Cribs

Regression modeling for home price

CSC 423 WINTER 2017

Jun Tae Son, Omer Saif Cheema, Yusheng Zhu

Table of Contents

1. Abstract..........3
2. Introduction..........3
3. Methodology..........3
4. Analysis, Results and Findings..........6
5. Future Work..........13
6. Conclusion..........13
Appendix..........14
Code..........35
References..........75

1. Abstract:

The dataset we used in the analysis had historical data on house sales in King County, Washington from May 2014 to May 2015. The goal of the analysis is to generate the best model for predicting home sale price in King county. We included the following features in the analysis; date of house sales, location of the property, the number of bedrooms and bathrooms, the size of the property, floors, waterfront, view index, conditions of house, levels of construction and design, renovation history and basement. We analyzed how influential these features are on house price and conducted prediction test using our final model. We assumed that the prices of the houses are only affected by the variables used in the data set and factors like interest rates, property taxes other political and economic factors stay constant. Our final model indicates that house prices in King County are most influenced by interior area of a house, the location of a house and whether a house has a basement or not. Finally, we are excited to declare that the predictors in our final model can explain over 75 percent of variation in house prices in King County.

2. Introduction:

In this report, we built a multivariate regression model of house prices using dataset composed of 21,613 houses in King County, Washington. The goal of the analysis is to generate the best fitted regression model for predicting home sale price in King county. Before starting the analysis, we hypothesized that location of the house, number of bedrooms and the quality of construction of a house are the biggest predictors of house prices. Also, during the analysis we assume that the prices of the houses are only affected by the variables used in the data set and factors like interest rates, property taxes other political and economic factors stay constant. Finally, you can see the name and the description of the variables from the data set in the appendix (see table 1). Below is the methodology we used to find the best regression model for predicting house prices in King County.

3. Methodology:

The data set used in the analysis was obtained from Kaggle.com and you can download it from the following link: https://www.kaggle.com/harlfoxem/housesalesprediction. The data set is composed of 21 attributes and 21,613 rows of observations. The data was imported by using the Proc Import function in SAS and then, 1500 rows were randomly taken out of the original data set using different seed values by all three members.

3.1 Preprocessing:

- **Recode Qualitative Covariates and Create Dummy Variables:**
    A. **DATE:** Every instance under the date features in the original dataset had 15 digits. We reduced the number of digits for every instance from 15 to 8 to make YYYYMMDD format, and then we assigned each date to one of the four quarters in a year; first, second, third and fourth quarter. Finally, by making Q1 as the base level, we created three dummy variables: Q2, Q3 and Q4 (see table 2 in the appendix).

B. **Location:** The location of each house purchased in the dataset was given in terms of its coordinates. Firstly, we calculated the medians for longitude and latitude of all the instances. Based on the median values for longitude and latitude, we divided the section from north(N) to south(S), and west(W) to east(E). We assigned N and W as the base level (where S=0 and E=0), and generated three location dummy variables NE, SW, and SE. Please see table 3 in the appendix for more detail.

C. **Grade:** The variable grade in the dataset could have a value from 1 to 13. We divided the grade index into four sections: 1-3 as short of building construction and design, 4-6 as below average level of construction and design, 7-10 as above average level of construction and design, and 11-13 as high-quality level of construction and design. (see table 4 for more details in the appendix).

D. **View:** The view index in the dataset had five grades from 0 to 4. So, we created binary dummy variable of view_good which had a value of 1 when the view index was 3 or 4, and a value of 0 otherwise.

E. **Floor:** The range of floor variable in the dataset was between 1 and 3.5. We calculated the mean value of the floor (1.5) and created a dummy variable, floor_h. If the floor>1.5 then floor_h=1.

F. **Condition:** The condition index in the dataset had five grades from 1 to 5. So, we created the dummy variable condition_good which had a value of 1 when the condition feature had a value greater than 4, and a value of 0 otherwise.

G. **Renovated:** The original dataset had yr_renovated variable which contained renovation history for the specific property. We created binary dummy variable 'renovated' which had a value of 1 when the property has a renovation record.

H. **Basement:** The original dataset had sqft_basement, the size of basement. We created a binary dummy variable 'basement' which had a value of 1 when the property had a basement.

- **Transformation of Variables:**
  We applied transformation method on dependent and independent variables that violated one of regression assumptions: linearity, constant variance, independence, and normality. We transformed dependent variable 'price' into 'ln_price', and independent variables 'sqft_living', 'sqft_lot' and 'sqft_above' into 'ln_sqft_living', 'ln_sqft_lot' and 'ln_sqft above' using log transformation. Please see the next section for more details.

3.2 Model Approach:

A. **Multicollinearity:** After the preprocessing, we created the Pearson correlation coefficient table to see the relationship among predictors, and ran the regression model with VIF option to check whether there was multicollinearity between independent variables. We removed predictors with multicollinearity problem if we found any.

B. **Interactive Variables:** We created the interactive variables by combining two different predictors that seemed to have a joint effect. If there was multicollinearity between the interactive variables and its parts, we centered the quantitative part(s) of the interactive variables and created new centered quantitative variable(s) and new centered interactive variables.
C. **Outliers and Influential Points:** We ran the regression model with option "r" and "influence" to check whether there were any outliers and influence points or not. If there were outliers and influence points, we kept on removing them until the R-square and Adj R-square stopped increasing.
D. **Data splitting:** After removing outliers and influence points, we split the sample into a training and testing set using 75/25 ratio and created a separate training set and a testing set.
E. **Model selections:** we ran two different selection methods on our full regression model to find the best fitted model. If two different models were given by the selection methods, we choose the one with the least number of predictors or/and with the highest adj R-square.
F. **Prediction test using two observations:** We copied the first two rows of predictors from the training set and merge them with the testing set and created a new dataset called predict. Then we ran the regression model with that dataset and calculated the predicted price, confidence interval, and prediction interval of all the data.

3.3 Validation Approach:

A. **Model validity test:** Based on the model selection results on training set, each team member generated their own regression model. And we conducted goodness of fit test and compared the number of predictors, RMSE, and adjusted R-square values.
B. **Predictive Performance Test:** Each team member used their train/test split dataset created previously and added a new response variable column called new_y. Then we added the observed values from ln_price values from the training set in the new_y and left the new_y values for test data empty. Then we ran the regression model for training set and predicted values on the test set. Finally, we calculated the R-square, adj. R-square, RMSE, MAE, cross validated R-square for all three models and selected the final model that showed the best performance.

3.4 Bonus Analysis (Partial Correlation) (Omer):

The Pearson correlations shows that some of the predictors are highly correlated with the response. However, as we have seen, some predictors are also highly correlated with other predictors, making it difficult to determine which predictors are actually important. Partial correlations allow us to see correlations between each predictor and the response, after adjusting for the other predictors. The partial methodology was copied from the report called Housing Prices Multiple Regression – Multicollinearity and Model Building written by M. Smith. You can access the report from the link given in the references and you can see that Omer used partial correlation by looking at his code. The Pearson correlation showed that the correlation of ln_price against ln_sqft_above, ln_sqft_lot and ln_sqft_above is 0.654, 01549 and 0.5632. That shows that ln_price has positive moderate relationship with ln_sqft_above; positive weak relationship with ln_sqft_lot and positive moderate relationship with ln_sqft_above. However, partial correlation for ln_price against ln_sqft_above, ln_sqft_lot and ln_sqft_above is 0.4026, -0.063 and -0.01033. This means that the true relationship between ln _sqft_living is still positive and moderate but the relation between ln_sqft_lot and ln_sqft_above is actually weak and negative (see figure 1 for more detail)

4. Analysis, Results and Findings:

4.1 Regression Assumption

The response variable in the initial regression model was price. The independent variables in the model were bathrooms, bedrooms, sqft_living, sqft_lot, sqft_above, floor_h, waterfront, view_good, condition_good, grade_b, grade_a, grade_h, renovated, basement, NE, SW, SE, Q2, Q3, Q4.

```
PROC REG;
model price =bathrooms bedrooms sqft_living sqft_lot sqft_above
    floor_h waterfront view_good condition_good grade_b grade_a
    grade_h renovated basement NE SW SE Q2 Q3 Q4;
run;
```

However, distribution of the dependent variable in the initial model was positively skewed as you can see in the figure 2 in the appendix. Since the normality assumption was violated (see figure 3), we decided to create a new dependent variable, ln_price, by taking the natural log of the variable price. By doing that the dependent variable in the model became normally distributed (see figure 4 and 5). According to the scatterplot matrix of dependent variables and continuous predictors (see figure 6), sqft_living and sqft_above appear to have moderate positive linear relationship with ln_price. And we couldn't find linearity relationship between dependent variable and sqft_lot. Residual plots of all three continuous predictors showed a discernible pattern (see figure 7). To solve the constant variance and independence problem, we applied log transformation and created new predictors: ln_sqft_living, ln_sqft_lot, and ln_sqft_abov. These new variables didn't violate constant variance and independence assumption and improved linearity with ln_price (see figure 8).

After the transformation of the dependent variable and the three continuous variables the initial regression model looked like below:

```
PROC REG;
model ln_price =bathrooms bedrooms ln_sqft_living ln_sqft_lot
ln_sqft_above floor_h waterfrontview_good condition_good grade_b
grade_a grade_h renovated basement NE SW SE Q2 Q3 Q4;
run;
```

4.2 Multicollinearity

There are two ways of determining multicollinearity among the independent variables.

- If correlation coefficient value of two independent variables has more than 0.9, we may conclude that there is multicollinearity problem.
- If two independent variables have higher than 10 variance influence, or lower than 0.1 tolerance value, then we may conclude that there is multicollinearity problem.

In this analysis, we generated Pearson correlation coefficient table and calculated variance influence to detect multicollinearity among the predictors. The correlation coefficient value between ln_sqft_living and ln_sqft_above was about 0.9 and VIF values for ln_sqft_living and ln_sqft_above were also higher than 10. We also found multicollinearity problems on grade dummy variables. Grade_b and grade_a had a high correlation value close to 0.9 and VIF values for all three grade dummy variables were larger than 10 (see figure 9). To solve the multicollinearity from our model, we took out one of the variables with multicollinearity and rerun the model to see how it changed (see table 5)

4.3 Interaction variables

In Jun's model, he created the interactive variables between the location of the house and the size of the living area (minus the basement). In the model he created three new interactive variables, above_ne, above_sw and above_se, by multiply ln_sqft_above with NE, SW and SE. However, when he rerun the regression model it showed that there was multicollinearity (VIF was greater than 10) between the interactive variables and location variables (NE, SW and SE) (See figure 8). To fix the multicollinearity problem, we centered ln_sqft_above and created a new variable called ln_sqft_above_c. Afterwards, we created new interactive variables by multiplying NE, SW and SE with ln_sqft_above_c and the new interactive variables were not multicollinear with NE, SW and SE anymore (see figure 10 & 11).

Omer also made the interactive variables between location of the house but he used the complete interior area of the house (ln_sqft_living) (including the basement). In his model he created three new interactive variables, ln_sqft_living_NE, ln_sqft_living_SW and ln_sqft_living_SE, with NE, SW and SE. However, just like Jun's interactive variable, Omer's variables were also multicollinear with location variables because their VIF value was greater than 10. He fixed the multicollinearity problem by centering ln_sqft_living (ln_sqft_living_c) and created centered interactive variables: ln_sqft_living_NE_c, ln_sqft_living_SW_c and ln_sqft_living_SE_c(look at figure 12 and 13 for more detail).

Yusheng create 7 interaction variables in her model. 3 interaction variables were made between ln_sqft_living and the 3 location dummy variables---by multiplying ln_sqft_living with NE, SW and SE respectively, based on the assumption that the Square footage of the apartments interior living space and the house location have a joint effect on the house value. Same as Jun and

Omer, it turned out that the VIF value of the 3 interaction variables appear to be much higher than 10 in her model. To fix this multicollinearity problem, she centered the ln_sqft_living into ln_sqft_living_c by subtracting the mean from the original data and recomputed the interaction variables--ln_sqft_living_NE_c, ln_sqft_living_SW_c and ln_sqft_living_SE_c in the new model, and the VIF of these interaction terms have all dropped to a value below 10 afterwards(see figure 14 & 15).

Apart from the joint effect assumption , Yusheng made another assumption that interaction terms should be added between predictors whose correlation coefficients is greater than 0.5, From the Pearson correlation coefficient table (Figure 16 & 17), 4 pairs : Cov (bathrooms, bedrooms) = 0.53987, Cov (bedrooms, ln_sqft_living) = 0.64290, Cov(bathrooms,ln_sqft_living) = 0.74995, Cov (bathrooms,floor_h)= 0.6007 can be found to satisfy the assumption .Thus, creating another 4 interaction variables accordingly, and then adding each interaction variable to the model one by one checking the P-value to test if each interaction variable is significant enough to stay in the model. Next, using the same mean centering method on the continuous variables to fix the multicollinearity problem.

4.4 Outliers and Influential points

An outlier is an observation that lies an abnormal distance from other values in a random sample from a population. There is an outlier in a dataset when its studentized residual values is greater than or less than +3 or -3.

Influence point is an observation which has excessive influence on the fit of the regression model. We used Cook's Distance to identify influential points in this dataset. In this model an observation is an influential point if it's Cook's Distance was greater than 0.0003. See table 6 to look at how many observations each member took out.

4.5 Splitting the data to training set and testing:

Before running the selection methods to find the best model for each member, we split our sample dataset to a training and testing set. 75% of the sample data was randomly assigned to a training set and the rest of the 25% data was stored in a testing set. Then each teammate ran the selection methods on the training set to produce the final model for each member.

4.6 Model Selection Methods

Jun: (see table 7)

Jun used Backward and Stepwise method to get the best set of predictors in the final regression model. After applying two selection methods, he removed 5 predictors but Adj-R-square and RMSE stayed the same (0.7307 and 0.2357 respectively). The possible reason for this is because he already got the model with low variance by eliminating 153 outliers and influential points in the previous section.

But when Jun ran the testing model using the set of 16 predictors driven from backward and stepwise selection methods, 'renovated' and 'NW' variables turned out to be not significant. Thus, he decided to remove those two insignificant predictors and finalized his model.

```
proc reg data=house_train;
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good
grade_b grade_h basement SW SE Q4 above_NE_c above_SW_c above_SE_c/stb.
run;
```

Omer:

Before running the two different selection methods, Omer had 21 predictors in his regression model. Running the Stepwise method gave Omer 14 predictors on training set and running the ADJRSQ on training set gave omer 16 predictors (see Table 8 for more detail about the selection methods and the final model). The final model Omer selected had 14 predictors given by the Stepwise method. However, when Omer ran his final model with the testing set 5 predictors became insignificant (as you can see in Figure 16). He had to take out variables, bathrooms, bedrooms, waterfront, NE and Q2. After taking out the variables Omer's real final model had only 9 predictors, as you can see below:

```
*Testing Final model after taking out bathrooms, bedrooms, waterfront, NE and Q2;
TITLE 'checking final model with testing set after taking out Q2';
PROC REG data=houseP_test;
model ln_price=ln_sqft_living_c view_good condition_good
    grade_b grade_a renovated SW SE ln_sqft_living_SE_c/stb vif;
run;
title;
```

Yusheng:

Before the selection method, Yusheng had 25 predictors in total. Running the forward selection method gave Yusheng 15 predictors on training set and running the Cp method on training set gave Yusheng 17 predictors.(See table 9 for more detail about the selection methods and the final model), Yusheng chose the model of Cp method as her final model because it has a better R square than another model. Then, she removed bathrooms_c and condition_good from her final model as they are not significant variables on training set.

However, when Yusheng ran her final model on testing set, 6 predictors became insignificant, She had to take out 6 insignificant variables, bedrooms_c, grade_b, grade_a, renovated, NE and Q2. So Yusheng had only 9 predictors remaining in her final model.

```
* remove insignificant predictors in testing set;
* remove bedrooms_c,grade_b,grade_a,renovated,NE,Q2;
title Final Model on testing;
proc reg data=house_test;
model ln_price = ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c bedliving_c living_SE_c/vif stb;
run;
```

4.7 Validation and Performance Diagnostics on the Regression Model
Based on the model selection analysis, each team member generated the following regression models:

Model 1 (Jun): ln_price = 13.3336 - 0.0746*bedrooms - 0.6983*ln_sqft_above_c + 0.7534*waterfront + 0.3226*view_good +0.1536*condition_good - 0.1123*grade_b + 0.4532*grade_h + 0.3768*basement - 0.5468*SW - 0.3844*SE - 0.1054*Q4 - 0.2190*above_NE_c + 0.2811*above_SW_c - 0.2688*above_SE_c

Model 2 (Omer): ln_price = 13.6272 - 0.6689*ln_sqft_living_c + 0.3597*view_good + 0.15775*condition_good - 0.5265*grade_b - 0.4731*grade_a + 0.1918*renovated - 0.4127*SW - 0.3761*SE - 0.2520*ln_sqft_living_SE_c

Model 3 (Yusheng): ln_price = 13.16315 - 0.6982*ln_sqft_living_c + 0.6044*waterfront + 0.4120*view_good - 0.4693*SW - 0.4133*SE - 0.0906*bb_c +0.2633*bathliving_c + 0.1038*bedliving_c - 0.2716*living_SE_c

To compare model validity and predictive performance, we compared the above fitted models as follows.
<Model validity test on training set>

	M1	*M2*	*M3*
The number of predictors	14	9	9
Goodness of Fit	p<0.001	p<0.001	p<0.001
RMSE	0.23755	0.29616	0.30424
R square	0.7301	0.6762	0.6637
Adjusted R square	0.7263	0.6743	0.6610

Based on Goodness of Fit test, we may conclude that all three models fitted the set of observations well enough. Even though the number of predictors on model 1 was higher than the rests, model 1 had the lowest RMSE and the highest adjusted R square value.

<Predictive performance test on testing set>

	M1	*M2*	*M3*
RMSE	0.2410	0.2993	0.2722
MAE	0.1962	0.2436	0.2209
R square	0.7665	0.6933	0.7179
Adjusted R square	0.7563	0.6857	0.7109
Cross-validated R square	0.0262 (<0.3)	0.0153 (<0.3)	0.0494 (<0.3)

Cross-validating statistics for all three models were less than 0.3, indicating that all of them were good for prediction. However, M1 had the lowest root mean squared error and mean absolute error. Adjusted R square for M1 was the highest among the regression models: 75.63% of the variability in home sale price can be explained by M1. Since M1 performed better on prediction, we decided to use M1 as our final regression model.

4.8 Interpretation of influential parameters

In this section, we are going to interpret regression coefficients of three predictors in the final model that had the strongest influence on home sale price (see Figure 19) When we determine the strongest predictor on the response variable, we need to compare standard estimates among predictors. In the final model, ln_sqft_above_c, SW, and basement variables were determined as the most influential predictors on the response because they had the highest standard estimates.

Sqft_above: Interpretation of parameter estimate for ln_sqft_above_c was complicated for the following reasons

1. We applied log transformation on both home sale price and sqft_above because distribution of home sale price was not normally distributed, and residual plot for sqft_above was not randomly scattered.
2. Price and sqft_above does not have linear relationship.
3. We centered ln_sqft_above to solve multicollinearity problem with interaction variables.

When we assigned sqft_above as 1, home sale price decreased by 99.43%. But whenever we input higher sqft_above value, the percentage of price significantly increased. And the percentage of price change became positive when sqft_above reached 1,654. From this analysis, we could determine that home sale price starts to increase if the size of living space without basement is larger than 1,654 sqft and the property is located in north west part of King county (see figure 20).

100*(e^(-0.6983*(7.4107-ln(1)))-1) = -99.43%

100*(e^(-0.6983*(7.4107-ln(1654)))-1) = 0.01%

SW: We cannot interpret SW variable without considering the interaction variable above_SW because above_SW is the interactive variable of SW. When we increase the sqft_above in the SW region by 1, the home sale price increased by 660.68%. And the percentage of price change became negative when sqft_above reached 473. Therefore, we may conclude that the effect on SW variable on home sale price starts to decrease if the size of living space without basement is larger than 473 sqft and the property is located in south west part of King county (see figure 21).
100*(e^(-0.5468*1)-1) + 100*(e^(0.2811*(7.4107-ln(1))*1)-1)= - 42.12% + 702.80% = 660.68%
100*(e^(-0.5468*1)-1) + 100*(e^(0.2811*(7.4107-ln(473))*1)-1)= - 42.12% + 42.08% = - 0.04%

Basement: basement is positively associated with home sale price. Assuming all other variables constant, home sale price increases by 45.76% if the property has a basement.
100*(e^0.3768 -1) = 100*(1.4576-1)=45.76%

4.9 Prediction results of two observations

We copied first two observations from training set and created new dataset called "pred". And we combined pred data and testing set to predict the home sale price for the two instances. The result is shown as below.

Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean		95% CL Predict		Residual
1	12.1495 .	12.5380	0.0522	12.4353	12.6407	12.0624	13.0136	-0.3885 .
2	12.7038 .	13.1537	0.0454	13.0643	13.2430	12.6808	13.6266	-0.4499 .

Since the actual values of ln_price fall in prediction interval, we may determine that the predictions were correct for both instances. Now we want to see the predicted value of price, thus we converted ln_price to price by applying inverse function of log().

Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean		95% CL Predict		Residual
1	$188,999.57	$278,730.32				$173,234.25	$448,471.31	.
2	$328,995.71	$515,916.39				$321,515.17	$827,860.55	.

- Predicted value of the first observation is $278,730.32 and prediction interval is in range between $173,234.25 and $448,471.31.
- Predicted value of the second observation is $328,995.71 and prediction interval is in rage between $321,515.17 and $827,860.55.

Although actual ln_price and predicted ln_price appeared to be similar, we noticed that actual prices in dollar deviated significantly from the predicted values. Therefore, we need further studies on real estate industry such as effects on global economy, regulatory from government, education institutions in the region, etc.

5. Future work:

Since the log transformation has caused the inaccuracy on model predictions, we may consider trying more transformation methods to improve the model's accuracy in the future. Unavoidably, there were some odd revelations about the influence of bedrooms on house prices in our model, which is violating our hypothesis mentioned in the introduction of this report. we want to investigate on a whole dataset or other large housing dataset to confirm this finding. Besides, latitude and longitude analysis is currently beyond the scope of this class, we may need to use third party software like tableau to analyze latitude and longitude in the future.

6. Conclusion

In this project, we presented a process of building a multivariate regression model for a simplified problem of estimating housing prices in King County, Washington. From the above model diagnostics, we could conclude that all the predictors in our final model are significant and there is no problem from the residual examinations. So, it is a valid model. Except for the intercept predictor, there are 14 predictors in the model. The model can explain 75.63% variations of the housing prices. Also, Our model has been proved to have good prediction performance with its cross validated R square being 0.0262 below to 0.3 . Especially, the predictor number of bedrooms with a very small coefficient and standardized estimates doesn't seem to have significant influence on home value. To conclude, the housing prices are closely related to the size of the house, the condition of construction and design of the house, house location and with or without basement. The number of bedrooms also affects the home prices but to a very limited degree.

However, we should also be aware that this is a simplified model and we only considered the information provided in this dataset alone. According to the hedonic pricing model suggested in our references, more reliable analysis should include both internal factors such as number of stories, heating/AC system, and age of the house, and external factors such as property taxes, school district, and air quality.

Appendix:
All relevant output should be included here and cross referenced in your Analysis, Results & Findings section.
Table 1:

Dependent variable	Description
Price	Price of each home sold

Independent variable	Description
ID	Unique ID for each home sold
Date	Date of the home sale
Bedrooms	Number of bedrooms
Bathrooms	Number of bathrooms, where 0.5 counts for a room with a toilet but no shower
Sqft_living	Square footage of the apartments interior living space
Sqft_lot	Square footage of the land space
Floors	Number of floors
Waterfront	A dummy variable for whether the apartment was overlooking the waterfront or not
View	An index from 0 to 4 of how good the view of the property was
Condition	An index from 1 to 5 on the condition of the apartment
Grade	An index from 1 to 13, where 1-3 falls short of building construction and design, 7 has an average level of construction and design, and 11-13 have a high-quality level of construction and design
Sqft_above	The square footage of the interior housing space that is above ground level

Sqft_basement	The square footage of the interior housing space that is below ground level
yr_built	The year the house was initially built
yr_renovation	The year of the house's last renovation
zipcode	What zipcode are the house is in
lat	Latitude
long	Longitude
Sqft_living15	The square footage of interior housing living space for the nearest 15 neighbors
Sqft_lot15	The square footage of the land lots of the nearest 15 neighbors

Table 2:

	Q2	Q3	Q4
First Quarter	0	0	0
Second Quarter	1	0	0
Third Quarter	0	1	0
Fourth Quarter	0	0	1

Table 3:

	NE	SW	SE
S =0 and E=0	0	0	0
S =0 and E=1	1	0	0
S=1 and E=0	0	1	0
S=1 and E=1	0	0	1

Table 4:

	grade_b	grade_a	grade_h
grade (1 to 3)	0	0	0
grade (4 to 6)	1	0	0
grade (7 to 10)	0	1	0
Grade (11 to 13)	0	0	1

Figure 1 (Omer's Partial Correlation)

Pearson Correlation Coefficients, N = 1500 Prob > |r| under H0: Rho=0				
	ln_price	ln_sqft_living	ln_sqft_lot	ln_sqft_above
ln_price	1.00000	0.65433 <.0001	0.15499 <.0001	0.56329 <.0001
ln_sqft_living	0.65433 <.0001	1.00000	0.30850 <.0001	0.87096 <.0001
ln_sqft_lot	0.15499 <.0001	0.30850 <.0001	1.00000	0.32287 <.0001
ln_sqft_above	0.56329 <.0001	0.87096 <.0001	0.32287 <.0001	1.00000

Pearson Partial Correlation Coefficients, N = 1500 Prob > |r| under H0: Partial Rho=0		
	ln_price	ln_sqft_living
ln_price	1.00000	0.40626 <.0001
ln_sqft_living	0.40626 <.0001	1.00000

Pearson Partial Correlation Coefficients, N = 1500 Prob > |r| under H0: Partial Rho=0		
	ln_price	ln_sqft_lot
ln_price	1.00000	-0.06354 0.0139
ln_sqft_lot	-0.06354 0.0139	1.00000

Pearson Partial Correlation Coefficients, N = 1500 Prob > |r| under H0: Partial Rho=0		
	ln_price	ln_sqft_above
ln_price	1.00000	-0.01033 0.6895
ln_sqft_above	-0.01033 0.6895	1.00000

Figure 2 (Histogram) (Before transformation of price):

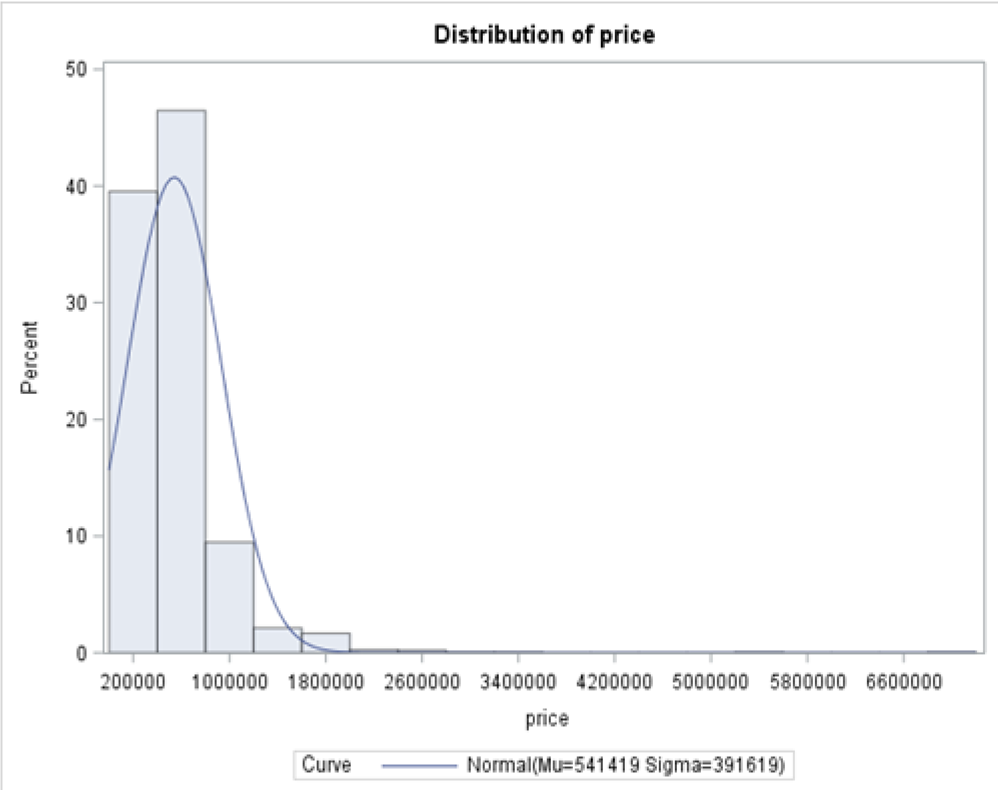


Figure 3 (NNP) (Before transformation of price):

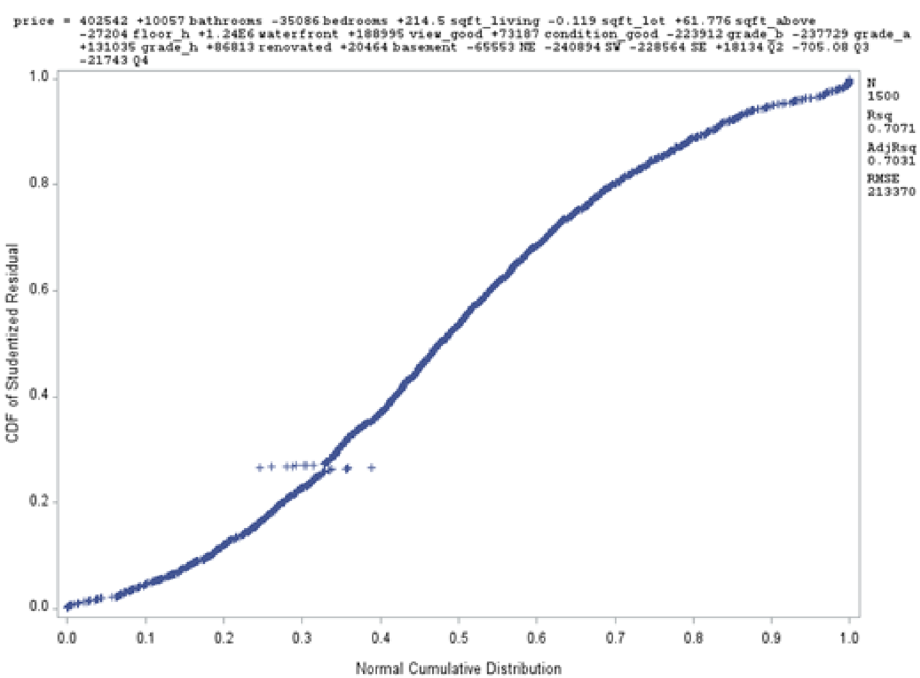


Figure 4 (Histogram) (After transformation of price to ln_price):

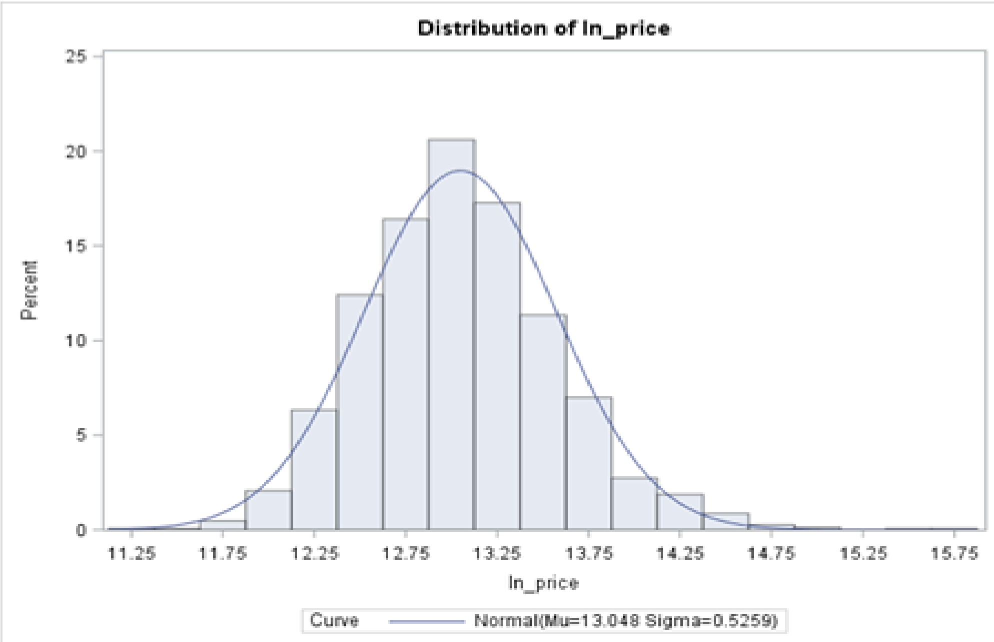


Figure 5 (NNP) (After transformation of price to ln_price):

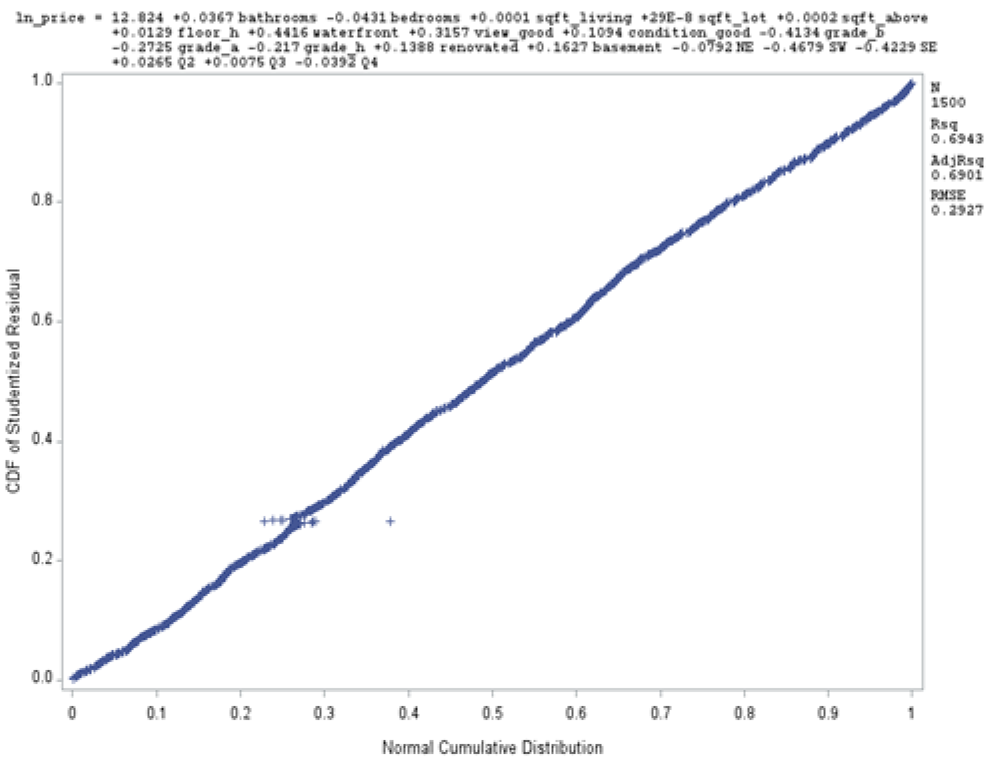


Figure 6 (Scatter plot matrix of ln_price against sqft_living, ln_sqft_living, sqft_lot, ln_sqft_lot, sqft_above, ln_sqft_above):

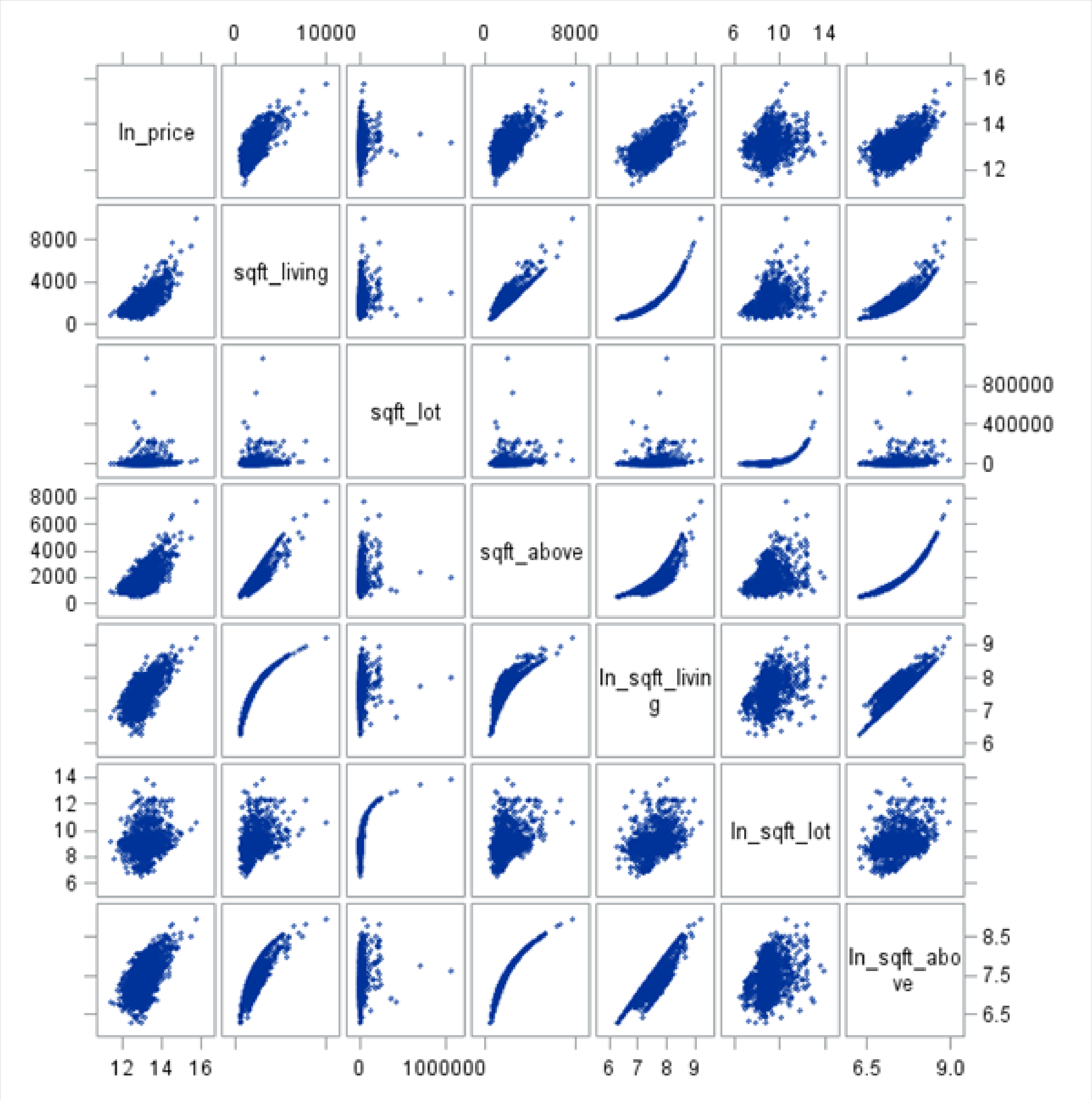


Figure 7 (Studentized Residual of sqft_lot, sqft_above and sqft_living):

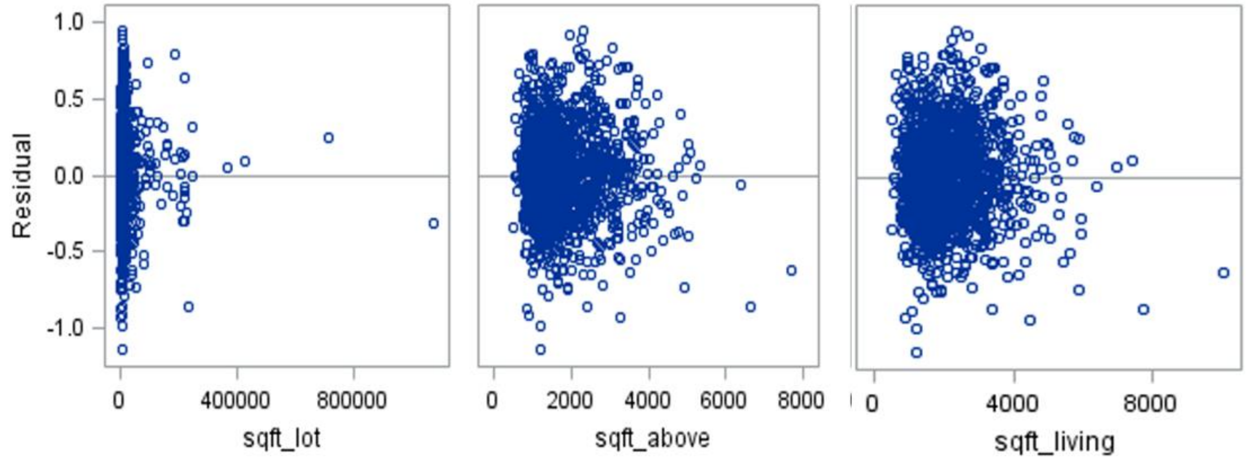


Figure 8 (Studentized Residual of ln_sqft_lot, ln_sqft_above and ln_sqft_living):

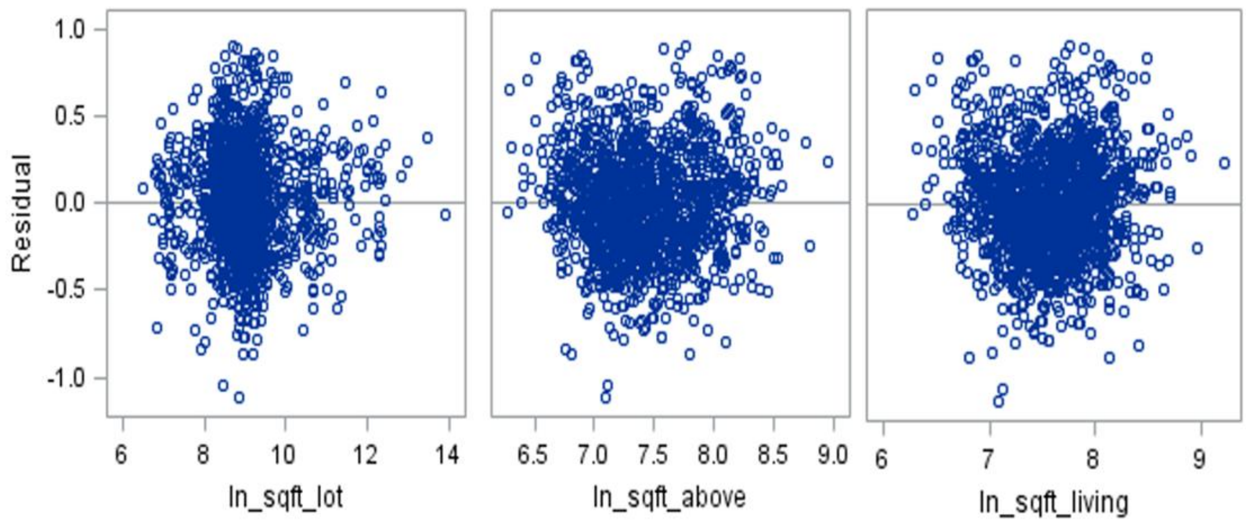


Figure 9 (Multicollinear variables):

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	8.39111	0.37662	22.28	<.0001	0	0
bathrooms	1	0.03855	0.01901	2.03	0.0427	0.05494	3.39315
bedrooms	1	-0.05085	0.01131	-4.50	<.0001	-0.08779	1.76245
ln_sqft_living	1	0.34843	0.07950	4.38	<.0001	0.27898	18.73892
ln_sqft_lot	1	0.00825	0.01104	0.75	0.4552	0.01403	1.63199
ln_sqft_above	1	0.37033	0.07790	4.75	<.0001	0.29872	18.26104
floor_h	1	0.02289	0.02481	0.92	0.3563	0.02135	2.47633
waterfront	1	0.57510	0.11616	4.95	<.0001	0.07968	1.19819
view_good	1	0.31097	0.04611	6.74	<.0001	0.11020	1.23478
condition_good	1	0.10497	0.01751	5.99	<.0001	0.09581	1.18166
grade_b	1	-0.64175	0.30235	-2.12	0.0340	-0.37891	147.40875
grade_a	1	-0.57728	0.30238	-1.91	0.0564	-0.37012	173.84572
grade_h	1	-0.22742	0.30962	-0.73	0.4627	-0.06346	34.52347
renovated	1	0.13142	0.04026	3.26	0.0011	0.04899	1.04177
basement	1	0.15484	0.03409	4.54	<.0001	0.14257	4.55776
NE	1	-0.08258	0.02525	-3.27	0.0011	-0.06479	1.81451
SW	1	-0.48057	0.02321	-20.70	<.0001	-0.37494	1.51725
SE	1	-0.43954	0.02306	-19.06	<.0001	-0.38602	1.89748
Q2	1	0.03738	0.02279	1.64	0.1011	0.03283	1.85217
Q3	1	0.00366	0.02339	0.16	0.8755	0.00309	1.79921
Q4	1	-0.03938	0.02405	-1.64	0.1018	-0.03187	1.75290

Table 5 (Multicollinear variables removed):

	Jun	Omer	Yusheng
The first variable each member removed	grade_a: It has the highest VIF value (173.84572)	Grade_h: It didn't had the biggest VIF (19. 12) out of all grade dummy variables but it was the only one that was insignificant.	ln_sqft_above: It has the highest coefficient with ln_sqft_living (0.87) in the Pearson correlation coefficient table(see at figure 13).
Rerun the model			
The second variable each member removed	ln_sqft_living: It has the highest VIF value (18.73723)	ln_sqft_above: it has the highest VIF value (19.59)	Didn't remove second variable that has multicollinearity,but created interaction terms between pairs whose correlation coefficient is greater than 0.5.

Figure 10 (Multicollinearity between locations and interactive variables - Jun's model):

NE	1	-0.98076	0.42302	-2.32	0.0206	-0.76947	526.01499
SW	1	1.50962	0.44492	3.39	0.0007	1.17778	575.41293
SE	1	-2.02487	0.39141	-5.17	<.0001	-1.77830	564.27863
Q2	1	0.03128	0.02243	1.39	0.1634	0.02747	1.85275
Q3	1	-0.00015690	0.02303	-0.01	0.9946	-0.00013230	1.80039
Q4	1	-0.03450	0.02367	-1.46	0.1451	-0.02792	1.75190
above_NE	1	0.12069	0.05680	2.12	0.0338	0.72373	554.07246
above_SW	1	-0.27541	0.06135	-4.49	<.0001	-1.55578	573.49987
above_SE	1	0.21326	0.05303	4.02	<.0001	1.40986	586.92954

Figure 11 (No Multicollinearity after centering ln_sqft_above_c -Jun's model):

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	13.00881	0.11171	116.45	<.0001	0	0
bathrooms	1	0.05282	0.01832	2.88	0.0040	0.07528	3.25660
bedrooms	1	-0.03928	0.01092	-3.60	0.0003	-0.06782	1.69892
ln_sqft_lot	1	0.01634	0.01075	1.52	0.1289	0.02779	1.59764
ln_sqft_above_c	1	-0.61638	0.05004	-12.32	<.0001	-0.49719	7.78024
floor_h	1	-0.00047173	0.02434	-0.02	0.9845	-0.00043987	2.46048
waterfront	1	0.62725	0.11502	5.45	<.0001	0.08691	1.21284
view_good	1	0.36256	0.04580	7.92	<.0001	0.12848	1.25805
condition_good	1	0.11594	0.01728	6.71	<.0001	0.10582	1.18773
grade_b	1	-0.10559	0.02813	-3.75	0.0002	-0.06234	1.31704
grade_h	1	0.28316	0.05946	4.76	<.0001	0.07901	1.31463
renovated	1	0.15590	0.03970	3.93	<.0001	0.05812	1.04591
basement	1	0.26620	0.02055	12.96	<.0001	0.24511	1.70912
NE	1	-0.08633	0.02578	-3.35	0.0008	-0.06773	1.95312
SW	1	-0.53139	0.02507	-21.20	<.0001	-0.41459	1.82621
SE	1	-0.44447	0.02329	-19.08	<.0001	-0.39034	1.99807
Q2	1	0.03128	0.02243	1.39	0.1634	0.02747	1.85275
Q3	1	-0.00015690	0.02303	-0.01	0.9946	-0.00013230	1.80039
Q4	1	-0.03450	0.02367	-1.46	0.1451	-0.02792	1.75190
above_NE_c	1	-0.12069	0.05680	-2.12	0.0338	-0.04982	2.62554
above_SW_c	1	0.27541	0.06135	4.49	<.0001	0.09448	2.11505
above_SE_c	1	-0.21326	0.05303	-4.02	<.0001	-0.09167	2.48124

Figure 12 (multicollinearity between location and interactive variables - Omer's model):

NE	1	-0.58897	0.42929	-1.37	0.1703	-0.45894	514.26479
SW	1	0.02222	0.38423	0.06	0.9539	0.01742	416.74522
SE	1	-1.96243	0.38322	-5.12	<.0001	-1.67702	492.88135
Q2	1	0.02684	0.02305	1.16	0.2444	0.02403	1.95671
Q3	1	-0.02309	0.02382	-0.97	0.3326	-0.01963	1.88480
Q4	1	-0.01307	0.02462	-0.53	0.5956	-0.01052	1.80565
ln_sqft_living_NE	1	0.06730	0.05648	1.19	0.2336	0.40292	525.52383
ln_sqft_living_SW	1	-0.06547	0.05161	-1.27	0.2047	-0.38046	413.26043
ln_sqft_living_SE	1	0.19953	0.05074	3.93	<.0001	1.30175	503.68349

Figure 13 (No Multicollinearity after centering ln_sqft_living_c -Omer's model):

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	13.61905	0.13592	100.20	<.0001	0	0
bathrooms	1	0.07111	0.01804	3.94	<.0001	0.10436	3.22209
bedrooms	1	-0.05537	0.01146	-4.83	<.0001	-0.09664	1.83823
ln_sqft_lot	1	0.01005	0.01137	0.88	0.3769	0.01689	1.67801
ln_sqft_living_c	1	-0.63372	0.04804	-13.19	<.0001	-0.51630	7.04092
floor_h	1	0.01637	0.02390	0.68	0.4936	0.01552	2.35983
waterfront	1	0.36661	0.09178	3.99	<.0001	0.06744	1.31001
view_good	1	0.29661	0.04679	6.34	<.0001	0.10941	1.36863
condition_good	1	0.10677	0.01794	5.95	<.0001	0.09523	1.17665
grade_b	1	-0.48549	0.06364	-7.63	<.0001	-0.28191	6.27573
grade_a	1	-0.47320	0.05412	-8.74	<.0001	-0.30110	5.45058
renovated	1	0.16780	0.04247	3.95	<.0001	0.06033	1.07181
basement	1	0.00191	0.01960	0.10	0.9225	0.00178	1.53030
NE	1	-0.08161	0.02482	-3.29	0.0010	-0.06359	1.71882
SW	1	-0.47137	0.02326	-20.27	<.0001	-0.36943	1.52677
SE	1	-0.45824	0.02286	-20.05	<.0001	-0.39159	1.75322
Q2	1	0.02684	0.02305	1.16	0.2444	0.02403	1.95671
Q3	1	-0.02309	0.02382	-0.97	0.3326	-0.01963	1.88480
Q4	1	-0.01307	0.02462	-0.53	0.5956	-0.01052	1.80565
ln_sqft_living_NE_c	1	-0.06730	0.05648	-1.19	0.2336	-0.02342	1.77568
ln_sqft_living_SW_c	1	0.06547	0.05161	1.27	0.2047	0.02640	1.99039
ln_sqft_living_SE_c	1	-0.19953	0.05074	-3.93	<.0001	-0.08024	1.91364

Figure 14 (Before centering for multicollinearity between interactive variables - Yusheng's model):

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	13.16448	0.69064	19.06	<.0001	0	0
bathrooms	1	-1.01280	0.24475	-4.14	<.0001	-1.49810	636.01222
bedrooms	1	-0.82097	0.20086	-4.09	<.0001	-1.44012	602.43419
ln_sqft_living	1	-0.03989	0.10769	-0.37	0.7111	-0.03269	37.78730
ln_sqft_lot	1	-0.00110	0.01093	-0.10	0.9199	-0.00185	1.64002
floor_h	1	-0.08420	0.08039	-1.05	0.2951	-0.08045	28.62760
waterfront	1	0.32937	0.09411	3.50	0.0005	0.05672	1.27456
view_good	1	0.30469	0.04529	6.73	<.0001	0.11165	1.33658
condition_good	1	0.12714	0.07920	1.61	0.1086	0.02525	1.20019
grade_b	1	-0.20254	0.24239	-0.84	0.4035	-0.11846	97.52511
grade_a	1	-0.10346	0.24717	-0.42	0.6756	-0.06634	121.89738
grade_h	1	0.14020	0.24900	0.56	0.5735	0.04034	24.90157
renovated	1	0.14389	0.04093	3.52	0.0005	0.05227	1.07288
NE	1	-0.06815	0.41286	-0.17	0.8689	-0.05356	510.83235
SW	1	-0.07257	0.38022	-0.19	0.8487	-0.05710	434.26487
SE	1	-1.61683	0.37653	-4.29	<.0001	-1.40284	517.92543
Q2	1	0.03033	0.02232	1.36	0.1744	0.02740	1.97319
Q3	1	-0.01777	0.02307	-0.77	0.4413	-0.01520	1.88897
Q4	1	-0.00433	0.02382	-0.18	0.8559	-0.00351	1.81614
bb	1	-0.06464	0.01635	-3.95	<.0001	-0.51979	83.89590
bathliving	1	0.16931	0.03439	4.92	<.0001	2.13653	913.90692
bedliving	1	0.11834	0.02875	4.12	<.0001	1.79770	925.95889
bathfloor	1	0.01913	0.03365	0.57	0.5698	0.05034	38.05805
living_NE	1	0.00143	0.05425	0.03	0.9789	0.00866	521.32727
living_SW	1	-0.05279	0.05108	-1.03	0.3016	-0.30786	430.61732
living_SE	1	0.15748	0.04976	3.17	0.0016	1.04303	527.01608

Figure 15 (After centering for multicollinearity between interactive variables - Yusheng's model):

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	13.20197	0.26051	50.68	<.0001	0	0
bathrooms_c	1	-0.04746	0.02179	-2.18	0.0296	-0.07020	5.04071
bedrooms_c	1	0.06513	0.01140	5.71	<.0001	0.11425	1.94178
ln_sqft_living_c	1	-0.71305	0.04559	-15.64	<.0001	-0.58430	6.77398
ln_sqft_lot	1	-0.00110	0.01093	-0.10	0.9199	-0.00185	1.64002
floor_h	1	-0.04386	0.02174	-2.02	0.0439	-0.04190	2.09420
waterfront	1	0.32937	0.09411	3.50	0.0005	0.05672	1.27456
view_good	1	0.30469	0.04529	6.73	<.0001	0.11165	1.33658
condition_good	1	0.12714	0.07920	1.61	0.1086	0.02525	1.20019
grade_b	1	-0.20254	0.24239	-0.84	0.4035	-0.11846	97.52511
grade_a	1	-0.10346	0.24717	-0.42	0.6756	-0.06634	121.89738
grade_h	1	0.14020	0.24900	0.56	0.5735	0.04034	24.90157
renovated	1	0.14389	0.04093	3.52	0.0005	0.05227	1.07288
NE	1	-0.05735	0.02398	-2.39	0.0169	-0.04507	1.72402
SW	1	-0.47057	0.02269	-20.74	<.0001	-0.37024	1.54706
SE	1	-0.42951	0.02189	-19.63	<.0001	-0.37266	1.74972
Q2	1	0.03033	0.02232	1.36	0.1744	0.02740	1.97319
Q3	1	-0.01777	0.02307	-0.77	0.4413	-0.01520	1.88897
Q4	1	-0.00433	0.02382	-0.18	0.8559	-0.00351	1.81614
bb_c	1	-0.06464	0.01635	-3.95	<.0001	-0.10841	3.64963
bathliving_c	1	0.16931	0.03439	4.92	<.0001	0.14299	4.09338
bedliving_c	1	0.11834	0.02875	4.12	<.0001	0.11048	3.49750
bathfloor_c	1	-0.01913	0.03365	-0.57	0.5698	-0.01753	4.61446
living_NE_c	1	-0.00143	0.05425	-0.03	0.9789	-0.00051103	1.81731
living_SW_c	1	0.05279	0.05108	1.03	0.3016	0.02095	1.99355
living_SE_c	1	-0.15748	0.04976	-3.17	0.0016	-0.06517	2.05722

Figure 16   (R (ln_sqft_living, ln_sqft_above) = 0.87 Close to 0.9 -Yusheng's model):

	bathrooms	bedrooms	ln_sqft_living	ln_sqft_lot	ln_sqft_above
bathrooms	1.00000	0.53987 <.0001	0.74995 <.0001	0.04890 0.0590	0.69527 <.0001
bedrooms	0.53987 <.0001	1.00000	0.64290 <.0001	0.19279 <.0001	0.54862 <.0001
ln_sqft_living	0.74995 <.0001	0.64290 <.0001	1.00000	0.31093 <.0001	0.87012 <.0001
ln_sqft_lot	0.04890 0.0590	0.19279 <.0001	0.31093 <.0001	1.00000	0.32291 <.0001
ln_sqft_above	0.69527 <.0001	0.54862 <.0001	0.87012 <.0001	0.32291 <.0001	1.00000
floor_h	0.60007 <.0001	0.22396 <.0001	0.43967 <.0001	-0.16111 <.0001	0.59946 <.0001
waterfront	0.07033 0.0066	-0.01767 0.4953	0.07347 0.0045	0.10215 <.0001	0.07122 0.0059

Figure 17 (Yusheng)

	ln_price	bathrooms	bedrooms	ln_sqft_living	ln_sqft_lot	floor_h	waterfront
ln_price	1.00000	0.54016 <.0001	0.33448 <.0001	0.65700 <.0001	0.15203 <.0001	0.28463 <.0001	0.16353 <.0001
bathrooms	0.54016 <.0001	1.00000	0.53987 <.0001	0.74995 <.0001	0.04890 0.0590	0.60007 <.0001	0.07033 0.0066
bedrooms	0.33448 <.0001	0.53987 <.0001	1.00000	0.64290 <.0001	0.19279 <.0001	0.22396 <.0001	-0.01767 0.4953
ln_sqft_living	0.65700 <.0001	0.74995 <.0001	0.64290 <.0001	1.00000	0.31093 <.0001	0.43967 <.0001	0.07347 0.0045
ln_sqft_lot	0.15203 <.0001	0.04890 0.0590	0.19279 <.0001	0.31093 <.0001	1.00000	-0.16111 <.0001	0.10215 <.0001
floor_h	0.28463 <.0001	0.60007 <.0001	0.22396 <.0001	0.43967 <.0001	-0.16111 <.0001	1.00000	0.05748 0.0264
waterfront	0.16353 <.0001	0.07033 0.0066	-0.01767 0.4953	0.07347 0.0045	0.10215 <.0001	0.05748 0.0264	1.00000

Table 6 (Outliers and Influential Points):

Model	Outliers removed	Influential points removed	Explanation
M1	153		I took out all possible outliers (>+-3) and influential points based on studentized residual values and Cook's Distance (>0.0003).
M2	8	0	When I removed the 8 outliers the RMSE decreased; R-square increased and Adj R-square increased. However, when I removed 20 or more influential points my R-square and Adj-R-square decreased instead of increasing. Therefore, I only took out the biggest outliers from my dataset. Also, since King County has one of the richest residents in United States, most of the outliers given by the model might not be outliers. It is also one of the richest county in the country.
M3	8		I remove 8 observations that are both outliers and influential points. My Adj-R-square increased from 0.66 to 0.67 after this step.

Table 7 (Jun’s Selection Methods):

Independent variables in the regression model before selection method	bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4 above_ne_c above_sw_c above_se_c	# of predictors= 21 Adj R-square = 0.7307 RMSE = 0.23566 The above values given by running the training set.
Independent variables given by both Backward and Stepwise method	bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b grade_h renovated basement NE SW SE Q4 above_NE_c above_SW_c above_SE_c	# of predictors= 16 Adj R-square = 0.7307 RMSE = 0.23566 The above values given by running the training set.
Final model variables after taking out two insignificant predictors	bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b grade_h renovated basement NE SW SE Q4 above_NE_c above_SW_c above_SE_c	# of predictors= 14 Adj R-square = 0.7263 RMSE = 0.23755 The above values given by running the training set.

Table 8 (Omer’s Selection Methods):

Independent variables in the regression model before selection method	bathrooms, bedrooms, ln_sqft_lot, ln_sqft_living_c, floor_h, waterfront ,view_good, condition_good, grade_b, grade_a, renovated, basement, NE, SW, SE, Q2, Q3, Q4, ln_sqft_living_NE_c, ln_sqft_living_SW_c, ln_sqft_living_SE_c.	# of predictors= 21
Independent variables given by Stepwise method	bathrooms, bedrooms, ln_sqft_living_c, waterfront ,view_good, condition_good, grade_b, grade_a, renovated, NE, SW, SE, Q2, ln_sqft_living_SE_c.	# of predictors=14 R-square = 0.6932 Adj R-square = 0.6894 RMSE = 0.28511 The above values given by running the training set.

Independent variables given by ADJRSQ method	bathrooms, bedrooms, ln_sqft_living_c, floor_h, waterfront, view_good, condition_good, grade_b, grade_a, renovated, NE, SW, SE, Q2, ln_sqft_living_SW_c, ln_sqft_living_SE_c.	# of predictors=16 R-square=0.6943 Adj R-square=0.6898 RMSE=0.28488 Variables floor_h, ln_sqft_living_SW_c are insignificant. The above values given by running the training set
Final Model Variables	bathrooms, bedrooms, ln_sqft_living_c, waterfront ,view_good, condition_good, grade_b, grade_a, renovated, NE, SW, SE, Q2, ln_sqft_living_SE_c.	# of predictors=14 R-square = 0.6932 Adj R-square = 0.6894 RMSE = 0.28511 The above values given by running the training set.

Figure 18 (Showing Omer's model after selection method on testing set):

Parameter Estimates							
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate	Variance Inflation
Intercept	1	13.61568	0.17496	77.82	<.0001	0	0
bathrooms	1	0.04730	0.03465	1.37	0.1731	0.06696	2.90327
bedrooms	1	-0.03794	0.02399	-1.58	0.1147	-0.06443	2.00353
ln_sqft_living_c	1	-0.68681	0.07311	-9.39	<.0001	-0.55047	4.14393
waterfront	1	0.07685	0.18282	0.42	0.6745	0.01467	1.46930
view_good	1	0.32628	0.08133	4.01	<.0001	0.13619	1.39087
condition_good	1	0.17464	0.03613	4.83	<.0001	0.14908	1.14786
grade_b	1	-0.46167	0.14371	-3.21	0.0014	-0.27594	8.90479
grade_a	1	-0.42327	0.12127	-3.49	0.0005	-0.27166	7.31114
renovated	1	0.17625	0.08199	2.15	0.0323	0.07181	1.34670
NE	1	-0.03112	0.04513	-0.69	0.4909	-0.02562	1.66618
SW	1	-0.42550	0.04715	-9.02	<.0001	-0.31601	1.47978
SE	1	-0.39518	0.04477	-8.83	<.0001	-0.32437	1.62992
Q2	1	0.00238	0.03399	0.07	0.9441	0.00204	1.02074
ln_sqft_living_SE_c	1	-0.22287	0.08605	-2.59	0.0100	-0.08931	1.43468

Table 9 (Yusheng's Selection Methods) :

Independent variables in the regression model before selection method	Bathrooms,bedrooms, ln_sqft_living_c, ln_sqft_lot , floor_h,waterfront,view_good , condition_good, grade_b, grade_a ,grade_h, renovated, NE, SW, SE, Q2 ,Q3, Q4, bb_c,bathliving_c,bedliving_c , bathfloor_c , living_NE_c ,living_SW_c, living_SE_c	# of predictors= 25 The above values given by running the training set.
Independent variables given by Forward Selection method	ln_sqft_living, SE, SW ,bathliving_c, view_good ,grade_h, bedrooms ,living_SE_c, renovated ,grade_b, waterfront, Q2 ,bedliving_c, bb_c, NE	# of predictors=15 R-square=0.6889 Adj R-square=0.6847 RMSE=0.29341 The above values given by running the training set
Independent variables given by Mallows'Cp Selection method	bathrooms, bedrooms ,ln_sqft_living_c, waterfront ,view_good, condition_good, grade_b, grade_a, renovated ,NE, SW, SE, Q2, bb_c, bathliving_c, bedliving_c, living_SE_c,	# of predictors=17 R-square=0.6904 Adj R-square=0.6857 RMSE=0.29297 Variables bathrooms_c and condition_good are insignificant.
Final Model Variables	Ln_sqft_living_c, waterfront view_good , SW, SE , bb_c, bathliving_c, bedliving_c, living_SE_c	# of predictors=9 R-square=0.6637 Adj R-square=0.6610 RMSE=0.30424 The above values given by

		running the training set.

Figure 19 (strongest parameters):

Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate
Intercept	1	13.33356	0.06481	205.74	<.0001	0
bedrooms	1	-0.07462	0.01863	-4.00	<.0001	-0.13250
ln_sqft_above_c	1	-0.69832	0.07163	-9.75	<.0001	-0.58981
waterfront	1	0.75344	0.25665	2.94	0.0036	0.08290
view_good	1	0.32257	0.09350	3.45	0.0006	0.10520
condition_good	1	0.15364	0.02909	5.28	<.0001	0.14629
grade_b	1	-0.11225	0.04717	-2.38	0.0179	-0.07263
grade_h	1	0.45324	0.15117	3.00	0.0029	0.08612
basement	1	0.37681	0.03111	12.11	<.0001	0.37483
SW	1	-0.54677	0.03925	-13.93	<.0001	-0.45987
SE	1	-0.38443	0.03249	-11.83	<.0001	-0.36842
Q4	1	-0.10542	0.03069	-3.43	0.0007	-0.09220
above_NE_c	1	-0.21899	0.10001	-2.19	0.0293	-0.08224
above_SW_c	1	0.28107	0.10550	2.66	0.0081	0.10071
above_SE_c	1	-0.26882	0.08775	-3.06	0.0024	-0.13145

Figure 20 (Effect of sqft_above on price):

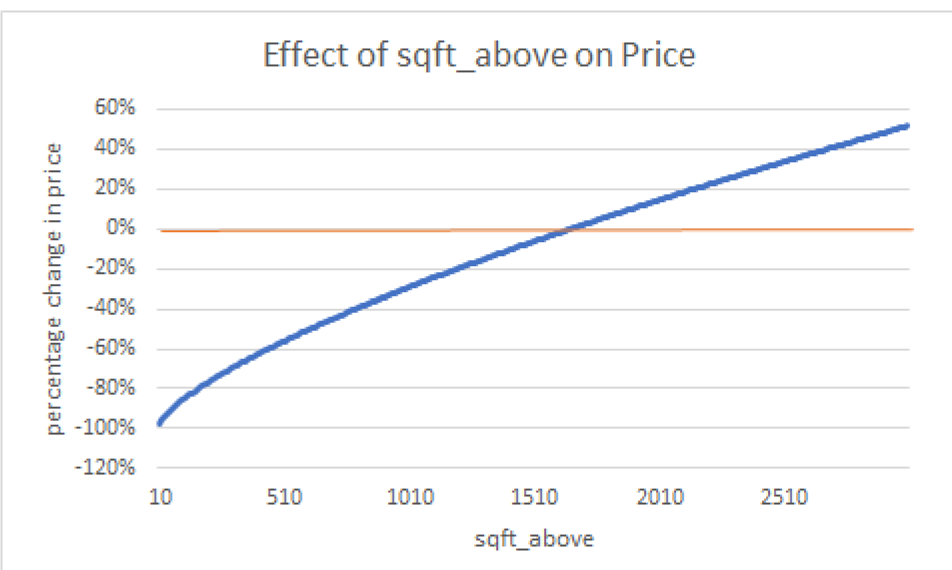
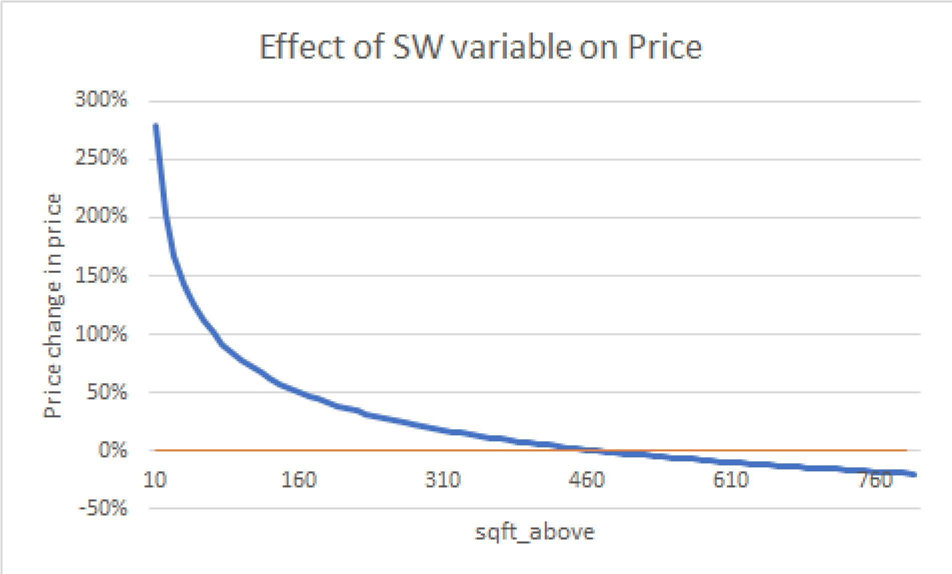


Figure 21 (Effect of SW variable on Price):



Code:
Attach the SAS code from each member to the word file. Make sure to include the name as a comment in your code.

Jun's Code:

```
proc import datafile = 'kc_house_data.csv' out=house replace;
delimiter=',';
getnames=yes;
run;

***
dependent variable = price

quantitative: bedrooms bathrooms sqft_living sqft_lot sqft_above
qualitative: floors waterfront view condition grade yr_renovated sqft_basement lat long date

not used: id zipcode yr_built sqft_living15 sqft_lot15

*sqft_above = sqft_living - sqft_basement
*sqft_living15 and sqft_lot15 = the average house and lot size of the 15 closest neighbors
*1 bath => Tub, shower, toilet, sink
*.75 bath => Shower, toilet, sink
*.5 bath => Toilet, sink
***;

proc surveyselect data=house
  method=srs n=1500 out=house_new seed=228247000;
run;

*Change the date format;
data house_new;
set house_new;
length date_var $ 8;
date_var=date;
run;
data house_new;
set house_new;
date_new = input(date_var, yymmdd8.);
format date_new date9.;
run;
```

```
data house_new;
set house_new;
qtr_var = qtr(date_new);
run;


*Dummy variables;
data house_new;
set house_new;
ln_price=log(price);
ln_sqft_living=log(sqft_living);
ln_sqft_lot =log(sqft_lot);
ln_sqft_above=log(sqft_above);
floor_h=(floors>1.5);
view_good=0;if view=3 or view=4 then view_good=1;
condition_good=0; if condition=4 or condition=5 then condition_good=1;
grade_b=0; if grade=4 or grade=5 or grade=6 then grade_b=1;
grade_a=0; if grade=7  or grade=8 or grade=9 or grade=10 then grade_a=1;
grade_h=0; if grade=11 or grade=12 or grade=13 then grade_h=1;
renovated= (yr_renovated>0);
basement= (sqft_basement>0);
S = (lat<47.57815);
E = (long>-122.237);
NE=0; if S=0 and E=1 then NE=1;
SW=0; if S=1 and E=0 then SW=1;
SE=0; if S=1 and E=1 then SE=1;
Q2=(qtr_var=2);
Q3=(qtr_var=3);
Q4=(qtr_var=4);
above_NE = ln_sqft_above*NE;
above_SW = ln_sqft_above*SW;
above_SE = ln_sqft_above*SE;
ln_sqft_above_c = 7.4107474 - ln_sqft_above;
above_NE_c = ln_sqft_above_c*NE;
above_SW_c = ln_sqft_above_c*SW;
above_SE_c = ln_sqft_above_c*SE;
run;

proc means mean median std stderr min p25 p50 p75 max clm;
var lat long qtr_var ln_sqft_living ln_sqft_above;
```

```
run;


*Check normality assumption / distribution for home saies by quarter;
proc univariate normal data=house_new;
var price;
histogram / normal (mu=est sigma=est);
run;
proc univariate normal data=house_new;
var ln_price;
histogram / normal (mu=est sigma=est);
run;

proc sgscatter data=house_new;
matrix ln_price sqft_living sqft_lot sqft_above ln_sqft_living ln_sqft_lot ln_sqft_above;
run;


*full model with log transformation on y;
proc reg data=house_new;
model ln_price =bathrooms bedrooms sqft_living sqft_lot sqft_above floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated basement NE SW SE Q2 Q3
Q4/vif stb;
plot student.*(sqft_living sqft_lot sqft_above predicted. npp.);
run;

*reduced model_1 with log transformation on x;
proc reg corr data=house_new;
model ln_price =bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_a grade_h renovated basement NE SW SE
Q2 Q3 Q4/vif stb;
plot student.*(ln_sqft_living ln_sqft_lot ln_sqft_above predicted. npp.);
run;



*reduced model_2: Remove a variable with multicollinearity problem;
* grade_a has the highest vif value (173.84572). Remove it and re-fit the model.;
proc reg corr data=house_new;
```

```
model ln_price =bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3
Q4/vif stb;
run;

*ln_sqft_living has the next highest vif value (18.73723). Remove it and re-fit the model;
proc reg corr data=house_new;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above floor_h waterfront view_good
condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4/vif stb;
run;




*reduced model_3: Make interaction variables;
proc reg corr data=house_new;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above floor_h waterfront view_good
condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4 above_ne above_sw
above_se /vif stb;
run;

*reduced model_4: Solve multicollinearity with interaction variable;
proc reg corr data=house_new;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c /vif stb;
run;



*reduced model_5: remove outliers and influential points;
proc reg corr data=house_new;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c /influence r;
run;

data house_new2;
set house_new;
```

```
if _n_ = 10 then delete;
if _n_ = 16 then delete;
if _n_ = 24 then delete;
if _n_ = 27 then delete;
if _n_ = 35 then delete;
if _n_ = 72 then delete;
if _n_ = 115 then delete;
if _n_ = 151 then delete;
if _n_ = 186 then delete;
if _n_ = 195 then delete;
if _n_ = 221 then delete;
if _n_ = 257 then delete;
if _n_ = 266 then delete;
if _n_ = 267 then delete;
if _n_ = 284 then delete;
if _n_ = 300 then delete;
if _n_ = 304 then delete;
if _n_ = 316 then delete;
if _n_ = 321 then delete;
if _n_ = 385 then delete;
if _n_ = 388 then delete;
if _n_ = 390 then delete;
if _n_ = 398 then delete;
if _n_ = 400 then delete;
if _n_ = 434 then delete;
if _n_ = 438 then delete;
if _n_ = 481 then delete;
if _n_ = 517 then delete;
if _n_ = 587 then delete;
if _n_ = 588 then delete;
if _n_ = 590 then delete;
if _n_ = 587 then delete;
if _n_ = 612 then delete;
if _n_ = 613 then delete;
if _n_ = 637 then delete;
if _n_ = 644 then delete;
if _n_ = 645 then delete;
if _n_ = 664 then delete;
if _n_ = 673 then delete;
if _n_ = 738 then delete;
```

```
if _n_ = 765 then delete;
if _n_ = 771 then delete;
if _n_ = 772 then delete;
if _n_ = 791 then delete;
if _n_ = 806 then delete;
if _n_ = 808 then delete;
if _n_ = 833 then delete;
if _n_ = 845 then delete;
if _n_ = 850 then delete;
if _n_ = 856 then delete;
if _n_ = 868 then delete;
if _n_ = 866 then delete;
if _n_ = 869 then delete;
if _n_ = 883 then delete;
if _n_ = 895 then delete;
if _n_ = 896 then delete;
if _n_ = 899 then delete;
if _n_ = 910 then delete;
if _n_ = 924 then delete;
if _n_ = 938 then delete;
if _n_ = 951 then delete;
if _n_ = 952 then delete;
if _n_ = 958 then delete;
if _n_ = 959 then delete;
if _n_ = 960 then delete;
if _n_ = 965 then delete;
if _n_ = 1013 then delete;
if _n_ = 1041 then delete;
if _n_ = 1044 then delete;
if _n_ = 1050 then delete;
if _n_ = 1074 then delete;
if _n_ = 1085 then delete;
if _n_ = 1098 then delete;
if _n_ = 1100 then delete;
if _n_ = 1103 then delete;
if _n_ = 1133 then delete;
if _n_ = 1147 then delete;
if _n_ = 1172 then delete;
if _n_ = 1182 then delete;
if _n_ = 1188 then delete;
```

```
if _n_ = 1189 then delete;
if _n_ = 1234 then delete;
if _n_ = 1278 then delete;
if _n_ = 1280 then delete;
if _n_ = 1282 then delete;
if _n_ = 1302 then delete;
if _n_ = 1303 then delete;
if _n_ = 1347 then delete;
if _n_ = 1348 then delete;
if _n_ = 1394 then delete;
if _n_ = 1414 then delete;
if _n_ = 1415 then delete;
if _n_ = 1452 then delete;
if _n_ = 1473 then delete;
if _n_ = 1484 then delete;
if _n_ = 1496 then delete;
run;

*Check for additional outliers and influential points;
proc reg corr data=house_new2;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c /influence r;
run;

data house_new3;
set house_new2;
if _n_= 7 then delete;
if _n_= 17 then delete;
if _n_= 106 then delete;
if _n_= 123 then delete;
if _n_= 193 then delete;
if _n_= 200 then delete;
if _n_= 223 then delete;
if _n_= 225 then delete;
if _n_= 312 then delete;
if _n_= 322 then delete;
if _n_= 350 then delete;
if _n_= 370 then delete;
if _n_= 386 then delete;
```

```
if _n_= 486 then delete;
if _n_= 495 then delete;
if _n_= 510 then delete;
if _n_= 563 then delete;
if _n_= 581 then delete;
if _n_= 604 then delete;
if _n_= 611 then delete;
if _n_= 612 then delete;
if _n_= 613 then delete;
if _n_= 622 then delete;
if _n_= 627 then delete;
if _n_= 642 then delete;
if _n_= 660 then delete;
if _n_= 671 then delete;
if _n_= 675 then delete;
if _n_= 680 then delete;
if _n_= 689 then delete;
if _n_= 691 then delete;
if _n_= 731 then delete;
if _n_= 736 then delete;
if _n_= 756 then delete;
if _n_= 781 then delete;
if _n_= 809 then delete;
if _n_= 824 then delete;
if _n_= 868 then delete;
if _n_= 898 then delete;
if _n_= 946 then delete;
if _n_= 949 then delete;
if _n_= 1000 then delete;
if _n_= 1004 then delete;
if _n_= 1006 then delete;
if _n_= 1011 then delete;
if _n_= 1022 then delete;
if _n_= 1023 then delete;
if _n_= 1047 then delete;
if _n_= 1058 then delete;
if _n_= 1074 then delete;
if _n_= 1124 then delete;
if _n_= 1136 then delete;
if _n_= 1144 then delete;
```

```
if _n_= 1205 then delete;
if _n_= 1211 then delete;
if _n_= 1247 then delete;
if _n_= 1323 then delete;
if _n_= 1389 then delete;
run;

proc reg corr data=house_new3;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c /influence r;
run;



*Split the original data;
PROC SURVEYSELECT DATA=house_new3
OUT = house_split seed=92595001
SAMPRATE = 0.75 OUTALL;
RUN;
data house_train (where = (Selected = 1));
set house_split;
run;
data house_test (where = (Selected = 0));
set house_split;
run;

proc reg data=house_train;
model  ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/stb;
run;

proc reg data=house_test;
model  ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/stb;
run;
```

```
*model selections;
proc reg data=house_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/selection=backward sle=0.05 sls=0.05;
run;

proc reg data=house_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/selection=stepwise sle=0.05 sls=0.05;
run;

proc reg data=house_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/selection=adjrsq sle=0.05 sls=0.05;
run;

proc reg data=house_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_above_c floor_h waterfront
view_good condition_good grade_b grade_h renovated basement NE SW SE Q2 Q3 Q4
above_ne_c above_sw_c above_se_c/selection=cp sle=0.05 sls=0.05;
run;

proc reg data=house_train;
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h renovated basement NE SW SE Q4 above_NE_c above_SW_c above_SE_c/stb;
run;

*We found two insignificant variables from testing set after model selection;
proc reg data=house_test;
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h renovated basement NE SW SE Q4 above_NE_c above_SW_c above_SE_c/stb;
run;
*Since standardized estimates for insignificant variables are significantly, there would be limited
or almost no influence on the final model. Thus, we decided to take them out;
*Renovated and NE;

proc reg data=house_train;
```

```
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h basement SW SE Q4 above_NE_c above_SW_c above_SE_c/stb;
run;

proc reg data=house_test;
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h basement SW SE Q4 above_NE_c above_SW_c above_SE_c/stb;
run;




*Take the first two observations from training set ;
proc print data=house_train;
run;

data pred;
input bedrooms      waterfront    view_good    condition_good     grade_b
        grade_h      basement      SW     SE     Q4     ln_sqft_above_c      above_NE_c
        above_SW_c   above_SE_c;
datalines;
2     0     0     1     0     0     0     0     1     1     0.32067     0
      0     0.32067
3     0     0     1     0     0     0     0     1     1     -0.3931     0
      0     -0.3931
;

*Compute two predictions including the prediction intervals using the regression model.;
data prediction;
set pred house_test;
run;

proc reg data=prediction;;
model ln_price =bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h basement SW SE Q4 above_NE_c above_SW_c above_SE_c / p clm cli alpha=0.05;
run;



*Validation test;
```

```
* create new variable new_y = ln_price for training set, and = NA * for testing set;
data house_validation;
set house_split;
if selected then new_y=ln_price;
run;

/* get predicted values for the missing new_y in test set for the fitted model*/
title "Validation - Test Set";
proc reg data=house_validation;
model new_y = bedrooms ln_sqft_above_c waterfront view_good condition_good grade_b
grade_h basement SW SE Q4 above_NE_c above_SW_c above_SE_c;
output out=outm1(where=(new_y=.)) p=yhat;
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_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;
```

```
*---------------------------------------;
*boxplot for bedrooms;
proc sort;
by bedrooms;
run;
proc boxplot;
plot ln_price*bedrooms;
run;

*boxplot for bathrooms;
proc sort;
by bathrooms;
run;
proc boxplot;
plot ln_price*bathrooms;
run;

*boxplot for floors;
proc sort;
by floors;
run;
proc boxplot;
plot ln_price*floors;
run;

*boxplot for waterfront;
proc sort;
by waterfront;
run;
proc boxplot;
plot ln_price*waterfront;
run;

*boxplot for view;
proc sort;
by view;
run;
proc boxplot;
plot ln_price*view;
run;
```

```
*boxplot for condition;
proc sort;
by condition;
run;
proc boxplot;
plot ln_price*condition;
run;

*boxplot for grade;
proc sort;
by grade;
run;
proc boxplot;
plot ln_price*grade;
run;

*boxplot for yr_built;
proc sort;
by yr_built;
run;
proc boxplot;
plot ln_price*yr_built;
run;

*boxplot for renovated;
proc sort;
by renovated;
run;
proc boxplot;
plot ln_price*renovated;
run;

*boxplot for basementt;
proc sort;
by basement;
run;
proc boxplot;
plot ln_price*basement;
run;
```

```
*boxplot for qtr_var;
proc sort;
by qtr_var;
run;
proc boxplot;
plot ln_price*qtr_var;
run;

*END OF JUN'S CODE;
```

Omer’s Code:

```
*START OF
OMER CHEEMA's
CODE;

*Importing the Data;
PROC IMPORT datafile = 'kc_house_data.csv' out=housePriceALL replace;
delimiter=',';
getnames=yes;
run;

*Choosing a random sample of 1500 rows from a file of 21,613 rows;
PROC SURVEYSELECT data=housePriceALL
   method=srs n=1500 out=housePrices seed=60001;
run;

PROC PRINT data=housePrices;
run;

*START OF PREPROCESSING;

DATA housePrices;
set housePrices;
length date_var $ 8;
date_var=date;
run;


DATA housePrices;
set housePrices;
date_new = input(date_var, yymmdd8.);
format date_new date9.;
run;


DATA housePrices;
set housePrices;
qtr_var = qtr(date_new);
```

```
run;

PROC MEANS mean median std stderr min p25 p50 p75 max clm;
var lat long qtr_var;
run;

*Creating all the dummy variables;
DATA housePrices;
set housePrices;
floor_h=(floors>1.5);
view_good=0;if view=3 or view=4 then view_good=1;
condition_good=0; if condition=4 or condition=5 then condition_good=1;
grade_b=0; if grade=4 or grade=5 or grade=6 then grade_b=1;
grade_a=0; if grade= 7 or grade=8 or grade=9 or grade=10 then grade_a=1;
grade_h=0; if grade=11 or grade=12 or grade=13  then grade_h=1;
renovated= (yr_renovated>0);
basement= (sqft_basement>0);
*Use the median latitude and longitude values;
S = (lat<47.5700);
E = (long>-122.2250);
NE=0; if S=0 and E=1 then NE=1;
SW=0; if S=1 and E=0 then SW=1;
SE=0; if S=1 and E=1 then SE=1;
Q2=(qtr_var=2);
Q3=(qtr_var=3);
Q4=(qtr_var=4);
run;


*\
END OF PREPROCESSING
From here every team member work should be their own.
*\;

*START OF TRANFORMATION;
*/CHECKING WHETHER TRANSFORMATION IS NEEDED OR NOT;
*DO RESIDUAL ANALYSIS;

*Checking if price has normal distribution or not;
PROC UNIVARIATE normal;
```

```
var price;
TITLE 'Histogram of price (before the transformation)';
histogram / normal (mu=est sigma=est);
run;


PROC REG data=housePrices;
model price =bathrooms bedrooms sqft_living sqft_lot sqft_above floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated basement NE SW SE Q2 Q3 Q4/vif stb;
plot student.*(sqft_living sqft_lot sqft_above);
plot student.*predicted.;
TITLE 'Normal Probability Plot before the transformation';
plot npp.*student.;
run;
title;

*Since the distribution for price is positively skewed, we are going to do a log transformation for
price;
*nnp show a S shape, so we need to transform price;
DATA housePrices;
set housePrices;
ln_price=log(price);
run;

*Checking the distribution for ln_price;
*The skeweness decreased from 2.9095 to 0.3486 hence, ln_price has almost normal
distribution;
*nnp S shape is gone;

PROC UNIVARIATE normal;
var ln_price;
TITLE 'Histogram of ln_price (after tranformation)';
histogram / normal (mu=est sigma=est);
run;

*Now lets see whether the quantitative independent variables need to be transformed or not;
PROC SGSCATTER;
matrix ln_price sqft_living sqft_lot sqft_above;
run;
```

```
PROC CORR;
var ln_price sqft_living sqft_lot sqft_above;
run;

*all three independent variable fail residual analysis, need to tranform them;

PROC REG data=housePrices;
model ln_price =bathrooms bedrooms sqft_living sqft_lot sqft_above floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated basement NE SW SE Q2 Q3
Q4/vif stb;
plot student.*(sqft_living sqft_lot sqft_above);
plot student.*predicted.;
TITLE 'Normal Probabilty Plot before transformation';
plot npp.*student.;
run;
title;

*Since the independent variables are (not linear) and does not have constant variance and
independence, we need to tranform it;
*BTW After using any transformation the sqft_lot does not become linear;
DATA housePrices;
set housePrices;
ln_sqft_living = log(sqft_living);
ln_sqft_above = log(sqft_above);
ln_sqft_lot = log(sqft_lot);
run;

*Lets look at the scatter plot after the transformation of quantitative variables;
PROC SGSCATTER;
matrix ln_price ln_sqft_living ln_sqft_lot ln_sqft_above;
run;

PROC CORR;
var ln_price ln_sqft_living ln_sqft_lot ln_sqft_above;
run;

PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_a grade_h renovated basement NE SW SE
Q2 Q3 Q4/vif stb;
```

```
plot student.*(ln_sqft_living ln_sqft_lot ln_sqft_above);
plot student.*predicted.;
plot npp.*student.;
run;

*END OF TRANFORMATION AND RESIDUAL ANALYSIS;

*IMPLEMENTING METHODOLOGY OBTAINED FROM REFERENCES;

*The pearson correlations show that some of the predictors are highly correlated with the
response.;
*However, as we have seen, some predictors are also highly correlated with other predictors,
making it difficult to determine which predictors are actually important.;
*A partial correlation is the correlation between two variables, while controlling for the
correlation with other variables.;
*Partial correlations allow us to see correlations between each predictor and the response, after
adjusting for the other predictors;
*https://www.jmp.com/content/dam/jmp/documents/en/academic/case-study-library/case-study-
library-12/business-case-studies/12-housingprices.pdf;


* Lets look at the Pearson Correlation between response and transformed predictors;
TITLE 'Pearson Correlation';
PROC CORR data=housePrices;
var ln_price ln_sqft_living ln_sqft_lot ln_sqft_above;
run;

*Now lets look at the partial correlation between the response and each predictor one by one;

*Lets look at the true relationship between ln_price and ln_sqft_living while controlling the
correlation of ln_sqft_lot and ln_sqft_above;
TITLE 'Partial Correlation between ln_price and ln_sqft_living';
PROC CORR data=housePrices;
var ln_price ln_sqft_living;
partial ln_sqft_lot ln_sqft_above;
run;

*Lets look at the true relationship between ln_price and ln_sqft_lot while controlling the
correlation of ln_sqft_living and ln_sqft_above;
TITLE 'Partial Correlation between ln_price and ln_sqft_lot';
```

```
PROC CORR data=housePrices;
var ln_price ln_sqft_lot;
partial ln_sqft_living ln_sqft_above;
run;

*Lets look at the true relationship between ln_price and ln_sqft_above while controlling the correlation of ln_sqft_living and ln_sqft_lot;
TITLE 'Partial Correlation between ln_price and ln_sqft_lot';
PROC CORR data=housePrices;
var ln_price ln_sqft_above;
partial ln_sqft_living ln_sqft_lot;
run;
title;

*As you can see the true relationship between ln_price and ln_sqft_living is still moderate.

*And the true relationship between ln_price and ln_sqft_lot and ln_sqft_above is weak negative instead of weak positive and moderate positive;
*END OF IMPLEMENTING METHODOLOGY OBTAINED FROM REFERENCES;

*CHECKING MULTICOLLINEARITY
*Initial Model;
PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_a grade_h renovated basement NE SW SE
Q2 Q3 Q4/vif stb;
run;

*The attributes ln_sqft_living and ln_sqft_above have multicollinearity.;
*ln_sqft_living has a higher VIF value compared to ln_sqft_above but I am going to take out ln_sqft_above because ln_sqft_living has higher standardized estimate;
PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated basement NE SW SE Q2 Q3 Q4/vif stb;
run;

*The variables grade_b, grade_a and grade_h are also mulitcollinear with each other;
*I am going to take out grade_h even though it has a lower VIF value, it is an insignifiacant variable;
*Taking out grade_h decrease the VIF for grade_b and grade_a below 10.;
```

```
PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4/vif stb;
run;

*END OF CHECKING MULTICOLLINEAR VARIABLES;


*TESTING INTERACTION VARIABLES;
*Creating interaction variables;
DATA housePrices;
set housePrices;
ln_sqft_living_NE = ln_sqft_living*NE;
ln_sqft_living_SW = ln_sqft_living*SW;
ln_sqft_living_SE = ln_sqft_living*SE;
run;

PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4 ln_sqft_living_NE
ln_sqft_living_SW ln_sqft_living_SE/vif stb;
run;

*After making the interaction variables, 6 predictors have become multicollinear with VIF
values in 500s;
*Since NE, SW and SE and there interaction variables have the VIF in 500s, lets center the
variables;

PROC CORR;
var ln_price ln_sqft_living NE SW SE ln_sqft_living_NE ln_sqft_living_SW ln_sqft_living_SE;
run;


DATA housePrices;
set housePrices;
ln_sqft_living_c =7.53876 - ln_sqft_living;
ln_sqft_living_NE_c = ln_sqft_living_c*NE;
ln_sqft_living_SW_c = ln_sqft_living_c*SW;
ln_sqft_living_SE_c = ln_sqft_living_c*SE;
run;
```

```
*Running the model again after centering ln_sqft_living;
PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/vif stb;
run;

*END OF TESING INTERACTIVE VARIABLE;


*CHECKING FOR OUTLIERS AND INFLUENTIAL POINTS;

PROC REG data=housePrices;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/stb vif r influence;
run;

*There are 8 outliers in the model, lets take them out;
*The observations which have studentized residual greater or less than 3 or -3 are outliers;
DATA housePricesNew;
set housePrices;
if _n_=21 then delete;
if _n_=27 then delete;
if _n_=69 then delete;
if _n_=297 then delete;
if _n_=443 then delete;
if _n_=581 then delete;
if _n_=854 then delete;
if _n_=1262 then delete;
run;

*Lets re-run the model after taking out the outliers;
*My R-square improved from 0.6784 to 0.6936;
*Adj R-square increased from 0.6738 to 0.6892;
*RMSE decreased from 0.29862 to 0.28927;
PROC REG data=housePricesNew;
```

```
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/stb vif r influence;
run;

*An obs is an influential point if it's cook'd distance is greater than 0.003;
*Now I will take out the new outliers and influential points with arrows;
DATA housePricesNew2;
set housePricesNew;
if _n_=32 then delete;
if _n_=82 then delete;
if _n_=287 then delete;
if _n_=297 then delete;
if _n_=351 then delete;
if _n_=404 then delete;
if _n_=424 then delete;
if _n_=430 then delete;
if _n_=474 then delete;
if _n_=551 then delete;
if _n_=570 then delete;
if _n_=667 then delete;
if _n_=705 then delete;
if _n_=751 then delete;
if _n_=779 then delete;
if _n_=797 then delete;
if _n_=912 then delete;
if _n_=980 then delete;
if _n_=984 then delete;
if _n_=1081 then delete;
if _n_=1100 then delete;
if _n_=1136 then delete;
if _n_=1138 then delete;
if _n_=1249 then delete;
if _n_=1275 then delete;
if _n_=1282 then delete;
if _n_=1383 then delete;
if _n_=1455 then delete;
run;

*lets re-run the model after taking out the biggest IPs (ones with cook's d >= 0.010;
```

```
*R-square decreased from 0.6936 to 0.6866;
*Adj R-square decreased from 0.6893 to 0.6820;
*RMSE decreased from 0.28927 to 0.27654;
PROC REG data=housePricesNew2;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/stb vif r influence;
run;

*After taking out the last set of observations (which had outliers and IPs) the RMSE further
decreased (which is good) but R-square and adj-R-square also decreased.;
*Therefore, I am going to ignore the removal of last set of observations and use
data=housePricesNew, instead of housePricesNew2.;

*END OF CHECKING FOR OUTLIERS AND INFLUENTIAL POINTS;


*START OF SPLITTING THE DATA TO TRAIN AND TEST;

*I am going to use dataset= housePricesNew2 for splitting;
PROC SURVEYSELECT DATA=housePricesNew
OUT = houseP_split SEED=56789
SAMPRATE = 0.75 OUTALL;
RUN;

*Making different datasets for training and testing data;
DATA houseP_train (where = (Selected = 1));
set houseP_split;
run;

DATA houseP_test (where = (Selected = 0));
set houseP_split;
run;

PROC PRINT data=houseP_train;
run;

PROC PRINT data=houseP_test;
run;
```

```
TITLE 'Full Regression model with training set before selection';
PROC REG data=houseP_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/stb vif;
run;

TITLE 'Full Regression model with testing set before selection';
PROC REG data=houseP_test;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/stb vif;
run;

*END OF SPLITTING DATA TO TRAINING AND TESTING;

*START OF SELECTION METHOD;


*Full Model with STEPWISE selection;
TITLE 'Doing Stepwise selection on full regression model with training set';
PROC REG data=houseP_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/selection=stepwise sle=0.05
sls=0.05;
run;

*Running the model selected by Stepwise;
proc reg data=houseP_train;
model ln_price = bathrooms bedrooms ln_sqft_living_c waterfront view_good condition_good
grade_b grade_a renovated NE SW SE Q2 ln_sqft_living_SE_c;
run;

*Full Model with ADJRSQ;
TITLE 'Doing ADJRSQ selection on full regression model with training set';
PROC REG data=houseP_train;
model ln_price =bathrooms bedrooms ln_sqft_lot ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated basement NE SW SE Q2 Q3 Q4
ln_sqft_living_NE_c ln_sqft_living_SW_c ln_sqft_living_SE_c/selection=adjrsq;
```

```
run;

*Running the model with the highest Adj r-square selected by ADJRSQ;
proc reg data=houseP_train;
model ln_price = bathrooms bedrooms ln_sqft_living_c floor_h waterfront view_good
condition_good grade_b grade_a renovated NE SW SE Q2 ln_sqft_living_SW_c
ln_sqft_living_SE_c/stb VIF;
run;

*Training Final model;
TITLE 'Checking the Final model with training set';
PROC REG data=houseP_train;
model ln_price=bathrooms bedrooms ln_sqft_living_c waterfront view_good condition_good
grade_b grade_a renovated NE SW SE Q2 ln_sqft_living_SE_c/stb vif;
run;

*Testing Final model;
TITLE 'Checking the Final model with testing set';
PROC REG data=houseP_test;
model ln_price=bathrooms bedrooms ln_sqft_living_c waterfront view_good condition_good
grade_b grade_a renovated NE SW SE Q2 ln_sqft_living_SE_c/stb vif;
run;
title;

*When running the final model with the testing set (houseP_test)bathrooms, bedrooms,
waterfront, NE and Q2 ;
*Lets take them out of the final model and re-run with the training and testing set;

*Training Final model after taking out bathrooms, bedrooms, waterfront, NE and Q2 ;
PROC REG data=houseP_train;
model ln_price=ln_sqft_living_c view_good condition_good grade_b grade_a renovated SW SE
ln_sqft_living_SE_c/stb vif;
run;

*Testing Final model after taking out bathrooms, bedrooms, waterfront, NE and Q2;
TITLE 'checking final model with testing set after taking out Q2';
PROC REG data=houseP_test;
model ln_price=ln_sqft_living_c view_good condition_good grade_b grade_a renovated SW SE
ln_sqft_living_SE_c/stb vif;
run;
```

```
title;

*END OF SELECTION METHOD;

*START OF PREDICTION;

PROC PRINT data=houseP_train;
run;

*choosing the first two rows from the training date set (houseP_train);
DATA pred;
input ln_sqft_living_c view_good condition_good grade_b grade_a renovated SW SE
ln_sqft_living_SE_c;
datalines;
-0.04194 0 1 0 1 0 1 0 0.00000
-0.63876 0 0 0 1 0 0 0 0.00000
;


*join the first two rows of houseP train with houseP_test to predict;
data predict;
set pred houseP_test;

proc print data= predict;
run;

*Compute Prediction;

PROC REG data=predict;
model ln_price=ln_sqft_living_c view_good condition_good grade_b grade_a renovated SW SE
ln_sqft_living_SE_c/p clm cli alpha=0.05;
run;

*NOTE: I got the same predicted price value when I ran the model with view_good, Q2 and
ln_sqft_living_SW in it. This means that those variable were truly insignificant;
*END OF PREDICTING;

*VALIDATION;
DATA housePricesNew_Val;
set houseP_split;
```

```
if selected then new_y=ln_price;
run;

PROC PRINT data=housePricesNew_Val;
run;

TITLE "Validation - Test Set";
PROC REG data=housePricesNew_Val;
model ln_price=ln_sqft_living_c view_good condition_good grade_b grade_a renovated SW SE
ln_sqft_living_SE_c;
output out=outm1(where=(new_y=.)) p=yhat;
run;

Title "Difference between Observed and Predicted in Test Set";
data outm1_sum;
set outm1;
d=ln_price-yhat;
absd=abs(d);
run;

Proc print;
run;

proc summary data=outm1_sum;
var d absd;
output out=outm1_stats std(d)=rmse mean(absd)=mae;
run;

TITLE 'Validation  statistics for Model';
proc print data=outm1_stats;
run;


proc corr data=outm1;
var ln_price yhat;
run;

*END OF VALIDATION;

*END OF
```

OMER CHEEMA's
CODE;

Yusheng's Code:

```
*** YUSHENG ZHU
*import the CSV file;
proc import datafile = 'kc_house_data.csv' out=house replace;
delimiter=',';
getnames=yes;
run;

***
dependent variable = price

quantitative: bedrooms bathrooms sqft_living sqft_lot sqft_above
qualitative: floors waterfront view condition grade yr_renovated  lat long date

not used: id zipcode yr_built sqft_living15 sqft_lot15 sqft_basement

*sqft_above = sqft_living - sqft_basement (remove sqft_basement because of perfect
multicollinearity)
*sqft_living15 and sqft_lot15 = the average house and lot size of the 15 closest neighbors
*1 bath => Tub, shower, toilet, sink
*.75 bath => Shower, toilet, sink
*.5 bath => Toilet, sink
***;


data house;
set house;
*Change the date format;
length date_var $ 8;
date_var=date;
run;
data house;
set house;
date_new = input(date_var, yymmdd8.);
format date_new date9.;
run;
data house;
set house;
```

```
qtr_var = qtr(date_new);
run;

*calculate the median latitude and longtitude;
proc means mean median std stderr min p25 p50 p75 max clm;
var lat long qtr_var;
run;

data house;
set house;
*Apply Log transformation on X-var and Y-var;
ln_price=log(price);
ln_sqft_living=log(sqft_living) ;
ln_sqft_lot =log(sqft_lot);
ln_sqft_above=log(sqft_above) ;
*Create Dummy variables;
floor_h=(floors>1.5);
view_good=0; if view=3 or view=4 then view_good=1;
condition_good=0; if condition=3 or condition=4 or condition=5 then condition_good=1;
grade_b=0; if grade=4 or grade=5 or grade=6  then grade_b=1;
grade_a=0; if grade=7 or grade=8 or grade=9 or grade=10 then grade_a=1;
grade_h=0; if grade=11 or grade=12 or grade=13 then grade_h=1;
renovated= (yr_renovated>0);
S = (lat<47.5718000);
E = (long>-122.2300000);
NE=0; if S=0 and E=1 then NE=1;
SW=0; if S=1 and E=0 then SW=1;
SE=0; if S=1 and E=1 then SE=1;
Q2=(qtr_var=2);
Q3=(qtr_var=3);
Q4=(qtr_var=4);
run;

*select n=1500 from the whole dataset as sample;
 PROC SURVEYSELECT DATA=house OUT=sample METHOD=SRS
 n=1500 seed=60001;
 RUN;


*Original/;
```

```
proc reg data=sample;
model price = bathrooms bedrooms sqft_living sqft_lot sqft_above floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4/vif stb;
plot npp.*student.;
run;

* Run model -after applying  Log Transformation on Y;
proc reg data=sample;
model ln_price = bathrooms bedrooms sqft_living sqft_lot sqft_above floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4/vif stb ;
plot npp.*student.;
run;
* Run model after Applying Log Transformation on X-var;
proc reg data=sample;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3
Q4/vif stb influence r;
plot npp.*student.;
run;
* remove both outliers and influential points;
data sample_R1;
set sample;
if _n_=21 then delete;
if _n_=27 then delete;
if _n_=69 then delete;
if _n_=443 then delete;
if _n_=581 then delete;
if _n_=803 then delete;
if _n_=854 then delete;
if _n_=1262 then delete;
run;

proc reg data=sample_R1;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h
waterfront view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3
Q4/vif stb ;
plot npp.*student.;
run;

proc corr data=sample_R1;
```

```
var bathrooms bedrooms ln_sqft_living ln_sqft_lot ln_sqft_above floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4;
run;

* R (ln_sqft_living,ln_sqft_above )=0.87~0.9--> remove ln_sqft_above to solve the
multicollinearity problem;

proc reg data=sample_R1;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4/vif stb ;
run;
* Refit the model;
* Assuming interaction terms should be added if correalation coefficient is greater than 0.5;
proc corr data=sample_R1;
var ln_price bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4;
run;

***creating interaction terms;
data sample_R1I;
set sample_R1;
*Based on the assumption that interaction terms should be added if correalation coefficient is
greater than 0.5;
bb=bathrooms*bedrooms;
bedliving=bedrooms*ln_sqft_living;
bathliving=bathrooms*ln_sqft_living;
bathfloor=bathrooms*floor_h;
******************************;
*Based on the assumption that location and  Square Feet has jonit effect on House Price;
living_NE = ln_sqft_living*NE;
living_SW = ln_sqft_living*SW;
living_SE = ln_sqft_living*SE;
RUN;


* Apply  interaction terms to the model---Location~Square footage;
proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 living_NE living_SW
living_SE /vif stb ;
```

```
run;

* VIF of interacton terms living_NE living_SW living_SE is > 10 , shows the Multicollinearity
problem. need to center these predictors.;
proc corr data=sample_R1I;
var ln_price bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 living_NE living_SW
living_SE;
run;

** adding interaction terms one by one and testing the P value;

*+ bb: interaction term is significant enough to stay in the model/ p<0.0001;
proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bb /vif stb ;
run;

*+ bedliving:  interaction term is significant enough to stay in the model/ p<0.0001;
proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bedliving /vif stb ;
run;

*+ bathliving: interaction term is significant enough to stay in the model/ p<0.0001;
proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bathliving /vif stb ;
run;

*+bathfloor: interaction term is significant enough to stay in the model/ p<0.0001;
proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bathfloor /vif stb ;
run;


proc reg data=sample_R1I;
model ln_price = bathrooms bedrooms ln_sqft_living ln_sqft_lot floor_h waterfront view_good
condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 living_NE living_SW
```

```
living_SE bb
bedliving bathliving bathfloor/vif stb ;
run;




**centering method,get the mean value of the continous variables;
proc corr;
var ln_price bathrooms bedrooms ln_sqft_living ;
run;

data sample_R1IC;
set sample_R1I;
*Center the continuous variables to handle the Multicollinearity Issues in Interaction Models ;
ln_sqft_living_c = 7.53931 - ln_sqft_living;
bathrooms_c = 2.10908 - bathrooms;
bedrooms_c = 3.34517 - bedrooms;
living_NE_c = ln_sqft_living_c*NE;
living_SW_c = ln_sqft_living_c*SW;
living_SE_c = ln_sqft_living_c*SE;
bb_c = bathrooms_c*bedrooms_c;
bathliving_c = bathrooms_c*ln_sqft_living_c;
bedliving_c = bedrooms_c*ln_sqft_living_c;
bathfloor_c = bathrooms_c*floor_h;
run;

* Apply centered interaction terms to the model;
proc reg data=sample_R1IC;
model ln_price = bathrooms_c bedrooms_c ln_sqft_living_c ln_sqft_lot floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bb_c
bathliving_c bedliving_c  bathfloor_c   living_NE_c living_SW_c living_SE_c /vif stb ;
run;

*Split the original sample data into training and testing set;
proc surveyselect data=sample_R1IC
out=house_split seed=1 samprate=0.75 outall;
run;
data house_train (where = (Selected = 1));
set house_split;
```

```
run;
data house_test (where = (Selected = 0));
set house_split;
run;


*model selection on the training set;
title M 1 forward selection method;
proc reg data=house_train;
model ln_price = bathrooms_c bedrooms_c ln_sqft_living_c ln_sqft_lot floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bb_c
bathliving_c bedliving_c  bathfloor_c   living_NE_c living_SW_c
living_SE_c/selection=forward sle=0.05 sls=0.05;
run;

title Final Model 1;
proc reg data=house_train;
model ln_price =ln_sqft_living_c SE SW bathliving_c view_good grade_h bedrooms_c
living_SE_c renovated grade_b waterfront Q2 bedliving_c bb_c NE/vif stb;
run;


title M 2 Cp selection method;
proc reg data=house_train;
model ln_price =bathrooms_c bedrooms_c ln_sqft_living_c ln_sqft_lot floor_h waterfront
view_good condition_good grade_b grade_a grade_h renovated NE SW SE Q2 Q3 Q4 bb_c
bathliving_c bedliving_c  bathfloor_c   living_NE_c living_SW_c living_SE_c /selection=cp;
run;


title Model 2;
proc reg data=house_train;
model ln_price =bathrooms_c bedrooms_c ln_sqft_living_c waterfront view_good
condition_good grade_b grade_a renovated NE SW SE Q2 bb_c bathliving_c bedliving_c
living_SE_c/vif stb;
run;


*remove bathrooms_c and condition_good as they are not significant predictors;
title Final Model 2;
```

```
proc reg data=house_train;
model ln_price = bedrooms_c ln_sqft_living_c waterfront view_good grade_b grade_a renovated
NE SW SE Q2 bb_c bathliving_c bedliving_c living_SE_c/vif stb;
run;

* select Model 2 as my final model;

proc reg data=house_test;
model ln_price = bedrooms_c ln_sqft_living_c waterfront view_good grade_b grade_a renovated
NE SW SE Q2 bb_c bathliving_c bedliving_c living_SE_c/vif stb;
run;

* remove insignificant predictors in testing set;
* remove bedrooms_c,grade_b,grade_a,renovated,NE,Q2;
title Final Model on testing;
proc reg data=house_test;
model ln_price = ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c bedliving_c
living_SE_c/vif stb;
run;


title Final Model on training;
proc reg data=house_train;
model ln_price = ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c bedliving_c
living_SE_c/vif stb;
run;



*Take the first two observations from training set ;
proc print data=house_train;
run;

data pred;
input ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c bedliving_c
living_SE_c;
datalines;
-0.04139  0  0   1  0     0.58340      0.03687       0.02710      0
-0.63821  0       0   0  0     -0.13493       0.24949       -0.22029     0
;
```

```
*Compute two predictions including the prediction intervals using the regression model.;
data prediction;
set pred house_test;
run;

proc reg data=prediction;;
model ln_price =  ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c
bedliving_c living_SE_c/ p clm cli alpha=0.05;
run;



*Cross-Validation test;
* create new variable new_y = ln_price for training set, and = NA * for testing set;
data house_validation;
set house_split;
if selected then new_y=ln_price;
run;

/* get predicted values for the missing new_y in test set for the fitted model*/
title "Validation - Test Set";
proc reg data=house_validation;
model new_y = ln_sqft_living_c waterfront view_good  SW SE  bb_c bathliving_c bedliving_c
living_SE_c/vif stb;
output out=outm1(where=(new_y=.)) p=yhat;
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_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;


*** apply the model on validation testing set;
proc reg data=outm1;
model ln_price = ln_sqft_living_c waterfront view_good SW SE bb_c bathliving_c bedliving_c
living_SE_c/vif stb;
run;

****END OF Yusheng Zhu's CODE;
```

Reference:

Smith, M. (2014). Case 12: Housing Prices Multiple Regression – Multicollinearity and Model Building. *University of Colorado Denver Business School.*, pp.3-9. Available at: https://www.jmp.com/content/dam/jmp/documents/en/academic/case-study-library/case-study-library-12/business-case-studies/12-housingprices.pdf [Accessed February 27, 2017]

Chica Olmo, J. (2007). Prediction of Housing Location Price By a Multivariate Spatial Method: Cokriging. *Journal of Real Estate Research*, 29(1), pp.7-12. Available at: http://pages.jh.edu/jrer/papers/pdf/past/vol29n01/05.91_114.pdf [Accessed February 27, 2017]

Glasserman, P. (2001). . Linear Regression Managerial Statistics. *Columbia Business School class materials*. Available at: https://www0.gsb.columbia.edu/faculty/pglasserman/B6014/Regression.pdf [Accessed February 27, 2017]

Hu, G., Wang, J. and Feng, W. (2013). Multivariate Regression Modeling for Home Value Estimates with Evaluation Using Maximum Information Coefficient. *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing 2012*, 443, pp.69-81. Available at: http://www.acisinternational.org/Springer/SamplePaper.pdf [Accessed February 27, 2017]

Ng, A. (2015). Machine learning for a London housing price prediction mobile application. *Imperial College London*. Available at: http://www.doc.ic.ac.uk/~mpd37/theses/2015_beng_aaron-ng.pdf [Accessed February 27, 2017]

Pardoe, I. (2007). Modeling home prices using realtor data. *Journal of Statistics Education*, 16(2), pp.6-9. Available at: http://ww2.amstat.org/publications/jse/v16n2/datasets.pardoe.pdf [Accessed February 27, 2017]