

Essential Mathematics for Data Scientists

Assessment 2A: Creating functions

In this exercise, you must write some scripts to 'complete' a program. You are provided with a file `master.m` and you must write the two files `randXY.m` and `findSlope.m`.

The `master.m` file needs to accomplish two goals:

- a) generate some random data as part of a simulation
- b) identify the slope of that data.

We will use this to demonstrate how even simulated data with simulated noise can be difficult to analyse.

The `master.m` file calls on the two files `randXY.m` and `findSlope.m`. You must write these two files from scratch. Each file should only be a few lines long and define an appropriate function. The task that each file must accomplish is described as follows:

`randXY.m`

This file should define a function that takes one argument, N , which is length of vectors that will be returned. It will generate a random set of data with x spaced from 0 to 1 (taken from a uniform random distribution) and $y = 5x + 3$ with additional noise (taken from a normal distribution with a standard deviation of 2). You should use the functions `rand` and `randn`.

`findSlope.m`

This file should define a function that takes two arguments, the vectors x and y . It will then find the least-squares slope of the line of best fit for this data and return this value. You should use the function `polyfit`.

Note: the function `rng` is called at the start of the `master.m` script. This should make the 'random' results the same for each run of the code. So long as you generate the random values in the same order as the solution, you will obtain the same results as indicated in the comments in the code.

Marking scheme

You are required to write two function files: `randXY.m` and `findSlope.m`.

Each function file is worth 5 marks. 1 mark for each item listed below.

`randXY.m`

- Function is structured correctly (includes "function" and "end")
- One input: N
- Two outputs: x and y
- Random numbers, uniformly distributed: $x = \text{rand}(N,1)$;
- Random numbers, normally distributed: $y = 5x + 3 + 2*\text{randn}(N,1)$

`findSlope.m`

- Function is structured correctly (includes “function” and “end”)
- Two inputs: x and y
- One output: slope
- Calculate coefficients: `coeffs = polyfit(x,y,1);`
- Slope equals first coefficient: `slope = coeffs(1);`