

GG250 Lab 9

Fitting a straight line to data

Our employee Joe Blow was developing some Matlab code for solving the problem of fitting a straight line to an arbitrary (x,y) data set when he was hit in the head by a meteorite. It is now up to us to finish his quest and obtain best-fitting lines for two crucial data sets of interest to us. The class web page has Joe's Matlab script `gg250_lab9_fitline.m` and his two functions `gg250_lab9_misfit.m` and `gg250_lab9_plotline.m`. The last thing Joe said before his life was tragically cut short was "I think I got the case of 3 points to work!!" He then took a short break to get some fresh air and the rest is history.

We will first see if we can get Joe's code to work. It turns out there are some problems but we will work through those collectively. Download the files and change the names so they end with `_name.m` (name being your name as usual). You will also have to make that change in the function definition lines and the function calls. Once it works you are faced with these tasks:

1. Joe's code is poorly documented. Please add comments that make it clear to others what is going on.
2. At the moment, Joe's code uses a hard-wired, 3-point x-y data set, and he has the limits for the slope and intercept search specified directly in the script. Modify `gg250_lab9_fitline_name.m` so it becomes a function that expects 5 input arguments (x, y, slopes, intercepts, mode) and returns 3 output arguments (best_slope, best_intercept, best_E). Here, E stands for our measurement of misfit. i.e., the vertical discrepancies between the line and the data points.
3. None of the plots are labeled and there are no titles. Label the axes and provide titles that include " $y = ? * x + ?$, with $E = ?$ ", where you will place the actual values for the best fit solution instead of the question marks.
4. Draw a horizontal and a vertical line that intersect at the location of the minimum E value.
5. Joe was planning to use the mode argument to switch between two ways of measuring the misfit. Mode = 0 (which is all that he implemented) calculates the root-mean-square misfit which gives us the standard least-squares estimate for the line. However, in the presence of outliers the least squares method is known to have problems, and in that case we will instead implement mode = 1 which should calculate the median of the squared errors instead. Make the changes to the code needed to implement that option.
6. Create a short script `gg250_lab9_name.m` that uses your new function to find the best-fit lines for the two data sets `Hawaii_age_dist.dat` and `hertzprung_russell.dat` (available on the web page). Run your function for both choices for the mode setting, resulting in a total of four answers (two for each data set).

Your email to gg250-labs@hawaii.edu should state the 4 sets of (slope, intercept, E) for the 4 cases your script will run through, and comment on the results. In particular, does the choice of misfit function have any significant effect on your results? Attach the final files `gg250_lab9_misfit_name.m`, `gg250_lab9_fitline_name.m`, `gg250_lab9_plotline_name.m`, and `gg250_lab9_name.m`