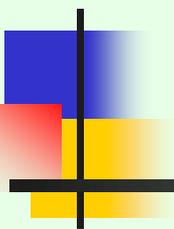


Teoria e Metodi Matematici per la Finanza

Prof.ssa Elisabetta Michetti

elisabetta.michetti@unimc.it



PARTE 1

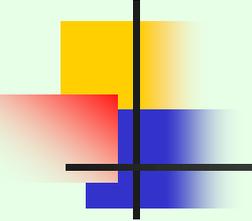
MatLab Base: Numeri, Vettori, Matrici



MatLab = MATrix LABoratory
(ambiente per il calcolo scientifico
e le simulazioni numeriche)

Desktop di MatLab:

- **Command Window:** finestra dei comandi (vengono indicati i comandi da eseguire ed il simbolo `>>` indica il **prompt** dei comandi)
- **Command History:** sequenza cronologica dei comandi eseguiti
- **Workspace:** spazio di lavoro (memoria operativa che contiene l'insieme delle variabili-**array** in memoria)
- **Current Directory:** elenco di tutti i file e le cartelle presenti nella directory corrente



Vettori

Le **variabili-array** in MatLab possono consistere in sequenze di caratteri alfanumerici, possono essere un numero, o una tabella di numeri o anche **VETTORI**.

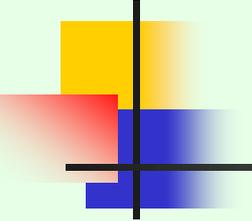
Un **VETTORE** di n elementi è una ennupla ordinata di numeri reali

I suoi **elementi** possono essere disposti in riga separati da una virgola (**vettore riga**) o in colonna (**vettore colonna**) e racchiusi fra parentesi quadre o tonde

ES: $x=[1, 3, -4, 11]$ è un vettore riga di 4 elementi

$y= \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$ è un vettore colonna di 3 elementi

$z=[133]$ è un vettore riga o colonna di un elemento



Memorizzare in MatLab un numero

I **COMANDI** vengono digitati nel **prompt** di MatLab. E' possibile, ad esempio, memorizzare un numero assegnandogli un nome.

L'**ASSEGNAZIONE** di un nome ad una variabile avviene attraverso il simbolo =

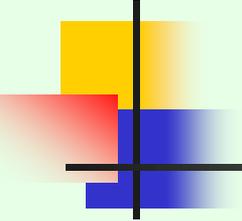
Es: memorizzare il numero 25 ed assegnargli un nome (o etichetta)

```
>> a=25
```

Dopo questo comando la variabile è stata memorizzata nel Workspace

NB: MatLab è *case sensitive*

Oss: Nella Command Window MatLab visualizza il risultato prodotto dal comando, per evitare la visualizzazione il comando può essere seguito da ;

**Esercizio 1**

Memorizzare in MatLab i seguenti numeri:

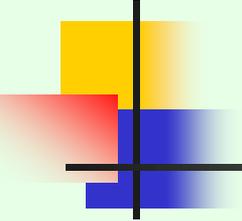
$$A = \frac{12}{5}, B = 3^6, C = 2,1 \cdot 7$$

Oss: la simbologia per le operazioni fra numeri è la consueta

Le variabili memorizzate nel Workspace possono ora essere utilizzate per compiere operazioni: basta richiamarle utilizzando il loro nome.

Si compiano le seguenti operazioni:

$$a = A + \frac{B}{C}, b = \frac{C - a}{B}, c = bB^A$$



Comandi iniziali

Nell'esecuzione di istruzioni, alcuni **comandi** risultano **di particolare comodità**:

>>**clc** cancella la command window

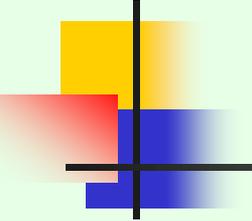
>>**clear** cancella il workspace

>> il comando ↑ richiama l'ultimo comando eseguito

>>**doc "argomento"** visualizza tutte le informazioni relative all' "argomento richiesto" interrogando il potente Help di MatLab

>>**format long** attiva il formato a 14 cifre decimali

>>**format short** riattiva il formato di default a 4 cifre decimali



Usare funzioni elementari

A volte può essere necessario utilizzare funzioni elementari

Es: memorizzare il numero e^2

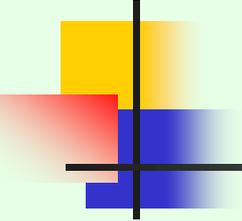
In tal caso bisogna conoscere la **sintassi** che MatLab usa per la funzione esponenziale

Bisognerà eseguire il seguente **comando**

```
>> w=exp(2)
```

Oss: l'argomento della funzione esponenziale viene indicato fra parentesi tonde

NB: il comando **>>doc elfun** permette di ottenere una lista completa delle funzioni elementari consultando l'help di MatLab



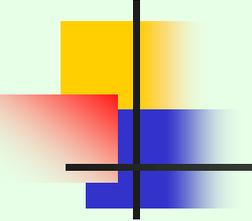
Esercizio 2

Memorizzare in MatLab i seguenti numeri:

$$x = \pi, y = \frac{1}{8}, z = -6$$

Quindi, utilizzando le funzioni elementari, **effettuare i seguenti calcoli:**

$$p = e^x - \ln 2, q = \sqrt{y} \sin z, r = |-10 + 2z|$$



Memorizzare in MatLab un vettore

VETTORE RIGA: per memorizzarlo basta elencare i suoi elementi separati da uno **spazio** o da una **virgola** e racchiusi fra parentesi quadre

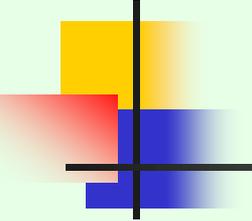
Es: memorizzare il vettore riga [10,-1,5]

```
>> A = [ 10 -1 5 ]
```

VETTORE COLONNA: per memorizzarlo basta elencare i suoi elementi separati da un **punto e virgola** e racchiusi fra parentesi quadre, oppure lo si può scrivere come vettore riga aggiungendo alla fine l'**apice** (che trasforma un vettore riga in un vettore colonna e viceversa)

Es: memorizzare il vettore colonna di elementi 3, 7, 9

```
>> B = [ 3 ; 7 ; 9 ]
```



Lavorare con i vettori

Oss: Il numero degli elementi del vettore si chiama **dimensione** del vettore (i vettori A e B degli esempi precedenti hanno dimensione 3)

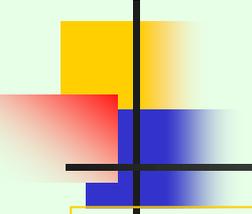
Per conoscere la **dimensione di un vettore** si usa il **comando length** seguito dal nome del vettore (fra parentesi tonde).

Es: individuare la dimensione di A

```
>> length(A)
```

Oss: ciascun elemento di un vettore è identificato da un **indice** che indica la posizione dell'elemento considerato. Nell'esempio precedente -1 è l'elemento di indice 2 del vettore A mentre 9 è l'elemento di indice 3 del vettore B.

Di conseguenza: è possibile **ACCEDERE AD UNO O PIU' ELEMENTI DI UN VETTORE** proprio specificando l'indice. Esso va indicato tra parentesi tonde dopo il nome del vettore



Lavorare con i vettori

Es: memorizzare il secondo elemento del vettore B

```
>> b=B(2)
```

Ora la variabile b assume valore 7

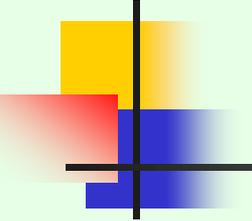
Es: creare un vettore riga di dimensione 2 avente come elementi il primo e terzo elemento di B

```
>> BB=[B(1) B(3)]
```

Oss: tramite l'indice è possibile puntare ad un elemento di un vettore quindi anche assegnargli un nuovo valore e di conseguenza **modificare il valore di un elemento del vettore**

Es: assegnare al primo elemento di A il valore 7

```
>> A(1)=7
```

**Esercizio 3**

Memorizzare in MatLab i seguenti vettori:

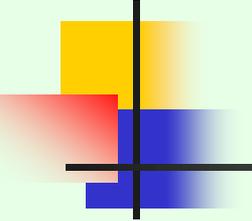
$$x = (-2, \sqrt{3}, e, 10), y = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, z = \begin{pmatrix} 3/5 \\ 7 \\ -4 \end{pmatrix}$$

Che dimensione hanno?

Creare due nuovi vettori: uno colonna (xx) avente come elementi gli elementi di x di indice pari e l'altro riga (zz) avente come elementi i primi due elementi di z.

Sostituire il primo elemento di xx con -20 e sostituire il secondo elemento di zz con il valore 31

Trasformare xx in vettore colonna e zz in vettore riga



Lavorare con i vettori

Oss: può essere necessario **accodare un elemento ad un vettore** in tal caso è sufficiente memorizzare un nuovo vettore che ha come primi elementi lo stesso vettore iniziale e come ultimo elemento il nuovo elemento

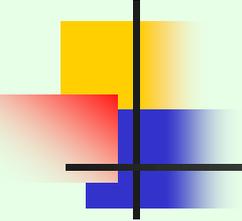
Es: Scrivere il vettore riga M di elementi 3,-3,7,-7,0 quindi aggiungere in coda l'elemento 10

```
>> M=[3 -3 7 -7 0];  
>> MM=[M 10];
```

Oss: in maniera analoga può essere necessario **eliminare un elemento da un vettore** in tal caso basta sostituire quell'elemento con un vettore vuoto []

Es: Eliminare il terzo elemento del vettore MM

```
>> MM(3) = []
```

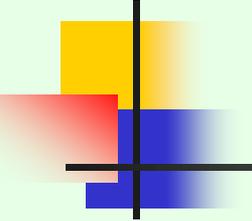


Esercizio 4

Memorizzare in MatLab il vettore colonna **P** avente i seguenti elementi:

10,-1,3,0,7,6

1. Creare il vettore **Q** dal vettore **P** aggiungendo l'elemento 0 in testa e l'elemento 1 in coda.
2. Creare il vettore **R** dal vettore **P** eliminando il secondo, terzo e quarto elemento.
3. Creare il vettore **S** che ha come primi elementi gli elementi del vettore **Q** e come successivi elementi quelli del vettore **R**.
4. Che dimensione ha il vettore **S**?
5. Assegnare al quinto elemento di **S** il valore 7. Che vettore ottenete?
6. Creare il vettore **T** formato dal primo e dal nono elemento di **S**



Operatore :

Oss: a volte, dato un vettore, può essere necessario **estrarre i primi k elementi del vettore** stesso; a tal fine di grande utilità è **l'operatore :** che viene letto da MatLab come **"da ... a ..."**

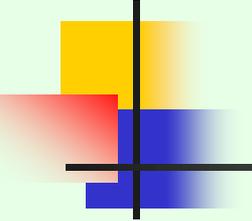
Es: memorizzare il vettore colonna X di elementi 2,4,6,8,10 quindi creare il vettore composto dai primi 3 elementi

```
>> X=[2 4 6 8 10]';  
>> X1=X(1:3)
```

NB: l'operatore : può anche essere usato in maniera più flessibile, ad esempio per estrarre gli elementi dall'indice i all'indice j del vettore

Es: dato il vettore X, creare un vettore formato dagli elementi di X che vanno dal secondo al quarto

```
>> X2=X(2:4)
```



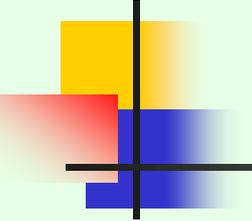
Operatore :

Oss: se volessimo estrarre da un vettore un nuovo vettore composto dagli elementi di indice dispari?

Dovremmo indicare a MatLab di prelevare gli elementi con indice che va da 1 ad n (ultimo indice) di passo 2

NB: utilizzando ancora l'operatore : è possibile richiedere di partire da un certo indice per arrivare ad un altro indice impiegando un certo **passo**.

La scrittura i:p:j viene letta da MatLab come "**da i a j con passo p**"; questo comando può essere utilizzato in diverse occasioni anche per definire vettori **equispaziati** (di elementi ordinati equidistanti tra di loro).



Operatore :

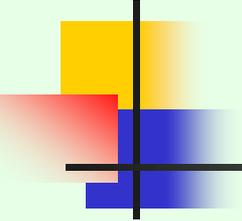
Es: memorizzare il vettore riga Y di elementi che vanno da 0 a 40 di passo 4

```
>> Y=0:4:40;
```

Es: dato il vettore Y, creare un vettore formato dagli elementi di Y che vanno dal secondo al sesto

```
>> Y1=Y(2:6)
```

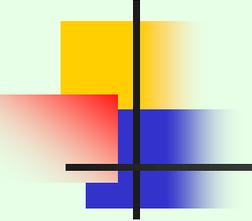
Oss: nell'ultimo esempio non è stato specificato un passo che di conseguenza è, per default, pari ad 1



Esercizio 5

Memorizzare in MatLab il vettore colonna P di elementi che vanno da -10 a 60 di passo 2 ed il vettore riga Q di 11 elementi che vanno da -1 a -6 di opportuno passo

1. Quanti elementi ha il vettore P? (**NB:** fissando gli elementi iniziale e finale ed il passo di conseguenza si ottiene la dimensione del vettore)
2. Quale passo hai usato per creare il vettore Q?
3. Dal vettore P si crei il vettore P1 formato dagli elementi di P di indice pari.
4. Trasformare il vettore riga Q nel vettore colonna Q1 quindi sopprimere da quest'ultimo gli elementi dal secondo al quinto



Operatore linspace

Oss: a volte non è noto il passo ma è noto il numero di elementi che deve avere il vettore a partire da i a j .

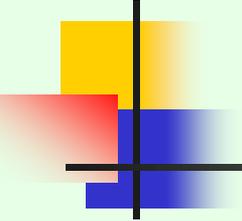
Ad esempio si desidera un **vettore composto da n elementi equispaziati che vanno da i a j** ; in tal caso si usa il comando **`linspace(i,j,n)`**

Es: memorizzare il vettore riga V composto da 20 elementi equispaziati che vanno da -2 a 6

```
>> V=linspace(-2,6,20)
```

Es: sopprimere gli elementi di V di indice pari

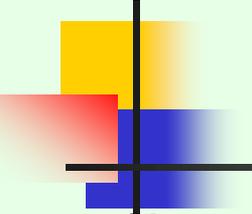
```
>> V(2:2:20)=[]
```



Esercizio 6

Memorizzare in MatLab il vettore riga **A** composto da 60 elementi che vanno da 3 a 20 ed il vettore colonna **B** composto da 10 elementi che vanno da 2 a -5

1. Che dimensione ha il vettore **A**?
2. Trasformare il vettore **B** in vettore riga e creare il vettore **C** avente come primi elementi gli elementi di **A** e come elementi successivi quelli di **B**
3. Creare il vettore **D** estraendo da **C** gli elementi che vanno dal sesto al diciottesimo di passo 3
4. Quanti elementi ha il vettore **D**?



Operazioni: somma

Siano X e Y due vettori riga (o due vettori colonna) di dimensione n

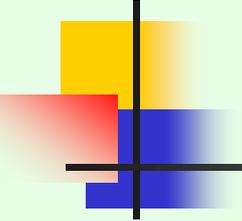
Allora è possibile la **somma** $X+Y$ dei due vettori ed il vettore somma è il vettore riga (o colonna) Z di dimensione n avente ciascun elemento pari alla somma degli elementi di X e di Y che hanno lo stesso indice

NB: una definizione analoga vale nel caso di **differenza** fra due vettori

Il comando per sommare (sottrarre) vettori in MatLab è l'operatore **+** (**-**)

Es: sia P il vettore riga di elementi che vanno da 6 a 20 di passo 2 e sia Q il vettore riga di 8 elementi equispaziati da 1 a 3. Si calcoli $R=P-Q$

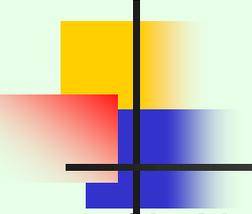
```
>> P=6:2:20;  
>> Q=linspace(1,3,8);  
>> R=P-Q
```



Esercizio 7

Memorizzare in MatLab il vettore riga **A** composto da 16 elementi che vanno da 3 a 20 ed il vettore colonna **B** i cui elementi vanno da 0 a 90 di passo 5

1. E' possibile sommare i due vettori? No...per due motivi, quali?
2. Trasformare opportunamente i due vettori in modo da rendere possibile la loro somma, quindi sommarli



Operazioni: moltiplicazione per uno scalare

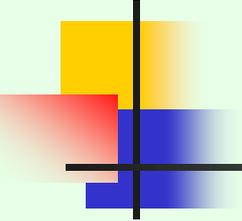
Sia X un vettore riga (o colonna) di dimensione n e k un numero reale

Allora è possibile la **moltiplicazione scalare** kX dello scalare per il vettore ed il vettore che si ottiene è il vettore riga (o colonna) Z di dimensione n avente ciascun elemento pari al prodotto di ciascun elemento di X per lo scalare k

Il comando per moltiplicare un numero per un vettore in MatLab è l'operatore $*$

Es: si calcoli il vettore $S=-10R$

```
>> S=-10*R
```



Esercizio 8

Memorizzare in MatLab il vettore riga X di elementi $-1,0,5,7$ ed il vettore riga Y di 4 elementi equispaziati da 10 a -8 .

1. Calcolare se possibile $Z=X+Y$. Che vettore si ottiene e di che dimensione?
2. Calcolare $V=0.5X$. Che vettore si ottiene e di che dimensione?
3. E' possibile l'operazione $Z-V$? Perché?

Operazioni: operatori puntuali

In MatLab l'**operatore .** permette di compiere altre **operazioni fra vettori "elemento per elemento"**. Quando si sommano vettori l'operazione viene compiuta elemento per elemento: l'elemento generico i-esimo del primo vettore è sommato all'elemento i-esimo del secondo vettore. La somma e anche la moltiplicazione scalare sono operazioni puntuali.

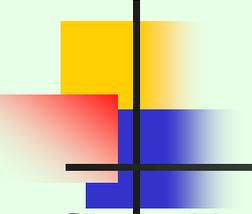
Siano X e Y due vettori riga (o due vettori colonna) di dimensione n

Allora è possibile il **prodotto puntuale $X.*Y$** dei due vettori ed il vettore che si ottiene è il vettore riga (o colonna) Z di dimensione n avente ciascun elemento pari al prodotto degli elementi di X e di Y che hanno lo stesso indice

Il comando per moltiplicare puntualmente due vettori in MatLab è l'operatore **.***

Es: siano dati i vettori colonna A di elementi 0,-1,3 e B di elementi -3,2,2 si calcoli il loro prodotto puntuale

```
>> A=[0;-1;3];B=[-3 2 2]';
>> C=A.*B
```



Operazioni: operatori puntuali

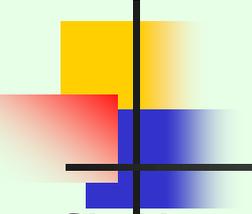
Siano X e Y due vettori riga (o due vettori colonna) di dimensione n con Y privo di elementi nulli.

Allora è possibile la **divisione puntuale $X./Y$** ed il vettore che si ottiene è il vettore riga (o colonna) Z di dimensione n avente ciascun elemento pari al rapporto fra gli elementi di X e gli elementi di Y che hanno lo stesso indice

Il comando per dividere puntualmente due vettori in MatLab è l'operatore **$./$**

Es: siano dati i vettori riga A di elementi $0,-1,1$ e B di elementi $-1,-2,-3$ si calcoli il rapporto puntuale $A./B$

```
>> A=[0 -1 1];B=[-1 -2 -3];  
>> C=A./B
```



Operazioni: operatori puntuali

Sia X un vettore riga (o colonna) di dimensione n e k un numero reale. Sia inoltre possibile elevare ogni elemento di X alla potenza k .

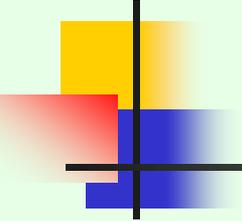
Allora è possibile l' **elevamento a potenza puntuale $X.^k$** ed il vettore che si ottiene è il vettore riga (o colonna) Z di dimensione n avente ciascun elemento pari all'elemento di X che ha lo stesso indice elevato alla potenza k -esima

Il comando per elevare puntualmente un vettore a potenza in MatLab è l'operatore `.`[^]

Es: dato il vettore riga A di 10 elementi equispaziati da 7 a 21 e sia $k=1/3$ si calcoli $A.^k$

```
>> A=linspace(7,21,10);  
>> B=A.^(1/3)
```

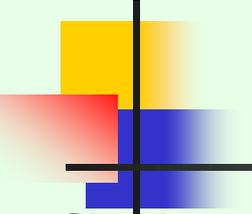
Oss: questa operazione corrisponde al calcolo della radice cubica di ogni elemento del vettore iniziale ed è quindi sempre possibile qualunque sia il vettore dato



Esercizio 9

Memorizzare in MatLab il vettore riga X di elementi $-1,0,1,2,3$ ed il vettore riga Y di elementi che vanno da -3 a 1 di passo 1 .

1. Calcolare Z prodotto puntuale dei due vettori. Il prodotto puntuale gode della proprietà commutativa?
2. Calcolare $V=X./Y$. E' possibile questa operazione? Se no, perché?
3. Assegnare agli elementi nulli di Y un valore unitario quindi calcolare $V=X./Y$. La divisione puntuale è commutativa?
4. Si elevi puntualmente il vettore V al quadrato. E' possibile elevare puntualmente X all'esponente $1/2$. Se no, perché? E' possibile elevare puntualmente Y all'esponente -1 ? Se no, perché?



Operazioni: funzione applicata ad un vettore

Sia X un vettore riga (o colonna) di dimensione n e **f una funzione reale di una variabile definita sui singoli elementi di X** .

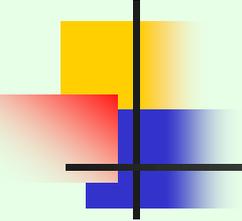
Allora è possibile **calcolare $f(X)$** e si ottiene un vettore riga (o colonna) Z di dimensione n in cui ciascun elemento si ricava applicando la funzione f al corrispondente elemento del vettore di partenza.

Anche questa è un'operazione puntuale.

Es: sia $A=[2,3,7,9]$ e $f(x)=e^{x-4}$ si calcoli $f(A)$

```
>> A=[2,3,7,9];  
>> B=exp(A-4);
```

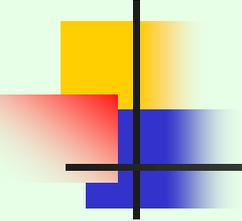
Oss: quando si richiede di calcolare $(A-4)$ MatLab sottrae il numero 4 ad ogni elemento di A , poi applica la funzione esponenziale ad ogni elemento del vettore così ottenuto



Esercizio 10

Memorizzare in MatLab i vettori riga X di elementi da 10 a 30 di passo $\frac{1}{4}$ ed Y di elementi $-1,3,7,0$

1. Sia $f(x)=\ln(x)-\sin x$. Si calcoli $Z=f(X)$. E' possibile? Quanti elementi ha Z ?
2. Sia $f(x)=1/x+2x-\ln(2x)$. Si calcoli $W=f(Y)$. E' possibile? Se no, perché?



Esercizi di ricapitolazione

1.1

Creare il vettore riga X di 24 elementi che vanno da -5 a 7

A. Calcolare $Y = \sqrt{2}X$

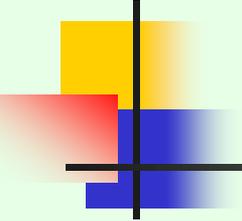
B. Dal vettore Y sopprimere gli elementi aventi indice multiplo di 3

C. Creare il vettore Z di elementi che vanno da 1 a 16 di passo 1

D. Calcolare, se possibile, $V = Y + Z$

E. Sia $f(x) = \frac{\sqrt[3]{2x^2 - 3}}{\cos x}$, calcolare $W = f(V)$

NB: prestare attenzione alle parentesi e all'uso degli operatori puntuali

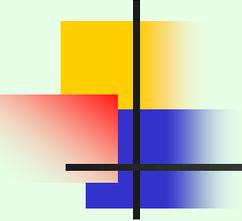


Esercizi di ricapitolazione

1.2

Creare il vettore colonna A di elementi che vanno da 9 a -18 di passo -0,5 e il vettore riga B di elementi (7,-1,3,5,8,e).

- A. Quanti elementi ha A ?
- B. Assegnare al decimo elemento di A il valore -3
- C. Trasformare il vettore riga B nel vettore colonna $B1$
- D. Creare il vettore C concatenando il vettore $B1$ con il vettore A
- E. Sia $f(x) = e^{\frac{|x+2|}{x^2}}$, calcolare $D = f(C)$

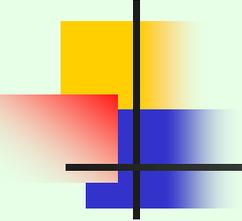


Esercizi di ricapitolazione

1.3

Creare il vettore riga X di 8 elementi equispaziati da 1 a 25, il vettore riga Y di elementi da 10 a 90 di passo 10, il vettore colonna Z di 8 elementi tutti unitari

- A. E' possibile la somma $X + Y$? Se no, perché?
- B. Sopprimere dal vettore Y il sesto elemento
- C. Trasformare il vettore colonna Z nel vettore riga V
- D. Calcolare $W = (X ./ Y).^ V$
- E. Sia $f(x) = (x + 30) \ln(x + 30)$, calcolare $f(W)$



Esercizi di ricapitolazione

1.4

Creare i vettori riga $x=(-1,1,-2,2,-3,3)$, $y=(7,5,-1,3,2,0)$ e $z=(e,e^2,e^3,e^4,e^5,e^6)$

NB: si cerchi un modo veloce per scrivere z che eviti di elencare tutti i suoi elementi

A. Memorizzare $a = e^3$, $b = \log_2 10$, $c = \sqrt[3]{5^2}$

B. Calcolare $V = (ax - by)cz$

C. Sia $f(x) = (\sin^2 x + \tan x^2) \log_{10} |x^4 + 6|$, calcolare $f(V)$.

Il vettore che si ottiene, quanti elementi ha?

Memorizzare in MatLab una matrice

MATRICE di m righe e n colonne

- Per memorizzarla basta elencare i suoi elementi **riga per riga**, racchiudendo gli elementi della matrice tra **parentesi quadrate**
- Gli elementi di una riga sono separati da uno **spazio** o da una **virgola**
- Le righe sono separate da un **punto e virgola**

ES. MEMORIZZARE LA SEGUENTE MATRICE:

$$A = \begin{pmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{pmatrix}$$

```
>> A=[1 3 5; 2 4 6]
```

Lavorare con le matrici

NB. Se sono in memoria i vettori riga (o colonna) della matrice

basta scriverla come **vettore colonna avente per elementi i vettori riga dati**
(o scriverla come **vettore riga avente per elementi i vettori colonna dati**)

ES. Consideriamo la matrice A dell'esempio precedente:

```
>> u=1:2:5;  
>> v=2:2:6;  
>> A=[u;v]
```

OSS: Ciascun elemento di una matrice è identificato da **due indici**:
il primo indica la riga e il secondo la colonna.

Nell'esempio precedente 6 è l'elemento di indici 2 (di riga) e 3 (di colonna).

DI CONSEGUENZA:

è possibile **ACCEDERE AD UN ELEMENTO DI UNA MATRICE** proprio specificando l'indice i di riga e l'indice j di colonna tra parentesi tonde dopo il nome della matrice: A(i,j)

```
>> A(2,3)
```

Lavorare con le matrici

OSS:

Tramite gli indici è possibile puntare ad un elemento di una matrice quindi anche assegnargli un nuovo valore e di conseguenza **modificare il valore di un elemento della matrice**

```
>> A=[1 3 5; 2 4 6]
```

ES. ASSEGNARE ALL'ELEMENTO DI POSTO (1,2) DELLA MATRICE A IL VALORE -1

```
>> A(1,2)=-1
```

Ora l'elemento di posto (1,2) assume valore -1

A =

```
1  -1  5
2   4  6
```

Lavorare con le matrici

(1) Può essere necessario **ACCEDERE A UN VETTORE RIGA (O COLONNA)**
in tal caso si usa l'operatore ":"

$A(:,j)$ individua la j-ma colonna di A, $A(i,:)$ individua la i-ma riga di A

ES.

- Individuare la prima colonna di A

```
>> A(:,1)
```

- Individuare la seconda riga di A

```
>> A(2,:)
```

Lavorare con le matrici

(2) In maniera analoga può essere necessario **ESTRARRE SOTTOMATRICI** in tal caso basta specificare l'intervallo degli indici di riga e di colonna

ES.

Estrarre la sottomatrice avente per righe tutte le righe di A e per colonne la prima e la seconda di A

```
>> A(:,1:2)
```

Sostituire la seconda riga di A con il vettore i cui elementi vanno da 2 a 6 di passo 2

```
>> A(2,:)=2:2:6
```

Estrarre la sottomatrice avente per righe tutte le righe di A e per colonne la prima e la terza di A

```
>> A(:, [1,3])
```

Lavorare con le matrici

RICAPITOLANDO

- $A(i,j)$ estrae l'elemento (i,j) : i-esima riga e j-esima colonna
- $A(i,:)$ estrae la i-esima riga
- $A(:,j)$ estrae la j-esima colonna
- $A(:,[n,m])$ estrae le colonne n-esima e m-esima

Lavorare con le matrici

(3) È inoltre possibile **SOPPRIMERE ALCUNI ELEMENTI DI UNA MATRICE**
DI CONSEGUENZA cambia la dimensione

ES.

Data la matrice

```
>> A=[1 2 3; 4 5 6; 7 8 9]
```

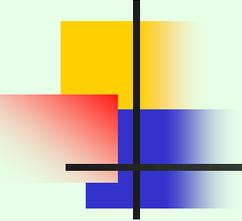
A =

```
1  2  3
4  5  6
7  8  9
```

Sopprimere la prima riga di A

```
>> A(1,:)=[]
```

ESERCIZIO PROPOSTO

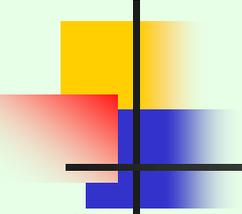


- MEMORIZZARE IN MatLab LA SEGUENTE MATRICE:

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 0 \end{pmatrix}$$

- elencando gli elementi
- usando vettori riga (o colonna) precedentemente memorizzati
- **ESTRARRE** la sottomatrice avente per righe le prime due righe di A e per colonne le ultime due di A
- sostituire la prima riga di A con il vettore [1,1,1]
- estrarre la sottomatrice avente per righe la prima e la terza di A e per colonne tutte le colonne di A

Matrici: applicazioni elementari



SOLO se A è una matrice QUADRATA:

(1) **DETERMINANTE** della matrice A

```
>> det(A)
```

Calcola il determinante di A

SOLO se A è una matrice QUADRATA CON DET DIVERSO DA ZERO:

(2) **MATRICE INVERSA**

```
>> inv(A)
```

Calcola la matrice inversa di A

Matrici: applicazioni elementari

Data una matrice A, MatLab offre diverse funzioni predefinite

(3) **DIMENSIONE** della matrice A

```
>> size(A)
```

Restituisce un vettore avente per elementi il n. di righe e il n. di colonne di A

(4) **MATRICE TRASPOSTA** mediante l'apice

```
>> A'
```

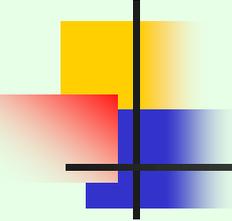
Scambia le righe con le colonne di A

(5) **RANGO** della matrice A

```
>> rank(A)
```

Calcola il rango di A

Matrici: funzioni predefinite



`zeros(m,n)`

genera una **matrice nulla** di m righe e n colonne

`ones(m,n)`

genera una **matrice unitaria** di m righe e n colonne

`eye(n)`

genera la **matrice identità** di ordine n

Il comando `>>doc elmat` permette di ottenere una lista completa delle funzioni elementari consultando l'help di MatLab

ES.

```
>> I=eye(3)
```

```
I =
```

```
1 0 0
0 1 0
0 0 1
```

```
>> A=ones(2,3)
```

```
A =
```

```
1 1 1
1 1 1
```

ESEMPIO

```
>> A=ones(2,2);  
>> A_inv=inv(A)  
Warning: Matrix is singular to working  
precision.
```

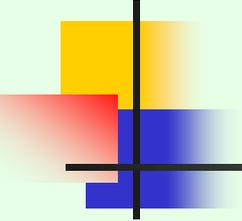
```
A_inv =
```

```
    Inf    Inf  
    Inf    Inf
```

```
>> d=det(A)
```

```
d =
```

```
    0
```

ESERCIZIO PROPOSTO 11

- MEMORIZZARE IN MatLab LA SEGUENTE MATRICE:

$$A = \begin{pmatrix} 2 & 0 & 2 \\ 1 & 2 & 1 \\ 4 & -1 & 3 \end{pmatrix}$$

- calcolare: la dimensione, la trasposta, il determinante.
- è possibile determinare la matrice inversa? Perché?

Matrici: operazioni

PRODOTTO DI UNA MATRICE PER UN VETTORE

Siano A una matrice con m righe e n colonne e x un vettore colonna di dimensione n

Allora è possibile il **prodotto** Ax mediante l'operatore *

Il risultato è un VETTORE COLONNA DI DIMENSIONE m

```
>> A=[1 2 3; -3 -1 2];  
>> x=[-1;0.5;3];  
>> y=A*x
```

```
y =
```

```
9.0000
```

```
8.5000
```

Matrici: operazioni

SOMMA TRA MATRICI

Siano A e B due con lo stesso numero di righe e di colonne
allora è possibile la **somma** A+B

Il risultato è una MATRICE della stessa dimensione di A e B

```
>> A=[1 2 3; -3 -1 2];  
>> B=[-1 2 0.5; 3 -3 6];  
>> C=A+B
```

```
C =
```

```
    0    4.0000    3.5000  
    0   -4.0000    8.0000
```

OSS. In maniera analoga è possibile effettuare A-B

Matrici: operazioni

PRODOTTO DI UNA MATRICE PER UNO SCALARE

Dato uno scalare a è possibile moltiplicarlo per una matrice mediante l'operatore *

Il risultato è una **MATRICE** della stessa dimensione di A

```
>> A=[1 2 3; -3 -1 2];
```

```
>> B=2*A
```

```
B =
```

```
2    4    6
```

```
-6   -2    4
```

Matrici: operazioni

PRODOTTO TRA MATRICI

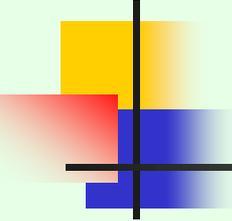
Siano A e B due matrici **conformabili** (numero di colonne di A = numero di righe di B)
allora è possibile il **prodotto** A per B mediante l'operatore *

```
>> A=[0 4 3; 0 -4 8];  
>> B=[0 1; -1 2; -2 3];  
>> C=A*B
```

```
C =
```

```
-10  17  
-12  16
```

ESERCIZIO PROPOSTO 12



1) MEMORIZZARE IN MatLab LA MATRICE A 3×5 AVENTE:

- prima riga il vettore di elementi da 0 a 12 equispaziati
- seconda riga unitaria
- terza riga gli elementi da 2 a -2 con opportuno passo

2) SI ESTRAGGA LA TERZA RIGA E SI MOLTIPLICHI OGNI SUO ELEMENTO PER 2, SIA x IL VETTORE OTTENUTO

3) SI MOLTIPLICHI LA MATRICE A PER IL VETTORE x (SI ESEGUANO LE OPERAZIONI NECESSARIE PER RENDERE POSSIBILE IL PRODOTTO)

4) SI ESEGUA LA TRASPOSTA DELLA MATRICE A E SI MOLTIPLICHI LA MATRICE OTTENUTA PER LA MATRICE B:

$$B = \begin{pmatrix} 2 & -1 \\ 1 & 3 \\ 0 & -2 \end{pmatrix}$$

Matrici: operazioni PUNTUALI

PRODOTTO (E RAPPORTO) PUNTUALE

Siano A e B due matrici aventi lo **STESSO NUMERO DI RIGHE E DI COLONNE**

allora è possibile il **prodotto (e rapporto) puntuale** mediante l'operatore **.*** (**./**)

```
>> A=[0 3; -4 2];
>> B=[7 2; -3 -1];
>> C=A.*B
```

C =

```
    0    6
   12   -2
```

```
>> D=A./B
```

D =

```
    0    1.5000
   1.3333 -2.0000
```

- 1) La matrice C ha per elemento (i,j) il prodotto tra gli elementi (i,j) delle matrici A e B
- 2) La matrice D ha per elemento (i,j) il rapporto tra gli elementi (i,j) delle matrici A e B

NB. Il rapporto puntuale di A e B è possibile solo se ogni elemento di B è diverso da zero.

Matrici: operazioni PUNTUALI

ELEVAMENTO A POTENZA PUNTUALE

Data una matrice A è possibile elevare ogni suo elemento ad uno scalare α mediante l'operatore `.`[^]

```
>> A=[1 2; 3 4];  
>> B=A.^2
```

B =

```
1    4  
9   16
```

OSS. L'operazione A^2 esegue il prodotto $A*A$:

```
>> C=A^2
```

C =

```
7    10  
15   22
```

In modo analogo è possibile **l'ELEVAMENTO A POTENZA PUNTUALE** tra due matrici

```
>> A=[1 2; 3 4];  
>> B=[5 3; 3 2];  
>> C=A.^B
```

C =

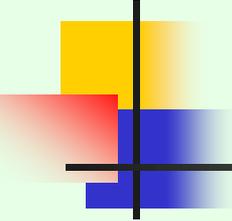
```
1    8  
27   16
```

NB: La matrice C ha per elemento (i,j)
l'elemento (i,j) della matrice A
elevato all'elemento (i,j) della
matrice B

Infine data una matrice A e una funzione $y=f(x)$ è possibile **applicare la funzione alla matrice**:

- Prestando attenzione all'utilizzo della corretta sintassi per le funzioni
- Prestando attenzione all'utilizzo degli operatori puntuali

ESERCIZIO PROPOSTO 13



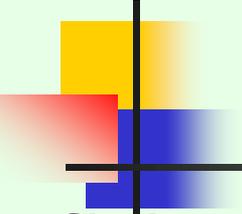
1) MEMORIZZARE IN MatLab LA MATRICE A 3×3 AVENTE:

- prima colonna il vettore di 3 elementi da 1 a -5 equispaziati
- seconda colonna il vettore composto da tutti elementi pari a due
- terza colonna gli elementi da 1 a 6 con opportuno passo

2) Si memorizzi la trasposta di A e si elevi ogni suo elemento al quadrato, sia B la matrice così ottenuta

3) Si moltiplichino ogni elemento di A con il corrispondente elemento di B, sia C la matrice così ottenuta

Matrici: funzioni applicate a MATRICI



Sia A una matrice $m \times n$ **f una funzione reale di una variabile definita sui singoli elementi di A .**

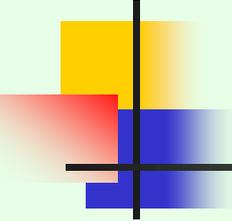
Allora è possibile **calcolare $f(X)$** e si ottiene una matrice B $m \times n$ in cui ciascun elemento si ricava applicando la funzione f al corrispondente elemento della matrice di partenza.

Anche questa è un'operazione puntuale.

Es: sia $A=[2,3;7,9]$ e $f(x)=\log(|x-10|)$ si calcoli $f(A)$

Oss: quando si richiede di calcolare $(A-10)$ MatLab sottrae il numero 4 ad ogni elemento di A .

ESERCIZIO PROPOSTO 14



1) MEMORIZZARE IN MatLab LA MATRICE A 2×3 AVENTE:

- prima colonna il vettore di 3 elementi da 1 a 7 equispaziati
- seconda colonna il vettore composto da tutti elementi pari a -1

2) Si determini la matrice B ottenuta da A applicando la funzione

$$F(x) = \frac{(x+1)}{(x-2)}$$