HOUSING PRICE PREDICTIONS

Submitted by:

Deepthi Prakashan

ACKNOWLEDGEMENT

I would like to sincerely thank Flip Robo Technologies for the opportunity to work in the project and the timely guidance provided.

I express my gratitude towards Data Trained for the training they have provided overtime and for the guidance and support.

I express my heartfelt support to my parents and friends who were constantly by my side providing me with motivation and inspiration to work through the completion of this project.

INTRODUCTION

Housing is one of the most basic needs of human being and real estate market is a major contributor in satisfying this basic need. It is a very large market and there are various companies working in the domain.

Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies.

1.1 Business goal

House prices are affected by various factors and understanding the trend added to the knowledge of the variable that affects the housing prices is essential in the real estate market to strategize or decide whether to make an investment. Machine learning hence simplifies this decision making by building a predictive model based on various features specific for that region that may affect

the price. Further, the model also helps the management to understand the pricing dynamics of a new market.

1.2 Conceptual Background

A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The data is provided in the CSV file below. The company is looking at prospective properties to buy houses to enter the market. You are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:

- Which variables are important to predict the price of variable?
- How do these variables describe the price of the house?

1.3 Review of Literature

Real estate market is a booming area that contributes largely to the economy. In the housing market it involves acquiring prospective houses, which can will profit on reselling. However, there are many attributes that can influence the price of a House. Making accurate price prediction hence increases the efficiency of the business, thus giving the investors to make well planned decision. Such problems have been approached through various traditional machine learning models such as SVM, Linear Regression, Random Forest etc. In [1] we can see how multiple linear regression efficiently evaluates the house price prediction and performs better than other algorithms like decision tree classifier and decision tree regressor. In paper [2], they use SVM to evaluate the housing price and uses PSO to optimise the parameters. This approach is advantageous to overcome non-linear relationship. In [3]. The authors adopt a different approach by involving a combination of two machine learning algorithms which adopted the Stacked Generalization technique. Such researches are intended to not just assist the investors, but it also gives insight into the growth in a country's economy as well.

[1]Thamarai, M. and Malarvizhi, S.P., 2020. House Price Prediction Modeling Using Machine Learning. *International Journal of Information Engineering & Electronic Business*, *12*(2).

[2] Wang, X., Wen, J., Zhang, Y. and Wang, Y., 2014. Real estate price forecasting based on SVM optimized by PSO. *Optik*, *125*(3), pp.1439-1443.

[3]Truong, Q., Nguyen, M., Dang, H. and Mei, B., 2020. Housing price prediction via improved machine learning techniques. *Procedia Computer Science*, *174*, pp.433-442.

Analytical Problem Framing

2.1 Mathematical/ Analytical Modeling of the Problem

To get an insight, analyse and draw conclusions from the data, statistical tools are essential. Here in this project, we have dissected the data using many such tools which helped in data cleaning, understanding the relationships between the variables that dictated the sale price of the houses and used algorithms to build a predictive model. The following have to used for understanding our data at hand:

Descriptive statistics:

isnull().sum() function is used for computing the null values present in each columns of the data

nunique() function helped in giving unique values in each columns, which is especially useful for categorical data. Also, it vaguely tells us the variability present in data.

value_counts() function gave the count of each categories present in each column. This function thus helped in determining if there are unnecessary or duplicate categories in each column

The describe() function computes the summary of the statistics the measures of the central tendency (mean, median and mode), the measure of variability(standard deviation, quartiles), the count, minimum and maximum values in each column.

Correlation:

Correlation between the features and the feature and label was analysed using the corr() function.

Distribution of the data:

It was checked to see if the data is normally distribution was present or if the data is skewed

Outliers:

The outliers were checked using boxplot.

2.2 Data Sources and their formats

- The data has been collected by a US based housing company from sale houses in Australia.
- The data is provided to us in the CSV file format.
- Data contains 1168 entries with 81 variables.
- Sales Price is the target column rest 80 being the features.
- It contains both numerical and categorical columns
- We have been provided with a test csv file separately to predict the values using the final model.
- In the test data set there are 292 entries and 80 columns

2.3 Data Pre-processing

Reducing data

Id and Utility columns we dropped. Utility contained only 1 value whereas Id had all unique values.

These features had above 90% of the values as zero: MiscVal(96.404110),

PoolArea(99.400685),

ScreenPorch(91.866438),

3SsnPorch(98.116438)

LowQualFinSF(98.030822),

BsmtHalfBath(94.6061)

PoolArea(99.400685).

These columns were dropped.

Alley, PoolQC and MiscFeature has 90% above values missing. These columns were dropped as well.

Imputation of missing values:

Rest of the missing values which were in categorical columns were replaced by ‘None’ due to the absence of that attribute in the house.

Missing values found in 'MasVnrArea' was filled with 0.

kNN imputer was finally used in the remaining numeric column (‘GarageYrBlt’) to ensure the distribution remained almost the same.

Normalising data: PowerTransformer was used to normalise the skewness present in the data.

Outliers’ treatment: We removed 2% data from each end to remove the outliers.

Encoding categorical data: Label encoder was used to encode the data.

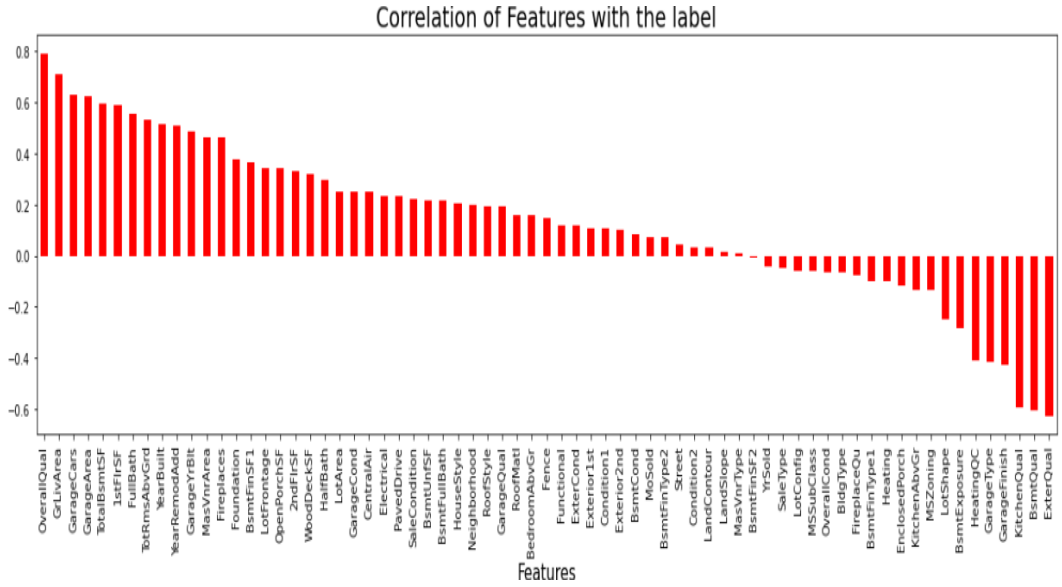
Scaling data: Scaling of the data was done by standard scaler

The data pre-processing step was then completed.

2.4 Data Inputs- Logic- Output Relationships

In order to decide the data inputs that are necessary to build our machine learning model, we are required to understand the pattern/relationship that exists between the label and features. For the same we have conducted an in-depth exploratory data analysis through techniques like correlation analysis and data visualization techniques that included barplot, regplots, catplot, lineplots and scatterplots.

Correlation Analysis was visualized through a heatmap and a bar plot, which expressed the features that had positive, negative and very less correlationship with the sale price of the house. It was observed that features like street, condition2, LandContour, LandSlope, MasVnrType, MSSubClass have least correlation with the target.



Strong positive correlation was shown by:

OverallQual	0.789185
GrLivArea	0.707300

Moderate positive correlation:

GarageCars	0.628329
GarageArea	0.619000
TotalBsmtSF	0.595042
1stFlrSF	0.587642
FullBath	0.554988
TotRmsAbvGrd	0.528363
YearBuilt	0.514408
YearRemodAdd	0.507831

Features with strong negative correlation:

BsmtQual	-0.601307
ExterQual	-0.624820

2.3 Hardware and Software Requirements and Tools Used

Hardware	

Processor	Dual Core or above
RAM	4GB or more
Cache	1MB or more
Hard Disk/SSD	180GB or more
Software	
OS	Window/Mac/Linux
Ide	Jupiter Notebook
Dataset	.csv file
Libraries	Pandas, numpy,matplotlib,seaborn, scikit.learn
Server	Web server with HTTP process

Model/s Development and Evaluation

With the pre-processed data, we proceeded with training the models. To do the same, the data was split into train and test dataset. Here, 80% of the data was used for trai ning and 20% for testing. Training the data allowed the models to learn the data and the underlying trend. Best ra ndom state was also checked, for which best r2 score was 0.90 at random state 8 (Using RandomForestRegressor), which was later used while training the models. Model's efficiency wa s then checked by predicting the house sales price from t he test dataset.

Evaluation metrics and cross-validation then helped us decide how good our prediction of house sale price was and if the predicted values weren't as a result of overfitting or underfitting.

The Algorithms that were used for training the models were :

- DecisionTreeRegressor,
- RandomForestRegressor,
- KNeighborsRegressor,
- GradientBoostingRegressor,
- AdaBoostRegressor
- Ridge Regressor.

Decision tree

DecisionTreeRegressor

```
dt = DecisionTreeRegressor(max_depth =22)
dt.fit(x_train,y_train)
preddt = dt.predict(x_test)

print('R2 Score',r2_score(y_test,preddt))
print('MAE',mean_squared_error(y_test,preddt))
print('MSE',mean_absolute_error(y_test,preddt))
print('RMSE',np.sqrt(mean_squared_error(y_test,preddt)))
print('Cross validation score',cross_val_score(dt,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,preddt)-cross_val_score(dt,x,y,cv=5).mean())*100)
```

```
R2 Score 0.7408676422594798
MAE 1484043414.2222223
MSE 25536.128205128207
RMSE 38523.2840529234
Cross validation score 0.6589066999875912
Difference in crossvalidation and R2 score 7.099356004214563
```

Decision tree is a non-parametric supervised machine learning model that solves both regression and classification tasks. In the case of a regression task, it splits the data such that it has a root node, which is considered the best splitting condition wherein on evaluation it gives maximum reduction in variance on splitting the data.

Here we set max_depth as 9, which restrict the trees to go only until 22. The result show the R2 score to be 71.51%. On applying evaluation metrics, we found MSE to be 26826.349 and, RMSE 40044.47. Cross validation score was 68.33%.

Random forest

RandomForestRegressor

```
rt = RandomForestRegressor()
rt.fit(x_train,y_train)
predrt = rt.predict(x_test)
```

```
print('R2 Score',r2_score(y_test,predrt))
print('MAE',mean_squared_error(y_test,predrt))
print('MSE',mean_absolute_error(y_test,predrt))
print('RMSE',np.sqrt(mean_squared_error(y_test,predrt)))
print('Cross validation score',cross_val_score(rt,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,predrt)-cross_val_score(rt,x,y,cv=5).mean())*100)
```

```
R2 Score 0.8849285812395178
MAE 659010640.9933602
MSE 15826.318076923078
RMSE 25671.20256227511
Cross validation score 0.8360519724772072
```

Random forest is a supervised machine learning algorithm that is used in solving both classification and regression problems. It is an ensemble learning technique, meaning, it creates multiple decision trees on different sample and obtain a better prediction by averaging over all the algorithms in the case of a RandomForestRegressor.

For RandomForestRegressor, we had a better R2 score of 90.19%. MSE 16243.53, RMSE 23491.89. On performing cross validation, it had a score of 83.60%

KNearestNeighbors

KNeighborsRegressor

```
knn = KNeighborsRegressor()
knn.fit(x_train,y_train)
predknn = knn.predict(x_test)

print('R2 Score',r2_score(y_test,predknn))
print('MAE',mean_squared_error(y_test,predknn))
print('MSE',mean_absolute_error(y_test,predknn))
print('RMSE',np.sqrt(mean_squared_error(y_test,predknn)))
print('Cross validation score',cross_val_score(knn,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,predknn)-cross_val_score(knn,x,y,cv=5).mean())*100)
```

```
R2 Score 0.8051363569316458
MAE 1115978369.8511112
MSE 22186.43675213675
RMSE 33406.26243462611
Cross validation score 0.7306367472297539
Difference in crossvalidation and R2 score 7.449960970189185
```

Knn is a non-parametric supervised machine learning model that predicts the target value by averaging the values of data points that are found in its close distance. This distance can be decided up that serves well for the dataset. This method is inefficient when the dimensionality of the dataset is big.

The model performed with an R2 score of 80.38%., MSE 22387.07, RMSE 33234.95. On performing cross validation, we the score as 73.06%

GradientBoostingRegressor

GradientBoostingRegressor

```
gbr = GradientBoostingRegressor()
gbr.fit(x_train,y_train)
predgbr = gbr.predict(x_test)

print('R2 Score',r2_score(y_test,predgbr))
print('MAE',mean_squared_error(y_test,predgbr))
print('MSE',mean_absolute_error(y_test,predgbr))
print('RMSE',np.sqrt(mean_squared_error(y_test,predgbr)))
print('Cross validation score',cross_val_score(gbr,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,predgbr)-cross_val_score(gbr,x,y,cv=5).mean())*100)
```

```
R2 Score 0.8854220537192234
MAE 656184538.5722169
MSE 15599.174902598133
RMSE 25616.09920679214
Cross validation score 0.8281507195263889
Difference in crossvalidation and R2 score 5.844172527830671
```

GradientBoostingRegressor is a non-parametric supervised machine learning model.It is an ensemble model that combines

multiple models and obtains the prediction. It is also called an additive model as it keeps adding weak learners such that we get a stronger model from the final model.

The model performed with an R2 score of 91.10%. MSE 15571.97, RMSE 22373.66. On cross-validation, the model scored 82.86%.

AdaBoostRegressor

AdaBoostRegressor

```
abr = AdaBoostRegressor()

abr.fit(x_train,y_train)
predabr = abr.predict(x_test)

print('R2 Score',r2_score(y_test,predabr))
print('MAE',mean_squared_error(y_test,predabr))
print('MSE',mean_absolute_error(y_test,predabr))
print('RMSE',np.sqrt(mean_squared_error(y_test,predabr)))
print('Cross validation score',cross_val_score(abr,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,predabr)-cross_val_score(abr,x,y,cv=5).mean())*100)
```

```
R2 Score 0.8190514371124087
MAE 1036287113.6888313
MSE 23492.96466488416
RMSE 32191.41366403208
Cross validation score 0.7614996238704369
Difference in crossvalidation and R2 score 4.822274300665708
```

AdaBoostRegressor is a non-parametric supervised machine learning model It is an ensemble model and works on the same principle as gradientboostingregressor. It utilises ‘stumps’ which is a tree with a node and two leaves. Each stump is made by taking the previous stump’s error into account so some stumps have more say in taking the decision than the other.

The model performed with an R2 score of 83.35%. MSE 23408.59, RMSE 30613.71. We got 76.89 on cross-validation.

RidgeRegressor

Ridge

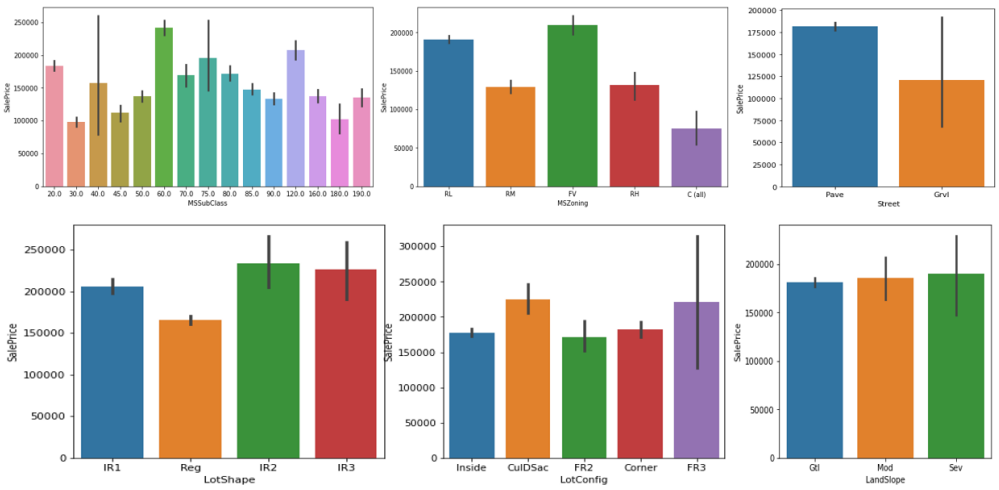
```
ridge= Ridge(alpha=.0001)

ridge.fit(x_train,y_train)
predridge = ridge.predict(x_test)

print('R2 Score',r2_score(y_test,predridge))
print('MAE',mean_squared_error(y_test,predridge))
print('MSE',mean_absolute_error(y_test,predridge))
print('RMSE',np.sqrt(mean_squared_error(y_test,predridge)))
print('Cross validation score',cross_val_score(ridge,x,y,cv=5).mean())

print('Difference in crossvalidation and R2 score',(r2_score(y_test,predridge)-cross_val_score(ridge,x,y,cv=5).mean())*100)
```

```
R2 Score 0.8706375642775803
MAE 740854876.0778047
MSE 19535.26572756185
RMSE 27218.64941685764
Cross validation score 0.8017087605250982
Difference in crossvalidation and R2 score 6.892880375248211
```

It is a type of linear regression which utilises L2 regularization technique in order to overcome over fitting of the model. It also solves problems with high collinearity between the independent variables.

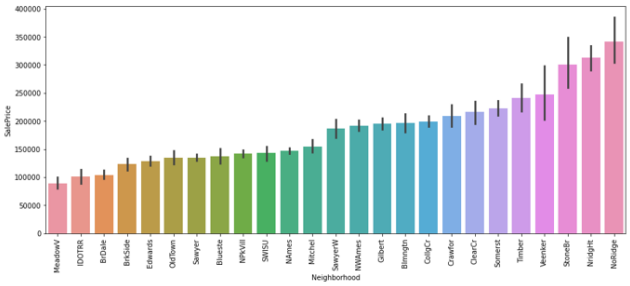
The model performed with an R2 score of 85.74% MSE 20189.37, RMSE 28329.75. On cross validation, the score was found to be 80.01%.

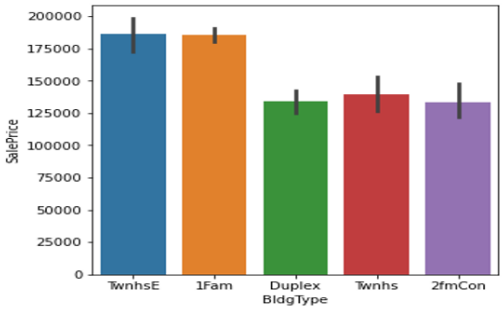
Visualization

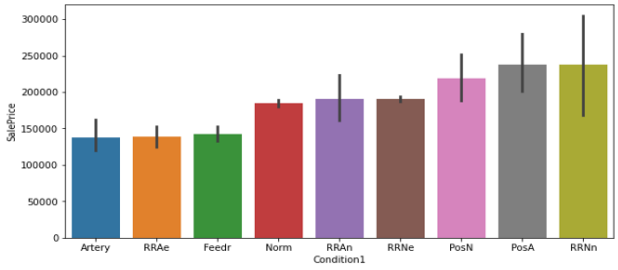


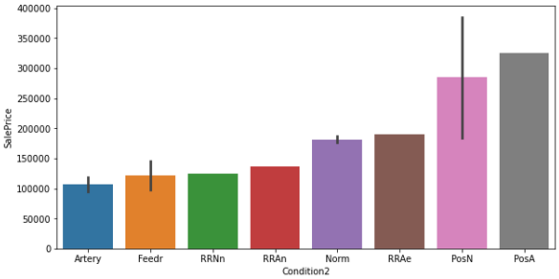
- MSSubClass: 60 : which is a 2-STORY 1946 & NEWER are seen to be to be priced the most
- MSZoning: FV Floating Village Residential, RL Residential Low Density are on the pricier side. C(all) is priced the least
- Street: We can asee that on average houses with gravel roads are monstly priced less, but we can observe a long error bar suggesting that the prices for the street with gravel roads have values that are highly variable from the mean price.
- LotShape: I f we look at the distribution for each, a lot of houses are Reg, but they are mostly priced less than 200000. IR2 and IR3 which are rare are found to be in a price range greater than 20,0000
- LotConfig: CulDSac is seen to be more on the costly side.

- LandSlope: There is not much difference seen with respect to the price.

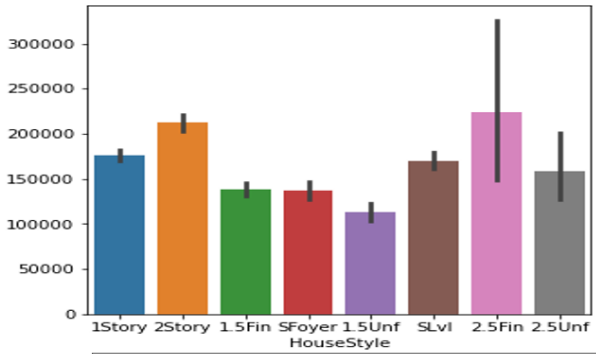


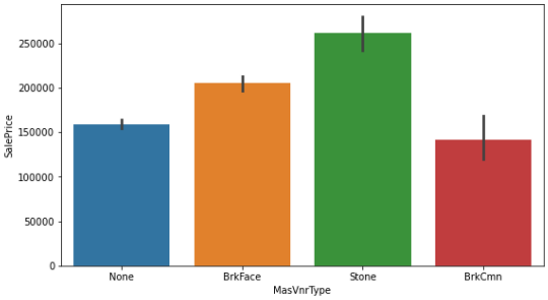


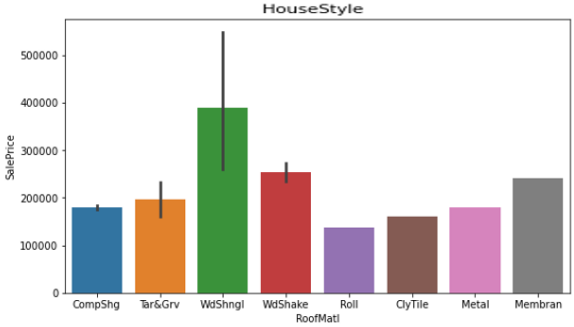


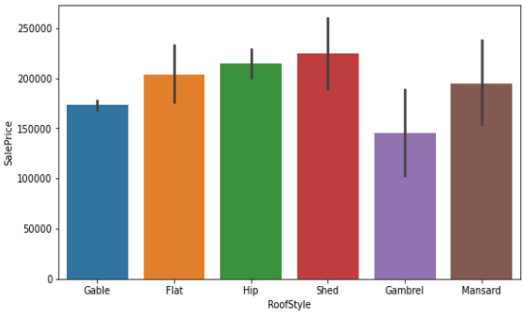


- Neighborhood: NoRidge,NridgHt are on pricier side where as MeadowV, IDOTRR, BrDale are cheap.
- BlidgType : Duplex. Twnhs, 2fmCon price race is comparatively less tha TwnhsE and 1Fam
- For Condition 1 Feedr, RRAe and RRAe falls on cheaper side where as RRNn and PosA are pricey
- Condition 2 Artery falls on cheaper side. PosA houses have high sale price

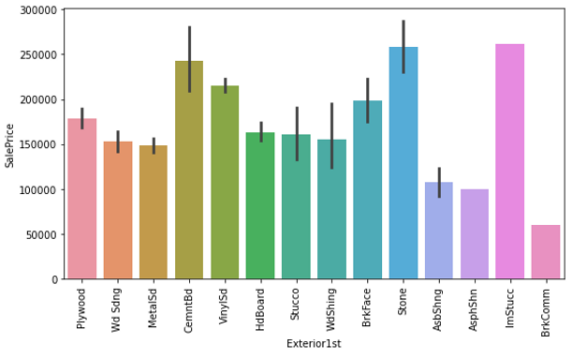


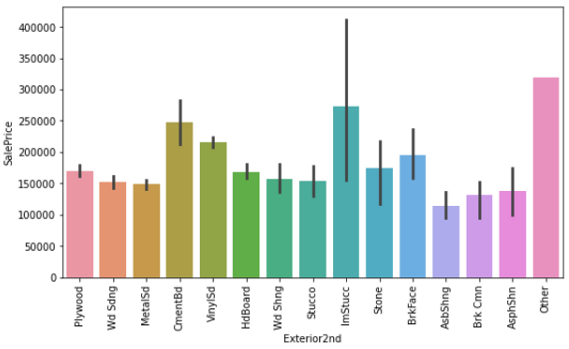






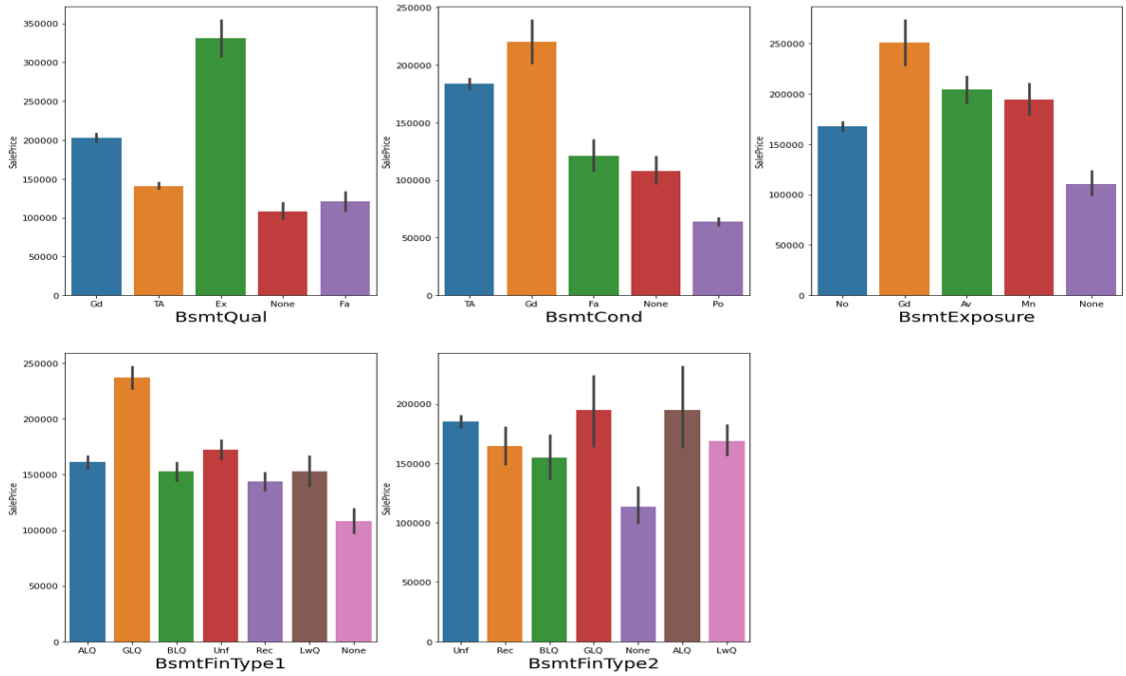
- HouseStyle : 2Story are price higher, 2.5 Fin is also price higher however the variability in prices is high. 1.5Unf is the cheapest.
- MasVnrTyp: While BrkCmn is the cheapest, Stone has high sale price.
- RoofMatt: House with roll are cheaper whereas WdShngl are pricy
- RoofStyle: Overall Shed has high sale price.





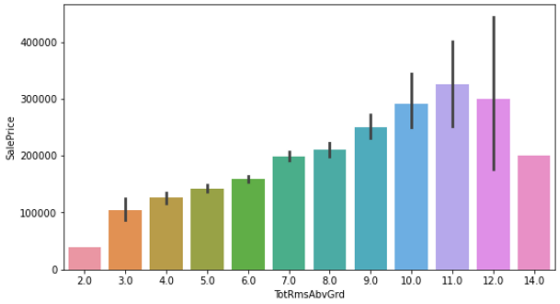
- Exterior1: BrkComm are priced the cheapes whereas lmStreet have high sale price.

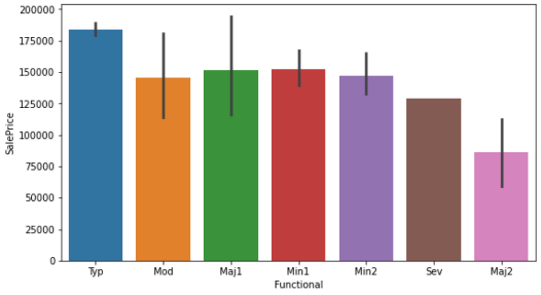
  Exterior 2: AsbbShng are priced cheaper

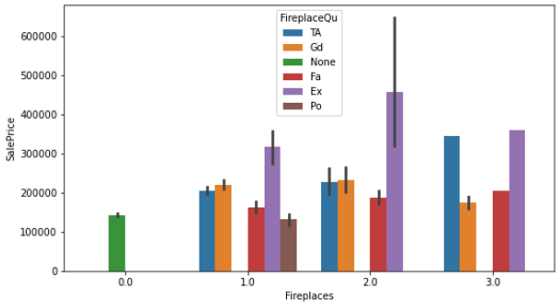


- BsmtQual: Houses with no basements are cheaper. Those rated as excellent have sale price.
- BsmtCond: House rated as poor for BsmtCond are priced less, while those price as Good have high sale price.
- BsmtFinType1: GLO is seen to have high sale price as per the plot.Rest are almost in the same range.

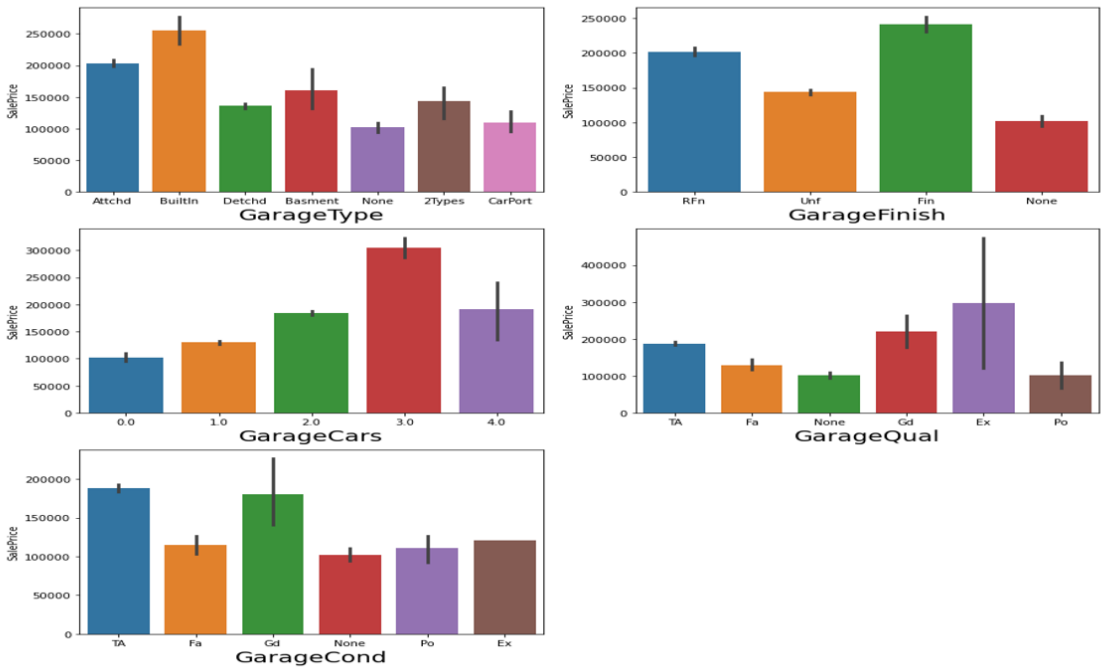
  BsmtFinType2: GLO and ALO have high sale price



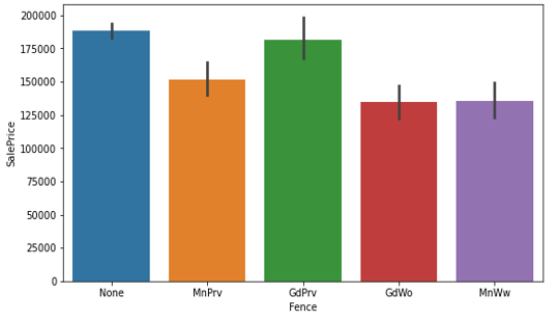


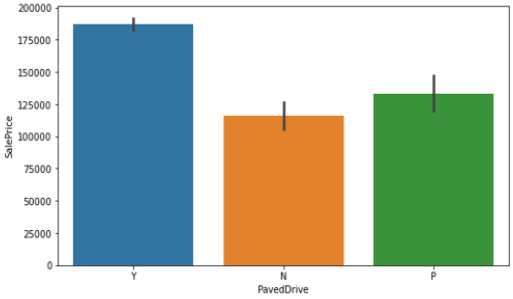


- TotRmsAbvGrd:11 has high sale price. 2 is the cheapest.
- Functional: Type has high sale price whereas Maj2 have less price
- Fireplace: Places with no fireplaces are the cheapest. Those with 2 fireplace have comparatively higher sale price, especially those rated as excellent.



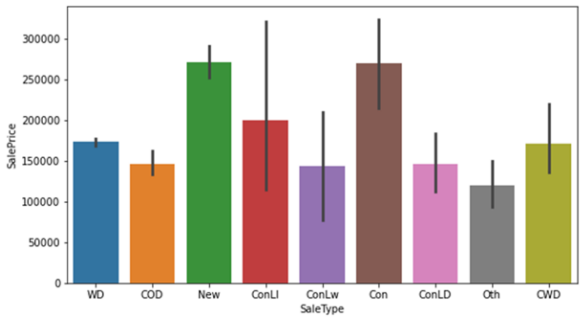
- GarageType: BuiltIn houses have higher sale price. Those that have no garages are priced lowest.
- GarageFinish: Fin are priced highest.
- GarageCars: Houses with 3 garage cars are price high. Those with none are priced the lowest
- GarageQual: Those rated as Excellent have high price rate buth the variability seen in price is high as well.
- GarageCond: TA and Gd are priced high whereas rest are of the same range.
- Overall we can see that houses with no garages are cheaper.

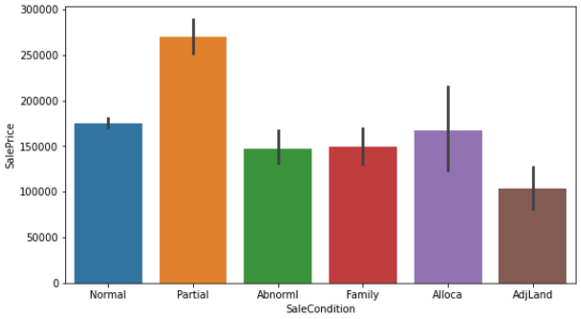




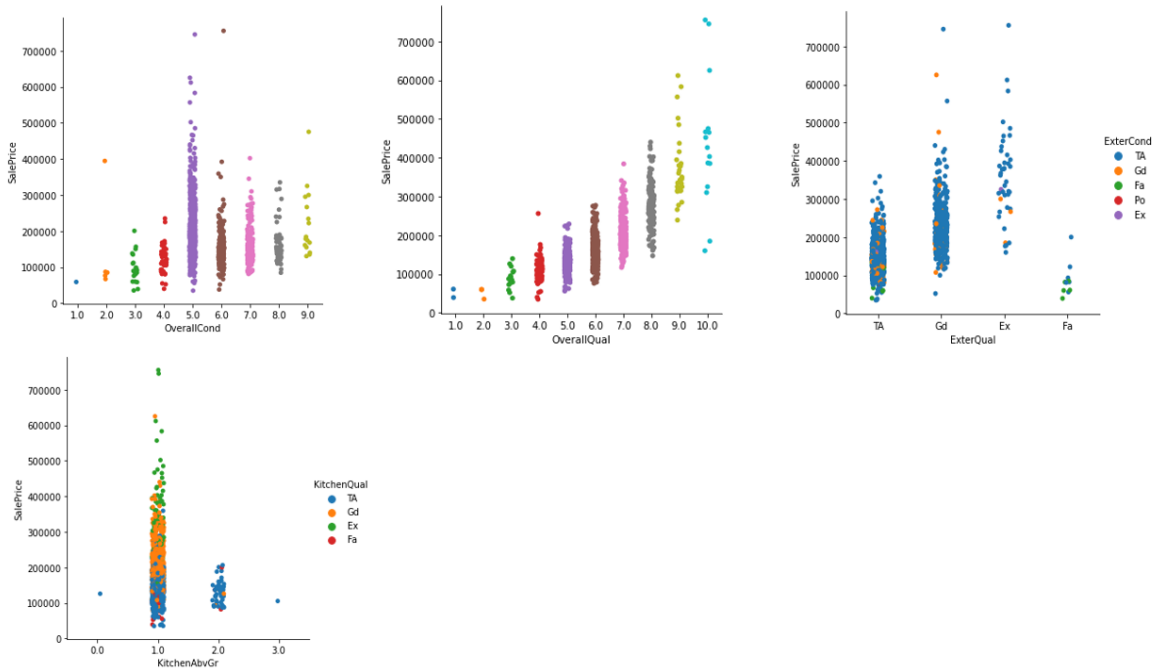
- Fence: Houses with no fences are seen to have slightly higher sale.

  PavedDrived: Houses with paved drive are priced high

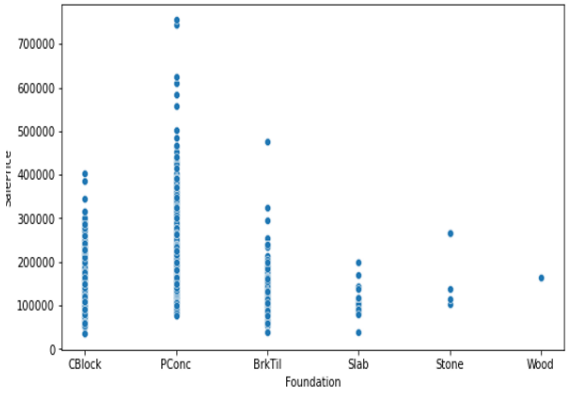


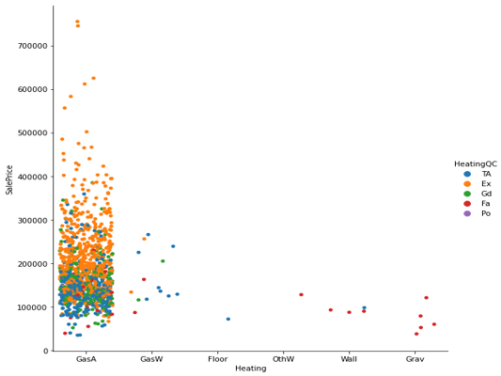


- SaleType: New houses are seen to have high sale price
- SaleCondition: Those with partial sale condition have high sale condition while AdjLand are seen to be cheaper

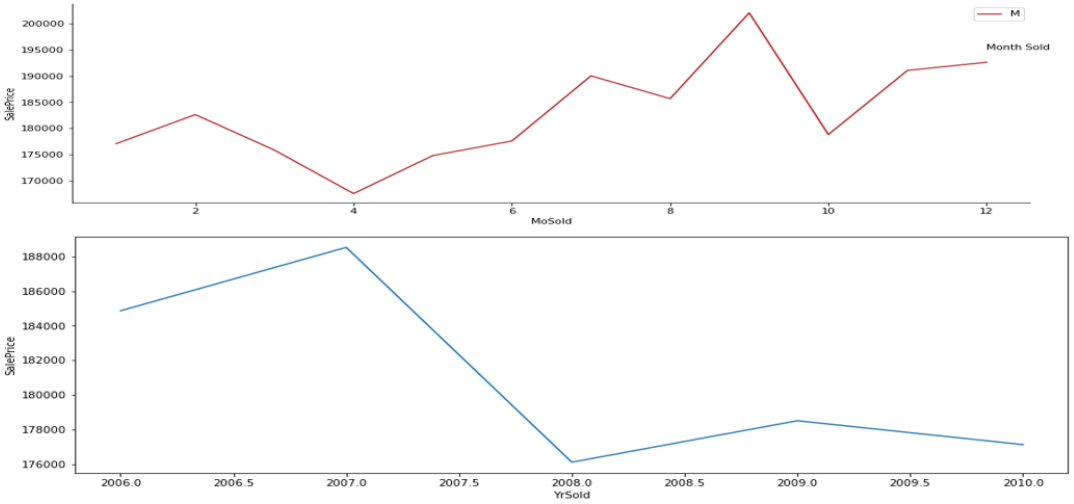


- OverallCondition : Houses with overall condition are seen to be most popular among buyers and are seen to have more houses at higher sale price though houses at
- OverallQual: We can observe a steady increase in prices as the rating increases.
- KitchenAbvGr:No of kitchens does not affect the increase in sale price, however quality does affect. Houses with 1 kitchens are seen more frequent, in which all ranges are seen. Those rated as Excellent quality have higher sale price.

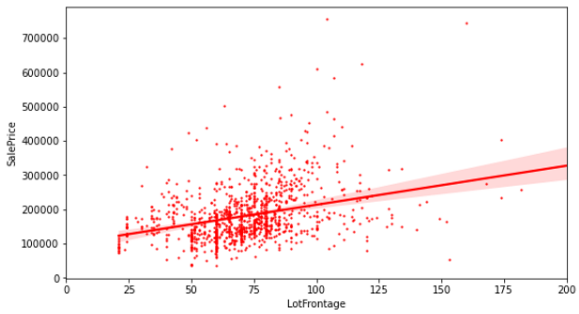


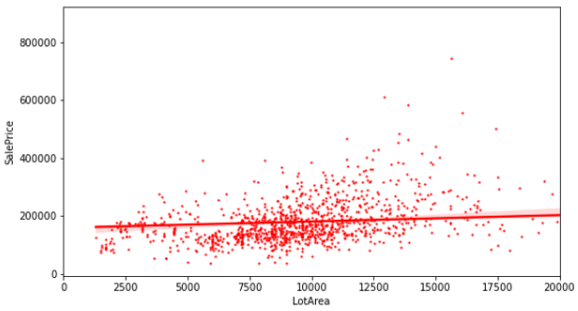


- Foundation: Pconc is seen to have higher sale price
- Heating: GasA is seen to be used more and those rated excellent are seen to be having higher sale prices

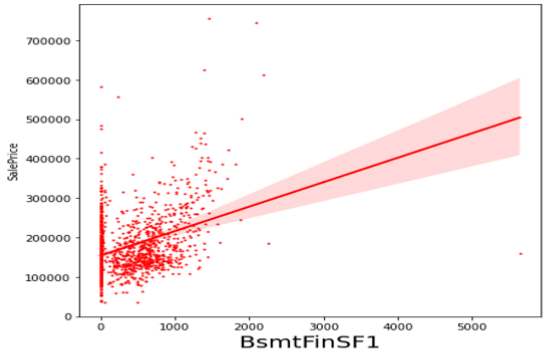


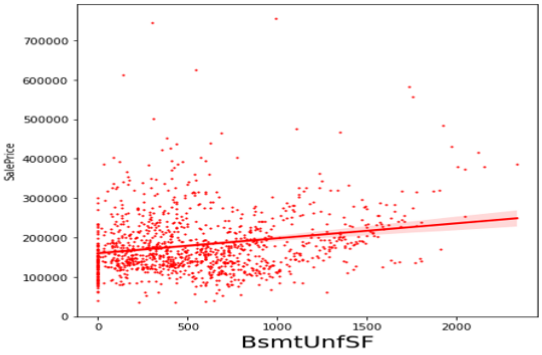
- We can observe price hike in the 9th month
- Sale Price was seen to have a steep decline from 2007 when the house was sold

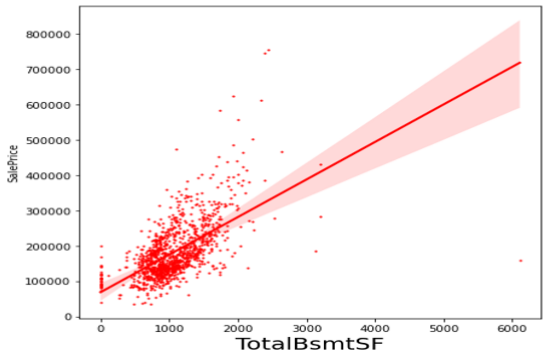




- We can observe positive correlation between LotFrontage and SalePrice
- Lot Area and sale price does not seem to have much correlation between each other

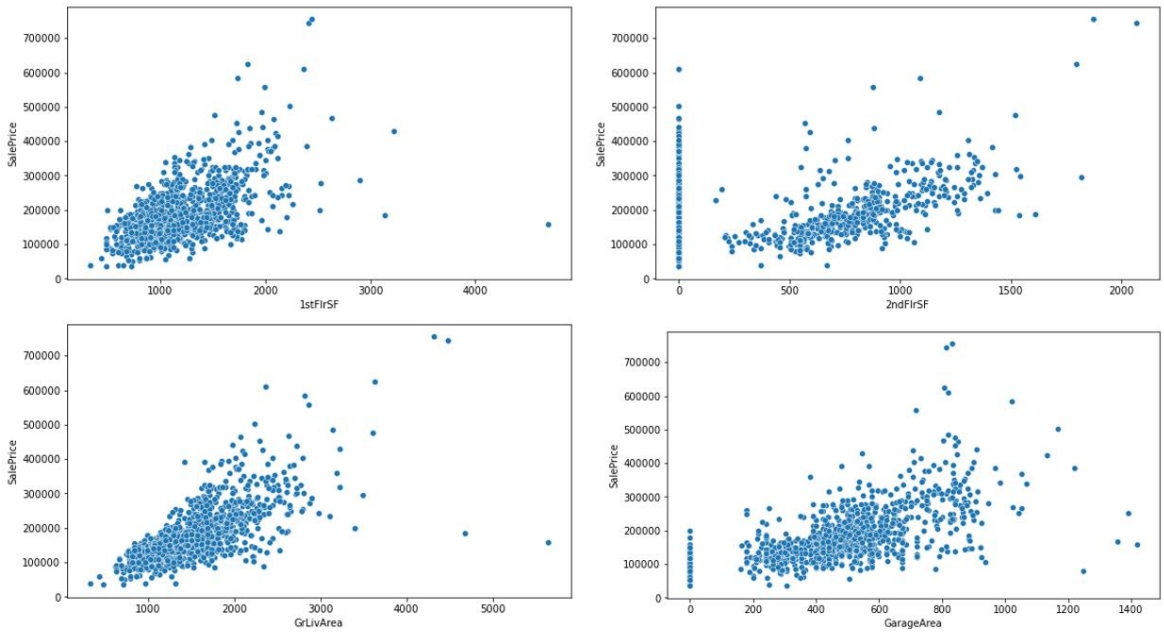




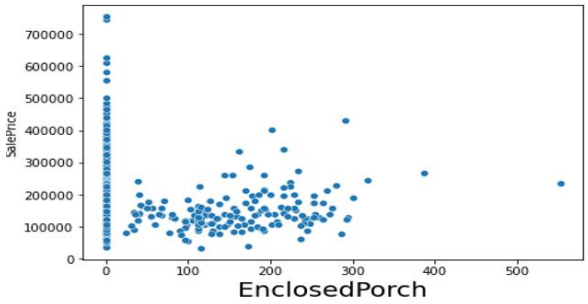


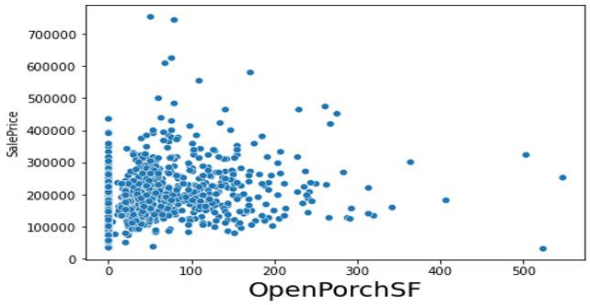
- BsmtFinSF1: Based on the plot it appears sale price is positively correlated with BsmtFinSF1.
- BsmtUnfSF is very weakly correlated with sale price as the regression line makes an angle almost close to zero.

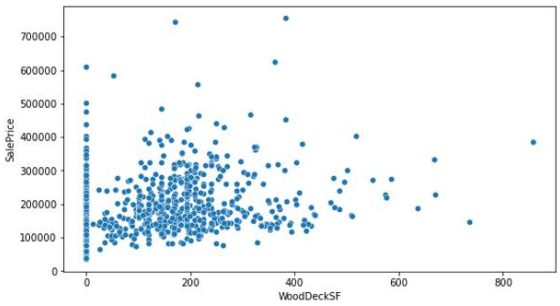
- TotalBsmtSF is also seen to be positively correlated with the sale price



- 1stFirSF: is seen to be positively correlated with the sale price
- 2ndFlrSF: Is seen to be weakly positively correlated with sale price.
- GrLivArea: Is seen to be strongly positively correlated.
- GarageArea: is seen to be weakly correlated with sale price.

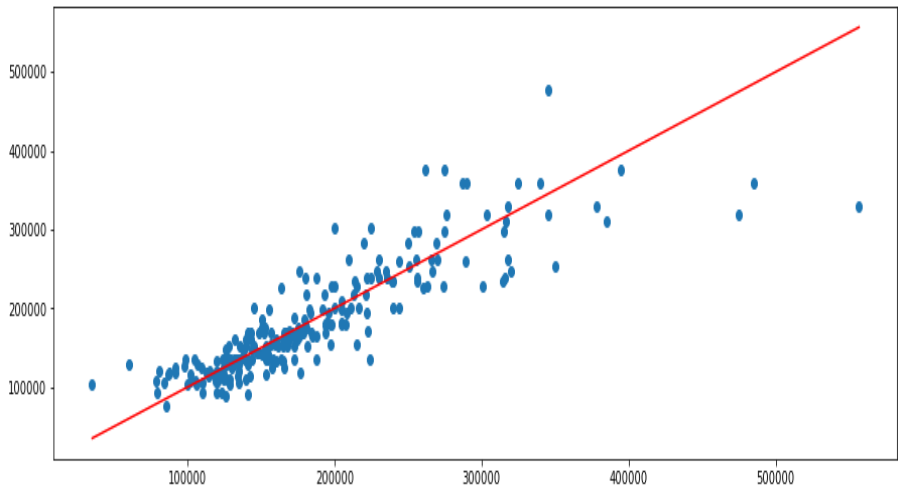






- EnclosedPorch is seen to have no correlation with the sale price.
- OpenPorchSF is positively correlated with the sale price.
- WoodDeckSF as per the graph is positively correlated with the sale price.

Best Model: DecisionTreeRegeessor



- From all the model, DecisionTreeRegressor() has the least difference in crossvalidation score vs the R2 score. So we selected it as the best model.

- We the proceeded to tune the model. After hyperparameter tuning, the R2 score was improved to 74.44% from 71.51%.
- We then proceeded to predict the housing sales price for test dataset

Interpretation of the results

From the dataset that was obtained, we observed that there were 81 columns that mostly described the attributes of a house that influenced the price of a house. From these there were few columns that had 90% and above data as null or zero values, or columns that were either highly variable or had zero variation. Since they did not contribute at all in the prediction, these columns were dropped. Most of the data was skewed which were normalized and the categorical data were encoded. We then proceeded to modelling once the data was scaled. The alogrithms that were used were Random Forest Regressor, Decision Tree Regressor, kNN Regressor, Ada Boost Regressor, Gradient Boosting Regressor, Ridge Regressor. While GradientBoostingRegressor achieved the best R2 score, we proceeded to select DecisionTreeRegressor as the best model as it had the least difference on cross-validation. The model's parameters were then tuned and used for prediction of the test data. On data visualization, we observed that QverallQual, or overall quality of the house was the major determining factor of a house price which indirectly depends on other factors such as Garage area, GrLivArea etc. Overall we saw a high correlation between house price and OverallQual, GrLivArea, GarageCars, GarageArea, BsmtQual and ExterQual.

CONCLUSION

- We saw that after analysis of our data, some of the important attributes that were important for our prediction were found to be:

  OverallQual, GrLivArea, GarageCars, GarageArea, BsmtQual and ExterQual

- As OverallQual of the houses increased the sale price also increased. This also turned out to be the most important predictor in the models.
- Similarly, for GrLivArea, GarageCars, GarageArea showed positive correlation with the sales price.
- In the case of BsmtQual and ExterQual, they showed negative correlation with the sales price, which does not really translate much into any predictive statements regarding housing sales price.
- Multiple models which were Random Forest Regressor, Decision Tree Regressor, kNN Regressor, Ada Boost Regressor, Gradient Boosting Regressor, Ridge Regressor were used for solving the project.
- The best model was selected to be using DecisionTreeRegressor that gave an R2 score of 74.44%. One of the most prominent issue faced during the project was over-fitting of the models. We mostly tackled it through trial and error with the parameters and hyperparameter tuning.
- Limitation of the study would be the high scores for the metrics such as MSE, MAE and RMSE.

- This project can further be extended by collecting data to present a time series data so as to forecast future price prediction since the main area that we focused here was on finding the qualitative aspects that influenced house price prediction.