Springboard--DSC Program

Capstone Project 2

An Exploration of Housing Sales in Washington D.C. and King County

By: Garrett Yamane

March 21st, 2019

Table of Contents

1. Introduction

Washington D.C., the capital of the United States of America, is home to many historical landmarks and our beloved government. King County, in Washington state, encompasses a variety of geographically diverse cities, from rural farms to the tech hub of Seattle and Bellevue. Both locations offer so much opportunity, and are places that are highly sought-after to move to. I am from the Bay Area where the cost of living has been increasing more and more throughout Silicon Valley's rapid progression.

When people move, do different housing features impact the price of a house sale? Are prices of houses being sold affected differently by these features based on location? These questions create the foundation of what I would like to explore in Washington D.C. and King County's housing sales.

2. Problem Statement

Moving can be a major hurdle for one's life. Whether it is the first time moving out on your own after graduating from college or if you are beginning to start a family and looking to settle down, buying one's own place can be quite expensive. Knowing the housing market and what to expect if you are moving to an unfamiliar place can be a challenging task. But what if there was a way to model housing trends and predict what you may have to pay? One model cannot accurately summarize every housing market, but being able to take two geographically separate locations and compare how house features are weighted differently when building housing price models can prove beneficial to both real estate agents and their clients when deciding where to move and what to buy. The goal of this project is to build individual accurate predictive machine learning model for both Washington D.C. and King County in order to both closely examine what features heavily impact the price of a house sale and to help prospective buyers understand the market they are buying into.

3. Target Client

Real estate agents need to be very educated and aware of the housing market when helping clients buy a new place. Not only is it important to know how much a house is going to cost, but it is also helpful to know what factors will increase the price and what features carry the highest value for different locations. People moving into the suburbs may value a place with more bedrooms and living space for raising a family, whereas those moving into a larger city may want a view with newer renovations. Real estate agents for major cities and their clients can benefit from a study that targets two different geographical locations through predictive modeling: King County in Washington State and D.C.

4. Data Wrangling

a) Gathering the Data

The data will come from two different datasets from Kaggle:

- King County, Washington Dataset [1]
- Washington D.C. Dataset [2]
-

There are 21.6K rows in the King County dataset, and 159K rows in the Washington D.C. dataset. However, the King Country dataset only contains house sales between May 2014 and May 2015, whereas the Washington D.C. dataset contains sales dating back to 1995. For consistency, I only sampled sale dates that both datasets have in common which decreased the number of rows of the D.C. dataset down to around 14K. I also looked at the following features that each dataset has in common:

- Price
- Sale Date
- # bathrooms
- # bedrooms
- Living SQFT
- Lot SQFT
- Stories

[1] https://www.kaggle.com/harlfoxem/housesalesprediction
[2] https://www.kaggle.com/christophercorrea/dc-residential-properties#DC_Properties.csv

- Condition
- Grade
- Year Built
- Year Remodeled

b) Cleaning and Wrangling the data

Ultimately, I wanted to combine the data sets into a single data frame for simplicity. To accomplish this, it was important to not only look at the columns that each data set has in common, but to also make sure that the data in each corresponding column is of the same type prior to merging the two data frames together.

The first step was to modify the Washington D.C. data. Renaming the columns in the Washington D.C. data frame made the merge much easier, so I changed the columns names to match those of the King County data frame. I then filtered down the data set to only contain housing sales between May 2014 and May 2015. I also want to make sure that missing values are accounted for and that I handled these cases prior to the merge. Thus, I removed the rows where the price of the sold housing unit is missing, since this was my dependent variable I built my regression models around. For any rows where the "Year Remodeled", "Year Built", "Living SQFT", or "# Floors" were missing, I imputed the values with the average for the entire column. For rows where the "Condition" or "Grade" was missing, I imputed the values with the label "Missing."

For the next part, I modified the King County data set. The only modification I made was to convert the "Grade" and "Condition" columns from numerical categorical values to the matching String values found in the Washington D.C. data. For example, the values found in the "Condition" column in the King County data set were numbered 1, 2, 3, 4, and 5. However, the grades in Washington D.C. were "Poor", "Average", "Good", "Very Good", and "Excellent." Because of this, I replaced each of the numerical values for the column with the corresponding grades.

	price	date	bathrooms	bedrooms	sqft_living	sqft_lot	floors	condition	grade	yr_built	yr_renovated	location
1	993500.0	2014-10-08	5.0	3	1148.0	814	2.0	Very Good	Average	1907	2014	DC
2	1280000.0	2014-08-19	2.5	3	1630.0	1000	2.0	Good	Good Quality	1906	2004	DC
4	1440000.0	2015-04-22	3.5	4	1686.0	1424	2.0	Very Good	Above Average	1908	2015	DC
5	1050000.0	2014-12-23	2.0	2	1440.0	1800	2.0	Average	Above Average	1885	1984	DC
8	900000.0	2014-06-05	1.5	2	1728.0	900	3.0	Good	Average	1880	2003	DC

Figure 1: A portion of the finalized data frame with labeled columns

```
price          0.0
date           0.0
bathrooms      0.0
bedrooms       0.0
sqft_living    0.0
sqft_lot       0.0
floors         0.0
condition      0.0
grade          0.0
yr_built       0.0
yr_renovated   0.0
location       0.0
dtype: float64
```

Figure 2: Missing value percentage for each column in the dataset

Finally, the Washington D.C. data frame consisted of 7160 entries and the King County data consisted of 21,613 entries. Overall, there were 12 features that described each house sale. Now, the respective columns all are of the same type and have no missing values.

5. Exploratory Data Analysis and Initial Findings

Because Washington D.C. and King County, Washington are located on opposite sides of the country, it is very interesting to look at how housing features and prices may differ and what could be in high demand depending on if you want to move to the east or west coast.

a) Are houses more likely to sell at specific times of the year?

The first thing I wanted to do was to look at when houses were most likely to be sold. Does the time of the year actually impact when houses sell?

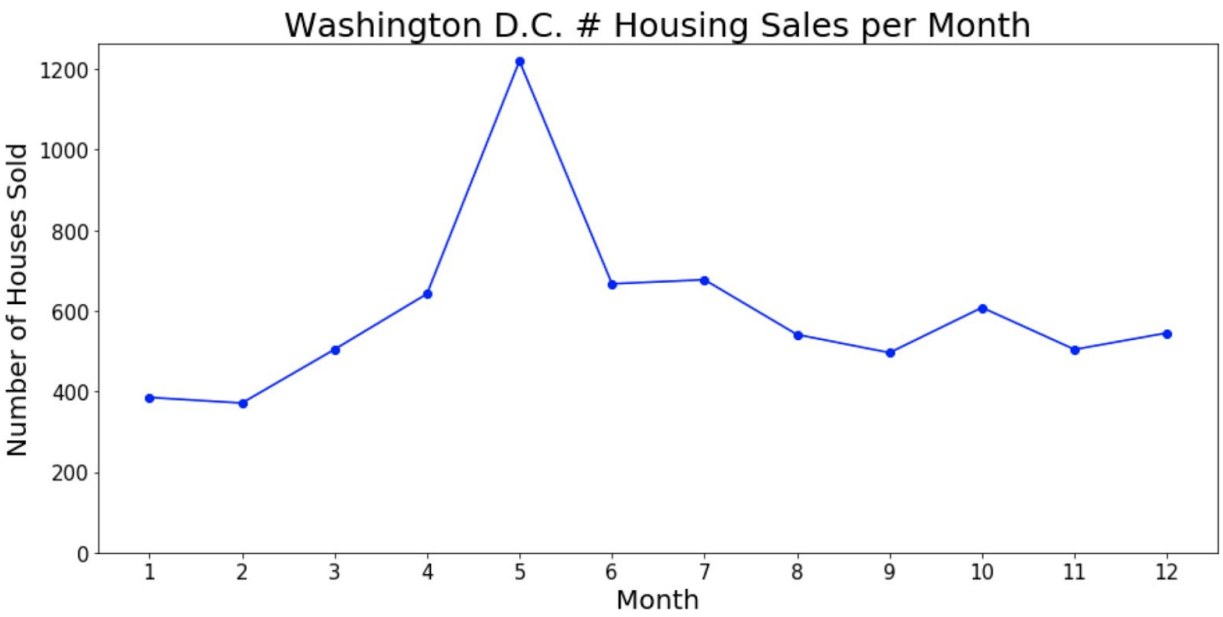


Figure 3: Washington D.C. House Sales by Month

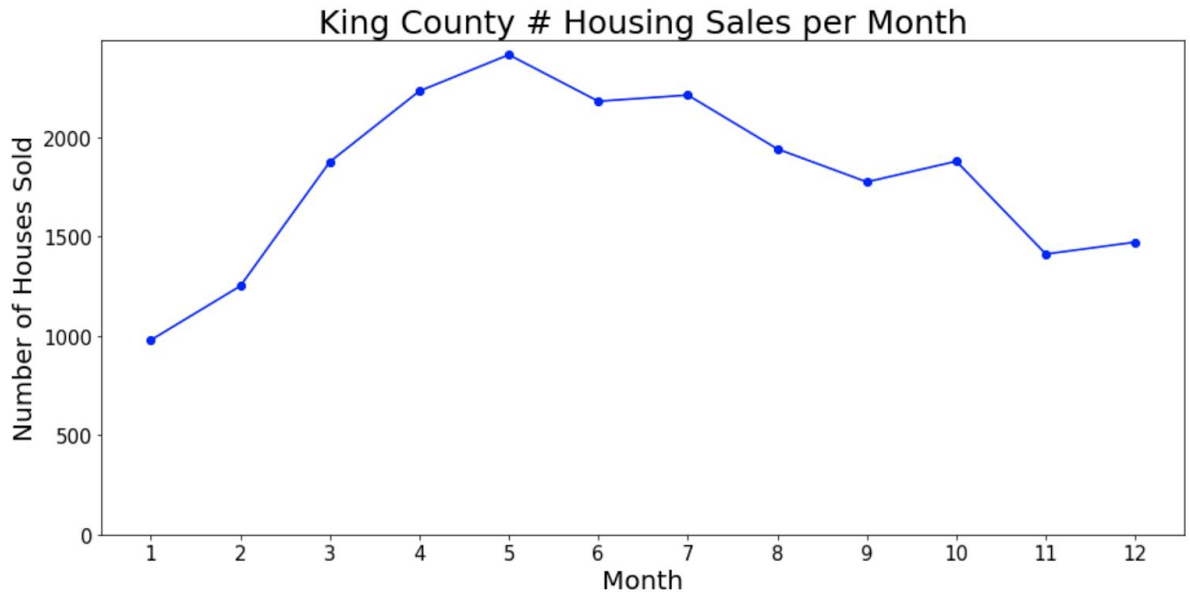


Figure 4: King County House Sales by Month

By aggregating the housing sales by month for each data set, figures 4 and 5 highlight that May is the month when most houses are sold for both regions. It is interesting to note that in King County, the immediate months preceding and proceeding May are much closer to the peak number of houses sold compared to those months in Washington D.C. There is an obvious increase in sales in May, with the surrounding months dropping in sales significantly.

b) Descriptive Statistics for Bedrooms, Bathrooms, Square Feet of Living, and Square Feet of Lot

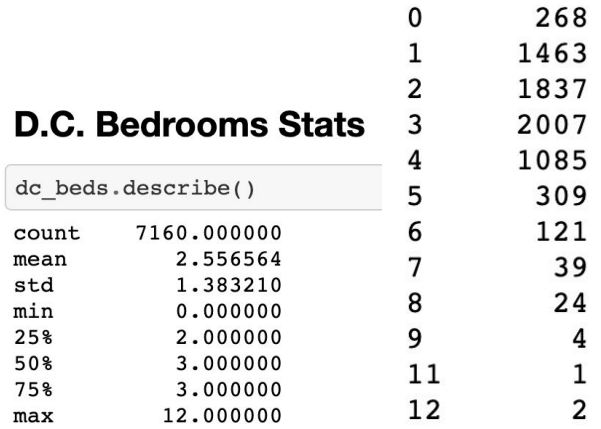


Figure 5: Washington D.C. bedrooms summary statistics and value counts

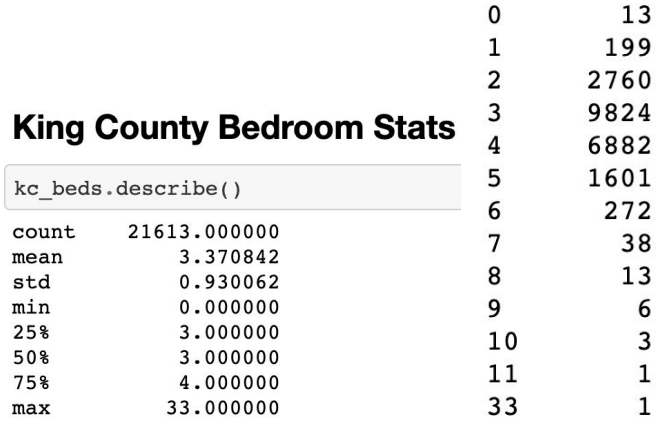


Figure 6: King County bedrooms summary statistics and value counts

Looking at the values shown in figures 5 and 6 helps to distinguish some interesting outliers. It appears that in King County, there is a housing unit that was sold with 33 bedrooms. There are only single cases where certain housing units had a significantly larger number of bedrooms. It is also worth noting that King County has an average of almost 1 more bedroom on average per house sale (which could be due to the outlier of the house with 33 bedrooms).

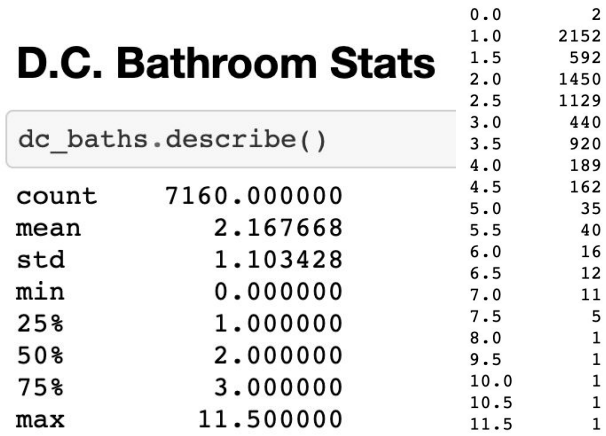


Figure 7: Washington D.C. bathroom summary statistics and value counts

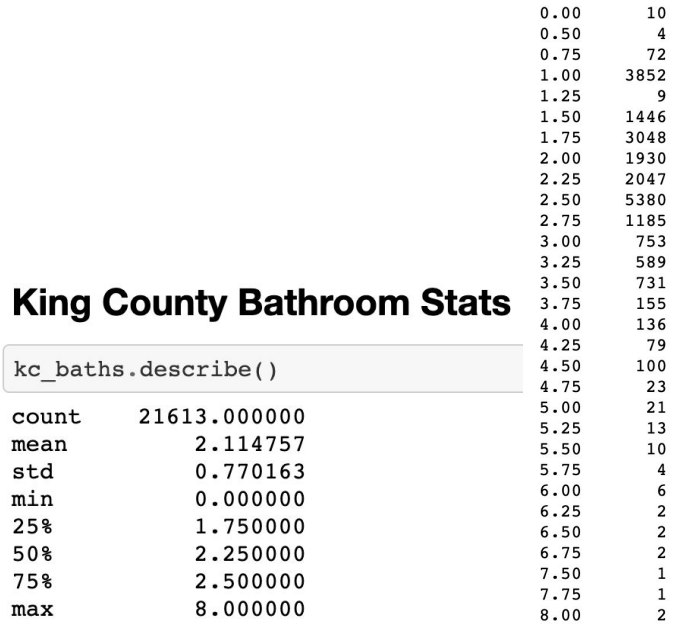


Figure 6: King County. bedrooms summary statistics and value counts

count	7160.000000
mean	1715.277034
std	601.743189
min	576.000000
25%	1457.750000
50%	1715.277034
75%	1715.277034
max	9817.000000

Figure 7: King County living square feet statistics

In the D.C. data set, the mean sqft_living is around 1715 per house sold. This number is 300 feet smaller than the average size home sold in King County, with King County's average living square feet having a larger maximum and smaller minimum.

count	7160.000000
mean	1875.414106
std	2525.770408
min	0.000000
25%	410.000000
50%	1209.000000
75%	2276.500000
max	67805.000000

Figure 8: Washington D.C. lot square feet summary statistics

count	21613.000000
mean	15106.967566
std	41420.511515
min	520.000000
25%	5040.000000
50%	7618.000000
75%	10688.000000
max	1651359.000000

Figure 9: King County lot square feet summary statistics

It appears that the average lot size for housing units in King County was much lower than those in Washington D.C. There was a much larger range of property lot sizes found in King Country, ranging from 520 square feet up to 1651359 square feet. These could be large outliers to the rest of the data set, and it is important to look into them when building my model. Washington D.C.

is a total of 61.05 square miles, whereas King County is 2,307 square miles. This leaves a lot more room for larger plots of land (such as farmland) to be sold.

c) What are the grade and condition distributions of houses sold?

For context, condition refers to the overall quality or build of the house. The better the condition, the more well-maintained the housing unit is. Low condition scores means that the housing unit is approaching a condition that might require reconstruction. Grade, on the other hand, refers to the evaluation of the construction materials and level of craftsmanship used to build the house. The higher the score, the higher the quality. For example, a score of 13 (the highest score) is mansion-level.

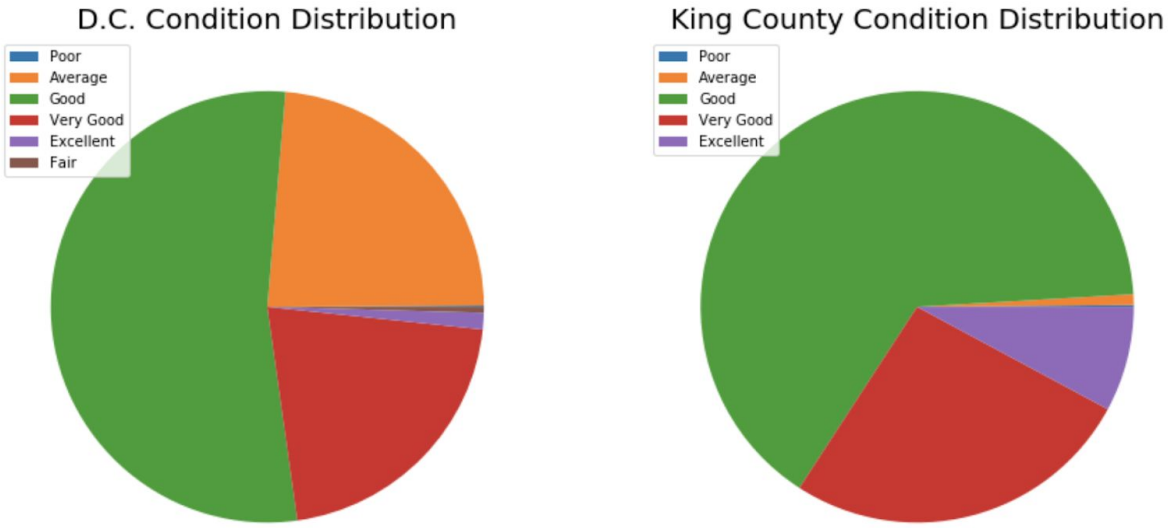


Figure 10: Condition categorical distribution for Washington D.C. and King County

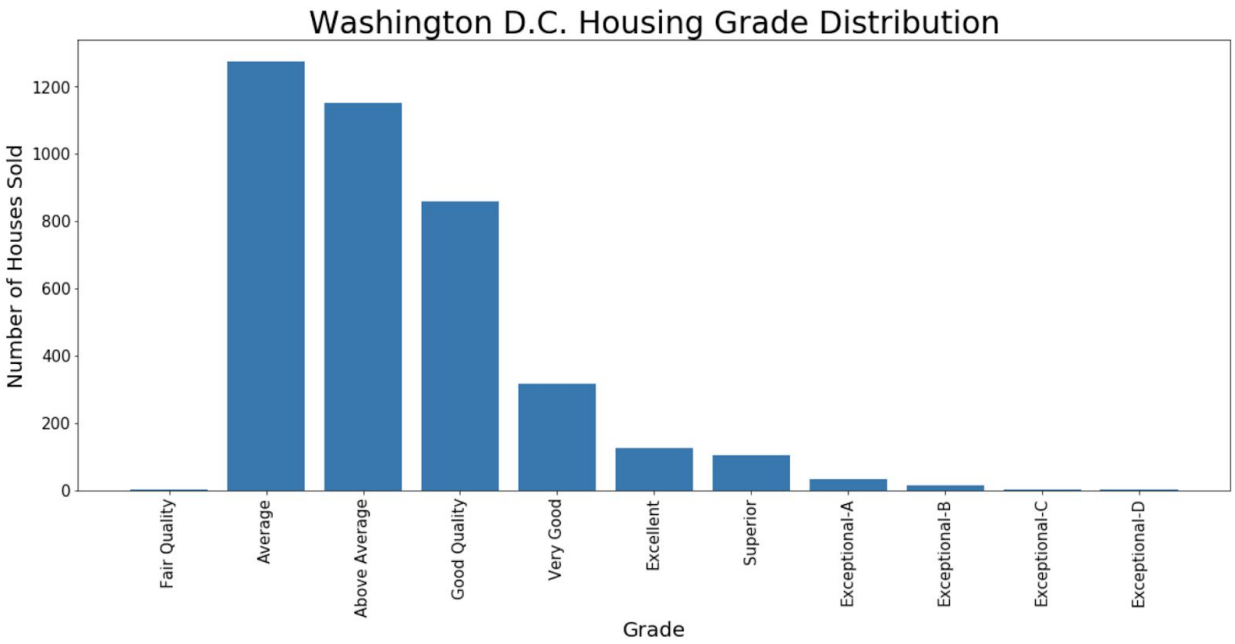


Figure 11: Washington D.C. housing grade categorical distribution

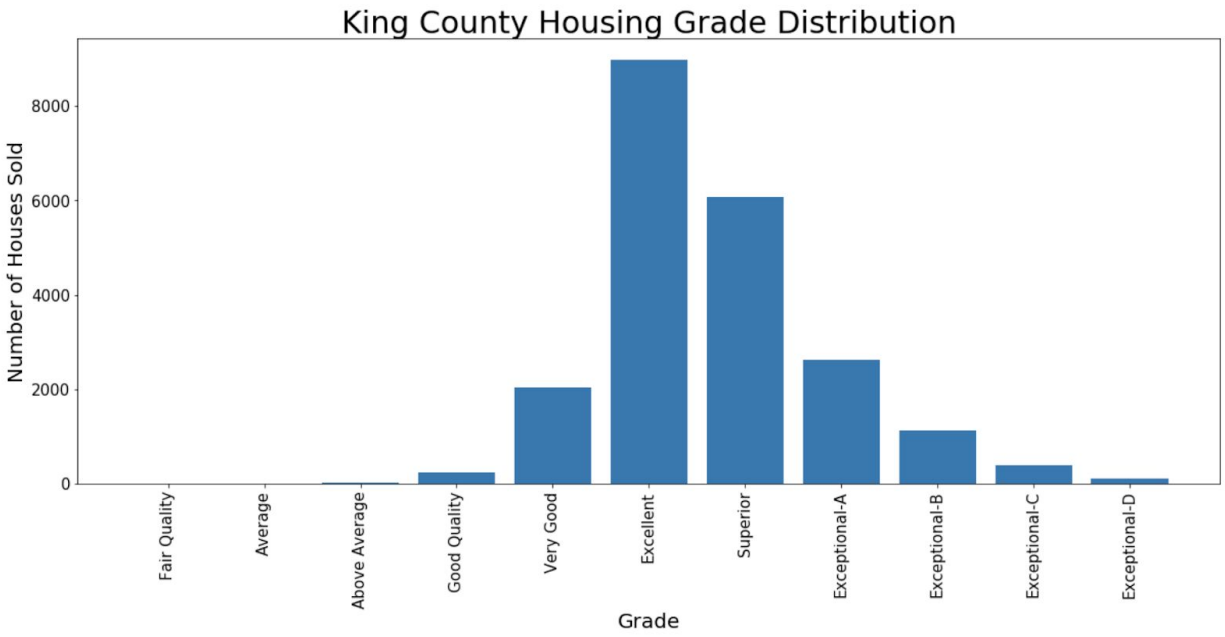


Figure 12: King County housing grade categorical distribution

The two distributions really highlight some important housing quality features between Washington D.C. and King County. Firstly, King County has much less 'Average' condition

houses sold and much more 'Good' condition houses sold compared to D.C. As a result, on average each house sold in King County was in better shape compared to those in D.C. Additionally, there is a clear visual distinction between the Washington D.C. Grade distribution and King County Grade distribution. Houses appear to be much nicer and high-end compared to those in Washington D.C. with the majority of houses sold being graded either "Excellent" or "Superior" compared to the majority of Washington D.C. houses being either "Average" or "Above Average".

d) Does condition affect the selling price?

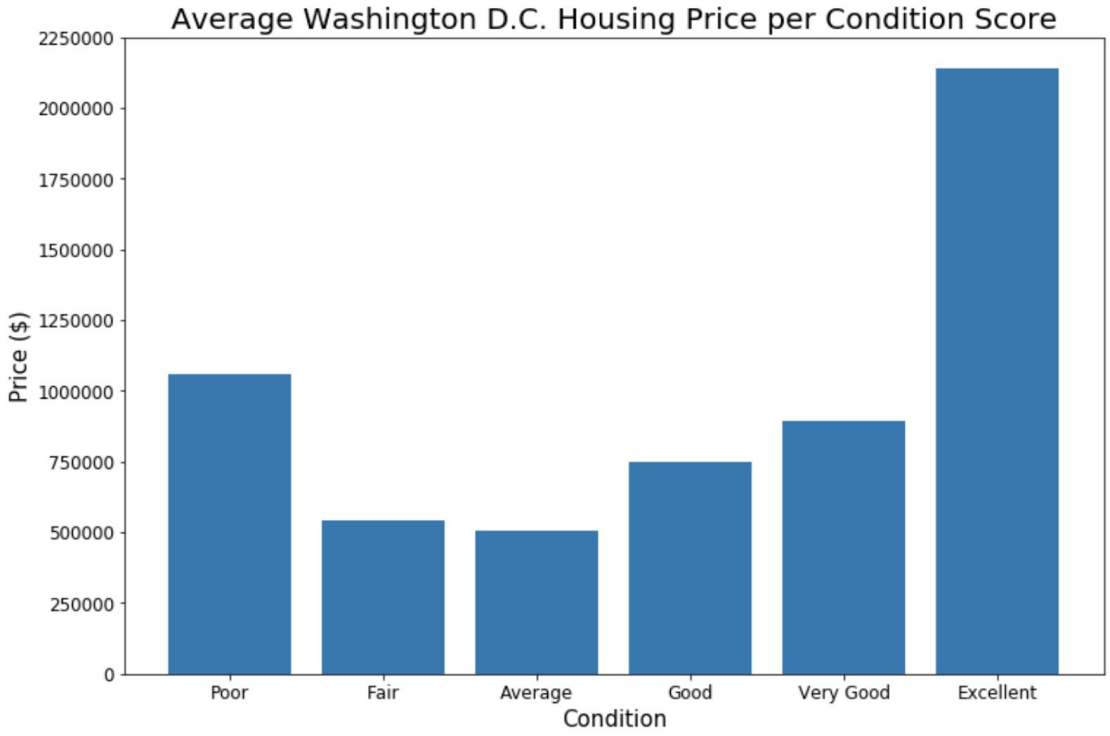


Figure 13: Washington D.C. Housing Price per Condition Score

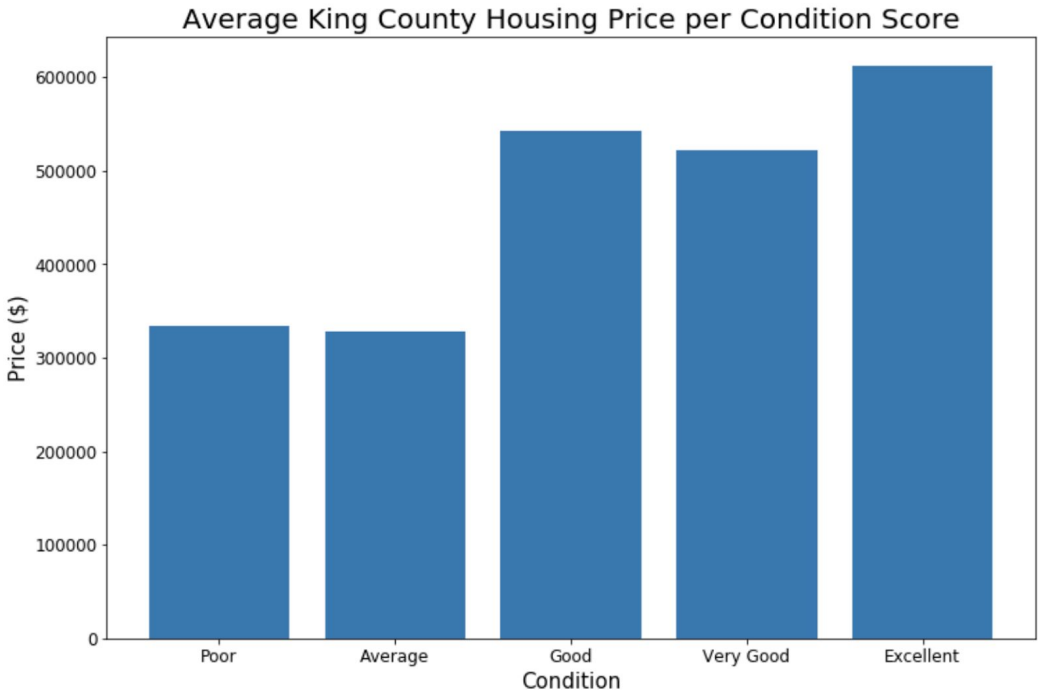


Figure 14: King County Housing Price per Condition Score

One of the most surprising things from the above graphs is the average price for 'Poor' condition houses in Washington D.C. The average price for 'Poor' condition houses in Washington D.C. was actually higher than any other condition other than the highest "Excellent" one. Unlike King County, which follows an expected pattern of higher prices for better housing conditions, the Washington D.C. data is unexpected here.

e) Does grade affect the selling price?

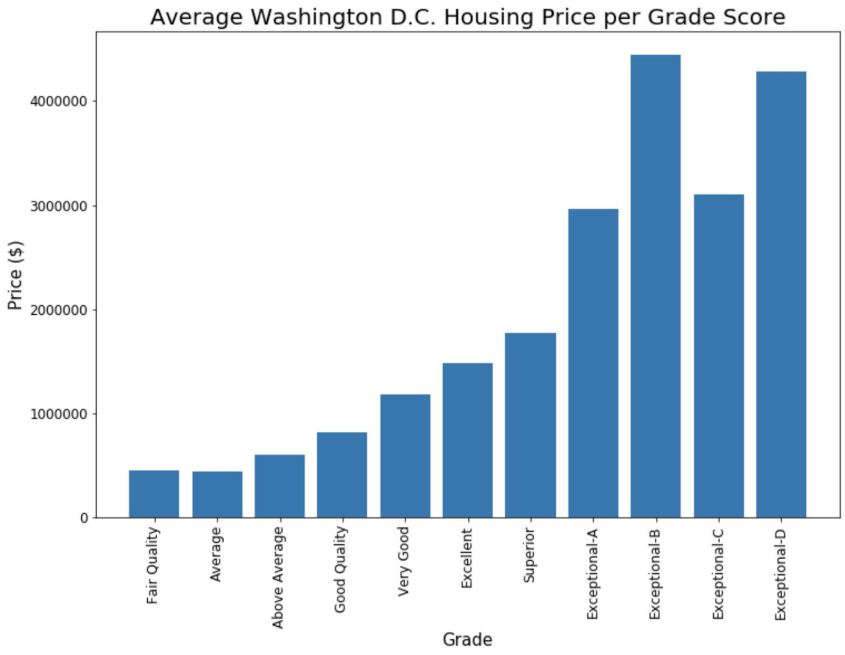


Figure 15: Washington D.C. Housing Price per Grade Score

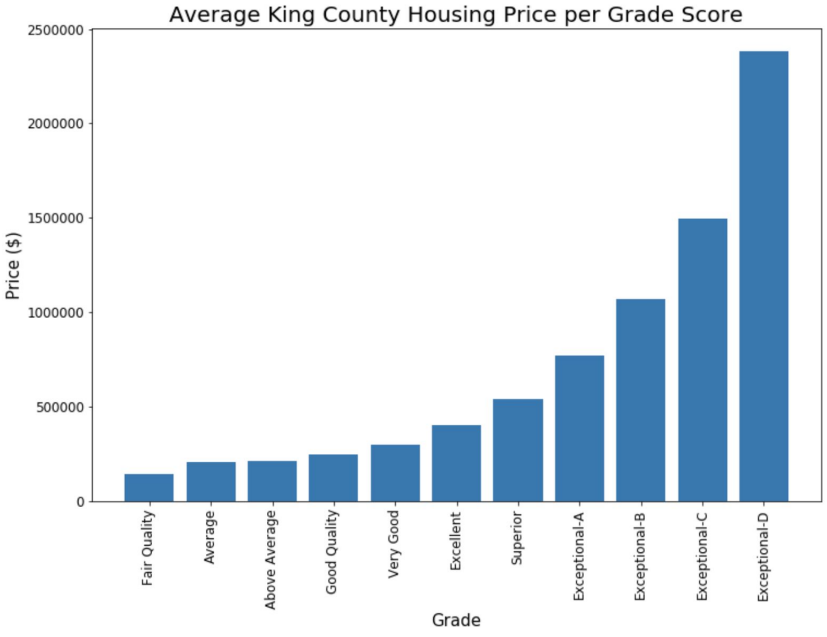


Figure 16: King County Housing Price per Grade Score

For both charts, it appears that the better grade a house received, the higher the average selling price for the house would be. There is one interesting thing about the Washington D.C. selling price for "Exceptional-C" grade. This grade is the second-highest grade a house can receive, yet there is a dip in the average selling price compared to grades lower than it.

f) Correlation Heatmap for Housing Features

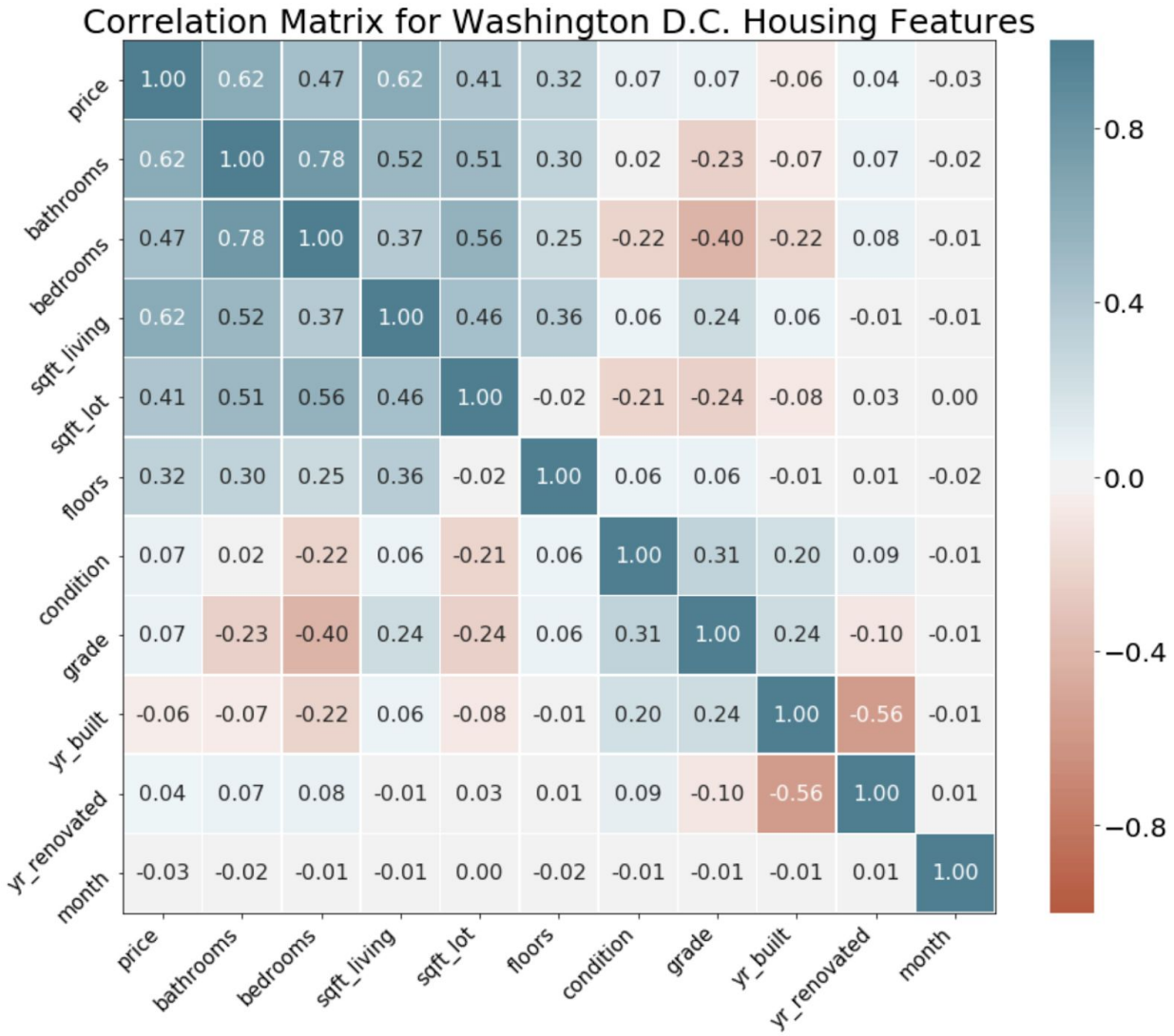


Figure 17: Washington D.C. housing feature correlation matrix

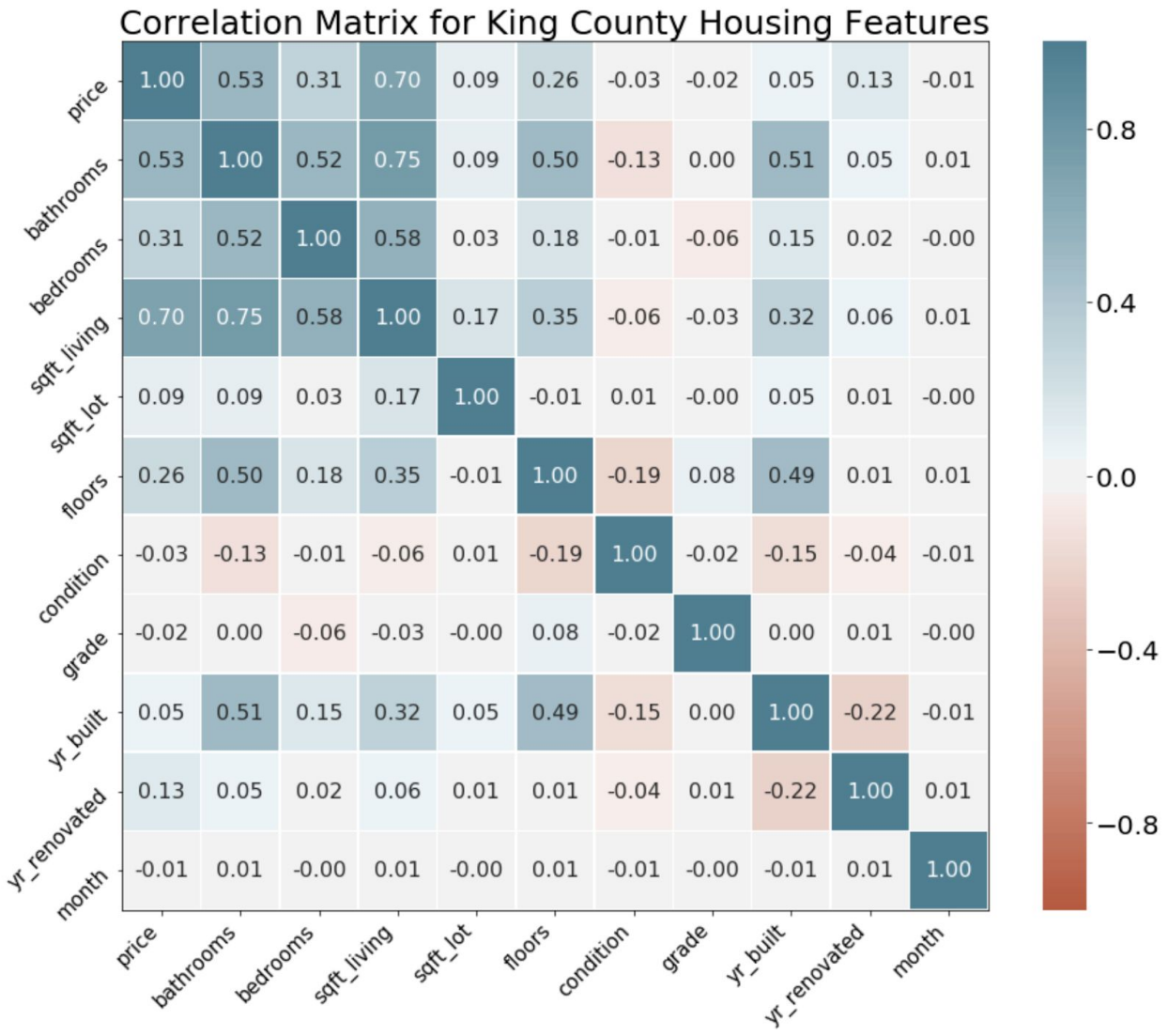


Figure 18: King County housing feature correlation matrix

Creating a correlation matrix between different housing features will help visualize which features have a high impact on the target and other features. One interesting column to look into is the 'price' column to see which other features have a high correlation with the dependent variable I will be examining. One distinct feature I would like to point out is 'sqft_lot'. In the Washington D.C. matrix, there is a value of 0.41, whereas in the King County matrix this value is 0.09. It is interesting that there is a large difference between the effect the lot square footage has on housing sales in these different locations. It could be that Washington D.C. doesn't have

as many houses with a large plot of land, so land can be more valuable when pricing a house compared to King County where there is a wide range of lot sizes.

6. Applications of Inferential Statistics

a) Average Housing Price: Is there a statistical significant difference between the average housing sale price between Washington D.C. and King County?

The code that I used for parts of this test can be seen in my notebook [3]. I compared the difference of means between the housing sale prices in Washington D.C and King County by testing the following hypotheses:

- H_0 : The true mean housing sale price between the Washington D.C. and King County are the same
- H_1 : The true mean housing sale price between the Washington D.C. and King County are not the same

For this test, I assumed an alpha-level of 0.05.

count	7160.000000
mean	627126.845391
std	510667.841622
min	5185.000000
25%	345000.000000
50%	517000.000000
75%	749600.000000
max	7395000.000000

Figure 19: Washington D.C. sale price summary statistics

[3] http://onlinestatbook.com/mobile/tests_of_means/difference_means.html

```
count      21613.000000
mean      540088.141767
std       367127.196483
min        75000.000000
25%       321950.000000
50%       450000.000000
75%       645000.000000
max      7700000.000000
```

Figure 20: King County sale price statistics

i) Compute the test statistic: Mean of the sampling distribution of the difference between means

```
test_stat = dc_df.price.mean() - kc_df.price.mean()
print("Mean difference of means for Washington D.C. and King County Housing Sales:", test_stat)
```

```
Mean difference of means for Washington D.C. and King County Housing Sales: 87038.70362453209
```

In order to test for the difference of means between the Washington D.C. and King County housing prices data sets, there are 3 assumptions that I am going to make:

1. The Washington D.C. and King County housing data sets have the same variance
2. Each housing price population is normally distributed
3. Each housing price is sampled independently from each other value. This assumption means that each housing unit sold is for one value only

ii) Calculate the Standard Error of the test statistic

The formula for the variance of the sampling distribution of the mean is:

$$\sigma_M^2 = \frac{\sigma^2}{N}$$

where σ is the standard deviation and N is the sample size. Because the Washington D.C. and King County are different populations and have different sample sizes, we need to distinguish between them via subscripts to represent each population:

$$\sigma^2_{M_1-M_2} = \frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}$$

Using the above formulas, I can use the following formula to calculate the standard error of the difference of means between the two populations.:

$$\sigma_{M_1-M_2} = \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}$$

The following is the code running these calculations:

```
# Calculate standard error of test statistic
dc_var = dc_df.price.var()
kc_var = kc_df.price.var()
dc_size = len(dc_df.price)
kc_size = len(kc_df.price)

standard_err = np.sqrt((dc_var / dc_size) + (kc_var / kc_size))
```

Now that I had the standard error, I plugged the value it into the equation below to get the t-statistic and used this to get the probability (p-value) of getting a t as large or larger than the t-statistic or as small or smaller than -(t-statistic). :

$$t = \frac{\text{statistic - hypothesized value}}{\text{estimated standard error of the statistic}}$$

```
# Compute t-statistic
t_stat = test_stat / standard_err

# Degrees of freedom
dof = dc_size + kc_size - 2

# Compute p-value
p_val = 1 - stats.t.cdf(t_stat,df=dof)

print("p-value:", p_val)
```

```
p-value: 0.0
```

Since the p-value was less than 0.05, I could confidently reject my null hypothesis and therefore conclude that the observed difference in the means is statistically significant--informally, this means that the observed difference is likely not to be due to chance.

7. Linear Relationships Between Non-categorical Housing Features and Housing Prices

By examining a linear regression model for each individual non-categorical housing features with the housing sale price, I could determine which features are statistically significant predictors of housing sale prices. I went through each non-categorical feature and plotted the feature against the price to first see any obvious correlations between the two.

Figure 21: Square Feet Linear vs. Price Linear Relationship

The above plot is one example of a linear regression model being built for each data set. By isolating each individual non-categorical feature with the price, the models could reveal some patterns and effects that the individual features had on the dependent variable I was examining. From above, it appears that there is a clear positive correlation between the living square feet and price. For each feature, I pulled the p-value of the resulting linear regression model to help determine if the non-categorical feature was a significant predictor for price:

P-values for each non-categorical variable

```
pvals_df
```

	bathrooms	bedrooms	sqft_living	sqft_lot	floors	yr_built	yr_renovated
D.C.	0.0	0.0	0.0	7.746301e-283	2.768112e-166	1.935821e-07	3.570190e-04
KC	0.0	0.0	0.0	7.972505e-40	0.000000e+00	1.929873e-15	1.021348e-77

Figure 21: P-values for each non-categorical feature in the Washington D.C. and King County data sets

Because the p-value for each non-categorical variable in both the Washington D.C. data set and King County data set is less than the alpha value of 0.05, each variable is deemed to be a statistically significant predictor for price.

8. Linear Regression Models for Categorical Features on Price

Now that I had looked at each non-categorical feature, now it was time to look at the categorical features: grade and condition. For each of these, I used *statsmodels "ols"* function to build the linear regression model. This function was very helpful because it automatically transforms each categorical variable to a dummy variable to be usable with predicting price (a continuous variable). Below is an example of the model built with the *grade* feature for the King County data set:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.007
Model:                            OLS   Adj. R-squared:                  0.007
Method:                 Least Squares   F-statistic:                     37.41
Date:                Wed, 04 Mar 2020   Prob (F-statistic):           3.12e-31
Time:                        22:07:45   Log-Likelihood:            -3.0753e+05
No. Observations:               21613   AIC:                         6.151e+05
Df Residuals:                   21608   BIC:                         6.151e+05
Df Model:                           4
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                     3.273e+05   2.79e+04     11.731      0.000    2.73e+05    3.82e+05
condition[T.Excellent]        2.851e+05   2.93e+04      9.739      0.000    2.28e+05    3.43e+05
condition[T.Good]             2.147e+05   2.81e+04      7.650      0.000     1.6e+05     2.7e+05
condition[T.Poor]             7144.5213   7.24e+04      0.099      0.921   -1.35e+05    1.49e+05
condition[T.Very Good]        1.939e+05   2.83e+04      6.848      0.000    1.38e+05    2.49e+05
==============================================================================
Omnibus:                    19204.851   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1164858.777
Skew:                           4.044   Prob(JB):                         0.00
Kurtosis:                      38.044   Cond. No.                         38.8
==============================================================================
```

Figure 22: OLS linear regression model using condition as the predictor

The above model works by using one of the *condition* categories as a baseline. From there, each other category is compared to this baseline model. For example, the condition category "average" is used as the baseline. "Poor" condition above has a p-value of .921. This p-value means that we cannot say that this category is a statistically significant predictor for "average" houses sold, whereas every other category has a p-value that is below the threshold I am using. The coefficients for each category also help distinguish how additional *conditions* affect the baseline model. I used this for both the Washington D.C. and King County data set on both the *grade* and *condition* features to see how each affects the price individually.

9. Baseline Model Analysis

First, I defined the data frames with a couple of modifications made. Rather than having sale dates by day, I am creating another column labeled 'month.' The 'condition' and 'grade' columns need to be represented with categorical codes to be used with the rest of the data, which all are

continuous variables. Additionally, I scaled the data to normalize each feature to have a consistent range of values with the rest of the columns.

a) Linear Regression Models: No Modifications

Before selecting the best features to build my models with, I was curious to see how well a linear regression model will perform without any modifications made to the scaled data. By including all features in the initial baseline model, I could see how well a generic model could represent the data for both Washington D.C. and King County. I wanted to explore how each feature could potentially affect (both positively and negatively) the overall fit of the models. The following was the resulting coefficients for each linear regression model:

i) Model Coefficients

Washington D.C. Model Coefficients ¶

```
# Washington D.C. linear regression model coefficients
lm_dc.params
```

```
Intercept       0.006840
bathrooms       0.415863
bedrooms       -0.014473
sqft_living     0.348688
sqft_lot        0.072633
floors          0.065963
condition       0.044293
grade           0.091308
yr_built       -0.094210
yr_renovated   -0.034802
month          -0.007449
dtype: float64
```

Figure 23: Washington D.C. linear regression model coefficients with no features excluded

King County Model Coefficients

```
# King County linear regression model coefficients
lm_kc.params
```

```
Intercept      -0.002498
bathrooms       0.139652
bedrooms       -0.170795
sqft_living     0.751963
sqft_lot       -0.036162
floors          0.079607
condition      -0.000993
grade          -0.014047
yr_built       -0.258198
yr_renovated    0.011617
month          -0.022038
dtype: float64
```

Figure 24: King County linear regression model coefficients with no features excluded

Looking at the initial coefficient values, there are some distinctions I would like to point out. Living square feet is by far the largest positive coefficient for both models, which makes sense due to the fact that more square footage tends to positively increase the price of a house. However, the magnitude of the coefficient for the King County model is significantly higher (0.75 compared to 0.34 in the Washington D.C. model). The coefficient values can range from -1 to 1, so 0.75 is very high. Another interesting difference is the "yr_built" coefficient. Although I expected newer buildings to increase the price, in both models it appears the more recent a house was built, the lower the price would be. Lastly, another interesting distinction is the "grade" coefficient. In Washington D.C., higher grades would positively impact the price of a house. However, in King County, this was not the case. One potential explanation for this could be due to King County's wider range of houses being sold (rural vs. urban). King County contains cities where a lot of the land is farmland where houses may have a lower grade compared to their city counterparts, and this may impact how the price is affected.

ii) Goodness of fit on Training Data

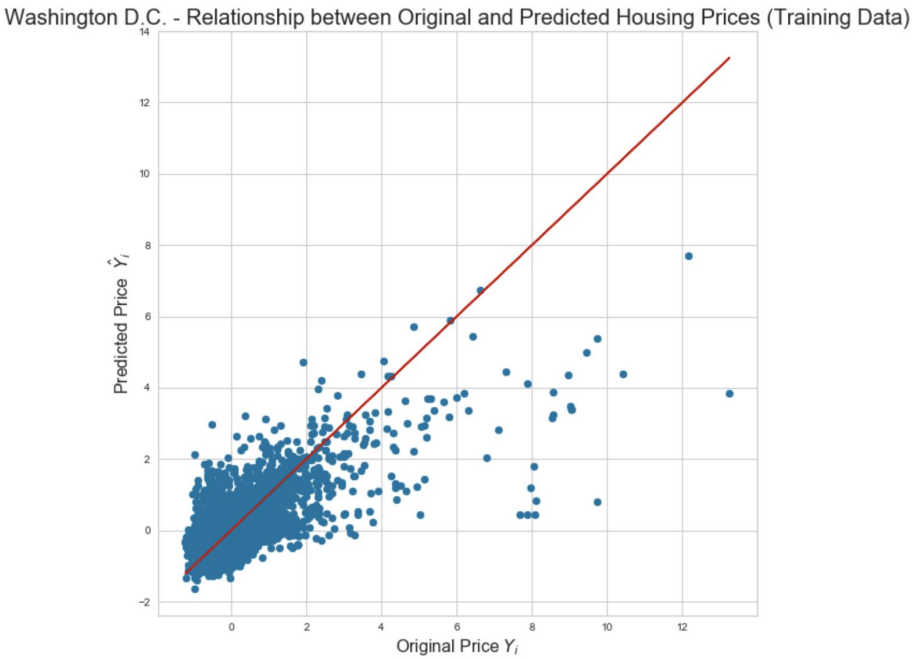


Figure 25: Washington D.C. Original vs. Predicted Housing Price on Training Data

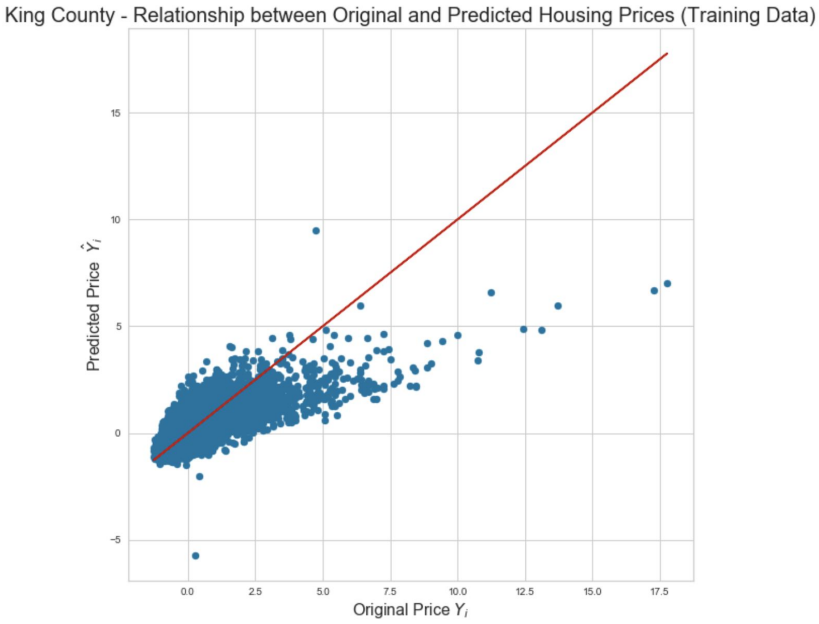


Figure 26: King County Original vs. Predicted Housing Price on Training Data

The first visualization I created for the linear regression model is illustrated in Figure 25 and 26 above. The line $y = x$ drawn through the graph represents perfect predictions, so points that are

closer to the line represent better predictions made by the model. As we can see, there are a few points that are plotted for higher housing sales. The model appears to have had more difficulty making accurate predictions for these points.

```
print("Washington D.C. Training Adjusted R-Squared:", lm_dc.rsquared_adj)
```

```
Washington D.C. Training Adjusted R-Squared: 0.5130895640248121
```

```
print("King County Training Adjusted R-Squared:", lm_kc.rsquared_adj)
```

```
King County Training Adjusted R-Squared: 0.5560741516357698
```

Figure 27: Adjusted R-Squared values for Washington D.C. and King County Models

The coefficient of determination, R^2, signifies the percentage of the variance in the housing sale prices that can be explained by each linear regression model. Higher R^2 values means that a higher percentage of the variance is explained by the model. By comparing the R^2 values of each model on the training and test data, I can see how the explained variance changes for predictions by the model. The reason I am using the adjusted R^2 value is because this value can increase *and* decrease based on additional features being used in the linear regression model. If necessary features are added to the model, the adjusted R^2 will decrease, thus implying that the additional features are unnecessary. The R^2 value alone can only increase, which means that adding more and more features will inevitably increase the R^2 value. The King County model had a slightly higher adjusted R^2 value, meaning that it accounted for a higher variance percentage than the Washington D.C. model.

```
Washington D.C. Training Mean Absolute Error: 0.4326660083645144
King County Training Mean Absolute Error: 0.43185433672425727
Washington D.C. Test Mean Absolute Error: 0.4002845605505214
King County Test Mean Absolute Error: 0.4414018780483613
```

Figure 28: Mean Absolute Error for Washington D.C. and King County Models

Another metric that I am using to evaluate my model is the mean absolute error. The mean absolute error measures the average magnitude of the errors in a set of predictions, without considering their direction. It helps me determine the average difference between my predicted sale prices and the observed sale prices. I first looked at the MAE for my training data, and then compared this value to my test data to determine if there is a significant flaw in my model. On just initial observation, it appears that there isn't a large difference between the two. However, one interesting note was that the MAE went down from the Washington D.C. training data to the test data. This means that the D.C. model was more accurate for data that it had not seen.

iii) Goodness of fit on Test Data

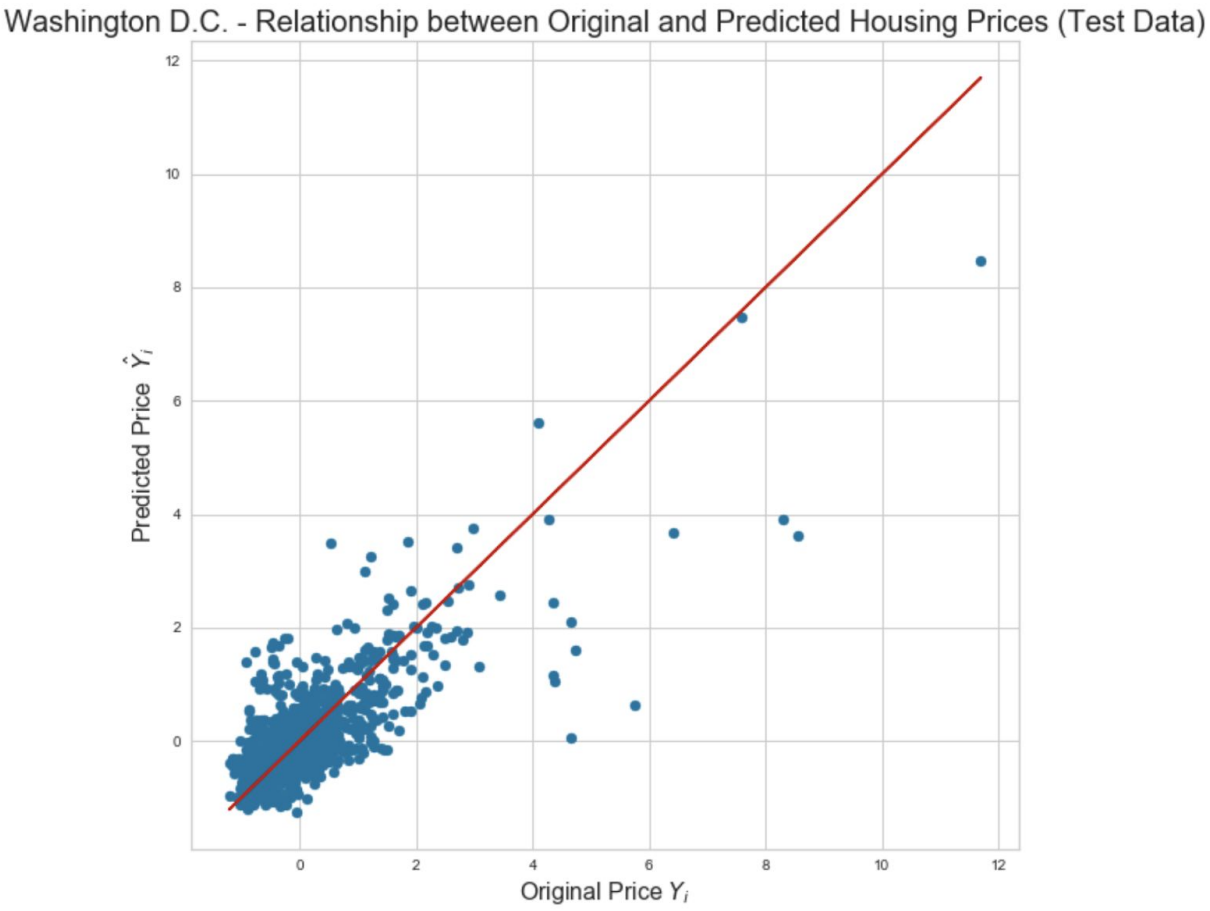


Figure 29: Washington D.C. Original vs. Predicted Housing Price on Test Data

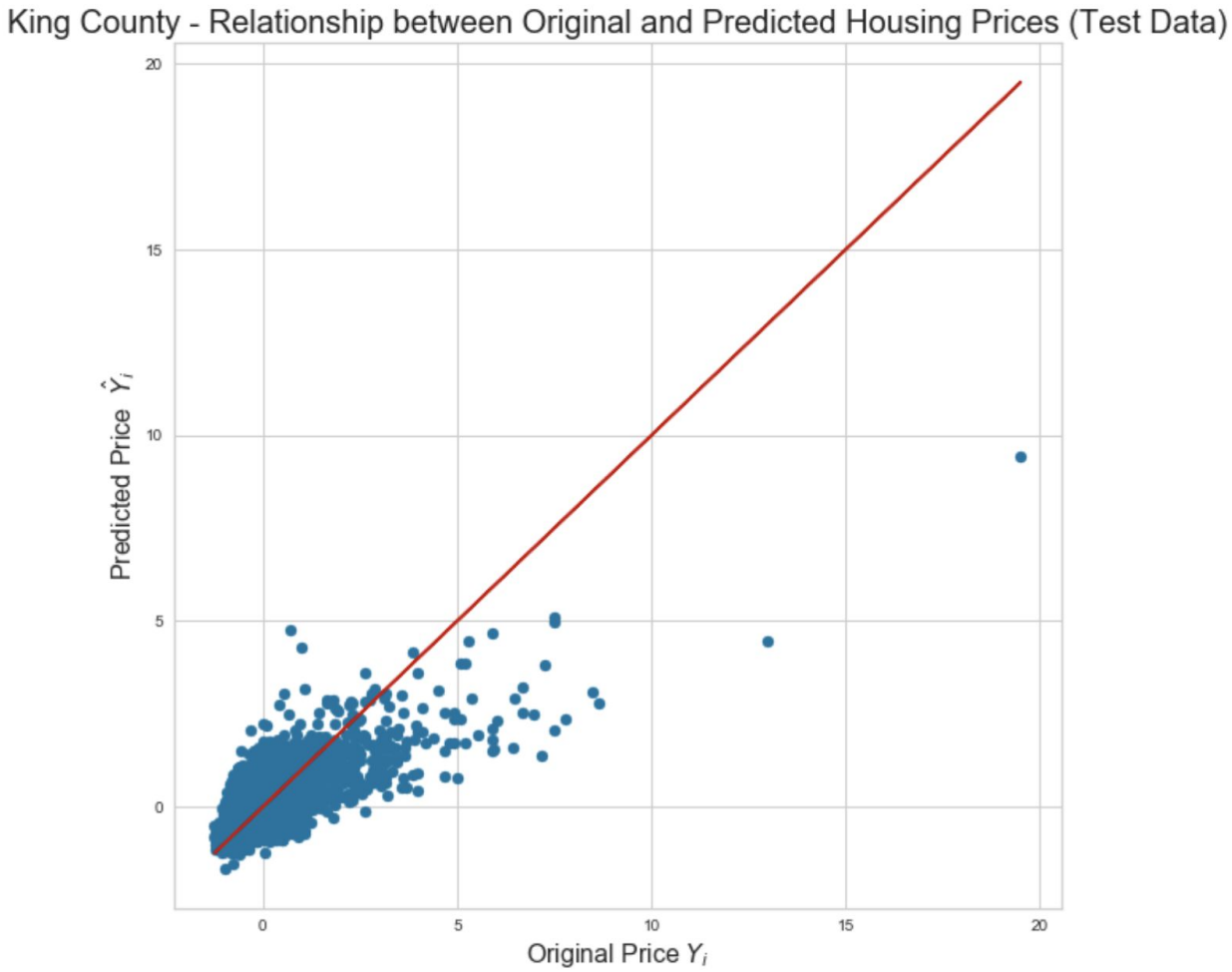


Figure 30: King County Original vs. Predicted Housing Price on Test Data

There are a few assumptions that Linear Regression models make:

- The standard deviation of *y* = 'price' should be constant for different values of *X*
- Normal distribution of errors (test for skew of model)
- Independence between errors (observations are obtained independently)

iv) Test for Constant Standard Deviation: Fitted vs. Residual Plot

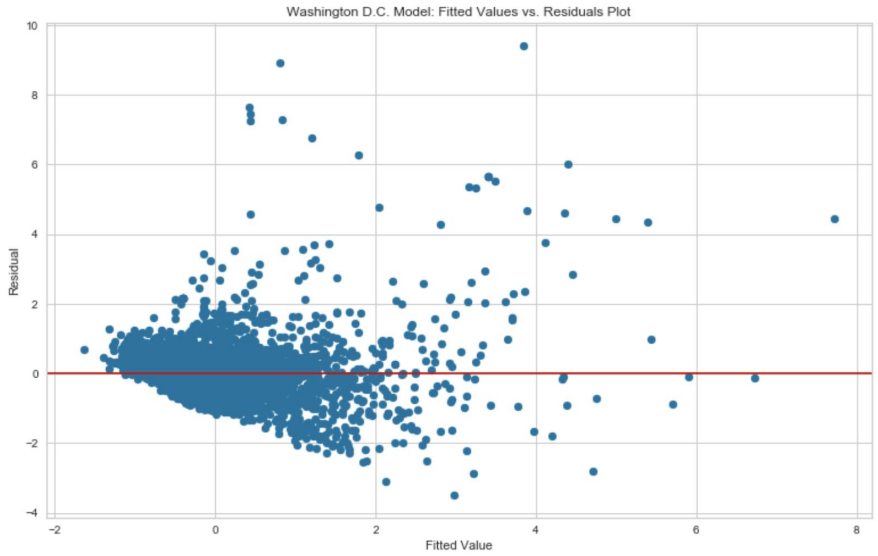


Figure 31: Washington D.C. Fitted Values vs. Residual Plot

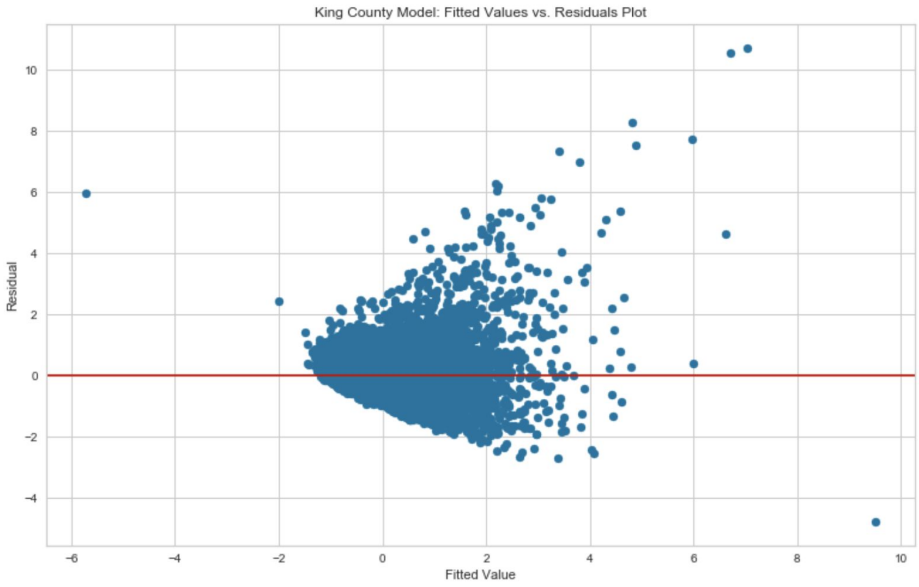


Figure 32: King County Fitted Values vs. Residual Plot

Because there appears to be a 'fanning' effect in the plot, this implies that the standard deviation is not constant for different values of *X*. For both models, it seems that my models will need some tuning for the features that I choose.

v) Test for Normal Distribution of Errors: Quantile Plots

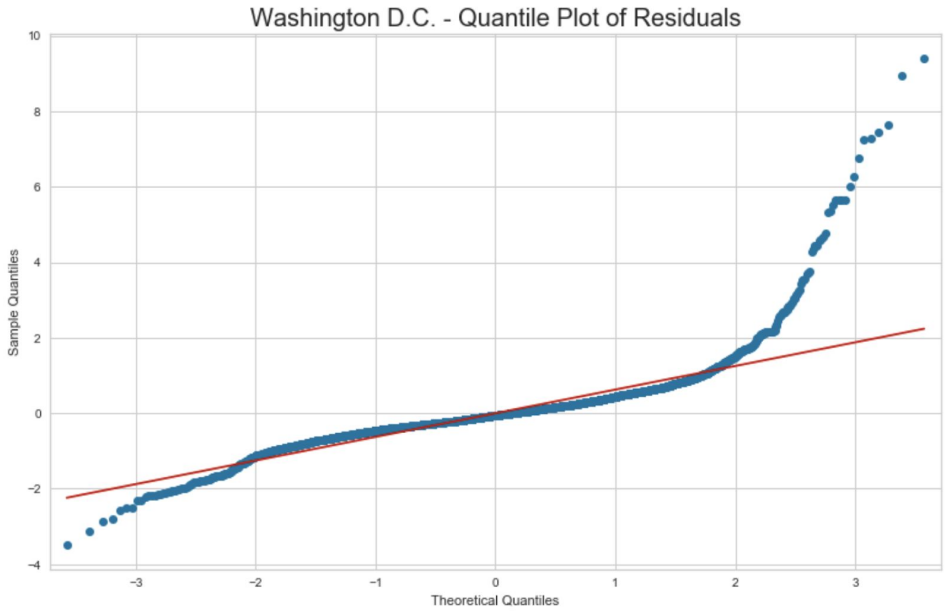


Figure 32: Washington D.C. Quantile Plot of Residuals

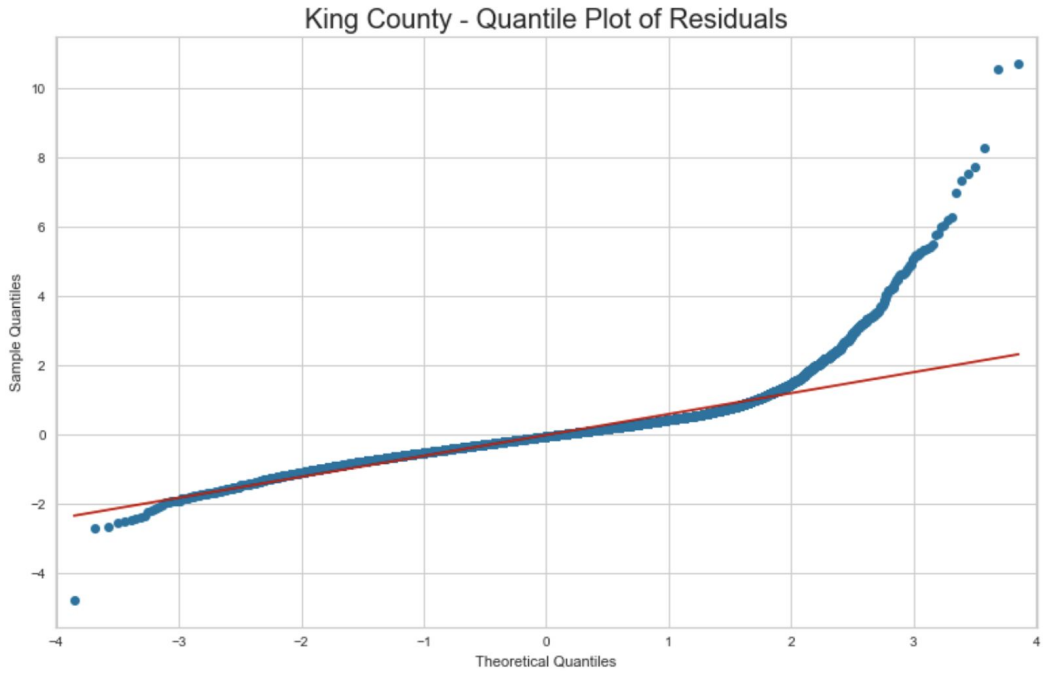


Figure 33: King County Quantile Plot of Residuals

The red line on the plot above signifies a normal distribution of errors. If the errors of each model were normally distributed, they should follow the red line. However, it appears that for both Washington D.C.'s model and King County's model there are clusters of values that deviate from the line, meaning that my model can improve. The points that deviate from the line could be considered outliers and may affect the overall accuracy of each model.

b) Feature Selection: Lasso Regression

One benefit of using Lasso Regression is to help induce sparsity into the model. Although my previous linear regression models from above do not drastically overfit the data (the adjusted R-squared values and Mean Absolute Values are not extreme between the test and data set to imply overfitting of the model), I would still like to introduce some form of feature selection to the models' predictor variables. Lasso Regression zeros out variables that have a high penalty for the model, thus eliminating poor predictor variables.

```
Washington D.C. training score: 0.5354283146472496
Washington D.C. test score: 0.4735110417537456
Washington D.C. number of features used: 8

King County training score: 0.5580081980330058
King County test score: 0.5460385081058753
King County number of features used: 8
```

Figure 34: Lasso Regression model scores and coefficients used

These scores are based on the *R-squared* value. Compared to my previous models built with no modifications, these scores are similar, with the Washington D.C. lasso regression model score being slightly lower.

	features	estimatedCoefficients
0	bathrooms	0.400750
1	bedrooms	0.000000
2	sqft_living	0.351308
3	sqft_lot	0.059128
4	floors	0.049667
5	condition	0.034085
6	grade	0.086690
7	yr_built	-0.073350
8	yr_renovated	-0.007563
9	month	-0.000000

Figure 35: Washington D.C. Lasso Regression Model Coefficients

	features	estimatedCoefficients
0	bathrooms	0.114235
1	bedrooms	-0.141313
2	sqft_living	0.749540
3	sqft_lot	-0.021016
4	floors	0.070464
5	condition	0.000000
6	grade	-0.000000
7	yr_built	-0.238361
8	yr_renovated	0.018938
9	month	-0.013999

Figure 36: King County Lasso Regression Model Coefficients

Running the Lasso Regression on the training data helped eliminate 2 features for each model that had a high prediction penalty. In the Washington D.C. lasso regression model, *bedrooms* and *month* were zeroed out. However, in the King County model *grade* and *condition* were zeroed out. This reveals the idea that certain housing features have more of a significant impact depending on the location that a house is being sold.

i) Evaluation of the Model: Mean Absolute Error

```
Washington D.C. Test Mean Absolute Error: 0.4259443903870488
King County Test Mean Absolute Error: 0.4244899316228654
```

Figure 37: Mean absolute error for Lasso Regression models

ii) Goodness of fit on Test Data

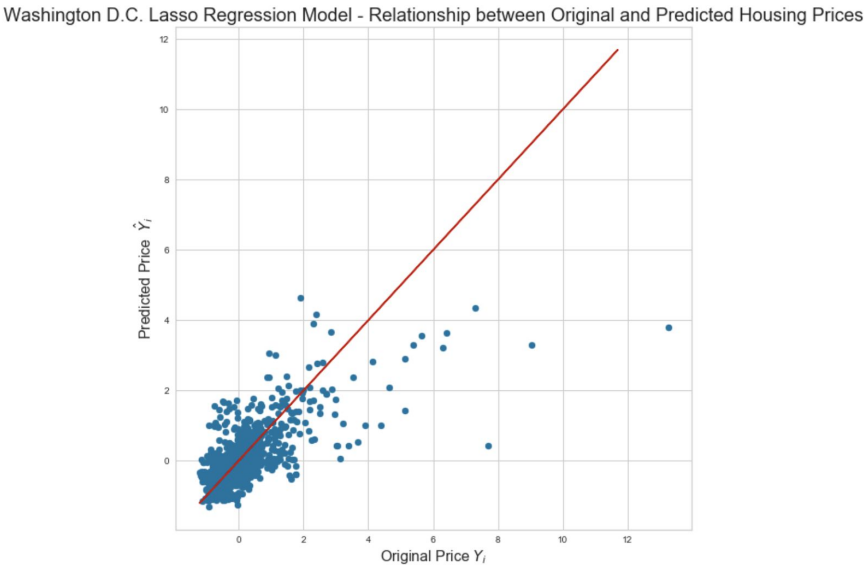


Figure 38: Washington D.C. Lasso Regression - Original vs. Predicted Model Housing Price

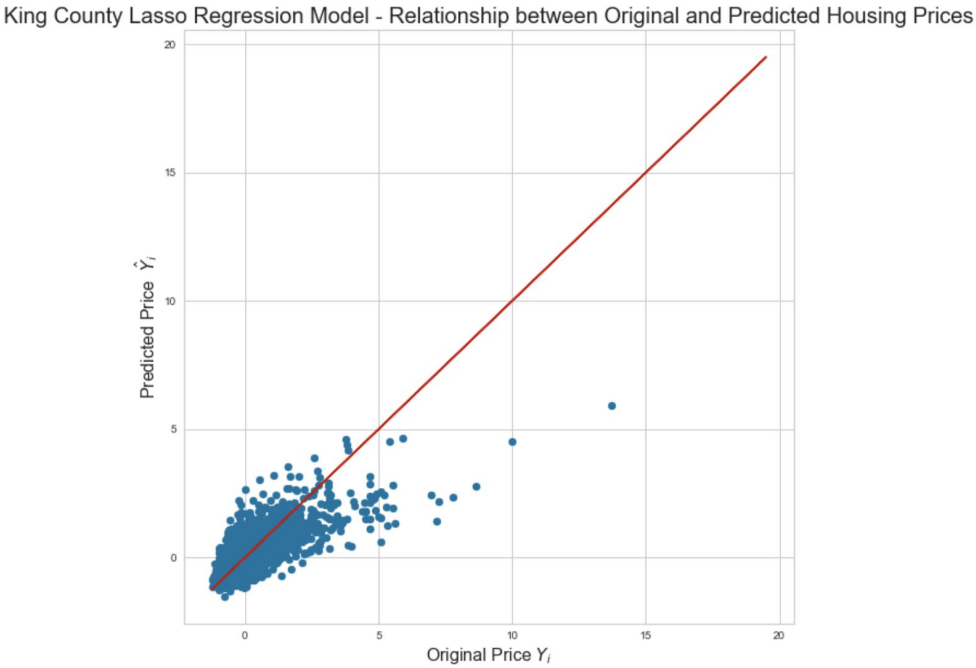


Figure 39: King County Lasso Regression - Original vs. Predicted Model Housing Price

iii) Test for Constant Standard Deviation: Predictions vs. Residual Plot

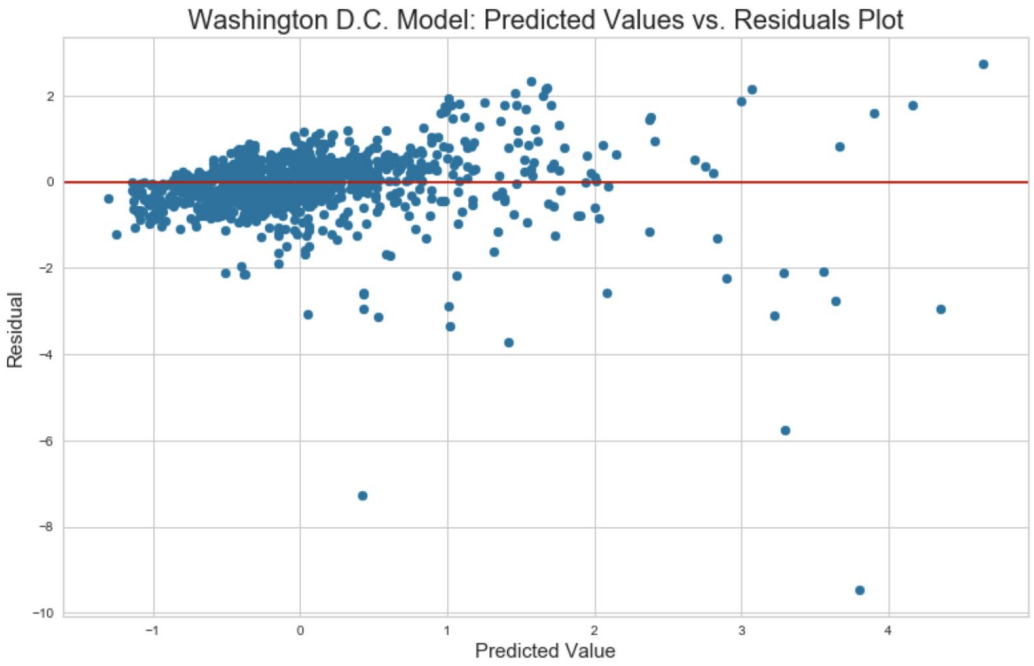


Figure 40: Washington D.C. Lasso Regression Model - Fitted Values vs. Residual Plot

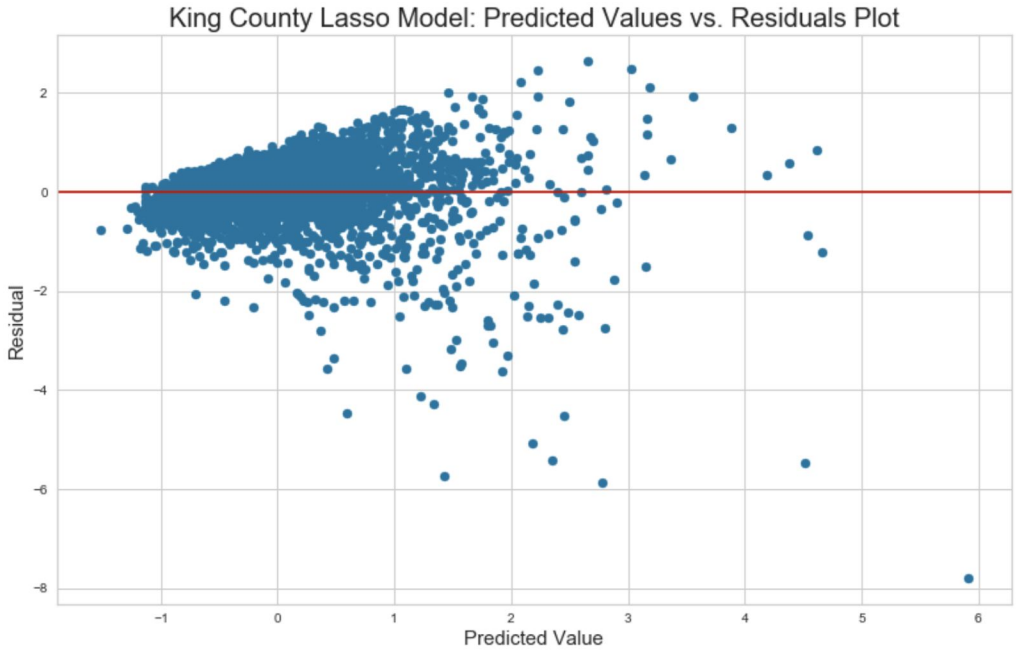


Figure 41: King County Lasso Regression Model - Fitted Values vs. Residual Plot

In Figure 40 and Figure 41, this is where it becomes much more clear that the Lasso Regression Model has been more powerful for predicting housing prices. There is not as much of a 'fanning' effect for the predicted values' residuals, meaning that they are closer to the true price values of the original data. They are more random about the Residual = 0 line, which indicates that the standard deviation is constant.

iv) Test for Normal Distribution of Errors: Quantile Plots

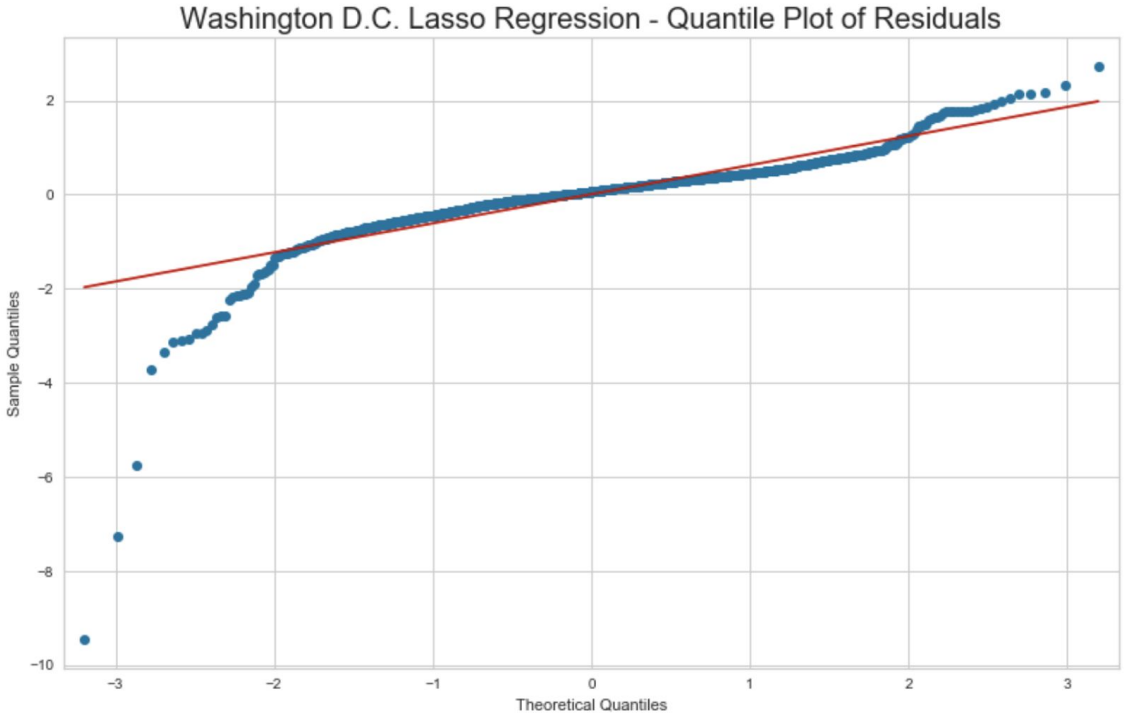


Figure 42: Washington D.C. Lasso Regression Model - Quantile Plot of Residuals

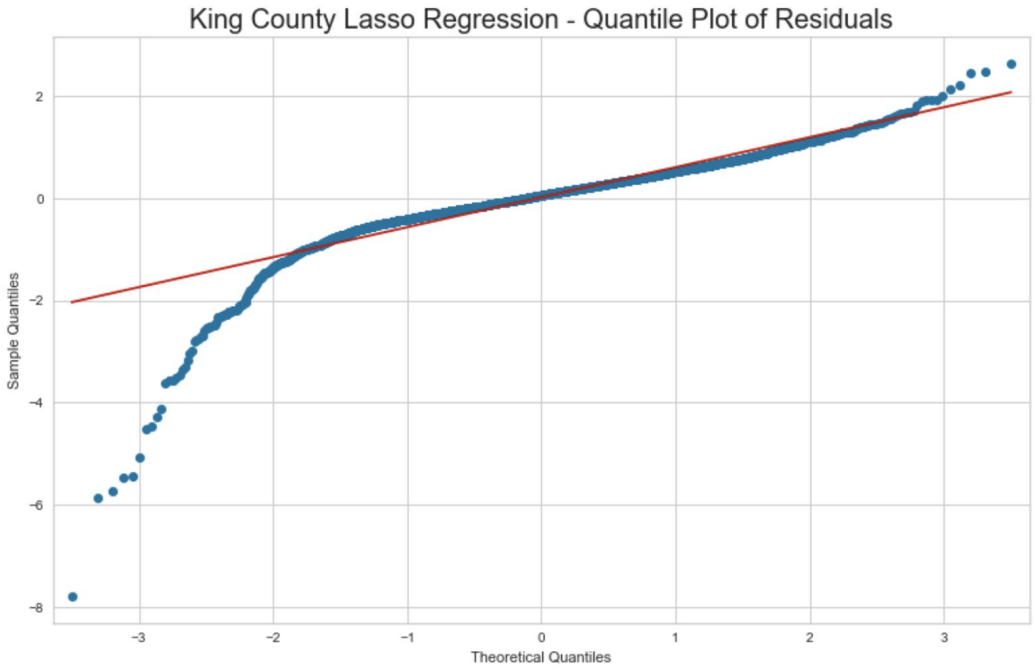


Figure 43: King County Lasso Regression Model - Quantile Plot of Residuals

As the theoretical quantile values get larger, the sample quantiles tend to the red line much closer than my original models. However, the Lasso Regression models for both the Washington D.C. and King County data struggle for lower theoretical quantiles. This implies that there is not a normal distribution of errors for each model, which highlights a weakness for the predictive power of them.

10. Baseline Model Extension: Random Forest

In my baseline model, I began with a linear regression model where I did not modify any of the features. I built the model solely with the original features to see how a model would score when considering all of the housing features. I then moved on to apply a lasso regression model to the data set, which in turn eliminated certain features from each data set that gave the models a high penalty. As an extension to the baseline models I defined, I applied an ensemble learning method: Random Forest.

Random forest regression is an ensemble of randomized decision trees. Decision trees fall victim to overfitting as a result of being trained and built on specific data. Different data can result with vastly different decision trees, which is an issue when we are trying to build a model that can generalize to data beyond the training subset. Random forest takes these decision trees and combines them together to form one cohesive model.

Random forest is also a bagging technique. Bagging is a combination of bootstrapping and aggregation. By sampling from the population with replacement (bootstrap) and aggregating these models built on these samples together, we can avoid overfitting a model to a single subset of the data. These individual models are run independently and in parallel, and do not interact with each other. This is good to keep training completely uninfluenced by each other.

Each model I built was on a random tree containing a bootstrapped sample of my housing data. The predictions for the housing price of each tree will be used to aggregate the models into a single general model.

a) Build Initial Random Forest Model and Hyperparameter Tuning

Since random forest regressors do not fall victim to overfitting, it didn't make sense to look at how well the model does based on the training data. This is where the OOB (out-of-bag) error comes in. The OOB error is what is used to measure the prediction errors of random forest models, since this error is based on how well the model predicts from the bootstrapped samples.

I first built a base set of random forest models with various settings for the *max_features* parameter. This *max_features* parameter represents the number of features to consider when deciding which feature at a split is the best one to use. For example, when given a node in my decision tree, the *max_features* parameter tells me how many random different features to consider in order to decide how to proceed down the tree. This process of selecting a certain number or random features helps prevent overfitting.

Once the initial base set of models were built, I built a visualization to illustrate the OOB errors of the models on the training data. This helped with determining how the models could be improved and led into the next step of hyperparameter tuning. By examining how manipulating the *max_features* and *n_estimators* (*n_estimators* is the number of trees that are made in the forest), I examined the model that returned me the best OOB error to use to compare to my base model:

Figure 44: Washington D.C. Random Forest Regressors with OOB error rate

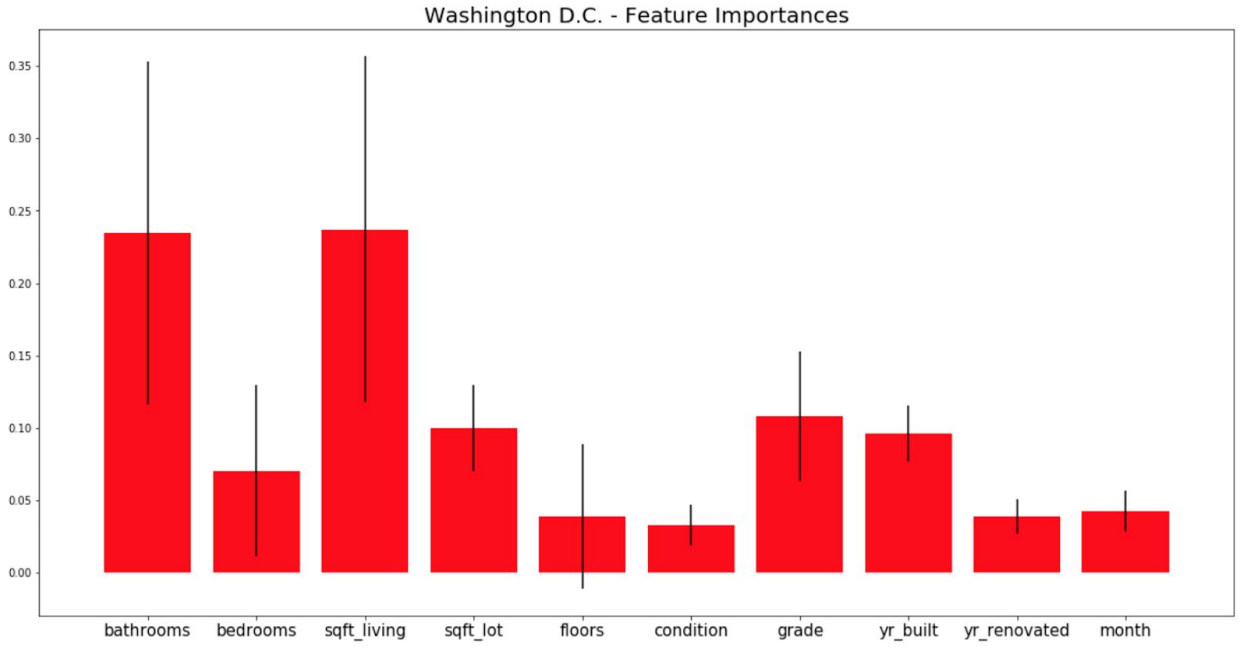
In the above graph, it appeared that when *max_feautures* = 'sqrt', the model's OOB error rate didn't even appear. This is because after looking deeper into the error rate, the error rate was identical to when the *max_features* = None. This highlights the fact that there appears to be a certain threshold for the *max_features* before the error rate does not change. However, when I set the *max_features* = 'log2', the error rate decreased significantly. The vertical line that I drew in the above plot is the *n_features* value where the error rate for the model was the lowest. This model (*max_features* = 'log2', *n_estimators* = 167) appeared to be the best model for the Washington D.C. training data set, and I used this same process to find the best two parameters to build out the Random Forest regressor for the King County data set:

Figure 45: King County Random Forest Regressors with OOB error rate

Similar to the Washington D.C. random forest regressor, the King County random forest regressor for *max_features* = 'sqrt' doesn't show on the graph. However, the trend for the OOB errors for *max_features* = 'log2' appeared to decrease as *n_estimators* increases. The best n_estimator for the King County random forest model is 300 when I tested the estimator values ranging from 100 to 300, so I used this parameter as the best model for the King County data. Now that I have the parameters for the random forest models, I will compare them to their baseline counterparts and see how well they perform on the test data.

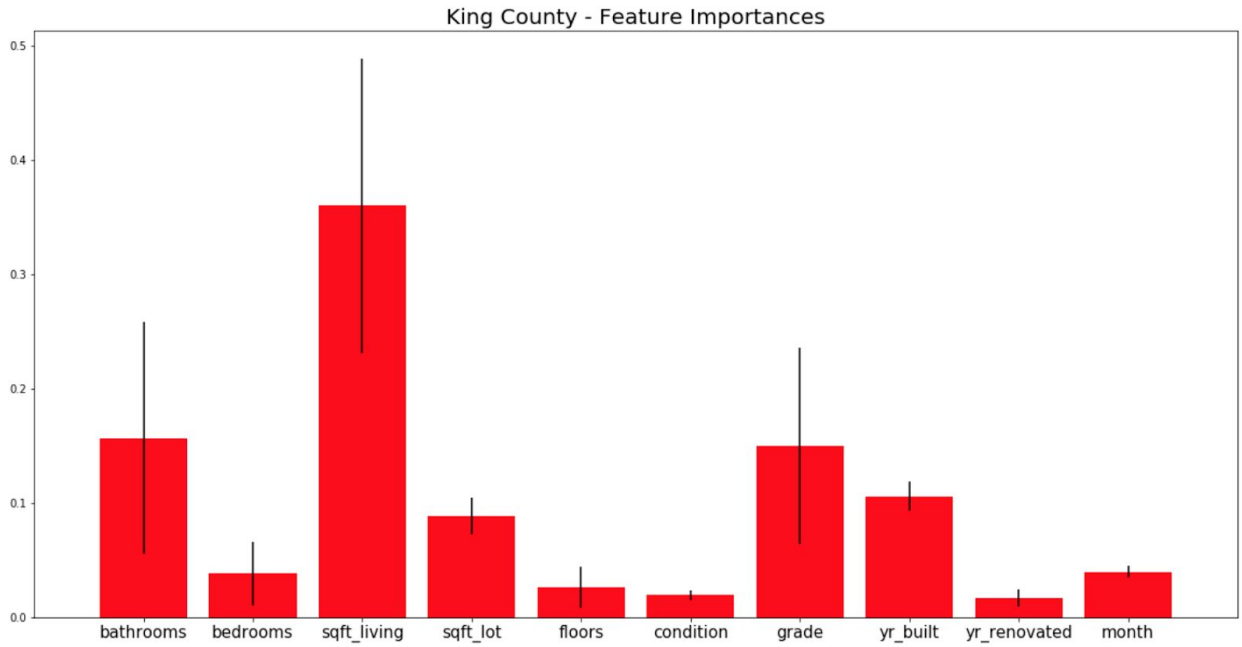
b) Feature Importance

Random Forest Regressors (RFR) belong to the family of non-parametric models, and do not possess a known closed form. What this means is that I couldn't directly look at the model coefficients like I did for the Linear Regression baseline models that took the base form $y = a + b1x1 + b2x2 + ...$ Instead, what I did was look at the feature importance of each housing feature for the random forest models. Scikit-learn's *feature_importances_* model attribute allowed me to examine the importance values of each feature. Higher values equates to higher importance. Below is a bar chart of the importances for each random forest model:



	Features	Importance Value
0	bathrooms	0.234591
1	bedrooms	0.070417
2	sqft_living	0.237226
3	sqft_lot	0.099992
4	floors	0.039018
5	condition	0.032830
6	grade	0.108064
7	yr_built	0.095940
8	yr_renovated	0.039032
9	month	0.042889

Figure 46: Washington D.C. Random Forest Regressor Feature Importance Values



	Features	Importance Value
0	bathrooms	0.156772
1	bedrooms	0.037921
2	sqft_living	0.360064
3	sqft_lot	0.088141
4	floors	0.025835
5	condition	0.019109
6	grade	0.150021
7	yr_built	0.105644
8	yr_renovated	0.016768
9	month	0.039724

Figure 47: King County Random Forest Regressor Feature Importance Values

From Figures 46 and 47, it is clear that the *sqft_living* had the highest importance for both the Washington D.C. random forest model and the King County random forest model. However, one distinction that I immediately noticed was the *bathrooms* feature. In the D.C. Random Forest model, the *bathrooms* and *sqft_living* had similar importance. However, the *bathrooms* in the King County model was significantly lower. This highlights a major difference in how the random forest regression algorithm was being affected by different features. Washington D.C. appeared to place higher importance on the number of bathrooms to determine the price of a

house, whereas King County placed significantly more importance on the living square feet than any other feature.

c) Evaluation of the Random Forest Models

Similar to my baseline model, I evaluated the Random Forest models using both the R-squared values and Mean Absolute Errors and compare them to their baseline counterparts.

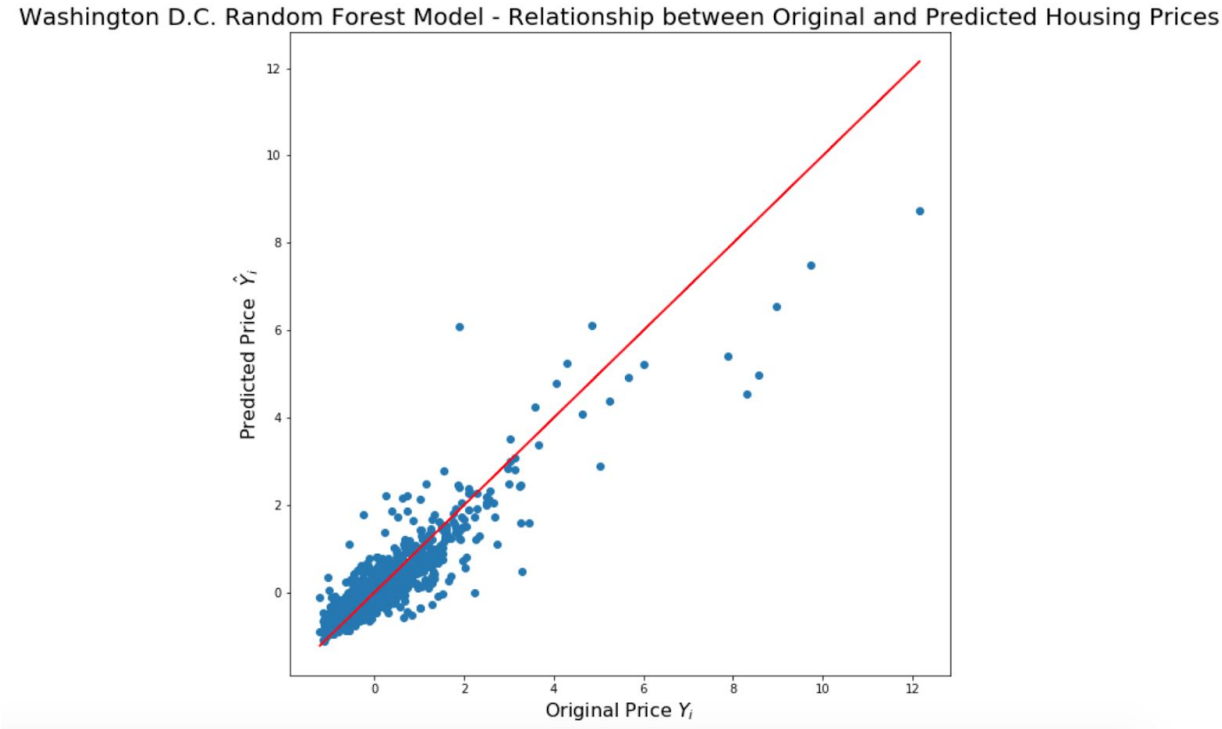


Figure 48: Washington D.C. Random Forest Original vs. Predicted Housing Price on Test Data

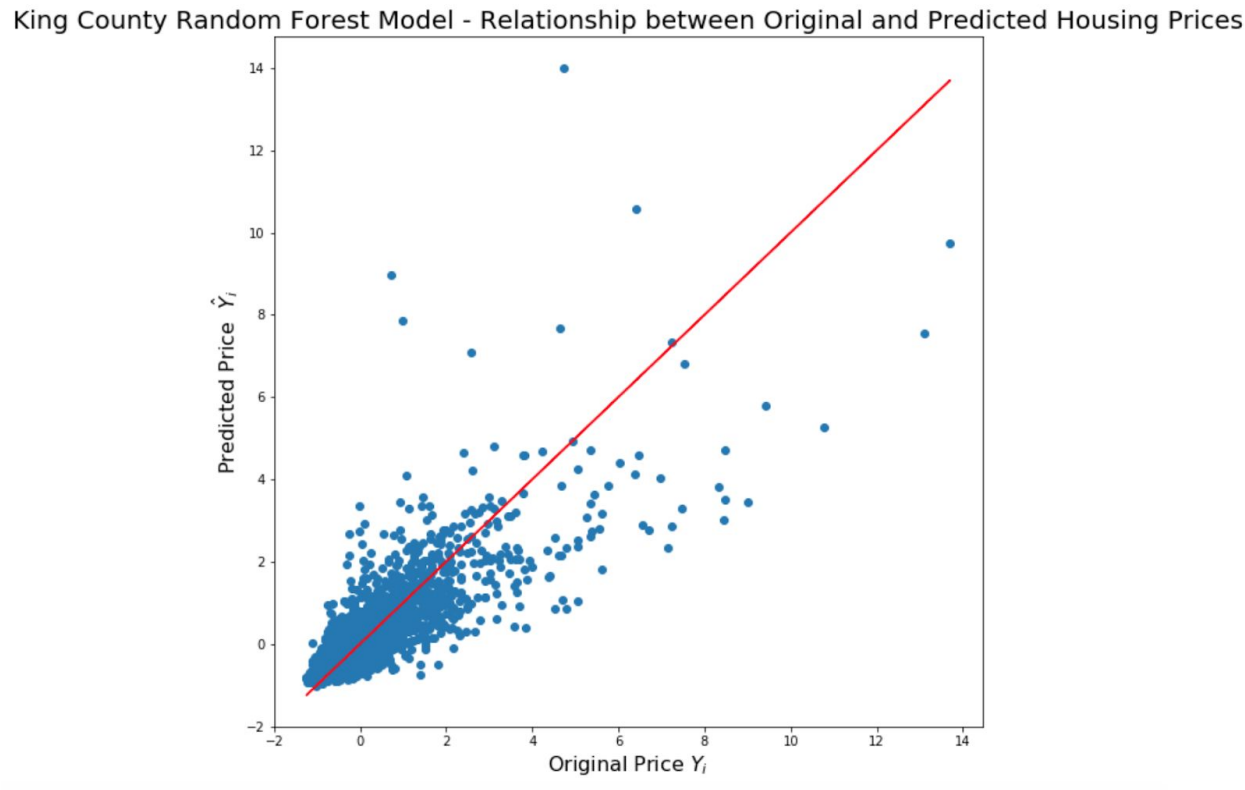


Figure 49: King County Random Forest Original vs. Predicted Housing Price on Test Data

i) R-Squared Values

```
Washington D.C. Random Forest Training R-Squared Score: 0.963895170964208
King County Random Forest Training R-Squared Score: 0.9593744130898185
Washington D.C. Random Forest Test R-Squared Score: 0.8241541547155612
King County Random Forest Test R-Squared Score: 0.6498936476589086
```

Figure 50: Random Forest Regressor R-Squared values

For the training data, the R-squared values were high. Both are above .95 out of 1, which means that 95% or more of the variance of the prices are accounted for in both models. It is important, however, to also look at how they did for the test data. The Washington D.C. R-squared value went down, but not as significantly as the King County R-squared value. This means that the Washington D.C. Random Forest model generalized better than the King County model.

Additionally, I compared the R-squared values to the Linear Regression models. The baseline R-squared values for the Linear Regression models were 0.5 for Washington D.C. and 0.56 for King County (for the training data). Already, the R-squared values were lower for the Linear Regression models, which supports the fact that the Random Forest models are stronger. Additionally, the Random Forest models generalized better on the test data.

ii) Mean Absolute Error

```
Washington D.C. Test Mean Absolute Error: 0.24543597359940245
King County Test Mean Absolute Error: 0.360429926738529
```

Figure 51: Random Forest Regressor Mean Absolute Error values

Similar to the R-squared values for the Random Forest models, the mean absolute errors were also significantly lower for the test data when using the Random Forest models. For the Washington D.C. Linear Regression model, the MAE was 0.42, and for the King County model the MAE was 0.45. For both data sets, the MAE dropped when using the Random Forest Regressor, thus making the predictions more accurate and reliable.

11. Conclusion

Creating both the Linear Regression models and Random Forest Regressor models on the Washington D.C. and King County housing sales highlights the fact that no one model can encompass all intricacies of large amounts of data. However, some models are better at capturing what the business problem is, and in my case the Random Forest Regressor was more accurate with predicting housing sale prices given the data I had to work with. The r-squared score and mean absolute values of the models' predictions on each data set were strong indicators of the fit of each model. Using the Random Forest Regressor model also allowed me to examine the "importance" of each housing feature, which told me how impactful certain features were for each model. Building individual models on geographically different locations also highlighted that housing features for different areas are treated differently and impact the sale price in various ways.

12. Future Work and Recommendations to Client

When I built each model for Washington D.C. and King County, I included all data points and kept each individual feature as-is. I didn't make modifications to the data set because I wanted to see how all the data would affect the models. However, I believe feature engineering could play a large part in the strength of the model. Engineering my own features from the raw data I was given could improve both the r-squared and mean absolute value metrics (the main metrics I used to determine the strength of my models) and I think this could be an interesting angle to approach building out the models. Additionally, when I built the Random Forest Regressors, the two main hyperparameters I manipulated were the *max_features* and *n_esimators*. I limited the *max_features* to three values: 'sqrt', 'log2', and 'None.' I also limited the *n_estimators* to the range of 100 to 300. Usually, increasing *n_estimators* results with lower OOB error rates, and *max_features* can range from 1 all the way to the total number of features the raw data set has. I would recommend tuning these hyperparameters even more and trying all different combinations to find the most optimal model, even if runtime at the testing stage increases. Overall, many different kinds of models can be used to represent and predict the housing data sale prices, and it would be interesting to see how different ones could fit various geolocations beyond Washington D.C. and King County.

I recommend prospective buyers to pay close attention to what is a hot commodity when it comes to buying a house in their area of interest. Some housing features can heavily impact the price of a house and buyers should be aware of what they feel is most important for their purchase. For example, if living square feet is the most important feature for a buyer, they should know that this will most likely increase the sale price (as observed for both Washington D.C. and King County). I also recommend to real estate agents to monitor housing sale trends. Although I only examined housing sales from May 2014 to May 2015, this data could change from location to location and throughout time certain housing features may become more important than others. 20 years ago, living square feet might not have impacted housing sales as much as it does recently, and being informed about trends like this could be very helpful to landing more sales.