Introduction to MATLAB

Instructor:

Muhammad Azeem Iqbal

Senior Lab Instructor

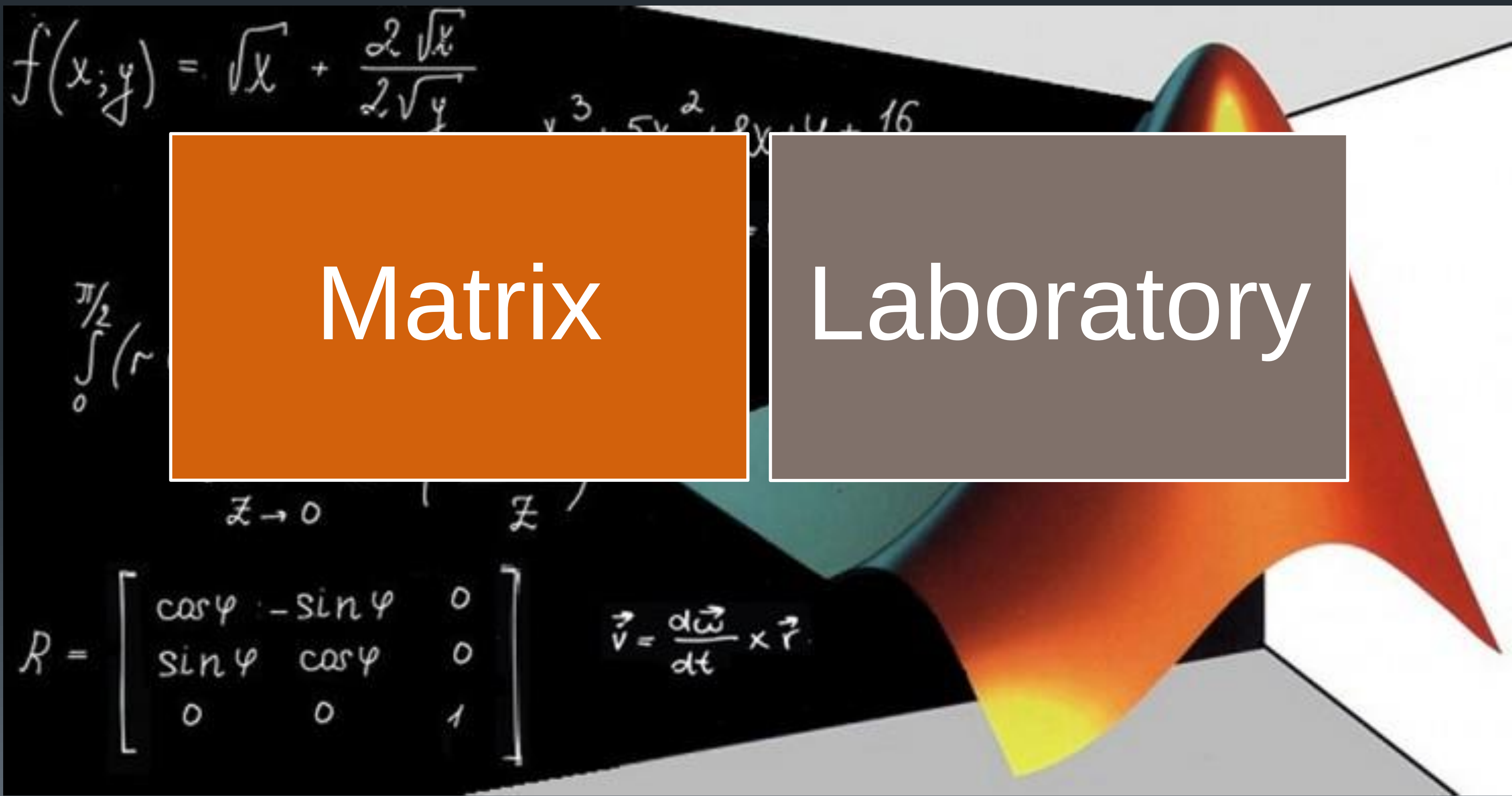
Centre for Experimental Physics Education (CEPE)

Lahore University of Management Sciences (LUMS)

Lahore, Pakistan

Contents

What MATLAB stands for?



Matrix

Laboratory

Who invented MATLAB?

MathWorks®

Dr. Cleve Moler

Chairman & Founder
PhD. Mathematics
Stanford University

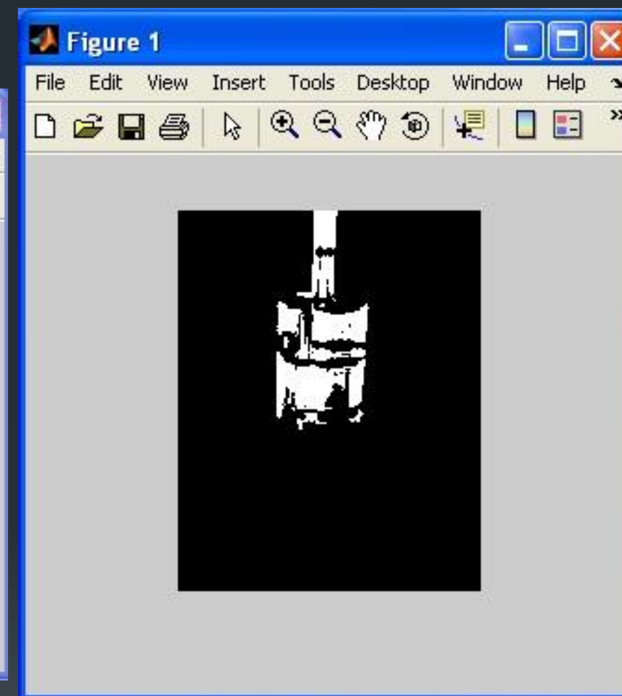
Jack Little

President & Co-Founder
MS Electrical Engineering
Massachusetts Institute of Technology

Introduction to MATLAB

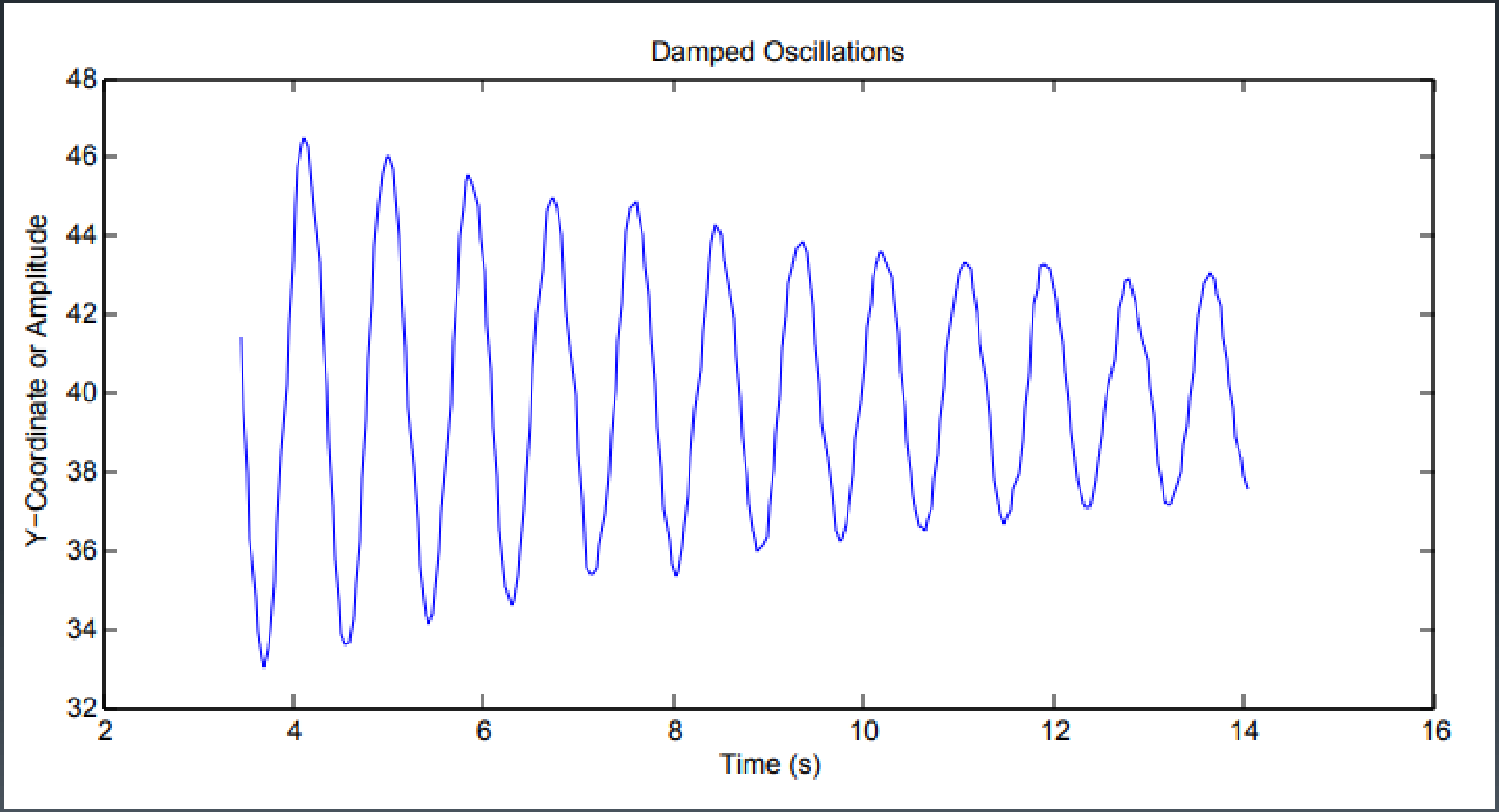
- Math and computation
- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including Graphical User Interface building
- Teaching of linear algebra, numerical analysis, and is popular amongst scientists involved in image processing.

Introduction to MATLAB



Observing simple harmonic motion using a webcam (1.1)

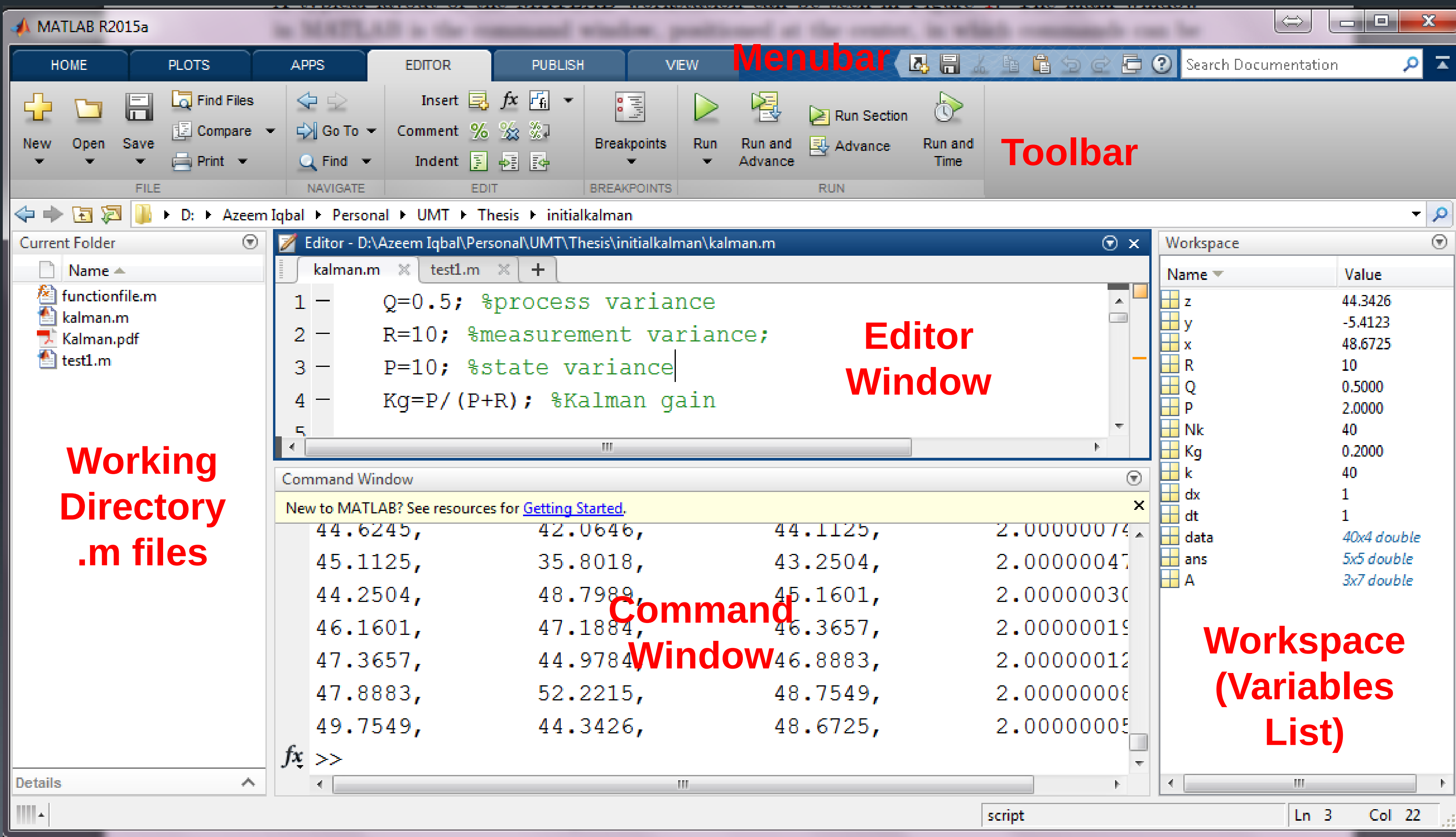
Introduction to MATLAB



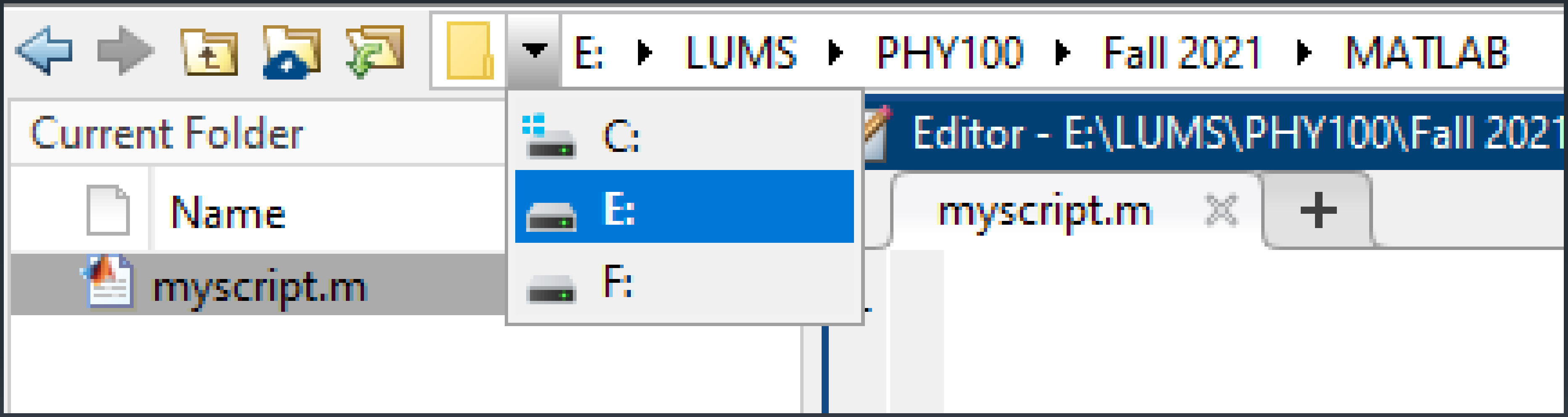
Introduction to MATLAB

Layout

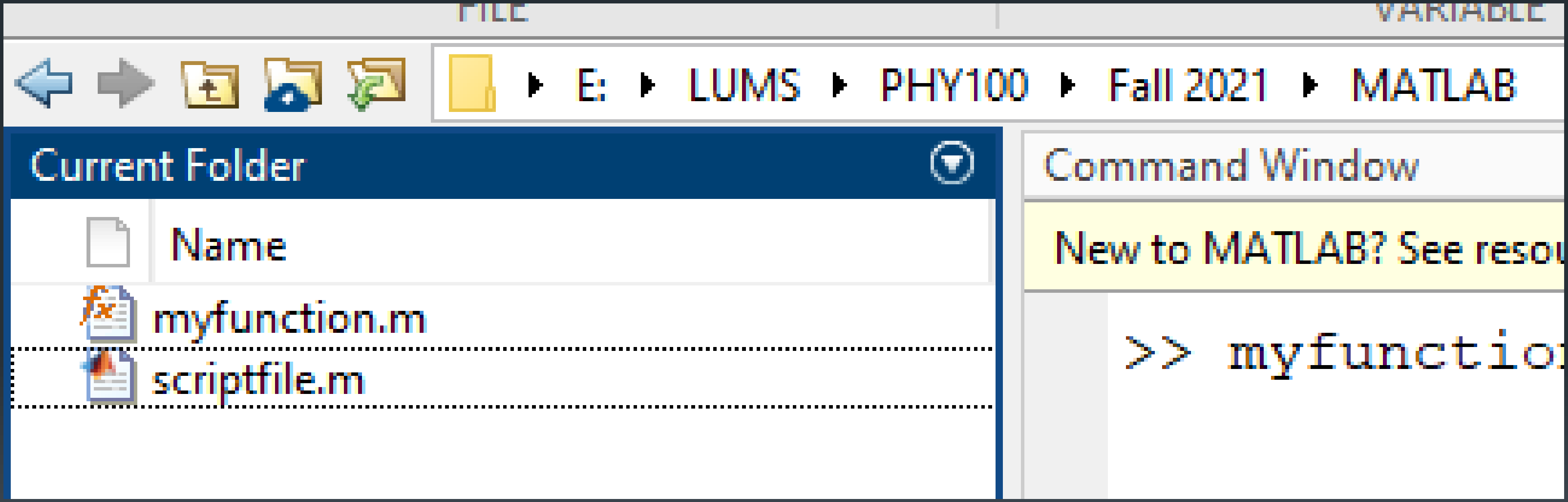
Layout


MATLAB R2015a
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Menubar
Search Documentation
New
Open
Save
Find Files
Compare
Print
FILE
Go To
Find
NAVIGATE
Insert
Comment
Indent
EDIT
Breakpoints
BREAKPOINTS
Run
Run and Advance
Run Section
Advance
Run and Time
RUN
Toolbar
D: Azeem Iqbal Personal UMT Thesis initialkalman
Current Folder
Name
functionfile.m
kalman.m
Kalman.pdf
test1.m
Working Directory .m files
Details
Editor - D:\Azeem Iqbal\Personal\UMT\Thesis\initialkalman\kalman.m
kalman.m
test1.m
Q=0.5; %process variance
R=10; %measurement variance;
P=10; %state variance
Kg=P/(P+R); %Kalman gain
Editor Window
Command Window
New to MATLAB? See resources for Getting Started.
45.1125, 35.8018, 43.2504,
44.2504, 48.7989, 45.1601,
46.1601, 47.1884, 46.3657,
47.3657, 44.9784, 46.8883,
47.8883, 52.2215, 48.7549,
49.7549, 44.3426, 48.6725,
>>
Command Window
Workspace
Name
Value
z 44.3426
y -5.4123
x 48.6725
R 10
Q 0.5000
P 2.0000
Nk 40
Kg 0.2000
k 40
dx 1
dt 1
data 40x4 double
ans 5x5 double
A 3x7 double
Workspace (Variables List)
script
Ln 3
Col 22

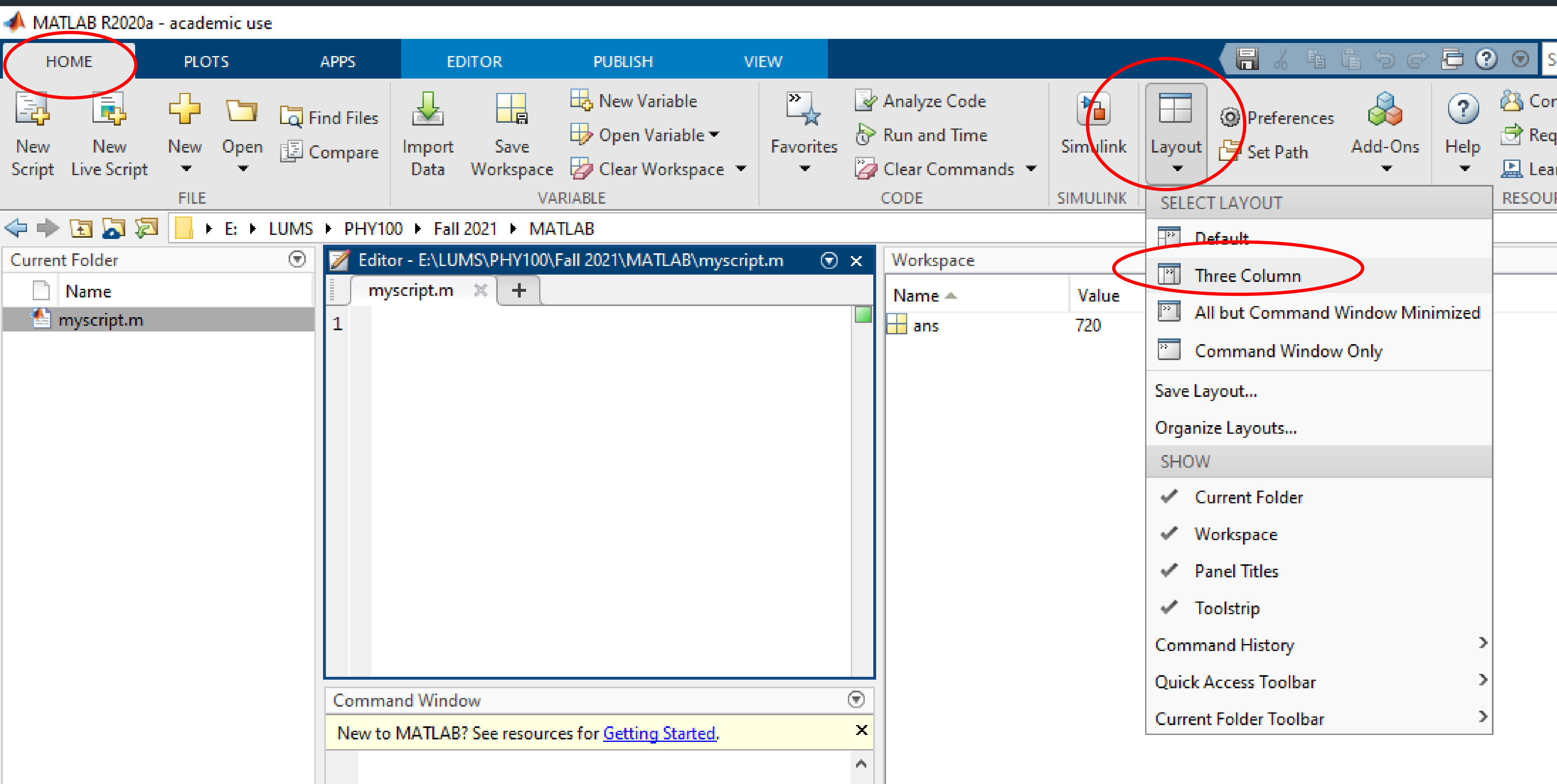
Current Folder Toolbar



Current Folder Toolbar



Don’t like the layout?



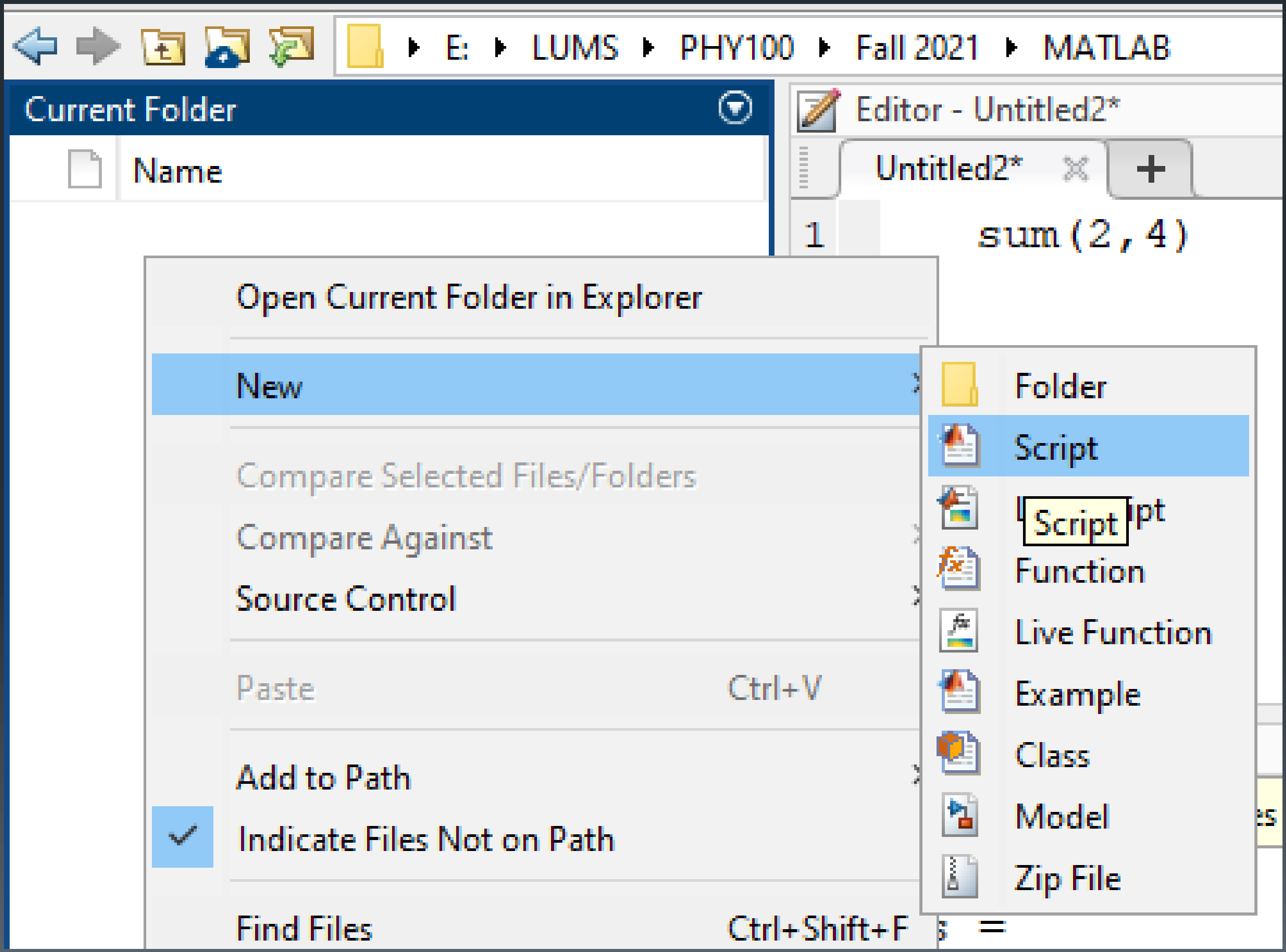
.m files

- MATLAB allows writing two kinds of program files:
- **Scripts** − script files are program files with ***.m extension***. In these files, you write series of commands, which you want to execute together. Scripts do not accept inputs and do not return any outputs. They operate on data in the workspace.
- **Functions** − functions files are also program files with ***.m extension***. Functions can accept inputs and return outputs. Internal variables are local to the function.

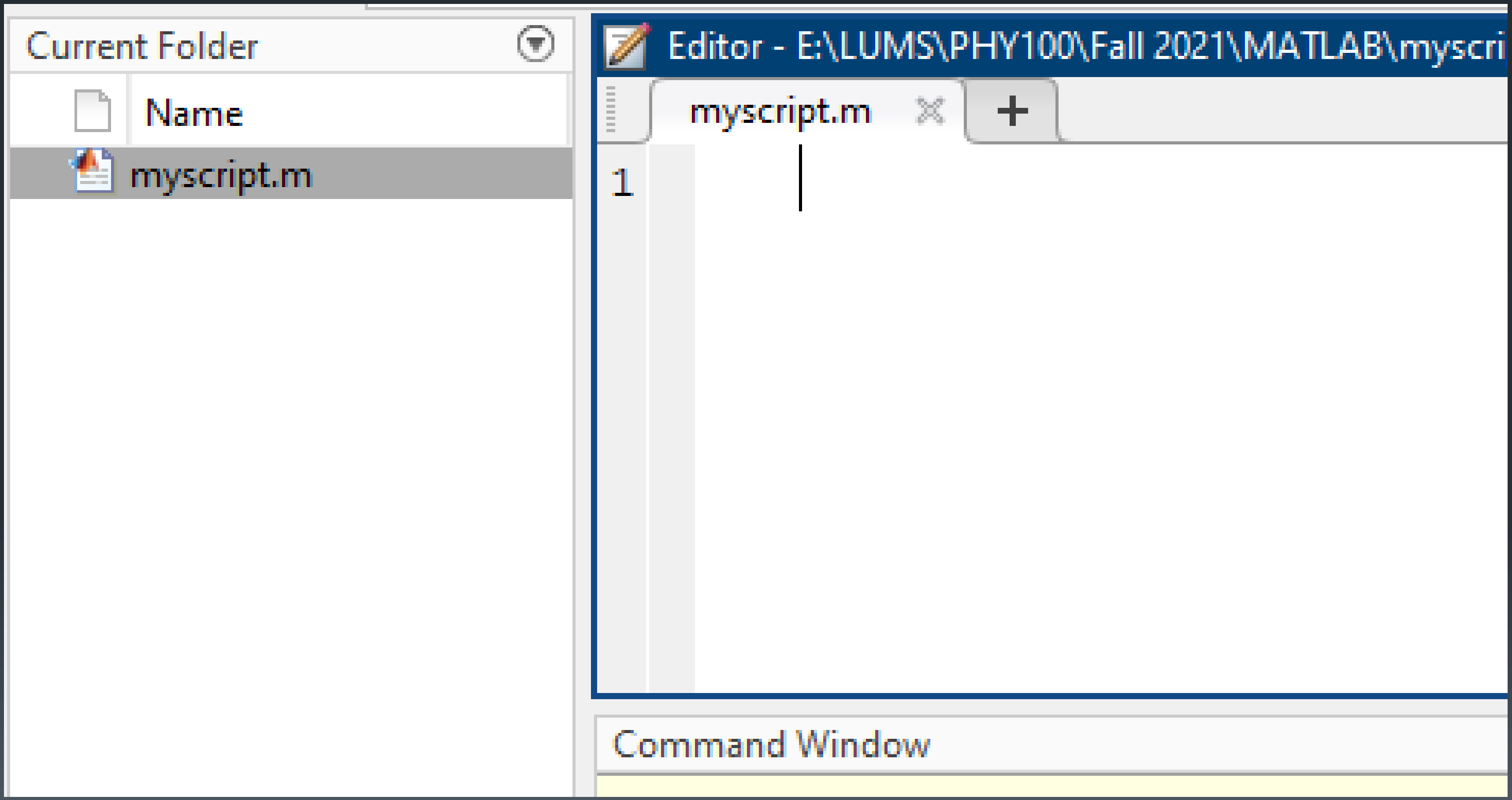
Introduction to MATLAB

Script Files

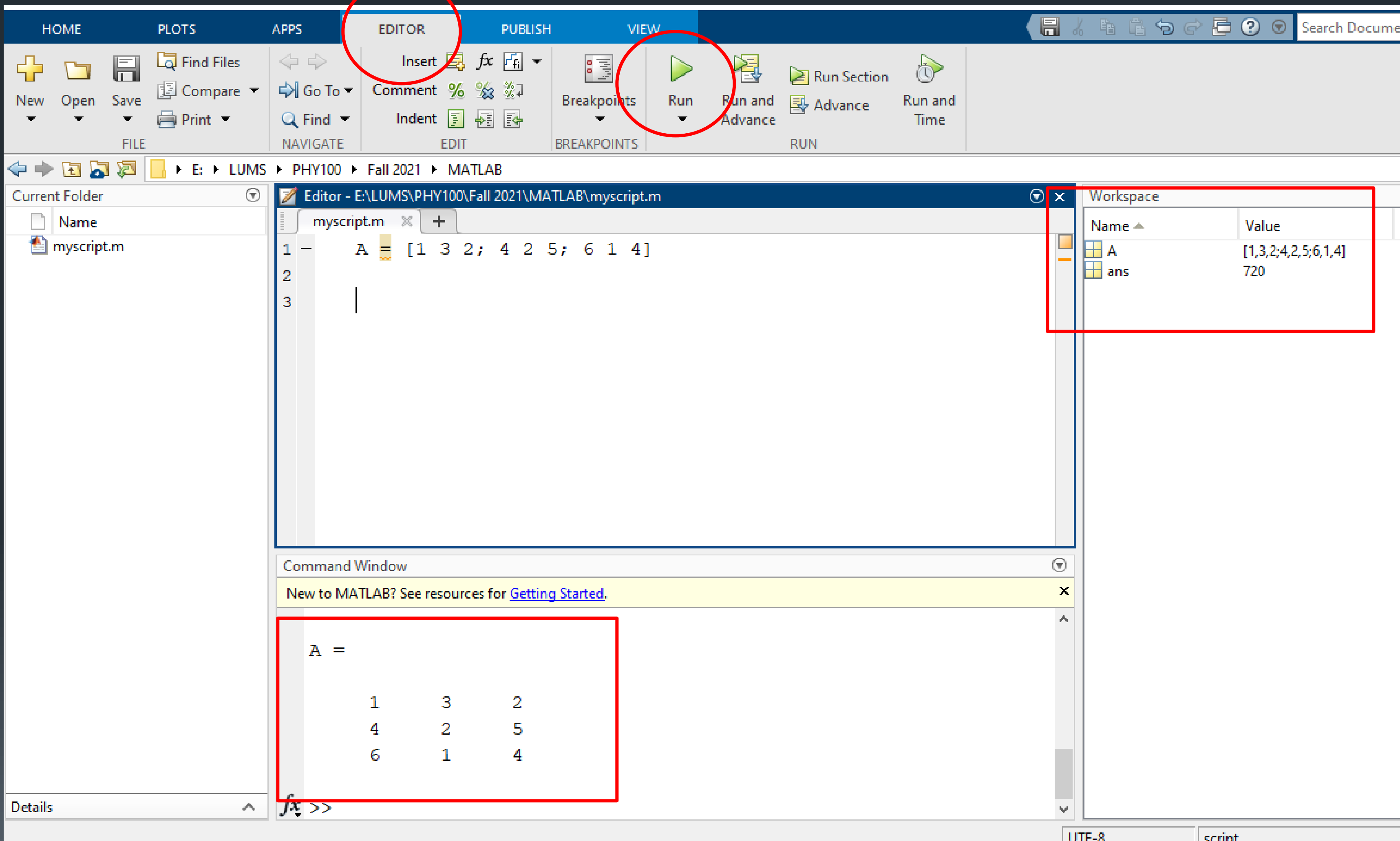
How to create Script files?



Script files



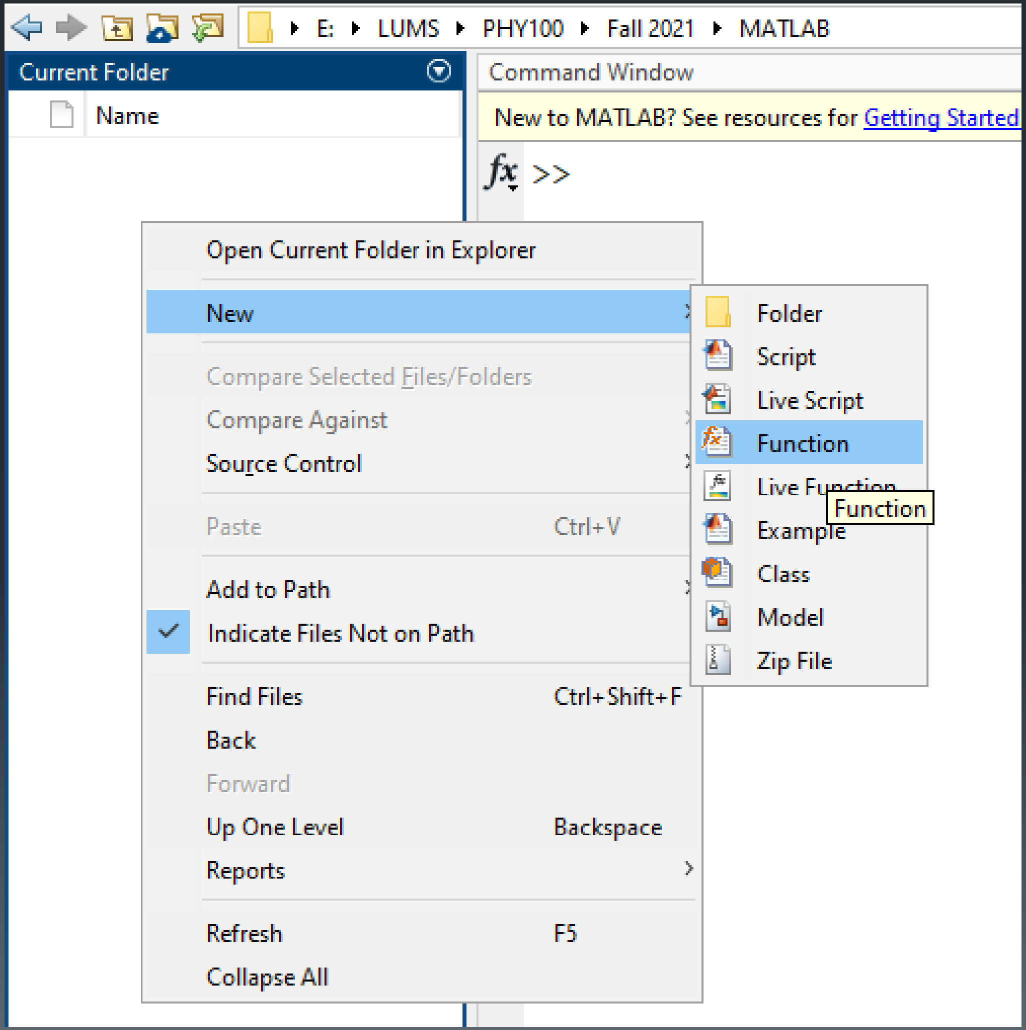
Press “Run” to Execute



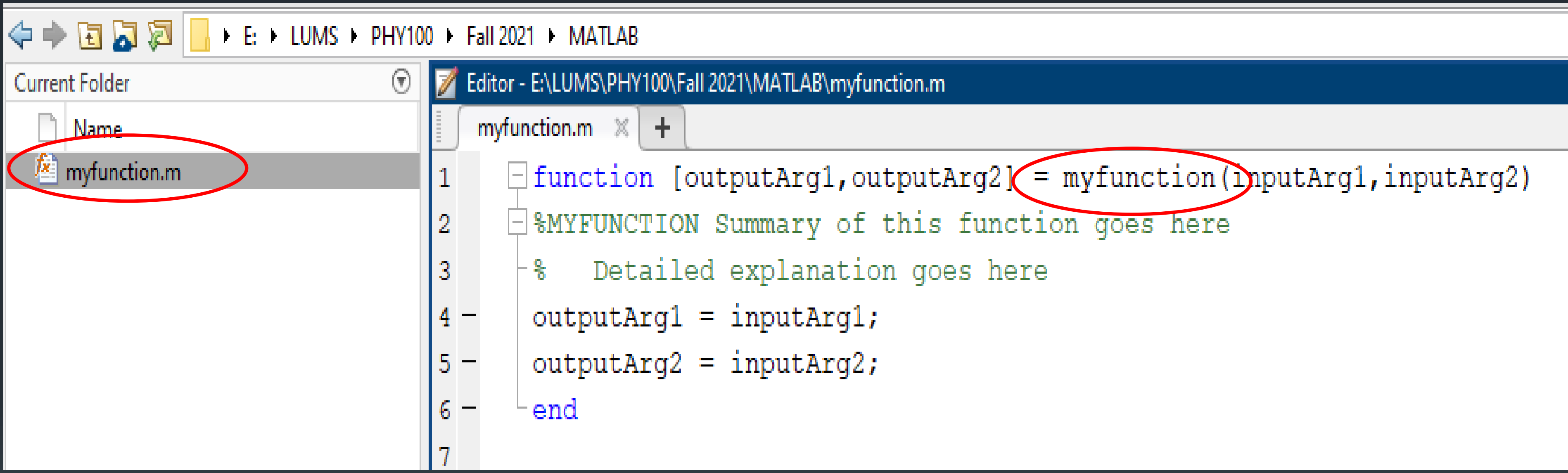
Introduction to MATLAB

Function Files

How to create a Function file?

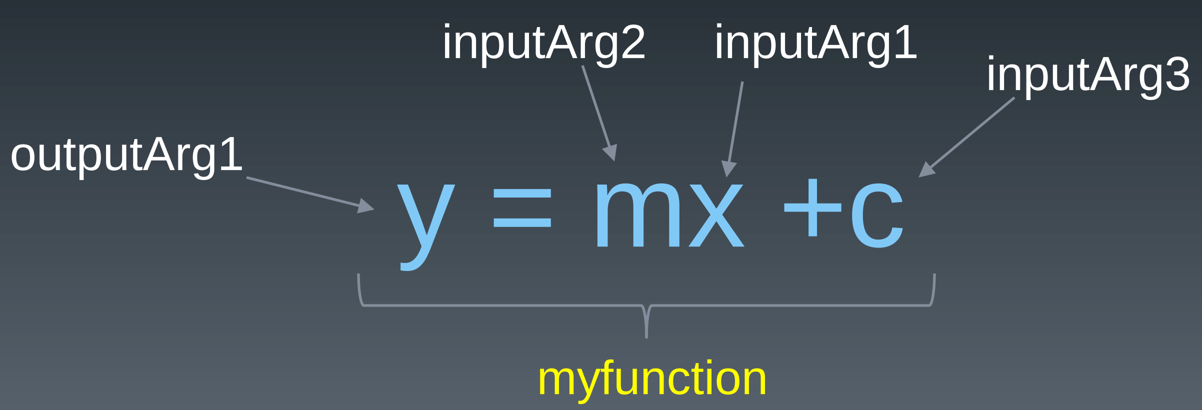


Structure of the function file



myfunction.m in **Editor** Window

1. function [outputArg1,outputArg2] = **myfunction**(inputArg1,inputArg2)
2. %MYFUNCTION Summary of this function goes here
3. %Detailed explanation goes here
4. outputArg1 = inputArg1;
5. outputArg2 = inputArg2;
6. end



How does a straight-line equation function file look like?

$$y = mx + c$$

```
function y = myfunction(m,x,c)
    y = mx + c
end
```

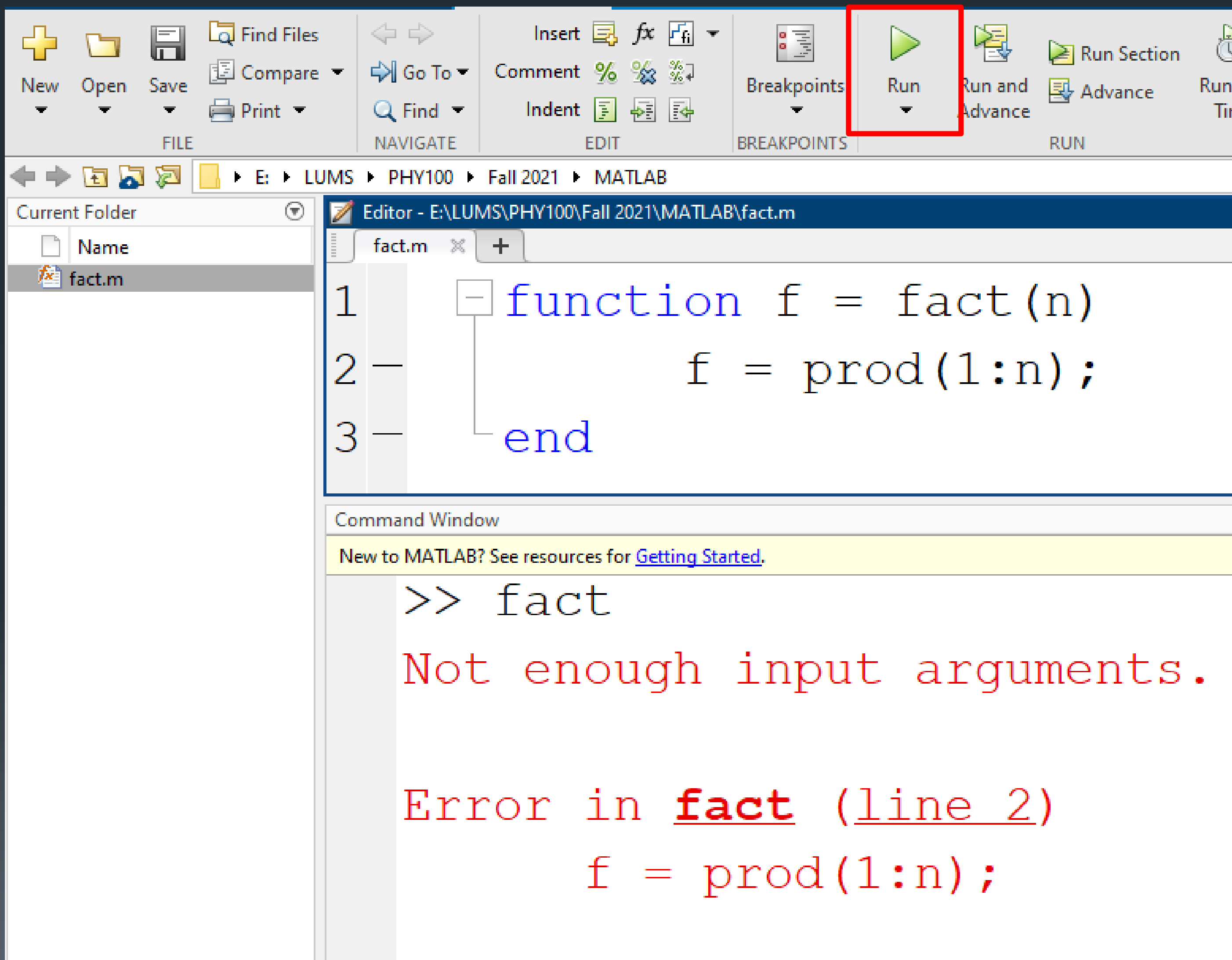
Functions provide more flexibility, primarily because you can pass input values and return output values. For example, this function named fact computes the factorial of a number

But what if …?

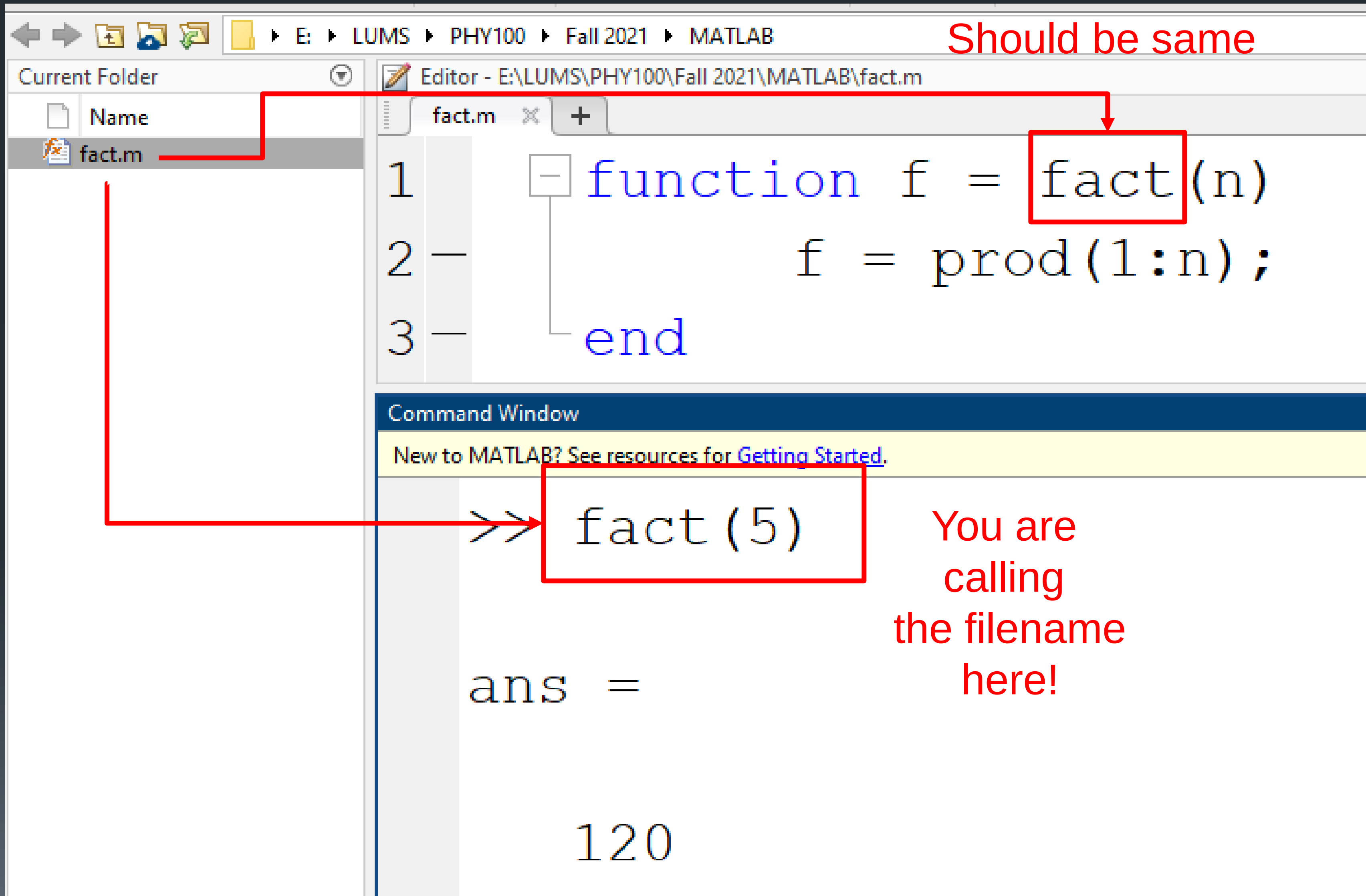
Create a function file named fact that computes the **factorial** of a number (n) and returns the result (f).

```
function f = fact(n)
        f = prod(1:n);
end
```

If you press “RUN”, error!



All names should be same …



Basic arithmetic operators

Introduction to MATLAB

Basic Arithmetics

Functional Names	Operators
Addition	+
Subtraction	-
Multiplication	*
Division	/
Power	^

Basic Arithmetics

Function Name	Command
Square root	sqrt()
Average	mean()
Standard Deviation	std()
Exponent	exp()
Sine	sin()
Natural Log	log()

Basic Arithmetics

a = 5;   b = 4;

Summation	=	a + b
Difference	=	a – b
Product	=	a * b
Division	=	a / b
Power	=	a^2
Square root	=	sqrt(b)
Exponent	=	exp(a)
Sine	=	sin(a)
Natural	=	log(b)

Basic Arithmetics

Concept of precedence:

P E M D A S

			Order of precedence	
P	=	Parentheses	1	
E	=	Exponents	2	
M	=	Multiplication	3	Which ever comes first in left to right order of equation
D	=	Division	4	
A	=	Addition	5	
S	=	Subtraction	6	

Basic Arithmetics

Solve:

$$6 \div 2\,(2 + 1)$$

What is the answer

1 or 9?

The correct answer is “9”

Introduction to MATLAB

Creating vectors and matrices

Vector and Matrices

- 1- Dimensional Vector

```
x = [1 2 5 1]    Row matrix
```

- Transpose   `y=x’`   `Column matrix`

- 2 – Dimensional Vectors

```
x = [1 2 3 ; 5 1 4 ; 3 2 -1]
```

$$x = \begin{pmatrix} 1 & 2 & 3 \\ 5 & 1 & 4 \\ 3 & 2 & -1 \end{pmatrix}$$

Vector and Matrices

- evenlist = [2 4 6 8 10 12 14 16 18];
- evenlist2 = 2:2:18;

Vector Addition

- summation = evenlist + evenlist2;

Vector Multiplication

- summation = evenlist .* evenlist2;

Matrix arithmetic

A = [2 4 6; 1 3 5; 7 9 11]; B = [1 2 3; 4 5 6;8 9 10];

$$\begin{pmatrix} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 7 & 9 & 11 \end{pmatrix} + \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & 10 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 7 & 9 & 11 \end{pmatrix} - \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & 10 \end{pmatrix}$$

Multiplication

$$a = \begin{pmatrix} 5 & 8 & 9 \\ 2 & 4 & 6 \\ 1 & 3 & 5 \end{pmatrix} \qquad b = \begin{pmatrix} 4 & 6 & 7 \\ 2 & 1 & 3 \\ 5 & 3 & 8 \end{pmatrix}$$

a * b	Matrix Multiplication
a .* b	Element by Element Multiplication

Extracting Values from Matrices

a = [2 4 6; 1 3 5; 7 9 11];

$$\begin{pmatrix} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 7 & 9 & 11 \end{pmatrix}$$

- You may want to extract a few values using:

  a(2,2) - Extracts "3" from the above matrix

  a(2,:) - Extracts 2nd row from the above matrix

  a(:,2) - Extracts 2nd column from the matrix

  a(2,[1 3]) - Extracts 1st and 3rd element from 2nd row

Data manipulation using “for” Loops

Basic structure:

```
for (condition)
statements
end
```

Generate the first 17 Fibonacci numbers

```
n=[1 1];

  for k=1:15
      n(k+2) = n(k+1) + n(k);
  end
```

Let’s practice!

Solve the first exercise …

Contents

- ✓Introduction to MATLAB
- ✓Layout
- ✓Basic arithmetic operations
- ✓Creating vectors and matrices
- ✓Matrix arithmetic
- ✓Extracting elements from matrices
- Data manipulation
  - Introduction to “for” Loops
- Graphs and plotting
  - Introduction to plotting
  - Multiple plots
  - Resolution of graph
- Curve fitting
  - Least square curve fitting of linear data
  - Fitting and plotting with error bars

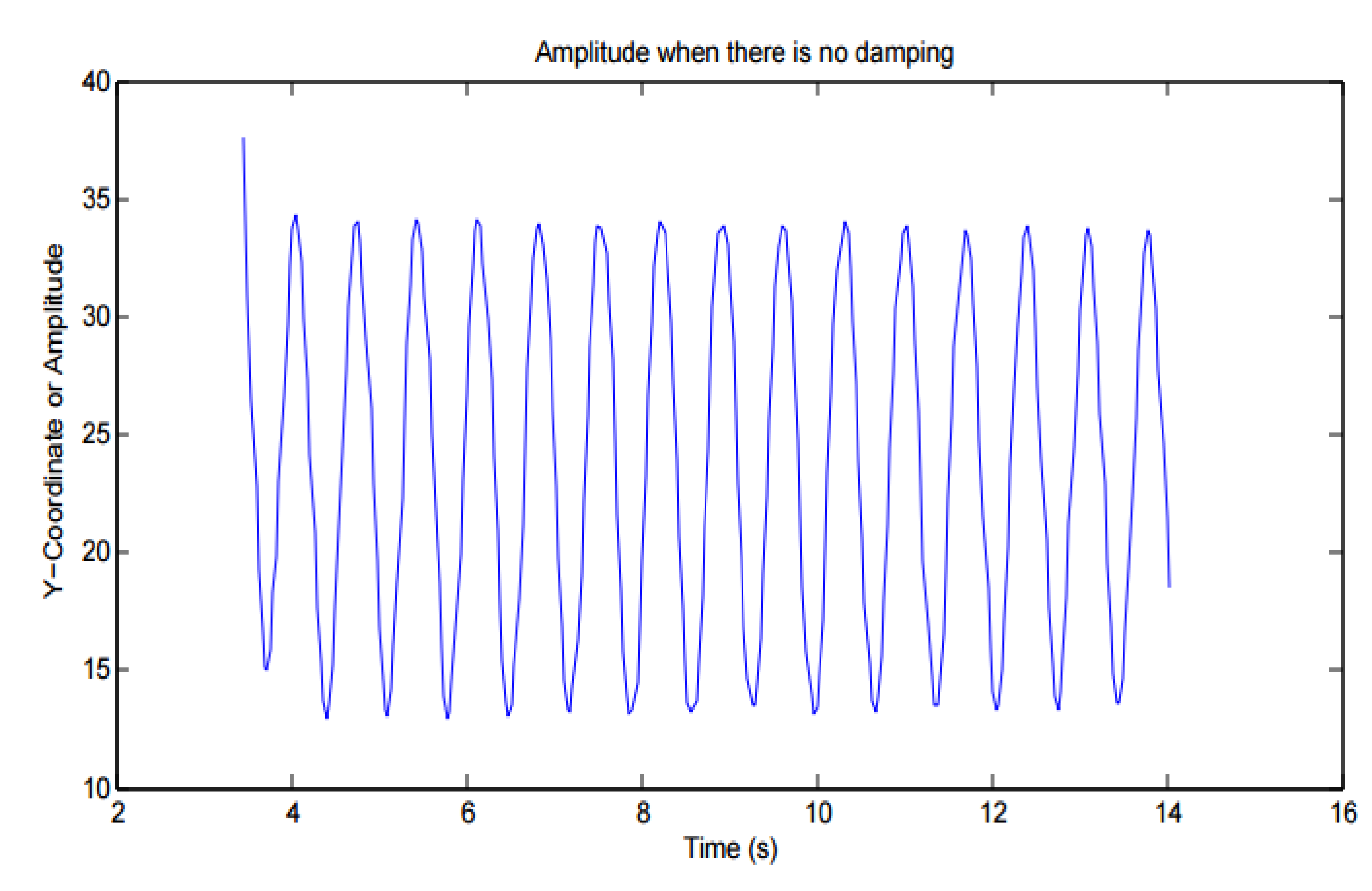
Introduction to MATLAB

Graphs and plotting

Graphs and plotting

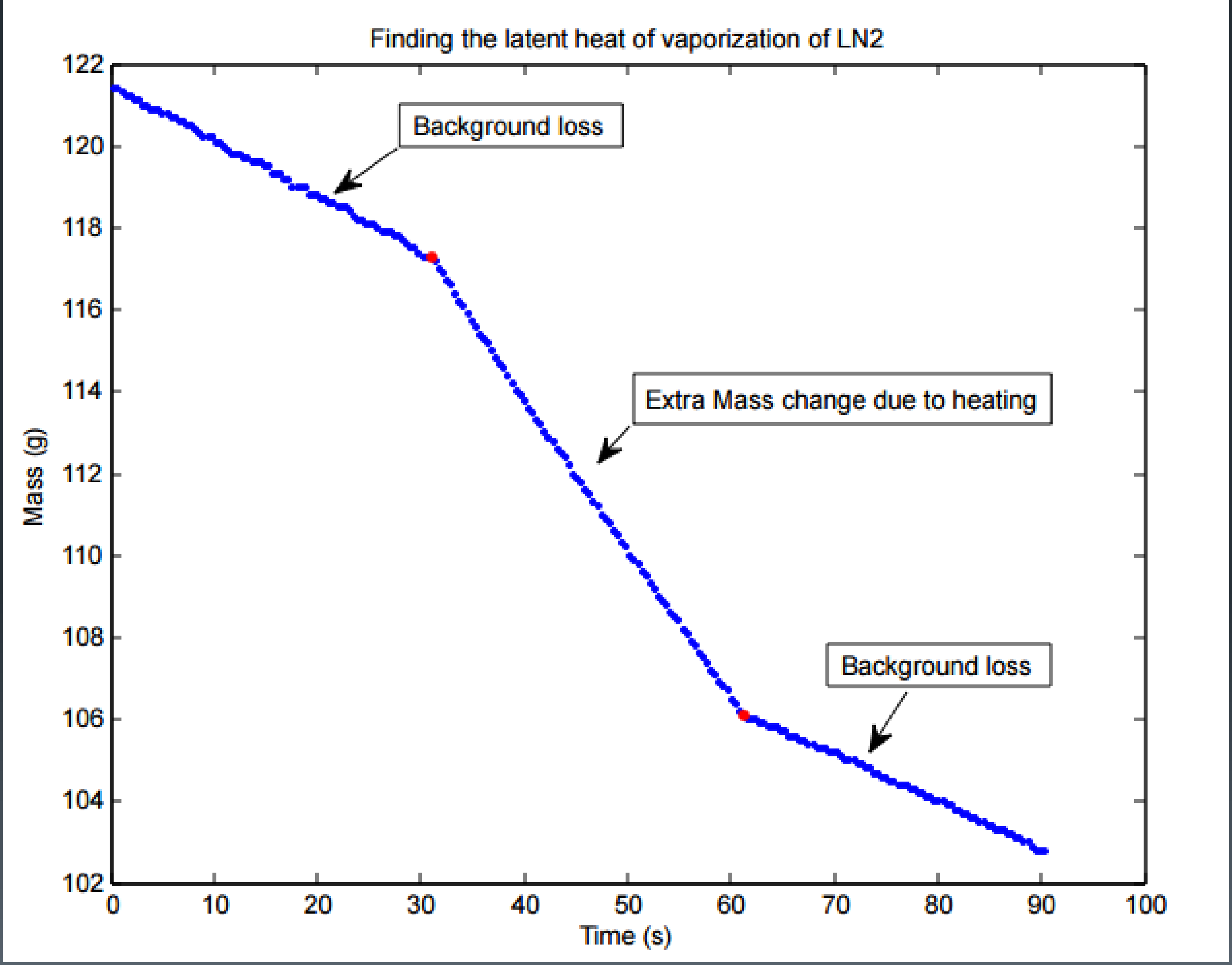
1. They act as visual aids indicating how one quantity varies when the other quantity is changed, often revealing subtle relationships.
2. Determine slopes and intercepts
3. Compare theoretical predictions and experimentally observed data.

Graphs and plotting



Sample Dataset

Sr#	Time (s)	Mass (g)
1	0.34	121.4
2	0.74	121.4
3	1.13	121.3
4	1.52	121.2
5	1.92	121.2
6	2.31	121.1
7	2.70	121.1
8	3.10	121.0
9	3.49	121.0
...	...	...
225	89.94	102.8
226	90.33	102.8


Finding the latent heat of vaporization of LN2
Background loss
Extra Mass change due to heating
Background loss
Mass (g)
Time (s)
122
120
118
116
114
112
110
108
106
104
102
0
10
20
30
40
50
60
70
80
90
100

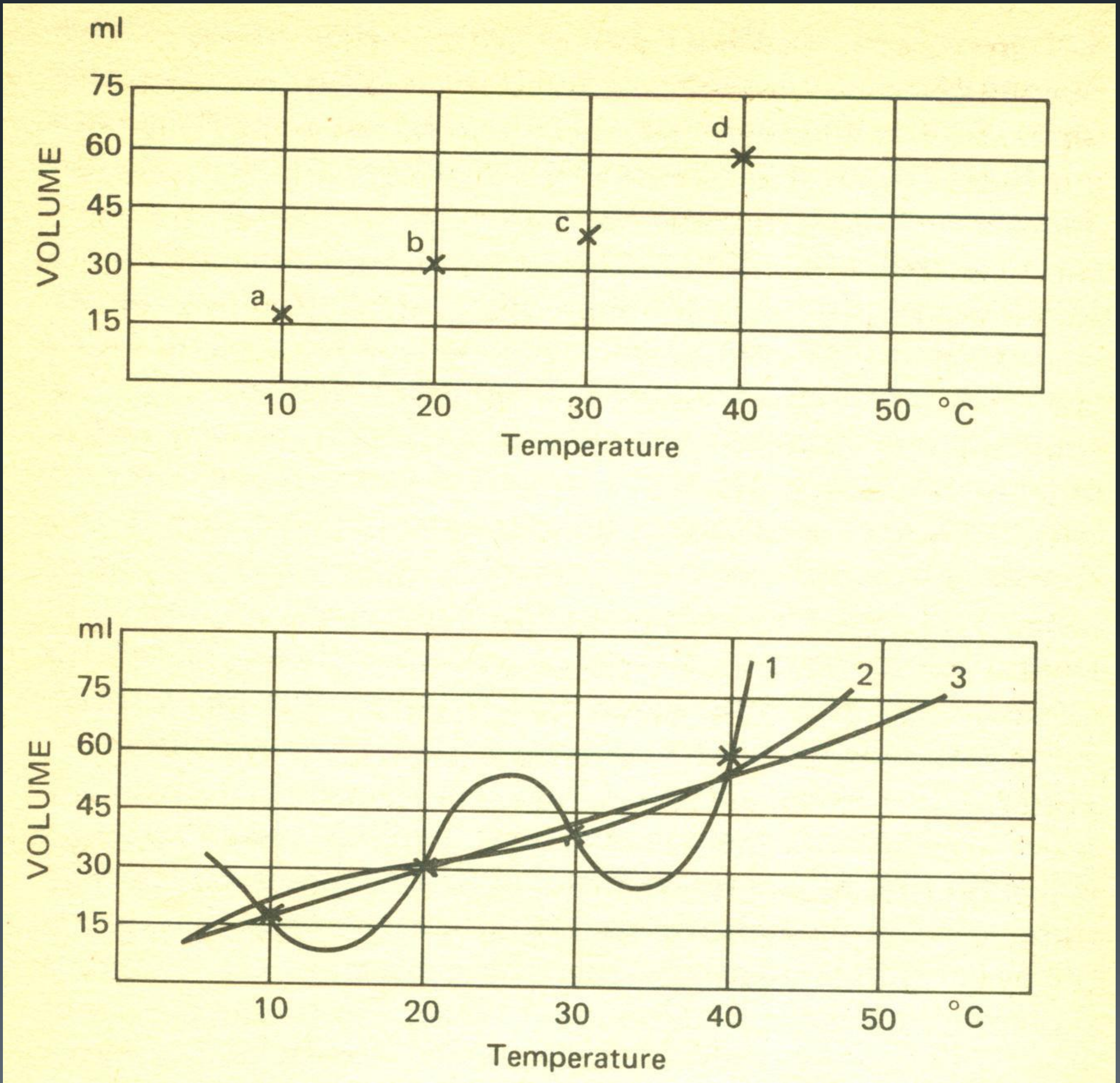
Introduction to plotting

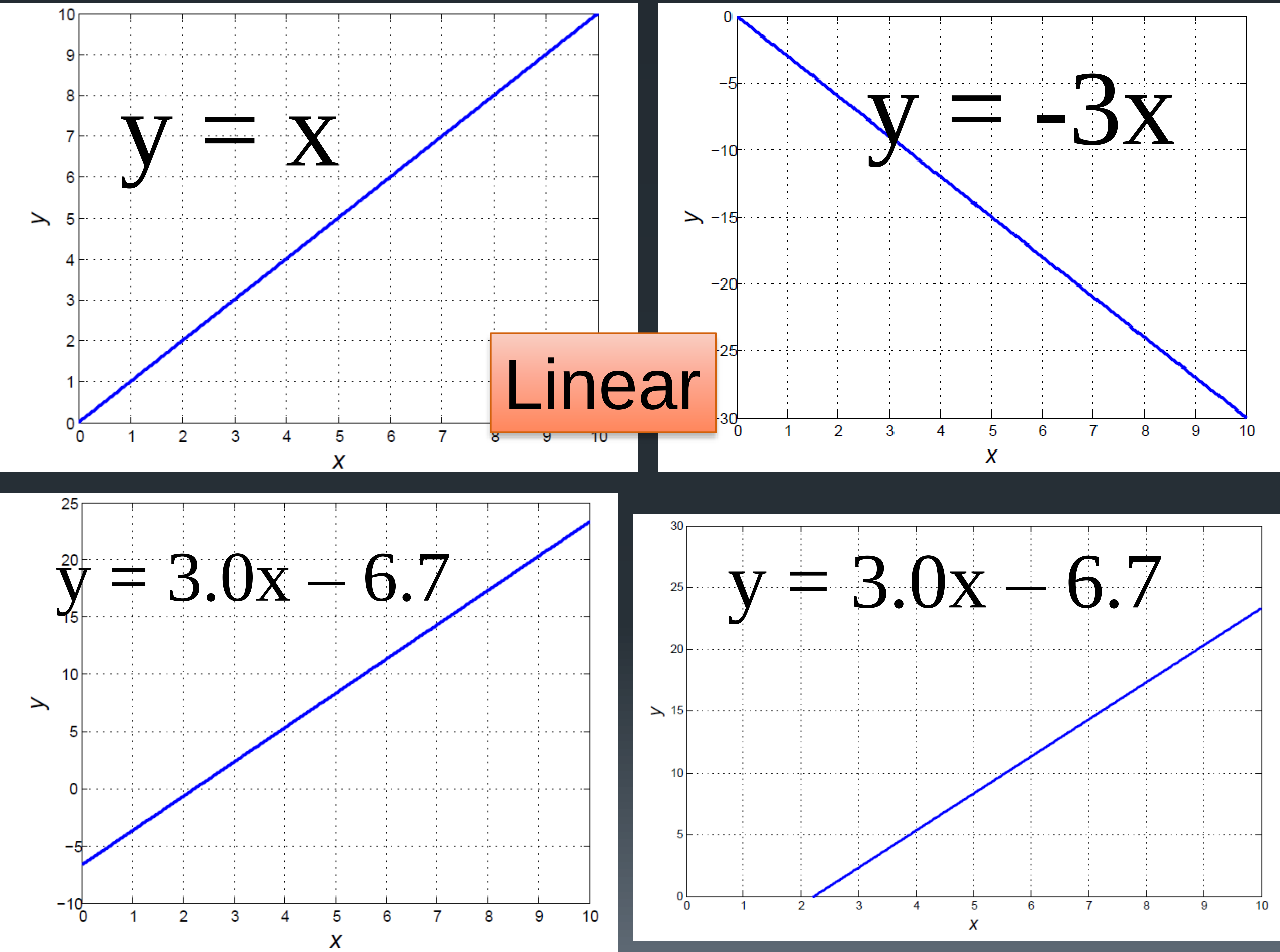
- Matlab can generate plots of a number of types
  - e.g.
    - Linear plots
    - Line plots
    - Logarithmic plots
    - Bar graphs
    - Three-dimensional plots

- In lab we will primarily work with two-dimensional plots by creating two “vectors” or an independent and dependent quantity.

- It is customary to plot independent variable (the “cause”) on horizontal axis and dependent variable (the “effect”) on the vertical axis.

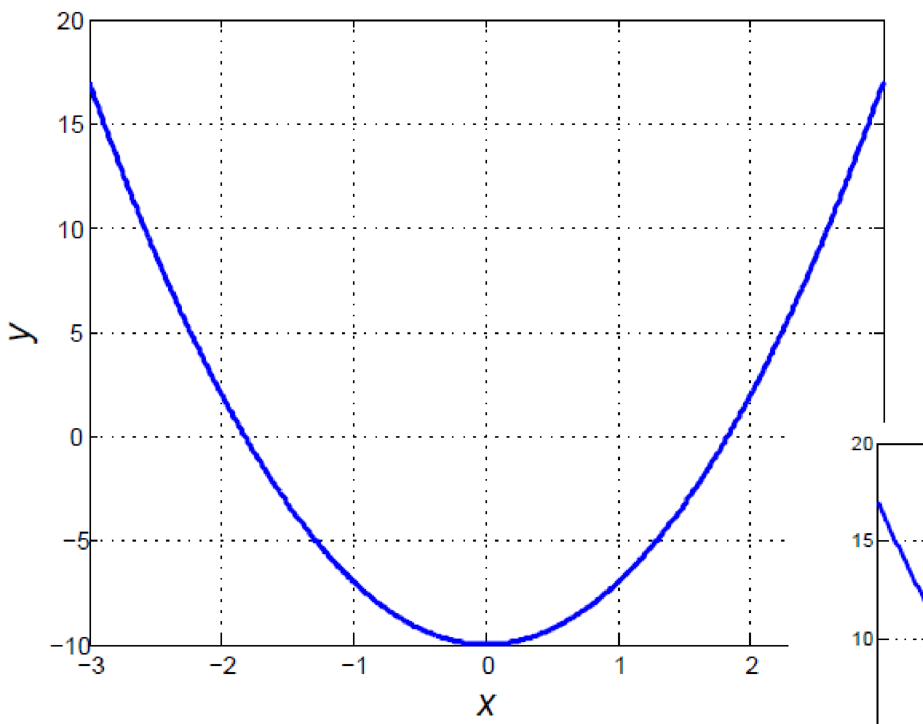
Typical models that fit typical experimental data



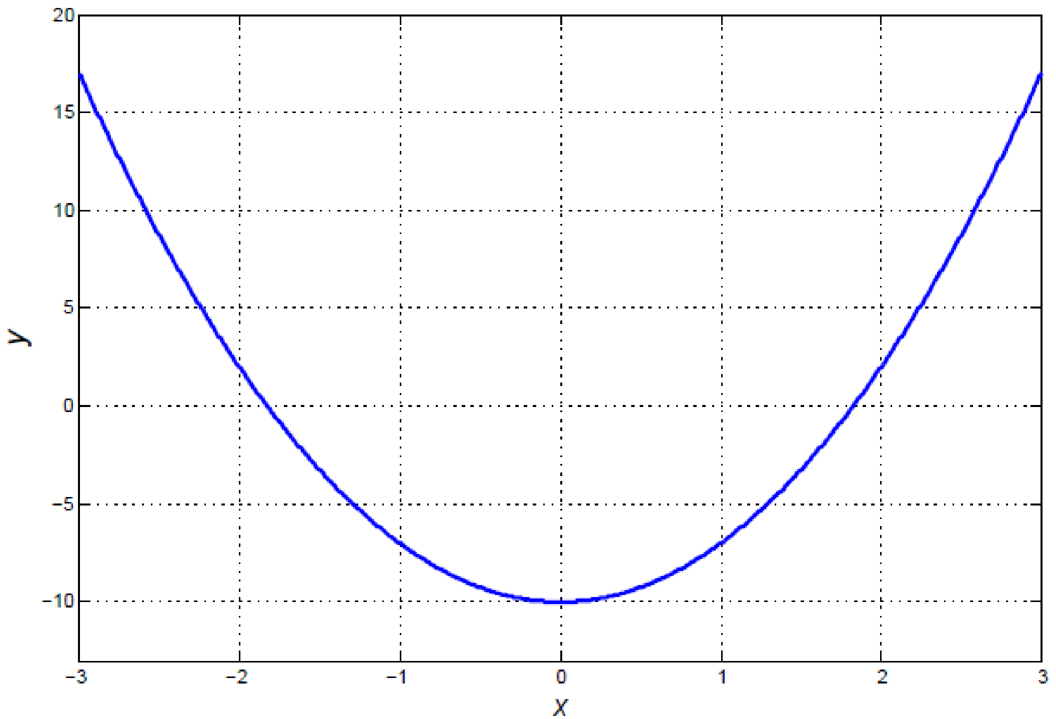


y = x
y = -3x
Linear
y = 3.0x – 6.7
y = 3.0x – 6.7
x
y

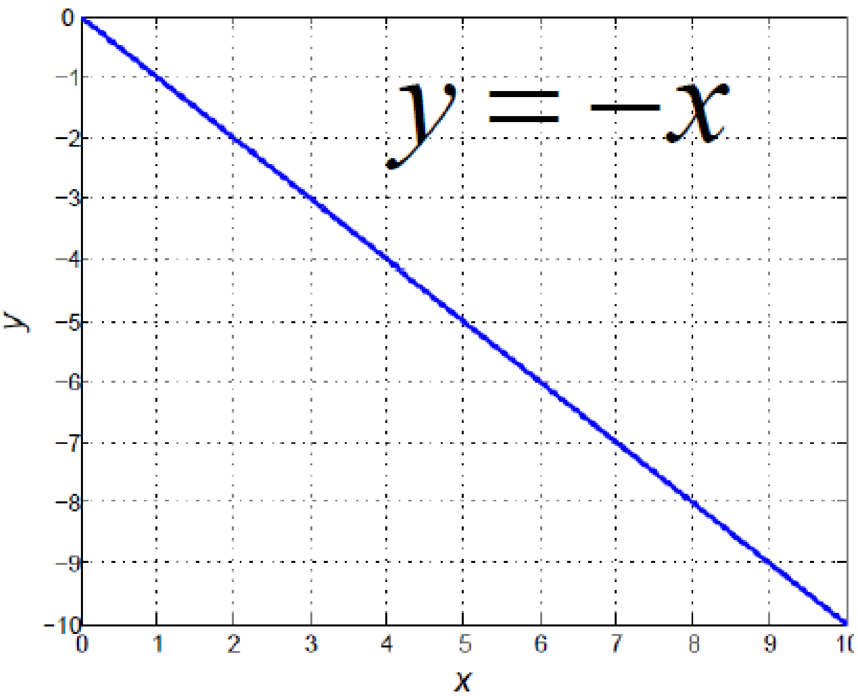
$Y = 3x^2 - 10$

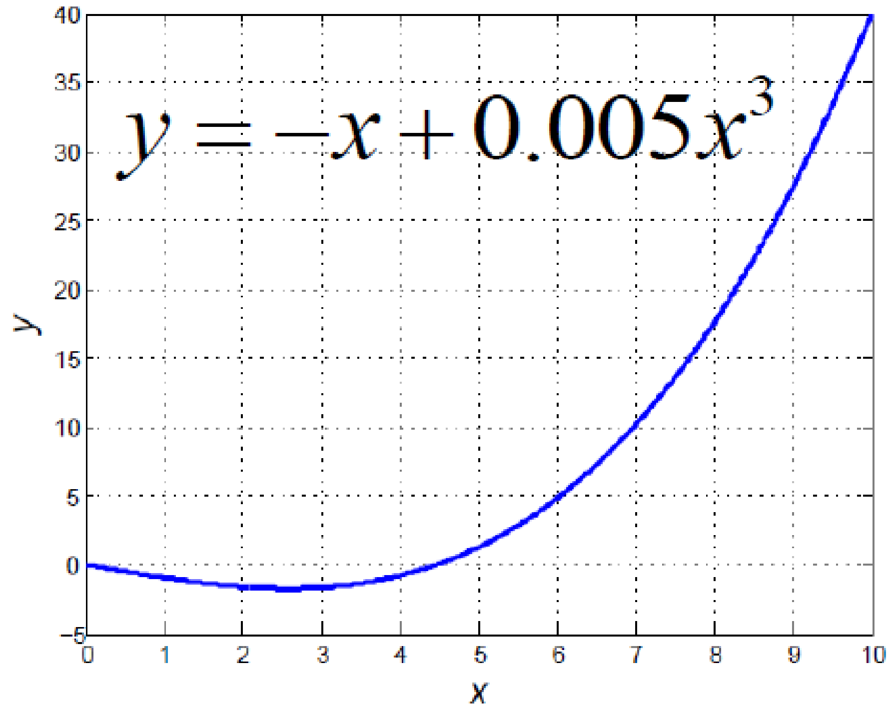


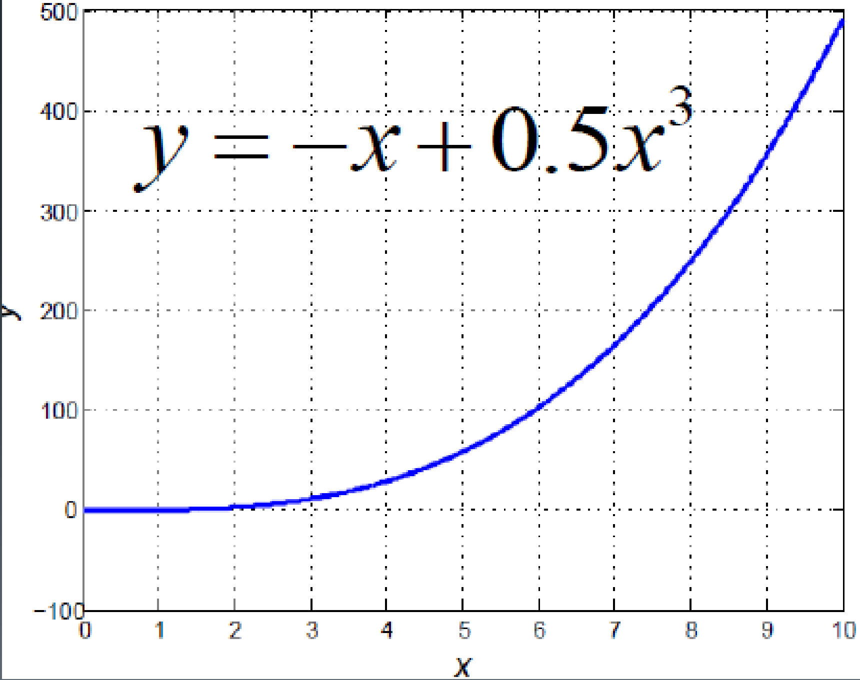
Quadratic

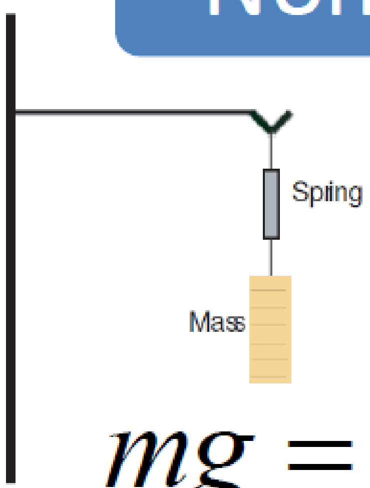


Nonlinearity



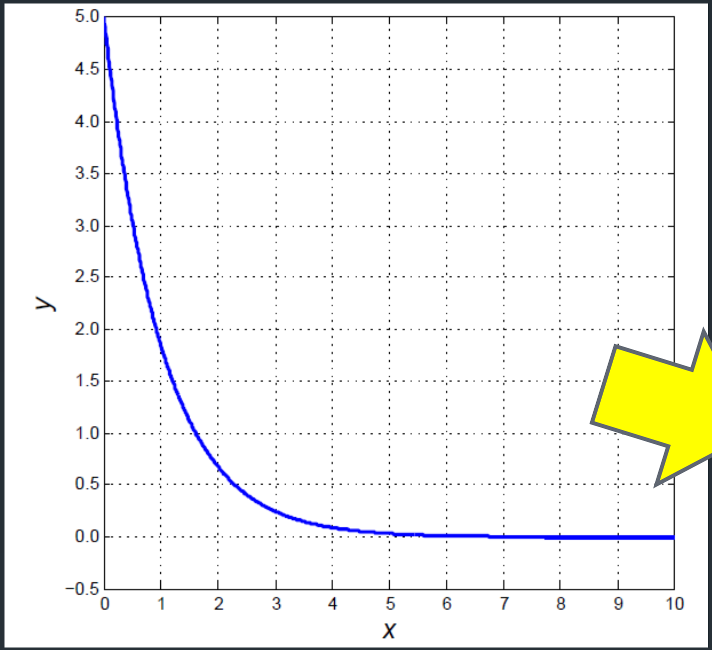






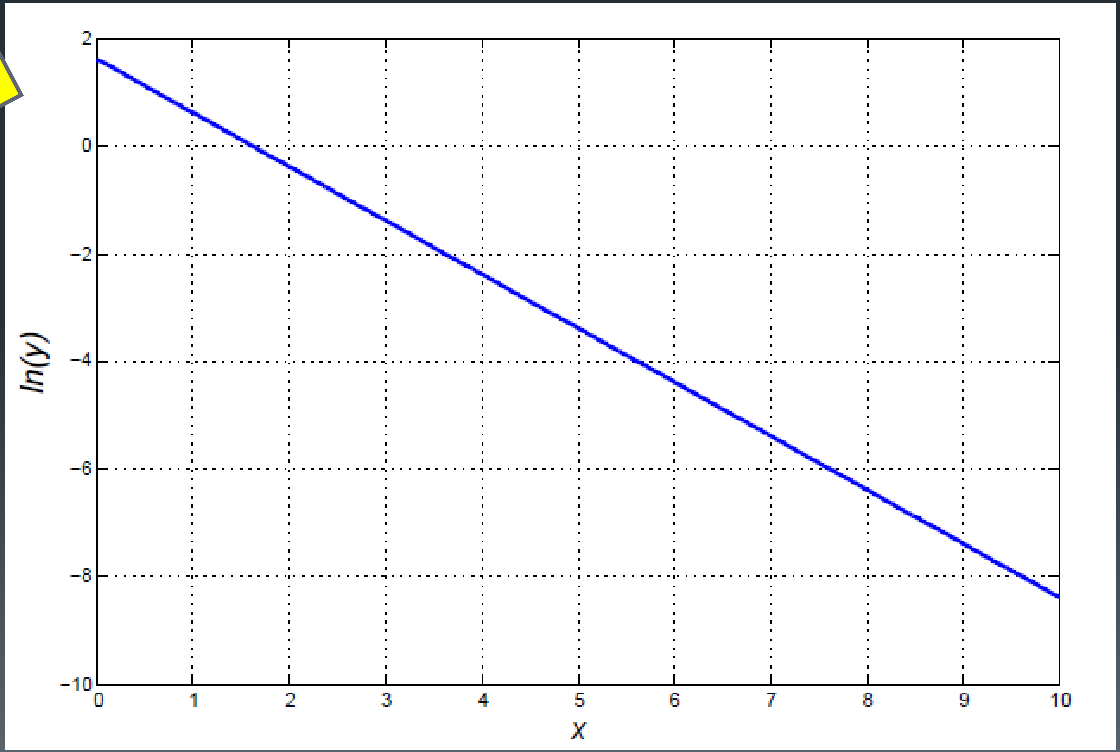
$$mg = -kx + \beta x^3$$

Linearization

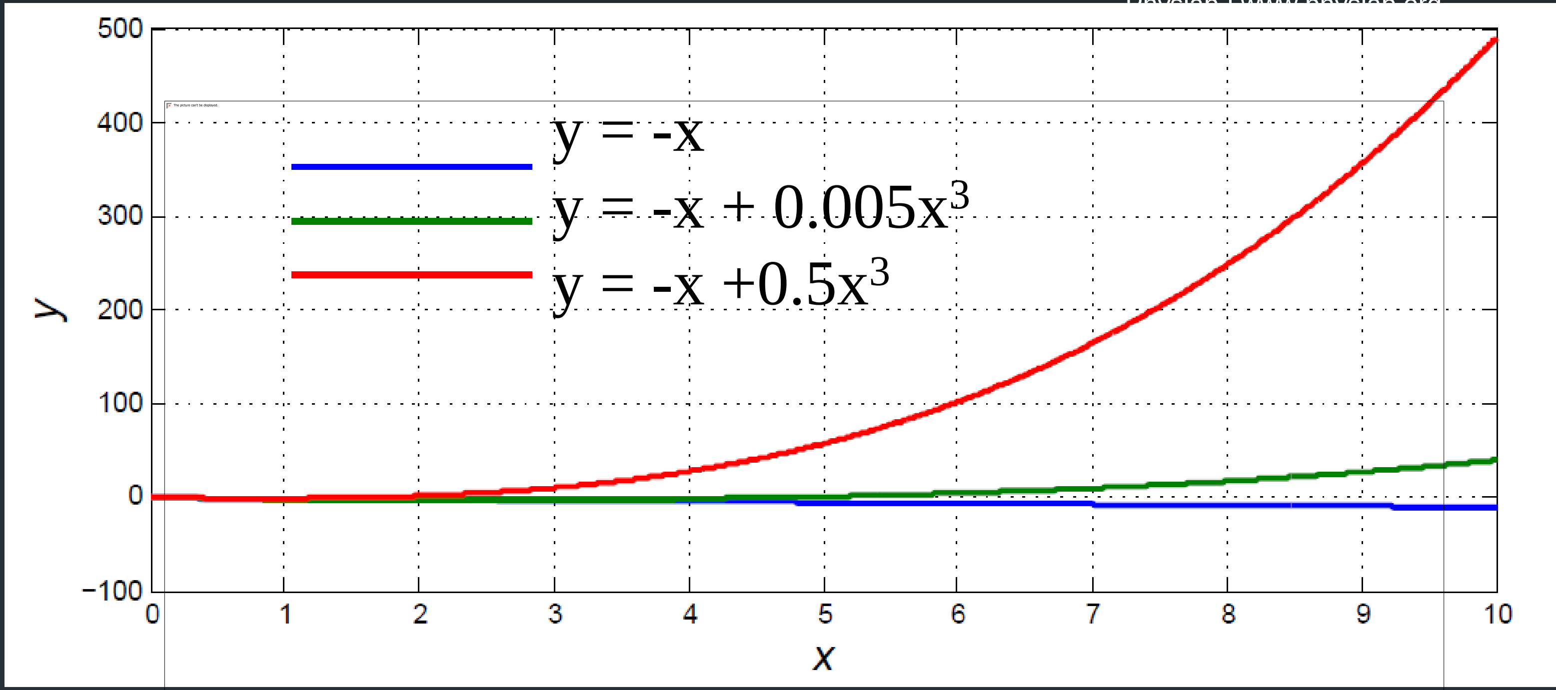


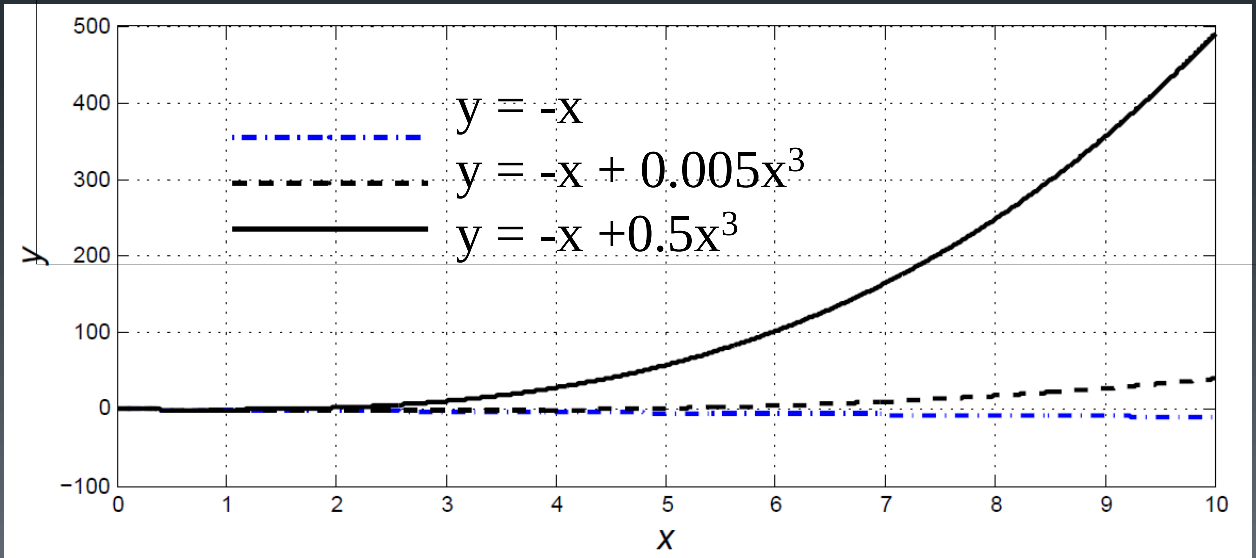
$$y = 5e^{-x}$$

$$\ln(y) = \ln(5) - x$$



Superposing Graphs





Example

- Let's consider an example of a stretched string fixed at one end to a rigid support, is strung over a pulley and a weight of 1.2 kg is attached at the other end. The string can be set under vibrations using a mechanical oscillator (woofer) connected to the signal generator.

- The relation of angular velocity (w) with the wave vector (k) is called the dispersion relation and given by,

$$w(k) = n\sqrt{\frac{T}{\mu}} \times \frac{\pi}{L}$$

$$w(k) = n\sqrt{\frac{T}{\mu}} \times \frac{\pi}{L}$$

$$2\pi f = n\sqrt{\frac{T}{\mu}} \times \frac{\pi}{L}$$

$$f = \frac{1}{2L}\sqrt{\frac{T}{\mu}} \times n$$

$$y = \underline{m}\,x + c$$

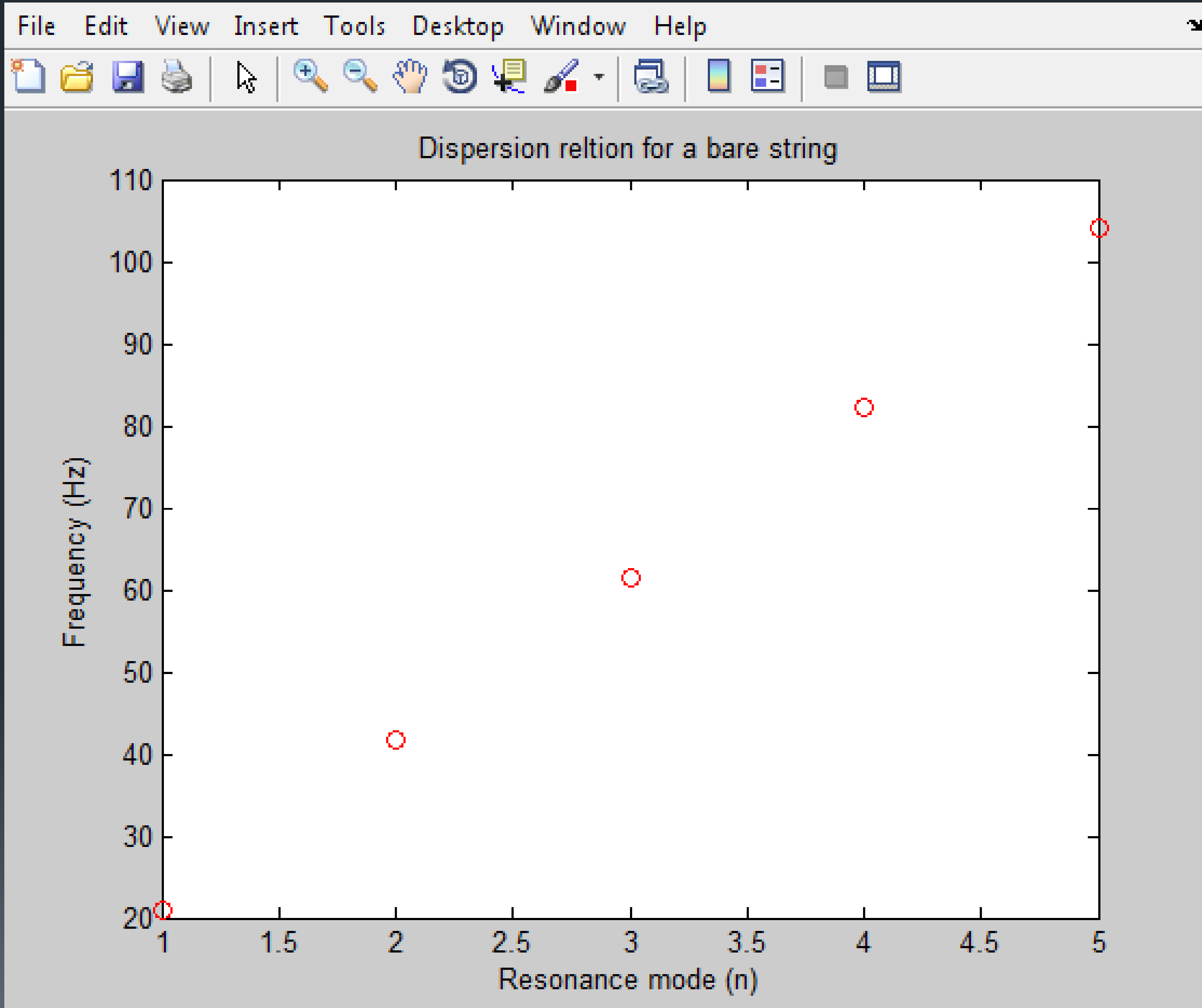
Example

Plot the following experimental data:

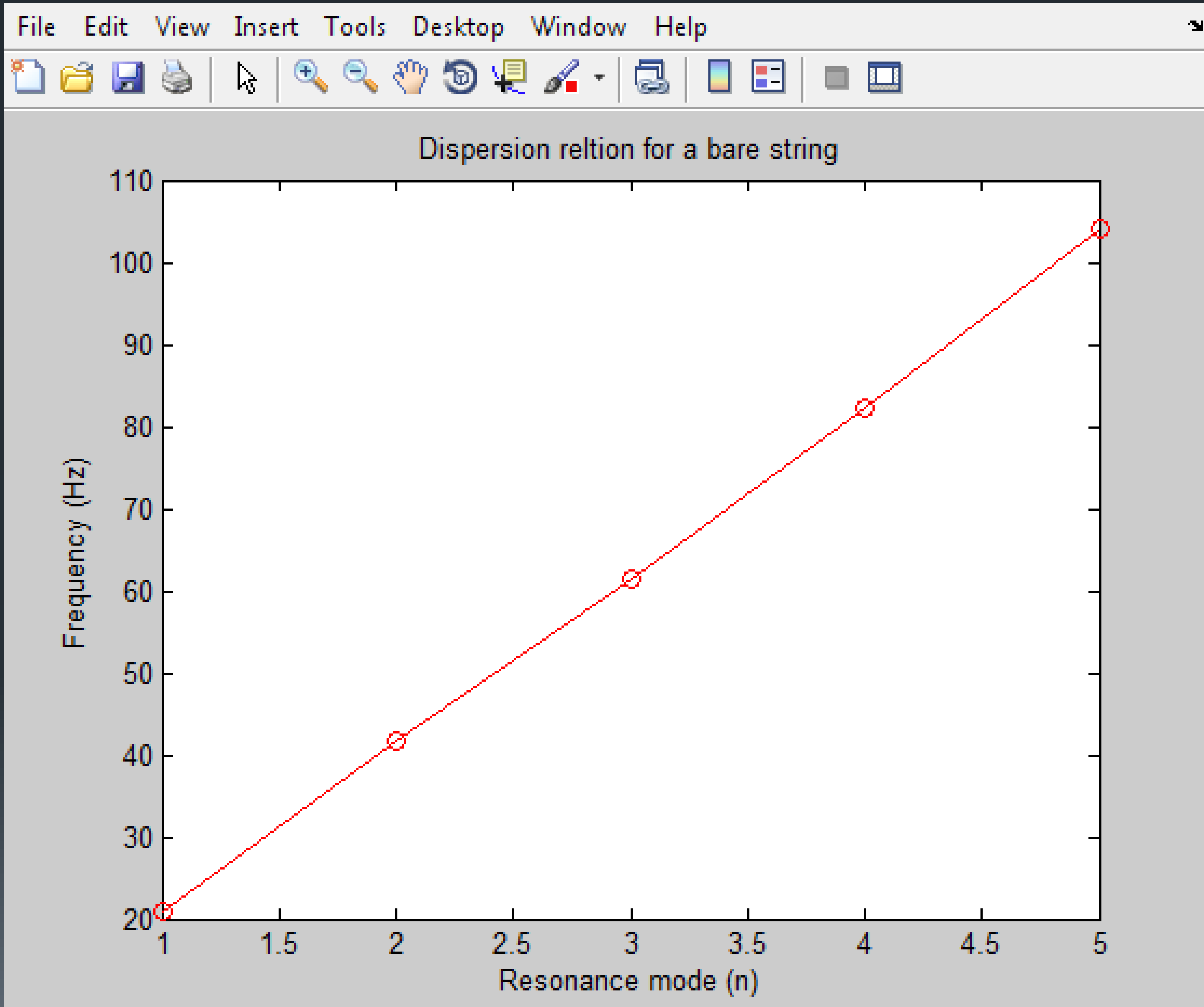
Resonance mode (n)	**1**	**2**	**3**	**4**	**5**
Frequency (Hz)	20.82	41.82	61.32	82.32	104.1

Commands:

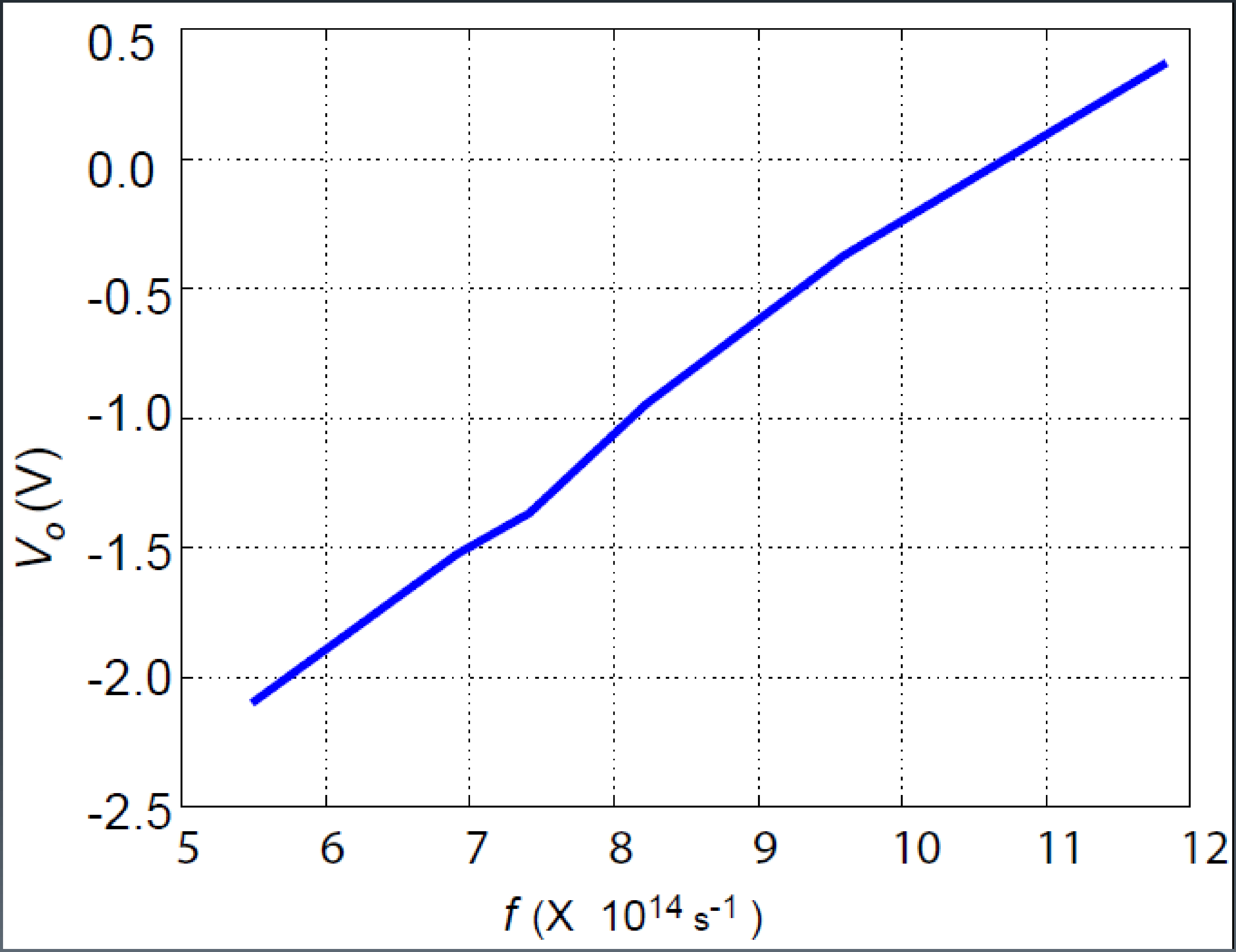
```
n=[1 2 3 4 5];
f=[20.82 41.82 61.32 82.32 104.1];
plot(n,f)
xlabel(`Resonance mode (n)')
ylabel(`Frequency (Hz))')
title('Dispersion relation for a bare string')
```



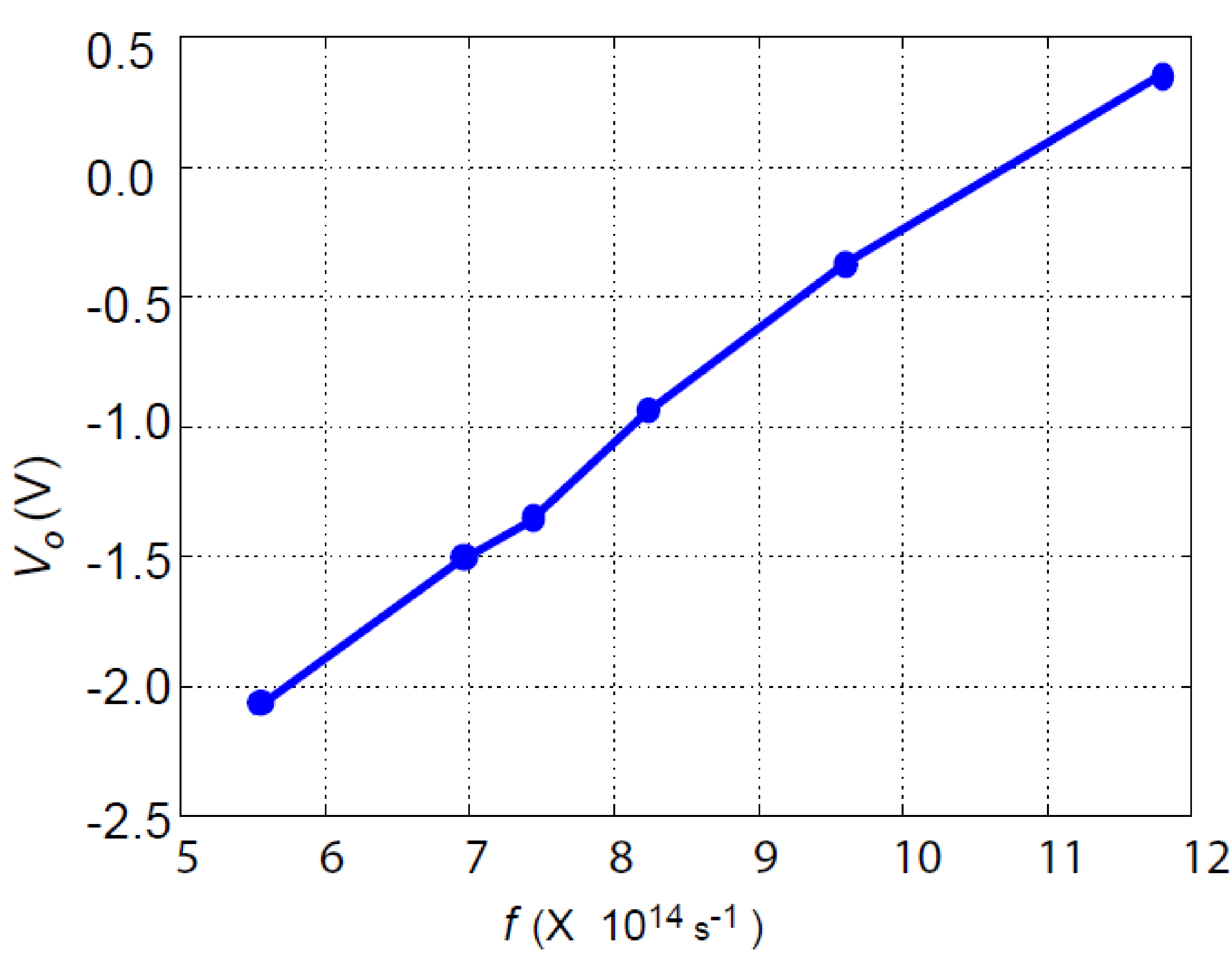
File Edit View Insert Tools Desktop Window Help
Dispersion reltion for a bare string
Frequency (Hz)
110
100
90
80
70
60
50
40
30
20
1
1.5
2
2.5
3
3.5
4
4.5
5
Resonance mode (n)


File Edit View Insert Tools Desktop Window Help
Dispersion reltion for a bare string
Frequency (Hz)
110
100
90
80
70
60
50
40
30
20
1
1.5
2
2.5
3
3.5
4
4.5
5
Resonance mode (n)

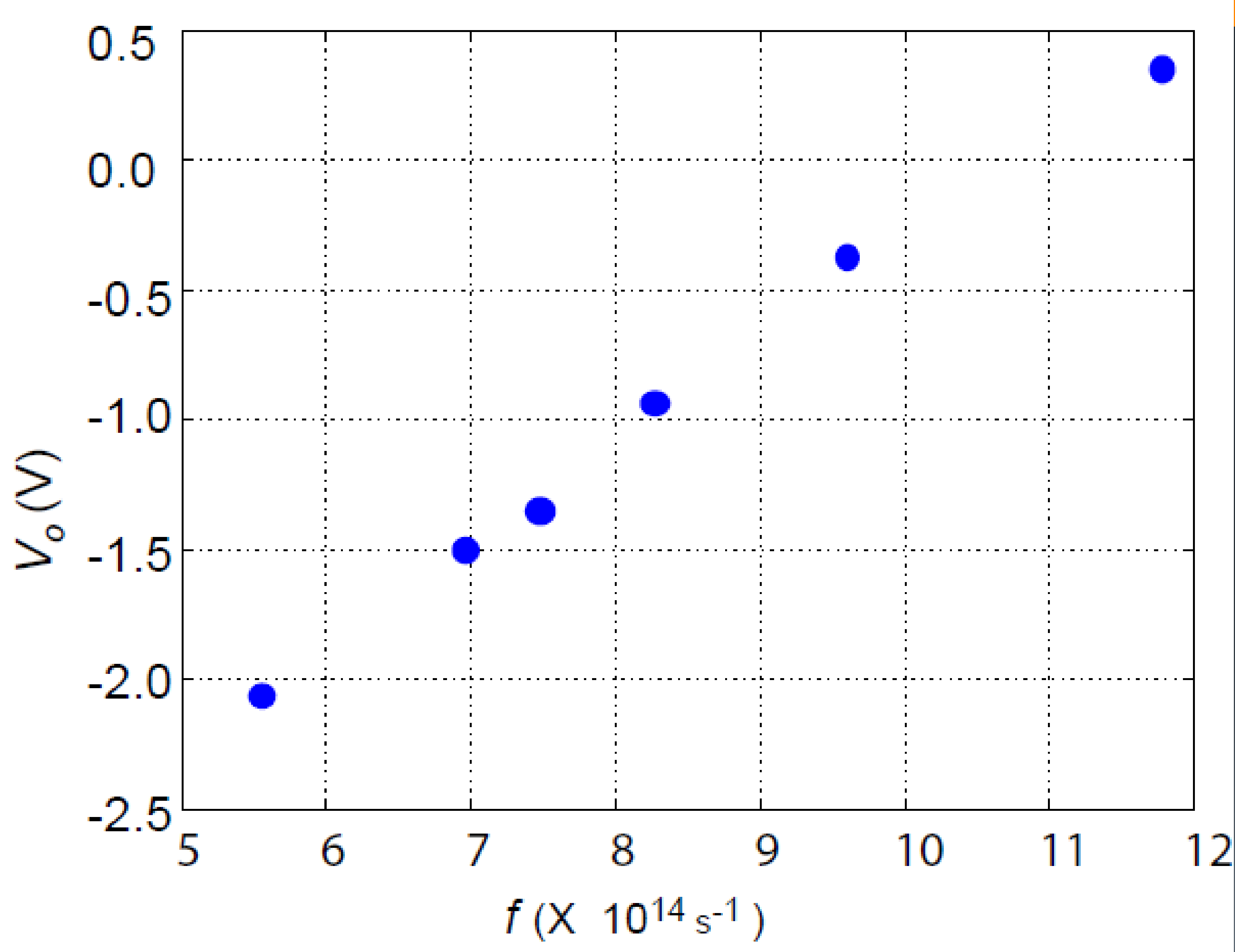
Possible Ways of Plotting
(a) Not acceptable



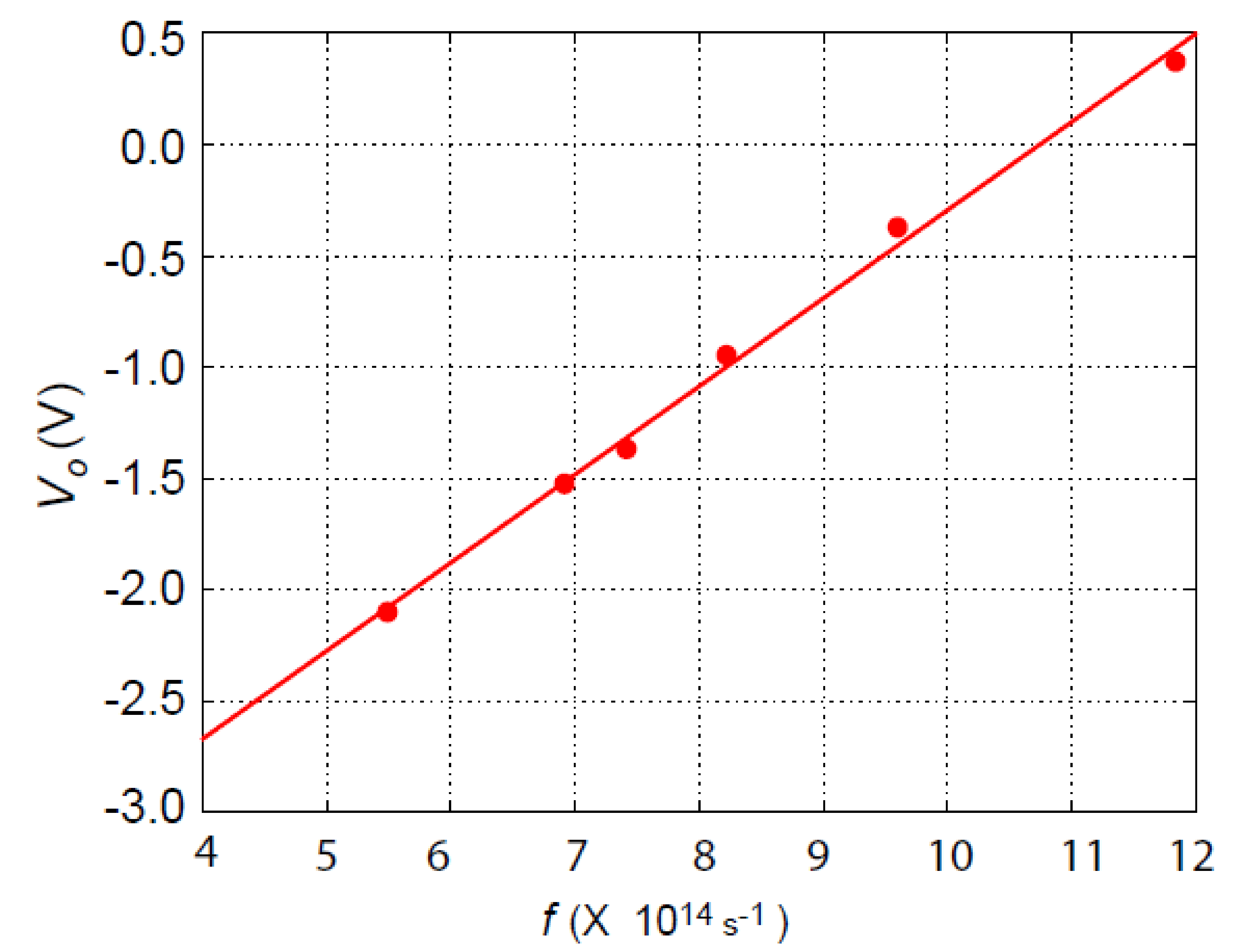
(b) Barely acceptable



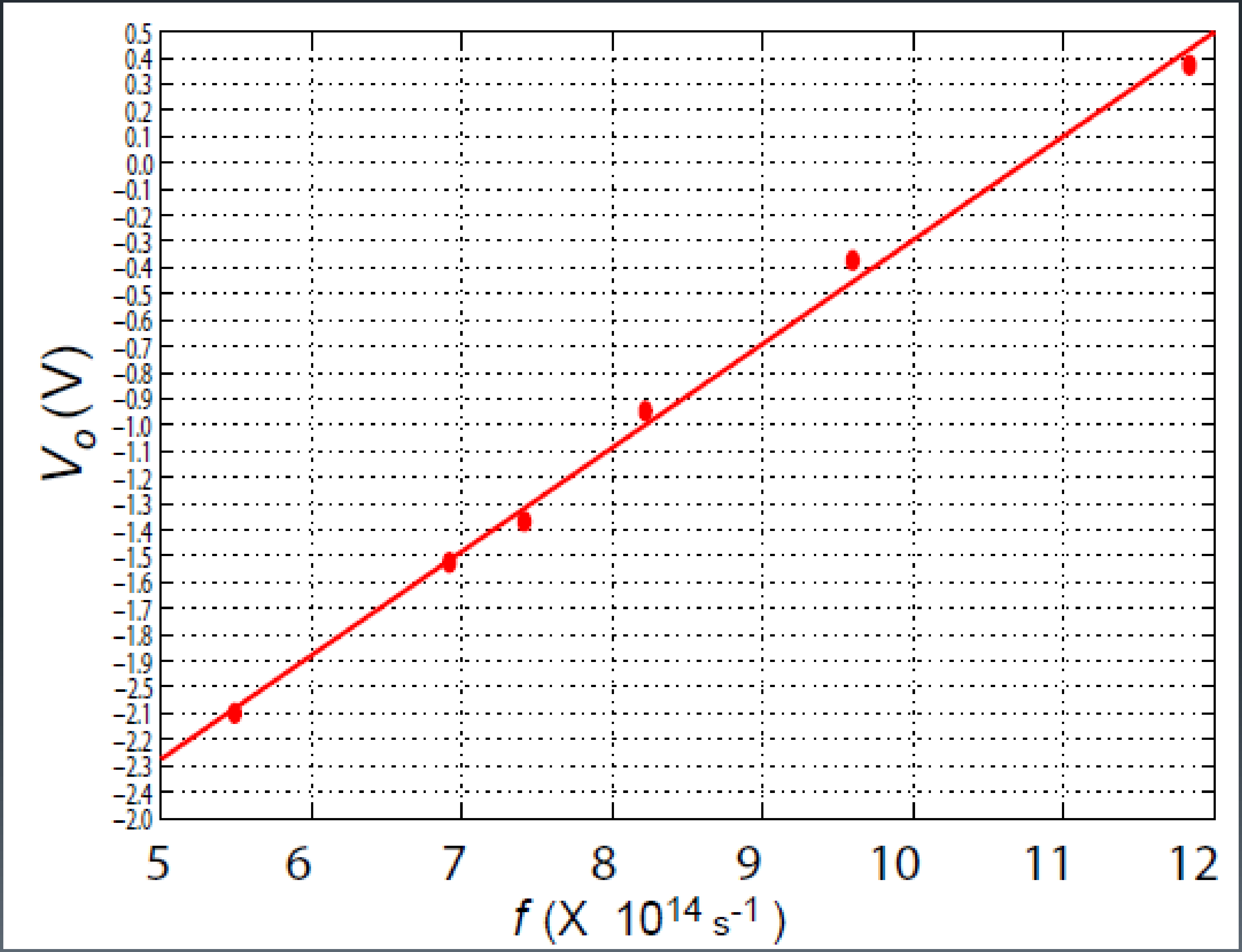
(c) Better



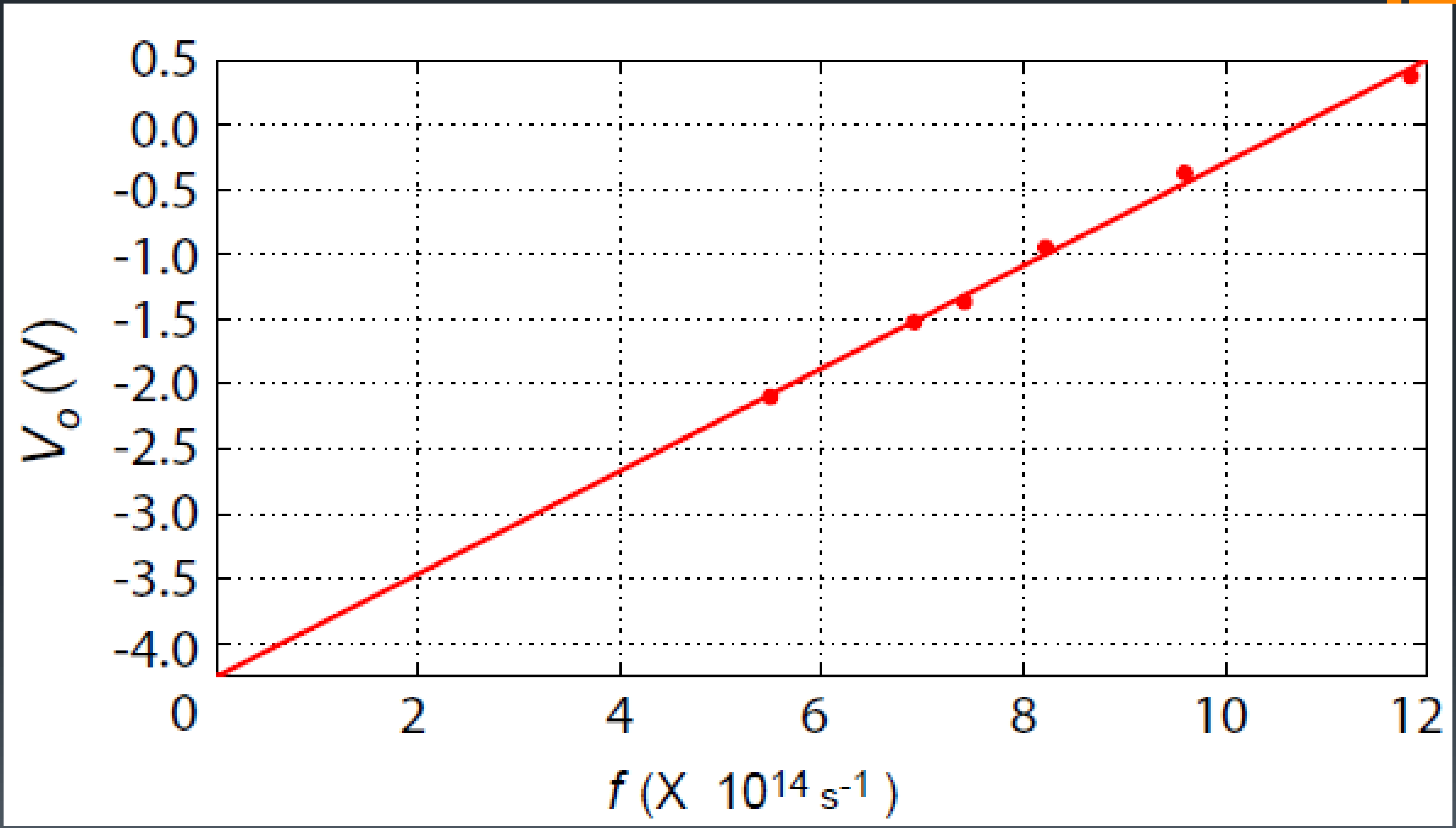
(d) Acceptable, good in all respects



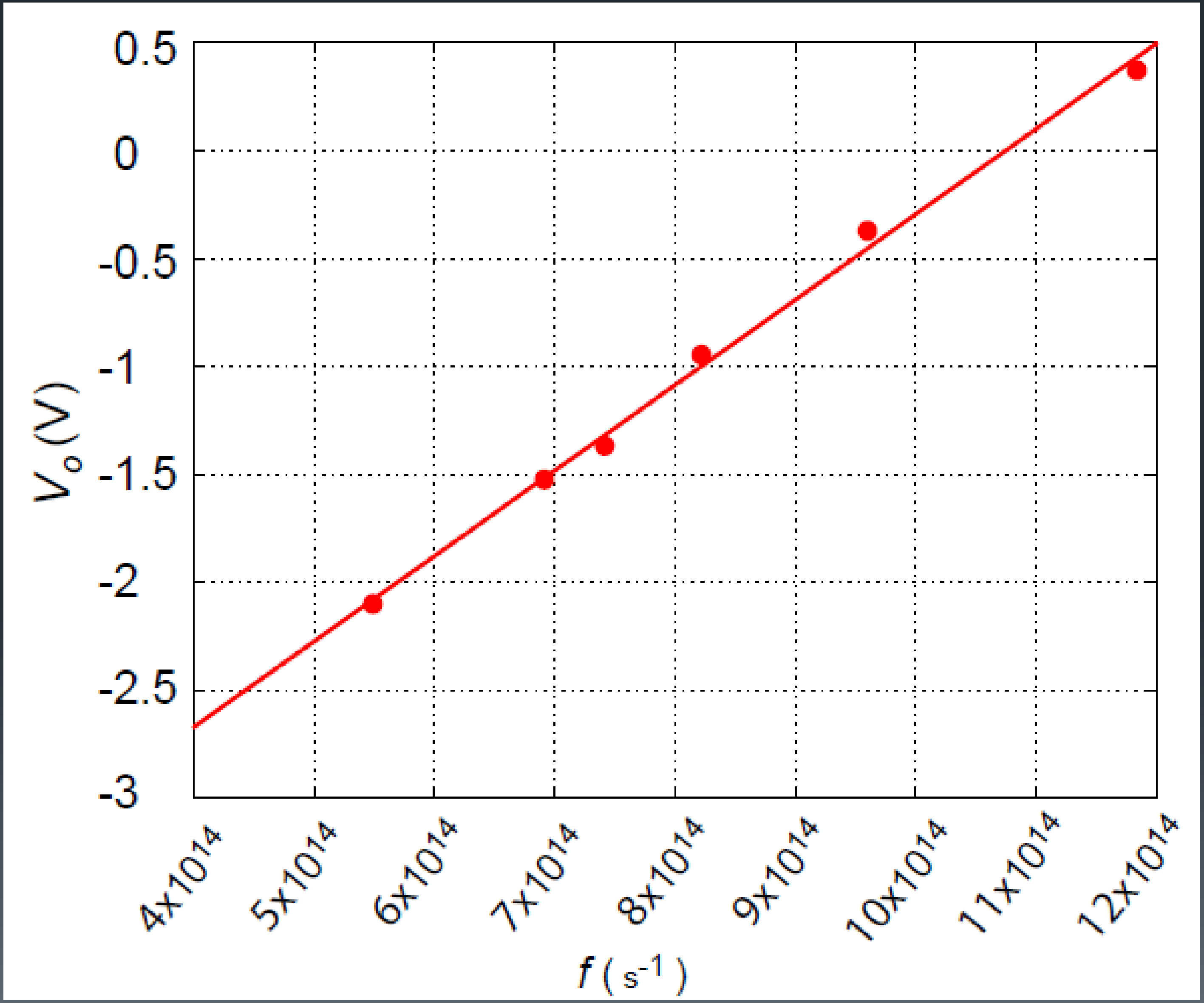
(e) Unnecessary detail or embellishment



(f) Axes too long

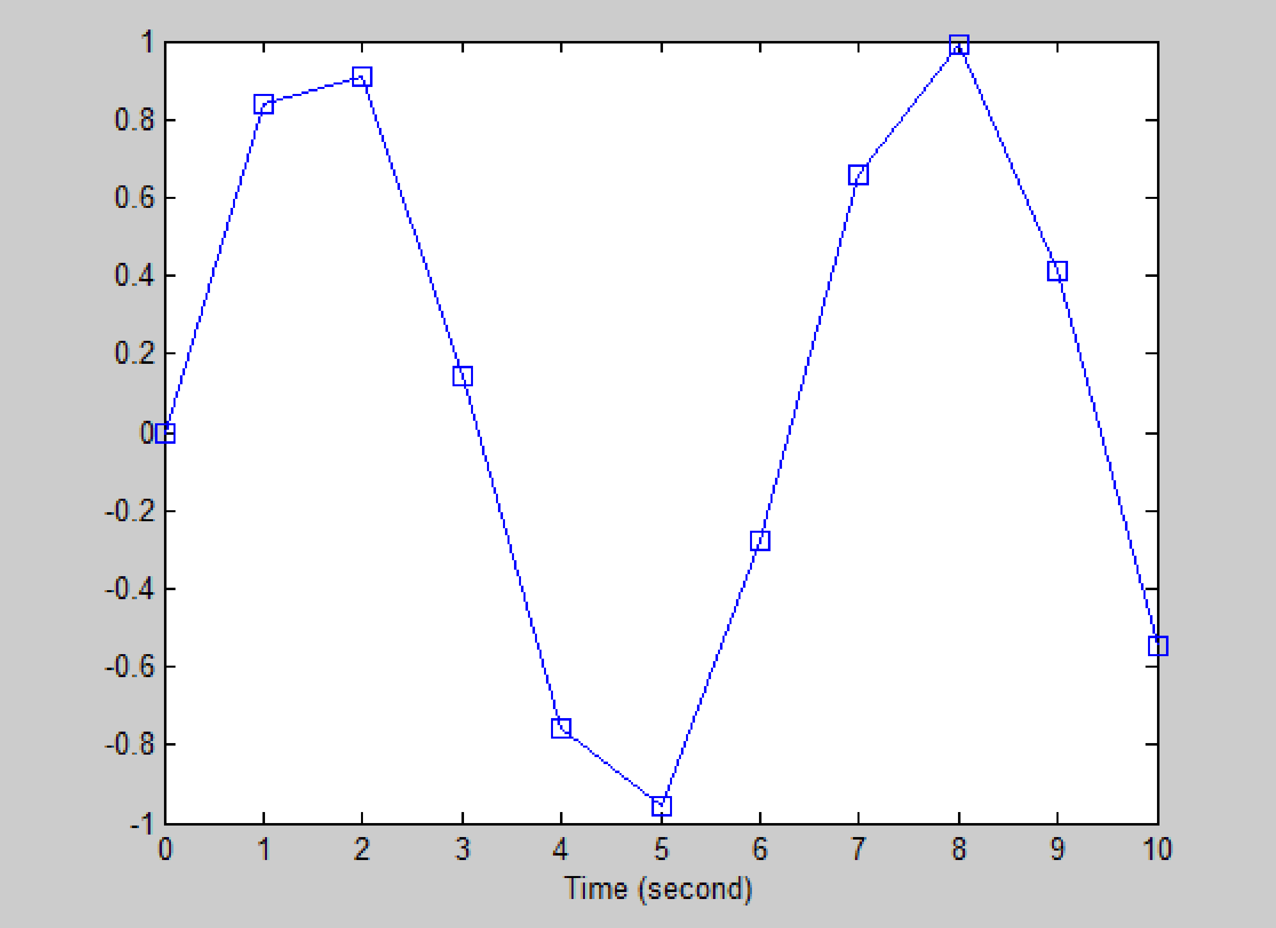


(g) Axes tick marks are inconsistent and clumsy



Resolution of Graph

Suppose we have a sine curve, sampled at interval of 1s for a duration of 10s, it means there are eleven data points contained within the sampled duration.



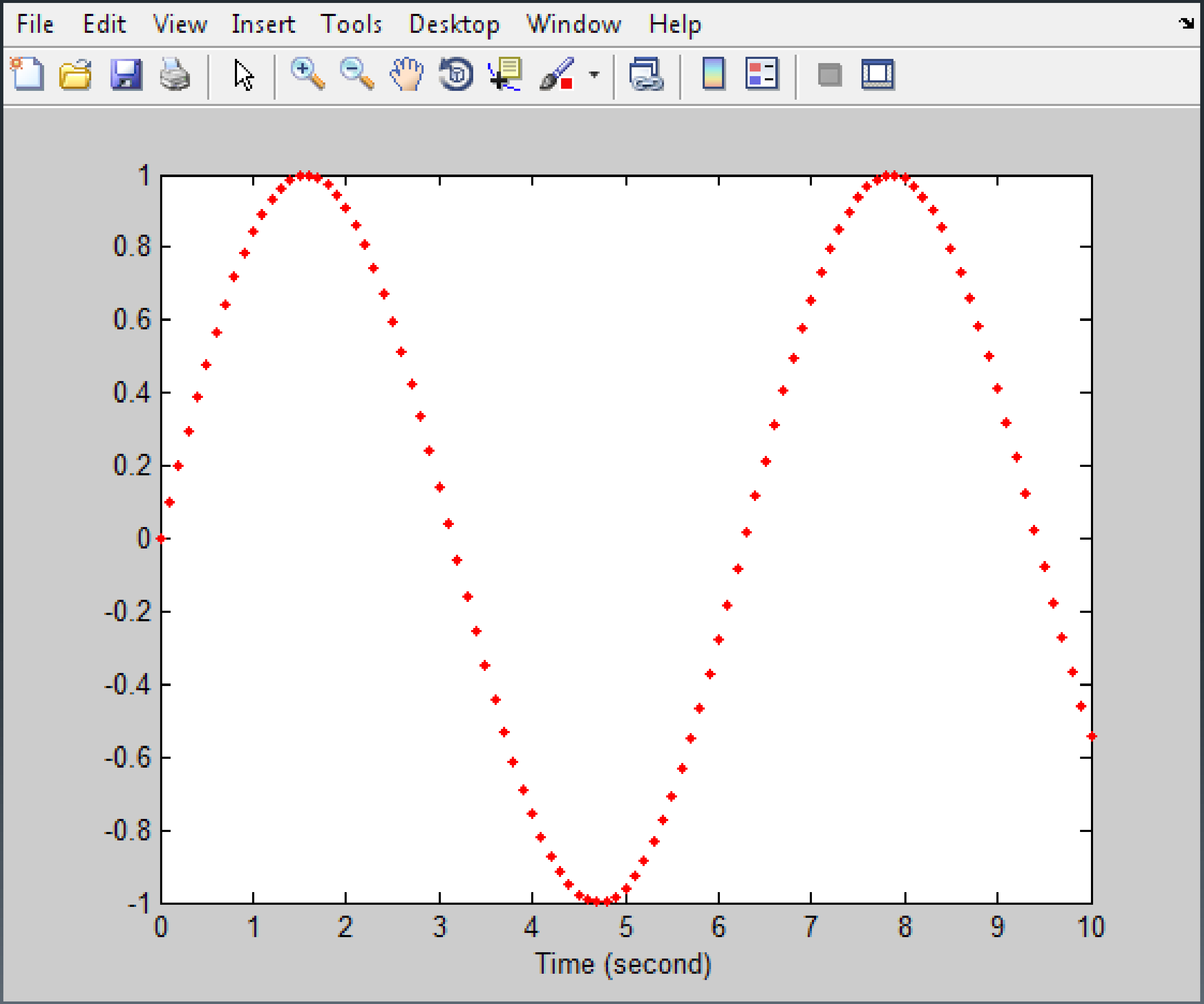
Resolution of Graph

- We know from experience that a plot of the sine function should be smooth.

- **Why is this discrepancy?**

The reason is that we have not sampled enough points. So, we decrease the sampling interval to 0.1 s and hence, increasing the number of samples to 101, we recover a smooth sine curve.

Resolution of Graph



Multiple plotting

Let's define input vectors:

x=0:0.1:4*pi;

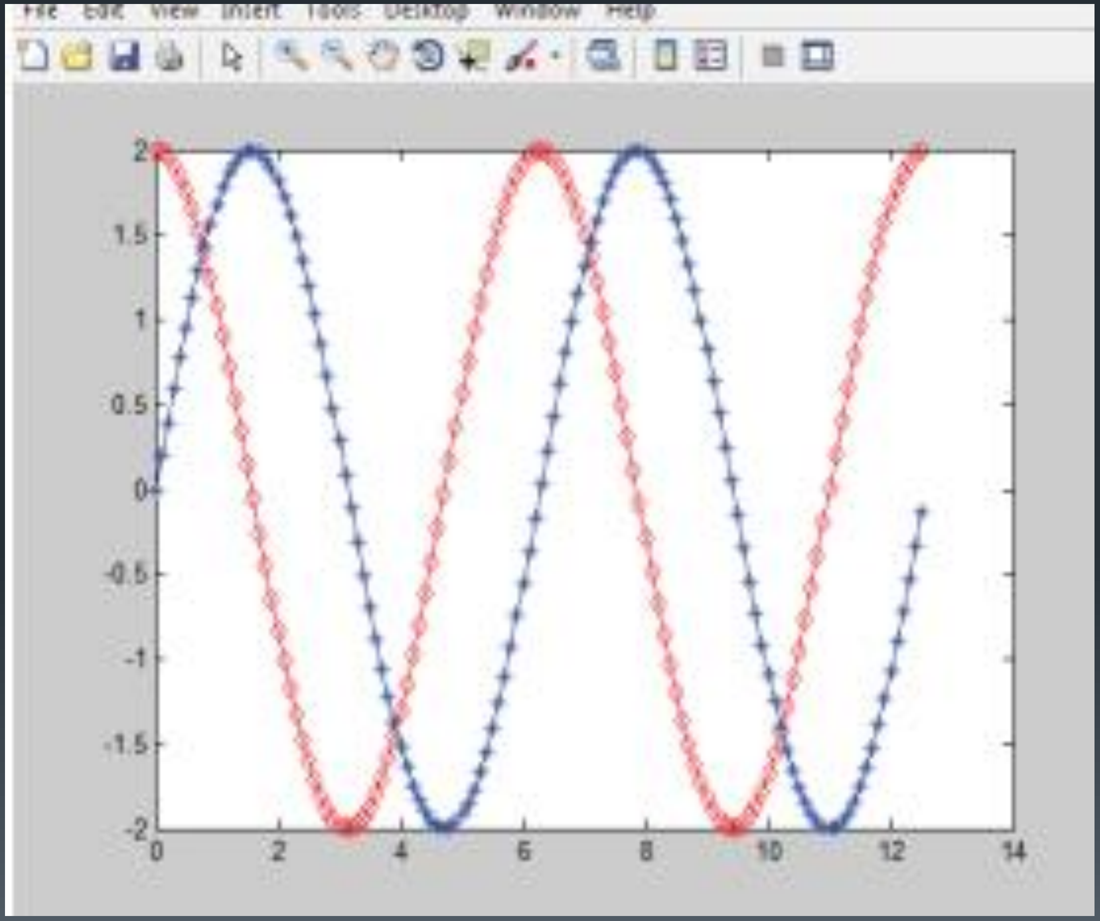
y=2*cos(x);

y1=2*sin(x);

figure; plot(x,y,`r-d')

hold on

plot(x,y1,`b-*')



Let’s practice!

Solve the second exercise …

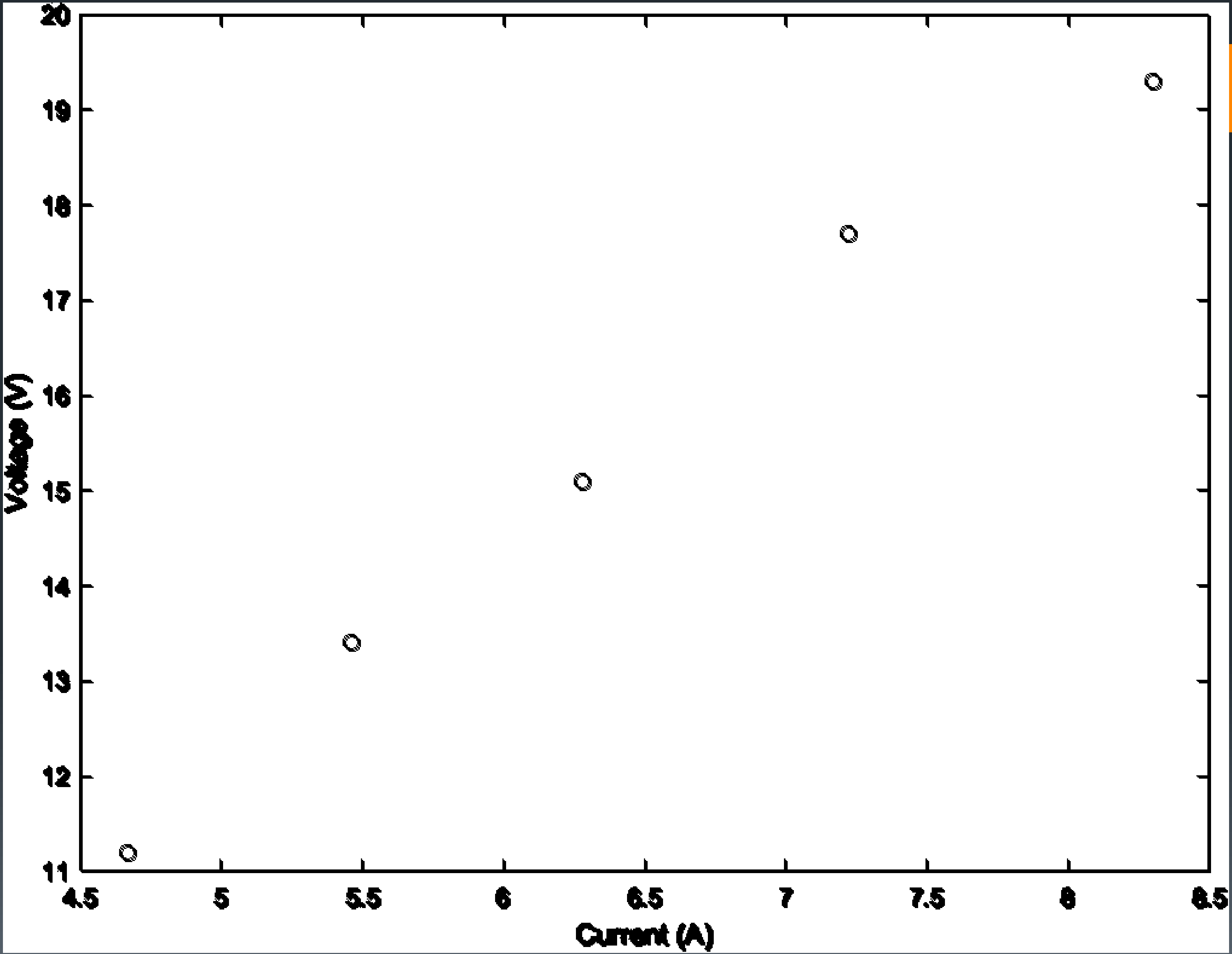
Introduction to MATLAB

Curve Fitting

Contents

Use case

Students performed an experiment in the lab and collected data …



20
19
18
17
16
15
14
13
12
11
Voltage (V)
4.5
5
5.5
6
6.5
7
7.5
8
8.5
Current (A)

Fitting a line to Data

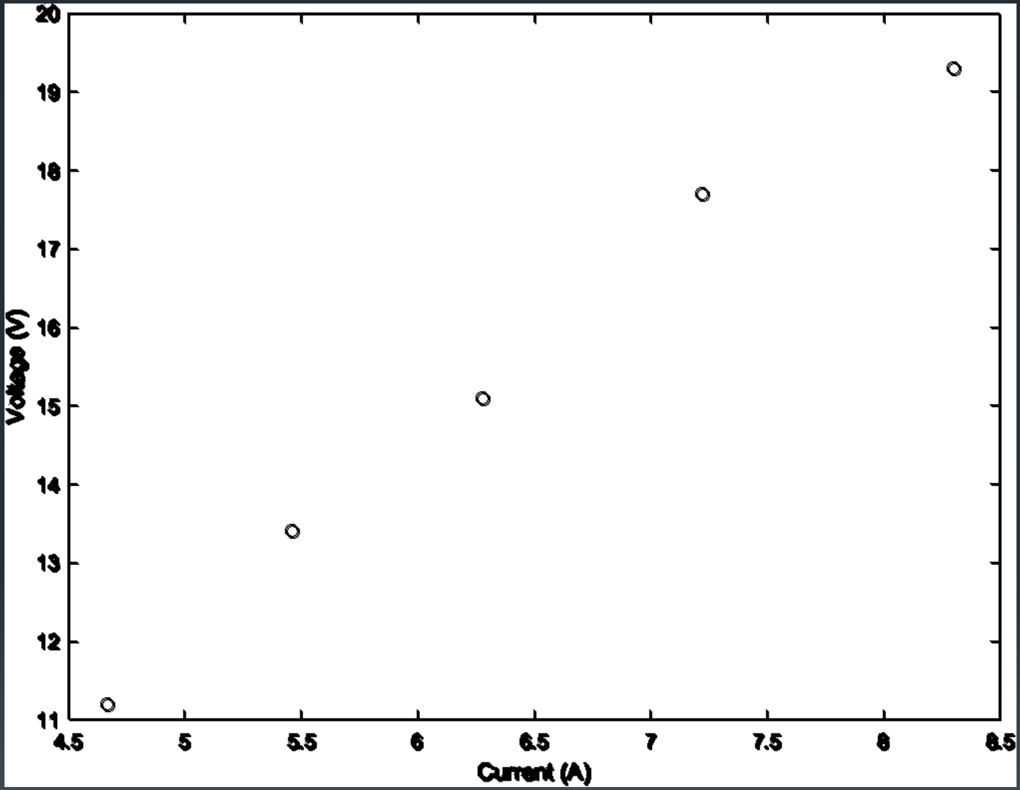
- Given n pairs of data points

$$(x_i, y_i), \quad i = 1, \dots, n$$

Find the coefficients m and c such that

$$F(x) = mx + c$$

is a good fit to the data

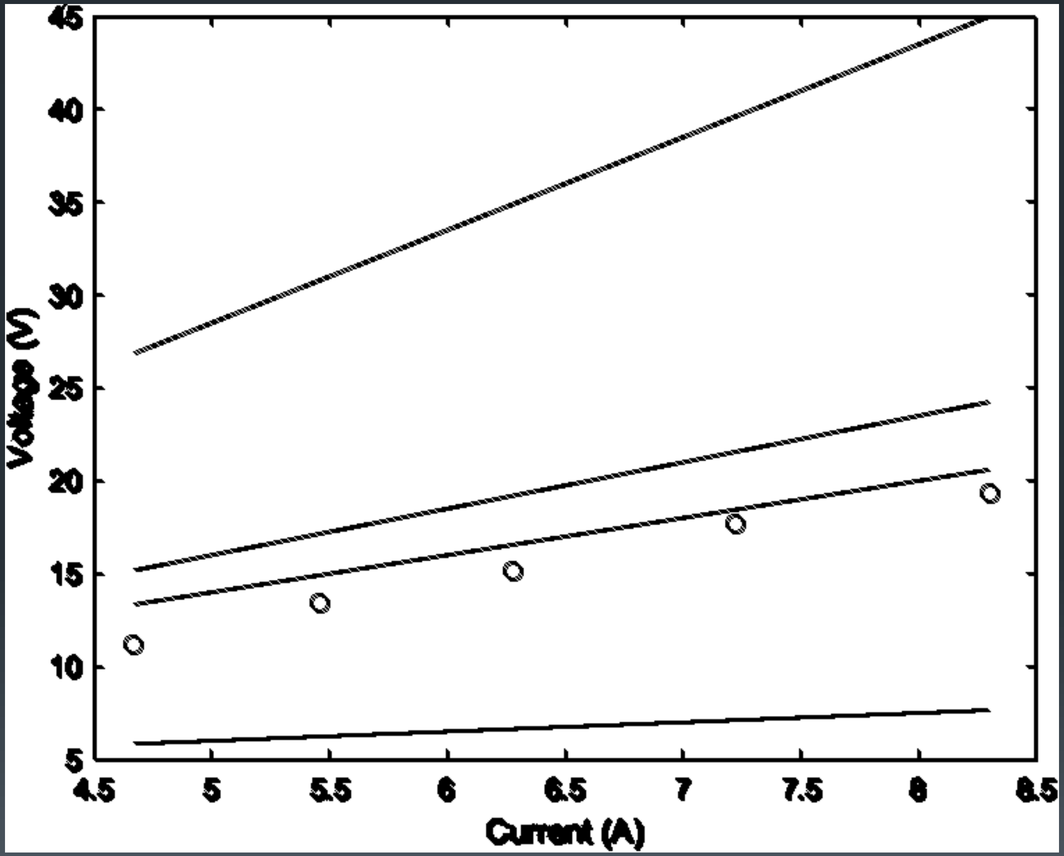


Questions:

1. How do we define a good fit?
2. How do we compute m and c after a definition of "good fit" is obtained?

Plausible Fits

- Plausible fits are obtained by adjusting the slope (m) and intercept (c).
- Here is a graphical representation of potential fits to a particular set of data.
- Which of the lines provides the best fit?



Best Fit & Residual

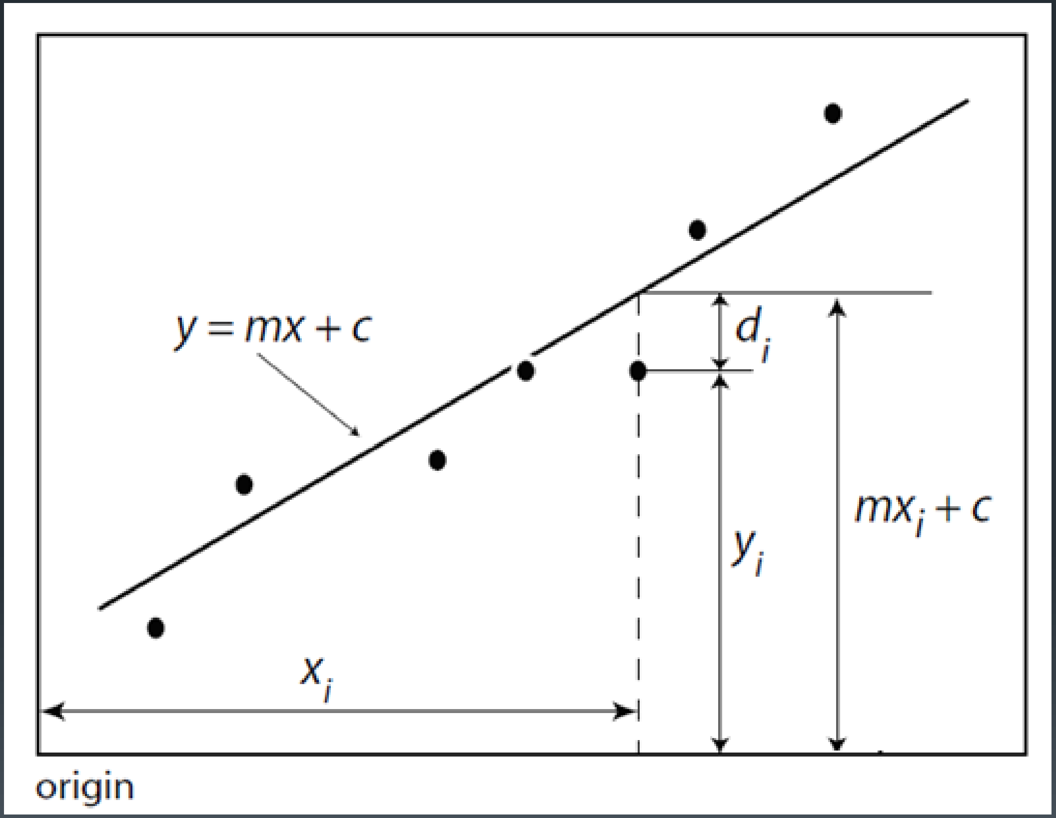
- If a straight line is drawn through these experimental points, represented by the equation,

$$y = mx + c$$

where m is the slope c is intercept.

- The deviation or the (residual) between the point y on the line and the measurement point y_i will be,

$$d_i = y_i - mx_i - c$$

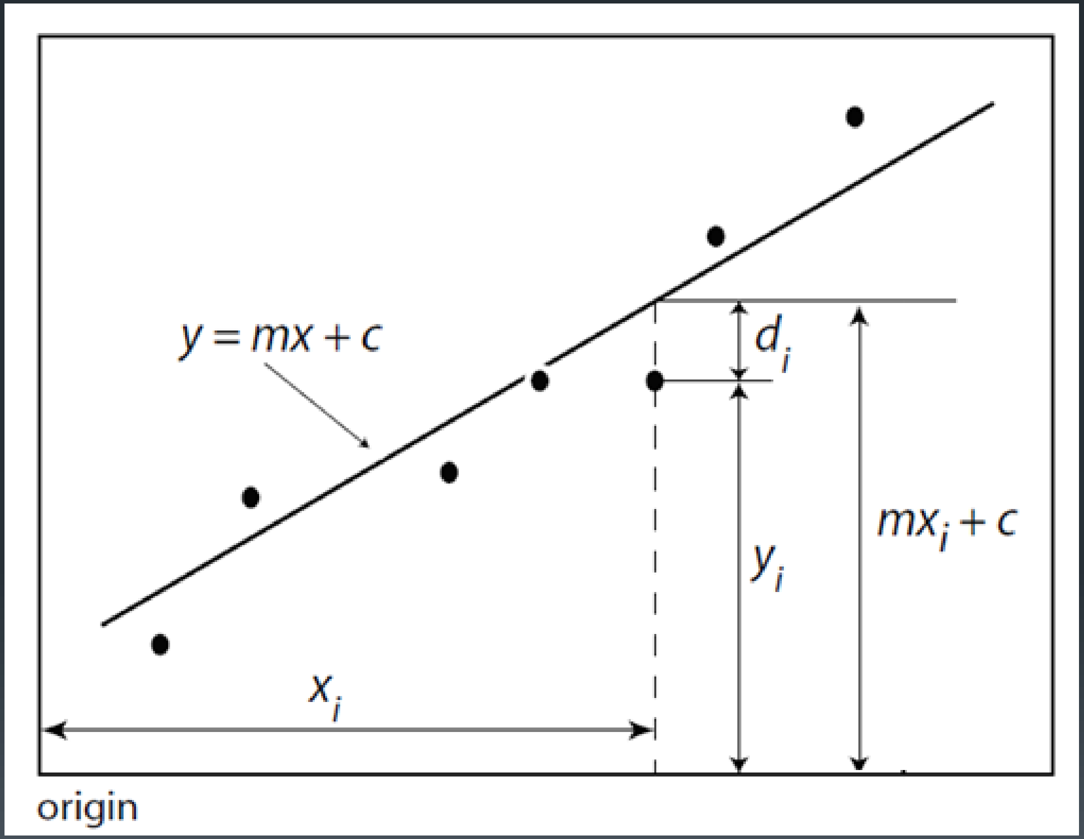


Best Fit & Residual

- This special line we have drawn has the property that it minimizes the sum of the squares of the deviations,

$$S = \sum_{i=1}^{N} d_i^2$$

- If the d_i's are the errors, the least squares curve fit is the best fit in the sense that it minimizes the squares of the errors.



Least squares curve fitting is minimization of S

$$\frac{\partial S}{\partial m} = -2\sum[x_i(y_i - mx_i - c)] = 0$$

$$\frac{\partial S}{\partial c} = -2\sum(y_i - mx_i - c) = 0$$

$$m = \frac{\sum y_i(x_i - \bar{x})}{\sum(x_i - \bar{x})^2}$$

$$c = \bar{y} - m\bar{x}$$

Minimization

Simultaneous equations

Least-squares fit

Detailed derivation (Available in the Data Processing Guide on the Physlab website)

Example

- A student wants to check the resistance of a resistor by measuring voltage (V) across it and the resulting current (I) through it and then calculating the resistance through Ohm's Law.
- *Using slope of the relation*

$$V = R\,I$$

$$y = m\,x + c$$

Data from the experiment

Voltage (V)	Current (A)
11.2	4.67
13.4	5.46
15.1	6.28
17.7	7.22
19.3	8.30

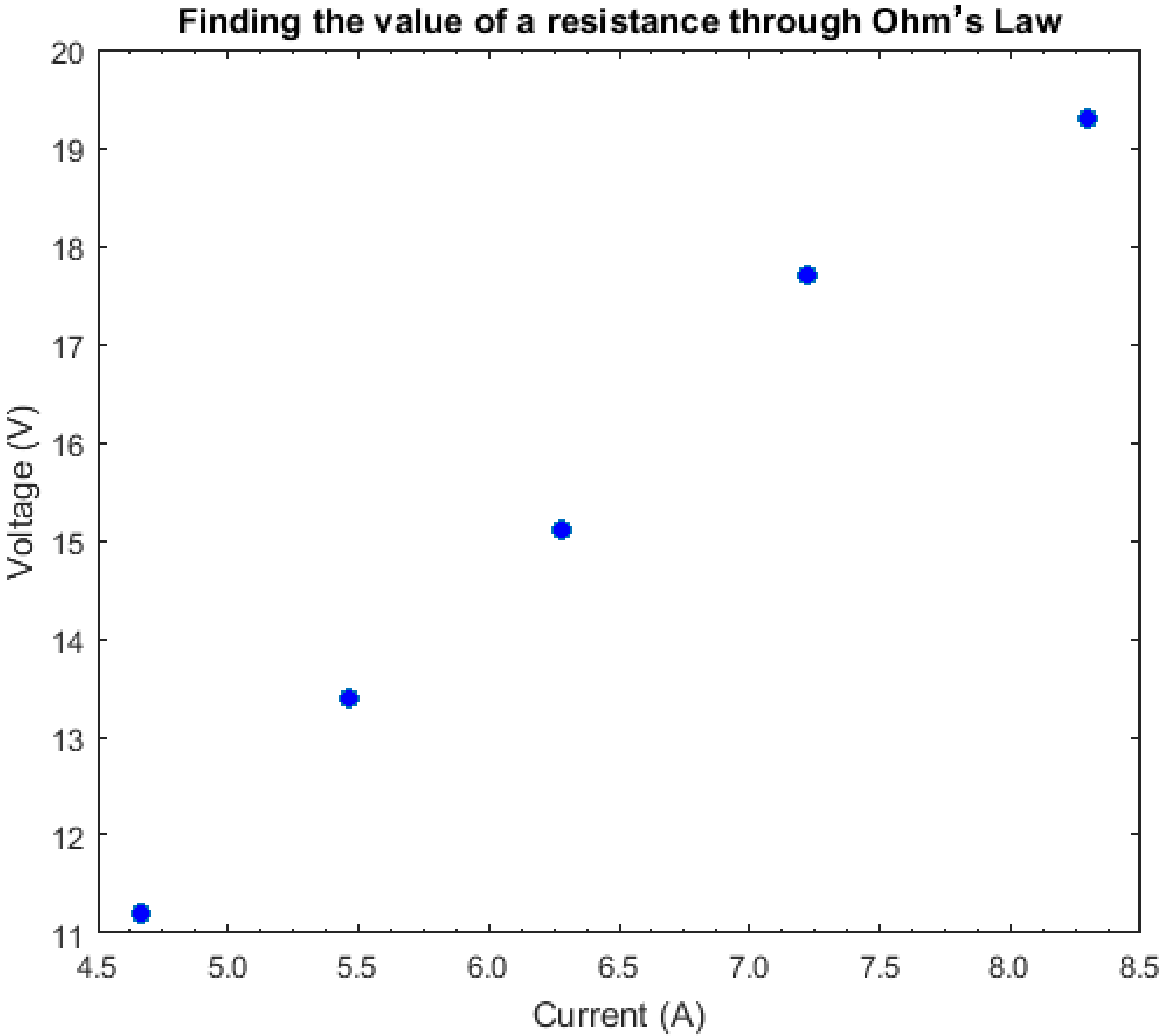
Curve Fitting Example

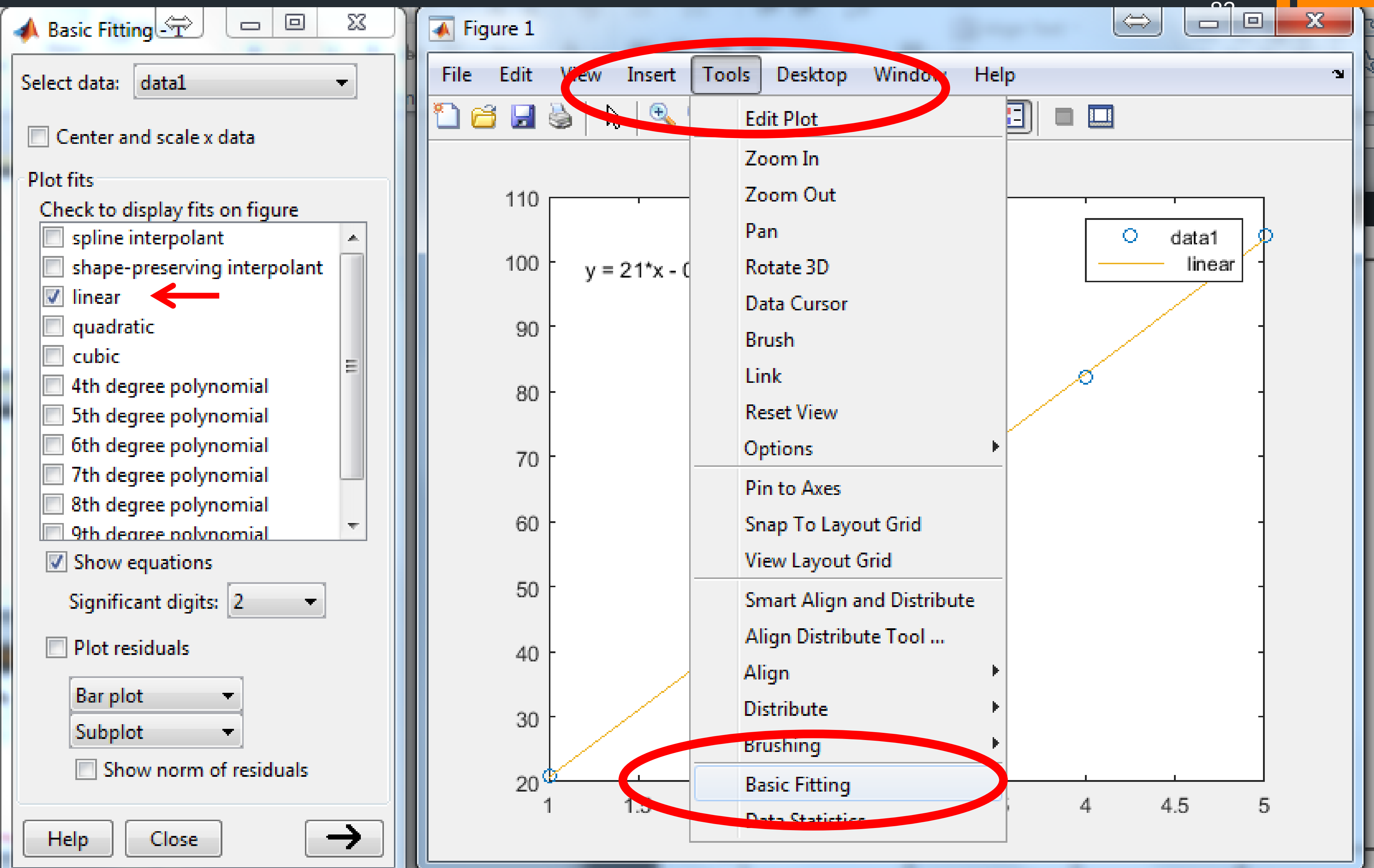
Plot the following experimental data:

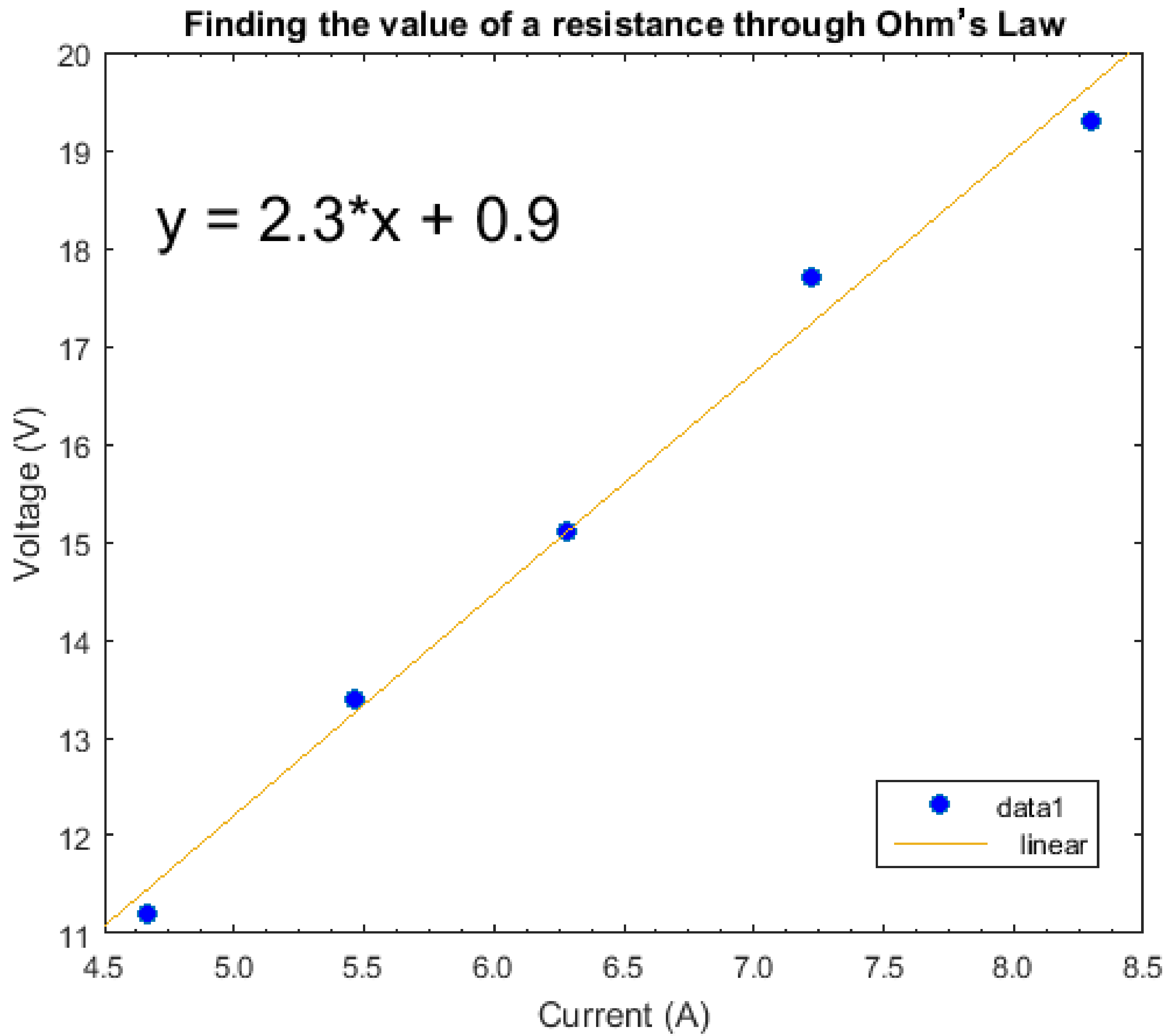
Voltage (V)	**11.2**	**13.4**	**15.1**	**17.7**	**19.3**
Current (A)	4.67	5.46	6.28	7.22	8.30

Commands:

```
v=[11.2 13.4 15.1 17.7 19.3];
a=[4.67 5.46 6.28 7.22 8.30 ];
plot(a,v,'o')
xlabel(`Current (A)')
ylabel(`Voltage (V))')
title('Finding the value of a resistance through Ohm's Law')
```

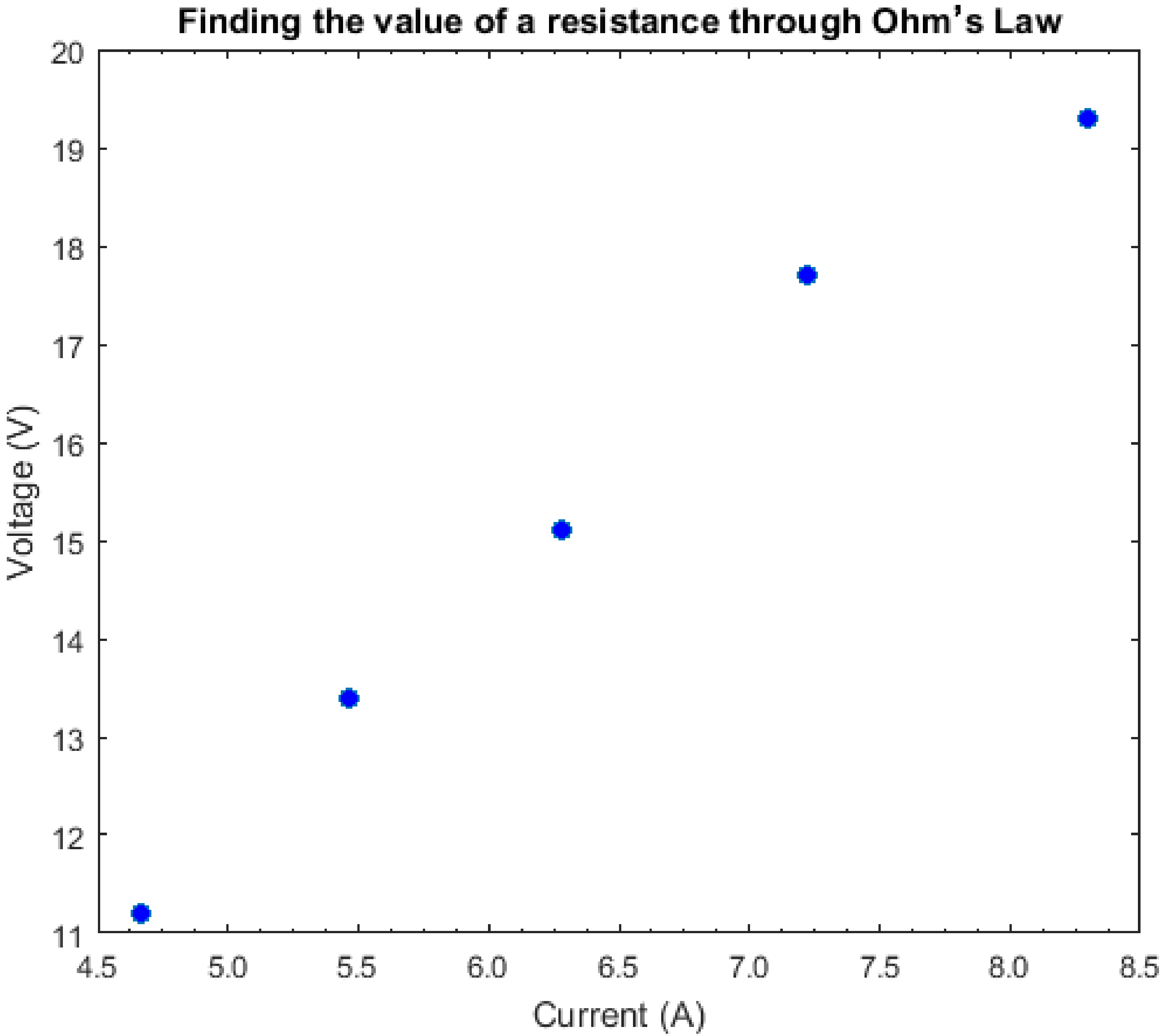

Finding the value of a resistance through Ohm's Law
Voltage (V)
Current (A)
20
19
18
17
16
15
14
13
12
11
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5


Basic Fitting
Select data:
data1
Center and scale x data
Plot fits
Check to display fits on figure
spline interpolant
shape-preserving interpolant
linear
quadratic
cubic
4th degree polynomial
5th degree polynomial
6th degree polynomial
7th degree polynomial
8th degree polynomial
9th degree polynomial
Show equations
Significant digits:
2
Plot residuals
Bar plot
Subplot
Show norm of residuals
Help
Close
Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
Edit Plot
Zoom In
Zoom Out
Pan
Rotate 3D
Data Cursor
Brush
Link
Reset View
Options
Pin to Axes
Snap To Layout Grid
View Layout Grid
Smart Align and Distribute
Align Distribute Tool ...
Align
Distribute
Brushing
Basic Fitting
Data Statistics
y = 21*x - 0
data1
linear
110
100
90
80
70
60
50
40
30
20
1
1.5
4
4.5
5

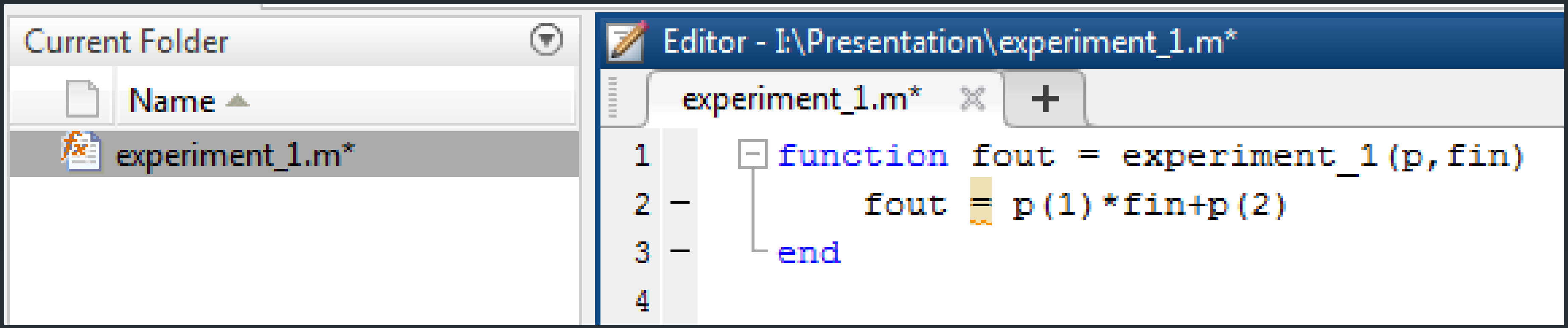

Finding the value of a resistance through Ohm's Law
y = 2.3*x + 0.9
Voltage (V)
Current (A)
20
19
18
17
16
15
14
13
12
11
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5
data1
linear

Curve Fitting

via lsqcurvefit command


Finding the value of a resistance through Ohm's Law
Voltage (V)
Current (A)
20
19
18
17
16
15
14
13
12
11
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5

1. Create a function file



2. Call the function via lsqcurvefit

experiment_1.m

```
function fout = experiment_1(p,fin)
    fout = p(1)*fin+p(2)
end
```

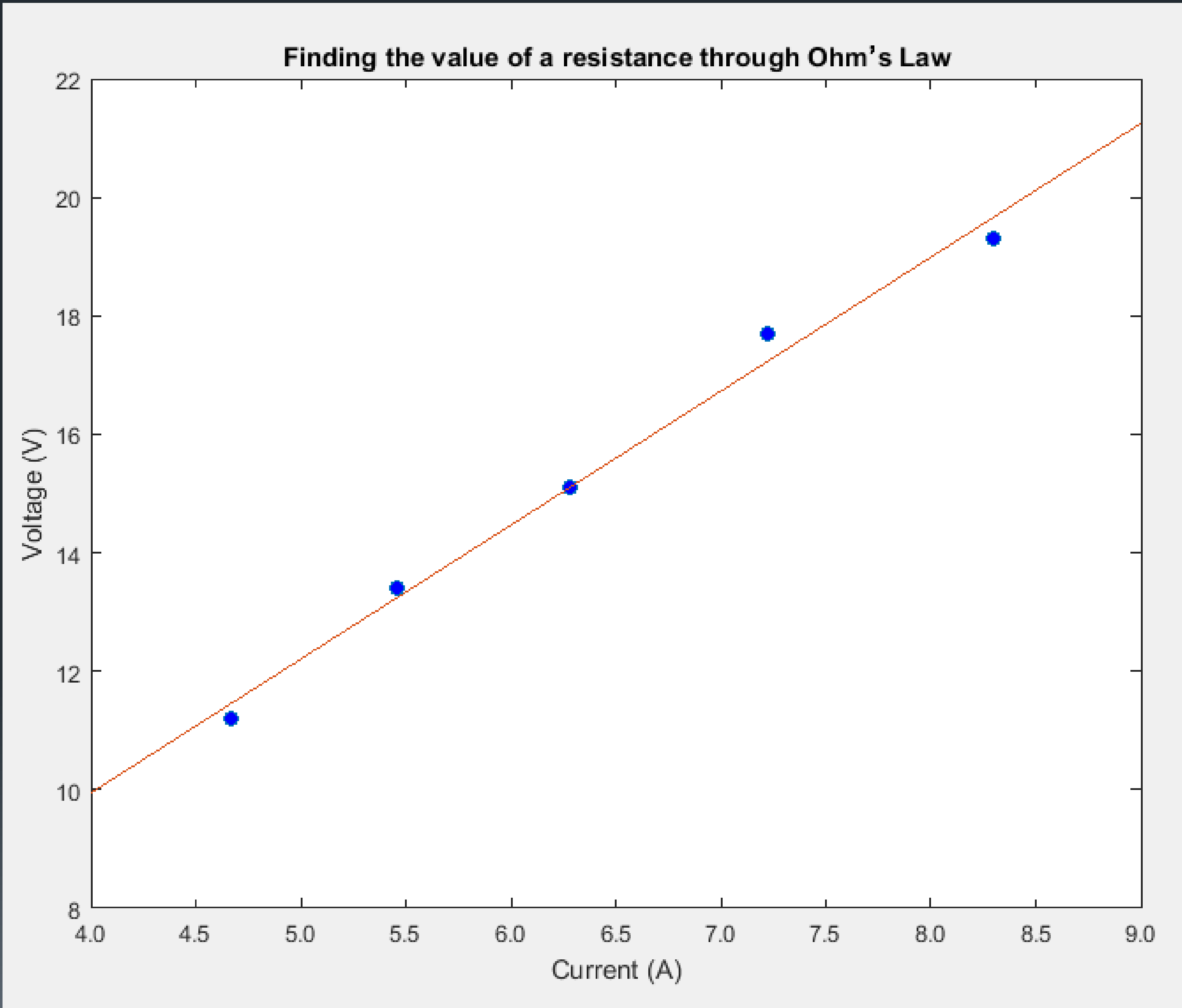
Command Window

New to MATLAB? See resources for Getting Started.

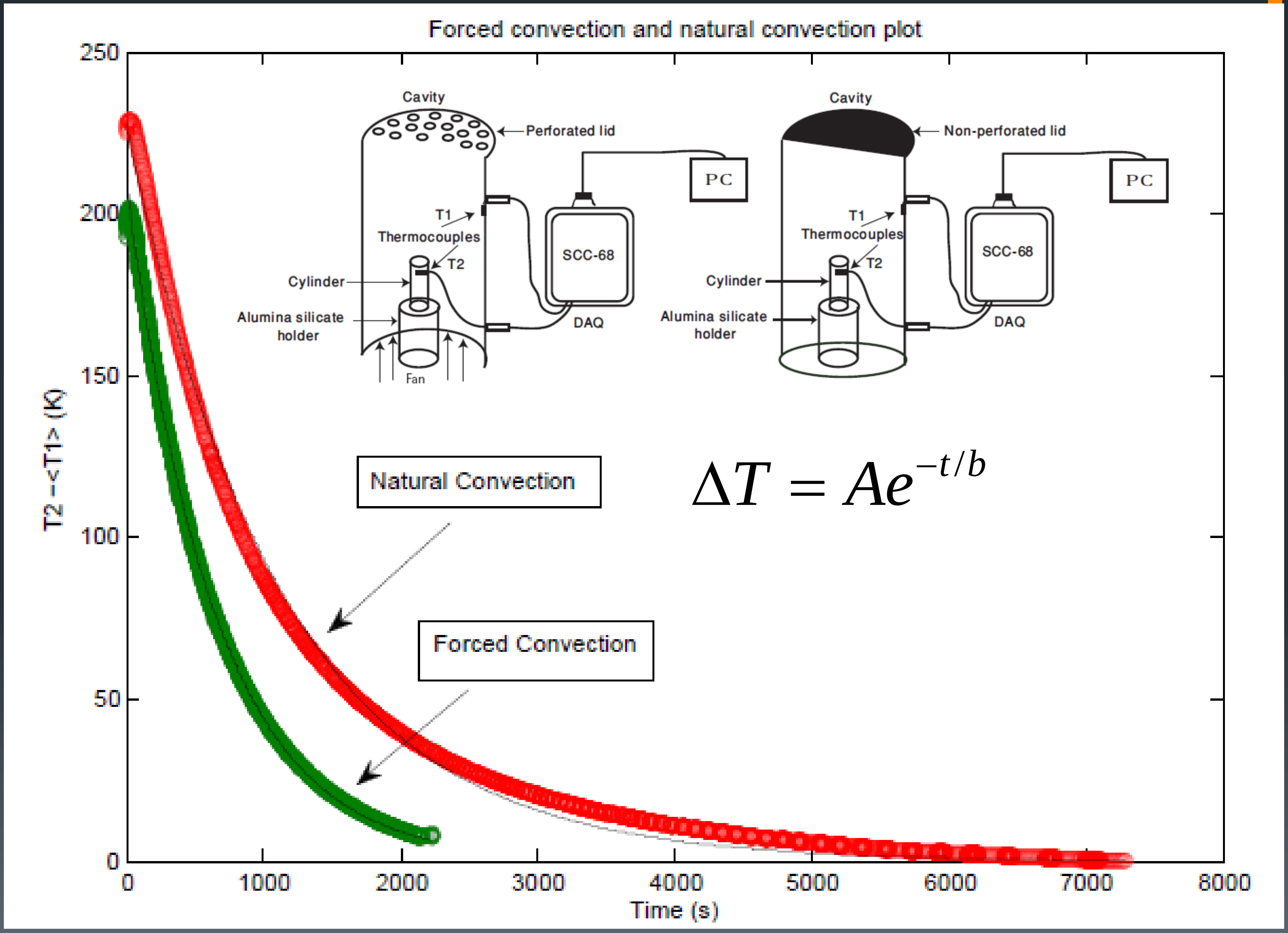
```
>> lsqcurvefit(@experiment_1, [1 1], I,V)
```

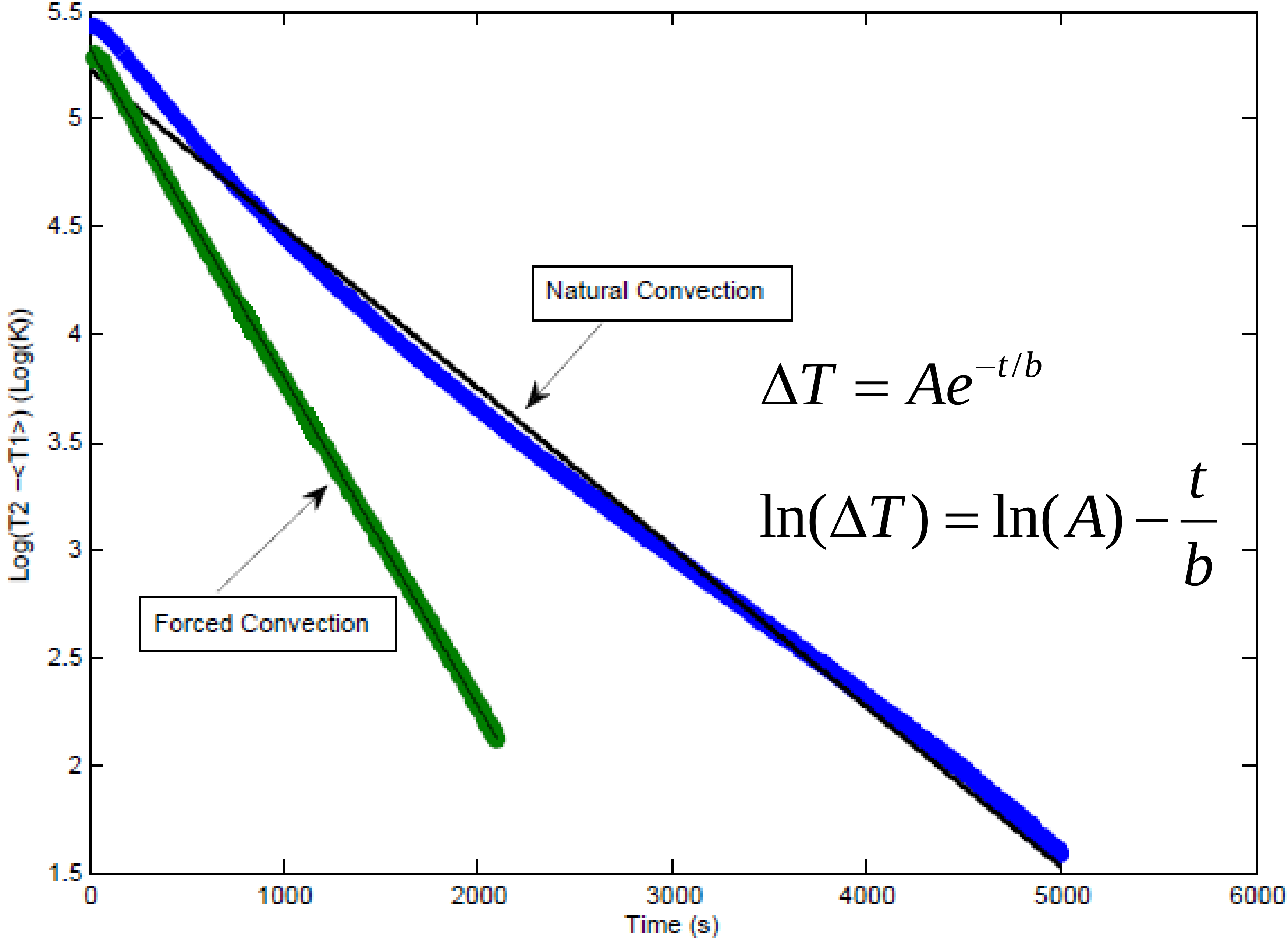
3. The function outputs the optimized values of parameters

```
Local minimum found.

Optimization completed because the size of the gradient is less than
the default value of the function tolerance.

<stopping criteria details>


ans =

    2.2605    0.9042

>>
```


Finding the value of a resistance through Ohm's Law
Voltage (V)
Current (A)
22
20
18
16
14
12
10
8
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5
9.0

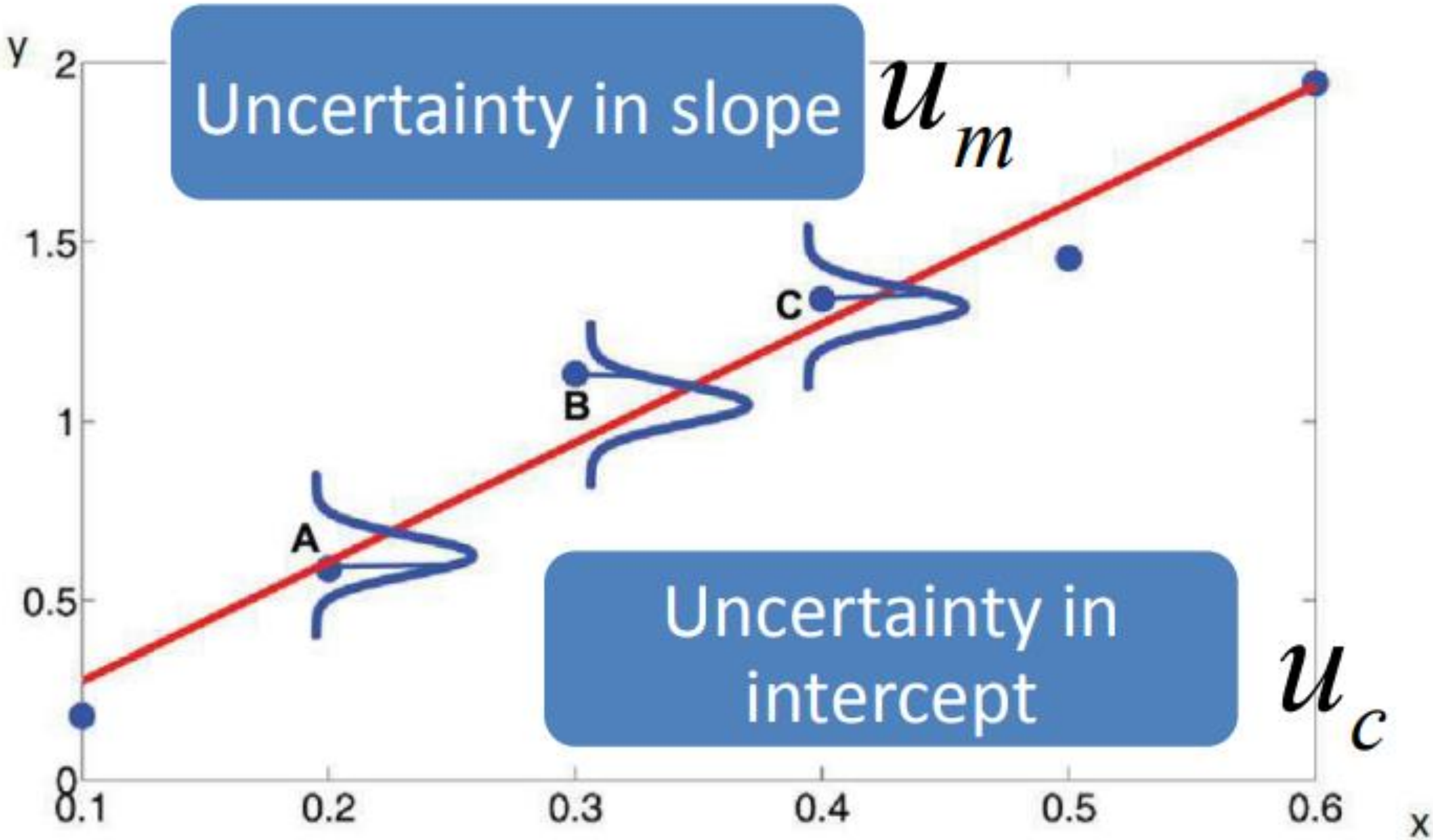
Linearizing Plots: Cooling Objects





Natural Convection
Forced Convection
Log(T2 −<T1>) (Log(K))
Time (s)
5.5
5
4.5
4
3.5
3
2.5
2
1.5
0
1000
2000
3000
4000
5000
6000
$\Delta T = Ae^{-t/b}$
$\ln(\Delta T) = \ln(A) - \frac{t}{b}$

Uncertainty in the least-squares fit



Uncertainty in slope and intercept

Uncertainty in slope

$$u_m = \sqrt{\frac{\sum_i^N d_i^2}{D(N-2)}}$$

Uncertainty in intercept

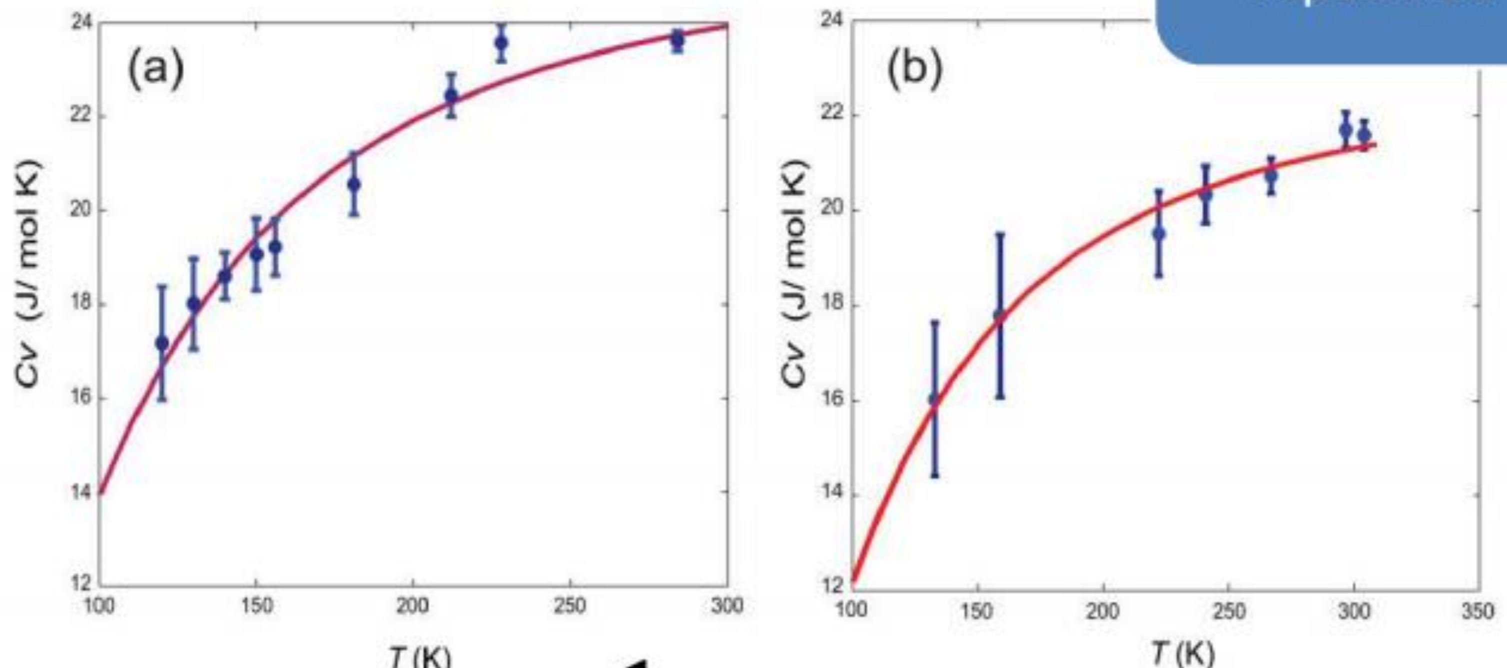
$$u_c = \sqrt{\left(\frac{1}{N} + \frac{\bar{x}^2}{D}\right)\left(\frac{\sum_i^N d_i^2}{(N-2)}\right)}$$

$$d_i = y_i - mx_i - c,$$

$$D = \sum_i^N (x_i - \bar{x})^2.$$

Concept of weighted fits

Measurement of heat capacities



$$w_i = \frac{1}{u_i^2}$$

Each data point is weighted differently.

Formulas for weighted fits

Slope and intercept

$$m = \frac{\Sigma_i w_i \, \Sigma_i w_i (x_i y_i) - \Sigma_i (w_i x_i) \, \Sigma_i (w_i y_i)}{\Sigma_i w_i \, \Sigma_i (w x_i^2) - (\Sigma_i w_i x_i)^2},$$

$$c = \frac{\Sigma_i (w_i x_i^2) \, \Sigma_i (w_i y_i) - \Sigma_i (w_i x_i) \, \Sigma_i (w_i x_i y_i)}{\Sigma_i w_i \, \Sigma_i (w_i x_i^2) - (\Sigma_i w_i x_i)^2}$$

Uncertainties therein

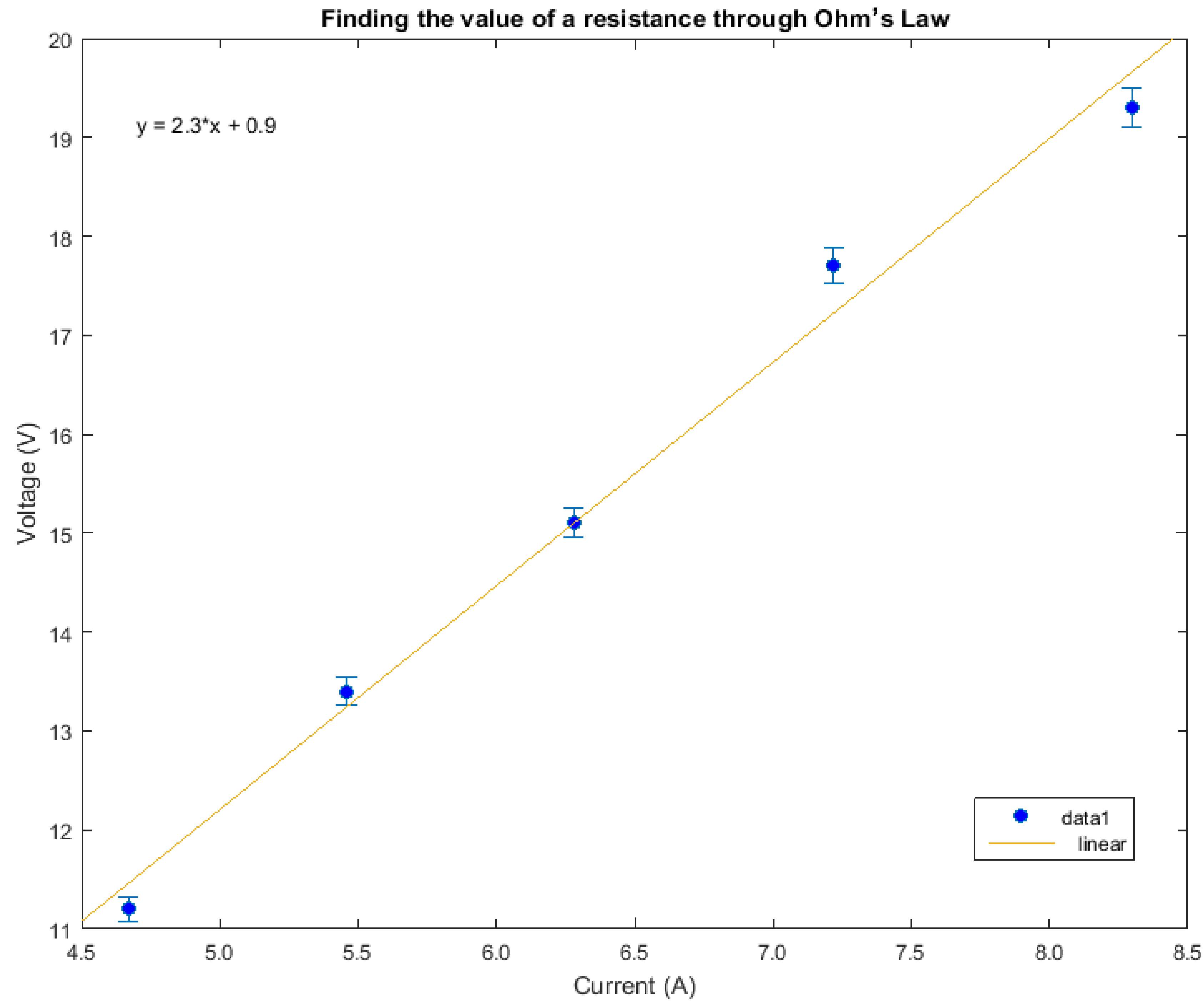
$$u_m = \sqrt{\frac{\Sigma_i w_i}{\Sigma_i w_i \, \Sigma_i (w_i x_i^2) - (\Sigma_i w_i x_i)^2}},$$

$$u_c = \sqrt{\frac{\Sigma_i (w_i x_i^2)}{\Sigma_i w_i \, \Sigma_i (w_i x_i^2) - (\Sigma_i w_i x_i)^2}}.$$

Error Bars

Uncertainties in Voltage and Current

Voltage (V)	Uncertainty due to resolution (U_s)	Uncertainty due to rating (U_r)	Total Uncertainty in Voltage (U_v)
11.2	0.03	0.11	0.12
13.4	0.03	0.13	0.14
15.1	0.03	0.15	0.15
17.7	0.03	0.18	0.18
19.3	0.03	0.19	0.20
Current (A)	**Uncertainty due to resolution (U_s)**	**Uncertainty due to rating (U_r)**	**Total Uncertainty in Voltage (I_v)**
4.67	0.003	0.047	0.047
5.46	0.003	0.055	0.055
6.28	0.003	0.063	0.063
7.22	0.003	0.072	0.072
8.30	0.003	0.083	0.083

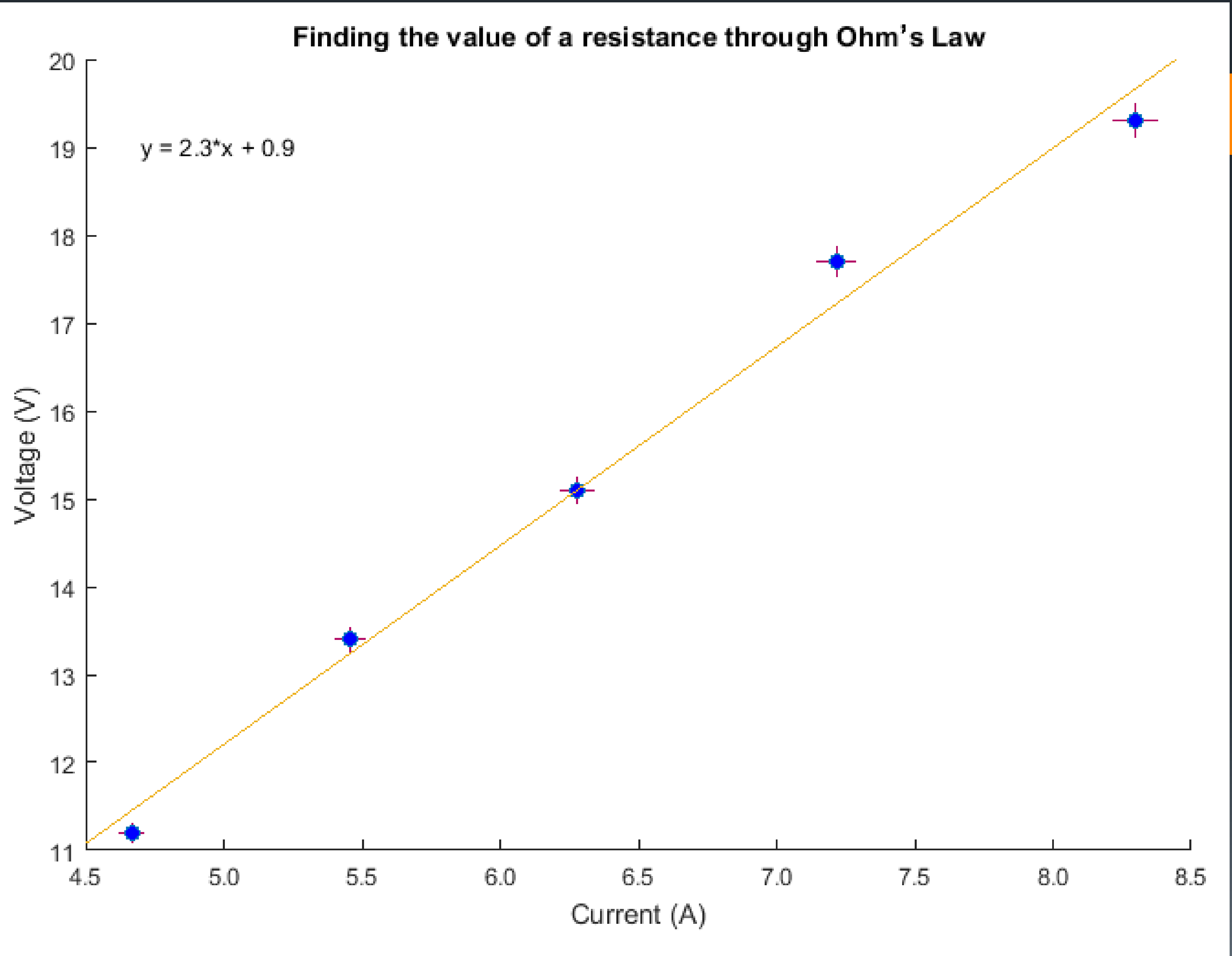

Finding the value of a resistance through Ohm's Law
y = 2.3*x + 0.9
Voltage (V)
Current (A)
data1
linear
20
19
18
17
16
15
14
13
12
11
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5

What if we want to plot uncertainties for both axis?

We'll use the "xyerrorbar" function file available on Physlab website.

Syntax:

```
xyerrorbar(x_vector, y_vector, u_x, u_y, 'o')
```



Finding the value of a resistance through Ohm's Law
y = 2.3*x + 0.9
Voltage (V)
20
19
18
17
16
15
14
13
12
11
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5
Current (A)

Let’s practice!

Solve last exercise of this session…

Introduction to MATLAB

End of session!