

Introduzione al linguaggio Matlab - 1

L'interfaccia Matlab é la seguente:

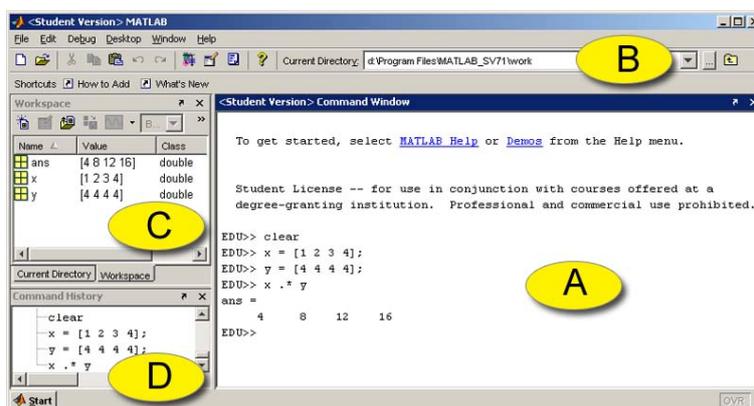


Figure 1: Interfaccia utente Matlab

- A) **Command Window**: é la finestra in cui si eseguono in comandi.
- B) **Current Directory**: indica l'indirizzo di memoria del sistema sul quale stiamo lavorando.
- C) **Workspace**: sono visualizzati i nomi dei variabili, i loro tipi e i loro valori.
- D) **Command History**: le istruzioni eseguite in precedenza ed eventualmente richiamabili.

Per poter eseguire un comando, esso va digitato nella *Command Window*, dopo il simbolo ">>", se il doppio maggiore é visualizzato significa che Matlab é pronto ad eseguire l'operazione.

Andiamo ora a vedere le prime semplici operazioni, se vogliamo eseguire delle operazioni algebriche é sufficiente digitarle con la classica sintassi aritmetica e premere *Invio*:

```
>> 7*3.5-2.1
ans =
    224
```

se l'operazione é stata eseguita correttamente, verrà visualizzato "ans =" seguito dal risultato dell'operazione. Se al comando viene preceduto un nome, quello sarà assegnato alla variabile:

```
>> b=7*3.5-2.1
b =
    224
```

Osservazione: Se si vuole sopprimere l'output a schermo bisogna mettere il ";" alla fine dell'espressione, in questa maniera Matlab salverá il risultato dell'operazione nel *Workspace* senza mostrarlo sulla *Command Window*.

Osservazione: Matlab é *case sensitive*, fa quindi differenza tra le maiuscole e le minuscole.

Nomi riservati e formati: in Matlab alcuni nomi sono riservati, ovvero associati a valori o operazioni predefinite, il che significa che non possono essere utilizzati nelle operazioni a piacimento dell'utente, alcuni esempi

```
>> pi
ans =
    3.1415
```

```
>> eps
ans =
    2.22..... (precisione di macchina)
```

É possibile poi scegliere i formati numerici (precisione che in default é a 5 cifre decimali) con i quali effettuare i calcoli, per scegliere il formato desiderato basta scrivere il formato preceduto dalla parola riservata "*format*", alcuni esempi

```
>> format short
>> format long
```

In Matlab l'elevamento a potenza si effettua con il comando "cappuccio", ad esempio se si vuole calcolare 2^3 :

```
>> 2^3
ans =
      8    (elevamento a potenza)
```

Il comando *clear* elimina tutte le variabili dal *Workspace*, mentre é possibile eliminare solo alcune variabili facendo seguire *clear* dai nomi delle variabili stesse.

Calcoli con variabili: di seguito alcuni esempi di calcolo con variabili che utilizzano funzioni matematiche predefinite:

```
>> y=cos(pi/3);
>> z=sin(pi/3);
>> a=log(y);
>> b=log(z);
```

Osservazione: log = ln.

Di uso fondamentale in Matlab é il comando di HELP in linea, digitando *help* seguito da una parola riservata si avrá visualizzato a schermo il funzionamento di quel comando:

```
>> help log

LOG natural .....
.....
```

Matrici e vettori: Ricordiamo che é buona norma digitare prima di qualunque sessione di lavoro il comando *clear*, per "ripulire" il *Workspace* da tutte le variabili già in uso.

Vediamo come si definiscono e si stampano matrici:

```
>> A = [ 1 2 3 ; 2 3 4 ; 3 4 5];
```

con lo "spazio" si dividono le varie entrate di ogni riga della matrice, mentre con il ";" si dividono le righe stesse:

```
>> A = [ 1 2 3 ; 2 3 4 ; 3 4 5]
```

```
A =
     1 2 3
     2 3 4
     3 4 5
```

Il comando "' ' " crea la trasposta di una matrice: se

$$B = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 2 & 1 \end{pmatrix}$$

allora

```
>> D=B'
```

```
D =
     1 4
     2 2
     3 1
```

Osservazione: Se durante la sessione di lavoro viene utilizzato un nome già in uso per definire una nuova variabile, allora Matlab sovrascriverá il nuovo valore sulla vecchia variabile.

Calcoli con matrici e vettori: Somma e sottrazione tra matrici

Il comandi per l'algebra tra matrici sono:

```
>> C = A+B      (somma tra matrici)
>> D = 3*A      (prodotto scalare-matrice)
>> D = 3.12*A + 4.5*B (combinazione lineare tra matrici)
```

Osservazione: I calcoli devono essere CONSISTENTI con l'algebra lineare delle matrici.

```
>> A*B          (prodotto righe per colonne)
>> A*b          (prodotto matrice-vettore)
>> D = 3.12*A + 4.5*B (combinazione lineare tra matrici)
```

Andiamo a vedere come si calcola il prodotto scalare tra vettori u e v :

```
>> u=.... ;
    v=.....;

>> u*v'        (prodotto scalare)
```

Osservazione: $u' * v$ da come risultato una matrice.

Calcoliamo ora la *norma* di un vettore u

```
>> u*u';
>> sqrt(u*u');
ans =
    .....norma di u
```

Il comando *norm* calcola questo risultato in una sola operazione

```
>> norm(u) ;
>> sqrt(u*u');
ans =
    norma euclidea
```

In Matlab é possibile effettuare il cosiddetto *prodotto di Hadamard* che non é altro che un operatore puntuale elemento per elemento di una matrice, l'operatore é il punto ".".

```
>> C=A.*B ;
>> D=A./B;
>> E=A.^B; (elevamento a potenza)
```

Osservazione: $A \pm B$ non esiste.

Matrici particolari: In generale una matrice A quadrata di ordine n in Matlab é definita con il comando relativo al tipo di matrice seguito da (n) nel caso di una matrice $n \times m$ il comando é come il precedente ma seguito da (n, m)

```
>> A=eye(3)
A =
    1 0 0
    0 1 0
    0 0 1 (la matrice unitaria)
```

```
>> A=eye(3,5)
A =
    1 0 0 0 0
    0 1 0 0 0
    0 0 1 0 0 (la matrice unitaria completata)
```

Altri esempi di matrici:

```
>> A = ones(5) (matrice 5x5 formata da soli zeri)
>> A = zeros(5) (matrice 5x5 formata da soli uno)
```

Matrici rettangolari:

```
>> A = ones(3,4) (matrice 3x4 formata da soli zeri)
>> A = zeros(3,2) (matrice 3x2 formata da soli uno)
>> A = rand(3)
A =
    ..... (matrice 3x3 con valori nell'intervallo [0,1])
```

Esercizio: studiare il comando

```
>> A = magic(3)
```

Esercizio: costruire la matrice C di dimensione (9×9) così fatta:

$$C = \begin{pmatrix} 12 & 1 & \dots & 1 \\ 1 & 12 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 1 & \dots & 1 & 12 \end{pmatrix}$$

```
>> A = ones(9);
```

```
>> B = eye(8);
```

```
>> C = A+11*B
```

```
C =
```

```
.....
```