House Price Prediction – Regression & Clustering

Introduction to Data Science – CS 418

Final Project

Submitted by:

Yukthi Papanna Suresh

Emily Lin

Caglar Kurtkaya

Problem Selection:

House Price Prediction of King county the most populous county in Washington.

Useful for prediction of real-estate prices which fluctuate every year. Our Aim would be to create a regression model to predict the house prices of King County based on the feature-set and estimate the price of a house based on its location using clustering techniques.

Data Collection:

The dataset can be found in Kaggle under the name **House sales on king county.**

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

Number of Features**: 21**

Number of observations: **21600**

Data Description:

S.No.	Attribute	Description
1	id	A notation for a house
2	date	Date house was sold
3	price	Price is prediction target
4	bedrooms	Number of Bedrooms/House
5	sqft_living	Square footage of the home
6	sqft_lot	Square footage of the lot

7	floors	Total floors (levels) in house
8	waterfront	House which has a view to a waterfront
9	view	Has been viewed
10	condition	How good the condition is (Overall)
11	grade	overall grade given to the housing unit, based on King County grading system
12	sqft_above	square footage of house apart from basement
13	sqft_basement	square footage of the basement
14	yr_built	Built Year
15	yr_renovated	Year when house was renovated
16	zip-code	zip
17	lat	Latitude coordinate
18	long	Longitude coordinate
19	sqft_living15	Living room area in 2015(implies-- some renovations) This might or might not have affected the lot-size area
20	lot-size area	sqft_lot15
21	sqft_lot15	lot-size area in 2015(implies-- some renovations)

Data-Preparation:

1. Trimmed the date format to year (noise):
   Ex: 20141013T000000 to 2014

2. Explore Missing values: (missing data)

```
o  id                 21613 non-null int64
o  date               21613 non-null object
o  price              21613 non-null float64
o  bedrooms           21613 non-null int64
o  bathrooms          21613 non-null float64
o  sqft_living        21613 non-null int64
o  sqft_lot           21613 non-null int64
o  floors             21613 non-null float64
o  waterfront         21613 non-null int64
o  view               21613 non-null int64
o  condition          21613 non-null int64
o  grade              21613 non-null int64
o  sqft_above         21613 non-null int64
o  sqft_basement      21613 non-null int64
o  yr_built           21613 non-null int64
o  yr_renovated       21613 non-null int64
o  zipcode            21613 non-null int64
o  lat                21613 non-null float64
o  long               21613 non-null float64
o  sqft_living15      21613 non-null int64
o  sqft_lot15         21613 non-null int64
```

   The dataset does not contain any missing values

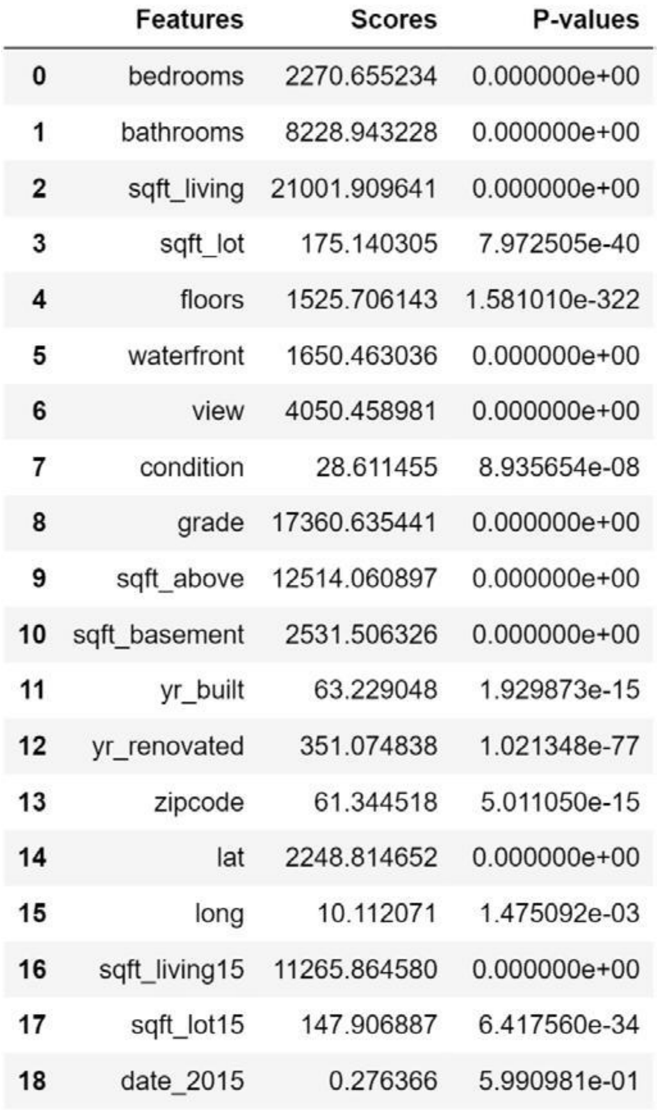
3. Created a dummy variable for year as “date_2015” with 2014 as 0 and 2015 as 1.

4. Removed outlier : (outliers)
   - During Exploratory data-analysis we found that there was a house with 33 rooms for just 100000. Clearly that was an outlier, so we removed it.

Exploratory Data-Analysis:

1. **Feature Selection**: The **SelectKBest** class just scores the features using a function (in this case **f_regression** but could be others) and then "removes all but the k highest scoring features".

   Ran Select K Best function on all the features. The results are as below:



	Features	Scores	P-values
0	bedrooms	2270.655234	0.000000e+00
1	bathrooms	8228.943228	0.000000e+00
2	sqft_living	21001.909641	0.000000e+00
3	sqft_lot	175.140305	7.972505e-40
4	floors	1525.706143	1.581010e-322
5	waterfront	1650.463036	0.000000e+00
6	view	4050.458981	0.000000e+00
7	condition	28.611455	8.935654e-08
8	grade	17360.635441	0.000000e+00
9	sqft_above	12514.060897	0.000000e+00
10	sqft_basement	2531.506326	0.000000e+00
11	yr_built	63.229048	1.929873e-15
12	yr_renovated	351.074838	1.021348e-77
13	zipcode	61.344518	5.011050e-15
14	lat	2248.814652	0.000000e+00
15	long	10.112071	1.475092e-03
16	sqft_living15	11265.864580	0.000000e+00
17	sqft_lot15	147.906887	6.417560e-34
18	date_2015	0.276366	5.990981e-01

Clearly from the above table all the scores are statistically significant **except date_2015** assuming **alpha = 0.01.**

Hence selecting the features with high scores, we get the below list.

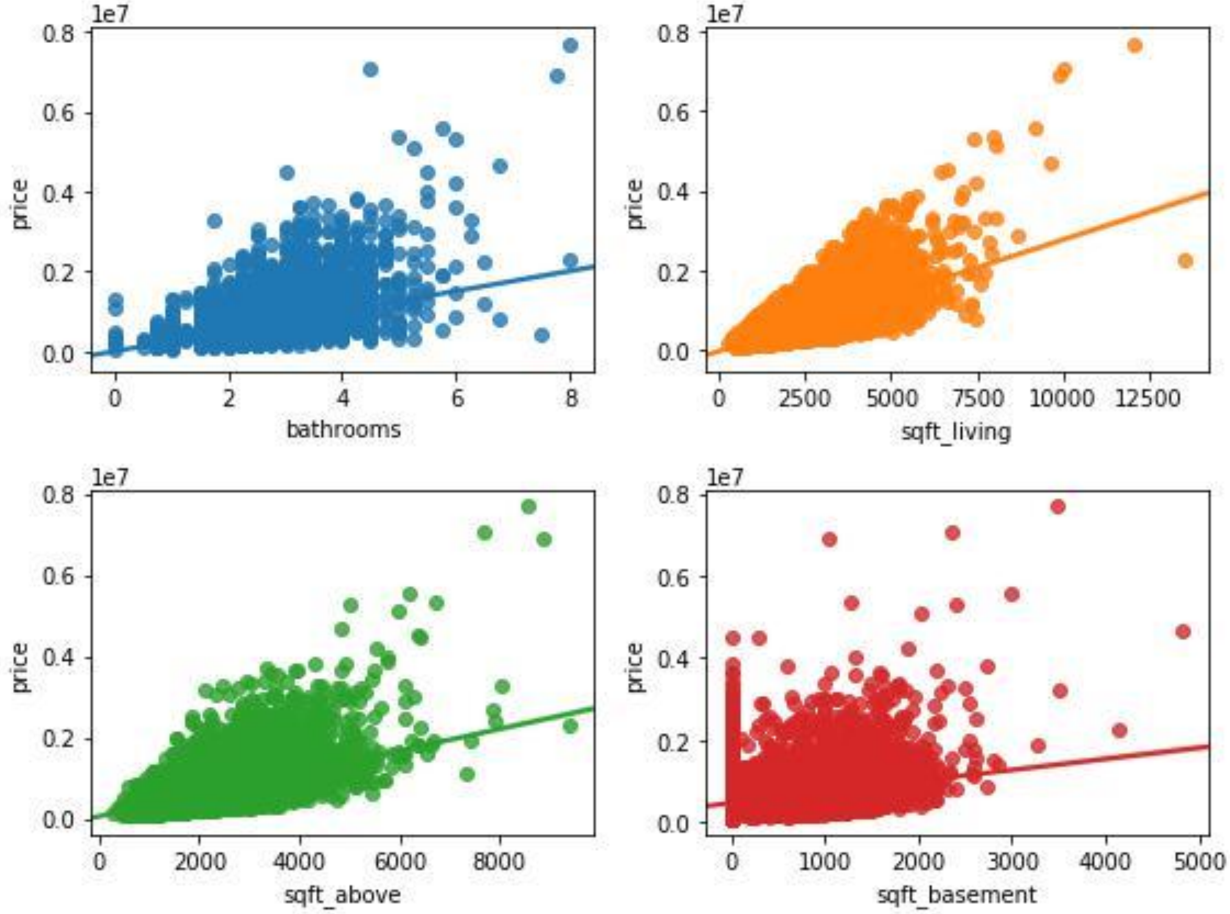
List of Important Features:

['bedrooms', 'bathrooms', 'sqft_living', 'floors', 'waterfront', 'view', 'grade','sqft_above', 'sqft_basement', 'yr_renovated', 'lat', 'sqft_living15']

2. **Price Info:** Price statistics are as shown below

```
count    2.161300e+04
mean     5.400881e+05
std      3.671272e+05
min      7.500000e+04
25%      3.219500e+05
50%      4.500000e+05
75%      6.450000e+05
max      7.700000e+06
```

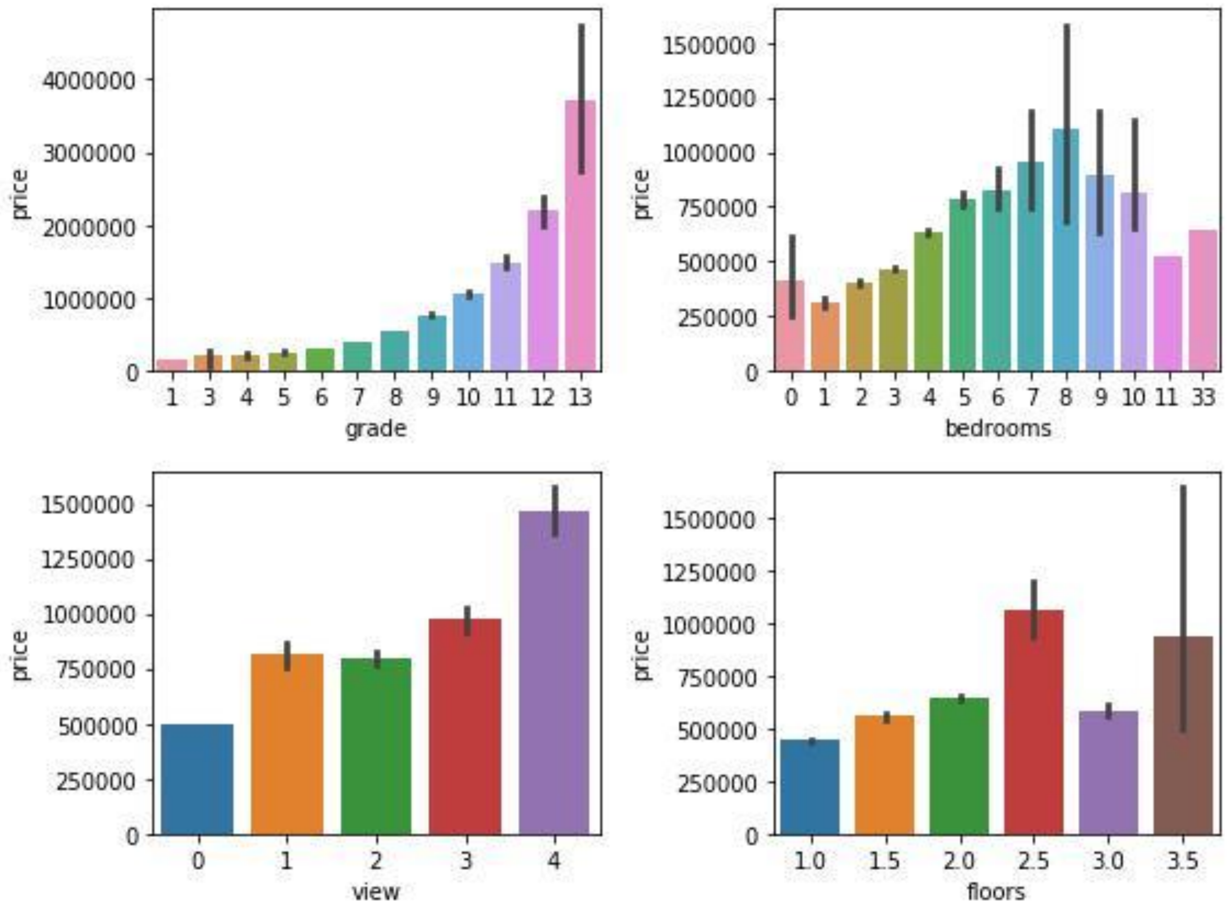
3. **Plots (Linear Regression)**



The above figure shows that the price of the houses is linearly dependent on the above numerical features and they fit the data well.

- The price of house increases with the increase in values of all features above.

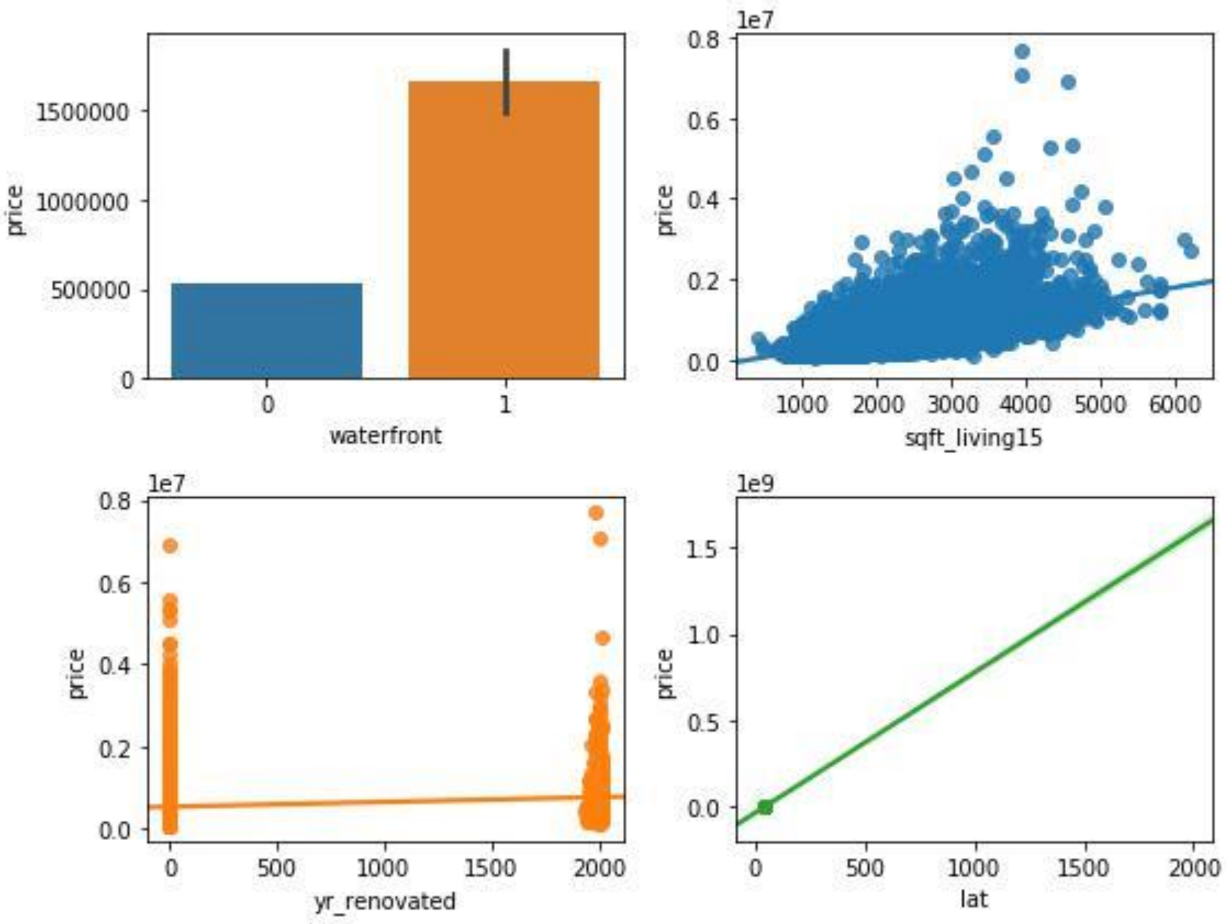
4. **Categorical (Bar plots)**



- The above shows average price of houses for different categorical features.
- The average price of house increases with increase in **grade** of the house.
- The average price of house with 8 **bedrooms** is the highest as there is a house with 33 rooms with very less price which suggests that that observation is an **outlier.**
- The average price of the house increases with the number of **views** present in the house

□ The average price of houses with 2.5 **floors** is highest which can also imply that the dataset contains more houses with 2.5 floors.

5. **Mixed Plots**



The above figure gives a mix of numerical and categorical regression and barplots of the remaining important features

□ The average price increases within presence of **waterfront**.

□ The price linearly increases with increase in **living space.**

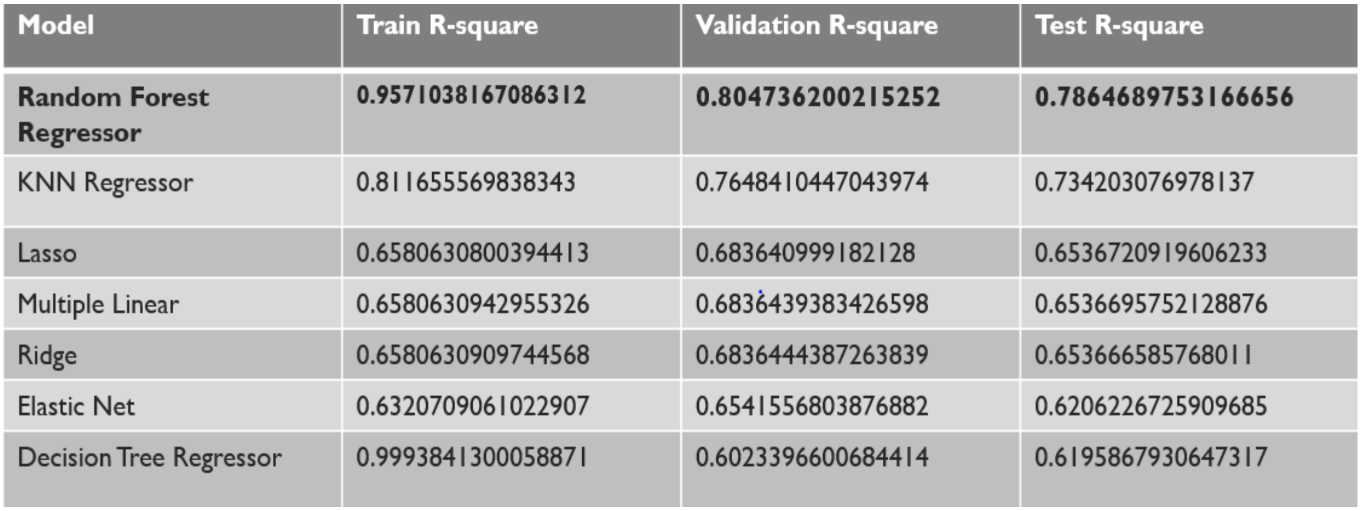
□ The houses are in the same **latitude** and hence the figure above.

Data Modelling:

- Divided the dataset into **train, test and validation sets.**
- **Used Standard Scaler** to fit the training set and transformed the train test and validation sets such that all the feature data have a **mean of zero and standard deviation of 1.**

Regression Models: (To predict house prices based on important features)

R_squared values with **Important features**



Model	Train R-square	Validation R-square	Test R-square
Random Forest Regressor	**0.9571038167086312**	**0.804736200215252**	**0.7864689753166656**
KNN Regressor	0.811655569838343	0.7648410447043974	0.734203076978137
Lasso	0.6580630800394413	0.683640999182128	0.6536720919606233
Multiple Linear	0.6580630942955326	0.6836439383426598	0.6536695752128876
Ridge	0.6580630909744568	0.6836444387263839	0.653666585768011
Elastic Net	0.6320709061022907	0.6541556803876882	0.6206226725909685
Decision Tree Regressor	0.9993841300058871	0.6023396600684414	0.6195867930647317

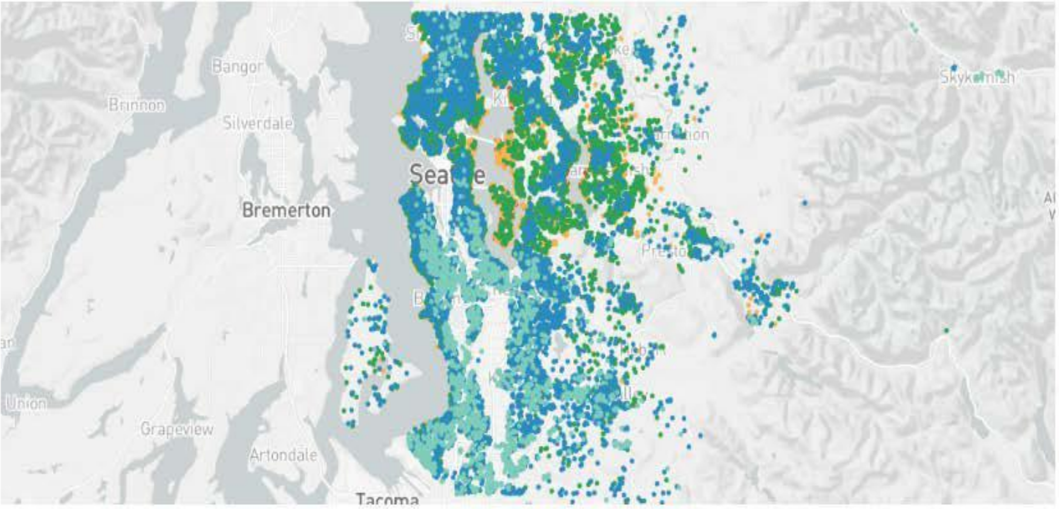
Clearly from above table we got **highest R squared** value on the test dataset using **Random Forest Regressor.**

Also got **highest adjusted R squared value**(0.8041145059734327) for **Random Forest Regressor.**

Conclusion: We got good regression models. Best regression model is given by Random Forest Regressor with explains almost 79% of variance in the observations.

Clustering Models:

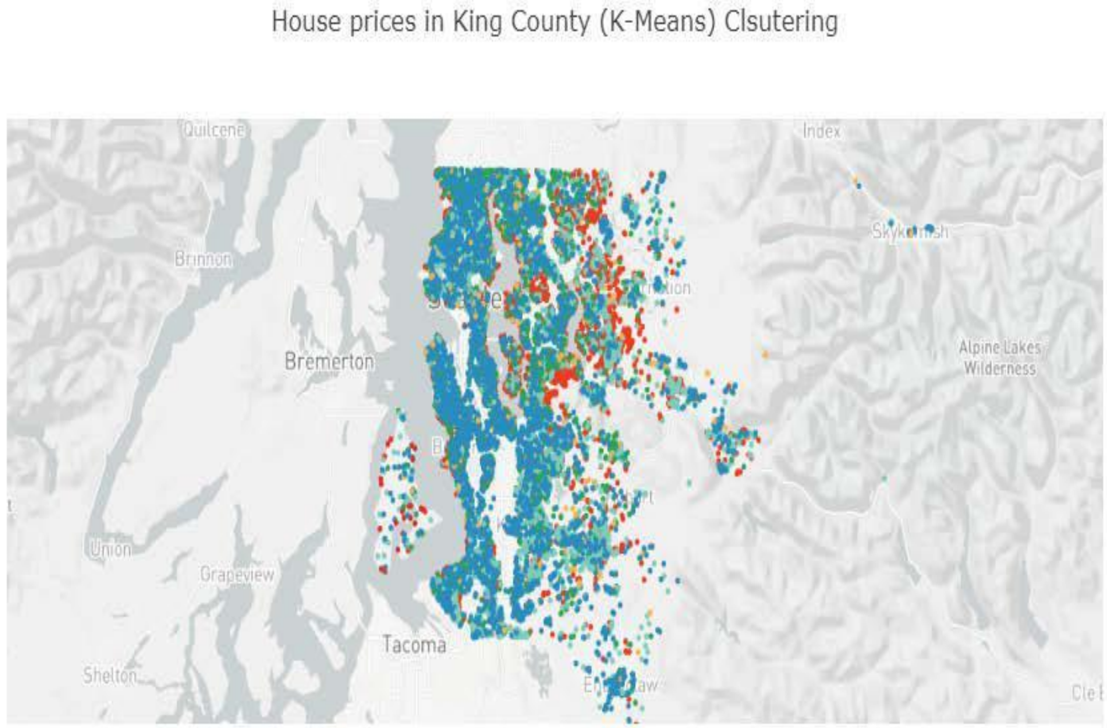
I. **Custom Cluster:** Using **longitude, latitude and price** of the house we gave color labels to **6 clusters** as following:
   - **a.** Price < 250000 – Light Blue
   - **b.** 250000<price<500000 – Dark Blue
   - **c.** 500000<price<1000000 – Green
   - **d.** 1000000<price<3000000 – Yellow
   - **e.** 3000000<price<5000000 – Light Red
   - **f.** Price>5000000 – Dark Red


House prices in King County (Custom Price Clustering)

Clearly The pricier houses lie at the center and least expensive houses lie on the south side.

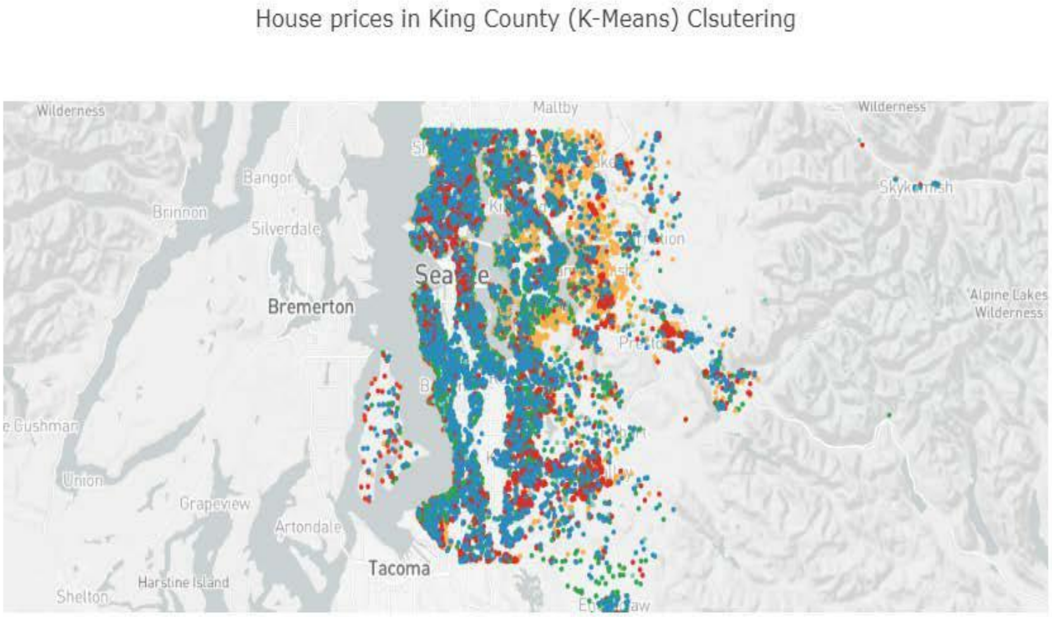
(See Jupyter Code map for a **zoomed** in view)

II. **K-Means Cluster:** Created 6 Clusters using K-means clustering on the important features. Below is the clustering figure.



Clearly the clusters are different from the custom clusters and clusters regions randomly.

III. **Ward's Linkage:** Created 6 Clusters using Ward's Linkage clustering on the important features. Below is the clustering figure.



Clearly ward's linkage also forms clusters randomly with respect to the desired clusters.

Conclusion: Bad Clustering Models due to the no clear trend in clusters of data points.