STA 160 Midterm Project Report

By John Tran 916232852

Overview: In this report, I analyzed two datasets from the UCI Machine Learning Repository website. The first one is seeds data that had 8 columns and 210 rows of data and the other dataset was automobile data with 26 columns and 205 rows of data. To simplify this report, these datasets are independent from each other and I will not be comparing these datasets through analysis and similarities. Hence, this report is basically two smaller reports in one. The first half will consist of an analysis of the seeds data and the second half will be the analysis of the automobile data.

Part 1: Seeds data

Introduction:

This data has been collected in the wild and donated in 2012. The dataset is about examined kernels belonging to three different varieties of wheat: Kama, Rosa and Canadian. These were randomly selected for experiment. The data collected attributed to seven geometric parameters of wheat kernels were measured: area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficients, and length of kernel groove. In this instance, the response variable is the labels of the seed type. Since the data is categorical and qualitative, the only reasonable way to utilized this is classification. Predicting a qualitative response for an observation can be referred to as *classifying* that observation. [1] A popular classification technique that I'll be using is **logistic regression** to predict a qualitative response.

Background:

Logistic Regression is very similar in structure to linear regression. It models the probability of the response variable or Y that belongs to a category. By using the logistic function, we can use manipulation to arrive at the logistic regression model.

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}. \quad \rightarrow \quad \log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X.$$

This particular model is for binary labels. With this seeds dataset, we have to use multinomial logistic regression. Multinomial logistic regression is simply the extended version of logistic regression. This classifier is for more than 2 labels.

$$\Pr(Y_i = 1) = \frac{e^{\boldsymbol{\beta}_1 \cdot \mathbf{X}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}$$

$$\Pr(Y_i = 2) = \frac{e^{\boldsymbol{\beta}_2 \cdot \mathbf{X}_i}}{1 + \sum_{k=1}^{K-1} e^{\boldsymbol{\beta}_k \cdot \mathbf{X}_i}}$$

.....

Summary and Analysis:

Let's plot the data first against each feature to see if there is anything reasonable.

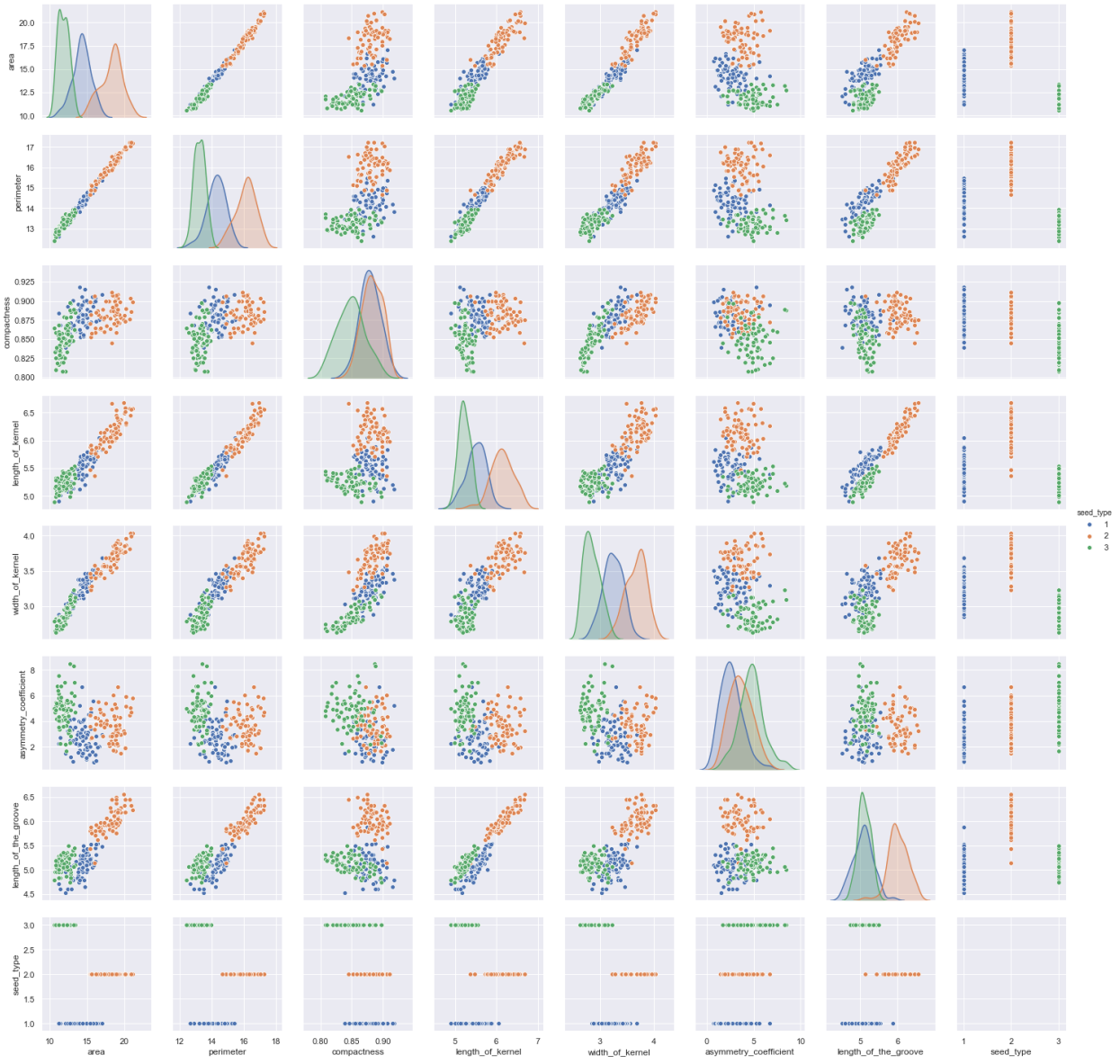

Figure 1: Pairplot of the Seeds Dataset.

As we can see the data seems predetermined and classified already. Let's try to train a model that correctly classifies this data. In this case, I used random sampling to train and split my data, and I also used stratified sampling to even distribute my data. Stratified sampling allows for more precision with each group. For my random sampling data, I split to a 75/25 split. This is often a common range for random sampling the data. Let's compare the proportion in the test data below.

Table 1: Proportions of the Stratified Sampling Data vs the Random Sampling

Stratified	Random
.35668	.333
.3312	.333
.3121	.333

We fit the full model for the multinomial logistic regression and computed the confusion matrix. The confusion matrix is a matrix that consist true positives, false positives, true negatives, and false negatives. It is a visual to greatly represent whether your model was precise or not. In this case, since we have two different sampling data, we can compare stratified and random sampling together to see which one is actually more accurate.

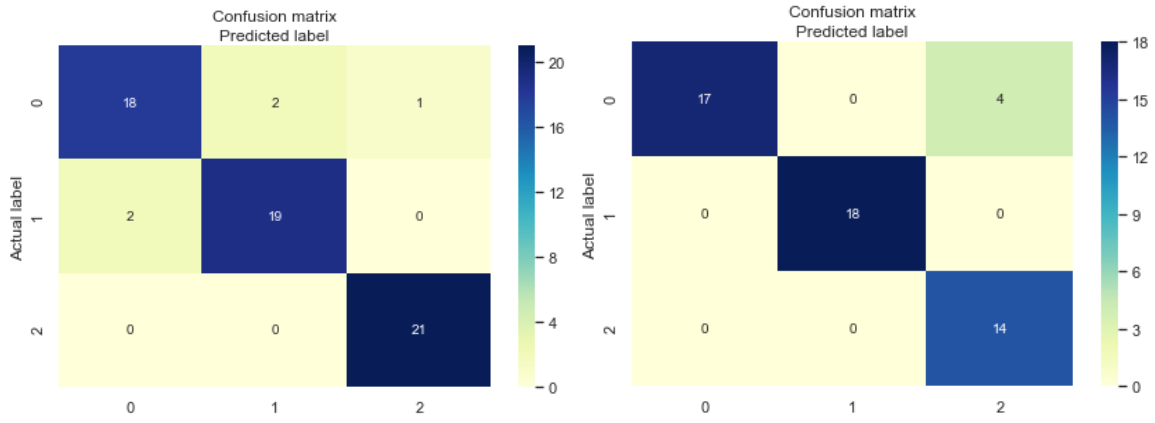

Figure 2: Stratified on the Right, Random Sampling on the Left

As we can see in each class, the stratified misclassified a few seeds in different labels, while for the random sampling it seems to be accurate for the first two labels, but the last one it seems to be off more than the stratified data. To dive deeper into this matrix, we can use classification reports to show more accurate results.

Table 2: Classification Reports

	Precision	**Accuracy**	**Precision**	**Accuracy**
1	**.90**	**.92063**	**1.00**	**.9403**
2	**.90**		**1.00**	
3	**.95**		**.78**	

In table 2, it shows the stratified data had more consistent precision than the random sampling data when running this model. However, the accuracy of the random sampling data proves that this model predicts better on it.

Along with the confusion, we decided to plot a ROC curve. The Receiver operating characteristic curve is a plot that shows the sensitivity and the specificity of the predictions for all the possible cutoffs probabilities. This curve can be more informative than a classification table, because it summarizes predictive power. The ROC curve plots sensitivity on the vertical axis and the 1- specificity on the horizontal axis. [2] Since our data has multiple labels, the ROC curve will have multiple ROC curves.

Figure 3: Multilabel ROC Curves

In **Figure 3,** the ROC curves show that almost all the classes have high predictive power. It tells us that the model is powerful and predicts relatively accurate. Class 2 seems to have the highest predictive power out of the 3 classes. However, this plot may have been incorrectly coded so these curves may be invalid.

Next let us see how well this logistic regression model draws decision boundaries. Decision boundaries are important when it comes to classification problems like this one. They are a boundary or a surface that partitions into regions of the problem space. Basically, it shows the separation between the classes. However, they may not accurately separate everything. We decided to impose decision boundaries on several variables to see whether or not there is a define line or definite boundary between the classes.

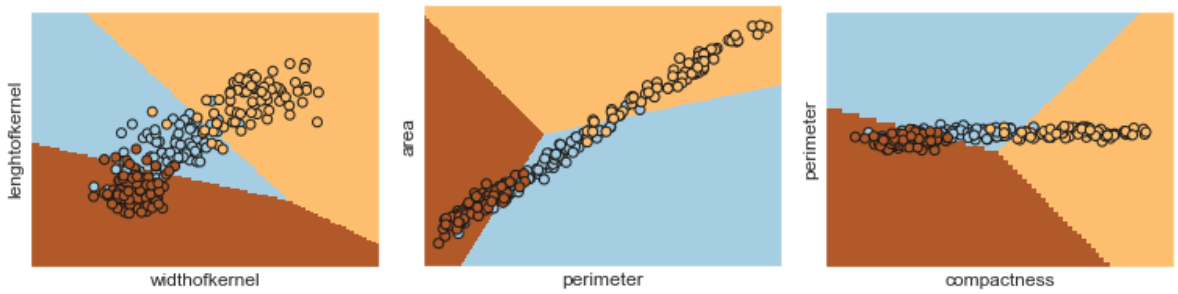

Figure 4: Decision boundaries on certain features

Based on **Figure 4,** the decision boundaries seem pretty close to the actual labels. With supporting evidence from the confusion matrix and the ROC curve. It seems that our model predicts these seeds pretty well.

Conclusion and Discussion:

With multinomial logistic regression, we were able to predict the classes of the seeds pretty well. With accuracy score of 94% this model indeed performed rather well. However, this data seemed pretty clean in the first place. If we took out the labels and try to train a model without it, it would have been a much more interesting case. Also, they are many other ways to classify this dataset such as linear discriminant analysis and KNN classifier. Besides those are the other two popular classifying techniques; we could also use random forest, decision trees, GAMs, and SVMs. We could as sample our data differently like cross validation or bootstrap. Overall, it would be nice to compare which classifier is the best out of all of these classification methods, but it just may be dependent on the data as always.

Part 2: Automobile Data

Introduction:

In this dataset, the data is vehicles from 1985 with many characteristic and features. These cars were model import cars and trucks with their specifications. This dataset have categorical and numerical variables, and it includes several NA's across the features. The dimensions of the data set are 2-5 rows by 26 columns. Each column represents an attribute of the vehicle, ranging from the model to the price of the vehicle. Our approach to this dataset was much different from the seeds data; we will be using linear regression and exploratory data analysis. With linear regression, we will be predicting certain variables on certain features and try to find the best model with certain techniques. By using exploratory data analysis, we can visualize certain categorical variables and numerical variables to simply see if there is a correlation between certain features that can help us with our linear regression model.

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^{\mathsf{T}} \boldsymbol{\beta} + \varepsilon_i,$$

According to linear regression, there are 4 assumptions as follow: linearity, independence, homoscedasticity, and equal variance. It will difficult to satisfy all these functions due to the categorical variables, but will try in the Summary and Analysis portion.

Cleaning the Data:

As we know our data is not exactly perfect, it is a mixed of everything: NAs, factors, and integers. So our approach to get rid of NA's, were to implement the mean of the corresponding column of that NA. This can caused our data to be skewed in a certain way. However, the primary reason for this is due to the fact the NA's occupy around 40 rows of our data. If we removed the NA's, the data will lose a lot of significance from certain features.

Summary and Analysis:

By using exploratory data analysis, we decided to see how many car brands our within our dataset. There are 23 car bands and 203 vehicles (after cleaning the data) in this dataset. Let's see which car brand is most pronounced within the data.

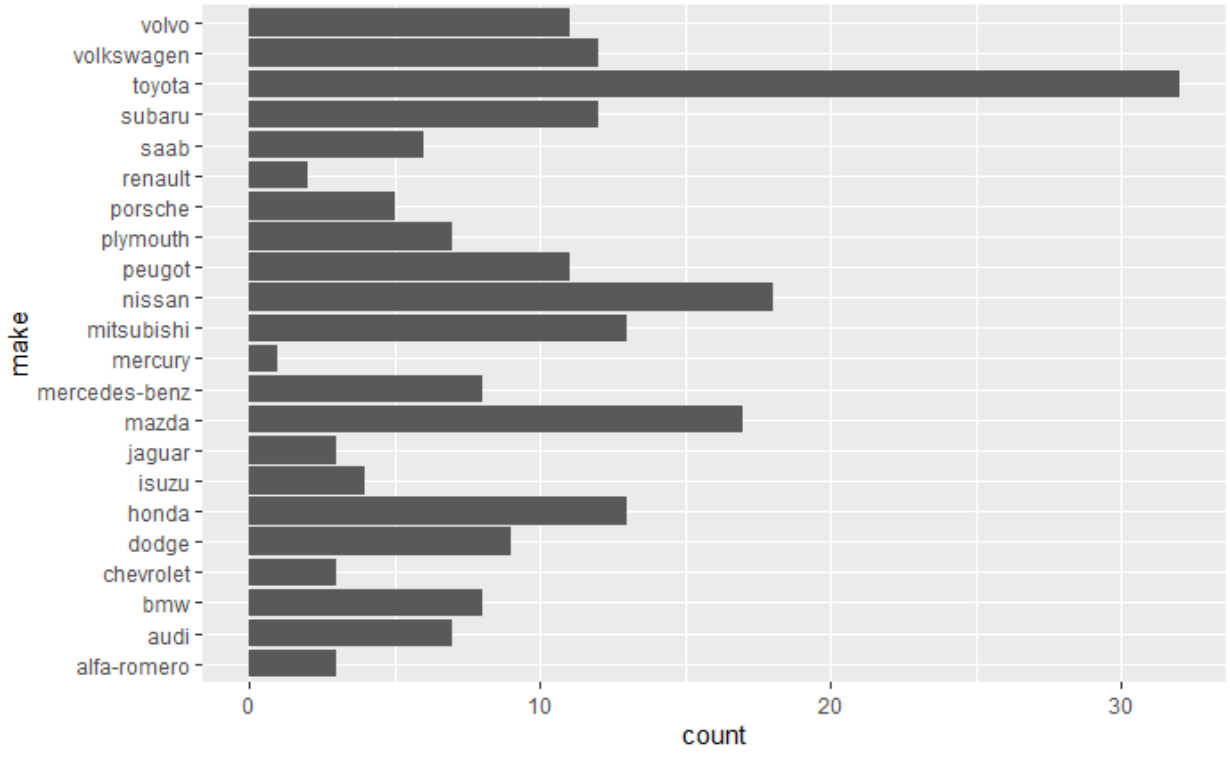

Figure 5: Counts of the car brands

This bar plot shows that Toyota has been collected the most in this 1985 dataset. While coming next are Mazda and Nissan. Japanese brands make over 25% percent of this dataset. Now since we know how many brand cars in this dataset, we can use this information to see which car brand is the most expensive in price.

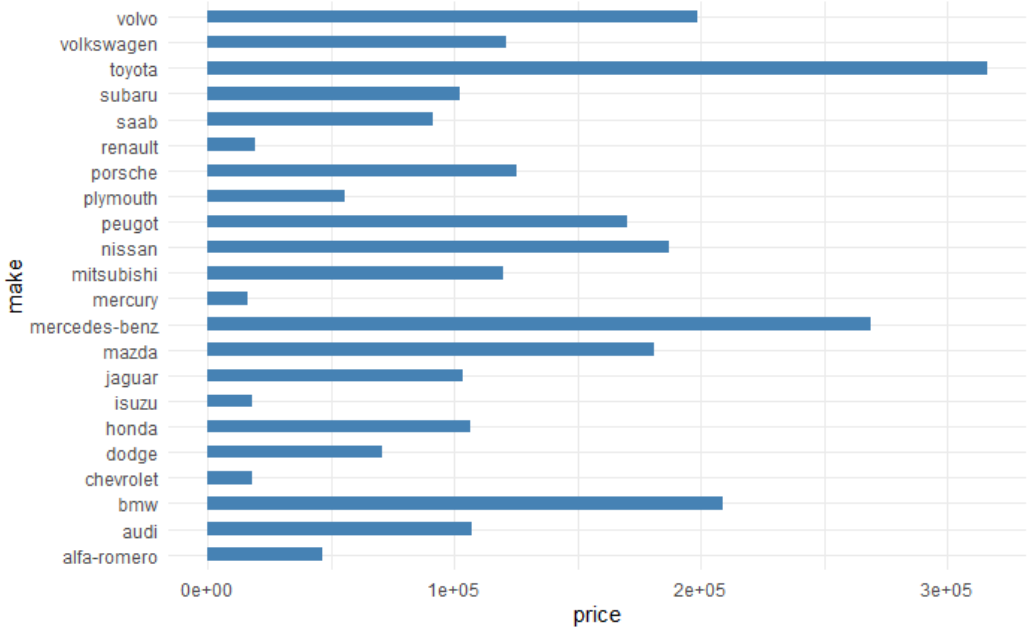

Figure 6

In **Figure 6,** it shows that Toyota is the most expensive; it is possible it can be altered from our data cleaning techniques. However, Toyota was the most collected vehicle out of the whole dataset; it does make sense why it would be the most expensive. Besides Toyota, our luxury brands such as BMW, Mercedes-Benz, and Volvo come up second in price. So car make or brand may have an effect on price, but since it is categorical variable we cannot be too sure. Now let's check out some numerical variables using exploratory data analysis.

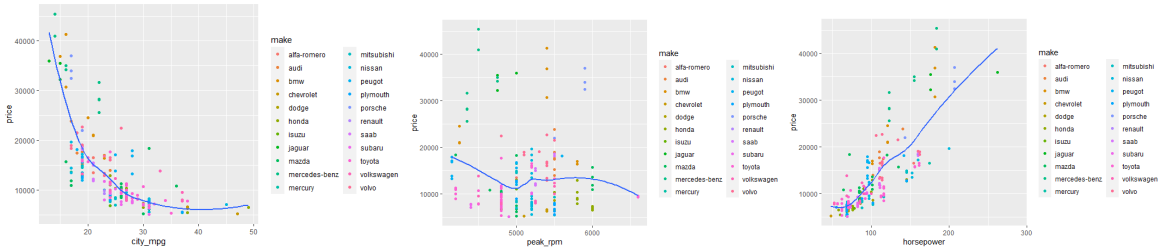

Figure 7: Certain variables on price

As we could see the variables have some correlation to price. In the first plot, it is a negative correlation. The second plot has a slightly negative correlation and the last plot has a positive correlation and it almost looks linear. These are very interesting plots that can help provide us certain variables for linear regression.

Linear regression only works with numerical data; categorical data cannot work in linear regression. So we will start with the full model with only numerical features. We saw a lot of correlation with price from **Figure 7**, so we will create a model with a response variable on price. So our full model has 15 features including symboling, normalized losses, wheelbase, length, width , height, curb weight, engine size, bore ,stroke, compression ratio, horsepower, peak rpm , city mpg, and highway mpg. Using R, we plotted out the diagnostic plots of this model.

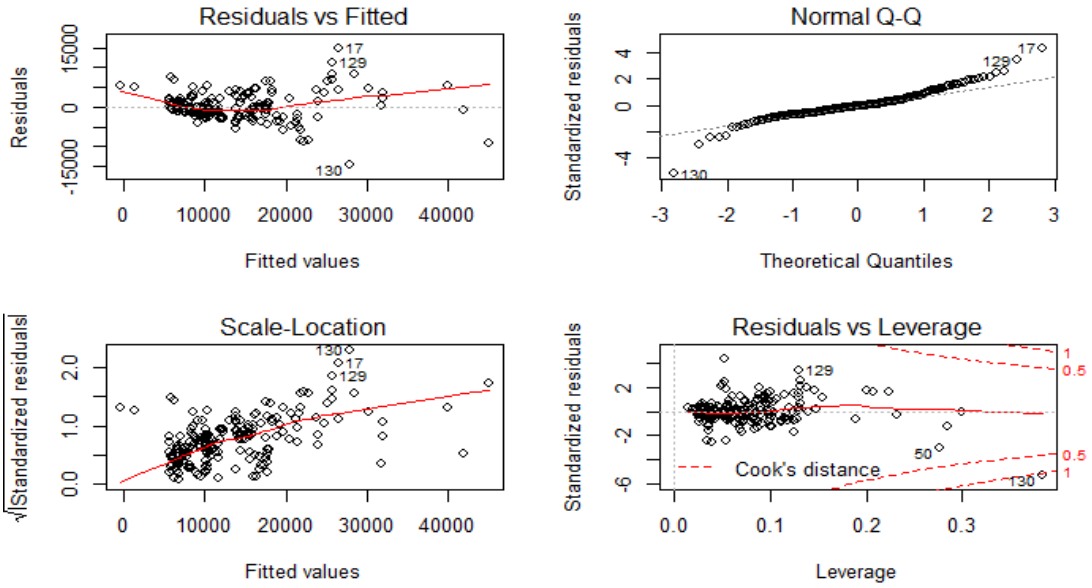

Figure 8

Based on the diagnostic plot of this model, we can see that the residuals are clustered together and show a lot of outliers. The QQ plot shows that there is some nonlinearity due to outliers. However, our R^2 is .813 which means the model fitted decently. But these diagnostic plots show a lot of violations towards the assumptions.

First we will test if there is any multicollinearity with between the variables. This is to see whether or not that the variables are independent of each other. This means that they are not highly correlated which can affect the model.

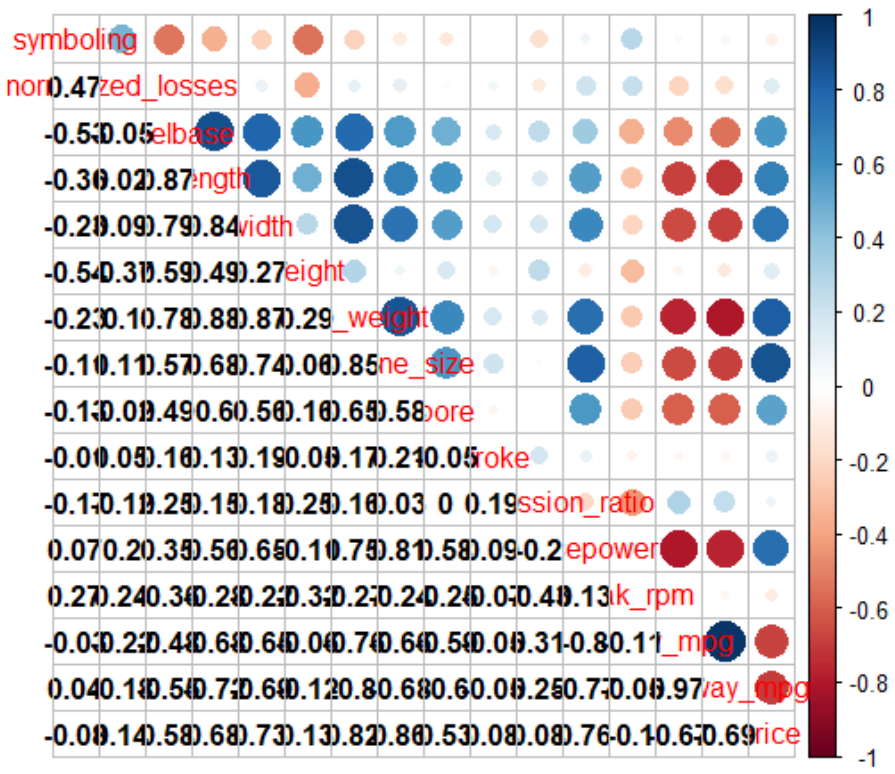

Figure 9: Correlation Plot of features of the full model

As we can see there is actually a lot correlation between many of the variables which results in multicollinearity. To further support this case, we use the Variance Inflation Factor, VIF. This measures the amount of multicollinearity in a set of multiple regression variables. According to results from R, city mpg, highway mpg, curb weight, length, wheelbase, horsepower, engine size, and width have a VIF score higher than 5 which indicates high multicollinearity. So we drop these variables. After dropping these variables, the diagnostic plots were still similar as in **Figure 8.** The R^2 for the model without high VIF variables was .37. In fact, this means that this model was under fitting data which is flawed.

So we will use the stepwise regression to get the best model. Stepwise regression utilizes AIC to get the best model. The lower the AIC the better the model would be. It can start with the full model and reduce to the model with the lowest AIC. After running it in R, the model had 7 predictor variables: symboling, wheelbase, engine size, stroke, compression ratio, peak rpm and city mpg. The R^2 was .81. The model fit decently again.

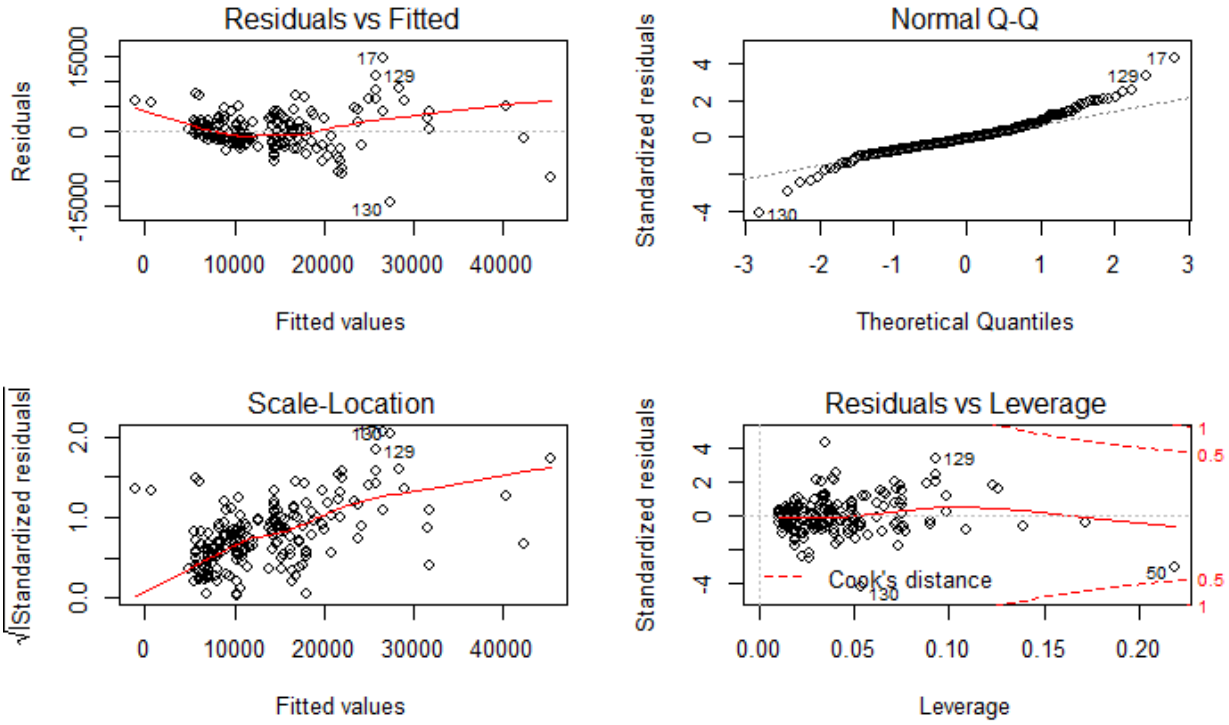

Figure 10: Diagnostic Plots for the stepwise linear model

In **Figure 10,** the residual plot shows the clustering still. The QQ plot is slightly better than the full model and certain outliers still have high leverage on this model. So our best model with just numerical variables was the full model, but it violates so many assumptions. So let's refer back to **Figure 6.** This figure showed association between the categorical variables and price. One way to implement categorical variables into the model is one hot encoding. One hot encoding allows us to create dummy variables with numeric values to implement in the linear regression model. After one hot encoding all the categorical variables, our dataset now has 77 columns. There is additional 51 dummy variables now. The only issue now that there is multicollinearity regardless due to the dummy variables.

Similar as before we check the full model with all 77 features to the response variable price. It returned an R^2 score of .96 but similar as the models before it violated many features as we can see here.

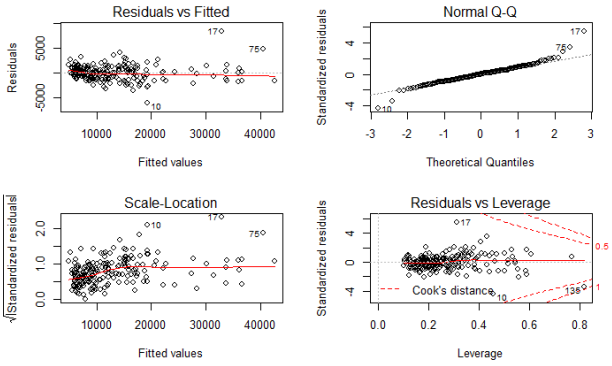

Figure 11 : Diagnostic Plot the full model (one hot)

This model is supposedly over fitting based on the diagnostic plots above. So let's decide to find the best model with the categorical dummy variables. We will use the stepwise regression and the VIF to find our best model. This way, we can find a decently fitted model with almost to none multicollinearity. So our current model **lm(formula = (1/price) ~ `make_alfa-romero` + make_audi + make_bmw + make_isuzu + `make_mercedes-benz` + make_mitsubishi + make_plymouth + make_porsche + make_saab + aspiration_std + body_style_hatchback + height + engine_type_dohc + engine_type_dohcv + fuel_system_2bbl + bore + stroke).** This has linear transformation applied to it due to the nonlinearity.

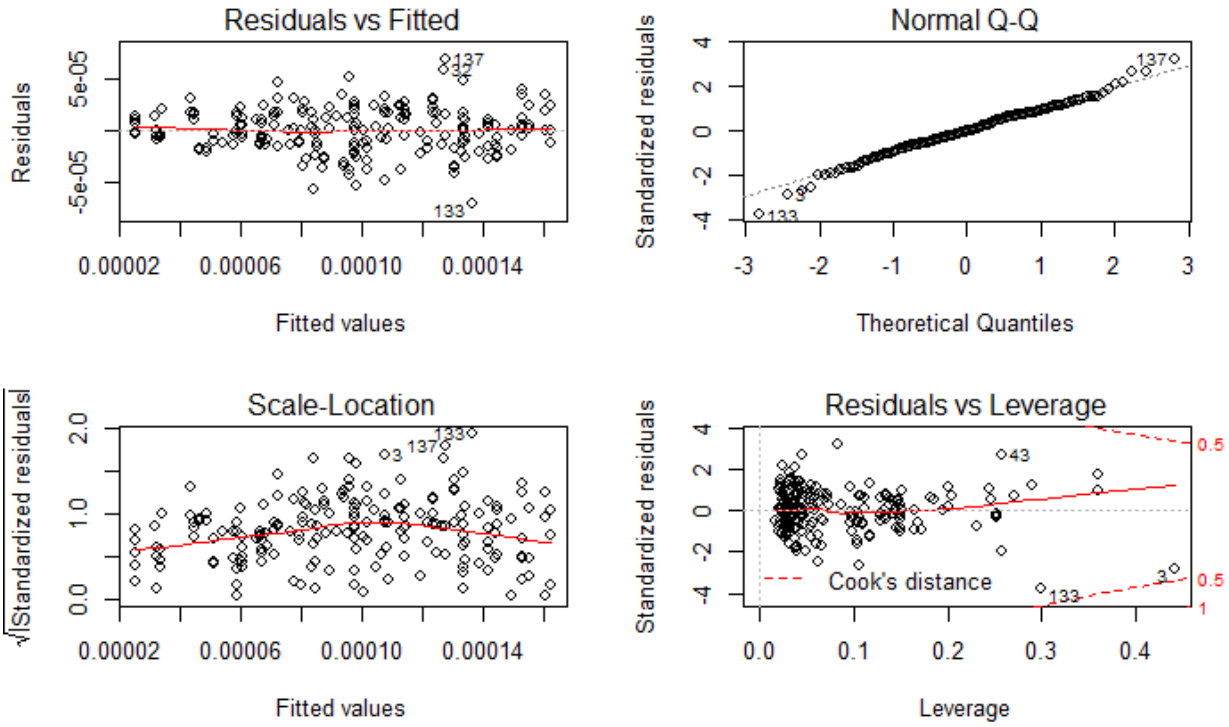

Figure 12 Diagnostic Plots with the current model

As seen in **Figure 12,** the residual plots look evenly distributed and pretty sparse, while the QQ plot is primarily linear. So far this model seems to pass each assumption of linear regression. The model has an R^2 of .7464. Compared to the previous models above, this model fits decently. Even though it may not fit well, the assumptions of linear regression were violated. This is the best model we can come us so far.

Conclusion and Discussion:

The final model may have not been the best model, but it fits the assumptions of linear regression. There were many issues that arose in this report such as multicollinearity, nonlinearity, and outliers. Yet, mulitcollinearity was most severe as seen in **Figure 9**, most of the variables were correlated with each other. In this case, a different approach can be viable for these issues. We can use LASSO or ridge regression to deal with the multicollinearity and probably get a better model.

References:

1. James, Gareth, et al. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2017.
2. Agresti, Alan. *An Introduction to Categorical Data Analysis*. Wiley, 2019.