RMIT Vietnam University
School of Science and Technology

COSC2789 | Practical Data Science

Assignment 2: Data Modelling

Due: 23:59, Sunday in Week 8

This assignment is worth 40% of your overall mark.

Introduction

In this assignment, you will examine a data file and carry out the steps of the data science process, including the cleaning, exploring and modelling. You will need to develop and implement appropriate steps, in IPython, to load a data file into memory, clean, process, and analyse it. This assignment is intended to give you practical experience with the typical first steps of the data science process.

The “Practical Data Science” Canvas contains further announcements and a discussion board for this assignment. Please be sure to check these on a regular basis - it is your responsibility to stay informed with regards to any announcements or changes.

Where to Develop Your Code

You are encouraged to develop and test your code in two environments: Jupyter Notebook (or Jupyter Lab) on Lab PCs or your laptop.

Plagiarism

RMIT University takes plagiarism very seriously. All assignments will be checked with plagiarism-detection software; any student found to have plagiarised will be subject to disciplinary action as described in the course guide. Plagiarism includes submitting code that is not your own or submitting text that is not your own. Allowing others to copy your work is also plagiarism. All plagiarism will be penalised; there are no exceptions and no excuses. More information on Academic Integrity is available at https://www.rmit.edu.vn/students/my-studies/assessment-and-exams/academic-integrity

General Requirements

This section contains information about the general requirements that your assignment must meet. Please read all requirements carefully before you start.
_ You must do the analysis and modelling in Python Jupyter Notebook/Jupyter Lab.
_ Please ensure that your submission follows the file naming rules specified in the tasks below. File names are case sensitive, i.e. if it is specified that the file name is gryphon, then that is exactly the file name you should submit; Gryphon, GRYPHON, griffin, and anything else but gryphon will be rejected.

Task 1.1: Data Preparation (4%)

First of all, you have to register for a Kaggle account with your student email address. Then, you can use this Kaggle account to participate in our course Kaggle competition here: https://www.kaggle.com/t/cb6996c0cfb34595bae674f4e138e5e8

After accepting to participate in the competition, you can download the datasets to work offline. The first task in this Assignment 2 will be data cleaning, similar to the first task in Assignment 1.

Being a careful data scientist, you know that it is vital to carefully check any available data before starting to analyse it. Your task is to prepare the provided data for analysis. You will start by loading the CSV data from the file (using appropriate pandas functions) and checking whether the loaded data is equivalent to the data in the source CSV file. Then, you need to clean the data by using the knowledge we taught in the lectures. You need to deal with all the potential issues/errors in the data appropriately (such as: typos, extra whitespaces, sanity checks for impossible values, and missing values etc).

Note: These steps must be performed consistently for train/val/test sets.

Task 1.2: Data Exploration (9%)

Explore at least 3 columns or column pairs using appropriate descriptive statistics and graphs (if appropriate), e.g. the distribution of a numerical attribute, the proportion of each value of a categorical attribute. For each explored column/pair, please think carefully and report in your notebook:
1) the way you used to explore a column (e.g. the graph);
2) what you can observe from the way you used to explore it.

Please format each graph carefully, and use it in your final report. You need to include appropriate labels on the x-axis and y-axis, a title, and a legend. The fonts should be sized for good readability. Components of the graphs should be coloured appropriately, if applicable.

Note: These steps are for the training dataset only.

Task 2: Feature Engineering (10%)

Use suitable Python approaches to extract potential features for model input. Conduct appropriate analysis to evaluate feature importance (e.g. correlation analysis), then use suitable method(s) to select the final feature for the model. The feature choices must be explained via analysis.

Note: These steps must be performed consistently for train/val/test sets.

Task 3: Modelling (10%)

You have to train 3 different models to predict the values of the label column in the validation set. You must report 3 evaluation metrics (RMSE, MAE, and R2) for each of these 3 models.

A result table should look like this:

Model	RMSE	MAE	R2
Model 1	0.43	0.54	0.87
Model 2	0.23	0.56	0.86
Model 3	0.45	0.53	0.89

You must briefly describe your model structure/configuration.

After evaluation and comparing the model performance, you save your best model to a local folder and use the best model to predict the values of the label in the test set. Finally, upload your predicted submission.csv to Kaggle to compete against your classmates.

Other Evaluation Criteria:

Innovative Model (2%)

Out of the 3 selected models, there should be at least 1 innovative model (the other 2 can be a simple model). A simple model using only one algorithm for model training with some parameter tuning is not considered as an innovative model. For example, using a linear regressor from scikit-learn without any modification like this will be considered a simple model and won’t have any point:

```
from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(X, Y)
model.predict(X_val)
```

If you use a model from any research work, you must cite the reference correctly. A sample scale of innovative model is as below:
1 point: linearly stacking of multiple algorithms or a simple neural network.
2 points: complex stacking or complex neural network.

Peer Ranking (5%)

The peer ranking will be based on the Kaggle leaderboard in a batch of 5 students. #1-5 students will get 5pts, #6-10 students will get 4.5pts, ... , #46-50 students will get 0.5pts. If you rank below 50 or not submit to Kaggle, there will be no point.

Important: You must set the seed of both numpy and random package to 42, and any algorithms that have random_state parameter, you must set it to 42. This is to ensure reproducibility and avoid gaming the result of the Kaggle competition.

What to Submit, When, and How

The assignment is due at 23:59, Sunday in Week 8.

Assignments submitted after this time will be subject to standard late submission penalties. You need to submit the following files:

- Notebook file containing your python commands named "assignment2.ipynb".
    - For the notebook files, please make sure to clean them and remove any unnecessary lines of code (cells). Follow these steps before submission:
        - Main menu → Kernel → Restart & Run All
        - Wait till you see the output displayed properly. You should see all the data printed and graphs displayed.

- The dataset and python file(s) (if needed) of model training, inference, final model file and predicted test values must be included in the submission. A README file must be included with a clear instruction on how to train/test the model if not coded in the notebook. If it requires any Python library, a "requirements.txt" must be provided with specific version number.

They must be submitted as ONE single zip file, named as your student number (for example, 1234567.zip if your student ID is s1234567). The zip file must be submitted in Canvas: Assignments/Assignment 1. Please do NOT submit other unnecessary files.