CMPT459 Spring 2021
Data Mining
Martin Ester
TAs: Madana Krishnan Vadakandara Krishnan and Arash Khoeini

Milestone 1 of the Course Project

Deadline: February 22nd

Total marks: 100

Introduction

In our course project, you will work with publicly available COVID-19 datasets.

- Case dataset for training (*cases_train.csv*)
  This contains the training data for individual cases, i.e. cases that tested positive for COVID-19.
- Case dataset for testing (*cases_test.csv*)
  This file contains the testing data for individual cases (similar to training data), but without the 'outcome' label.
- Location dataset (*location.csv*)
  This file contains the number of cases for each location.

These files were obtained from the following open-source repositories, which provide more information:

- https://github.com/beoutbreakprepared/nCoV2019
- https://github.com/CSSEGISandData/COVID-19

The datasets have been filtered by the TAs for use in our project. To ensure consistency, the datasets have been frozen on September 20th, 2020. You can download the datasets from:

https://github.com/MadanKrishnan97/CMPT459CourseProjectSpring2021/tree/main/dataset

The data mining task will be to predict the outcome of a case. In this first milestone of the course project, you will do the data preprocessing as specified further in the document. In the next milestones, you will be building models and using this model to predict the outcomes on *cases_test.csv* dataset. The performance of these models will be evaluated based on the predictions on the test set, and hence it is important to treat *cases_test.csv* and *cases_train.csv* similarly.

Preprocessing Tasks

Note that the following tasks should be performed on the following datasets:

- Tasks 1.1 and 1.3 on *cases_train.csv*,
- Tasks 1.2 and 1.5 on *cases_train.csv* and *cases_test.csv*, and
- Tasks 1.1, 1.4 and 1.5 on *location.csv*.

1.1 Exploratory Data Analysis **(20 marks)**

Perform exploratory data analysis to get an understanding of the datasets. Show visualizations and statistics for all attributes of both datasets for which that makes sense. It does not make sense, e.g., for textual attributes such as the description. For some attributes, e.g. longitude and latitude, it may make sense to visualize the combination of two attributes. Finally, for every attribute print the number of missing values.

1.2 Data cleaning and Imputing missing values **(10 + 15 marks)**

Perform data cleaning steps, mainly on the age column. Reduce different formats (ex. 20-29, 25-, 13 months), to a standard format (ex. 25)

For all attributes with missing values, discuss why and how (if applicable) you impute missing values. Apply your imputation strategy to your datasets.

1.3 Dealing with outliers **(20 marks)**

Which attributes have outliers? How do you deal with them? Apply your strategy of dealing with outliers to your datasets.

1.4 Transformation **(10 marks)**

In the location dataset, transform the information for cases from the US from the country level, used in the location dataset, to the state level, used in the cases dataset. Explain your method of transformation, and why you use a particular type of aggregation on any column.

1.5 Joining the cases and location dataset **(20 marks)**

The two datasets can be joined using some shared features. You can use either 'province, country' or 'latitude, longitude'. Present your strategy for joining the datasets and motivate your design decisions. Apply your join strategy to create a dataset of cases with additional features inherited from their locations.

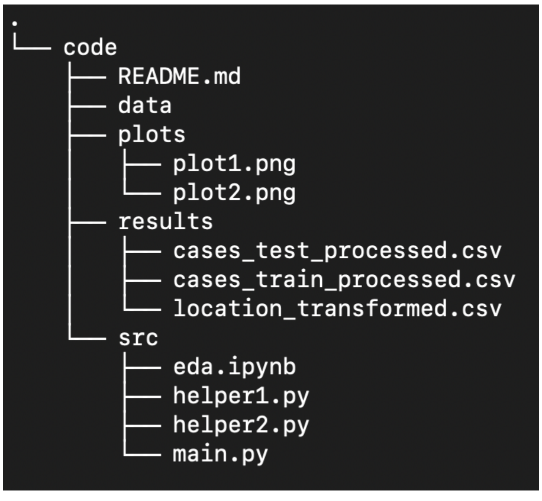
1.6 Outcome labels **(5 marks)**

What are the different ‘outcome’ labels in the *cases_train.csv* dataset? Based on your understanding what do these labels mean? What type of data mining task is the prediction of the outcome labels?

Submission (Code + Report)

2.1 Code

Submit a ‘code.zip’ file with the following contents. It should contain the codes, results and figures obtained. The structure should look like this.



```
.
└── code
    ├── README.md
    ├── data
    ├── plots
    │   ├── plot1.png
    │   └── plot2.png
    ├── results
    │   ├── cases_test_processed.csv
    │   ├── cases_train_processed.csv
    │   └── location_transformed.csv
    └── src
        ├── eda.ipynb
        ├── helper1.py
        ├── helper2.py
        └── main.py
```

- *data/* folder structure for the original dataset (location for the three input csv files)
- *plots/* for the figures obtained
- *result/* for the results obtained after preprocessing
- *src/* for codes and scripts
    - *src/eda.ipynb* for task 1.1 (EDA)
    - *src/main.py* for tasks 1.2, 1.3, 1.4 and 1.5
    - *src/helper-n.py* for any helper functions that you want to use in main.py
- *README.md* for any special instructions for code execution (ex: if third party APIs used, high running time etc.)

Running '*src/eda.ipynb --> Cell --> Run All*' and '*python src/main.py*' should reproduce all the reported results with the same directory structure as described above. You are free to use (rather encouraged) to split codes into helper functions and use them in main.py. Ensure to use relative paths to access files and folders within the code.

2.2 Report

Briefly explain the approaches and steps followed in the five preprocessing tasks and answer the questions asked in section 1.6. The report should **NOT** be more than 2 pages. Submit a 'report.pdf' file.

NOTE:

- '*cases_test_processed.csv*' and '*cases_train_processed.csv*' will contain the cleaned, joined result obtained from 1.5. '*location_transformed.csv*' will contain the transformed location dataset.
- Do NOT include the three input csv files in the submission. Retain only the '*data/*' folder structure as shown in the above image.