Linear Regression Analysis

A. Nagar

Due Date Mentioned in eLearning

Instructions

- This assignment requires you to build linear regression model in Python using standard machine learning libraries.
- You should store your dataset under your account in the UTD server or any other public location, such as Google drive. Do not submit the dataset (which could be quite large) on eLearning,
- You are allowed to work in teams of maximum two students. Please write the names and NetIDs of each group member on the cover page.
  Only 1 final submission per team.
- **You have a total of 4 free late days for the entire semester. You can use at most 2 days for any one assignment. After four days have been used up, there will be a penalty of 10% for each late day. The submission for this assignment will be closed 2 days after the due date.**
- Please ask all questions on Piazza, not via email.

1 Project Selection

One of the most popular applications of linear regression is in the area of house price prediction. For this assignment, you can choose any one of the datasets below and perform regression analysis. Include a report of your results, with plots and summary data. Also, indicate what interesting information did you obtain

- California Housing Dataset
  https://www.kaggle.com/camnugent/california-housing-prices
- Advanced Regression Techniques competition
  https://www.kaggle.com/c/house-prices-advanced-regression-techniques/overview
- Zillow Home Price Dataset
  https://www.kaggle.com/c/zillow-prize-1
- Russian Housing Dataset
  https://www.kaggle.com/c/sberbank-russian-housing-market
- Melbourne Housing Dataset
  https://www.kaggle.com/anthonypino/melbourne-housing-market
- Ames Housing Dataset
  https://www.kaggle.com/c/ames-housing-data

2 Model Constructions and Results Analysis

In this section, you will perform data pre-processing, loading, model creation and results analysis. You are free to use any data science library, such as Scikit-learn, statsmodels, Numpy or Pandas. The following could be some of the suggested steps:

- Loading the data into Pandas DataFrame object. Remember to use public URLs to read the file.
- Examining data for consistency: Check for null values, missing data, and any data inconsistency and handle them before proceeding forward.
- Examining attributes and target variable(s): Be sure you clearly understand each of the attributes and the target variable. Examine the various attributes and convert any categorical ones to numerical ones, if needed. Obtain and output summary of the attributes. Are the attributes normally distributed? If not, what could be the reason?
- Standardize and normalize the attributes.
- Find how the attributes are correlated to each other and the target variable. Perform numerical and visual analysis and output plots and results.
- Identify a few important attributes and proceed forward. Do not use all attributes blindly.

- Split the data into training and testing parts. The ratio is up to you.
- Construct a model and tune the various parameters such as learning rate, maximum iterations, etc. Make sure to keep track of the parameters used and results obtained. Do not just use all default values.
- Apply the model on the training and test datasets and report diagnostic parameters such as R-squared and adjusted R-squared, Mean Squared Error, variable weights and significances (t-values, p-values, importance, etc), F-statistic, etc.
- Try different sets of attributes, perhaps by adding more to the initial list. What is your interpretation of the results? Which are the significant variables? What are the best results that you obtained?

You should create a report that includes results and your interpretation for each of the above steps. You are free to add any additional detail. Visual plots are preferred in all cases followed by your interpretation. Remember not to copy definitions or long explanations from external sources, but to write *your* analysis and interpretation of the results. **Please do not include code or code snippets in your report.**

3 Requirements

The following are parameters that **cannot be changed**

1. You are allowed to work in teams of maximum size 2
2. Treat this as a data science project. You have to interpret the output diagnostics. Also, try to include as many plots as you can. As stated previously, your interpretation and analysis of results is what we want to see.
3. You cannot copy any publically available solutions. There will be penalty for plagiarism.
4. Submit your Python code file and report file. Please do not hard code any paths in your code. You can put the data in your UTD web account and read from that link.
5. Python code can be on Google Colab or Jupyter Notebook.

You need to tune as many of the parameters as possible. The list will not be mentioned here, but you can see them in the documentation and sample code. You have to keep a log of your experiments with the parameters used and results obtained.

If you have made any assumptions, please state them completely. Also include instructions on how to compile and run your code in a README file.