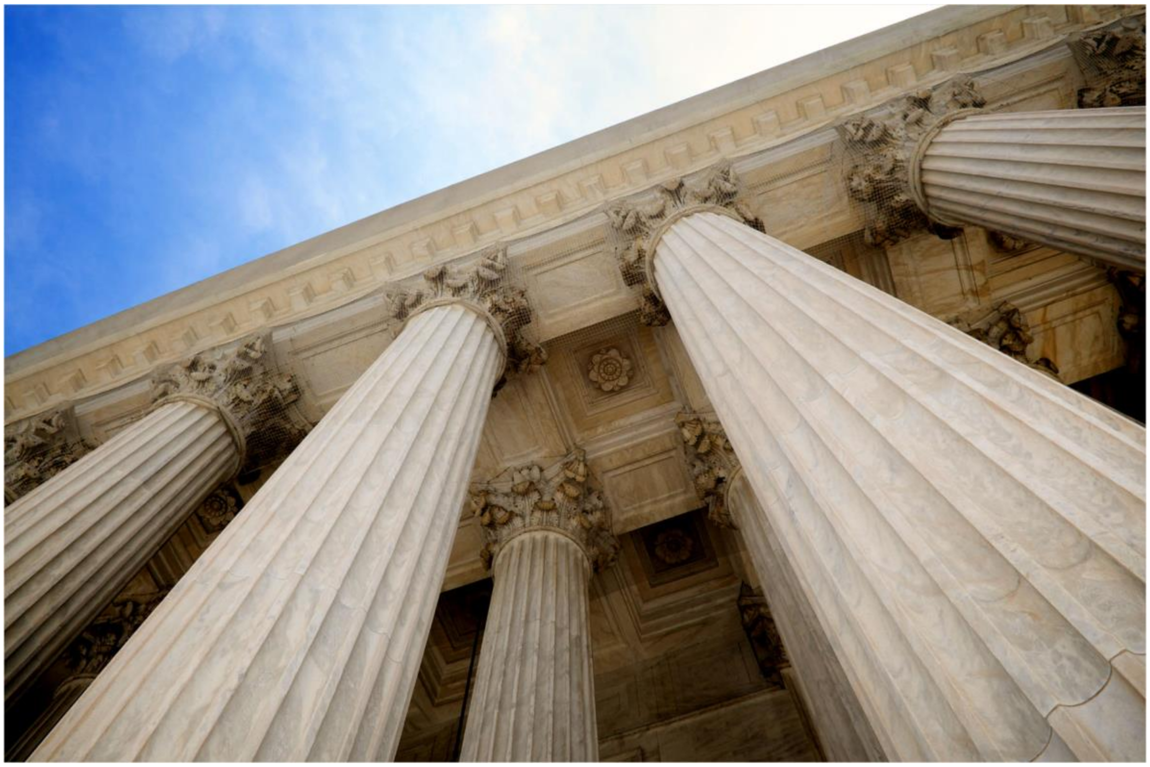


Property Price Prediction

DATA MINING PROJECT REPORT

U01382090| Rakesh Parappa | CS619 – Data Mining

Introduction

The report entails analyzing different variables related to house sale prices for King County, which includes Seattle. Below are the questions answered in the report:

- To see if there is any relationship between attributes like the area, grade and zip code with the price of the property.
- To predict the price of the unknown or new data based on the attribute values.

The models used for predicting the price are linear regression and k-nearest neighbor. The associations are defined based on k-means clustering

Dataset Description

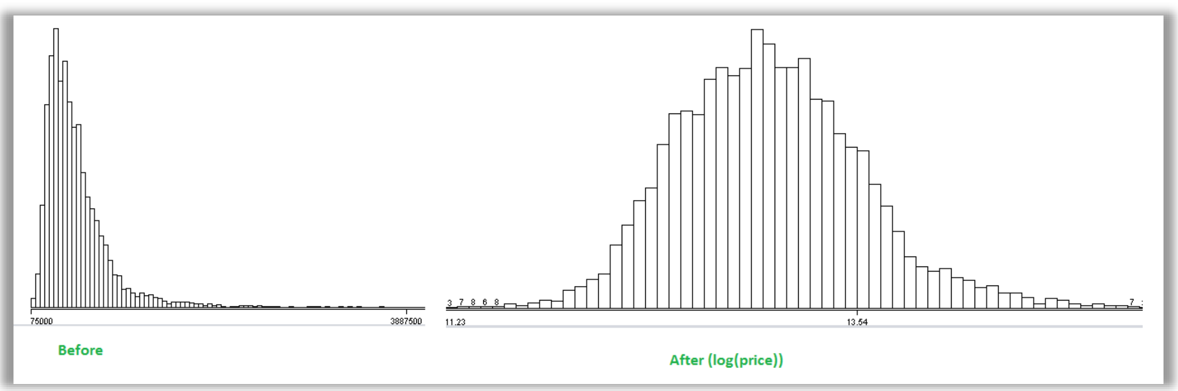
The dataset for this project is obtained from kaggle's website. It includes homes sold between May 2014 and May 2015. It has around 19 house features or variables with 21613 observations. Below are the considered key attributes and their descriptions:

Variables	Description
Price	Price of the property
Bedrooms	Discrete value of the number of bedrooms in the house
Bathrooms	Continuous variable of the number of bathrooms in the house
Sqft_living	Area of the living room
Sqft_lot	Area of the lot
Floors	Discrete value of the number of floors
Waterfront	Binary variable with values 0(No) or 1 (Yes)
View	Discrete value with range 0 being the lowest to 4 being highest
Condition	Discrete value with range 1 being the lowest to 5 being the highest
Grade	Discrete value with range 1 being the lowest to 13 being the highest
Sqft_above	Area of the top floor
Sqft_basement	Area of the basement
Yr_built	Year the house was built
Yr_renovated	Year the house was renovated
Zipcode	Zip code of the place where the house is situated

Data Preparation

To accommodate the variables into models and extract the right meaning from the observations, I have done some of the transformations which are as follows:

- I used Weka to transform numeric data of the attribute "waterfront" to binary, where the value 0 signifies "No" and 1 signifies "Yes".
- I used Weka to transform the numeric data of "view", "condition", and "grade" to nominal
- I used Wek to perform equal-height discretization of 10 bins on the attribute "yr_built".
- I used Weka to convert the numeric values of the below attributes to binary as there were many observations with the value 0:
  - "yr_renovated" – 0 for not renovated and 1 for renovated
  - "sqft_basement" – 0 for no basement and 1 for basement.
- I have added new attribute "city" with 24 distinct values replacing attribute "zipcode" which had 70 distinct values.
- I used Weka to convert the left skewed variable "price" to normally distributed by applying log. This also reduced the high range values of price from millions to tens.



Data Analysis

After the initial preparation of the data, I have split the data into a training set with 80% of data and a test set with 20% data. The training dataset is used to build a model whereas the test dataset is used to evaluate. Below are the models used.

Linear Regression

Linear regression model is a good place to start with if your predicting variable is numerical and based on one or more independent variables. The independent variables could be either numerical or nominal.

Given a dataset with y_i being independent variable and x_i being dependent variable, linear regression model fits a function to calculate y_i such that:

$$y = \sum_{i=0}^{\infty} \beta_i * x_i$$

Where, i = Number of variables

β_i = Regression coefficients

The above equation states that the change in the value of x_i by one unit, changes the value of y by β_i.

Based on Linear Regression schema - `LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4` and 17920 instances and 15 attributes, Weka has constructed a below regression model.

```
Log(Price) =

       -0.01 * bedrooms +
        0.05 * bathrooms +
        0    * sqft_living +
        0    * sqft_lot +
        0.01 * floors +
        0.32 * waterfront=1 +
        0.13 * view=1,2,3,4 +
       -0.02 * view=2,3,4 +
        0.06 * view=3,4 +
        0.12 * view=4 +
        0.13 * condition=2,4,3,5 +
        0.2  * condition=4,3,5 +
       -0.06 * condition=3,5 +
        0.13 * condition=5 +
        0.15 * grade=6,7,1,8,9,10,11,12,13 +
        0.26 * grade=7,1,8,9,10,11,12,13 +
        0.09 * grade=1,8,9,10,11,12,13 +
        0.09 * grade=8,9,10,11,12,13 +
        0.15 * grade=9,10,11,12,13 +
        0.09 * grade=10,11,12,13 +
        0.07 * grade=11,12,13 +
        0    * sqft_above +
        0.09 * sqft_basement_binarized=1 +
```

```
        0.18 * yr_built='(1934.5-1946]','(1946-1957.5]','(1969-
1980.5]','(1980.5-1992]','(1911.5-1923]','(1992-2003.5]','(1923-
1934.5]','(2003.5-inf)','(-inf-1911.5]' +
       -0.09 * yr_built='(1946-1957.5]','(1969-1980.5]','(1980.5-
1992]','(1911.5-1923]','(1992-2003.5]','(1923-1934.5]','(2003.5-
inf)','(-inf-1911.5]' +
       -0.11 * yr_built='(1969-1980.5]','(1980.5-1992]','(1911.5-
1923]','(1992-2003.5]','(1923-1934.5]','(2003.5-inf)','(-inf-1911.5]'
+
        0.03 * yr_built='(1980.5-1992]','(1911.5-1923]','(1992-
2003.5]','(1923-1934.5]','(2003.5-inf)','(-inf-1911.5]' +
        0.29 * yr_built='(1911.5-1923]','(1992-2003.5]','(1923-
1934.5]','(2003.5-inf)','(-inf-1911.5]' +
       -0.27 * yr_built='(1992-2003.5]','(1923-1934.5]','(2003.5-
inf)','(-inf-1911.5]' +
        0.27 * yr_built='(1923-1934.5]','(2003.5-inf)','(-inf-1911.5]'
+
       -0.26 * yr_built='(2003.5-inf)','(-inf-1911.5]' +
        0.33 * yr_built='(-inf-1911.5]' +
        0.04 * yr_renovated_binarized=1 +
        0.06 * City=Kent,Enumclaw,Maple Valley,Renton,Black
Diamond,North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
       -0.04 * City=Enumclaw,Maple Valley,Renton,Black Diamond,North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.15 * City=Maple Valley,Renton,Black Diamond,North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.07 * City=Renton,Black Diamond,North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.02 * City=Black Diamond,North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.08 * City=North
Bend,Carnation,Duvall,Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.07 * City=Kenmore,Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
       -0.08 * City=Vashon,Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.13 * City=Bothell,Seattle,Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
```

```
       -0.04 * City=Fall
City,Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
       -0.03 *
City=Snoqualmie,Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellev
ue,Mercer Island,Medina +
        0.25 *
City=Kirkland,Woodinville,Issaquah,Redmond,Sammamish,Bellevue,Mercer
Island,Medina +
       -0.16 *
City=Woodinville,Issaquah,Redmond,Sammamish,Bellevue,Mercer
Island,Medina +
        0.03 * City=Issaquah,Redmond,Sammamish,Bellevue,Mercer
Island,Medina +
        0.09 * City=Redmond,Sammamish,Bellevue,Mercer Island,Medina +
       -0.07 * City=Sammamish,Bellevue,Mercer Island,Medina +
        0.21 * City=Bellevue,Mercer Island,Medina +
        0.1  * City=Mercer Island,Medina +
        0.36 * City=Medina +
       11.24
```

K-nearest neighbor

K-nearest neighbor is an instance-based learning algorithm which uses instances themselves to represent what is learned, rather than inferring a rule set or decision tree and storing it instead.

In KNN, each new instance is compared with existing ones using a distance metric, and the closest existing instance is used to assign the class to the new one. Sometimes more than one nearest neighbor is used, and the majority class of the closest k neighbors (or the distance weighted average if the class is numeric) is assigned to the new instance.

The distance between the instances is calculated using Euclidean distance. The distance between an instance with attribute values a_1^1, a_2^1, a_3^1,……, a_k^1 and another instance with values a_1^2, a_2^2, a_3^2,……, a_k^2 is defined as:

$$\sqrt{(a_1^1 - a_1^2)^2 + (a_2^1 - a_2^2)^2 + \cdots + (a_k^1 - a_k^2)^2}$$

In Weka I have used IBk schema to build K-nearest neighbor model. Below are the metrics for different K value.

	K = 1	K = 5	K = 7	K= 8	K = 10
Correlation coefficient	0.8002	0.8578	0.8616	0.8624	0.863
Mean absolute error	0.2374	0.1977	0.1958	0.1956	0.1959
Root mean-squared error	0.331	0.2724	0.2691	0.2686	0.2684

Relative absolute error	56.9881%	47.448%	47.0004%	46.9514%	47.0205%
Root relative squared error	62.535%	51.4662%	50.8461%	50.7449%	50.6993%

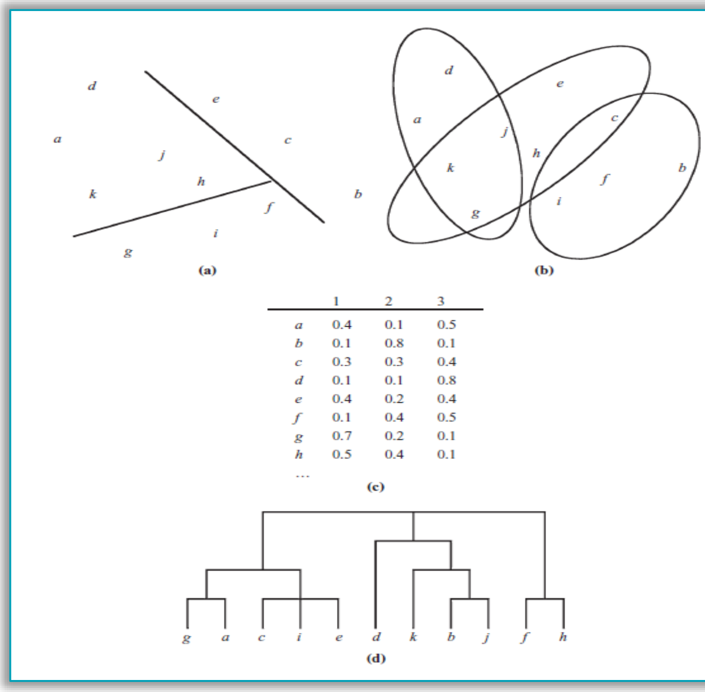
Based on the above table K = 8 seems to be the best. It has a good correlation coefficient and smallest value for each error measure.

K-means clustering

k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

First, cluster size is mentioned(K). Then each instance is added to the cluster based on Euclidian distance. Next, the mean or centroid is calculated for the clusters which would be the new centers of the clusters. Then the process is repeated until the means of clusters remains constant.

Once the iteration has stabilized, each point is assigned to its nearest cluster center, so the overall effect is to minimize the total squared distance from all points to their cluster centers. But the minimum is the local one, not the global as it depends on K value. To increase the chance of finding a global minimum people often run the algorithm several times with different initial choices and choose the best final result—the one with the smallest total squared distance. The diagram represents the clusters in different forms.



I have used “SimpleKMeans” schema in Weka for cluster analysis. By applying the model for different values of k, I found that k=5 segregates the instances quite evenly.

Results

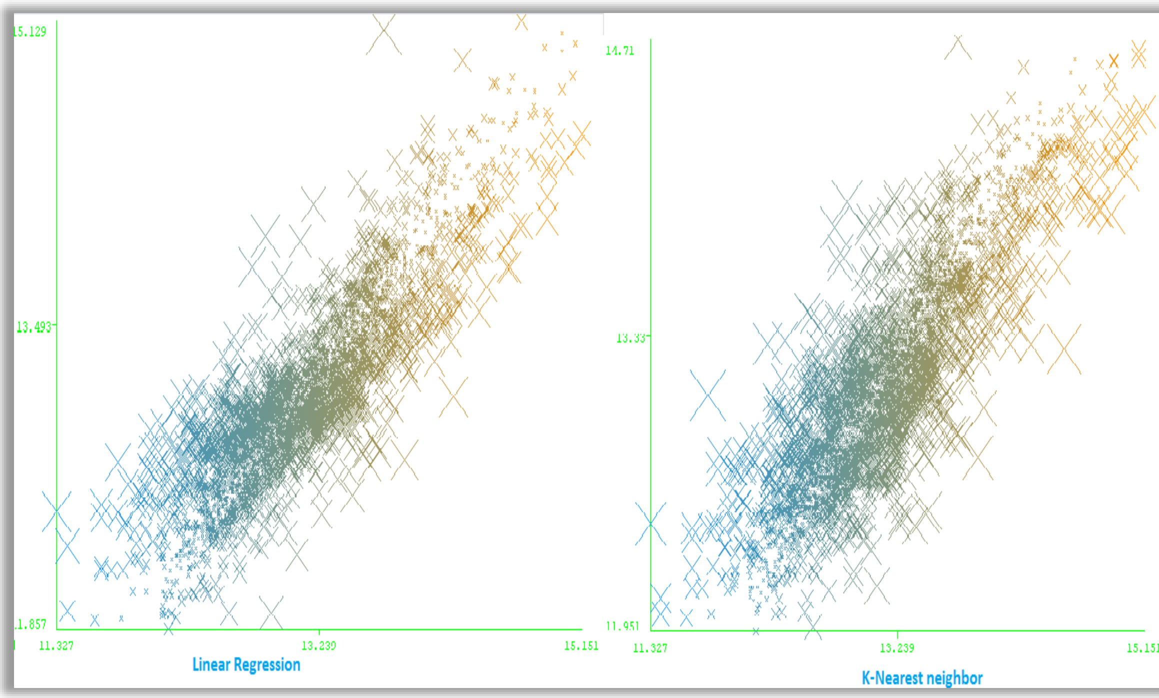
After building the model on the training data, I have used test data to predict the attribute "price".

Below are the metrics obtained for Linear regression and nearest neighbor with k = 8.

	Linear Regression	KNN
Correlation coefficient	0.8903	0.8578
Mean absolute error	0.1727	0.1977
Root mean-squared error	0.2337	0.2724
Relative absolute error	43.1505 %	47.448%
Root relative squared error	45.1417 %	51.4662%

Based on the above metrics, linear regression model outperforms KNN, by having higher correlation and smaller error metrics.

Below are the graph showing actual vs predicted values of price for both the models.



We can see that the KNN graph is wide when compared to the Linear regression model.

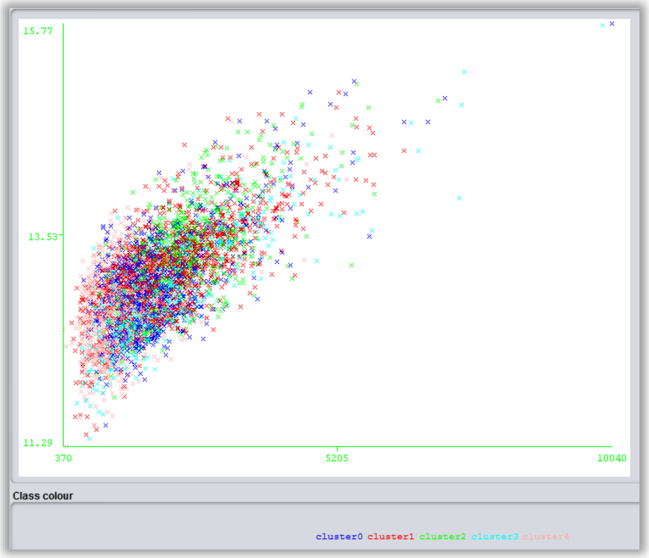
Below are the results of some of the records having actual, predicted and error of price converted in dollars.

Actual($)	Predicted(LR)	Predicted(KNN)	Error(LR)	Error(KNN)
380028	589482	515040	-209454	-135012
624683	589482	515040	35201	109643
379648	485046	403124	-105398	-23476
340102	417483	438450	-77381	-98348
515555	465096	522824	50459	-7269
600189	571489	546342	28700	53847
399912	231886	204843	168026	195069
364033	287506	314582	76527	49451
440207	398714	400312	41493	39895
284930	244752	252963	40178	31967
669308	514525	581287	154783	88021
474967	431059	391601	43908	83366
747134	458172	530195	288962	216939
317109	272938	334703	44171	-17594
390038	313640	261712	76398	128326
1069819	651479	692456	418340	377363
1209842	729416	1022744	480426	187098
580126	560172	475442	19954	104684
518140	631593	588305	-113453	-70165

Through the cluster analysis using simple k-means algorithm, I found that k=5 i.e. 5 clusters seems optimum and distributes the data pretty much evenly. Below are the cluster results:

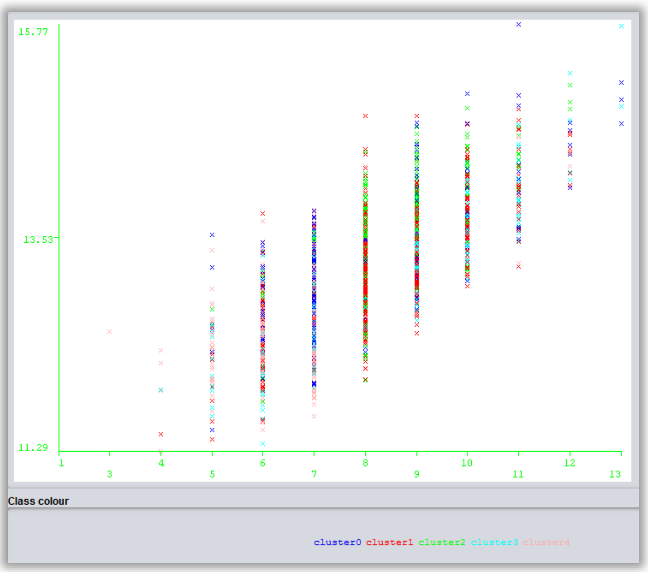
```
Final cluster centroids:
                                      Cluster#
Attribute                 Full Data          0            1            2            3            4
                          (17290.0)   (3527.0)     (5127.0)     (2351.0)     (2472.0)     (3813.0)
=====================================================================================================
bedrooms                     3.3728     3.4352       3.4328       3.7639       3.4013       2.9748
bathrooms                    2.1124     2.0568       2.4705       2.3619       2.2298       1.4522
sqft_living               2079.7926  2026.0037    2318.9945    2553.3369    2174.1561    1454.7613
sqft_lot                 15011.8397  11919.224   13372.8172   17209.2446   22344.7265   13967.4988
floors                       1.4964     1.1902       1.9568       1.2886         1.68       1.1694
waterfront                        0          0            0            0            0            0
view                              0          0            0            0            0            0
condition                         3          3            3            4            3            4
grade                             7          7            8            8            7            7
sqft_above                 1789.561  1375.1296    2218.5444    1777.4504    2125.8107    1385.5652
sqft_basement_binarized           0          1            0            1            0            0
yr_built             '(2003.5-inf)' '(1980.5-1992]' '(2003.5-inf)' '(1969-1980.5]' '(1992-2003.5]' '(1946-1957.5]'
yr_renovated_binarized            0          0            0            0            0            0
City                        Seattle    Seattle      Seattle      Seattle         Kent      Seattle
Log(Price)                  13.0469    13.0228      13.1761      13.3508      12.9264      12.7864
```

In this graph, we can see that there seems to be a positive relationship between variables price and sqft_living. Cluster#4 with houses having the lower living area, built in 1946-1957 and not renovated seems to have a lower price. Seems like Cluster#2 has the houses having a bigger area and are renovated, hence having a higher price.



Although we could not make any relationship of the price with the zip code, we can see that Seattle has houses with the complete range of price.

We can see from the below graph that price seems to increases with the increase in the grade of the property. Cluster#0 seems to have maximum houses with grade 7. Cluster#1 and Cluster#2 have houses with grade 8.



Conclusion

Below is the conclusion of the project:

- Linear regression model works well with predicting the price of the property; however, the error margin doesn't seem to be too far from KNN.

- For k=5, clusters seem to segregate the data well.

- There seems to be a positive relationship with the living area and price. Bigger the living area higher will be the price.
- There seems to be a positive relationship with the grade and price. Higher the grade, higher will be the price.

References

https://www.kaggle.com/harlfoxem/housesalesprediction

https://en.wikipedia.org/wiki/Linear_regression

https://en.wikipedia.org/wiki/K-means_clustering

https://www.kaggle.com/auygur/step-by-step-house-price-prediction-r-2-0-77

Data Mining, third edition by Ian Witten and Frank Eibe. Errata