



Università di Cagliari

Facoltà di Ingegneria e Architettura
Ingegneria Elettronica Magistrale

TESINA DEL CORSO *Ottimizzazione*

Metodi iterativi per il calcolo approssimato di autovalori

Professore:

Prof. Giuseppe Rodriguez

Studente:

Rita Delussu

Matricola 70/83/65009

Indice

1	Metodo delle potenze	2
2	Algoritmo del metodo delle potenze	4
2.1	Metodo delle potenze inverse	5
2.2	Metodo delle potenze inverse con shift	6
3	Esperimenti numerici e risultati	7
3.1	Metodo delle potenze	7
3.1.1	Caso 1	8
3.1.2	Caso 2	9
3.1.3	Caso 3	10
3.1.4	Caso 4	11
3.2	Metodo delle potenze inverse	12
3.2.1	Caso 1	12
3.2.2	Caso 2	13
3.3	Metodo delle potenze inverse con shift	14

Introduzione

La conoscenza di autovalori e autovettori di una matrice quadrata è richiesta non solo nell'ambito matematico, ma anche in molte applicazioni, nelle quali si deve disporre di una loro buona approssimazione.

Per stimare gli autovalori e gli autovettori di una matrice A solitamente si ricorre alla approssimazione delle radici dell'equazione caratteristica

$$\det(A - \lambda I) = 0. \tag{1}$$

Tuttavia, tranne per qualche caso speciale, è sconsigliabile seguire tale procedimento, a causa degli errori che si introducono nel calcolo dei coefficienti della (1). Nei paragrafi successivi vengono esposti alcuni metodi iterativi utili per l'individuazione di alcuni autovalori di una matrice.

Capitolo 1

Metodo delle potenze

Non sempre è necessario conoscere lo spettro di una matrice A , cioè l'insieme di tutti i suoi autovalori; spesso ci si può limitare ad individuare quelli estremi, cioè quelli di modulo massimo e/o di modulo minimo. Il metodo delle potenze è un metodo iterativo per il calcolo dell'autovalore di massimo modulo di una matrice e il corrispondente autovettore. Esso converge quando sono verificate le seguenti condizioni:

- A è diagonalizzabile
- il vettore iniziale $x^{(0)}$ ha una componente non nulla lungo l'autovettore v_1 , corrispondente a λ_1
- l'autovalore di modulo massimo è separato dagli altri, ovvero $|\lambda_1| > |\lambda_i|, i = 2, \dots, n$

Il metodo delle potenze consiste nel fissare un vettore iniziale $x^{(0)}$ e calcolare la successione di vettore il cui termine k -esimo è $x^{(k)} = A^k x^{(0)}$. La diagonalizzabilità di A implica l'esistenza di n autovettori $x^{(i)}$ ($i = 1, \dots, n$), linearmente indipendenti e quindi che il vettore $x^{(0)}$ possa essere rappresentato nella forma

$$x^{(0)} = \sum_{i=1}^n \alpha_i v_i \quad (1.1)$$

Sfruttando la relazione (1.1) e applicando l'ultima ipotesi si ottiene

$$x^{(k)} = \alpha_1 \lambda_1^k \left(v_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k v_i \right) \quad (1.2)$$

Per la terza ipotesi, il rapporto $\left(\frac{\lambda_i}{\lambda_1} \right)^k$ converge a zero per $k \rightarrow \infty$ e di conseguenza la direzione vettore $x^{(k)}$ tende a quella dell'autovettore v_1 . Un

algoritmo di questo tipo può dar luogo a fenomeni di overflow o underflow in quanto può produrre vettori con componenti di valore assoluto eccessivamente grande o eccessivamente piccolo. Pertanto si preferisce normalizzare il vettore x_k ad ogni iterazione:

$$\begin{aligned}x_k &= Aq_{(k-1)} \\ q_k &= x_k / \|x_k\|\end{aligned}\tag{1.3}$$

Nell'applicazione di un metodo iterativo risulta necessario un criterio di arresto e la scelta di quest'ultimo determina la qualità dell'approssimazione della soluzione. Per motivi numerici il criterio di arresto viene implementato nella forma

$$|\lambda^{(k+1)} - \lambda^{(k)}| < \tau |\lambda^{(k+1)}|\tag{1.4}$$

dove $\tau > 0$ è la tolleranza che, per questo lavoro, è dell'ordine di 10^{-10} .

Esistono numerose varianti del metodo delle potenze. Una di queste è il metodo delle potenze inverse che permette di calcolare l'autovalore minimo della matrice in esame e il suo rispettivo autovettore. Questa generalizzazione del metodo delle potenze consiste nell'applicare il metodo, nel caso in cui la matrice in esame sia diagonalizzabile, alla matrice inversa per approssimare l'autovalore di modulo minimo, in quanto gli autovalori della matrice inversa sono i reciproci degli autovalori della matrice A.

Il metodo delle potenze può essere utilizzato anche per calcolare l'autovalore di A più vicino ad un dato valore μ . In questo caso al posto della matrice A, si utilizza $(A - \mu I)$. Questo algoritmo è noto come metodo delle potenze con shift ed il valore μ viene chiamato "shift". Si può ottenere il metodo delle potenze inverse imponendo lo shift pari a zero.

Capitolo 2

Algoritmo del metodo delle potenze

L'algoritmo che descrive il metodo delle potenze è il seguente:

```
1 function [lambda, k] = MP(A, x0, Nmax)
   q = x0/norm(x0);
3   k=0;
   tau = 1e-10;
5   lambda = q'*A*q;
   flag = true;
7   while(flag)
       k = k +1;
9       x_old = q;
       l_old = lambda;
11      x = A*q;
       q = x/norm(x);
13      lambda=q'*A*q;
       flag = and((abs(lambda - l_old) >=
15              tau*abs(lambda)), (k <= Nmax));
end
```

Listing 2.1: Algoritmo del metodo delle potenze.

L'algoritmo richiede in ingresso una matrice A di cui si vuole calcolare l'autovalore dominante, un vettore x_0 di innesco e il numero di iterazioni massimo (N_{max}) e restituisce in uscita l'autovalore massimo (λ) e il numero di iterazioni impiegate per ottenere la soluzione.

2.1 Metodo delle potenze inverse

Una variante particolarmente interessante del metodo delle potenze è utile nel caso in cui A sia una matrice quadrata con n autovettori linearmente indipendenti

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0 \quad (2.1)$$

e si decida di calcolare l'autovalore più piccolo in modulo. Il metodo delle potenze, applicato alla matrice inversa A^{-1} , fornisce l'autovalore massimo dell'inversa e quindi l'autovalore minimo della matrice A . Nell'algoritmo precedente si deve sostituire ad A la sua inversa, e quindi ad ogni iterazione si dovrebbe calcolare l'inversa di A . In genere si evita di calcolare la matrice inversa, infatti si preferisce risolvere un sistema lineare, utilizzando, ad esempio, la fattorizzazione $PA = LU$. Il costo computazionale è maggiore rispetto al metodo delle potenze, tuttavia è coinvolta solo la matrice A . Si riporta di seguito il codice relativo al metodo delle potenze inverse:

```

function [lambda, q, k, l_old, x_old, x] = MP(A, x0, Nmax)
2     q = x0/norm(x0);
     k=0;
4     tau = 1e-10;
     lambda = q'*A*q;
6     flag = true;
     while(flag)
8         k = k +1;
         [L, U, P] = lu(A);
10        x_old = q;
         l_old = lambda;
12        y = L\P*q;
         x = U\y;
14        q = x/norm(x);
         lambda=q'*A*q;
16        flag = and((abs(lambda - l_old) >=
                    tau*abs(lambda)),(k <= Nmax));
18 end

```

Listing 2.2: Algoritmo del metodo delle potenze inverse.

2.2 Metodo delle potenze inverse con shift

Questo metodo applica il metodo delle potenze inverse alla matrice $(A - \mu I)$, dove μ è l'approssimazione dell'autovalore che si vuole trovare. L'algoritmo restituisce l'autovalore più vicino a μ . Rispetto agli algoritmi precedenti la funzione richiede in ingresso anche il parametro n , il quale definisce la dimensione della matrice identità.

```
function [lambda, k] = MP(A, mu, n, x0, Nmax)
2     q = x0/norm(x0);
     k=0;
4     tau = 1e-10;
     lambda = q'*A*q;
6     flag = true;
     while(flag)
8         k = k +1;
         [L, U, P] = lu(A - mu * eye(n));
10        x_old = q;
         l_old = lambda;
12        y = L\P*q;
         x = U\y;
14        q = x/norm(x);
         lambda=q'*A*q;
16        flag = and((abs(lambda - l_old) >=
                    tau*abs(lambda)),(k <= Nmax));
18    end
```

Listing 2.3: Algoritmo del metodo delle potenze inverse con shift.

Capitolo 3

Esperimenti numerici e risultati

3.1 Metodo delle potenze

Gli esperimenti numerici sviluppati mostrano che il metodo converge o non converge a seconda delle caratteristiche della matrice A in ingresso. Sono stati esaminati quattro casi in cui la matrice A presenta le seguenti caratteristiche:

1. autovalore di modulo massimo con molteplicità algebrica pari a 1
2. autovalore di modulo massimo con molteplicità superiore a 1
3. autovalore di modulo massimo non unico
4. autovalore di modulo massimo complesso e coniugato

La matrice A e il vettore di innesco x_0 sono stati generati utilizzando due script: uno per gli autovalori reali e uno per gli autovalori complessi. In entrambi i casi il vettore iniziale è generato attraverso la funzione `rand` di Matlab. Inoltre l'algoritmo del metodo delle potenze restituisce due grafici in cui vengono mostrati l'autovalore e l'autovettore. Il grafico relativo all'autovalore mostra l'errore relativo (in scala semi-logaritmica) in funzione delle iterazioni; mentre il grafico inerente all'autovettore presenta l'errore assoluto in funzione delle iterazioni.

3.1.1 Caso 1

La matrice A presenta un autovalore massimo con molteplicità algebrica pari a 1. Il vettore degli autovalori scelto è:

$$au = [7 \ 5 \ 5 \ 9 \ 2 \ 1 \ 4]^T \quad (3.1)$$

L'algoritmo delle potenze, che termina dopo 46 iterazioni, restituisce il grafico Fig. 3.1.

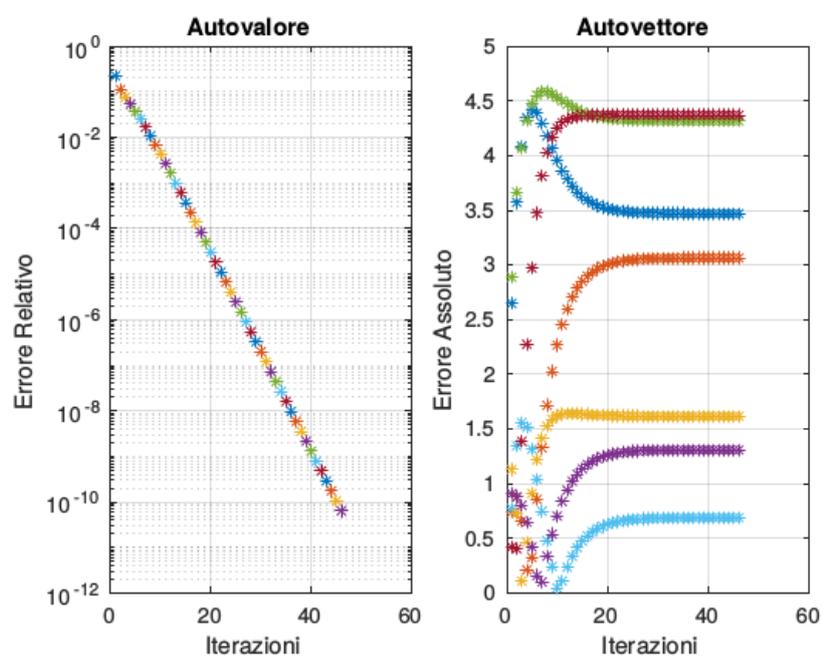


Figura 3.1: Autovalore e autovettore nel caso in cui la matrice abbia un autovalore massimo con molteplicità algebrica unitaria.

In questo caso il metodo delle potenze converge all'autovalore massimo.

3.1.2 Caso 2

Per questo caso viene utilizzato il vettore di autovalori del caso precedente (3.1) in cui vengono aggiunti altri elementi il cui valore è pari all'autovalore massimo. Quindi si ottiene:

$$au = [7 \ 5 \ 5 \ 9 \ 2 \ 1 \ 4 \ 9 \ 9]^T \quad (3.2)$$

Il metodo delle potenze è convergente anche nel caso in cui l'autovalore di modulo massimo abbia molteplicità algebrica maggiore di 1. Le iterazioni eseguite dall'algorithm prima di convergere sono 40.

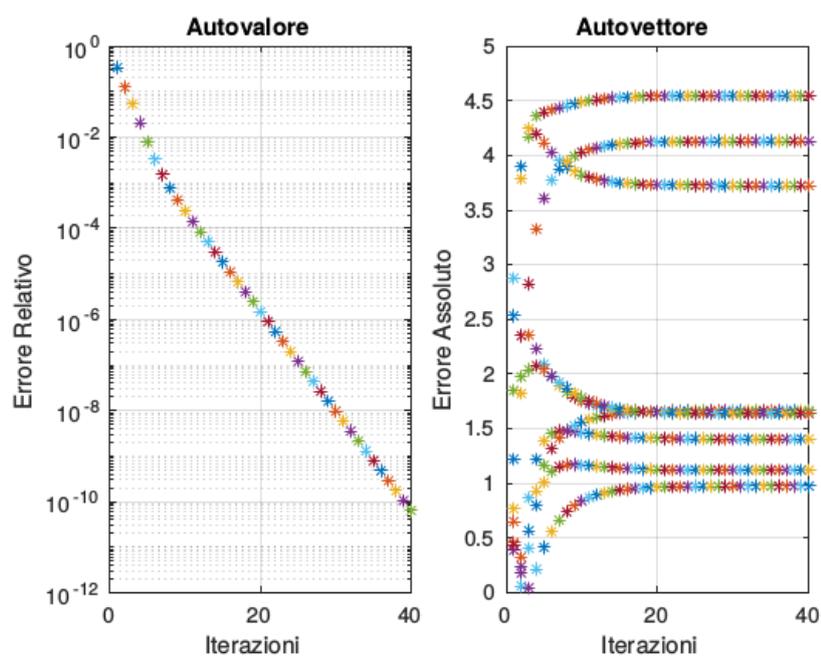


Figura 3.2: Autovalore e autovettore nel caso in cui l'autovalore massimo presenti una molteplicità algebrica maggiore di uno.

3.1.3 Caso 3

Utilizzando il vettore (3.1.2) e cambiando il segno dell'ultimo elemento si ottiene

$$au = \begin{bmatrix} 7 & 5 & 5 & 9 & 2 & 1 & 4 & 9 & -9 \end{bmatrix} \quad (3.3)$$

In questo caso l'autovalore di modulo massimo non è più unico. Dal grafico 3.3 a sinistra si potrebbe desumere che il metodo converga all'autovalore massimo, ma ciò non è vero in quanto l'autovettore non converge. Nonostante l'algoritmo termini alla 31-esima iterazione, il valore dell'autovalore è errato ($\lambda = 7.7307$).

Quindi nel caso in cui l'autovalore di modulo massimo non sia unico, i risultati di convergenza non sono più applicabili.

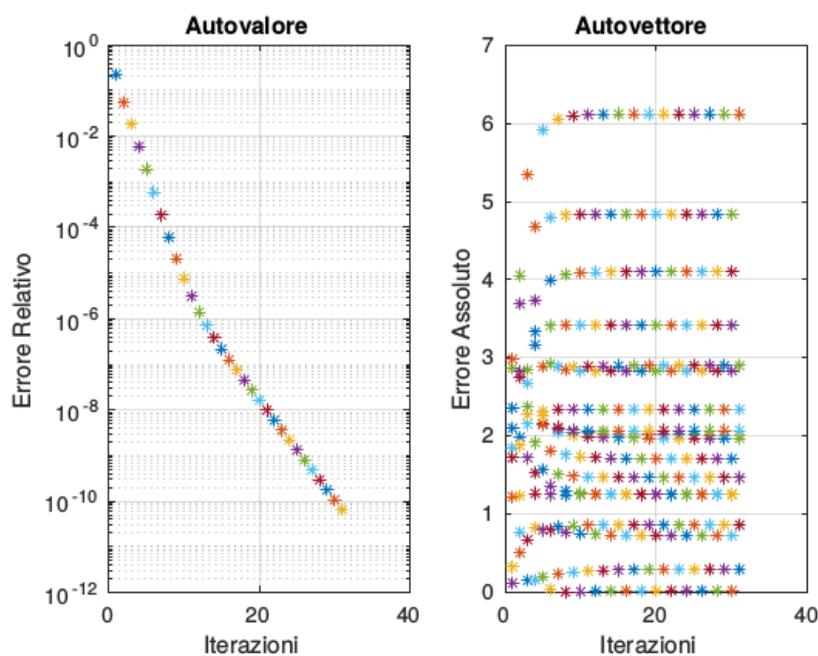


Figura 3.3: Autovalore e autovettore nel caso in cui l'autovalore massimo non è unico.

3.1.4 Caso 4

Per questo caso è stato utilizzato lo script che crea una matrice avente autovalori complessi coniugati e reali. Sono stati scelti i seguenti autovalori:

$$au = [1 \quad -1 \quad 0 \quad 1]^T \quad auC = [2 - i \quad 2 + i \quad 0 \quad 0]^T \quad (3.4)$$

dove con au vengono indicati gli autovalori reali, mentre con auC gli autovalori complessi. Anche in questo caso il metodo delle potenze non converge. Il grafico dell'autovalore sembrerebbe convergere, ma l'autovettore continua a oscillare. Quindi il metodo delle potenze non converge nel caso in cui la matrice A presenti un autovalore massimo complesso e coniugato.

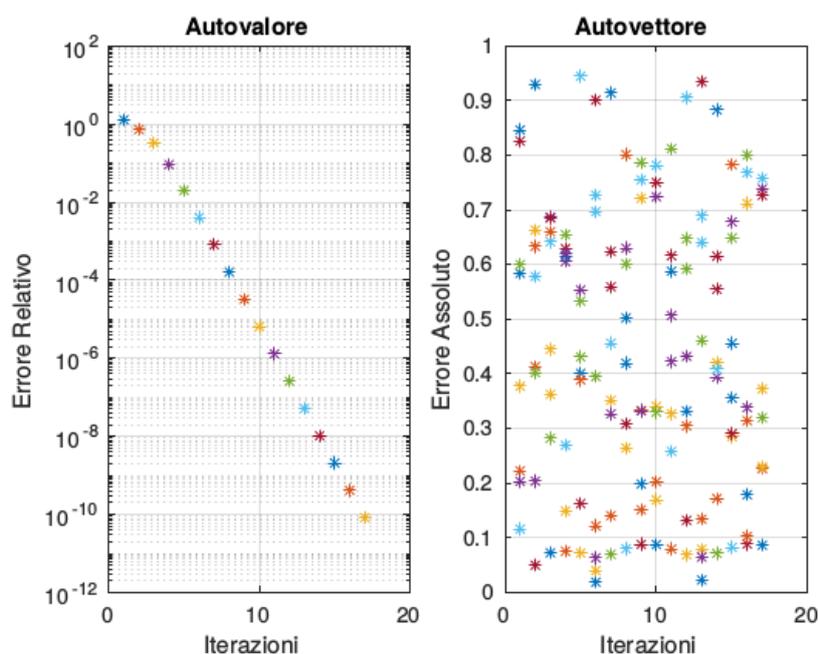


Figura 3.4: Autovalore e autovettore nel caso in cui la matrice A presenti un autovalore massimo complesso e coniugato.

3.2 Metodo delle potenze inverse

Gli esperimenti generati per questo metodo sono due: uno convergente e uno non convergente. Nei paragrafi successivi si riportano i risultati ottenuti attraverso gli esperimenti numerici.

3.2.1 Caso 1

Il vettore di autovalori a cui ci si riferisce è (3.1). In questo caso, a differenza del caso trattato nel metodo delle potenze, l'algoritmo tenderà all'autovalore minimo che risulta essere di valore pari a 1. L'algoritmo converge all'autovalore minimo dopo 18 iterazioni.

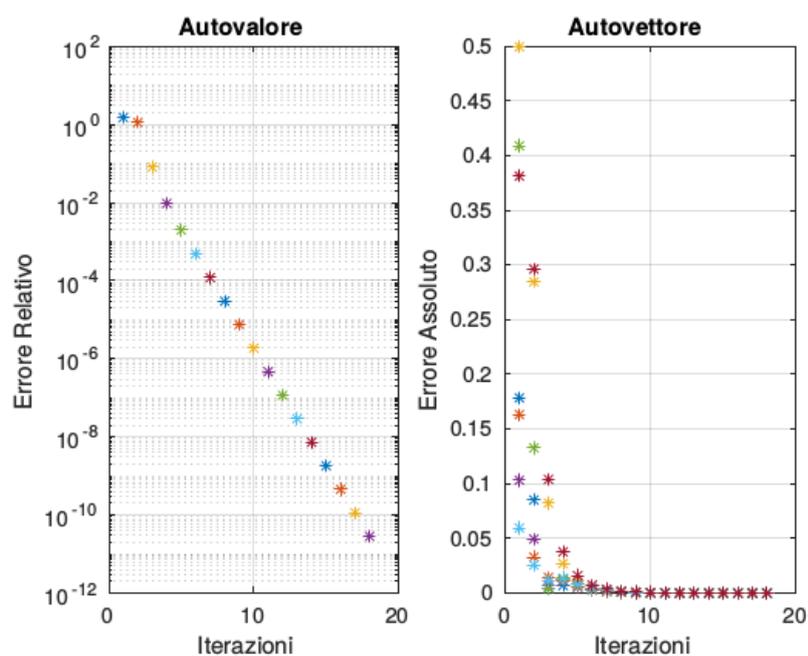


Figura 3.5: Autovalore e autovettore nel caso in cui la matrice sia caratterizzata da un autovalore minimo con molteplicità algebrica unitaria.

3.2.2 Caso 2

Un caso in cui non converge il metodo delle potenze inverse è quello in cui la matrice in esame abbia l'autovalore minimo prossimo allo zero. Il vettore scelto per questo esperimento numerico è

$$au = \left[7 \quad 5 \quad 5 \quad 9 \quad 2 \quad 1 \quad 2 \cdot 10^{-12} \right]^T \quad (3.5)$$

L'algoritmo termina dopo 4 iterazioni. Questo potrebbe portare a dire che l'esperimento sia andato a buon fine, ma ciò viene smentito dal grafico dell'autovalore che, come mostra la Fig. 3.6, continua a oscillare.

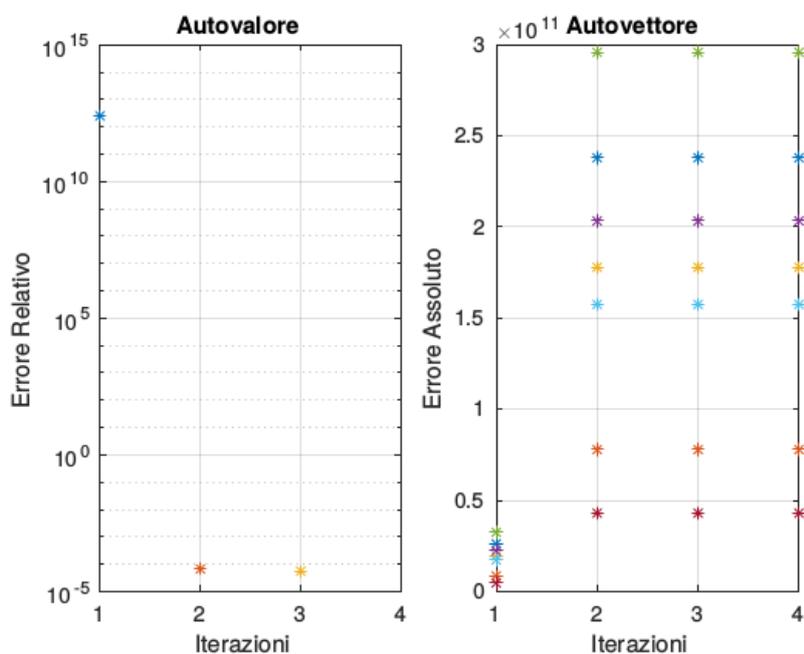


Figura 3.6: Autovalore e autovettore nel caso in cui la matrice sia caratterizzata da un autovalore minimo quasi nullo.

3.3 Metodo delle potenze inverse con shift

Per questa sezione sono stati implementati degli esperimenti numerici in cui μ è un'approssimazione dell'autovalore della matrice in esame. Nel caso in cui $\mu = \lambda$ l'algoritmo non converge.

Dato che si deve utilizzare un'approssimazione dell'autovalore della matrice, è necessario studiare quanto ci si può avvicinare all'autovalore vero.

Si prendano in considerazione i seguenti autovalori reali:

$$au = [10 \ 5 \ 7 \ 8 \ 3 \ 2 \ 75 \ 4]^T \quad (3.6)$$

In questo caso si ricerca l'autovalore pari a 10. Inizialmente viene utilizzato un valore di $\mu = 9$. Utilizzando questo valore come approssimazione dell'autovalore, l'algoritmo converge dopo 17 iterazioni (Fig. 3.7).

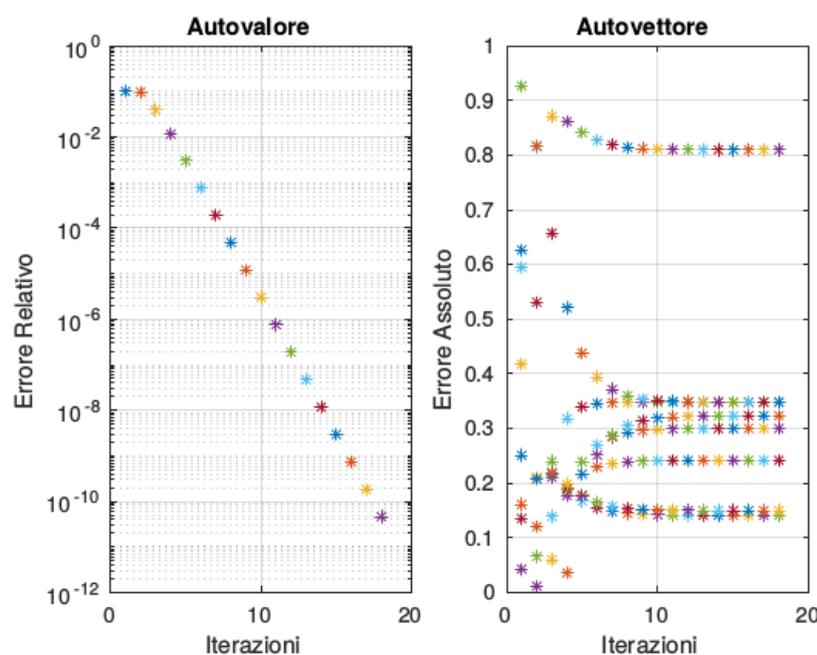


Figura 3.7: Autovalore e autovettore. Metodo delle potenze con shift utilizzato nel caso in cui si approssima l'autovalore con un valore di μ pari a 9.

Successivamente il valore di μ viene approssimato ulteriormente, ossia $\mu = 9.9$. A partire da questo valore l'algoritmo inizia a non convergere (Fig. 3.8 in alto). Avvicinandosi ulteriormente all'autovalore vero, l'autovettore diverge velocemente (Fig. 3.8 in basso).

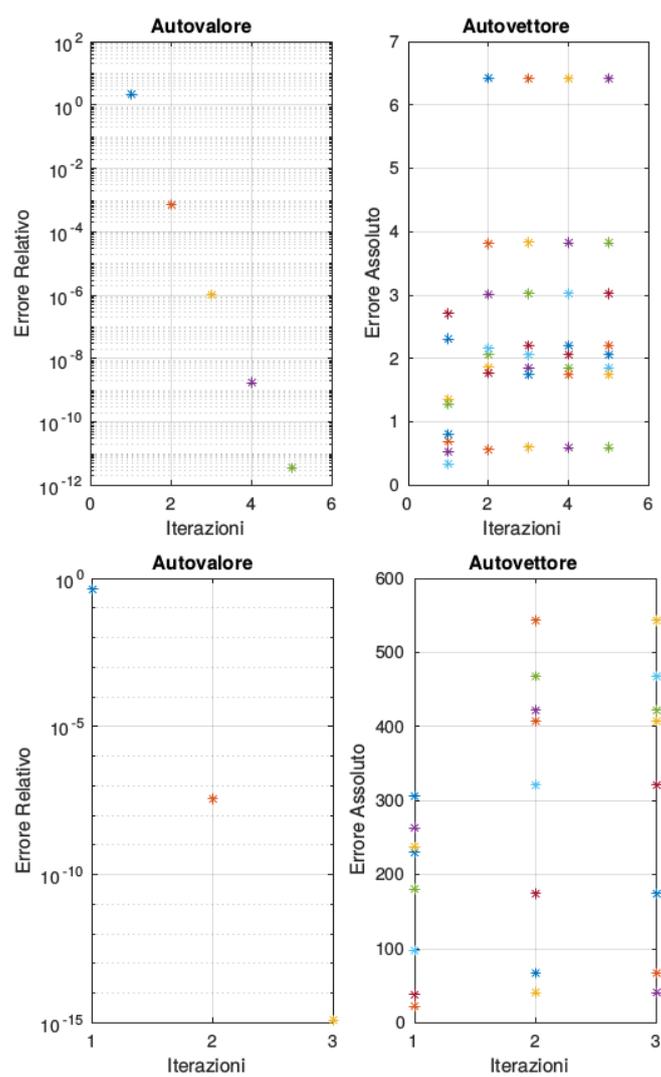


Figura 3.8: Autovalore e autovettore nel caso in cui il valore di $\mu = 9.9$ (in alto), $\mu = 9.999$ (in basso).

Conclusioni

In questo lavoro sono stati presentati alcuni metodi iterativi utilizzabili nel momento in cui è necessario conoscere gli autovalori di una matrice. A seconda dell'applicazione non sempre è utile conoscere tutti gli autovalori. Attraverso questi metodi iterativi è possibile ricavare delle informazioni sull'autovalore massimo (metodo delle potenze), sull'autovalore minimo (metodo delle potenze inverse) e su un autovalore che si vuole trovare (metodo delle potenze inverse con shift). Gli esperimenti numerici implementati hanno dimostrato in quali casi i vari metodi esposti convergono o tendono a divergere.

Appendice

Per maggiore chiarezza vengono qui presentati i due script utilizzati per la creazione della matrice A.

Il primo script di Matlab è relativo alla generazione di una matrice avente autovalori reali. L'algoritmo necessita di due elementi in ingresso chiamati *au* e *dim*. Il primo si riferisce a una lista di autovalori desiderati, mentre il secondo è necessario per definire la dimensione della matrice ortogonale Q.

```
function[A, x0] = autovalori (au, dim)
2       Q = orth(rand(dim));
        D = diag(au);
4       A = Q'*D*Q;
        x0 = rand(dim, 1);
6 end
```

Listing 3.1: Autovalori reali. Algoritmo implementato per la generazione di una matrice A caratterizzata da autovalori scelti dall'utente.

Questo script restituisce una matrice A e un vettore x_0 . Questi elementi possono essere utilizzati come parametri di ingresso per l'algoritmo del metodo delle potenze.

Il secondo script è stato utilizzato per creare una matrice con alcuni autovalori complessi. Questo script richiede in ingresso due liste di autovalori. In questo lavoro è stato scelto di utilizzare *au* per indicare gli autovalori reali e con *auC* gli autovalori complessi. Si ottiene in uscita una matrice A la quale sarà caratterizzata dagli autovalori scelti e un vettore x_0 .

```
function[A, x0] = autovaloriC(au, auC)
2       D = diag(au);
        C = diag(auC);
4       dim = size(au,2);
        dimC = size(auC,2);
6       AD = [C zeros(dimC); zeros(dim) D];
        dimAD = size(AD);
```

```
8      Q = orth(rand(dimAD));  
      A = Q'*AD*Q;  
10     dimX = dim + dimC;  
      x0 = rand(dimX, 1);  
12 end
```

Listing 3.2: Autovalori complessi. Algoritmo implementato per la generazione di una matrice A caratterizzata da autovalori complessi scelti dall'utente.