Deakin University
School of IT
Unit Team: SIT742

Trimester 2, 2023
Assessment Task 2 (Full Mark: 100)

SIT742: Modern Data Science

Extension Request Students with difficulty in meeting the deadline because of illness, etc. must apply for an assignment extension no later than 8:00pm on 20/09/2023 (Wednesday). Apply via '*CloudDeakin*', the menu item '*Extension Request*' under the '*Assessment*' drop-down menu.

Academic Integrity All assignment will be checked for plagiarism, and any academic misconduct will be reported to unit chair and university. Current usage of generative AI tools for SIT742 assessment should following the guideline II.

Instructions

Assignment Questions

There are total **2** parts in this assessment task:

Part 1 The first part will focus on the data acquisition and manipulation which includes the `numpy`, the `pandas`, the `Data Wrangling`, the `EDA` from **M03**, and **M04**.

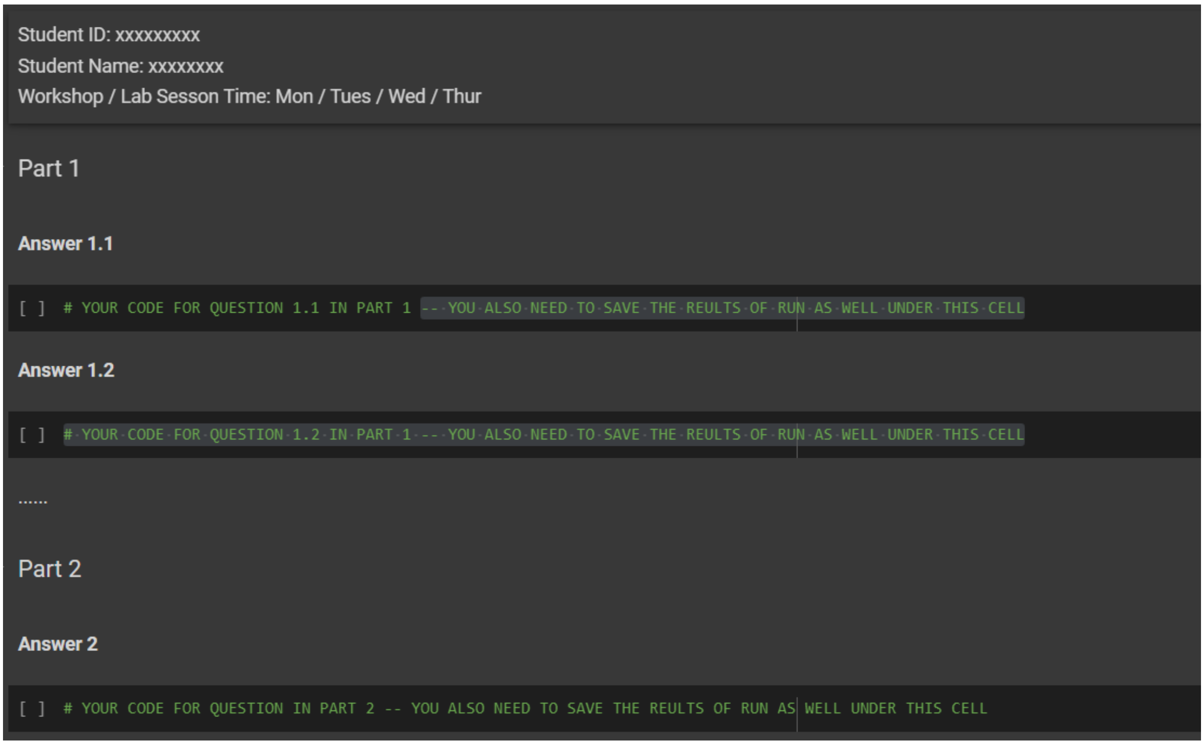
Part 2 The second part will require to perform the time series analysis exercise, which includes the `time series`, the `ARIMA`, the `IsolationForest` **M05**.

What to Submit?

You (your group) are required to submit the following completed files to the corresponding *Assignment* (Dropbox) in *CloudDeakin*:

`SIT742Task2.ipynb` The completed notebook with `all the run-able code` for all requirements (part 1 and part2).

In general, you (your group) need to complete, **save** the results of running, download/export the notebook as a local file, and submit your **notebook** from Python platform such as `Google Colab`. You need to clearly list the answer for each question, and the expected format from your notebook will be like in Figure 1 (**One notebook** for each group).



Student ID: xxxxxxxxx
Student Name: xxxxxxxx
Workshop / Lab Sesson Time: Mon / Tues / Wed / Thur

Part 1

Answer 1.1

```
[ ]  # YOUR CODE FOR QUESTION 1.1 IN PART 1 -- YOU ALSO NEED TO SAVE THE REULTS OF RUN AS WELL UNDER THIS CELL
```

Answer 1.2

```
[ ]  # YOUR CODE FOR QUESTION 1.2 IN PART 1 -- YOU ALSO NEED TO SAVE THE REULTS OF RUN AS WELL UNDER THIS CELL
```

......

Part 2

Answer 2

```
[ ]  # YOUR CODE FOR QUESTION IN PART 2 -- YOU ALSO NEED TO SAVE THE REULTS OF RUN AS WELL UNDER THIS CELL
```

Figure 1: Notebook Format

`SIT742Task2report.pdf` You (group) are also required to put your answer (code) and running results from `SIT742Task2.ipynb` into a `pdf` as the report for your task2 assignment (the code comments, and results including plot images are all required in the report, the code format such as Indentation should be same in the ipynb notebook).

In this report (**one for each group**), you will also need to cover all the questions (`Part 1` and `Part 2`). And you will need to provide a clear explanation on your logic for solving each question (you could write explanation below your solution and results). In the explanation, you will need to cover below parts: 1). why you decide to choose your solution; 2). are there any other solutions that could solve the question; 3). whether your solution is the optimal or not? why? The length of the explanation part for each question is limited below 100 words.

In the end of your report, you also need to discuss below three points:

- What you have learned with your team member from the second assignment.
- What is the contribution of each the team member for finishing the second assignment.

`SIT742Task2video.avi` A video demonstration between 10 and 15 minutes, and the file format can be other common video formats, such as '`MKV`', '`WMV`', '`MOV`' etc.

For the your group, one important submission is a **short video** in which each of *You* orally present the solutions that you provide in the notebook and illustrate the running of code line by line. In the video, your group need to work together to discuss below three points:

- Which question(s) you have worked on and how did you collaborate with other team members.
- What is the logic behind the your solution on the question(s) and is there any other optimized ways to resolve the question.
- What is your understanding of `Code collaboration`? How do you collaborate with coding in your group? What are the common tools to support the `Code collaboration`?

Part I

Data Acquisition and Manipulation

There are **10** questions in this part for total **60** marks, and each question is for **5** marks. The quality of your explanation in the report and video will be **10** marks for all questions.

You are required to use `Google Colab` to finish all the coding in the *code block cell*, and provide sufficient coding comments, and also save the result of running as well.

The (`Item_listing_category.zip`) data used for this part could be found in **here**. You will need to use `Pandas` to read the unzipped (csv) data for starting.

Question 1.1

Find the missing values:

- Write the function `missing_values_table` and use the dataframe as the input. The function should return the information of missing values by column (only for columns which have missing values and the returned value should be the count of rows has missing values);
- For columns which have missing values, could you impute the missing values with the mean value of the particular columns? (if you think it could not be done with mean value, write down the reason in comments and report rather than code)

Question 1.2

Find the price information from the data:

- Write code to print the median price of the items in the data;
- What is the 90th percentile value on the price;
- Draw the histogram chart for the price of the items in the data with 50 bins.

Question 1.3

Exploring the shipping information from the data:

- Write code to find out the percentage of the items that are paid by the buyers.
- Draw (two) histogram graphs in one plot on the price for seller pays shipping and buyer pays shipping (50 bins).
- When buying the items online, do you need to pay higher price if seller pays for the shipping? Write the code to find out (Compare the median price of items paid by buyers and items paid by sellers, and explain the result in the comment and report).

(Optional: You could use the subplot from `M04B-EDA`)

Question 1.4

You are required to find out the item condition information from the data. Lower the number (value), the better condition of the item.

- Write the code to find out (print) the count of the rows on each number (value) in column `item_condition_id`.
- Draw the boxplot graphs (one plot) on the price for each item condition value, and find out out

whether the better condition of the item could have higher median price (draw the plot and answer this question in the comment and report).

Question 1.5

Conduct the category analysis and find out the relevant information:

- Write the code to find out (print) how many unique categories you could find from column `category_name`.
- For the items with worst condition only (highest value from `item_condition_id`), write code to (print) find out the top 3 categories (now you probably understand the findings you had in Question 1.4).

Question 1.6

The categories in column `category_name` have 3 parts. The three parts (`main_cat`,`subcat_1` and `subcat_2`) are concatenated with '/' character sequentially in the data now.

- Write the **function** (must be function) to split the text content (string value in each row) in column `category_name` by `'/'` character. you need to handle the exception in the function for those has missing values (`NaN`). For missing values (`NaN`), the results from splitting should be "Category Unknown", "Category Unknown", "Category Unknown".
- Use the above function you wrote to create three new columns `main_cat`,`subcat_1` and `subcat_2` with corresponding values from the result of splitting. Print out the dataframe to show the top 5 rows for three new columns `main_cat`,`subcat_1` and `subcat_2`.

Question 1.7

After splitting the category for column `category_name`, we now have the three main details regarding to the category information. However, we need to clean the text in each of the new three columns in lowercase.

- Write code (or function) to change the text (value in each row) from the new three columns to lowercase.
- Draw the bar chart to find out the top 5 most popular `main categories` (in column `main_cat`) in the data (only showing the top 5).
- Write code (or function) to (print) find out how many unique `main categories` (in column `main_cat`), unique `first sub-categories` (in column `subcat_1`) and unique `second sub-categories` (in column `subcat_2`) respectively.

Question 1.8

Exploring the price and categories.

- Write code to (print) find out the median price for all the categories in new column `main_cat`.
- Draw the bar chart to find out the top 10 most expensive first sub-categories (in column `subcat_1`) in the data.
- Draw the bar chart to find out the top 10 cheapest second sub-categories (in column `subcat_2`) in the data.

Question 1.9

Exploring the price and brand.

- Write code to (print) find out the median price for all the brands (fill `NaN` with 'brand unavailable').
- Draw the bar chart to find out the top 10 most popular brands in the data.

Question 1.10

Item Description Analysis.

- Could you draw the wordcloud chart by using the column `clean_description`.
- Divide the data with quantiles of the price (using `qcut` from `pandas` to obtain the first/second/third/fourth quantile).
- Draw the wordcould by using the column `clean_description` on each quantile of price data.

Part II

Time series analysis exercise

There are **3** questions in this part for total **40** marks, and each question is for **10** marks. The quality of your explanation in the report and video will be **10** marks for all questions.

You are required to use `Google Colab` to finish all the coding in the *code block cell*, and provide sufficient coding comments, and also save the result of running as well.

The (`nyc_taxi.csv`) data used for this part could be found in **this link**. You will need to use `Pandas` to read the csv data for starting.

Question 2.1

The dataset used here is the New York City Taxi Demand dataset. The raw data is from the NYC Taxi and Limousine Commission. The data included here consists of aggregating the total number of taxi passengers into 30 minute buckets. In this question, we will simply process the data and explore the time series.

- Create two new dataframes `df_day` and `df_hour` by aggregating the demand value on daily and hourly level.
- Plot the demand value in two line charts for both `df_day` and `df_hour` dataframes.
- Plot the seasonal decomposition components (`Trend`, `Seasonal`, `Residual`) from `df_day` dataframe, also find out the p value from `adfuller test`. Do you think the `df_day` is stationary enough (please explain your reasons in comments and report)?

Question 2.2

In this question, we will try to use time series model such as ARIMA and others to build the model(s) for forecasting the future.

- Create the `acf` and `pacf` plots for `df_day` dataframe.

- Find the best model with different parameters on ARIMA model. The parameter range for p,d,q are all from $[0, 1, 2]$. In total, you need to find out the best model with lowest `Mean Abosulate Error` from 27 choices based on the time from ”Jul-01-2014” to ”Dec-01-2014”.
- Using the best model in above steps to forecast the time from ”Jan-01-2015” to ”Jan-31-2015”. Plot the predicted value and the true demand value from ”Jan-01-2015” to ”Jan-31-2015”.
- Could you think of **any other model** (not as same as ARIMA) could do the forecasting for demand value from ”Jan-01-2015” to ”Jan-31-2015”? You could choose **one** model (except ARIMA) and train the model based on the demand value from ”Jul-01-2014” to ”Dec-01-2014” (same training data as the ARIMA). Hint: there are some resources regarding other time series forecasting models such as `prophet` here and also the `exponential smoothing` here.

Question 2.3

In this question, we will detect the anomaly within the `df_day` dataframe.

- Create the `Weekday` column according to the `timestamp` column in `df_day` dataframe. The value in `Weekday` column should be from [’Monday’, ’Tuesday’, ’Wednesday’, ’Thursday’,’Friday’, ’Saturday’, ’Sunday’]. Also create the `Hour`, `Day`, `Month`, `Year`, `Month_day` (numeric format on day of the month), `Lag` (yesterday’s demand value), and `Rolling_Mean` (rolling 7 days mean demand value, minimized period is 1) 7 new columns in `df_day` dataframe according to the `timestamp` column.
- Using `Isolation Forest` with above crafted features in `df_day` to find out the date which is identified as ’outlier’.

Guideline for the Application of Generative Artificial Intelligence in SIT742 Assignments

This section delineates the expectations for SIT742 students regarding the conscientious and ethical employment of generative Artificial Intelligence (AI) tools in their assignments.

Introduction

Generative AI, with its capacity to spontaneously produce content like audio, code, written text, images, and videos, harbors enormous transformative potential for society and economy. Despite its vast opportunities, there are significant risks and challenges associated with its use.

In academia, generative AI can be a valuable tool, aiding in tasks such as text summarization and refinement for improved readability. However, it also introduces potential pitfalls including IT security issues, intellectual integrity and property protection concerns, and potential breaches of confidentiality. When data is input into generative AI tools, it enters the public domain, accessible by undefined third parties, thereby opening up debates about the rightful authorship of the generated content.

As generative AI technologies and their applications continue to evolve rapidly, they will undoubtedly introduce new facets and concerns to consider.

Directives for Using Generative AI in SIT742 Assignments

AI tools, such as `ChatGPT`, can generate a broad range of content, from essays and code to poems, based on user input. While these tools serve as a valuable platform for learning and exploration, they should not supersede original work and critical thought. Therefore, SIT742 students are urged to be prudent when employing generative AI tools in their assignments. When incorporating AI tools in SIT742 assessments, please adhere to these guidelines:

Reference AI-generated content Acknowledge and cite the AI tool as the source if you utilize or alter any AI-generated content in your assessment. Use the Harvard referencing style for proper citation format. Neglecting to accurately cite AI-generated content may lead to plagiarism allegations and academic misconduct penalties.

Ensure content appropriateness[Refrain from employing AI tools to generate content that is irrelevant, inappropriate, offensive, or harmful to yourself or others. Uphold the intellectual property rights of others by avoiding the use of AI tools to generate content that infringes on copyrights.

Exercise responsible and ethical use of AI tools Misusing or abusing AI tools in ways that contradict the university's policies, values, or codes of conduct is unacceptable.

Refrain from using AI tools for coding tasks Coding tasks are intended to assess your programming skills and knowledge, which cannot be evaluated through AI-generated code. Using AI tools to generate code for coding tasks will result in a zero score for that portion of the assessment and may also lead to academic misconduct penalties.

By employing AI tools in SIT742 assessments, you consent to comply with these rules and accept the repercussions of any violation. Deakin University strictly enforces its policy against plagiarism or collusion. The Deakin Academic Integrity Committee will review and make decisions on allegations of academic integrity breaches involving AI tools or other sources. Students could face severe consequences, including exclusion from the unit or the university in extreme cases.

For any questions or concerns about using AI tools for university assessments, please reach out to the teaching team for clarification.