INDIAN STATISTICAL INSTITUTE, DELHI CENTRE

Exploratory Data Analysis

and

Prediction of CarPrice Data

Project

Name: Kaulik Poddar

Roll Number: MD2207

Supervisor: Prof. Dr. Deepayan Sarkar

Session: 2022 - 2024

Date of Submission: 6 November 2022

Contents

1 Introduction

In this current generation car has become one most important thing in human life. When we talk about thing a thing, automatically the first question is asked about it's price.In this case, our motive is also to do work with car price. Now, there are so many factors on which price of a car depends. In this project I will show how different factors affect the price of car. Ofcourse here my response is price of car. There are so many predictors among which, some are numerical variable and some are categorical predictor. In this project with data analysis I also some car price prediction using a regression model.

2 Objective

In this project my two prior objectives are doing Exploratory Data Analysis(EDA) and prediction of car price.

a. Under EDA I did the followings ————-

i. To check whether there is any missing values.

ii. To see the variability and skewness of the numerical variables.

iii. To check presence of outliers in each of the numerical variables.

iv. Checking collinearity among the numerical predictors.

v. Observing the relation of continuous variables with car price using scatterplot.

vi. Observing the relation of categorical variables with car price using boxplot.

b. Under Prediction of car price I did the followings————

i. Breaking the original dataset into train set and test set.

ii. Fit a proper model on train set.

iii. Then finally predict the response variable for test set.

3 Description of the Dataset

Source: The dataset is obtained from Kaggle : CarPricePrediction

Statement: A Chinese automobile company Geely Auto aspires to enter the US market by setting up their manufacturing unit there and producing cars locally to give competition to their US and European counterparts. They have contracted an automobile consulting company to understand the factors on which the pricing of cars depends. Specifically, they want to understand the factors affecting the pricing of cars in the American market, since those may be very different from the Chinese market.

The company wants to know: Which variables are significant in predicting the price of a car How well those variables describe the price of a car Based on various market surveys, the consulting firm has gathered a large dataset of different types of cars across the Americal market.

Description:

• The dataset has total 205 data points.

• The dataset has 26 columns and they are given below ———

i. Car ID

ii. symboling

iii. CarName : Name of the cars.

iv. fueltype : Type of fuel in car.

v. aspiration : Type of aspiration in car.

vi. doornumber : Number of doors in car.

vii. carbody : Type of car body.

viii. drivewheel : Drive wheel of the car.

ix. enginelocation : Location of engine in the car.

x. wheelbase : Base of wheel.

xi. carlength : Length of car.

xii. carwidth : Width of car.

xiii. carheight : Height of car.

xiv. curbweight : Weight of curb.

xv. enginetype : Type of engine.

xvi. cylindernumber : Number of cylinder.

xvii. enginesize : Size of engine.

xviii. fuelsystem : Type of fuelsystem.

xix. boreratio : Bore ratio of car.

xx. stroke : Stroke of car.

xxi. compressionratio : Ratio of compression.

xxii. horsepower : Horsepower of car.

xxiii. peakrpm : Maximum value of RPM of car.

xxiv. citympg : The score a car will get on average in city conditions, with stopping and starting at lower speeds.

xxv. highwaympg : the average a car will get while driving on an open stretch of road without stopping or starting, typically at a higher speed.

xxvi. price : Price of car

Among these columns there are some non numerical variables. Now we will note down their levels ————-

i. **CarName** : ”alfa-romero ”,”audi ”,”bmw”,”buick ”,”chevrolet”, ”dodge ”,”honda ”,”isuzu”,”jaguar”,”maxda ”,”mazda ”,”mercury cougar”,”mitsubishi ”,”Nissan ”,”peugeot ”,”plymouth ”,”porsche”, ”renault ”,”saab ”,”subaru”,”toyota ”, ”volkswagen ”,”volvo ”.

ii. **fueltype** : ”gas” , ”diesel”.

iii. **aspiration** : ”std” , ”turbo”.

iv. **doornumber** : ”two”, ”four”.

v. **carbody** : ”convertible”, ”hardtop”, ”hatchback”, ”sedan”, ”wagon”.

vi. **drivewheel** : ”fwd” , ”rwd”.

vii. **enginelocation** : ”front” , ”rear”.

viii. **cylindernumber** : ”two” , ”three” , ”four” , ”five”, ”six”, ”eight”, ”twelve”.

ix. **fuelsystem** : ”1bbl”, ”2bbl”, ”4bbl”, ”idi”, ”mfi”, ”mpfi”, ”spdi”, ”spfi”.

4 Analysis of Dataset

In this section, we will do exploratory analysis of the dataset.

4.1 Checking Missing Values

In the dataset, there is no missing value.

4.2 Analysis of Numerical Variables

Let us first report the values of average, variabilty and skewness of each of the variables.

	Median	IQR	Skewness
wheelbase	97.000000	7.9000000	1.04251361
carlength	173.200000	16.8000000	0.15481032
carwidth	65.500000	2.8000000	0.89737535
carheight	54.100000	3.5000000	0.06265992
curbweight	2414.000000	790.0000000	0.67640218
enginesize	120.000000	44.0000000	1.93337485
boreratio	3.310000	0.4300000	0.02000863
stroke	3.290000	0.3000000	-0.68464767
compressionratio	9.000000	0.8000000	2.59171962
horsepower	95.000000	46.0000000	1.39500643
peakrpm	5200.000000	700.0000000	0.07460766
citympg	24.000000	11.0000000	0.65883775
highwaympg	30.000000	9.0000000	0.53603793
price	9.239414	0.7509581	0.66795492

Figure 1: Reporting median,IQR,skewness

Here, we report the average values by taking median and variability by Inter Quartile Range (IQR).Hence, we can see that except stroke every variable is positvely skewed.

To understand the skewness graphically, we plot histogram of every numerical variables.

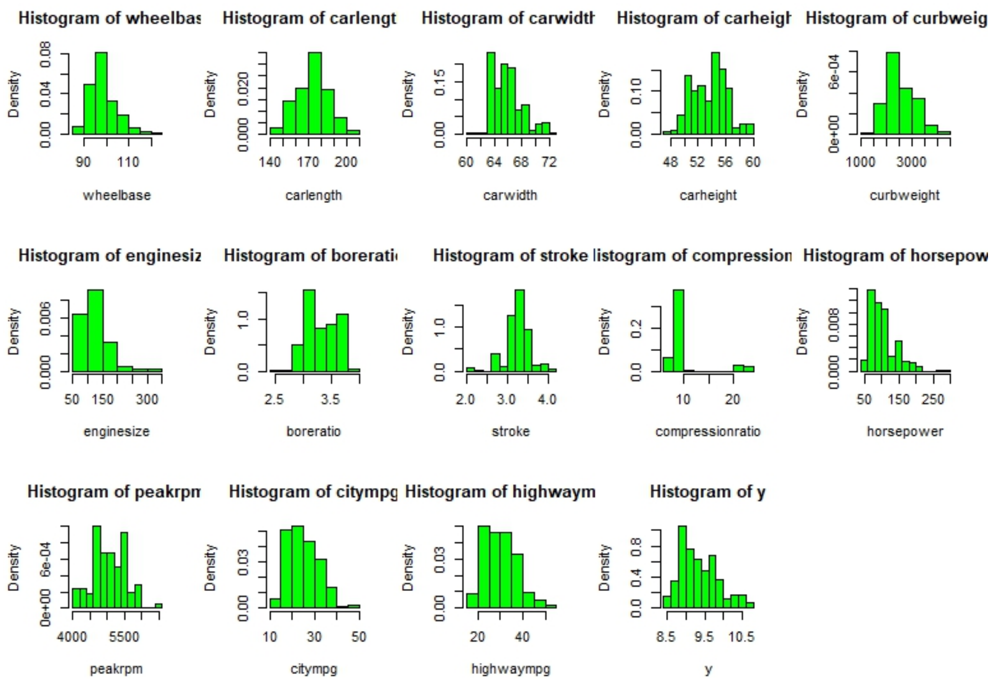

Figure 2: Histogram of numerical variables

Now we will see the presence of outliers in each numerical variables. For this I use Boxplot. The boxplot also gives us the visualization of median, variability, whiskers etc which helps us very much.

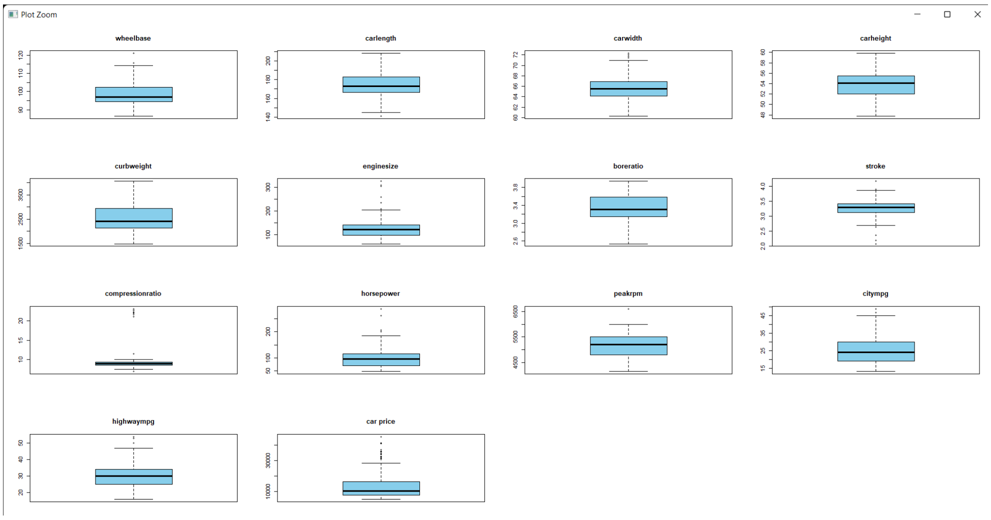

Figure 3: Boxplopt of numerical variables

The above diagram tells us that wheelbase, carwidth, enginesize, stroke, compressionratio,horsepower,citympg,highwaympg and carprice have outliers.

4.3 Relationship Between ”carprice” and Different Numerical Variables

In this section we will see how much different numerical variables affect carprice. For this we will use a graphical tool - Scatterplot. By watching the graph, we will comment whether that variable affects car price or not.

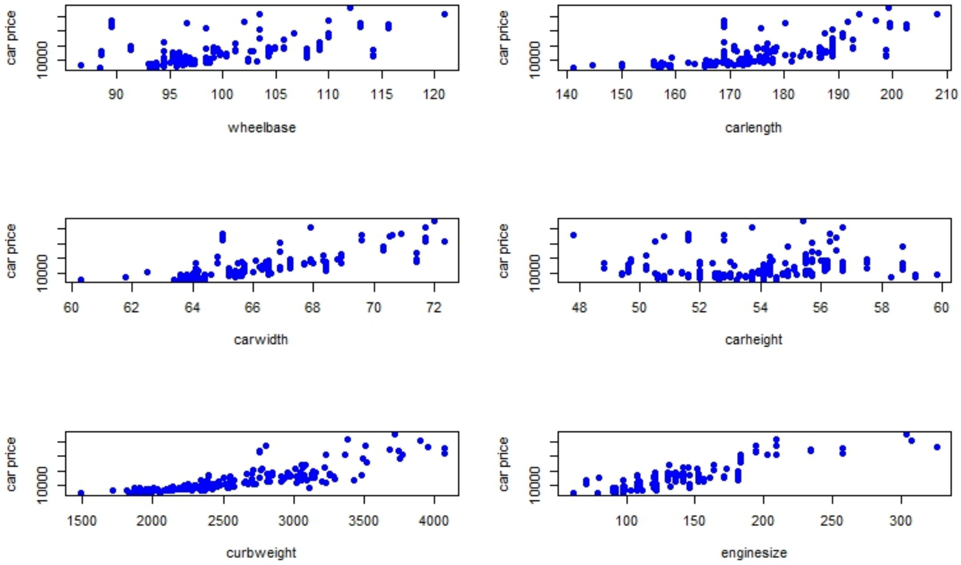

Figure 4: Scatterplot of different numerical variables vs carprice

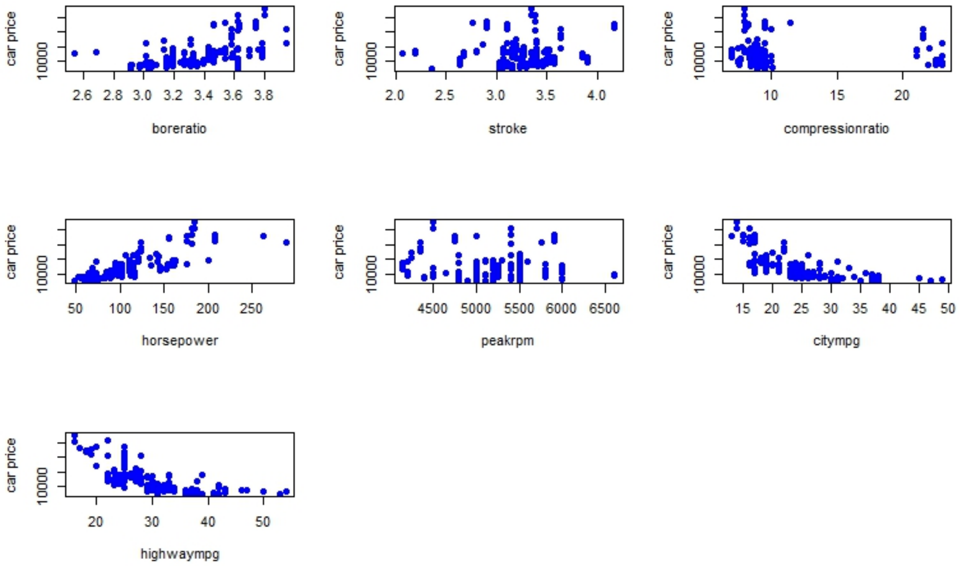

Figure 5: Scatterplot of different numerical variables vs carprice

From the above figure we can say wheelbase, carlength, carwidth,curbweight, enginesize, boreratio, horsepower, citympg and highwaympg affect carprice significantly. So we can consider them as significant predictors for predicting carprice.

4.4 Relationship Between ”carprice” and Different Categorical Variables

Now we want to see whether the categorical variables affect the price of car. To see this we draw boxplot of ”carprice” vs each of the categorical variables.

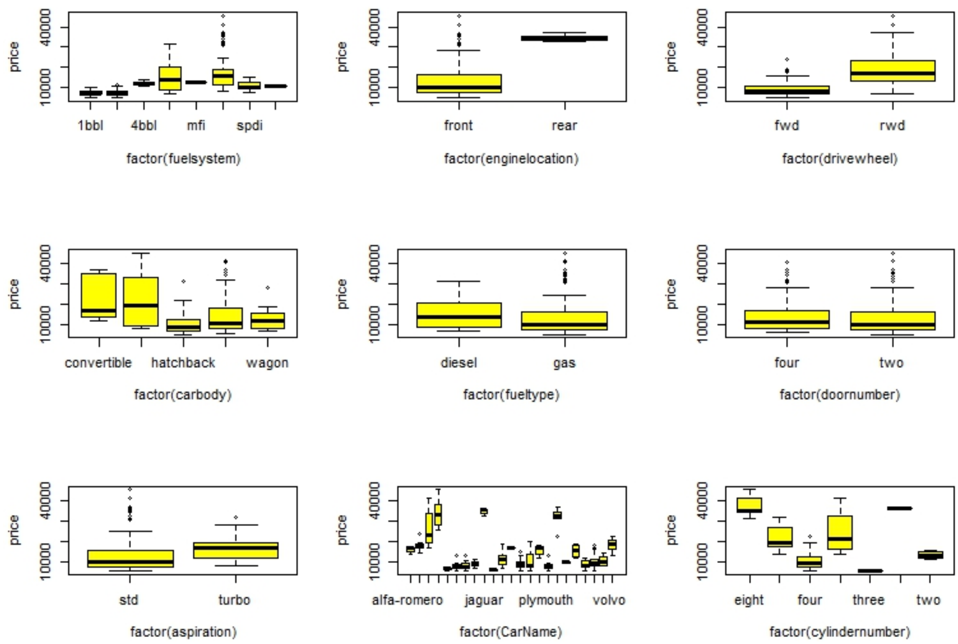

Figure 6: Boxplot of Different Categorical Variables vs ”carprice”

From the above figure we can see that on an average the price of car is same whether the car has ”two” or ”four” door. Hence, we can conclude that ”doornumber” does not affect the ”carprice”. Hence it is not a significant predictor. But price of car is mostly changed w.r.t ”enginelocation”, ”drivewheel”, ”carbody”, ”CarName” and ”cylindernumber”.

4.5 Checking Collinearity in Numerical Predictors

In section 4.3, we saw that there are some numerical variables which affects ”carprice”. Now it may happen that the predictors are correlated among themselves. Then it is not necessary to consider each variable as predictor, because if one variable affect our response variable then automatically the correlated variables do the same.

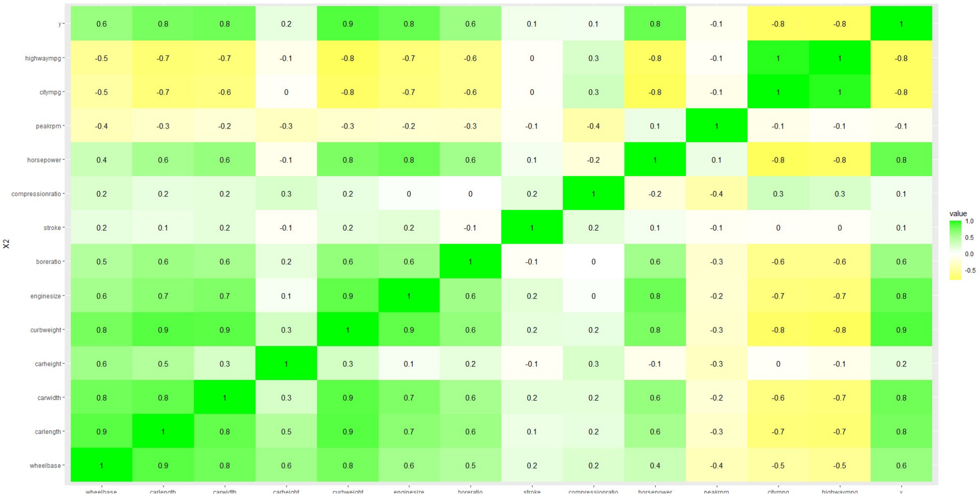

Figure 7: Correlation Heatmap of Numerical Variables

From the above figure, we can see that there is traces of collinearity in the variables. We assume the cut off point as $|0.8|$, ie if the value of correlation coefficient is either $>= 0.8$ or $<= -0.8$, then we consider that two predictors to be correlated.

5 Data Preparation

5.1 Transformation of Categorical Predictors

Now we have a dataset with a response variable and some numerical and categorical predictors. Now we convert the categorical variables into dummy variables so that we can use it as a regression model. And we delete the first column dummy set of each variable to remove multicollinearity from the new dataset.

After doing this dummification we have now total 63 predictors.

5.2 Breaking the Dataset into Train Set and Test Set

Now I want to predict some car price based on my given data. So I have decided to break my dataset into train set and test set. Based on the data points in train set I build a model and through that model I will predict the value of response variable of test set.

In the original dataset I have total 205 data points. From that I took 171 data points in train set and 34 in test set.

6 Model Building and Model Modification

Now we want to fit different model on the train set and select the best one. Here, our variable of interest is price of car.

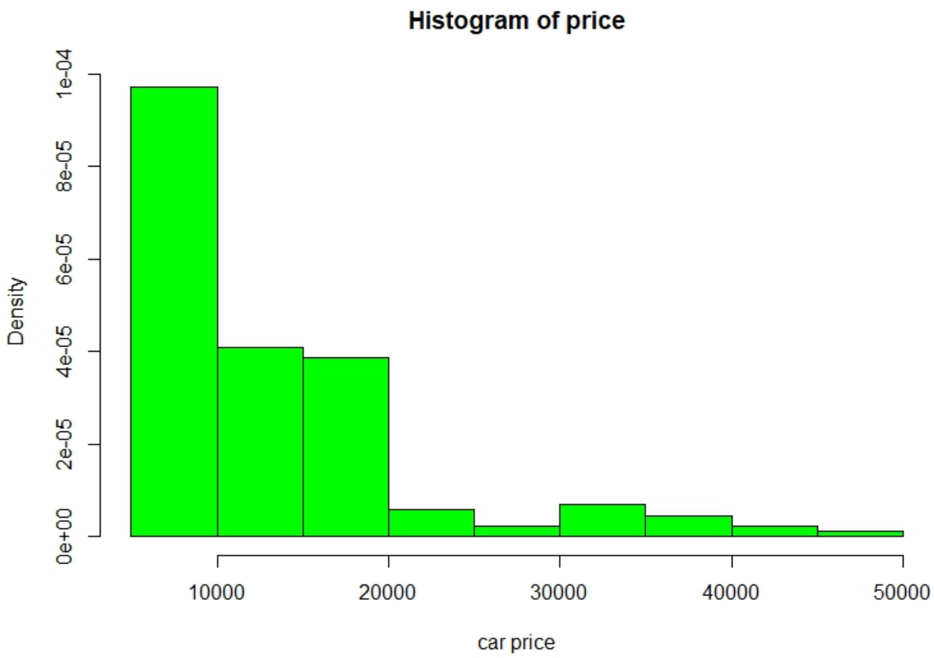

Figure 8: Histogram of ”car price” in train set

Clearly, the price is positively skewed, so I do log transformation of price and consider that as my response variable.

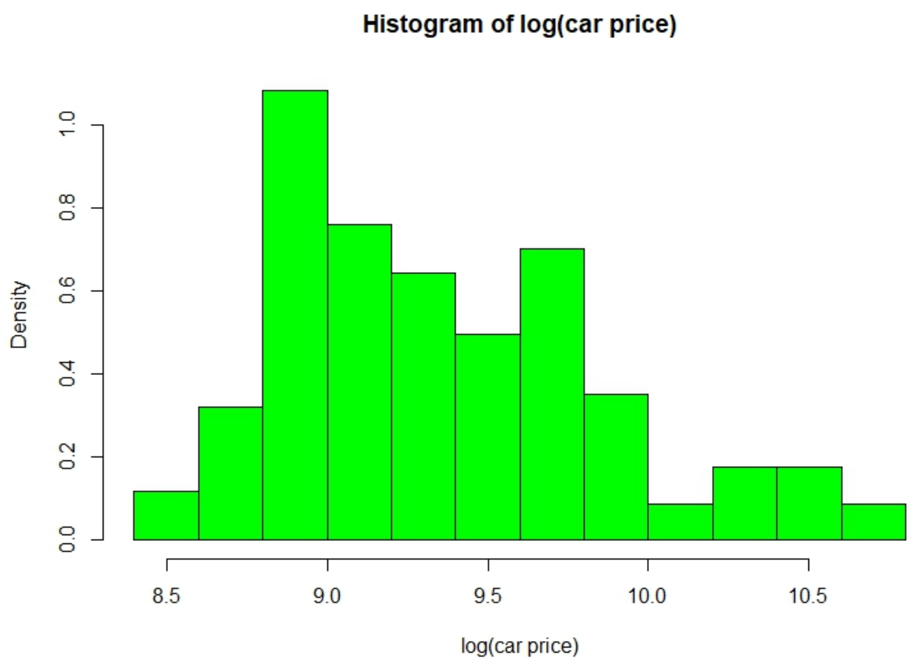

Figure 9: Histogram of "log(car price)" in train set

From the above diagram we can see that there is slightly positive skewness in the dataset. But we can consider it more or less symmetric.

6.1 Model 1:Considering All the Predictors

Here, we fit our response variable on all predictors. The following table gives us the output of regression model.

Predictors	Estimate	Std.Error	t value	p value	Predictors	Estimate	Std.Error	t value	p value
(Intercept)	7.752436	1.198647	6.467657	2.58E-09	`CarName_plymouth `	-0.29418	0.1261624	-2.331764012	0.021469
wheelbase	0.015431	0.006056	2.548188	0.012159	CarName_porsche	0.42197	0.3775071	1.11777986	0.266011
carlength	-0.00737	0.00341	-2.16089	0.032796	`CarName_renault `	-0.21178	0.1505396	-1.406818121	0.162202
carwidth	0.02796	0.01547	1.807402	0.073337	`CarName_saab `	0.209766	0.1350574	1.553163043	0.123156
carheight	-0.01813	0.009863	-1.83862	0.068575	CarName_subaru	-0.11038	0.1327501	-0.831485959	0.407437
curbweight	0.000487	0.000106	4.611819	1.05E-05	`CarName_toyota `	-0.14179	0.1068615	-1.326862096	0.187206
enginesize	0.004654	0.001704	2.730917	0.00732	`CarName_volkswagen `	-0.05775	0.1189067	-0.485654828	0.628144
boreratio	-0.27005	0.126882	-2.12837	0.035459	`CarName_volvo `	0.035935	0.1514937	0.237205156	0.812923
stroke	0.049935	0.072704	0.686823	0.493589	fueltype_gas	-0.69263	0.4571538	-1.515086876	0.132518
compressionratio	-0.03724	0.034103	-1.09187	0.277194	aspiration_turbo	0.121016	0.0582384	2.077947743	0.039959
horsepower	-0.00103	0.001717	-0.60185	0.548467	doornumber_two	-0.06446	0.0337206	-1.911542808	0.058445
peakrpm	6.24E-05	4.75E-05	1.314041	0.19147	carbody_hardtop	-0.22055	0.0908454	-2.427799452	0.016755
citympg	-0.01163	0.009508	-1.22269	0.22397	carbody_hatchback	-0.23296	0.0783261	-2.974241035	0.003586
highwaympg	0.010583	0.008438	1.254129	0.212361	carbody_sedan	-0.21742	0.0853928	-2.546133328	0.012227
`CarName_audi `	0.188856	0.145723	1.295996	0.197594	carbody_wagon	-0.24032	0.0934683	-2.571097342	0.011425
CarName_bmw	0.48008	0.16331	2.939679	0.003978	drivewheel_rwd	0.09161	0.0522602	1.752952567	0.082298
`CarName_buick `	0.066165	0.172975	0.382516	0.702791	enginelocation_rear	0.457927	0.3048936	1.501924211	0.135882
CarName_chevrolet	-0.18159	0.172041	-1.05548	0.293436	enginetype_l	0.390885	0.295286	1.32375163	0.188234
`CarName_dodge `	-0.28034	0.127011	-2.20723	0.029301	enginetype_ohc	-0.12105	0.0866446	-1.397125832	0.165089
`CarName_honda `	0.076707	0.146767	0.522645	0.602236	enginetype_ohcv	-0.0952	0.0831082	-1.145482382	0.254408
CarName_isuzu	-0.10917	0.143054	-0.76317	0.446937	enginetype_rotor	0.647304	0.3103358	2.085819308	0.039226
CarName_jaguar	-0.41501	0.190653	-2.17676	0.03156	cylindernumber_five	0.180442	0.2050649	0.879926914	0.38075
`CarName_maxda `	-0.20868	0.139423	-1.49675	0.137221	cylindernumber_four	0.295509	0.2458866	1.201811561	0.231927
`CarName_mazda `	0.021067	0.118082	0.178406	0.858721	cylindernumber_six	0.114487	0.1813645	0.63125389	0.529138
`CarName_mercury cougar`	0.027795	0.196602	0.141377	0.887822	cylindernumber_twelve	0.312323	0.3573049	0.874106702	0.383898
`CarName_mitsubishi `	-0.29196	0.129252	-2.25882	0.025796	fuelsystem_2bbl	0.225919	0.1005945	2.245835605	0.026642
`CarName_Nissan `	-0.04627	0.11448	-0.40419	0.68683	fuelsystem_mpfi	0.285393	0.107183	2.662673136	0.008873
`CarName_peugeot `	-0.75137	0.314527	-2.38888	0.018542	fuelsystem_spdi	0.271975	0.125901	2.160231674	0.032848
					fuelsystem_spfi	2.649e-01	1.831e-01	1.447	0.15071

Figure 10: Output when all the predictors are considered

With Multiple R-square = 0.9858,which is quite high. But only 20 out of 57 predictors are significant (from p-value). So, there is something suspicious in the model.The yellow coloured variable denotes significant predictors.

Now I want to see whether the residuals are homoschedastic or not.For that I plot the residuals vs the fitted values of response variable.

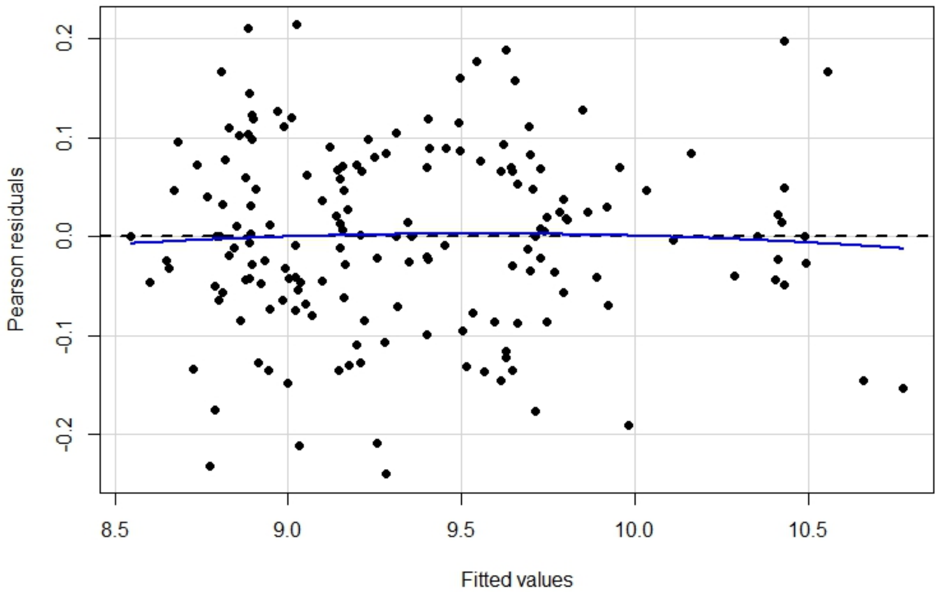

Figure 11: Plot of Residuals vs Fitted values of Response Variable

From the plot, I observe that the residuals are more or less homoschedastic. Now I want to see whether the residuals follow normal dist or not. For that I use qqplot.

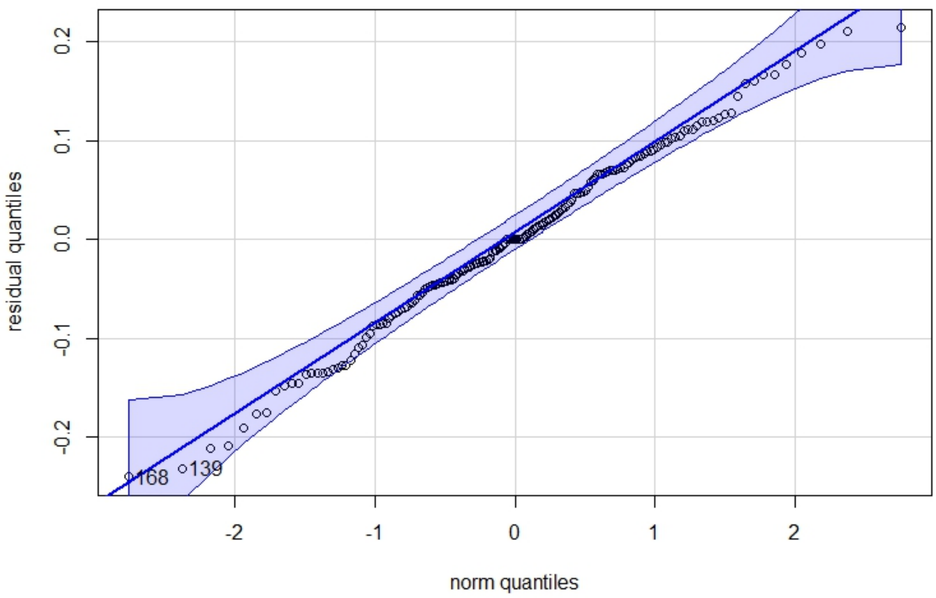

Figure 12: qqplot of residuals

From the plot we can see that residual quantiles coincides more or less with actual normal quantiles. Hence,the residuals follow normal distribution.

6.2 Model 2: Removing the Correlated Variables

Here, we want to fit a model removing the correlated predictors.Besides that, from the boxplot of car price vs doornumber, we can see that car price is invariant w.r.t the doornumber.Hence, we assume doornumber does not affect our response variable.

The following table gives us the output of regression model.

Predictor	Estimate	Std.Error	t value	p value	Predictor	Estimate	Std.Error	t value	p value
(Intercept)	9.004058	0.98782	9.115079	1.95E-15	`CarName_saab `	0.041976	0.13637	0.307811	0.758751
carheight	-0.01019	0.008944	-1.13912	0.256885	CarName_subaru	-0.2029	0.129924	-1.56172	0.120945
curbweight	0.000584	8.14E-05	7.17547	6.11E-11	`CarName_toyota `	-0.20838	0.105712	-1.97118	0.050968
boreratio	0.037366	0.108572	0.344156	0.731322	`CarName_volkswagen `	-0.13387	0.124179	-1.07802	0.283151
stroke	0.036392	0.075921	0.479335	0.632558	`CarName_volvo `	-0.02806	0.149582	-0.1876	0.851501
compressionratio	-0.02629	0.031715	-0.82909	0.408672	fueltype_gas	-0.55097	0.443022	-1.24365	0.216012
peakrpm	3.77E-05	4.43E-05	0.849497	0.397269	aspiration_turbo	0.08227	0.047527	1.731027	0.085975
`CarName_audi `	0.131124	0.153128	0.856299	0.393511	carbody_hardtop	-0.11891	0.090287	-1.31699	0.19031
CarName_bmw	0.314643	0.165693	1.898951	0.059932	carbody_hatchback	-0.1242	0.077526	-1.60209	0.111722
`CarName_buick `	0.138618	0.17788	0.779277	0.437326	carbody_sedan	-0.09732	0.077894	-1.24943	0.213899
CarName_chevrolet	-0.19685	0.171196	-1.14986	0.25245	carbody_wagon	-0.16421	0.088228	-1.86117	0.065127
`CarName_dodge `	-0.26892	0.126338	-2.12858	0.0353	drivewheel_rwd	0.038089	0.050299	0.757258	0.450355
`CarName_honda `	0.002349	0.152355	0.015415	0.987726	enginelocation_rear	0.579228	0.243853	2.375315	0.019092
CarName_isuzu	-0.17575	0.141689	-1.2404	0.21721	enginetype_l	-0.22201	0.259261	-0.85631	0.393508
CarName_jaguar	-0.14282	0.160491	-0.88989	0.375278	enginetype_ohc	-0.00318	0.086599	-0.03672	0.97077
`CarName_maxda `	-0.29826	0.142431	-2.09407	0.038324	enginetype_ohcv	-0.09144	0.084655	-1.08013	0.282216
`CarName_mazda `	-0.0112	0.119778	-0.0935	0.925658	enginetype_rotor	-0.06228	0.222902	-0.27939	0.78042
`CarName_mercury cougar`	-0.08508	0.192999	-0.44085	0.660102	cylindernumber_five	-0.2793	0.16273	-1.71635	0.088637
`CarName_mitsubishi `	-0.30021	0.133272	-2.25262	0.026071	cylindernumber_four	-0.27573	0.200942	-1.37219	0.172521
`CarName_Nissan `	-0.14853	0.116888	-1.27067	0.206263	cylindernumber_six	-0.16301	0.163324	-0.99807	0.320219
`CarName_peugeot `	-0.01807	0.280905	-0.06433	0.94881	cylindernumber_twelve	0.141168	0.282121	0.50038	0.617709
`CarName_plymouth `	-0.2836	0.127939	-2.21667	0.028499	fuelsystem_2bbl	0.14883	0.103211	1.441998	0.151865
CarName_porsche	0.070221	0.284374	0.24693	0.805378	fuelsystem_mpfi	0.252515	0.108137	2.335129	0.02117
`CarName_renault `	-0.28116	0.153395	-1.83292	0.069253	fuelsystem_spdi	0.203695	0.131418	1.54997	0.123739
					fuelsystem_spfi	0.13291	0.188555	0.704884	0.482226

Figure 13: Model Output

With Multiple R-square = 0.9572.

Now R-square is quite high, but only 12 out of 49 predictors are significant (from p-value). So, there is something suspicious in the model.The yellow coloured variable denotes significant predictors.

Now I want to see whether the residuals are homoschedastic or not.For that I plot the residuals vs the fitted values of response variable.

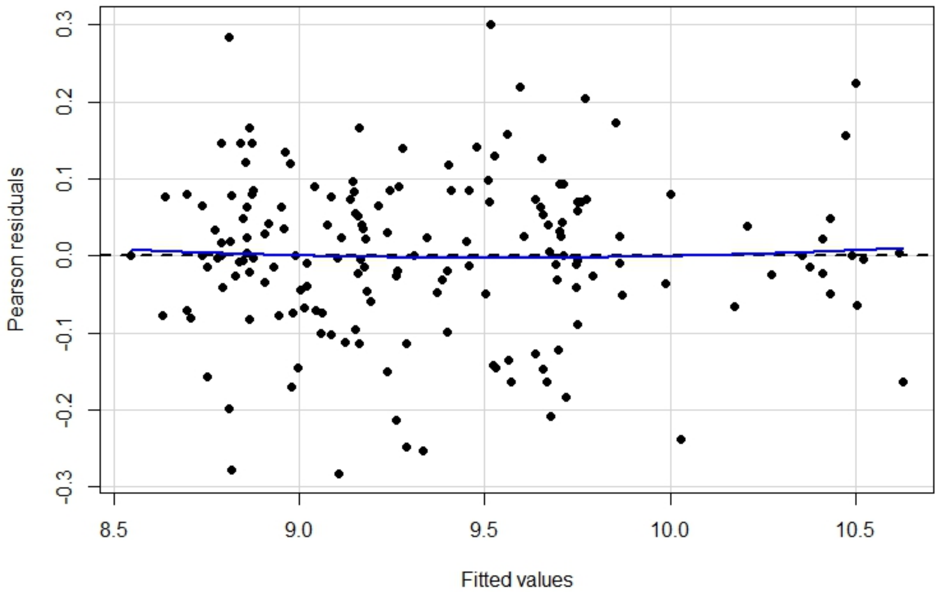

Figure 14: Plot of Residuals vs Fitted values of Response Variable

From the plot, I observe that the residuals are more or less homoschedastic. Now I want to see whether the residuals follow normal dist or not. For that I use qqplot.

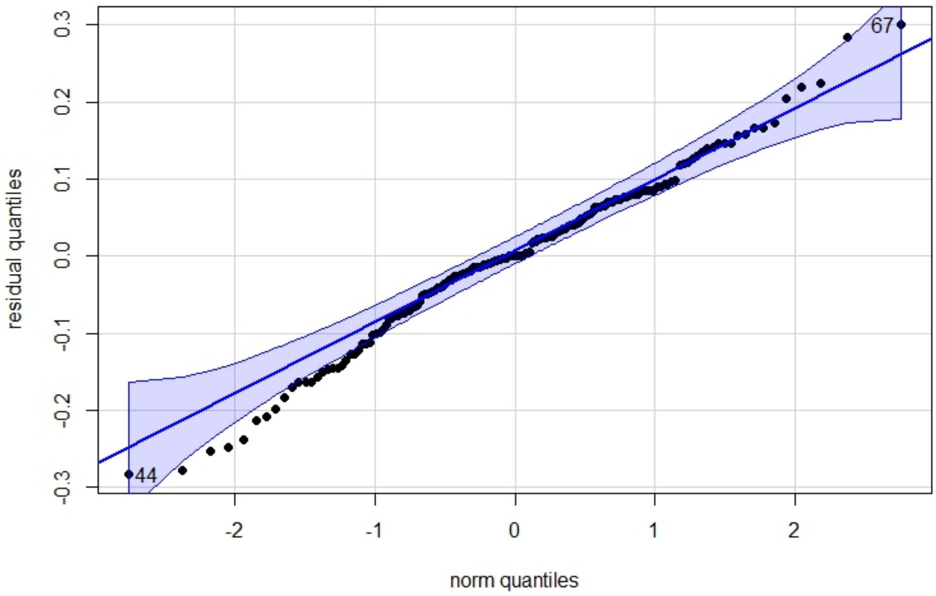

Figure 15: qqplot of residuals

From the plot we can see that residual quantiles does not coincide with actual normal quantiles. Hence,the residuals does not follow normal distribution.

6.3 Model 3: Removing all Insignificant and Correlated Predictors

Here, we want to fit a model removing the correlated and insignificant predictors.

The following table gives us the output of regression model.

Predictors	Estimate	Std.Error	t value	p value
(Intercept)	7.413231187	0.069777	106.2411	7.4E-149
curbweight	0.0007433	2.95E-05	25.16632	3.74E-57
CarName_bmw	0.407531766	0.065122	6.257976	3.51E-09
`CarName_dodge `	-0.108269445	0.060933	-1.77685	0.077516
`CarName_maxda `	-0.186446505	0.109268	-1.70633	0.089912
`CarName_mitsubishi `	-0.119146146	0.053948	-2.20855	0.028646
`CarName_plymouth `	-0.107914255	0.065232	-1.65431	0.10005
`CarName_toyota `	-0.117658551	0.032978	-3.56784	0.000477
carbody_wagon	-0.177997097	0.03602	-4.94157	1.96E-06
enginelocation_rear	0.823962887	0.108788	7.574	2.85E-12
fuelsystem_mpfi	0.126669195	0.03008	4.211028	4.26E-05
`CarName_renault `	-0.155137526	0.110209	-1.40766	0.161196
aspiration_turbo	0.061414298	0.034807	1.764439	0.07959

Figure 16: Model Output

With Multiple R-square = 0.9162.

Now R-square is quite high, and 10 out 12 predictors come to be significant, which implies the model fit is good.The sky coloured variable denotes insignificant predictors.

Now I want to see whether the residuals are homoschedastic or not.For that I plot the residuals vs the fitted values of response variable.

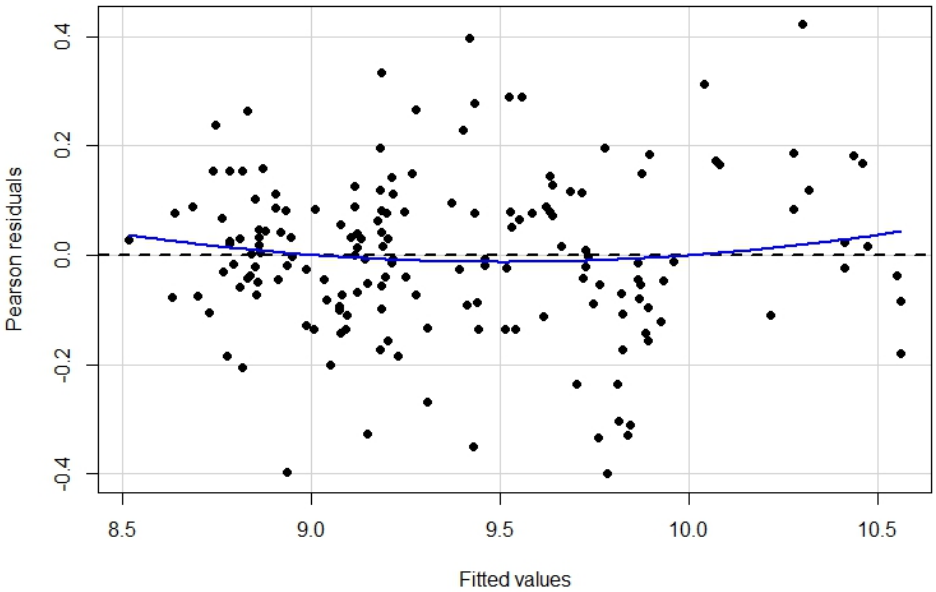

Figure 17: Plot of Residuals vs Fitted values of Response Variable

From the plot, I observe that the residuals are more or less homoschedastic. Now I want to see whether the residuals follow normal dist or not. For that I use qqplot.

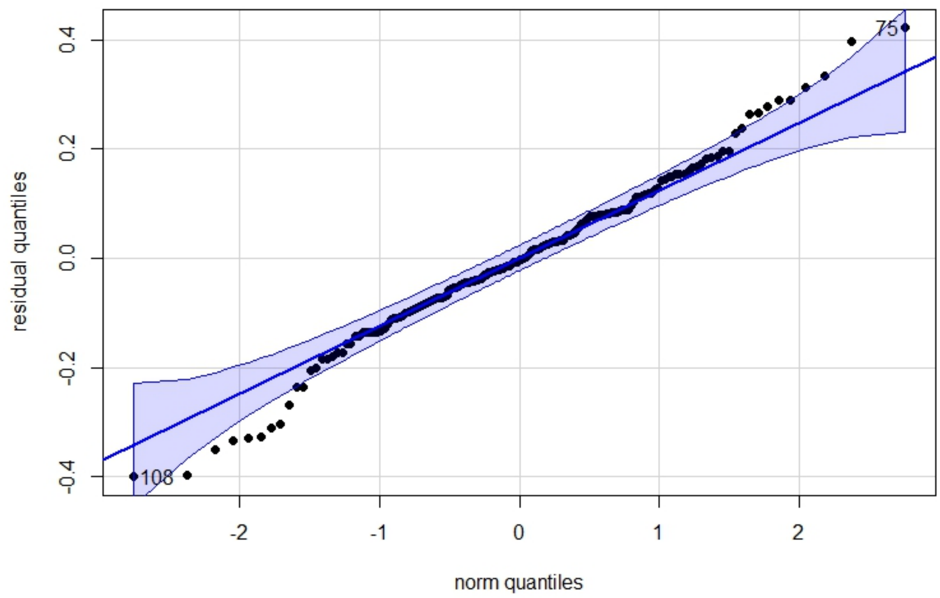

Figure 18: qqplot of residuals

From the plot we can see that residual quantiles does not coincide with actual normal quantiles at the two ends,but it coincides good in the middle portion. So we do Shapiro-Wilk Normality Test to come to a rigid decision.

```
        Shapiro-Wilk normality test

data:  resid(lm(y ~ ., train4))
W = 0.9872, p-value = 0.1218
```

Figure 19: Output of Shapiro-Wilk Normality Test

Here, we can see H0 is accepted and we conclude that residuals follows normal distribution.

6.4 Model-4: Lasso Regression

Now, we will do lasso regression. We mainly use lasso here for variable selection. We take an optimal choice of lambda such that penalty term will be minimized and at the same time the model is good.

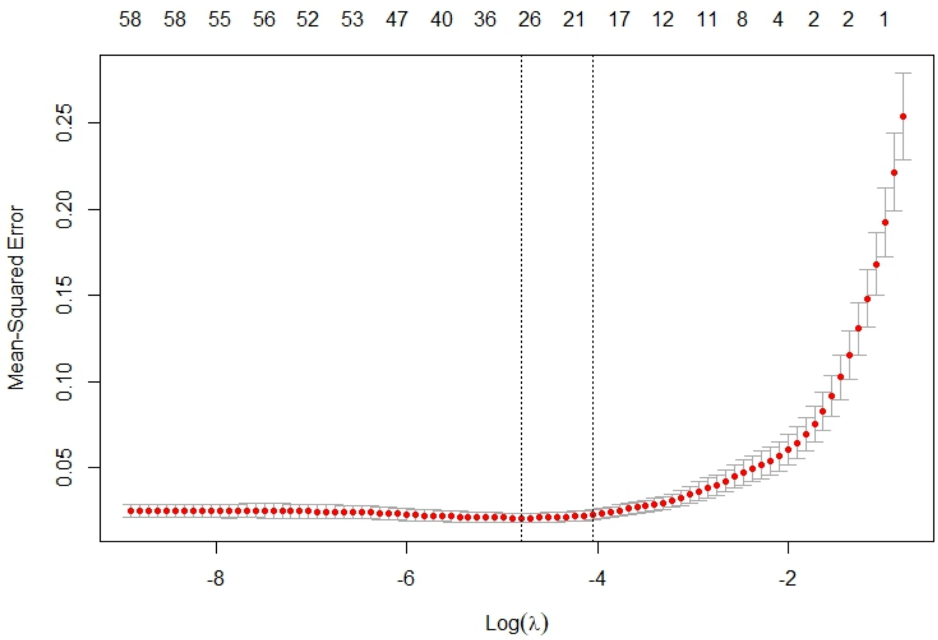

Figure 20: Plotting Mean Square Error vs log(lambda)

Here, we consider the largest value of lambda such that error is within 1 standard error of the minimum and the corresponding number non zero coefficient is 19.

Here, lambda.min = 0.009, and lambda.1se = 0.015.

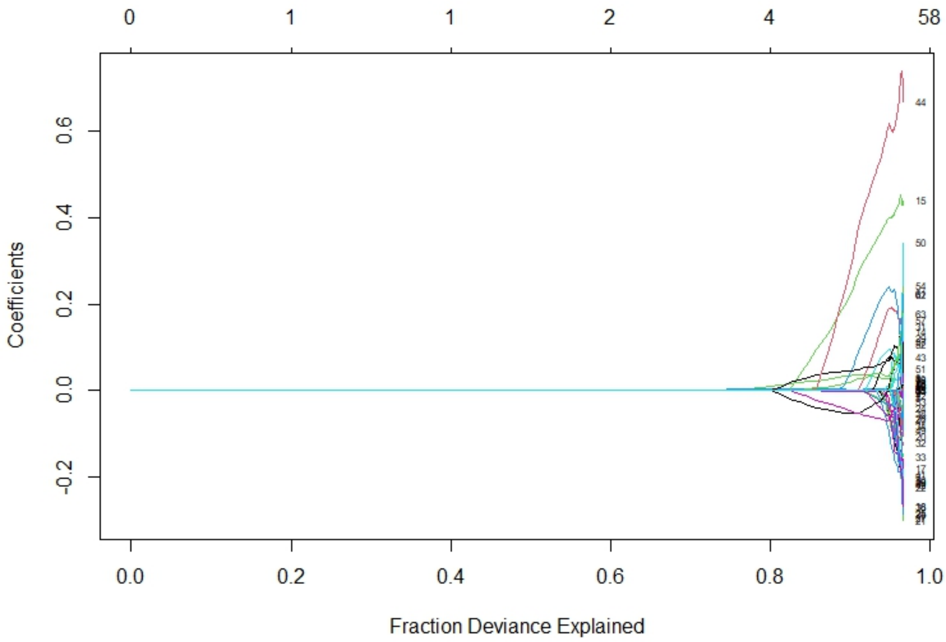

Figure 21: Plotting Coefficients vs Fraction Deviance Explained

From the graph, we can see that just 4 predictors gives the multiple R-square 0.8 and approximately 15 predictors gives us that value 0.9.

Now, we will collect those variable for which the value of coefficient are non zero which is provided by lasso model by taking lambda = 0.015. Here, we want to fit a model by collecting those predictors.

The following table gives us the output of regression model.

Predictor	Estimate	Std.Error	t value	p value
(Intercept)	6.561133	0.602934	10.88201	9.36E-21
carwidth	0.019365	0.010683	1.81269	0.071853
curbweight	0.000442	7.04E-05	6.278195	3.41E-09
enginesize	0.000208	0.000646	0.322823	0.747273
horsepower	0.002462	0.000644	3.826441	0.000189
citympg	0.001792	0.003081	0.581466	0.561788
`CarName_audi `	0.268456	0.064705	4.148925	5.54E-05
CarName_bmw	0.438541	0.056914	7.705389	1.57E-12
`CarName_buick `	0.309213	0.067623	4.572589	9.92E-06
`CarName_mazda `	0.164375	0.042838	3.837144	0.000182
`CarName_saab `	0.151588	0.065212	2.324543	0.02142
`CarName_toyota `	-0.04335	0.028633	-1.51415	0.132065
carbody_hatchback	-0.05256	0.024197	-2.17203	0.031404
carbody_wagon	-0.08197	0.033609	-2.4388	0.015888
drivewheel_rwd	0.082946	0.032508	2.551602	0.01171
enginelocation_rear	0.660289	0.101782	6.487288	1.16E-09
cylindernumber_four	-0.00469	0.036061	-0.12998	0.896754
fuelsystem_2bbl	-0.07848	0.030568	-2.56724	0.011216
fuelsystem_mpfi	0.04996	0.033339	1.498553	0.136064

Figure 22: Model Output

With Multiple R-square = 0.9463.

Now I want to see whether the residuals are homoschedastic or not.For that I plot the residuals vs the fitted values of response variable.

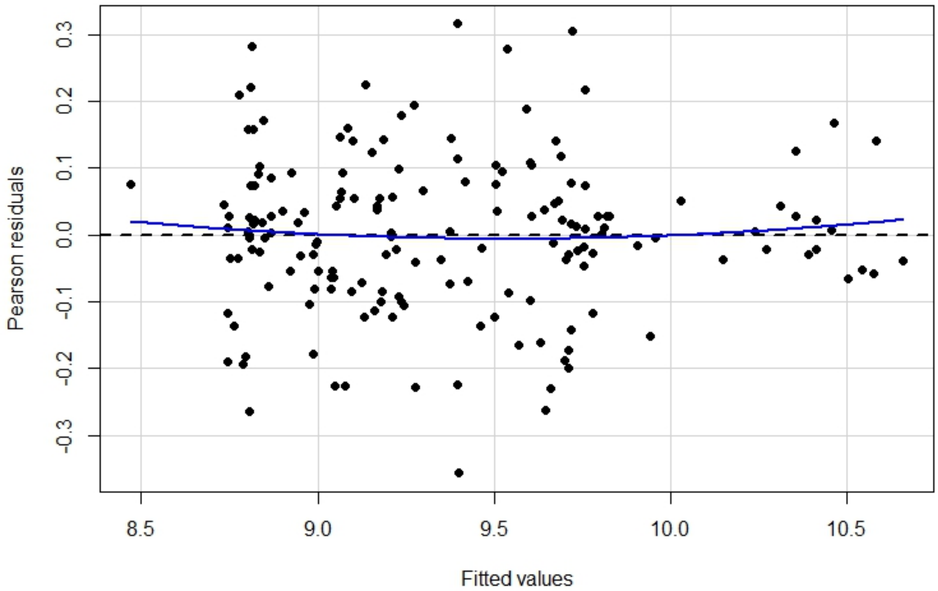

Figure 23: Plot of Residuals vs Fitted values of Response Variable

From the plot, I observe that the residuals are more or less homoschedastic. Now I want to see whether the residuals follow normal dist or not. For that I use qqplot.

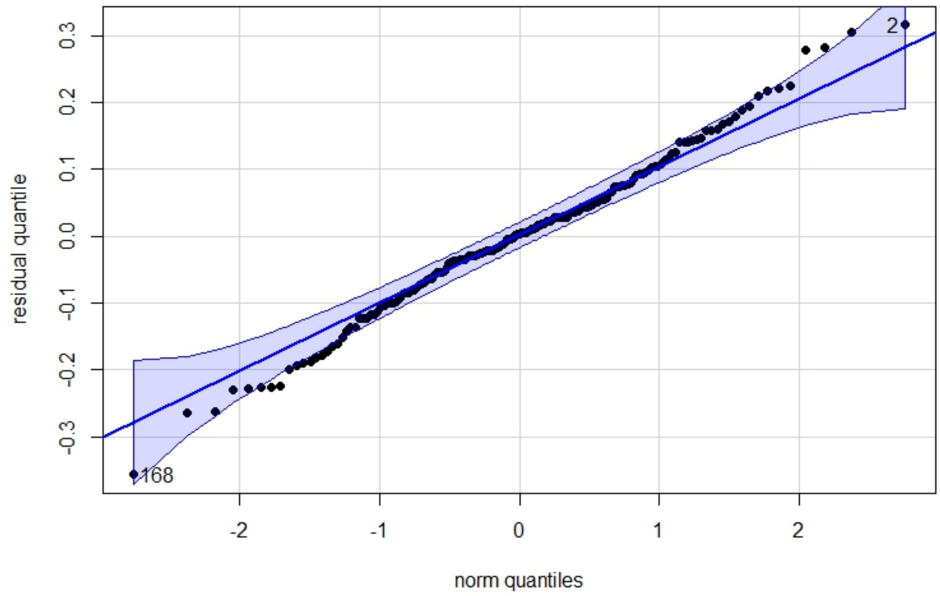

Figure 24: qqplot of residuals

From the plot we can see that residual quantiles coincides more or less with actual normal quantiles.

Now we do Shapiro-Wilk Normality Test to come to a rigid decision.

```
> shapiro.test(resid(s5))

        Shapiro-Wilk normality test

data:  resid(s5)
W = 0.99239, p-value = 0.5087

>
```

Figure 25: Output of Shapiro-Wilk Normality Test

Here, we can see H0 is accepted and we conclude that residuals follows normal distribution.

7 Model Comaparison

We saw that among the first three models model-3 was best. Now we add another model (Model-4) by lasso. Now , we want to compare these two models using ”PRESS” statistic.

The expression of PRESS Statistic is, $PRESS = \sum_{i=1}^{n}(e_i - e_{-i})^2$

For Model-3, PRESS = 4.011418.

For Model-4, PRESS = 2.859928.

Hence, we can see that w.r.t PRESS, Model-4 is better than Model-3.

8 Prediction

8.1 Prediction by Multiple Linear Regression

In this section, we predict the response variable of test set by Model-3.

The results is shown in the following diagram,

Actual log (price)	Fitted log (price)	Actual Price	Fitted Price	Actual log (price)	Fitted log (price)	Actual Price	Fitted Price
9.781884731	9.653845268	17710	15581	9.099297073	9.140660131	8949	9326
9.957265258	10.00265634	21105	22085	9.487972109	9.850975232	13200	18976
10.33397042	10.34829079	30760	31203	9.722864552	9.803643423	16695	18099
8.747510946	8.806175181	6295	6675	8.937087036	8.933886991	7609	7584
8.760453046	8.699392335	6377	5999	9.99961558	9.604787475	22018	14835
9.469931564	9.45579203	12964	12781	10.51942966	10.44510296	37028	34375
8.894944461	8.907263965	7295	7385	9.649240256	9.589921477	15510	14616
9.087607607	9.125794133	8845	9189	9.328923088	9.466997402	11259	12926
9.095658772	8.806175181	8916.5	6675	8.584477938	8.771022917	5348	6444
8.823942327	8.829217477	6795	6830	8.974364842	8.986579885	7898	7995
9.657906656	9.398150106	15645	12066	9.133243322	8.8862344	9258	7231
9.286838343	9.229112818	10795	10189	9.209239767	9.318399849	9989	11141
10.14847087	10.0873446	25552	24036	9.296334565	9.216567764	10898	10062
8.592115118	8.719734229	5389	6122	8.984066928	9.181849838	7975	9719
8.730528802	8.739060026	6189	6242	9.535679436	9.3916159	13845	11987
9.047703788	9.024487184	8499	8303	9.849559211	9.769915334	18950	17499
8.902319529	8.739673067	7349	6245	10.01993637	9.86584123	22470	19261

Figure 26: Actual Values and Fitted Values of Response variable using Model-3

8.2 Prediction by Lasso

In this section, we predict the response variable of test set by Model-4.

The results is shown in the following diagram,

Actual log (price).	Fitted log (price)	Actual Price	Fitted Price	Actual log (price).	Fitted log (price)	Actual Price	Fitted Price
9.781884731	9.853110928	17710	19017	9.099297073	9.027951198	8949	8332
9.957265258	9.979903541	21105	21588	9.487972109	9.69347826	13200	16211
10.33397042	10.37682322	30760	32106	9.722864552	9.677590336	16695	15956
8.747510946	8.744945028	6295	6278	8.937087036	8.925753953	7609	7523
8.760453046	8.73223667	6377	6199	9.99961558	9.605525568	22018	14846
9.469931564	9.454447953	12964	12764	10.51942966	10.43185459	37028	33923
8.894944461	8.94472153	7295	7667	9.649240256	9.599091944	15510	14751
9.087607607	9.120968606	8845	9145	9.328923088	9.321663242	11259	11177
9.095658772	8.797501359	8916.5	6617	8.584477938	8.726018282	5348	6161
8.823942327	8.917390608	6795	7460	8.974364842	8.987967762	7898	8006
9.657906656	9.56143112	15645	14206	9.133243322	8.871016894	9258	7122
9.286838343	9.336438286	10795	11343	9.209239767	9.35093645	9989	11509
10.14847087	10.24945246	25552	28267	9.296334565	9.218667759	10898	10083
8.592115118	8.773594739	5389	6461	8.984066928	9.1320585	7975	9247
8.730528802	8.774342477	6189	6466	9.535679436	9.239211213	13845	10292
9.047703788	9.008265356	8499	8170	9.849559211	9.76123188	18950	17347
8.902319529	8.772193952	7349	6452	10.01993637	9.738735969	22470	16962

Figure 27: Actual Values and Fitted Values of Response variable using Model-4

8.3 Comparison of Prediction

Now we compute Residual Sum of Square (RSS) for the two models.

The expression of RSS is, $RSS = \sum_{i=1}^{n}(y_i - \widehat{y_i})^2$

For Model-3, RSS $= 0.757578$

For Model-4, RSS $= 0.7118$

Hence, we can see that RSS Model-4 is slightly higher than that of Model-3.

9 Conclusion

I start the EDA by checking whether there is any missing data or not. We saw that there is no missing data points in the dataset. Then we individually analyse each of the numerical variable. We compute their average, variability and skewness. Then we saw whether there is any outlier in each variable. We saw that by boxplot. But we do not remove any point as the number of data points is less in the dataset. Then we check for presence of dependence of car price on each variable individually. After that I have checked for presence of multicollinearity in the numerical predictors. And there is multicollinearity among some variables. Now, I want to predict some car price. For that first I do dummification of categorical variables and then I break my dataset into train and test set. Now as car price based on train set is positively skewed, I do log transformation and consider that as my response variable. Now I fit several models and want to choose the best one. In the Model-1, I consider all the variables as predictors. Then in Model-2, I remove the correlated variables and refit the model. In Model-3, I remove all insignificant predictors (obtained from Model-2) and the correlated variables and refit the model. Among these three models, Model-3 is an optimized model. In Model-4 we took those predictors which are suggested by lasso. Now we saw that Model-4 is better than Model-3 w.r.t PRESS statistic. And finally we saw that Model-4 predicts test set slightly better than Model-3. Hence, we can conclude Model-4 is slightly better representation than Model-3.

10 Acknowledgement

I would like to express my gratitude to Dr. Deepayan Sarkar for giving me the opportunity to work on this project. I would like to thank my family and my friends as well. They kept me motivated to work constantly on my project. They also helped me to understand some of the concepts and helped me to write some codes. My work was made easier by my family and my friends.

11 Appendix

Listing 1: R code used for the analysis and model fitting

```
rm(list=ls())
setwd("D:/ISI/Deepayan sir")
C=read.csv(file="CarPricedata.csv")
attach(C)
E=C[,-c(1,2,26)]
y=log(price)
Z=cbind(E,price,y)
D=cbind(E,y)
attach(D)
View(D)
attach(Z)
View(Z)

#EDA

# Missing Values Check
any(is.na(C))

# Checking Outliers
par(mfrow=c(4,4))
boxplot(C[,10],main="wheelbase",col="skyblue")
```

```
boxplot(C[,11],main="carlength",col="skyblue")
boxplot(C[,12],main="carwidth",col="skyblue")
boxplot(C[,13],main="carheight",col="skyblue")
boxplot(C[,14],main="curbweight",col="skyblue")
boxplot(C[,17],main="enginesize",col="skyblue")
boxplot(C[,19],main="boreratio",col="skyblue")
boxplot(C[,20],main="stroke",col="skyblue")
boxplot(C[,21],main="compressionratio",col="skyblue")
boxplot(C[,22],main="horsepower",col="skyblue")
boxplot(C[,23],main="peakrpm",col="skyblue")
boxplot(C[,24],main="citympg",col="skyblue")
boxplot(C[,25],main="highwaympg",col="skyblue")
boxplot(C[,26],main="car price",col="skyblue")
par(mfrow=c(1,1))

# Checking collinearity
library(dplyr)
library(reshape)
library(ggplot2)
corr = data.matrix(cor(D[sapply(D,is.numeric)]))
mel = melt(corr)
ggplot(mel, aes(X1,X2))+geom_tile(aes(fill=value)) +
  geom_text(aes(label = round(value, 1)))+
  scale_fill_gradient2(low='yellow',mid = 'white' ,high=
     'green')

```

```
# Observing the relation of continuous variables with
   car price using scatterplot
par(mfrow=c(3,2))
plot(wheelbase,price,xlab="wheelbase",ylab="car price",
   pch=19,col="blue")
plot(carlength,price,xlab="carlength",ylab="car price",
   pch=19,col="blue")
plot(carwidth,price,xlab="carwidth",ylab="car price",pch
   =19,col="blue")
plot(carheight,price,xlab="carheight",ylab="car price",
   pch=19,col="blue")
plot(curbweight,price,xlab="curbweight",ylab="car price"
   ,pch=19,col="blue")
plot(enginesize,price,xlab="enginesize",ylab="car price"
   ,pch=19,col="blue")
par(mfrow=c(1,1))
par(mfrow=c(3,3))
plot(boreratio,price,xlab="boreratio",ylab="car price",
   pch=19,col="blue")
plot(stroke,price,xlab="stroke",ylab="car price",pch=19,
   col="blue")
plot(compressionratio,price,xlab="compressionratio",ylab
   ="car price",pch=19,col="blue")
plot(horsepower,price,xlab="horsepower",ylab="car price"
   ,pch=19,col="blue")
plot(peakrpm,price,xlab="peakrpm",ylab="car price",pch
```

```
=19,col="blue")
plot(citympg,price,xlab="citympg",ylab="car price",pch
=19,col="blue")
plot(highwaympg,price,xlab="highwaympg",ylab="car price"
,pch=19,col="blue")
par(mfrow=c(1,1))

# Observing the relation of categorical variables with
car price using boxplot
par(mfrow=c(3,3))
boxplot(price~factor(fuelsystem),data=C,col="yellow")
boxplot(price~factor(enginelocation),data=C,col="yellow"
)
boxplot(price~factor(drivewheel),data=C,col="yellow")
boxplot(price~factor(carbody),data=C,col="yellow")
boxplot(price~factor(fueltype),data=C,col="yellow")
boxplot(price~factor(doornumber),data=C,col="yellow")
boxplot(price~factor(aspiration),data=C,col="yellow")
boxplot(price~factor(CarName),data=C,col="yellow")
boxplot(price~factor(cylindernumber),data=C,col="yellow"
)
par(mfrow=c(1,1))

# Observing the skewness of variables
N=D[sapply(D,is.numeric)]
View(N)
```

```
ncol(N)
col.names=colnames(N)
col.names
par(mfrow=c(3,5))
for(i in 1:14)
{
  hist(N[,i],main=paste("Histogram of",col.names[i]),freq = F,xlab=col.names[i],
       col="green")
}

par(mfrow=c(1,1))

# Reporting values
library(moments)
R=matrix(0,nrow=14,ncol=3)
R[,1]=as.vector(apply(N,2,median))
R[,2]=as.vector(apply(N,2,IQR))
R[,3]=as.vector(apply(N,2,skewness))
colnames(R)=c("Median","IQR","Skewness")
rownames(R)=c("wheelbase","carlength","carwidth","carheight","curbweight","enginesize",
              "boreratio","stroke","compressionratio","horsepower","peakrpm","citympg",
              "highwaympg","price")
R
```

```
# Data Preparation
library(fastDummies)
data=dummy_cols(D,select_columns = c("CarName","fueltype
   ","aspiration","doornumber",
                                     "carbody","
                                        drivewheel","
                                        enginelocation",
                                        "enginetype",
                                     "cylindernumber","
                                        fuelsystem"),
                remove_first_dummy = T,
                  remove_selected_columns = T)
View(data)

# Breaking the dataset into train and test set
library(caTools)
set.seed(seed=2207)
sample=sample.split(data[,1],SplitRatio = 0.8)
train=subset(data,sample==T)
test=subset(data,sample==F)


hist(subset(C$price,sample==T),freq=F,col="green",main="
   Histogram of carprice of train set"
     ,xlab="car price")
hist(log(subset(C$price,sample==T)),freq=F,col="green",
```

```
    main="Histogram of carprice of train set"
      ,xlab="car price")
# Model fiiting

#Model-1
train1=train
View(train1)
s1=summary(lm(y~.,train1))
s1
library(readxl)
coeff.pval=data.frame(s1$coefficients)
t1=cbind(rownames(coeff.pval),coeff.pval)
writexl::write_xlsx(t1,'D:/ISI/Deepayan sir/t1.xlsx')
nrow(s1$coefficients)
library(car)
residualPlot(lm(train1$y~.,train1[-14]),pch=19)
shapiro.test(resid(lm(y~.,train1)))
ncvTest(lm(y~.,train1))
qqPlot(resid(lm(y~.,train1)),ylab="residual quantiles",
    main="Q-Q plot of Residuals")

#Model-2
train3=subset(train,select=-c(wheelbase,carlength,
    carwidth,horsepower,citympg,highwaympg,
                                    enginesize,doornumber_two)
                                        )
```

```
attach(train3)
s3=summary(lm(y~.,train3))
s3
library(readxl)
coeff.pval=data.frame(s3$coefficients)
t3=cbind(rownames(coeff.pval),coeff.pval)
writexl::write_xlsx(t3,'D:/ISI/Deepayan sir/t3.xlsx')
nrow(s3$coefficients)
library(car)
residualPlot(lm(y~.,train3),pch=19,main="")
shapiro.test(resid(lm(y~.,train3)))
ncvTest(lm(y~.,train3))
qqPlot(resid(lm(y~.,train3)),ylab="residual quantiles",
   pch=19)

#Model-3
train4=subset(train3,select=c(y,curbweight,CarName_bmw,‘
   CarName_dodge‘,‘CarName_maxda‘,
                              ‘CarName_mitsubishi‘,‘
                                 CarName_plymouth‘,‘
                                 CarName_toyota‘,
                              carbody_wagon,
                                 enginelocation_rear,
                                 fuelsystem_mpfi,
                              ‘CarName_renault‘,
                                 aspiration_turbo))
```

```r
attach(train4)
s4=lm(y~.,train4)
summary(s4)
residualPlot(lm(y~.,train4),
             main="Residual Plot vs Fitted Values of Response Variable",pch=19)
library(readxl)
coeff.pval=data.frame(summary(s4)$coefficients)
t4=cbind(rownames(coeff.pval),coeff.pval)
writexl::write_xlsx(t4,'D:/ISI/Deepayan sir/t4.xlsx')
nrow(s4$coefficients)
library(car)
residualPlot(lm(y~.,train4),pch=19)
shapiro.test(resid(lm(y~.,train4)))
ncvTest(lm(y~.,train4))
qqPlot(resid(lm(y~.,train4)),ylab="residual quantiles",pch=19,main="Q-Q Plot of Residuals")

#Prediction-------------
P=matrix(0,nrow=nrow(test),ncol=4)
P[,2]=predict(s4,newdata=test)
P[,1]=test$y
P[,3]=subset(C$price,sample==F)
P[,4]=floor(exp(P[,2]))
colnames(P)=c("Actual log(price)","Fitted log(price)","Actual Price","Fitted Price")
```

```
P
library(readxl)
writexl::write_xlsx(data.frame(P),'D:/ISI/Deepayan sir/P
    .xlsx')
SSE1=sum((P[,1]-P[,2])^2)
SSE1

#PRESS--------------
h1=hatvalues(s4)
e1=resid(s4)
PRESS1=sum((e1/(1-h1))^2)
PRESS1

#Lasso----------------
library(lattice)
library(glmnet)
str(train)
y1= train$y
X= model.matrix( ~ . - y - 1,train)
fm.lasso=cv.glmnet(X, y1, alpha = 1)
s.cv <- c(lambda.min = fm.lasso$lambda.min, lambda.1se =
     fm.lasso$lambda.1se)
s.cv
round(coef(fm.lasso, s = s.cv), 3)
#Choosing lambda
cv.lasso <- cv.glmnet(X, y1, alpha = 1, nfolds = 50)
```

```
plot(cv.lasso)

f1=glmnet(X, y1, alpha = 1)
plot(f1, xvar = "dev", label = TRUE)
#Fitting Model from Lasso---------
rownames(coef(fm.lasso, s = 'lambda.1se'))[coef(fm.lasso
   , s = 'lambda.1se')[,1]!= 0]
s5=lm(y~carwidth+curbweight+enginesize+horsepower+
   citympg+`CarName_audi `+CarName_bmw+
        `CarName_buick `+`CarName_mazda `+`CarName_saab
           `+`CarName_toyota `+
        carbody_hatchback+carbody_wagon+drivewheel_rwd+
           enginelocation_rear+
        cylindernumber_four+
      fuelsystem_2bbl+fuelsystem_mpfi,data=train)
summary(s5)
library(readxl)
coeff.pval=data.frame(summary(s5)$coefficients)
t5=cbind(rownames(coeff.pval),coeff.pval)
writexl::write_xlsx(t5,'D:/ISI/Deepayan sir/t5.xlsx')
h2=hatvalues(s5)
e2=resid(s5)
PRESS2=sum((e2/(1-h2))^2)
PRESS2

#Prediction---------------
```

```
Q=matrix(0,nrow=nrow(test),ncol=4)
Q[,1]=test$y
Q[,2]=predict(s5,newdata = test)
Q[,3]=subset(C$price,sample==F)
Q[,4]=floor(exp(Q[,2]))
colnames(Q)=c("Actual log(price)","Fitted log(price)","
   Actual Price","Fitted Price")
Q
writexl::write_xlsx(data.frame(Q),'D:/ISI/Deepayan sir/Q
   .xlsx')
SSE2=sum((Q[,1]-Q[,2])^2)
SSE2
residualPlot(s5,pch=19)
qqPlot(resid(s5),ylab="residual quantile",pch=19)
shapiro.test(resid(s5))
ncvTest(s5)
```