FIT1043 Assignment 2 Description

Due date: Friday 16 Oct 2020 - 11:55 pm

This is an individual assessment and worth 20% of your total mark for FIT1043. You need to use Python in this assignment.

Aim

This assignment aims to analyse, visualise, and model data using Python. It will test your ability to:

1. Use **Python** to transform data and extract information
2. Use various graphical and non-graphical tools for performing exploratory data analysis and visualisation
3. Understand the strengths and weaknesses of certain visualisation tools
4. Implement and interpret the results of machine learning models in Python

Data

1. The data set "**CityPairs.csv**" contains information about the scheduled operations of international airlines operating to and from Australia. Data mainly shows the passengers, freight and mail carried between city pairs connected by a single flight number service. The description of each data column is shown in Table 1.

Table1: Description of the "CityPairs" columns

Column Name	Description
Month	A unique identifier for Year-Month
AustralianPort	Australian port where traffic is uplifted or discharged within a single flight number
ForeignPort	The foreign port where traffic is uplifted or discharged within a single flight number
Country	Based on the international uplift or discharge port within a single flight number
Passengers_In	Number of passengers inbound to Australia
Freight_In_(tonnes)	Freight inbound to Australia in tonnes
Mail_In_(tonnes)	Mail inbound to Australia in tonnes
Passengers_Out	Number of passengers outbound from Australia
Freight_Out_(tonnes)	Freight outbound from Australia in tonnes
Mail_Out_(tonnes)	Mail outbound from Australia in tonnes
Passengers_Total	Passengers_In+ Passengers_Out
Freight_Total_(tonnes)	Freight_In_(tonnes)+Freight_Out_(tonnes)
Mail_Total_(tonnes)	Mail_In_(tonnes)+Mail_Out_(tonnes)
Year	Year ranging from 1985 to 2016
Month_num	Month ranging from 1 to 12

2. The data set "**ClusteringData.csv**" contains information about the 'suicide rate' and 'GDP per capita'.

Hand-in Requirements

Please hand in **two** files including a **PDF file** containing your answer, and a **Jupyter notebook** file **(.ipynb)** containing your Python code to all the questions respectively. You need to consider the following cases for your submission:

1. PDF file should contain:
   Answers to the questions. Make sure to include screenshots/images of the graphs you generate in your report (you will need to use screen-capture functionality to create appropriate images). Moreover, please include your Python code, **not the screenshot of your codes**, to justify your answers to all the questions. The Turnitin would not be generated if you include a screenshot of your codes and you will lose **20% of the assignment mark** if you include a screenshot of the codes instead of writing/copying your codes.
   To generate a pdf report, you can use Word to write your report, but you need to convert it to PDF before your submission. Alternatively, an easier way is to generate a pdf version of your Jupyter notebook by using Ctrl+P in the Jupyter notebook. This pdf file is a mandatory requirement to check the Turnitin by Monash University.

2. Ipynb file should contain:
   Your Python codes for this assignment.

You will need to submit two **separate** files. "**Zip"**, "**rar"** or any other similar file compression format have a **penalty of 10%** of your assignment mark as they can not be processed by Turnitin system automatically.

You will be penalized by 5% of the assignment mark (5% out of 20 marks) if you submit after the due date for every day that you are late. If you could not submit your assignment before the due date, please make sure to submit your files at most 7 days after the assignment due date, we do not mark assignments which will be submitted after 23th of October 11:55 pm.

Assignment Tasks

There are three parts that you need to complete for this assignment. Parts A and B involve graphing and fitting regression lines to the "CityPairs.csv" data in Python, as well as analysing the results by answering further questions. Part C involves creating a proper clustering model for "ClusteringData.csv" data. **There are two challenge questions which you need to answer if you want to get 85 and above.**

Part A - Analysing Mail Flow in Australian Capital Cities

We will investigate the amount of traffic flowing to/from different Australian State Capitals in the following questions. The Australian State Capital ports are as follows: Adelaide, Brisbane, Darwin, Hobart, Melbourne, Perth, and Sydney.

Remember to add proper labels and titles to all plots in this assignment.

1. We want to explore the mail traffic flow of each Australian capital port. To calculate the mail traffic flow, you need to calculate the total Mail_In for each of the ports as well as the total Mail_Out for each of them. Create a bar chart which shows the total Mail_In and total Mail_Out for each of the ports and answer the following questions.
    1.1. Which city has the largest amount of mail flowing in?
    1.2. Can you properly compare the values for all cities by looking at the plot? Why?
    1.3. Why do you think the mail traffic amount is significantly higher for some of the ports?
2. Create a line chart to show the trend of total mail traffic to each of the following ports against year: Perth and Brisbane.
    2.1. How was the mail package traffic to each of the Brisbane and Perth ports in the mid-80s?
    2.2. How was the total mail traffic to Brisbane and Perth in 2016? (You should analyse the data and see if 2016 has anything different from the previous years or not. Make sure to use the output of the function "describe()" of Pandas data frame to answer this question).

Part B1 - Linear Regression and Prediction

We explore the annual freight at all Australian state capital ports against the year in this part. Create a scatter plot in Python showing the total annual freight in tonnes at all Australian state capital ports against the year and answer the following questions.

1. Does the data show a clear pattern? Describe the relationship you observe.
2. Are there any outliers? If so, use the IQR rule to remove the outliers. IQR rule is a simple method which can help you to detect the outliers based on the output of function "describe()".
3. Create a simple linear regression to model the relationship between Year and the Total Freight. Does the linear fit look to be a good fit? Justify your answer.
4. How fast is the total amount of freight increasing each year? [Hint: Think about what parameter in the regression model represents the rate of change]
5. What does the linear model predict for the total freight volume at Australian state capital ports in 2020?
6. Try fitting the linear model only to the data from the year 2005 onwards. What happens to the prediction for 2020? Which prediction could you trust more? Why?

Part B2 - Comparing Traffic Volumes

We explore the distribution of total monthly passenger traffic at all of the Australian ports in this question.

1. You first need to calculate the total number of Passengers_In and the total number of Passengers_Out for each unique month over the years for all the Australian ports. (i.e consider each month of each year as a unique month to calculate the total number of passengers. You can make use of the column "Month" for this). Next, create histograms to check the distribution of monthly Passengers_In and monthly Passengers_Out. Describe the distributions. Can you see any outliers in the plots? Discuss your answer.
2. Use boxplots to visualise the information of question 1.1. How many outliers can you see in the plots? Use the IQR rule to show the data points which are considered as outliers of monthly Passengers_In and monthly Passengers_Out)
3. As you can see in question 1.2, the information which is provided by a boxplot is so similar to the information which we saw in the output of the function "describe()". However, they are not the same. What are the differences between the information which are shown in a box plot and the output of function "describe()"?

Part C - Clustering task

Use the dataset 'ClusteringData.csv' to cluster the dataset using KMeans. Try different values of K and visualise your clusters. What is the best value of K based on your visualisation? Why do you think it is the best value for K? Describe the clusters which you see in your visualisation for the best value of K.

Challenge1:

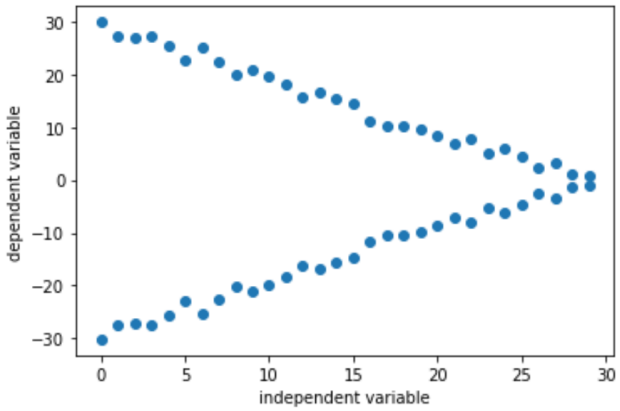
An important challenge in KMeans is about how to evaluate the quality of clusters. As there is no dependent variable for clustering models, we cannot check the accuracy or error of our model and we need other approaches to check the quality of the models. There are many other metrics/approaches which you can use to evaluate the performance of a clustering model; Silhouette score is one of them.

1. Explain how the Silhouette score works and what is the meaning of having the following results as a Silhouette score?
   1.1. Silhouette Score is 0.02
   1.2. Silhouette Score is -0.06
   1.3. Silhouette Score is 0.97
   1.4. Silhouette score is -0.9
2. Implement Silhouette score in Sklearn and find out the best number of clusters (K) based on the silhouette score.

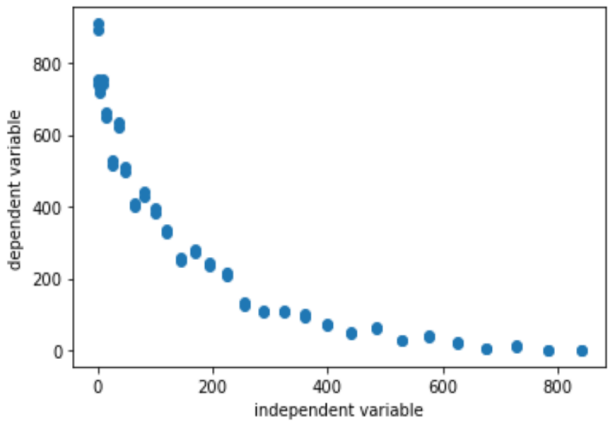
Challenge 2:

Imagine you are going to start working as a data scientist in a company. One of your tasks is to check if you can create a linear/polynomial regression model for a small dataset which has an independent variable and a dependent variable. As you have a small dataset, you would first plot the data and see the following relationship between your dependent and independent dataset.

```
plt.scatter(independent,dependent)
plt.xlabel('independent variable')
plt.ylabel('dependent variable')
plt.show()
```



Then, you would decide to use a transformation and transform the data to use a regression model. After applying the transformation, you plot the data again and this time you see the following relationship.



1. What is the transformation which you used?
2. Do you think your decision to transform the data with that transformation is a good idea? Discuss your answer.

Good Luck!