L2 Regression Assignment (Michaelmas 2016)

This assignment should take about 10 hours to complete (from data familiarisation, code development through to report writing).

Car MPG data

The car MPG data was taken from a US study (Heavenrich et al, 1991; EPA/AA/CTAB/91-02), which measured the fuel efficiency of various cars and light trucks. As part of this study, each vehicle was characterised by its cab volume (VOL), power output (HP), maximum speed (SP), and weight (WT).

Data source

The tab separated value file can be found on duo under:

Level 2 Engineering Mathematics > Assessment > Regression Assignment > carmpgdat.txt.

You should copy this file into your Matlab working directory.

This can be loaded into your matlab environment using:

```
mpg = importdata('carmpgdat.txt', '\t', 1);
```

The data will then be placed into a Matlab data structure (`mpg`). Within this structure there are three elements: `mpg.data` (a 82x5 matrix of double precision values), and `diel.textdata` (containing the non-numerical data which includes the column headers in row 1 and the car models in column 1). Critically, the miles per gallon (MPG) is to be found in column 3 of `mpg.data`.

You will also find a template Matlab function (`mpg_regression.m`) which you should use as the basis for your code. Do not change either the file name or the first line in this file. This is needed for the automated code analysis.

Report and code

You will need to write some Matlab code to analyse the data and estimate the model parameters. These model parameters do not need to be reported, but you will need them to compute residuals for your models. An important transformation you need will be to compute the inverse of MPG: GPM (gallons per mile).

Specifically, your report should include three section, each briefly answering or describing the following points:

1. Provide a brief summary of the data you are analysing. You should provide and describe the following scatter plots as part of your summary:
   a) HP v MPG
   b) WT v MPG
   c) HP v GPM
   d) WT v GPM
2. Regression modelling. Populate a table containing the R^2, R^2_{adj}, p and K-S test results for the following data subsets: full data v MPG; HP, WT v MPG; SP, WT v MPG; HP v MPG; SP v MPG; WT v MPG; full data v GPM; HP, WT v GPM; SP, WT v GPM; HP v GPM; SP v GPM; and WT v GPM.

[Note: *p* represents the number of regression coefficiencts (including the constant term) and K-S test is the Kolomogorov-Smirnov test, you may use matlabs `kstest`)

3. Plot the residue histogram for the best and worst cases based on the R^2_{adj} values from your table. Comment on these results: what makes for a good and bad regression?

Helpful hints:

You may find that using Matlab's regression function (`regress()`) very helpful!

The Kolmogorov-Smirnov function (`kstest()`) will test a dataset to see if it appears to be drawn from a N(0,1) distribution. Remember to transform your residues appropriately!

Both these functions are part of Matlab's Statistics Toolbox, make sure this is accessible from your environment. If you have trouble, contact the IT Service desk.

Submission instructions

You need to submit the following via duo:

1. Brief report (as indicated above), submitted in PDF format.
   - The report should be no more than 2 pages with a minimum of 10 point font.
   - Do not forget to include your anonymous Z-code at the top of the document!
   - Use PDF format, as other file formats can get incorrectly displayed due to different software versions. This can include graphs and/or equations not displaying, and this will count against you.
2. Your Matlab code (save as `mpg_regression.m`).
3. These must be submitted (along with the statistics assignment files) as a single zip file. The zip file should not create subfolders. Name your zip file using your anonymous Z-code (eg: `Z0123456` should submit as `Z0123456.zip`). This ensures that all submissions have a unique file name.

Your report will be read and marked in conjunction with your code. The code will be tested and automated scoring will be applied (syntactic correctness, execution time), but may also be manually reviewed. Therefore, you should use an appropriate amount of commenting to guide the reader through your code!

Submission deadline: 14:00 on 16 January 2017 using TurnItIn via DUO.