

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE
Corso di Laurea in Matematica

Strategie di accelerazione del metodo di PageRank

Tesi di Laurea in Analisi Numerica

Relatore:
Chiar.ma Prof.ssa
Valeria Simoncini

Presentata da:
Giulia Veronesi

Anno Accademico 2024/2025

Alla mia famiglia

Indice

Introduzione	i
1 Nozioni preliminari	1
2 Il metodo PageRank	3
2.1 Il vettore di PageRank	3
2.2 La matrice di Hyperlink e il problema agli autovalori	4
2.2.1 Il Web come grafo diretto	5
2.2.2 Modello del <i>random surfer</i> e interpretazione markoviana	7
3 Il calcolo del vettore di PageRank	11
3.1 Il metodo delle potenze	12
3.1.1 Teoria generale	12
3.1.2 Applicazione al PageRank	13
3.1.3 Scelta del metodo	15
3.1.4 Costo computazionale e velocità di convergenza	16
3.2 Estrapolazione di Aitken	17
3.2.1 Formulazione	17
3.2.2 Estrapolazione di Aitken nel metodo delle potenze	18
3.3 Estrapolazione quadratica	20
3.3.1 Formulazione	20
3.3.2 Estrapolazione quadratica nel metodo delle potenze	23
4 Risultati sperimentali	27
4.1 Matrice <i>web-NotreDame</i>	27
4.1.1 Metodo delle potenze	28
4.1.2 Estrapolazione di Aitken	29
4.1.3 Estrapolazione quadratica	34
4.2 Matrice <i>web-Google</i>	38

4.2.1	Metodo delle potenze	39
4.2.2	Estrapolazione di Aitken	40
4.2.3	Estrapolazione quadratica	43
4.3	Matrice <i>soc-LiveJournal1</i>	47
4.3.1	Metodo delle potenze	48
4.3.2	Estrapolazione di Aitken	48
4.3.3	Estrapolazione quadratica	49
4.3.4	Tempo di calcolo	52
4.4	Conclusioni	53
	Bibliografia	55
	Ringraziamenti	57

Introduzione

Con la crescita esponenziale del Web e della quantità di informazioni disponibili online, è emersa la necessità di sviluppare strumenti efficienti per organizzare e classificare le pagine web in base alla loro rilevanza. In questo contesto si colloca l'algoritmo *PageRank*, proposto alla fine degli anni Novanta da Lawrence Page e Sergey Brin, fondatori del motore di ricerca Google. L'idea alla base dell'algoritmo è quella di assegnare a ciascuna pagina web un punteggio che ne misuri l'importanza relativa all'interno della rete di collegamenti ipertestuali del Web.

All'interno di questo elaborato, dopo aver introdotto il vettore di PageRank, la matrice di Hyperlink, la Google matrix e averne discusso le principali proprietà, entreremo nel vivo del problema del calcolo del vettore di PageRank. In particolare, oltre a presentare il metodo classico utilizzato per il suo calcolo, descriveremo due tecniche estrapolative in grado di accelerarne significativamente la convergenza. Infine, verranno presentati alcuni esperimenti numerici su collezioni di pagine Web di dimensioni molto diverse tra loro, al fine di valutare gli effetti di tali tecniche.

Capitolo 1

Nozioni preliminari

Definizione 1.1. Un **grafo** è una coppia $G = (V, E)$ in cui V è un insieme di nodi (o vertici) ed E è un insieme di archi che collegano i nodi tra loro.

Un grafo si dice **connesso** se per ogni coppia di vertici esiste una sequenza di archi, (detta cammino), che li collega.

Definizione 1.2. Un **grafo diretto** è un grafo i cui archi sono orientati. Un grafo diretto si dice **fortemente connesso** se per ogni coppia di nodi (V_i, V_j) esiste una sequenza di archi orientati che porta da V_i a V_j .

Definizione 1.3. La **matrice delle adiacenze** di un grafo (diretto) con n vertici è una matrice di dimensione $n \times n$ A i cui elementi sono definiti come segue:

$$a_{i,j} = \begin{cases} 1 & \text{se esiste un arco (orientato) da } V_j \text{ a } V_i \\ 0 & \text{altrimenti} \end{cases}$$

Osservazione 1.4. La matrice delle adiacenze di un grafo non diretto è simmetrica, in quanto gli archi non presentano un'orientazione. Nel caso di un grafo diretto, invece, questa proprietà non vale in generale, poiché gli archi sono orientati. Per ogni $i = 1, \dots, n$, la colonna i -esima della matrice di adiacenza A contiene degli 1 nelle posizioni corrispondenti ai vertici che hanno un arco diretto verso V_i e degli 0 nelle restanti. Analogamente, la riga i -esima contiene degli 1 nelle posizioni dei vertici verso cui parte un arco da V_i . In altre parole, la colonna i -esima descrive gli archi entranti in V_i , mentre la riga i -esima descrive gli archi uscenti da V_i .

Definizione 1.5. Una matrice a coefficienti reali non negativi si dice **stocastica per colonne** se la somma degli elementi su ogni colonna è uguale a 1.

Definizione 1.6. Un **processo stocastico** è una successione di variabili aleatorie $\{X_t, t \in T\}$ a valori in $\{S_1, S_2, \dots, S_n\}$ (detto **spazio degli stati** del processo).

Il parametro t è generalmente pensato come il tempo, e X_t rappresenta lo stato del processo al tempo t .

Definizione 1.7. Una **catena di Markov** è un processo stocastico che soddisfa la **proprietà di Markov**:

$$P(X_{t+1} = S_j | X_t = S_{i_t}, X_{t-1} = S_{i_{t-1}}, \dots, X_0 = S_{i_0}) = P(X_{t+1} = S_j | X_t = S_{i_t})$$

Osservazione 1.8. Dalla proprietà di Markov segue che lo stato del processo all'istante successivo dipende solo da quello corrente e non anche dai precedenti.

Definizione 1.9. Data $(X_t)_{t \in \mathbb{N}}$ una catena di Markov, la **matrice di transizione** $P = (p_{ij})$ è la matrice delle **probabilità di transizione**

$$p_{ij} = P(X_{t+1} = i | X_t = j).$$

Definizione 1.10. Una matrice stocastica P si dice **aperiodica** se la catena di Markov da essa definita è aperiodica, cioè se per ogni stato i vale:

$$\text{MCD}\{n \geq 1 : (P^n)_{ii} > 0\} = 1,$$

ossia se il massimo comune divisore dei tempi $n \geq 1$ per i quali la probabilità di tornare nello stato i dopo n passi è positiva è uguale a 1.

Teorema 1.11 (Teorema Ergodico per le catene di Markov). Sia $(X_n)_{n \geq 0}$ una catena di Markov finita, irriducibile e aperiodica, con matrice di transizione P . Allora esiste un'unica distribuzione stazionaria π tale che, indipendentemente dalla distribuzione iniziale, la distribuzione di X_n converge a π al crescere di n .

Teorema 1.12 (Teorema di Cayley-Hamilton). Sia $A \in \mathbb{R}^{n \times n}$. Se $p_A(\lambda)$ è il polinomio caratteristico di A , allora $p_A(A) = 0$.

Capitolo 2

Il metodo PageRank

2.1 Il vettore di PageRank

La definizione del vettore di PageRank segue da un concetto fondamentale, quello di **importanza** di una pagina Web.

Consideriamo una collezione di pagine Web; un'approssimazione dell'importanza o qualità di una determinata pagina può essere ottenuta, in modo piuttosto intuitivo, contando il numero di links che puntano a quest'ultima (detti **inlinks** o **backlinks**).

Il PageRank estende questo concetto conferendo un peso maggiore agli inlinks provenienti da pagine a loro volta importanti: questo criterio aggiuntivo è fondamentale, ad esempio, ai fini di non conferire un'importanza ingiustamente alta a eventuali pagine a cui puntano tante pagine spam.

L'idea che permette di dare la definizione di PageRank di una pagina Web P_k è la seguente: ciascuna pagina P_j che punta ad essa le conferisce una frazione della sua importanza, uguale a $1/l_j$, dove l_j è il numero dei links della pagina P_j , (detti **outlinks**). L'importanza di P_k è, dunque, data dalla somma dei contributi ricevuti dalle pagine che puntano ad essa, ciascuno ponderato rispetto all'importanza della rispettiva pagina.

Definizione 2.1. *Siano P_i l' i -esima pagina della nostra collezione, l_i il numero dei suoi outlinks, \mathcal{S}_{P_i} l'insieme delle pagine che puntano ad essa. Il **PageRank** (o importanza) della pagina k -esima è dato da:*

$$\mathcal{I}(P_k) = \sum_{P_i \in \mathcal{S}_{P_k}} \frac{1}{l_i} \mathcal{I}(P_i) \quad (2.1)$$

Definizione 2.2. Il **vettore di PageRank** è il vettore \mathcal{I} di lunghezza uguale al numero di pagine Web della collezione in esame, che ha come i -esima componente l'importanza della i -esima pagina Web:

$$\mathcal{I} = \begin{pmatrix} \mathcal{I}(P_1) \\ \mathcal{I}(P_2) \\ \vdots \\ \mathcal{I}(P_n) \end{pmatrix}$$

Osservazione 2.3. Ognuna delle pagine che puntano a P_k cede ad essa una parte della propria importanza inversamente proporzionale al numero degli outlinks, pertanto il contributo di una pagina è maggiore se quest'ultima punta a un basso numero di pagine, ossia se ha un basso numero di outlinks.

Osservazione 2.4. Dall'equazione (2.1), notiamo che il calcolo del PageRank di ogni pagina sfrutta quelli dei suoi inlinks che, però, sono anch'essi non noti. Pertanto, è necessario ricorrere a una procedura iterativa.

A tal fine, nel prossimo paragrafo presentiamo una formulazione matriciale del problema che sarà definitiva per determinare il vettore di PageRank.

2.2 La matrice di Hyperlink e il problema agli autovalori

Consideriamo una collezione di n pagine Web.

Definizione 2.5. La **matrice di Hyperlink** è una matrice $H \in \mathbb{R}^{n \times n}$ i cui elementi sono definiti come segue:

$$h_{i,j} = \begin{cases} \frac{1}{l_j} & \text{se } P_j \text{ punta a } P_i \\ 0 & \text{altrimenti} \end{cases} \quad (2.2)$$

dove l_j è il numero degli outlinks della pagina P_j .

Osservazione 2.6. Gli elementi della matrice di Hyperlink sono tutti non negativi e, fatta eccezione per le colonne nulle (corrispondenti alle pagine senza outlinks), la somma degli elementi su ciascuna colonna è uguale a 1.

Da quest'ultima definizione e da quella di PageRank, segue che:

$$\mathcal{I} = H\mathcal{I},$$

ossia \mathcal{I} è autovettore relativo all'autovalore $\lambda = 1$ della matrice di Hyperlink.

La matrice H per come è stata definita nella (2.2) presenta, però, alcune problematiche:

1. La mancata gestione dei cosiddetti **dangling nodes**, ossia delle pagine senza outlinks (anche, banalmente, pagine corrispondenti a file pdf o immagini) che, nella matrice, sono rappresentate da colonne nulle,
2. Non è detto nè che H abbia un autovalore 1, nè che esso sia semplice in caso sia presente nello spettro di H .

Per comprendere come ovviare a queste problematiche, è utile collocare la matrice di Hyperlink all'interno di due quadri teorici equivalenti: il primo relativo prettamente alla struttura del Web, il secondo riguardante il modello probabilistico del random surfer e, di conseguenza, le catene di Markov.

2.2.1 Il Web come grafo diretto

La struttura del Web è modellabile in modo naturale mediante un **grafo diretto** i cui nodi rappresentano le pagine Web e gli archi orientati gli inlinks e gli outlinks.

Il ruolo della matrice di Hyperlink in questa rappresentazione è quello di **matrice delle adiacenze pesata**, infatti $h_{i,j}$ è non nullo se esiste un arco orientato dalla pagina j alla pagina i .

Per risolvere il problema della gestione dei dangling nodes, data $\{P_i\}_{1,\dots,n}$ la nostra collezione di pagine Web, definiamo il vettore d di lunghezza n come segue:

$$d_j = \begin{cases} 1 & \text{se } l_j = 0 \\ 0 & \text{altrimenti} \end{cases} \quad (2.3)$$

e modifichiamo H :

$$G = H + \frac{1}{n}\mathbf{1}d^T \quad (2.4)$$

G è una matrice stocastica per colonna, ovvero tutti i suoi elementi sono non negativi e la somma degli elementi su ogni colonna è 1, da ciò segue che:

$$G^T\mathbf{1} = \mathbf{1},$$

perciò G ha autovalore $\lambda = 1$, da cui

$$G\mathcal{I} = \mathcal{I}.$$

Affinchè \mathcal{I} sia l'unico autovettore di λ , bisogna assicurare che $\lambda = 1$ sia autovalore semplice. Per fare ciò, ricordiamo le definizioni di matrice irriducibile e di grafo diretto fortemente connesso.

Definizione 2.7. $A \in \mathbb{R}^{n \times n}$ si dice **irriducibile** se esiste una matrice di permutazione T tale che:

$$TAT^T = \begin{bmatrix} X & Y \\ 0 & Z \end{bmatrix}$$

con X, Z quadrate. Altrimenti si dice che A è **irriducibile**.

Definizione 2.8. Il grafo diretto di una matrice A si dice **fortemente connesso** se per ogni elemento $A_{i,j}$ diverso da zero, ossia per ogni coppia di vertici P_i, P_j , esiste un cammino nel grafo che porta da P_i a P_j .

Teorema 2.9 (Variante del teor. di Perron-Frobenius). Sia $A \in \mathbb{R}^{n \times n}$ irriducibile e stocastica per colonna. Allora:

1. L'autovalore dominante λ_1 è uguale a 1;
2. Esiste un unico autovettore associato a λ_1 con componenti tutte positive ed è l'unico autovettore con tutte componenti non negative;
3. Se A ha tutti elementi positivi, si ha $|\lambda_i| < 1$ per ogni altro autovalore dello spettro di A .

Inoltre vale il seguente:

Teorema 2.10. Una matrice A è irriducibile se e solo se il suo grafo diretto è fortemente connesso.

Da questi ultimi due risultati, segue che affinché l'autovalore dominante della matrice G sia 1 e sia semplice è sufficiente che G , oltre a essere stocastica per colonne, sia irriducibile.

Per assicurare l'irriducibilità di G , come abbiamo visto, è sufficiente che il suo grafo diretto sia fortemente connesso, ossia che ciascuna pagina Web sia raggiungibile da qualsiasi altra.

A tal fine, procediamo "aggiungendo" un link da ogni pagina verso ogni altra pagina:

$$G = \alpha H + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T, \quad \alpha \in [0, 1] \quad (2.5)$$

La matrice di Hyperlink aggiustata per stocasticità per colonne e irriducibilità, è detta **Google matrix**.

Teorema 2.11. *Sia $\text{spec}(H) = \{1, \lambda_2, \dots, \lambda_n\}$. Allora*

$$\text{spec}(G) = \{1, \alpha\lambda_2, \dots, \alpha\lambda_n\}$$

Dimostrazione. Sia $\bar{\mathbf{1}} = \frac{1}{\sqrt{n}} \mathbf{1}$, si ha $G^T \bar{\mathbf{1}} = \alpha G^T \bar{\mathbf{1}} + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T \bar{\mathbf{1}} = \bar{\mathbf{1}}$.

Sia ora $U = [\bar{\mathbf{1}}, U_0]$ unitaria, allora $U^* H U = \begin{bmatrix} 1 & 0 \\ * & T \end{bmatrix}$, da cui

$$U^* G U = \alpha \begin{bmatrix} 1 & 0 \\ * & T \end{bmatrix} + (1 - \alpha) \frac{1}{n} n e_1 \bar{\mathbf{1}}^T U = \begin{bmatrix} 1 & 0 \\ * & \alpha T \end{bmatrix}.$$

□

Osservazione 2.12. *Da questo teorema segue che anche nel caso in cui H abbia autovalore 1 multiplo, G ha autovalore 1 semplice.*

Il fatto che l'autovettore relativo all'autovalore principale 1 sia unico e che esso sia proprio il vettore di PageRank, garantirà di poter calcolare quest'ultimo mediante un procedimento iterativo.

2.2.2 Modello del *random surfer* e interpretazione markoviana

Analizziamo un approccio probabilistico attraverso il quale sarà possibile giungere alle stesse conclusioni del paragrafo precedente. A tal fine, presentiamo un modello che è stato introdotto da Sergey Brin e Lawrence Page nel loro lavoro originale sull'algoritmo PageRank[1, 2].

Il PageRank può essere pensato come un modello del comportamento dell'utente. Immaginiamo un "navigatore" (*random surfer*) che passa in modo casuale da una pagina all'altra secondo la struttura del grafo diretto del Web: quando arriva a una pagina con più outlinks, ne sceglie uno in modo casuale, continuando questo processo di decisioni casuali indefinitamente.

Nel lungo termine, il tempo che il navigatore trascorre su una determinata pagina è una misura dell'importanza di quest'ultima, infatti, se seguendo in modo casuale la struttura del Web, si è ritrovato ripetutamente su tale pagina, essa dev'essere stata puntata da altre pagine importanti.

Osservazione 2.13. *Il modello del random surfer è privo di memoria, nel senso che la scelta della pagina successiva dipende unicamente dalla pagina corrente e non dalla storia pregressa della navigazione, pertanto possiamo dire che esso soddisfa la **proprietà di Markov**.*

Consideriamo la **catena di Markov** introdotta dal percorso casuale del navigatore sul grafo diretto del Web, i cui stati sono dati dai nodi del grafo e la cui matrice di transizione H descrive la probabilità di transizione dalla j -esima pagina alla i -esima come $H_{ij} = \frac{1}{l_j}$, (dove l_j è il numero di outlinks della pagina P_j).

Affinché H possa essere interpretata come matrice di transizione di una catena di Markov, è necessario che ogni nodo presenti almeno un outlink, ovvero che H non contenga colonne nulle, poiché ciascuna colonna deve rappresentare una distribuzione di probabilità.

Osservazione 2.14. *Nel modello del random surfer, la presenza di dangling nodes causerebbe l'arresto forzato del navigatore ogni qualvolta egli capitasse su uno di essi.*

Per ovviare a questo problema, si introduce l'**aggiustamento stocastico** di H effettuato in (2.4), ossia $G = H + \frac{1}{n}\mathbf{1}d^T$.

In termini del random surfer, l'effetto dell'aggiunta di questo termine di rango 1 modifica le probabilità di transizione cosicché in caso il navigatore incorra in un dangling node, egli salta in modo casuale in un'altra pagina nell'istante seguente, secondo la distribuzione di probabilità data dal vettore $\frac{1}{n}\mathbf{1}$.

Dal Teorema Ergodico per le catene di Markov (Teorema 1.11), segue che la catena di Markov definita dalla matrice G ha un'unica distribuzione stazionaria di probabilità se G è aperiodica e irriducibile.

Modifichiamo G come in (2.5):

$$G = \alpha H + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T, \quad \alpha \in [0, 1],$$

Osservazione 2.15. *In termini del modello del random surfer, α è il parametro che controlla la proporzione di tempo in cui il navigatore segue la struttura del Web: l' $\alpha \cdot 100\%$ del tempo egli segue il grafo diretto, nel restante $(1 - \alpha) \cdot 100\%$ interrompe il percorso teletrasportandosi in una nuova pagina Web scelta casualmente.*

Il "teletrasporto" è uniforme in quanto la matrice data da $\frac{1}{n} \mathbf{1}\mathbf{1}^T$ è uniforme, pertanto ogni pagina ha la stessa probabilità di essere scelta.

Si ottiene, così, una matrice con le seguenti proprietà:

- Stocasticità per colonne;
- Irriducibilità, infatti ogni pagina è direttamente connessa a ogni altra pagina;
- Aperiodicità: i self-loops $G_{ii} > 0$ per ogni i creano aperiodicità.

Sono così soddisfatte le ipotesi del Teorema Ergodico, pertanto G ha un'unica distribuzione stazionaria di probabilità, da cui segue, anche in questo caso, l'unicità del vettore di PageRank associato a G .

Capitolo 3

Il calcolo del vettore di PageRank

Il problema del PageRank può essere impostato in due modi:

- Come problema agli autovalori per \mathcal{I} :

$$\mathcal{I} = G\mathcal{I},$$

- Come sistema lineare omogeneo nell'incognita \mathcal{I} :

$$(I - G)\mathcal{I} = 0,$$

dove I è la matrice identità di dimensione $n \times n$.

Osservazione 3.1. *Entrambi gli approcci sono soggetti alla normalizzazione $\|\mathcal{I}\|_1$, che assicura che \mathcal{I} sia un vettore di probabilità.*

Per via delle enormi dimensioni che il grafo del Web può raggiungere, (alcuni miliardi di nodi), il calcolo del vettore di PageRank porta con sé un problema non banale: quello della memorizzazione dei dati.

Per quanto riguarda l'impostazione del problema come sistema lineare omogeneo, l'utilizzo di un *metodo diretto* sarebbe impraticabile per via dello spazio di archiviazione dei dati che esso richiederebbe; per questo motivo, è necessario ricorrere all'applicazione di un *metodo iterativo*.

All'interno di questo elaborato ci soffermeremo sul calcolo del vettore di PageRank nell'ottica di un problema agli autovalori.

3.1 Il metodo delle potenze

3.1.1 Teoria generale

Il metodo delle potenze è uno dei più semplici metodi iterativi per determinare l'autocoppia dominante (in modulo) di una matrice.

Siano $A \in \mathbb{R}^{n \times n}$ e $x^{(0)} \in \mathbb{R}^n$ un arbitrario vettore iniziale di norma unitaria. Supponiamo che A sia diagonalizzabile con autovalori $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, $|\lambda_1|$ semplice e relativi autovettori u_1, \dots, u_n . Consideriamo per k intero positivo l'iterazione costituita da:

$$\begin{aligned}\hat{x}^{(k)} &= Ax^{(k-1)} \\ x^{(k)} &= \hat{x}^{(k)} / \|\hat{x}^{(k)}\|_2 \\ \lambda^{(k)} &= (x^{(k)})^* Ax^{(k)}\end{aligned}\tag{3.1}$$

Siccome A è diagonalizzabile, è possibile scrivere $x^{(0)}$ come combinazione lineare degli autovettori u_1, \dots, u_n :

$$x^{(0)} = \sum_{i=1}^n \alpha_i u_i, \quad \alpha_i \in \mathbb{R}, \quad i = 1, \dots, n.$$

Poichè $Au_i = \lambda_i u_i$, raccogliendo opportunamente segue:

$$A^k x^{(0)} = \alpha_1 \lambda_1^k \left(u_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k u_i \right), \quad k = 1, 2, \dots\tag{3.2}$$

Dal momento che $|\lambda_i/\lambda_1| < 1$ per ogni i , le componenti di $A^k x^{(0)}$ (e quindi di $x^{(k)}$) nelle direzioni degli autovettori u_2, \dots, u_n decresceranno, mentre quella lungo la direzione di u_1 sarà sempre più significativa al crescere di k .

Il metodo delle potenze, dunque, genera due successioni: $\{x^{(k)}\}_{k \in \mathbb{N}}$ e $\{\lambda^{(k)}\}_{k \in \mathbb{N}}$ che convergono, rispettivamente, a u_1 e a λ_1 per $k \rightarrow +\infty$.

Osservazione 3.2. *L'ipotesi di semplicità in modulo dell'autovalore λ_1 è cruciale ai fini della convergenza del metodo delle potenze, infatti, in caso contrario, al variare del vettore iniziale $x^{(0)}$ varierebbe il limite della successione $\{x^{(k)}\}_{k \in \mathbb{N}}$, che convergerebbe a un generico vettore dell'autospazio associato a λ_1 .*

Consideriamo, ora, il seguente risultato, che fornisce delle importanti informazioni sulla velocità di convergenza del metodo:

Teorema 3.3. *Sia $A \in \mathbb{R}^{n \times n}$ diagonalizzabile con autovalori $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, $|\lambda_1|$ semplice e sia u_1 l'autovettore a esso corrispondente. Posto $x^{(0)} = \sum_{i=1}^n \alpha_i u_i$ con $\alpha_i \in \mathbb{R}$ per ogni i , assumendo $\alpha_1 \neq 0$, esiste $C > 0$ tale che:*

$$\|x^{(k)} - u_1\|_2 \leq C \left| \frac{\lambda_2}{\lambda_1} \right|^k, \quad k \geq 1, \quad (3.3)$$

con $x^{(k)}$ normalizzato in modo opportuno.

Dimostrazione. Poniamo

$$v^{(k)} = \frac{A^k x^{(0)}}{\alpha_1 \lambda_1^k} \stackrel{(3.2)}{=} u_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k u_i,$$

si ha

$$\|v^{(k)} - u_1\|_2 \leq \sum_{i=2}^n \left| \frac{\alpha_i}{\alpha_1} \right| \left| \frac{\lambda_i}{\lambda_1} \right|^k \|u_i\|_2 \leq \left| \frac{\lambda_2}{\lambda_1} \right|^k \sum_{i=2}^n \left| \frac{\alpha_i}{\alpha_1} \right| \|u_i\|_2 =: C \left| \frac{\lambda_2}{\lambda_1} \right|^k.$$

$v^{(k)}$ differisce da $x^{(k)}$ per un fattore di normalizzazione. Segue la tesi. \square

Osservazione 3.4. *L'ipotesi di diagonalizzabilità della matrice può essere indebolita. Ai fini della dimostrazione è infatti sufficiente assumere che l'iterata iniziale $x^{(0)}$ appartenga al sottospazio generato dagli autovettori della matrice relativi agli autovalori di A $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_m|$, $|\lambda_1|$ semplice.*

Osservazione 3.5. *Dal teorema segue che la convergenza di $\{x^{(k)}\}_{k \in \mathbb{N}}$ a u_1 è lineare rispetto al fattore $|\lambda_2/\lambda_1|$, pertanto, maggiore è la distanza di λ_1 dal resto dello spettro, maggiore sarà la velocità di convergenza del metodo.*

Il metodo delle potenze richiede la definizione di un *criterio d'arresto*. Una quantità che può svolgere questo servizio è la norma del residuo $r^{(k)} = Ax^{(k)} - \lambda^{(k)}x^{(k)}$; usando un criterio relativo, il test può essere scritto come $\|r^{(k)}\|_2/|\lambda^{(k)}|$.

Dopo aver definito una tolleranza *tol* anche sulla base del contesto nel quale si sta operando, se k è il primo valore tale per cui $\|r^{(k)}\|_2/|\lambda^{(k)}| < tol$ allora il processo si arresterà al termine della k -esima iterazione.

3.1.2 Applicazione al PageRank

Consideriamo una collezione di n pagine Web, G la corrispondente Google matrix con i relativi autovalori $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_m|$, $\lambda_1 = 1$ semplice e autovettori u_1, \dots, u_m .

Supponiamo che il vettore iniziale $x^{(0)}$ sia sufficientemente generale in modo che sia esprimibile come combinazione lineare di tutti gli autovettori di G . Allora, scalando opportunamente $x^{(0)}$, possiamo scrivere:

$$x^{(0)} = u_1 + \alpha_2 u_2 + \cdots + \alpha_m u_m, \quad \alpha_i \in \mathbb{R}, \quad i = 2, \dots, m.$$

Poichè $\lambda_1 = 1$, possiamo scrivere

$$x^{(1)} = Gx^{(0)} = u_1 + \alpha_2 \lambda_2 u_2 + \cdots + \alpha_m \lambda_m u_m$$

e, per k intero positivo,

$$x^{(k)} = G^k x^{(0)} = u_1 + \alpha_2 \lambda_2^k u_2 + \cdots + \alpha_m \lambda_m^k u_m.$$

Osservazione 3.6. *Dal momento che l'autovalore dominante della Google matrix G è noto, l'applicazione del metodo delle potenze al problema del PageRank richiede la costruzione di una sola successione, ossia $\{x^{(k)}\}_{k \in \mathbb{N}}$, convergente all'autovettore dominante u_1 , il quale, come visto in precedenza, coincide con il vettore di PageRank.*

Osservazione 3.7. *Dal punto di vista teorico, la normalizzazione dell'iterata non influisce sulla convergenza del metodo, tuttavia, in implementazioni numeriche, l'introduzione di una normalizzazione può risultare vantaggiosa per migliorare la stabilità numerica contenendo l'effetto degli errori di arrotondamento.*

Poichè vengono trattate enormi quantità di dati, risulta conveniente sostituire la norma 1 alla norma euclidea, in quanto il suo calcolo è meno oneroso dal punto di vista computazionale, richiedendo un numero inferiore di operazioni aritmetiche.

Per per k intero positivo e tol fissata, modificando la (3.1) per il PageRank e inserendo un criterio di arresto per il contesto considerato si ha:

function $x = Power(tol, v)$

$x^{(0)} = v$

per $k = 1, 2, \dots$

$$\hat{x}^{(k)} = Gx^{(k-1)}$$

$$x^{(k)} = \hat{x}^{(k)} / \|\hat{x}^{(k)}\|_1$$

Se $\|x^{(k)} - x^{(k-1)}\|_1 < tol$, Stop

3.1.3 Scelta del metodo

La scelta di questo metodo è giustificata da alcune ragioni:

La Google matrix G in (2.5) è una matrice completamente densa e quindi non viene mai costruita esplicitamente. Infatti ogni iterazione può essere espressa in termini della matrice H che, a differenza di G , è molto sparsa. L'operazione matrice-vettore viene quindi effettuata come segue:

$$\begin{aligned} x^{(k)} = Gx^{(k-1)} &= \left(\alpha H + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T \right) x^{(k-1)} = \\ &= \alpha Hx^{(k-1)} + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T x^{(k-1)} \end{aligned} \quad (3.4)$$

Per massimizzare l'efficienza di questa scrittura è fondamentale prendere alcuni accorgimenti pratici:

Osservazione 3.8. Per quanto riguarda l'addendo $\alpha Hx^{(k-1)}$, eseguire i prodotti nell'ordine $\alpha (Hx^{(k-1)})$ permette di risparmiare notevolmente sul numero di elementi da moltiplicare per α , infatti, posto m il numero di elementi non nulli della riga i -esima di $\alpha (Hx^{(k-1)})$,

- se $m \in \{0, 1\}$, il numero di prodotti da eseguire non cambia,
- se $m > 1$, bisognerà eseguire un solo prodotto (quello tra α e l' i -esima componente del vettore $Hx^{(k-1)}$), anzichè m .

Osservazione 3.9. In merito all'addendo $(1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T x^{(k-1)}$, è cruciale dal punto di vista computazionale eseguire prima il prodotto $\mathbf{1}^T x^{(k-1)}$, in modo da moltiplicare i restanti fattori per uno scalare.

Eseguire per primo il prodotto $\mathbf{1}\mathbf{1}^T$ vorrebbe dire costruire una matrice $n \times n$ a ogni iterazione, facendo esplodere il costo computazionale.

Un altro vantaggio dell'utilizzo del metodo delle potenze per il calcolo del vettore di PageRank è che esso è un metodo cosiddetto **matrix-free**, ovvero tale per cui i coefficienti della matrice sono accessibili soltanto attraverso le moltiplicazioni matrice-vettore.

Non viene eseguita nessuna manipolazione di H che, nonostante la sparsità della matrice, risulterebbe molto onerosa per via della sua grandezza e della mancanza di una

struttura.

Un ulteriore pregio di questo metodo è costituito dal fatto che ogni iterazione salvi in memoria un solo vettore, ovvero l'iterata corrente.

3.1.4 Costo computazionale e velocità di convergenza

Il costo computazionale di una iterazione del metodo delle potenze applicato al PageRank è principalmente dovuto al prodotto matrice-vettore $Gx^{(k-1)}$.

Con matrici piene, il costo di questa operazione è $O(n^2)$, ma poiché, come abbiamo visto, tale prodotto viene effettuato utilizzando H , la sparsità di quest'ultima fa sì che il costo computazionale di ogni iterazione diventi $O(n)$.

Velocità di convergenza: Sia $\{1, \mu_2, \dots, \mu_m\}$ l'insieme degli autovalori di H e $\{1, \lambda_2, \dots, \lambda_m\}$ quello di G . Dal (Teorema 2.11), si ha $\text{spec}(G) = \{1, \alpha\mu_2, \dots, \alpha\mu_m\}$, da cui segue

$$|\lambda_2| = \alpha|\mu_2| \leq \alpha,$$

infatti sappiamo che $|\mu_2| \leq 1$.

Come abbiamo visto nel (Teorema 3.3), la velocità di convergenza del metodo è lineare rispetto al rapporto $|\lambda_2/\lambda_1|$ che, nel nostro caso, può essere maggiorato come segue:

$$\left| \frac{\lambda_2}{\lambda_1} \right| = |\lambda_2| \leq \alpha.$$

Risulta dunque evidente l'importanza del ruolo di $\alpha \in (0, 1)$ per quanto riguarda la velocità di convergenza del metodo: a bassi valori di α corrispondono bassi valori di $|\lambda_2|$, pertanto $\|x^{(k)} - u_1\|_1 (\leq C|\lambda_2/\lambda_1|^k)$ convergerà velocemente a zero.

Alla luce di questo, verrebbe naturale optare per valori bassi di α , in modo da velocizzare il più possibile la convergenza, ma ciò è sconsigliato. Recuperiamo la definizione della matrice G :

$$G = \alpha H + (1 - \alpha) \frac{1}{n} \mathbf{1}\mathbf{1}^T.$$

La *probabilità di teletrasporto* $1 - \alpha$ del navigatore nel modello del random surfer, è il parametro che determina il peso della componente data da $\frac{1}{n} \mathbf{1}\mathbf{1}^T$: al crescere di $1 - \alpha$, tale componente assume un ruolo sempre più rilevante, a scapito dell'informazione strutturale contenuta nella matrice di Hyperlink. Pertanto:

- al calare di α corrisponde un aumento della velocità di convergenza e del parametro $1 - \alpha$ a discapito della reale struttura del grafo del Web,

- a valori di α prossimi a 1 segue un notevole rallentamento della convergenza del metodo, ma una significativa valorizzazione della struttura effettiva del Web.

Per questi motivi, il parametro generalmente utilizzato è $\alpha = 0.85$, in modo da non penalizzare oltremodo nessuno dei due aspetti.¹

Nei prossimi paragrafi, presenteremo delle varianti del metodo delle potenze che permetteranno di velocizzare notevolmente il calcolo del vettore di PageRank apportando un costo computazionale aggiuntivo minimo e permettendoci di incrementare ulteriormente il valore di α (fino a 0.99) senza far esplodere il numero di iterazioni necessarie alla convergenza.

3.2 Estrapolazione di Aitken

Incominciamo introducendo un algoritmo detto *estrapolazione di Aitken*, che sviluppiamo come segue.

3.2.1 Formulazione

Assumiamo che l'iterata $x^{(k-2)}$ possa essere espressa come combinazione lineare dei primi due autovettori della Google matrix G . In questo modo sarà possibile risolvere per l'autovettore principale u_1 in forma chiusa utilizzando le iterate successive $x^{(k-2)}, x^{(k-1)}, x^{(k)}$.

Poichè $x^{(k-2)}$ può solo essere approssimata come combinazione lineare di tali autovettori, l'autovettore dominante ottenuto sarà solo una stima del vero u_1 ; tuttavia, vedremo che questa stima diventerà sempre più accurata al crescere di k .

Stando a queste premesse, possiamo scrivere:

$$x^{(k-2)} = u_1 + \alpha_2 u_2, \quad \alpha_2 \in \mathbb{R} \quad (3.5)$$

Poichè $\lambda_1 = 1$, possiamo formulare le due iterate successive come:

$$x^{(k-1)} = Gx^{(k-2)} = u_1 + \alpha_2 \lambda_2 u_2 \quad (3.6)$$

e

$$x^{(k)} = Gx^{(k-1)} = u_1 + \alpha_2 \lambda_2^2 u_2. \quad (3.7)$$

¹Tale impostazione risale alle prime implementazioni dell'algoritmo effettuate dai suoi ideatori, Larry Page e Sergey Brin.

Siano, ora, $g, h \in \mathbb{R}^n$, la cui i -esima componente è definita come segue per $i = 1, \dots, n$:

$$g_i := (x_i^{(k-1)} - x_i^{(k-2)})^2, \quad (3.8)$$

$$h_i := x_i^{(k)} - 2x_i^{(k-1)} + x_i^{(k-2)}, \quad (3.9)$$

dove $x_i^{(j)}$ rappresenta la i -esima componente del vettore $x^{(j)}$.

Sostituendo (3.5), (3.6) e (3.7) e raccogliendo opportunamente, otteniamo:

$$g_i = \alpha_2^2 (\lambda_2 - 1)^2 (u_2)_i^2, \quad (3.10)$$

$$h_i = \alpha_2 (\lambda_2 - 1)^2 (u_2)_i. \quad (3.11)$$

Definiamo f_i come il quoziente g_i/h_i , ovvero:

$$f_i := \frac{g_i}{h_i} = \frac{\alpha_2^2 (\lambda_2 - 1)^2 (u_2)_i^2}{\alpha_2 (\lambda_2 - 1)^2 (u_2)_i} = \alpha_2 (u_2)_i, \quad (3.12)$$

per cui

$$f = \alpha_2 u_2. \quad (3.13)$$

A questo punto, dall'equazione (3.5) otteniamo una soluzione in forma chiusa per u_1 :

$$u_1 = x^{(k-2)} - \alpha_2 u_2 = x^{(k-2)} - f \quad (3.14)$$

Ricordiamo che quest'ultima formulazione rappresenta solo un'approssimazione dell'autovettore dominante, in quanto è espressa in termini delle iterate successive $x^{(k-2)}, x^{(k-1)}, x^{(k)}$, le quali abbiamo supposto essere formulabili come combinazione lineare di u_1 e u_2 .

3.2.2 Estrapolazione di Aitken nel metodo delle potenze

L'estrapolazione di Aitken può essere eseguita periodicamente all'interno del metodo delle potenze ai fini di accelerarne la convergenza.

Tale estrapolazione deriva dal noto metodo Δ^2 di Aitken: riprendendo le definizioni (3.10) e (3.11), infatti, si ha

$$g_i = (x_i^{(k-1)} - x_i^{(k-2)})^2 = (\Delta x_i^{(k-2)})^2,$$

$$h_i = x_i^{(k)} - 2x_i^{(k-1)} + x_i^{(k-2)} = (x_i^{(k)} - x_i^{(k-1)}) - (x_i^{(k-1)} - x_i^{(k-2)}) = \Delta^2 x_i^{(k-2)}.$$

Il vettore g descrive l'ampiezza quadratica dello spostamento tra due iterate consecutive (rappresentando il quadrato della *prima differenza* $\Delta x^{(k-2)}$), mentre h ne quantifica

la variazione attraverso la *seconda differenza* $\Delta^2 x^{(k-2)}$. In particolare, valori piccoli di h indicano che la sequenza procede con incrementi pressoché costanti, mentre valori elevati segnalano una variazione significativa tra incrementi successivi. In altre parole, h misura quanto la successione si discosta da un andamento localmente lineare; una seconda differenza di piccola entità è indicativa di un regime asintotico regolare, condizione che rende efficace l'applicazione dell'estrapolazione di Aitken.

Il rapporto

$$f_i = \frac{g_i}{h_i} = \frac{(\Delta x_i^{(k-2)})^2}{\Delta^2 x_i^{(k-2)}}$$

combina informazioni sull'ampiezza degli incrementi e sulla loro variazione, consentendo di ottenere una stima asintotica del contributo residuo alla convergenza. Sottraendo il vettore f all'iterata corrente, l'estrapolazione di Aitken compie un avanzamento che riduce il termine dominante dell'errore, corrispondente alla componente di convergenza lineare associata all'autovalore secondario λ_2 .

Il metodo delle potenze modificato con estrapolazione di Aitken può essere implementato algoritmicamente come segue:

```
function x = AitkenPower(tol, v)
x^(0) = v
per k = 1, 2, ...
    x^(k) = Gx^(k-1)
    x^(k) = x^(k) / ||x^(k)||_1
    periodicamente x^(k) = Aitken(x^(k-2), x^(k-1), x^(k))
        x^(k) = x^(k) / ||x^(k)||_1
    se ||x^(k) - x^(k-1)||_1 < tol Stop
end
```

```
function x = Aitken(x^(k-2), x^(k-1), x^(k))
g_i = (x_i^(k-1) - x_i^(k-2))^2
h_i = x_i^(k) - 2x_i^(k-1) + x_i^(k-2)
x_i = x_i^(k) - g_i/h_i
end
```

Osservazione 3.10. *L'introduzione della normalizzazione nel primo algoritmo dopo l'esecuzione della funzione "Aitken", pur non influenzando la direzione dell'autovettore sul piano teorico, risulta fondamentale per la stabilità numerica e garantisce che l'iterata rimanga un vettore di probabilità valido durante l'intero processo di calcolo.*

Costo computazionale: il costo aggiuntivo prodotto dall'algoritmo del blocco di Aitken a quello di una iterazione del metodo delle potenze è $O(n)$ (ricordiamo che n è il numero di pagine Web). Dal momento che l'estrapolazione viene eseguita solo periodicamente, possiamo dire che il costo aggiuntivo dell'estrapolazione di Aitken è minimo.

3.3 Estrapolazione quadratica

Assumiamo, ora, che la Google matrix G abbia solo tre autovettori e che l'iterata $x^{(k-3)}$ possa essere espressa come loro combinazione lineare.

Queste assunzioni ci permetteranno di risolvere per l'autovettore principale in forma chiusa utilizzando le iterate successive $x^{(k-3)}, x^{(k-2)}, x^{(k-1)}, x^{(k)}$.

Come nel caso di Aitken, sotto queste ipotesi $x^{(k-3)}$ può essere approssimata come combinazione lineare dei soli primi tre autovettori, pertanto, il vettore u_1 calcolato tramite questo algoritmo sarà solo una stima del vero autovettore dominante. Mostriamo sperimentalmente che tale stima si avvicinerà all'effettivo valore di u_1 al crescere di k .

3.3.1 Formulazione

Iniziamo la formulazione dell'estrapolazione quadratica approssimando $x^{(k-3)}$. Stando a quanto detto sopra, possiamo scrivere:

$$x^{(k-3)} = u_1 + \alpha_2 u_2 + \alpha_3 u_3 \quad (3.15)$$

Dopodichè, definiamo le iterate successive:

$$x^{(k-2)} = Gx^{(k-3)} \quad (3.16)$$

$$x^{(k-1)} = Gx^{(k-2)} \quad (3.17)$$

$$x^{(k)} = Gx^{(k-1)} \quad (3.18)$$

Poichè assumiamo che G abbia solo tre autovettori, il polinomio caratteristico $p_G(\lambda)$ è dato da:

$$p_G(\lambda) = \gamma_0 + \gamma_1 \lambda + \gamma_2 \lambda^2 + \gamma_3 \lambda^3. \quad (3.19)$$

Come sappiamo, l'autovalore dominante della Google matrix è $\lambda_1 = 1$. Gli autovalori di G , inoltre, sono gli zeri del polinomio caratteristico $p_G(\lambda)$, pertanto

$$p_G(1) = \gamma_0 + \gamma_1 + \gamma_2 + \gamma_3 = 0. \quad (3.20)$$

Per il Teorema di Cayley-Hamilton (1.12), si ha $p_G(G) = 0$, da cui, per ogni $z \in \mathbb{R}^n$,

$$p_G(G)z = [\gamma_0 I + \gamma_1 G + \gamma_2 G^2 + \gamma_3 G^3]z = 0. \quad (3.21)$$

Ponendo $z = x^{(k-3)}$,

$$[\gamma_0 I + \gamma_1 G + \gamma_2 G^2 + \gamma_3 G^3]x^{(k-3)} = 0. \quad (3.22)$$

Dalle equazioni (3.16), (3.17) e (3.18), sostituendo,

$$\gamma_0 x^{(k-3)} + \gamma_1 x^{(k-2)} + \gamma_2 x^{(k-1)} + \gamma_3 x^{(k)} = 0. \quad (3.23)$$

Da (3.20) segue

$$x^{(k-3)}(-\gamma_1 - \gamma_2 - \gamma_3) + \gamma_1 x^{(k-2)} + \gamma_2 x^{(k-1)} + \gamma_3 x^{(k)} = 0. \quad (3.24)$$

Riscrivendo otteniamo:

$$(x^{(k-2)} - x^{(k-3)})\gamma_1 + (x^{(k-1)} - x^{(k-3)})\gamma_2 + (x^{(k)} - x^{(k-3)})\gamma_3 = 0. \quad (3.25)$$

Introduciamo le seguenti definizioni:

$$y^{(k-2)} := x^{(k-2)} - x^{(k-3)}, \quad (3.26)$$

$$y^{(k-1)} := x^{(k-1)} - x^{(k-3)}, \quad (3.27)$$

$$y^{(k)} := x^{(k)} - x^{(k-3)}. \quad (3.28)$$

Possiamo, ora riscrivere l'equazione (3.25) in notazione matriciale:

$$\begin{pmatrix} y^{(k-2)} & y^{(k-1)} & y^{(k)} \end{pmatrix} \boldsymbol{\gamma} = 0 \quad (3.29)$$

dove $\boldsymbol{\gamma} = (\gamma_1 \ \gamma_2 \ \gamma_3)^T$.

Escludendo la soluzione banale $\boldsymbol{\gamma} = 0$, dal momento che vincolare un singolo coefficiente del polinomio non influenza gli zeri (ma fissa una scelta di scala per $\boldsymbol{\gamma}$), imponiamo

$$\gamma_3 = 1. \quad (3.30)$$

Da ciò segue:

$$\begin{pmatrix} y^{(k-2)} & y^{(k-1)} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -y^{(k)}. \quad (3.31)$$

Si tratta di un sistema sovradeterminato che può essere risolto come problema ai minimi quadrati

$$\min_{\gamma_1, \gamma_2 \in \mathbb{R}} \left\| \begin{pmatrix} y^{(k-2)} & y^{(k-1)} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} + y^{(k)} \right\|,$$

la cui soluzione si ottiene risolvendo il sistema triangolare superiore

$$R \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -Q^T y^{(k)}, \quad (3.32)$$

dove le matrici Q e R costituiscono la fattorizzazione QR di $Y = \begin{pmatrix} y^{(k-2)} & y^{(k-1)} \end{pmatrix}$. Le equazioni (3.20), (3.30) e (3.32) determinano completamente i coefficienti del polinomio caratteristico $p_G(\lambda)$.

Dividendo $p_G(\lambda)$ per $\lambda - 1$ otteniamo il polinomio $q_G(\lambda)$ avente come radici i secondi due autovalori di G : λ_2 e λ_3 .

$$q_G(\lambda) = \frac{\gamma_0 + \gamma_1\lambda + \gamma_2\lambda^2 + \gamma_3\lambda^3}{\lambda - 1} =: \beta_0 + \beta_1\lambda + \beta_2\lambda^2. \quad (3.33)$$

Effettuando la divisione tra polinomi otteniamo:

$$\beta_0 = \gamma_1 + \gamma_2 + \gamma_3 \quad (3.34)$$

$$\beta_1 = \gamma_2 + \gamma_3 \quad (3.35)$$

$$\beta_2 = \gamma_3 \quad (3.36)$$

Grazie al Teorema di Cayley-Hamilton (Teorema 1.12), si ha:

$$0 = p_G(G) = (G - I)q_G(G), \quad (3.37)$$

dunque, per ogni $z \in \mathbb{R}^n$,

$$(G - I)(q_G(G)z) = 0, \quad (3.38)$$

ossia,

$$q_G(G)z \in \text{Ker}(G - I). \quad (3.39)$$

Poichè $\text{Ker}(G - I)$ è proprio l'autospazio dominante, segue $q_G(G)z = u_1$.²

Ponendo $z = x^{(k-2)}$,

$$u_1 = q_G(G)x^{(k-2)} = [\beta_0 I + \beta_1 G + \beta_2 G^2]x^{(k-2)}, \quad (3.40)$$

dalle equazioni (3.16), (3.17) e (3.18), otteniamo una soluzione in forma chiusa per u_1 :

$$u_1 = \beta_0 x^{(k-2)} + \beta_1 x^{(k-1)} + \beta_2 x^{(k)} \quad (3.41)$$

Come detto in precedenza, quest'ultima equazione fornisce solo un'approssimazione di u_1 in quanto tale formulazione segue dall'ipotesi che G abbia solo tre autovettori.

²A meno di multipli scalari.

3.3.2 Estrapolazione quadratica nel metodo delle potenze

Anche in questo caso, l'estrapolazione quadratica può essere eseguita periodicamente all'interno del metodo delle potenze per accelerare la convergenza all'autovettore principale.

Possiamo riassumere il funzionamento dell'estrapolazione quadratica come segue. Ad ogni iterazione in cui viene eseguita l'estrapolazione, a partire dall'iterata corrente e dalle tre precedenti, vengono calcolati i coefficienti di un polinomio di secondo grado; tale polinomio è costruito in modo da attenuare i contributi associati ai principali autovalori subdominanti λ_2 e λ_3 . Sfruttando la definizione delle iterate precedenti, l'iterata estrapolata $x^{(k)}$ può essere interpretata come il risultato dell'applicazione di tale polinomio a $x^{(k-2)}$. Per costruzione, le componenti di errore lungo le direzioni dei due autovettori subdominanti risultano fortemente ridotte. Si ottiene così una stima dell'autovettore dominante depurata dai contributi più lenti e perturbata soltanto da termini residui che convergono a zero molto più rapidamente.

Osservazione 3.11. *Mentre l'estrapolazione di Aitken agisce riducendo la componente dell'errore lungo la direzione dell'autovettore associato a λ_2 , l'estrapolazione quadratica, coinvolge simultaneamente gli autospazi relativi a λ_2 e λ_3 . Questo approccio permette di eliminare più componenti di disturbo, offrendo una stima dell'autovettore dominante significativamente più precisa.*

Il metodo delle potenze modificato con estrapolazione quadratica può essere implementato come segue:

```
function  $x = \text{QuadraticPower}(tol, v)$ 
```

```
 $x^{(0)} = v$ 
```

```
per  $k = 1, 2, \dots$ 
```

```
     $\hat{x}^{(k)} = Gx^{(k-1)}$ 
```

```
     $x^{(k)} = \hat{x}^{(k)} / \|\hat{x}^{(k)}\|_1$ 
```

```
    periodicamente  $\hat{x}^{(k)} = \text{Quadratic}(x^{(k-3)}, x^{(k-2)}, x^{(k-1)}, x^{(k)})$ 
```

```
         $x^{(k)} = \hat{x}^{(k)} / \|\hat{x}^{(k)}\|_1$ 
```

```
    se  $\|x^{(k)} - x^{(k-1)}\|_1 < tol$  Stop
```

```
end
```

```

function  $x = \text{Quadratic}(x^{(k-3)}, x^{(k-2)}, x^{(k-1)}, x^{(k)})$ 
per  $j = k - 2, k - 1, k$ 
     $y^{(j)} = x^{(j)} - x^{(k-3)}$ 
end
 $Y = \begin{pmatrix} y^{(k-2)} & y^{(k-1)} \end{pmatrix}$ 
 $\gamma_3 = 1$ 
 $[Q.R] = \text{QR\_householder}(Y)$ 
 $R \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -Q^T y^{(k)}$ 
 $\gamma_0 = -(\gamma_1 + \gamma_2 + \gamma_3)$ 
 $\beta_0 = \gamma_1 + \gamma_2 + \gamma_3$ 
 $\beta_1 = \gamma_2 + \gamma_3$ 
 $\beta_2 = \gamma_3$ 
 $x = \beta_0 x^{(k-2)} + \beta_1 x^{(k-1)} + \beta_2 x^{(k)}$ 
end

```

Osservazione 3.12. *Come nel caso dell'estrapolazione di Aitken, l'introduzione della normalizzazione nel primo algoritmo dopo l'esecuzione della funzione "Quadratic", sebbene non influenzi la direzione dell'autovettore sul piano teorico, risulta cruciale per la stabilità numerica e per garantire che l'iterata rimanga un vettore di probabilità valido durante l'intero processo di calcolo.*

Per calcolare $\begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix}$ si eseguono i seguenti passaggi:

1. Mediante le trasformazioni di Householder³ applicate a $(Y \mid -y^{(k)})$ si ottiene la matrice $(R \mid -Q^T y^{(k)})$ dove Q e R costituiscono la fattorizzazione QR di Y.⁴ Poichè $Y \in \mathbb{R}^{n \times 2}$, la fattorizzazione richiede soltanto due trasformazioni di Householder.

2. Si risolve il sistema triangolare superiore $R \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = -Q^T y^{(k)}$ per $\begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix}$ utilizzando le sostituzioni all'indietro.

³Questa scelta è stata effettuata per ragioni di stabilità numerica.

⁴Q non viene calcolata esplicitamente durante la procedura: è applicata implicitamente come prodotto di trasformazioni di Householder.

Costo computazionale: il costo aggiuntivo prodotto dall'algoritmo dell'estrapolazione quadratica rispetto a quello di una iterazione del metodo delle potenze deriva principalmente dalla risoluzione del problema ai minimi quadrati di γ_1 e γ_2 . La fattorizzazione QR calcolata mediante trasformazioni di Householder, richiede, per una matrice di dimensioni generiche $n_1 \times n_2$ un costo computazionale uguale a $O(n_1 n_2^2)$. Nel nostro caso, la matrice Y è di dimensione $n \times 2$, pertanto il costo computazionale richiesto è uguale a $O(n)$. Le altre operazioni all'interno dell'algoritmo hanno un costo computazionale di $O(1)$ o $O(n)$.

Dal momento che l'estrapolazione viene eseguita solo periodicamente, possiamo dire che il costo aggiuntivo dell'estrapolazione quadratica è minimo.

Capitolo 4

Risultati sperimentali

I tre datasets utilizzati per valutare la convergenza dei metodi analizzati sono tratti dallo Stanford Network Analysis Project (SNAP).

Essi sono: *web-NotreDame*, che consiste nel grafo diretto del dominio Web dell'università di Notre Dame collezionato nel 1999, *web-Google*, ossia un grafo diretto del Web rilasciato da Google nel 2002 come parte del Google Programming Contest e *soc-LiveJournal1*, ovvero il grafo diretto delle amicizie tra i membri del social network LiveJournal.

4.1 Matrice *web-NotreDame*

Le proprietà del grafo diretto di *web-NotreDame* sono le seguenti:

Nodi	Links	Densità
325729	≈ 1.5 milioni	1.4×10^{-5}

Osservazione 4.1. *Nonostante le loro dimensioni, tutte le matrici considerate sono estremamente sparse: la densità di ciascuna di esse è nell'ordine di grandezza di 10^{-5} o 10^{-6} .*

In Figura 4.1 riportiamo la **struttura Spy** della matrice, ovvero una rappresentazione della densità della matrice nella quale le aree colorate e quelle bianche descrivono, rispettivamente, la disposizione degli elementi non zero e degli zeri della matrice stessa.

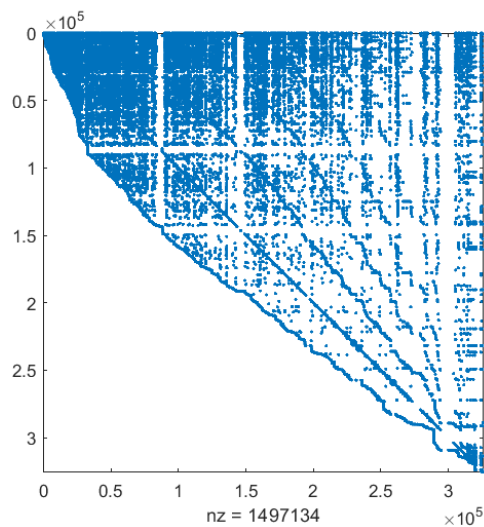


Figura 4.1: Visualizzazione degli elementi non zero della matrice *web-NotreDame*

4.1.1 Metodo delle potenze

Analizziamo, innanzitutto, la convergenza del metodo delle potenze applicato alla Google matrix G costruita a partire dalla matrice *web-NotreDame*. Come abbiamo visto, la matrice G non viene mai calcolata esplicitamente durante l'esecuzione dell'algoritmo.

Poichè la velocità di convergenza del metodo è influenzata dalla scelta di $\alpha \in (0, 1)$, esaminiamo l'andamento di $\|x^{(k)} - x^{(k-1)}\|_1$ per $k \geq 1$ al variare di tale parametro. Indichiamo con **residuo** L_1 la norma 1 della differenza tra due iterate successive[4].

All'interno di questo elaborato, a meno che non venga specificato diversamente, la tolleranza che definisce il criterio di arresto dell'algoritmo è fissata a 10^{-8} . Inoltre, in caso il metodo non converga entro le prime 1000 iterazioni, esso viene forzatamente arrestato al termine della millesima iterazione.

I valori che consideriamo sono i seguenti:

- $\alpha = 0.5$ (conferisce una scarsa rilevanza all'effettiva struttura del Web)
- $\alpha = 0.7$
- $\alpha = 0.85$ (scelta standard)
- $\alpha = 0.95$
- $\alpha = 0.99$ (massima importanza conferita alla struttura del Web)

La convergenza del metodo delle potenze è illustrata in Figura 4.2, al variare del parametro α :

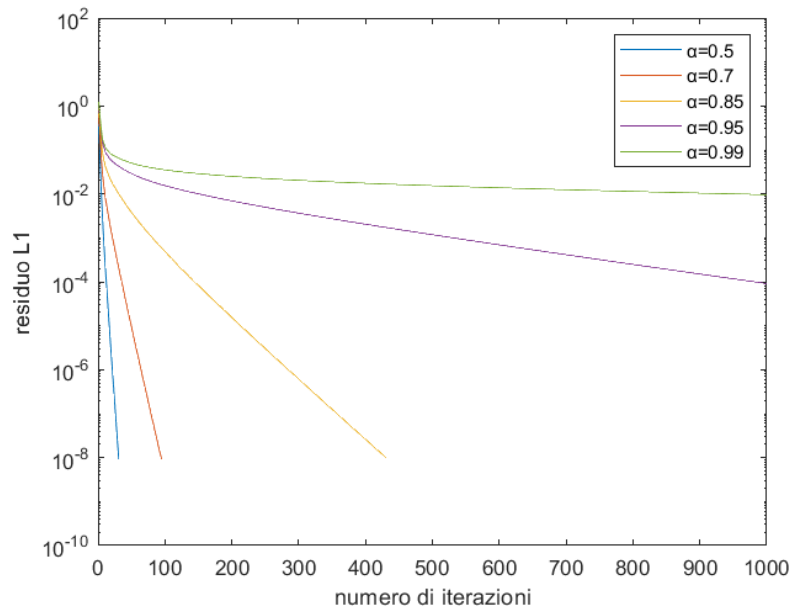


Figura 4.2: *Convergenza del metodo delle potenze al variare di α*

Come ci aspettavamo, la velocità di convergenza cala al crescere del valore del parametro. In particolare, la convergenza del metodo per $\alpha = 0.5$ e $\alpha = 0.7$ è piuttosto rapida ed avviene, rispettivamente, in 31 e 95 iterazioni.

Per $\alpha = 0.85$, il metodo impiega 431 iterazioni affinché il valore del residuo scenda al di sotto della tolleranza fissata, mentre per $\alpha = 0.95$ e $\alpha = 0.99$ quest'ultima non viene raggiunta entro le prime mille iterazioni.

4.1.2 Estrapolazione di Aitken

In questa sezione, studiamo gli effetti dell'estrapolazione di Aitken sulla velocità di convergenza del metodo delle potenze applicato alla matrice *web-NotreDame* al variare di α . Per $\alpha = 0.5$ e $\alpha = 0.7$, applichiamo l'estrapolazione di Aitken una volta in fasi diverse della procedura.

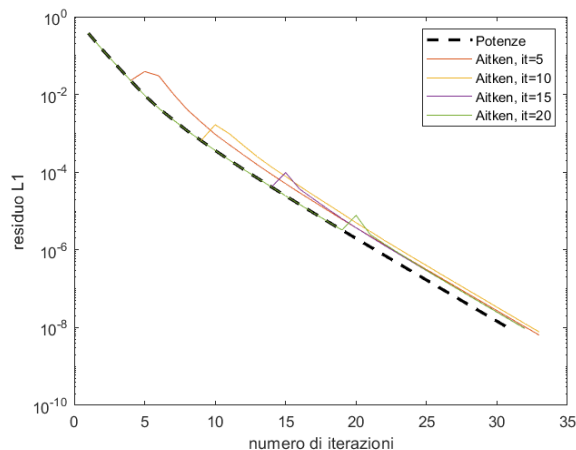


Figura 4.3: *Confronto per $\alpha = 0.5$*

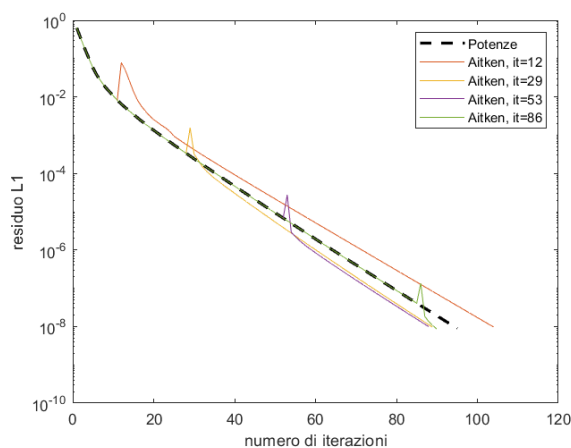


Figura 4.4: *Confronto per $\alpha = 0.7$*

Nel primo caso, a prescindere dal momento in cui si esegue l'estrapolazione, essa rallenta la convergenza; questo risultato è riconducibile al fatto che il metodo delle potenze, per questo valore del parametro, risulta di per sè molto efficiente.

Nel secondo caso, invece, possiamo notare che applicando l'estrapolazione anzitempo, ad esempio alla dodicesima iterazione, la convergenza della variante modificata richiede 104 iterazioni a fronte delle 95 richieste dal metodo delle potenze classico. Non è opportuno applicare l'estrapolazione di Aitken nelle prime fasi dell'iterazione poiché in tale regime la direzione asintotica della sequenza non è ancora stabilizzata, correndo il rischio di ottenere un'accelerazione meno efficace. Per quanto riguarda gli altri casi riportati in Figura 4.4, l'estrapolazione produce una riduzione del numero di iterazioni necessarie alla convergenza uguale al 5.3-7.4%.

Per la scelta $\alpha = 0.85$, riportiamo in Figura 4.5 il confronto tra il metodo standard e quello modificato mediante una singola estrapolazione:

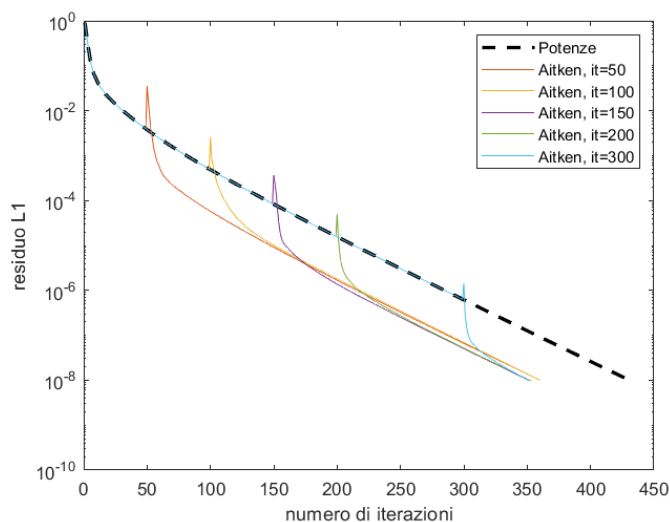


Figura 4.5: *Confronto per $\alpha = 0.85$*

Applicando l'estrapolazione all'iterazione 150, 200 o 300 si ottiene un risparmio iterativo di circa il 18.3%, in quanto la convergenza avviene in 352 iterazioni a fronte delle 431 richieste dal metodo standard. Estrapolando alla 50^a o alla 100^a iterazione, i benefici ottenuti sono leggermente inferiori.

In seguito a questo risultato, ci domandiamo se l'applicazione multipla dell'estrapolazione possa giovare ulteriormente alla convergenza.

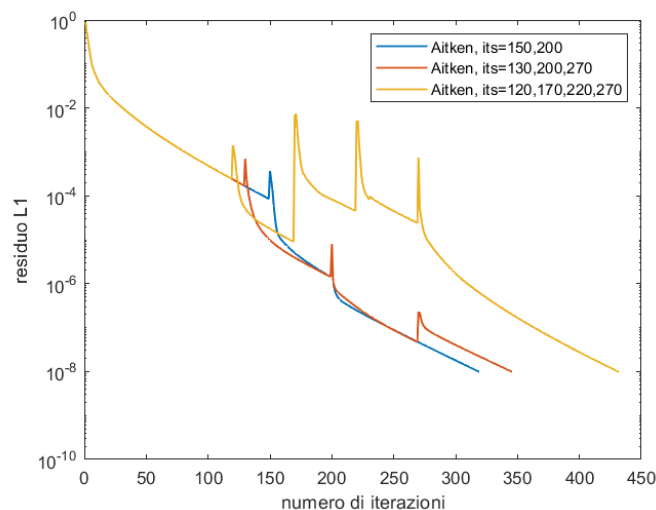


Figura 4.6: *Confronto del metodo modificato con due, tre e quattro estrapolazioni per $\alpha = 0.85$*

In Figura 4.6 è riportato il confronto tra le varianti del metodo delle potenze modificate mediante due, tre e quattro applicazioni dell'estrapolazione di Aitken.

Estrapolando due volte alle iterazioni 150,200, si ottiene il risultato migliore: il residuo assume un valore inferiore alla tolleranza fissata in 319 iterazioni, assicurando un risparmio iterativo intorno al 26% rispetto al metodo delle potenze standard. Un simile risparmio in termini di iterazioni comporta benefici anche relativamente al tempo di esecuzione della procedura. Come si vedrà in seguito, tale aspetto risulta particolarmente importante quando si considerano quantità di dati più elevate.

Aumentando il numero di estrapolazioni, notiamo che la convergenza del metodo rallenta. Tale comportamento è attribuibile ai picchi che si manifestano in corrispondenza di ciascuna estrapolazione, i quali sono riconducibili a una non ancora accurata approssimazione del secondo autovettore dominante. I risultati ottenuti indicano che un'applicazione periodica dell'estrapolazione di Aitken non è consigliabile, infatti, supponendo di applicarla ogni 20 iterazioni, la decrescita del residuo viene perturbata come segue:

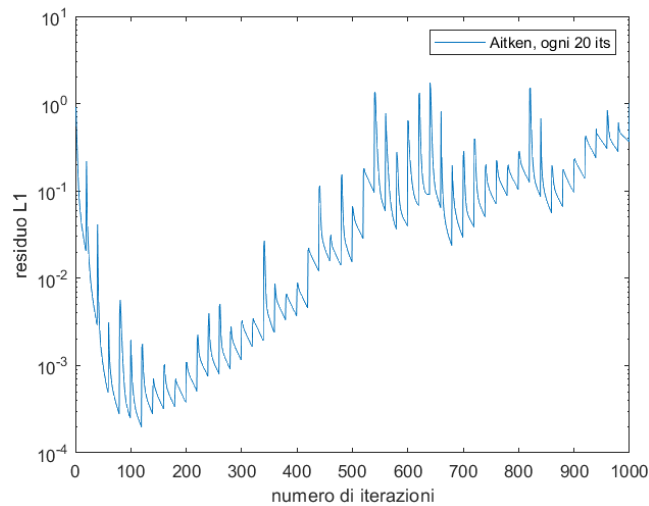


Figura 4.7: *Comportamento del metodo delle potenze modificato con estrapolazione di Aitken periodica per $\alpha = 0.85$*

In questa situazione, la convergenza risulta fortemente penalizzata diventando enormemente più lenta rispetto a quella del metodo delle potenze standard; in particolare, il residuo, per quanto riguarda le prime 1000 iterazioni, mostra addirittura un andamento crescente.

Nei casi $\alpha = 0.95$ e $\alpha = 0.99$, come abbiamo visto, il residuo del metodo delle potenze classico è ben lontano dal raggiungimento della tolleranza fissata al termine di mille iterazioni. Per la scelta $\alpha = 0.95$, applicando l'extrapolazione di Aitken una singola volta in diversi momenti della procedura e confrontando l'andamento dei residui dei metodi modificati con quello del metodo delle potenze standard, si ha:

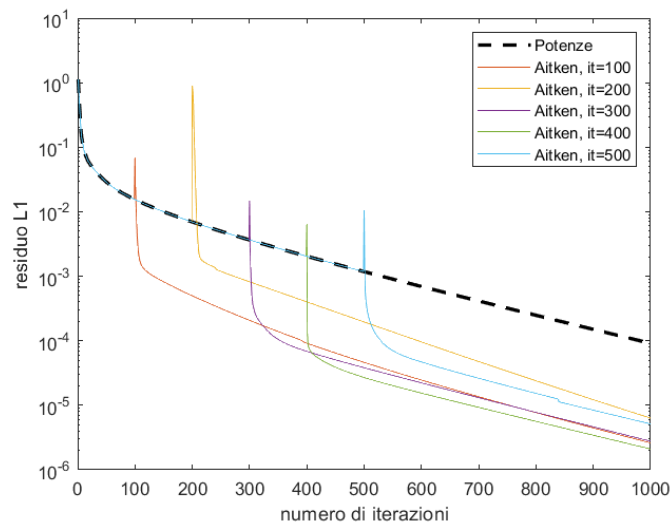


Figura 4.8: *Confronto per $\alpha = 0.95$*

In particolare, il valore del residuo del metodo delle potenze standard dopo le prime mille iterazioni è 9.13×10^{-5} , mentre quello del metodo modificato, ottenuto applicando l'extrapolazione alla 400^a iterazione, è uguale a 2.10×10^{-6} , risultando quindi inferiore di poco meno di due ordini di grandezza rispetto al precedente.

Per quanto riguarda il caso $\alpha = 0.99$, invece, si ha:

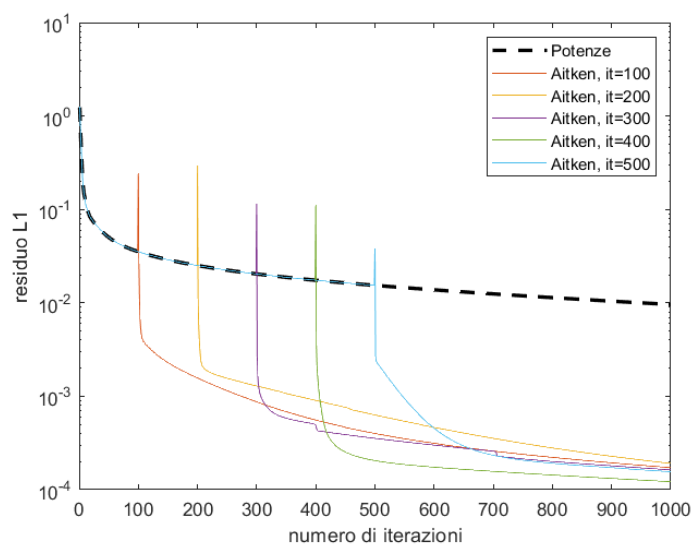


Figura 4.9: confronto per $\alpha = 0.99$

Il residuo del metodo delle potenze dopo mille iterazioni è uguale a 9.60×10^{-3} , quello del metodo con applicazione dell'extrapolazione di Aitken alla 400^a iterazione ha valore 1.22×10^{-4} , superando anche in questo caso di quasi due ordini di grandezza la precisione del metodo delle potenze standard.

Tentando anche per questi ultimi due valori di α applicazioni multiple dell'extrapolazione, si ha:

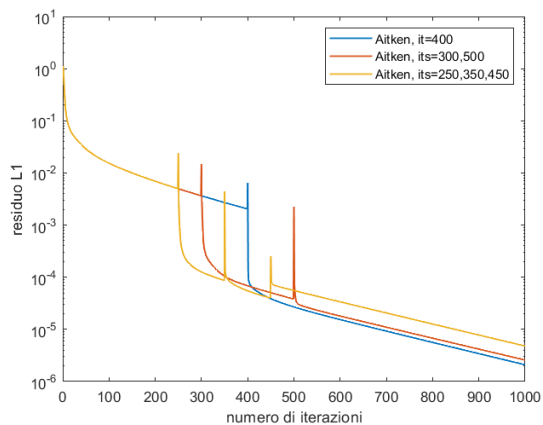


Figura 4.10: Confronto per $\alpha = 0.95$

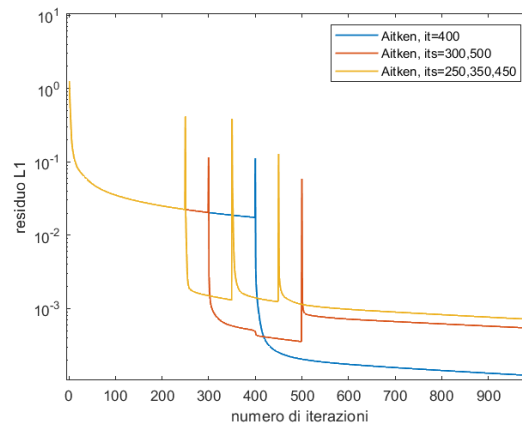


Figura 4.11: Confronto per $\alpha = 0.99$

In entrambi i casi, aumentare di poco il numero di estrapolazioni influisce negativamente sulla convergenza. Per $\alpha = 0.99$ tale peggioramento risulta ancora più evidente.

4.1.3 Estrapolazione quadratica

Procediamo esaminando l'effetto dell'estrapolazione quadratica sulla convergenza del metodo delle potenze applicato alla matrice *web-NotreDame*.

Per quanto riguarda i casi $\alpha = 0.5$ e $\alpha = 0.7$, confrontando la velocità di convergenza del metodo delle potenze standard con quella di alcune sue varianti — che prevedono una singola applicazione dell'estrapolazione in diverse fasi della procedura oppure un'applicazione periodica (rispettivamente ogni 5 e ogni 20 iterazioni) — si ottiene quanto segue:

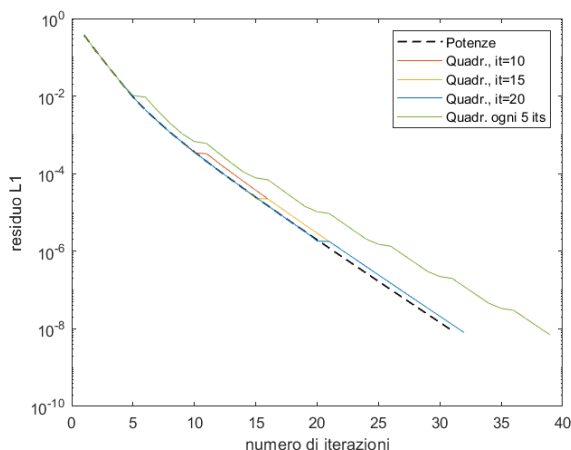


Figura 4.12: Confronto per $\alpha = 0.5$

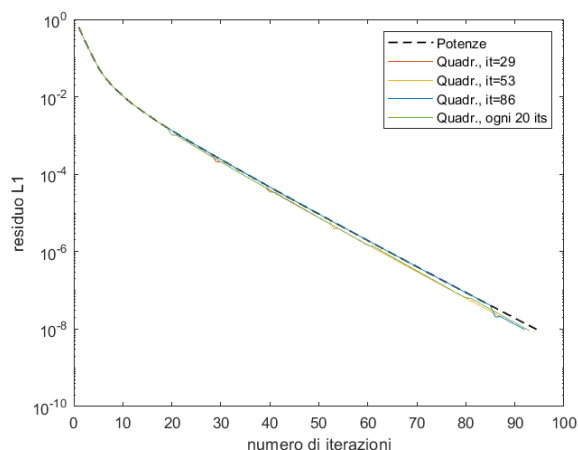


Figura 4.13: Confronto per $\alpha = 0.7$

Come nell'analogo caso riguardante l'estrapolazione di Aitken, l'estrapolazione quadratica non produce benefici osservabili in nessuno dei due casi. In particolare, come si evince dal grafico in Figura 4.12, per $\alpha = 0.5$ una sua applicazione periodica penalizza in modo piuttosto significativo la convergenza del metodo. Osservando la Figura 4.13, invece, notiamo che per $\alpha = 0.7$, il comportamento del metodo modificato mediante estrapolazione periodica risulta allineato a quello delle altre varianti.

Consideriamo il caso $\alpha = 0.85$. A differenza di quanto fatto per l'estrapolazione di Aitken, l'attenzione è fin da subito rivolta a casi che differiscono tra loro per il numero di applicazioni dell'estrapolazione all'interno della procedura iterativa.

Questa scelta è motivata dalla struttura teorica dell'estrapolazione quadratica, che coinvolge un numero maggiore di iterate rispetto all'estrapolazione di Aitken e consente di approssimare simultaneamente più componenti dell'errore, in particolare quelle associate ai due autovalori subdominanti che rallentano la convergenza. Di conseguenza, ci si

aspetta che l'applicazione dell'estrapolazione non generi picchi significativi nell'andamento del residuo non correndo il rischio di penalizzare un suo utilizzo ripetuto all'interno della procedura iterativa.

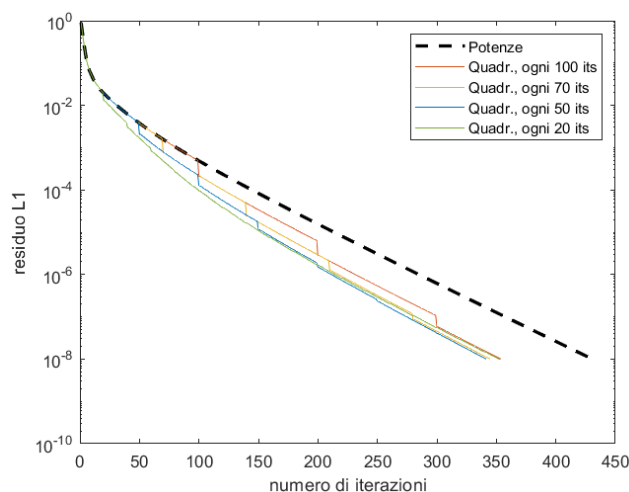


Figura 4.14: *Confronto per $\alpha = 0.85$*

Sebbene ciascuna frequenza di estrapolazione analizzata produca un beneficio apprezzabile, il risultato migliore si ha in corrispondenza dell'applicazione dell'estrapolazione ogni 50 iterazioni. In questo caso, il metodo converge in 342 iterazioni, impiegando il 20.6% di iterazioni in meno rispetto al metodo standard, che ne sfrutta 431.

Confrontando la convergenza del metodo delle potenze standard con quello modificato con l'estrapolazione quadratica applicata con periodo 20 e con due varianti che sfruttano l'estrapolazione di Aitken si ha:

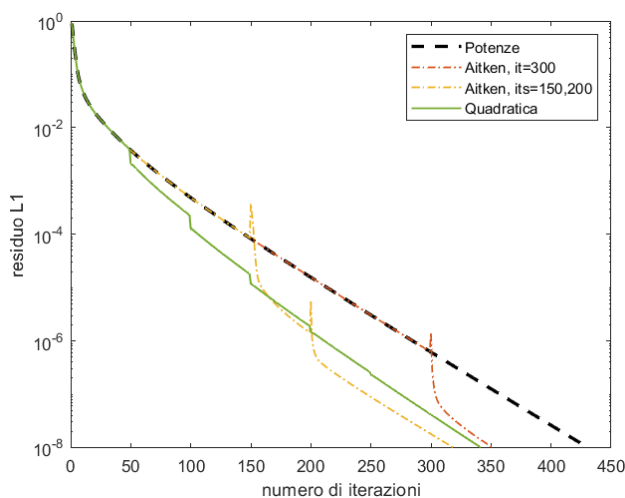


Figura 4.15: *Confronto dei tre metodi per $\alpha = 0.85$*

Il metodo modificato mediante estrapolazione quadratica converge più velocemente rispetto alla versione modificata con una singola applicazione dell'extrapolazione di Aitken all'iterazione 300, ma il beneficio maggiore resta comunque quello apportato dall'esecuzione di quest'ultima effettuata due volte.

Infine, esaminiamo i casi $\alpha = 0.95$ e $\alpha = 0.99$. Riportando in due figure separate i rispettivi confronti tra metodo standard e alcune varianti con estrapolazione quadratica otteniamo:

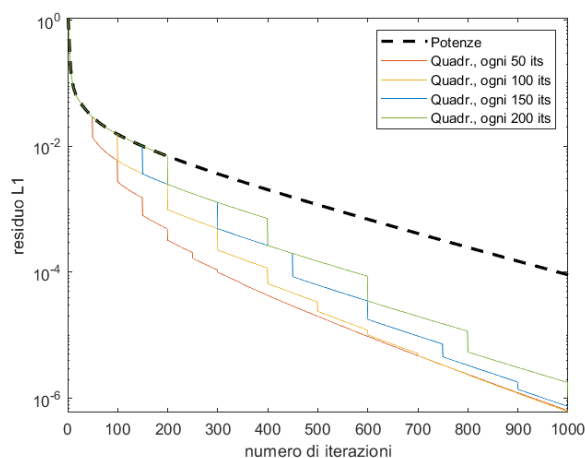


Figura 4.16: *Confronto per $\alpha = 0.95$*

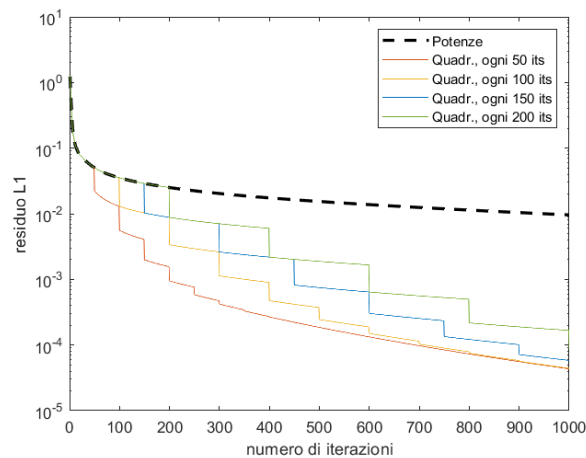


Figura 4.17: *Confronto per $\alpha = 0.99$*

In nessuno dei due casi l'introduzione dell'extrapolazione quadratica permette il raggiungimento della tolleranza fissata da parte del residuo entro le prime mille iterazioni. Inoltre, sia in Figura 4.16, che in Figura 4.17, osserviamo che le varianti del metodo che risultano più performanti sono quelle alle quali viene applicata l'extrapolazione quadratica con un periodo non troppo ampio, uguale a 50 o 100 iterazioni.

Concludiamo l'analisi confrontando, anche per questi ultimi due valori del parametro α , la convergenza del metodo delle potenze classico, di quello modificato mediante estrapolazione di Aitken eseguita alla 400^a iterazione¹ e della variante che prevede l'applicazione dell'extrapolazione quadratica ogni 100 iterazioni.

Per $\alpha = 0.95$ si ha:

¹Si tratta della versione che apporta i benefici maggiori tra quelle analizzate.

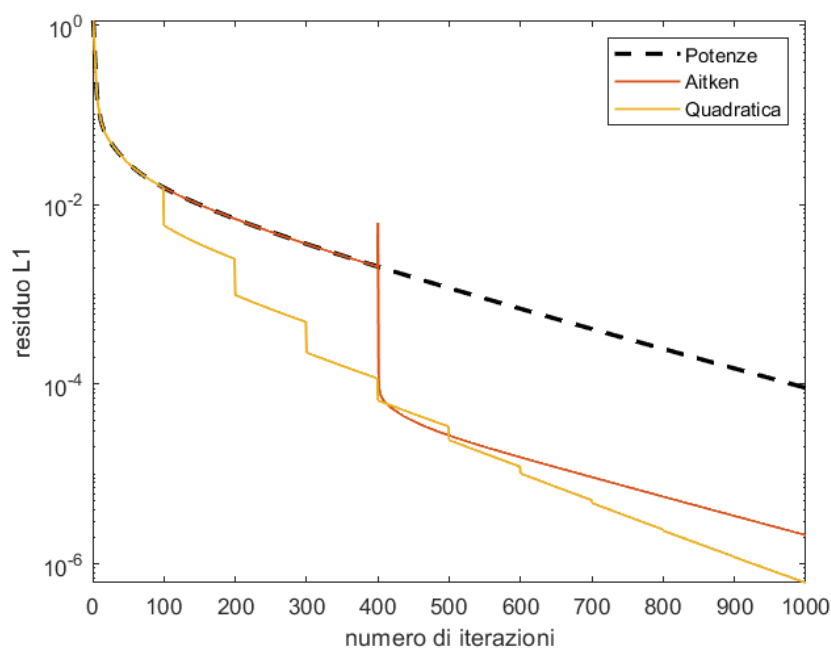


Figura 4.18: *Confronto dei tre metodi per $\alpha = 0.95$*

Riportiamo in una tabella i valori dei residui di ciascun metodo dopo mille iterazioni:

metodo	potenze	estr. Aitken	estr. quad.
residuo L_1	9.13×10^{-5}	2.10×10^{-6}	6.18×10^{-7}

Tra il valore del residuo del metodo delle potenze standard e quello ottenuto mediante l'extrapolazione di Aitken si osserva una differenza di circa 1.64 ordini di grandezza; in altre parole, il residuo del metodo modificato risulta circa 43 volte inferiore rispetto a quello del metodo classico.

Considerando invece la variante con extrapolazione quadratica, il residuo differisce da quello del metodo standard di circa 2.17 ordini di grandezza, risultando quindi circa 148 volte più piccolo dopo le prime mille iterazioni.

Questi risultati evidenziano come l'impiego delle tecniche di extrapolazione permetta di ottenere una riduzione significativamente più rapida del residuo, indicando quindi una velocità di convergenza notevolmente maggiore rispetto al metodo delle potenze standard.

Per $\alpha = 0.99$, invece, si ha il seguente grafico:

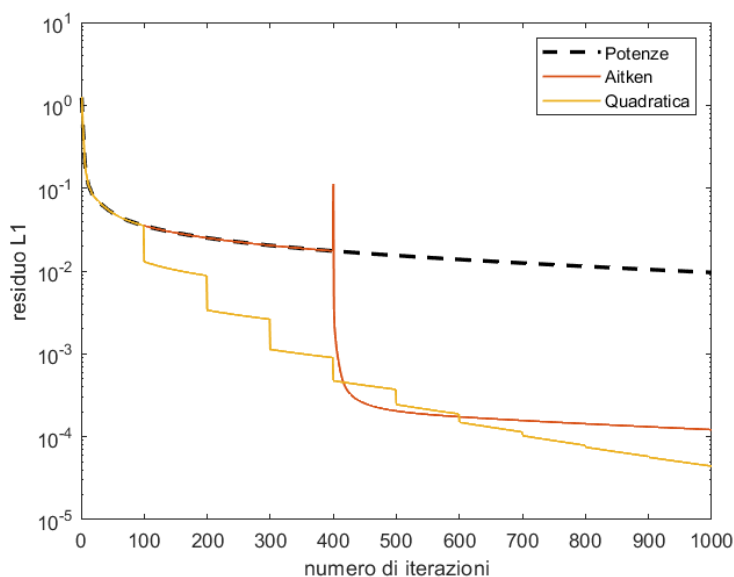


Figura 4.19: Confronto dei tre metodi per $\alpha = 0.99$

Riportando i valori in una tabella, si ha:

metodo	potenze	estr. Aitken	estr. quad.
residuo L_1	9.60×10^{-3}	1.22×10^{-4}	4.40×10^{-5}

Per questo valore di α , la differenza osservata tra il valore del residuo del metodo delle potenze standard e quello del metodo modificato mediante l'extrapolazione di Aitken è di circa 1.90 ordini di grandezza: il residuo del metodo modificato risulta circa 79 volte inferiore rispetto a quello del metodo classico.

Per quanto riguarda la variante con extrapolazione quadratica, il residuo differisce da quello del metodo standard di circa 2.34 ordini di grandezza, risultando circa 218 volte più piccolo dopo le prime mille iterazioni.

4.2 Matrice *web-Google*

Le proprietà del grafo diretto di *web-Google* sono le seguenti:

Nodi	Links	Densità
916428	≈ 5.1 milioni	6.1×10^{-6}

Al fine di analizzare la struttura di sparsità del dataset, anche in questo caso si riporta, in Figura 4.20, la rappresentazione Spy della matrice di Hyperlink associata. A causa delle dimensioni del grafo, la visualizzazione globale appare visivamente densa. Per

questo motivo, la stessa figura include anche ingrandimenti di sottomatrici di dimensione ridotta, che consentono di apprezzare chiaramente la reale natura sparsa della matrice.

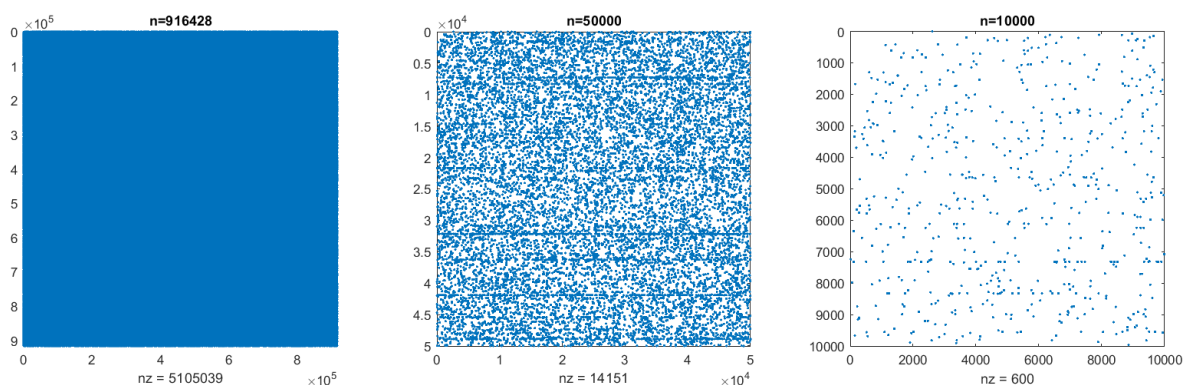


Figura 4.20: Da sinistra, *Spy* globale e *Spy* di due sottomatrici di dimensione rispettivamente 50000×50000 e 10000×10000 .

4.2.1 Metodo delle potenze

Analizziamo la convergenza del metodo delle potenze applicato alla Google matrix costruita a partire dalla matrice *web-Google*.

In Figura 4.21, mostriamo l'andamento del residuo L_1 del metodo delle potenze al variare di α :

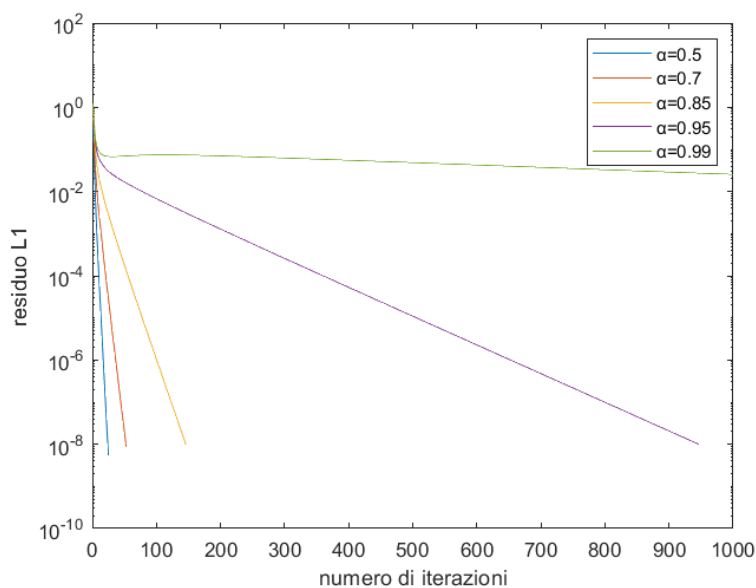


Figura 4.21: Convergenza del metodo delle potenze al variare di α applicato alla matrice *web-Google*

In particolare, riportando il numero di iterazioni necessarie alla convergenza in una tabella, si ha:

α	0.5	0.7	0.85	0.95	0.99
n° di iterazioni	25	53	146	947	>1000

Ai primi due valori di α corrisponde una rapida convergenza del metodo, mentre per $\alpha \geq 0.95$ esso rallenta drammaticamente.

Il compromesso migliore si ha per $\alpha = 0.85$, infatti esso concilia efficacemente due esigenze contrapposte: la necessità di una convergenza spedita e la volontà di valorizzare la struttura Hyperlink del Web.

4.2.2 Estrapolazione di Aitken

Esaminiamo l'effetto dell'estrapolazione di Aitken sulla convergenza del metodo. Per valori moderati del parametro α , quali $\alpha = 0.5$ e $\alpha = 0.7$, al fine di confrontare l'effetto dell'estrapolazione applicata una singola volta in momenti diversi della procedura, si riportano i seguenti grafici:

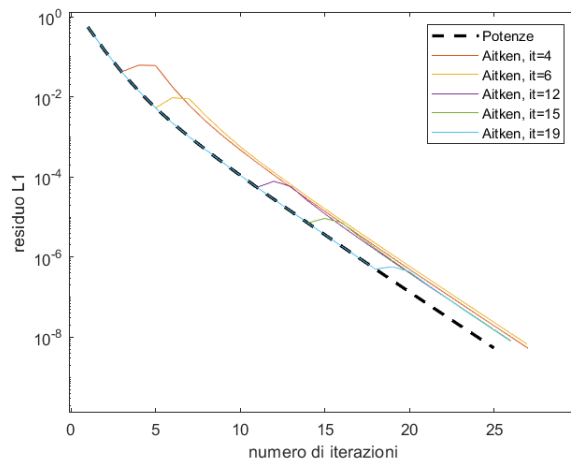


Figura 4.22: Confronto per $\alpha = 0.5$

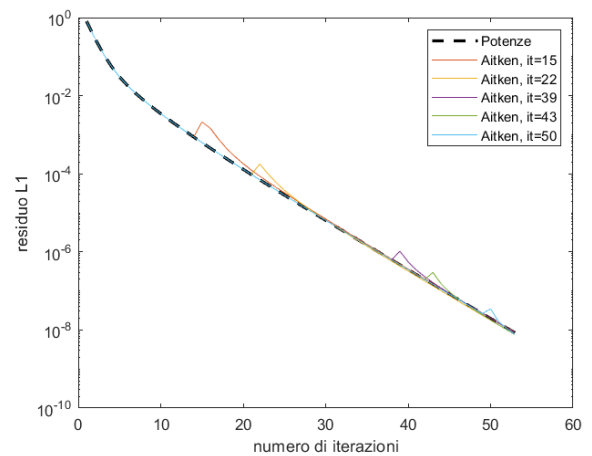


Figura 4.23: Confronto per $\alpha = 0.7$

In entrambi i casi considerati, l'estrapolazione non produce benefici apprezzabili in termini di convergenza; in particolare, per $\alpha = 0.5$, il metodo modificato richiede un numero di iterazioni leggermente superiore rispetto al metodo delle potenze standard. Per $\alpha = 0.7$, l'estrapolazione non influisce sul numero di iterazioni necessarie alla convergenza. Tali esiti sono giustificati dall'efficienza del metodo delle potenze per tali valori del parametro.

Nel caso $\alpha = 0.85$, invece, è possibile iniziare ad apprezzare l'utilità dell'estrapolazione:

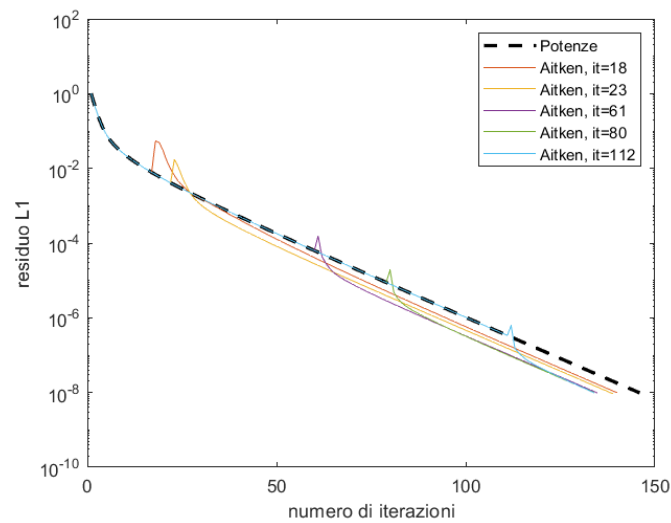


Figura 4.24: *Confronto per $\alpha = 0.85$*

Il metodo delle potenze converge in 146 iterazioni, mentre l'integrazione di una singola estrapolazione di Aitken consente una convergenza più rapida. In particolare, applicando l'estrapolazione alle iterazioni 61, 80 o 112, l'algoritmo si arresta dopo 134 iterazioni, con una riduzione del numero di iterazioni corrispondente circa all'8.2%.

Considerando il caso $\alpha = 0.95$, la riduzione di iterazioni necessarie alla convergenza risulta ancora più evidente:

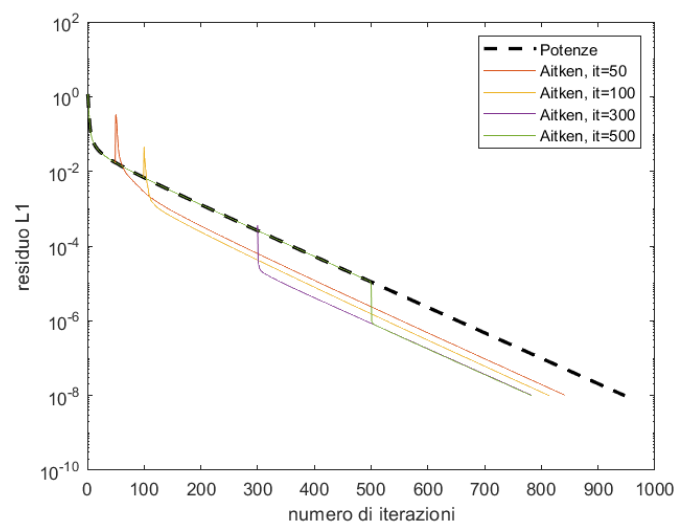


Figura 4.25: *Confronto con $\alpha = 0.95$*

Mentre il metodo delle potenze converge in 947 iterazioni, estrapolando alla trecentesima iterazione o alla cinquecentesima, la convergenza viene anticipata a 783 iterazioni con un risparmio iterativo di circa il 17.3%.

Nel caso $\alpha = 0.99$, assistiamo a un notevole incremento della velocità di convergenza:

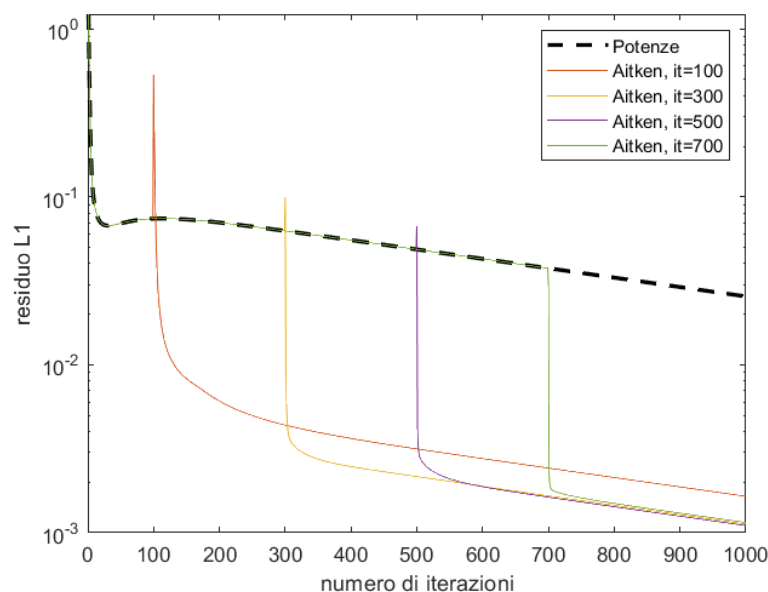


Figura 4.26: *Confronto per $\alpha = 0.99$*

Analogamente a quanto osservato in precedenza, l'applicazione dell'estrapolazione in una fase troppo iniziale della procedura, ad esempio alla centesima iterazione, risulta meno efficace.

In ognuno dei casi, sebbene dopo mille iterazioni il residuo risulti ancora lontano dal raggiungimento della tolleranza prefissata (10^{-8}), è chiaramente visibile l'accelerazione impressa al metodo dall'applicazione dell'estrapolazione di Aitken, in particolare quando essa viene eseguita in una fase sufficientemente avanzata dell'iterazione.

Ci domandiamo se l'applicazione ripetuta dell'estrapolazione possa ulteriormente migliorare la convergenza del metodo.

Rimanendo nel caso $\alpha = 0.99$, in Figura 4.27 viene confrontato l'andamento del residuo del metodo con una sola applicaione dell'estrapolazione di Aitken all'iterazione 300 e quello dei metodi modificati mediante due (alle iterazioni 300,400), tre (alle iterazioni 300,400,500) e quattro (alle iterazioni 300,400,500,600) applicazioni.

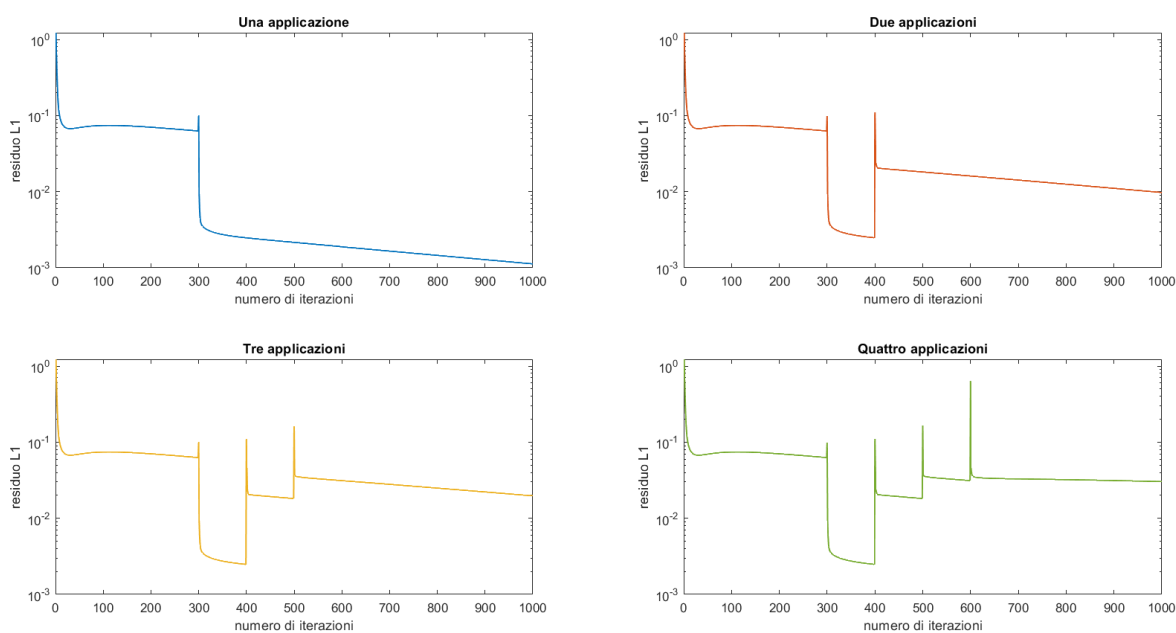


Figura 4.27: *Effetto di applicazioni multiple dell'extrapolazione di Aitken sull'andamento del residuo nel caso $\alpha = 0.99$*

All'aumentare del numero di estrapolazioni eseguite, segue una diminuzione della velocità di decrescita del residuo; più precisamente, riportando i valori all'interno di una tabella, dopo le prime mille iterazioni si ha:

n° di applicazioni	una	due	tre	quattro
residuo L_1	1.13×10^{-3}	9.76×10^{-3}	1.97×10^{-2}	3.03×10^{-2}

La causa di tale comportamento è riconducibile ai picchi che si manifestano in corrispondenza di ciascuna estrapolazione causati da un'approssimazione povera dell'autovettore subdominante.

4.2.3 Estrapolazione quadratica

In accordo con quanto emerso per l'extrapolazione di Aitken, nei casi $\alpha = 0.5$ e $\alpha = 0.7$ la rapidità della convergenza del metodo delle potenze non lascia margine a un miglioramento significativo mediante l'extrapolazione quadratica.

Per la scelta $\alpha = 0.85$, dal confronto della convergenza del metodo delle potenze standard e di alcune sue varianti ottenute eseguendo l'extrapolazione quadratica con frequenze diverse, si ottiene il seguente grafico:

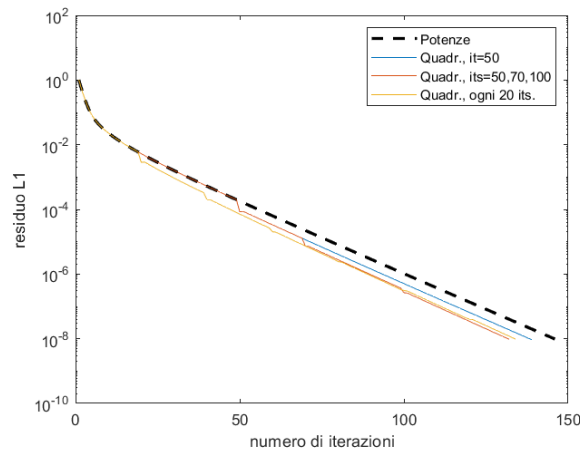


Figura 4.28: *Confronto per $\alpha = 0.85$*

In Figura 4.28 abbiamo confrontato la convergenza del metodo delle potenze con quella del metodo modificato applicando l'extrapolazione rispettivamente una, tre volte e periodicamente, ogni 20 iterazioni.

Rispetto a quanto osservato nel caso dell'extrapolazione di Aitken, in questo contesto il metodo che prevede tre applicazioni dell'extrapolazione quadratica risulta il più efficace in termini di velocità di convergenza. Esso converge in 132 iterazioni, contro le 146 richieste dal metodo delle potenze standard, con una riduzione del numero di iterazioni uguale a circa il 9.6%. Tale miglioramento risulta leggermente superiore a quello ottenuto mediante l'extrapolazione di Aitken, che si attestava intorno all'8.2%.

Consideriamo il caso $\alpha = 0.95$.

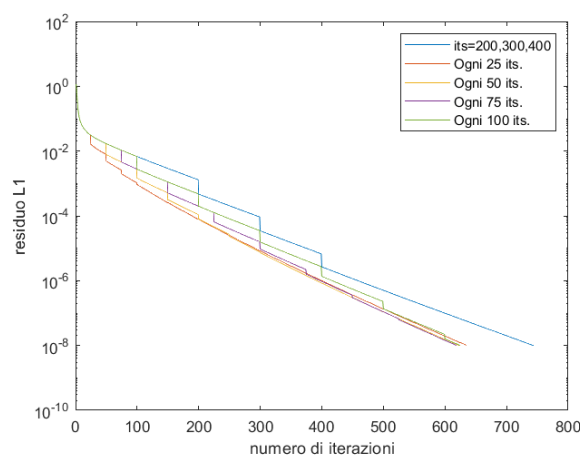


Figura 4.29: *Confronto di alcune varianti del metodo delle potenze modificato con extrapolazione quadratica per $\alpha = 0.95$*

La Figura 4.29 mostra che i metodi che prevedono un'applicazione periodica dell'estrapolazione quadratica convergono più rapidamente rispetto alla variante che la applica soltanto tre volte.

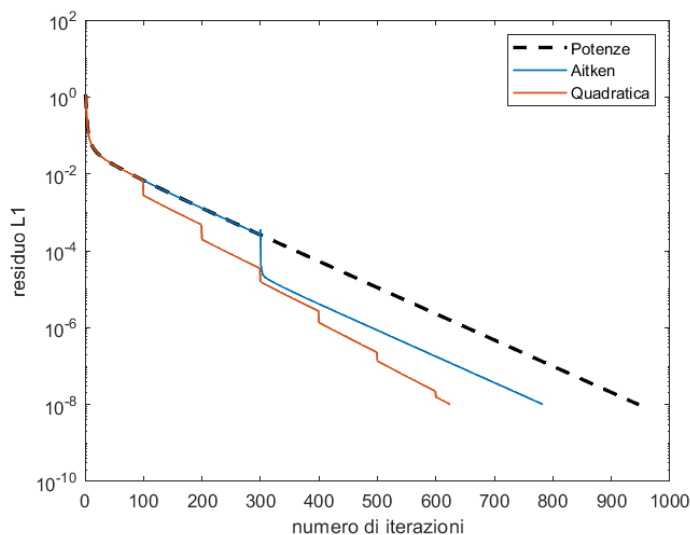


Figura 4.30: Confronto dei tre metodi nel caso $\alpha = 0.95$

In Figura 4.30, vengono confrontati gli andamenti del residuo L_1 del metodo delle potenze standard e di due sue varianti modificate mediante, rispettivamente, una applicazione dell'estrapolazione di Aitken (alla 300^a iterazione) e l'applicazione periodica dell'estrapolazione quadratica (ogni 100 iterazioni).

La convergenza del metodo con estrapolazione quadratica viene raggiunta in 625 iterazioni, apportando una riduzione di circa il 34% rispetto al metodo standard, a fronte di quella del 17.3% ottenuta mediante estrapolazione di Aitken. Possiamo dunque concludere che per $\alpha = 0.95$, l'estrapolazione quadratica fornisce un risparmio iterativo sensibilmente superiore rispetto all'estrapolazione di Aitken, risultando in questo caso circa doppio.

Esaminiamo, infine, il caso $\alpha = 0.99$. Analogamente a quanto osservato per il metodo delle potenze modificato mediante l'estrapolazione di Aitken, per questo valore del parametro non è possibile raggiungere la tolleranza prefissata entro il limite di mille iterazioni. L'analisi si concentra pertanto sul confronto dell'andamento dei residui e sulla loro riduzione rispetto al metodo delle potenze standard.

Alla luce dei risultati ottenuti per $\alpha = 0.95$, che evidenziano l'efficacia dell'applicazione periodica dell'estrapolazione quadratica, consideriamo direttamente questa formu-

lazione: nella Figura 4.31 analizziamo la convergenza del metodo delle potenze al quale applichiamo l'estrapolazione quadratica con periodi 25,50,75,100.

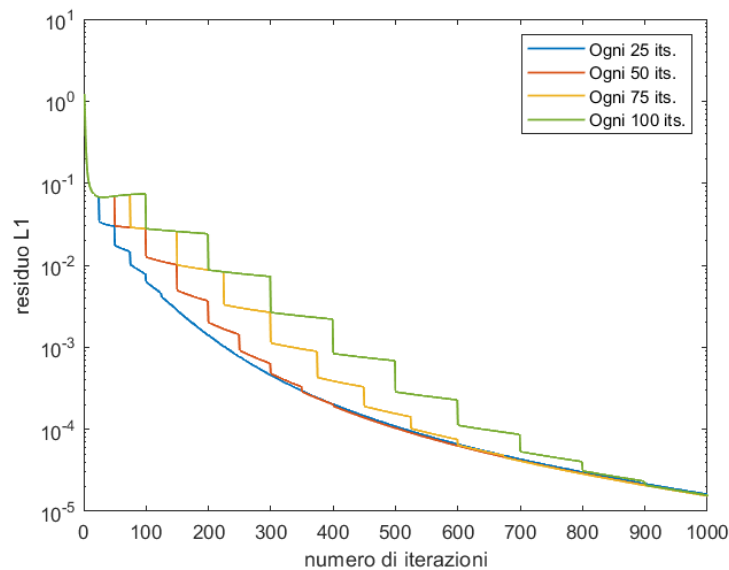


Figura 4.31: *Confronto per $\alpha = 0.99$*

Dopo le prime mille iterazioni i diversi andamenti del residuo risultano pressoché coincidenti. Per questo motivo, anche in questo caso, al fine di confrontare il metodo in questione con quello delle potenze standard e quello modificato mediante estrapolazione di Aitken, consideriamo la formulazione che applica l'estrapolazione quadratica ogni 100 iterazioni, così da contenere il più possibile il costo computazionale complessivo della procedura.

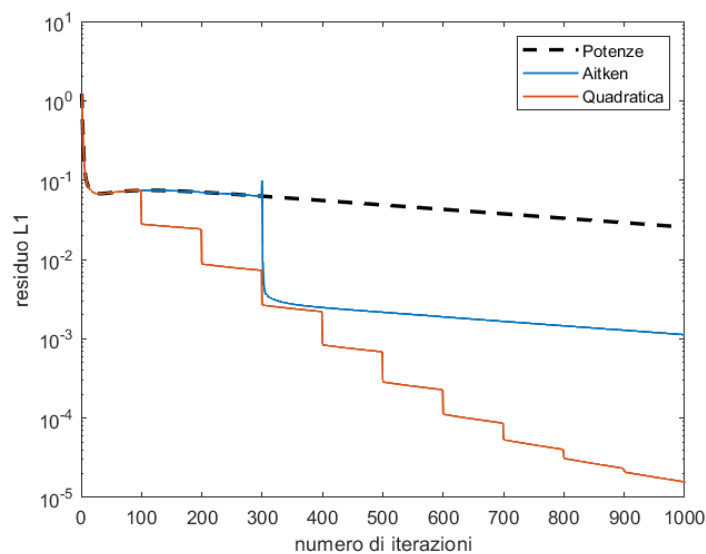


Figura 4.32: *Confronto dei tre metodi per $\alpha = 0.99$*

Al termine della millesima iterazione, la situazione è la seguente:

metodo	potenze	estr. Aitken	estr. quad.
residuo L_1	2.55×10^{-2}	1.13×10^{-3}	1.54×10^{-5}

La differenza osservata tra il valore del residuo del metodo delle potenze standard e quello del metodo modificato mediante l'extrapolazione di Aitken è di circa 1.35 ordini di grandezza: il residuo del metodo modificato risulta circa 23 volte inferiore rispetto a quello del metodo classico.

Per quanto riguarda la variante con extrapolazione quadratica, il residuo differisce da quello del metodo standard di circa 3.22 ordini di grandezza, risultando circa 1656 volte più piccolo dopo le prime mille iterazioni.

4.3 Matrice *soc-LiveJournal1*

Le proprietà della matrice *soc-LiveJournal1* sono, invece, le seguenti:

Nodi	Links	Densità
≈ 4.8 milioni	≈ 69.0 milioni	2.9×10^{-6}

Anche in questo caso ci troviamo di fronte a una matrice di enormi dimensioni, ma fortemente sparsa. Come illustrato nello Spy plot di Figura 4.33, l'apparente densità della vista globale è dovuta esclusivamente alle dimensioni del sistema; gli ingrandimenti proposti (di dimensione 10000×10000 e 5000×5000), permettono invece di apprezzare la reale sparsità della matrice.

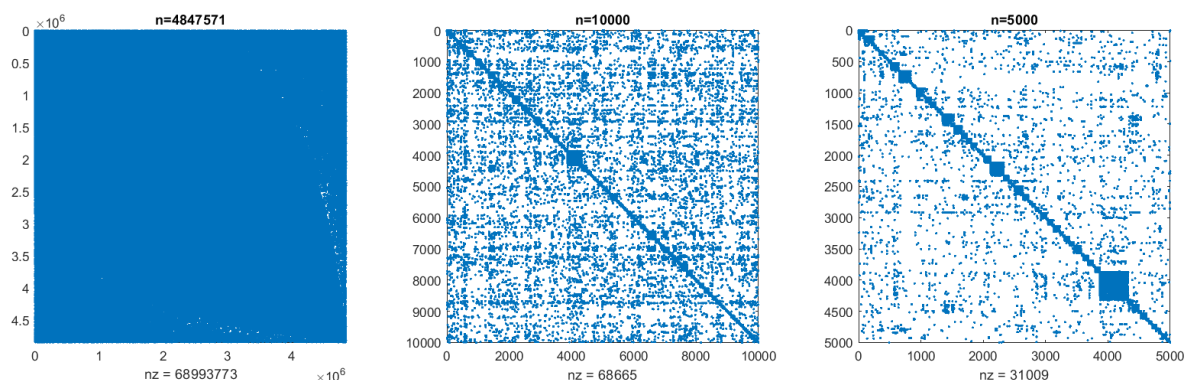


Figura 4.33: *Spy globale e di due sottomatrici*

4.3.1 Metodo delle potenze

Procediamo con l'analisi di convergenza del metodo delle potenze applicato alla matrice *soc-LiveJournal1*. Esaminiamo l'andamento del residuo L_1 al variare del parametro $\alpha \in \{0.5, 0.7, 0.85, 0.95, 0.99\}$, con tolleranza fissata uguale a 10^{-8} .

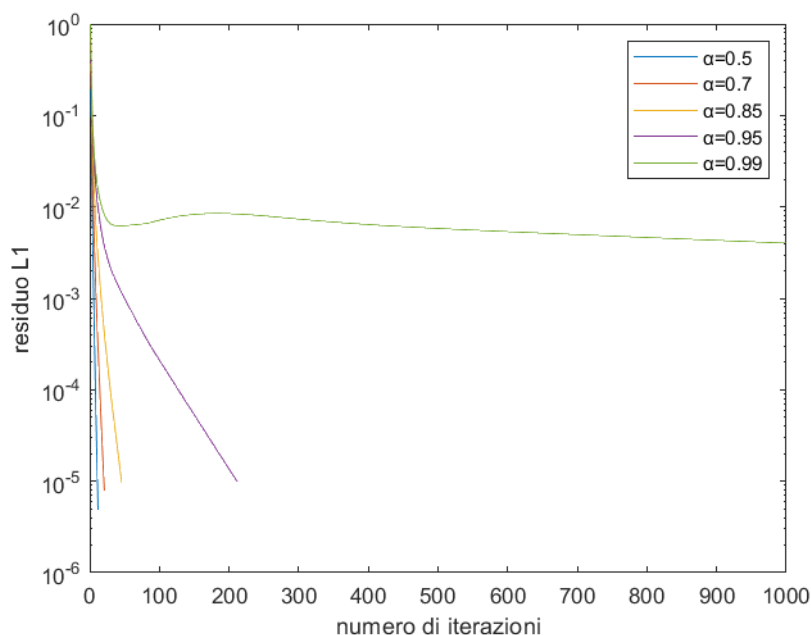


Figura 4.34: Convergenza del metodo delle potenze al variare di α applicato alla matrice *soc-LiveJournal1*

Riportando in una tabella il numero di iterazioni necessarie alla convergenza del metodo al variare di α , si ha:

α	0.5	0.7	0.85	0.95	0.99
iterazioni	12	21	46	212	>1000

Poiché, per la matrice *soc-LiveJournal1*, il metodo delle potenze converge in un numero piuttosto ridotto di iterazioni per $\alpha \leq 0.95$, l'analisi sperimentale si concentra sul caso più critico $\alpha = 0.99$, per il quale la convergenza è significativamente più lenta.

4.3.2 Estrapolazione di Aitken

Esaminiamo l'effetto dell'estrapolazione di Aitken sulla convergenza del metodo per $\alpha = 0.99$.

Confrontando il metodo delle potenze standard con i metodi modificati applicando l'extrapolazione di Aitken una sola volta in momenti diversi della procedura otteniamo il seguente grafico dei residui:

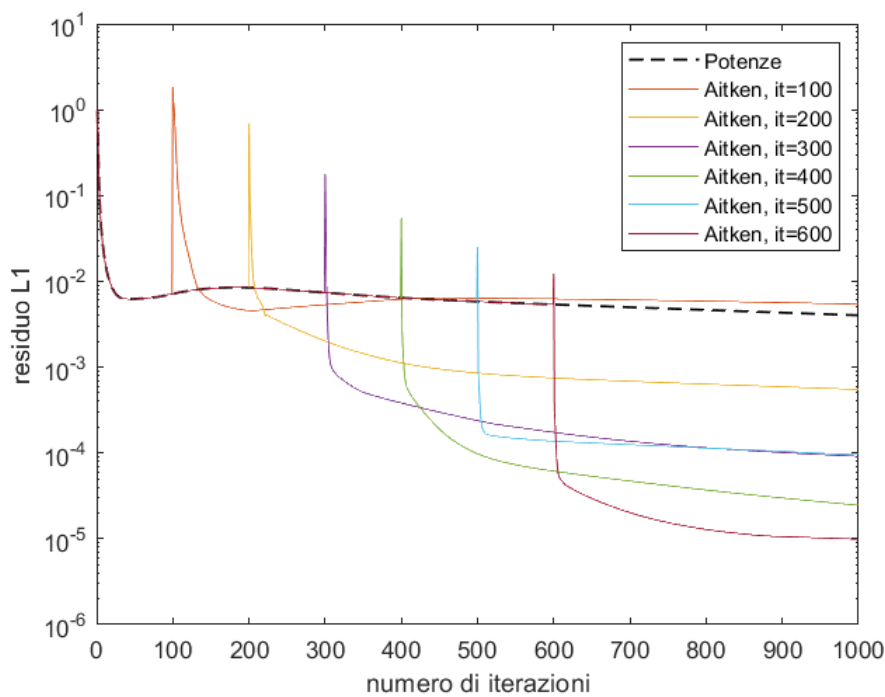


Figura 4.35: *Confronto per $\alpha = 0.99$*

Il residuo del metodo delle potenze modificato mediante extrapolazione applicata alla 600ª iterazione risulta inferiore di circa tre ordini di grandezza rispetto a quello ottenuto con il metodo delle potenze standard.

Anche in questo caso emerge chiaramente l'importanza del momento in cui viene eseguita l'extrapolazione. Nel nostro esperimento, infatti, applicarla alla 100ª iterazione si rivela svantaggioso: il residuo alla millesima iterazione risulta addirittura leggermente superiore a quello prodotto dal metodo delle potenze classico.

Un'applicazione alla 200ª iterazione produce invece un miglioramento limitato, con un residuo inferiore di poco più di un ordine di grandezza rispetto a quello del metodo standard.

4.3.3 Estrapolazione quadratica

Valutiamo, ora, l'effetto dell'extrapolazione quadratica sulla convergenza del metodo nel caso $\alpha = 0.99$.

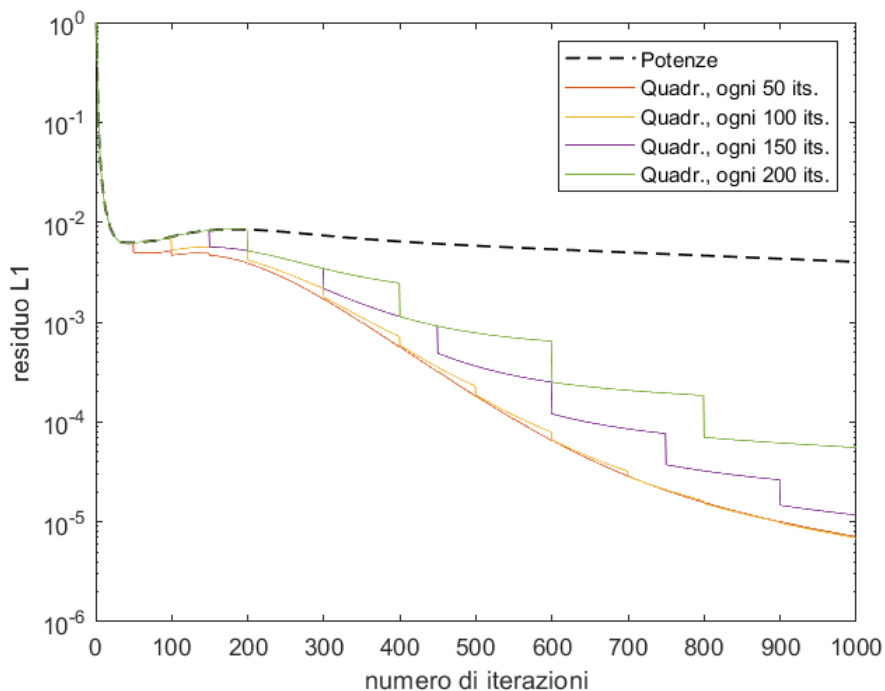


Figura 4.36: Confronto per $\alpha = 0.99$

In Figura 4.36 è riportato il confronto tra il metodo delle potenze standard applicato alla matrice *soc-LiveJournal1* e le varianti del metodo in cui viene eseguita periodicamente l'extrapolazione quadratica con periodi uguali a 50, 100, 150 e 200 iterazioni.

I risultati migliori in termini di convergenza si osservano quando l'extrapolazione viene applicata con una frequenza relativamente elevata. In particolare, eseguendola ogni 50 o 100 iterazioni, il residuo alla millesima iterazione è circa tre ordini di grandezza più piccolo rispetto a quello prodotto dal metodo delle potenze standard.

Nè tramite extrapolazione di Aitken, nè mediante extrapolazione quadratica il metodo converge entro le prime mille iterazioni. Inoltre, dalle Figure 4.35 e 4.36, pare che le due extrapolazioni, (quella di Aitken applicata alla 600^a e quella quadratica eseguita ogni 100 iterazioni), producano un effetto simile sulla convergenza: in entrambi i casi, infatti, il residuo alla millesima iterazione è leggermente inferiore a 10^{-5} . Per approfondire questo aspetto, procediamo come segue.

Alziamo il numero di iterazioni massime da 1000 a 2000: confrontando la convergenza del metodo delle potenze standard e dei due appena citati, si ha:

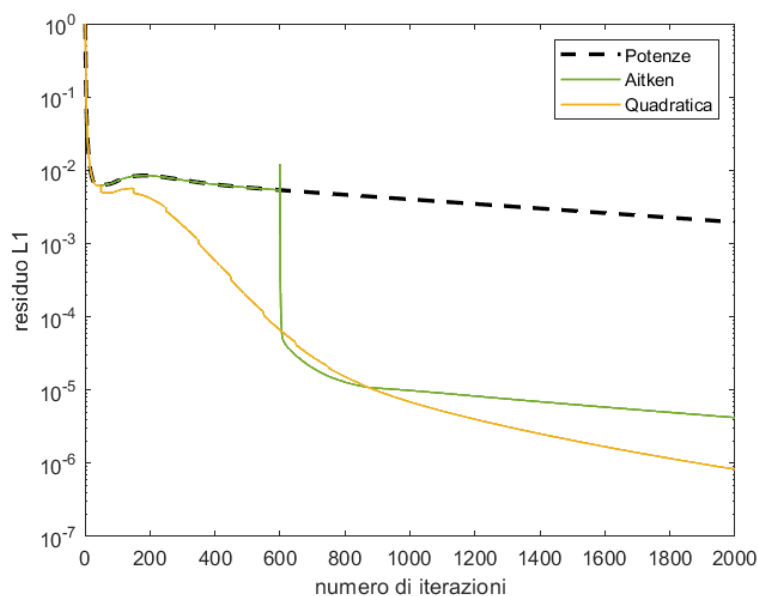


Figura 4.37: *Confronto dei tre metodi per $\alpha = 0.99$ e $maxit = 2000$*

Dopo 2000 iterazioni, la situazione è la seguente:

metodo	potenze	estr. Aitken	estr. quad.
residuo L_1	1.96×10^{-3}	4.21×10^{-6}	8.29×10^{-7}

Il residuo prodotto dal metodo delle potenze standard rimane ancora piuttosto elevato; come si osserva in Figura 4.37, il suo andamento decresce infatti molto lentamente. Per quanto riguarda i due metodi modificati mediante estrapolazione, si nota che inizialmente essi sembrano presentare un comportamento simile. Tuttavia, dopo un brusco calo iniziale, il residuo del metodo con estrapolazione di Aitken inizia a decrescere più lentamente rispetto a quello del metodo modificato con estrapolazione quadratica.

La differenza dei residui dei primi due metodi è uguale a circa 2.67 ordini di grandezza, in quanto il metodo con estrapolazione di Aitken produce un residuo dal valore 465 volte inferiore a quello del metodo standard.

La differenza tra il valore del residuo del metodo modificato con estrapolazione quadratica dopo mille iterazioni e quello del metodo classico, invece, è uguale a 3.37 ordini di grandezza: il primo risulta essere inferiore di circa 2364 volte rispetto al secondo, producendo un risultato decisamente migliore di quello prodotto dall'extrapolazione di Aitken.

4.3.4 Tempo di calcolo

Poiché, con tolleranza 10^{-8} , nessuno dei metodi considerati raggiunge la convergenza entro 2000 iterazioni, si considera anche una tolleranza meno stringente uguale a 10^{-5} , ai fini di evidenziare un aspetto molto significativo della convergenza: il tempo di calcolo, fattore estremamente importante nel momento in cui si ha a che fare con dati di queste dimensioni.

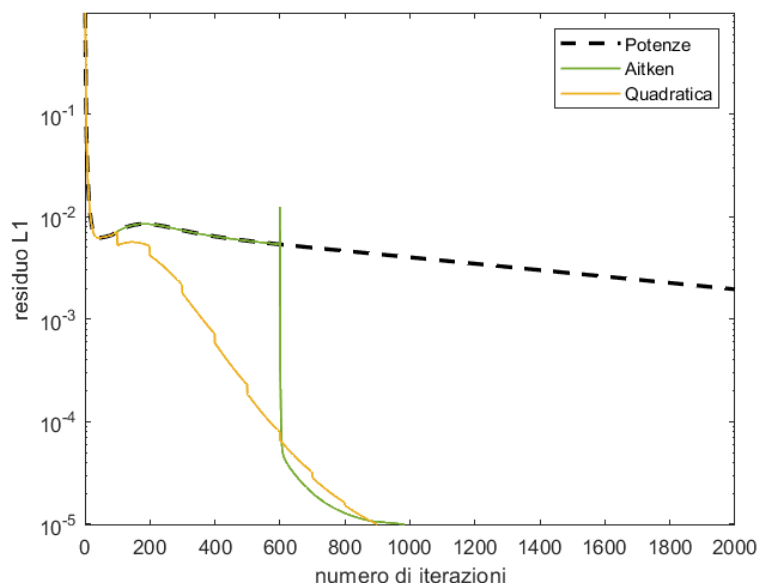


Figura 4.38: Confronto dei tre metodi per $\alpha = 0.99$, $maxit = 2000$ e $tol = 10^{-5}$

Riportando il numero di iterazioni necessarie alla convergenza e i tempi di esecuzione in una tabella si ha:

metodo	potenze	estr. Aitken	estr. quad.
n^o di iterazioni	>2000	983	899
tempo (secondi)	1.32×10^3	6.56×10^2	6.02×10^2

Si osserva che il metodo delle potenze impiega circa 22 minuti² per eseguire 2000 iterazioni senza tuttavia raggiungere la tolleranza prefissata. I metodi modificati mediante estrapolazione di Aitken ed estrapolazione quadratica, invece, convergono entro le prime 1000 iterazioni e richiedono rispettivamente circa 11 e 10 minuti.

²Il tempo di calcolo è una grandezza che può dipendere da diversi fattori; esecuzioni diverse possono produrre tempi di calcolo leggermente differenti.

4.4 Conclusioni

In questo capitolo sono stati analizzati gli effetti di due tecniche di estrapolazione sulla convergenza del metodo delle potenze applicato a tre matrici caratterizzate da un livello di sparsità simile ma da dimensioni differenti.

Dall'analisi condotta emerge che, per valori relativamente piccoli di α , l'utilizzo di tali tecniche tende ad appesantire inutilmente la procedura. Al contrario, per valori elevati di α esse risultano particolarmente efficaci nel ridurre in maniera significativa il numero di iterazioni necessarie alla convergenza del metodo.

Nel caso $\alpha = 0.85$, il risparmio iterativo osservato è stato moderato. Per la matrice *web-NotreDame*, la riduzione massima del numero di iterazioni è stata di circa il 26% nel caso dell'extrapolazione di Aitken e al 20.6% nel caso dell'extrapolazione quadratica. Questo rappresenta inoltre l'unico scenario in cui i benefici ottenuti mediante l'extrapolazione di Aitken risultano superiori a quelli della variante quadratica. Per la matrice *web-Google*, invece, il risparmio iterativo è stato uguale all'8.2% con l'extrapolazione di Aitken e al 9.6% con l'extrapolazione quadratica.

Per valori più elevati del parametro, in particolare $\alpha = 0.95$ e $\alpha = 0.99$, entrambe le tecniche hanno mostrato benefici molto più evidenti. Nel caso della matrice *web-NotreDame*, tuttavia, in entrambi i casi non è stato possibile raggiungere la tolleranza fissata entro le prime mille iterazioni; di conseguenza, l'analisi del risparmio iterativo è stata sostituita da un confronto tra gli ordini di grandezza dei residui ottenuti al termine delle mille iterazioni.

Per $\alpha = 0.95$, l'extrapolazione di Aitken e quella quadratica hanno prodotto una riduzione del residuo pari rispettivamente a 1.64 e 2.17 ordini di grandezza. Nel caso $\alpha = 0.99$, le riduzioni osservate sono state pari rispettivamente a 1.90 e 2.34 ordini di grandezza.

Per quanto riguarda la matrice *web-Google*, invece, nel caso $\alpha = 0.95$ il metodo delle potenze converge entro il termine fissato; l'applicazione delle due tecniche di estrapolazione consente di ottenere un risparmio iterativo del 17.3% nel caso dell'extrapolazione di Aitken e al 34% nel caso dell'extrapolazione quadratica.

Nel caso $\alpha = 0.99$, la convergenza del metodo richiede più di mille iterazioni e pertanto il confronto viene nuovamente effettuato in termini di residuo al termine di tali iterazioni. In questo scenario, l'extrapolazione di Aitken e quella quadratica producono una riduzione pari rispettivamente a 1.35 e 3.22 ordini di grandezza, evidenziando in

modo particolarmente marcato l'efficacia dell'estrapolazione quadratica.

Successivamente è stata considerata la matrice *soc-LiveJournal1*, di dimensioni significativamente maggiori rispetto alle precedenti. In questo caso, oltre alla convergenza dei metodi per $\alpha = 0.99$, è stato analizzato anche un aspetto particolarmente rilevante nei problemi di grandi dimensioni, ovvero il tempo di calcolo. A tal fine, la tolleranza è stata fissata a 10^{-5} e il numero massimo di iterazioni è stato portato a 2000.

I risultati mostrano che il metodo delle potenze, anche dopo 2000 iterazioni, risulta ancora lontano dal raggiungimento della convergenza e richiede circa 22 minuti di tempo di esecuzione. I metodi modificati mediante estrapolazione di Aitken ed estrapolazione quadratica, invece, raggiungono la convergenza in meno di mille iterazioni e impiegano rispettivamente circa 11 e 10 minuti, evidenziando quindi una riduzione significativa del tempo di calcolo rispetto al metodo delle potenze.

Bibliografia

- [1] S. Brin, L. Page, R. Motwani, T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*, Technical Report, Stanford InfoLab (1998).
- [2] S. Brin, L. Page, *The anatomy of a large-scale hypertextual Web search engine*, Computer Networks and ISDN Systems 30, pp. 107-117 (1998).
- [3] G. Grimmett, D. Stirzaker, *Probability and Random Processes*, Terza Edizione, Oxford University Press, pp. 213-215,222,260,261 (2001).
- [4] S.D. Kamvar, T.H. Haveliwala, C.D. Manning, G.H. Golub: *Extrapolation Methods for Accelerating PageRank Computations*, Proceedings of the 12th International World Wide Web Conference (WWW 2003), ACM (2003).
- [5] A. N. Langville, C. D. Meyer, *Deeper Inside PageRank*, Internet Mathematics, 1 (3), pp. 335–380 (2004).
- [6] A.N. Langville, C.D. Meyer : *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press, pp. 31-42 (2006).
- [7] A. Quarteroni, R. Sacco, F. Saleri, P. Gervasio, *Matematica Numerica*, Quarta Edizione, Springer, pp. 174,175 (2014).
- [8] V. Simoncini, D. Palitta, *Dispense del corso di Calcolo Numerico, Modulo di Algebra Lineare Numerica*, II edizione, v.7 (2022).
- [9] V. Simoncini, *Lucidi del corso di Matematica Computazionale: Metodi Matriciali per il Data Science* (2025).

Ringraziamenti

Ringrazio, innanzitutto, Valeria Simoncini per essere stata una relatrice presente, disponibile e scrupolosa, ma soprattutto perchè, grazie ai suoi corsi di Calcolo Numerico e Matematica Computazionale, mi ha permesso di scoprire gli ambiti della matematica sui quali desidero costruire il mio percorso.

Grazie mamma e papà per avermi sempre supportata, ascoltata e capita, ma al contempo ripresa nei momenti in cui, pur negandolo, ne avevo bisogno. Grazie, per i vostri sacrifici, sappiate che per me hanno un valore immenso. Grazie perchè senza il vostro supporto incondizionato e la vostra fiducia nei miei confronti, gli ostacoli incontrati lungo questo percorso sarebbero stati alti cento metri.

Grazie Elena per esserci sempre, quando eravamo più piccole e questi quattro anni di differenza pesavano di più non sapevo, ingenuamente, che crescendo saremmo diventate complici in tutto. Mi hai sempre compresa e ho sempre capito e apprezzato il tuo modo di starmi vicina nei momenti più bui facendomi ridere e non lasciandomi sola. Anch'io ci sarò sempre per te, ti voglio tantissimo bene (dato che non te lo dico spesso lo scrivo, così almeno "scripta manent").

Grazie nonna Mariarosa, nonostante abbiamo 62 anni di differenza io e te ci siamo sempre capite, sei una donna di larghe vedute e sei estremamente moderna nel modo di pensare. Grazie per la tua sensibilità, per l'attenzione che mi hai sempre dedicato, per la tua straordinaria empatia e per aver sempre saputo che ce l'avrei fatta. Sappi che anche io amo ascoltarti.

Grazie nonna Emma, per la tua stima incondizionata nei miei confronti, per la fiducia che hai da sempre nelle mie capacità, per pensarmi sempre. Grazie per tutti i racconti di quando eri più giovane, per me sono tanto preziosi e mi stimo molto per avere una nonna che è da sempre una donna così forte e in gamba.

Grazie ai miei zii Maurizio e Daniela, a Marco e Francesca per avermi sempre supportata e aver sempre creduto in me.

Grazie Andrea, perchè da quando sei nella mia vita anch'io credo di più in me stessa. Sei un meraviglioso valore aggiunto alla mia vita. Grazie per esserci sempre, per rendere più leggeri tutti i momenti pesanti, per farmi ridere anche quando lo vedo impensabile, per tenermi per terra ogni volta che rischio di prendere il volo assieme ai brutti pensieri, per ascoltarmi e sapermi sempre dare dei buoni consigli. Sei davvero un uomo speciale.

Grazie Alyssa, la nostra amicizia è da sempre uno dei miei tesori più preziosi. Siamo sintonizzate sulla stessa frequenza, dall'umorismo agli aspetti più profondi della vita. Da te mi sono sempre sentita compresa e mai giudicata, mi aiuti da sempre ad affrontare nel modo giusto le cose della vita, spazzando via le mie inutili paranoie. Grazie per essere la *k* della mia vita.

Grazie ai miei amici dell'università, in particolare grazie a Laura e Gaia: voi due siete il regalo più bello che l'università mi ha fatto, ci siamo sempre supportate e ogni volta che ci vediamo torno a casa con le guance che mi fanno male dal gran che ho riso. Nonostante la distanza fisica che ci separa vi sento sempre vicine, ogni volta che vi incontro mi sembra di non vedervi dal giorno prima. Siete preziose.

Grazie Anna per la tua sensibilità unica, per la tua empatia e per quanto sei capace di farmi ridere. Sei una persona davvero rara, spero di non perderti mai.

Grazie Jo, Smanz, Gabri, Busi, Fusto e Deiv per essere quelli che siete. Ogni volta che ci vediamo mi diverto tantissimo, mi avete sempre sostenuta e avete sempre avuto fiducia nelle mie capacità. Anche le vostre *prese* sono state una parte importante del mio percorso, che mi hanno sempre aiutata a non prendermi troppo sul serio.

Grazie Alessia, Dora, Enrico, Eleonora, Mery, Margherita, Federica: siete stati, ognuno a proprio modo, di grande supporto durante il mio percorso, nutro un grande affetto verso di voi.

Grazie a Enrico Montanari per avermi mostrato per primo la bellezza e la potenza della matematica. Non dimenticherò mai la passione con la quale l'ha fatto, è un insegnante speciale.