Day 10

- (Concept Question)
- Matlab plotting concepts
- Plotting data
- Adding plots to technical reports
- Plotting results from formulas
- Plotting multiple curves
- `close all`
- (Exercises)

Matlab plotting concepts

We can only plot *vectors* in Matlab

- We don't (usually) plot one point at a time
- We don't plot equations

The basic command is

```
plot(x, y)
```

where x and y are both vectors, and are the *same length*

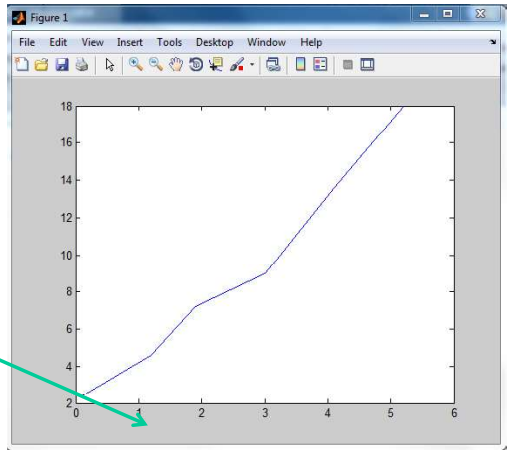
Plotting data

A simple script that makes a plot of data

Day10_InClassExample1.m

```
1 -    clc
2 -    clear variables
3 -    x = [0.1   1.2    1.9    3.0    4.1    5.2];
4 -    y = [2.4   4.6    7.2    9.0    13.6   18.0];
5 -    plot(x,y);
```

Notice that x values come first in the plot command



Plotting data

This plot is not complete because it does not have axis labels or a title.

Two ways to add these features:

1. Add additional commands to the script; or
2. Use the interactive edit menu

Plotting data

1. Adding axis labels and a title by adding additional commands to the script:

Day10_InClassExample1.m

```
1 -    clc
2 -    clear variables
3 -    x = [0.1   1.2    1.9    3.0    4.1    5.2];
4 -    y = [2.4   4.6    7.2    9.0    13.6   18.0];
5 -    plot(x,y);
6 -    xlabel('X Axis Descriptor (units!)');
7 -    ylabel('Y Axis Descriptor (units!)');
8 -    title('A Decent Plot Title');
```


Plotting data

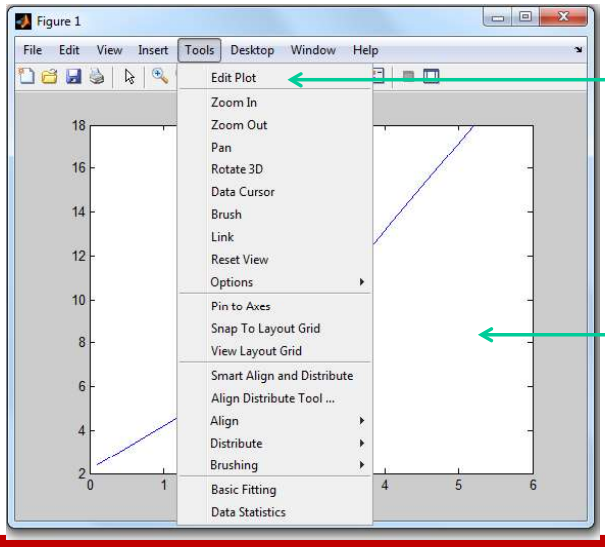
Many other good script commands for plotting:

`grid`	makes a grid on the plot
`plot(x,y,'o');`	plots the points with just circle symbols – no line
`plot(x,y,'r-');`	plots the points with a red solid line
`plot(x,y,'r--o');`	plots the points with a red dashed line and red circle symbols

Type “help plot” for more options!

Plotting data

2. Adding axis labels and a title by using the interactive edit menu



1. Click on Tools, Edit Plot

2. Double-click in the white part of the plot

Plotting data

Once you are in the plot editor you can poke around and add labels to the plot.

You can also double-click on the curve and add symbols, change the color, and change the line to a dashed line.

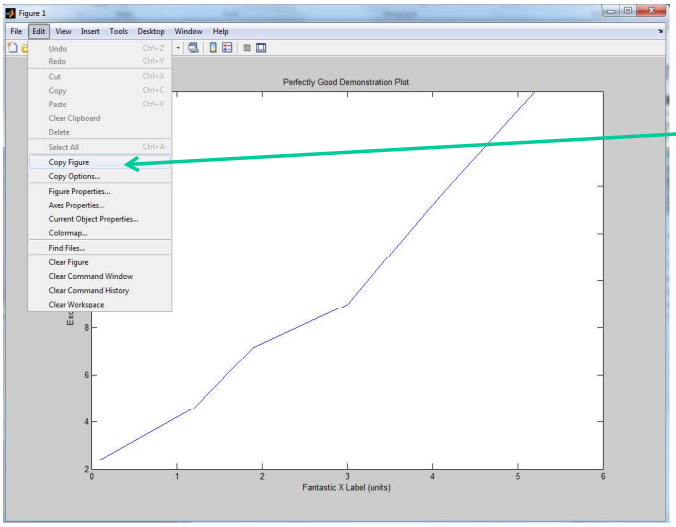
Plotting data

You can choose which approach to use to modify the plot to suit your needs:

- If you want to be able to reproduce exactly the same plot the next time you run the script, put the commands in the script.
- If we ask you to turn in a plot, put the commands in the script.
- If you want to experiment with different ways to visualize the data then the plot editor is useful.

Adding plots to technical reports

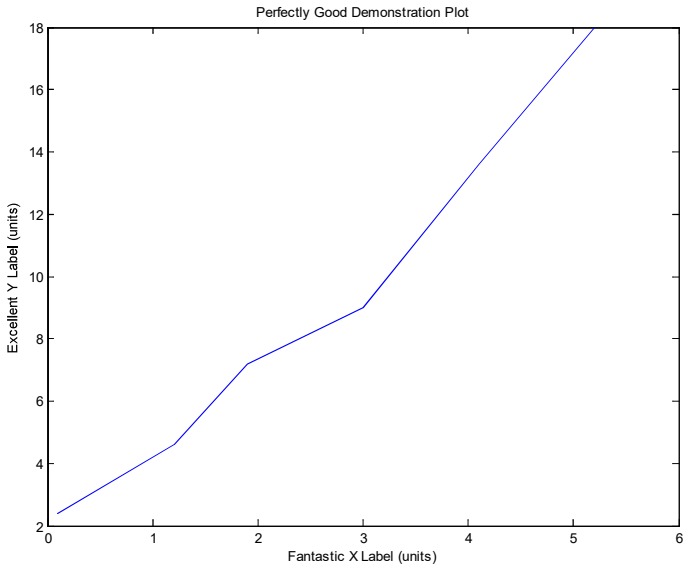
We often want our (very professional) plots to go in a paper report or on a slide.



Use Edit → Copy Figure to copy the final figure. Then paste the figure into your Word or Powerpoint document

Adding plots to technical reports

Edit →Copy Figure gives a high-quality copy of the figure



Plotting results from formulas

To plot a formula or equation, we *must first* create *vectors* to hold the independent (x-axis) values and the dependent (y-axis) results.

Often we use a for loop to create the vectors.

The script on the next slide shows how we would do this to create a sine curve.

Plotting results from formulas

Day10_InClassExample2.m

```
1 -    clc
2 -    clear variables
3 -    index=0;
4 -    for theta=0:1:360
5 -        index=index+1;
6 -        angle(index)=theta;
7 -        sine_of_angle(index)=sind(theta);
8 -    end
9 -    plot(angle,sine_of_angle)
10 -   xlabel('Angle (degrees)');
11 -   ylabel('Sine of the Angle');
12 -   title('A Typical Sine Curve');
```

Use a loop and a recursive assignment to create two vectors: one for the angle and one for its sine

Make the plot: angle is first because it goes on the x-axis

Notice that we plot angle– not theta– because angle is a vector and theta is not a vector

Plotting multiple curves

If we want to display two curves on a single graph, we must first create vectors for each of them. Then we issue the plot command with *two pairs* of vectors:

```
plot(angle,sine_of_angle,angle,cosine_of_angle)
```

x, y x, y

Plotting multiple curves

To plot each curve with a different style and color of line, put that information *after each* pair:

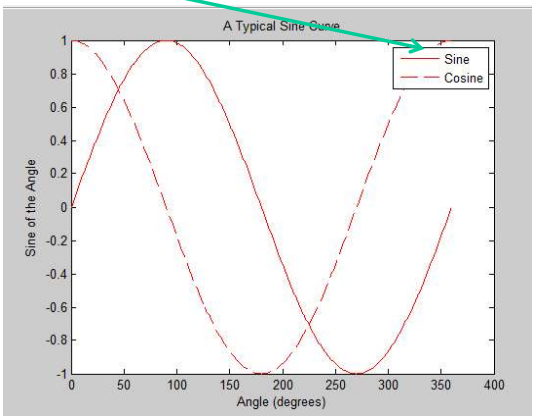
```
plot(angle,sine_of_angle,'r-',angle,cosine_of_angle,'r--')
```


Plotting multiple curves

When you plot multiple curves you should add a legend to the figure with the legend command:

```
legend('Sine','Cosine')
```

The legend command should come after the plot command in your script



close all

To close all existing figures and start fresh, add `close all` to the beginning of your scripts:

```
1 -    clc
2 -    clear variables
3 -    close all
```

This also will make your new figure “pop up” when you create it, so you don’t have to hunt it down among all your open Matlab windows.