Assignment – 1

Predictive Modelling of Eating-Out Problem

Due on Sunday, 29th September (23:59)

Objective

In this assignment, you will be provided with a real-world dataset, and you are required to use the feature engineering, modelling and deployment skills that you have gained in this unit so far to draw some conclusions supported by code and graphs, building predictive models, and deploying your work on a public repository.

Data Description

This assignment is accompanied by a real-world dataset containing more than 10K records for restaurants in the Sydney area in 2018. For every record, information about the restaurant goes from basic details such as name, address, and location to advanced information such as rating. You must use your data science skills to **predict the restaurant's success using different machine learning algorithms**. The table below describes the variables in the attached data.

Table 1. Data columns and their description

Column	Description	Example
'address'	restaurant's address (text)	371A Pitt Street, CBD, Sydney
'cost'	the average cost for two people in AUD (numeric)	50.0
'cuisine'	cuisines served by the restaurant (list)	[Thai, Salad]
'lat'	Latitude (numeric)	-33.876059
'link'	Url (text)	https://www.zomato.com/sydney/sydney-madang-cbd
'lng'	longitude (numeric)	151.207605
'phone'	phone number (numeric)	02 8318 0406
'rating_number'	restaurant rating (numeric)	4.0
'rating_text'	resturnat rating (text)	Very Good
'subzone'	The suburb in which restaurant resides (text)	CBD
'title'	restaurant's name (text)	Sydney Madang
'type'	business type (list)	[Casual Dining]
'votes'	Number of users who provided the rating (numeric)	1311.0
'groupon'	is the restaurant promoting itself on Groupon.com? (boolean)	False

Tasks

The tasks of this assignment are divided into three parts as follows:

Part A –Importing and Understanding Data (20 marks)

In this part, you are expected to:

- Understand the dataset and develop intuition about the data.
- Document an exploratory data analysis and, whenever possible, conclude the analysis.
- Employ popular graphical modules (e.g., matplotlib and seaborn) to answer the questions below.

1- **Provide plots/graphs to support**:

  - How many unique cuisines are served by Sydney restaurants?
  - which suburbs (top 3) have the highest number of restaurants?
  - "*Restaurants with 'excellent' ratings are mostly costly while those with 'Poor' ratings are rarely expensive*". Do you agree with this statement or not? Please support your answer with numbers and visuals. (hint: use stacked bar chart or histogram to relate 'cost' to 'rating_text')

2- Perform exploratory analysis for the variables of the data. This can be done by producing histograms, distribution plots, and descriptive insights about these variables. This can be performed at least for the following variables.

  - Cost
  - Rating
  - Type

3- **Produce Cuisine Density Map**: Using the restaurant geographic information and the provided "sydney.geojson" file, write a Python function to show a cuisine density map in which each suburb is colour-coded by the number of restaurants that serve a particular cuisine.

(Hint: use the spatial join in geopandas)

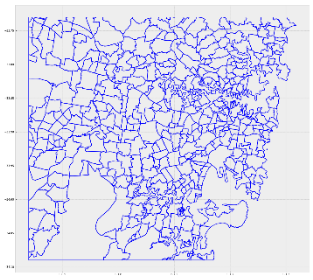
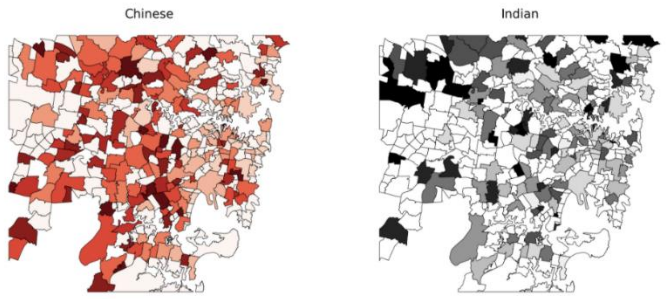
"sydney.geojson" file plotted	Example output from *show_cuisine_densitymap*
	

Figure 1. Example of input and output of the density map function

4. Please investigate employing interactive plotting libraries (such as Bokeh, Plotly, ... etc.) in a use case where you think the non-interactive plotting is limiting. Explain the limitation and how the interactive libraries will solve it.

5- Tableau Dashboard for quick insights: Can you generate a Tableau dashboard that visualises some graphs/plots to answer some of the EDA questions above? Also, can you share this dashboard on the Tableau public?

Part B – Predictive Modelling (20 marks)

In this part, you are expected to apply predictive modelling to predict/classify the success of the restaurants.

I. **Feature Engineering**: Implement the feature engineering approaches to:

1. Perform data cleaning to remove/impute any useless records in the predictive task (such as NA, NaN, etc.)
2. Use proper label/feature encoding for each feature/column you consider, preparing the data for the modelling step.

II. **Regression:**

3. Build a linear regression model (model_regression_1) to predict the restaurant rating (numeric rating) from other features (columns) in the dataset. Please consider splitting the data into train (80%) and test (20%) sets.
   [Hint: please use sklearn.model_selection.train_test_split and set random_state=0 " while splitting]
4. Build another linear regression model (model_regression_2) using the Gradient Descent as the optimisation function.
5. Report the mean square error (MSE) on the test data for both models.

III. **Classification:**

6. Simplify the problem into binary classifications where class 1 contains 'Poor' and 'Average' records while class 2 contains 'Good', 'Very Good' and 'Excellent' records.
7. Build a logistic regression model (model_classification_3) for the simplified data, with 80% training data and 20% test data.
   [Hint: please use sklearn.model_selection.train_test_split and set random_state=0 " while splitting]
8. Use the confusion matrix to report the results of using the classification model on the test data.
9. Draw your conclusions and observations about the performance of the model relevant to the classes' distributions.
10. Repeat the previous classification task using three other models of your choice (example suggestions here (on Scikit-Learn website) and report the performance.

Part C – Deployment (10 marks)

Step 1: Deploy the code on GitHub

In this step, you are required to deploy your source code with its dependencies to a repository and then push this repository to your GitHub account.

- Please use the Git commands to connect your code files to the created repository
- Push the committed files to the GitHub repository
- Create a “readme.md” markdown file that describes the code of this repository, how to run it, and what the user would expect if they got the code running.

Step 2: Deploy a Docker image to the Docker Hub

In this step, you need to create a Docker image with all the trained models, the data, and code to run these models one after another and produce the results. The user using this Docker image will be able to see the output results (accuracy, confusion matrix, etc.) of applying all three models to the accompanying data.

Deliverables

You are required to submit a compressed (e.g. ZIP) file to the Canvas website of the unit with the following files:

1- Python Jupyter Notebook(s) with the code for parts A & B with all explanation, discussion and insights added as Markdown cells or comments into the notebook(s).

2- A PDF document with the following:

   a. Link to the Tableau Dashboard

   b. The results of the (regression and classification) trained models on the test data. These results must be listed in two tables: one for the regression and another for the classification.

   c. The list of commands you have used to deploy your source code to the GitHub repository.

   d. The list of commands you have used to create and push the Docker image to the Docker Hub.

   e. The Link of the source code you have deployed on GitHub (please add me as a collaborator; my GitHub account is radwanebrahim@gmail.com)

   f. The link to the Docker image you deployed on the Docker Hub.