Coursework – Data Analysis

This coursework is an individual assignment. You need to write your own Python code in a Jupyter Notebook. The deadline of submission is at noon on the 4th of January (Week 24), 2021 (UK time). The submission should be done via Canvas.

Submission of this coursework is mandatory. Some questions relating to this coursework will be presented in Online Test Part 1 (the 5th of January 2021). No marks will be granted to those coursework related questions if no coursework submission has been received from a student.

The programming language you should use to finish this assessment is Python (Version 3 and above only). In particular, you can use functions from the following packages: **Numpy, Pandas, Matplotlib, Seaborn** and **Sklearn**.

The Python skills needed to do this assessment are covered in the first five practical sessions – practical notes are available on Canvas.

The information of the dataset which you will work with can be viewed in the following link: https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/

Note that you should work on the white wine dataset (winequality-white.csv) only, which can be downloaded from 'Data Folder' in the link given above.

Task 1: Data pre-processing and data exploration

a. Use **Pandas** to load the data and report the number of data points (rows) in the dataset.
b. Consider "quality" as class labels. Report the number of features in the dataset and the number of data points in each class.
c. Perform random permutations of the data using the function, **shuffle**, from **sklearn.utils.** You must set a value to the parameter, **random_state**. Assign the data to a new variable as **white_wine**.

   The details on how to use **shuffle** can be viewed from the following link: https://scikit-learn.org/stable/modules/generated/sklearn.utils.shuffle.html
d. Produce one scatter plot, that is, one feature against another feature. You are free to choose which two features you want to use.

Task 2: PCA Analysis on the white-wine dataset Using Scikit-Learn

a. Perform a PCA analysis on the whole **white_wine** dataset.
b. Plot the data in the PC1 and PC2 projections and label/colour the data in the plot according to their class labels. Details on how to use **matplotlib.pyplot.scatter** can be viewed from the following link: https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.scatter.html

c. Report the variance captured by each principal component.

Task 3: Divide the white_wine dataset into a training set, a validation set, and a test set.

a. Take out the first 1000 rows from **white_wine** and save it as the validation set.

b. Take out the last 1000 rows from **white_wine** and save it as the test set.

c. Save the rest of rows from **white_wine** as the training set.

Task 4: Investigate how the size of the training dataset affects the model performance on the test set.

In this task, let us consider the last column 'quality' of the white_wine dataset as a real-valued target rather than a class label. You need to use the linear regression model to finish the following tasks (a)-(c). Note that you should use all available features in the dataset.

a. Produce a learning curve of the size of training set against the performance measurements. The performance should be measured on both the training set and the validation set. You need to choose at least 10 different sizes for the training set. For example, the first size may be 10% of the total training set produced in Task 3.
    - Remember to scale the corresponding training set and the validation set.
b. Report what the best training data size you would like to use for this work is and explain why you choose it.
c. Report the performance on the test set obtained using the model trained from the best size.
    - Remember to scale the corresponding training and test sets.

Task 5: Critical Discussion: write your conclusions using critical thinking (no more than 150 words) in your Jupyter notebook submission.

a. Summarize your findings for each task.
b. For Task 4, discuss whether there is any problem with that experimental design. If there is, what is it? How may you further improve it so that the experimental results are more reliable?

Hand in date: by noon on 04/01/2021 via Canvas.

What to submit:

A Jupyter Notebook submission showing your completed Python code and critical discussion. The submission should be identified by your student ID number.