FLIP ROBO

Housing Price Prediction Project

Submitted by:

Mekhala Misra

ACKNOWLEDGMENT

I took help from following websites:

1)Geek for geeks

2)Pandas documentation

3)researchgate.net

INTRODUCTION

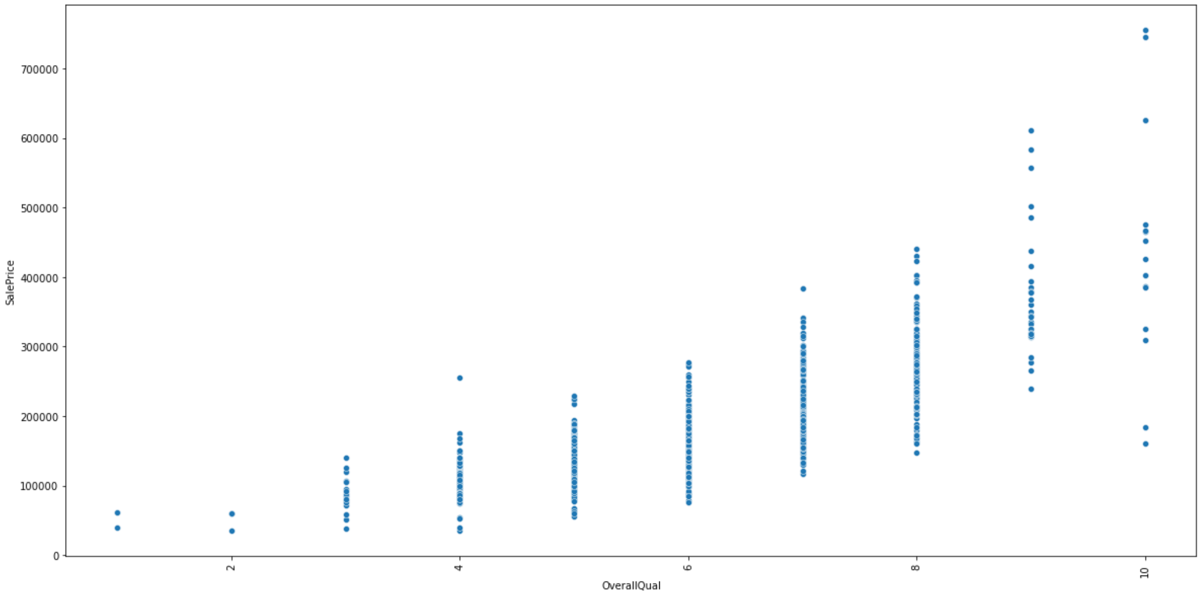
- ## Business Problem Framing

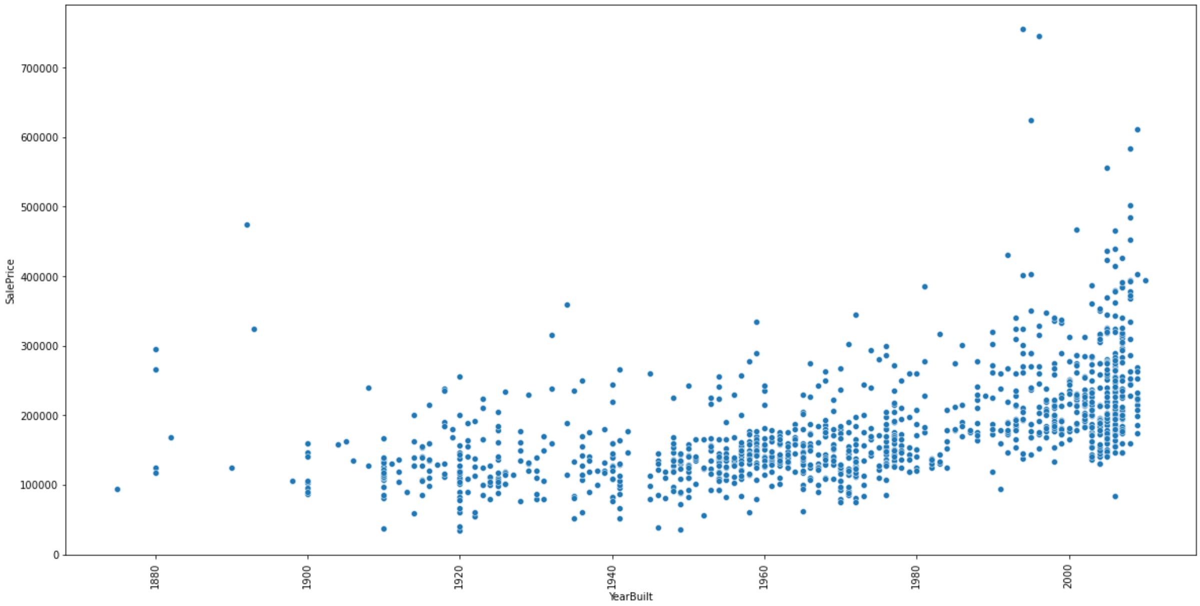
  A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. Now we have to analyse this dataset of independent variables, create a model and predict the price of a house depending on its independent variable values.

- ## Review of Literature

  Using various scatter graph I came to certain conclusion:

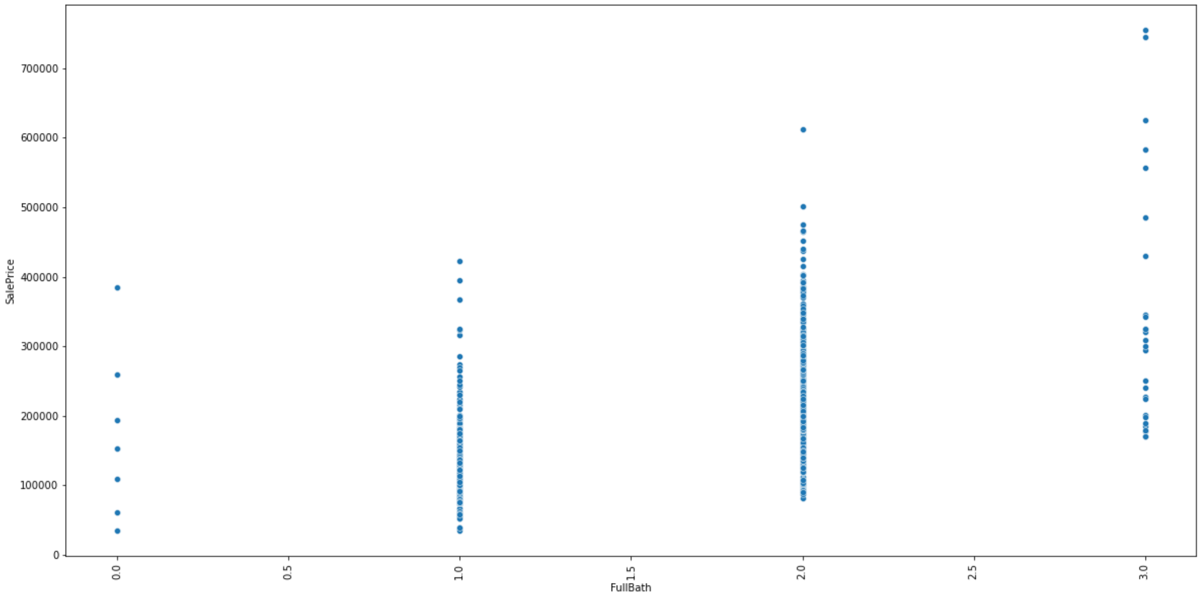
1) Here we can see that overall quality of a house has linear effect on its price.





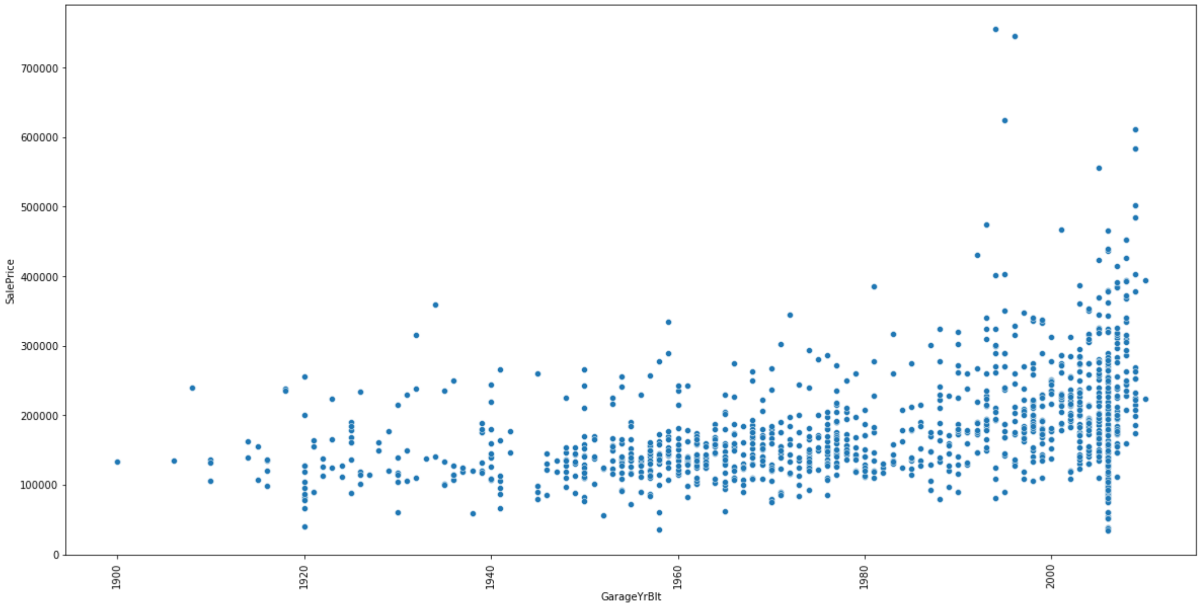
2)

YearBuilt is also linearly related with sale price means newer the house higher its price



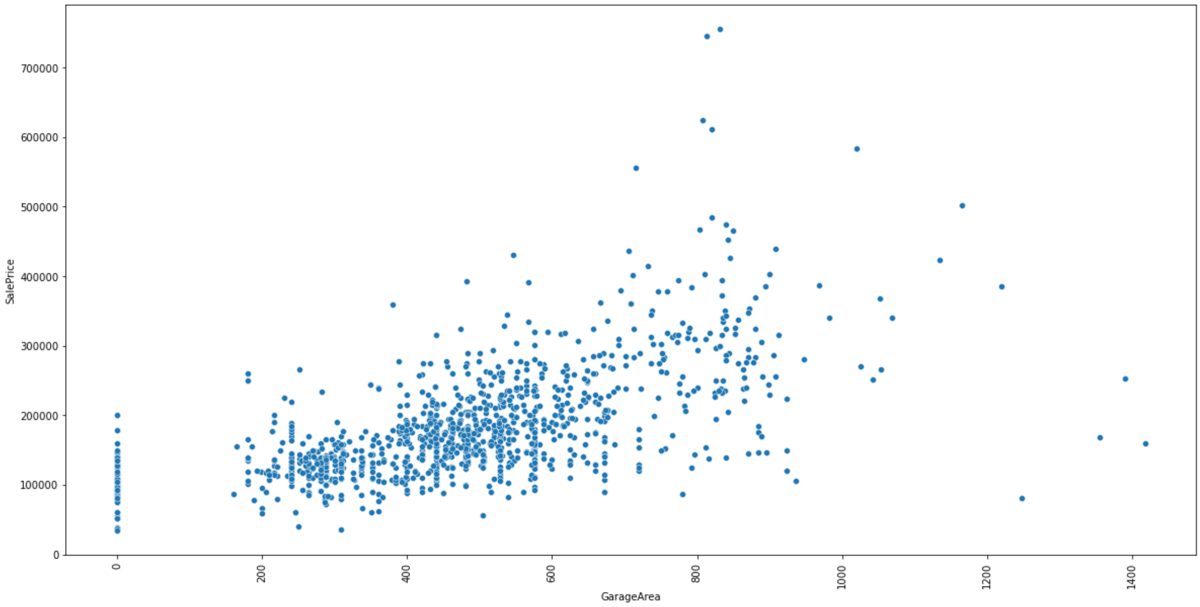
3)

FullBath also positively affects the sales price of house



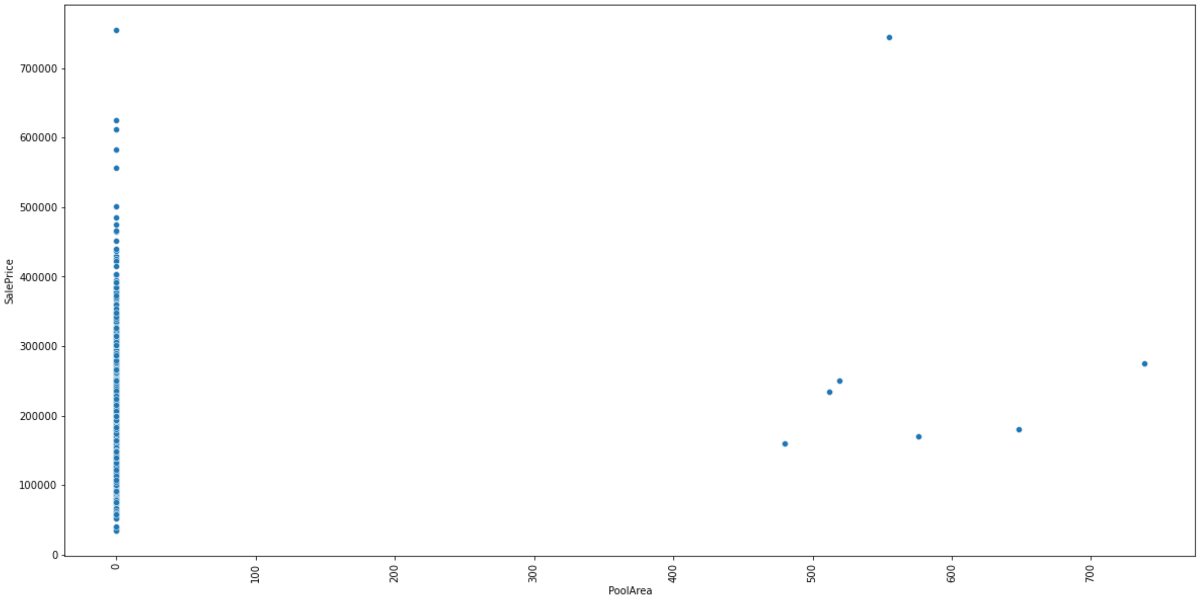
4)

5) Latest the garage built higher is the house price.



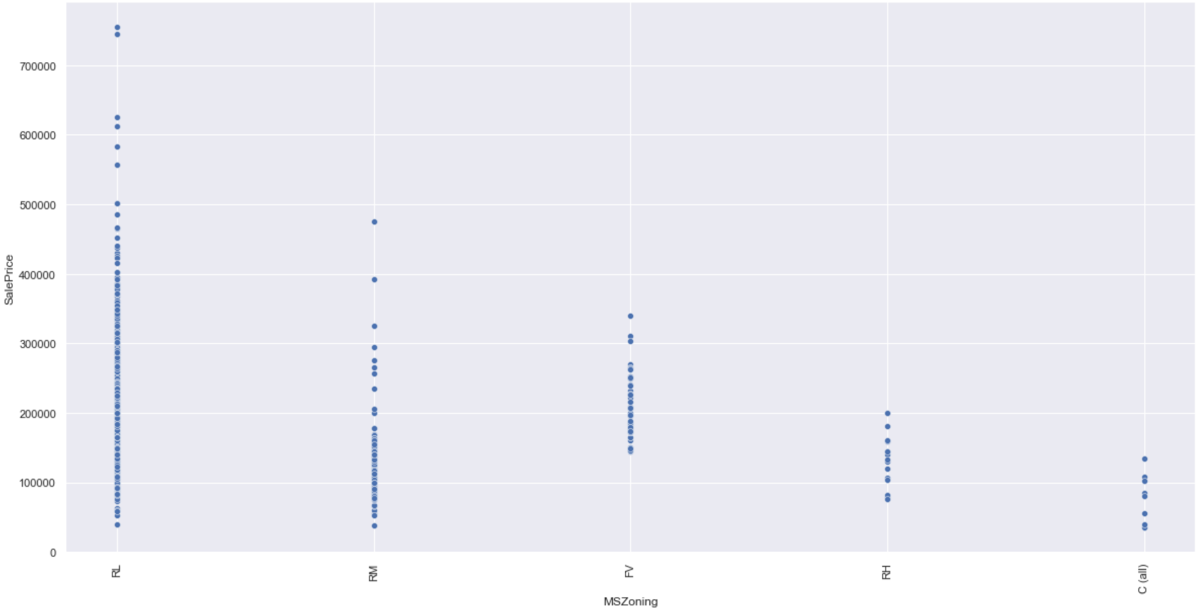
6)

Garage Area affects price positively.



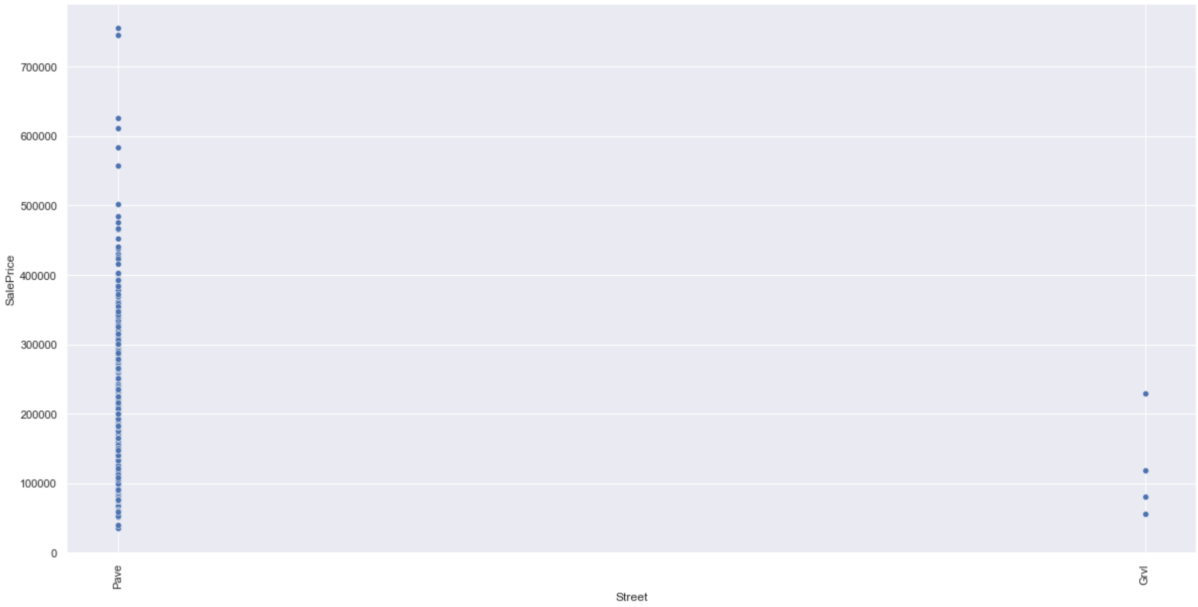
7)

PoolArea doesn’t have any impact on house price.



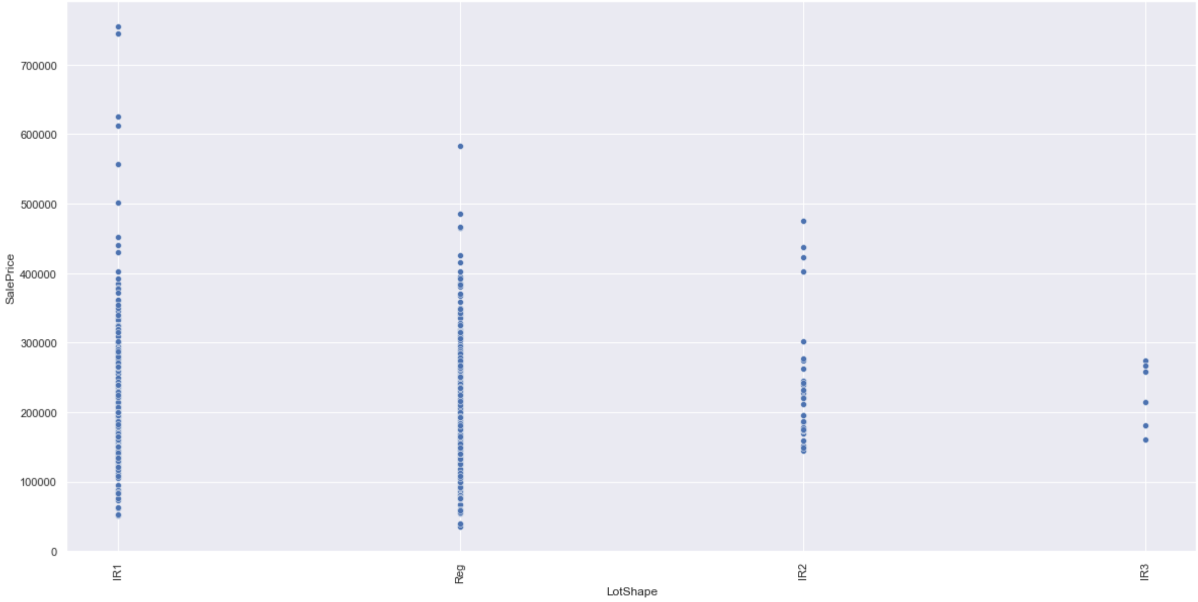
8)

MSZoning is negatively related to sales price



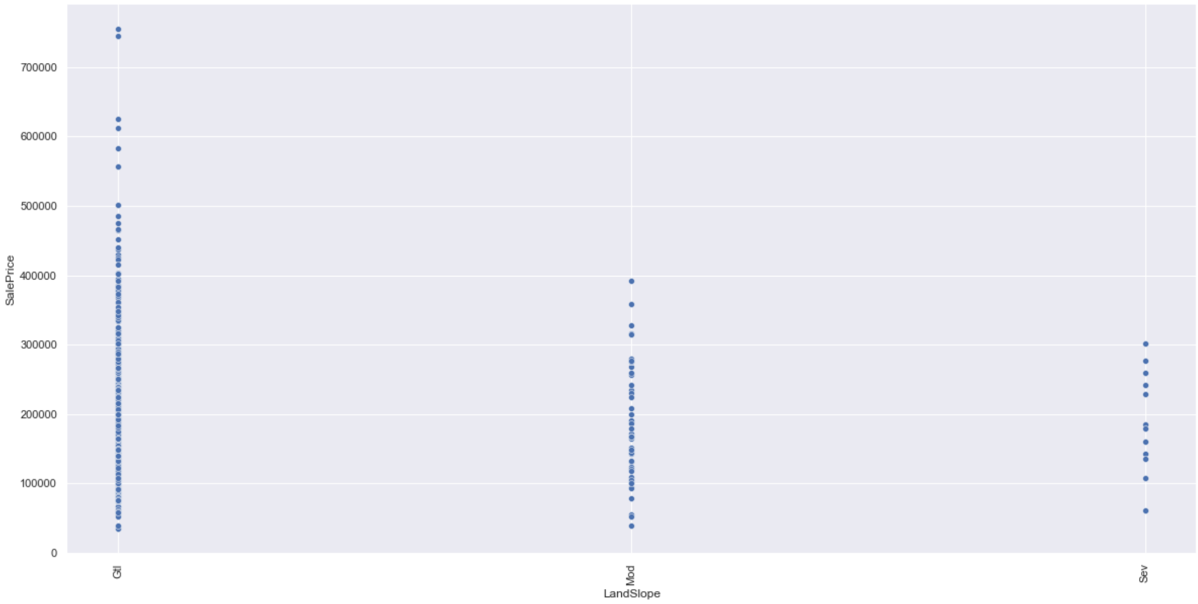
9)

Houses with paved streets are more costly then gravel.



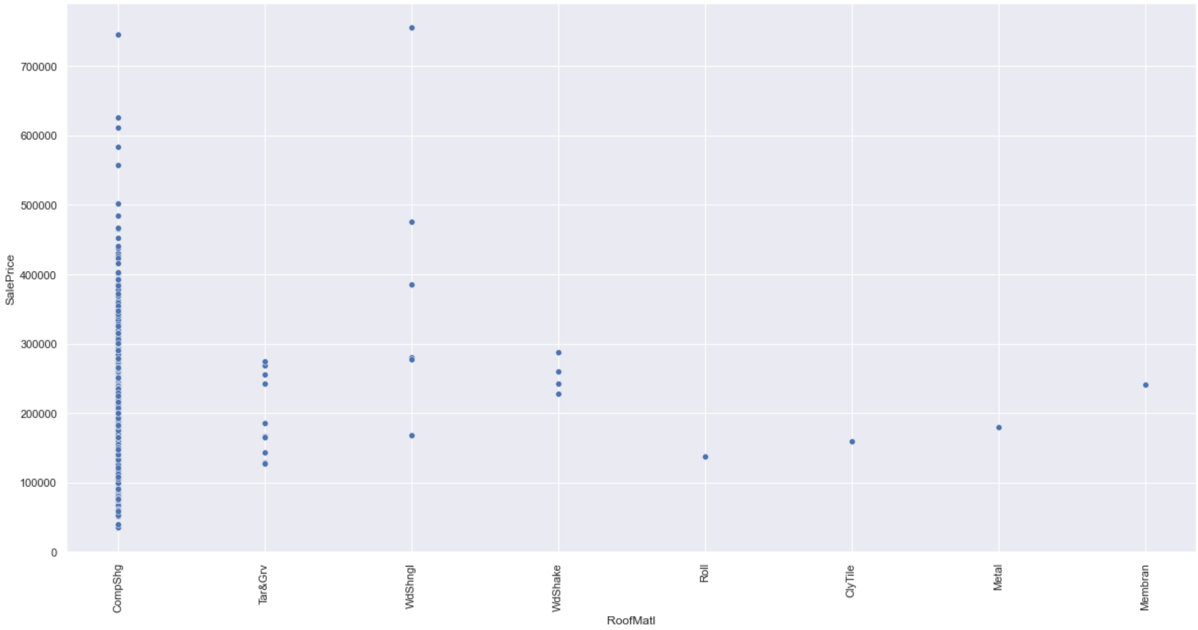
10)

Higher the price if lot shape is IR1.



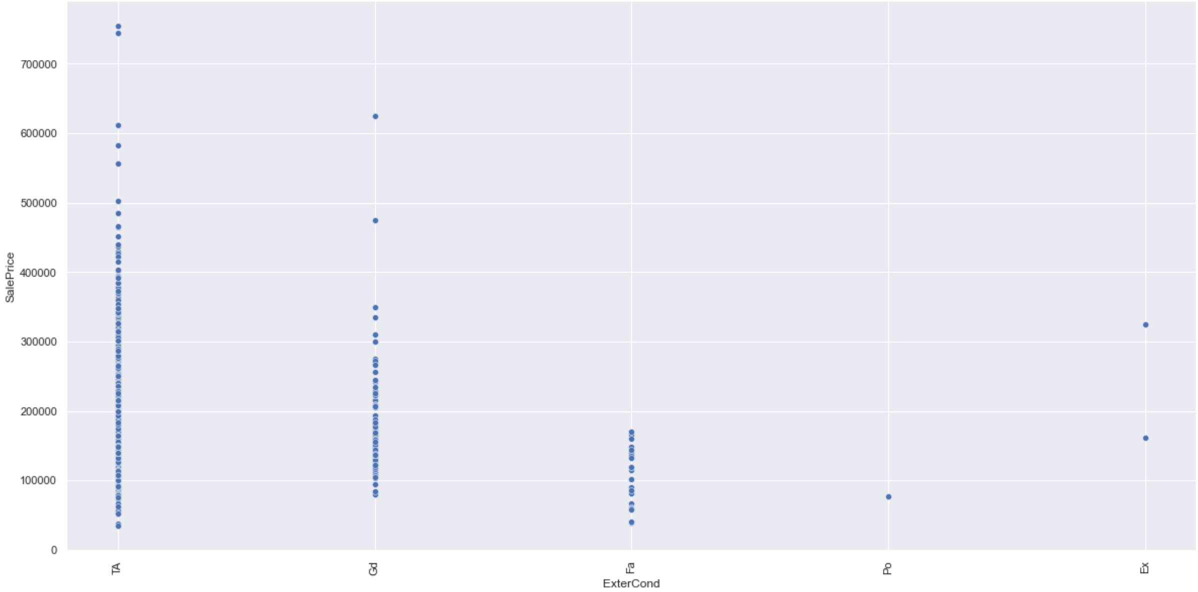
11)

Its is negative linearly related with salesprice.



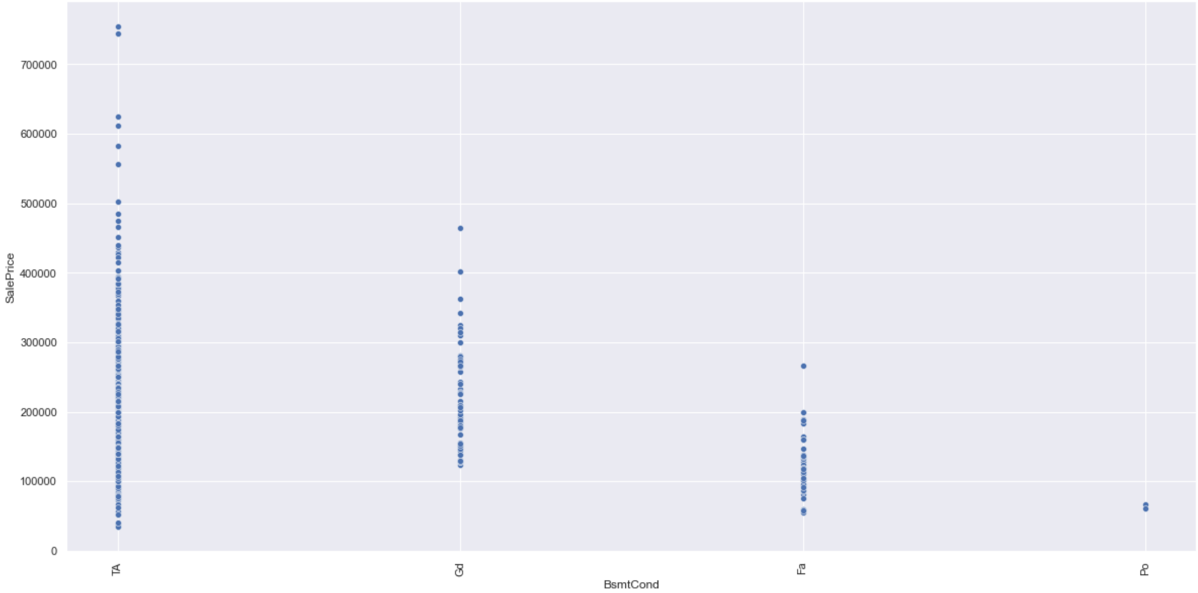
12)

Compshg roofmatl has higher house prices as compared to others.



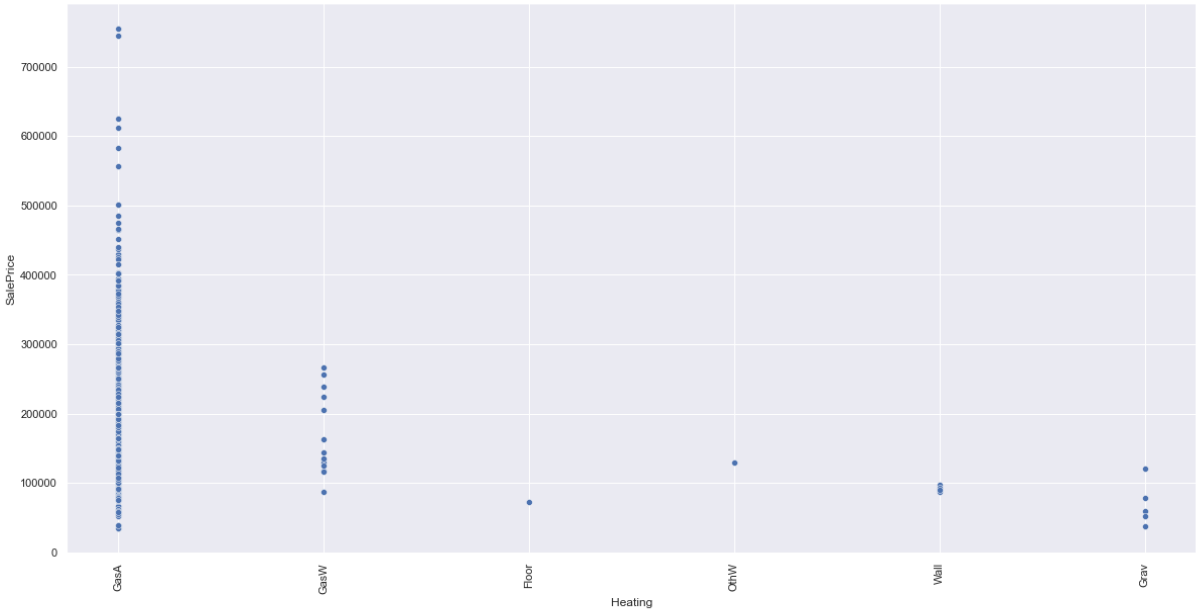
13)

If the exterior has TA then higher the price.



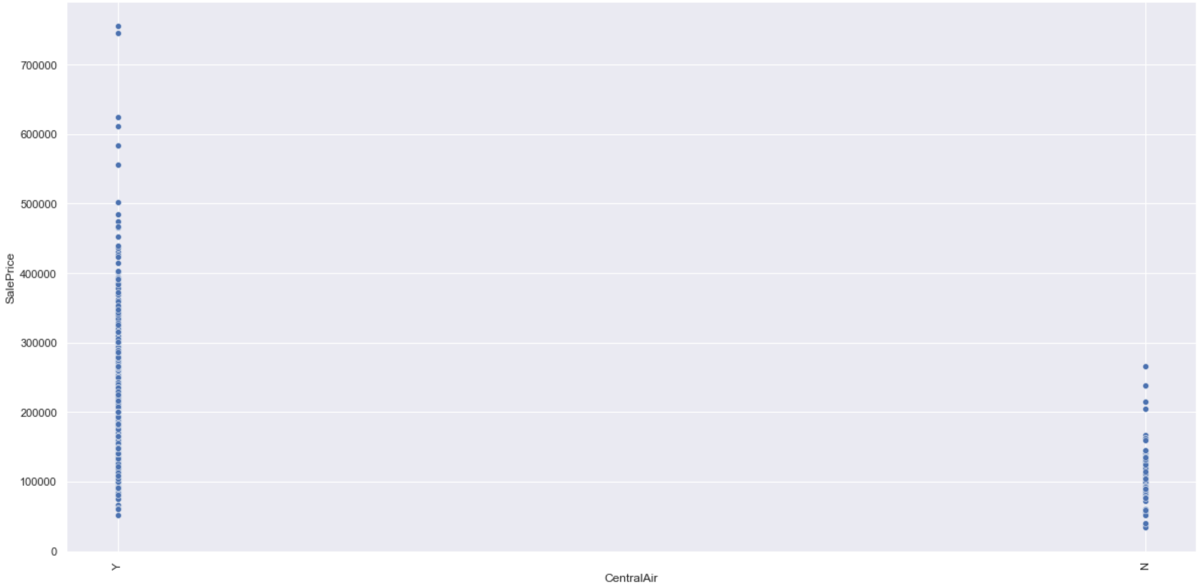
14)

It is also negatively related with sales price.



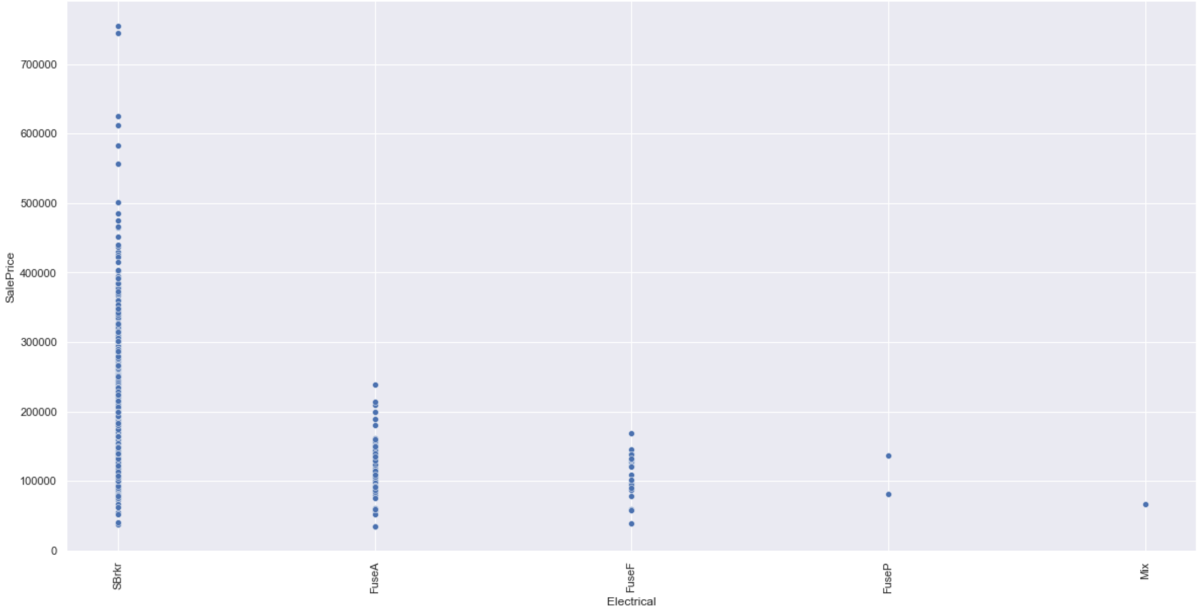
15)

If heating is GasA, so highest the house price.



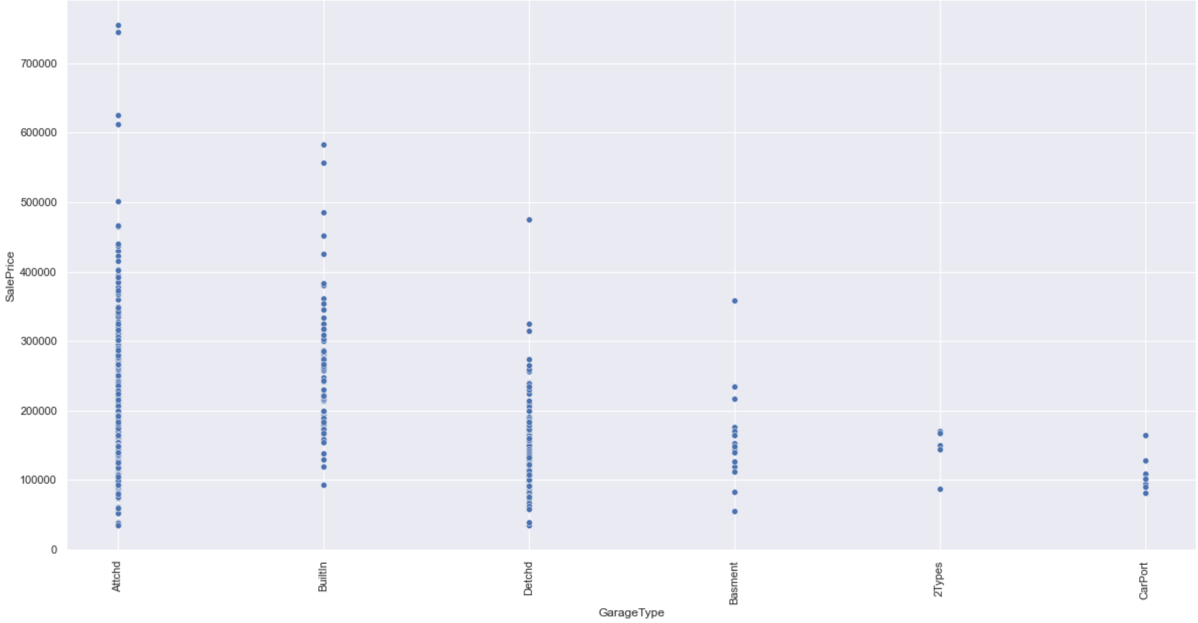
16)

If the house is centrally air conditioned then higher the price.



17)

Electrical is negatively related to sales price.



18)

Attached garage has higher the house price.

- ## Motivation for the Problem Undertaken

In todays time when real estate is the hottest place for investment, all the building company's will be keen in knowing what all aspects should be considered so that more customers are attracted and the price of house is also higher.

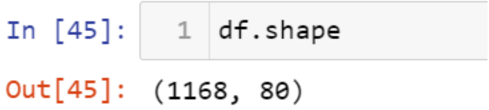
Analytical Problem Framing

- ## Mathematical/ Analytical Modeling of the Problem

  1) The dataset had lot of NAN values so these are imputed with either mean or mode depending on there data type
  2) There were many categorical features so all of them were encoded using ordinal encoder.
  3) After creating heatmap as well as after finding vif score of features we found that there was a problem of multicollinearity as many features vif score was greater than 10. So all those features were deleted that caused multicollinearity.
  4) Outliers were also present but since our dataset is very small so we could not delete those outliers and as this is sales price prediction so outliers cannot be actual outliers as well.
  5) The data was also skewed so we removed skewness using power transform method.
  6) Data was then normally distributed using standardisation technique.
  7) We applied Linear regression, random forest regressor and XGBoost algorithms on the clean data and got 82.6%,87.9% and 81% accuracy respectively.
  8) Depending on the model accuracy we opted random forest regressor and applied hyperparameter tuning on it but the accuracy cannot be improved.
  9) So we saved our previous random forest regressor model.

- ## Data Sources and their formats

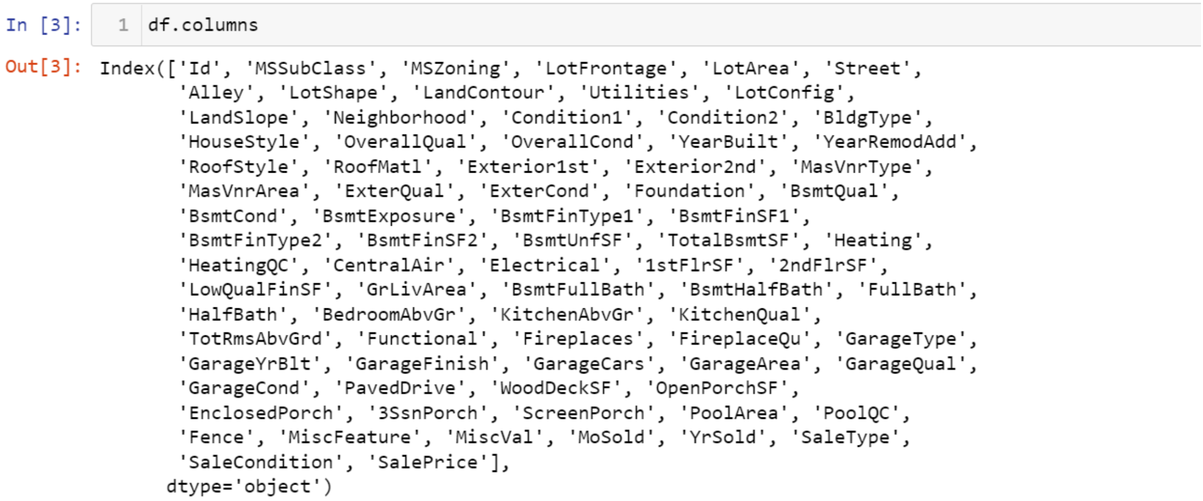
  Data set is provided by the FlipRobo technologies and it has 1168 rows and 81 columns.



```
In [45]: df.shape

Out[45]: (1168, 80)
```

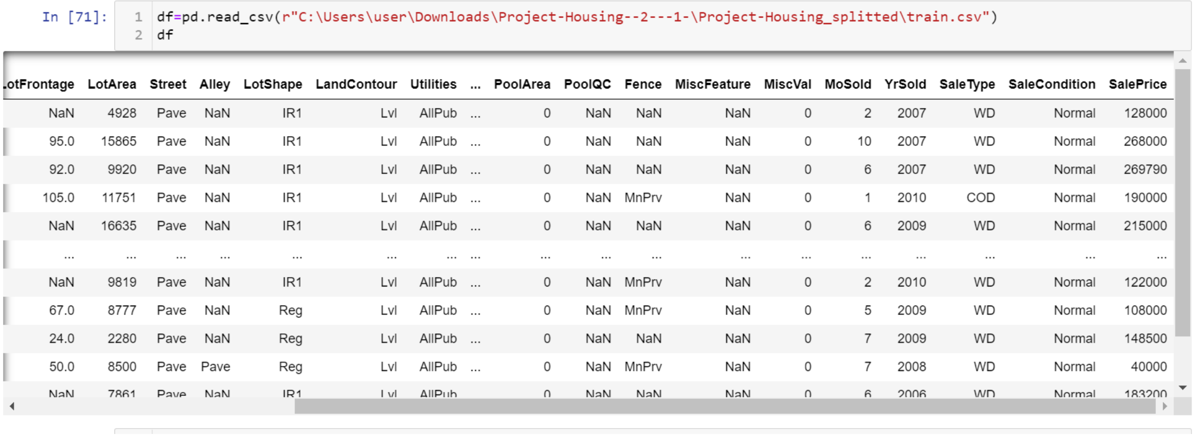
Following are the columns present:



```
In [3]: df.columns

Out[3]: Index(['Id', 'MSSubClass', 'MSZoning', 'LotFrontage', 'LotArea', 'Street',
               'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig',
               'LandSlope', 'Neighborhood', 'Condition1', 'Condition2', 'BldgType',
               'HouseStyle', 'OverallQual', 'OverallCond', 'YearBuilt', 'YearRemodAdd',
               'RoofStyle', 'RoofMatl', 'Exterior1st', 'Exterior2nd', 'MasVnrType',
               'MasVnrArea', 'ExterQual', 'ExterCond', 'Foundation', 'BsmtQual',
               'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinSF1',
               'BsmtFinType2', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', 'Heating',
               'HeatingQC', 'CentralAir', 'Electrical', '1stFlrSF', '2ndFlrSF',
               'LowQualFinSF', 'GrLivArea', 'BsmtFullBath', 'BsmtHalfBath', 'FullBath',
               'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr', 'KitchenQual',
               'TotRmsAbvGrd', 'Functional', 'Fireplaces', 'FireplaceQu', 'GarageType',
               'GarageYrBlt', 'GarageFinish', 'GarageCars', 'GarageArea', 'GarageQual',
               'GarageCond', 'PavedDrive', 'WoodDeckSF', 'OpenPorchSF',
               'EnclosedPorch', '3SsnPorch', 'ScreenPorch', 'PoolArea', 'PoolQC',
               'Fence', 'MiscFeature', 'MiscVal', 'MoSold', 'YrSold', 'SaleType',
               'SaleCondition', 'SalePrice'],
              dtype='object')
```

Here is the glimpse of data:



```
In [71]: df=pd.read_csv(r"C:\Users\user\Downloads\Project-Housing--2---1-\Project-Housing_splitted\train.csv")
         df
```

otFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
NaN	4928	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2007	WD	Normal	128000
95.0	15865	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	10	2007	WD	Normal	268000
92.0	9920	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2007	WD	Normal	269790
105.0	11751	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	1	2010	COD	Normal	190000
NaN	16635	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2009	WD	Normal	215000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
NaN	9819	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	2	2010	WD	Normal	122000
67.0	8777	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	5	2009	WD	Normal	108000
24.0	2280	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	7	2009	WD	Normal	148500
50.0	8500	Pave	Pave	Reg	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	7	2008	WD	Normal	40000
NaN	7861	Pave	NaN	IR1	Lvl	AllPub		0	NaN	NaN	NaN	0	6	2006	WD	Normal	183200

Data Preprocessing Done

- The dataset had lot of NAN values so these are imputed with either mean or mode depending on there data type
- There were many categorical features so all of them were encoded using ordinal encoder.
- After creating heatmap as well as after finding vif score of features we found that there was a problem of multicollinearity as many features vif score was greater than 10. So all those features were deleted that caused multicollinearity.
- Outliers were also present but since our dataset is very small so we could not delete those outliers and as this is sales price prediction so outliers cannot be actual outliers as well.
- The data was also skewed so we removed skewness using power transform method.
- Data was then normally distributed using standardisation technique.

Hardware and Software Requirements and Tools Used

We imported following packages:

Import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import LabelEncoder

```
from statsmodels.stats.outliers_influence import variance_inflation_factor
from scipy.stats import zscore
from sklearn.model_selection import GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import power_transform
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn import metrics
from sklearn.metrics import mean_squared_error, mean_absolute_erro
```

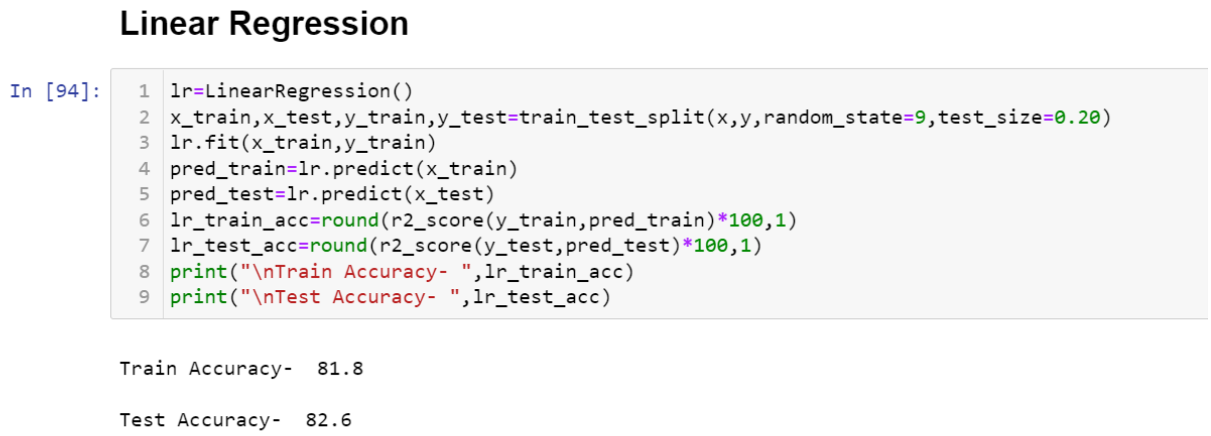
We also installed XGBoost Regressor.

Model/s Development and Evaluation

- ## Identification of possible problem-solving approaches (methods)
- The dataset had lot of NAN values so these are imputed with either mean or mode depending on there data type
- There were many categorical features so all of them were encoded using ordinal encoder.
- After creating heatmap as well as after finding vif score of features we found that there was a problem of multicollinearity as many features vif score was greater than 10. So all those features were deleted that caused multicollinearity.
- Outliers were also present but since our dataset is very small so we could not delete those outliers and as this is sales price prediction so outliers cannot be actual outliers as well.
- The data was also skewed so we removed skewness using power transform method.
- Data was then normally distributed using standardisation technique.

- ## Testing of Identified Approaches (Algorithms)
- We applied Linear regression, random forest regressor and XGBoost algorithms on the clean data and got 82.6%,87.9% and 81% accuracy respectively.
- Depending on the model accuracy we opted random forest regressor and applied hyperparameter tuning on it but the accuracy cannot be improved.
- So we saved our previous random forest regressor model.
- ## Run and Evaluate selected models

  1) Linear Regression



Linear Regression

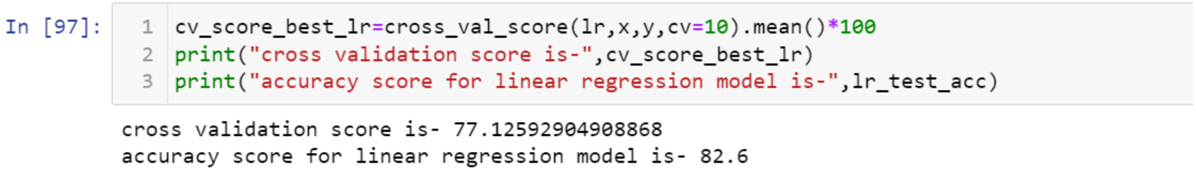
In [94]:

```
lr=LinearRegression()
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=9,test_size=0.20)
lr.fit(x_train,y_train)
pred_train=lr.predict(x_train)
pred_test=lr.predict(x_test)
lr_train_acc=round(r2_score(y_train,pred_train)*100,1)
lr_test_acc=round(r2_score(y_test,pred_test)*100,1)
print("\nTrain Accuracy- ",lr_train_acc)
print("\nTest Accuracy- ",lr_test_acc)
```

```

Train Accuracy-  81.8

Test Accuracy-  82.6
```

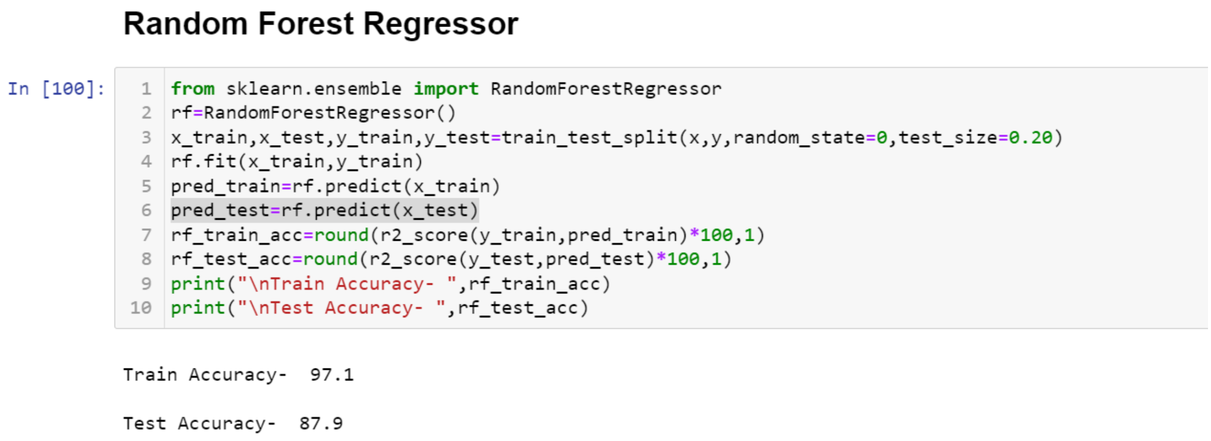


In [97]:

```
cv_score_best_lr=cross_val_score(lr,x,y,cv=10).mean()*100
print("cross validation score is-",cv_score_best_lr)
print("accuracy score for linear regression model is-",lr_test_acc)
```

```
cross validation score is- 77.12592904908868
accuracy score for linear regression model is- 82.6
```

2) Random Forest Regressor



Random Forest Regressor

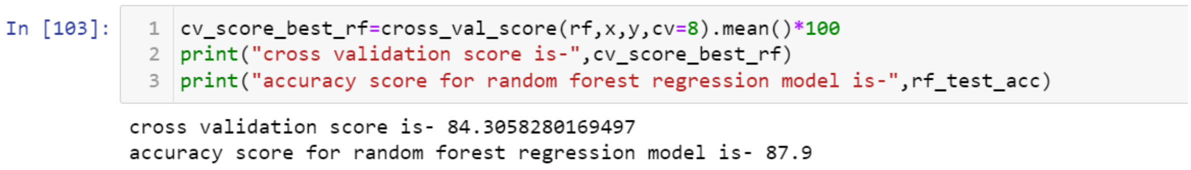
In [100]:

```
from sklearn.ensemble import RandomForestRegressor
rf=RandomForestRegressor()
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=0,test_size=0.20)
rf.fit(x_train,y_train)
pred_train=rf.predict(x_train)
pred_test=rf.predict(x_test)
rf_train_acc=round(r2_score(y_train,pred_train)*100,1)
rf_test_acc=round(r2_score(y_test,pred_test)*100,1)
print("\nTrain Accuracy- ",rf_train_acc)
print("\nTest Accuracy- ",rf_test_acc)
```

```

Train Accuracy-  97.1

Test Accuracy-  87.9
```

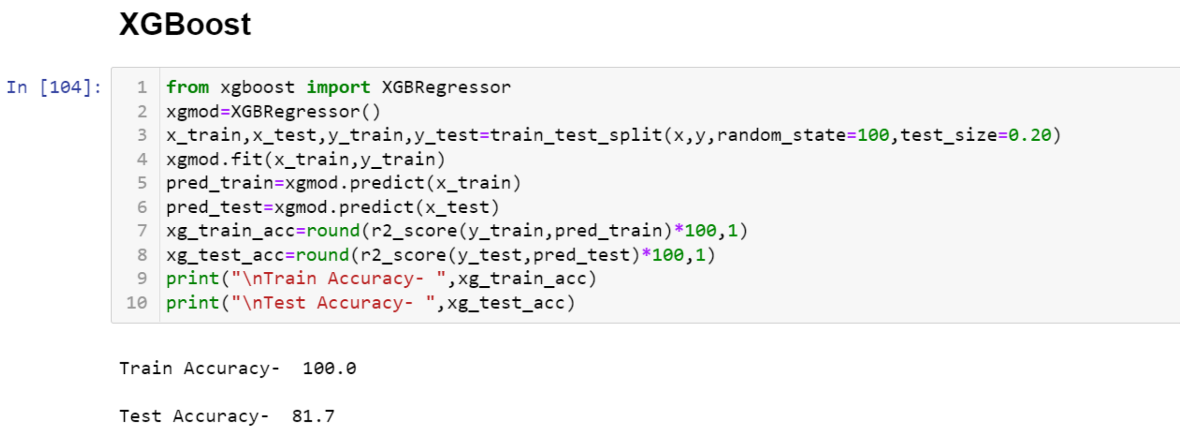


In [103]:

```
cv_score_best_rf=cross_val_score(rf,x,y,cv=8).mean()*100
print("cross validation score is-",cv_score_best_rf)
print("accuracy score for random forest regression model is-",rf_test_acc)
```

```
cross validation score is- 84.3058280169497
accuracy score for random forest regression model is- 87.9
```

3) XGBoost Regressor



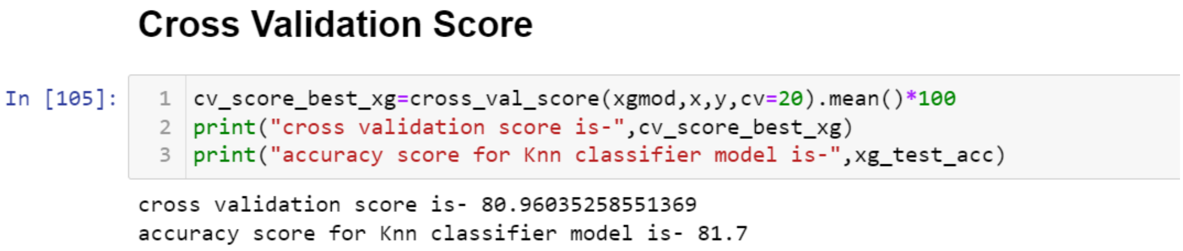
XGBoost

In [104]:

```python
from xgboost import XGBRegressor
xgmod=XGBRegressor()
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=100,test_size=0.20)
xgmod.fit(x_train,y_train)
pred_train=xgmod.predict(x_train)
pred_test=xgmod.predict(x_test)
xg_train_acc=round(r2_score(y_train,pred_train)*100,1)
xg_test_acc=round(r2_score(y_test,pred_test)*100,1)
print("\nTrain Accuracy- ",xg_train_acc)
print("\nTest Accuracy- ",xg_test_acc)
```

```
Train Accuracy-  100.0

Test Accuracy-  81.7
```



Cross Validation Score

In [105]:

```python
cv_score_best_xg=cross_val_score(xgmod,x,y,cv=20).mean()*100
print("cross validation score is-",cv_score_best_xg)
print("accuracy score for Knn classifier model is-",xg_test_acc)
```

```
cross validation score is- 80.96035258551369
accuracy score for Knn classifier model is- 81.7
```

CONCLUSION

- Key Findings and Conclusions of the Study

I found out that the key factors for the house sales price are the lot area, year of establishment, garage, exterior work, streets, electrical work, heating, centrally airconditioned, basements.