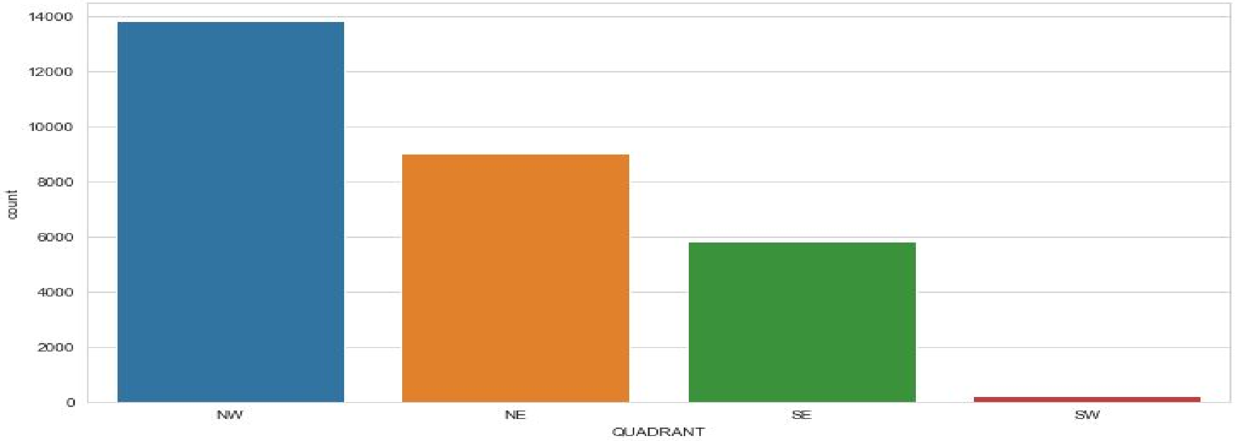
D.C Property Project

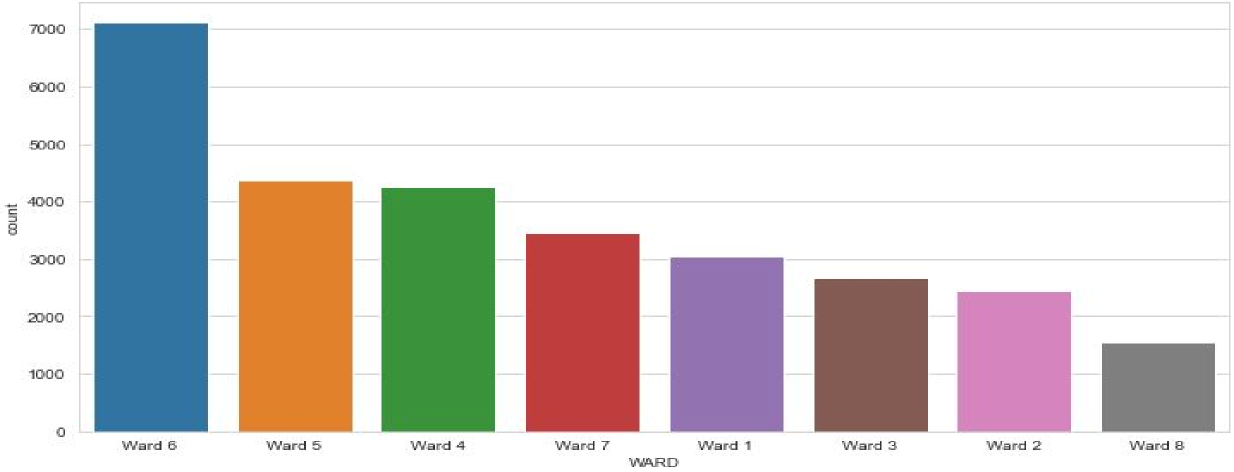
Business Problems:

1. Provide price guidelines for people who want to sell or buy properties in D.C area based on related features.
2. Understanding key factors affecting housing price.
3. How to know about the condition of housing without seeing it’s picture?

Exploratory Data Analysis:

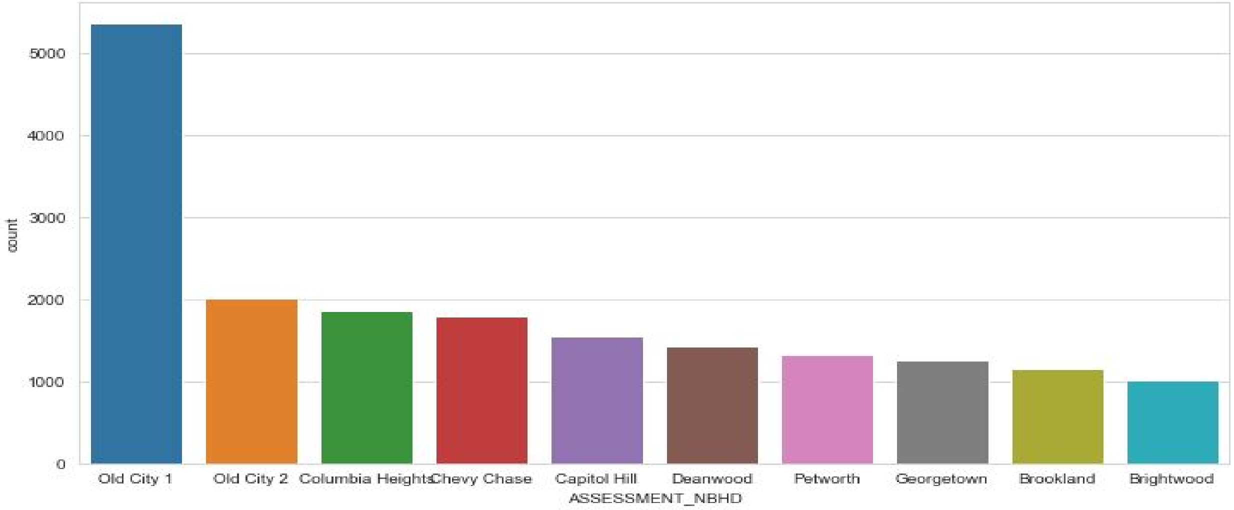
The dataset contains 28900 entries of 46 variables. The D.C area is divided into 4 **Quadrants** and each **Quadrant** further divided into 8 **Ward**.



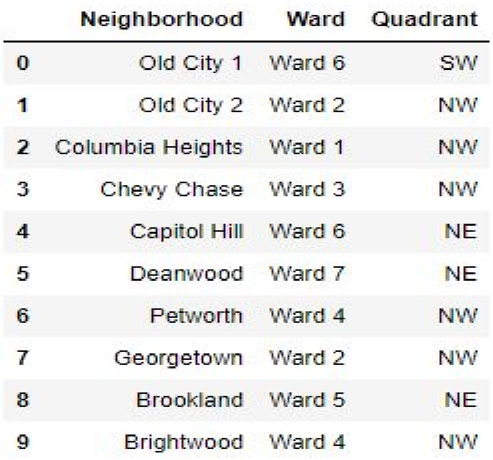


From the above two graphs we can see that **Quandant NW(Ward 6)** has the most representation on the dataset and **Quadrant SW(Ward 8)** has least representation.

Further the area divided into **33 Neighborhoods** each Neighborhoods divided into **120 Sub -Neighborhoods.**

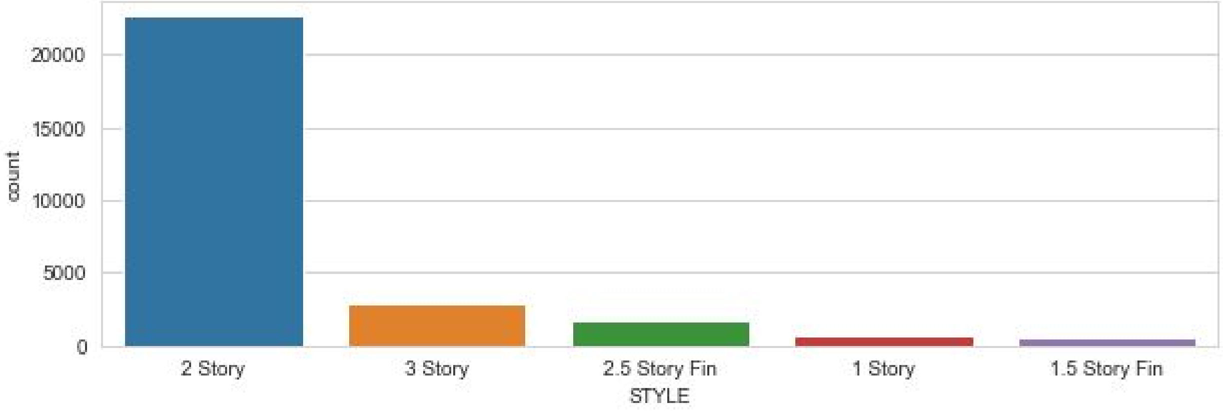


From these two graphs we can understand that **Old City-1** in **Ward 6 of SW Quadrant** is the most dominant **Neighborhood** in the dataset. For simple models we are not considering Sub-Neighborhood, Full Address, Zip Code, Latitude, Longitude of those properties.

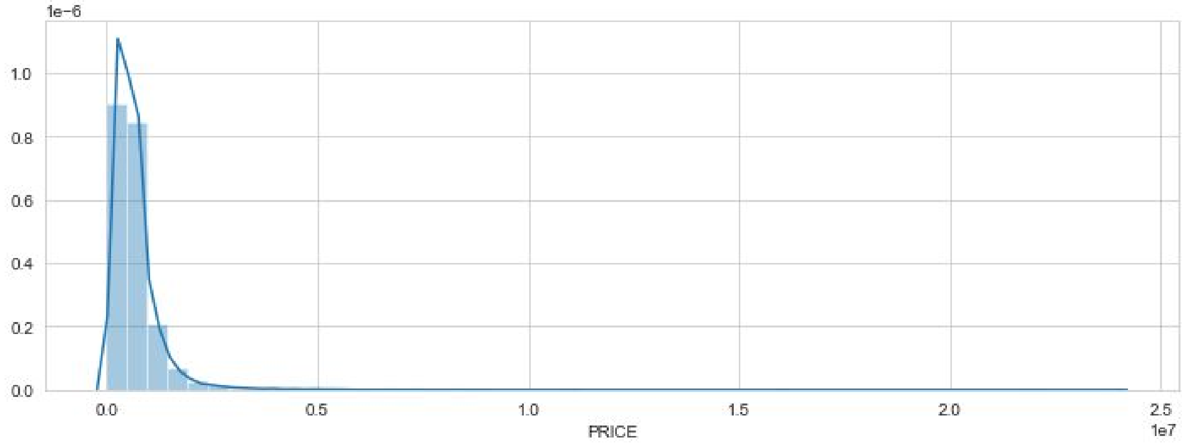


	Neighborhood	Ward	Quadrant
0	Old City 1	Ward 6	SW
1	Old City 2	Ward 2	NW
2	Columbia Heights	Ward 1	NW
3	Chevy Chase	Ward 3	NW
4	Capitol Hill	Ward 6	NE
5	Deanwood	Ward 7	NE
6	Petworth	Ward 4	NW
7	Georgetown	Ward 2	NW
8	Brookland	Ward 5	NE
9	Brightwood	Ward 4	NW

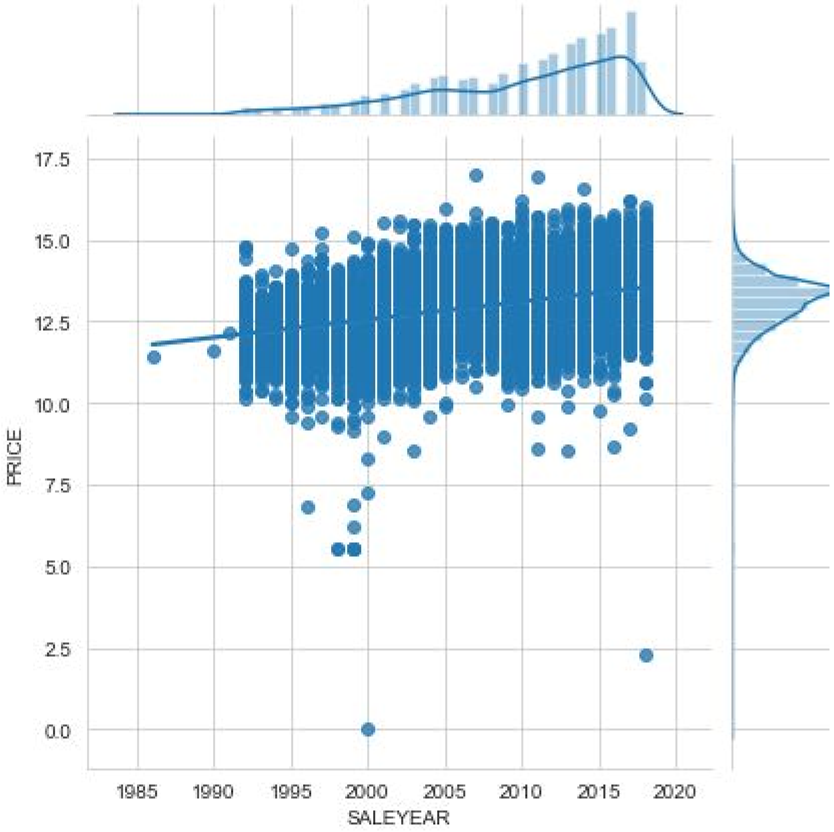
Now, residential property further categorized into 17
Style, out of which **2 Story** and **3 Story** house dominates (See the graph below)



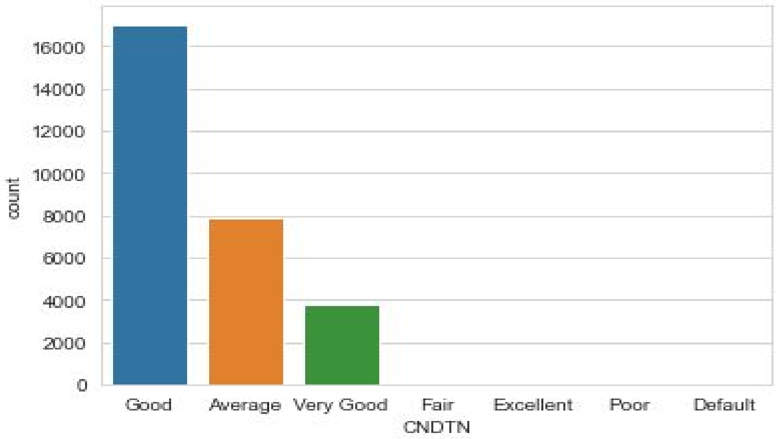
Now we will explore the price columns. From the below graph we can understand that it is highly biased. So, we will transform our target with Log Transformation:



Next, for the **Sale Date** there are 5678 unique dates, to make model simple, I extracted only Year and then compare with Price for clear guideline:



Next, for our second business objective: the **Condition** column consists of 7 labels out of which only **Good Condition** is the majority. So, I will map all other conditions in one separate class.



Data Cleaning and Preprocessing:

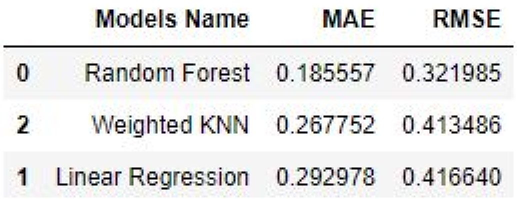
1. **Removed redundant columns**
   - I have removed 20 columns unnecessary columns
2. **Check Missing Value**
   - This is no missing value in data using df.isnull().sum()
3. **Log Transformation**
   - I have used Log Transformation to Price to transform because it is highly skewed and have outliers
4. **Label Encoding**
   - Use LabelEncoder() for categorical columns for new labels and use manual mapping to re-label **Condition** variable
5. **Normalization**
   - Used MinMaxScaler() to normalize all variables to same range

Feature Engineering:

1. Created **Sale Year** out of **Sale Date** column for easy understanding of housing price trend

Modelling:

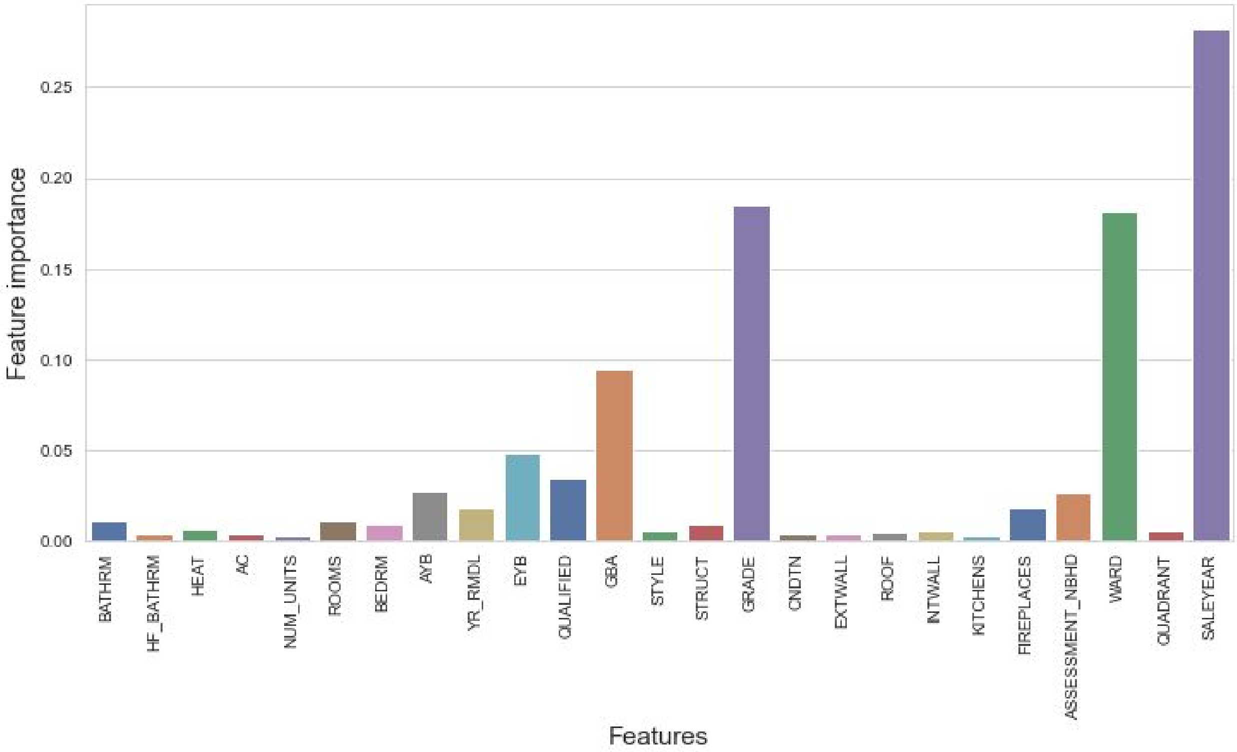
1. Created **Feature** data frame consists of all variables except our Target i.e., **Price**
2. Split both Target and Feature data frame into 80% training set and 20% testing set
3. Create a pipeline of three machine learning models (Linear Regression, Random Forest Regression and Weighted KNN) and report their respective Mean Absolute Error, Root Mean Squared Error on the test set.
4. Comparing four models, I have chosen Random Forest Regression as my baseline model because it has lowest RMSE of 0.321985 (See the report below)



	Models Name	MAE	RMSE
0	Random Forest	0.185557	0.321985
2	Weighted KNN	0.267752	0.413486
1	Linear Regression	0.292978	0.416640

Feature Importance:

1. Created a graph of **Feature Importance** of all variables to have better understanding of factors affecting Price
2. It's turn out that **SALEYEAR**, **Ward** and **Grad** have most importance and **Heat**, **Num_Units**, **Kitchens, INTWALL, AC, EXTWALL** have least importance
3. I have then run my baseline model after removing those least important variables but the RMSE value didn't change a lot.



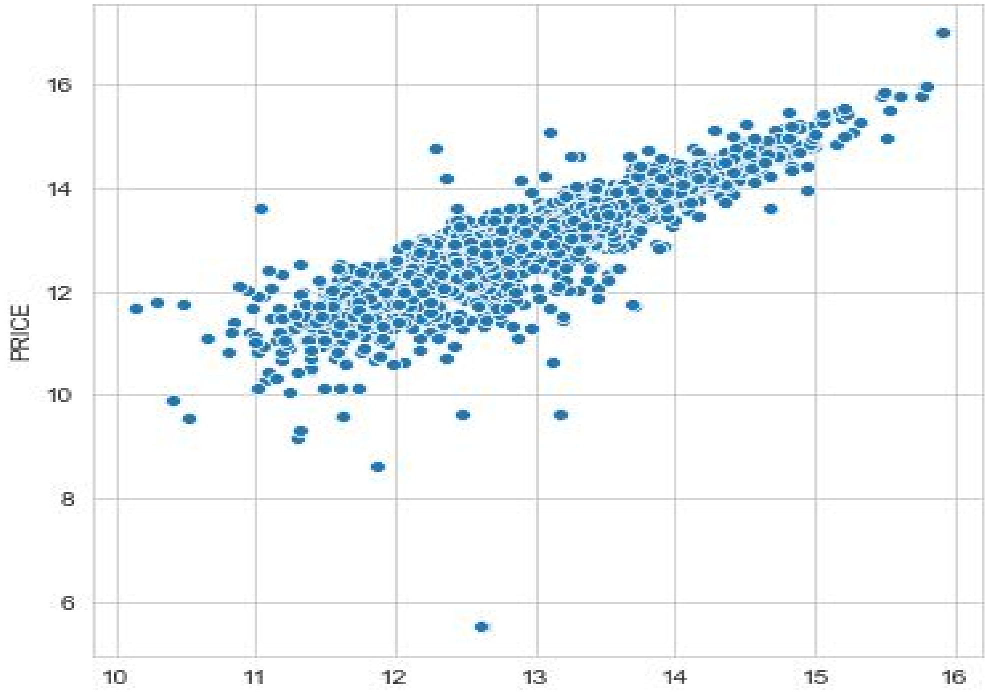
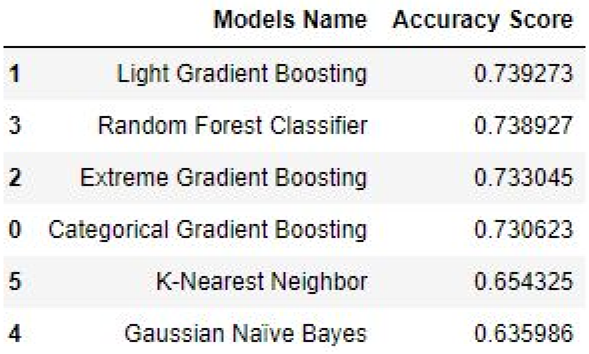


Fig: Predicted Price vs. Test set Price

Modelling for Condition:

The third business problem is to predict whether the house is in Good condition or not without looking at the picture of the house. So, it's a binary classification problem.

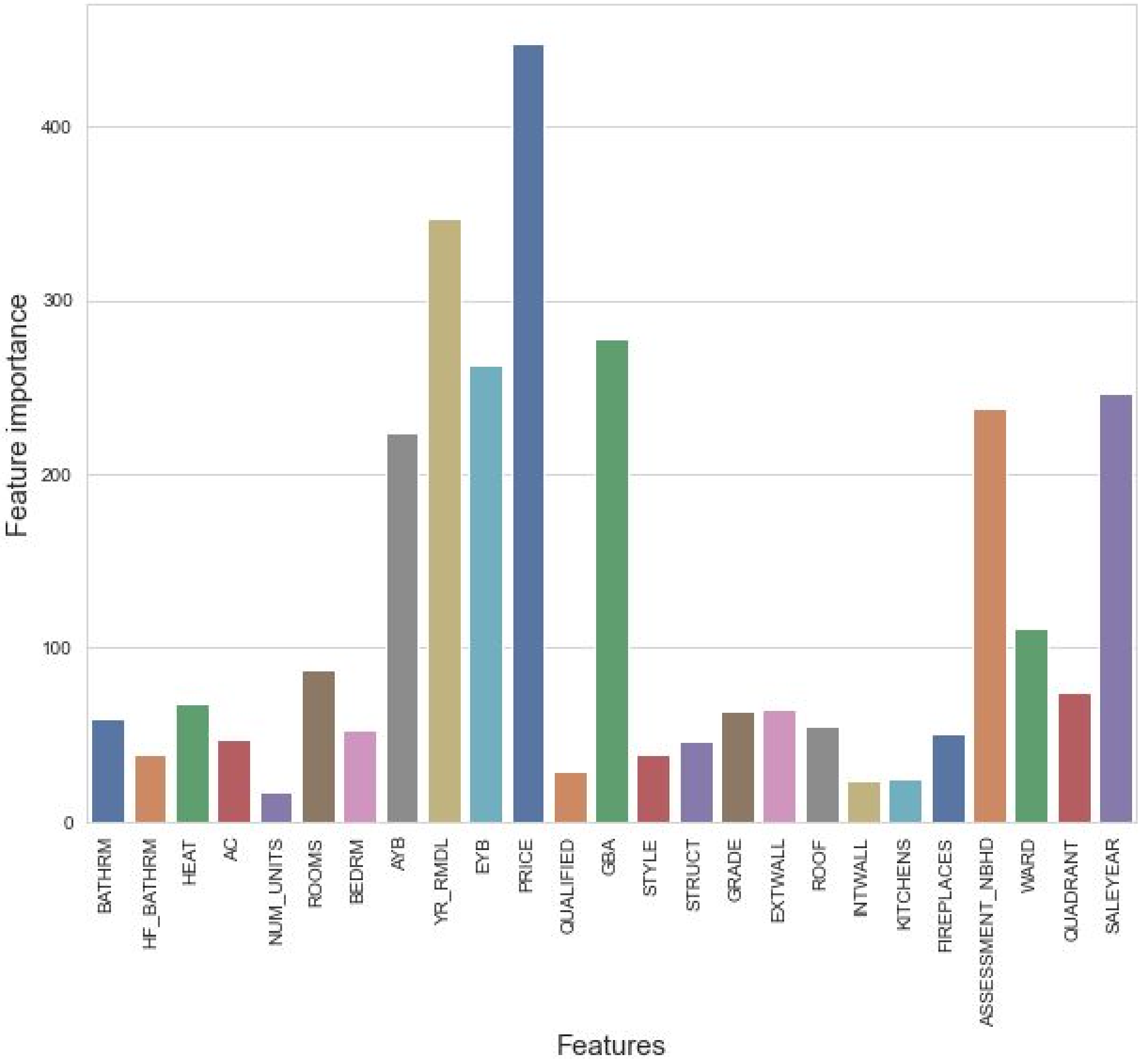
1. In this case, I have used **Condition** as the target variable and training and testing set ratio remains same as before.
2. Create a pipeline of six machine learning models (Categorical Gradient Boosting, Light Gradient Boosting, Extreme Gradient Boosting, Random Forest Classifier, Gaussian Naïve Bayes, K-Nearest Neighbor) and report their respective **Accuracy Score** on the test set.
3. Comparing four models, I have chosen Random Forest Regression as my baseline model because it has highest **Accuracy Score** of 0.739273 (See the report below)



	Models Name	Accuracy Score
1	Light Gradient Boosting	0.739273
3	Random Forest Classifier	0.738927
2	Extreme Gradient Boosting	0.733045
0	Categorical Gradient Boosting	0.730623
5	K-Nearest Neighbor	0.654325
4	Gaussian Naïve Bayes	0.635986

Feature Importance:

4. Created a graph of **Feature Importance** of all variables to have better understanding of factors affecting Condition
5. It's turn out that **Price**, **Year of Remodel** and **Gross Building Area** have most importance and **Num_Units**, **Kitchens,** and **INTWALL** have least importance



Conclusion: Both Price and Condition can be predicted with higher accuracy. Imbalance of data over some variables (like Ward, Neighborhood) can be a factor affecting model accuracy.