**Introduction**

This mini project aims to show what we have learnt from the art and craft of Data Science - from Problem Formulation, Data Preparation, Exploratory Analysis, Pattern Recognition to Machine Learning. The source of the data we used was gotten from SgCarMart, Singapore’s most popular car buying and selling website with over 20,000 car listings.

**Motivation**

We all know that cars here in Singapore are very expensive as compared to other countries. Therefore, it takes considerable amount of time in deciding to get and own a car. Prices for brand new cars in Singapore are price higher mainly because limited road space. Thus, many people have opt to get a second-hand or used car instead of a brand new one. As for most used car are priced more reasonably and fit in more people’s budget of getting a car.

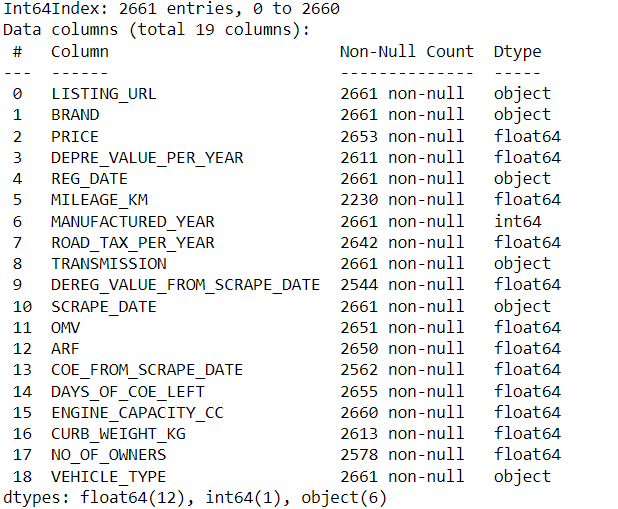
Therefore, our aim in this mini-project is to implement a Linear Regression Model, using a dataset to determine or form some sort of range of prices, what quantify as a reasonable price range for a used car should be.

**Data Collection**

The way we got our data was through using a web-scraper. The web-scraping was done through using Python, together with a package called BeautifulSoup. This allows us to retrieve the data, as the web-scraper will iterate through the search pages of the SgCarMart website.

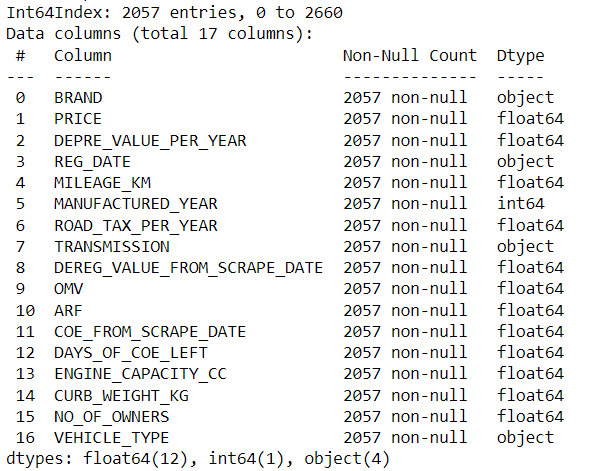
As it is iterating through the search pages which contains 100 search result of the listing posted of used car, it stores the links of those listing which is appended to a list. Through those individual listing links, it is able to scrap the features of that specific used car. All that information is appended to a dataframe which is stored later into a CSV file.

**Data Preparation**

****

After web-scraping, we had a dataframe that contained 2661 rows(entries) and 19 columns(features) of unique listings of vehicles.

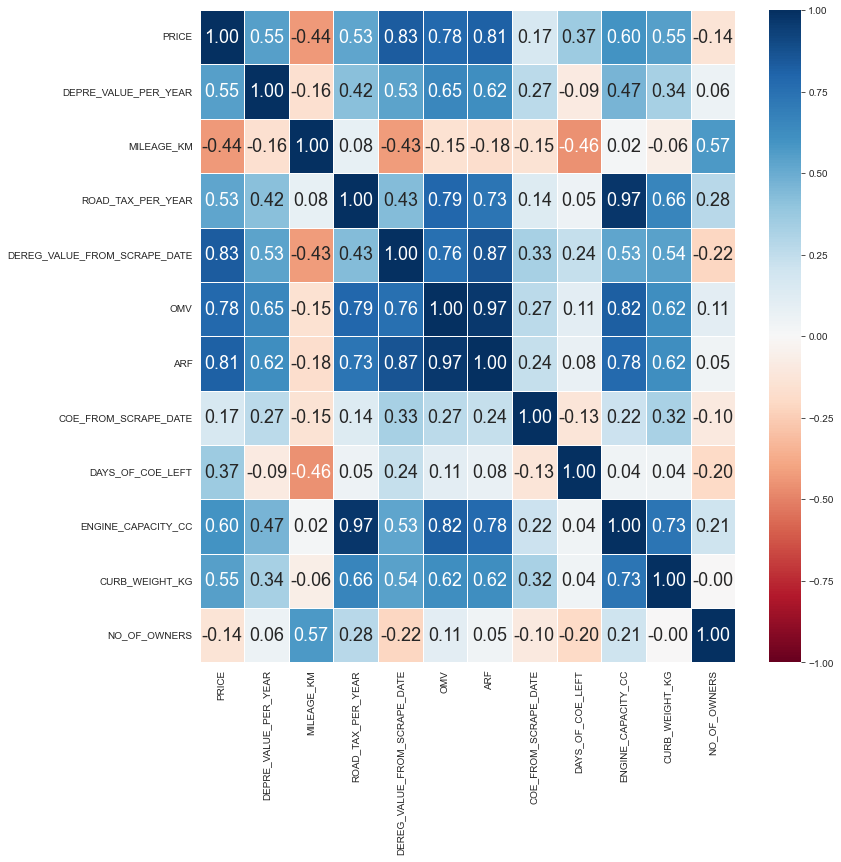
With a glance, we can see that there are missing entry (NULL) for some of the rows. We proceed to clean the data by drop the rows containing missing features and it could affect the result of the machine learning later on. We felt that the link URL and Scrape date, as we think it is irrelevant to predicting the resale value of a car.



After cleaning, this bought our dataframe down to 2057 entries with 17 columns.

**Data Visualization**

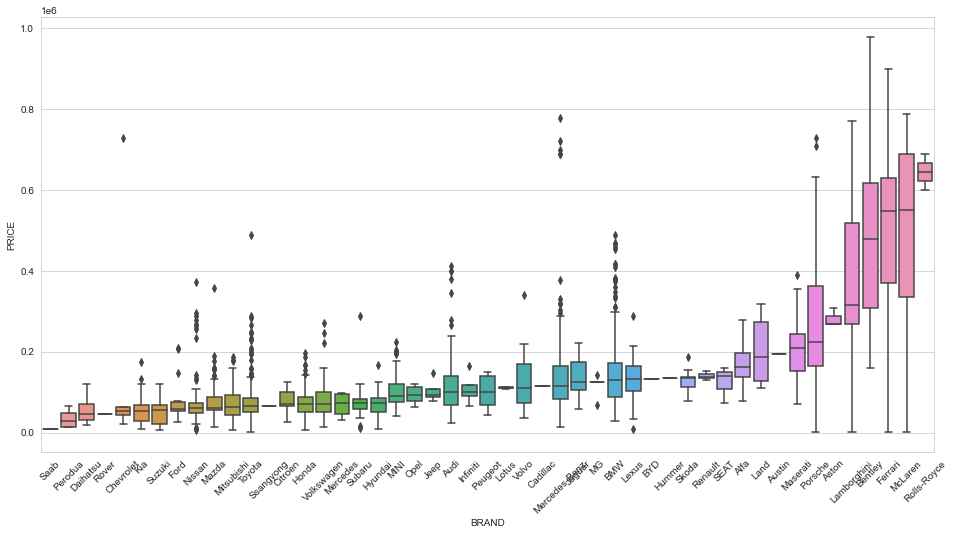
We went on to create a heatmap of a correlation matrix for the numerical data, which are the features with respect to the price, This will help us to easily visualize any variables that will be good in predicting the price.

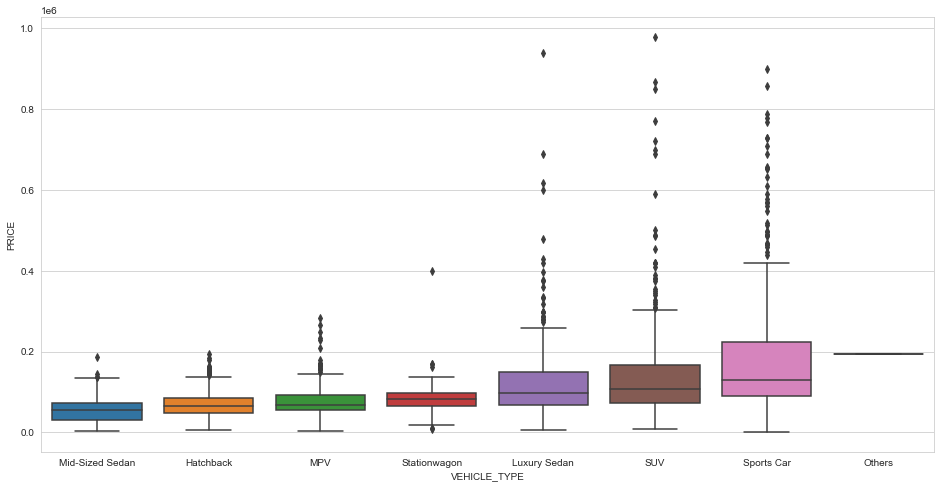


From the heatmap, we can easily see that both ARF ( Additional Registraion Fee ) and OMV ( Open Market Value) are Highly correlated ( >0.7 correlation with Price ). There are also Road Tax, Curb Weight and Engine Capacity which are decently correlated ( >0.5 correlation )

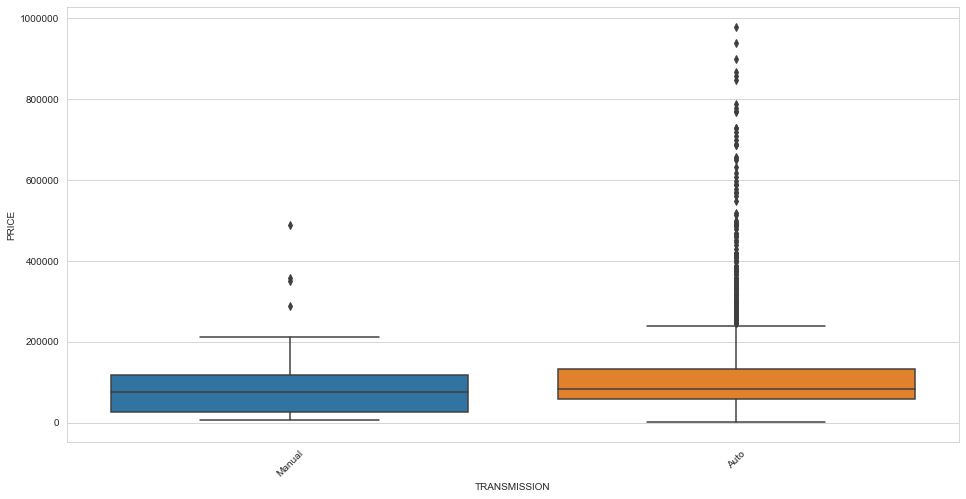
Next, we want to see the correlation between categorical data and the price. Since we need to see the correlation between categorical data and a numerical data, we decided to use boxplot.

Between Prices and Car Brands,



Between Prices and Vehicle Type,

And lastly between Prices and the type of transmission used.



From the boxplot, we can that Brand is the more strongly correlated to prices as the boxes overlap lesser with one another when compared to vehicle type and transmission.

Ideally, we want our model to be able to determined/predict the price using variables (features) that are not as easy to determine the car price with just common human knowledge. Meaning, with higher OMV and ARF, it will easily means that the price of that car will be higher as well. A person could just roughly guess the price range of a vehicle without using our model.

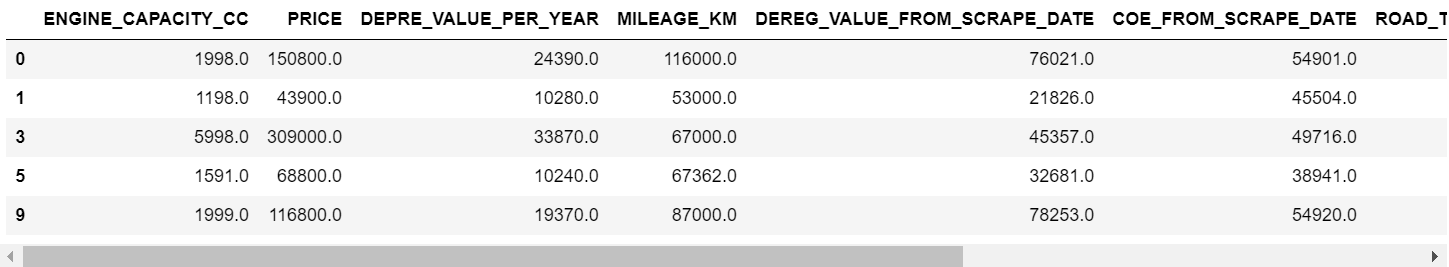
So for our model, ideally we wanted to use features that are decently correlated, we left us with:

Response Variable: Price

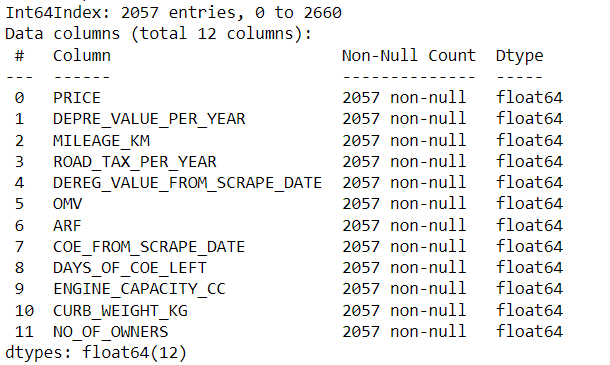
Predictor Variables: Brand, Vehicle Type, Transmission, Curb Weight (KG), Engine Capacity (CC), Road Tax per Year.

**Machine Learning**

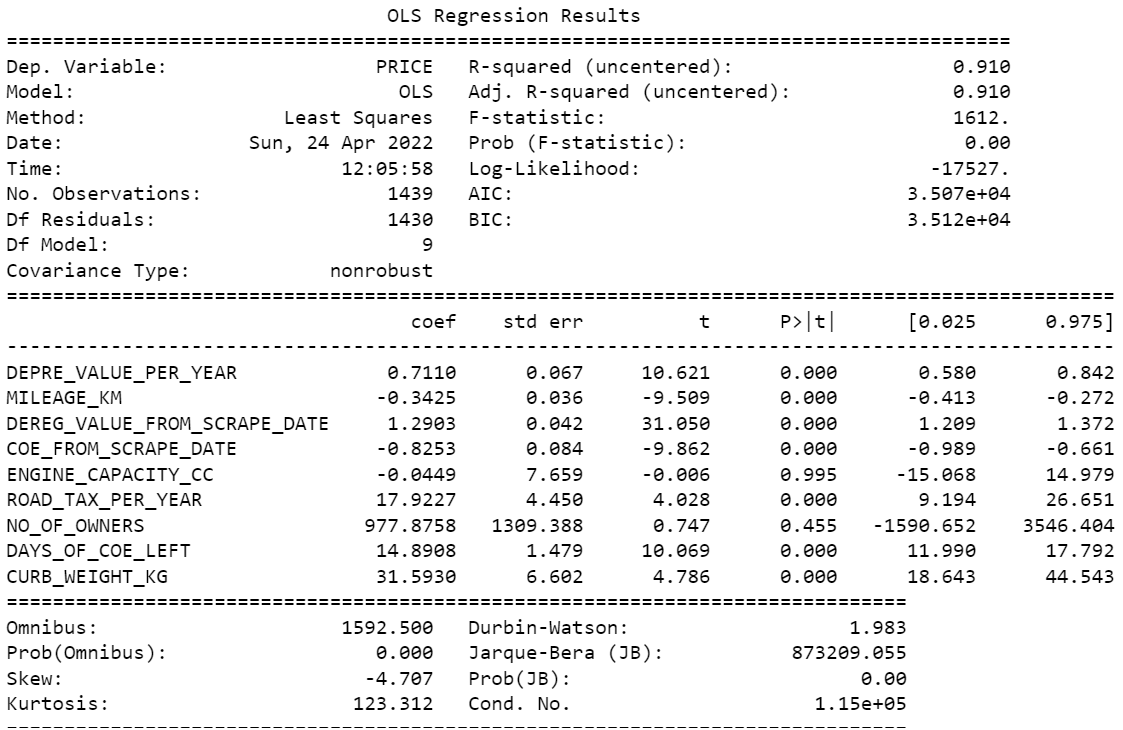
After cleaning the data again to be left with the features that we needed for the model, we have:



WIth the data ready, we went and split the data into train and test set (70% training and 30% test).



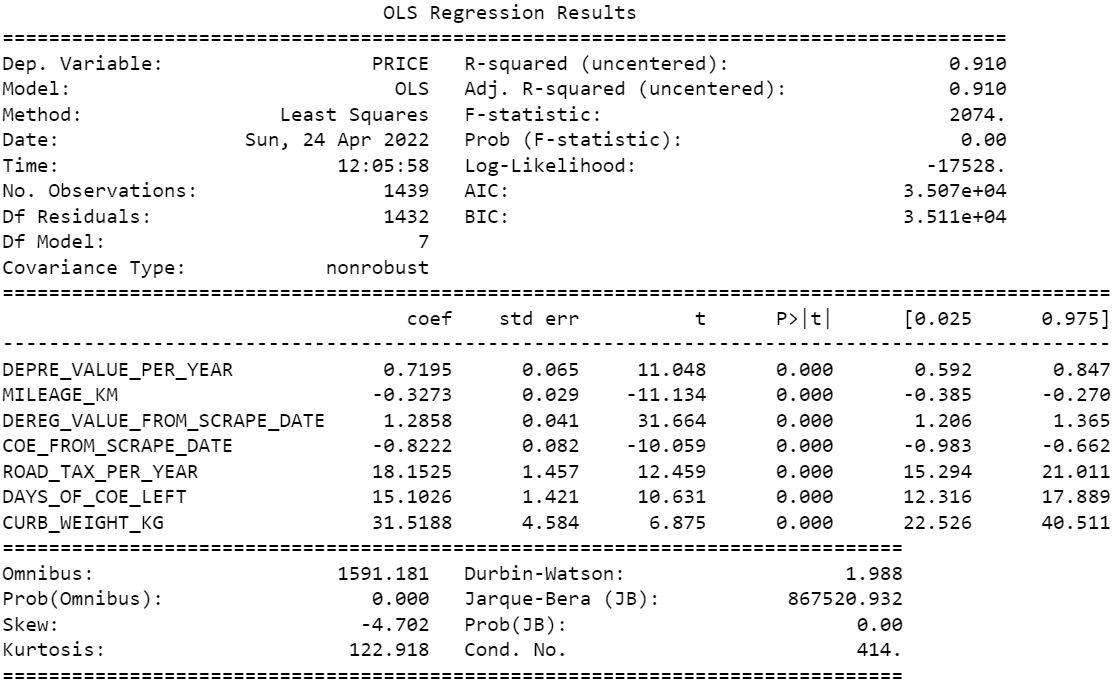
With the dataset ready, we went on to create our linear regression model. We gotten the result:



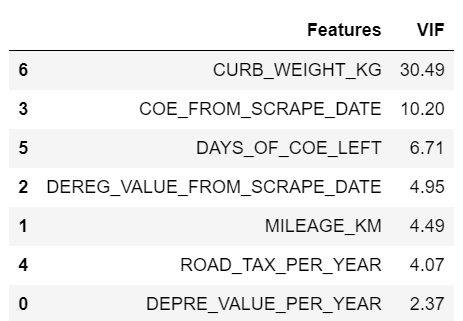
With the result, we can see that Number of previous owners has a high p value ( >0.05 ) indicate that our variable is not strong enough to suggest an effect exists in the population. An effect might exist but it's possible that the effect size is too small, the sample size is too small, or there is too much variability for the hypothesis test to detect it. Hence dropping it as it is insignificant in presence of other variables.

The same goes to the engine capacity of the vehicle. (0.995)

This gotten us a new result:

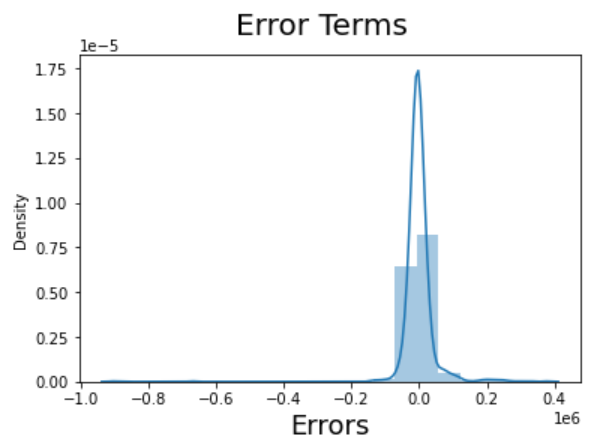


After making sure all the P values are looking good. We went on to check the VIF values.



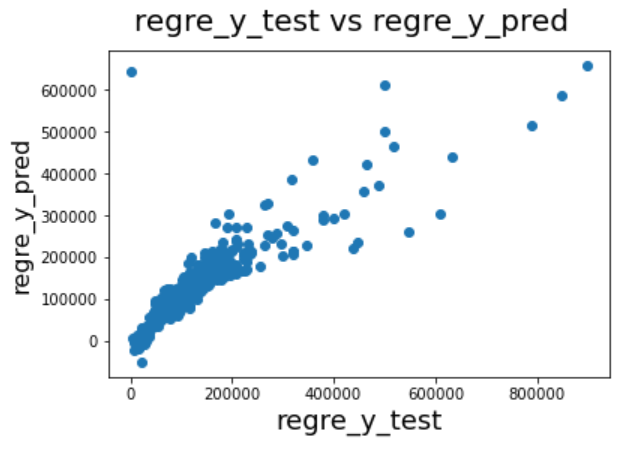
We can see that the curb weight of the car has a high VIF (30.49). This shows that curb weight has high multicollinearity, a statistical concept where several independent variables in a model are correlated. Therefore, we decided to drop it along with the coe value gain from scrapping the car today (10.20).

With that, we plot the residual analysis of the model.



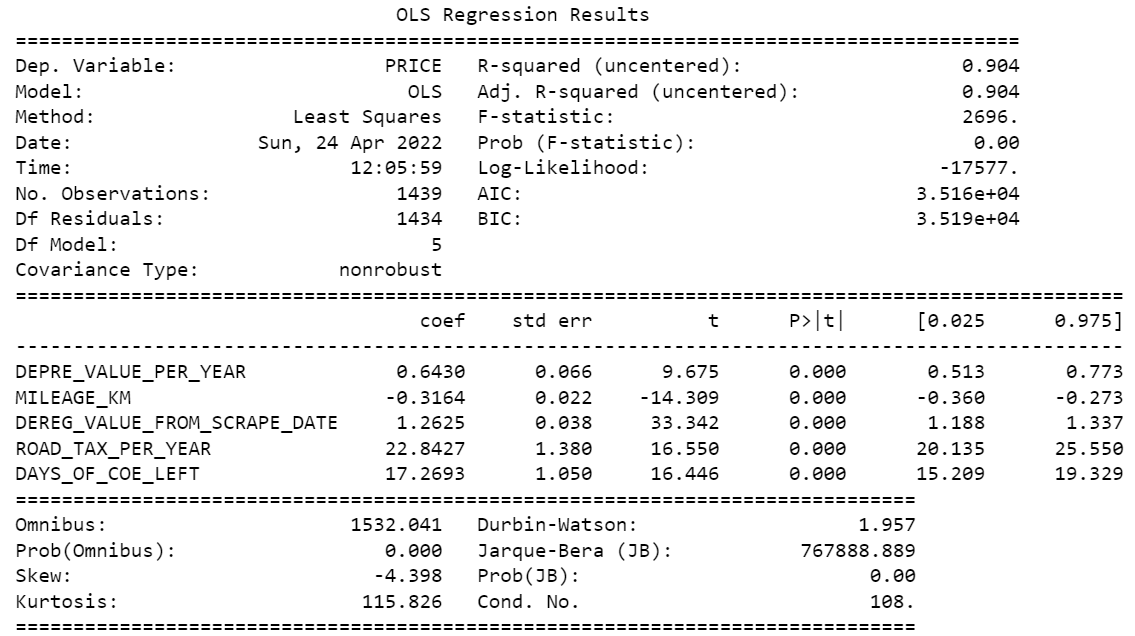
With the error terms not too far off from being normally distributed, it shows the linear modeling is on the right track.

Moving on to the last part, where the prediction is being done.



We got a decent linear regression with a few outliers.

This concludes our findings.



*R-squared and Adjusted R-squared (extent of fit)* - 0.904 and 0.904 - 90% variance explained.

*p-values* of all the coefficients are 0.00. - means that all the predictors are statistically significant.