GENERAL MOTORS COMPANY

Project Title: Just Another Random Car

Team Member:

Ruoxi Wang

Li-An Fan Jiang

Cong Minh Tran

Non-technical Summary Report

When people want to purchase the car, most people put price as their most important factor for considering. However, there are many other factors can also effect people to make buying decision. Different people care about different things. What the car being used for? For individual, business, family; What is the purpose of buying the car? As the gifts to give others, transport tools or car fans who want to make the collection; Does the car's configuration really matter, such as cruise leather or sound; How to set the price based on the mileage for used car and so on. There are really a lot of factors people should be considering, when they plan to purchase the car. Therefore, how to choose the car with reasonable price and good performance becomes important for people who want to purchase the car.

The goal of this project is to create a regression model which can predict the price of a GM car based on the predictors provided, so that could help people to make a best decision on choosing high cost-performance car.

This project used multi-variables regression analysis to build the model which mean use the given indicators to predict the price of GM cars. In this method we use both qualitative and quantitative variables assuming that there is a linear relationship between the price and those variables.

Technical Summary Report

Abstract

Our goal is to create a regression model which can predict the price of a GM car based on the predictors provided.

Our methodology: Multi-variable and reclassify variables regression analysis two methods

Our findings and recommendations:

After finishing the mode to predict the price of GM car based on given variables, we've found that the 3 most important indicators are number of cylinder-liter, type of car, car brand,….

Recommendations

Introduction

There are 804 observations in GM_Cars dataset and the original dataset provide the following 12 variables:

Dataset: GM car value dataset contains over eight hundred records about 2005 used GM cars. Each record shows variety of characteristics such as mileage, make, model, engine size, interior style, and cruise control. The main purpose is predicting the retail price of car based on other predictors which are provided in the dataset.

- **Dependent Variable:**

  Price: suggested retail price of the used 2005 GM car in excellent condition. The condition of a car can greatly affect price. All cars in this data set were less than one year old when priced and considered to be in excellent condition.

- **Independent Variables:**

  Two quantitative: Mileage, Liter;

  Nine qualitative: Make, Model, Trim, Type, Cylinder, Doors, Cruise, Sound, Leather

  - Mileage: number of miles the car has been driven

- Make: manufacturer of the car such as Saturn, Pontiac, and Chevrolet
- Model: specific models for each car manufacturer such as Ion, Vibe, Cavalier
- Trim (of car): specific type of car model such as SE Sedan 4D, Quad Coupe 2D
- Type: body type such as sedan, coupe, etc
- Cylinder: number of cylinders in the engine
- Liter: a more specific measure of engine size
- Doors: number of doors
- Cruise: indicator variable representing whether the car has cruise control (1=cruise)
- Sound: indicator variable representing whether the car has upgraded speakers (1=upgraded)
- Leather: indicator variable representing whether the car has leather seats (1=leather)

Methodology

This project use multi-variable and reclassify variables regression analysis two methods

Step 1: Data collecting

GM_Cars dataset collected from Kelly Blue Book for several hundred 2005 used GM cars; For this data set, a representative sample of over eight hundred, 2005 GM cars were selected, then an algorithm was developed following the 2005 Central Edition of the Kelly Blue Book to estimate retail price.

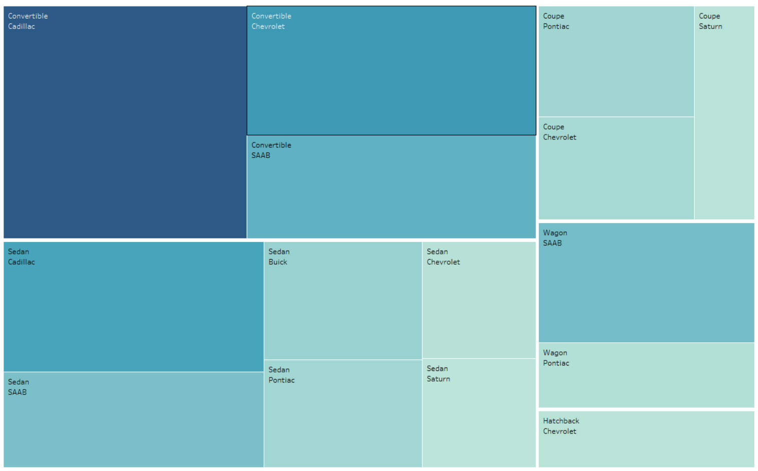
Step2: Data exploration

- We explored the data mainly by SAS Procedures, we also did use Tableau to visualize the data.

There are some patterns from tableau and these pattern are similar to the procedures we created in SAS

	Make					
Type	Buick	Cadillac	Chevrol..	Pontiac	SAAB	Saturn
Convertible		Abc	Abc		Abc	
Coupe			Abc	Abc		Abc
Hatchback			Abc			
Sedan	Abc	Abc	Abc	Abc	Abc	Abc
Wagon				Abc	Abc	

The average price of car based on type



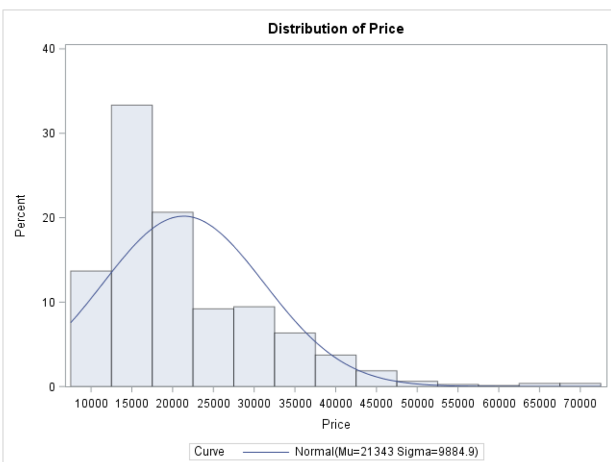
- This dataset split the data by 80% for training set and 20% for testing set.
- For both methods, this project delete one variable ‘Trim’. Due to ‘Trim’ include 47 variables, so we consider to delete it. Then we search the description of the ‘Trim’ to help us make the decision. According to the Wikipedia, the ‘Trim’ means “a model may be offered in varying *trim levels*, which denotes different configurations of standard equipment and amenities. For instance, the base trim may have only basic features (wheel covers, cloth seats) compared to the top-of-the-line model (alloy wheels, leather upholstery)”. Therefore, we think ‘Trim’ is not an important factor for the car, then we decided to delete this variable, so currently we have 11 variables in GM_Cars dataset.

In Leanne and minh’s method, we also delete the ‘Model’ variable. But in Ruoxi’s model not.

- Creating following dummy variables:
  - Five dummy variables for ‘Make’;
  - Four dummy variables for ‘Type’;
  - Two dummy variable for ‘Cylinder’;

- One dummy variable for 'Doors';
- In Ruoxi's method, we re-classify the 'Model' variable, due to there are 32 terms in this variable, which is a lot. So we set three levels combine with the price, which are 'Economy', 'Standard' and 'Luxury'. The cars under \$15,000 which are economy cars; the price between \$15,000-\$27,000 belongs to standard level; the car price higher than \$27,000 are luxury cars.

- Transformation: After exploring the data by boxplot, frequency table and histogram for descriptive statistics, the output shows that the price are not normally distributed, which has a long tail and right skewness, so we decided to apply log transformation and create a new dependent variable log(price), in order to make the distribution normally.



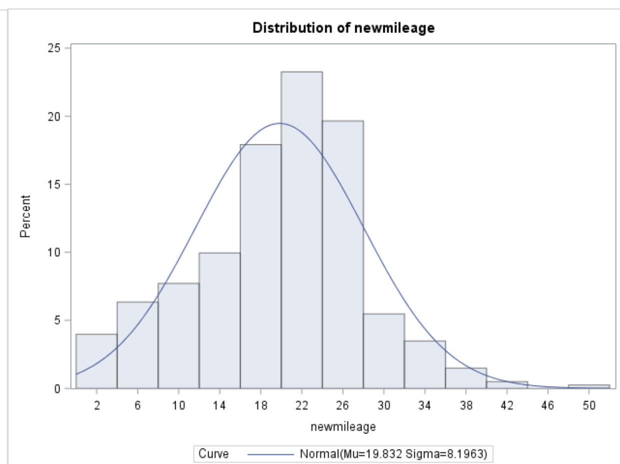


Figure 1 Figure2

- Interaction Terms and center method: Based on the output (Figure3), it showed that there is a strong correlation between Cylinder and Liter (Figure4), so we create an interaction term cylinder_liter. And we also use the center method to solve the multicollinearity issue between interaction term and main terms (Figure5).

Pearson Correlation Coefficients
Prob > |r| under H0: Rho=0
Number of Observations

	Selected	price	mileage	cylinder	liter	doors	cruise	sound	leather	train_price	ln_price
Selected Selection Indicator	1.00000 804	0.03502 0.3214 804	-0.02743 0.4373 804	0.01569 0.6568 804	0.01035 0.7695 804	0.03805 0.2812 804	0.07506 0.0333 804	-0.06235 0.0772 804	-0.02912 0.4096 804	. . 644	. . 644
price	0.03502 0.3214 804	1.00000 804	-0.14305 <.0001 804	0.56909 <.0001 804	0.55815 <.0001 804	-0.13875 <.0001 804	0.43085 <.0001 804	-0.12435 0.0004 804	0.15720 <.0001 804	1.00000 <.0001 644	0.96802 <.0001 644
mileage	-0.02743 0.4373 804	-0.14305 <.0001 804	1.00000 804	-0.02946 0.4041 804	-0.01864 0.5977 804	-0.01694 0.6314 804	0.02504 0.4784 804	-0.02615 0.4591 804	0.00101 0.9773 804	-0.17744 <.0001 644	-0.17964 <.0001 644
cylinder	0.01569 0.6568 804	0.56909 <.0001 804	-0.02946 0.4041 804	1.00000 804	0.95790 <.0001 804	0.00221 0.9502 804	0.35428 <.0001 804	-0.08970 0.0109 804	0.07552 0.0323 804	0.56369 <.0001 644	0.57720 <.0001 644
liter	0.01035 0.7695 804	0.55815 <.0001 804	-0.01864 0.5977 804	0.95790 <.0001 804	1.00000 804	-0.07926 0.0246 804	0.37751 <.0001 804	-0.06553 0.0633 804	0.08733 0.0132 804	0.55459 <.0001 644	0.58666 <.0001 644
doors	0.03805 0.2812 804	-0.13875 <.0001 804	-0.01694 0.6314 804	0.00221 0.9502 804	-0.07926 0.0246 804	1.00000 804	-0.04767 0.1769 804	-0.06253 0.0764 804	-0.06197 0.0791 804	-0.14810 0.0002 644	-0.10297 0.0089 644
cruise	0.07506 0.0333 804	0.43085 <.0001 804	0.02504 0.4784 804	0.35428 <.0001 804	0.37751 <.0001 804	-0.04767 0.1769 804	1.00000 804	-0.09173 0.0093 804	-0.07057 0.0454 804	0.42215 <.0001 644	0.48835 <.0001 644
sound	-0.06235 0.0772 804	-0.12435 0.0004 804	-0.02615 0.4591 804	-0.08970 0.0109 804	-0.06553 0.0633 804	-0.06253 0.0764 804	-0.09173 0.0093 804	1.00000 804	0.16544 <.0001 804	-0.10383 0.0084 644	-0.11693 0.0030 644
leather	-0.02912 0.4096 804	0.15720 <.0001 804	0.00101 0.9773 804	0.07552 0.0323 804	0.08733 0.0132 804	-0.06197 0.0791 804	-0.07057 0.0454 804	0.16544 <.0001 804	1.00000 804	0.16695 <.0001 644	0.14126 0.0003 644
train_price	. . 644	1.00000 <.0001 644	-0.17744 <.0001 644	0.56369 <.0001 644	0.55459 <.0001 644	-0.14810 0.0002 644	0.42215 <.0001 644	-0.10383 0.0084 644	0.16695 <.0001 644	1.00000 644	0.96802 <.0001 644
ln_price	. . 644	0.96802 <.0001 644	-0.17964 <.0001 644	0.57720 <.0001 644	0.58666 <.0001 644	-0.10297 0.0089 644	0.48835 <.0001 644	-0.11693 0.0030 644	0.14126 0.0003 644	0.96802 <.0001 644	1.00000 644

The SAS System

The CORR Procedure

3 Variables:	Cylinder Liter Cylinder_Liter

Simple Statistics

Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Cylinder	804	5.26866	1.38753	4236	4.00000	8.00000
Liter	804	3.03731	1.10556	2442	1.60000	6.00000
Cylinder_Liter	804	17.47015	10.98335	14046	6.40000	48.00000

Pearson Correlation Coefficients, N = 804
Prob > |r| under H0: Rho=0

	Cylinder	Liter	Cylinder_Liter
Cylinder	1.00000	0.95790 <.0001	0.97474 <.0001
Liter	0.95790 <.0001	1.00000	0.98542 <.0001
Cylinder_Liter	0.97474 <.0001	0.98542 <.0001	1.00000

Figure4

The SAS System

The CORR Procedure

3 Variables:	Cylinder Liter Cylinder_Liter_c

Simple Statistics

Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Cylinder	804	5.26866	1.38753	4236	4.00000	8.00000
Liter	804	3.03731	1.10556	2442	1.60000	6.00000
Cylinder_Liter_c	804	1.46775	1.75781	1180	-0.17029	8.16966

Pearson Correlation Coefficients, N = 804
Prob > |r| under H0: Rho=0

	Cylinder	Liter	Cylinder_Liter_c
Cylinder	1.00000	0.95790 <.0001	0.53686 <.0001
Liter	0.95790 <.0001	1.00000	0.56512 <.0001
Cylinder_Liter_c	0.53686 <.0001	0.56512 <.0001	1.00000

Figure5

Step3: Build the models

This project build three models. We used all variables to fit the first model to research what are the problems, then remove the infects and finally improve the model's performance.

Step4: Test models

After the build the models, we checked the 4 assumptions, parameters, R^2, AdjR^2 and GOF, remove collinearity, remove outliers, use selection methods to improve model's performance. After came up with the final model, we test the model by the testing set and draw the conclusion.

Analysis, Results and Findings

For both two methods, we split the data by 80% as training set, and 20% as test set. In addition, all three models do the transformation and interaction term, which we mentioned in step two, therefore, for the following model analysis, we **will not** mention again for this part.

Model1: Ruoxi's model (model1):

- In Ruoxi's model, we reclassify the 'model' into three different level, which we mentioned in step two. So the dummy variables for model 1 is 19 variables.
- We use stepwise method to select model, and then we got 16 variables.

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.47239	0.03294	287.58	<.0001	0	0
newmileage	1	-0.00797	0.00036107	-22.06	<.0001	-0.15919	1.01403
Make1	1	0.35967	0.02603	13.82	<.0001	0.26268	7.03519
Make2	1	-0.10680	0.01219	-8.76	<.0001	-0.12755	4.12599
Make3	1	-0.10432	0.01242	-8.40	<.0001	-0.09916	2.71375
Make4	1	0.28009	0.02921	9.59	<.0001	0.23839	12.03239
Make5	1	-0.07015	0.01625	-4.32	<.0001	-0.04498	2.11313
Standard	1	0.13528	0.01920	7.04	<.0001	0.16330	10.46458
Luxury	1	0.32420	0.02820	11.49	<.0001	0.35265	18.32827
Type1	1	0.30700	0.01430	21.46	<.0001	0.18089	1.38346
Type2	1	-0.03796	0.01304	-2.91	0.0037	-0.02434	1.36172
Type4	1	0.11499	0.01521	7.56	<.0001	0.07594	1.96454
Cylinder	1	-0.06585	0.01191	-5.53	<.0001	-0.22279	31.62436
Liter	1	0.23461	0.01226	19.13	<.0001	0.63244	21.27363
Cylinder_Liter_c	1	-0.00634	0.00409	-1.55	0.1210	-0.02719	5.97509
Cruise	1	0.01941	0.00859	2.26	0.0240	0.02044	1.59134
Leather	1	0.01850	0.00733	2.52	0.0118	0.02018	1.24468

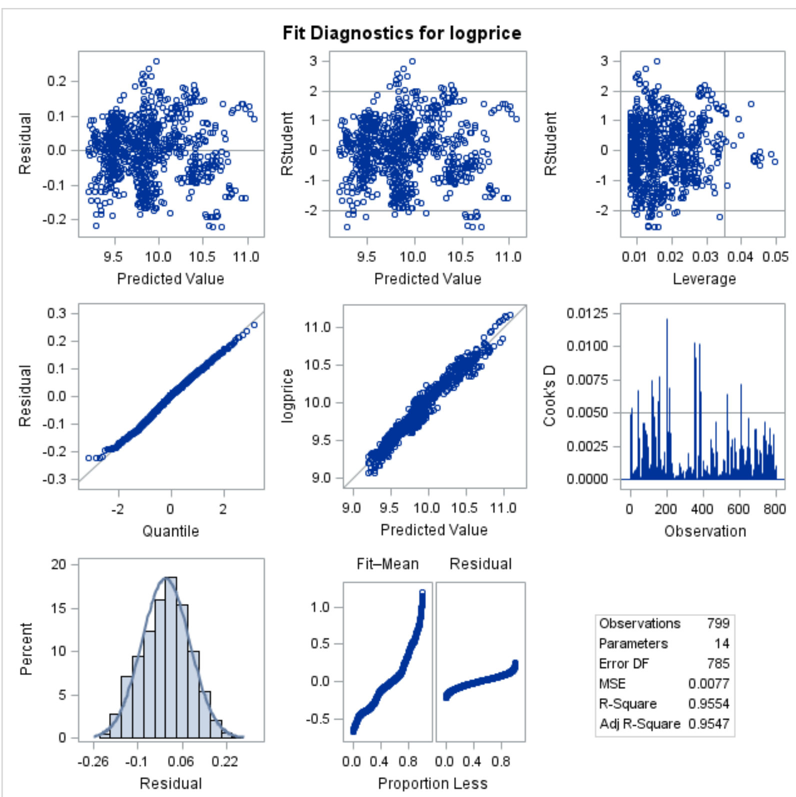
- We mentioned we create the interaction term in step two. However, according to the output, we can find that the interaction didn't solve the moticlinearity, therefore, we remove the highly colinearity variables and interaction term. And then check are there any outliers in the model. After we remove the outliers, we got the following variables:

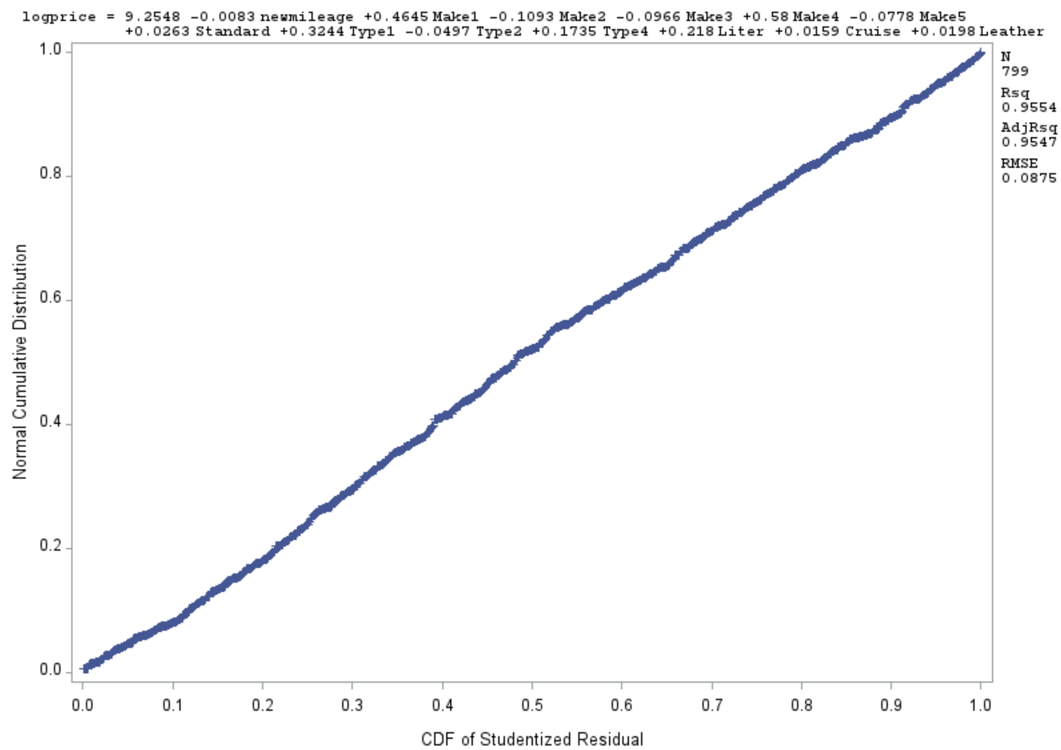
Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.34391	0.01991	469.22	<.0001	0	0
newmileage	1	-0.00798	0.00036635	-21.79	<.0001	-0.15924	1.01223
Make1	1	0.30773	0.02210	13.93	<.0001	0.22484	4.94235
Make2	1	-0.11193	0.01239	-9.03	<.0001	-0.13350	4.14247
Make3	1	-0.11237	0.01227	-9.16	<.0001	-0.10684	2.57893
Make4	1	0.32217	0.02833	11.37	<.0001	0.27430	11.03331
Make5	1	-0.07215	0.01643	-4.39	<.0001	-0.04628	2.10543
Standard	1	0.11409	0.01350	8.45	<.0001	0.13757	5.02029
Luxury	1	0.28282	0.02769	10.21	<.0001	0.30766	17.20250
Type1	1	0.29628	0.01416	20.93	<.0001	0.17466	1.32065
Type2	1	-0.06109	0.01262	-4.84	<.0001	-0.03919	1.24292
Type4	1	0.12230	0.01338	9.14	<.0001	0.08080	1.48235
Liter	1	0.16613	0.00637	26.08	<.0001	0.44782	5.58992
Cruise	1	0.01890	0.00860	2.20	0.0282	0.01990	1.55443
Leather	1	0.02594	0.00737	3.52	0.0005	0.02822	1.21756

We found that still has the variables which have higher VIF, then we remove the highest one ‘Luxury’ first to see what happened. After removing the ‘Luxury’, we find that all the variables seems good and we got 14 variables.

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.25478	0.01910	484.47	<.0001	0	0
newmileage	1	-0.00834	0.00037957	-21.98	<.0001	-0.16643	1.00973
Make1	1	0.46451	0.01771	26.22	<.0001	0.33950	2.95269
Make2	1	-0.10928	0.01284	-8.51	<.0001	-0.13031	4.13307
Make3	1	-0.09656	0.01260	-7.66	<.0001	-0.09181	2.52891
Make4	1	0.58002	0.01707	33.98	<.0001	0.48661	3.61269
Make5	1	-0.07778	0.01699	-4.58	<.0001	-0.04991	2.09480
Standard	1	0.02629	0.00979	2.69	0.0074	0.03163	2.44477
Type1	1	0.32441	0.01431	22.67	<.0001	0.19132	1.25519
Type2	1	-0.04974	0.01301	-3.82	0.0001	-0.03192	1.22755
Type4	1	0.17350	0.01369	12.67	<.0001	0.11133	1.36007
Liter	1	0.21800	0.00410	53.22	<.0001	0.58778	2.14918
Cruise	1	0.01588	0.00891	1.78	0.0750	0.01673	1.55055
Leather	1	0.01981	0.00765	2.59	0.0098	0.02152	1.21552

- We do the residual analysis to see the model violate any assumptions or not.





We find that although it is not perfect, it still looks good and not violate the assumptions.

- Finally, we do the validation for model, we use hold-out validation, and got the following results:

Test and Train Sets for gmcars

The REG Procedure
Model: MODEL1
Dependent Variable: logprice

Number of Observations Read	799
Number of Observations Used	799

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	13	128.76513	9.90501	1294.73	<.0001
Error	785	6.00546	0.00765		
Corrected Total	798	134.77059			

Root MSE	0.08747	R-Square	0.9554
Dependent Mean	9.87762	Adj R-Sq	0.9547
Coeff Var	0.88549		

Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	9.25478	0.01910	484.47	<.0001
newmileage	1	-0.00834	0.00037957	-21.98	<.0001
Make1	1	0.46451	0.01771	26.22	<.0001
Make2	1	-0.10928	0.01284	-8.51	<.0001
Make3	1	-0.09656	0.01260	-7.66	<.0001
Make4	1	0.58002	0.01707	33.98	<.0001
Make5	1	-0.07778	0.01699	-4.58	<.0001
Standard	1	0.02629	0.00979	2.69	0.0074
Type1	1	0.32441	0.01431	22.67	<.0001
Type2	1	-0.04974	0.01301	-3.82	0.0001
Type4	1	0.17350	0.01369	12.67	<.0001
Liter	1	0.21800	0.00410	53.22	<.0001
Cruise	1	0.01588	0.00891	1.78	0.0750
Leather	1	0.01981	0.00765	2.59	0.0098

Test and Train Sets for gmcars

Obs	_TYPE_	_FREQ_	rmse	mae
1	0	159	0.091128	0.073133

Test and Train Sets for gmcars

The CORR Procedure

2 Variables:	logprice yhat

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
logprice	159	9.84063	0.41476	1565	9.06403	11.16699	
yhat	159	9.84714	0.39898	1566	9.21673	11.07734	Predicted Value of logprice

Pearson Correlation Coefficients, N = 159 Prob > |r| under H0: Rho=0		
	logprice	yhat
logprice	1.00000	0.97566 <.0001
yhat Predicted Value of logprice	0.97566 <.0001	1.00000

Ruoxi's Model	Training Set – 644 Obs	Testing – 160 Obs
RMSE	0.087	0.0911
R^2	0.9554	0.9533
AdjR^2	0.9547	0.9756
GOF	OK	OK
Residual	OK	OK

CVR2=0.003

The final model:

Logprice = 9.25478 -0.00834*newmileage + 0.46451*Make1 - 0.10928*Make2 - 0.09656*Make3 + 0.58002*Make4 - 0.07778*Make5 + 0.02629*Standard + 0.32441*Type1 - 0.04974*Type2 + 0.1735*Type4 + 0.218*Liter + 0.01588*Cruise + 0.01981*Leather

Model2: Leanne's model

- Data exploring and cleaning, after import the data file there are some columns that is empty. So, we have to drop the empty column and rerun it to make it the data set that we want.

Figure 1. Original data set

Obs	Price	Mileage	Make	Model	Trim	Type	Cylinder	Liter	Doors	Cruise	Sound	Leather	VAR13	VAR14	VAR15	VAR16	VAR17	VAR18	VAR19	VAR20	VAR21	VAR22	VAR23	VAR24
1	17314.10313	8221	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	1												
2	17542.03608	9135	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0												
3	16218.84786	13196	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0												
4	16336.91314	16342	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	0												
5	16339.17032	19832	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	1												
6	15709.05282	22236	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0												
7	15230.00339	22576	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0												
8	15048.04218	22964	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0												
9	14862.09387	24021	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	1												
10	15295.01827	27325	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	1												

Figure 2. Cleaning the data set

Obs	Price	Mileage	Make	Model	Trim	Type	Cylinder	Liter	Doors	Cruise	Sound	Leather
1	17314.10313	8221	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	1
2	17542.03608	9135	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
3	16218.84786	13196	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
4	16336.91314	16342	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	0
5	16339.17032	19832	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	1
6	15709.05282	22236	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
7	15230.00339	22576	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0

- Then create histogram to see whether the variables are normally distributed. It turns out the distribution has a long tail and right skewness which means this distribution needs to be transformed. So, we apply log transformation on the and Price and named it as “Logprice”.

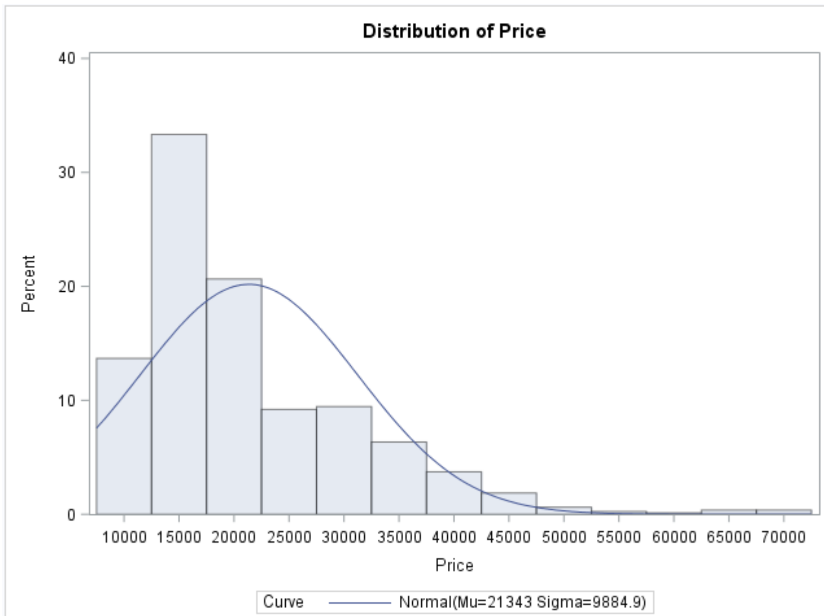


Figure 3. Distribution of Price

- After re-organizing the dataset, we have to deal with the problem with dummy variables. we use FREQ to see how many dummy variables in each independent variable that we have to create since “Model” and “Trim” are too complicated, we have to create too many dummy variables then we decide to drop them first. Then, we use “Make” to create 5 dummy variables and “Type” to create 4 dummy variables. After creating those dummy variables, drop “Make”, “Model”, “Trim”, and “Type” to make the dataset more clearly. So, there are 17 independent variables included dummy variables in total are listed below:

Figure 4. Total variables

17 Variables:	logprice Mileage M1 M2 M3 M4 M5 T1 T2 T3 T4 Cylinder Liter Doors Cruise Sound Leather

- To be prepare for the validation and final model2, we split the data into 80% for Training and 20% for Testing sets.

Figure 5. Data splitting

Test and Train Sets for car

Obs	Selected	Price	Mileage	Cylinder	Liter	Doors	Cruise	Sound	Leather	logprice	train_price	M1	M2	M3	M4	M5	T1	T2	T3	T4
1	1	17314.10313	8221	6	3.1	4	1	1	1	9.7593	17314.10	0	0	0	0	0	0	0	0	0
2	0	17542.03608	9135	6	3.1	4	1	1	0	.	.	0	0	0	0	0	0	0	0	0
3	1	16218.84786	13196	6	3.1	4	1	1	0	9.6939	16218.85	0	0	0	0	0	0	0	0	0
4	0	16336.91314	16342	6	3.1	4	1	0	0	.	.	0	0	0	0	0	0	0	0	0
5	0	16339.17032	19832	6	3.1	4	1	0	1	.	.	0	0	0	0	0	0	0	0	0

- ➢ Use Pearson Correlation Coefficient Matrix to check the correlation between each variable has multicollinearity problem or not. According to Figure 6, we can know that Cylinder and Liter has the multicollinearity problem because their value is higher than 0.9 which mean they have the strongest relationship in all of the variables. In this case, we have to create interaction term (Figure 7) to see whether this problem can be solved or not. We use center method to create interaction term.

Cylinder_m=5.26866-Cylinder

Liter_m=3.03731-Liter

Cylinder_Liter_m=Cylinder_m*Liter_m

As the result shows in Figure 8, we can know that the interaction term has reduce the connection between them.

Figure 6. Pearson Correlation Coefficient

	logprice	Mileage	M1	M2	M3	M4	M5	T1	T2	T3	T4	Cylinder	Liter	Doors	Cruise	Sound	Leather
logprice	1.00000 644	-0.14035 0.0004 644	0.57444 <.0001 644	-0.46626 <.0001 644	-0.09276 0.0185 644	0.39927 <.0001 644	-0.24375 <.0001 644	0.41581 <.0001 644	-0.25356 <.0001 644	-0.17294 <.0001 644	0.07799 0.0479 644	0.57182 <.0001 644	0.58514 <.0001 644	-0.07745 0.0495 644	0.50807 <.0001 644	-0.17239 <.0001 644	0.15140 0.0001 644
Mileage	-0.14035 0.0004 644	1.00000 804	-0.03747 0.2886 804	-0.01751 0.6202 804	-0.02989 0.3973 804	0.05618 0.1114 804	0.01747 0.6209 804	0.02744 0.4372 804	-0.02569 0.4669 804	0.00151 0.9659 804	0.02702 0.4443 804	-0.02946 0.4041 804	-0.01864 0.5977 804	-0.01694 0.6314 804	0.02504 0.4784 804	-0.02615 0.4591 804	0.00101 0.9773 804
M1	0.57444 <.0001 644	-0.03747 0.2886 804	1.00000 804	-0.27029 <.0001 804	-0.15920 <.0001 804	-0.13512 0.0001 804	-0.09440 0.0074 804	0.08646 0.0142 804	-0.09440 0.0074 804	-0.15264 <.0001 804	-0.09776 0.0055 804	0.53490 <.0001 804	0.40622 <.0001 804	0.08710 0.0135 804	0.19064 <.0001 804	-0.09193 0.0091 804	0.20530 <.0001 804
M2	-0.46626 <.0001 644	-0.01751 0.6202 804	-0.27029 <.0001 804	1.00000 804	-0.38941 <.0001 804	-0.33051 <.0001 804	-0.23091 <.0001 804	-0.10417 0.0031 804	0.34925 <.0001 804	0.22969 <.0001 804	-0.23913 <.0001 804	-0.15754 <.0001 804	-0.12405 0.0004 804	-0.14581 <.0001 804	-0.29319 <.0001 804	0.25957 <.0001 804	0.15549 <.0001 804
M3	-0.09276 0.0185 644	-0.02989 0.3973 804	-0.15920 <.0001 804	-0.38941 <.0001 804	1.00000 804	-0.19466 <.0001 804	-0.13600 0.0001 804	-0.12333 0.0005 804	-0.13600 0.0001 804	0.03267 0.3549 804	0.21302 <.0001 804	0.11444 0.0012 804	0.11386 0.0012 804	0.04094 0.2462 804	0.00094 0.9788 804	-0.07431 0.0351 804	-0.08985 0.0108 804
M4	0.39927 <.0001 644	0.05618 0.1114 804	-0.13512 0.0001 804	-0.33051 <.0001 804	-0.19466 <.0001 804	1.00000 804	-0.11543 0.0010 804	0.33825 <.0001 804	-0.11543 0.0010 804	-0.18664 <.0001 804	0.32833 <.0001 804	-0.37188 <.0001 804	-0.32675 <.0001 804	-0.02568 0.4671 804	0.23312 <.0001 804	-0.08721 0.0134 804	0.00381 0.9141 804
M5	-0.24375 <.0001 644	0.01747 0.6209 804	-0.09440 0.0074 804	-0.23091 <.0001 804	-0.13600 0.0001 804	-0.11543 0.0010 804	1.00000 804	-0.07313 0.0382 804	-0.08065 0.0222 804	0.11922 0.0007 804	-0.08351 0.0179 804	-0.19155 <.0001 804	-0.18094 <.0001 804	-0.06485 0.0661 804	-0.19904 <.0001 804	-0.13937 <.0001 804	-0.15279 <.0001 804
T1	0.41581 <.0001 644	0.02744 0.4372 804	0.08646 0.0142 804	-0.10417 0.0031 804	-0.12333 0.0005 804	0.33825 <.0001 804	-0.07313 0.0382 804	1.00000 804	-0.07313 0.0382 804	-0.11824 0.0008 804	-0.07573 0.0318 804	0.06153 0.0812 804	0.06589 0.0619 804	-0.46292 <.0001 804	0.14769 <.0001 804	-0.04364 0.2165 804	0.00928 0.7927 804
T2	-0.25356 <.0001 644	-0.02569 0.4669 804	-0.09440 0.0074 804	0.34925 <.0001 804	-0.13600 0.0001 804	-0.11543 0.0010 804	-0.08065 0.0222 804	-0.07313 0.0382 804	1.00000 804	-0.13040 0.0002 804	-0.08351 0.0179 804	-0.05502 0.1190 804	-0.12525 0.0004 804	0.15797 <.0001 804	-0.26485 <.0001 804	0.07354 0.0371 804	0.09070 0.0101 804
T3	-0.17294 <.0001 644	0.00151 0.9659 804	-0.15264 <.0001 804	0.22969 <.0001 804	0.03267 0.3549 804	-0.18664 <.0001 804	0.11922 0.0007 804	-0.11824 0.0008 804	-0.13040 0.0002 804	1.00000 804	-0.13504 0.0001 804	-0.04166 0.2381 804	0.04683 0.1847 804	-0.82544 <.0001 804	-0.04065 0.2497 804	0.09784 0.0055 804	0.06351 0.0719 804
T4	0.07799 0.0479 644	0.02702 0.4443 804	-0.09776 0.0055 804	-0.23913 <.0001 804	0.21302 <.0001 804	0.32833 <.0001 804	-0.08351 0.0179 804	-0.07573 0.0318 804	-0.08351 0.0179 804	-0.13504 0.0001 804	1.00000 804	-0.26906 <.0001 804	-0.25531 <.0001 804	0.16359 <.0001 804	-0.04428 0.2097 804	-0.14236 <.0001 804	-0.00337 0.9239 804
Cylinder	0.57182 <.0001 644	-0.02946 0.4041 804	0.53490 <.0001 804	-0.15754 <.0001 804	0.11444 0.0012 804	-0.37188 <.0001 804	-0.19155 <.0001 804	0.06153 0.0812 804	-0.05502 0.1190 804	-0.04166 0.2381 804	-0.26906 <.0001 804	1.00000 804	0.95790 <.0001 804	0.00221 0.9502 804	0.35428 <.0001 804	-0.08970 0.0109 804	0.07552 0.0323 804
Liter	0.58514 <.0001	-0.01864 0.5977	0.40622 <.0001	-0.12405 0.0004	0.11386 0.0012	-0.32675 <.0001	-0.18094 <.0001	0.06589 0.0619	-0.12525 0.0004	0.04683 0.1847	-0.25531 <.0001	0.95790 <.0001	1.00000	-0.07926 0.0246	0.37751 <.0001	-0.06553 0.0633	0.08733 0.0132

Figure 7. Relationship between Cylinder and Liter

Test and Train Sets for car

The CORR Procedure

3 Variables:	Cylinder Liter C_L

Simple Statistics						
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Cylinder	804	5.26866	1.38753	4236	4.00000	8.00000
Liter	804	3.03731	1.10556	2442	1.60000	6.00000
C_L	804	17.47015	10.98335	14046	6.40000	48.00000

Pearson Correlation Coefficients, N = 804 Prob > |r| under H0: Rho=0			
	Cylinder	Liter	C_L
Cylinder	1.00000	0.95790 <.0001	0.97474 <.0001
Liter	0.95790 <.0001	1.00000	0.98542 <.0001
C_L	0.97474 <.0001	0.98542 <.0001	1.00000

Figure 8. After creating interaction term in between

Test and Train Sets for car

The CORR Procedure

3 Variables:	Cylinder Liter Cylinder_Liter_m

Simple Statistics						
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Cylinder	804	5.26866	1.38753	4236	4.00000	8.00000
Liter	804	3.03731	1.10556	2442	1.60000	6.00000
Cylinder_Liter_m	804	1.46759	1.74004	1180	-0.17355	8.09211

Pearson Correlation Coefficients, N = 804 Prob > |r| under H0: Rho=0			
	Cylinder	Liter	Cylinder_Liter_m
Cylinder	1.00000	0.95790 <.0001	0.52413 <.0001
Liter	0.95790 <.0001	1.00000	0.55256 <.0001
Cylinder_Liter_m	0.52413 <.0001	0.55256 <.0001	1.00000

➢ After dealing with all the problem, we use stepwise method to run the model2 and the result shows in Figure 9. We can know that there are 15 variables in the first draft. Also, we use Variance inflation (VIF) to check whether there is a multicollinearity problem or not and we can get the result in Figure 11. By checking Figure 10, we notice that Cylinder and Liter still has multicollinearity problem then we remove the variable has the highest value of VIF which is Cylinder. After we rerun the model2, we will get Figure 11 since the VIF values are lower than 10, we can know that the multicollinearity problem is solved. Next, we have to remove the variable which is insignificant. Also, remove the outlier in the draft model2 (Figure 12).

Figure 9. Result of Stepwise selection

Summary of Stepwise Selection								
Step	Variable Entered	Variable Removed	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	Liter		1	0.3424	0.3424	7984.75	334.27	<.0001
2	M4		2	0.4089	0.7513	2623.51	1054.06	<.0001
3	M1		3	0.1339	0.8853	868.718	747.22	<.0001
4	Mileage		4	0.0246	0.9098	548.371	174.21	<.0001
5	T1		5	0.0215	0.9313	268.485	199.72	<.0001
6	T4		6	0.0095	0.9408	146.109	102.08	<.0001
7	M2		7	0.0041	0.9450	93.9532	47.71	<.0001
8	M3		8	0.0018	0.9468	72.1024	21.69	<.0001
9	M5		9	0.0025	0.9493	40.6662	31.89	<.0001
10	T2		10	0.0011	0.9504	28.2769	14.01	0.0002
11	Leather		11	0.0007	0.9511	20.9207	9.23	0.0025
12	Cruise		12	0.0003	0.9515	18.4153	4.47	0.0349
13	Cylinder		13	0.0003	0.9517	16.8862	3.51	0.0614
14	Cylinder_Liter_m		14	0.0002	0.9520	15.8098	3.07	0.0801
15	Sound		15	0.0002	0.9522	15.4952	2.32	0.1285

Figure 10. Multicollinearity checking

Root MSE	0.08972	R-Square	0.9522
Dependent Mean	9.88798	Adj R-Sq	0.9510
Coeff Var	0.90733		

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.30680	0.03422	271.96	<.0001	0	0
Mileage	1	-0.00000839	4.460571E-7	-18.82	<.0001	-0.16538	1.01406
M1	1	0.44934	0.02059	21.83	<.0001	0.32957	2.99238
M2	1	-0.13051	0.01452	-8.99	<.0001	-0.15726	4.01591
M3	1	-0.09642	0.01473	-6.55	<.0001	-0.09299	2.64777
M4	1	0.54420	0.01820	29.89	<.0001	0.48057	3.39232
M5	1	-0.09437	0.01914	-4.93	<.0001	-0.06060	1.98333
T1	1	0.32527	0.01700	19.13	<.0001	0.18923	1.28427
T2	1	-0.03881	0.01546	-2.51	0.0123	-0.02564	1.36911
T4	1	0.15498	0.01549	10.01	<.0001	0.10239	1.37410
Cylinder	1	-0.02364	0.01215	-1.95	0.0522	-0.08144	23.00265
Liter	1	0.24784	0.01420	17.45	<.0001	0.68462	20.20877
Cylinder_Liter_m	1	-0.00449	0.00282	-1.59	0.1115	-0.01981	2.02738
Cruise	1	0.02153	0.01039	2.07	0.0385	0.02297	1.61126
Sound	1	0.01243	0.00817	1.52	0.1285	0.01438	1.17248
Leather	1	0.02535	0.00879	2.88	0.0041	0.02769	1.20987

Figure 11. Model2 without multicollinearity problem

Root MSE	0.08992	R-Square	0.9519
Dependent Mean	9.88798	Adj R-Sq	0.9508
Coeff Var	0.90934		

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.25866	0.02369	390.83	<.0001	0	0
Mileage	1	-0.00000842	4.468895E-7	-18.83	<.0001	-0.16583	1.01336
M1	1	0.43011	0.01810	23.77	<.0001	0.31547	2.30261
M2	1	-0.12781	0.01448	-8.83	<.0001	-0.15400	3.97899
M3	1	-0.09865	0.01471	-6.70	<.0001	-0.09514	2.63172
M4	1	0.55124	0.01788	30.83	<.0001	0.48679	3.25820
M5	1	-0.08999	0.01905	-4.72	<.0001	-0.05779	1.95589
T1	1	0.32226	0.01697	18.99	<.0001	0.18748	1.27361
T2	1	-0.04882	0.01461	-3.34	0.0009	-0.03226	1.21747
T4	1	0.15735	0.01547	10.17	<.0001	0.10396	1.36554
Liter	1	0.22208	0.00516	43.08	<.0001	0.61346	2.65025
Cylinder_Liter_m	1	-0.00373	0.00280	-1.33	0.1830	-0.01644	1.98806
Cruise	1	0.02082	0.01040	2.00	0.0457	0.02222	1.60927
Sound	1	0.01388	0.00815	1.70	0.0890	0.01607	1.16266
Leather	1	0.02729	0.00875	3.12	0.0019	0.02980	1.19437

Figure 12. Model2 without multicollinearity problem and insignificant variables

Root MSE	0.09014	R-Square	0.9515
Dependent Mean	9.88798	Adj R-Sq	0.9506
Coeff Var	0.91158		

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.28105	0.02148	432.14	<.0001	0	0
Mileage	1	-0.00000842	4.479587E-7	-18.80	<.0001	-0.16595	1.01321
M1	1	0.41920	0.01696	24.72	<.0001	0.30747	2.01151
M2	1	-0.13224	0.01386	-9.54	<.0001	-0.15934	3.62659
M3	1	-0.10366	0.01441	-7.19	<.0001	-0.09997	2.51236
M4	1	0.54311	0.01734	31.32	<.0001	0.47961	3.04935
M5	1	-0.10030	0.01847	-5.43	<.0001	-0.06441	1.82895
T1	1	0.31604	0.01653	19.12	<.0001	0.18386	1.20303
T2	1	-0.04907	0.01463	-3.36	0.0008	-0.03242	1.21429
T4	1	0.15060	0.01521	9.90	<.0001	0.09950	1.31403
Liter	1	0.21764	0.00435	50.09	<.0001	0.60121	1.87367
Cruise	1	0.02188	0.01035	2.11	0.0349	0.02335	1.58655
Leather	1	0.02853	0.00869	3.28	0.0011	0.03117	1.17304

- As a result, we get in Figure 13, then we have to check on the Goodness of Fit (GOF) and the assumptions.

  - ✓ GOF:

    H0: b1=bi=0

    Ha: At least one coefficient bi≠0

    Test Statistic: F==1084.71

    p-value < 0.0001

    reject the H0 and there is at least one coefficient parameter in the model2.

  - ✓ In order to check the regression model2 is valid or not (Figure 14), we have to check the assumptions to see whether there is a problem or not. Four assumptions are listing below:
  - Linearity: As we check the scatter plot, we can see that the pattern of the spread show a straight line in between.

- Independent: The points are mostly randomly scattered around the zero line so we can assume the errors are independent to each other.
- Constant variance: The points are mostly randomly scattered around the zero line and the pattern of the spread in the residuals didn't increase or decrease, so the errors have constant variance.
- Normality: We can see that the points lie close to the line and the pattern of the spread show a 45degree straight line. So, we can assume that the errors are probably normal.

Finally, we can fit the final model2 equation:

LogPrice = 9.27455 - 0.000008*Mileage + 0.41921*M1 - 0.13005*M2 - 0.10618*M3 + 0.55124*M4 – 0.0977*M5 +0.31129*T1 – 0.04842*T2 + 0.1693*T4 + 0.21959*Liter + 0.02195*Cruise + 0.02672*Leather

By checking the Standard Estimate (STB), we can know that Liter has the strongest influence on the variance of car price.

Figure 13. Final model2

The REG Procedure
Model: MODEL1
Dependent Variable: logprice

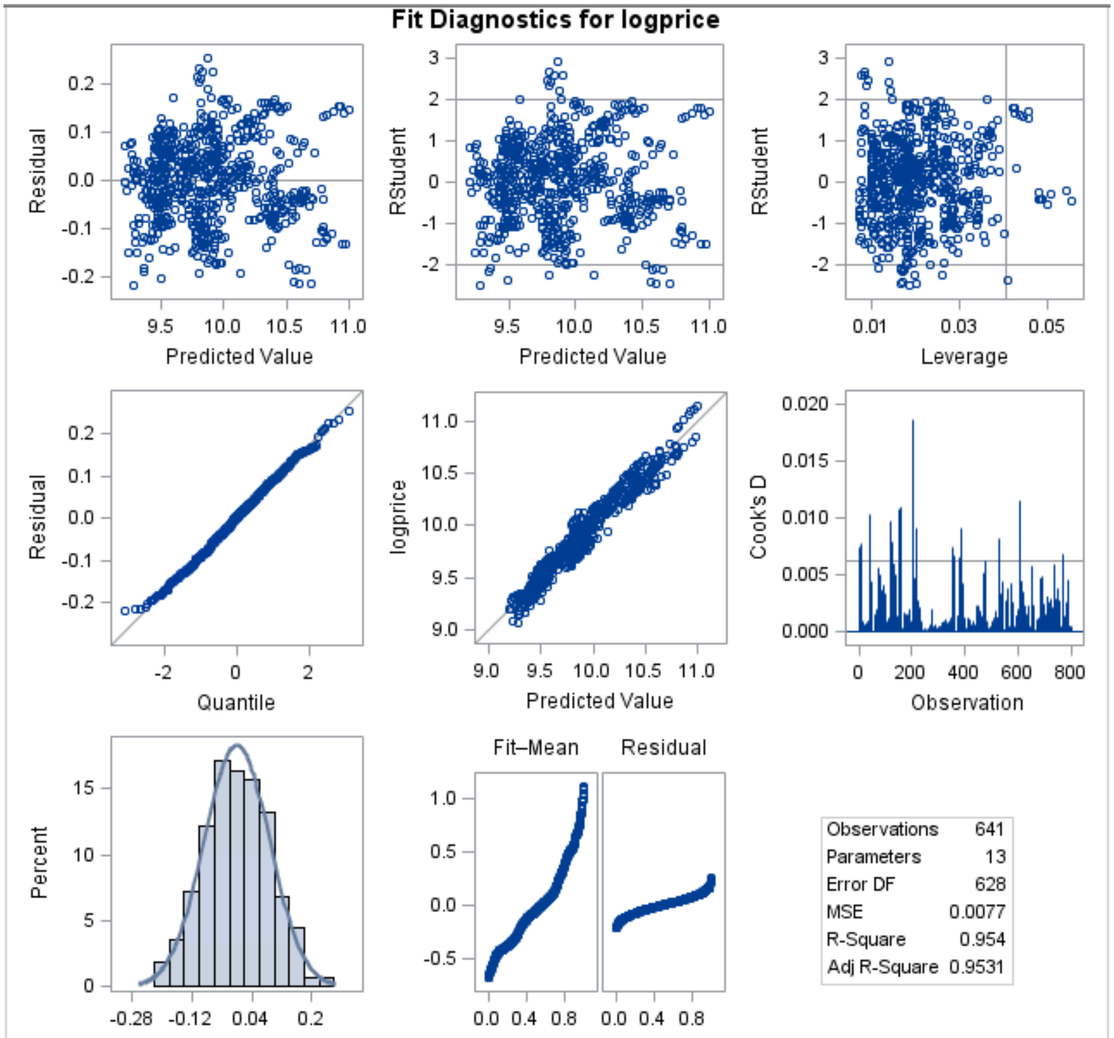
Number of Observations Read	801
Number of Observations Used	641
Number of Observations with Missing Values	160

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	12	100.66181	8.38848	1084.71	<.0001
Error	628	4.85657	0.00773		
Corrected Total	640	105.51838			

Root MSE	0.08794	R-Square	0.9540
Dependent Mean	9.88703	Adj R-Sq	0.9531
Coeff Var	0.88945		

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	9.27455	0.02098	441.97	<.0001	0	0
Mileage	1	-0.00000842	4.372025E-7	-19.25	<.0001	-0.16586	1.01303
M1	1	0.41921	0.01655	25.33	<.0001	0.30759	2.01129
M2	1	-0.13005	0.01353	-9.61	<.0001	-0.15656	3.61934
M3	1	-0.10618	0.01407	-7.55	<.0001	-0.10241	2.51219
M4	1	0.55214	0.01699	32.50	<.0001	0.48141	2.99350
M5	1	-0.09770	0.01802	-5.42	<.0001	-0.06277	1.82938
T1	1	0.31129	0.01615	19.27	<.0001	0.18118	1.20568
T2	1	-0.04842	0.01427	-3.39	0.0007	-0.03200	1.21388
T4	1	0.16930	0.01518	11.15	<.0001	0.10877	1.29757
Liter	1	0.21959	0.00425	51.64	<.0001	0.60661	1.88291
Cruise	1	0.02195	0.01010	2.17	0.0302	0.02341	1.58424
Leather	1	0.02672	0.00851	3.14	0.0018	0.02913	1.17528

Figure 14. Residual analysis of the model2



- In order to test the performance of the model2, we randomly choose 20% observations from the original dataset as the test set. All the result are showed in Figure 15. It's a good case it because the value of CV-R^2 is ≤ 0.3.

Figure 15. Result of Validation

Test and Train Sets for car

Obs	_TYPE_	_FREQ_	rmse	mae
1	0	160	0.094729	0.073411

Test and Train Sets for car

The CORR Procedure

2 Variables:	logprice yhat

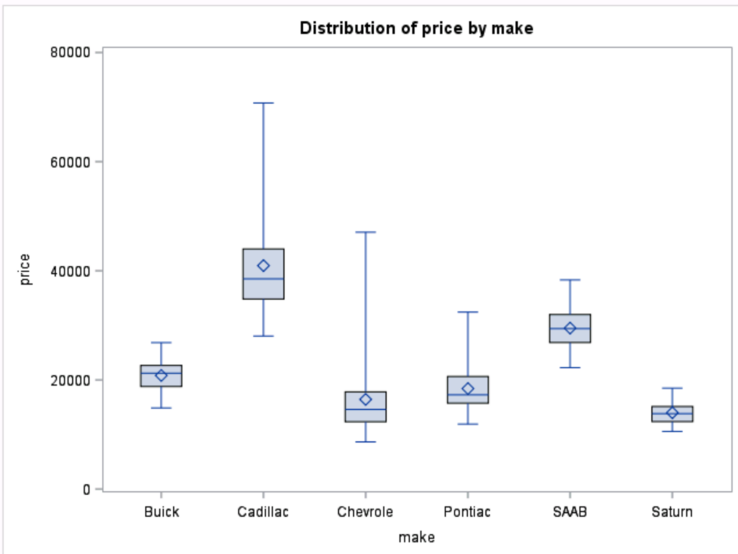
Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
logprice	160	9.84310	0.42816	1575	9.07898	11.16699	
yhat	160	9.84503	0.40669	1575	9.19975	11.06295	Predicted Value of logprice

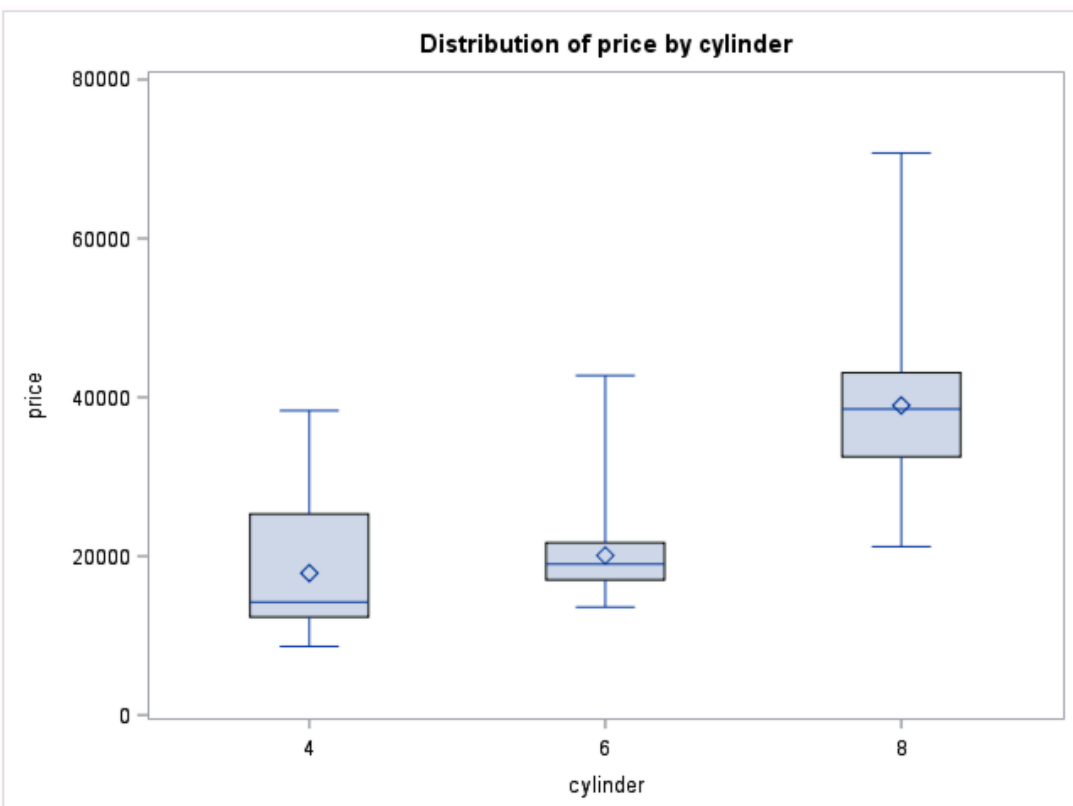
Pearson Correlation Coefficients, N = 160 Prob > |r| under H0: Rho=0		
	logprice	yhat
logprice	1.00000	0.97556 <.0001
yhat Predicted Value of logprice	0.97556 <.0001	1.00000

Result	Training set 641 observations	Testing set 160 observations
RMSE	0.08794	0.09473
MAE	N/A	0.073411
R^2	0.9540	0.9518
Adj-R^2	0.9531	0.9479
CV-R^2	N/A	0.002
GOF	OK	N/A
Residuals	OK	N/A

Model3: Minh's model

We sort the data by some categories to make boxplots in order to find the patterns about ln_price:





- We mentioned in step two for interaction tem. So after create interacation term. We then ran the correlation model. There is no significant relationship between ln_price and any predictors. But we noticed that there is a high association between cylinder and liter. At the end of this phase, we concluded that there is no significant association between ln_price and any variables. But there are some interesting patterns:

  The car with more cylinder will have higher average ln_price

  The convertible cars have higher average ln_price

  Trim is redundant of Type

  Model depends on other qualitative variables such as cruise, leather, the Model will change for some feature of the car, for e.g 2 cars with same type same brand will have model if there have different number of cylinder.

Parameter Estimates								
Variable	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Standardized Estimate**	**Tolerance**	**Variance Inflation**
Intercept	1	9.46372	0.05009	188.93	<.0001	0	.	0
mileage	1	-0.00000802	4.551307E-7	-17.63	<.0001	-0.15969	0.98892	1.01121
Make0	1	0.57514	0.01956	29.41	<.0001	0.42406	0.39032	2.56199
Make1	0	0	.	.	.	.	.	.
Make2	1	-0.00963	0.01124	-0.86	0.3919	-0.00909	0.72146	1.38608
Make3	1	0.64916	0.01595	40.70	<.0001	0.55107	0.44286	2.25804
Make4	1	0.00856	0.01568	0.55	0.5854	0.00558	0.77617	1.28837
Type0	1	-0.34472	0.02041	-16.89	<.0001	-0.30871	0.24294	4.11633
Type1	0	0	.	.	.	.	.	.
Type2	1	0.06575	0.01725	3.81	0.0002	0.07696	0.19906	5.02352
Type3	1	0.23091	0.02346	9.84	<.0001	0.15059	0.34677	2.88377
d	1	-0.40174	0.02375	-16.92	<.0001	-0.40721	0.14007	7.13914
Cylinder0	1	-0.02300	0.03782	-0.61	0.5434	-0.02707	0.04097	24.40909
Cylinder1	1	-0.18643	0.11931	-1.56	0.1186	-0.14930	0.00889	112.45751
liter	1	0.20625	0.05530	3.73	0.0002	0.54597	0.00379	264.03905
cruise	1	0.02751	0.01099	2.50	0.0125	0.02801	0.64833	1.54242
sound	1	0.00600	0.00847	0.71	0.4795	0.00684	0.86964	1.14990
leather	1	0.01004	0.00939	1.07	0.2851	0.01092	0.77914	1.28347
Cylinder_Liter	1	0.00613	0.00826	0.74	0.4583	0.16146	0.00171	583.23530

The model shows a high collinearity between, Cylinder_Liter, Cylinder0, Cylinder1, Liter. Also, Cylinder_liter is not significant so we removed this interaction variable.

- We then ran the model with different methods:

a. Stepwise

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	10	104.55986	10.45599	1129.44	<.0001
Error	633	5.86011	0.00926		
Corrected Total	643	110.41997			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	9.51583	0.03441	708.09510	76487.3	<.0001
mileage	-0.00000804	4.618986E-7	2.80368	302.85	<.0001
Make0	0.54726	0.01664	10.01183	1081.46	<.0001
Make3	0.63756	0.01497	16.78842	1813.46	<.0001
Type0	-0.33586	0.02013	2.57732	278.40	<.0001
Type2	0.05561	0.01610	0.11045	11.93	0.0006
Type3	0.20457	0.02152	0.83680	90.39	<.0001
d	-0.37412	0.02320	2.40673	259.97	<.0001
cylinder	-0.02047	0.01247	0.02496	2.70	0.1011
liter	0.24717	0.01434	2.74935	296.98	<.0001
cruise	0.03661	0.01081	0.10619	11.47	0.0008

b. Forward

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	12	104.57703	8.71475	941.14	<.0001
Error	631	5.84294	0.00926		
Corrected Total	643	110.41997			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	9.50211	0.03594	647.27844	69901.9	<.0001
mileage	-0.00000803	4.623034E-7	2.79615	301.97	<.0001
Make0	0.54075	0.01780	8.54304	922.59	<.0001
Make2	-0.01023	0.01126	0.00765	0.83	0.3638
Make3	0.63572	0.01531	15.96476	1724.09	<.0001
Type0	-0.33667	0.02017	2.58023	278.65	<.0001
Type2	0.05971	0.01646	0.12185	13.16	0.0003
Type3	0.21502	0.02319	0.79621	85.99	<.0001
d	-0.37853	0.02345	2.41303	260.59	<.0001
cylinder	-0.01623	0.01289	0.01468	1.59	0.2085
liter	0.24344	0.01462	2.56860	277.39	<.0001
cruise	0.03703	0.01082	0.10850	11.72	0.0007
sound	0.00791	0.00830	0.00842	0.91	0.3406

c. Backward

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	104.53490	11.61499	1251.29	<.0001
Error	634	5.88507	0.00928		
Corrected Total	643	110.41997			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	9.47627	0.02460	1377.36583	148384	<.0001
mileage	-0.00000804	4.624991E-7	2.80841	302.55	<.0001
Make0	0.53243	0.01399	13.43813	1447.69	<.0001
Make3	0.64411	0.01445	18.44471	1987.05	<.0001
Type0	-0.33007	0.01984	2.56795	276.65	<.0001
Type2	0.06183	0.01567	0.14453	15.57	<.0001
Type3	0.21188	0.02108	0.93775	101.02	<.0001
d	-0.37909	0.02304	2.51377	270.81	<.0001
liter	0.22484	0.00456	22.56914	2431.38	<.0001
cruise	0.03479	0.01077	0.09690	10.44	0.0013

d. AdjRsqrt

Number in Model	Adjusted R-Square	R-Square	Variables in Model
11	0.9461	0.9470	mileage Make0 Make3 Type0 Type2 Type3 d cylinder liter cruise sound
10	0.9461	0.9469	mileage Make0 Make3 Type0 Type2 Type3 d cylinder liter cruise
11	0.9461	0.9470	mileage Make0 Make2 Make3 Type0 Type2 Type3 d cylinder liter cruise
12	0.9461	0.9471	mileage Make0 Make2 Make3 Type0 Type2 Type3 d cylinder liter cruise sound
11	0.9460	0.9470	mileage Make0 Make2 Make3 Type0 Type2 Type3 d liter cruise sound
10	0.9460	0.9468	mileage Make0 Make2 Make3 Type0 Type2 Type3 d liter cruise
12	0.9460	0.9470	mileage Make0 Make3 Make4 Type0 Type2 Type3 d cylinder liter cruise sound
12	0.9460	0.9470	mileage Make0 Make3 Type0 Type2 Type3 d cylinder liter cruise sound leather
11	0.9460	0.9469	mileage Make0 Make3 Type0 Type2 Type3 d cylinder liter cruise leather
11	0.9460	0.9469	mileage Make0 Make3 Make4 Type0 Type2 Type3 d cylinder liter cruise
12	0.9460	0.9470	mileage Make0 Make2 Make3 Make4 Type0 Type2 Type3 d cylinder liter cruise
12	0.9460	0.9470	mileage Make0 Make2 Make3 Type0 Type2 Type3 d cylinder liter cruise leather
13	0.9460	0.9471	mileage Make0 Make2 Make3 Make4 Type0 Type2 Type3 d cylinder liter cruise sound
13	0.9460	0.9471	mileage Make0 Make2 Make3 Type0 Type2 Type3 d cylinder liter cruise sound leather
10	0.9460	0.9468	mileage Make0 Make3 Type0 Type2 Type3 d liter cruise sound
9	0.9459	0.9467	mileage Make0 Make3 Type0 Type2 Type3 d liter cruise
12	0.9459	0.9470	mileage Make0 Make2 Make3 Type0 Type2 Type3 d liter cruise sound leather

- We decided to choose the ninth model in the AdjR^2 method which has 9 variables and also one of cylinder and liter has been removed.

My Model: ln_price = B0 + B1*mileage + B2*Make0 + B3*Make3 + B4*Type0 + B5*Type2 + B6*Type3 + B7*D + B8 * liter + B9*cruise

The output is pretty good with high AdjR^2, the Goodness of Fit Test, all variables are significant.

Number of Observations Read	804
Number of Observations Used	644
Number of Observations with Missing Values	160

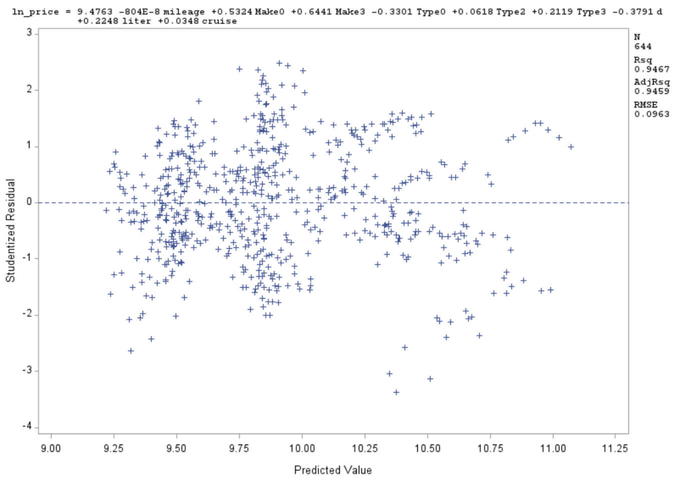
Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	104.53490	11.61499	1251.29	<.0001
Error	634	5.88507	0.00928		
Corrected Total	643	110.41997			

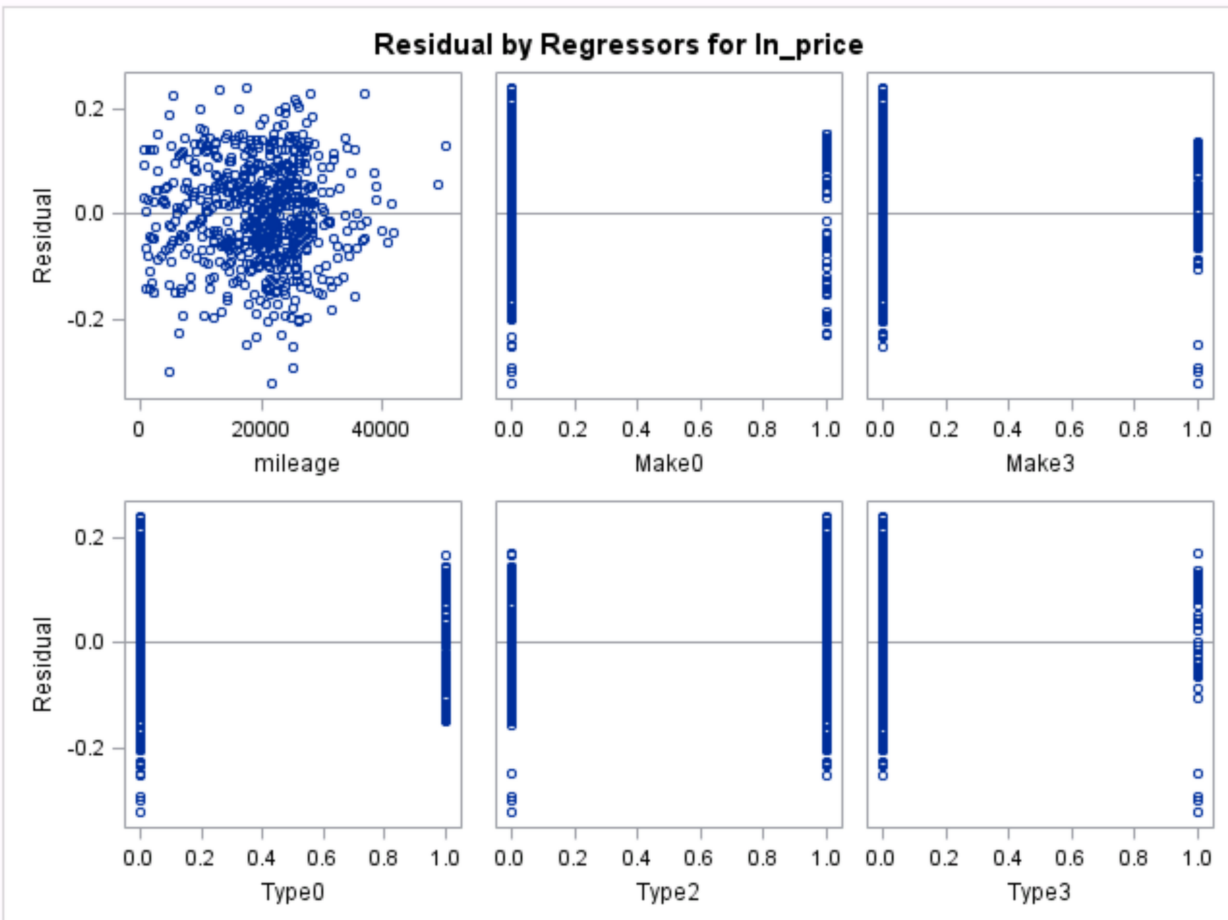
Root MSE	0.09635	R-Square	0.9467
Dependent Mean	9.88536	Adj R-Sq	0.9459
Coeff Var	0.97463		

Parameter Estimates								
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Tolerance	Variance Inflation
Intercept	1	9.47627	0.02460	385.21	<.0001	0	.	0
mileage	1	-0.00000804	4.624991E-7	-17.39	<.0001	-0.16014	0.99180	1.00827
Make0	1	0.53243	0.01399	38.05	<.0001	0.39257	0.78969	1.26632
Make3	1	0.64411	0.01445	44.58	<.0001	0.54677	0.55874	1.78975
Type0	1	-0.33007	0.01984	-16.63	<.0001	-0.29559	0.26617	3.75694
Type2	1	0.06183	0.01567	3.95	<.0001	0.07238	0.24987	4.00209
Type3	1	0.21188	0.02108	10.05	<.0001	0.13818	0.44480	2.24819
d	1	-0.37909	0.02304	-16.46	<.0001	-0.38425	0.15419	6.48546
liter	1	0.22484	0.00456	49.31	<.0001	0.59518	0.57700	1.73310
cruise	1	0.03479	0.01077	3.23	0.0013	0.03543	0.69908	1.43046

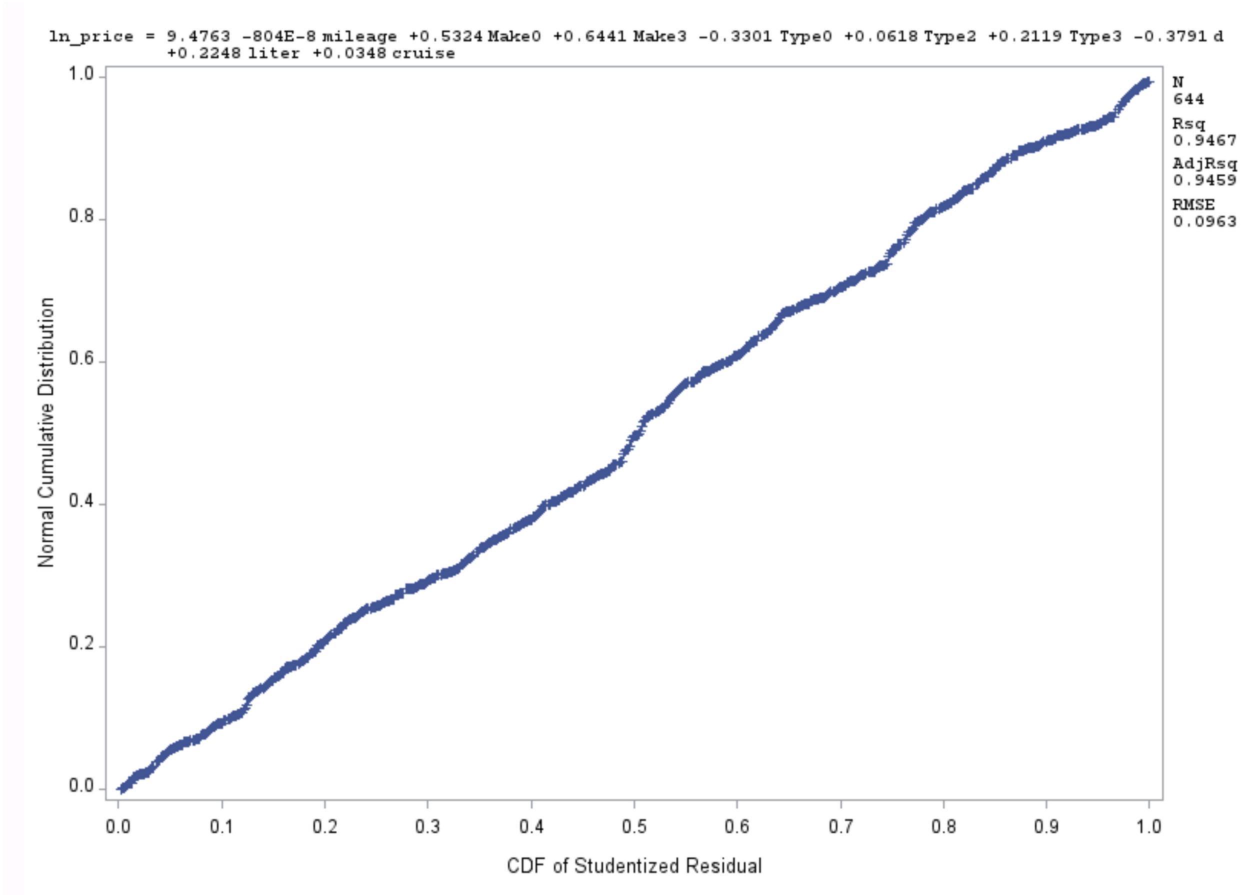
- Then we checked the 4 assumptions of the model

For Residual vs Predicted value and Residual vs vars, points randomly scattered around the zeroline.





For the QQ plot, the graph didn’t show a fine straight line but it is almost a straight line.

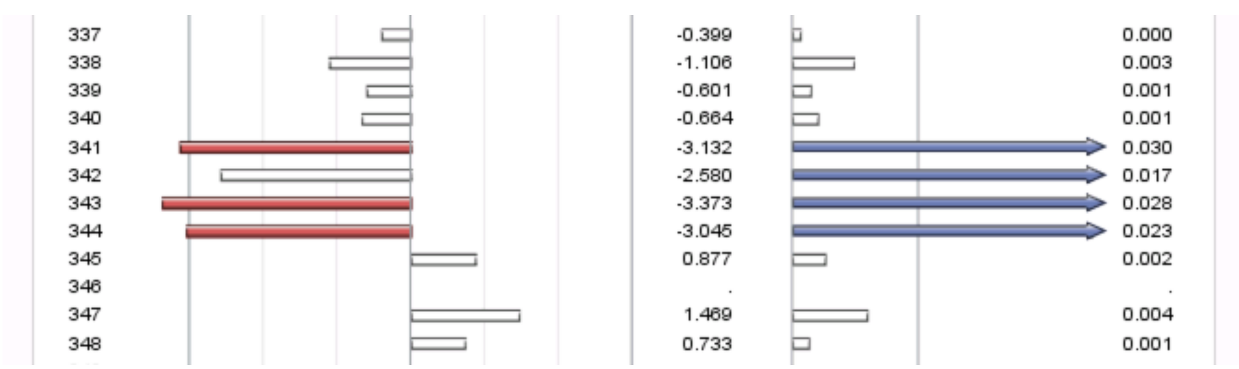


Based on the residual plots above, we concluded that there is no failure in the assumptions of the selected model.

- We then check the influence points and outliers.

We found and removed the observations which are both influence points and outliers below:

N = 341, 343, 344



- At this step, we was satisfied with our model because the result is much more better than the result we showed in the presentation.

Number of Observations Read	801
Number of Observations Used	641
Number of Observations with Missing Values	160

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	104.66488	11.62943	1313.20	<.0001
Error	631	5.58799	0.00886		
Corrected Total	640	110.25287			

Root MSE	0.09411	R-Square	0.9493
Dependent Mean	9.88431	Adj R-Sq	0.9486
Coeff Var	0.95207		

Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate
Intercept	1	9.46784	0.02408	393.16	<.0001	0
mileage	1	-0.00000813	4.532104E-7	-17.93	<.0001	-0.16143
Make0	1	0.53198	0.01367	38.92	<.0001	0.39243
Make3	1	0.65326	0.01420	46.00	<.0001	0.54722
Type0	1	-0.32446	0.01941	-16.72	<.0001	-0.29065
Type2	1	0.06018	0.01531	3.93	<.0001	0.07023
Type3	1	0.22668	0.02075	10.93	<.0001	0.14386
d	1	-0.37266	0.02253	-16.54	<.0001	-0.37775
liter	1	0.22631	0.00446	50.73	<.0001	0.59917
cruise	1	0.03452	0.01052	3.28	0.0011	0.03516

Validation:

At the beginning, we use 80% of the dataset for training, 20% for testing.

We used the selected model to predict the ln_price of the test set and export these value as yhat, we then compare this column with the real ln_price.

ln_price	Make0	Make1	Make2	Make3	Make4	Type0	Type1	Type2	Type3	Cylinder0	Cylinder1	d	Cylinder_Liter	yhat	absd
9.7593	0	0	0	0	0	0	0	1	0	1	0	-0.06607	18.6	9.8253	0.06607
9.7724	0	0	0	0	0	0	0	1	0	1	0	-0.04560	18.6	9.8180	0.04560
9.9300	0	0	0	0	0	0	0	1	0	1	0	0.04726	21.6	9.8828	0.04726
9.8921	0	0	0	0	0	0	0	1	0	1	0	0.07641	21.6	9.8157	0.07641
9.7964	0	0	0	0	0	0	0	1	0	1	0	0.07208	21.6	9.7243	0.07208
9.9940	0	0	0	0	0	0	0	1	0	1	0	0.12293	21.6	9.8711	0.12293
9.9652	0	0	0	0	0	0	0	1	0	1	0	0.11349	21.6	9.8517	0.11349
9.9476	0	0	0	0	0	0	0	1	0	1	0	0.13763	22.8	9.8100	0.13763
9.9084	0	0	0	0	0	0	0	1	0	1	0	-0.06011	22.8	9.9685	0.06011

Validation statistics for Model

Obs	_TYPE_	_FREQ_	rmse	mae
1	0	159	0.094578	0.077442

Validation statistics for Model

The CORR Procedure

2 Variables:	ln_price yhat

Simple Statistics							
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum	Label
ln_price	159	9.84856	0.38851	1566	9.09054	11.14380	
yhat	159	9.84137	0.37996	1565	9.21724	11.01844	Predicted Value of ln_price

Pearson Correlation Coefficients, N = 159 Prob > |r| under H0: Rho=0		
	ln_price	yhat
ln_price	1.00000	0.96995 <.0001
yhat Predicted Value of ln_price	0.96995 <.0001	1.00000

Minh's Model	Training Set – 644 Obs	Testing Set – 160 Obs
RMSE	0.09411	0.0946
R^2	0.9493	0.9409
AdjR^2	0.9486	0.937
GOF	OK	OK
Residual	OK	OK

5-fold validation

The result for 5-fold validation is pretty much the same as the previous model.

Parameter Estimates										
						Cross Validation Estimates				
Parameter	DF	Estimate	Standardized Estimate	Standard Error	t Value	1	2	3	4	5
Intercept	1	9.489708	0	0.028093	337.80	9.50E+00	9.49E+00	9.49E+00	9.49E+00	9.49E+00
mileage	1	-0.000008240	-0.166055	0.000000528	-15.60	-8.12E-06	-8.36E-06	-8.29E-06	-8.31E-06	-8.14E-06
Make0	1	0.530489	0.392570	0.016110	32.93	5.22E-01	5.33E-01	5.26E-01	5.39E-01	5.32E-01
Make3	1	0.632057	0.558868	0.016256	38.88	6.31E-01	6.28E-01	6.33E-01	6.33E-01	6.36E-01
Type0	1	-0.327527	-0.295642	0.022191	-14.76	-3.33E-01	-3.26E-01	-3.26E-01	-3.28E-01	-3.25E-01
Type2	1	0.062595	0.074201	0.018194	3.44	6.29E-02	5.46E-02	6.55E-02	6.30E-02	6.67E-02
Type3	1	0.215969	0.150972	0.023489	9.19	2.08E-01	2.10E-01	2.20E-01	2.20E-01	2.21E-01
d	1	-0.376114	-0.387576	0.025965	-14.49	-3.82E-01	-3.70E-01	-3.74E-01	-3.76E-01	-3.79E-01
liter	1	0.221966	0.587390	0.005305	41.84	2.22E-01	2.20E-01	2.22E-01	2.22E-01	2.24E-01
cruise	1	0.037486	0.038858	0.012532	2.99	3.54E-02	4.35E-02	3.63E-02	4.13E-02	3.10E-02

Comparison

AdjRsqr Formula:

1-(((1-R^2)(n-1))/(n-k-1))

CV-R^2 Fomula:

|ModelR^2 – R^2(CV)|

We did use the same rate for 80/20 splitting the data to training and testing set.

For model assumption (will be mentioned below), all three models meet the assumptions and also pass the GOF test.

a. Ruoxi’s model contains 14 variables, has the lowest RMSE and also has the highest AdjR^2,

CV-R^2 = 0.0021

b. Leanne’s model contains 12variables, has the average RMSE and also has the average AdjR^2,

CV-R^2 = 0.0003

c. Minh’s model contains 9 variables, has the highest RMSE and also the lowest AdjR^2, and

CV-R^2 = 0.007

Detail information given through table below.

So basically, out three models seem pretty good, we then concluded to choose Minh’s model because this one is pretty straight forward and 9 variables is a suitable numer.

Ruoxi’s Model	Training Set – 644 Obs	Testing – 160 Obs
RMSE	0.087	0.0911
R^2	0.9554	0.9533
AdjR^2	0.9547	0.9756
GOF	OK	OK
Residual	OK	OK
Leanne’s Model	**Training Set – 644 Obs**	**Testing Set – 160 Obs**
RMSE	0.08992	0.094446
R^2	0.9519	0.9516
AdjR^2	0.9508	0.9974
GOF	OK	OK
Residual	OK	OK
Minh’s Model	**Training Set – 644 Obs**	**Testing Set – 160 Obs**
RMSE	0.09411	0.0946
R^2	0.9493	0.9409
AdjR^2	0.9486	0.937
GOF	OK	OK

Residual	OK	OK

Our best model:

Number of Observations Read	801
Number of Observations Used	641
Number of Observations with Missing Values	160

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	104.66488	11.62943	1313.20	<.0001
Error	631	5.58799	0.00886		
Corrected Total	640	110.25287			

Root MSE	0.09411	R-Square	0.9493
Dependent Mean	9.88431	Adj R-Sq	0.9486
Coeff Var	0.95207		

Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate
Intercept	1	9.46784	0.02408	393.16	<.0001	0
mileage	1	-0.00000813	4.532104E-7	-17.93	<.0001	-0.16143
Make0	1	0.53198	0.01367	38.92	<.0001	0.39243
Make3	1	0.65326	0.01420	46.00	<.0001	0.54722
Type0	1	-0.32446	0.01941	-16.72	<.0001	-0.29065
Type2	1	0.06018	0.01531	3.93	<.0001	0.07023
Type3	1	0.22668	0.02075	10.93	<.0001	0.14386
d	1	-0.37266	0.02253	-16.54	<.0001	-0.37775
liter	1	0.22631	0.00446	50.73	<.0001	0.59917
cruise	1	0.03452	0.01052	3.28	0.0011	0.03516

The strongest predictors are liter, Make3 (SAAB), Make0 (Cadillac) and then Doors.

If the liter increases by 1, the price increases (exp(0.226) – 1)*100% = 25.35%

If the car is made by SAAB, the price increases (exp(0.653) – 1)*100% = 92.1%

If the car is made by Cadillacs, the price increases (exp(0.532) – 1)*100% = 70.2%

Test 2 predictions:

*I chose a Cadillac(Make0) Sedan(Type2) 4doors(d=1) liter=3.8 cruise=yes mileage=1000;

Output Statistics								
Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean		95% CL Predict		Residual
1	.	10.5724	0.0146	10.5438	10.6010	10.3809	10.7639	.

Predicted price = exp(10.5724) = 39042 ($)

Confidence Interval (37941; 40175) ($)

Predict Interval (32238; 47282) ($)

*I chose a SAAB(Make3) Coupe(Type0) 2doors(d=0) liter=3.1 cruise=no mileage=2000;

Output Statistics								
Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean		95% CL Predict		Residual
1	.	10.4712	0.0225	10.4271	10.5154	10.2768	10.6656	.

Predicted price = exp(10.4712) = 35284 ($)

Confidence Interval (33762; 36879) ($)

Predict Interval (29050; 42856) ($)

Limitation & Future Work

For the dataset, the number of observation is pretty low for each specific model/trim of car, the data also have redundant variables. We also notice that these variables are very sensitive with the change of some other specific variables. So in the future, we could use some technique such as feature extraction to decrease the number of variables.

In the first phase, we discovered some interesting pattern from the data but couldn't apply it in the model. In the future, develop a model based on these patterns could be an effective way.

Also, we dropped trim and model but we still would like to find out whether trim and model could improve the model's performance, and also to check the other interaction variable options. To do this, develop the model by reclassifying or clustering observations as Ruoxi did could be a promising method.

Recommendation

The most important indicators in this model are mostly car brands(Make).

In my opinion, for car buyer, the most important indicator should be first type of car, for e.g the average price of a convertible car is almost always higher than those of a coupe. Then for each type of car, they should check the engine(in this case we don't have this variable) and the cylinder to the suitable one. The third important element is mileage to check the condition of the car, if the car is too old, the price should be low but the engine could not be in good condition. And the last thing is Brand name, of course the brand name is very important for many people, but the luxury brand comes with higher price, so buyer should do cross check between car with similar type and model before made the decision.

Appendix

Ruoxi's code

```
*Import the data;
proc import out=gmcars replace
datafile='C:\Users\rwang37\Desktop\gmcar_price.txt' ;
```

```
delimiter='09'x;
getnames=yes;
run;
proc print;
run;
*split the original sample data;
proc surveyselect data=gmcars out=car_all seed=495857
samprate=0.80 outall;
run;
data car_all;
set car_all;
if selected then train_price=price;
logprice=log(train_price);
run;
proc print data=car_all;
run;
*test frequency and to see how many terms each variable have;
proc freq;
table make;
run;
proc freq;
table model;
run;
proc freq;
table trim;
run;
proc freq;
table type;
run;
```

```
proc freq;
table cylinder;
run;
*calculate the minimum, maximum,median, p25 and p75, so that I can define the price range for each reclassify level;
proc means min max median p25 p75;
var price;
run;
data gmcars;
set gmcars;
Level=0;
if Model='Century' then Level=1;
if Model='Lacrosse' then Level=1;
if Model='Lesabre' then Level=1;
if Model='Park Ave' then Level=1;
if Model='CST-V' then Level=2;
if Model='CTS' then Level=2;
if Model='Deville' then Level=2;
if Model='STS-V6' then Level=2;
if Model='STS-V8' then Level=2;
if Model='XLR-V8' then Level=2;
if Model='AVEO' then Level=0;
if Model='Cavalier' then Level=0;
if Model='Classic' then Level=0;
if Model='Cobalt' then Level=0;
if Model='Corvette' then Level=2;
if Model='Impala' then Level=1;
if Model='Malibu' then Level=1;
if Model='Monte Ca' then Level=1;
if Model='Bonnevil' then Level=1;
```

```
if Model='G6' then Level=1;
if Model='Grand Am' then Level=1;
if Model='Grand Pr' then Level=1;
if Model='GTO' then Level=2;
if Model='Sunfire' then Level=0;
if Model='Vibe' then Level=1;
if Model='9_3' then Level=2;
if Model='9_3 HO' then Level=2;
if Model='9_5' then Level=2;
if Model='9_5 HO' then Level=2;
if Model='9-2X AWD' then Level=1;
if Model='Ion' then Level=0;
if Model='L Series' then Level=1;
run;
proc print;
run;
data gmcars;
set gmcars;
drop var13;
drop var14;
drop var15;
drop var16;
drop var17;
drop var18;
drop var19;
drop var20;
drop var21;
drop var22;
drop var23;
```

```
drop var24;
run;
proc print;
run;
proc univariate normal;
var price;
histogram / normal(mu=est sigma=est);
run;
*Transformation;
data gmcars;
set gmcars;
logprice=log(price);
newmileage=mileage/1000;
run;
proc univariate normal;
var logprice;
histogram / normal(mu=est sigma=est);
run;
proc univariate normal;
var newmileage;
histogram / normal(mu=est sigma=est);
run;
*creat dummy variables;
data gmcars;
set gmcars;
Make1=(Make='Cadil');
Make2=(Make='Chevr');
Make3=(Make='Ponti');
Make4=(Make='SAAB');
```

```
Make5=(Make='Satur');
run;
proc print;
run;
data gmcars;
set gmcars;
Standard=(Level=1);
Luxury=(Level=2);
run;
proc print;
run;
data gmcars;
set gmcars;
Type1=(Type='Conve');
Type2=(Type='Hatch');
Type3=(Type='Coupe');
Type4=(Type='Wagon');
run;
proc print;
run;
data gmcars;
set gmcars;
drop mileage;
drop make;
drop Model;
drop trim;
drop type;
run;
proc print data=gmcars;
```

```
run;

*check the significance and moticolinearity of the variables;

proc corr data=gmcars;

var logprice newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type3
Type4 Cylinder Liter Doors Cruise Sound Leather;

run;

proc reg data=gmcars;

model logprice= newmileage Cylinder Doors sound Liter Cruise Leather Type1 Type2 Type3 Type4
standard luxury make1 make2 make3 make4 make5 /vif stb;

run;

*create interaction term and use center method;

data gmcars;

set gmcars;

Cylinder_Liter=Cylinder*liter;

run;

proc corr;

var Cylinder Liter Cylinder_Liter;

run;

data gmcars;

set gmcars;

Cylinder_c=5.25156-Cylinder;

Liter_c=3.02753-Liter;

Cylinder_Liter_c=Cylinder_c*Liter_c;

run;

proc corr;

var Cylinder Liter Cylinder_Liter_c;

run;

*use stepwise method to select the model;

proc reg data=gmcars;
```

```
model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type3
Type4 Cylinder Liter Cylinder_Liter_c Doors Cruise Sound Leather /vif stb selection=stepwise;

run;

*the interaction didn't solve the moticlinearity

*remove the highly colinearity variables and interaction term ;

*check are there any outliers in the model;

proc reg data=gmcars;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4
Liter Cruise Leather /vif stb r influence;

run;

*remove the ourlier;

data gmcarsmodel1;

set gmcars;

if _n_=388 then delete;

if _n_=382 then delete;

run;

proc reg data=gmcarsmodel1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4
Liter Cruise Leather/vif stb r influence;

run;

*the model without outliers;

proc reg data=gmcarsmodel1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4
Liter Cruise Leather/vif stb;

run;

*one variable the VIF still very high, so remove it;

proc reg data=gmcars;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather /vif stb r influence;

run;

*remove outliers;
```

```
data gmcarsmodel1;
set gmcars;
if _n_=388 then delete;
if _n_=741 then delete;
if _n_=742 then delete;
if _n_=743 then delete;
if _n_=744 then delete;
run;
proc reg data=gmcarsmodel1;
model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather/vif stb r influence;
run;
*the model without the outliers;
proc reg data=gmcarsmodel1;
model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather/vif stb;
run;
*Do the residual analysis to see does the model violate any assumptions;
proc reg corr;
 *full model;
 model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather;
 * reduced model ;
 model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather;
 * RESIDUAL PLOT: RESIDUALS VS X-VARIABLES;
 plot student.*predicted.;
 plot npp.*student.;
 run;
quit;
*use backward method;
```

```
proc reg data=gmcars;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type3 Type4 Cylinder Liter Cylinder_Liter_c Doors Cruise Sound Leather /vif stb selection=backward;

run;

*the interaction didn't solve the moticlinearity

*remove the highly colinearity variables and interaction term;

*check are there any outliers in the model;

proc reg data=gmcars;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4 Liter Cruise Leather /vif stb r influence;

run;

*remove outliers;

data gmcarsmodel2;

set gmcars;

if _n_=382 then delete;

if _n_=388 then delete;

run;

proc reg data=gmcarsmodel2;

model logprice=newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4 Cylinder Cylinder_Liter_c Liter Cruise Leather/vif stb r influence;

run;

*the model without outliers;

proc reg data=gmcarsmodel1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Luxury Type1 Type2 Type4 Liter Cruise Leather/vif stb;

run;

proc reg data=gmcars;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter Cruise Leather /vif stb r influence;

run;

*remove outliers;
```

```
data gmcarsmodel1;

set gmcars;

if _n_=388 then delete;

if _n_=741 then delete;

if _n_=742 then delete;

if _n_=743 then delete;

if _n_=744 then delete;

run;

proc reg data=gmcarsmodel1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather/vif stb r influence;

run;

proc reg data=gmcarsmodel1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather/vif stb;

run;

*Validation model;

title "Test and Train Sets for gmcars";

proc surveyselect data=gmcarsmodel1 out=car_all1 seed=495857

samprate=0.80 outall;

run;

data car_train1(where=(selected=1));

set car_all1;

run;

data car_test1(where=(selected=0));

set car_all1;

run;

data car_all1;

set car_all1;

if selected then new_y=logprice;
```

```
run;

proc reg data=car_all1;

model logprice= newmileage Make1 Make2 Make3 Make4 Make5 Standard Type1 Type2 Type4 Liter
Cruise Leather;

output out=outm1(where=(new_y=.)) p=yhat;

run;

data outm1_sum;

set outm1;

d=logprice-yhat;

absd=abs(d);

run;

proc summary data=outm1_sum;

var d absd;

output out=outm1_stats std(d)=rmse mean(absd)=mae;

run;

proc print data=outm1_stats;

run;

proc corr data=outm1;

var logprice yhat;

run;
```

Leanne's code

```
*Import data;

proc import out=car replace

datafile='S:\Final Project\gmcar_price.txt' ;

delimiter='09'x;

getnames=yes;

run;
```

```
proc print;
run;
data car;
set car;
drop var13;
drop var14;
drop var15;
drop var16;
drop var17;
drop var18;
drop var19;
drop var20;
drop var21;
drop var22;
drop var23;
drop var24;
run;
proc print;
run;

proc univariate;
var price;
run;
proc univariate data=car;
var price;
histogram/normal(mu=est sigma=est);
run;

data car;
```

```
set car;

logprice=log(price);

run;


proc univariate data=car;

var logprice;

histogram/normal(mu=est sigma=est);

run;


*Creates a next dataset xv_all - adds a column splitting train and test sets;

title " Test and Train Sets for car";

proc surveyselect data=car out=xv_all seed=495857 samprate=0.8 outall; *outall - show all the data selected (1) and not selected (0) for training;

run;


*dataset xv_all content: selected (1) for Train, Seleted (0) for Test;

proc print;

run;


*create new variable logprice = car for training set, and = NA for testing set;

data xv_all;

set xv_all;

if selected then train_price=price;

logprice=log(train_price);

run;

proc print;

run;


*test frequency;

proc freq;
```

```
tables make;
run;
proc freq;
tables model2;
run;
proc freq;
tables trim;
run;
proc freq;
tables type;
run;
proc freq;
tables cylinder;
run;

*create dummy variables;
data xv_all;
set xv_all;
M1=(Make='Cadil');
M2=(Make='Chevr');
M3=(Make='Ponti');
M4=(Make='SAAB');
M5=(Make='Satur');
run;
proc print;
run;
data xv_all;
set xv_all;
Standard=(Level=1);
```

```
Luxury=(Level=2);
run;
proc print;
run;
data xv_all;
set xv_all;
drop Standard;
drop Luxury;
drop Level
run;
proc print;
run;

data xv_all;
set xv_all;
T1=(Type='Conve');
T2=(Type='Hatch');
T3=(Type='Coupe');
T4=(Type='Wagon');
run;
proc print;
run;
data xv_all;
set xv_all;
drop make;
drop Model2;
drop trim;
drop type;
run;
```

```
proc print data=xv_all;
run;

*initial model2 to check the moticolinearity;
proc corr data=xv_all;
var logprice mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 cylinder liter doors cruise sound leather;
run;

*create interaction term;
data xv_all;
set xv_all;
C_L=Cylinder*liter;
run;
proc corr;
var Cylinder Liter C_L;
run;
data xv_all;
set xv_all;
Cylinder_m=5.26866-Cylinder;
Liter_m=3.03731-Liter;
Cylinder_Liter_m=Cylinder_m*Liter_m;
run;
proc corr;
var Cylinder Liter Cylinder_Liter_m;
run;

*model2 1;
*try different method to see the difference;
proc reg data=xv_all;
```

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 cylinder liter Cylinder_Liter_m doors cruise sound leather/vif stb selection=stepwise;

run;

*stepwise

*keep what's in the result and remove the highly colinearity variable - cylinder, check outliers and rerun the model21;

proc reg data=xv_all;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter Cylinder_Liter_m cruise sound leather/vif stb influence r;

run;

*remove the variable which is insignificant - Cylinder_Liter_m;

proc reg data=xv_all;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise sound leather/vif stb;

run;

*remove the variable which is insignificant - sound;

proc reg data=xv_all;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise leather/vif stb influence r;

run;

*remove outliers - model21;

data car_model21;

set xv_all;

if _n_=743 then delete;

if _n_=744 then delete;

run;

*check if there still has outliers;

```
proc reg data=car_model21;
model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise leather/ vif stb influence r;
run;

*remove outliers - model21;
data car_model21;
set car_model21;
if _n_=742 then delete;
run;

*check if there still has outliers;
proc reg data=car_model21;
model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise leather/ vif stb influence r;
run;

*model21 with 12 variables and no outlier;
proc reg data=car_model21;
model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise leather/ vif stb;
run;


*model2 2;
proc reg data=xv_all;
model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 cylinder liter Cylinder_Liter_m doors cruise
sound leather/vif stb selection=forward;
run;

*forward;
*keep what's in the result and remove the highly colinearity variable - cylinder, check outliers and rerun
the model22;
```

```
proc reg data=xv_all;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 liter Cylinder_Liter_m cruise sound leather/ vif
stb influence r ;

run;


*remove outliers - model22;

data car_model22;

set xv_all;

if _n_=384 then delete;

run;


*check if there still has outliers;

proc reg data=car_model22;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 liter Cylinder_Liter_m cruise sound leather/ vif
stb influence r;

run;


*remove outliers - model22;

data car_model22;

set car_model22;

if _n_=742 then delete;

if _n_=743 then delete;

run;


*check if there still has outliers;

proc reg data=car_model22;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 liter Cylinder_Liter_m cruise sound leather/ vif
stb influence r;

run;
```

```
*remove outliers - model22;

data car_model22;

set car_model22;

if _n_=741 then delete;

run;


*check if there still has outliers;

proc reg data=car_model22;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 liter Cylinder_Liter_m cruise sound leather/ vif
stb influence r;

run;


*model21 with 15 variables and no outlier;

proc reg data=car_model22;

model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t3 t4 liter Cylinder_Liter_m cruise sound leather/ vif
stb influence r;

run;



*model2 testing;

title "Validation - Test and Train Set";

proc surveyselect data=car out=xv_all seed=495857

samprate=0.8 outall; *outall - show all the data selected (1) and not selected (0) for training;

run;


data xv_all;

set xv_all;

if selected then train_price=price;

logprice=log(train_price);

run;
```

```
proc print data= xv_all;
run;

proc reg data=xv_all;
* MODEL21;
model2 logprice = mileage m1 m2 m3 m4 m5 t1 t2 t4 liter cruise leather;
output out=outm1(where=(logprice=.)) p=yhat;
run;

data outm1;
set outm1;
logprice=log(price);
run;
proc print data=outm1;
run;

data outm1_sum;
set outm1;
d=logprice-yhat;
absd=abs(d);
run;
proc summary data=outm1_sum;
var d absd;
output out=outm1_stats std(d)=rmse mean(absd)=mae;
run;
proc print data=outm1_stats;
run;
proc corr data=outm1;
var logprice yhat;
```

```
run;



* MODEL22;
proc surveyselect data=car out=xv_all seed=495857
samprate=0.6 outall; *outall - show all the data selected (1) and not selected (0) for training;
run;

data xv_all;
set xv_all;
if selected then new_price=logprice;
run;
proc print data= xv_all;
run;

proc reg data=xv_all;
model2 logprice = mileage m4 t3 t4 Cylinder_Liter_m doors cruise leather;
output out=outm2(where=(logprice=.)) p=yhat;
run;

data outm2_sum;
set outm2;
d=logprice-yhat;
absd=abs(d);
run;
proc summary data=outm2_sum;
var d absd;
output out=outm2_stats std(d)=rmse mean(absd)=mae;
```

```
run;
proc print data=outm2_stats;
run;
proc corr data=outm2;
var logprice yhat;
run;
```

Minh's code

```
*Import;
data car;
infile 'gmcar_price.txt' firstobs=2 delimiter='09'x MISSOVER;
input price mileage make $ model $ trim $ type $ cylinder liter doors cruise sound leather;
run;

proc print;
run;
*Hold out for cross validation;
proc surveyselect data=car out=xv_all seed=241993
samprate=0.8 outall;
run;

data xv_all;
set xv_all;
if selected then train_price=price;
ln_price=log(train_price);
run;
proc print data=xv_all;
run;
```

```
*test frequency;

proc freq data=xv_all;
tables make;
run;
proc freq data=xv_all;
tables model;
run;
proc freq data=xv_all;
tables trim;
run;
proc freq data=xv_all;
tables type;
run;
proc freq data=xv_all;
tables cylinder;
run;
proc univariate data=xv_all;
var train_price;
histogram/normal(mu=est sigma=est);
run;

proc univariate data=xv_all;
var ln_price;
histogram/normal(mu=est sigma=est);
run;

proc corr data=xv_all;
run;
```

```
*sort by type;
proc sort data=xv_all;
by type;
run;

proc boxplot data=xv_all;
plot ln_price*type;
run;

*sort by make;
proc sort data=xv_all;
by make;
run;

proc boxplot data=xv_all;
plot price*make;
run;

*sort by cylinder;
proc sort data=xv_all;
by cylinder;
run;

proc boxplot data=xv_all;
plot price*cylinder;
run;

proc sort data=xv_all;
```

```
by type;
run;

proc boxplot data=xv_all;
plot cylinder*type;
run;

proc boxplot data=xv_all;
plot doors*type;
run;



proc print data=xv_all;
run;

data pika;
set xv_all;
*all 0 -> Buick;
Make0=0;
if make='Cadillac' then Make0=1;
Make1=0;
if make='Chevrolet' then Make1=1;
Make2=0;
if make='Pontiac' then Make2=1;
Make3=0;
if make='SAAB' then Make3=1;
Make4=0;
if make='Saturn' then Make4=1;
*type;
*all 0 -> Convertible;
```

```
Type0=0;
if type='Coupe' then Type0=1;
Type1=0;
if type='Hatchback' then Type1=1;
Type2=0;
if type='Sedan' then Type2=1;
Type3=0;
if type='Wagon' then Type3=1;
*cylinder;
*0 -> cylinder = 4;
Cylinder0=0;
if cylinder=6 then Cylinder0=1;
Cylinder1=0;
if cylinder=8 then Cylinder1=1;
*doors;
*0 -> 2 doors;
d=0;
if doors=4 then d=1;
Cylinder_Liter=Cylinder*liter;
run;

data pika;
set pika;
drop trim;
drop model;
run;

proc print data=pika;
run;
```

```
*draft model;

*full;

proc reg data=pika;
model ln_price = mileage make0 make1 make2 make3 make4 type0 type1 type2 type3 d
cylinder liter cruise sound leather/stb vif tol;
run;


proc reg data=pika;
model ln_price = mileage make0 make1 make2 make3 make4 type0 type1 type2 type3 d
cylinder liter cruise sound leather/selection=stepwise;
run;

proc reg data=pika;
model ln_price = mileage make0 make1 make2 make3 make4 type0 type1 type2 type3 d
cylinder liter cruise sound leather/selection=backward;
run;

proc reg data=pika;
model ln_price = mileage make0 make1 make2 make3 make4 type0 type1 type2 type3 d
cylinder liter cruise sound leather/selection=forward;
run;

proc reg data=pika;
model ln_price = mileage make0 make1 make2 make3 make4 type0 type1 type2 type3 d
cylinder liter cruise sound leather/selection=adjrsq;
run;
```

```
*New model;

proc reg data=pika;
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise /stb vif tol;
plot student.*predicted.;
plot npp.*student.;
run;


proc reg data=pika;
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise/ influence r;
run;


*remove outliers;
data pika;
set pika;
if _n_=341 then delete;
if _n_=343 then delete;
if _n_=344 then delete;
run;

proc print data=pika;
run;

proc reg data=pika;
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise/ stb;
run;
```

```
*I chose a Cadillac(Make0) Sedan(Type2) 4doors(d=1) liter=3.8 cruise=yes mileage=1000;
data pred;
input mileage Make0 Make3 Type0 Type2 Type3 d liter cruise;
datalines;
1000 1 0 0 1 0 1 3.8 1
;

*I chose a SAAB(Make3) Coupe(Type0) 2doors(d=0) liter=3.1 cruise=no mileage=2000;
data pred;
input mileage Make0 Make3 Type0 Type2 Type3 d liter cruise;
datalines;
2000 0 1 1 0 0 0 3.1 0
;

data predict;
set pred pika;

proc reg data=predict;
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise/p clm cli alpha=0.05;
run;

*validation;
title "Validation - Test Set";
*proc reg data=xv_all;
proc reg data=pika;
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise;
output out=outm1(where=(ln_price=.)) p=yhat;
run;

data outm1;
```

```
set outm1;
ln_price=log(price);
run;

proc print data=outm1;
run;


/* summarize the results of the cross-validations for model-1*/
title "Difference between Observed and Predicted in Test Set";
data outm1_sum;
set outm1;
d=ln_price-yhat; *d is the difference between observed and predicted values in test set;
absd=abs(d);
run;
/* computes predictive statistics: root mean square error (rmse)
and mean absolute error (mae)*/
proc summary data=outm1_sum;
var d absd;
output out=outm1_stats std(d)=rmse mean(absd)=mae ;
run;

proc print data=outm1_sum;
run;

proc print data=outm1_stats;
title 'Validation statistics for Model';
run;
*computes correlation of observed and predicted values in test set;
proc corr data=outm1;
```

```
var ln_price yhat;
run;


*5-fold validation;
title "5-fold crossvalidation + 25% testing set";
proc glmselect data=pika
plots=(asePlot Criteria);
partition fraction(test=0.25);
model ln_price = mileage Make0 Make3 Type0 Type2 Type3 d liter cruise/
selection=stepwise(stop=cv) cvMethod=split(5) cvDetails=all stb;
run;
```