ENG1060: COMPUTING FOR ENGINEERS

Lab 8 – Week 9

2020 OCT NOV

Welcome to lab 8. Remember that laboratories continuously build on previously learned concepts and lab tasks. Therefore, it is crucial that you complete all previous labs before attempting the current one.

Self-study:
Students are expected to attempt these questions during their own self-study time, prior to this lab session. There may be questions that require functions not covered in the workshops. Remember to use MATLAB's built-in help for documentation and examples.

Learning outcomes:

1. To identify situations where curve fitting is required
2. To distinguish between linear and polynomial regressions
3. To differentiate between linear and non-linear models
4. To perform linearization on non-linear models
5. To use curve fitting techniques to develop relationships and interpolate/extrapolate data

Background:
Engineers are limited in their collection of data – collection at discrete values along a continuum. However, data may be needed at points between the discrete values. Therefore, it is necessary to develop relationships that are representative of the existing data, which may display characteristics consistent with non-linear models (e.g. exponential, power and saturation-growth rate) or polynomials. The developed relationship can be used to interpolate/extrapolate to the desired points.

Primary workshops involved:

- Workshop 8: Curve fitting

Assessment:
This laboratory comprises **2.5%** of your final grade. The questions are designed to test your recollection of the workshop material and to build upon important programming skills. You will be assessed on the quality of your programming style as well as the results produced by your programs during your laboratory session by the demonstrators. Save your work in **m-files** named ***lab1t1.m***, ***lab2t2.m,*** etc. Inability to answer the demonstrator's questions will result in zero marks, at the demonstrator's discretion.

Team tasks begin at the start of the lab session so please ensure you arrive on time to form your groups. Students who arrive late will not be able to participate in the team tasks as teams will have already formed and will therefore forfeit all associated marks. These tasks will be assessed during class.

Lab submission instructions

Follow the instructions below while submitting your lab tasks.

Team tasks:
The team tasks are designed for students to test and demonstrate their understanding of the fundamental concepts specific to that lab. These tasks will occur at the start of the lab and will be assessed on the spot. Demonstrators will advise on how these will be conducted. Most team tasks do not require the use of MATLAB but MATLAB should be used for checking purposes.

Individual tasks:
The individual tasks are designed for students to apply the fundamentals covered in the team tasks in a variety of contexts. These tasks should be completed in separate m-files. There is typically one m-file per task unless the task requires an accompanying function file (lab 3 onwards). Label the files appropriately. E.g. lab6t1.m, lab6t2.m, eridium.m, etc.

Deadline:
The lab tasks are due next Friday at 9am (MYT) or 12pm (AEDT). Late submissions will not be accepted. Students will need to apply for special consideration after this time.

Submission:
Submit your lab tasks by:

1) Answering questions in Google Form, and
2) Submitting one .zip file which includes all individual tasks.

The lab .zip file submission links can be found on Moodle under the weekly sections, namely Post-class: Lab participation & submission. The submission box ("Laboratory 8") will only accept one .zip file. Zipping instructions are dependent on the OS you are using.

Your zip file should include the separate m-files for the individual tasks including function files.

It is good practice to download your own submission and check that the files you have uploaded are correct. Test run your m-files that you download. You are able to update your submission until the deadline. Any update to the submission after the deadline will be considered late.

Grade and feedback:
The team will endeavour to grade your lab files by Tuesday of the following week. Grades and feedback can be viewed through the Moodle Gradebook, which is available on the left side pane on the ENG1060 Moodle site.

 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

Lab 8 – Assessed questions

TASK 1

[2 MARKS – L08TD]

Note: Team tasks are designed for students to recall material that they should be familiar with through the workshops and practice of the individual questions prior to this lab session.

Students will be split into groups of 3-4 for the team tasks. Students in each group must explain aspects of the question below to receive the marks. Ensure that everyone has equal learning opportunities. Additionally, ask your table for help.

<u>Curve fitting:</u>

Each group will be assigned one of the following non-linear models.

- Exponential model $y = \alpha e^{\beta x}$
- Power model $y = \alpha x^{\beta}$
- Saturation-growth model $y = \frac{\alpha x}{\beta + x}$

Complete the following:

1. Type the non-linear model name and the equation
2. Illustrate an example of the non-linear model on Cartesian axes.
3. List the steps to linearise the equation into the form of $y_{\text{lin}} = a_0 + a_1 x_{\text{lin}}$ and map the relationships between the nonlinear variables (e.g. y, x, α, β) and the linear variables (e.g. y, x, a_0, a_1).

$\boldsymbol{x}$	$\boldsymbol{y}$	$\boldsymbol{a_0}$	$\boldsymbol{a_1}$
???	???	???	???

4. Illustrate an example of the linearised model on Cartesian axes
5. List the steps required to obtain the nonlinear coefficients, starting from step 5
6. Discuss the task and explore any misunderstandings. Also, browse the work of other teams related to the other Codes and ensure that you have understood it as concepts from all sets may be required for the individual tasks.
7. Have a demonstrator assess your understanding

Remember good programming practices for all tasks even if not specifically stated. This includes, but is not limited to:

- using clc, close all, and clear all, where appropriate
- suppressing outputs where appropriate
- labelling all plots, and providing a legend where appropriate
- fprintf statements containing relevant answers

PRELIMINARY

Create a function file implementing the linear regression method following the function declaration provided below. This will help you consolidate your understanding of the techniques involved.

- Linear regression   function [a0,a1,r2] = linreg(x,y)

 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 2

[3 MARKS]

The solubility of oxygen in water *S*, is a function of the water temperature *T*. The solubility of O_2 as millimoles of O_2 per litre of water has been measured for several temperatures as shown in the table below. It is believed that the data follows the power relationship. i.e. $S = \alpha T^{\beta}$ where α and β are coefficients.

T **(°C)**	***S*** **(mmol O_2 /L H_2O)**
5	1.95
10	1.70
15	1.55
20	1.40
25	1.30
30	1.15
35	1.05
40	1.00
45	0.95

A. Show **by hand with pen and paper** the linearisation of this nonlinear model.
B. Use the linreg() function in MATLAB to ultimately determine the non-linear coefficients α and β and the r^2 value. Print the non-linear equation and r^2 value.
C. Plot the raw data as blue circles. Use the non-linear function to interpolate/extrapolate the solubility values for temperatures ranging 0°C to 60°C, and then plot this as a black continuous line on the same figure.
D. Determine the solubility of O_2 in water at a temperature of 50°C and plot this as a red diamond on the same figure. Print the non-linear equation as the title of the plot.

TASK 3

[5 MARKS]

Data for metabolic rate as a function of mass is provided below. It is believed that the data follows the power relationship. i.e. $metabolism = \alpha\, mass^{\beta}$ where α and β are coefficients.

Animal	**Mass (kg)**	**Metabolism (watts)**
Cow	400	270
Human	70	82
Sheep	45	50
Hen	2	4.8
Rat	0.3	1.45
Dove	0.16	0.97

A. Use the polyfit() function in MATLAB to determine the non-linear coefficients and r^2 value. Print the non-linear equation and r^2 value using fprintf. Use sprintf to print the non-linear equation into the title of the plot.
B. Plot the data as red diamonds. Then use the non-linear function to interpolate/extrapolate the metabolic rate values for the mass values starting from 0kg to 500kg. Plot this as a blue line on the same figure. Also, determine the metabolic rate of a 200kg tiger and plot this as a black diamond on the same figure.

 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 4

[5 MARKS]

The xy_data.txt file contains the x and y coordinates. Values in the first row correspond to the x coordinates and values in the second row correspond to the y coordinates.

A. Use importdata() to import the x and y coordinates. Plot y against x as black circles.
B. Determine the value of y at $x = 0$ using a 7th-order polynomial that is fitted to the data. Use fprintf to print this information. Example output is shown below.

    For 7th polynomial degree, y=??? at x=0

C. Use a **for loop** to fit the data with polynomial degrees of 2, 3, 5 and 9. For each polynomial degree, plot the fitted curve using x values from -3 to 3.5 using 100 points. Allow MATLAB to automatically colour the curves by not specifying a colour. Label your plot accordingly and include a legend.

TASK 5

[5 MARKS]

The acceleration of a heavily loaded vehicle can be modelled by the generalised saturation growth rate (SGR) model:

$$a = \frac{F[m]^n}{V^n + [m]^n}$$

where a is the acceleration of the vehicle, F and V are the nonlinear coefficients of the model, $[m]$ is the independent variable, and n is the order of the SGR model.

A. Show **by hand with pen and paper** the linearisation of this nonlinear model.
B. Write a function that performs the linear regression for this model, taking inputs of the raw/nonlinear x and y experimental data, and the order of the SGR model. Follow the function declaration provided below

```
function [F, V, a0, a1, r2] = LinRegrSGR(x, y, n)
```

C. Assume a second order SGR model (n=2) to determine the values of the nonlinear coefficients F and V given the experimental data below. Then, plot the **raw data below as red squares** and print the a_0, a_1 and r^2 values.

[m]	1.3	1.8	3	4.5	6	8	9
a	0.07	0.13	0.22	0.275	0.335	0.35	0.36

D. Interpolate/extrapolate this fitted function from m = 1 to 10 with a spacing of 0.1, and plot it with **a blue dashed line on the same figure from part C**. Place the legend in the north-west corner of the plot window.

2 marks deducted for poor programming practices (missing comments, unnecessary outputs, no axis labels, inefficient coding, etc.)

END OF ASSESSED QUESTIONS

The remainder of this document contains supplementary and exam-type questions for extended learning. Use your allocated lab time wisely!

 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

Lab 8 – Supplementary questions

These questions are provided for your additional learning and are not assessed in any way. You may find some of these questions challenging and may need to seek and examine functions that are not taught in this unit. Remember to use the help documentation. Coded solutions will not be provided on Moodle. Ask your demonstrators or use the discussion board to discuss any issues you are encountering.

TASK 1S

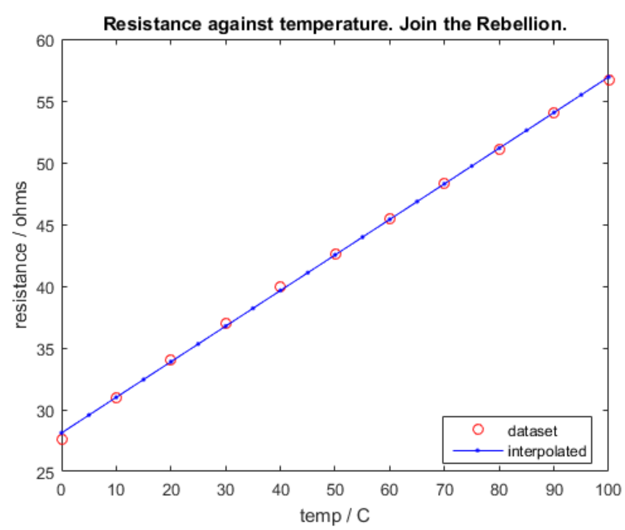
In a typical resistor, the resistance *R* in Ohms increases with an increase in temperature *T* (degrees Celsius). The temperature increments and the observed resistance values are given in the following table.

T (deg C)	0	10	20	30	40	50	60	70	80	90	100
R (Ohms)	27.6	31	34	37	40	42.6	45.5	48.3	51.1	54	56.7

A. Use the 'polyfit' function in MATLAB to calculate the coefficients of the 'linear' line of best fit ('a1' and 'a0'). Use these coefficients to determine the coefficient of determination, r^2.
B. Use the 'polyval' function to interpolate resistance values for temperatures between 0 and 100 with increments of 5 degrees C.

SOLUTION

```
polyfit: y = 0.2881x + 28.1227,   r^2 = 0.9994
```



 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 2S

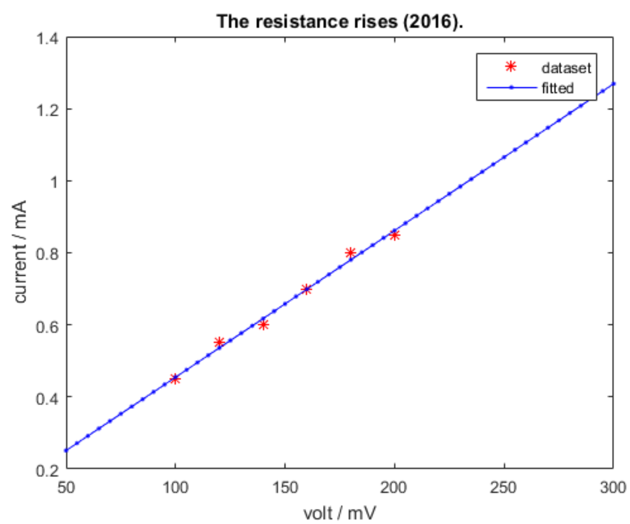
Consider the following data measured from a non-linear device:

Voltage / mV	100	120	140	160	180	200
Current / mA	0.45	0.55	0.60	0.70	0.80	0.85

A. Use the ‘polyfit’ function in MATLAB to calculate the coefficients of the ‘linear’ line of best fit (‘a1’ and ‘a0’). Use these coefficients to determine the coefficient of determination, r^2.
B. Confirm the a0, a1 and r^2 values through manual calculations (without using polyfit).
C. Use the ‘polyval’ function to predict current values for voltages between 50 mV and 300 mV with increments of 5 mV.

SOLUTION

```
    method        a0        a1        r2
   polyfit    0.0476    0.0041    0.9911
    linreg    0.0476    0.0041    0.9911
Both methods give the exact same values.
```



 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 3S

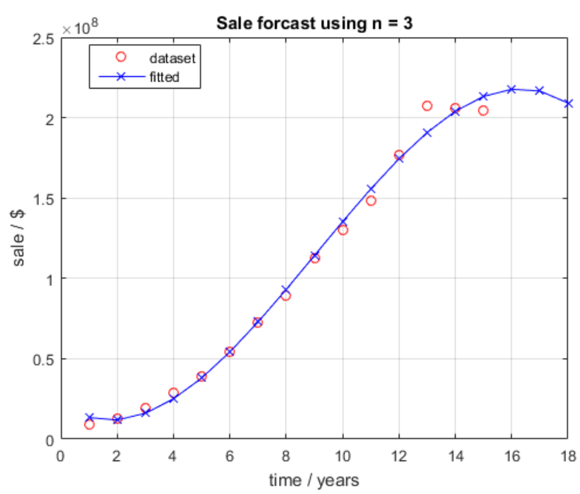
A sales manager wishes to forecast sales for the next 5 years based on data from the past 15 years. The data for the past 15 years is given in the following table.

Year	Sale ($)
1	9,149,548
2	13,048,745
3	19,147,687
4	28,873,127
5	39,163,784
6	54,545,369
7	72,456,782
8	89,547,216
9	112,642,574
10	130,456,321
11	148,678,983
12	176,453,837
13	207,547,632
14	206,147,352
15	204,456,987

Experiment with different polynomial degrees using the polyfit and polyval functions. Choose an appropriate polynomial to approximate the next three years and plot the results.

SOLUTION

```
Using polynomial order of 3 to fit the data.
      Year           Predicted sale/$
        16               217685877.44
        17               216583640.85
        18               209114251.44
```



 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 4S

Consider the following function

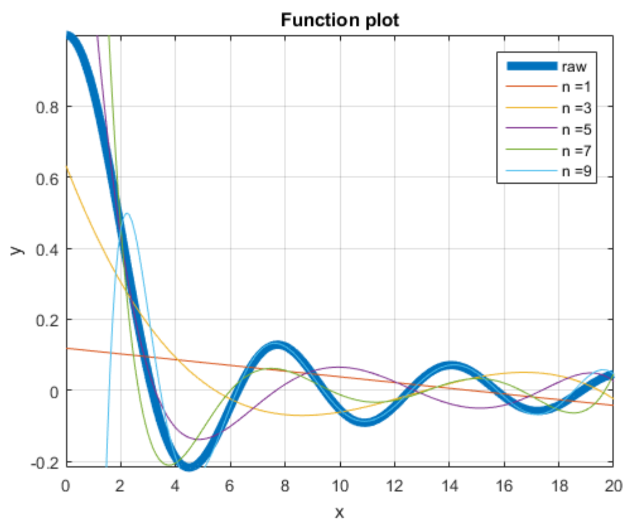
$$y = \frac{\sin(x)}{x}$$

and plot it for *x*=0 to *x*=20 with increments of 0.01.

Calculate the values of *y* for *x*=0 to *x*=20 with increments of 2. Use this data to fit multiple polynomial degrees (e.g. *n*=3, 5, 7, etc.) using polyfit and polyval. On the same figure as the previous, plot the fitted curves between *x*=0 to *x*=20 using increments of *x*=0.1. Remember to include a legend. Provide an fprintf statement on which polynomial order is appropriate for this fit.

SOLUTION

```
All fitted curves are inaccurate.
```



 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.

TASK 5S

Torricelli's principle of hydraulic resistance states that the volume flow rate *f* of a liquid through a restriction, such as an opening valve, is proportional to the square root of the pressure drop *p* across the restriction which can then be related to the volume of the liquid in the tank *V* through

$$f = c\sqrt{V}$$

where *c* is a constant. This can be generalised to

$$f = cV^e$$

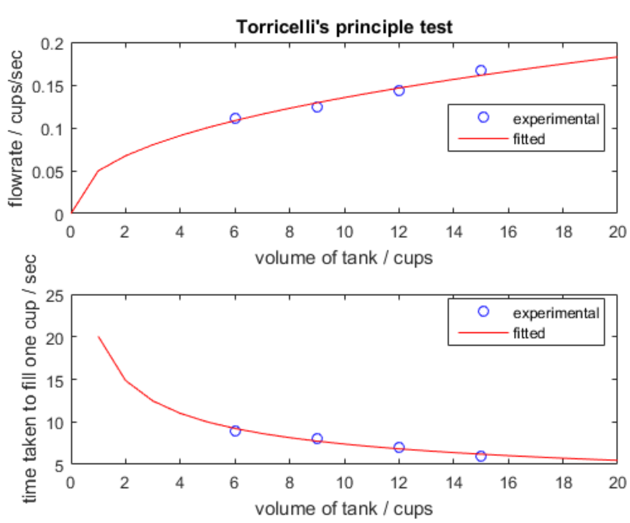
and an attempt can be made to experimentally verify that *e*=0.5. Several experiments were conducted with measurements shown in the following table, where the fill time *t* is the time taken to fill up one cup given the tank has a constant volume of *V*.

Volume *V* (cups)	Fill time *t* (s/cup)
15	6
12	7
9	8
6	9

Use polyfit together with the data to determine the constant *c* and the exponent *e*. Plot the raw data as blue circles and use polyval to interpolate values between *V*=0 to 20 with increments of 1.
Plot the flowrate and the time taken to fill one cup against the volume of liquid in the tank as separate subplots, using blue circle markers for the experimental data provided and a red solid line for the fitted model.

SOLUTION

```
e = 0.4331, close enough to theory e = 0.5.
Experimental model: f = 0.0499*V^0.4331
```



 Important: If you are struggling with a task, ensure that you have performed hand-written work (e.g. hand calculations, pseudocode, flow charts) to better understand the processes involved. Do this before asking demonstrators for help and use it to assist with your illustration of the problem.