Assessment 2019

Machine Learning and Statistics

Due: November 29$^{\text{th}}$, 2019

This document contains the instructions for Assessment 2019 for the module Machine Learning and Statistics. Your submission for this assessment must be in the form of a single GitHub repository and the URL for this repository must be submitted using the link on the Moodle page on or before October 11$^{\text{th}}$. You may use a private repository if you wish but make sure to add `ianmcloughlin` as a collaborator, again on or before October 11$^{\text{th}}$.

Your submission must be substantially your own work, otherwise it will not be considered for marking. In completing the project, you will be expected to carry out your own investigations and reference all sources used. Please be advised that all students are bound by the Quality Framework [4] at GMIT which includes the Code of Student Conduct and the Policy on Plagiarism.

Instructions

This assessment concerns the well-known Boston House Prices [1] dataset and the Python [3] packages scipy [2], keras [7], and jupyter [6]. The project is worth 100% of your marks for this module, but it is broken into three parts, as follows.

(20%) Describe: Create a git repository and make it available online for the lecturer to clone. The repository should contain all your work for this assessment. Within the repository, create a jupyter [6] notebook that uses descriptive statistics and plots to describe the Boston House Prices [1] dataset. This part is worth 20% of your overall mark.

(20%) Infer: To the above jupyter notebook, add a section where you use inferential statistics to analyse whether there is a significant difference in median house prices between houses that are along the Charles river and those that aren't. You should explain and discuss your findings within the notebook. This part is also worth 20%.

(60%) Predict: Again using the same notebook, use keras [7] to create a neural network that can predict the median house price based on the other variables in the dataset.

You are free to interpret this as you wish — for example, you may use all the other variables, or select a subset. This part is worth 60%.

Minimum Viable Project

The minimum standard for this assessment is a git repository containing a README file written in Markdown [5] and a jupyter notebook containing your work. The README should contain a summary of your work and provide instructions as to how to run the jupyter notebook and the web application. A better project will be well laid out, clear and concise, and easily understood and run.

Marking scheme

The following marking scheme will be used to mark the project out of 100%. Students should note, however, that in certain circumstances the examiner's overall impression of the project may influence marks in each individual component.

20%	**Describe**	Good summary of the dataset, repository well laid-out and organised. Reasonable commits to the repository.
20%	**Infer**	Appropriate analysis of the relationship between the median house price and whether the house borders the river, with good explanations of the analysis.
60%	**Predict**	Well-trained neural network providing decent predictions of house prices based on the other variables. Good explanations of how to use the neural network and how it works.

Advice for students

- Your git commit history should be extensive. A reasonable unit of work for a single commit is a small function, or a handful of comments, or a small change that fixes a bug. If you are well organised you will find it easier to determine the size of a reasonable commit, and it will show in your git history.

- Using information, code and data from outside sources is sometimes acceptable — so long as it is licensed to permit this, you clearly reference the source, and the overall project is substantially your own work. Using a source that does not meet these three conditions could jeopardise your mark.

- You must be able to explain your project during and after its completion. Bear this in mind when you are writing your README. If you had trouble understanding something in the first place, you will likely have trouble explaining it a couple of weeks later. Write a short explanation of it in your README, so that you can jog your memory later.

- Everyone is susceptible to procrastination and disorganisation. You are expected to be aware of this and take reasonable measures to avoid them. The best way to do this is to draw up an initial straight-forward project plan and keep it updated. You can show the examiner that you have done this in several ways. The easiest is to summarise the project plan in your README. Another way is to use a to-do list like GitHub Issues.

- Students have problems with projects from time to time. Some of these are unavoidable, such as external factors relating to family issues or illness. In such cases allowances can sometimes be made. Other problems are preventable, such as missing the submission deadline because you are having internet connectivity issues five minutes before it. Students should be able to show that up until an issue arose, they had completed a reasonable and proportionate amount of work and took reasonable steps to avoid preventable issues.

- Go easy on yourself - this is one project in one module. It will not define you or your life. A higher overall course mark should not be determined by a single project, but rather your performance in all your work in all your modules. Here, you are just trying to demonstrate to yourself, to the examiners, and to prospective future employers, that you can take a reasonably straight-forward problem and solve it within a few weeks.

References

[1] Boston Standard Metropolitan Statistical Area. Boston house prices dataset.
`https://www.kaggle.com/c/boston-housing`.

[2] SciPy developers. The scipy python package.
`https://www.scipy.org/`.

[3] Python Software Foundation. Welcome to python.org.
`https://www.python.org/`.

[4] GMIT. Quality assurance framework.
`https://www.gmit.ie/general/quality-assurance-framework`.

[5] GitHub Guides. Mastering markdown.
`https://guides.github.com/features/mastering-markdown/`.

[6] Project Jupyter. Project jupyter.
`https://jupyter.org/`.

[7] Keras Team. Keras: The python deep learning library.
`https://keras.io/`.