CS 552 – Data Science with Python Assignment-1

Car Pricing Prediction by Linear Regression

Due: 12 December 2021– 22:00

1. Definition

In this assignment, you will implement a **Jupyter notebook** that solves one of the most well-known regression problems, which is Car Pricing Prediction by some features (e.g. the age of car, selling price, fuel type etc.). You need to implement the notebook solving car pricing prediction by **linear regression**. We expect you to process the data, and train **different linear regression models** that predict the price of a car by excluding some certain features, and lastly report the performance of each model **qualitatively** and **quantitatively**. The data contains 301 unique entries with 9 different features without any partition, so you are expected to split the data as training and test set.

In this assignment, you are only allowed to use the scientific computation libraries, which are introduced in the lectures (NumPy [docs], Pandas [docs], Scikit-learn [docs]), and also you are free to use any visualization library (e.g. Matplotlib [docs], Seaborn [docs]). However, we kindly expect you **to conduct a comprehensive experimental setup and to visualize the data and the results in your notebooks**, where each cell introduces one thing and has a comment for what it does. The details of the task are in the following sections.

In the report, you are expected to introduce the project and the aim of this project, to explain the algorithms/models employed in this project in detail. Also, it is expected to show the experimental setup (i.e., each parameter in your model or each feature in the data that you changed for training), and to report the training performance in all experimental settings and the test performance of tuned settings for each algorithm/model. Moreover, descriptive tables, plots and figures are required **both to observe the data better and to show the results for all settings**. Please do **NOT** forget to add visualization for the data and the results in your notebooks.

We strongly recommend you reading about the algorithms and the problem before starting to implement your assignment.

2. Implementation Details

For this assignment, you may not use more than one notebook, which means **you need to have a single "*.ipynb" file that is responsible for all tasks**. For each cell, a descriptive comment in the first line is **must**. Please use **the newest** version of the libraries.

The detail of this task as follows:

- Giving information about the linear regression (report)
- Giving information about the dataset (e.g., what are the features? how many rows/columns? Numeric/categoric features? Any missing information in rows? The statistical information about the dataset and **more**)
- Fetching the data
- Checking out the data (e.g., head, description, info, # of missing rows, correlation)
- Visualizing the data for each dependent feature with independent feature (e.g., Owner Type vs. Selling Price, Transmission vs. Selling Price)
- **Feature engineering (e.g., only categorical features to one-hot dummy (*pandas.get_dummies()*), drop rows with missing info, scale the data to the range of 0 to 1, extracting and visualizing the correlation between features, picking different feature sets including at least 4 different features in 3 different settings (e.g. F1=[f1,f2,f4,f5,f6], F2=[f2,f4,f5,f6], F3=[f1,f2,f2,f6]), split the data for training and test set and more)** (notebook) (hints: choose dependent variable carefully, eliminate unnecessary columns, convert year to age, remove unit names from the values, do not forget convert categorical features to one-hot format, check missing values)
- Training LR models with at **least two different regression algorithms** (e.g., LinearRegression, RandomForestRegressor, RANSACRegressor) for **3 different feature setups** (e.g., F1, F2, F3)
- Evaluate the models with different metrics (mean-squared error, mean-absolute error, root-mean-squared error, r2 score)
- Visualize the results for 3 different setups for each metrics
- Printing the general formula of the model with best performing setup
- Discussing the model coefficients for 3 different setups (report)
- Discussing the extracted information which may be useful for, for example, a car seller company, that's why we try to use ML or predictive analysis on such a case. (**MOST IMPORTANT PART** in report) (for example, present price of a car directly influences selling price prediction, since as can be seen in Figure X, they are highly correlated. The age of a car negatively affects the selling price as shown in Figure X. etc.)
- Please do **NOT** include any other 3rd-party library to your notebook (except NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn), because you do not need them. Including such libraries may cause a problem for running your code in different local environments (e.g., dependencies).
- Linear regression and car pricing prediction problem are well-studied in the literature and on online resources, and thus there are many different coding examples online. We have almost all of them, at least we have a chance to compare your notebooks with our collected online resource database for linear regression problem. Please do **NOT** try to use them directly. In any circumstances of copying online resources directly, **you will get 0 points**.
- Please follow the technical report writing rules (i.e., *Introduction, Methodology, Implementation Details, Results, Conclusion*). It is **highly** recommended to add the visual contents that you extract in your notebooks to your reports.

The notebook file (*.IPYNB) and the report file (*.PDF) should be zipped (.ZIP) together. The filename of final submission file should be in the format of "**NAME_SURNAME_ID_hw1.zip**". Please follow this structure for your submission. You should **NOT** include the data files (.CSV) which is provided by us to your submission.

You may discuss the algorithms and so forth with your friends, but this is an individual work. Therefore, you have to submit your original work. **In any circumstances of plagiarism, first, you will fail the course, then the necessary actions will be taken immediately.**