

Introduzione all'ambiente MATLAB

*Utilizzo di Matlab ed operazioni fondamentali.
Primi rudimenti di grafica.*

MATLAB = MATrix LABoratory

è un ambiente di programmazione orientato al calcolo scientifico.

- ha una buona potenzialità grafica,
- esistono versioni per Unix/Linux, Windows, Mac,
- i files sono portabili da una piattaforma all'altra.

MATLAB: www.mathworks.com

Link alla licenza unibs:

[https://www.unibs.it/servizi-online/servizi-line/
altri-servizi-informatici/software/matlab](https://www.unibs.it/servizi-online/servizi-line/altri-servizi-informatici/software/matlab)

MATLAB ha **funzioni intrinseche** molto potenti
(es: risoluzione di sistemi lineari, calcolo di autovalori e autovettori di una matrice)
e sono completati con **toolbox** (MATLAB), ovvero librerie di software mirato a particolari problematiche

Prompt di Matlab:

>>

Definizione e disegno di una funzione

Problema 0: disegnare il grafico di $f(x) = \tan(x)$ sull'intervallo $[-2\pi, 2\pi]$ e la sua inversa $f^{-1}(x) = \arctan(x)$.

```
% definisco la funzione con il function handle  
f=@(x)tan(x);  
% apro una finestra grafica (che ha numero 1)  
figure(1)  
% fplot(f,[a,b]), [a,b] intervallo di definizione  
fplot(f,[-2*pi,2*pi])  
% pi e' una variabile predefinita di matlab  
xlabel('x') % aggiungo label all'asse x  
ylabel('y') % aggiungo label all'asse y  
legend('f(x)=tan(x)') % aggiungo la legenda  
grid on % griglia
```

Ora evidenzio di rosso il ramo della funzione quando

$$x \in [-\pi/2, \pi/2]$$

```
hold on % dico a matlab di mantenere tutto cio'
        % che ho disegnato prima
fplot(f, [-pi/2, pi/2], 'r') % 'r' sta per red
% vedo che e' cambiato il range in y
% per riportarlo ai valori di prima (circa)
% axis([xmin, xmax, ymin, ymax])
axis([-pi/2, pi/2, -8, 8])
% definisco la funzione inversa (arctangente)
finv=@(x) atan(x);
fplot(finv, [-2*pi, 2*pi], 'g') % e la disegno, g=green
% definisco la bisettrice del primo e terzo quadrante
b=@(x) x;
fplot(b, [-2*pi, 2*pi], 'k--') % k=black
```

Problema 1: disegnare $f(x) = \sin(x^2)$ sull'intervallo $I = [-2\pi, 2\pi]$.

Lavoriamo come prima:

```
% definisco la funzione con il function handle  
f=@(x) sin(x^2);  
% apro una finestra grafica (che ha numero 1)  
figure(1)  
% fplot(f,[a,b]), [a,b] intervallo di definizione  
fplot(f,[-2*pi,2*pi])
```

Matlab si lamenta un po'....

Vediamo come procedere correttamente, dobbiamo parlare di **array**

Assegnazione di variabili scalari

```
>>a=1.54
```

- a nome della variabile (max 31 caratteri alfanumerici, il primo dei quali non deve essere un numero),
- 1.54 valore numerico assegnato alla variabile,
- Di default lettere maiuscole e minuscole sono considerate diverse sia nei comandi che nei nomi delle variabili.

Il comando

```
>> a=1.54
```



produce

```
a =
```

```
1.5400
```

```
>> a=1.54;
```



non produce risposta

```
>> 1.67
```



produce

```
ans =
```

```
1.6700
```

ans è il nome della variabile di default.

per visualizzare il contenuto della variabile a

```
>> a
```



produce

```
a =
```

```
1.5400
```

per poter spezzare un'istruzione troppo lunga:
tre punti in sequenza

```
>> b=1+1/2+5/3+1/4+23/6+...  
2/9+1/10;
```


Operazioni aritmetiche

^ potenza

* prodotto

/ divisione

+ somma

- differenza

Es: per calcolare $x = \frac{3 + 5^3 - 2/3}{4(5 + 2^4)}$ il comando da dare è:

```
>> x = (3+5^3-2/3)/(4*(5+2^4))
```

- Sono osservate le precedenze classiche dell'aritmetica
- Per alterare le precedenze si utilizzano esclusivamente le parentesi **tonde**

Variabili predefinite

`pi` π

`i, j` $\sqrt{-1}$ unità immaginaria

`NaN` not a number

`eps` $2.2204e-16$ precisione di macchina

Il contenuto di queste variabili può essere variato con una semplice operazione di assegnazione:

```
>> pi = 18  
pi =  
    18
```

Per riassegnare alla variabile `pi` il valore π :

```
>> clear pi  
>> pi  
ans =  
    3.1416
```

Per cancellare il contenuto di tutte le variabili:

```
>> clear
```

Assegnazione di array

```
>> a=[1 2 3 4];  
>> a=[1,2,3,4];  
>> a=(1:4);
```

*Modi equivalenti per generare un array
1x4, 1 riga e 4 colonne, vettore riga*

```
>> a  
a =  
     1     2     3     4
```

```
>> b=[2;4;1;-2]  
b =  
     2  
     4  
     1  
    -2
```

*Per generare un array 4x1, 4 righe e 1
colonna, vettore colonna*

```
>> c=[5 3 4; 2 4 -2]
```

```
c =
```

```
    5     3     4
    2     4    -2
```

Per generare un array 2x3, matrice 2 righe e 3 colonne

Lo spazio o la virgola separano elementi sulla stessa riga. Il punto e virgola separa le righe.

Operazione di trasposizione:

```
>> a'
```

```
ans =
```

```
    1
    2
    3
    4
```

Il vettore trasposto di a viene memorizzato nella variabile ans

```
>> d=a'
```

Il vettore trasposto di a viene memorizzato nella variabile a1

Analogo discorso vale per la trasposizione di matrici:

```
>> e=c'
```

```
e =
```

```
    5    2  
    3    4  
    4   -2
```

```
>> whos
```

Name	Size	Bytes	Class
a	1x4	32	double array
ans	4x1	32	double array
b	4x1	32	double array
c	2x3	48	double array
d	4x1	32	double array
e	3x2	48	double array

Operazioni su array

- + somma di vettori o matrici (elemento per elemento)
- differenza di vettori o matrici (elemento per elemento)
- * prodotto tra vettori e/o matrici (righe per colonne)

Sono le operazioni dell'algebra lineare; quindi:

- per somma e differenza: gli operandi devono avere le stesse dimensioni
- per il prodotto: la dimensione interna dei due array deve coincidere.

```
>> f=d+b
```

```
f =
```

```
3
```

```
6
```

```
4
```

```
2
```

$$f_i = d_i + b_i$$

$$\text{con } d = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \text{ e } b = \begin{bmatrix} 2 \\ 4 \\ 1 \\ -2 \end{bmatrix}$$

OK, d e b sono entrambi vettori colonna (4x1)

```
>> g=a-b
```

a=vettore riga (1x4)

b=vettore colonna (4x1)

L'OPERAZIONE NON HA SENSO IN ALGEBRA LINEARE, MA
MATLAB LA SVOLGE, espandendo i vettori a matrici e facendo la
somma di matrici

Prodotto scalare tra vettori

Se $a \in \mathbb{R}^{1 \times n}$ e $b \in \mathbb{R}^{n \times 1}$

$$a \cdot b = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

Dimensioni: $(1 \times n)(n \times 1) \rightarrow (1 \times 1)$

quindi il risultato è uno scalare

```
>> a*b
```

```
ans =  
    6
```

$$a = [1, 2, 3, 4], \quad b = \begin{bmatrix} 2 \\ 4 \\ 1 \\ -2 \end{bmatrix}$$

(1x4)(4x1) -prodotto scalare- OK

```
>> a*a
```

```
Error using *
```

```
Inner matrix dimensions must agree.
```

(1x4)(1x4) -prodotto non possibile

Prodotto di matrici

Ricordiamo che se $A \in \mathbb{R}^{n \times m}$ e $B \in \mathbb{R}^{p \times q}$, il prodotto AB è possibile solo se $m = p$ e che se $A \in \mathbb{R}^{n \times m}$ e $B \in \mathbb{R}^{m \times q}$, allora $C \in \mathbb{R}^{n \times q}$ ($(n \times \cancel{m})(\cancel{m} \times q) \rightarrow (n \times q)$) e

$$C_{ij} = \sum_{k=1}^m A_{ik} B_{kj}, \quad i = 1, \dots, n, \quad j = 1, \dots, q$$

l'operazione $*$ realizza il prodotto tra matrici:

```
A=[2 3 4; 1 -2 1]; % (2x3)
B=[3 1 4; 2 -1 0; 2 7 -1]; % (3x3)
C=A*B % (2x3) (3x3) --> (2,3) OK
```

C =

```
20    27    4
 1    10    3
```

```
>> B*A
?? Error using ==> *
Inner matrix dimensions must agree.
```

(3x3)(2x3) - prodotto non possibile-

Operazioni punto

Esistono poi le operazioni "punto" che agiscono su array che abbiano le stesse dimensioni:

. * prodotto elemento per elemento

. / divisione elemento per elemento

. ^ potenza elemento per elemento

```
>> b2=b .* b
```

$$(b2)_i = b_i * b_i$$

```
b2 =
```

```
    4  
   16  
    1  
    4
```

essendo $b = \begin{bmatrix} 2 \\ 4 \\ 1 \\ -2 \end{bmatrix}$

Avrei ottenuto lo stesso risultato con

```
>> b2=b .^2
```

Semplici comandi sulle matrici

```
[n,m]=size(A) % restituisce il n. di righe
                % e colonne di A
det(A)          % calcola il determinante di A (n x n)
rank(A)        % calcola il rango di A
inv(A)         % calcola l'inversa di A (n x n)
eig(A)         % calcola gli autovalori di A (n x n)
x=A\b          % risolve il sistema lineare
                % A x = b, dove A (n x n), b(n x 1)
% attenzione a non confondere
% / (slash) con \ (backslash)
```

Funzioni matematiche e grafica

Problema 1: disegnare $f(x) = \sin(x^2)$ sull'intervallo $I = [-2\pi, 2\pi]$

```
% definisco la funzione con il function handle  
f=@(x) sin(x.^2);  
% apro una finestra grafica (che ha numero 1)  
figure(1)  
% fplot(f,[a,b]), [a,b] intervallo di definizione  
fplot(f,[-2*pi,2*pi])  
xlabel('x') % aggiungo label all'asse x  
ylabel('y') % aggiungo label all'asse y  
legend('f(x)=sin(x^2)') % aggiungo la legenda  
grid on % griglia  
axis equal % uso stessa scala in x e y
```

Problema 2: valutare $f(x) = \sin(x^2)$ in 100 punti equispaziati $x_i \in [-2\pi, 2\pi]$ con $i = 1, \dots, 100$ e disegnare la funzione utilizzando i punti $(x_i, f(x_i))$.

```
>> x=linspace(-2*pi,2*pi,100);  
% x=linspace(a,b,n) crea un vettore riga di n elem,  
% contenenti le ascisse di n punti equispaziati  
% sull'intervallo chiuso [a,b]  
  
>> f=@(x) sin(x.^2);  
% definisco f  
>> y=f(x);  
% valuto f  
  
>> figure(2) % apro una finestra grafica  
>> plot(x,y) % matlab disegna la spezzata che  
% congiunge i punti (x_i,y_i)
```

La sintassi del comando plot è:

```
plot(x,y, 'color_linestyle_marker')
```

```
>> plot(x,y, 'm-*')
```

color: c,m,y,r,b,g,w,k

linestyle: -,--, :,-.,none

marker: +,o,*,.,x,s

Per disegnare 2 o più coppie di vettori sullo stesso grafico, ad esempio $f(x) = \sin(x^2)$ e $g(x) = (\sin(x))^2$:

```
>> g=@(x)(sin(x)).^2;
```

```
>> yg=g(x);
```

```
>> plot(x,y, 'b-',x,yg, 'r--');
```

bisogna ripetere: 'ascisse, ordinate, specifiche' per ogni coppia di vettori. Le specifiche sono opzionali.

Per conoscere nel dettaglio tutte le opzioni di un comando, oppure se non ci si ricorda la sintassi del comando:

```
help nome_comando
```

```
>> help plot
```

Se non ci si ricorda il nome del comando, ma si vuole fare una ricerca per *parola_chiave* (in inglese), oppure se si cercano tutti i comandi che facciano riferimento ad una *parola_chiave*:

```
lookfor parola_chiave
```

```
>> lookfor plot
```


Funzioni matematiche intrinseche

<code>sqrt(x)</code>	\sqrt{x}
<code>round(x)</code>	arrotondamento: <code>round(3.6)=4</code>
<code>fix(x)</code>	parte intera: <code>fix(3.6)=3</code>
<code>sign(x)</code>	segno di x (vale -1, 0 o 1)
<code>sin(x), cos(x), tan(x)</code>	<code>sin(x), cos(x), tan(x)</code>
<code>sinh(x), cosh(x), tanh(x)</code>	<code>sinh(x), cosh(x), tanh(x)</code>
<code>asin(x), acos(x), atan(x)</code>	<code>asin(x), acos(x), atan(x)</code>
<code>exp(x), log(x), log10(x)</code>	$e^x, \log_e(x), \log_{10}(x)$

Per z complesso:

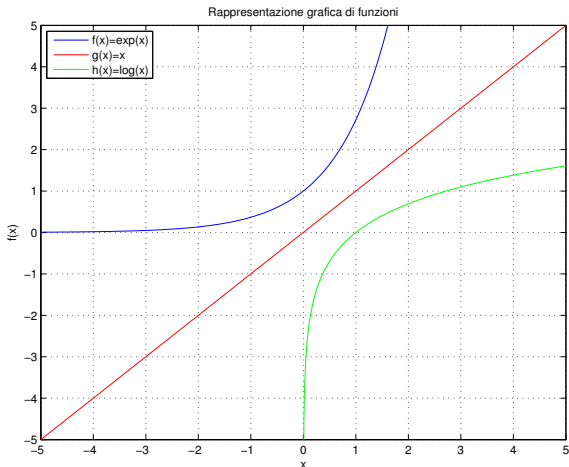
```
>> z=3+i*4
```

<code>real(z)</code>	parte reale di z
<code>imag(z)</code>	parte immaginaria di z
<code>conj(z)</code>	complesso coniugato di z
<code>abs(z)</code>	modulo di z

Generazione di un m-file

Problema 2: Creazione di un grafico 2D.

Disegnare $f(x) = \exp(x)$, $g(x) = x$, $h(x) = \log(x)$ sull'intervallo $I = [-5, 5]$.



Dal menù **Home** selezionare **New Script**. Si apre una finestra di **Editor** in cui si possono scrivere i comandi matlab (non compare più il prompt)

```
f=@(x) exp(x);  
figure(1); clf  
fplot(f, [-5,5], 'b')  
xlabel('x'); ylabel('y')  
title('Rappresentazione grafica di funzioni')  
hold on % mantiene il grafico fatto e  
g=@(x)x;  
fplot(g, [-5,5], 'r')  
h=@(x) log(x);  
fplot(h, [0.00001,5], 'g')  
l=legend('f(x)=exp(x)', 'g(x)=x', 'h(x)=log(x)');  
set(l, 'Location', 'Northwest')  
grid on % disegna la griglia  
axis([-5,5,-5,5]) % fissa il box della figura  
hold off %
```

- Per salvare il contenuto del file: dal menù dell'Editor selezionare **Save as**.
Specificare il direttorio in cui salvare (es: `c:\tmp` o `e:\`) ed il nome per il file (es: `dis2d.m`)
N.B. L'estensione dei file matlab è sempre **m**.
- Dalla finestra dei comandi matlab:
`>> addpath c:\tmp` oppure
`>> addpath e:\`
per dire di cercare il file in tale direttorio, quindi richiamare il file generato, dando il nome del file stesso:
`>> dis2d`

Matlab segnala errori?

- 1 Leggere il tipo di errore
- 2 Tornare nell'editor, cercare l'errore e modificare il file
- 3 Salvare le modifiche effettuate
- 4 Tornare alla finestra dei comandi Matlab e ridare il comando

```
>> dis2d
```