Project Assessment

Rushit Shah, Harshi Thaker, Manthan Mehta

Introduction of Dataset

The **King County House Dataset** offers a plethora of information about the price, size, location, condition, and other characteristics of houses in Washington's King County. In this post, we'll show you how we used R to create a multivariate linear regression model to forecast property prices. The following is a comprehensive list of the modules that we utilized in this analysis. The code contained below includes many, but not all of them. Some of the variable names may be unclear, so here is a quick summary of each:

Column/Variable definitions

id - Unique ID for each home sold
date - Date of the home sale
price - Price of each home sold
bedrooms - Number of bedrooms
bathrooms - Number of bathrooms, where .5 accounts 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 in the house
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_renovated - The year of the house's last renovation
zip code - What zip code area 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

Data cleaning
Firstly, The date column in the original dataset was nonnumeric in the format **'yyyymmddT000000'** as shown below.
Secondly, we also filtered out the houses with more than 11 bedrooms as after doing an overview of the data, we found that these houses were outliers.

Hence, data cleaning was performed in order to transform the data into a consistent and usable format.

Code:

```
str_replace(kc_data$date, 'T000000', '')

kc_data$date = as.Date(kc_data$date, format = '%Y%m%d')

kc_data = kc_data %>% filter(bedrooms <= 11)

```

Comparison:

Original data	Cleaned data
20141013T000000	2014-10-13
20141209T000000	2014-12-09
20150225T000000	2015-02-25
20141209T000000	2014-12-09

Module 1: Statistics

What is the average home price in the zip code 98034 and what is the standard deviation?

Code:

```
kc_zip = kc_data %>% filter(zipcode == 98034)
mean(kc_zip$price)
sd(kc_zip$price)

```

Result:

```
> kc_zip = kc_data %>% filter(zipcode == 98034)
> mean(kc_zip$price)
[1] 521652.9
> sd(kc_zip$price)
[1] 309625.6
> 
```

Module 2: Linear Regression

What are the best predictors for home price from the ones in the file? Show the model?

The following steps were followed for the regression model.

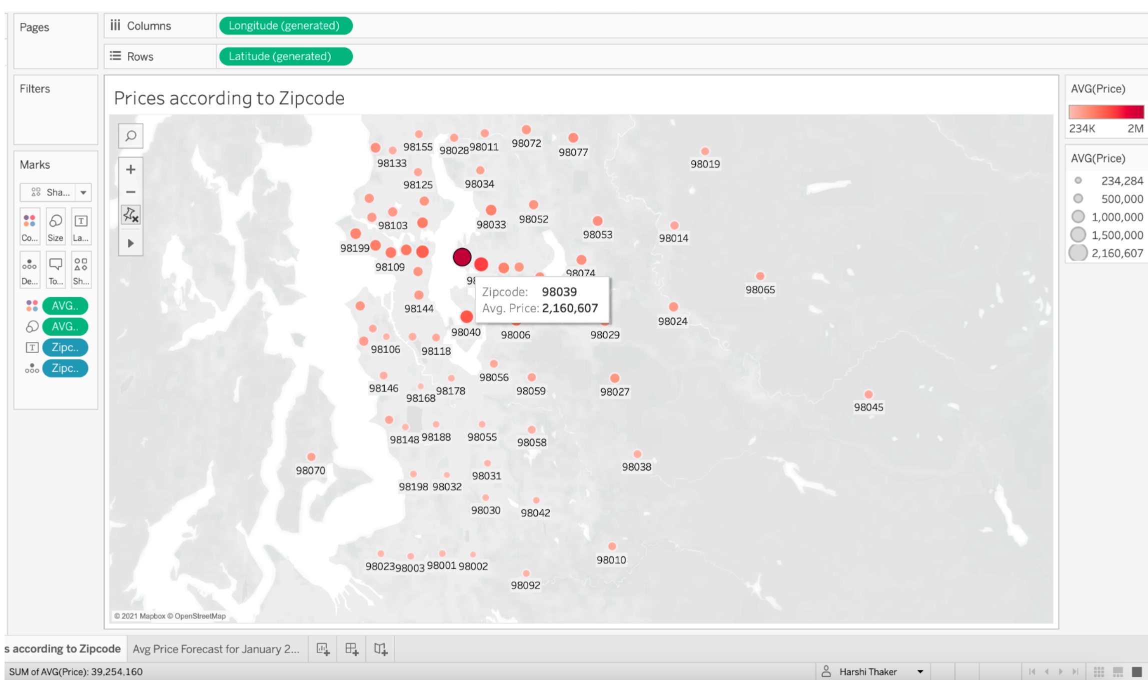
- *Data preparation*
  The columns yr_built, yr_renovated, lat, long and zip code even though important did not have any quantitative data which could be utilized for the analysis. In order to use them in an efficient manner we created the following new columns from the aforementioned ones;

  yr_sold - This column was created to calculate the age of the houses sold and was derived from the **‘date’** column as shown below.

  age - We created this column to calculate the age of the house. This was derived from the difference between the **‘yr_sold’** column and **‘yr_built’** column.

  renovated - This column was created to help identify if a recently renovated house had an effect on the price. To further simplify, if the house was renovated in the past 10 years or built in the last 5 years we assigned a value 1 and for all other conditions we assigned a value of 0. S

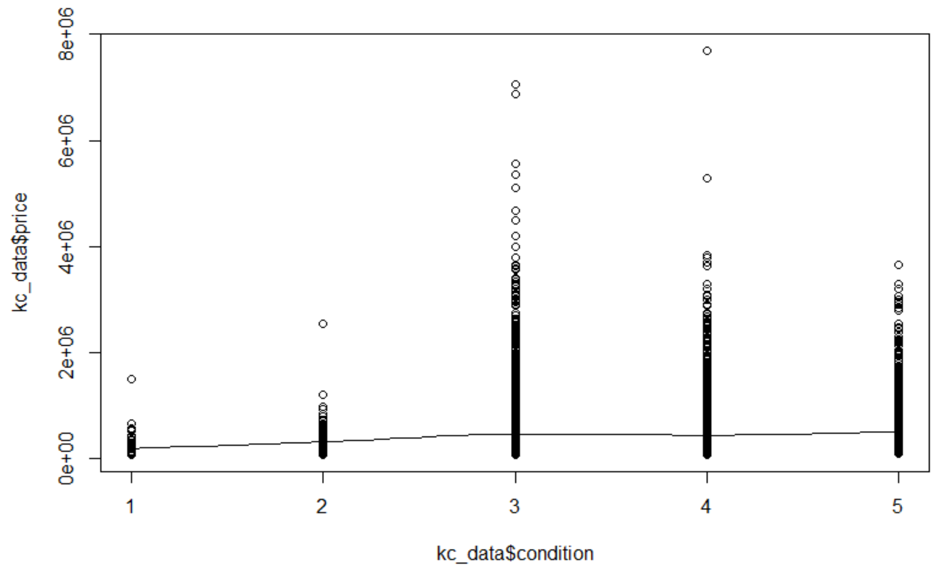
  dist - Based on the below plot, we realized that from a point (downtown Seattle), the prices might vary for the houses at different locations. So we calculated the distance of the houses from that one point (as labelled below) to take into consideration its effect on the price model.



```r
kc_data$yr_sold = year(kc_data$date)
kc_data$age = kc_data$yr_sold - kc_data$yr_built
kc_data$renovated = ifelse(((kc_data$yr_sold - kc_data$yr_renovated) <= 10) | (kc_data$age <= 5), 1, 0)


kc_dist_df = data.frame(lat = 47.6062, long = -122.3321)

for(i in 1:nrow(kc_data)){
  vec <- c(47.6062, -122.3321)
  kc_dist_df[i,] <- vec
}
kc_dist_df

latitude = kc_data$lat
longitude = kc_data$long

kc_data_dist_df = data.frame(latitude, longitude)

kc_data$dist = geodist(kc_dist_df, kc_data_dist_df, paired = TRUE,
                       sequential = FALSE, pad = FALSE, measure = "haversine")
view(kc_data)
```

	yr_sold	age	renovated	dist
16	2014	23	0	32411.781
16	2015	24	0	32411.781
91	2014	67	0	18067.438
23	2014	62	0	18157.391
20	2015	85	0	15104.253
04	2015	64	1	4061.260
00	2015	64	0	4077.100
58	2015	55	0	13575.661
34	2014	9	0	7463.856
03	2014	69	0	23129.056
10	2014	90	0	16279.256
69	2014	89	0	16366.182
69	2015	90	0	16366.182
20	2014	89	0	9075.206
00	2014	112	0	2288.871
07	2014	0	1	2679.655
07	2014	113	0	1491.580
01	2014	7	0	1426.214
35	2014	73	0	8481.399
99	2014	16	0	14626.452

- *Linear Model*
  For the linear model, we initially took into consideration the 'condition' column as we thought it should have a higher impact on the model. But based on the below correlation graph, we see that the condition column has minimal impact with respect to price.

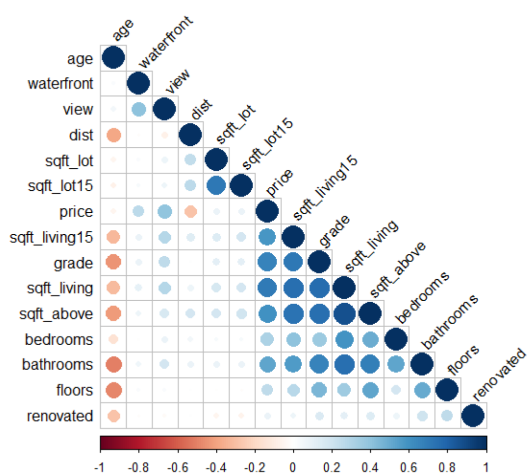


To find the correlation between the different columns in the dataset, we designed the below graph.

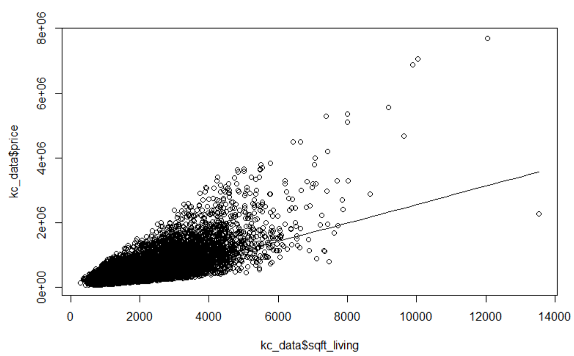
Code:

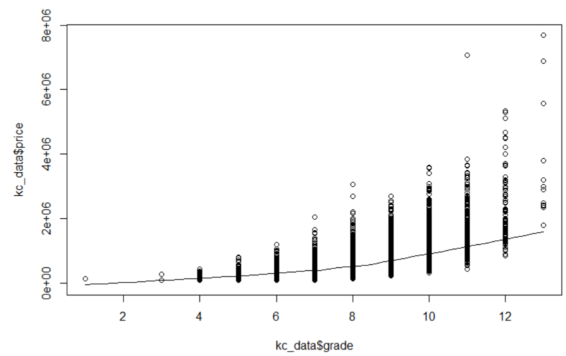
```
kc_corr = kc_data %>% select(price,bedrooms, bathrooms, sqft_living, sqft_lot, floors, waterfront, view, grade,
                             sqft_above,sqft_living15, sqft_lot15, age, renovated, dist)

corr = cor(kc_corr)

library(corrplot)

corrplot(corr, type = "lower", order = "hclust",
         tl.col = "black", tl.srt = 45)
```

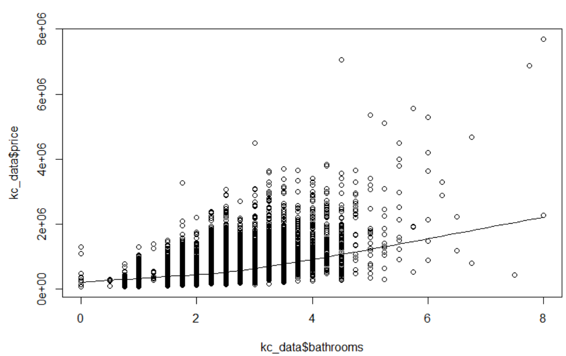
Result:

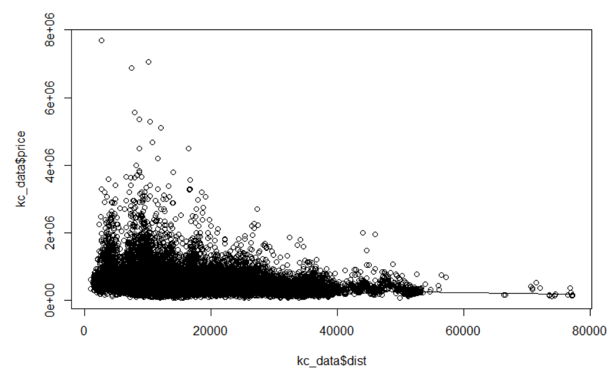


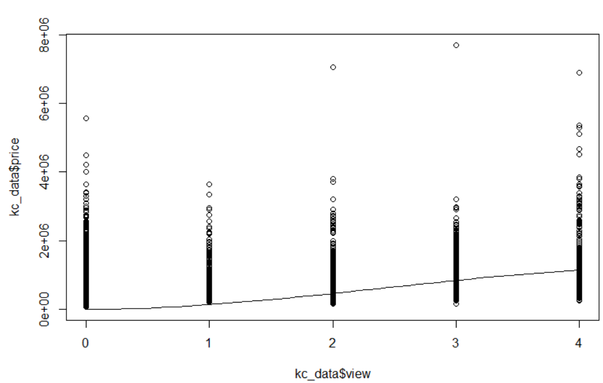
- **Modelling:**
  - The correlation graph points out that columns like sqft_living15 and sqft_above had high correlation with sqft_living and therefore only sqft_living was taken into account during the modelling process.
  - Next, the price column does not show that high of a correlation with other columns and hence we have restricted the model to the below mentioned columns.











Code:

```
kc_data_model = lm(price ~ sqft_living + grade + bathrooms + dist + view - 1, kc_data)
summary(kc_data_model)
```

```
78:1    (Top Level) ÷

Console   Terminal ×   Jobs ×

R 4.1.1 · ~/

Call:
lm(formula = price ~ sqft_living + grade + bathrooms + dist + 
    view - 1, data = kc_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-1245555  -113029   -16300    76697  4335150 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
sqft_living  2.321e+02  2.614e+00  88.802  < 2e-16 ***
grade        3.801e+04  8.065e+02  47.127  < 2e-16 ***
bathrooms   -1.675e+04  2.972e+03  -5.635 1.78e-08 ***
dist        -1.156e+01  1.343e-01 -86.109  < 2e-16 ***
view         8.560e+04  2.003e+03  42.733  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 215200 on 21607 degrees of freedom
Multiple R-squared:  0.8914,	Adjusted R-squared:  0.8914 
F-statistic: 3.548e+04 on 5 and 21607 DF,  p-value: < 2.2e-16
```

Result:
The linear model gives a R-squared value of **0.8914**.

Module3: Supervised Clustering
Cluster the data using your choice of columns.

For supervised clustering, we have taken the **‘waterfront’** column as the dependent variable. Here we try to identify and analyze if the waterfront variable contributes to the view rating, i.e. how correlated they are.

Code:

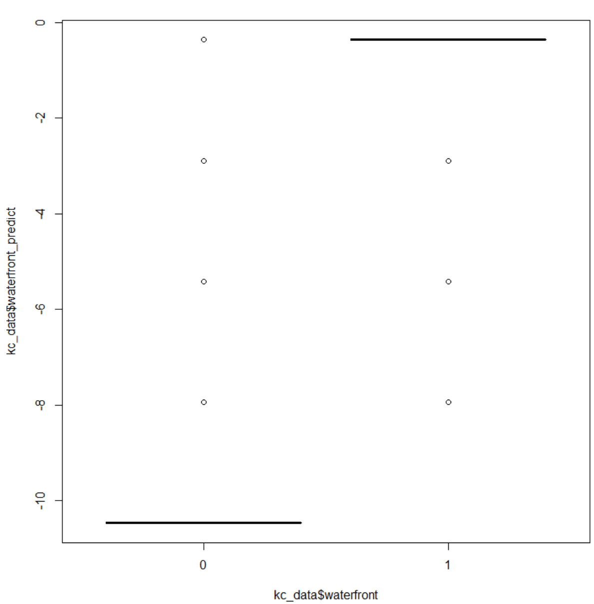
```
waterfront_model = glm(kc_data$waterfront ~ kc_data$view, family = 'binomial')
summary(waterfront_model)

kc_data$waterfront_predict = predict(waterfront_model)
boxplot(kc_data$waterfront_predict ~ kc_data$waterfront)

threshold_wf = -5
kc_data$wf = ifelse(kc_data$waterfront_predict <= threshold_wf, 0, 1)

confusionMatrix(as.factor(kc_data$waterfront),
                as.factor(kc_data$wf))
```

Result:
Based on the results as displayed down in the confusion matrix, the following observations are evident;

- Out of 163 houses which have a waterfront, 154 houses have a high rating in terms of views
- 94.5% of the people agree that having a waterfront provides a better view and ultimately a higher view rating
- Finally, out of 21449 houses which do not have a waterfront, 96.8% of them had a lower rating in terms of views.

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 20774   675
         1     9   154

               Accuracy : 0.9684
                 95% CI : (0.9659, 0.9706)
    No Information Rate : 0.9616
    P-Value [Acc > NIR] : 6.951e-08

                  Kappa : 0.3017

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9996
            Specificity : 0.1858
         Pos Pred Value : 0.9685
         Neg Pred Value : 0.9448
             Prevalence : 0.9616
         Detection Rate : 0.9612
   Detection Prevalence : 0.9925
      Balanced Accuracy : 0.5927

       'Positive' Class : 0
```

Box plot for predicted waterfront values based on view vs waterfront columns. We decided to take a threshold value of -5 based on the plot.

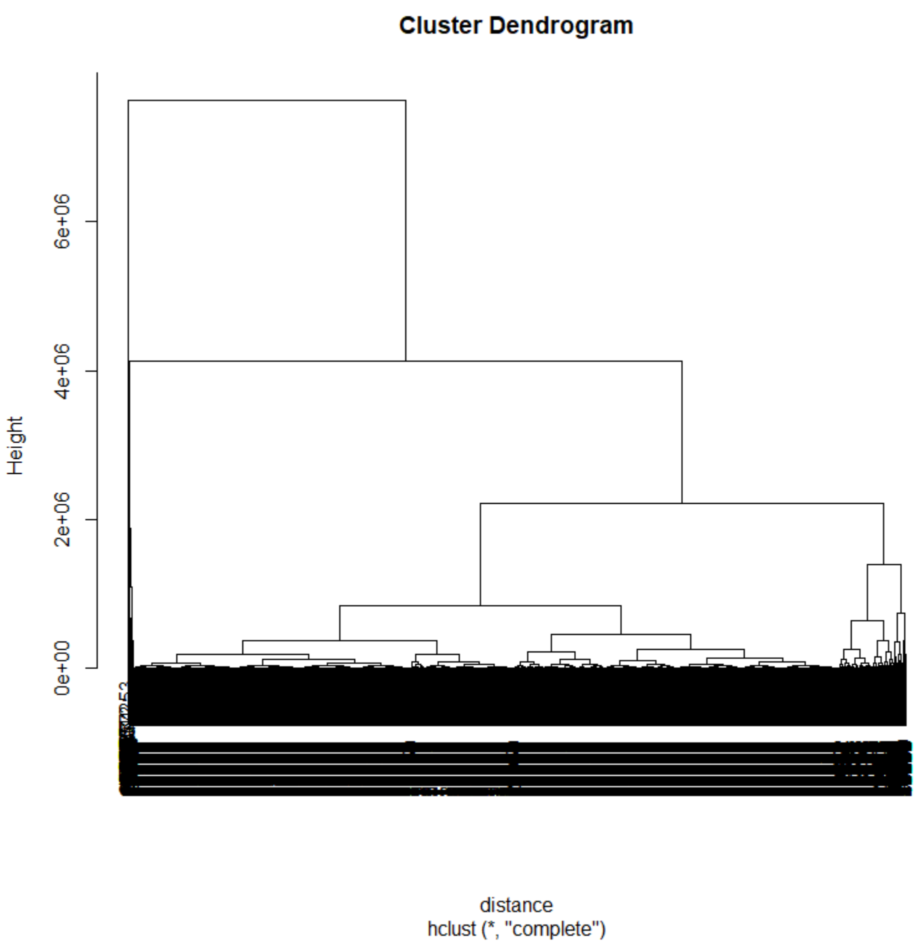


Module 4: Unsupervised Clustering

Cluster the data using these columns: bedrooms, bathrooms, sqft_living, floors, waterfront, price. Name the clusters.

Code:

```
kc_cluster_data = kc_data %>% select(bedrooms,bathrooms,sqft_living,floors,waterfront,price)

set.seed(1234)

distance = dist(kc_cluster_data[,1:6])
cl = hclust(distance)
plot(cl)

kc_cluster_data$cluster = kmeans(kc_cluster_data, 3)$cluster
clusplot(kc_cluster_data, kc_cluster_data$cluster)

```



Based on the height vs distance plot in the above cluster dendrogram, we observed that there are 3 distinct clusters.

Below mentioned is the summary of the 3 clusters

```
> summary(kc_cluster_data %>% filter(cluster == 1))
    bedrooms        bathrooms       sqft_living       floors        waterfront         price           cluster
 Min.   : 0.000   Min.   :0.000   Min.   : 290   Min.   :1.000   Min.   :0.00000   Min.   : 75000   Min.   :1
 1st Qu.: 3.000   1st Qu.:1.500   1st Qu.:1280   1st Qu.:1.000   1st Qu.:0.00000   1st Qu.:281000   1st Qu.:1
 Median : 3.000   Median :2.000   Median :1652   Median :1.000   Median :0.00000   Median :368000   Median :1
 Mean   : 3.189   Mean   :1.897   Mean   :1730   Mean   :1.407   Mean   :0.00141   Mean   :372012   Mean   :1
 3rd Qu.: 4.000   3rd Qu.:2.500   3rd Qu.:2100   3rd Qu.:2.000   3rd Qu.:0.00000   3rd Qu.:460000   3rd Qu.:1
 Max.   :11.000   Max.   :7.500   Max.   :5461   Max.   :3.500   Max.   :1.00000   Max.   :584000   Max.   :1
> summary(kc_cluster_data %>% filter(cluster == 2))
    bedrooms        bathrooms       sqft_living       floors        waterfront          price             cluster
 Min.   : 0.000   Min.   :0.000   Min.   : 860   Min.   :1.000   Min.   :0.000000   Min.   : 584950   Min.   :2
 1st Qu.: 3.000   1st Qu.:2.000   1st Qu.:2120   1st Qu.:1.000   1st Qu.:0.000000   1st Qu.: 650785   1st Qu.:2
 Median : 4.000   Median :2.500   Median :2630   Median :2.000   Median :0.000000   Median : 749450   Median :2
 Mean   : 3.726   Mean   :2.507   Mean   :2703   Mean   :1.669   Mean   :0.009875   Mean   : 797824   Mean   :2
 3rd Qu.: 4.000   3rd Qu.:2.750   3rd Qu.:3200   3rd Qu.:2.000   3rd Qu.:0.000000   3rd Qu.: 885438   3rd Qu.:2
 Max.   :10.000   Max.   :6.750   Max.   :7480   Max.   :3.500   Max.   :1.000000   Max.   :1393000   Max.   :2
> summary(kc_cluster_data %>% filter(cluster == 3))
    bedrooms       bathrooms      sqft_living        floors        waterfront         price            cluster
 Min.   :2.000   Min.   :1.500   Min.   : 1890   Min.   :1.000   Min.   :0.0000   Min.   :1395000   Min.   :3
 1st Qu.:4.000   1st Qu.:2.750   1st Qu.: 3452   1st Qu.:2.000   1st Qu.:0.0000   1st Qu.:1550000   1st Qu.:3
 Median :4.000   Median :3.250   Median : 4095   Median :2.000   Median :0.0000   Median :1750000   Median :3
 Mean   :4.178   Mean   :3.449   Mean   : 4285   Mean   :1.862   Mean   :0.1269   Mean   :1989852   Mean   :3
 3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.: 4830   3rd Qu.:2.000   3rd Qu.:0.0000   3rd Qu.:2220000   3rd Qu.:3
 Max.   :9.000   Max.   :8.000   Max.   :13540   Max.   :3.500   Max.   :1.0000   Max.   :7700000   Max.   :3
>
```

Cluster 1 - Economic houses
Cluster 2 - Mid-range houses
Cluster 3 - High-end houses

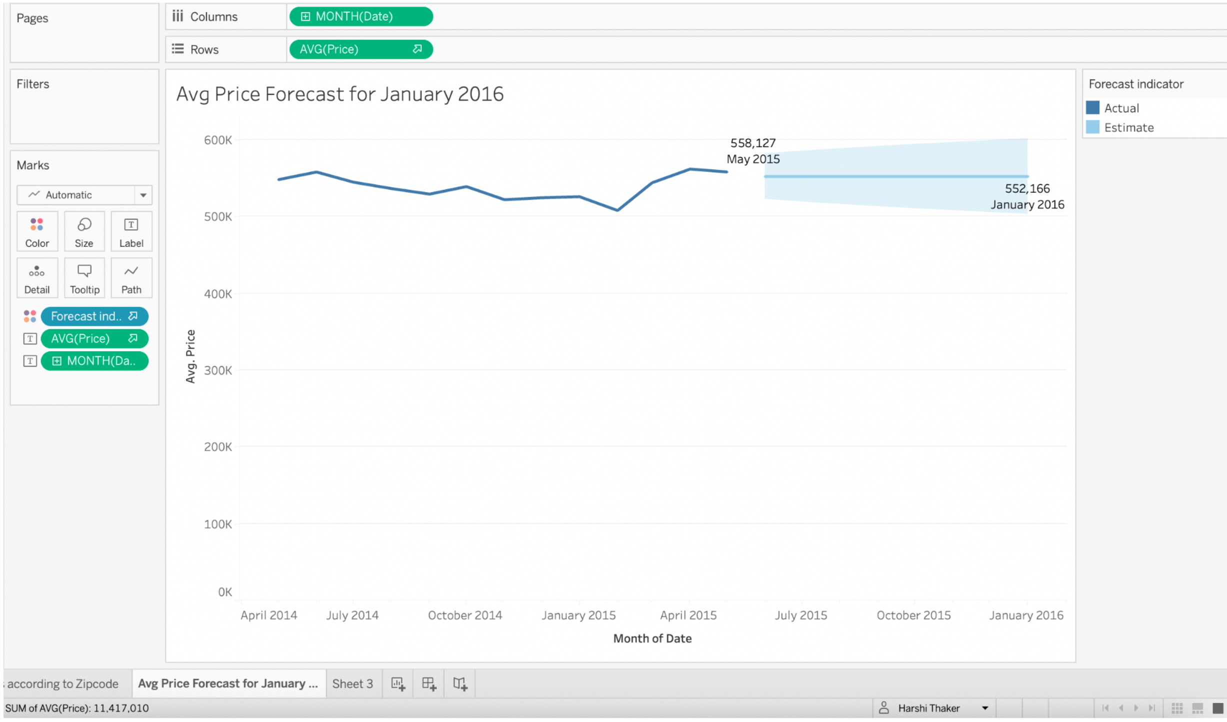
The clusters have been classified based on the house features and amenities.

- Cluster 1 groups basic houses with small bedrooms and sufficient bathrooms but less space comparatively.
- Cluster 2 groups slightly better houses with almost the same number of bedrooms and bathrooms but with larger space as compared to cluster 1.
- Cluster 3 groups luxurious houses with the largest area along with the possibility of having a waterfront which ultimately results in a good view.

Module 5: (Neural Network)
Does the model improve if instead of a linear model we use a neural network?

Analysis: The neural network model is not comparable to the linear model since we were unable to find a neural network with the same parameters as the linear model, causing our model to converge. Because the neural network model utilizes scaled pricing whereas the linear model uses real prices, the comparison would need further modifications.

Module 6: Forecasting
What is the expected average home price for January 2016 based on the average home prices from previous months?



Conclusion - The forecasting feature of Tableau predicts that in the month of January 2016, the average price of the houses will stabilize in the range of $550000.