Introduction to Scientific Computing Lab 6

MATLAB I

2017

1 Objectives

After completing these exercises you will be able to:

- Use the basic functionalities of MATLAB
- Create matrices and solve linear systems
- Plot data

2 Notes

Work individually

Create a directory for each week so you can come back to your codes in the future. Create files for each of the different exercises and name them in a logical manner, so `exercise1.m` for example.

When changing the source code, MATLAB will automatically save the code for you when you run.

When you have completed all exercises, ask a demonstrator to assess your work. They will test your code and ensure it is formatted well with good commenting, structure and variable names. This is a useful feedback mechanism, so listen to what the demonstrator has to say and their recommendations for improving your code.

In case of an error, read the compiler error. This will often tell you the line (or close to the line) where the error is occurring. Fix it, test and repeat for the errors you have. If you are getting nowhere then it can often be useful to copy the error into google, or use some keyword searches. If you are really stuck on one error then call over an assistant who will be able to point you in the right direction.

3 Data Fitting

In this exercise you will fit lines to given data points of increasing degrees of complexity. You will firstly fit a simple straight line to two data points, which will fit exacly. You will then fit a straight line to three data points, and a quadratic line to the same three points to investigate the differences. Finally, you will fit a line to a set of airline passenger data and use this to approximate future passenger growth.

3.1 Exercise 1: Linear Fit to Two Data Points

Create a new MATLAB script and fit a straight line through two data points given as $\mathbf{x}_1$ and $\mathbf{x}_2$. You are trying to solve the following equation for both known data points.

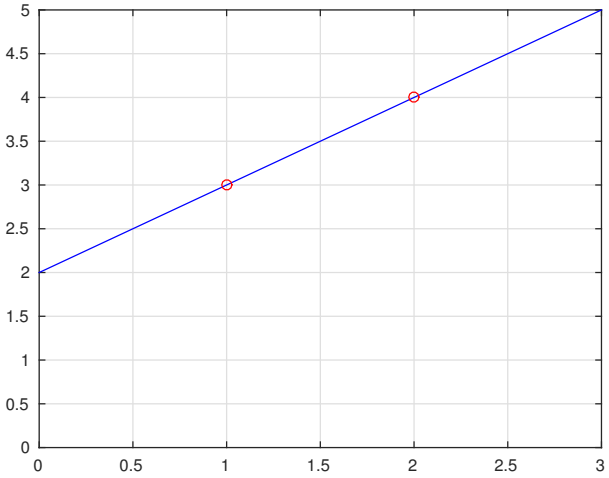
$$y = ax + b \tag{1}$$

To do this you should:

- Hard-code the following data points:

$$\mathbf{x}_1 = (1.0, 3.0) \quad , \quad \mathbf{x}_2 = (2.0, 4.0) \tag{2}$$

- Construct the linear system of equations and solve to find the coefficients a and b
- Plot:
  - The straight line fit from 0 to 3 in 0.1 intervals
  - The known data points
  - A grid, and an axis range of $x = 0:3$, $y = 0:5$
  - Your graph should look something like:



3.2 Exercise 2: Linear and Quadratic Fit to Three Data Points

Create a new script a fit two different lines (linear and quadratic) through three points $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$. For the quadratic fit you are trying to solve the following equation for the known data points

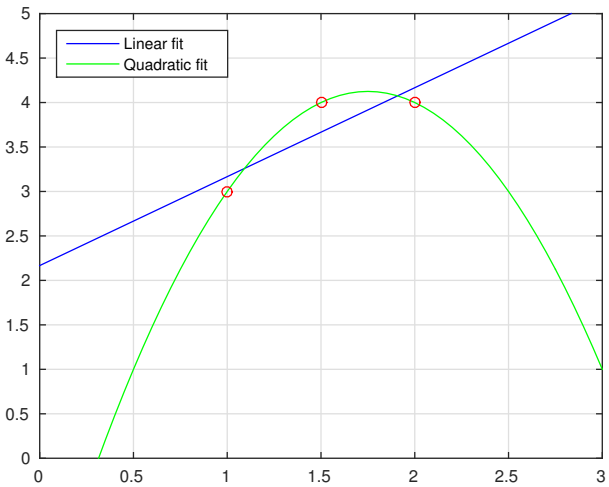
$$y = dx^2 + ex + f \tag{3}$$

You should:

- Hard-code the following data points:

$$\mathbf{x}_1 = (1.0, 3.0) \quad , \quad \mathbf{x}_2 = (2.0, 4.0) \quad , \quad \mathbf{x}_3 = (1.5, 4.0) \tag{4}$$

- Construct the linear system for the straight line and solve to find the coefficients a and b
  - NOTE: You will have a non-square matrix, hence you will have to use `pinv` to get an approximate inverse
- Construct the linear system for the quadratic line and solve to find the coefficients d, e and f
- Plot:
  - The straight line fit from 0 to 3 in 0.1 intervals
  - The quadratic line fit from 0 to 3 in 0.05 intervals
  - The known data points
  - A grid, and an axis range of $x = 0:3$, $y = 0:5$ and a legend
  - Your graph should look something like:



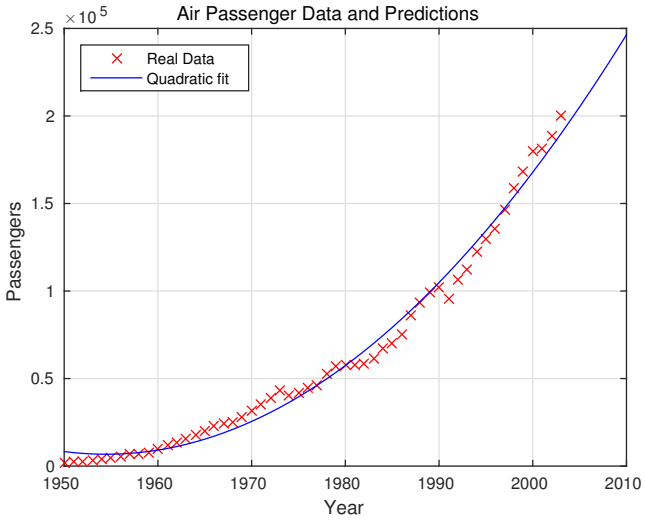
3.3 Exercise 3: Quadratic Fit to Airline Passenger Data

Create a new script. In this final exercise you will fit a quadratic curve to a set of historic airline passenger data. Firstly, download `passengerdata.mat` from blackboard. When you load it into MATLAB it will create two vectors for you, `xs` and `ys`. `xs` is the year entry for each piece of data and `ys` is the number of passengers the airline had in that year. To load this file into blackboard, you can do:

```
>> load('passengerdata.mat')
```

Now:

- Construct the linear system for the quadratic line and solve to find the coefficients d, e and f
- Plot:
  - The known data points
  - The quadratic line fit from 1950 to 2010 in 1 year intervals
  - A grid, and an axis range of $x = 1950 : 2010$, $y = 0 : 250000$ and a legend
  - A title and axis labels
  - Your graph should look something like:



3.4 Optional Extension

Extend exercise 3 to be able to easily change the order of your curve and fit polynomials of any given order. So in your script, you would have a line:

```
N = <something>
```

And then fit a curve that is described by:

$$y = \sum_{i=0}^{N} a_i x^i = a_0 + a_1 x^1 + a_2 x^2 + \cdots + a_N x^N \tag{5}$$

You will only be able to see the difference between N=1,2,3,4. Any higher and the system becomes extremely badly conditioned and the solutions stop changing.

HINTS:

- You will have to look at next week's notes for loop syntax
- The '.^' operator will perform elemental power indexing