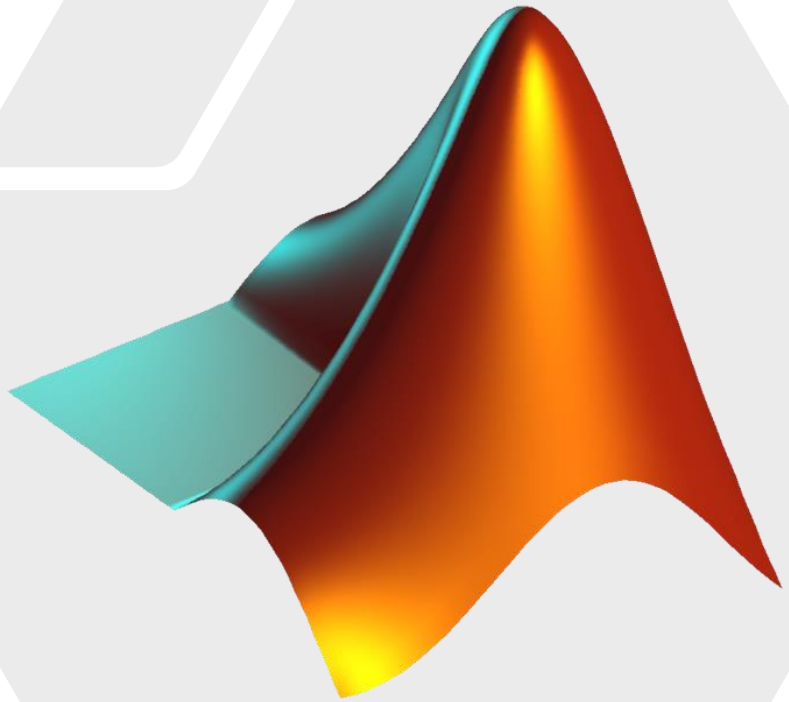
Data Analysis 1

Week 4 (16:00, 27/10/2021)



Vivatsathorn Thitasirivit

Student. Faculty of Engineering, Chulalongkorn University.

Week 4 Outlines

Second order of average measurement

- Covariance & Correlation

MATLAB

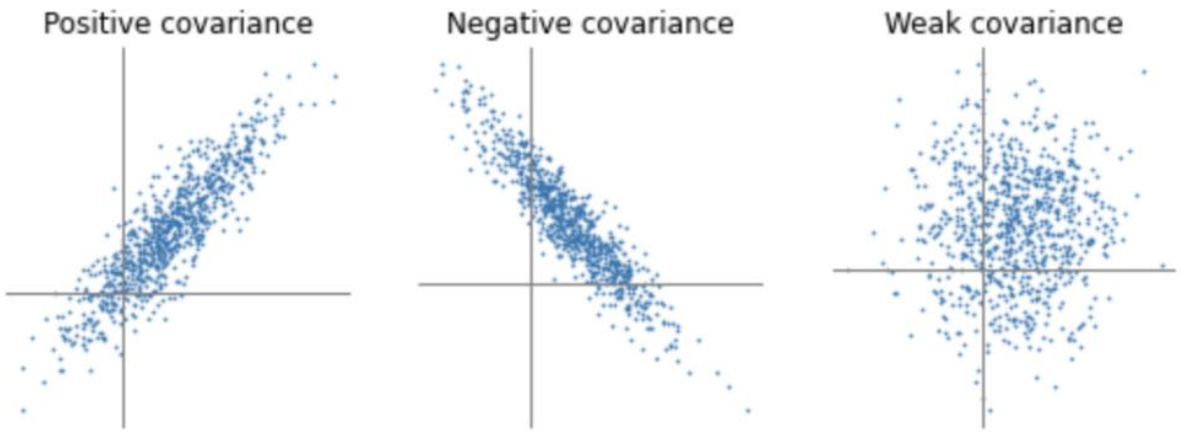
- Basic Matrices Operations
- Elemental Operations
- Figures
- Next week: Functions

Assignments

Data Samples

Covariance

$$\mathrm{Cov}(X, Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$$

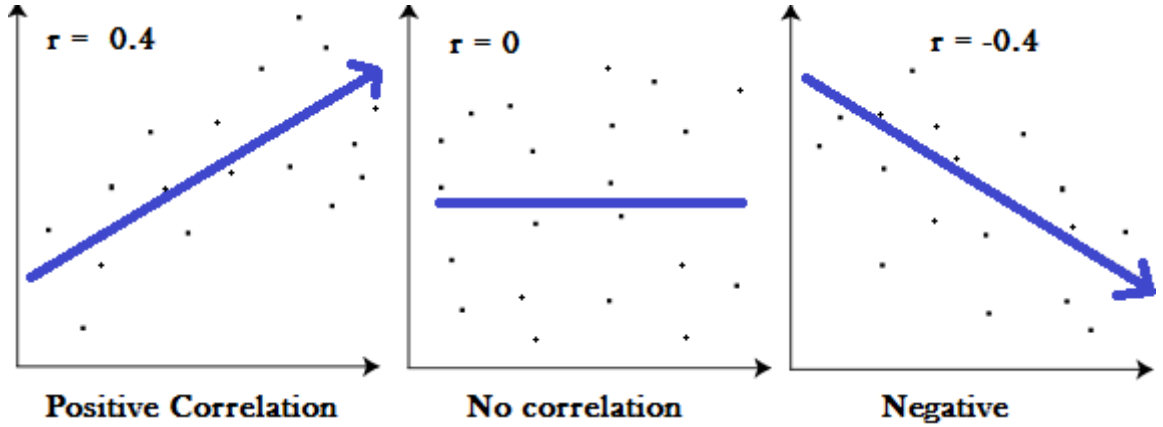


Programmathically: Covariance Sign

Correlation Coefficient

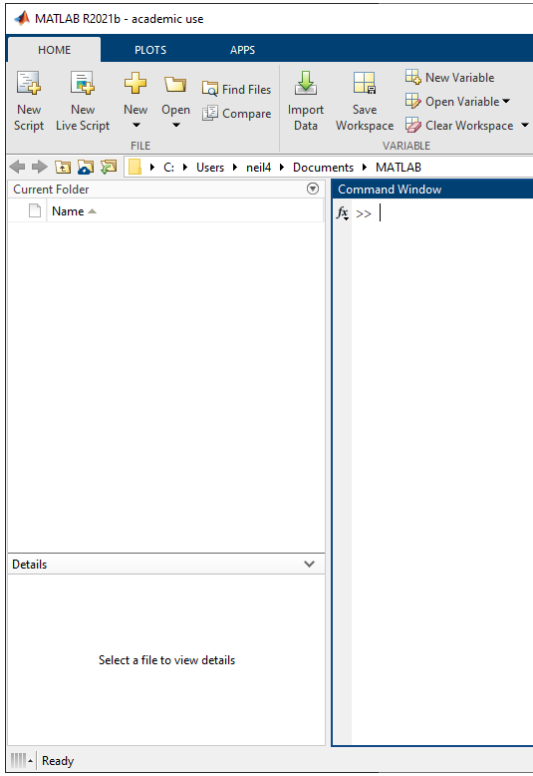
$$\mathrm{Cov}(X,Y) \xrightarrow{\text{Normalization}} r$$

$$r = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}} = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma(x_i - \bar{x})^2}\sqrt{\Sigma(y_i - \bar{y})^2}}$$



Stats How To: Correlation Coefficient Sign

MATLAB for Engineer & Scientist



A Matrix Laboratory for Matrix-based Calculations

It is very high-performance + fast language for technical computing, visualization, and analysis.

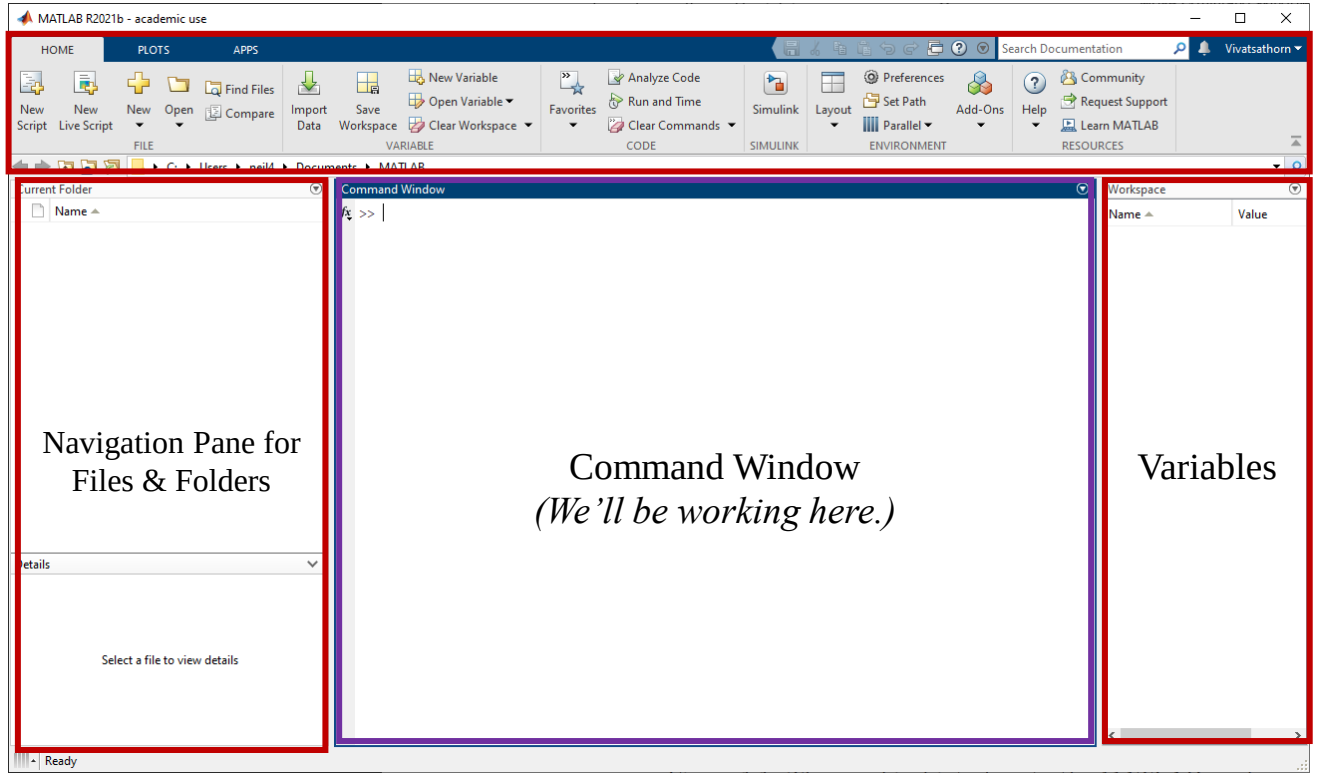
The data do not have to be dimensioned like an array (like in C, except for Python, of course)

Wide mathematical library

Python is a general-purpose programming language, whereas MATLAB is a technical computing language. The performance between two is significantly different when come to Big Data. Python will have its advantage in machine-learning-based analysis and computing.

MATLAB Graphical Interface

Toolbar



MATLAB Basic Commands

Addition & Subtraction

```
>> 1-2+3
ans =
    2
```

Multiplication & Division

```
>> 1/2*3
ans =
    1.5000
```

Power

```
>> 2.^3
ans =
    8

>> 2.^(1/2)
ans =
    1.4142
```

Symbol	Operation	Example
+	Addition	1+2
-	Subtraction	2-5
*	Multiplication	6*7
/	Division	1/3
.^	Power of	3.^2
exp()	Exponential	exp(3)
sqrt()	Square Root	sqrt(3)

MATLAB Basic Commands

Constant pi

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

Constant e

```
>> exp(1)
ans =
    2.7183
```

Square root of 2

```
>> sqrt(2)
ans =
    1.4142
```

MATLAB Basic Commands

Variables

```
>> x=3
x =
     3

>> x*8
x =
     24
```

Overriding Variables

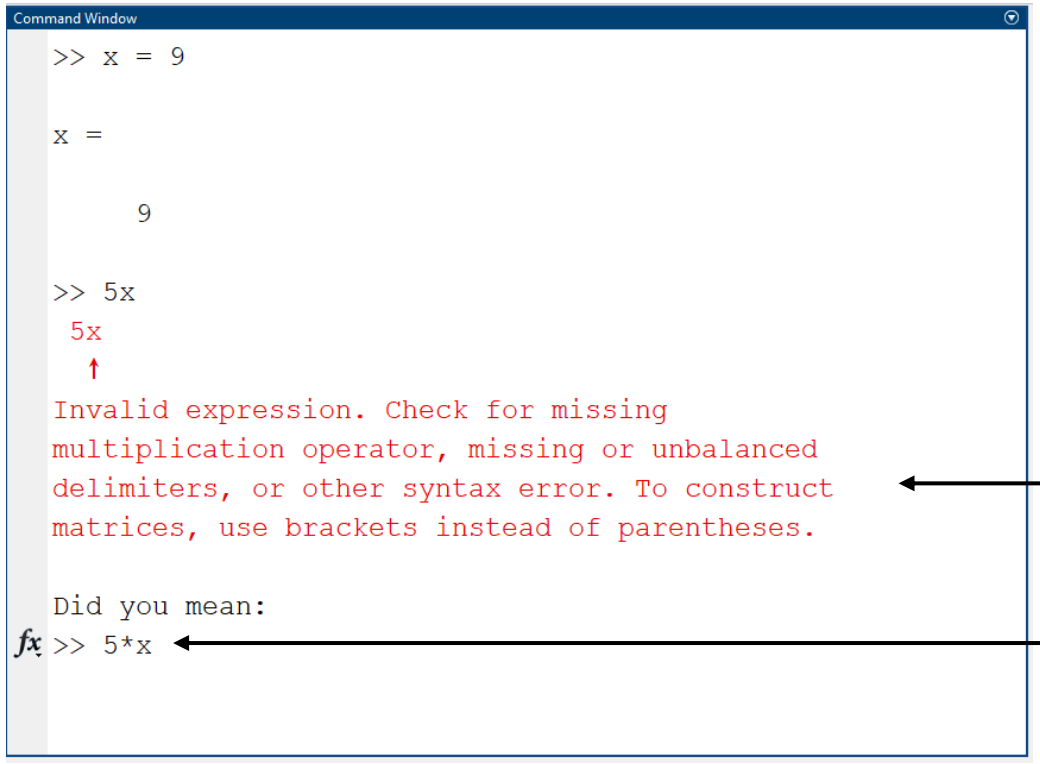
```
>> x=7

or

>> x=x+5
```

MATLAB Basic Commands

Error Handling



Error Message

Correction Recommendation

MATLAB Basic Commands

Hierarchy of Arithmetic Operations

Level	Operation	
First	Elements inside all parentheses are evaluated first, starting from the innermost parentheses, then working inside out.	(…)
Second	All exponents are evaluated from left to right	.^
Third	All multiplications and divisions are evaluated from left to right. Content between symbols will act as only one element, e.g., 2/5*3 is equivalent to (2/5)*3.	* /
Fourth	All additions and subtractions are evaluated from left to right.	+ -

MATLAB Basic Commands

Hierarchy of Arithmetic Operations

$$\frac{1}{2+3^2}+\left(\frac{4}{7}*\frac{5}{6}\right)^3$$

```
>> 1/(2+3.^2)+(4/7*5/6).^3
ans =
    0.1989
```

Number Decimal Format

```
>> format short        3.1416

>> format long         3.141592653589793
```

Use semicolon to hide output

```
>> x=3   ⟶   >> x=3;
x =
     3
```

Level	Operation
First	Elements inside all parentheses are evaluated first, starting from the innermost parentheses, then working inside out.
Second	All exponents are evaluated from left to right
Third	All multiplications and divisions are evaluated from left to right. Content between symbols will act as only one element, e.g., 2/5*3 is equivalent to (2/5)*3.
Fourth	All additions and subtractions are evaluated from left to right.

MATLAB Basic Commands

Managing Workspace

Clear Variables

```
>> clear
```

Clear Command Screen

```
>> clc
```

My Variable List

```
>> who
```

Variable Details

```
>> whos
```

Statement before semicolon will get hidden

Multiple Statement (with ; or ,)

```
>> a=7, b=sin(a); c=acos(a), d=-c
a =

     7
c =

   0.0000 + 2.6339i
d =

   0.0000 - 2.6339i
```

MATLAB Basic Commands

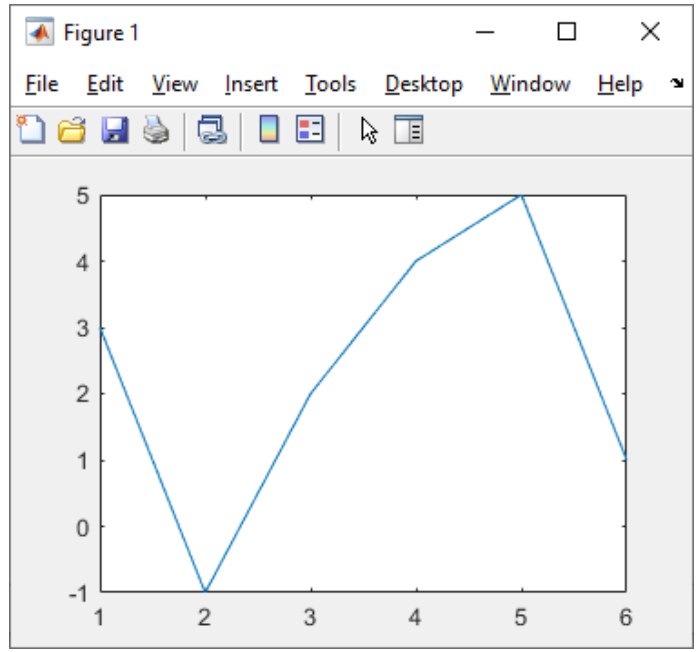
Functions

Symbol	Operation	Symbol	Operation	Symbol	Operation
`cos(x)`	Cosine	`sec(x)`	Secant	`abs(x)`	Absolute
`sin(x)`	Sine	`csc(x)`	Cosecant	`sign(x)`	Sign
`tan(x)`	Tangent	`cot(x)`	Cotangent	`max(x)`	Maximum
`acos(x)`	Arc Cosine	`asec(x)`	Arc Secant	`min(x)`	Minimum
`asin(x)`	Arc Sine	`acsc(x)`	Arc Cosecant	`ceil(x)`	Ceiling
`atan(x)`	Arc Tangent	`acot(x)`	Arc Cotangent	`floor(x)`	Floor
`exp(x)`	Exponential	`pi`	Pi	`round(x)`	Round to nearest integer
`sqrt(x)`	Square Root	`i, j`	Imaginary unit	`rem(x)`	Remainder after division
`log(x)`	Natural Logarithm	`Inf`	The infinity	`angle(x)`	Phase angle of Trig. Function
`log10(x)`	Base 10 Logarithm	`NaN`	Not a number	`conj(x)`	Complex conjugate

MATLAB Basic Plotting

Simple plot

```
>> x = [1,2,3,4,5,6];
>> y = [3, -1, 2, 4, 5, 1];
>> plot(x,y)
```

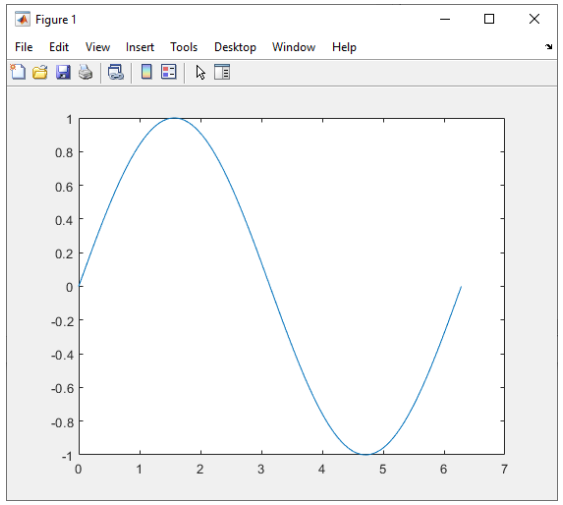


MATLAB Basic Plotting

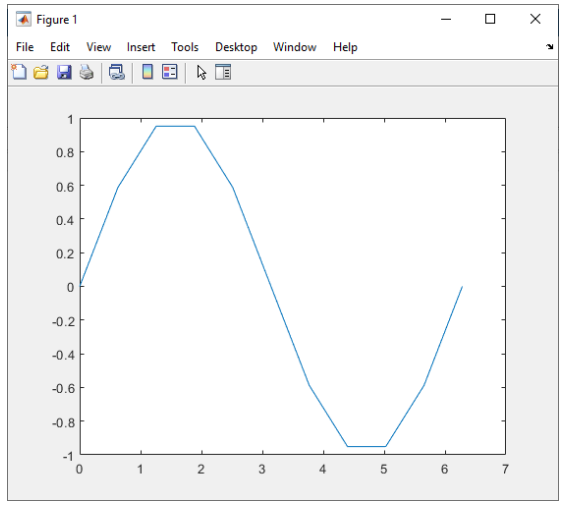
Interval and Step

Syntax for x variable in interval [*a*,*b*] with step *s* in between: `x = a:s:b`

Example




```
>> x = 0:pi/100:2*pi;
>> y = sin(x);
>> plot(x,y)
```

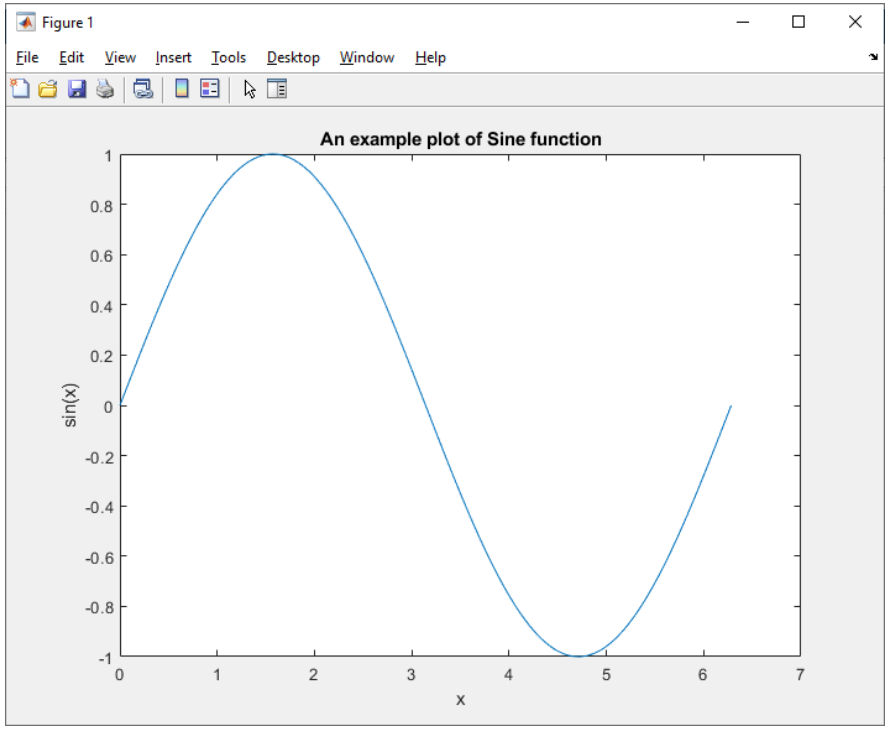



```
>> x = 0:pi/5:2*pi;
>> y = sin(x);
>> plot(x,y)
```

MATLAB Basic Plotting

Labeling Plot

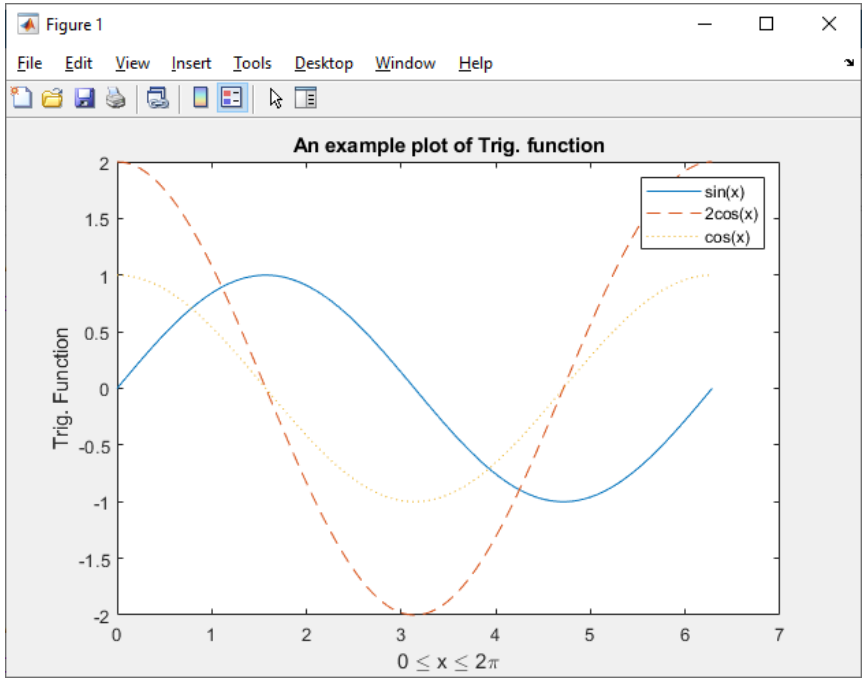
```
>> x = 0:pi/100:2*pi;
>> y = sin(x);
>> plot(x,y)
>> xlabel('0 \leq x \leq 2\pi')
>> ylabel('sin(x)')
>> title('An example plot of Sine function')
```



MATLAB Basic Plotting

Multiplot & Legends

```
>> x = 0:pi/100:2*pi;
>> y1 = sin(x);
>> y2 = 2*cos(x);
>> y3 = cos(x);
>> plot(x,y1,'-',x,y2,'--',x,y3,':')
>> xlabel('0 \leq x \leq 2\pi')
>> ylabel('Trig. Function')
>> title('An example plot of Trig. function')
>> legend('sin(x)','2cos(x)','cos(x)')
```



MATLAB Basic Plotting

Chart Style

Axis Range

```
>> axis([x1,x2,y1,y2])
```

Line Colors & Styles

```
>> plot(x,y,'style1','style2',…)
```

Symbol	Color	Symbol	Line Style	Symbol	Marker
r	Red	-	Solid	+	Plus sign
g	Green	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
c	Cyan	-.	Dash-dot (Phantom)	.	Point
m	Magenta	none	No line	x	Cross
y	Yellow			s	Square
k	Black			d	Diamond

MATLAB Basic Plotting

Chart Style

Axis Range

```
>> axis([x1,x2,y1,y2])
```

Line Colors & Styles

```
>> plot(x,y,'style1','style2',…)
```

Symbol	Color	Symbol	Line Style	Symbol	Marker
r	Red	-	Solid	+	Plus sign
g	Green	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
c	Cyan	-.	Dash-dot (Phantom)	.	Point
m	Magenta	none	No line	x	Cross
y	Yellow			s	Square
k	Black			d	Diamond

MATLAB Matrices

Generating a Vector: *Vector is a* 1x*N or M*x1 *dimensional matrix.*

Row vector

```
>> v=[1 4 7 10 13] or >> v=[1,4,7,10,13]
```

$$\mathbf{v} = [1\ \ 4\ \ 7\ \ 10\ \ 13]$$

Column vector

```
>> w=[1;4;7;10;13]
```

$$\mathbf{w} = \begin{bmatrix} 1 \\ 4 \\ 7 \\ 10 \\ 13 \end{bmatrix}$$

Transpose

```
>> u=v'
```

Value of Element(s)

```
>> a=v(1)          >> d=v(1,end)
>> b=v(2)          >> e=v(1:end)
>> c=v(1:3)        >> f=v(:)
```

MATLAB Matrices

Generating a Matrix: *Usual Matrix dimension is M row* x *N column.*

Matrix generation

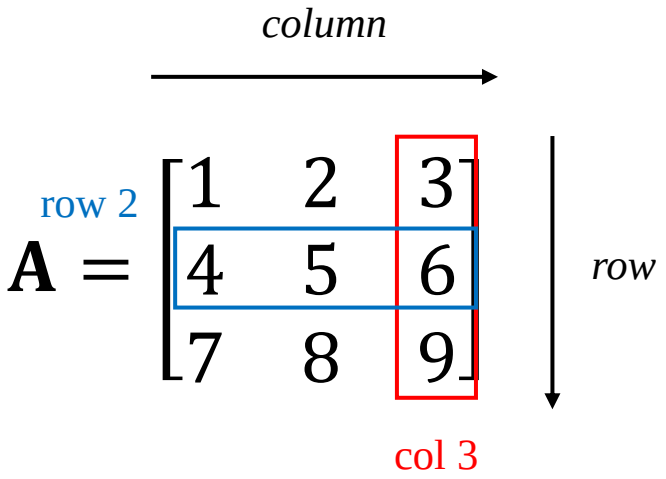
```
>> A=[1 2 3;4 5 6;7 8 9] or
>> A=[1,2,3;4,5,6;7,8,9]
```

Matrix element indexing: Use `A(row,col)`

```
>> A(2,3)
ans =
     6
```

Matrix element overriding

```
>> A(2,3)=0
A =
      1      2      3
      4      5      0
      7      8      9
```



MATLAB Matrices

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Generating a Matrix: *Usual Matrix dimension is M row* x *N column.*

Dimension

```
>> size(A) or >> [m,n] = size(A)
```

Colon Operator

`>> x=0:0.1:5` Means this vector *x* has 51 elements (50 subintervals) ranging from 0 to 5 with 0.1 interval: (0,0.1,0.2,0.3,…,4.9,5)

Linear Spacing

`>> y=linspace(0,2*pi,101)` Means this vector *y* has 101 elements (100 subintervals) from 0 to 2π

Colon Operator in Matrix *for selecting range of element in a Matrix*

`>> A(m:n,k:j)` Means selecting from row *m* to *n* and column *k* to *j*.
Leave *m* and *n* blank for selecting ALL rows `>> A(:,k:j)`
Leave *k* and *j* blank for selecting ALL columns `>> A(m:n,:)`

`>> A(:)` Means iterating elements in **A** into a single column vector