

# THE UNIVERSITY OF MELBOURNE

## CVEN30008 Risk Analysis

### MATLAB *Introduction*

## Starting MATLAB

After logging into your student account, you can enter MATLAB by double-clicking on the MATLAB icon. The MATLAB desktop as shown Figure 1 will appear after you start MATLAB.

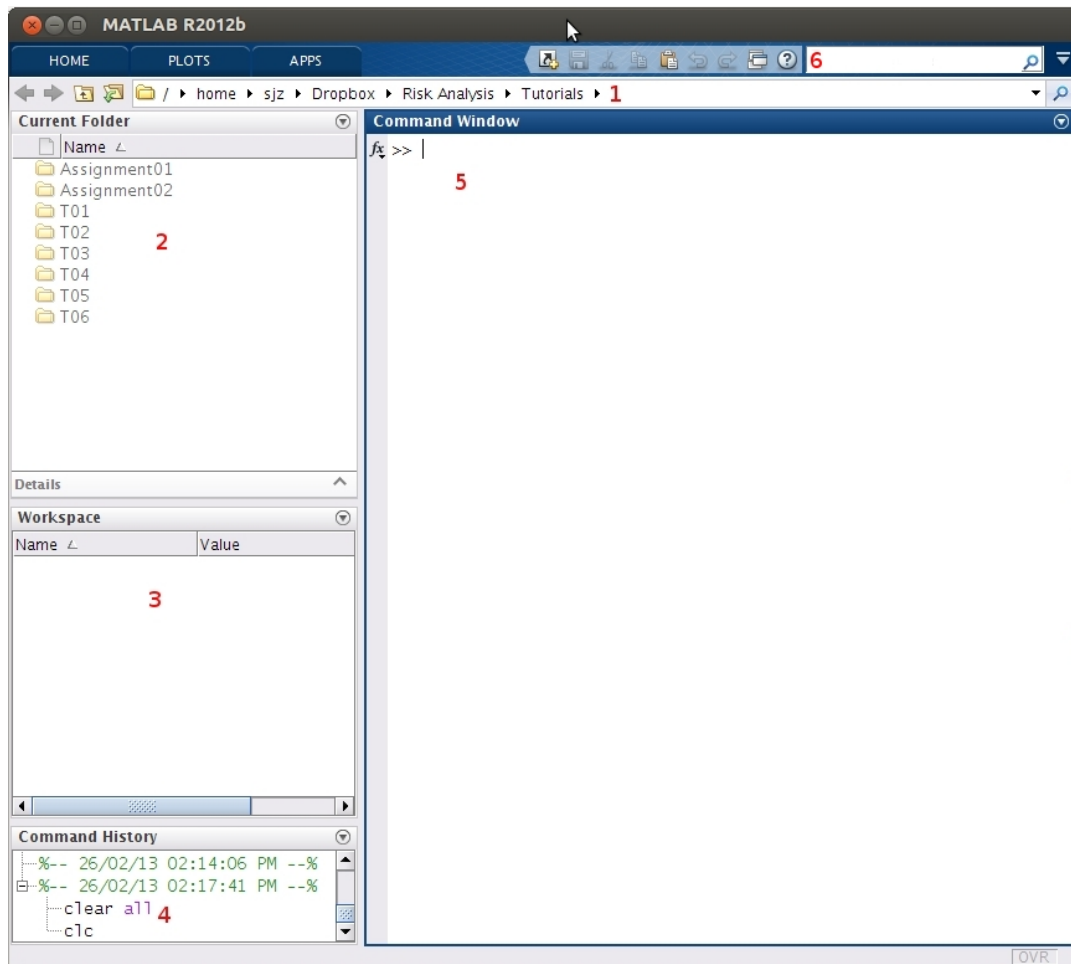


Figure 1: The MATLAB desktop.

1. The CURRENT DIRECTORY: view or change current directory
2. The CURRENT FOLDER: view files in the current folder (Note: you can set up your file system as shown in Figure 1)
3. The WORKSPACE: view current available variables
4. The COMMAND HISTORY: view or execute previously run functions
5. The COMMAND WINDOW: enter MATLAB functions at command-line prompt
6. The SEARCH WINDOW: search for function help

## Preliminaries [1]

5 mins

This section gives an overview of basic commands and operations in MATLAB.

### Need help with MATLAB functions?

- Getting help from the command line, eg. type the following in the command window:

```
» help plot
```

- Getting help using the MATLAB Help Browser by selecting Help → MATLAB Help

### Variables

- Begin with an alphabetic character: `a`
- Case sensitive: `a`, `A`
- Data type detection: `a=3`, `a=3.5`, `a='hi'`
- Default output variable: `ans`
- Built-in variables: `pi`, `i`, `j`, `Inf`
- To remove variables: `clear`
- To list variables: `who`
- Special Characters: `[ ] ( ) { } = ' . ... , ; : % ! @`
- Avoid using special characters, operators, keywords, function names and built-in variables in variable name.

### Operators

- Arithmetic Operators: `+` `-` `*` `/` `\` `^` `'`
- Arithmetic Operators (element-wise): `.*` `./` `.\` `.^` `.'`
- Relational Operators: `<` `>` `<=` `>=` `==` `~=`
- Logical Operators: `&` `|` `~` `&&` `||` `true` `false`
- Operator precedence: `()` `{}` `[]` → Arithmetic → Relational → Logical

### Matrix

- Row vector

```
» vec_r1 = [3 1 4 1 5 9]
» vec_r2 = [0:10]
» vec_r3 = [0:2:10]
```

- Column vector

```
» vec_c1 = [3; 1; 4; 1; 5; 9]
» vec_c2 = vec_r2' %transpose of the row vector 'vec_r2'
» vec_c3 = [0:2:10]
```

- Matrix

```
» mat_1 = [3 1 4; 1 5 9]
» mat_2 = [vec_r1; vec_r3]
```

- Accessing elements

```
» vec_r1(1)
» vec_r1(4:end)
» mat_1(2,3)
» mat_1(:,2:3)
```

## Plotting

5 mins

The MATLAB plotting procedure is to take a vector of  $x$ -coordinates,  $x = (x_1, \dots, x_N)$ , and a vector of  $y$ -coordinates,  $y = (y_1, \dots, y_N)$ , locate the points  $(x_i, y_i)$ , for  $i = 1, 2, \dots, N$  and then join them by straight lines. Both  $x$  and  $y$  need to be in an identical vector form (both are row vectors or column vectors of the same length).

### Basic plotting

The six measurements of signal A in a 6 seconds interval is given as the vectors  $\text{time} = (1, 2, 3, 4, 5, 6)$  and  $\text{signal\_A} = (300, 100, 250, 400, 350, 200)$ . Produce the graph for signal A with the given data.

```
» time = [1, 2, 3, 4, 5, 6];
» signal_A = [300, 100, 250, 400, 350, 200];
» plot(time, signal_A);
```

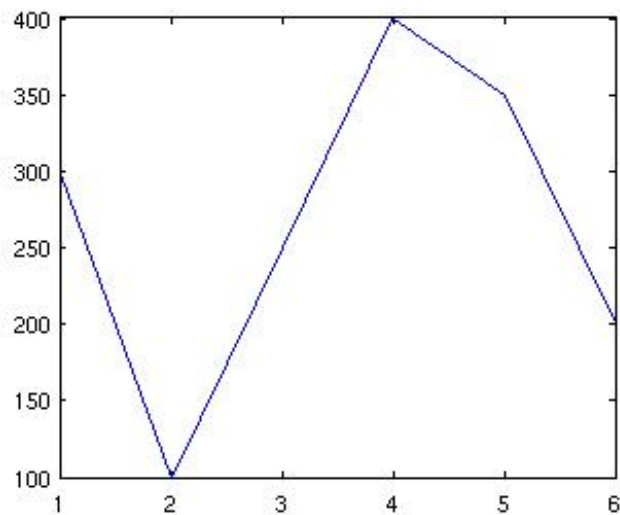


Figure 2: Signal A discrete plot.

## Adding titles, axis labels, and annotations

Signal B follows the equation:  $\text{signal\_B} = 150 \times \sin(\text{time}) + 350$ . Produce the graph for signal B over a period of  $2\pi$  seconds.

```

» time = 0:pi/10:2*pi;           %create time vector with an increment of pi/10
» signal_B = 150*sin(time)+350; %create signal B
» plot(time,signal_B,'r*-');      %plot signal B with LineSpec r*-
» xlabel('time (seconds)');        %label x-axis
» ylabel('Signal B');              %label y-axis
» title('Signal B vs Time');       %label title
» legend('Signal B');              %display legend
» axis([0 2*pi 150 550]);         %control axis scaling and appearance
» grid on;                        %turn on grid lines

```

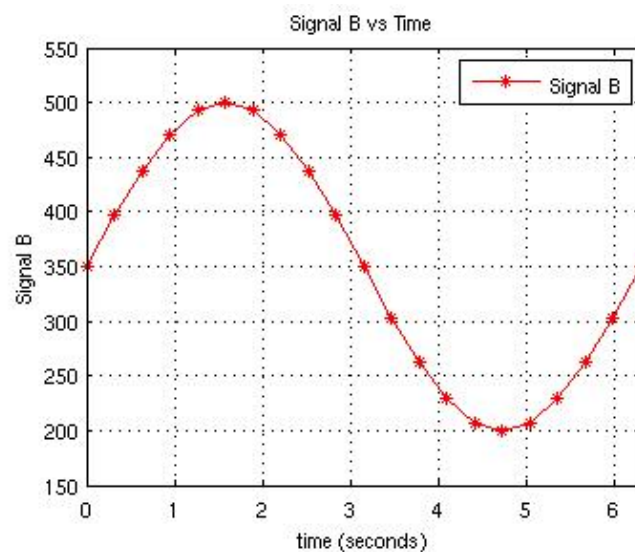


Figure 3: Signal B vs time.

By default, the data points will be joined with line segments in MATLAB. If there are plenty of data points, the curve will be smooth. We can customise the line by `plot(x,y,LineSpec)`. *LineSpec* is a string, defined by quotation marks. It may contain line style, color, width, marker type, marker size. To find additional information, type `doc plot`.

SYMBOL	COLOR	SYMBOL	LINE STYLE	SYMBOL	MARKER
k	Black	—	Solid	+	Plus sign
r	Red	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
g	Green	-.	Dash-dot	.	Point
c	Cyan	none	No line	x	Cross
m	Magenta			s	Square
y	Yellow			d	Diamond

Figure 4: LineSpec for plot function.

## Multiple data sets in one plot

Signal C follows the equation:  $\text{signal\_C} = 150 \times \cos(\text{time}) + 350$ . Produce the graph for signal B and C over a period of  $2\pi$  seconds.

```

» time = 0:pi/10:2*pi;
» signal_B = 150*sin(time)+350;
» signal_C = 150*cos(time)+350;
» plot(time,signal_B,'r*-');
» hold on;                                %hold current graph
» plot(time,signal_C,'bo-')
» hold off;                               %returns to the default plot mode
» xlabel('time (seconds)');
» ylabel('Signals');
» title('Signals vs Time');
» legend('Signal B','Signal C');
» axis([0 2*pi 150 550]);
» grid on;

```

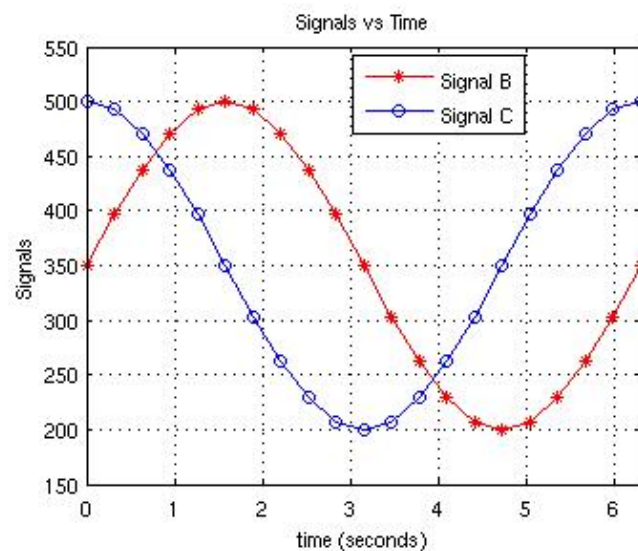


Figure 5: Signal B and C vs time.

## References

- [1] V. Ivanova. Matlab quick start. <http://web.mit.edu/ist/topics/math/quickstartmatlab.pdf>, Last Accessed On: 20 July 2010.