Software Design for Intelligent Systems Project 1

Serena Shah | ss94574 | serena.shah@utexas.edu

The dataset analyzed was one that gave technical information about almost 400 automobile vehicles and this project aimed at using linear regression to predict fuel efficiency of the vehicles given this technical information. In its raw form, the dataset was provided in `.csv` file format. The data was read into a `pandas` dataframe object, which associated a column with each variable type and contained all the recorded values for that variable. To make the data workable for analysis, it was important to address missing values in columns, data types of variables, and consistency in size across the dataset.

First, the shape of the data was identified as 398 by 9, indicating 398 values for 9 recorded variables. These variables are `mpg`, `cylinders`, `displacement`, `horsepower`, `weight`, `acceleration`, `model_year`, `origin`, and `car_name`. Looking at the data type of `horsepower` gave insight into potential data handling difficulties– although horsepower is a numerical value (and therefore should have a type `float64`), its data type was listed as an `object`. Through error outputs when trying to convert the data type, it was revealed that there were 6 values that were '`?`' strings. These were replaced with `NaN` values, then the column was converted to `float` type, and finally the `NaN` values were replaced with the median value for horsepower in the dataset.

The `origin` variable was dropped as its values were not interpretable for analysis. No variables had categorical values and so one-hot encoding was not needed.

To begin preparation for the linear regression fitting, the `car_name` variable had to be dropped from the dependent variables' data frame due to its `string` variables, which are incompatible with the numeric process of linear regression. The dataset was then split into two sets: one training set and one testing set. These were split into a 1:1 ratio to maintain the same shape for both sets. To train the linear regression model, the `scikit-learn` library's linear model was fitted using the training data, with all numeric variables other than `mpg` being the dependent *x* and the `mpg` values being the independent *y*.

The `predict` functionality of the `scikit-learn` library output its predicted `mpg` values given the other variables' test values. The magnitude difference between the real and estimated values was on average 2.65 mpg. The score given on the regression model for testing data was 0.8152, relatively close to an absolutely accurate score of 1.0. The model, therefore, does a good job of predicting fuel efficiency. I am confident that the model can generally predict the fuel efficiency of a car given the numeric variables listed above within an approximately 20% accuracy, though feeding more robust data to the regression model would likely improve its predictive abilities.