

LAB 3 MATLAB

Esercizio 1

L'obiettivo di questo esercizio è quello di calcolare gli autovalori di due matrici in due modi diversi e confrontare gli output risultanti.

Come indicato dal testo partiamo calcolando n usando la formula:

$$n = 10(d_1 + 1) + d_0$$

d_1 e d_0 sono dati dalla penultima e ultima cifra della matricola

La matrice A viene calcolata come indicata dal testo in questo modo:

$$A = \text{diag}(\text{ones}(1, n - 1), 1) + \text{eye}(n)$$

Calcoliamo E e B in questo modo:

```
E = zeros(n);  
E(n, 1) = 2^(-n);  
  
B = A + E;
```

Creiamo la matrice E pari a 0 di dimensioni $n \times n$. Questa matrice è destinata a rappresentare la perturbazione rispetto alla matrice A .

Inseriamo la perturbazione con il comando $E(n, 1) = 2^{-n}$. Infine sommiamo le matrici come richiesto dal testo.

Parte a

Per quanto riguarda gli autovalori usiamo la funzione `eig` di matlab.

Una volta generati gli autovalori dobbiamo confrontarli in questi due modi:

- Confronto puntuale
- Confronto in norma

A livello di codice è il seguente:

```
% Confronta con norma  
norma1 = norm(B - A) / norm(A); %  
norma2 = norm(VB - VA) / norm(VA)
```

L'output del confronto puntuale è:

	1
	MATLAB Online
2	0.5000
3	0.5000
4	0.5000
5	0.5000
6	0.5000
7	0.5000
8	0.5000
9	0.5000
10	0.5000
11	0.5000
12	0.5000
13	0.5000
14	0.5000
15	0.5000
16	0.5000
17	0.5000
18	0.5000
19	0.5000
20	0.5000
21	0.5000
22	0.5000
23	0.5000
24	0.5000
25	0.5000
26	0.5000
27	0.5000
28	0.5000
29	0.5000
30	0.5000
31	0.5000
32	0.5000
33	0.5000
34	0.5000

La differenza puntuale calcolata tra gli autovalori di A e B risulta costante e pari a 0.5 per ogni componente. Questo accade perché la matrice E, che introduce una piccola perturbazione in A, modifica tutti gli autovalori in modo uniforme a causa della struttura regolare di A.

Per quanto riguarda l'output della norma1 abbiamo ottenuto **2.913402281235099e-11** mentre l'output della norma2 è **0.5000000000000000**. La norma1 misura quanto la matrice perturbata B differisce dalla matrice originale A in termini relativi. L'output ottenuto infatti è molto piccolo ed indica che la perturbazione introdotta da E ha avuto un effetto poco rilevante sull'intera matrice B. Mentre per la norma2 l'output 0.5 indica che nonostante la perturbazione E sia piccola, il cambiamento è stato sufficiente a modificare proporzionalmente tutti gli autovalori in modo uniforme.

Parte b

Ripetiamo l'esercizio per $A^t A$ e $B^t B$.

L'output del confronto puntuale è:

1
3.0704e-13
9.1671e-13
1.5131e-12
2.0882e-12
2.6321e-12
3.1395e-12
3.6016e-12
4.0131e-12
4.3685e-12
4.6636e-12
4.8955e-12
5.0623e-12
5.1612e-12
5.1956e-12
5.1641e-12
5.0715e-12
4.9201e-12
4.7158e-12
4.4635e-12
4.1691e-12
3.8409e-12
3.4861e-12
3.1140e-12
2.7303e-12
2.3443e-12
1.9655e-12
1.6009e-12
1.2603e-12
9.4458e-13
6.6924e-13
4.3521e-13
2.4913e-13
1.1147e-13
2.8422e-14

La differenza puntuale tra gli autovalori di $A^t A$ e $B^t B$ è estremamente piccola. La perturbazione introdotta da E è trascurabile sul prodotto matriciale rispetto alla matrice di partenza, e i valori decrescono progressivamente, mostrando un impatto maggiore sugli autovalori più grandi.

Per quanto riguarda l'output della norma3 abbiamo ottenuto **1.458212324478072e-11** mentre l'output della norma2 è **1.404970982458071e-12**. L'output della norma3 indica che l'effetto della perturbazione introdotta da E è estremamente piccolo anche a livello del prodotto matriciale $A^t A$ rispetto alla sua norma. Questo risultato conferma che il prodotto matriciale $A^t A$ è molto stabile rispetto alla perturbazione.

Mentre l'output della norma4 è ancora più piccolo rispetto alla norma3, evidenziando che la perturbazione ha un effetto praticamente trascurabile sugli autovalori del prodotto matriciale. Gli autovalori di $A^t A$ sono molto stabili rispetto a piccole perturbazioni.

Esercizio 2

Generiamo A manualmente. La diagonale D viene costruita tramite la formula fornita dal testo:

$$D = \text{diag}(g_1 \dots g_n)$$

La matrice G è ottenuta moltiplicando A per l'inversa di D:

$$G = A * D^{-1}$$

In matlab per rappresentare la funzione inversa di D usiamo (D^{-1}) .

Gli autovalori e autovettori di G sono stati calcolati utilizzando:

```
% Calcolo degli autovalori e autovettori
[autovettori, autovalori] = eig(G);
```

Output ottenuti

Autovalori di G:

	1	2	3	4	5	6	7	8	9	10	11
1	1.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
2	0.0000 + 0.0000i	0.7640 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
3	0.0000 + 0.0000i	0.0000 + 0.0000i	0.5774 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
4	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.2241 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
5	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	-0.8824 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
6	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	-0.7255 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
7	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	-0.5774 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
8	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	-0.3802 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
9	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0
10	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	3.5724 + 0.0000i	0
11	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i	-8.128 + 0.0000i

Gli autovalori della matrice G confermano le proprietà teoriche attese. Notiamo come il primo autovalore sia uguale ad 1 mentre gli altri hanno modulo minore di 1.

Autovettori di G:

	1	2	3	4	5	6	7	8	9	10	11
1	-0.6124 + 0.0000i	-0.0521 + 0.0000i	-0.7385 + 0.0000i	-0.2314 + 0.0000i	0.6272 + 0.0000i	0.3765 + 0.0000i	0.7385 + 0.0000i	0.0708 + 0.0000i	0.0000 - 0.0000i	0.0000 + 0.0000i	-0.0000 + 0.0000i
2	-0.1021 + 0.0000i	-0.0114 + 0.0000i	-0.2132 + 0.0000i	-0.1721 + 0.0000i	-0.1185 + 0.0000i	-0.0865 + 0.0000i	-0.2132 + 0.0000i	-0.0310 + 0.0000i	-0.5890 + 0.0000i	-0.5890 + 0.0000i	0.1711 + 0.0000i
3	-0.2041 + 0.0000i	-0.2634 + 0.0000i	0.0000 + 0.0000i	-0.2283 + 0.0000i	-0.3212 + 0.0000i	0.0441 + 0.0000i	-0.0000 + 0.0000i	-0.1304 + 0.0000i	0.2488 - 0.0000i	0.2488 + 0.0000i	-0.4699 + 0.0000i
4	-0.3062 + 0.0000i	-0.1727 + 0.0000i	0.0000 + 0.0000i	0.4857 + 0.0000i	-0.3307 + 0.0000i	0.1606 + 0.0000i	0.0000 + 0.0000i	0.3644 + 0.0000i	0.0087 + 0.0000i	0.0087 - 0.0000i	-0.5233 + 0.0000i
5	-0.4082 + 0.0000i	0.2772 + 0.0000i	-0.0000 + 0.0000i	0.6400 + 0.0000i	0.0335 + 0.0000i	-0.3382 + 0.0000i	-0.0000 + 0.0000i	-0.7525 + 0.0000i	-0.0000 + 0.0000i	-0.0000 - 0.0000i	0.0000 + 0.0000i
6	-0.3062 + 0.0000i	0.3086 + 0.0000i	-0.0000 + 0.0000i	0.2539 + 0.0000i	-0.1622 + 0.0000i	-0.2735 + 0.0000i	-0.0000 + 0.0000i	0.5011 + 0.0000i	-0.0087 - 0.0000i	-0.0087 + 0.0000i	0.5233 + 0.0000i
7	-0.1021 + 0.0000i	-0.0114 + 0.0000i	-0.2132 + 0.0000i	-0.1721 + 0.0000i	-0.1185 + 0.0000i	-0.0865 + 0.0000i	-0.2132 + 0.0000i	-0.0310 + 0.0000i	0.4646 + 0.0000i	0.4646 - 0.0000i	0.0639 + 0.0000i
8	-0.3062 + 0.0000i	0.5255 + 0.0000i	0.3693 + 0.0000i	-0.1936 + 0.0000i	0.0905 + 0.0000i	0.6606 + 0.0000i	-0.3693 + 0.0000i	-0.0425 + 0.0000i	0.0000 + 0.0000i	0.0000 - 0.0000i	-0.0000 + 0.0000i
9	-0.1021 + 0.0000i	0.2293 + 0.0000i	0.2132 + 0.0000i	-0.2880 + 0.0000i	-0.0342 + 0.0000i	-0.3036 + 0.0000i	0.2132 + 0.0000i	0.0373 + 0.0000i	0.0029 + 0.0000i	0.0029 - 0.0000i	-0.1744 + 0.0000i
10	-0.3062 + 0.0000i	-0.5776 + 0.0000i	0.3693 + 0.0000i	-0.0378 + 0.0000i	0.5367 + 0.0000i	-0.2842 + 0.0000i	-0.3693 + 0.0000i	0.1133 + 0.0000i	-0.0000 - 0.0000i	-0.0000 + 0.0000i	-0.0000 + 0.0000i
11	-0.1021 + 0.0000i	-0.2520 + 0.0000i	0.2132 + 0.0000i	-0.0562 + 0.0000i	-0.2028 + 0.0000i	0.1306 + 0.0000i	0.2132 + 0.0000i	-0.0994 + 0.0000i	-0.1273 + 0.0000i	-0.1273 - 0.0000i	0.4094 + 0.0000i

L'autovettore associato all'autovalore $\lambda=1$ è dato dalla prima colonna della matrice degli autovettori:

$$x = [-0.6124, -0.1021, -0.2041, -0.3062, -0.4082, -0.3062, -0.1021, -0.3062, -0.1021, -0.3062, -0.1021]^T$$

Questo autovettore deve essere normalizzato per verificare che le sue componenti siano comprese tra 0 e 1:

$$x_{norm} = \frac{x}{\max(x)}$$

Dove :

$$\begin{aligned}\min(x) &= -0.6124 \\ \max(x) &= -0.1021\end{aligned}$$

Per verificare che le componenti risultino effettivamente comprese tra 0 e 1 dobbiamo svolgere i calcoli manualmente per ciascuna x , a seguito mettiamo sotto i primi due calcoli effettuati:

Per $x_1 = -0.6124$:

$$x_{norm,1} = \frac{-0.6124}{-0.1021} = 5.99$$

Per $x_2 = -0.1021$:

$$x_{norm,2} = \frac{-0.1021}{-0.1021} = 1.00$$

Il risultato normalizzato è:

$$x_{norm} = [5.99, 1.00, 1.99, 2.99, 3.99, 2.99, 1.00, 2.99, 1.00, 2.99, 1.00]$$

NOTA: per verificare la veridicità dei calcoli elencati qua sopra guardare il codice commentato su matlab (riga 88)

Infine per il terzo punto della parte C notiamo dall'analisi degli autovettori che il corrispondente autovettore ha componenti sia positive che negative.

Per ogni stazione ferroviaria (nodo del grafo), è stata calcolata una stima intuitiva della sua importanza relativa sulla base dell'autovettore normalizzato associato all'autovalore $\lambda = 1$. La tabella seguente riporta le stazioni ferroviarie con la loro rispettiva importanza:

Nodo	Stazione	Importanza relativa (x_{norm})
1	Milano	5.99
2	Pavia	1.00
3	Lodi	1.99
4	Brescia	2.99
5	Bergamo	3.99
6	Como	2.99
7	Varese	1.00
8	Lecco	2.99
9	Sondrio	1.00
10	Cremona	2.99
11	Mantova	1.00

Le stazioni con importanza più alta sono quelle centrali o con molte connessioni, come **Milano**, **Bergamo**, **Brescia**, e **Cremona**. Questa analisi riflette il concetto di centralità

utilizzato nei motori di ricerca, come il [PageRank](#), per determinare la rilevanza dei nodi in un grafo.

Esercizio 3

L'obiettivo di questo esercizio è utilizzare il metodo delle potenze e il metodo delle potenze inverse per stimare gli autovalori dominanti di una matrice data e analizzare il comportamento di convergenza dei due metodi.

Parte a

Per il metodo delle potenze, sono stati scelti due vettori iniziali:

$$v_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad v_2 = \begin{bmatrix} 3 \\ 10 \\ 4 \end{bmatrix}$$

Il metodo delle potenze converge all'autovalore di massimo modulo, l'algoritmo segue questa scala:

- a) Si riordinano gli autovalori in base al massimo modulo, ottenendo $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$ dove λ_1 è l'autovalore di massimo modulo
- b) Converge se $\lambda_1 > \lambda_2$
- c) Si trova la velocità di convergenza con

$$V = \left(\frac{\lambda_2}{\lambda_1}\right)^k$$

Nel nostro caso gli autovalori riordinati in base al massimo modulo sono:

$$|\lambda_1|=5, |\lambda_2|=3, |\lambda_3|=1$$

L'autovalore di massimo modulo è λ_1

Nel nostro caso il punto b è soddisfatto.

La velocità di convergenza è:

$$V = \left(\frac{\lambda_2}{\lambda_1}\right)^k = \left(\frac{3}{5}\right)^k$$

Output ottenuti

A	[1.-1.2:-2.0.5:6.-3.6]	3x3	double
autovalore 1	5.0000	1x1	double
autovalore 2	3.0000	1x1	double
autovettore 1	[0.2033:0.6505:0.7318]	3x1	double
autovettore 2	[0.1374:0.8242:0.5494]	3x1	double
iter 1	29	1x1	double
iter 2	16	1x1	double
max iter	1000	1x1	double
tolerance	1.0000e-06	1x1	double
v1	[1:1:1]	3x1	double
v2	[3:10:4]	3x1	double

Parte b

L'algoritmo del metodo delle potenze inverse segue questa scala:

- a) Dato uno shift p , bisogna trovare l'autovalore più vicino a p (λ_1) ed il secondo più vicino a p (λ_2)

- b) Si calcola:

$$\mu_i = \frac{1}{\lambda_i - p}$$

per ogni lambda

- c) μ_i sarà quello di massimo modulo tra tutti μ calcolati, mentre μ_2 sarà quello subito più piccolo.

- d) La velocità di convergenza si calcola:

$$V = \left(\frac{\mu_2}{\mu_1}\right)^k$$

Se si hanno due autovalori con lo stesso modulo massimo, allora non converge.

Output ottenuti

A	[1.-1.2:-2.0.5:6.-3...	3x3	double
autovalore 1	5.0000	1x1	double
autovalore 2	3.0000	1x1	double
autovettore 1	[0.2033:0.6505:0...	3x1	double
autovettore 2	[0.1374:0.8242:0...	3x1	double
autovettore inv	[0.2033:0.6505:0...	3x1	double
iter 1	29	1x1	double
iter 2	16	1x1	double
iter inv	40	1x1	double
lambda inv	15.0000	1x1	double
max iter	1000	1x1	double
mu	10	1x1	double
tolerance	1.0000e-06	1x1	double
v1	[1:1:1]	3x1	double
v2	[3:10:4]	3x1	double