King County House Sales Study

Author: Kamile Yagci

Overview

In this project, I analyzed the King County House Sales. I used the Multiple Linear Regression to model the data and predict the house sale prices in King County.

Business Problem

The Windermere Real Estate Agency hired me to develop a model to predict the house sale prices in King County. The agency plans to use the results of this study when advising their customers/homeowners on determining the value of their houses. They believe that the pricing the house correctly will increase the efficiency of sales. The agency also would like to learn about the effect of renovation on house sale price, so they can advise the customers to do renovation or not.

Questions:

- What are the main predictors for House Sale Price?
- Create a model to predict the House Sale Price.
- Do house renovation affects the Sale Price?

Method

I followed the following steps in this project:

1. Data
    - Load
    - Explore
    - Clean
    - Check correlation
2. Model - Apply multiple regression and do validation
    - Baseline model with one best predictor
    - Second and Third models (Search for the next best predictors)
    - Final Model with five best predictors
3. Interpret
    - Interpret Final Model
    - Check multiple regression assumptions
4. Future work

Data

Source

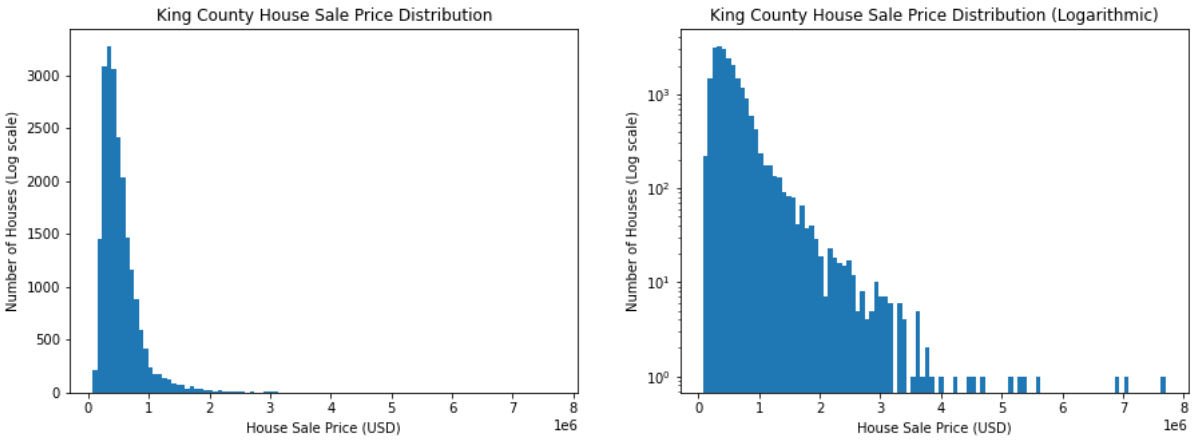
I used the King County House Sales Data for this study. The data file is 'kc_house_data.csv'

Data Exploring

Data contains

- information on 21597 houses sold between May 2014 - May 2015.
- 21 columns: 'id', 'date', 'price', 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'grade', 'sqft_above', 'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode', 'lat', 'long', 'sqft_living15', 'sqft_lot15'

Since our main goal is predicting sale price, let's look at the distribution of the House Sale Prices. The left plot shows the number of houses in y-axis and sale price on the x-axis. The right plot shows the same distribution on log scale. It will be easier to see the outliers on the log scale.



The 'price' looks like a left-skewed normal distribution. There are fewer houses above $3,000,000. I guess they are outliers on the data.

Data Cleaning

The Project 2 infomation page recommends to remove these columns to ease the analysis: ['date', 'view', 'sqft_above', 'sqft_basement', 'yr_renovated', 'zipcode', 'lat', 'long', 'sqft_living15', 'sqft_lot15']

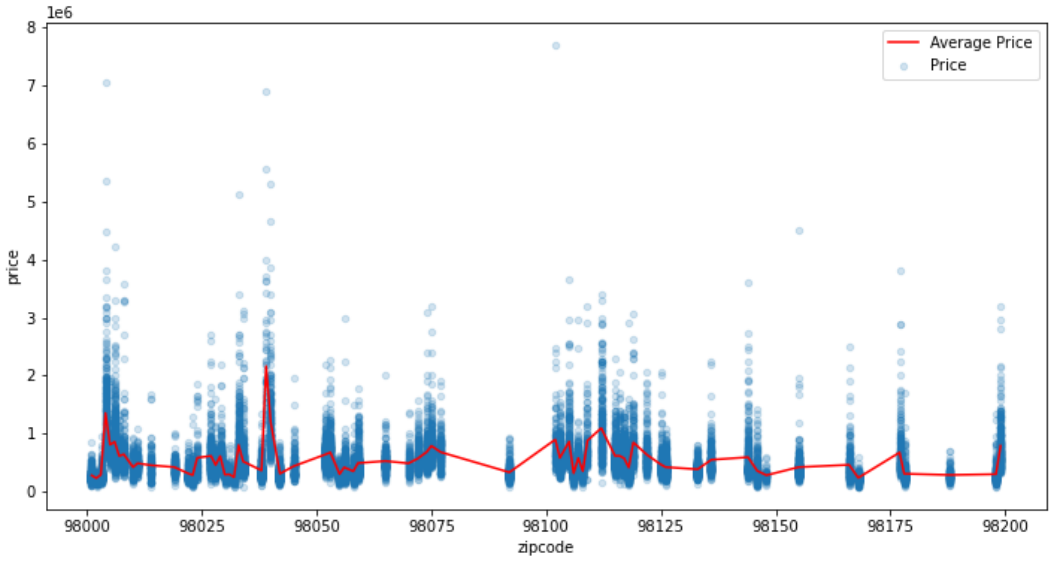
I follow the advise and remove these variables except 'yr_renovated' and maybe 'zipcode'

- 'date' and 'view' are apparently not significant predictors for Sale Price; good to remove

- sqft_above' and 'sqft_basement' will have multicollinearity with 'sqft_living'; so better to remove
- 'lat' and 'long' determine the location; I will use the zipcode for location and so no need for these variables
- 'sqft_living15' and 'sqft_lot15' may have multicollinearity with 'sqft_living' and 'sqft_lot'; OK to remove

One of my business question is about the effect of renovation on sale price; so needs to keep 'yr_renovated'

In general, location is an important factor in house prices. I want to take a closer look at 'zipcode', before making a decision on keeping it or not. The below plot shows the sale price vs zipcode and the average price per zipcode.



The house prices peak at few zipcodes. How significant is this? I decided to keep 'zipcode' in my data. I have applied hot-encoding to create dummy variables for each zipcode.

Remove unneccessary columns

The columns: 'date', 'view', 'sqft_above', 'sqft_basement', 'lat', 'long', 'sqft_living15', 'sqft_lot15' are dropped.

Handle missing values

- Null values in 'waterfront' is filled with zero.
- A boolean variable is created for 'yr_renovated' and the null/zero values in yr-renovated are filled with yr_built.

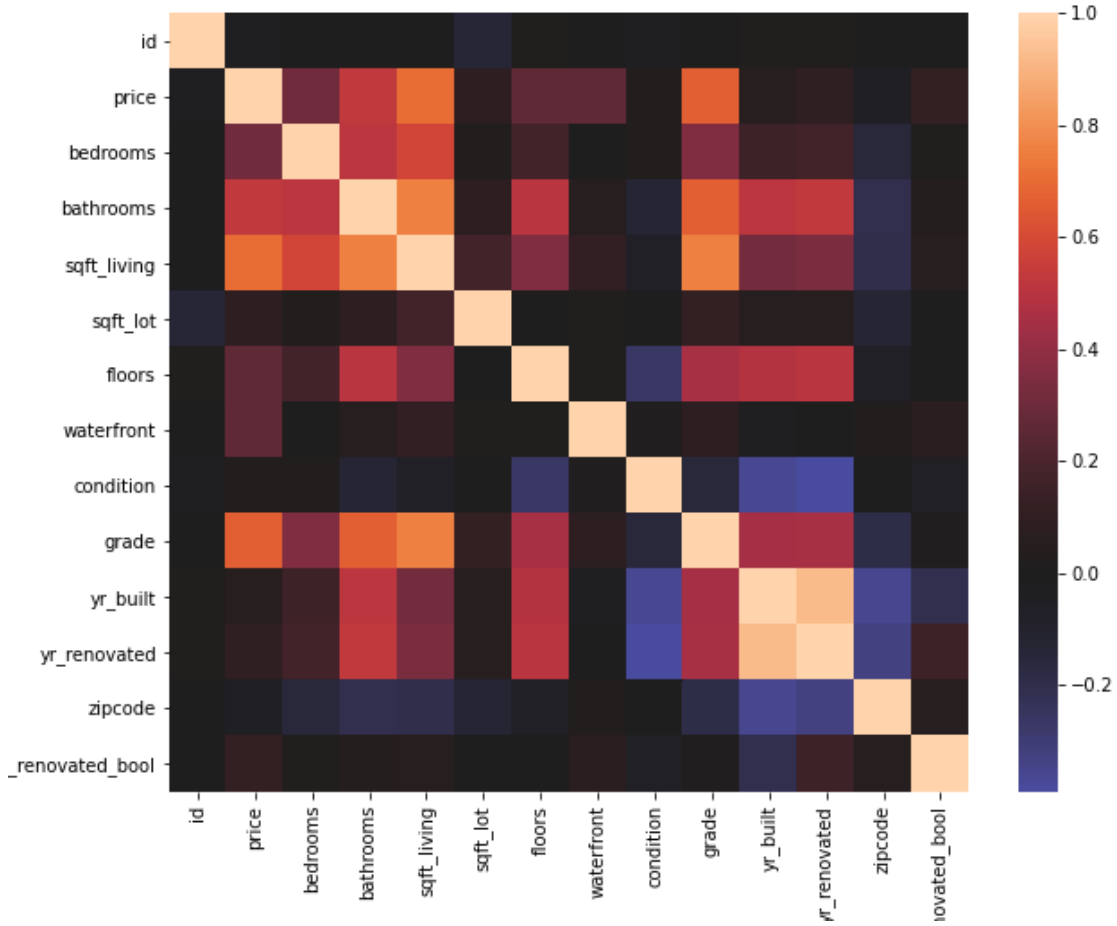
Final data columns

After the completion of data cleaning, the full list of data columns are:

'id', 'price', 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', 'waterfront', 'condition', 'grade', 'yr_built', 'yr_renovated', 'yr_renovated_bool', 'zip_98002', 'zip_98003', 'zip_98004', 'zip_98005', 'zip_98006', 'zip_98007', 'zip_98008', 'zip_98010', 'zip_98011', 'zip_98014', 'zip_98019', 'zip_98022', 'zip_98023', 'zip_98024', 'zip_98027', 'zip_98028', 'zip_98029', 'zip_98030', 'zip_98031', 'zip_98032', 'zip_98033', 'zip_98034', 'zip_98038', 'zip_98039', 'zip_98040', 'zip_98042', 'zip_98045', 'zip_98052', 'zip_98053', 'zip_98055', 'zip_98056', 'zip_98058', 'zip_98059', 'zip_98065', 'zip_98070', 'zip_98072', 'zip_98074', 'zip_98075', 'zip_98077', 'zip_98092', 'zip_98102', 'zip_98103', 'zip_98105', 'zip_98106', 'zip_98107', 'zip_98108', 'zip_98109', 'zip_98112', 'zip_98115', 'zip_98116', 'zip_98117', 'zip_98118', 'zip_98119', 'zip_98122', 'zip_98125', 'zip_98126', 'zip_98133', 'zip_98136', 'zip_98144', 'zip_98146', 'zip_98148', 'zip_98155', 'zip_98166', 'zip_98168', 'zip_98177', 'zip_98178', 'zip_98188', 'zip_98198', 'zip_98199'

Correlation

The correlation heat map will help to observe the correlations between variables. The dummy zipcodes are not included in the correlation study.



I also looked at the correlation values for 'price':

- id: -0.016772
- price: 1.000000
- bedrooms: 0.308787
- bathrooms: 0.525906
- sqft_living: 0.701917
- sqft_lot: 0.089876
- floors: 0.256804
- waterfront: 0.264306
- condition: 0.036056
- grade: 0.667951
- yr_built: 0.053953
- yr_renovated: 0.097541
- zipcode: -0.053402
- yr_renovated_bool: 0.117543

Observations:

- There is good correlation between Price and (bedrooms, bathrooms, sqft_living, grade).
- There is also correlation among bedrooms, bathrooms, sqft_living, grade. We have to consider multicollinearity effects using them in modeling. I should only use one of them in my model.
- There is also high correlation between yr_renovated and yr_built. Again, only one of them can be used in my model.
- The variables that have highest correlation with Sale Price are 'sqft_living' and 'grade'. Definition of these variables:
  - sqft_living: square footage of the home (continious variable)
  - grade: overall grade given to the housing unit, based on King County grading system (categorical varible)

Model

My main goal for this project is predicting the House Sale Price. Therefore 'price' variable is my target, dependent varaible (X). And all other variables are predictors, independent variables (y).

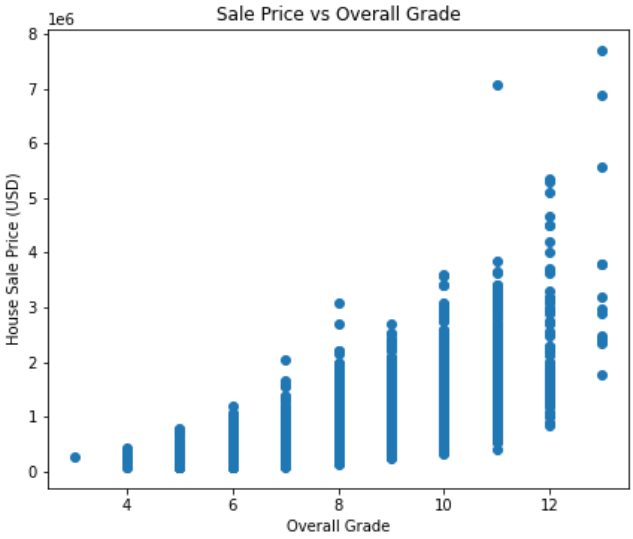
This modeling process will tell me which variables are good predictors and it will produce a fit algorithm which calculates the predicted sale price.

I will use multiple linear regression for this study.

Baseline Model

I will start the modeling with only one variable. Then I will try to improve my baseline model by adding more variables step by step.

Based on my correlation study, the Sale price has best correlation with 'sqft_living' and 'grade'.



The high correltaion is visible on both plots.

I choose to use 'sqft_living' in my baseline model since it is a continuous variable.

There are three steps in modeling:

1. Seperate data into train and test splits

   I used sklearn train_test_split function to split data. I allocated 75% of the data for training and 25% for testing (default).

2. Apply Linear Fit to training data and make predictions

   I used sklearn LinearRegression function to fit the data.

   Linear equation with one independent variable is y = m.x + b, where x is the independent variable, y is dependent varaible, m is slope and b is y-intercept.

   In our fit, independent variable (x) is 'sqft_living' and dependent variable (y) is 'price'. When Linear Regression fit applied, it produces two values: slope (coefficient) and y-intercept (intercept).

   Here are the results of our baseline fit:

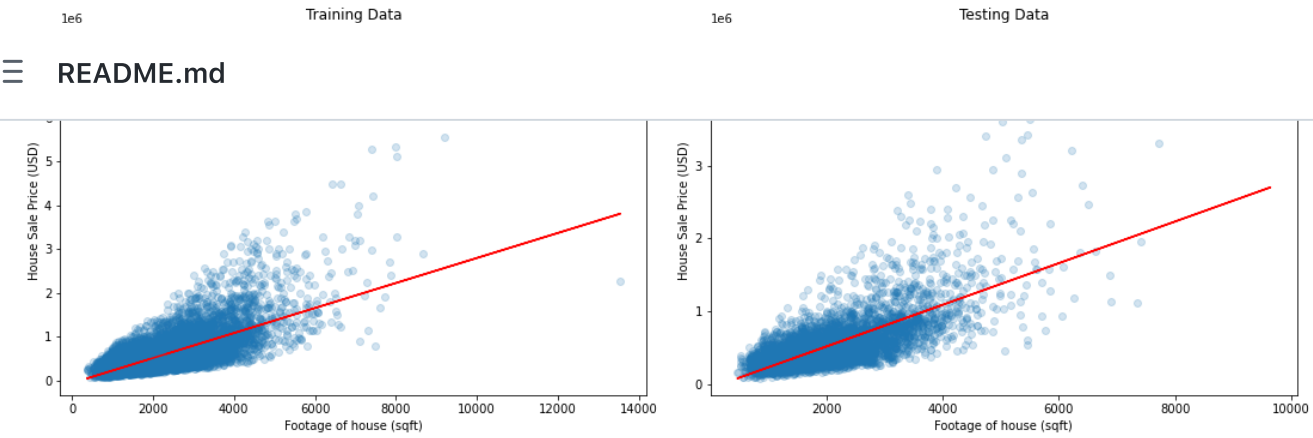
   - Slope: 285.58593563
   - y-intercept: -53321.493253810564

3. Validate model:

   I calculated the predicted price and then calculated R squared for both training and testing data, and then plotted:

- R squared for Training: 0.4951005996564265
- R squared for Testing: 0.48322207729033984

README.md



The R squared score for training and test scores are similar. The Training and Testing graphs also show similar fitting behavior. Therefore, we can say that the baseline model is a good fit, not overfitted or underfitted.

Model Validation with Multiple splits of data

Seperating the data in training and test splits is a random process. We can improve the validation by repeating the process (split, fit, predict, validate) multiple times and finding the mean R squared scores.

I used the sklearn functions cross_validate and ShuffleSplit moethods for this purpose.

Here is the R squared results:

- Mean R squared for Training: 0.49093640472300526
- Mean R squared for Testing: 0.4960198399390211

As the number of splits increase, the difference between the Train and Test score decreases.

The R squared value for baseline model with one predictor (sqft_living) is around 0.49. However, the score is not high enough. We should try to improve our model by adding more predictors.

Second Model with 2 predictors

In the second model, I plan to add a second predictor to my fit in order to improve the R squared score.

Firstly, I iterate over all the remaining predictors to find the best 2nd predictor.

Here is the R squared scores for LinearRegression fits on training and testing data using two predictors ('sqft_living' + 2nd predictor)

	two_predictors	R_squared_train	R_squared_test
0	[sqft_living, zip_98004]	0.533440	0.520127
1	[sqft_living, waterfront]	0.529563	0.528285
2	[sqft_living, yr_built]	0.526124	0.518243
3	[sqft_living, zip_98039]	0.517281	0.506499
4	[sqft_living, yr_renovated]	0.516954	0.507128
5	[sqft_living, zip_98112]	0.511713	0.506115
6	[sqft_living, zip_98040]	0.508561	0.496405
7	[sqft_living, zip_98023]	0.504133	0.492635
8	[sqft_living, zip_98038]	0.502710	0.491157
9	[sqft_living, zip_98042]	0.502537	0.492249
10	[sqft_living, zip_98105]	0.502214	0.491189
11	[sqft_living, zip_98092]	0.501713	0.491271
12	[sqft_living, yr_renovated_bool]	0.501480	0.491107
13	[sqft_living, zip_98119]	0.501165	0.493318
14	[sqft_living, condition]	0.501093	0.489588

Observations:

- Adding a 2nd predictor to baseline model improved the R squared value both on training and testing data.
- As I guessed, zipcode plays a significant role in House Sale price.
- The R_squared values for top 2nd predictors are very close to each other.
- Moreover, model validation on test data is also good for all.

I will not choose a 2nd predictor for this step. Instead I will search for the best set of predictors.

Third Model with 3 predictors

In this step, I will try to find the best set of 2nd and 3rd predictors in addition to the 'sqft_living'. Here are the results:

	three_predictors	R_squared_train	R_squared_test
0	[sqft_living, zip_98004, waterfront]	0.569135	0.566727
1	[sqft_living, yr_built, zip_98004]	0.562154	0.552604
2	[sqft_living, yr_built, waterfront]	0.556858	0.558546
3	[sqft_living, zip_98039, zip_98004]	0.556583	0.544304
4	[sqft_living, zip_98004, yr_renovated]	0.553657	0.543087
5	[sqft_living, zip_98039, waterfront]	0.551630	0.552382
6	[sqft_living, zip_98112, zip_98004]	0.551043	0.544249
7	[sqft_living, waterfront, yr_renovated]	0.549054	0.549292
8	[sqft_living, zip_98040, zip_98004]	0.548339	0.534251
9	[sqft_living, yr_built, zip_98039]	0.546934	0.539806
10	[sqft_living, zip_98112, waterfront]	0.546886	0.552160
11	[sqft_living, zip_98040, waterfront]	0.541900	0.539191
12	[sqft_living, zip_98023, zip_98004]	0.541846	0.528913
13	[sqft_living, zip_98105, zip_98004]	0.541012	0.528563
14	[sqft_living, zip_98038, zip_98004]	0.540323	0.527330

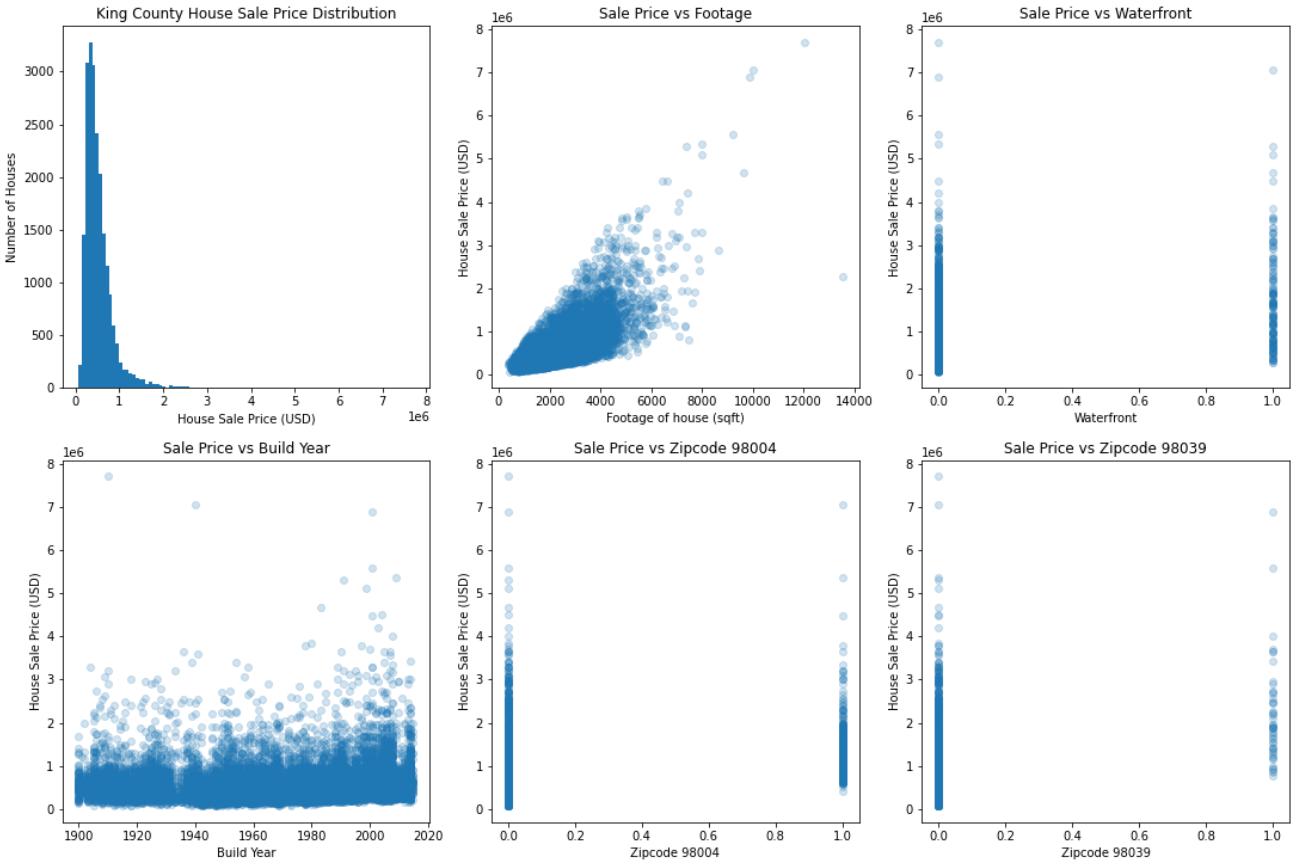
Observations:

- As the number of the predictors increase, R squared value increased.
- The validation is still good.
- The predictors sets which includes variables "sqft_living, waterfront, zip_98004, yr_built, and zip_98039" give similar R_squared values for training and test.

Final Model with 5 predictors

In final model, I will use all the top 5 predictors: sqft_living, waterfront, zip_98004, yr_built, and zip_98039.

Let's first look at the scatter graphs of these predictors.



Now, I apply Multiple Linear regression on top 5 predictors to see how much my model improves.

Results of final model:

- Final model predictors: ['sqft_living', 'waterfront', 'yr_built', 'zip_98004', 'zip_98039']
- R squared for Training: 0.6159271322430281
- R squared for Testing: 0.6178922353975156

As I guessed, adding more zipcodes will improve the model. However, I stop adding new predictors at this point.

Validation looks pretty good.

Let's double check the validation with multisplitter:

- Final model predictors: ['sqft_living', 'waterfront', 'yr_built', 'zip_98004', 'zip_98039']
- Mean R squared for Training: 0.6132165670165712
- Mean R squared for Testing: 0.6242740771569494

RMSE calculation

Here are the MSE and RMSE values for final model.

- MSE = 45489203300.832375
- RMSE = 213281.9807223113

RMSE value is quite large. This is not a good sign.

Multiple Linear Regression in Statsmodels

We can also use statmodels for multiple linear regression modeling. I did run the Statmodel OLS on different set of predictors. Here are my findings:

As the number of predictors used in modeling increase, the R squared value increases. However, also the conditon number increases, which is not good.

- R_squared = 0.795 and Cond. No. = 6.17e+11 when all predictors used (including hot-encoded zipcode and boolean yr_renovated).
- R_squared = 0.617 Cond. No. = 2.07e+05 when five final model predictors used ('sqft_living', 'waterfront', 'yr_built', 'zip_98004', 'zip_98039').
- R_squared = 0.592 Cond. No. = 4.75e+04 when four predictors used ('sqft_living', 'waterfront', 'zip_98004', 'zip_98039').
- R_squared = 0.493 Cond. No. = 5.63e+03 when one predictor, 'sqft_living', used

Effect of House Renovations

I would like to see if and how much house renovations effect the House Sale Prices.

Let's apply the linear fit with one predictor only: 'yr_renovated'.

Result:

- Slope: 1300.35687138
- y-intercept: -2022535.979409572
- R squared for Training: 0.010206187304151793
- R squared for Testing: 0.006126113636745867

R squared value is very low. Apparently this is not a good fit.

I will now apply the linear regression on final model predictors + renovation year. I want to observe how much the R squared will improve.

- Final model + Renovation predictors: ['sqft_living', 'waterfront', 'yr_built', 'zip_98004', 'zip_98039', 'yr_renovated']
- R squared for Training: 0.6167249032808759

- R squared for Testing: 0.618146687048683

Renovation does improve the R squared score very slightly on training data, but not on testing data.

I conclude that House Renovation doesn't have significant effect on House Sale Price.

Final Model Interpretation

Here is the parameters for my final model:

- Coefficients:
  - sqft_living: 2.844667e+02
  - waterfront: 8.040627e+05
  - yr_built: -2.071868e+03
  - zip_98004: 6.201505e+05
  - zip_98039: 1.162237e+06
- Intercept: 4014990.7576159136

Observations:

- The y-intercept is quite large, about 4,000,000, and the price goes down with year_built.
- The 'year_built' has negative coefficient, which is interesting. The old houses are more expensive than the new ones. Are the old, established neigborhoods more valuable? Are old neighborhoods close to downtown? Why? This can be investigated for future study.

Here is the linear equation for our final model:

House Sale Price = 4014990.7576159136 + (2.844667e+02 * sqft_living) + (8.040627e+05 * waterfront) + (-2.071868e+03 * yr_built) + (6.201505e+05 * zip_98004) + (1.162237e+06 * zip_98039)

In general, the format of the linear equation:

y = b + mx (for one independent variable)

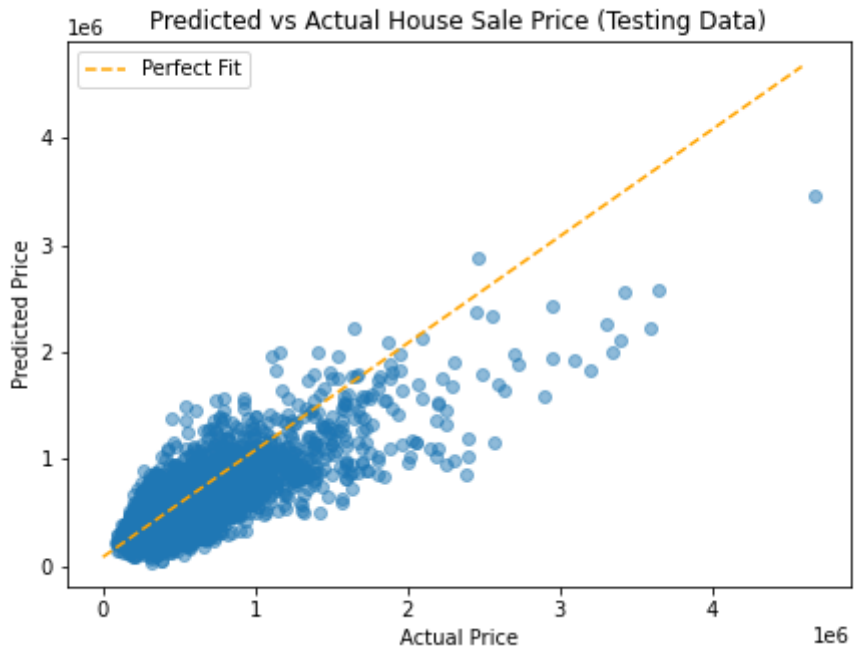
y = b + m1x1 + m2x2 + m3x3 + m4x4 + m4x5 (for five independent varibles)

Some sale price prdictions:

1. sqft_living=2800, waterfront=0, year_built=1978, zipcode_98004=0, zipcode_98039=0 => price = 713342.1052993718 # control set
2. sqft_living=4200, waterfront=0, year_built=1978, zipcode_98004=0, zipcode_98039=0 => price = 1111595.4730236786 # sqft_living effect
3. sqft_living=2800, waterfront=1, year_built=1978, zipcode_98004=0, zipcode_98039=0 => price = 1517404.8278655098 # waterfront effect
4. sqft_living=2800, waterfront=0, year_built=2015, zipcode_98004=0, zipcode_98039=0 => price = 636682.9802501751 # year_built effect effect
5. sqft_living=2800, waterfront=0, year_built=1978, zipcode_98004=1, zipcode_98039=0 => price = 1333492.6308398792 # zipcode 98004 effect
6. sqft_living=2800, waterfront=0, year_built=1978, zipcode_98004=0, zipcode_98039=1 => price = 1875578.8361759782 # zipcode 98039 effect

Investigating Linearity

I will check the linearity between the model predicted value and actual value on the test data.



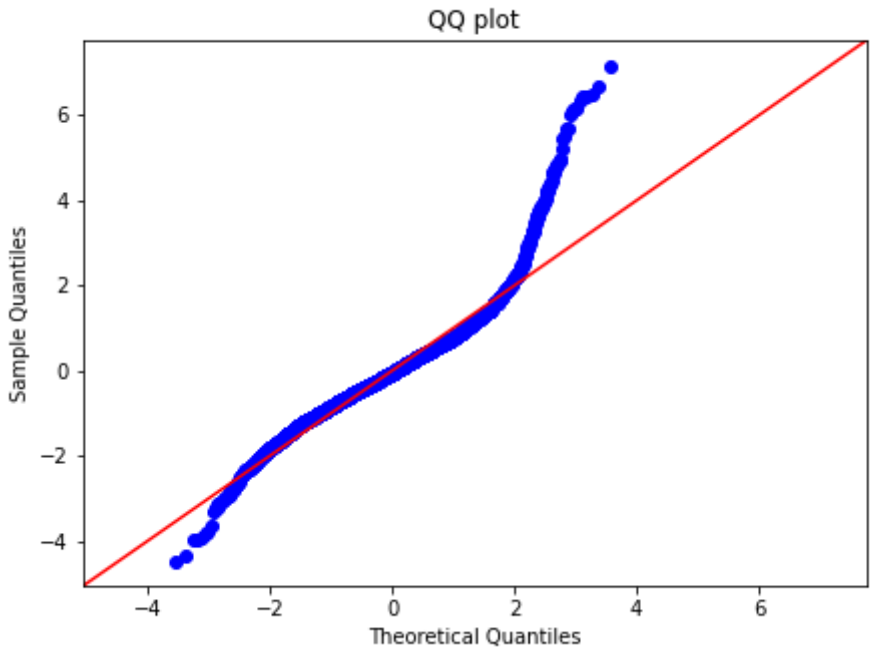
Observations:

- The plot shows linear relation between the Actual and Predicted price.
- However, it is important to note that, at high sale prices (above 2,000,000), the predicted value is deflecting away from the perfect fit. I believe these are outliners.

I conclude that Linearity assumption holds for the majority of the data, except outliers at high sale prices.

Investigating Normality

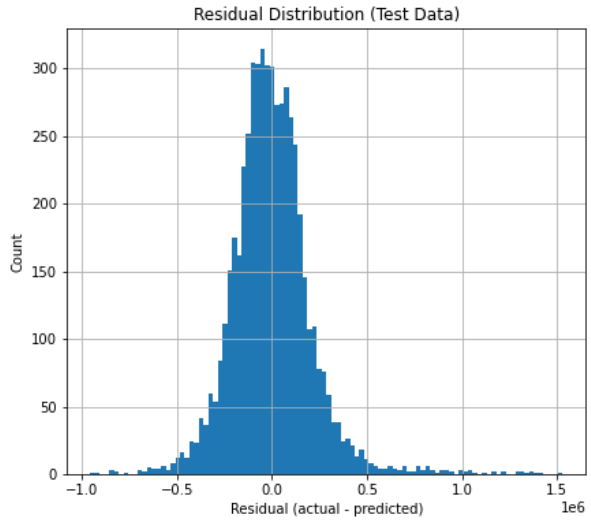
We will check normality by plotting the residual distribution vs normal distribution.

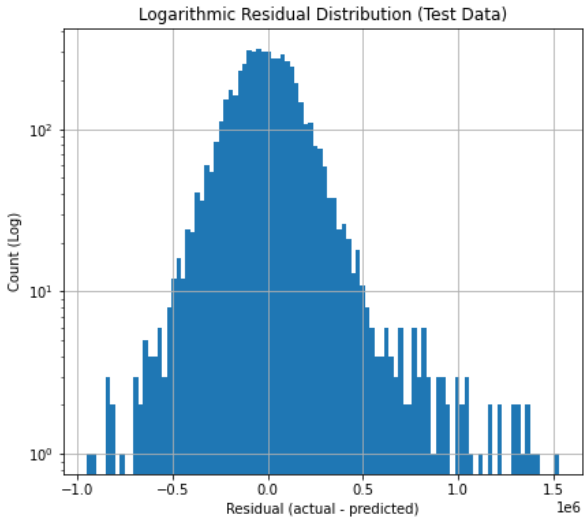


Observations:

- The data looks like normal distribution close to center, but it is skewed at the tails/edges.
- I believe skewness is caused by the outliners at the high sale prices.

Let's look at the residual distributions closely.





Residual distribution looks normal except on the tails. The data in tails causes a bit skewness. I believe the residual values on the tails are caused by outliers, the houses at high sale prices.

I conclude that Normality assumption holds for the majority of the data, except outliers at high sale prices.

Investigating Multicollinearity (Independence Assumption)

I used statsmodel variance_inflation_factor to calculate the Multicollinearity.

Multicollinearity scores:

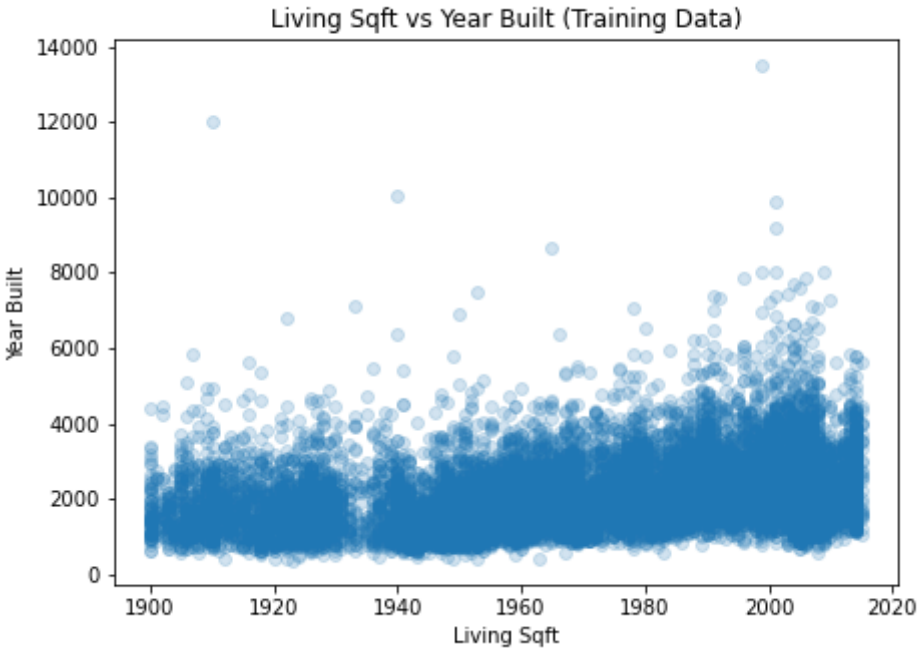
- sqft_living: 6.460982
- waterfront: 1.019214
- yr_built: 6.320784
- zip_98004: 1.029405
- zip_98039: 1.010471

Observations:

- The VIF values for variables, 'waterfront' 'zip_98004', 'zip_98004' are around 1. They are not correlated.
- However, the VIF values are around 6.4 for sqft_living and yr_built. These variables look correlated and causes multicollinearity.

I conclude that Independence Assumption is violated since significant multicollinearity is observed.

Even though the correlation is not high for the sqft_living and yr_built, it still caused considerable multicollinearity. Why are they correlated?



The plot shows slight correlation. The new houses looks like a bit larger.

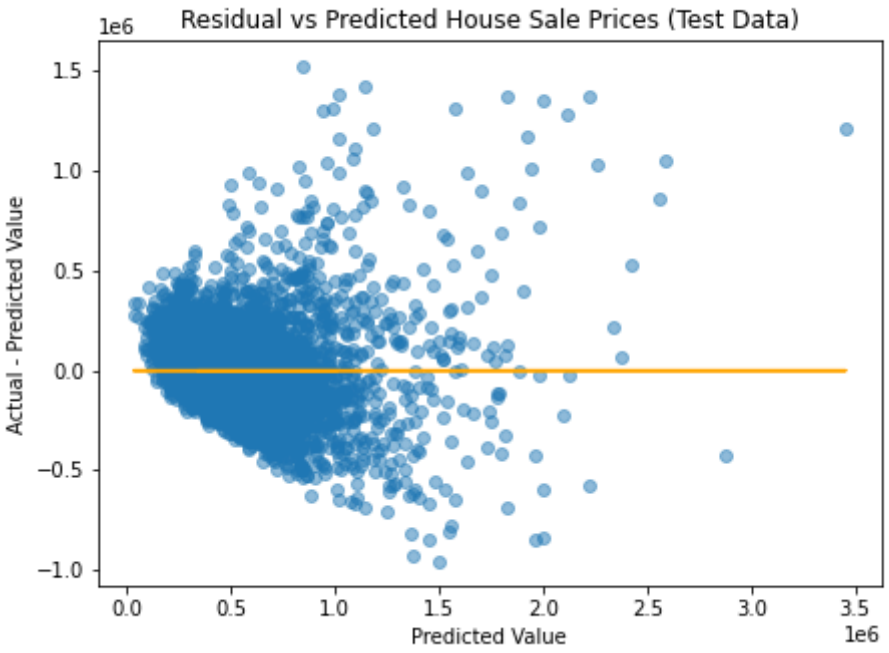
The 'sqft_living' and 'yr_built' are main predictors. Should I remove 'year_built' from model? But it will decrease the R squared.

Using Statmodel OLS fit, I calculated R squared with and without 'yr_built'

- R_squared = 0.617 Cond. No. = 2.07e+05 when five final model predictors used ('sqft_living', 'waterfront', 'yr_built', 'zip_98004', 'zip_98039').
- R_squared = 0.592 Cond. No. = 4.75e+04 when four predictors used ('sqft_living', 'waterfront', 'zip_98004', 'zip_98039').

Investigating Homoscedasticity

I will look at the Residual vs Predicted values for house prices on testing data. The shape of the graph will tell me about the Homoscedasticity.



Observations:

- The cone/funnel shape is observed on data.
- Funnel gets larger at high house sale prices.

I conclude that Homoscedasticity assumption is violated.

Linear Regression Assumptions Conclusion

- Linearity assumption holds for the majority of the data, except outliers at high sale prices.
- Normality assumption holds for the majority of the data, except outliers at high sale prices.
- Independence Assumption is violated since significant multicollinearity is observed.
- I conclude that Homoscedasticity assumption is violated.

Future Work

- Study outliers:
  - I guess one of the main causes for the assumption violations is outliers.
  - Remove from analysis?
- Study correlation between sqft_living and yr_built:
  - The multicollinearity caused by their correlation affects the model.
  - Should 'yr_built' be removed from model? Advantages and disadvantages?

- Explore Homoscedasticity:
  - Homoscedasticity is observed at low house sale prices as well as high.
  - How can we avoid it?

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

- **Jupyter Notebook** 100.0%