

10 Plot in MATLAB

Il comando `plot` viene utilizzato per la visualizzazione di grafici. Con

```
1 plot(x,y)
```

viene disegnato un grafico che passa per i punti le cui ascisse e ordinate sono contenute rispettivamente nei vettori `x` e `y`.

Per disegnare il grafico della parabola $y = x^2$ nell'intervallo $[-1, 1]$:

```
1 x = linspace(-1,1,100);  
2 y = x.^2;  
3 plot(x,y)
```

Si possono specificare diversi stili per il plot

```
1 plot(x,y) % linea continua blu  
2 plot(x,y,'r') % linea continua rossa  
3 plot(x,y,'b--') % linea tratteggiata blu  
4 plot(x,y,'g-.') % linea-punto verde  
5 plot(x,y,'c.') % punti azzurri
```

Per aprire una nuova figura:

```
1 figure
```

Ogni nuovo plot sovrascrive il precedente. Per disegnare più di un plot sulla stessa figura:

```
1 x = linspace(-1,1,100);  
2 figure  
3 hold on  
4 plot(x,x)  
5 plot(x,x.^2,'r')  
6 hold off
```

Per aggiungere un titolo alla figura

```
1 title('Due plot')
```

È possibile utilizzare dei marker oltre alle linee nei plot

```
1 plot(x, x.^2, '-o')
2 plot(x, x, 'rx')
```

Si possono disegnare anche punti singoli

```
1 x = 1;
2 y = 2;
3 plot(x, y, 'o')
```

Per una lista completa degli stili di linea, dei colori e dei marker che si possono usare

```
1 help plot
2 doc plot
```

Si possono disegnare anche semplici forme geometriche, come ad esempio dei rettangoli

```
1 x = 0;
2 y = 1 ;
3 base = 3;
4 altezza = 5;
5 rectangle('Position',[x, y, base, altezza], 'FaceColor', 'b')
```

Per ridefinire i limiti del plot

```
1 xlim([-1 5])
2 ylim([0 10])
```

10.1 Esercizi

Esercizio 10.1

Visualizzare la soluzione dell'equazione $x = \cos(x)$ nel piano cartesiano. In particolare, il grafico deve mostrare:

- La funzione $y = x$;
- La funzione $y = \cos(x)$ (sullo stesso grafico);

- *Bonus*: una linea verticale tratteggiata in corrispondenza dell'intersezione delle due curve.

Esercizio 10.2

La cicloide è la curva tracciata nell'aria da un punto situato sul bordo di una ruota in movimento. Disegnare la cicloide generata da un disco di raggio $r = 1$, sapendo che la curva è descritta dalle seguenti equazioni:

$$\begin{aligned}x &= r * (t - \sin(t)), \\y &= r * (1 - \cos(t)),\end{aligned}$$

dove t è un parametro che varia, ad esempio, da 0 a 10.

Esercizio 10.3

Disegnare la superficie:

$$f(x, y) = \frac{1}{2\pi} e^{-x^2 - y^2}$$

sul dominio $\Omega = [-5; 5] \times [-4; 6]$ usando 100 punti per le x e 50 per le y . Dare inoltre i nomi agli assi e un titolo al grafico.

Successivamente disegnare anche un cerchio di raggio 1 situato interamente su questa superficie.

Esercizio 10.4

(TdE 2010 - modificato) Dopo una gara automobilistica si ha come risultato una matrice le cui righe rappresentano gli n partecipanti (numerati da 1 a n) e le colonne gli m giri di pista effettuati. Il valore di ogni generico elemento (i, j) della tabella rappresenta il tempo impiegato (in secondi) dal partecipante i per percorrere il giro j .

Si scrivano le istruzioni per:

- creare una tabella di valori casuali che contenga i tempi sul giro di ogni partecipante; si noti che, seppur espressi in secondi, i tempi sul giro sono accurati al millesimo di secondo; inoltre, nessun tempo risulta essere maggiore di 3 minuti netti; si assuma $n = 10$ e $m = 30$.
- calcolare il tempo medio che è stato impiegato da ciascun partecipante per completare la gara;
- determinare il vincitore della gara (cioè il numero del partecipante il cui tempo di percorrenza totale è minore di quello degli altri partecipanti);
- disegnare un grafico in cui l'asse delle x rappresenta i partecipanti. Tracciare quindi

- una linea continua nera indicante i tempi medi (in secondi) sul giro per ciascun partecipante;
- una linea tratteggiata rossa indicante i tempi minimi (in secondi) sul giro per ciascun partecipante;
- una linea tratteggiata verde indicante i tempi massimi (in secondi) sul giro per ciascun partecipante;
- un asterisco blu in corrispondenza del tempo medio sul giro ottenuto dal vincitore;

Esercizio 10.5

Si costruisca una matrice in *MATLAB* per la memorizzazione delle precipitazioni atmosferiche registrate da una stazione meteorologica. La matrice deve avere 4 righe e n colonne, dove n indica il numero dei giorni monitorati (tutti i giorni degli anni dal 2012 al 2014). Ogni colonna fa riferimento ad un determinato giorno: i primi tre elementi di ciascuna colonna indicano rispettivamente il giorno, il mese e l'anno, mentre l'ultimo elemento rappresenta il valore in mm della quantità di pioggia caduta, come intero casuale nell'intervallo $[0, 100]$.

Si tracci un grafico in cui l'asse delle ascisse rappresenta i mesi del 2014. In corrispondenza di ogni mese tracciare un segmento verticale le cui coordinate sono espresse dai valori massimo e minimo delle precipitazioni in quel mese. Si mostri inoltre con un asterisco il valore medio delle precipitazioni di ciascun mese.

Esercizio 10.6

Si prepari un programma *MATLAB* in grado di disegnare la configurazione iniziale di una partita di battaglia navale. Si assuma che il piano di battaglia sia una griglia 10×10 e che la disposizione delle navi venga fornita attraverso 4 vettori contententi rispettivamente: l'ascissa e l'ordinata della prua della nave, e l'ascissa e l'ordinata della poppa.

Il piano di battaglia deve essere disegnato mediante la costruzione di una griglia, mentre le navi verranno rappresentate come rettangoli rossi. Le ascisse delle coordinate verranno indicate con le lettere maiuscole dalla A alla J, mentre le ordinate con numeri da 1 a 10.

10.1.1 Strutture e salvataggio su file

Esercizio 10.7

Scrivere uno script MATLAB che gestisca un sistema di raccomandazione di ristoranti. Lo script permette all'utente di inserire un nuovo ristorante: ciascun ristorante ha un nome, un indice di qualità (da 1 a 5) e un flag che specifica se il ristorante è economico. Inoltre, il sistema permette all'utente di visualizzare l'elenco dei nomi dei ristoranti economici che hanno un indice di qualità maggiore o uguale della media di tutti i ristoranti memorizzati.

Esercizio 10.8

Scrivere uno script MATLAB per svolgere le seguenti operazioni:

- caricare dal file *incassi.mat* la struttura *botteghino*, avente campi *titolo* e *incasso*;
- inserire i dati relativi ad un nuovo film nella struttura;
- salvare sul file *incassi.mat* i dati contenuti nella struttura.

Soluzioni

Soluzione dell'esercizio 10.1

```
1 close all
2 clear
3 clc
4
5 % costruisco il dominio
6 x = linspace(0, 1, 100);
7
8 % disegno i grafici
9 figure();
10 hold on;
11 % plot della funzione y = x
12 plot(x, x, 'b');
13 % plot della funzione y = cos(x)
14 plot(x, cos(x), 'r');
15
16 % cerco ascissa più vicina all'intersezione
17 diff = abs(x - cos(x));
18 [~, pos] = min(diff);
19 % disegno una linea verticale sull'intersezione
20 plot([x(pos), x(pos)], [1, 0], 'k--');
21
22 xlabel('x');
23 legend('x', 'cos(x)');
24 title('coseno')
25 hold off;
```

Soluzione dell'esercizio 10.2

```
1 close all
2 clear
3 clc
4
5 % intervallo dei tempi
6 t = 0: 0.1: 10;
7
8 % costruisco i vettori per x e y
9 x = t - sin(t);
10 y = 1 - cos(t);
11
```

```
12 % disegno la funzione
13 plot(x, y);
14 xlabel('x');
15 ylabel('y');
16 title('cicloide');
```

Soluzione dell'esercizio 10.3

```
1 close all
2 clear
3 clc
4
5 % costruisco il dominio
6 x = linspace(-5, 5, 100);
7 y = linspace(-4, 6, 50);
8 [xx, yy] = meshgrid(x, y);
9
10 % funzione
11 f = @(x, y) (1/(2*pi) * exp(- x.^2 - y.^2));
12
13 % disegno il grafico della superficie
14 figure();
15 grafico1 = surf(xx, yy, f(xx, yy));
16 xlabel('ascisse');
17 ylabel('ordinate');
18 title('campana');
19 set(grafico1, 'EdgeColor', 'none');
20
21 % disegno la circonferenza
22 hold on;
23 t = linspace(0, 2*pi, 100);
24 grafico2 = plot3(cos(t), sin(t), f(cos(t), sin(t)), 'r');
25 set(grafico2, 'LineWidth', 3);
```

Soluzione dell'esercizio 10.4

```
1 clear
2 clc
3 close all
4
5 n_piloti = 10;
6 n_giri = 30;
7
```

```

8 minuti = randi(2,n_piloti,n_giri);
9 secondi = randi(60,n_piloti,n_giri) - 1;
10 millesimi = randi(1000,n_piloti,n_giri) - 1;
11
12 tempi = minuti * 60 + secondi + millesimi / 1000;
13 tempi_medi = mean(tempi, 2);
14
15 [tempo_vinc, vinc] = min(tempi_medi);
16 disp(['Il vincitore e ' ' num2str(vinc)]);
17
18 tempi_minimi = min(tempi, [], 2);
19 tempi_massimi = max(tempi, [], 2);
20
21 figure
22 hold on
23 title('Tempi ottenuti da ciascun partecipante')
24 piloti = 1:n_piloti;
25 piloti = piloti';
26 plot(piloti,tempi_medi,'k');
27 plot(piloti,tempi_minimi,'r--');
28 plot(piloti,tempi_massimi,'g--');
29 plot(vinc,tempi_medi(vinc),'b*');
30 grid on

```

Soluzione dell'esercizio 10.5

```

1 clear
2 clc
3 close all
4
5 n_giorni = 366+365*2;
6 prec = zeros(4,n_giorni);
7
8 c = 0;
9 for anno=2012:2014
10     for mese=1:12
11         switch mese
12             case {1,3,5,7,8,10,12}
13                 giorni_mese = 31;
14             case 2
15                 if(mod(anno,4) == 0)
16                     giorni_mese = 29;
17             else

```

```
18         giorni_mese = 28;
19     end
20     otherwise
21         giorni_mese = 30;
22     end
23     for giorno=1:giorni_mese
24         c = c + 1;
25         prec(1,c) = giorno;
26         prec(2,c) = mese;
27         prec(3,c) = anno;
28         prec(4,c) = randi(101)-1;
29     end
30 end
31 end
32
33 anno = 2014;
34
35 prec_min = zeros(1,12);
36 prec_max = zeros(1,12);
37 prec_medie = zeros(1,12);
38 for mese=1:12
39     colonne = (prec(3,:) == 2014) & (prec(2,:) == mese);
40     prec_min(mese) = min(prec(4,colonne));
41     prec_max(mese) = max(prec(4,colonne));
42     prec_medie(mese) = mean(prec(4,colonne));
43 end
44
45 figure
46 hold on
47 for mese=1:12
48     plot([mese mese], [prec_min(mese) prec_max(mese)], 'b');
49     plot(mese, prec_medie(mese), 'r*');
50 end
```

Soluzione dell'esercizio 10.6

```
1 clear
2 clc
3
4 figure
5 hold on
6
7 for x=0:10
```

```

8     plot([x x],[0 10], 'k');
9 end
10 for y=0:10
11     plot([0 10], [y y], 'k');
12 end
13
14 ascisse_prua    = ['A', 'G', 'E'];
15 ordinate_prua  = [ 3 ,  8 ,  2 ];
16 ascisse_poppa  = ['A', 'I', 'E'];
17 ordinate_poppa = [ 4 ,  8 ,  6 ];
18
19
20 n_navi = length(ascisse_prua);
21
22 for nave=1:n_navi
23     x_prua = ascisse_prua(nave) - 'A';
24     y_prua = ordinate_prua(nave) - 1;
25     x_poppa = ascisse_poppa(nave) - 'A';
26     y_poppa = ordinate_poppa(nave) - 1;
27
28     base = abs(x_prua-x_poppa) + 0.8;
29     altezza = abs(y_prua-y_poppa) +0.8;
30     x_start = min(x_prua,x_poppa) + 0.1;
31     y_start = min(y_prua,y_poppa) + 0.1;
32     rectangle('Position',[x_start y_start base altezza], '
        FaceColor','r');
33 end
34
35 title('Battaglia navale')
36
37 set(gca,'XTick',[0.5:1:9.5])
38 set(gca,'XTickLabel',['A' 'B' 'C' 'D' 'E' 'F' 'G' 'H' 'I' 'J'
    ])
39
40 set(gca,'YTick',[0.5:1:9.5])
41 set(gca,'YTickLabel',[1:1:10]')

```

Soluzione dell'esercizio 10.7

```

1 clear
2 clc
3
4 % preparo struttura ristoranti

```

```
5 ristoranti.nome = 'Primo';
6 ristoranti.indice = 3;
7 ristoranti.economico = true;
8
9 % stampo il menu sulla command window
10 fprintf('--- GESTIONE RISTORANTI ---\n\n');
11 fprintf('[i]: inserire un nuovo ristorante\n');
12 fprintf('[s]: stampa ristoranti economici\n');
13 fprintf('[q]: termina\n\n');
14
15 menu = input('inereire un comando: ', 's');
16 while menu ~= 'q'
17     switch menu
18         case 'i'
19             % inserisco i campi del nuovo ristorante
20             fprintf('\n')
21             ristoranti(end + 1).nome = input('inserire nome: ',
22                 's');
23             ristoranti(end).indice = input('inserire indice: ')
24             ;
25             ristoranti(end).economico = input('inserire
26                 economico: ');
27         case 's'
28             % calcolo la media dell'indice di qualità
29             media = mean([ristoranti.indice]);
30             % scorro i ristoranti e stampo quelli economici
31             fprintf('\n')
32             for ii = ristoranti
33                 if ii.economico && ii.indice > media
34                     str_out = ['nome: ' ii.nome ...
35                         ' qualità: ' num2str(ii.indice) '/5'];
36                     disp(str_out);
37                 end
38             end
39         end
40     end
41
42     % stampo il menu sulla command window
43     fprintf('\n--- GESTIONE RISTORANTI ---\n\n');
44     fprintf('[i]: inserire un nuovo ristorante\n');
45     fprintf('[s]: stampa ristoranti economici\n');
46     fprintf('[q]: termina\n\n');
47
48     menu = input('inereire un comando: ', 's');
49 end
```

Soluzione dell'esercizio 10.8

```
1 clear
2 clc
3
4 % inserisco titolo e incasso del film
5 titolo = input('inserire titolo del film: ', 's');
6 incasso = input('inserire incasso del film: ');
7
8 % carico il contenuto da file
9 load('incassi');
10 % alternativa
11 load incassi
12 % se voglio caricare solo botteghino
13 load('incassi', 'botteghino');
14 % alternativa
15 load incassi botteghino
16
17 % inserisco il nuovo film in botteghino
18 botteghino(end + 1).titolo = titolo;
19 botteghino(end).incasso = incasso;
20
21 % salvo la struttura botteghino su file
22 save('incassi', 'botteghino');
23 % alternativa
24 save incassi botteghino
```