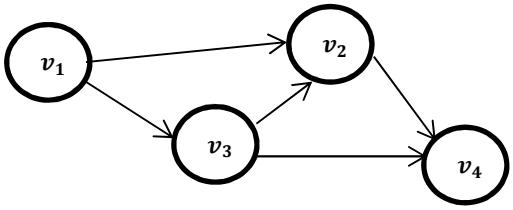

Figura 1.7 Grafo orientato ma non fortemente connesso

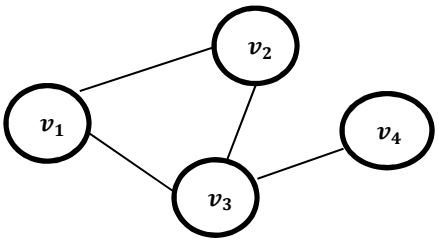

Figura 1.8 Grafo non orientato e non connesso

Nel caso del grafo della figura 1.7 si ha:

$$V = \{1,2,3,4\} \quad \text{e} \quad E = \{(1,2); (1,3); (2,4); (3,2); (3,4)\}$$

e la matrice di adiacenza è

$$\mathrm{A} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Invece, nel caso del grafo della figura 1.8 si ha:

$$V = \{1,2,3,4\} \quad \text{e} \quad E = \{(1,2); (1,3); (2,3); (3,4)\}$$

e la matrice di adiacenza è

$$\mathrm{A} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Notiamo che se il grafo è non orientato, la matrice di adiacenza è simmetrica rispetto alla diagonale principale poiché, per definizione di grafo non orientato, risulta che

$$(v_i, v_j) \in E \Leftrightarrow (v_j, v_i) \in E$$

Invece che partire da un grafo per costruire la relativa matrice di adiacenza, possiamo fare l'inverso, cioè partire da una matrice e costruire il grafo. Data una matrice $\mathrm{A}_{n\times n}$, il grafo di A è definito come il grafo orientato $G(\mathrm{A})$ su un insieme di nodi $\{v_1, \dots, v_n\}$ tra i quali c'è un collegamento diretto tra il nodo v_i e il nodo v_j se e solo se $a_{i,j} \neq 0$.

Definizione 1.1.10. *Data una matrice* $\mathrm{A}_{n\times n}$ *,ogni prodotto della forma* $\widetilde{\mathrm{A}} = \mathrm{P}^T\mathrm{AP}$*, dove* P *è una matrice di permutazione (matrice ottenuta dalla matrice identità permutando le sue righe o colonne), si chiama permutazione simmetrica della matrice* A.

L'effetto della permutazione simmetrica su una matrice è quello di scambiare righe o colonne, l'effetto di una permutazione simmetrica sul grafo di una matrice è quello di rinumerare i nodi. Infatti osserviamo che i grafi associati alle matrici A e $\widetilde{\mathrm{A}}$ differiscono univocamente per la numerazione dei nodi. Poiché

$$a_{i,j} = \tilde{a}_{\sigma_i,\sigma_j}$$

si ha che

$$a_{i,j} \neq 0 \Leftrightarrow \tilde{a}_{\sigma_i,\sigma_j} \neq 0$$

Quindi un arco orientato unisce il nodo v_i con il nodo v_j del grafo associato ad A se e solo se un arco unisce il nodo v_{σ_i} con il nodo v_{σ_j} del grafo associato a $\widetilde{\mathrm{A}}$. Di conseguenza il grafo orientato di una matrice è invariante per permutazioni simmetriche, cioè se P è una matrice di permutazione allora

$$G\left(\mathrm{P^T AP}\right) = G(\mathrm{A})$$

Per esempio, data $\mathrm{A} = \begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix}$, il grafo di A risulta essere quello della figura 1.9

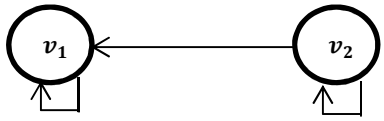

Figura 1.9 Grafo dedotto da una matrice

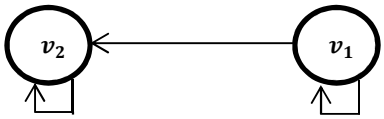

Figura 1.10 Effetto della permutazione simmetrica sul grafo 1.8

Se consideriamo come matrice di permutazione $\mathrm{P} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ avremo

$$\widetilde{\mathrm{A}} = \mathrm{P}^T\mathrm{AP} = \begin{pmatrix} 3 & 2 \\ 0 & 1 \end{pmatrix}$$

e il grafico $G\left(\mathrm{P^T AP}\right)$ è quello mostrato in figura 1.10.

Definizione 1.1.11. *Una matrice* A *si dice matrice riducibile se esiste una matrice di permutazione* P *tale che:*

$$\widetilde{\mathrm{A}} = \mathrm{P}^T\mathrm{AP} = \begin{pmatrix} \mathrm{X} & \mathrm{Y} \\ 0 & \mathrm{Z} \end{pmatrix} \quad \textit{dove } \mathrm{X} \textit{ e } \mathrm{Z} \textit{ sono matrici quadrate.} \qquad (1.1.1)$$

Una matrice si dirà irriducibile se non è riducibile.

Teorema 1.1.1. *Una matrice quadrata* A *è irriducibile se e solo se il suo grafo orientato è fortemente connesso.*

Dimostrazione. $\Leftarrow$ Supponiamo che la matrice A sia riducibile, allora esiste una matrice di permutazione P tale che $\widetilde{\mathrm{A}} = \mathrm{P}^T\mathrm{AP}$ è triangolare a blocchi, cioè è del tipo

$$\widetilde{\mathrm{A}} = \begin{pmatrix} \mathrm{X} & \mathrm{Y} \\ 0 & \mathrm{Z} \end{pmatrix}$$

con X matrice $m \times m$. Quindi il grafo associato a $\widetilde{\mathrm{A}}$ non ha archi che uniscono i nodi con $i > m$ ai nodi $j \leq m$. Quindi non è fortemente connesso.

$\Longrightarrow$ Se il grafo non è fortemente connesso si trova la matrice di permutazione che porta A nella forma triangolare a blocchi nel modo seguente. Supponiamo che esista una coppia di nodi

(v_p, v_q) per cui a partire da v_p non si possa raggiungere v_q percorrendo archi orientati nel grafo. Allora costruiamo l'insieme $\mathcal{P}$ dei nodi raggiungibili da v_p e $\mathcal{Q}$ l'insieme dei nodi non raggiungibili da v_p. $\mathcal{Q}$ è sicuramente non vuoto in quanto contiene v_q, inoltre non possono esserci archi orientati che connettono nodi di $\mathcal{P}$ con nodi di $\mathcal{Q}$. Allora basta ordinare le righe e le colonne di A in modo che in testa ci siano gli indici di $\mathcal{Q}$ e in coda i nodi di $\mathcal{P}$. In questo modo il blocco in basso a sinistra con indice di riga in $\mathcal{P}$ e indice di colonna in $\mathcal{Q}$ sarà costituito tutto da elementi nulli.

■

Per esempio, data una qualunque matrice quadrata A, non è facile affermare se essa sia riducibile o irriducibile, in quanto non è immediato trovare una matrice di permutazione P tale che sia verificata la (1.1.1). Grazie a questo teorema la questione diventa abbastanza semplice, infatti, basta esaminare il grafo della matrice A e se esso risulta fortemente connesso allora la matrice A è irriducibile.

Utilizzando la teoria dei grafi si possono schematizzare, ad esempio, reti di computer e mappe di siti. Infatti, l'insieme dei documenti presenti sul web si può visualizzare come un grafo, detto grafo del web, in cui:

- Gli URL (universal resource locator, indirizzi in formato specifico che possono identificare in modo univoco la posizione di un oggetto nel web) sono i nodi.
- Gli hyperlinks (collegamenti ipertestuali tra i diversi documenti) sono gli archi diretti che connettono i nodi. C'è un arco tra il nodo x e il nodo y quando la pagina che corrisponde all'URL x contiene un link verso l'URL y. Ciascuna pagina web è caratterizzata da archi entranti e da archi uscenti.

Il grafo web è un grafo dinamico che cambia in continuazione e ha dimensioni enormi, si parla di un numero di nodi di circa 2-4 miliardi e un numero di archi di circa 60-100 miliardi.

Concludiamo il paragrafo con le seguenti definizioni:

Definizione 1.1.12. *Sia $G = (V, E)$ un grafo orientato. Si definisce grado entrante di un nodo $v_i \in V$, lo si indica con $indegree(v_i)$, il numero di archi entranti nel nodo v_i (inlinks). Si definisce grado uscente di un nodo $v_i \in V$. lo si indica con $outdegree(v_i)$, il numero di archi uscenti dal nodo v_i (outlinks).*

Definizione 1.1.13. *Sia $G = (V, E)$ un grafo orientato, con un numero di nodi pari a n. Si definisce grado di un vertice $v_i \in V$, lo si indica con $degree(v_i)$, la somma del grado entrante e del grado uscente, cioè*

$$degree(v_i) = indegree(v_i) + outdegree(v_i)$$

Si definisce grado di un grafo, lo si indica con $degree(G)$, il massimo dei gradi dei suoi vertici, cioè

$$degree(G) = \max_{1 \le i \le n} \{degree(v_i) | v_i \in V\}$$

1.2 Metodo delle potenze (The power method)

Prima di affrontare la descrizione matematica di tale metodo ricordiamo alcune definizioni e risultati riguardanti autovalori e autovettori.

Definizione 1.2.1. *Data una matrice quadrata* A *di ordine n, di elementi reali o complessi, si chiama autovalore di* A *un numero $\lambda \in \mathbb{K}$, per il quale esiste un vettore non nullo, $\boldsymbol{v} \in \mathbb{K}^n$,tale che*

$$\mathrm{A}\boldsymbol{v} = \lambda \boldsymbol{v}.$$

Si dice che $v \neq 0 \in \mathbb{R}^n$ è un autovettore sinistro per A *se*

$$\boldsymbol{y}^T \mathrm{A} = \lambda \boldsymbol{y}^T$$

Denotiamo con:

- $Sp(\mathrm{A})$ spettro di A, l'insieme degli autovalori della matrice A

$$Sp(\mathrm{A}) = \{\lambda \in \mathbb{K} | \exists \boldsymbol{v} \neq 0, \mathrm{A}\boldsymbol{v} = \lambda \boldsymbol{v}\}.$$

- $\rho(\mathrm{A}) = \max_{1 \leq i \leq n} |\lambda_i|$ raggio spettrale della matrice A.

- V_λ autospazio di A, l'insieme degli autovettori relativi a λ

$$V_\lambda = \{\boldsymbol{v} \in V | \mathrm{A}\boldsymbol{v} = \lambda \boldsymbol{v}\}$$

Definizione 1.2.2. *Per un vettore arbitrario $\boldsymbol{x} \neq 0$ di $\mathbb{C}^n$, il numero λ, dato da*

$$\lambda = \frac{\boldsymbol{x}^* \mathrm{A} \boldsymbol{x}}{\boldsymbol{x}^* \boldsymbol{x}} \tag{1.2.1}$$

è detto quoziente di Rayleigh, dove $\boldsymbol{x}^* = \overline{\boldsymbol{x}}^T$.

Definizione 1.2.3. *Una matrice A si dice non negativa se tutti i suoi elementi sono numeri reali non negativi, $a_{i,j} \geq 0$.*

Definizione 1.2.4. *Una matrice A si dice matrice primitiva se* A *è una matrice non negativa e se possiede un solo autovalore, $r = \rho(\mathrm{A})$, nel cerchio spettrale (cerchio del piano complesso centrato nell'origine e di raggio pari al raggio spettrale).*

Per poter affermare se una matrice non negativa è primitiva, non c'è bisogno di calcolare gli autovalori e di contare quanti di essi cadano nel cerchio spettrale, possiamo riferirci al teorema 1.2.1.

Teorema 1.2.1. *Data una matrice* A *non negativa,* A *è primitiva* $\Leftrightarrow \exists\, m > 0 \in \mathbb{N}$ *tale che* $\mathrm{A}^m > 0$.

Dimostrazione. Viene ommessa. Si veda [4], Capitolo 8, pag. 678.

Teorema 1.2.2 (Perron-Frobenius) *Sia* $\mathrm{A} \in \mathbb{R}^{n\times n}$ *una matrice non negativa e irriducibile. Allora valgono i seguenti fatti:*

I. $r = \rho(\mathrm{A}) \in Sp(\mathrm{A})$ *e* $r = \rho(\mathrm{A}) > 0$

II. *La molteplicità algebrica di* r *è* 1, $m_a(r) = 1$

III. *Esiste un autovettore* $\boldsymbol{x} > 0$ *relativo all'autovalore* $\rho(\mathrm{A})$

IV. *Il vettore di Perron* $\boldsymbol{p}$ *è l'unico vettore tale che*

$$\mathrm{A}\boldsymbol{p} = \rho(\mathrm{A})\boldsymbol{p}, \text{ con } \boldsymbol{p} > 0 \text{ e } \|\boldsymbol{p}\|_1 = 1$$

dove $\|\boldsymbol{p}\|_1 = \sum_{i=1}^{n} |p_i|$, p_i *componente i-esima di* $\boldsymbol{p}$

V. *Se* $\mathrm{A} > 0$, $\rho(\mathrm{A})$ *è l'unico autovalore dominante*

Dimostrazione. Viene ommessa. Si veda [4], Capitolo 8, pag. 673.

Affinché una matrice sia primitiva, un' ulteriore condizione necessaria e sufficiente è data dal seguente teorema

Teorema 1.2.3. *Una matrice* A *non negativa e irriducibile, con* $r = \rho(\mathrm{A})$, *è primitiva se e solo se esiste*

$$\lim_{k\to\infty} \left(\frac{\mathrm{A}}{r}\right)^k$$

In tal caso

$$\lim_{k\to\infty} \left(\frac{\mathrm{A}}{r}\right)^k = \frac{\boldsymbol{p}\boldsymbol{q}^{\boldsymbol{T}}}{\boldsymbol{q}^T\boldsymbol{p}} > 0$$

dove $\boldsymbol{p}$ *e* $\boldsymbol{q}^T$ *sono rispettivamente il vettore destro e sinistro di Perron della matrice* A .

Dimostrazione. Viene ommessa. Si veda [4], Capitolo 8, pag. 674.

Il metodo delle potenze è un metodo iterativo particolarmente adatto per approssimare l'autovalore di modulo massimo (autovalore dominante) di un matrice ed un corrispondente autovettore. Esso trova impiego in specifici problemi, ad esempio è alla base del metodo utilizzato da Google per il proprio algoritmo di PageRank.

Consideriamo il caso di una matrice $\mathrm{A}_{n\times n}$ con n autovettori di $\mathbb{C}^n$ linearmente indipendenti, associati agli autovalori $\lambda_1, \dots, \lambda_n$, tali che

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$$

Stiamo cioè supponendo che l'autovalore dominante λ_1 (autovalore di modulo massimo) abbia molteplicità algebrica 1.

Denotiamo con $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ gli n autovettori di A associati agli autovalori $\lambda_1, \dots, \lambda_n$, quindi

$$\mathrm{A}\boldsymbol{x}_i = \lambda_i \boldsymbol{x}_i \;\; i = 1, \dots, n$$

Consideriamo un vettore iniziale $\boldsymbol{y}_0 \in \mathbb{C}^n$ e scriviamolo come combinazione lineare degli autovettori $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$:

$$\boldsymbol{y}_0 = \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i$$

supponendo che $\boldsymbol{y}_0$ non sia ortogonale a $\boldsymbol{x}_1$, cioè $\alpha_1 \neq 0$.

A partire da $\boldsymbol{y}_0$ generiamo la successione:

$$\boldsymbol{y}_1 = \mathrm{A}\boldsymbol{y}_0, \qquad \boldsymbol{y}_2 = \mathrm{A}\boldsymbol{y}_1, \qquad \dots \qquad , \boldsymbol{y}_k = \mathrm{A}\boldsymbol{y}_{k-1}$$

Teorema 1.2.4. *La successione* $\{\boldsymbol{y}_k\}$ *sopra descritta , al crescere di* k*, converge all'autovettore relativo all'autovalore dominante.*

Dimostrazione. Infatti, esaminando il comportamento della successione $\{\boldsymbol{y}_k\}$, per $k \to \infty$, si ha

$$\boldsymbol{y}_k = \mathrm{A}\boldsymbol{y}_{k-1} = \mathrm{A}^2\boldsymbol{y}_{k-2} = \dots = \mathrm{A}^{\mathrm{k}}\boldsymbol{y}_0 = \mathrm{A}^{\mathrm{k}} \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i = \sum_{i=1}^{n} \alpha_i \mathrm{A}^{\mathrm{k}} \boldsymbol{x}_i =$$

$$= \sum_{i=1}^{n} \alpha_i \lambda_i^k \boldsymbol{x}_i = \lambda_1^k \left[\alpha_1 \boldsymbol{x}_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k \boldsymbol{x}_i \right]$$

Posto

$$\boldsymbol{g}_k = \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k \boldsymbol{x}_i$$

avremo

$$\boldsymbol{y}_k = \lambda_1^k (\alpha_1 \boldsymbol{x}_1 + \boldsymbol{g}_k) \qquad (1.2.2)$$

dalla quale si vede che $\boldsymbol{y}_k$ tende a disporsi nella direzione di $\boldsymbol{x}_1$ per $k \to \infty$. Infatti

$$\lim_{k \to \infty} \boldsymbol{g}_k = 0$$

poiché $\left| \frac{\lambda_i}{\lambda_1} \right| < 1$ per $i \geq 2$.

■

Osservazione 1.2.1. La velocità di convergenza del metodo dipende dai rapporti $\left| \frac{\lambda_i}{\lambda_1} \right|$, per $i = 2,3, \dots, n$. Questi rapporti sono tutti maggiorati da $\left| \frac{\lambda_2}{\lambda_1} \right|$.

Consideriamo il quoziente di Rayleigh (1.2.1), associato al vettore $\boldsymbol{y}_k \simeq \lambda_1^k \alpha_1 \boldsymbol{x}_1$

$$\sigma_k = \frac{\boldsymbol{y}_k^* \boldsymbol{y}_{k+1}}{\boldsymbol{y}_k^* \boldsymbol{y}_k} = \frac{\boldsymbol{y}_k^* \mathrm{A} \boldsymbol{y}_k}{\boldsymbol{y}_k^* \boldsymbol{y}_k} \tag{1.2.3}$$

Teorema 1.2.5. *La successione di scalari σ_k, generata dal metodo delle potenze, converge all'autovalore dominante λ_1, cioè*

$$\lim_{k\to\infty} \sigma_k = \lambda_1$$

Dimostrazione. Calcoliamo il quoziente di Rayleigh dato da (1.2.3)

$$\sigma_k = \frac{\overline{\lambda}_1^k (\alpha_1 x_1 + g_k)^* \lambda_1^{k+1} (\alpha_1 x_1 + g_{k+1})}{\overline{\lambda}_1^k (\alpha_1 x_1 + g_k)^* \lambda_1^k (\alpha_1 x_1 + g_k)} = \lambda_1 \frac{(\overline{\alpha}_1 x_1^* + g_k^*)(\alpha_1 x_1 + g_{k+1})}{(\overline{\alpha}_1 x_1^* + g_k^*)(\alpha_1 x_1 + g_k)}$$

Per $k \to \infty$, $\boldsymbol{g}_k$ e $\boldsymbol{g}_{k+1}$ tendono a zero.

■

Se $\boldsymbol{y}_k^{(m_k)}$ è una componente di massimo modulo di $\boldsymbol{y}_k$, segue che

$$\boldsymbol{y}_k^{(m_k)} \simeq \lambda_1^k \alpha_1 \boldsymbol{x}_1^{(m)}$$

dove $\boldsymbol{x}_1^{(m)}$ è la componente di massimo modulo di $\boldsymbol{x}_1$. Avremo allora

$$\lim_{k\to\infty} \frac{\boldsymbol{y}_k}{\boldsymbol{y}_k^{(m_k)}} = \frac{\boldsymbol{x}_1}{\boldsymbol{x}_1^{(m)}}$$

cioè $\dfrac{\boldsymbol{y}_k}{\boldsymbol{y}_k^{(m_k)}}$ converge all'autovettore dominante normalizzato in norma infinito, dove, dato $\boldsymbol{x} \in \mathbb{C}^n$, $||\boldsymbol{x}||_\infty = \max_i |x_i|$.

Nella forma precedente, l'algoritmo può presentare problemi di underflow o overflow dovuti al fatto che λ_1^k può tendere a zero o a infinito. Infatti si può osservare che:

- Se $|\lambda_1| < 1 \Longrightarrow \|\boldsymbol{y}_k\|_2 \simeq \left\|\lambda_1^k \alpha_1 \boldsymbol{x}_1\right\|_2 \to 0$ (underflow)
- Se $|\lambda_1| > 1 \Longrightarrow \|\boldsymbol{y}_k\|_2 \simeq \left\|\lambda_1^k \alpha_1 \boldsymbol{x}_1\right\|_2 \to \infty$ (overflow)

con $||\cdot||_2$ norma euclidea. Per evitare che $\boldsymbol{y}_k$ diventi troppo piccolo o troppo grande (in norma), si usa la tecnica di normalizzazione, la quale impone il vincolo che la successione $\{\boldsymbol{y}_k\}$ sia sulla superficie sferica di centro l'origine e raggio 1.

Si parte sempre da $\boldsymbol{z}_0 = \boldsymbol{y}_0$ e lo si normalizza

$$\boldsymbol{t}_0 = \frac{\boldsymbol{z}_0}{\|\boldsymbol{z}_0\|_2}$$

Si pone quindi

$$\boldsymbol{z}_1 = \mathrm{A}\boldsymbol{t}_0 \quad \text{e} \quad \boldsymbol{t}_1 = \frac{\boldsymbol{z}_1}{\|\boldsymbol{z}_1\|_2}$$

Si costruisce così la successione di vettori $\{\boldsymbol{t}_k\}$, con $\|\boldsymbol{t}_k\|_2 = 1$, mediante la seguente relazione

$$\boldsymbol{t}_k = \frac{\boldsymbol{z}_k}{\|\boldsymbol{z}_k\|_2}$$

e si pone

$$\boldsymbol{z}_k = \mathrm{A}\boldsymbol{t}_{k-1}$$

Si verifica facilmente che

$$\sigma_k = \frac{\boldsymbol{t}_k^* \boldsymbol{z}_{k+1}}{\boldsymbol{t}_k^* \boldsymbol{t}_k} \tag{1.2.4}$$

Infatti basta porre al secondo membro della (1.2.4)

$$\boldsymbol{t}_k = \frac{1}{\prod_{i=0}^{k}\|\boldsymbol{z}_i\|_2}\mathrm{A}^k\boldsymbol{y}_0 = \frac{1}{\prod_{i=0}^{k}\|\boldsymbol{z}_i\|_2}\boldsymbol{y}_k$$

$$\boldsymbol{z}_{k+1} = \frac{1}{\prod_{i=0}^{k}\|\boldsymbol{z}_i\|_2}\mathrm{A}^{k+1}\boldsymbol{y}_0 = \frac{1}{\prod_{i=0}^{k}\|\boldsymbol{z}_i\|_2}\boldsymbol{y}_{k+1}$$

Osservazione 1.2.2. Il metodo delle potenze è convergente anche nel caso in cui l'autovalore di modulo massimo abbia molteplicità algebrica maggiore di 1, cioè $\lambda_1 = \lambda_2 = \ldots = \lambda_r$ con:

$$|\lambda_1| = |\lambda_2| = \ldots = |\lambda_r| > |\lambda_{r+1}| \geq \ldots \geq |\lambda_n|$$

Quando, invece, l'autovalore di modulo massimo non è unico, i risultati di convergenza precedenti non sono più applicabili.

1.3 Concetti fondamentali sulle catene di Markov

Definizione 1.3.1. (Processo Aleatorio o Stocastico) *Sia $\{\Omega, \mathcal{A}, \mathcal{P}\}$ uno spazio di probabilità, T l'insieme dei possibili valori di un parametro t, ξ una σ-algebra e $\{E, \xi\}$ uno spazio misurabile. Si definisce processo aleatorio di parametro t, una famiglia di variabili aleatorie $\{X_t\}_{t\in T}$ definite in Ω e a valori in E, indicizzate dal parametro t.*

- I valori che possono assumere le variabili aleatorie X_t sono detti stati.

- L'insieme degli stati, E, è detto lo spazio degli stati e può essere un insieme finito, infinito numerabile ed infinito con cardinalità del continuo.
  Ad esempio se $X_t = k \in E$ diremo che il sistema al tempo t si trova nello stato k.

Il processo stocastico è detto:

- A tempo discreto, se l'insieme T è composto da valori discreti

- A tempo continuo, se l'insieme T ha cardinalità del continuo

Definizione 1.3.2. (Proprietà di Markov) *Un processo stocastico verifica la proprietà di Markov se per ogni istante di tempo $t \in T$, per ogni coppia di stati $i, j \in E$ e per ogni sequenza di stati $k_o, \dots, k_{t-1} \in E$, risulta*

$$\mathbb{P}(X_{t+1} = j | X_0 = k_0, \dots, X_{t-1} = k_{t-1}, X_t = i) = \mathbb{P}(X_{t+1} = j | X_t = i)$$

La probabilità condizionata

$$\mathbb{P}(X_{t+1} = j | X_t = i) = p_{i,j}(t)$$

è detta probabilità di transizione al tempo t dallo stato i allo stato j.

Definizione 1.3.3. (Catena di Markov) *Un processo stocastico discreto $\{X_t\}$ con $t \in T$, $T = \mathbb{N}_{\{0\}}$, avente un insieme di stati E numerabile, è detto catena di Markov se è verificata la propietà di Markov.*

Dalla definizione (1.3.2) si deduce una caratteristica fondamentale delle catene di Markov, cioè l'indipendenza dello stato futuro, condizionatamente a quello presente, dagli stati passati. Questa proprietà indica che la probabilità condizionata di un evento futuro, dati lo stato attuale e tutti gli eventi passati, dipende esclusivamente dallo stato attuale del processo e non dai precedenti. In pratica è come se il processo avesse una perdita di memoria riguardo al suo passato.

Definizione 1.3.4. *Una catena di Markov si dice omogenea o stazionaria (nel tempo) se, per ogni $i, j \in E$, si ha che $\mathbb{P}(X_{t+1} = j | X_t = i)$ è indipendente da t, quindi*

$$p_{i,j}(t) = p_{i,j} \; \forall \, i, j$$

Le catene di Markov a stati finiti possono essere facilmente studiate per mezzo di un approccio basato sulle matrici.

Definizione 1.3.5. (Matrice di transizione): *La matrice* $\mathrm{P} = p_{i,j}$ *è detta matrice delle probabilità di transizione (o semplicemente matrice di transizione) della catena se i suoi elementi soddisfano le seguenti proprietà:*

i. $p_{i,j} \geq 0 \; \forall \, i, j \in E$

ii. $\sum_j p_{i,j} = 1 \; \forall i \in E$, *equivalentemente* $\mathrm{P}\boldsymbol{e} = \boldsymbol{e}$ *con* $\boldsymbol{e} = (1, \dots, 1)^T$

Una matrice i cui elementi soddisfano le precedenti proprietà si dice matrice stocastica.

Possiamo affermare che ogni catena di Markov definisce una matrice stocastica e ogni matrice stocastica definisce una catena di Markov.

Osservazione 1.3.1. Se $\mathrm{P}_{n \times n}$ è una matrice stocastica, allora $\rho(\mathrm{P}) = 1$. Infatti avendo la somma delle righe uguali a 1 si ha che

$$\|\mathrm{P}\|_\infty = \max_{i=1,\dots,n} \sum_{j=1}^{n} |p_{i,j}| = 1.$$

Poiché $\rho(P) \leq \|P\|$ per ogni norma di matrice e, sfruttando la condizione $P\boldsymbol{e} = \boldsymbol{e}$, cioè che, per le matrici stocastiche il vettore $\boldsymbol{e}$ è un autovettore corrispondente all'autovalore $\lambda = 1$, si ha

$$1 \leq \rho(P) \leq \|P\|_\infty = 1 \implies \rho(P) = 1$$

Un metodo pratico per studiare le proprietà degli stati di una catena di Markov è quella di analizzarne il grafo associato, nel quale ad ogni stato corrisponde un nodo ed ad ogni valore $p_{i,j} > 0$ corrisponde un arco orientato dal nodo *i* al nodo *j*.
Uno stato *j* risulta accessibile dallo stato *i* se e solo se esiste un cammino orientato dal nodo *i* al nodo *j*. Due stati *i* e *j* risultano comunicanti se e solo se appartengono ad uno stesso ciclo orientato.

Definizione 1.3.6 *Una catena di Markov si dice irriducibile se la sua matrice di transizione* P *è una matrice irriducibile, cioè per ogni coppia di indici* (h, k)*, lo stato* i_k *può essere raggiunto dallo stato* i_h*. Si dice catena di Markov aperiodica se è una catena irriducibile e se la matrice di transizione* P *è una matrice primitiva.*

Definizione 1.3.7. *Un vettore di distribuzione di probabilità (o semplicemente vettore di probabilità) è un vettore riga non negativo*

$$\boldsymbol{p}^T = (p_1, \dots, p_n) \quad \textit{tale che} \quad \sum_k p_k = 1.$$

Data una catena di Markov con matrice di transizione P*, si dice vettore di probabilità stazionario un vettore di probabilità* $\boldsymbol{p}^T$ *tale che*

$$\boldsymbol{p}^T P = \boldsymbol{p}^T$$

Osservazione 1.3.2. Ogni riga di una matrice stocastica è un vettore di probabilità.

L'irriducibilità della matrice di transizione è una proprietà molto importante perché garantisce, grazie al Teorema di Perron-Frobenius, l'esistenza e l'unicità del vettore di probabilità stazionario, che risulta essere un autovettore sinistro per una catena di Markov.

Definizione 1.3.8. *Il k-esimo passo di un vettore di probabilità per una catena con n stati è*

$$\boldsymbol{p}^T(k) = (p_1(k), \dots, p_n(k)) \quad \textit{dove} \quad p_j(k) = \mathbb{P}(X_k = S_j)$$

In altre parole, $p_j(k)$ *è la probabilità di essere allo stato j-esimo dopo il k-esimo passo.*
Il vettore di probabilità iniziale è definito come

$$\boldsymbol{p}^T(0) = (p_1(0), \dots, p_n(0)) \quad \textit{dove} \quad p_j(0) = \mathbb{P}(X_0 = S_j)$$

cioè la probabilità che la catena inizi dallo stato S_j*.*

Tutta l'analisi Markoviana è imperniata sul comportamento transitorio della catena, in aggiunta al suo comportamento limite. Dato un vettore di probabilità iniziale $\boldsymbol{p}^T(0)$, il primo scopo è

quello di calcolare la probabilità di essere in un generico stato dopo la prima transizione, cioè di determinare

$$\boldsymbol{p}^T(1) = \big(p_1(1), \dots, p_n(1)\big)$$

Per ogni j avremo

$$p_j(1) = \mathbb{P}\big(X_1 = S_j\big) = \mathbb{P}\big[X_1 = S_j \cap (X_0 = S_1 \cup X_0 = S_2 \cup \dots \cup X_0 = S_n)\big] =$$

$$= \mathbb{P}\big[\big(X_1 = S_j \cap X_0 = S_1\big) \cup \big(X_1 = S_j \cap X_0 = S_2\big) \cup \dots \cup \big(X_1 = S_j \cap X_0 = S_n\big)\big] =$$

$$= \sum_{i=1}^{n} \mathbb{P}\big(X_1 = S_j \cap X_0 = S_i\big) = \sum_{i=1}^{n} \mathbb{P}(X_0 = S_i)\mathbb{P}\big(X_1 = S_j | X_0 = S_i\big) = \sum_{i=1}^{n} p_i(0)p_{i,j}$$

In altre parole

$$\boldsymbol{p}^T(1) = \boldsymbol{p}^T(0)\mathrm{P}$$

la quale descrive l'evoluzione dalla distribuzione iniziale alla distribuzione dopo il primo passo. Sfruttando la proprietà di Markov riguardante l' assenza di memoria, e partendo con vettori di distribuzione iniziale $\boldsymbol{p}^T(1)$, $\boldsymbol{p}^T(2)$, etc., avremo

$$\boldsymbol{p}^T(2) = \boldsymbol{p}^T(1)\mathrm{P}\ ,\ \boldsymbol{p}^T(3) = \boldsymbol{p}^T(2)\mathrm{P}\ \text{, etc.}$$

Sostituendo otterremo

$$\boldsymbol{p}^T(k) = \boldsymbol{p}^T(0)\mathrm{P}^k$$

che è un caso speciale del metodo delle potenze.

Analizziamo ora il limite per $k \to \infty$ della matrice P^k. Ci mettiamo nel seguente caso:
P è irriducibile ed esiste

$$\lim_{k\to\infty} \mathrm{P}^k$$

Ciò significa che la matrice P è primitiva per il teorema 1.2.3. In tale situazione il limite può essere calcolato facilmente. Il vettore di Perron per la matrice P è

$\frac{\boldsymbol{e}}{n}$, vettore di probabilità uniforme, con $\boldsymbol{e} = (1, \dots, 1)^T$.

Se $\boldsymbol{\pi} = (\pi_1, \dots, \pi_n)^T$ è il vettore di Perron per la matrice P^T , cioè $\boldsymbol{\pi}^T\mathrm{P} = \boldsymbol{\pi}^T$, allora per il teorema 1.2.3. si ha

$$\lim_{k\to\infty} \mathrm{P}^k = \frac{\left(\frac{\boldsymbol{e}}{n}\right)\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T\left(\frac{\boldsymbol{e}}{n}\right)} = \frac{\boldsymbol{e}\boldsymbol{\pi}^T}{\boldsymbol{\pi}^T\boldsymbol{e}} = \boldsymbol{e}\boldsymbol{\pi}^T = \begin{pmatrix} \pi_1 & \cdots & \pi_n \\ \vdots & \ddots & \vdots \\ \pi_1 & \cdots & \pi_n \end{pmatrix} > 0$$

Inoltre, se P è primitiva, esiste il limite della distribuzione ed è dato da

$$\lim_{k\to\infty} \boldsymbol{p}^T(k) = \lim_{k\to\infty} \boldsymbol{p}^T(0)\mathrm{P}^k = \boldsymbol{p}^T(0)\boldsymbol{e}\boldsymbol{\pi}^T = \boldsymbol{\pi}^T$$

Poiché $\sum_k p_k(0) = 1,$ il termine $\boldsymbol{p}^T(0)\boldsymbol{e}$ sparisce, e in tal modo il valore del limite è indipendente dal valore della distribuzione di probabilità iniziale $\boldsymbol{p}^T(0)$.

Capitolo 2

Nascita di PageRank

2.1 Storia dell’ Algoritmo Pagerank

La ricerca di informazioni sul web è effettuata per la maggior parte tramite i motori di ricerca, uno dei più famosi è Google Search che fu creato il 27 settembre 1997. Negli anni ' 90, Internet cominciò a riempirsi di milioni di documenti: si avverti l' esigenza di catalogarli, riordinarli e indicizzarli. In altri termini, era necessario adottare un sistema "bibliotecario" per consentire agli utenti di rinvenire, in tempi brevi, le informazioni desiderate. Nel 1998 due studenti della Stanford University si armarono di un idea e trovarono un finanziatore che staccasse loro un assegno affinché la sviluppassero. I due ragazzi si chiamano Larry Page e Sergey Brin e la loro idea si chiama “Google”.

Figura 2.1 Logo del motore di ricerca Google

La parola "Google" deriva da uno scherzoso riferimento al termine “Googol” , coniato da Milton Sirotta (nipote del matematico americano Edward Kasner) nel 1938, per riferirsi al numero rappresentato da 1 seguito da 100 zeri. Durante la stesura del saggio *Mathematics and the Imagination*, Kasner decise di chiedere al nipote di otto anni di inventare un vocabolo che potesse designare il numero 1 seguito da 100 zeri (ossia 1×10^{100}), da inserire nel suo libro. Il bambino pronunciò una parola che resterà nella storia: “Googol”.

$$1 \text{ Googol} = 1.0 \times 10^{100}$$

Un Googol è circa pari a 70!. In matematica esso non ha un particolare significato, se non quello di essere utile per un confronto con altri numeri incredibilmente grandi, come quello degli atomi nell’universo visibile (stimato tra 10^{72} e 10^{82}). Edward Kasner , in un suo libro, propose anche il termine “googolplex” per indicare il numero (ancora più grande) che si ottiene

elevando dieci alla “googolesima” potenza. Cosa leghi questo episodio al nome del motore di ricerca ce lo svela, nel 2004, David Koller, ricercatore del dipartimento di informatica della prestigiosa Università di Stanford. David racconta in un’intervista che nel 1997 partecipò, assieme a un gruppo di studenti, ad una riunione di progetto organizzata da Brin e Page. I fondatori della nuova società avevano intenzione di trovare un nome che si riferisse alle finalità dell’azienda, vale a dire indicizzare l’infinito contenuto di informazioni disponibili in rete. Uno degli studenti presenti, Sean Anderson, memore del saggio pubblicato sessant’anni prima da Kasner, propose allora il termine “Googolplex”, a cui fu preferito il più breve “Googol”. Una volta scelto il nome, Sean si assicurò che non fosse già presente nel registro dei domini pubblicato su Internet (ossia, che il nome non fosse già stato utilizzato o prenotato da qualche altro sito web) e che potesse dunque essere depositato. Basandosi sulla pronuncia della parola, Sean digitò erroneamente la sequenza, ossia “google.com” anziché "googol.com". La nuova versione della parola piacque a tal punto ai due ricercatori californiani da spingerli ad impiegarla definitivamente per la loro neonata invenzione. L’uso che Google fa del termine riflette la missione del motore di ricerca: organizzare un immenso, praticamente infinito, insieme di informazioni e documenti disponibili sul Web.
Il funzionamento di Google è molto diverso da quello di tutti gli altri motori di ricerca che producono i risultati in relazione alla frequenza con cui determinate parole chiave compaiono all’ interno di ogni sito web; Esso, invece, ha sviluppato una tecnologia di ricerca che si basa su una serie di calcoli simultanei. Tale tecnologia è chiamata PageRank. Il PageRank (PR) è un valore che Google assegna a ogni sito presente nel suo database. Il nome deriva dall’unione delle due parole Page e Rank. Rank sta per classifica, quindi classificazione dei siti; Page deriva dal nome di chi lo ha progettato in parte, Larry Page, ma anche dal termine pagina. Google usa una formula matematica chiamata appunto PageRank per giudicare l’importanza delle pagine che corrispondono ad una ricerca. Sergey Brin ebbe l’idea che l’informazione sul web potesse essere ordinata in maniera gerarchica tramite la popolarità dei link: una pagina è considerata di migliore qualità rispetto a un'altra se possiede un numero maggiore di inlinks[1].

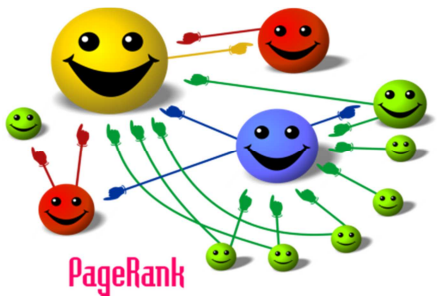

Figura 2.2 Schematizzazione del principio base di PageRank

The Anatomy of a Large-Scale Hypertextual Web Search Engine, pubblicato nel 1998, è il nome del primo documento riguardante il progetto di Brin e Page, il quale descrive PageRank e il prototipo iniziale del motore di ricerca Google.

[1] Sergey Brin & Lawrence Page, *The Anatomy of a Large-Scale Hypertextual Web Search Engine*

2.2 Information Retrieval e gli Algoritmi di Link Analysis Ranking

Il World Wide Web (WWW) è una vasta struttura di collegamenti ipertestuali e informazioni eterogenee che includono testi, immagini, video e metadati. La quantità di informazioni disponibili in formato elettronico è cresciuta in maniera esponenziale negli ultimi anni, basti pensare che i navigatori di Internet trovano sul web centinaia di milioni di pagine. L'aumento della mole di dati consultabili ha, però, avuto come conseguenza da parte degli utenti l'accresciuta difficoltà di ricercare informazioni interessanti per i propri scopi e, con essa, la necessità di avere a disposizione degli strumenti efficaci per il recupero dell'informazione. È in questo scenario che si colloca quel settore dell'informatica che va sotto il nome di Information Retrieval (recupero dell'informazione, in sigla (IR)). In breve esso si occupa dei problemi relativi alla memorizzazione, rappresentazione e reperimento dei documenti. La maggior parte degli utenti utilizza, come strumenti per il reperimento delle informazioni, i motori di ricerca che permettono di trovare informazioni dal web. Essi rispondono ogni giorno a milioni di richieste e agiscono come un recipiente di contenuti, come se mantenessero un ricordo di ogni informazione disponibile nel WWW.

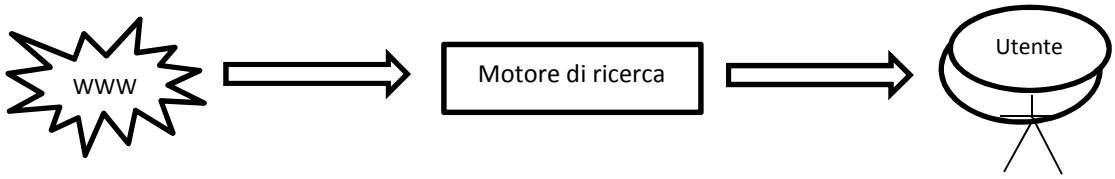

Figura 2.3 Idea su come agiscono i motori di ricerca

Un motore di ricerca si occupa di tre attività logicamente distinte:

1) Raccolta dei dati
2) Elaborazione dei dati raccolti
3) Risposta alle richieste (query) degli utenti

Per quanto concerne il punto 1) si tratta di recuperare il contenuto delle pagine Web (di solito, limitandosi a quelle testuali). Viene svolta da un'apposita componente, detta spider.

Nella elaborazione dei dati raccolti, fase 2), occorre indicizzare i documenti recuperati. L'indicizzazione deve consentire in modo efficiente di rispondere alle query e in particolare, l'indicizzazione deve permettere il ranking (classificazione) dei documenti. Il problema del ranking si può formulare nel seguente modo:

Definizione 2.2.1. *Dato un insieme $\mathcal{P}$ di pagine e una query Q, il ranking è una funzione*

$$r_q: \mathcal{P} \to \mathbb{R}$$

che associa ad ogni pagina un numero reale (rank), che indica il grado di rilevanza di quella pagina a fronte di quella query.

Il ranking è una componente integrale di ogni sistema software utilizzato per il recupero dell'informazione (Information Retrieval System). Un algoritmo di link analysis ranking ha lo scopo di analizzare i links contenuti all'interno delle pagine web del sito in esame, in modo da

associare ad esse un determinato punteggio detto appunto rank. Quindi il rank di una pagina non è altro che un indice di gradimento (un peso numerico) della pagina stessa. Quando si effettua una interrogazione (query) su un motore di ricerca, l'elenco di siti che si ottiene è ordinato in base al rank attribuito a ciascun di essi: più alto è il rank maggiore è la possibilità che un sito appaia tra le prime posizioni dell'elenco. In passato il processo del reperimento di informazione operato dai motori di ricerca prevedeva solamente un analisi dei contenuti testuali delle pagine, confrontandoli con la query formulata nell'interrogazione e verificando la presenza o meno di compatibilità. Si trattava di tecniche di ranking basate sull'analisi del contenuto testuale, come ad esempio Altavista, uno dei primi motori di ricerca veloci della rete. Il ranking moderno esamina una nuova categoria di informazioni rappresentata da dati estrapolati dal web, cioè si basa sull'analisi della struttura degli hyperlink, come per esempio il motore di ricerca Google.

Un algoritmo di link analysis ranking comincia con un insieme di pagine web. Il passo successivo è quello di costruire un grafo $G = (V, E)$ di collegamenti ipertestuali. Come abbiamo già detto nel Capitolo 1, un nodo rappresenta una singola pagina web mentre un arco diretto è posto tra due nodi se c'è un collegamento ipertestuale tra le due pagine web. Se ci sono collegamenti multipli tra due pagine viene posto un solo arco e non sono permessi loop. L'output di un algoritmo di link analysis è un vettore n-dimenensionale $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_i, \dots, \alpha_n)$, dove la i-esima componente rappresenta il peso attribuito al nodo i-esimo, cioè alla i-esima pagina del web. Tali pesi sono utilizzati per attribuire un determinato punteggio alle pagine web.

Citiamo due degli algoritmi di link analysis ranking che si svilupparono alla fine del secolo scorso, tralasciando Pagerank la cui trattazione dettagliata sarà fatta nel prossimo capitolo.

I. Algoritmo Indegree

II. Algoritmo HITS (Hyperlink Induced Topic Search)

I. Indegree è stato il capostipite di tutti gli algoritmi di analisi dei link, considerando una pagina web secondo la sua popolarità. Esso computa un valore di importanza per ciascun nodo prendendo in esame solamente il valore informativo derivante dalla quantità di link afferenti alla pagina stessa e provenienti da altri documenti web. In questo modo più link entranti presenta una pagina web, più elevato è il suo grado di importanza. Avremo che il rank della pagina i è

$$PR(i) = |indegree(i)|$$

II. Indipendentemente da Brin e Page, J. M. Kleinberg nel 1998 propose una definizione differente riguardo l'importanza di una pagina, utilizzando sia gli inlinks che gli outlinks, creando così due punteggi di popolarità per ogni pagina web. Una pagina web è considerata una "hub page" se contiene molti outlinks. Invece si parla di "authority page" se la pagina web contiene molti inlinks. Ovviamente una pagina può essere sia una "hub page" che una "auth page".

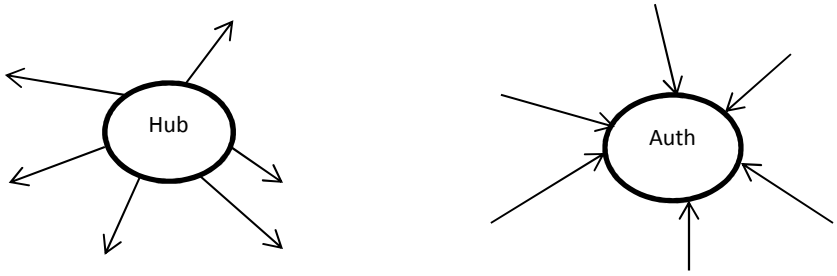

Figura 2.4 Schematizzazione “hub page” e “auth page”

HITS assegnerà ad ogni pagina web due punteggi detti “hub score” e “auth score” i quali si rafforzano a vicenda. Infatti, tale algoritmo concretizza la relazione di mutuo rinforzo:

Un buon “hub” rappresenta una pagina afferente a molte buone “authority”; una buona ”authority” identifica una pagina che viene puntata da molti buoni “hub”.

Ogni pagina i ha sia un authority score x_i, sia un hub score y_i. Sia E l’insieme di tutte le connessioni del grafo associato al web e sia $e_{i,j}$ l’arco diretto tra il nodo i e il nodo j. A ciascuna pagina è stato assegnato in qualche modo un’ authority score iniziale x_i^0 e un hub score iniziale y_i^0. HITS si basa sul raffinamento successivo dei punteggi x_i e y_i calcolando

$$x_i^{(k)} = \sum_{j:e_{j,i}\in E} y_j^{(k-1)}$$

$$y_i^{(k)} = \sum_{j:e_{i,j}\in E} x_j^{(k)} \qquad (2.2.1)$$

Queste sono le equazioni originali di Kleinberg, le quali possono essere scritte in forma matriciale sfruttando la matrice di adiacenza L del grafo diretto del web.
In notazione matriciale le equazioni (2.2.1) diventano

$$\boldsymbol{x}^{(k)} = \mathrm{L}^T\boldsymbol{y}^{(k-1)} \quad \text{e} \quad \boldsymbol{y}^{(k)} = \mathrm{L}\boldsymbol{x}^{(k)} \qquad (2.2.2)$$

dove $\boldsymbol{x}^{(k)}$ e $\boldsymbol{y}^{(k)}$ sono vettori colonna $n \times 1$, i quali rappresentano gli indicativi punteggi di authority e hub a ogni iterazione.
Questo conduce all’algoritmo iterativo per il calcolo di tali punteggi, definitivi rispettivamente $\boldsymbol{x}$ e $\boldsymbol{y}$. Le equazioni (2.2.2) possono essere semplificate con la sostituzione

$$\boldsymbol{x}^{(k)} = \mathrm{L}^T\mathrm{L}\boldsymbol{x}^{(k-1)} \quad \text{e} \quad \boldsymbol{y}^{(k)} = \mathrm{L}\mathrm{L}^T\boldsymbol{y}^{(k-1)} \qquad (2.2.4)$$

Riportiamo di seguito l'algoritmo HITS originale

Inizializzazione $\boldsymbol{y}^0 = \boldsymbol{e}$ dove $\boldsymbol{e}$ è un vettore colonna unitario

$$\boldsymbol{x}^{(k)} = \mathrm{L}^T \boldsymbol{y}^{(k-1)}$$

$$\boldsymbol{y}^{(k)} = \mathrm{L}\boldsymbol{x}^{(k)}$$

$k = k + 1$, si itera fino alla convergenza

Le nuove equazioni (2.2.4) definiscono il metodo delle potenze per calcolare l'autovalore dominante per le matrici $\mathrm{L}^T\mathrm{L}$ (detta "matrice authority") e $\mathrm{L}\mathrm{L}^T$ (detta "matrice hub"). Le matrici $\mathrm{L}^T\mathrm{L}$ e $\mathrm{L}\mathrm{L}^T$ sono simmetriche, semidefinite positive e non negative. In tal modo, i loro autovalori distinti $\{\lambda_1, \ldots, \lambda_k\}$, sono necessariamente reali e non negativi, con $\lambda_1 > \lambda_2 > \ldots > \lambda_k \geq 0$. In altre parole non è possibile avere autovalori multipli sul cerchio spettrale. Di conseguenza, il metodo delle potenze applicato all'algoritmo HITS converge, però non è garantita l'unicità del limite dei vettori authority e hub. Infatti, le matrici $\mathrm{L}^T\mathrm{L}$ e $\mathrm{L}\mathrm{L}^T$ non sono in generale irriducibili e quindi non è possibile sfruttare il teorema di Perron-Frobenius riguardante l'unicità dell'autovettore.

Esempio 2.1
Dato il grafico in figura 2.5

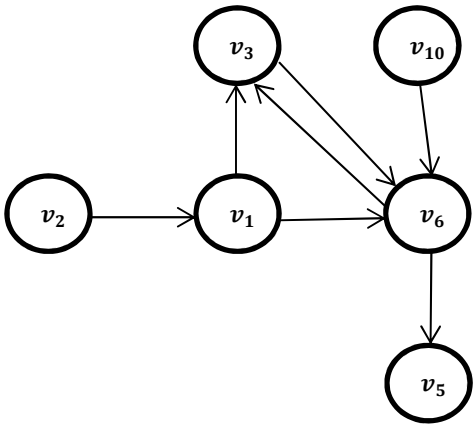

Figura 2.5 Grafo web di quattro pagine

La matrice di adiacenza associata al grafo di figura 2.4 è

$$\mathrm{L} = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_5 \\ v_6 \\ v_{10} \end{array} \begin{array}{c} \begin{array}{cccccc} v_1 & v_2 & v_3 & v_5 & v_6 & v_{10} \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

La “matrice authority” e la “matrice hub” sono rispettivamente

$$L^T L = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad LL^T = \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Gli autovettori normalizzati dei punteggi “authority” $\boldsymbol{x}$ e “hub” $\boldsymbol{y}$ sono

$$\boldsymbol{x}^T = (0 \quad 0 \quad 0.366 \quad 0.134 \quad 0.5 \quad 0)$$

$$\boldsymbol{y}^T = (0.366 \quad 0 \quad 0.2113 \quad 0 \quad 0.2113 \quad 0.2113)$$

Le pagine web della figura 2.5 saranno ordinate, in maniera decrescente, nei seguenti modi

Punteggio “authority” = $(v_6\ v_3\ v_5\ v_1\ v_2\ v_{10})$

Punteggio “hub” = $(v_1\ v_3\ v_6\ v_{10}\ v_2\ v_5)$

Capitolo 3

Trattazione matematica del Pagerank

PageRank è un algoritmo di analisi dei link che assegna un peso numerico ad ogni elemento di un collegamento ipertestuale di un insieme di documenti, come ad esempio il World Wide Web, con lo scopo di quantificare la sua importanza relativa all'interno della serie. Esso è una valore che Google assegna a ogni sito presente nel suo database. Tale algoritmo ha dimostrato una maggiore efficacia, rispetto a molti altri sistemi concorrenti, nell'attribuzione della rilevanza di un contenuto. Il peso numerico assegnato ad un dato elemento E è chiamato il pagerank di E. Esso può essere applicato a tutti gli insiemi di oggetti collegati da citazioni e riferimenti reciproci. Brin e Page, nel 1998, estesero l'idea dell'algoritmo Indegree, osservando che non tutti i collegamenti portano lo stesso peso. I collegamenti provenienti da pagine di alta qualità dovrebbero conferire una maggiore authority. Non è importante solamente il numero di link che puntano verso quella pagina ma anche se la qualità di tali collegamenti è alta o bassa. Il PageRank è facilmente riconducibile al concetto di popolarità tipico delle relazioni sociali umane e si ripromette di indicare le pagine o i siti di maggiore rilevanza in termini ricercati. Tale metodo può essere descritto come analogo ad una elezione nella quale ha diritto al voto chi può pubblicare una pagina web e il voto viene espresso attraverso i collegamenti in essa presenti. I voti non hanno tutti lo stesso peso: le pagine web più popolari esprimeranno, coi propri link, voti di valore maggiore. Il PageRank viene ricalcolato per tutti i siti indicizzati da Google ogni due-quattro mesi e quindi il PageRank non riflette mai la situazione attuale, ma quella di un periodo anteriore. Sinora abbiamo utilizzato solo parole per presentare tale algoritmo, ora invece traduciamo il discorso in equazioni matematiche. La traduzione rileverà che i punteggi del PageRank sono i valori stazionari di un enorme catena di Markov e molte proprietà interessanti di tale algoritmo saranno spiegate tramite la teoria di Markov.

3.1 Formula originale del Pagerank

Il PageRank di una pagina P_i, denotato con $r(P_i)$ è la somma dei PageRanks di tutte le pagine che puntano a P_i:

$$r(P_i) = \sum_{P_j \in B_{P_i}} \frac{r(P_j)}{|P_j|} \tag{3.1.1}$$

dove B_{P_i} è l'insieme delle pagine che puntano a P_i e $|P_j|$ è il numero di outlinks della pagina P_j. Il problema dell'equazione (3.1.1) è che i valori $r(P_j)$ sono sconosciuti. Per ovviare a questo problema Brin e Page usarono una procedura iterativa che agisce nel seguente modo: siano $P_1, P_2, \dots, P_n$ le pagine presenti nel web e supponiamo inizialmente che ogni pagina abbia PageRank uguale, diciamo $\frac{1}{n}$, dove n è il numero di pagine. Poniamo

$$r_0(P_i) = \frac{1}{n}$$

e calcoliamo ricorsivamente il nuovo rank

$$r_{k+1}(P_i) = \sum_{P_j \in B_{P_i}} \frac{r_k(P_j)}{|P_j|} \tag{3.1.2}$$

dove $r_{k+1}(P_i)$ è il pagerank della pagina P_i alla k+1-esima iterazione.

Tale processo è avviato con valore inziale $r_0(P_i) = \frac{1}{n}$ per ogni pagina P_i e si ripete con la speranza che converga a qualche valore finale stabile. L'equazione (3.1.2) può essere scritta in forma matriciale. Introduciamo un vettore riga $1 \times n$, $\boldsymbol{\pi}^T$, e una matrice $\mathrm{H}_{n\times n}$ con

$$\boldsymbol{\pi}^{(k)T} = (r_k(P_1), \dots, r_k(P_n))$$

e H matrice normalizzata per righe tale che

$\mathrm{H}_{i,j} = \frac{1}{|P_i|}$ se c'è un collegamento tra il nodo i e il nodo j

$\mathrm{H}_{\mathrm{i,j}} = 0$ altrimenti

Sebbene H abbia la stessa struttura della matrice binaria adiacente di un grafo, definizione 1.1.9, i suoi elementi non nulli sono probabilità. Infatti l'elemento $h_{i,j}$ della matrice rappresenta la probabilità che la pagina P_i sia connessa alla pagina P_j. Ricordiamo ora cosa si intende per limite di una matrice, definizione che ci occorre per poter definire il vettore PageRank.

Definizione 3.1.1. *Sia* A_i *,* $i = 0, 1, \dots$ *una successione di matrici* $n \times m$ *a elementi in* $\mathbb{K}$ *, con* $\mathbb{K}$ *campo reale o complesso. Indichiamo con* $a^i_{h,k}$ *l'elemento di posto* h, k *in* A_i*. Diremo che la successione di matrici* $\{\mathrm{A}_i\}$ *converge alla matrice* A*, e scriveremo*

$$\lim_{i\to\infty} \mathrm{A}_i = \mathrm{A}$$

Se, per $i \to \infty$, $a^i_{h,k} \to a_{h,k}$ *per ogni h e k.*

Tenendo presente tale definizione, possiamo dire che: se per $k \to \infty$, esiste

$$\lim_{k\to\infty} \boldsymbol{\pi}^{(k)T} = \boldsymbol{\pi}^T$$

allora $\boldsymbol{\pi}^T$ è detto vettore PageRank.

Usando la notazione matriciale, l'equazione (3.1.2) assume la forma:

$$\boldsymbol{\pi}^{(k+1)T} = \boldsymbol{\pi}^{(k)T}\mathrm{H} \qquad (3.1.3)$$

Osservazione 3.1.1.

1) La forma del processo iterativo dell'equazione (3.1.3) è studiata in analisi numerica e rappresenta il metodo delle potenze applicato alla matrice H.

2) Se non ci fossero pagine pozzo nel grafo associato alla matrice H, che rendono le righe corrispondenti nulle, tale matrice rivestirebbe il ruolo di matrice di transizione, mentre ciascuna pagina rappresenterebbe uno stato della catena di Markov. Inoltre risulterebbe soddisfatta la proprietà di Markov, in quanto la probabilità di essere nella pagina web *j* al passo $n+1$, dipende esclusivamente da dove ci si trova al passo n.

Esempio 3.1.

Consideriamo il piccolo web della figura sottostante:

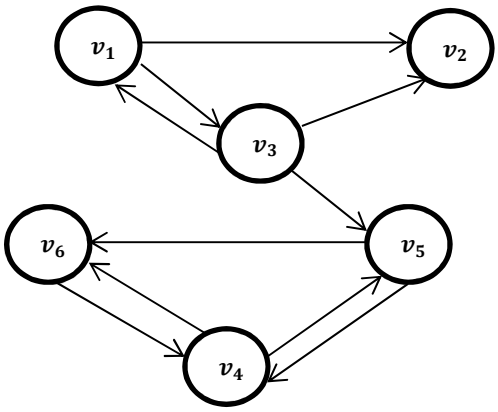

Figura 3.1 Grafo diretto che rappresenta un web di sei pagine

Applicando l'equazione (3.1.3) sino alla seconda iterazione, si hanno i seguenti valori:

Iterazione 0	Iterazione 1	Iterazione 2	Rank all'interazione 2^a
$r_0(P_1) = 1/6$	$r_1(P_1) = 1/18$	$r_2(P_1) = 1/36$	5
$r_0(P_2) = 1/6$	$r_1(P_2) = 5/36$	$r_2(P_2) = 1/18$	4
$r_0(P_3) = 1/6$	$r_1(P_3) = 1/12$	$r_2(P_3) = 1/36$	5
$r_0(P_4) = 1/6$	$r_1(P_4) = 1/4$	$r_2(P_4) = 17/72$	1
$r_0(P_5) = 1/6$	$r_1(P_5) = 5/36$	$r_2(P_5) = 11/72$	3
$r_0(P_6) = 1/6$	$r_1(P_6) = 1/6$	$r_2(P_6) = 14/72$	2

La matrice del grafo è:

$$H = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

L'equazione (3.1.3) solleva diverse problematiche:

1. Tale processo iterativo continua indefinitamente o converge?
2. Sotto quali condizioni e proprietà della matrice H è garantita la convergenza?
3. La convergenza dipende dalla scelta del vettore iniziale $\boldsymbol{\pi}^{(0)T}$?

Brin e Page originariamente iniziarono il processo iterativo con

$$\boldsymbol{\pi}^{(0)T} = \frac{1}{n}\boldsymbol{e}^T$$

dove $\boldsymbol{e}^T$ è un vettore riga unitario. Tale scelta crea una serie di problemi quando si va a utilizzare l'equazione (3.1.3) con quel vettore iniziale. Ad esempio, si verifica il così detto problema dei pozzi (rank-sinks), ovvero i pozzi sono quelle pagine che hanno outdegree zero, cioè pagine che non hanno archi uscenti. I pozzi accumulano PageRank ad ogni iterazione senza condividere, in tal modo altri nodi potrebbero trovarsi con un PageRank nullo dopo un tot di iterazioni.

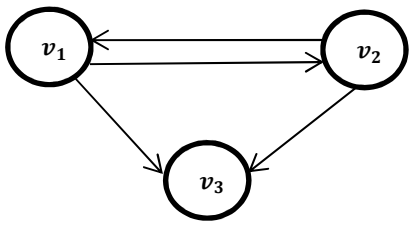

Figura 3.2 Grafo semplice con un pozzo

Nell'esempio 3.1 il gruppo di nodi v_4, v_5 e v_6 aspira ad accumulare PageRank e ciò può accadere involontariamente oppure con intento. Infatti, dopo 13 iterazioni il vettore $\boldsymbol{\pi}^{(k)T}$ risulta essere

$$\boldsymbol{\pi}^{(13)T} = (0,0,0,\tfrac{2}{3},\tfrac{1}{3},\tfrac{1}{5})$$

C'è anche un altro problema che potrebbe verificarsi ed è quello dei cicli. Consideriamo il caso della figura 3.3. Queste due pagine creano un ciclo infinito in quanto la pagina 1 punta soltanto alla pagina 2 e viceversa. Infatti, se supponiamo che il processo iterativo dell'equazione (3.1.3) inizi con vettore $\boldsymbol{\pi}^{(0)T} = (1,0)$, le iterazioni successive non convergeranno mai, in quanto l'iterazione k-esima $\boldsymbol{\pi}^{(k)T}$ oscillerà tra $(1,0)$ quando k è pari e tra $(0,1)$ quando k è dispari.

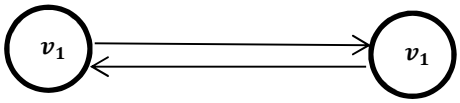

Figura 3.3 Grafo semplice con un ciclo

3.2 Le prime modifiche apportate al modello

L'equazione (3.1.3) assomiglia al metodo delle potenze applicato a una catena di Markov con matrice di transizione H. Questa osservazione è molto utile perché ci permette di applicare la teoria delle catene di Markov al problema del PageRank, in modo tale da correggere l'equazione (3.1.3) e assicurarne la convergenza. In particolare, per ogni vettore iniziale il metodo delle potenze, applicato a una matrice di Markov P, converge a un unico vettore positivo, chiamato vettore stazionario, fintanto che P è stocastica, irriducibile e primitiva. I problemi causati dai pozzi e dai cicli possono essere superati modificando la matrice H.

Nel 1998 Brin e Page apportarono le prime modifiche al loro modello. È interessante notare che in nessuno dei loro documenti c'è il riferimento alle catene di Markov, anche se oggi essi sono perfettamente consci della forte connessione tra il modello PageRank e la teoria di Markov. Essi, per descrivere le modifiche, utilizzarono la nozione di navigatore casuale (random surfer). Immaginiamo che un navigatore virtuale stia facendo una passeggiata aleatoria sul grafo del web, scelga a caso un link della pagina in cui si trova e che continui tale processo indefinitamente. Tale processo incontra dei problemi quando egli capita in un pozzo, in quanto è incapace di muoversi. Per ovviare a ciò la matrice H viene modificata sostituendo le sue righe di vettori nulli, $\mathbf{0}^T$, con il vettore $\frac{1}{n}\boldsymbol{e}^T$, dove $\boldsymbol{e}^T$ è un vettore riga unitario. In tal modo la matrice H diventa una matrice stocastica, denominiamola S, dove risulta:

$$\mathrm{S} = \mathrm{H} + \boldsymbol{a}\,(\frac{1}{n}\boldsymbol{e}^T)$$

con $\boldsymbol{a}$ vettore colonna binario tale che la sua i-esima componente, a_i, valga 1 se la pagina i-esima è un pozzo, altrimenti 0.

Essendo una matrice stocastica, S rappresenta la matrice di transizione di una catena di Markov. In tal modo si da al navigatore la possibilità di uscire dalla situazione di stallo collegandosi a una qualsiasi altra pagina della rete. Non siamo ancora in grado di affermare la convergenza dell'equazione (3.1.3), cioè che esista e sia unico un vettore positivo $\boldsymbol{\pi}^T$ tale che l'equazione (3.1.3) converga a esso. Con un' ulteriore modifica, Brin e Page trasformarono la matrice S in una matrice primitiva. In tal modo il vettore stazionario della catena (che nel nostro caso è il vettore PageRank), esiste ed è unico e può essere trovato facilmente con un iterazione basata sul metodo delle potenze. Sempre riferendosi a un navigatore casuale, egli segue la struttura di collegamenti ipertestuali sul web ma a volte, annoiandosi, potrebbe abbandonare tale metodo di navigazione ed entrare in una nuova URL del browser, cominciando così una nuova navigazione ipertestuale. Per modellizzare matematicamente ciò, Brin e Page inventarono una nuova matrice G tale che

$$\mathrm{G} = \alpha \mathrm{S} + (1-\alpha)\frac{1}{n}\boldsymbol{e}\boldsymbol{e}^T \qquad (3.2.1)$$

dove α è uno scalare, $\alpha \in (0,1)$. La matrice G si chiama matrice di Google. La formula (3.2.1) può essere anche espressa nella forma

$$\mathrm{G} = \alpha \mathrm{S} + (1-\alpha)\mathrm{E} \qquad (3.2.2)$$

dove la matrice

$$\mathrm{E} = \frac{1}{n}\boldsymbol{e}\boldsymbol{e}^T$$

si chiama matrice di perturbazione.

Lo scalare α è la proporzione di tempo che il navigatore impiega navigando tramite i collegamenti ipertestuali. Per esempio se $\alpha = 0.6$ significa che il 60% del tempo il navigatore segue la struttura ipertestuale e il restante 40% si trasferisce in un'altra pagina. Il trasferimento è casuale perché la matrice di perturbazione è uniforme. Ciò significa che il navigatore ha la stessa probabilità di approdare a una qualunque pagina durante il trasferimento.

L'ulteriore modifica della matrice H conduce a diverse conseguenze:

1) G è stocastica. Essa è combinazione lineare di due matrici stocastiche S e E.

2) G è irriducibile. Infatti abbiamo introdotto archi in tutte le pagine e quindi ognuna di esse è direttamente connessa ad ogni altra pagina, in tal modo l'irriducibilità è banalmente verificata.

3) G è primitiva. $\mathrm{G}^k > 0$ per qualche k, e infatti per $k = 1$ $\mathrm{G}^1 > 0$.

Ciò implica che esiste un unico vettore positivo $\boldsymbol{\pi}^T$, e il metodo delle potenze applicato alla matrice G ci garantisce la convergenza a tale vettore.

In definitiva la formula (3.1.3) diventa

$$\boldsymbol{\pi}^{(k+1)T} = \boldsymbol{\pi}^{(k)T}\mathrm{G}$$

che è semplicemente il metodo delle potenze applicato alla matrice G.

Osservazione 3.2.1. La matrice G è artificiale, nel senso che per ottenerla H è stata modificata due volte al fine di produrre le proprietà desiderate di convergenza. Un vettore stazionario per H non esiste, così Brin e Page la modificarono per ottenere i risultati sperati.

La matrice di perturbazione E, introdotta da Brin e Page, subì un cambiamento. Invece che usare

$$\mathrm{E} = \frac{1}{n}\boldsymbol{e}\boldsymbol{e}^T$$

Brin e Page posero

$$\mathrm{E} = \boldsymbol{e}\boldsymbol{v}^T$$

dove $\boldsymbol{v}^T > 0$ è un vettore di probabilità, chiamato vettore di perturbazione.

Usare $\boldsymbol{v}^T$, al posto di $\frac{1}{n}\boldsymbol{e}^T$, significa che il navigatore non ha più la stessa probabilità di approdare a una pagina web durante il trasferimento. Ogniqualvolta il navigatore si trasferisce in un'altra pagina, egli segue la distribuzione di probabilità data dal vettore $\boldsymbol{v}^T$. La matrice di Google diventa:

$$\mathrm{G} = \alpha \mathrm{S} + (1-\alpha)\boldsymbol{e}\boldsymbol{v}^T \tag{3.2.3}$$

Tornando all'esempio 3.1, la matrice stocastica S è

$$\mathrm{S} = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Ponendo $\alpha = 0{,}9$ la matrice primitiva G è:

$$\mathrm{G} = 0{,}9\mathrm{H} + \left(0{,}9 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + 0{,}1 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right) \frac{1}{6}(1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1) =$$

$$= \begin{pmatrix} 1/60 & 7/15 & 7/15 & 1/60 & 1/60 & 1/60 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 19/60 & 19/60 & 1/60 & 1/60 & 19/60 & 1/60 \\ 1/60 & 1/60 & 1/60 & 1/60 & 7/15 & 7/15 \\ 1/60 & 1/60 & 1/60 & 7/15 & 1/60 & 7/15 \\ 1/60 & 1/60 & 1/60 & 11/12 & 1/60 & 1/60 \end{pmatrix}$$

Utilizzando ad esempio il comando "eig" di Matlab, si trova che il vettore PageRank di Google, per il piccolo web dell'esempio 3.1, è il vettore stazionario della matrice G ed è dato da:

$$\boldsymbol{\pi}^T = \begin{pmatrix} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \\ 0{,}03721 & 0{,}05396 & 0{,}04151 & 0{,}3751 & 0{,}206 & 0{,}2862 \end{pmatrix}$$

L'interpretazione di $\pi^1 = 0.03721$, prima componente del vettore $\boldsymbol{\pi}^T$, è che per il 3,721% del tempo il navigatore sosta nella pagina 1. Inoltre, le pagine nel piccolo web dell'esempio 3.1 possono essere classificate in base alla loro importanza come: $(v_4, v_6, v_5, v_2, v_3, v_1)$, significa che P_4 è la pagina più importante, quella che ha il più alto valore di pagerank. La pagina P_4 sarà quella che verrà visualizzata per prima dall'indicizzazione di Google.

3.3 Calcolo del vettore PageRank

Il problema del PageRank può essere formulato in due modi:

1) Risolvere il problema agli autovalori per $\boldsymbol{\pi}^T$:

$$\boldsymbol{\pi}^T = \boldsymbol{\pi}^T \mathrm{G}$$
$$\boldsymbol{\pi}^T \boldsymbol{e} = 1$$

L'obiettivo è di trovare l'autovettore sinistro dominante normalizzato di G corrispondente all'autovalore dominante λ_1=1 (G è stocastica, ciò implica λ_1=1)

2) Risolvere il seguente sistema lineare omogeneo:

$$\boldsymbol{\pi}^T (\mathrm{I} - \mathrm{G}) = \mathbf{0}^T$$
$$\boldsymbol{\pi}^T \boldsymbol{e} = 1$$

In entrambi i casi $\boldsymbol{\pi}^T \boldsymbol{e} = 1$ è un' equazione di normalizzazione, assicura che $\boldsymbol{\pi}^T$ sia un vettore di probabilità.

Soffermiamoci sul caso 1). Abbiamo già detto che $\boldsymbol{\pi}^T$ è il vettore stazionario di una catena di Markov con matrice di transizione G e molte ricerche sono state fatte per calcolare tale vettore. Brin e Page, invece, utilizzarono un metodo di analisi numerica, il metodo delle potenze. Tale metodo applicato a G può essere espresso in termini della matrice sparsa H

$$\boldsymbol{\pi}^{(k+1)T} = \boldsymbol{\pi}^{(k)T} \mathrm{G} = \alpha \boldsymbol{\pi}^{(k)T} \mathrm{S} + \frac{(1-\alpha)}{n} \boldsymbol{\pi}^{(k)T} \boldsymbol{e}\boldsymbol{e}^T =$$

$$= \alpha \boldsymbol{\pi}^{(k)T} \mathrm{H} + (\alpha \boldsymbol{\pi}^{(k)T} \boldsymbol{a} + 1 - \alpha) \frac{\boldsymbol{e}^T}{n}$$

Tra tutti i metodi iterativi disponibili, il metodo delle potenze è il più lento, ma nonostante ciò fu scelto ugualmente dai due fondatori di Google per la sua semplicità e per la sua velocità di convergenza molto rapida. Infatti sono necessarie circa 50-100 iterazioni prima di giungere alla convergenza. Ci si chiede come mai il metodo delle potenze applicato a G richieda così poche iterazioni per convergere e se ciò sia dovuto alla struttura della matrice G. La risposta viene fornita dalla teoria delle catene di Markov. In generale il tasso asintotico di convergenza del metodo delle potenze applicato a una matrice dipende dal rapporto dei due autovalori maggiori, λ_1 e λ_2. Noi sappiamo che

$$\left|\frac{\lambda_2}{\lambda_1}\right|^k \to 0$$

Per le matrici stocastiche $\lambda_1 = 1$, quindi la convergenza dipende solo da $|\lambda_2|$. Poiché G è anche primitiva $|\lambda_2| < 1$. In generale, trovare numericamente λ_2 per una matrice richiede sforzi computazionali che uno non è disposto a spendere solo per ottenere una stima del tasso asintotico di convergenza. Fortunatamente, per il problema del PageRank, è facile mostrare il seguente teorema

Teorema 3.3.1. *Se lo spettro della matrice stocastica* S *è* $\{1, \lambda_2, \dots, \lambda_n\}$*, allora lo spettro della matrice di Google,* $\mathrm{G} = \alpha \mathrm{S} + (1-\alpha)\boldsymbol{e}\boldsymbol{v}^T$*, è* $\{1, \alpha\lambda_2, \dots, \alpha\lambda_n\}$.

Dimostrazione. Poiché S è stocastica, $\boldsymbol{e} = (1, \dots, 1)^T$ è l'autovettore relativo all'autovalore 1. Sia $\mathrm{Q} = (\boldsymbol{e}\,,\ \mathrm{X})$ una matrice non singolare che ha come prima colonna l'autovettore $\boldsymbol{e}$**.** Sia $\mathrm{Q}^{-1} = \begin{pmatrix} \boldsymbol{y}^T \\ \mathrm{Y}^T \end{pmatrix}$. Allora

$$Q^{-1}Q = \begin{pmatrix} \boldsymbol{y}^T\boldsymbol{e} & \boldsymbol{y}^T X \\ Y^T\boldsymbol{e} & Y^T X \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & I \end{pmatrix}$$

Quest'ultima uguaglianza ci fornisce due utili relazioni, che sono $\boldsymbol{y}^T\boldsymbol{e} = 1$ e $Y^T\boldsymbol{e} = 0$.

Adesso consideriamo la trasformazione di similitudine

$$Q^{-1}SQ = \begin{pmatrix} \boldsymbol{y}^T\boldsymbol{e} & \boldsymbol{y}^T SX \\ Y^T\boldsymbol{e} & Y^T SX \end{pmatrix} = \begin{pmatrix} 1 & \boldsymbol{y}^T SX \\ 0 & Y^T SX \end{pmatrix}$$

Essa rivela che $Y^T SX$ contiene i rimanenti autovalori di S, $\lambda_2, \dots, \lambda_n$. Applichiamo la trasformazione di similitudine alla matrice $G = \alpha S + (1-\alpha)\boldsymbol{e}\boldsymbol{v}^T$

$$Q^{-1}(\alpha S + (1-\alpha)\boldsymbol{e}\boldsymbol{v}^T)Q = \alpha Q^{-1}SQ + (1-\alpha)Q^{-1}\boldsymbol{e}\boldsymbol{v}^T Q =$$

$$= \begin{pmatrix} \alpha & \alpha\boldsymbol{y}^T SX \\ 0 & \alpha Y^T SX \end{pmatrix} + (1-\alpha)\begin{pmatrix} \boldsymbol{y}^T\boldsymbol{e} \\ Y^T\boldsymbol{e} \end{pmatrix}(\boldsymbol{v}^T\boldsymbol{e} \quad \mathbf{v}^T X) =$$

$$= \begin{pmatrix} \alpha & \alpha\boldsymbol{y}^T SX \\ 0 & \alpha Y^T SX \end{pmatrix} + \begin{pmatrix} (1-\alpha) & (1-\alpha)\boldsymbol{v}^T X \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & \alpha\mathbf{y}^T SX + (1-\alpha)\boldsymbol{v}^T X \\ 0 & \alpha Y^T SX \end{pmatrix}$$

Perciò gli autovalori di G sono $\{1, \alpha\lambda_2, \dots, \alpha\lambda_n\}$.

■

L'enunciato del teorema 3.3.1 equivale ad affermare che

Se $Sp(S) = \{1, \mu_2, \dots, \mu_n\}$ e $Sp(G) = \{1, \lambda_2, \dots, \lambda_n\}$ allora $\lambda_k = \alpha\mu_k \ \ per \ \ k = 2,3, \dots, n$

La struttura a hyperlink del web rende molto probabile che $|\mu_2| \simeq 1$. Ciò significa che $|\lambda_2(G)| \simeq \alpha$. Quindi si ha come risultato che il parametro α interviene nella convergenza dell'algoritmo. Nelle loro trattazioni Brin e Page usano $\alpha = 0{,}85$, e tale valore è tutt'oggi usato da Google. Il parametro $\alpha \in (0,1)$ viene denominato fattore di smorzamento (damping factor). Ci si può chiedere come mai Brin e Page abbiano scelto proprio il valore $\alpha = 0{,}85$. Come già detto, α controlla il tasso di convergenza asintotico del metodo delle potenze del PageRank. Se $\alpha \to 1$ il numero di iterazioni previste cresce notevolmente, come ripotato in tabella 3.1.

α	Numero di iterazioni
0,5	34
0,75	81
0,85	104
0,9	219
0,99	2.292
0,999	23015

Tabella 3.1 Numero di iterazioni previste al crescere di α

La scelta $\alpha = 0{,}85$ non è ottimale come numero di iterazioni previste, ciò significa che Page e Brin furono costretti a eseguire un delicato atto di bilanciamento , come $\alpha \to 1$ l'artificiosità introdotta dalla matrice di perturbazione E si riduce, ma il tempo di calcolo aumenta. Sembra che, ponendo $\alpha = 0{,}85$, si trovi un compromesso accettabile tra efficienza e efficacia. È

interessante notare che la costante α controlla molto di più della sola convergenza. Essa riguarda anche la sensibilità del vettore risultante PageRank. Infatti come $\alpha \to 1$ i punteggi PageRank diventano molto più volatili e oscillano notevolmente anche per piccoli cambiamenti nella struttura del web.

3.4 Analisi di sensibilità dell'algoritmo

La sensibilità del vettore PageRank può essere analizzata esaminando separatamente ciascun parametro della matrice di Google G (3.2.3). Abbiamo visto che essa dipende da tre parametri: il fattore di smorzamento α, la matrice H, e il vettore di perturbazione $\boldsymbol{v}^T$.

1. Modifica del parametro α

   Useremo la derivata per mostrare gli effetti delle variazioni di α su $\boldsymbol{\pi}^T$. La derivata di $\boldsymbol{\pi}^T$ rispetto ad α, denotiamola con

   $$\frac{d\boldsymbol{\pi}^T(\alpha)}{d\alpha} = \left(\frac{d\pi_1}{d\alpha}, \ldots, \frac{d\pi_n}{d\alpha}\right) \tag{3.4.1}$$

   ci dice quanto gli elementi del vettore $\boldsymbol{\pi}^T$ variano, quando α subisce delle leggere variazioni. Il segno della derivata ci da informazioni importanti:

   - se $\frac{d\pi_j(\alpha)}{d\alpha} > 0$ allora piccoli aumenti di α implicano che il PageRank di P_j aumenterà.
   - se $\frac{d\pi_j(\alpha)}{d\alpha} < 0$ allora piccoli aumenti di $\propto$ implicano che il PageRank di P_j diminuirà.

   La (3.4.1) ci fornisce il tasso di variazione di $\boldsymbol{\pi}^T(\alpha)$ rispetto a piccole variazioni del parametro α. È importante sottolineare che la (3.4.1) è solo un'approssimazione di quanto gli elementi di $\boldsymbol{\pi}^T$ cambiano al variare di α. La matrice G dipende da α per la (3.2.3). Per capire quanto sia sensibile $\boldsymbol{\pi}^T(\alpha)$ per cambiamenti del parametro α, dobbiamo essere certi che la derivata (3.4.1) sia ben definita. Non è detto infatti che le componenti di un autovettore siano funzioni differenziabili. Il problema viene risolto con il seguente teorema

   Teorema 3.4.1. *Il vettore PageRank è dato da*

   $$\boldsymbol{\pi}^T(\alpha) = \frac{1}{\sum_{i=1}^{n} D_i(\alpha)} (D_1(\alpha), \ldots, D_n(\alpha))$$

   dove $D_i(\alpha)$ è il determinante dell' i-esimo minore principale di ordine $n-1$ di $\mathrm{I} - \mathrm{G}(\alpha)$. Poiché ciascun $D_i(\alpha) > 0$ è una somma di prodotti di elementi di $\mathrm{I} - \mathrm{G}(\alpha)$, segue che ciascuna componente di $\boldsymbol{\pi}^T(\alpha)$ è una funzione differenziabile di α nell'intervallo (0,1).

Dimostrazione Viene ommessa. Si veda [3], Capitolo 6, pag.66.

Teorema 3.4.2. *Se $\boldsymbol{\pi}^T(\alpha)$ è il vettore PageRank associato alla matrice di Google*

$$\mathrm{G}(\alpha) = \alpha\mathrm{S} + (1-\alpha)\boldsymbol{e}\boldsymbol{v}^T$$

Allora

$$\frac{d\boldsymbol{\pi}^T(\alpha)}{d\alpha} = -\boldsymbol{v}^T(\mathrm{I}-\mathrm{S})(\mathrm{I}-\alpha\mathrm{S})^{-2}$$

Dimostrazione. Consideriamo

$$\mathbf{0}^T = \boldsymbol{\pi}^T(\alpha)(\mathrm{I}-\mathrm{G}) = \boldsymbol{\pi}^T(\alpha)(\mathrm{I}-\alpha\mathrm{S}-(1-\alpha)\boldsymbol{e}\boldsymbol{v}^T) \qquad (3.4.2)$$

La quantità $(\mathrm{I}-\alpha\mathrm{S})$ è non singolare perché[2]

$$\alpha < 1 \Longrightarrow \rho(\alpha\mathrm{S}) < 1 \Longrightarrow \exists\ (\mathrm{I}-\alpha\mathrm{S})^{-1}$$

Moltiplichiamo la relazione (3.4.2) a destra per $(\mathrm{I}-\alpha\mathrm{S})^{-1}$, ottenendo

$$\mathbf{0}^T = \boldsymbol{\pi}^T(\alpha)(\mathrm{I}-(1-\alpha)\boldsymbol{e}\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}) \Longrightarrow \boldsymbol{\pi}^T(\alpha) = (1-\alpha)\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}$$

Utilizzando la formula[3]

$$\frac{d\mathrm{A}(\alpha)^{-1}}{d\alpha} = -\mathrm{A}^{-1}(\alpha)\left[\frac{d\mathrm{A}(\alpha)}{d\alpha}\right]\mathrm{A}^{-1}(\alpha)$$

dove A è una matrice le cui entrate sono funzioni differenziabili di α e il fatto che $(\mathrm{I}-\mathrm{S})$ commuta con $(\mathrm{I}-\alpha\mathrm{S})^{-1}$ avremo

$$\begin{aligned}\frac{d\boldsymbol{\pi}^T(\alpha)}{d\alpha} &= (1-\alpha)\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}\mathrm{S}(\mathrm{I}-\alpha\mathrm{S})^{-1} - \boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1} = \\ &= -\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}[\mathrm{I}-(1-\alpha)\mathrm{S}(\mathrm{I}-\alpha\mathrm{S})^{-1}] = \\ &= -\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}(\mathrm{I}-\alpha\mathrm{S}-(1-\alpha)\mathrm{S})(\mathrm{I}-\alpha\mathrm{S})^{-1} = \\ &= -\boldsymbol{v}^T(\mathrm{I}-\alpha\mathrm{S})^{-1}(\mathrm{I}-\mathrm{S})(\mathrm{I}-\alpha\mathrm{S})^{-1} = -\boldsymbol{v}^T(\mathrm{I}-\mathrm{S})(\mathrm{I}-\alpha\mathrm{S})^{-2}\end{aligned}$$

■

In particolare, sfruttando il teorema 3.4.1 si ottengono

$$\lim_{\alpha\to 0}\frac{d\boldsymbol{\pi}^T(\alpha)}{d\alpha} = -\boldsymbol{v}^T(\mathrm{I}-\mathrm{S})$$

[2] Carl D. Meyer, *Matrix Analysis and Applied Algebra*, Capitolo 7, pag. 618.
[3] Carl D. Meyer *Matrix Analysis and Applied Linear Algebra*, Capitolo 3, pag. 130.

$$\lim_{\alpha \to 1} \frac{d\boldsymbol{\pi}^T(\alpha)}{d\alpha} = -\boldsymbol{v}^T(\mathrm{I} - \mathrm{S})^{\#}$$

dove $(\cdot)^{\#}$ denota l'inversa (nel senso dei gruppi), cioè data una matrice $\mathrm{A}_{n\times n}$ reale, l'inversa di A è una matrice X tale che

$$\begin{aligned} &\text{I.} && \mathrm{AXA} = \mathrm{A} \\ &\text{II.} && \mathrm{XAX} = \mathrm{X} \\ &\text{III.} && \mathrm{AX} = \mathrm{XA} \end{aligned}$$

Quando esiste una tale matrice essa è unica e si denota con $\mathrm{A}^{\#}$.
Quindi al tendere di α a 1, la sensibilità di $\boldsymbol{\pi}^T$ è governata dalle entrate della matrice $(\mathrm{I} - \mathrm{S})^{\#}$. Per piccoli valori di α, $\alpha \to 0$, si sfrutta la seguente disuguaglianza[4]

$$\left|\frac{d\pi_j(\alpha)}{d\alpha}\right| \leq \frac{1}{1-\alpha} \qquad (3.4.3)$$

dove $\boldsymbol{\pi}^T(\alpha) = (\pi_1(\alpha), \dots, \pi_2(\alpha))$ è il vettore PageRank. L'utilità della (3.4.3) è limitata solo a piccoli valori di α. Essa ci assicura che i PageRanks non sono eccessivamente sensibili come funzioni del parametro α. I valori grandi di α sono quelli che hanno maggiore interesse, perché sono quelli che danno peso alla struttura di collegamenti ipertestuali del web, mentre, per valori di α piccoli aumenta l'influenza del vettore di perturbazione $\boldsymbol{v}^T$.

2. Modifica della matrice H
   Ci chiediamo quanto sia sensibile $\boldsymbol{\pi}^T$ a cambiamenti della matrice H. Alcuni risultati[5] hanno provato che per una catena di Markov con matrice di transizione P e vettore stazionario $\boldsymbol{\pi}^T$

$$\boldsymbol{\pi}^T \text{ è sensibile a cambiamenti della matrice P} \iff |\lambda_2(\mathrm{P})| \simeq 1$$

Per il problema di PageRank, sappiamo che $|\lambda_2(\mathrm{G})| \leq \alpha$. Perciò, come $\alpha \to 1$, il vettore PageRank diventa sempre più sensibile ai cambiamenti della matrice G. Poiché G dipende da α, H, e $\boldsymbol{v}^T$ vorremmo isolare gli effetti della matrice H sulla sensibilità di $\boldsymbol{\pi}^T$. Allora riadoperiamo la derivata

$$\frac{d\boldsymbol{\pi}^T(h_{i,j})}{dh_{i,j}} = \alpha\pi_i(\boldsymbol{e}_j^T - \boldsymbol{v}^T)(\mathrm{I} - \alpha\mathrm{S})^{-1}$$

L'effetto di α è chiaro, come $\alpha \to 1$ gli elementi di $(\mathrm{I} - \alpha\mathrm{S})^{-1}$ tendono a ∞. In tal modo il vettore PageRank è più sensibile a piccoli cambiamenti della struttura del grafo web.

[4] Amy N. Langville & Carl D. Mayer *Google's PageRank and beyond*, Capitolo 6, pag. 66.
[5] Carl D. Meyer *The character of a finite Markov chain*,

3. Modifica del vettore di perturbazione $\boldsymbol{v}^T$
   Calcoliamo anche in questo caso la derivata di $\boldsymbol{\pi}^T$ ma, stavolta, rispetto a $\boldsymbol{v}^T$.

$$\frac{d\boldsymbol{\pi}^T(\boldsymbol{v}^T)}{d\boldsymbol{v}^T} = (1 - \alpha + \alpha \sum_{i\in D} \pi_i)(\mathrm{I} - \alpha \mathrm{S})^{-1} \qquad (3.4.4)$$

   dove D è l'insieme dei pozzi. Innanzitutto nell'equazione (3.4.2) c'è la dipendenza da α. Come nel caso 2., quando $\alpha \to 1$ gli elementi di $(\mathrm{I} - \alpha \mathrm{S})^{-1}$ tendono a ∞, quindi $\boldsymbol{\pi}^T$ diventa sempre più sensibile al tendere di α a 1. Inoltre se le pagine pozzo sono combinate assieme in modo tale da avere una porzione di PageRank alta (cioè è grande il valore $\sum_{i\in D} \pi_i$), allora il vettore $\boldsymbol{\pi}^T$ è più sensibile nei confronti del vettore $\boldsymbol{v}^T$. Infatti se l'insieme delle pagine pozzo ha un importanza considerevole, allora il navigatore casuale rivisiterà tali pagine molto spesso e in tal modo opterà frequentemente per un indirizzo URL differente, cioè modificando il vettore $\boldsymbol{v}^T$. Quindi le azioni del navigatore casuale, e quindi il vettore PageRank, sono sensibili ai cambiamenti del vettore di perturbazione $\boldsymbol{v}^T$.

3.5 Codice Matlab per il calcolo del PageRank di un sito web

Tramite il linguaggio Matlab calcoliamo il vettore PageRank partendo dal sito della Microsoft: http://www.microsoft.com. La funzione surfer ci permette di scaricare una porzione della rete, immettendo una pagina da cui partire e il numero di nodi da considerare. Come output otteniamo il vettore $\boldsymbol{U}$, che contiene gli indirizzi dei siti visitati e H la matrice sparsa, il cui generico elemento $h_{i,j}$ vale 1 se la pagina $\boldsymbol{U}(j)$ punta verso la pagina $\boldsymbol{U}(i)$, zero altrimenti.

```
>> [U,H] = surfer ('http://www.microsoft.com/ ', 150 )
```

In figura 3.4 è riportato lo spy plot di H, che ci permette di visualizzare la sparsità della matrice. Gli elementi diversi da zero sono riportati con un punto blu sul grafico.

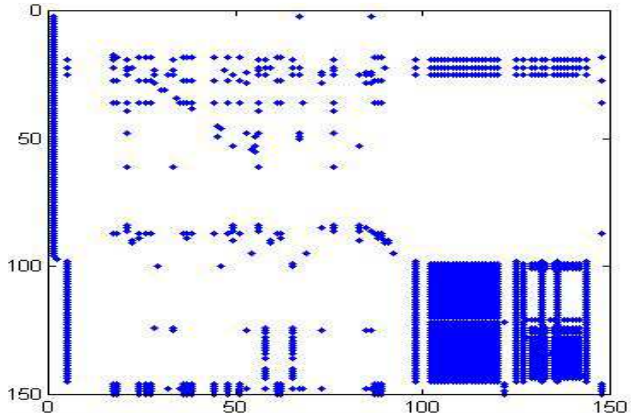

Figura 3.4 Spy plot della matrice H

Creiamo ora le matrici e i vettori di cui abbiamo bisogno nell'implementazione dell'algoritmo di PageRank:

```
>> M=double(full(H)');

>>e=ones(150,1);

>>v=1/150 * e;

>> S=creaS(M);
```

Dove creaS è la seguente funzione:

```
function S = creaS (M)
link_usc = sum (M,2);
n = length (M);
S = M;
for i = 1:n
      if link_usc(i) == 0
      S(i,:) = 1 / n;
  else
      S(i,:) = S(i,:) / link_usc(i);
  end
end
```

Con questi comandi otteniamo la matrice S. Adesso calcoliamo la matrice di Google G, utilizzando $\alpha = 0.85$.

```
>> G = 0.85 * S + 0.15 * e * v';
```

Per calcolare il vettore PageRank dobbiamo utilizzare il metodo delle potenze, tramite la funzione powermethod, definita su Matlab. Riportiamo qui il codice:

```
function[eigenvect,eigenvalue,it] = powermethod (A,t_0)
it = 0;
err = 1;
gamma = 1;
t_prec = t_0;
tau_prec = 1;
while err >= 1.0e-16
     it = it + 1;
     u_att = A * t_prec;
     tau_att = sum(abs(u_att));
     gamma = gamma * tau_att;
     t_att = 1 / tau_att * u_att;
     err = abs((tau_att - tau_prec)) / tau_att;
      t_prec = t_att;
      tau_prec = tau_att;
end
u_att = A * t_prec;
[massimo,j] = max(abs(t_prec));
eigenvalue = u_att(j) / t_prec(j)
alpha_1x_1 = gamma / eigenvalue * u_att;
norma = sum(alpha_1x_1);
eigenvect = 1/norma * alpha_1x_1;
```

Digitando il comando seguente nel prompt di Matlab, si ottiene il vettore PageRank (pagerank) del grafo web che abbiamo scaricato con la funzione surfer (figura 3.5), l'autovalore dominante (lambda) e il numero di iterazioni effettuate (it).

```
>> [pagerank, lamba, it]=powermethod(G',v)
```

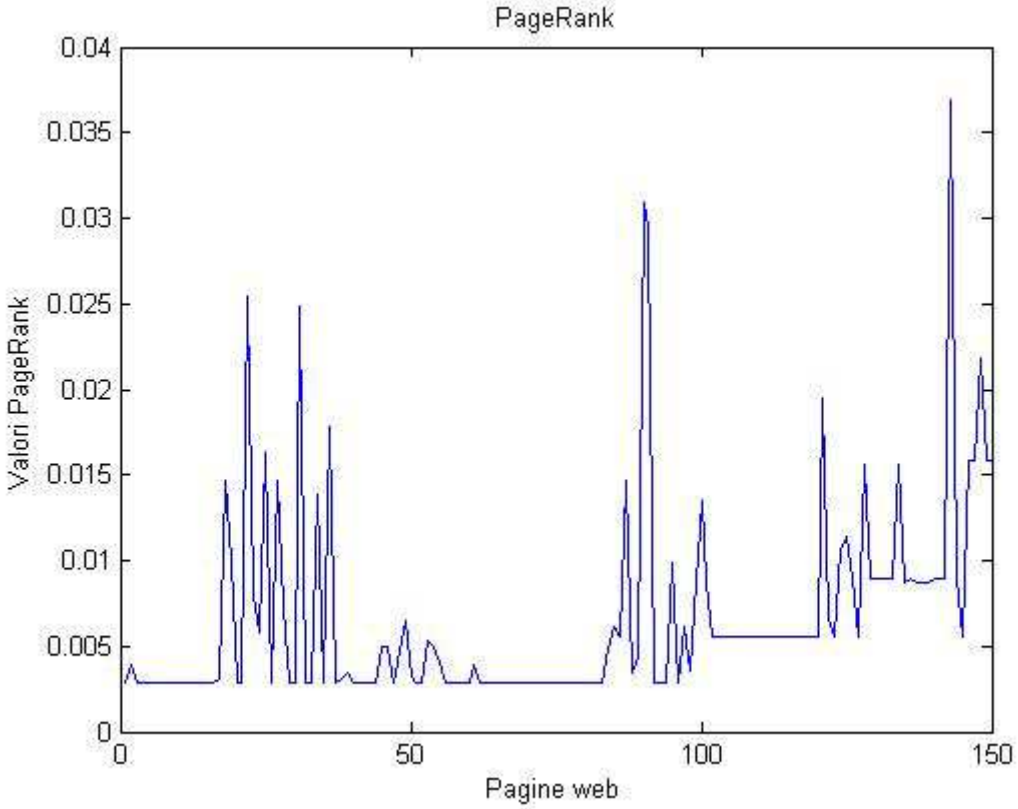

Figura 3.5 PageRanks della porzione di rete scaricata con pagina di partenza il sito web della Microsoft

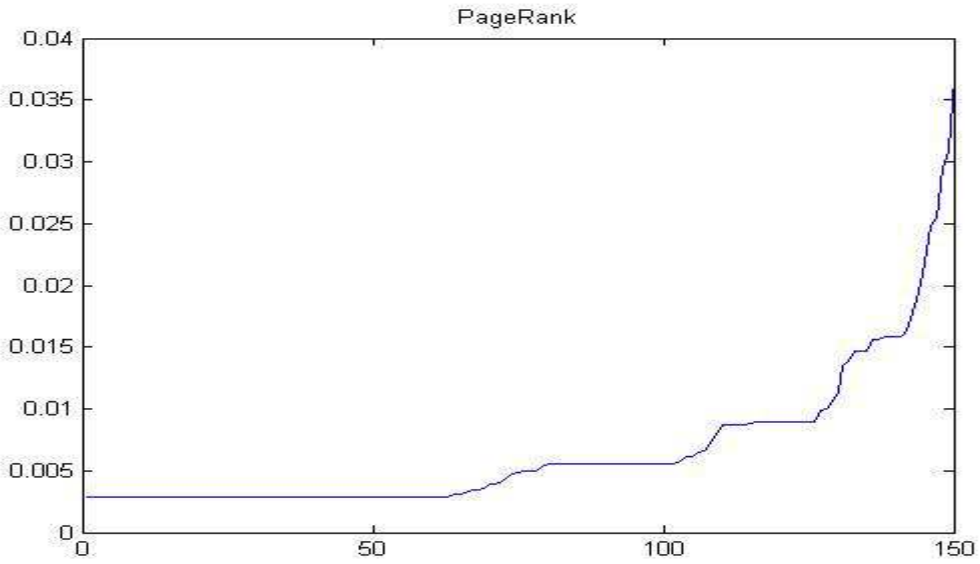

Figura 3.6 Valori PageRanks in ordine crescente

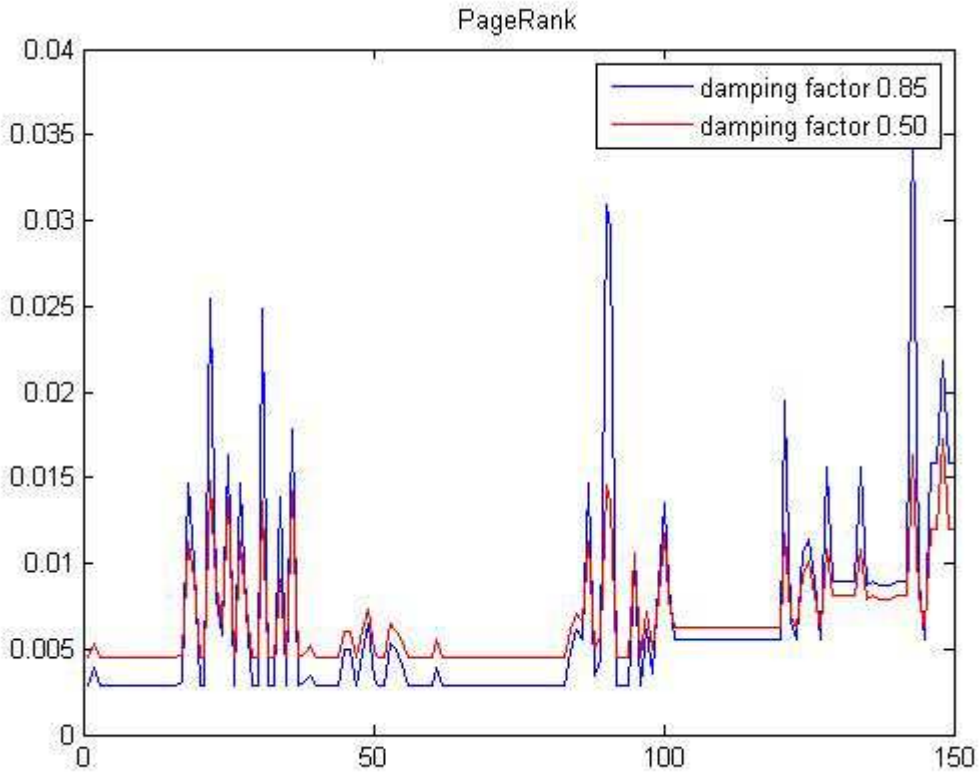

Figura 3.7 Valori PageRanks per $\alpha = 0{,}85$ e $\alpha = 0{,}50$

Nella figura 3.5 sono rappresentati i valori del vettore PageRank della porzione di rete scaricata, partendo dal nodo: http://www.microsoft.com. Sull'asse delle ascisse sono riportate le pagine web e sull'asse delle ordinate i loro relativi valori di PageRank. Ordinando i valori del vettore PageRank in ordine crescente si ottiene il grafico 3.6. Dalla figura si osserva che il numero di pagine web aventi un buon punteggio PageRank è molto limitato rispetto al totale (complessivamente 250). Infine, nella figura 3.7 sono rappresentati i PageRanks del grafo considerato per due valori diversi del parametro α, $\alpha = 0{,}85$ e $\alpha = 0{,}50$. Si osserva che, i due andamenti riportati in figura 3.7 sono differenti. Infatti, come discusso nel paragrafo 3.4, il vettore PageRank, per piccole variazione di α, risulta molto sensibile e ne possiamo vedere gli effetti in tale esempio.

Concludiamo questo paragrafo, calcolando il vettore PageRank della porzioni di rete scaricata partendo dal sito dell'Università degli Studi di Cagliari: http://www.unica.it, considerando un grafo di 30 pagine web. Riportiamo, nella pagina seguente,la tabella 3.2, dove sono riportati i nodi, i valori del vettore PageRank, il numero di inlinks e outlinks e il vettore $\boldsymbol{U}$ che contiene le pagine URL (i nodi) del grafo web scaricato.

Nodi	PageRank	In	Out	URL (Vettore $\boldsymbol{U}$)
2	0.0489	13	0	http://people.unica.it/rubrica
3	0.0489	13	0	http://people.unica.it/wifi
4	0.0474	12	27	http://elearning.unica.it
9	0.0367	11	0	http://elearning.unica.it/comments/feed
7	0.0367	11	0	http://elearning.unica.it/xmlrpc.php
11	0.0367	11	0	http://elearning.unica.it/wp-includes/wlwmanifest.xml
15	0.0367	11	0	http://people.unica.it/immagini/sfondocercapiatto.png
30	0.0367	11	0	http://elearning.unica.it/moodle-docenti/ guida-per-il-docente
6	0.0367	11	0	http://gmpg.org/xfn/11
8	0.0367	11	0	http://elearning.unica.it/feed
14	0.0367	11	4	http://www.unica.it
16	0.0367	11	0	http://people.unica.it/immagini/sfondocercatondo.png
12	0.0352	10	20	http://elearning.unica.it/help-desk
27	0.0352	10	20	http://elearning.unica.it/moodle-per-gli-studenti
26	0.0352	10	20	http://elearning.unica.it/corsi-di-riallineamento
28	0.0352	10	20	http://elearning.unica.it/moodle-per-gli-studenti/ guida-per-lo-studente
29	0.0352	10	20	http://elearning.unica.it/moodle-docenti
23	0.0352	10	20	http://elearning.unica.it/il-progetto
25	0.0352	10	20	http://elearning.unica.it/informazioni-portale
24	0.0351	10	19	http://elearning.unica.it/i-servizi
22	0.0337	9	20	http://elearning.unica.it/author/admin
5	0.0330	2	0	http://statistiche.unica.it/piwik
19	0.0223	1	0	http://elearning.unica.it/wp-content/uploads/2011/09/ pizza1.png
21	0.0223	1	0	http://elearning.unica.it/wp-content/uploads/2011/09/ wecam_ico3.png
10	0.0223	1	0	http://elearning.unica.it/home/feed
17	0.0223	1	0	http://elearning.unica.it/wp-content/uploads/2011/09/ cubo_elearning.png
13	0.0223	1	21	http://elearning.unica.it/piattamoodle
18	0.0223	1	0	http://elearning.unica.it/wp-content/uploads/2011/09/ moodle_circle1.png
20	0.0223	1	0	http://elearning.unica.it/wp-content/uploads/2011/09/ universe.png
1	0.0208	0	4	http://www.unica.it/

Tabella 3.2

I nodi 2 e 3 sono quelli con valore PageRank maggiore, infatti sono le pagine con un numero più alto di inlinks (13). Ciò conferma il fatto che le pagine web sono ordinate in maniera gerarchica tramite la popolarità dei link: una pagina è considerata di migliore qualità rispetto a un'altra se c'è un numero elevato di pagine che punta ad essa. Infine riportiamo i grafici del vettore PageRank.

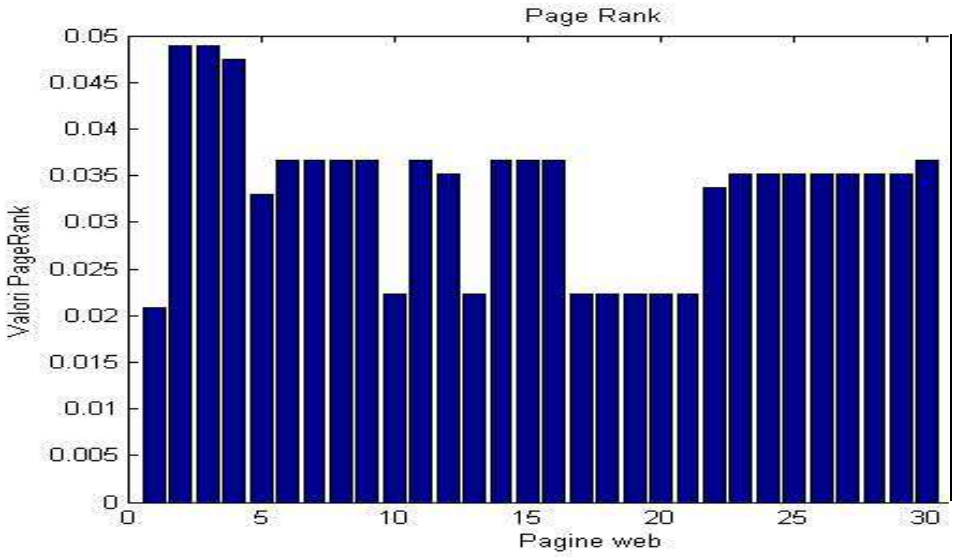

Figura 3.8 Grafico a barre del vettore PageRank della porzione di rete scaricata con pagina di partenza il sito web dell'Università di Cagliari.

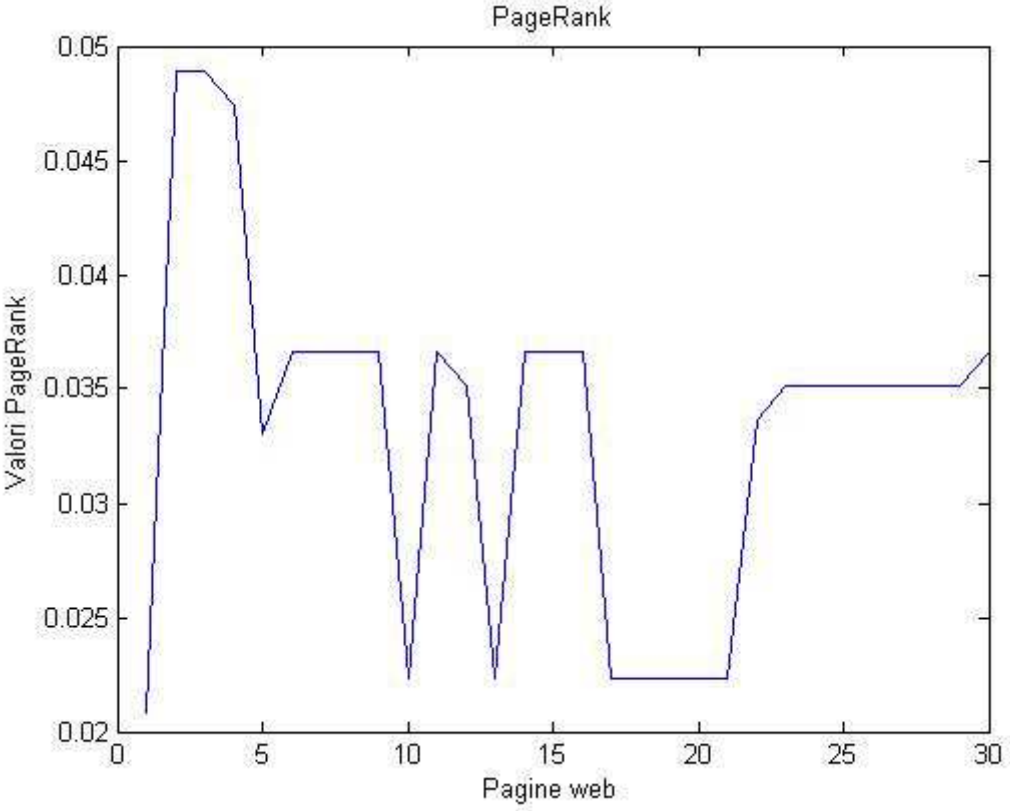

Figura 3.9 Valori PageRanks della porzione di rete scaricata con pagina di partenza il sito web dell'Università di Cagliari.

Bibliografia

[1]. *The Anatomy of a Large-Scale Hypertextual Web Search Engine,* Sergey Brin & Larry Page, 1998

[2]. *Analisi Numerica, Metodi Modelli Applicazioni*, Valerio Comincioli

[3]. *Google's PageRank and Beyond,* Amy N. Langville & Carld D. Meyer

[4]. *Matrix Analysis and applied linear algebra*, Carl D. Meyer

Sitografia

[1]. MathAcademy.ws *I teoremi di Gerschgorin*, Dario A. Bini, Università di Pisa

[2]. *Dispense sulla teoria dei grafi*, Prof. Antonio Sassano, Università Sapienza di Roma

[3]. *Link Analysis Ranking: Algorithms, Theory, and Experiments,* A. Borodin, G. O. Roberts, J. S. Rosenthal, P. Tsaparas

[4]. *Dispense sulla teoria dei grafi*, Prof. Vincenzo Acciaro, Università G. D'Annunzio, Chieti-Pescara

[5]. *Page Ranking Algorithms: A Survey*, Neelam Duhan, A. K. Sharma, Komal Kumar Bhatia, YMCA Institute of Engineering, Faridabad, India

[6]. *Dispense sulle catene di Markov,* Prof. Riccardo Cambini, Università di Pisa

[7]. Sito Internet *www.mathworks.com/matlabcentral*