Matlab Functions

Lab 1

Note: In Matlab, you can find help document by typing "help" followed by the command name. For example: "help find" returns detailed explanation on how to use find function.

find Find indices of nonzero elements.

I = find(X) returns the linear indices corresponding to the nonzero entries of the array X. X may be a logical expression

Example:

```
>> x = [1 2 3 0 9 8 0 3 9];
>> i = find(x)

i =

     1     2     3     5     6     8     9

>> x(i)

ans =

     1     2     3     9     8     3     9

>> j = find(x>3)

j =

     5     6     9

>> x(j)

ans =

     9     8     9

>>
```

mean Average or mean value.

For vectors, mean(X) is the mean value of the elements in X. For matrices, mean(X) is a row vector containing the mean value of each column. For N-D arrays, mean(X) is the mean value of the elements along the first non-singleton dimension of X.

Example:

```
>> x = 0:10

x =

     0     1     2     3     4     5     6     7     8     9    10

>> mean(x)

ans =

     5

fx >>
```

plot Linear plot.

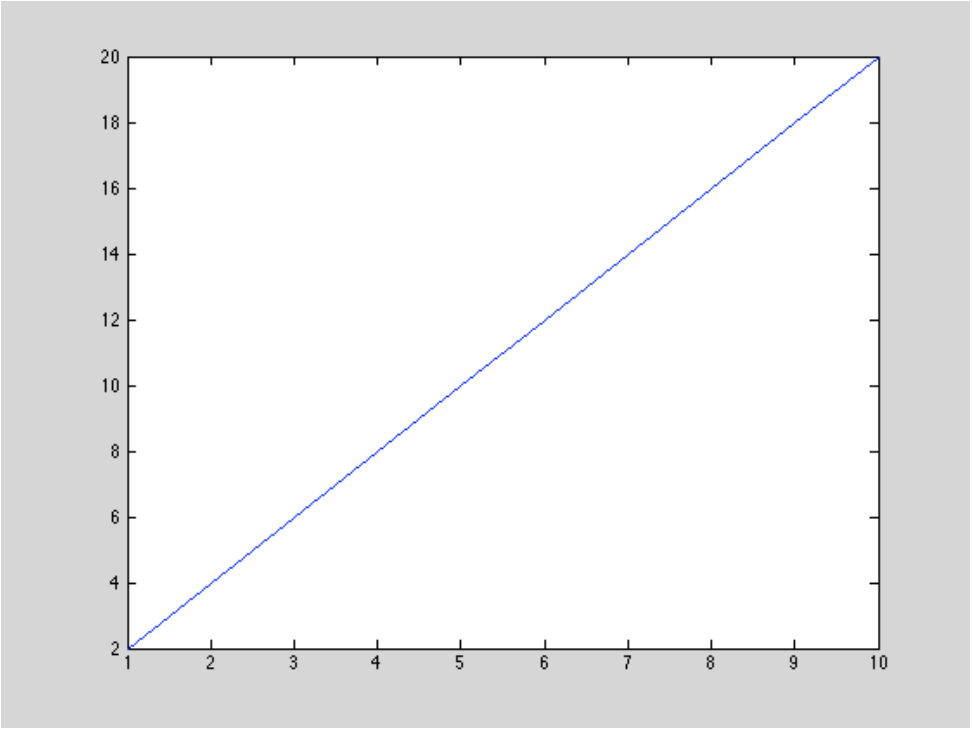
plot(X,Y) plots vector Y versus vector X. If X or Y is a matrix, then the vector is plotted versus the rows or columns of the matrix, whichever line up. If X is a scalar and Y is a vector, disconnected line objects are created and plotted as discrete points vertically at X.

plot(Y) plots the columns of Y versus their index.
If Y is complex, plot(Y) is equivalent to plot(real(Y),imag(Y)).
In all other uses of plot, the imaginary part is ignored.

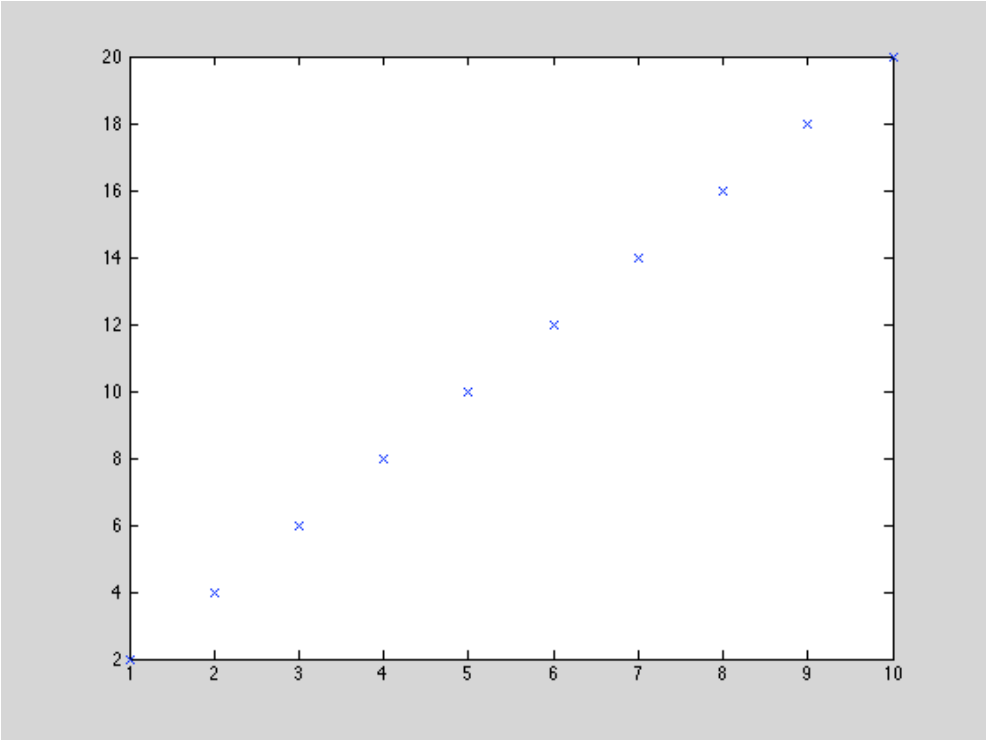
Various line types, plot symbols and colors may be obtained with plot(X,Y,S) where S is a character string made from one element from any or all the following 3 columns:

b	blue	.	point	-	solid
g	green	o	circle	:	dotted
r	red	x	x-mark	-.	dashdot
c	cyan	+	plus	--	dashed
m	magenta	*	star	(none)	no line
y	yellow	s	square		
k	black	d	diamond		
w	white	v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		

```
>> x = 1:10

x =

     1     2     3     4     5     6     7     8     9    10

>> y = 2*x

y =

     2     4     6     8    10    12    14    16    18    20

>> plot(x,y)
```


20
18
16
14
12
10
8
6
4
2
1
2
3
4
5
6
7
8
9
10

```
>> plot(x,y,'bx')
>>
```


20
18
16
14
12
10
8
6
4
2
1
2
3
4
5
6
7
8
9
10

legend Display legend.
legend(string1,string2,string3, ...) puts a legend on the current plot using the specified strings as labels. legend works on line graphs, bar graphs, pie graphs, ribbon plots, etc. You can label any solid-colored patch or surface object. The fontsize and fontname for the legend strings matches the axes fontsize and fontname.

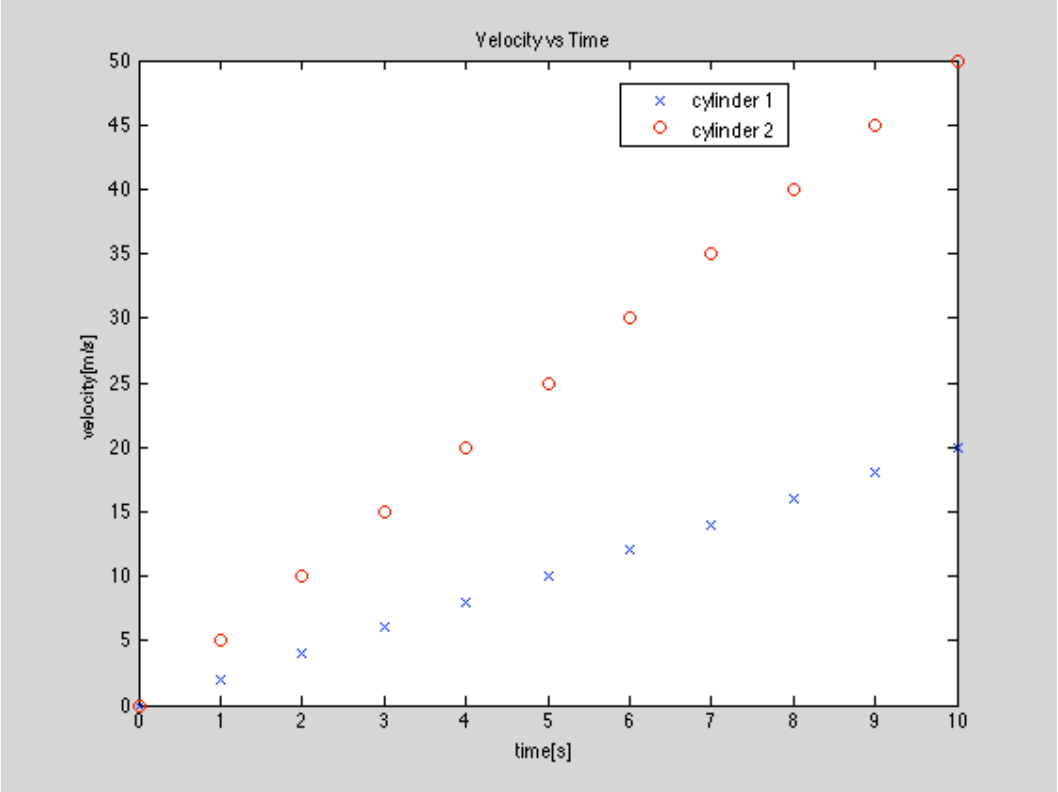
xlabel X-axis label.
xlabel('text') adds text beside the X-axis on the current axis.

ylabel Y-axis label.
ylabel('text') adds text beside the Y-axis on the current axis.

title Graph title.
title('text') adds text at the top of the current axis.

Example

```
>> x = 0:10

x =

     0     1     2     3     4     5     6     7     8     9    10

>> y = 2*x

y =

     0     2     4     6     8    10    12    14    16    18    20

>> z = 5*x

z =

     0     5    10    15    20    25    30    35    40    45    50

>> plot(x,y,'bx',x,z,'ro')
>> legend('cylinder 1','cylinder 2')
>> title('Velocity vs Time')
>> xlabel('time[s]')
>> ylabel('velocity[m/s]')
>>
```


Velocity vs Time
cylinder 1
cylinder 2
velocity[m/s]
time[s]
0
5
10
15
20
25
30
35
40
45
50
0
1
2
3
4
5
6
7
8
9
10