Università degli Studi di Cagliari

FACOLTÀ DI INGEGNERIA E ARCHITETTURA
Corso di Laurea in Ingegneria Elettrica ed Elettronica

Analisi dei metodi HITS e PageRank per il ranking di reti complesse

Relatore:
Prof. Giuseppe Rodriguez

Candidato:
Roberto Carboni

Anno Accademico 2019-2020

Indice

Introduzione

La grande quantità di pagine Web ha portato negli anni '90 allo sviluppo di metodi di ranking per poterle ordinare, non solo attraverso una ricerca testuale, ma anche in base ad una gerarchia che abbia un riscontro con i reali interessi di un utente che naviga nel web.
Lo scopo di questa tesi è quello di analizzare e confrontare due metodi di ranking, in particolare il metodo HITS e il PageRank. Verranno prima analizzati i metodi singolarmente, evidenziando le particolarità ed esponendo la definizione sia dal punto di vista matematico che concettuale. Successivamente i due metodi verranno implementati in Matlab per poter eseguire dei test su reti reali. Confrontando poi i risultati ottenuti con i due metodi.
Verrà poi proposta una variante, di tipo concettuale, applicabile su entrambi i metodi. Infine si confronteranno i metodi originali con la variante proposta.

Capitolo 1

Grafi e matrici

1.1 Grafi

In generale un *grafo* è una struttura composta da:

nodi o **vertici** gli oggetti del grafo.

archi le relazioni tra i nodi.

Definita questa struttura, si indica con *degree* o *grado* il numero di archi connessi ad un nodo. Se due archi sono connessi ad uno stesso nodo sono detti *adiacenti*. A partire da un grafo si può ottenere un *sotto-grafo*, ossia un sottoinsieme di nodi e archi del grafo. Un esempio di grafo è riportato in Figura 1.1.

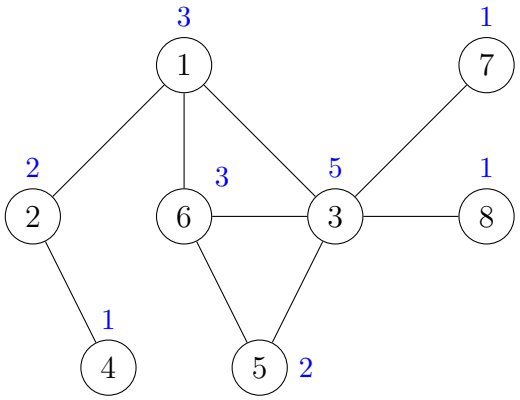


Figura 1.1: Esempio di grafo.

Nella Fiugra 1.1 i numeri neri cerchiati sono i nodi, le linee che uniscono i nodi sono gli archi e i numeri in blu rappresentano il degree di ogni nodo.

Nel seguito un grafo come quello di Figura 1.1, dove ad ogni nodo è assegnato un punteggio, verrà chiamato *grafo classificato*.
Attraverso questa struttura si può rappresentare, ad esempio, il web, dove i nodi rappresentano le pagine, gli archi i link tra le varie pagine. Nel caso del web il grafo che si ottiene ha una forma particolare, poiché le relazioni hanno una direzione, ossia una pagina che ha un link verso un'altra pagina non necessariamente riceve un link da quest'ultima. Per indicare che una pagina p_i ha un link verso una pagina p_j si può usare il termine ***puntare***, ossia p_i *punta* p_j.
Quello che si ottiene è un grafo *orientato*. Un esempio di questo tipo di grafo è riportato in Figura 1.2.

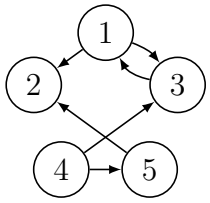


Figura 1.2: Esempio di grafo orientato.

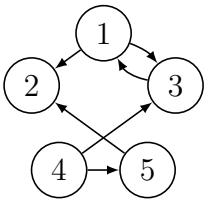
In altri termini, chiamando $\boldsymbol{V} = \{v_1, \ldots, v_n\}$ l'insieme dei nodi ed $\boldsymbol{E} = \{e_1, \ldots, e_n\}$ l'insieme degli archi, si può definire il grafo come l'insieme $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$. A partire da questa definizione, si dice *cammino* una sequenza di n nodi $u_1 \to u_2 \to \cdots \to u_n$ tali che per ogni $i = 1, \ldots, n$ si ha $(u_{i-1}, u_i) \in \boldsymbol{E}$ oppure $(u_i, u_{i-1}) \in \boldsymbol{E}$.Se in un cammino ogni arco è percorso una sola volta è detto cammino *semplice*. Un cammino semplice in cui il primo ed ultimo nodo coincidono viene detto *ciclo*. Chiaramente a seconda del grafo di partenza i cammini e i cicli potranno essere orientati o meno.

1.2 Matrice di adiacenza

Prendendo in considerazione un grafo possiamo studiarne le proprietà e il comportamento attraverso l'uso delle matrici. In particolare si può costruire una matrice che rappresenti le interazioni tra i nodi. Partendo da un grafo, come quello in Figura 1.2 a sinistra, si costruisce una matrice $\boldsymbol{A}$ ponendo un *1* in posizione *(i,j)* se il nodo i ha un arco in direzione j e uno *0* in caso contrario, ossia:

$$a_{ij} = \begin{cases} 1 & se \quad (i,j) \in \boldsymbol{E} \\ 0 & \text{altrimenti} \end{cases}$$

Si ottiene una matrice quadrata di grandezza $n \times n$, dove n è il numero di nodi, chiamata *matrice di adiacenza*. Un esempio è riportato in Figura 1.2 a destra.
Nel caso di Figura 1.2 la matrice di adiacenza è una matrice sparsa non simmetrica. Si dice che una matrice è sparsa se solo pochi elementi sono diversi da zero, tipicamente meno del 10% [8]. Invece una matrice $\boldsymbol{A}$ è non simmetrica se $\boldsymbol{A} \neq \boldsymbol{A}^T$. La matrice di adiacenza assume questa forma particolare perché il grafo preso in considerazione è un grafo orientato, quindi il legame tra due nodi può essere sia unidirezionale, come quello tra i nodi 1 e 2 di Figura 1.2, sia bidirezionale, come quello tra i nodi 1 e 3.



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

Figura 1.2: A destra la matrice di adiacenza costruita a partire dal grafo di sinistra.

Un altro fenomeno da considerare è quello dei *dangling-nodes*, ossia dei nodi che non hanno archi verso altri nodi, oppure che non hanno archi entranti(come nel caso dei nodi 2 e 4 di Figura 1.2). La presenza di questi nodi produce rispettivamente delle righe o delle colonne di soli zeri nella matrice di adiacenza
Considerando il web, un nodo senza archi entranti può essere una pagina che non riceve link da altre pagine e che non ha link verso se stessa. Mentre un nodo che non ha archi uscenti può essere una pagina web che non ha link né verso altre pagine né verso se stessa.

Capitolo 2

Metodi di ranking

Un primo modo per classificare i nodi può essere quello di utilizzare il degree, considerando importante un nodo connesso a molti archi e poco importante un nodo connesso a pochi archi. Questo tipo di punteggio considera però soltanto il numero di archi connessi ad un nodo ma non la loro *qualità*. Quindi un arco prodotto da un nodo molto importante ha lo stesso peso di un arco prodotto da un nodo meno importante.
I metodi di ranking che verranno descritti in questo capitolo definiscono invece degli archi con un peso, si ha quindi una distinzione tra un arco proveniente da un nodo molto importante e quello proveniente da un nodo poco importante.

2.1 PageRank

Il "PageRank" fu sviluppato da Larry Page e Sergey Brin come metodo di ranking per un nuovo motore di ricerca.
L'idea di questo metodo è di assegnare, a partire dalla matrice del web, un *punteggio di importanza* ad ogni pagina.

2.1.1 Teleportation parameter

Un aspetto distintivo di questo metodo è l'uso di una costante α chiamata *teleportation parameter* o *damping factor*.
Questo parametro viene usato per definire la probabilità α che un utente, che visita una pagina, segua un link piuttosto che un altro. Al contrario il termine 1-α definisce la probabilità che un utente inizi una nuova ricerca, senza seguire i link all'interno dell'ultima pagina che ha visitato.
Essendo un parametro che definisce una probabilità può assumere un valore

compreso tra 0 e 1, generalmente si assegna $\alpha = 0.85$ [2].
La presenza di α permette di poter personalizzare il PageRank. Ad esempio, assegnando questo valore ad una sola pagina o ad un gruppo di pagine si può impedire al proprietario di una pagina o di un sito di poter aumentare il proprio punteggio in modo arbitrario [2]. Inoltre α permette di personalizzare il modello di un utente, ad esempio si può modificare il valore di questo parametro in base alle abitudini di un utente.

2.1.2 Definizione

Considerando una pagina p e chiamando $c(p)$ il numero di link uscenti da p, si può definire una prima forma semplificata di PageRank [6] come:

$$pr(p) = \alpha \sum_{i \in \mathcal{I}} \frac{pr(t_i)}{c(t_i)}$$

Dove t_i è una pagina che ha un link verso p e $\mathcal{I}$ è l'insieme degli indici delle pagine che hanno un arco verso la pagina p.
Da questa definizione segue che il punteggio di un nodo non dipende solo dal numero di archi entranti, ma anche dal punteggio dei nodi da cui arrivano. In Figura 2.1 viene evidenziato questo concetto, ossia come il punteggio di una pagina venga partizionato sulle pagine che *punta*.

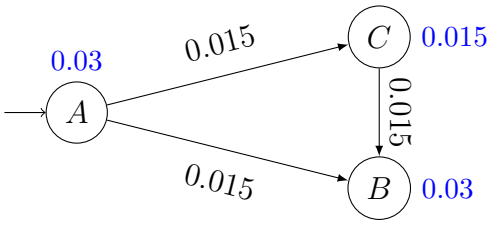


Figura 2.1: PageRank semplificato

Se nel caso di un sotto-grafo come quello di Figura 2.1 ma con un arco aggiuntivo da B a C, si provasse a calcolare il PageRank nella versione semplificata, il punteggio dei nodi B e C tenderebbe all'infinito, perché non partizionerebbero il loro punteggio con altri nodi all'esterno del sotto-grafo.
Per ovviare a questo problema, Il punteggio di PageRank pr di una pagina p è allora definito come [2]:

$$pr(p) = \frac{(1-\alpha)}{n} + \alpha \sum_{i \in \mathcal{I}} \frac{pr(t_i)}{c(t_i)}$$

Dove n è il numero di pagine web.
Calcolando il PageRank su tutte le pagine si ottiene un vettore pr che contiene quindi, i punteggi di tutte le pagine. Il vettore pr è un *vettore stocastico*, ossia la somma di tutti i suoi termini è pari a 1. Applicando il PageRank al grafo di Figura 1.2 si ottiene il grafo classificato di Figura 2.2.

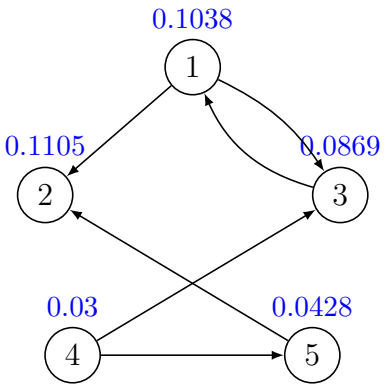


Figura 2.2: Esempio di grafo classificato con il PageRank.

Usando il PageRank, una pagina che ha un link verso un'altra pagina le conferisce un certo grado importanza. Tuttavia, una pagina importante contribuirà maggiormente in questo processo rispetto ad una pagina meno importante. Segue che *una pagina è importante se è* ***puntata*** *da altre pagine importanti*[5]. L'importanza di un nodo, per questo metodo, viene definita come la probabilità che, dopo un tempo sufficientemente grande, un utente si trovi proprio in quel nodo [6].
Un altro aspetto da considerare è che, siccome questo metodo è stato sviluppato per un motore di ricerca, uno degli obbiettivi di Page e Brin era quello di implementare un metodo che restituisse le pagine più importanti entro le prime decine di risultati di una ricerca, poiché, secondo una loro ipotesi, l'attenzione di un utente è focalizzata, appunto, sulle prime decine di pagine [2].

2.2 HITS

Il metodo "HITS" (Hyperlink-Induced Topic Search), o anche "Hubs and Authorities", è un metodo di ranking sviluppato da Jon M. Kleinberg [4].

2.2.1 Definizione

Anche per questo metodo l'idea è quella di classificare le pagine web in base ad un punteggio di importanza.
Questo algoritmo è suddiviso in due fasi: nella prima viene costruita una *sotto-matrice* del web attraverso una ricerca testuale su tutte le pagine del web, che verrà poi elaborata nel secondo step.
Nella seconda fase vengono calcolati due punteggi per ogni pagina: un punteggio di *hub* e uno di *authority*. Il punteggio di hub viene assegnato in base ai link uscenti dalla pagina, mentre il punteggio di authority in base ai link verso la pagina di cui si vuole calcolare il punteggio [4].
Quindi, *per avere un punteggio di hub alto una pagina deve* ***puntare*** *molte pagine con un punteggio di authority alto, viceversa per avere un punteggio di authority alto una pagina deve essere* ***puntata*** *da molte pagine con un punteggio di hub alto*[4].
Secondo quanto descritto, il punteggio di authority viene calcolato come:

$$x_k = \sum_{i=1}^{n} a_{ik} y_i \tag{2.2.1}$$

e, in modo analogo, quello di hub come:

$$y_k = \sum_{j=1}^{n} a_{kj} x_j \tag{2.2.2}$$

I vettori x e y che contengono i punteggi di authority e di hub rispettivamente sono, come nel caso del vettore pr di PageRank, dei vettori stocastici.
Considerando il metodo in forma matriciale si ottiene:

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

$$y^{(k)} = \boldsymbol{A} x^{(k-1)}$$

dove A è una matrice di adiacenza. Le equazioni precedenti possono essere riscritte come:

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

viene quindi esplicitato l'uso di una matrice simmetrica, poiché il prodotto di una matrice per la sua trasposta è proprio una matrice simmetrica.
Quello appena definito è un metodo iterativo, che converge per $k \to \infty$. Quindi $x^{(k)}$ e $y^{(k)}$ sono l'approssimazione di x e y alla k-esima iterazione. I primi vettori $x^{(0)}$ e $y^{(0)}$ vengono inizializzati con tutti i termini pari a 1.

In Figura 2.2 è riportato un esempio di grafo classificato con i punteggi di hub e authority. Si nota subito che ci sono dei nodi con punteggio nullo, in particolare il nodo 2 ha punteggio di hub nullo perché non ha archi uscenti, mentre il nodo 4 ha punteggio di authority nullo perché non ha archi entranti.

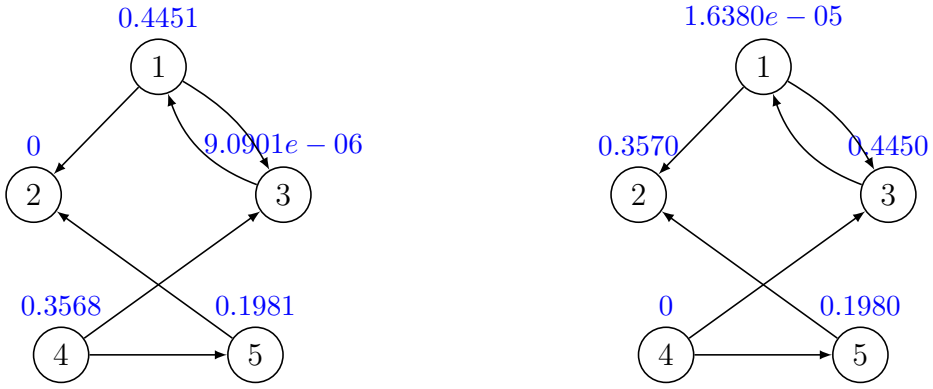


Figura 2.3: Esempio di grafo classificato con il metodo HITS, a sinistra con il punteggio di hub, a destra con quello di authority.

2.2.2 Formulazione matriciale

Come visto questo metodo prende in ingresso una matrice di adiacenza tipicamente non simmetrica, che poi nel calcolo dei punteggi viene trasformata in matrice simmetrica. Questa trasformazione può essere vista con la matrice aumentata[1]:

$$\boldsymbol{B} = \begin{pmatrix} 0 & \boldsymbol{A} \\ \boldsymbol{A}^T & 0 \end{pmatrix}$$

dove $\boldsymbol{B}$ è una matrice simmetrica di grandezza $2n \times 2n$ ed n è il numero dei nodi. Con questa formulazione le (2.2.1) e (2.2.2) possono essere riscritte come:

$$\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{x} \end{pmatrix} = \boldsymbol{B} \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{x} \end{pmatrix}$$

In termini di grafi questa riformulazione può essere espressa prendendo in considerazione due vettori identici che contengono tutti i nodi:

[1]Matrice ottenuta aggiungendo delle colonne ad una data matrice.

$$\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 2 \\ \vdots \\ n \end{pmatrix} \qquad \boldsymbol{v}_2 = \begin{pmatrix} 1 \\ 2 \\ \vdots \\ n \end{pmatrix}$$

Partendo da un grafo orientato, attraverso questi vettori si costruisce un *grafo bipartito* in cui è presente un arco non orientato tra il nodo i e il nodo j se nel grafo orientato è presente un arco orientato da i a j.
Quindi riprendendo il grafo orientato di Figura 1.2, riportato in Figura 2.1 a sinistra, si ottiene il grafo non orientato di Figura 2.3 a destra.

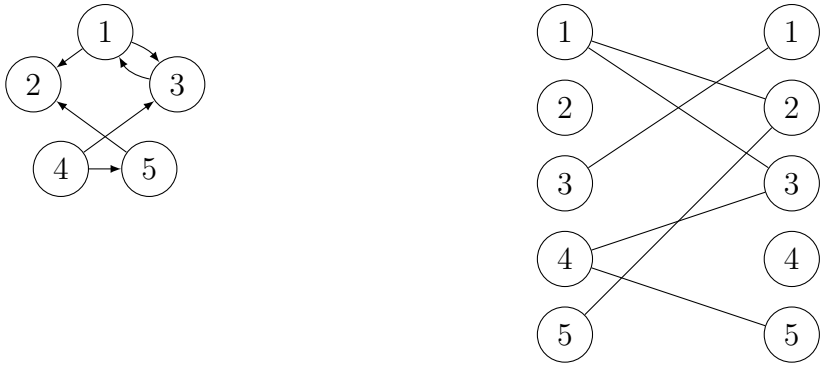


Figura 2.3: Grafo orientato a sinistra, grafo bipartito non orientato a destra.

Si nota quindi che la matrice aumentata $\boldsymbol{B}$ è la matrice di adiacenza del grafo bipartito.

2.3 Metodi a confronto

Come specificato precedentemente, il metodo di Kleinberg, a differenza del PageRank, restituisce due punteggi per ogni nodo.
Mentre il punteggio di authority, che definisce *l'importanza* di una pagina, può essere confrontato con il punteggio definito da Page e Brin, il punteggio di hub non può essere paragonato a quello del PageRank, o comunque non a quello del PageRank in modalità standard, poiché quello di hub non è un punteggio che conferisce un *tipo* di importanza come quello del PageRank.
Se l'algoritmo di Page e Brin viene calcolato non sulla matrice di adiacenza ma sulla sua trasposta, allora il risultato ha un altro significato, ossia non definisce *quanto* è importante un nodo ma *perché* è importante. Questa modalità prende il nome di Reverse PageRank. Ed è questa modalità che può essere confrontata con il punteggio di hub del metodo HITS.

Un'altra differenza importante sta nell'utilizzo della matrice di adiacenza. Mentre il PageRank usa una matrice di adiacenza non simmetrica, il metodo HITS non usa direttamente questa matrice ma ne costruisce una simmetrica. Questo è dovuto proprio all'utilizzo dei due punteggi di hub e authority. Poiché il punteggio di hub considera gli archi uscenti dai nodi e quello di authority considera gli archi entranti, in modo implicito quello che si prende in considerazione è un grafo bipartito non orientato, che ha come matrice di adiacenza proprio una matrice simmetrica.
Dai grafi classificati si è notato come il metodo HITS assegni un punteggio nullo, sia di hub che di authorities, a nodi che non hanno archi uscenti o entranti rispettivamente, a differenza del PageRank che assegna sempre un punteggio diverso da zero.
Infine, mentre il metodo HITS restituisce sempre vettori stocastici, il PageRank, senza opportune modifiche, in presenza di dangling nodes produce un vettore pr che non è stocastico. Tuttavia, come verrà spiegato in seguito, si può modificare il PageRank per ovviare a questo problema.

Capitolo 3

Algoritmi e implementazione

Dopo aver visto l'idea dietro i due metodi, verranno ora visti nel dettaglio gli algoritmi.

3.1 PageRank

Per la definizione di questo metodo, viene preso in considerazione un utente casuale che, con probabilità α, si sposta da una pagina web ad un'altra attraverso dei link.
Ora, chiamando $\boldsymbol{P}$ la matrice in cui il valore P_{ij} è la probabilità che l'utente passi dalla pagina i alla pagina j, si può dare una prima definizione di PageRank, si definisce il vettore **pr** del PageRank la soluzione al problema agli autovalori:

$$(\alpha \boldsymbol{P} + (1-\alpha)\boldsymbol{v}\boldsymbol{e}^{\boldsymbol{T}})\boldsymbol{pr} = \boldsymbol{pr}$$

dove $\boldsymbol{e}$ è un vettore colonna di soli *1* e $\boldsymbol{v}$ un vettore colonna di n termini, in cui ogni termine è pari a $1/n$.
Questa prima definizione fa quindi uso diretto degli autovalori.
La definizione che verrà usata per l'implementazione in questo capitolo è però un'altra, ovvero quella proposta da David F. Gleich [3]. In questo secondo caso viene definito il vettore $\boldsymbol{pr}$ del PageRank come la soluzione al sistema lineare:

$$(\boldsymbol{I} - \alpha \boldsymbol{P})\boldsymbol{pr} = (1-\alpha)\boldsymbol{v}$$

dove $\boldsymbol{pr} \geq 0$ e $\boldsymbol{e}^{\boldsymbol{T}}\boldsymbol{pr} = 1$. Quello che si ottiene è un vettore stocastico, ossia un vettore in cui la somma di tutti gli elementi è pari a uno.

Standard Random Walk Nel prossimo capitolo, verranno eseguite alcune prove numeriche in cui il PageRank verrà utilizzato in diverse modalità, in

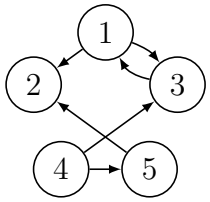
particolare in modalità "Standard Random Walk" e "Reverse". La differenza, dal punto di vista matematico, si trova nella costruzione della matrice $\boldsymbol{P}$. Definendo $\boldsymbol{d}$ il vettore colonna come $\boldsymbol{d} = \boldsymbol{Ae}$, tale vettore presenterà in posizione i la somma degli *1* nella riga *i-esima* della matrice $\boldsymbol{A}$:

$$\boldsymbol{d_i} = \sum_{j=0}^{n} \boldsymbol{A_{ij}}$$

segue quindi che il termine $\boldsymbol{d_i}$ è il degree del nodo *i-esimo*. Chiamando poi $\boldsymbol{D}$ la matrice con $\boldsymbol{d}$ in diagonale, la matrice $\boldsymbol{P}$ per la modalità Standard viene costruita come:

$$\boldsymbol{P} = \boldsymbol{A^T D^{-1}} \tag{3.1.1}$$

dove $\boldsymbol{D^{-1}}$ è la matrice inversa di $\boldsymbol{D}$.
Utilizzando il grafo orientato visto nel Capitolo 1 la matrice $\boldsymbol{P}$, ottenuta usando la (3.1.1), è riportata in Figura 2.1.



$$\boldsymbol{P} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 1 \\ 1/2 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 \end{pmatrix}$$

Figura 3.1: A destra la matrice $\boldsymbol{P}$ in modalità "Standard Random Walk".

Dalla Figura 2.1 si nota che la matrice $\boldsymbol{P}$ ha una colonna di soli zeri, questo perché come già visto, il nodo 2 è un dangling node. Usando la matrice $\boldsymbol{P}$ costruita in questo modo, in presenza di dangling nodes il PageRank restituisce un vettore non stocastico.

Weakly Preferential Definiamo ora la matrice $\boldsymbol{P}_s$ come:

$$\boldsymbol{P}_s = \boldsymbol{P} + \boldsymbol{vc}^T \tag{3.1.2}$$

come proposto da Boldi et al.(2008) [1]. Dove $\boldsymbol{c}$ prende il nome di *vettore di correzione*, i cui valori sono definiti come:

$$c_i = \begin{cases} 1 & \text{se il nodo } i\text{-esimo è un dangling node} \\ 0 & \text{altrimenti} \end{cases}$$

Usando la 3.1.2 il PageRank restituisce sempre un vettore stocastico, anche in presenza di dangling-nodes. Questa variante prende il nome di *Weakly Preferential PageRank*.
Prendendo sempre in considerazione il grafo di Figura 2.1 la matrice $\boldsymbol{P}_s$ risultante è:

$$\boldsymbol{P}_s = \begin{pmatrix} 0 & 1/5 & 1 & 0 & 0 \\ 1/2 & 1/5 & 0 & 0 & 1 \\ 1/2 & 1/5 & 0 & 1/2 & 0 \\ 0 & 1/5 & 0 & 0 & 0 \\ 0 & 1/5 & 0 & 1/2 & 0 \end{pmatrix}$$

Figura 3.2: Matrice $\boldsymbol{P}_s$ in modalità *Weakly Preferential PageRank*.

Quella che si ottiene è una matrice molto simile alla $\boldsymbol{P}$ descritta precedentemente ma con $1/5$ nella seconda colonna ossia quella corrispondente al dangling-node.
In generale quindi, questa variante pone $1/n$ in tutti gli elementi della colonna i-esima della matrice $\boldsymbol{P}$ di grandezza $n \times n$, se il nodo i è un dangling-node. In altri termini, in questa modalità se un utente visita una pagina senza link, ha una probabilità $1/n$ di spostarsi su una qualsiasi altra pagina.

Reverse PageRank Come detto in precedenza esiste una variante del PageRank che produce un punteggio con significato analogo a quello di hub del metodo HITS.
In questa modalità $\boldsymbol{P}$ si costruisce come nel caso Standard con la sola differenza che si prende come matrice di adiacenza la trasposta della reale matrice di adiacenza, ossia $\boldsymbol{d} = \boldsymbol{A^T e}$. Segue quindi che la struttura matriciale resta invariata, cambia però concettualmente il significato del vettore $\boldsymbol{pr}$ del PageRank.

Implementazione Per le prove numeriche queste modalità sono state implementate in Matlab come segue:

Listing 3.1: Implementazione del PageRank

```
1 function [pr] = PR(A, mode, max, tau)
2 alpha = 0.85;
3 n = size(A, 1);
4 I = speye(n, n);
5 v = ones(n, 1)*1/n;
6 i = [1:n];
7 switch mode
```

```
8           case 'standard'
9                   e = ones(n, 1);
10                  d = A*e;
11                  D = spdiags(d, 0, n, n);
12                  P = A'*dinv(D);
13          case 'weak'
14                  d = A*e;
15                  D = spdiags(d, 0, n, n);
16                  P_s = A'*dinv(D);
17                  c = zeros(n, 1);
18                  P_s(:, d==0)=1/n;
19                  P = P_s + v*c';
20          case 'reverse'
21                  d = A'*e;
22                  D = spdiags(d, 0, n, n);
23                  P = A*dinv(D);
24  end
25  pr = jacobi( (I-alpha*P), ((1-alpha)*v), max, tau);
26  end
```

Si noti che le matrici diagonali $\boldsymbol{D}$ ed $\boldsymbol{I}$ sono state definite usando le funzioni *speye()* e *spdiags()*, che creano delle matrici sparse memorizzando però solo il valore e la posizione dei termini non nulli, in questo caso i termini in diagonale. In particolare i termini in diagonale della matrice $\boldsymbol{I}$ sono pari a *1*. L'uso di queste funzioni è fondamentale se si usano, come nel caso del web, matrici di dimensioni elevate.
La funzione *dinv()* è stata implementata per invertire la matrice $\boldsymbol{D}$ che, essendo diagonale, ha come matrice inversa ancora una matrice diagonale che può essere scritta come:

$$d_{ij}^{-1} = \begin{cases} 0 & se \quad i \neq j \\ 1/d_{ij} & se \quad i = j \end{cases}$$

La matrice $\boldsymbol{D}$ può essere *singolare*, ossia con determinante nullo. In questo caso la funzione *dinv()* calcola la matrice *pseudo-inversa* $\boldsymbol{D}^{+}$ definita come:

$$d_{ij}^{+} = \begin{cases} 0 & se \quad i \neq j \\ 0 & se \quad i = j, d_{i,i} = 0 \\ 1/d_{ij} & se \quad i = j, d_{i,i} \neq 0 \end{cases}$$

La funzione *jacobi()* è il metodo iterativo di Jacobi per la risoluzione dei sistemi lineari, implementato come riportato nel Listing 3.2 [7].

Listing 3.2: Implementazione del metodo di Jacobi

```
function [x] = jacobi(A, b, max, tau)
% metodo di Jacobi
% per la soluzione di sistemi lineari nella forma: Ax=b
% max indica in numero massimo di iterazioni
% tau indica la precisione richiesta,
%  usata per il criterio di arresto
n = size(A, 1);
x = zeros(n, 1);
k=0;
my_norm = inf;
while (my_norm>tau*norm(x)) && (k<max)
    x0 = x;
    for i = 1:n
        sigma = 0;
        for j = 1:n
            if (j ~= i)
                sigma = sigma + A(i, j)*x(j);
            end
        end
        x(i) = (b(i)-sigma)/(A(i, i));
    end
    k = k+1;
    my_norm = norm(x-x0);
end
end
```

Metodo di Jacobi Il metodo di Jacobi, utilizzato nel paragrafo precedente per risolvere sistemi lineari nella forma $\boldsymbol{Ax} = \boldsymbol{b}$, è un metodo iterativo del prim'ordine, ossia un metodo in cui il vettore dei termini noti $\boldsymbol{x}^{(k+1)}$ alla iterazione *k+1*, dipende solo dal vettore dei termini noti all'iterazione precedente.
Definendo le matrici $\boldsymbol{P}$ ed $\boldsymbol{N}$ come segue:

$$\boldsymbol{P} = \begin{pmatrix} a_{11} & & 0 \\ & \ddots & \\ 0 & & a_{nn} \end{pmatrix} \qquad \boldsymbol{N} = -\begin{pmatrix} 0 & & a_{ij} \\ & \ddots & \\ a_{ji} & & 0 \end{pmatrix}$$

E definendo poi $\boldsymbol{B} = \boldsymbol{P}^{-1}\boldsymbol{N}$, il metodo di Jacobi converge se e solo se il raggio spettrale $\rho(\boldsymbol{B})$ è minore di 1.

	$\rho(\boldsymbol{B})$	
	Standard random walk	Weakly preferential
YouTube	0.0654	0.8500
Twitter	0.0654	0.8500
Web Stanford	0.8500	0.8500

Tabella 3.1: Raggio spettrale nelle tre reti

In generale un metodo iterativo si dice *consistente* se:

$$\boldsymbol{x}^{(k)} = \boldsymbol{x} \Rightarrow \boldsymbol{x}^{(k+1)} = \boldsymbol{x}$$

dove $\boldsymbol{x}$ è la soluzione esatta del sistema.
Infine un metodo iterativo si dice *globalmente convergente* se per ogni vettore iniziale $\boldsymbol{x}^{(0)}$ il metodo converge alla soluzione esatta del sistema.
Nel capitolo seguente il metodo di Jacobi verrà applicato per la risoluzione del PageRank applicato a tre reti reali. In particolare verranno usati degli snapshot di YouTube [1], Twitter [2] e del web di Stanford [3].
In Tabella 3.1 è riportato il raggio spettrale della matrice $\boldsymbol{B}$ costruita considerando $\boldsymbol{A} = \boldsymbol{I} - \alpha\boldsymbol{P}$ nel caso del Random Standard Walk e $\boldsymbol{A} = \boldsymbol{I} - \alpha\boldsymbol{P_s}$ nel caso del Weakly Preferential, per ciascuna delle tre reti. Dai dati riportati in Tabella 3.1 risulta che nelle due modalità del PageRank il metodo di Jacobi converge sempre, poiché il raggio spettrale della matrice $\boldsymbol{B}$ è sempre minore di 1.
Ora, riprendendo il significato di convergenza globale, è stato calcolato il vettore di PageRank usando tre diversi vettori iniziali $\boldsymbol{x}^{(0)}$ per il metodo di Jacobi, in particolare è stato utilizzato un vettore $\boldsymbol{x_0^{(0)}}$ di soli *0*, un vettore $\boldsymbol{x_1^{(0)}}$ di soli *1* e un vettore $\boldsymbol{x_c^{(0)}}$ di numeri casuali ottenuto con la funzione *rand()* di Matlab. I risultati ottenuti sono poi stati confrontati con il vettore di PageRank calcolato usando l'operatore "\" di Matlab per la risoluzione di sistemi lineari, che verrà considerato come valore esatto. I dati sono riportati in Tabella 3.2 per la modalità Standard Random Walk e Tabella 3.3 per la modalità Weakly Preferential, dove l'errore relativo e_r è stato calcolato come:

$$\frac{\| \boldsymbol{pr} - \tilde{\boldsymbol{pr}} \|}{\| \boldsymbol{pr} \|}$$

I dati riportati nelle Tabelle 3.2 e 3.3 sono stati ottenuti eseguendo dieci iterazioni del metodo di Jacobi e con una precisione $\tau = 10^{-5}$.
Risulta quindi che la scelta migliore sia quella di usare come vettore iniziale un vettore di soli *0*.

	e_r		
	$x_0^{(0)}$	$x_1^{(0)}$	$x_c^{(0)}$
YouTube	6.7405e-04	3.4830	1.7727
Twitter	0.0391	165.2446	84.3775
Web Stanford	0.0395	389.8996	218.8184

Tabella 3.2: Errore relativo nelle tre reti(Modalità Standard Random Walk)

	e_r		
	$x_0^{(0)}$	$x_1^{(0)}$	$x_c^{(0)}$
YouTube	0.1870	226.3847	112.2823
Twitter	0.0497	195.0082	96.6018
Web Stanford	0.0593	609.8164	308.1800

Tabella 3.3: Errore relativo nelle tre reti(Modalità Weakly Preferential)

3.2 HITS

Come visto precedentemente il metodo HITS è diviso in due fasi, nella prima viene fatta una ricerca testuale su tutto il web, nella seconda viene applicato un algoritmo di ranking. In questo paragrafo verrà analizzata questa seconda fase.
L'algoritmo calcola i punteggi: h di hub e a di authority. Il valori non sono calcolati separatamente, ma al contrario dipendono strettamente l'uno dall'altro, poiché secondo l'idea di Kleinberg il valore hub di una pagina (o più generalmente di un nodo) dipende dal valore authority delle pagine che *punta*, così come il valore di authority di una pagina dipende dal valore di hub delle pagine da cui è *puntata*.
Per l'implementazione è stato usato il seguente codice Matlab:

Listing 3.3: Implementazione del metodo HITS

```
function [ha] = hits(A)
tau = 1e-7; % tolleranza
n = size(A, 1);
a = ones(n, 1)/sqrt(n);
h = ones(n, 1)/sqrt(n);
while 1
        a0 = a;
        h0 = h;
```

```
9           a = A'*h);
10          a = a/sum(a);
11          h = A*a0;
12          h = h/sum(h);
13          if ((abs(a-a0) < tau) & (abs(h-h0) < tau))
14                  break;
15          end
16  end
17  ha=[h,a];
```

Dove con "tau" si è indicata la tolleranza. Si noti che tutti i termini dei vettori $\boldsymbol{h}$ e $\boldsymbol{a}$ rispettivamente dei punteggi di hub e authority, sono inizializzati con il valore $\frac{1}{\sqrt{n}}$.
Il metodo così implementato, seguendo l'algoritmo descritto da Kleinberg [4], produce due vettori h e a che risultano essere stocastici anche in presenza di dangling-nodes.

Capitolo 4

Confronto dei metodi su reti reali

Per confrontare i due metodi, sono state utilizzate le seguenti reti reali:

1. YouTube (1 000 nodi, 8 822 archi). È un sotto-grafo di uno snapshot di YouTube[1] [1]. I nodi solo gli utenti e gli archi sono link di utenti ad altri utenti.

2. Twitter (3 656 nodi, 188 712 archi) è uno snapshot della rete di Twitter [2]. I nodi sono utenti e gli archi sono citazioni e re-tweet tra gli utenti.

3. Web di Stanford(10 000 nodi, 4 402 archi). È un sotto-grafo del web di Stanford[2] [3]. I nodi sono pagine e gli archi sono link di una pagina verso un'altra.

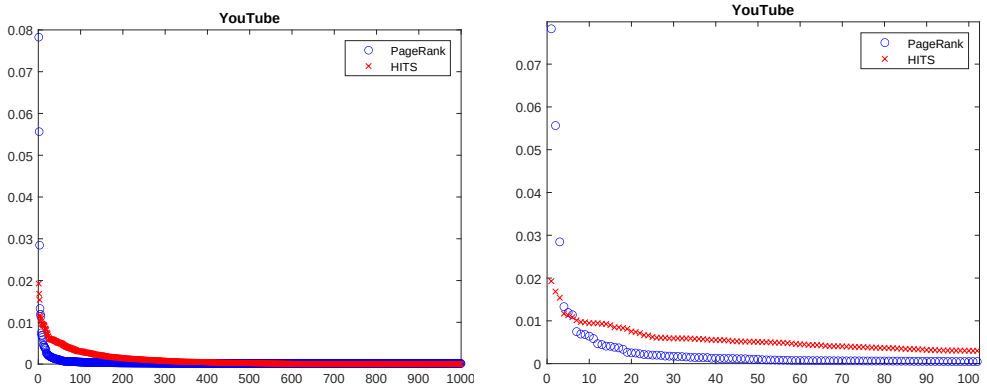
Verranno eseguite delle prove per confrontare il punteggio del PageRank in modalità standard e reverse con rispettivamente i punteggi authority e hub di HITS. Nel calcolo del vettore di PageRank è stato utilizzato il metodo di Jacobi con una precisione $\tau = 10^{-7}$ e con un massimo di cinquanta iterazioni.

4.1 Caso 1

Per questo caso verrà utilizzato un sotto-grafo di uno snapshot di YouTube. In Figura 4.1 sono riportati i punteggi calcolati con il PageRank e con il metodo HITS, di cui si considera in questo grafico il solo punteggio di authority. I nodi, dopo essere stati numerati, sono stati riportati sull'asse delle

[1] è un sotto-grafo ricavato considerando i primi 1000 nodi di un grafo di YouTube[1]

[2] come per YouTube, il sotto-grafo è stato costruito considerando i primi 10 000 nodi di un grafo del Web di Stanford



(a) *Standard PageRank e authority* (b) *Ingrandimento dei primi 250 nodi*

Figura 4.1: Prove su snapshot di YouTube

ascisse in base al punteggio in ordine decrescente. Dall'ingrandimento riportato in Figura 4.1(b) si può notare che, in generale, il metodo HITS assegni i punteggi in modo più omogeneo, a differenza del PageRank che accentra maggiormente i punteggi su pochi nodi, creando quindi dei gap molto ampi tra i nodi. Calcolando invece il Reverse PageRank e il punteggio di hub (dati riportati in Figura 4.2), si nota che la distribuzione dei punteggi è simile a quella di Figura 4.1, ma in modo più accentuato. Questa differenza sulla distribuzione dei punteggi si può notare anche dal grafo classificato della rete, ossia un grafo in cui si evidenzia il punteggio di ogni nodo. In Figura 4.3 e 4.4 [3] sono riportati i grafi (in cui sono stati omessi i dangling-nodes) classificati con i punteggi di PageRank e Authority. Si ritrova quindi che il metodo HITS assegna i punteggi in modo più omogeneo rispetto al PageRank.
Un altro aspetto che si può notare, con l'uso dei grafi, è che i nodi con punteggi più alti siano quelli con un maggiore numero di archi. Chiaramente questo aspetto si nota usando entrambi i metodi. Le stesse considerazioni possono essere fatte se si considerano i grafi classificati con i punteggi di Reverse PageRank e Hub del metodo HITS, riportati in Figura 4.5 e 4.6.

In Figura 4.7 è riportato il grafico con i punteggi di hub e di authority, in cui è stato aggiunto il sotto-grafo bipartito(gli archi sono rappresentati dalle linee in giallo) che contiene i primi dieci nodi con punteggio di hub più alto e i primi dieci nodi con punteggio di authority più alto[4]. Sono stati riportati

[3] I grafi classificati riportati in questa pagina e nel seguito sono stati prodotti seguendo, in parte, uno script del blog di Matlab [4].

[4] Sono stati considerati solo i primi dieci nodi con punteggio più alto, sia di hub che di authority, poiché sono i nodi con maggiore peso, conferiscono quindi maggiore punteggio ad altri nodi, e anche per semplicità di rappresentazione.

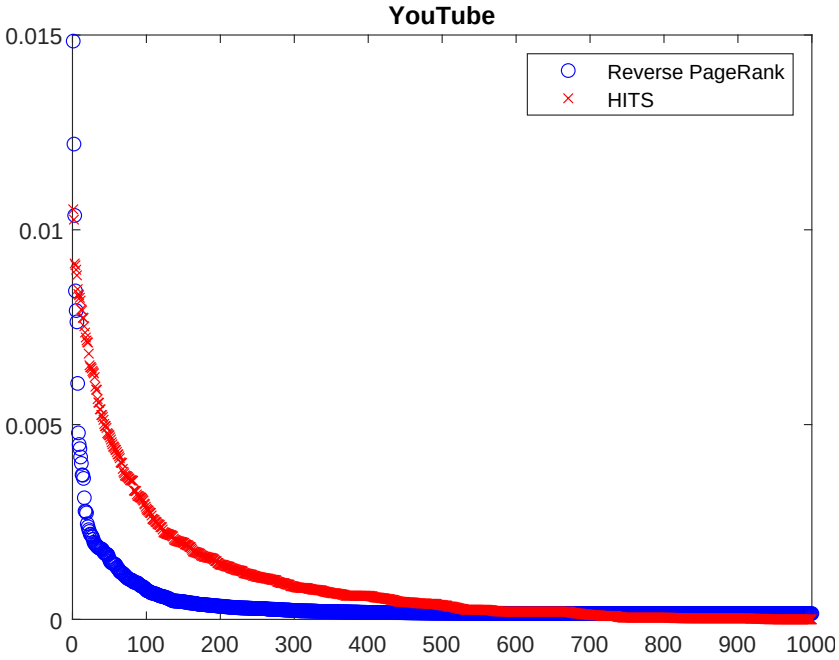


Figura 4.2: Punteggi di Reverse Pagerank e hub di YouTube

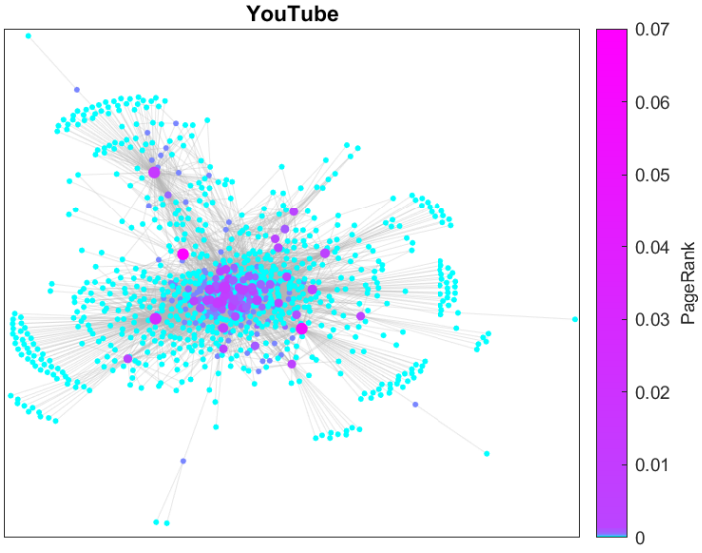


Figura 4.3: Grafo di YouTube classificato con PageRank

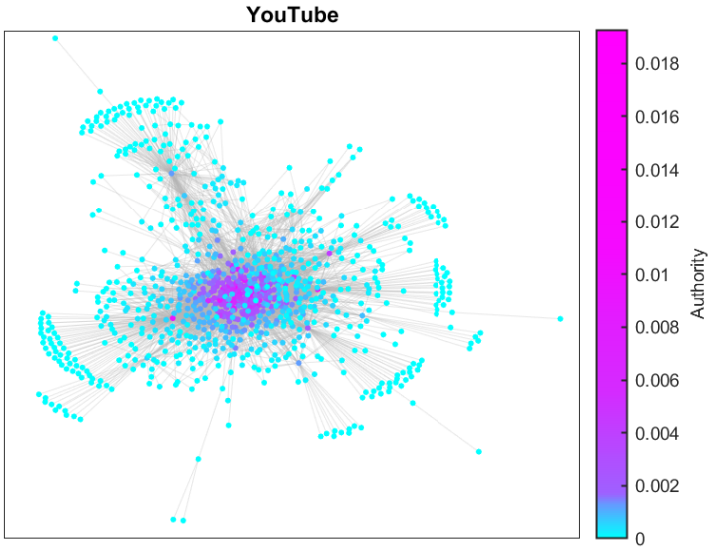


Figura 4.4: Grafo di YouTube classificato con HITS(authority)

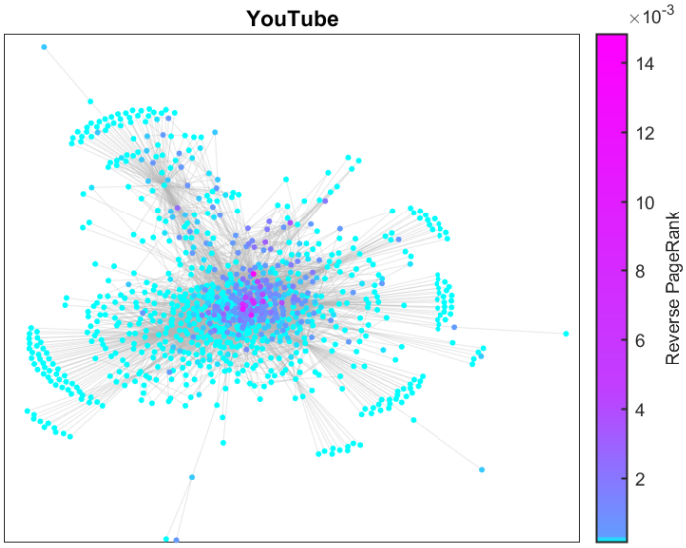


Figura 4.5: Grafo di YouTube classificato con Reverse PageRank

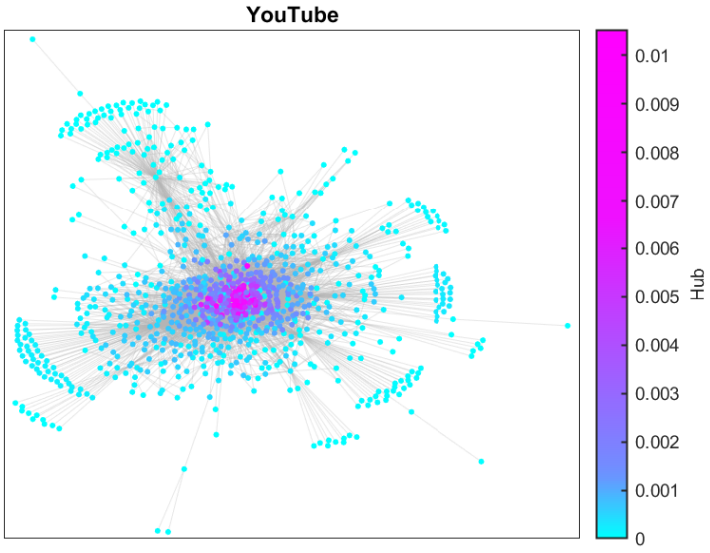


Figura 4.6: Grafo di YouTube classificato con HITS(hub)

quindi solo gli archi che coinvolgono questi venti nodi. Da questa figura si nota una stretta relazione tra i nodi con un alto hub e i nodi con un alto authority, concorde con quanto detto nei capitoli precedenti. Questa relazione si nota anche considerando il PageRank, infatti questo metodo in modalità Reverse ha significato analogo al punteggio di hub, come si può notare dalla Figura 4.8. In questa ultima figura è evidenziato come i nodi conferiscano una certa importanza ai nodi che puntano, infatti il nodo con punteggio di PageRank più alto non ha un legame diretto con i nodi con punteggio di Reverse PageRank più alto. La sua importanza è quindi dovuta al legame diretto con i nodi con un punteggio di PageRank elevato. In Tabella 4.1 sono riportati i nodi con relativi punteggi ottenuti con entrambi i metodi (PageRank in modalità standard e il punteggio di authority di HITS).

Sono stati evidenziati i tre nodi comuni nelle prime dieci posizioni di entrambi i metodi. Si nota quindi che solo un nodo ricopre la stessa posizione mentre gli altri due sono in posizioni diverse. Si può dedurre che i due metodi restituiscano classifiche piuttosto diverse.

Dai punteggi riportati in Tabella 4.1, si nota anche come il PageRank assegni punteggi più alti rispetto al metodo HITS. Uno degli effetti di questa diversità, come già visto, è che nel PageRank i punteggi sono molto distanziati tra di loro, ottenendo quindi pochi nodi con punteggi molto alti e molti nodi con punteggi molto bassi, mentre la distribuzione dei punteggi nel metodo

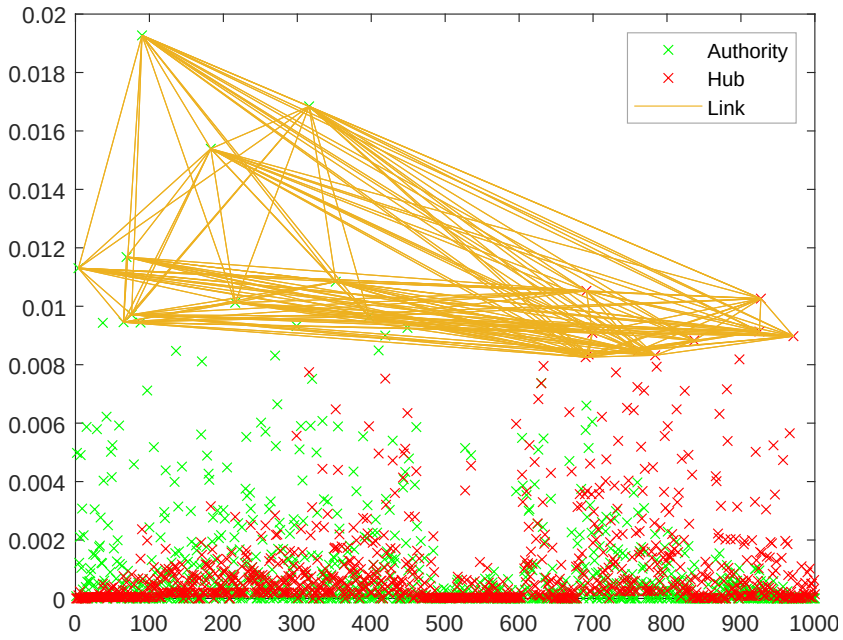


Figura 4.7: Legame tra i punteggi di Hub e Authority

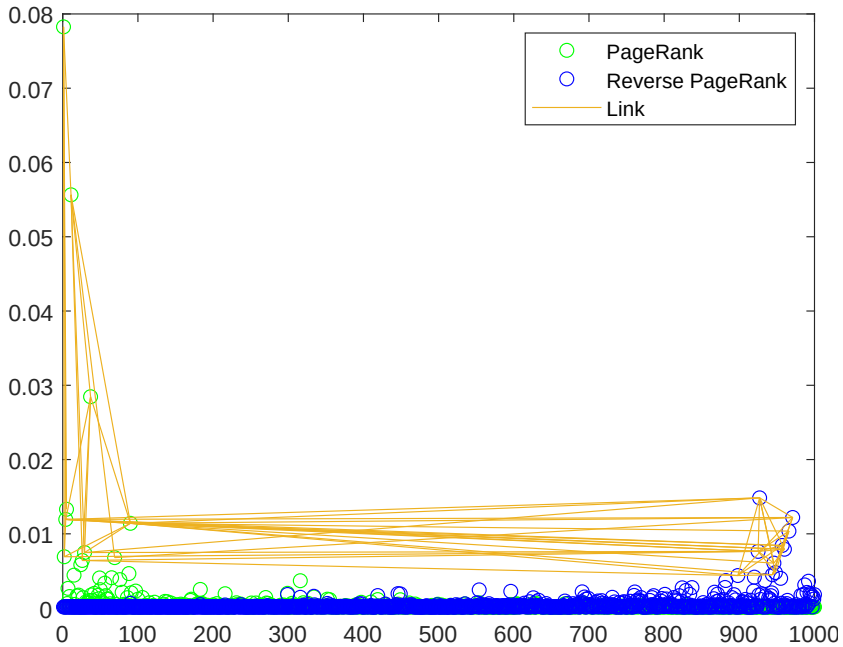


Figura 4.8: Legame tra i punteggi di PageRank e Reverse PageRank

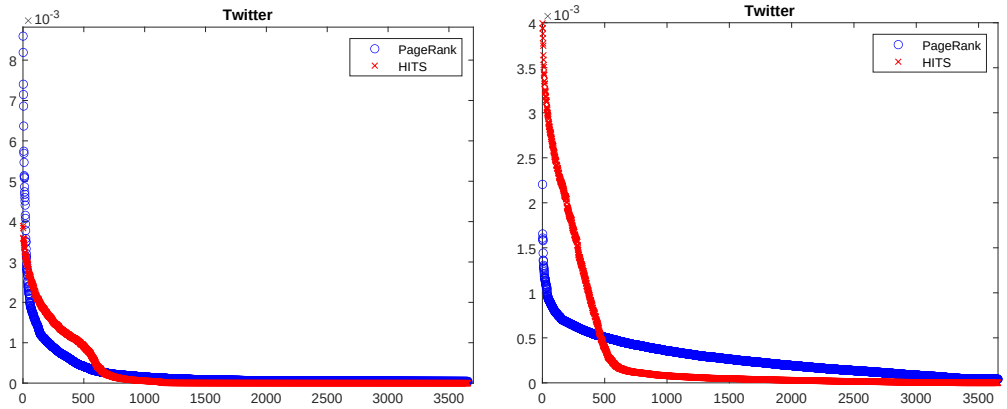
PageRank		HITS(authority)	
N^o nodo	Punteggio	N^o nodo	Punteggio
1	0.0783	**90**	0.0193
11	0.0556	316	0.0168
37	0.0285	183	0.0154
5	0.0133	**69**	0.0117
4	0.0113	**4**	0.0113
90	0.0114	352	0.0108
29	0.0075	216	0.0101
2	0.0069	76	0.0097
69	0.0068	397	0.0097
26	0.0054	65	0.0095

Tabella 4.1: Primi dieci risultati

HITS è più omogenea.

4.2 Caso 2

Per il secondo caso viene utilizzata una sotto-rete di Twitter.
Calcolando i punteggi e rappresentandoli graficamente, si ottengono i grafici in Figura 4.9. Come nel primo caso i punteggi sono disposti in ordine decrescente.
Si nota, dalla Figura 4.9(a) che i punteggi massimi del PageRank sono più alti rispetto a quelli di HITS, mentre HITS ha punteggi minimi più piccoli rispetto al PageRank. Come invece si nota dalla Figura 4.9(b), nella modalità reverse la situazione è diversa rispetto alla precedente. Infatti ora è HITS che assegna punteggi massimi più alti, mentre assegna punteggi minimi più bassi. Si ritrovano quindi gli stessi andamenti del caso precedente.
Per questa rete, nella classifica dei dieci nodi più importanti, non vi sono nodi in comunque. Questo sottolinea il fatto che i due metodi *premiano* i nodi in modo diverso, anche se in entrambi i metodi il significato di *importanza* è simile.
Andando a considerare i grafi classificati di Figura 4.10, si nota ancora una volta come il punteggio di authority del metodo HITS distribuisca i punteggi in modo più uniforme, rispetto al PageRank che accentra gran parte dei punteggi in pochi nodi.
La situazione è opposta nel caso del PageRank in modalità reverse e del punteggio di hub. Adesso è il PageRank che distribuisce i punteggi in modo



(a) *Standard PageRank e authority* (b) *Reverse PageRank e hub*

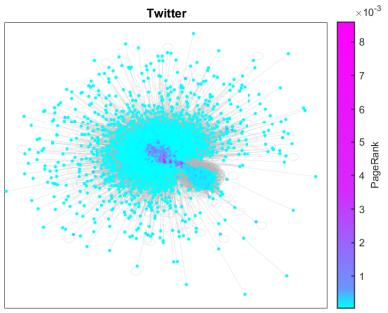
Figura 4.9: Prove su snapshot di Twitter

uniforme, mentre il metodo HITS *premia* con un alto punteggio di hub gli stessi nodi con un punteggio di authority elevato.

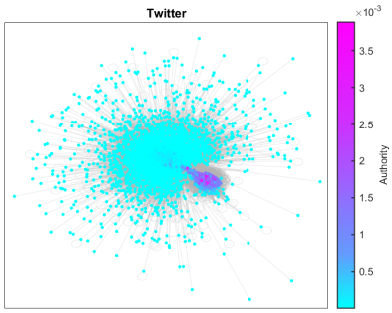
4.3 Caso 3

In questo caso viene presa in considerazione una sotto-rete di uno snapshot del web di Stanford.
Calcolando i punteggi con entrambi i metodi otteniamo i risultati di Figura 4.12. Dal grafico di Figura 4.12 si nota che il metodo HITS restituisce alcuni punteggi molto più alti rispetto a quelli del PageRank in modalità standard. Ma in generale è il PageRank che assegna punteggi più alti. Il grafico è in scala semi-logaritmica quindi non sono presenti i nodi con un punteggio pari a zero, segue che in questo caso il metodo HITS ha assegnato un punteggio nullo a gran parte dei nodi, al contrario del PageRank.
Considerando il Reverse PageRank e hub di HITS, si nota che l'andamento è analogo al grafico precedente.

L'andamento di questi punteggi può essere spiegato attraverso il grafo di questa sotto-rete.
In questo caso il nodo più importante è lo stesso in entrambi i metodi(standard PageRank e authority di HITS). Il sotto-grafo classificato a cui appartiene è riportato in Figura 4.14, da cui si nota subito l'elevato numero di archi entranti nel nodo. Per spiegare il punteggio di questo nodo si possono considerare i punteggi di Reverse PageRank e di hub di HITS. I grafi ottenuti con questi punteggi sono riportati in Figura 4.15. Si nota che i nodi con un punteggio di hub più alto fanno parte del sotto-grafo del nodo visto in pre-

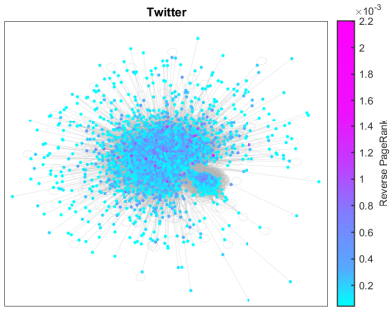


(a) *Grafo classificato con PageRank*

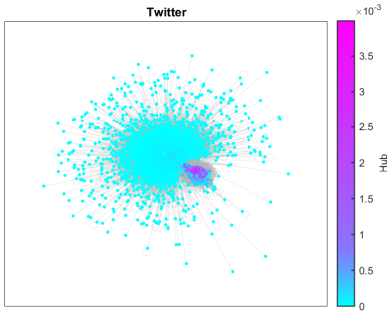


(b) *Grafo classificato con HITS(authority)*

Figura 4.10: Grafi classificati



(a) *Grafo classificato con Reverse PageRank*



(b) *Grafo classificato con HITS(hub)*

Figura 4.11: Grafi classificati

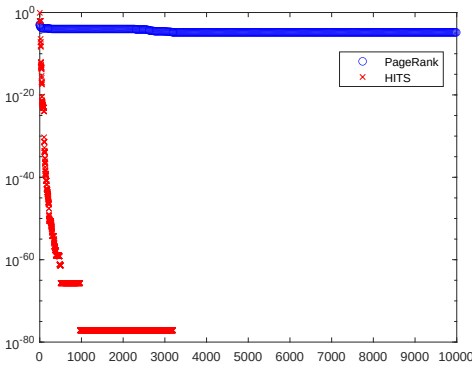


Figura 4.12: Punteggi PageRank e HITS(authority)

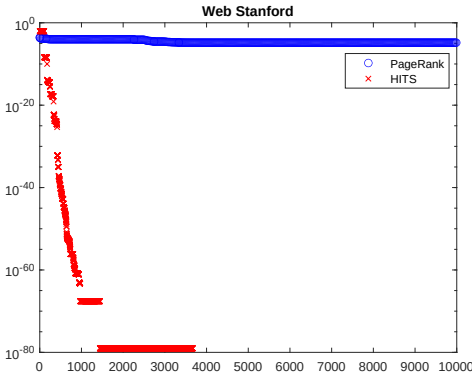
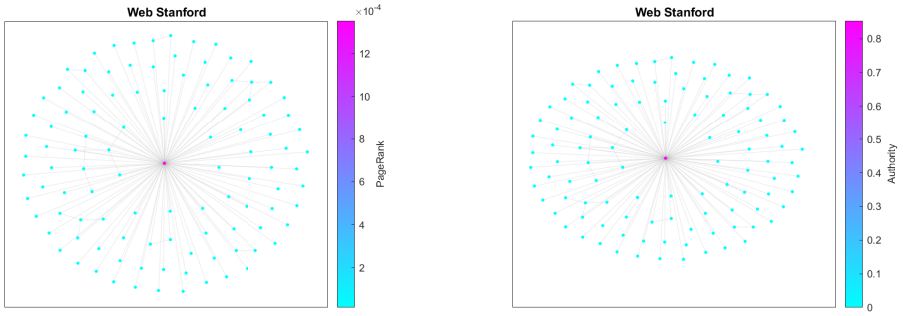


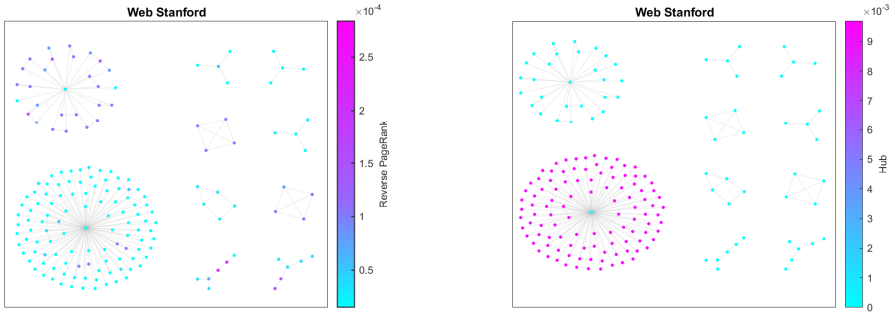
Figura 4.13: Punteggi PageRank e HITS(hub)



(a) *Sotto-grafo classificato con PageRank*

(b) *Sotto-grafo classificato con HITS(Authority)*

Figura 4.14: Grafi classificati



(a) *Sotto-grafo classificato con Reverse PageRank*

(b) *Sotto-grafo classificato con HITS(hub)*

Figura 4.15: Grafi classificati

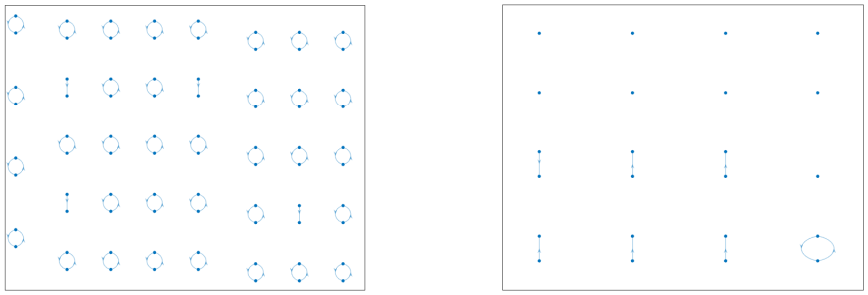


Figura 4.16: Sotto-grafi dello snapshot di Stanford

cedenza, mentre nel caso del Reverse PageRank i nodi più importanti fanno parte di altri sotto-grafi, ad evidenziare il fatto che nonostante questi punteggi abbiano significato analogo, producono risultati differenti. Ora, andando a considerare altri sotto-grafi, riportati in Figura 4.16, si nota la presenza di nodi con pochi archi sia entranti che uscenti, in altri casi invece, alcuni nodi non hanno interazioni con nessun nodo. La presenza di questi nodi era prevedibile dalla Figura 4.12. Risulta quindi che il metodo HITS assegna un punteggio pari a zero a questo tipo di nodi, mentre il PageRank assegna comunque un punteggio non nullo.

Capitolo 5

Variante dei metodi

Dopo aver implementato i metodi come sono stati descritti da Page e Brin [2] [6] e Kleinberg [4], di seguito verrà proposta una variante dei due metodi. Come visto nei grafi classificati del capitolo precedente, gran parte del punteggio è concentrato su pochi nodi, mentre gli altri hanno un punteggio molto basso.
La variante proposta, per entrambi i metodi, è quella di calcolare un primo vettore di punteggi eseguendo poche iterazioni, per poi eseguirne altre sui soli primi n nodi. In questo modo si esegue il metodo su una matrice di adiacenza di dimensioni ridotte. Nel seguito il numero n dei primi nodi più importanti corrisponde al 10% dei nodi totali, questo è solo uno dei possibili modi per scegliere i primi n nodi, altre scelte potrebbero migliorare il metodo variante, portando ad ottenere risultati molto simili a quelli dei metodi "originali". Chiaramente più nodi vengono usati nel secondo step e più i risultati saranno vicini a quelli dei metodi "originali".
Quello che si vuole ottenere è un vettore di punteggi che abbia lo stesso andamento del vettore ottenuto con il metodo originale, in altri termini si vuole che un nodo copra la stessa posizione di importanza assegnata dal metodo originale.

5.1 Confronto con i metodi varianti

Per confrontare i metodi originali con la variante proposta, verranno eseguite delle prove numeriche sulle reti reali viste precedentemente.

PageRank Per le prove seguenti è stata usata la modalità Weakly Preferential. Per la risoluzione del sistema lineare è stato usato il metodo di Jacobi, con un massimo di dieci iterazioni per il PageRank "originale", men-

YouTube		Twitter		Web Stanford	
Originale	Variante	Originale	Variante	Originale	Variante
1	**1**	**2356**	**2356**	2260	9279
11	**11**	**1665**	**1665**	7203	8121
37	**4**	**1815**	**1815**	3164	7895
5	**37**	**3481**	**92**	**8025**	4385
4	**2**	**92**	**3481**	7432	783
90	**90**	**2749**	**2749**	5562	5772
29	**29**	**2497**	**131**	2656	2137
69	15	**131**	**3562**	8261	1103
2	**26**	862	821	9168	**8025**
26	76	**3562**	**2497**	8083	6674

Tabella 5.1: Confronto con i punteggi del PageRank

tre nel metodo variante un massimo di cinque nel primo step e un massimo di dieci nel secondo.
Nella Tabella 5.1 sono riportati i dieci nodi più importanti ottenuti con il metodo originale e quello variante, evidenziando i nodi in comune. Si nota che nei casi di YouTube e Twitter le classifiche sono molto simili, con rispettivamente 8 e 9 nodi in comune. Mentre nel caso del web di Stanford è presente un solo nodo in comune nelle due classifiche, in particolare il nodo in comune non ricopre la stessa posizione nelle due classifiche.

HITS Per le prove con il metodo HITS "originale" è stato usato il codice presentato nei capitoli precedenti, mentre per il metodo variante sono state eseguite 5 iterazioni nel primo step e 10 nel secondo.

Hubs Le classifiche ottenute calcolando il punteggio di hub nelle due modalità, sono molto simili alle classifiche ottenute nel paragrafo precedente. Infatti considerando la Tabella 5.2 si nota che nei casi di YouTube e Twitter sono presenti rispettivamente 6 e 8 nodi in comune, ma rispetto al caso precedente i nodi in comune ricoprono spesso posizioni differenti. Considerando il web di Stanford si nota che in questo caso non sono presenti nodi in comune.

Authorities Considerando ora le classifiche ottenute con i punteggi di authorities, i dati sono riportati nella Tabelle 5.3. I dati sono molto simili al caso dei punteggi di hubs, sono presenti infatti 7 e 8 nodi comuni rispettivamente nelle reti di YouTube e Twitter. In questo caso però è presente un nodo in comune nella classifica del web di Stanford.

YouTube		Twitter		Web Stanford	
Originale	Variante	Originale	Variante	Originale	Variante
691	**927**	**751**	**3238**	1901	3557
927	898	**3561**	**3561**	2339	1830
925	**925**	**3238**	**3305**	5806	9815
699	**971**	**3305**	**135**	7369	9802
971	881	**135**	**751**	1277	9647
837	**699**	467	**3560**	4559	9254
761	**837**	**608**	**608**	4747	9078
694	**761**	**3560**	**1698**	5764	8952
784	786	**1698**	667	6323	7729
690	871	3045	952	6675	7564

Tabella 5.2: Confronto con i punteggi di hub

YouTube		Twitter		Web Stanford	
Originale	Variante	Originale	Variante	Originale	Variante
90	**90**	**622**	**622**	**2260**	3163
316	**183**	**2702**	**2702**	2055	3408
183	**4**	**703**	**703**	3316	8176
69	**76**	**3347**	**1719**	3735	9067
4	37	**2322**	**695**	5053	8529
352	**316**	**1719**	**2322**	5149	6829
216	**216**	3209	**3347**	7380	4067
76	**65**	**695**	3417	7949	2365
397	136	**961**	**961**	8401	**2260**
65	299	3488	1634	9284	8983

Tabella 5.3: Confronto con i punteggi di authorities

Capitolo 6

Conclusioni

Dalle prove eseguite su reti reali si è notato che i due metodi di ranking, nonostante siano basati entrambi sul calcolo di autovettori e abbiano un significato di *importanza* analogo, restituiscono dei punteggi complessivamente diversi. Quindi i due metodi, su questo tipo di reti, non possono essere applicati indifferentemente uno dall'altro.
Dal confronto tra i metodi *originali* e quelli varianti si nota che le classifiche dei primi dieci nodi più importanti sono molto simili nei casi di Twitter e YouTube, mentre si ottengono risultati diversi nel caso del web di Stanford. Poiché la rete di Stanford è composta da molti sotto-web e sono presenti pochi archi e molti nodi dangling, si può ipotizzare che i risultati ottenuti con il metodo variante, utilizzando questa rete, siano dovuti a questa particolare struttura, differente dalle reti di Twitter e YouTube. Per applicare il metodo variante ad una rete come quella di Stanford, si possono quindi estrarre i sotto-web ed applicare il metodo ad ognuno di questi, considerando singolarmente ogni sotto-web.

Bibliografia

[1] P. Boldi, R. Posenato, M. Santini, S. Vigna. (2008). Traps and Pitfalls of Topic-Biased PageRank. 107-116. 10.1007/978-3-540-78808-$9_1$0.

[2] S. Brin, L. Page. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 33:10717, 1998.

[3] D. Gleich. PageRank Beyond the Web, 2014.

[4] J. Kleinberg. Authoritative Sources in a Hyperlinked Environment. Journal of the ACM, 1999.

[5] Amy N. Langville, Carl D. Meye. Googles PageRank and Beyond: The Science of Search Engine Rankings, 2006.

[6] L. Page, S. Brin, R. Motwani, T. Winograd. The PageRank Citation Ranking: Bringing Order to the WebTechnicalReport1999-0120, Computer Science Department, Stanford University, 1999

[7] G. Rodriguez. Algoritmi numerici, Pitagora Editrice Bologna, 2008.

[8] G. Rodriguez, S. Seatzu. Introduzione alla Matematica Applicata e Computazionale, Pitagora Editrice Bologna, 2010.

[9] A. Sanfilippo. Encyclopedia of Language and Linguistics, Graph Theory, Elsevier, 2006.

Sitografia

[1] Ryan A. Rossi, Nesreen K. Ahmed. The Network Data Repository with Interactive Graph Analytics and Visualization, 2015. http://networkrepository.com

[2] https://github.com/gephi/gephi/wiki/Datasets

[3] J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. arXiv.org:0810.1355, 2008. https://www.cise.ufl.edu/research/sparse/matrices/SNAP/web-Stanford.html

[4] https://blogs.mathworks.com/loren/