Assignment – 1

Predictive Modelling and Deployment

Due on Sunday, 3 October (23:55)

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 knowledge that you have gained in this unit so far to draw some conclusions, supported by code and graphs, build predictive models and deploy your work on public repository.

Data Description

This assignment is accompanied with a real-world dataset containing more than 10K records for restaurants existing in the Sydney area in the year 2018. For every single record, information about the restaurant goes from basic details such as name, address and location to advanced details such as rating. You are required to use your data science skills to predict the restaurant's success using different machine learning techniques.

In the lecture of week 7, we invite Dr Abdelwahed Khamis (Postdoctoral Research Fellow @CSIRO) who wrote the original script to crawl the data and has shared it with us for the assignment purposes. He will tell us the story behind the dataset and how and why it was collected. Then, he will explain the dataset composition and share some visual insights (some of which you will be asked to reproduce in this assignment). The below table shows the description of the variables in this data. For more information about the data, please check the recording of the guest lecture (under week 7).

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 the following three parts:

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 draw conclusions about the analysis;
- Employ popular graphical modules (matplotlib and seaborn) to answer below questions.

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' rating are mostly very expensive while those with 'Poor' rating are rarely expensive*". Do you agree on this statement or not? Please support your answer by 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 and distribution plots and descriptive insights about these variables. This can be performed at least for the following variables.

  - ○ Cost
  - ○ Rating
  - ○ Type

3- **Produce Cusine Density Map**: Using the restaurant geographic information and the provided "sydney.geojson" file, write a python function to show a cuisine density map where each suburb is colour-coded by the number of restaurants that serve a particular cuisine. This function can be called as: "show_cuisine_densitymap(cuisine='Indian')". (Hint: use the spatial join in geopandas)

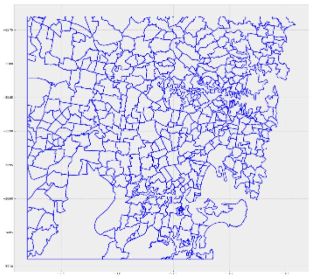
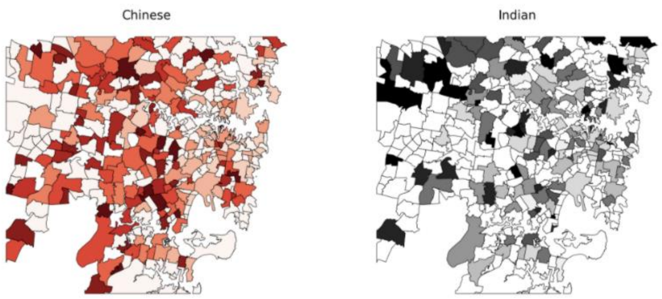
"sydney.geojson" file plotted	Example output from *show_cuisine_densitymap*
	

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

Bonus: (*i.e.,* Extra marks for this section)

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.

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 records that are useless in the predictive task (such as NA, NaN, etc.)
2. Use proper label/feature encoding for each feature/column you consider making the data ready for the modelling step.

II. **Regression:**

3. Build a linear regression model (model_regression_1) to predict the restaurants 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) with 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, where training data is 80% and the test data is 20%.

   [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.

Bonus: 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 GitLab

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

- Please use the Git commands to connect your code files to the created repository
- Push the committed files to the GitLab repository
- Create a "readme.md" markdown file that describes the code of this repository and how to run it and what the user would expect if 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 with the data and code to run these models one after another and produce the results. So, the user who would use this Docker image will be able to see the output results (accuracy, confusion matrix etc.) of applying all the three models on the accompanied 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. Results of the (regression and classification) trained models on the test data. These results need to be listed into two tables; one for the regression and another for the classification.

   b. The list of the commands that you have used to deploy your source code to the GitLab repository.

   c. The list of the commands that you have used to create and to bush the Docker image to the Docker Hub.

   d. The Link of the source code you have deployed on the GitLab.

   e. The Link of the Docker image you have deployed on the Docker Hub.