Nadiem Ahmed October 11, 2019

Project Deliverable 2

1. **Problem Statement:**

The goal of this project is to accurately predict the time it takes for Mercedes to test a car given a set of custom features that the car will have.

There are approximately 380 variables, most of which are binary. It will likely be necessary to employ feature selection and reduce the dimensionality of the data. Before doing so, I will conduct some data exploration to better understand the nature of the data set.
PCA - mainly for continuous
Variance threshold

2. **Data Preprocessing:**

The data set has not changed. There are approximately 380 anonymized features. 8 of the features are categorical, while the rest are binary. Since there are only around 4200 samples the set suffers from the curse of dimensionality, which means that we do not have enough samples to accurately describe the features. Therefore, It will likely be necessary to employ feature selection and reduce the dimensionality of the data. Before doing so, I will conduct some data exploration to better understand the nature of the data set.

To begin my data exploration, I checked the type of data I had, which I mentioned already. I then took a look at my target variable ("y), which is the seconds it takes for a Mercedes car to pass testing. I looked at the target variable's summary statistics, and used both a scatter and bar plot to visualize its' distribution. I then checked if I had any missing data. I then took a look at the distribution of my 8 categorical features using bar and violin plots. Next, I attempted to visualize the correlation of my features using a heatmap, but given the large numbers of features, I did not really gain much insight from it. I also removed an outlier and columns that had only one unique value from my data. Last but not least, I used one-hot encoding to convert my categorical features into binary features that I could use in my model. This increased the number of features in the data significantly.

3. **Machine Learning Model**

To see whether or not this dataset does indeed suffer from the curse of dimensionality, I think it's appropriate to initially keep the majority of the features and create an initial model, before reducing the number of features to see if it improves results. Therefore, I have thus far avoided using any dimensionality reduction models (PCA, etc.)

As mentioned previously, since we have a continuous target variable, we must use a regression algorithm. Given the large number of features I have, I opted to initially use a Lasso regression. Lasso regression is a supervised machine learning algorithm that uses shrinkage, or eliminating certain variables/features, to conduct a regression. Given that we have a large number of

variables, this is quite helpful. In Lasso regression, the coefficients (weights) of variables that are deemed unimportant are reduced to 0, and they are eliminated. This is controlled by the lambda parameter; a higher lambda value will lead to more variables being eliminated.

4. Preliminary Results

Given that we are solving a regression problem, the performance of our model should be measured using the mean squared error. However, the Kaggle competition is scored using the R^2 value so we will use that. So far I have used the Lasso regression model with numerous different hyperparameters. As you decrease the “alpha” hyperparameter, the R^2 value increases, but if you decrease it too much the model becomes inaccurate. So far, the highest R^2 value I have is 0.54567, with “alpha=0.025”.

5. Next Steps

I am approaching my model selection from an experimental perspective. I want to try numerous models and see if they improve performance, so I will definitely be trying to use different regression techniques as we move forward. I will also be concurrently reducing the dimensionality of the data. There are numerous techniques for this as well, which I will also experiment with.