MIE 1624 Introduction to Data Science and Analytics – Winter 2020

Assignment 1

Due Date: 11:59pm, February 23, 2020

Submit via Quercus

Background:

Kaggle has hosted an open data scientist competition in 2019 titled "**2019 Kaggle ML & DS Survey Challenge.**" The purpose of this challenge was to "*tell a data story about a subset of the data science community represented in this survey, through a combination of both narrative text and data exploration.*" More information on the competition, data, and prizes can be found on: https://www.kaggle.com/c/kaggle-survey-2019

The original dataset (**multiple_choice_responses.csv**) contains the survey results provided by Kaggle. The survey results from 19717 participants are shown in 246 columns, representing survey questions. Not all questions are answered by each participant, and responses contain various data types.

In the original dataset, column Q10 "*What is your current yearly compensation (approximate $USD)?"* contains the ordinary categorical target variable. The original data (**multiple_choice_responses.csv**) has been transformed to **Kaggle_Salary.csv** as per the code given in **KaggleSalary_DataSet.ipynb.** In the dataset to be used for Assignment 1 (**Kaggle_Salary.csv- File to be read in notebook for this Assignment**), rows with the null values of salaries have been dropped. In addition, two columns ('Q10_Encoded' and 'Q10_buckets') has been added at the end. Column 'Q10_buckets'(**Target Variable for Assignment 1**) has been obtained by combining some salary buckets in the column 'Q10'. Column 'Q10_Encoded' has been obtained by label encoding the column 'Q10_buckets'.

The purpose of this assignment is to
1) understand and explore employment in the data science community, as represented in a survey conducted by Kaggle.
2) train, validate, and tune multi-class ordinary classification problem that can classify, given a set of survey responses by a data scientist, what a survey respondent's current yearly compensation bucket is.

Classification is a supervised machine learning approach used to assign a discrete value of one variable when given the values of others. Many types of machine learning models can be used for training classification problems, such as logistic regression, decision trees, kNN, SVM, random forest, gradient-boosted decision trees and neural networks. In this assignment you are **required to use logistic regression algorithm**, but feel free to experiment with other algorithms.

For the purposes of this assignment, any subset of data can be used for data exploration and for classification purposes. For example, you may focus only on one country, exclude features, or engineer new features. If a subset of data is chosen, **it must contain at least 5000 training points.** You must **justify and explain** why you are selecting a subset of the data, and how it may affect the model.

Data is often split into training and testing data. The training data is typically further divided to create validation sets, either by just splitting, if enough data exists, or by using **cross-validation** within the training

set. The model can be iteratively improved by tuning the hyperparameters of the model or by feature selection.

Submission:

1) Produce a report in the form of an IPython Notebook detailing the analysis you performed to determine the best classifier (ordinary multi-class classification model) for the given data set. Your analysis must include the following steps: data cleaning, exploratory data analysis, feature selection (or model preparation), model implementation, model validation, model tuning, and discussion. When writing the report, make sure to explain for each step, what it is doing, why it is important, and the pros and cons of that approach.

2) Create 5 slides in PowerPoint and PDF describing the findings from exploratory analysis, model feature importance, model results and visualizations.

Learning objectives:

1. Understand how to clean and prepare data for machine learning, including working with multiple data types, incomplete data, and categorical data. Perform data standardization/normalization, if necessary, prior to modeling.
2. Understand how to explore data to look for correlations between the features and the target variable.
3. Understand how to apply machine learning algorithms (logistic regression) to the task of classification.
4. Improve on skills and competencies required to compare performance of classification algorithm, including application of performance measurements, statistical hypothesis testing, and visualization of comparisons.
5. Understand how to improve the performance of your model.
6. Improve on skill and competencies required to collate and present domain specific, evidence-based insights.

To do:

The following sections should be included but the order does not need to be followed. The discussion for each section is included in that section's marks.

1. **Data cleaning (20 marks):**

   While the data is made ready for analysis, several values are missing, and some features are categorical. Note that some values that appear "null" indicate that a survey respondent did not select that given option from a multiple-choice list. For example – "*Who/what are your favorite media sources that report on data science topics? (Select all that apply) - Selected Choice - Twitter (data science influencers)*"

   For the data cleaning step, handle missing values however you see fit and justify your approach. Provide some insight on why you think the values are missing and how your approach might impact the overall analysis. Suggestions include filling the missing values with a certain value (e.g. mode for categorical data) and completely removing the features with missing values. Secondly, convert

categorical data into numerical data by encoding and explain why you used this particular encoding method.

These tasks can be done interchangeably, e.g., encoding can be done first.

2. **Exploratory data analysis (15 marks):**

   a. Present 3 graphical figures that represent trends in the data. How could these trends be used to help with the task of predicting yearly compensation or understanding the data? All graphs should be ***readable*** and presented in the notebook. All axes must be ***appropriately labelled***.
   b. Visualize the order of feature importance. Some possible methods include correlation plot, or a similar method. Given the data, which of the original attributes in the data are most related to a survey respondent's yearly compensation?

   The steps specified before are not in a set order.

3. **Feature selection (10 marks):**

   Explain how feature engineering is a useful tool in machine learning. Then select the features to be used for analysis either manually or through some feature selection algorithm (e.g. regularized regression).

   Not all features need to be used; features can be removed or added as desired. If the resulting number of features is very high, dimensionality reduction can also be used (e.g. PCA). Use at least one feature selection technique – describe the technique and ***provide justification*** on why you selected that set of features.

4. **Model implementation (25 marks):**

   Implement **logistic regression** algorithm on the training data using 10-fold cross-validation. How does your model accuracy compare across the folds? What is average and variance of accuracy for folds? Treating each value of hyperparameter(s) as a new model, which model performed best? Give the reason based on bias-variance trade-off. An output of your algorithm should be a probability of belonging to each of the salary buckets. Apply scaling/normalization of features, if necessary.

5. **Model tuning (20 marks):**

   Improve the performance of the models from the previous step with hyperparameter tuning and select a final optimal model using grid search based on a metric (or metrics) that you choose. Choosing an optimal model for a given task (comparing multiple classifiers on a specific domain) requires selecting performance measures, for example accuracy, precision, recall and/or F1-score to compare the model performance.

   There is no minimum model accuracy, as long as your methodology is reasonable and well explained.

6. **Testing & Discussion (10 marks):**

   Use your optimal model to make classifications on the test set. How does your model perform on the test set vs. the training set? The overall fit of the model, how to increase the accuracy (test, training)? Is it overfitting or underfitting? Why? Plot the distribution.

   Insufficient discussion will lead to the deduction on marks.

Tools:

- **Software:**
    - **Python Version 3.X** is required for this assignment. our code should run on the CognitiveClass Virtual Lab http://labs.cognitiveclass.ai (Kernel 3). All libraries are allowed but here is a list of the major libraries you might consider: Numpy, Scipy, Sklearn, Matplotlib, Pandas.
    - No other tool or software besides Python and its component libraries can be used to touch the data files. For instance, using Microsoft Excel to clean the data is not allowed.
    - Read the required data file from the same directory as your notebook on the CognitiveClass Virtual Lab – for example pd.read_csv("Kaggle_Salary.csv").
- **Required data files:**
    - **Kaggle_Salary.csv**: survey responses with yearly compensation.
    - The data file cannot be altered by any means. The Jupyter notebook will be run using local version of this data file. Do not save anything to file within the notebook and read it back.

What to submit:

1. Submit via Quercus a Jupyter (IPython) notebook containing your implementation and motivation for all the steps of the analysis with the following naming convention:

   lastname_studentnumber_assignment1.ipynb

   Make sure that you **comment** your code appropriately and describe each step in sufficient detail. Respect the above convention when naming your file, making sure that all letters are lowercase and underscores are used as shown. **A program that cannot be evaluated because it varies from specifications will receive zero marks.**

2. Submit 5 slides in PowerPoint and PDF describing the findings from exploratory analysis, model feature importance, model results and visualizations. Use the following naming conventions **lastname_studentnumber_assignment1.pptx** and **lastname_studentnumber_assignment1.pdf**

Late submissions will receive a standard penalty:

- up to one hour late - no penalty
- one day late - 15% penalty
- two days late - 30% penalty
- three days late - 45% penalty
- more than three days late - 0 mark

Other requirements:

1. A large portion of marks are allocated to analysis and justification. Full marks will not be given for code alone.

2. Output must be shown and readable in the notebook. The only files that can be read into the notebook are the files posted in the assignment without modification. All work must be done within the notebook.

3. The notebook should be presentable, do not show large amounts of raw output.

4. Ensure the code runs in full before submitting. Open the code in CognitiveClass Virtual Lab (Kernel 3) and navigate to Kernel -> Restart Kernel and Run all Cells. Ensure that there are no errors.

5. Do not re-run cross-validation (it can run for a very long time). When cross-validation is finished, output (print) the results (optimal model parameters). Hard-code the results in the model parameters and comment out the cross-validation code used to generate the optimal parameters.

Tips:

1. You have a lot of freedom with however you want to approach each step and with whatever library or function you want to use. As open-ended as the problem seems, the emphasis of the assignment is for you to be able to explain the reasoning behind every step.

2. While some suggestions have been made in certain steps to give you some direction, you are not required to follow them. Doing them, however, guarantees full marks if implemented and explained correctly.

3. The output of the classifier when evaluated on the training set must be the same as the output of the classifier when evaluated on the testing set, but you may clean and prepare the data as you see fit for the training set and the testing set.

4. When evaluating the performance of your algorithm, keep in mind that there can be an inherent trade-off between the results on various performance measures.