FLIP ROBO

House Price Prediction

Submitted by:

Akash Kanjwani

ACKNOWLEDGMENT

During the process of completing this project, I have referred following materials for which I owe them great gratitude.

1.Data Source - Shared by Flip Robo Technology

2. Data trained video tutorials.

3. Scikit-learn https://scikit-learn.org/stable/

4. Machine Learning for Dummies by John Mueller and Luca Massaron - Easy to understand for a beginner book.

5. Geeksforgeeks. https://www.geeksforgeeks.org/

Besides that all the observation, creations of the models and graphs done by self help.

Problem Statement:

Houses are one of the necessary needs of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. 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. Our problem is related to one such housing company. 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.

Business Goal:

You are required to model the price of houses with the available independent variables. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market.

Review of Literature:

Fristly it is important to know which type of model we are going to construct and which parameters and techniques will be used.

We have to see the dataset first by this we get, which type of information we have, what is the pattern and what type of model we are going to construct.

* EDA (Exploratory data analysis) is the most import part of machine learning model, without it we cannot built our perfect model because model requires some filtration and cleaning, here EDA helps us to do find these garbage present inside the data.

* When we get the huge amount of data there is very high chances of outliers and skewness present inside the dataset. So it become necessary to remove zscore as well as skewness .

* Data scaling is one of the most important thing to do in the machine learning. It helps us to scale the data.

* When we have the dataset which contains many columns, which gives the same information, So here we have to adopt the Variance inflation factor.

Outliers, Skewness, Data scaling, Variance inflation factor

We had outliers in some columns and also skewed data was present in our data set . To Clean the data we have performed following steps.

- For outliers removing we used z score method
- For skewness removing we used power transformation
- For Data scaling we used Min max scaler.
- We had to drop some columns because many column was providing the similar results for this we used VIF and dropped some columns.

Model Building

Since our problem was based on the prediction of house price which shows that our problem was based on the regression model, hence we used regression technique and used many regressior like:

- Linear Regression
- Decision Tree Regressor
- KNeighbors Regressor
- Support Vector Regressor

For Bagging and boosting:

- Random Forest Regresssor
- Gradient Bossting Regressor
- AdaBoost Regressor
- XgBoost Regressor

For Regression model we Found some metrix like:

1. R2 score.
2. Mean absolute error.
3. Mean squared error.
4. Root mean squared error.

For overfitting and underfitting and for Hyperparameters tuning :

Grid Search CV

Cross_validation_Score

In order to find out our best model we compared all the model and found :

R2 Score : 88.88

Cross_validation score : 84.65

Diffrence : 4.23

Mean Suqared error : 15085.184

Root mean absolute error: 20344.926

And when we did hyper parameter tuning we found the 86.34 % accuracy

Model name - Gradient Boosting Regressor

Feature importance :

Feature Importance refers to techniques that calculate a score for all the input features for a given model — the scores simply represent the "importance" of each feature. A higher score means that the specific feature will have a larger effect on the model that is being used to predict a certain variable.

Now we imported the text dataset for prediction but before putting the values in the model we apply preprocessing step onto the test data set like Removing outliers, removing skewness, droping column, data scaling.

Motivation for the Problem Undertaken

The aim is to predict the efficient house pricing for real estate customers with respect to their budgets and priorities. By analyzing previous market trends and price ranges, and also upcoming developments future prices will be predicted.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem

Outliers

1) In Our dataset some columns have outliers, As we can see LotFrontage column the mean is 68.97 and standard daviation is 22.83 and also the maximum value is 313 means the data are highly spreded therefor the chances of outliers are present.

2) Similarly in the LotArea mean is 10484 and the standard daviation is 8957 and the maximum values is 164660, shows that outliers are present in this column.

3) In the MasVnrArea column and BsmtFinSF2 column outliers are present. mean is 101.69 and max vlaue is 1600 for Masvnrarea mean is 46.64 and max is 1474 for BsmtFinSF2

WHich shows the data are highly spreded in these column.

4) In the GrLivArea column mean is 1525 and std is 528 and max values is 1795, these data shows that outliers can be present in this columns.

5) 1stFlrSF column and GrLivArea column have the high diffrence between mean and maximum values Mean is 1169 and standard daviation is 391 for 1stflrSF Mean is 1525 and standard daviation is 528 for GrLivArea Above reult shows that the data are highly spreded in these column.

Skewness

In our dataset Some column showing skewness

1) In Miscval(23.06), PoolArea(13.24),LotArea(10.65), LowQualFinSF(8.66) and 3SsnPorch(9.77) columns high skewness is present means the data are not equally distributed.

2) In the ScreenPorch(4.10), EnclosedPorch(3.04),OpenPorchSF(2.41), kitchenAbvgr(4.36), BsmtHalfBath(4.26) and BsmtFinSF2(4.36) columns skewed data is present.

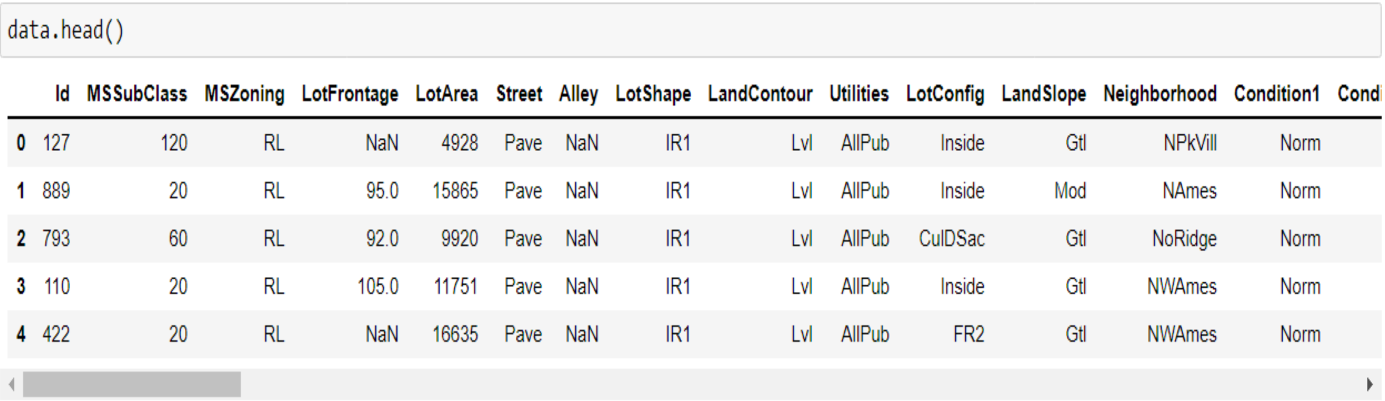
3) MSSubClass(1.42), LotFrontage(2.81), MasVnrArea(2.83), BsmtFinSF1(1.87), TotalBsmtSF(1.74) and 1stFlrSF(1.51) columns have skewed data, means in these columns the data are not equally distributed.

Data Sources and their formats-

We have got the data by the Flip Robo Technologies, The data are available in Csv format .So for this we had to use pandas lib. for converting the csv data into the table format.

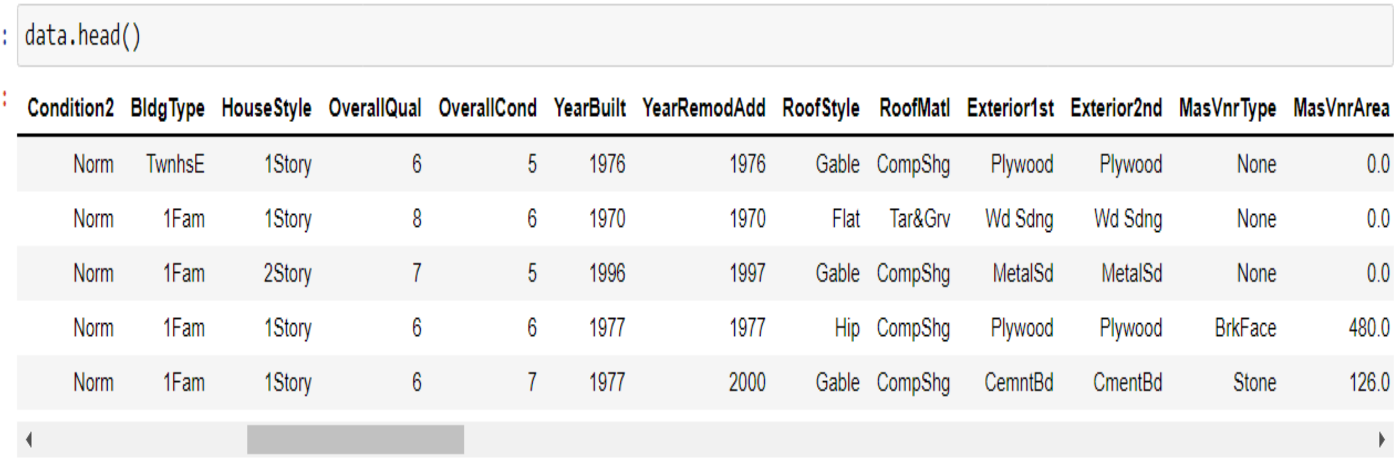
```
import pandas as pd

data=pd.read_csv(r"C:\Users\ABC\OneDrive\Desktop\train.csv")
```



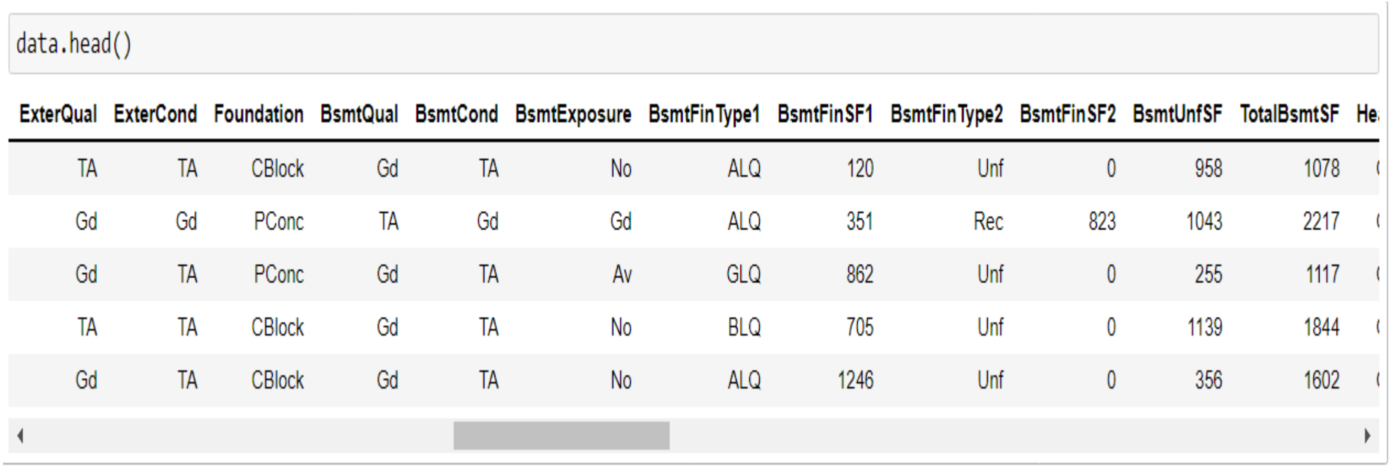
```
data.head()
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood	Condition1	Cond
0	127	120	RL	NaN	4928	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	NPkVill	Norm	
1	889	20	RL	95.0	15865	Pave	NaN	IR1	Lvl	AllPub	Inside	Mod	NAmes	Norm	
2	793	60	RL	92.0	9920	Pave	NaN	IR1	Lvl	AllPub	CulDSac	Gtl	NoRidge	Norm	
3	110	20	RL	105.0	11751	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	NWAmes	Norm	
4	422	20	RL	NaN	16635	Pave	NaN	IR1	Lvl	AllPub	FR2	Gtl	NWAmes	Norm	



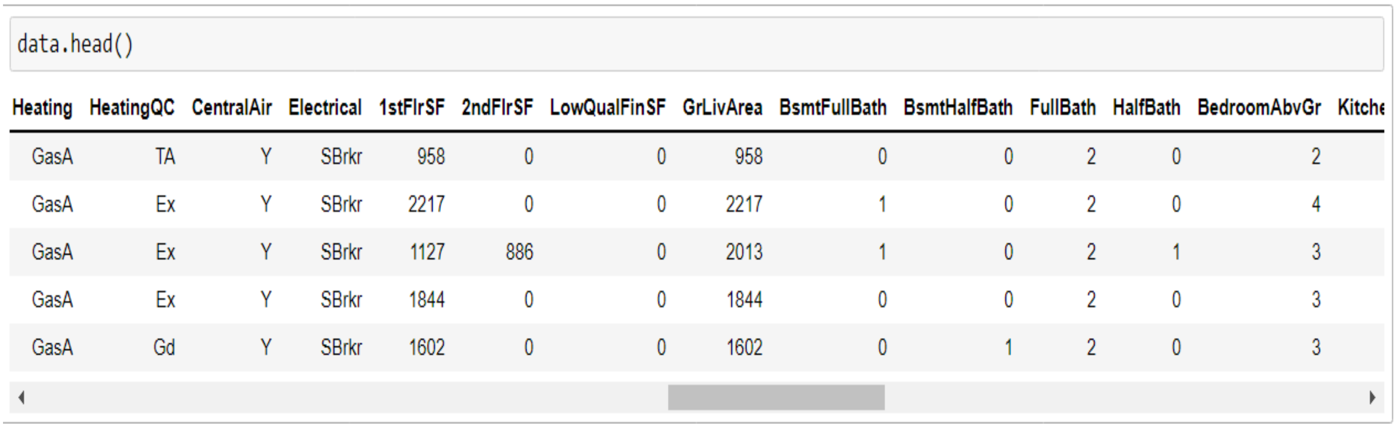
```
data.head()
```

Condition2	BldgType	HouseStyle	OverallQual	OverallCond	YearBuilt	YearRemodAdd	RoofStyle	RoofMatl	Exterior1st	Exterior2nd	MasVnrType	MasVnrArea
Norm	TwnhsE	1Story	6	5	1976	1976	Gable	CompShg	Plywood	Plywood	None	0.0
Norm	1Fam	1Story	8	6	1970	1970	Flat	Tar&Grv	Wd Sdng	Wd Sdng	None	0.0
Norm	1Fam	2Story	7	5	1996	1997	Gable	CompShg	MetalSd	MetalSd	None	0.0
Norm	1Fam	1Story	6	6	1977	1977	Hip	CompShg	Plywood	Plywood	BrkFace	480.0
Norm	1Fam	1Story	6	7	1977	2000	Gable	CompShg	CemntBd	CmentBd	Stone	126.0



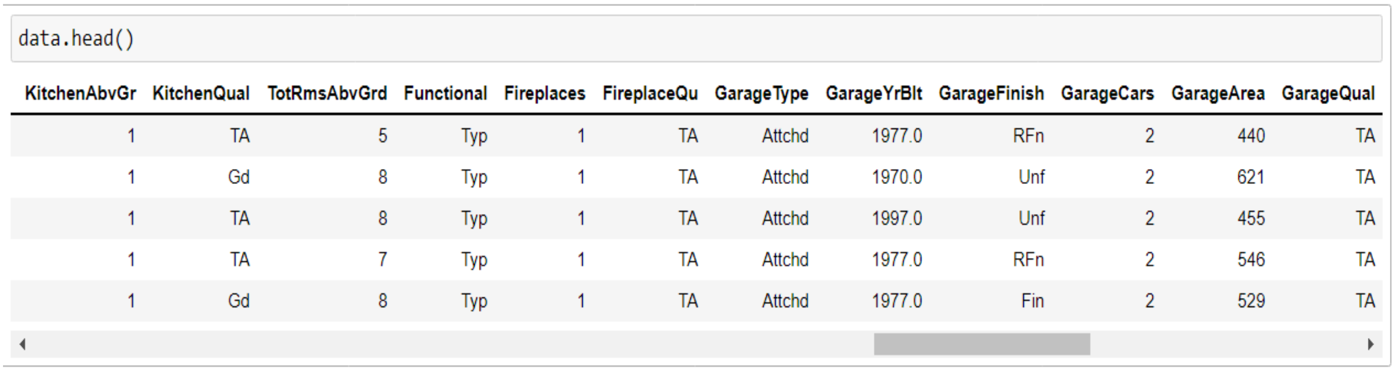
```
data.head()
```

ExterQual	ExterCond	Foundation	BsmtQual	BsmtCond	BsmtExposure	BsmtFinType1	BsmtFinSF1	BsmtFinType2	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	He
TA	TA	CBlock	Gd	TA	No	ALQ	120	Unf	0	958	1078	(
Gd	Gd	PConc	TA	Gd	Gd	ALQ	351	Rec	823	1043	2217	(
Gd	TA	PConc	Gd	TA	Av	GLQ	862	Unf	0	255	1117	(
TA	TA	CBlock	Gd	TA	No	BLQ	705	Unf	0	1139	1844	(
Gd	TA	CBlock	Gd	TA	No	ALQ	1246	Unf	0	356	1602	(



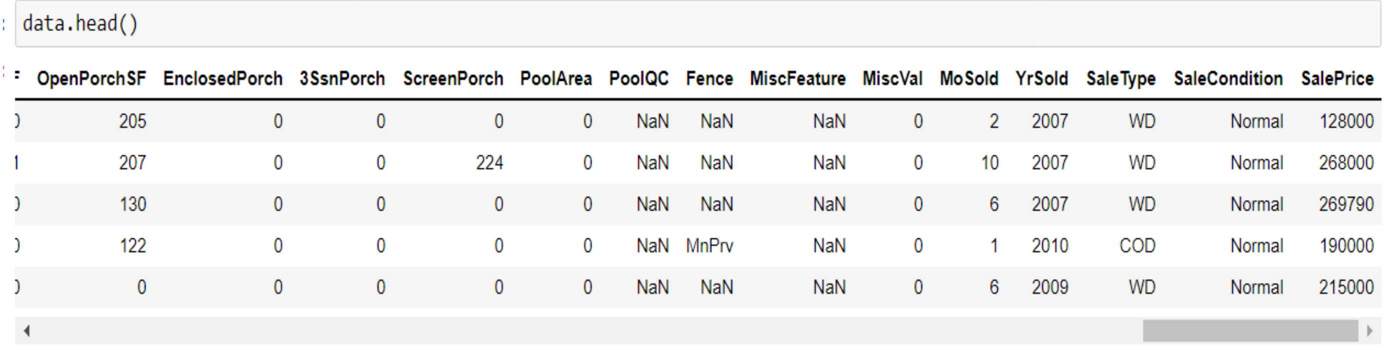
```
data.head()
```

Heating	HeatingQC	CentralAir	Electrical	1stFlrSF	2ndFlrSF	LowQualFinSF	GrLivArea	BsmtFullBath	BsmtHalfBath	FullBath	HalfBath	BedroomAbvGr	Kitche
GasA	TA	Y	SBrkr	958	0	0	958	0	0	2	0	2	
GasA	Ex	Y	SBrkr	2217	0	0	2217	1	0	2	0	4	
GasA	Ex	Y	SBrkr	1127	886	0	2013	1	0	2	1	3	
GasA	Ex	Y	SBrkr	1844	0	0	1844	0	0	2	0	3	
GasA	Gd	Y	SBrkr	1602	0	0	1602	0	1	2	0	3	



```
data.head()
```

KitchenAbvGr	KitchenQual	TotRmsAbvGrd	Functional	Fireplaces	FireplaceQu	GarageType	GarageYrBlt	GarageFinish	GarageCars	GarageArea	GarageQual
1	TA	5	Typ	1	TA	Attchd	1977.0	RFn	2	440	TA
1	Gd	8	Typ	1	TA	Attchd	1970.0	Unf	2	621	TA
1	TA	8	Typ	1	TA	Attchd	1997.0	Unf	2	455	TA
1	TA	7	Typ	1	TA	Attchd	1977.0	RFn	2	546	TA
1	Gd	8	Typ	1	TA	Attchd	1977.0	Fin	2	529	TA



```
data.head()
```

OpenPorchSF	EnclosedPorch	3SsnPorch	ScreenPorch	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
205	0	0	0	0	NaN	NaN	NaN	0	2	2007	WD	Normal	128000
207	0	0	224	0	NaN	NaN	NaN	0	10	2007	WD	Normal	268000
130	0	0	0	0	NaN	NaN	NaN	0	6	2007	WD	Normal	269790
122	0	0	0	0	NaN	MnPrv	NaN	0	1	2010	COD	Normal	190000
0	0	0	0	0	NaN	NaN	NaN	0	6	2009	WD	Normal	215000

* Table size=1168*81

*Total number of rows=1168

* Total number of columns=81

* Float dtype column=3

* Int dtype column=35

* Object dtype column=43

* We had some null values which created the problem to prediction time so we replace this null values with the most frequent one, using imputation technique.

Data Pre-processing Done

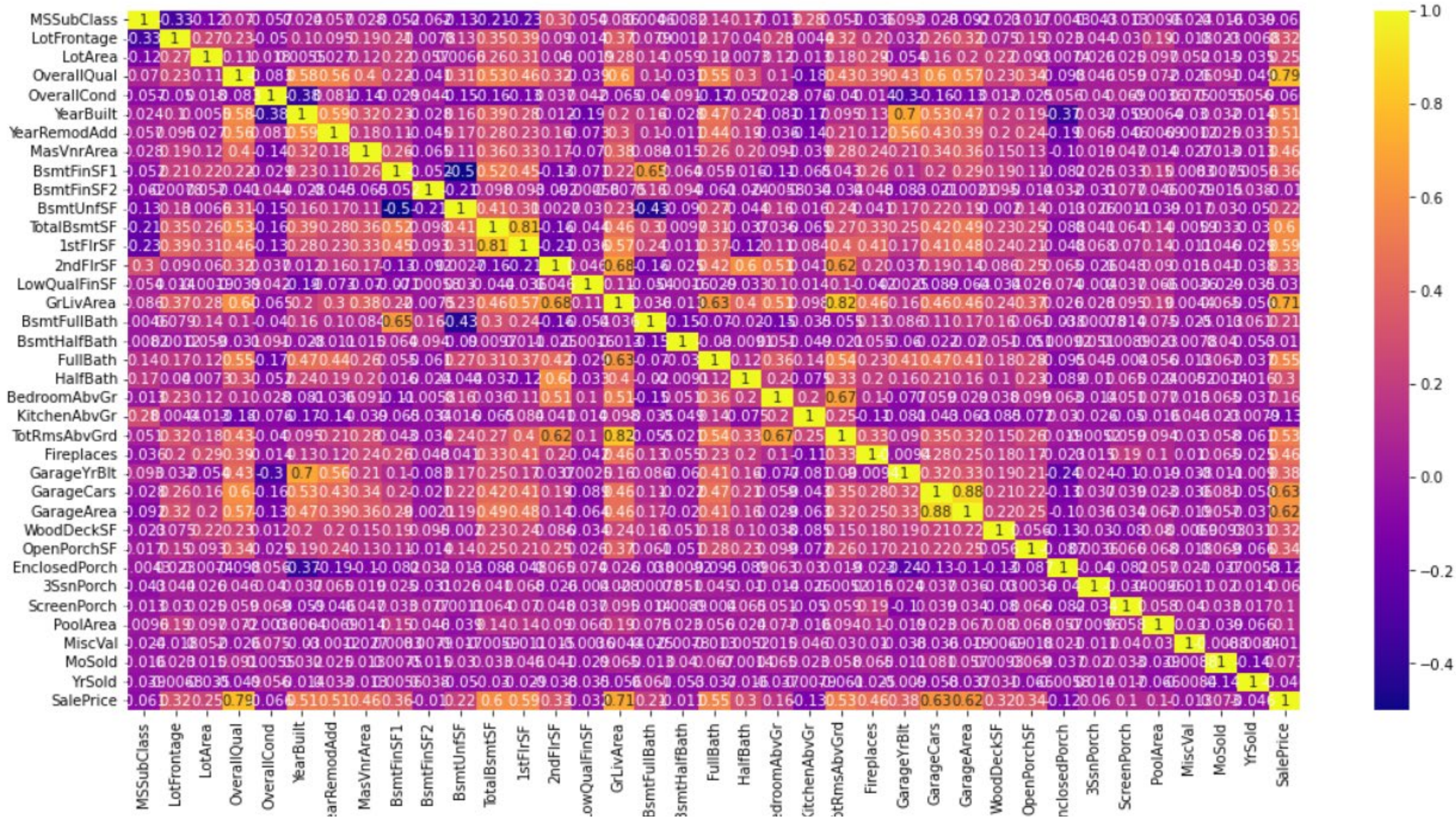
Outliers, Skewness, Data scaling, Variance inflation factor

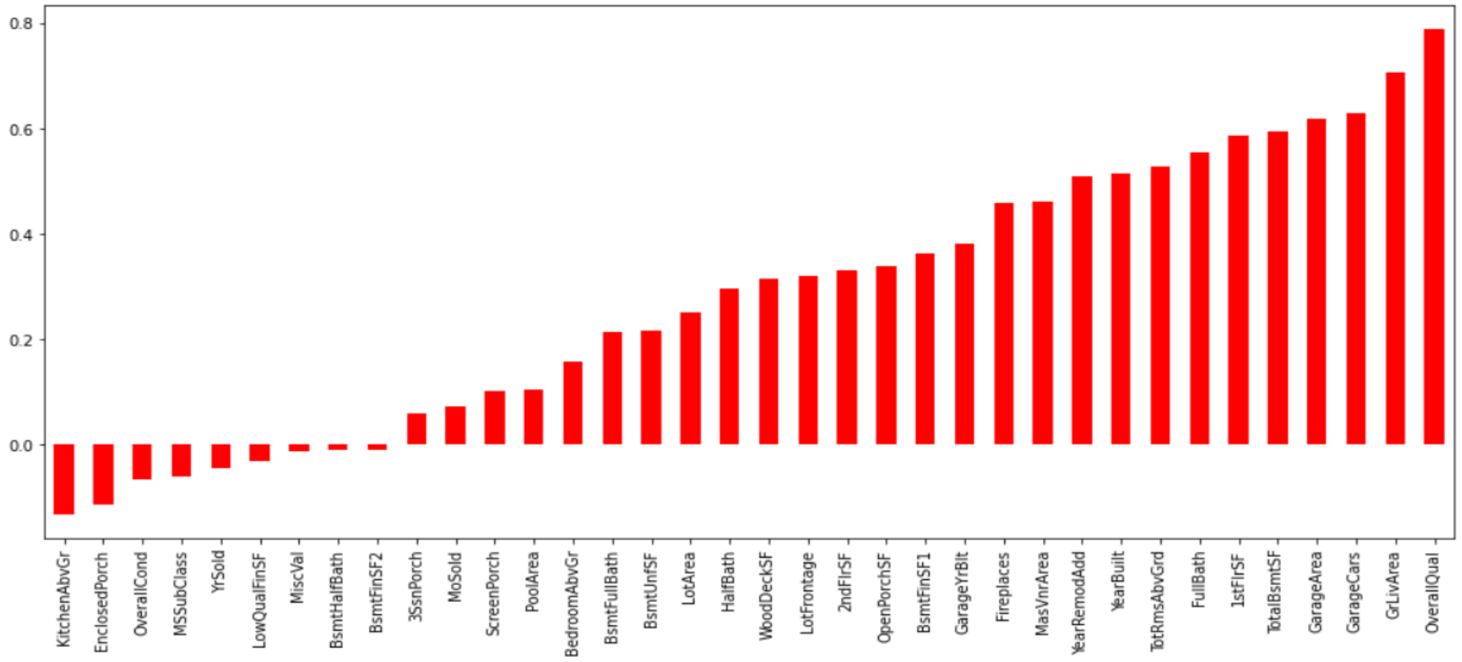
We had outliers in some columns and also skewed data was present in our data set . To Clean the data we have performed following steps.

- For outliers removing we used z score method.
- For skewness removing we used power transformation
- For Data scaling we used Min max scaler.

- We had to drop some columns because many column was providing the similar results for this we used VIF and dropped some columns.

Data Inputs- Logic- Output Relationships





1) Most of the columns are making positive correlation with the target variable excluding kitchenAbvGr, EnclodesPorch, OverallCond, YrSold, LowQualfinSF and MiscVal. MiscVal, bsmtHalfBath and BsmtFinSF2 columns have zero correlation with target variable.

OverallQual, GrLivArea, Garagecars, TotalBSmtSf and 1stFlrsf columns making highly positive correlation with sale_price target column, which can be considered as a strong bond, means if any of these column increases, sale price is also increases.

2) The house price is higher in 2-STORY 1946 & NEWER and 1-STORY 1946 & NEWER ALL STYLES type MSsubclass.

3) The price is average in 2-STORY 1945 & OLDER, 2-1/2 STORY ALL AGES, SPLIT OR MULTI-LEVEL, SPLIT FOYER, DUPLEX - ALL STYLES AND AGES type MSsubclass.

4) House price are very less in PUD - MULTILEVEL - INCL SPLIT LEV/FOYER, 1-1/2 STORY - UNFINISHED ALL AGES type MSsubclass.

5) House prices are very high in gentle slpoe, avg in moderate and less in severe slope.

6) House price are very high in Slightly irregular lotshape, average in regular and Moderately Irregular and very less in Irregular.

7) House price are very high when we talk about 2 story and 1 story house and avg.-less in rest house type.

8) House prices are high when the houses are built with gable and hip style and very less when it is built with flat, shed, gambrel and mansaed type.

9) House prices are very high when we make the foundation with poured contrete, avg in Cinder block & Brick-Tile and less in stone, slab and wood.

10) When the GarageArea is increasing simultaneously Sale price of house is increasing, means both are directly praportional to each other.

11) House prices are high when the quality of kitchen is Excellent and good.

12) When we see the relation between year built and saleprice, We foun that most of the houses are started to built from 1920 but initially the prices was less but we see the price pattern it started to raise from 1990.

13) Totalbasment surface and the total price are directly praportional to each other, if one increases another will also increases.

Hardware and Software Requirements and Tools Used

Anaconda Navigator

Jupyter Notebook

Languge-Python

Many lib.-------

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import warnings
warnings.filterwarnings('ignore')
from sklearn.preprocessing import power_transform
from scipy.stats import zscore
from sklearn.preprocessing import MinMaxScaler
import statsmodels.api as si
from scipy import stats
from statsmodels.stats.outliers_influence import variance_inflation_factor
import sklearn
from sklearn.linear_model import LinearRegression,Lasso,Ridge,ElasticNet
from sklearn.model_selection import train_test_split,GridSearchCV,cross_val_score
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import RandomForestRegressor,AdaBoostRegressor,GradientBoostingRegressor
import xgboost as xg
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score
```

Pandas- For making data frame

Matplotlib and seaborn- For data visualization

Numpy- For numerical python

ZScore- For removing outliers

MInMAxScaler- For data scaling

Power transform- For removing skewness

From metrice - mean_squared_error,mean_absolute_error,r2_score

-For checking the model accuracy, error

Regression- For regression modeling

Ensamble- For boosting and bagging

Grid search cv- For hyperparameter tuning

Cross_Val_Score- For cross validation

Model/s Development and Evaluation

Approaches

Firstly it is import to know about which type of modelling we are going to construct, For this problem we used regression models because our target variable is numerical type and we had to predict the house prices.

When we go for regression we have to use some metrics like mean_squared_error, mean_absolute_error, r2_score

In order to do this work we have to find out the best random state by which we can achieve good accuracy.

Then we split our data set into the train part and test part using train test split.

When we done with the modelling we have to use cross validation for real accuracy(without underfitting and overfitting).

Hyperpameter is must for increasing the model accuracy in order to build good model for this we use Grid search cv.

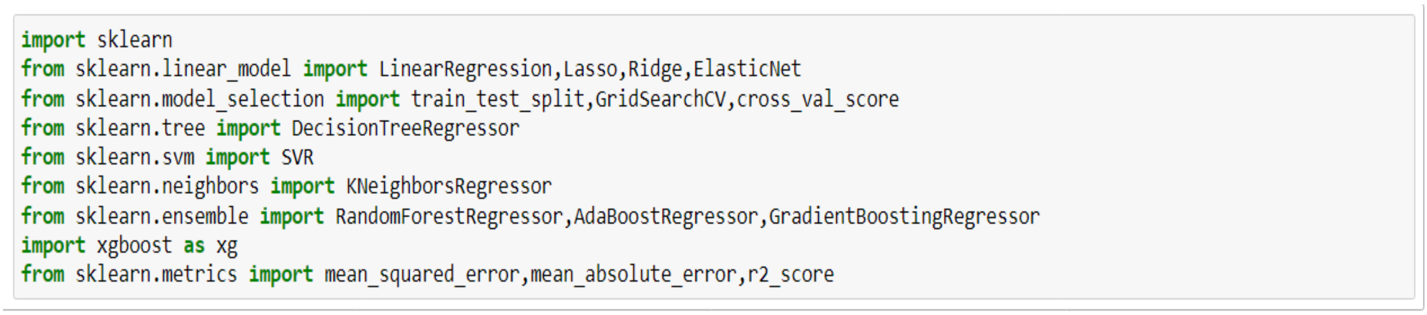
We followed all the above approaches to build our Machine learning model.

Algorithms

- Linear Regression
- Decision Tree Regressor
- KNeighbors Regressor
- Support Vector Regressor

For Bagging and boosting:

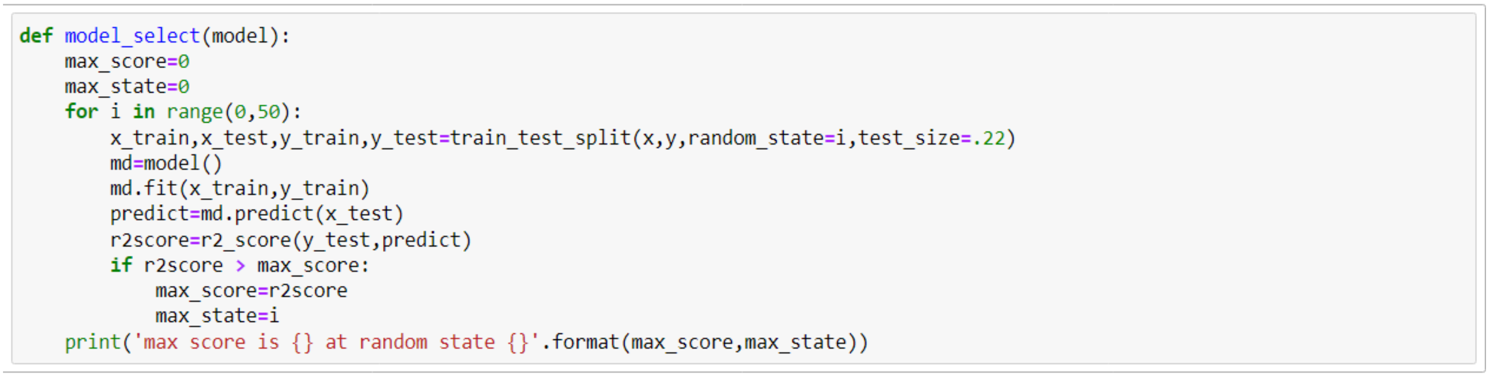
- Random Forest Regresssor
- Gradient Bossting Regressor
- AdaBoost Regressor
- XgBoost Regressor



```
import sklearn
from sklearn.linear_model import LinearRegression,Lasso,Ridge,ElasticNet
from sklearn.model_selection import train_test_split,GridSearchCV,cross_val_score
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import RandomForestRegressor,AdaBoostRegressor,GradientBoostingRegressor
import xgboost as xg
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score
```

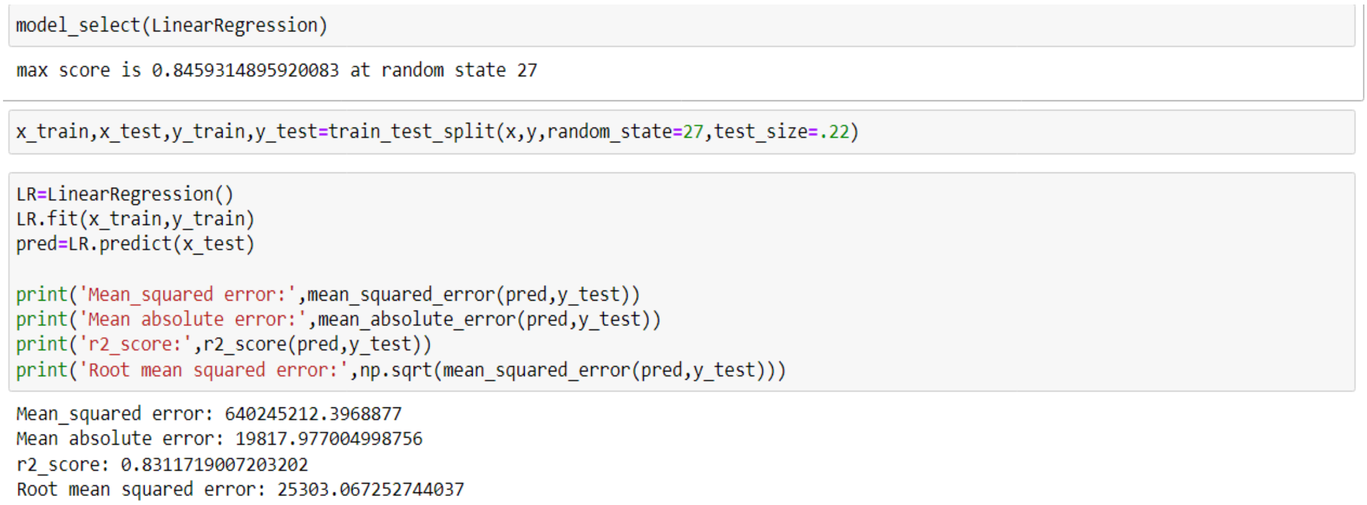
Run and Evaluate selected models and Key Metrics

In order to achieve good accuracy, we find the best random state and then this random state is applied on the train test split.



```
def model_select(model):
    max_score=0
    max_state=0
    for i in range(0,50):
        x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=i,test_size=.22)
        md=model()
        md.fit(x_train,y_train)
        predict=md.predict(x_test)
        r2score=r2_score(y_test,predict)
        if r2score > max_score:
            max_score=r2score
            max_state=i
    print('max score is {} at random state {}'.format(max_score,max_state))
```

Linear Regression



```
model_select(LinearRegression)
```

```
max score is 0.8459314895920083 at random state 27
```

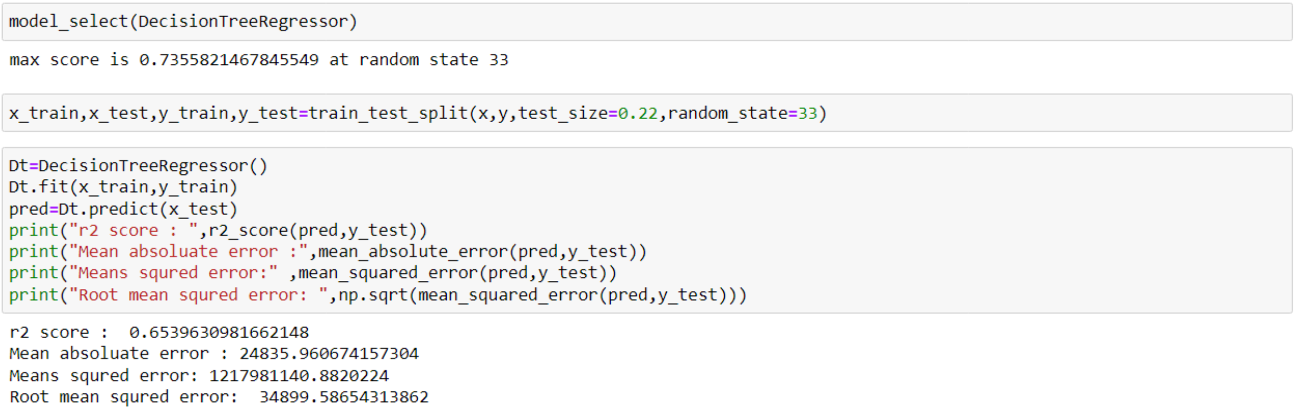
```
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=27,test_size=.22)
```

```
LR=LinearRegression()
LR.fit(x_train,y_train)
pred=LR.predict(x_test)

print('Mean_squared error:',mean_squared_error(pred,y_test))
print('Mean absolute error:',mean_absolute_error(pred,y_test))
print('r2_score:',r2_score(pred,y_test))
print('Root mean squared error:',np.sqrt(mean_squared_error(pred,y_test)))
```

```
Mean_squared error: 640245212.3968877
Mean absolute error: 19817.977004998756
r2_score: 0.8311719007203202
Root mean squared error: 25303.067252744037
```

Decision tree regressor



```
model_select(DecisionTreeRegressor)
```

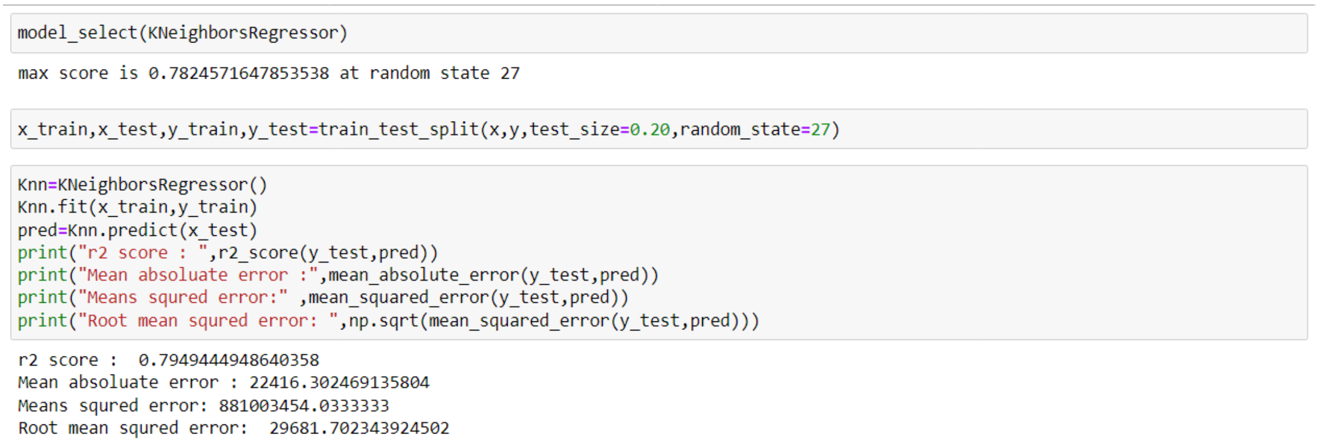
```
max score is 0.7355821467845549 at random state 33
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.22,random_state=33)
```

```
Dt=DecisionTreeRegressor()
Dt.fit(x_train,y_train)
pred=Dt.predict(x_test)
print("r2 score : ",r2_score(pred,y_test))
print("Mean absoluate error :",mean_absolute_error(pred,y_test))
print("Means squred error:" ,mean_squared_error(pred,y_test))
print("Root mean squred error: ",np.sqrt(mean_squared_error(pred,y_test)))
```

```
r2 score :  0.6539630981662148
Mean absoluate error : 24835.960674157304
Means squred error: 1217981140.8820224
Root mean squred error:  34899.58654313862
```

K neighbours regressor



```
model_select(KNeighborsRegressor)
```

```
max score is 0.7824571647853538 at random state 27
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=27)
```

```
Knn=KNeighborsRegressor()
Knn.fit(x_train,y_train)
pred=Knn.predict(x_test)
print("r2 score : ",r2_score(y_test,pred))
print("Mean absoluate error :",mean_absolute_error(y_test,pred))
print("Means squred error:" ,mean_squared_error(y_test,pred))
print("Root mean squred error: ",np.sqrt(mean_squared_error(y_test,pred)))
```

```
r2 score :  0.7949444948640358
Mean absoluate error : 22416.302469135804
Means squred error: 881003454.0333333
Root mean squred error:  29681.702343924502
```

RandomForestRegressor

```
model_select(RandomForestRegressor)
```

```
max score is 0.8680190598809524 at random state 46
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=46)
```

```
rfr=RandomForestRegressor()
rfr.fit(x_train,y_train)
predi=rfr.predict(x_test)
print("r2 score:",r2_score(y_test,predi))
print("Mean absolute error",mean_absolute_error(y_test,predi))
print("Mean squared error:",mean_squared_error(y_test,predi))
print("Root mean squared error:",np.sqrt(mean_squared_error(y_test,predi)))
```

```
r2 score: 0.865207400910482
Mean absolute error 18818.82771604938
Mean squared error: 676879188.3494377
Root mean squared error: 26016.901974474935
```

Adaboost Regressor

```
model_select(AdaBoostRegressor)
```

```
max score is 0.8298096981699499 at random state 1
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.22,random_state=1)
```

```
adaa=AdaBoostRegressor()
adaa.fit(x_train,y_train)
pre=adaa.predict(x_test)
print("r2 score:",r2_score(y_test,pre))
print("Mean absolute error",mean_absolute_error(y_test,pre))
print("Mean squared error:",mean_squared_error(y_test,pre))
print("Root mean squared error:",np.sqrt(mean_squared_error(y_test,pre)))
```

```
r2 score: 0.8303833848389313
Mean absolute error 20247.93814169605
Mean squared error: 659092059.9729753
Root mean squared error: 25672.788317067847
```

GradientBoosting Regressor

```
model_select(GradientBoostingRegressor)
```

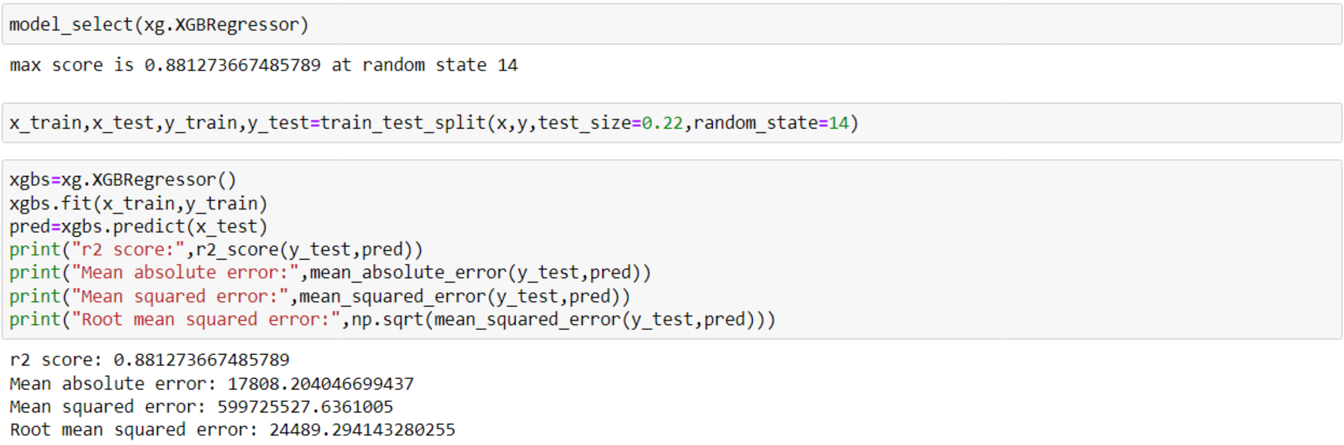
```
max score is 0.8901174002843901 at random state 18
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.22,random_state=18)
```

```
gbr=GradientBoostingRegressor()
gbr.fit(x_train,y_train)
prt=gbr.predict(x_test)
print("r2 score:",r2_score(y_test,prt))
print("Mean absolute error:",mean_absolute_error(prt,y_test))
print("Mean squared error:",mean_squared_error(prt,y_test))
print("Root mean squared error:",np.sqrt(mean_squared_error(prt,y_test)))
```

```
r2 score: 0.8888381082342846
Mean absolute error: 15085.18469907603
Mean squared error: 413916038.7421715
Root mean squared error: 20344.926609407357
```

XGBoost Regressor



```
model_select(xg.XGBRegressor)
```

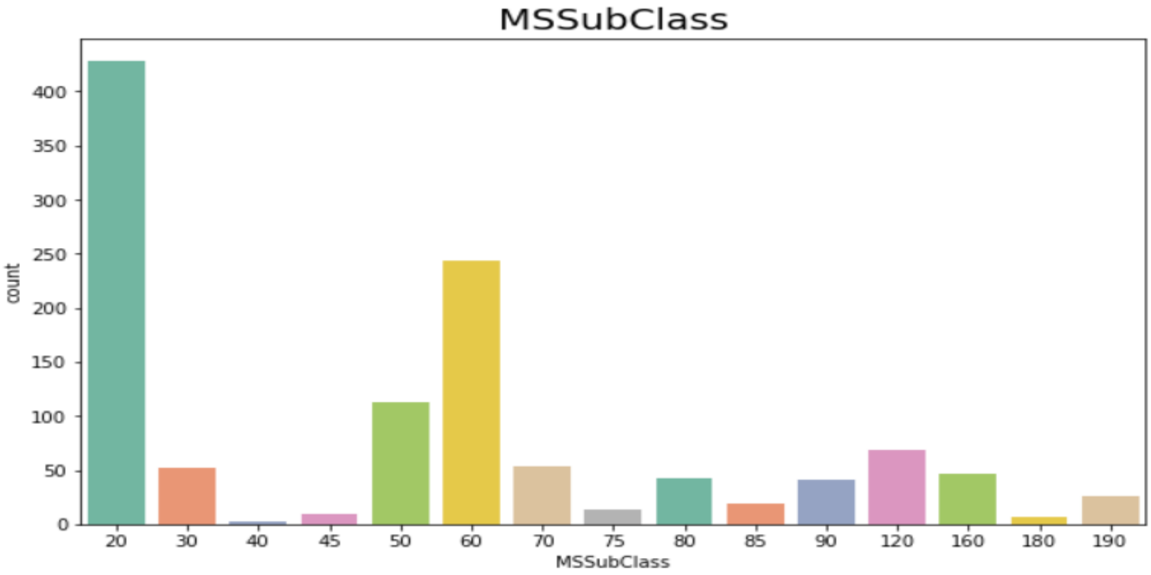
```
max score is 0.881273667485789 at random state 14
```

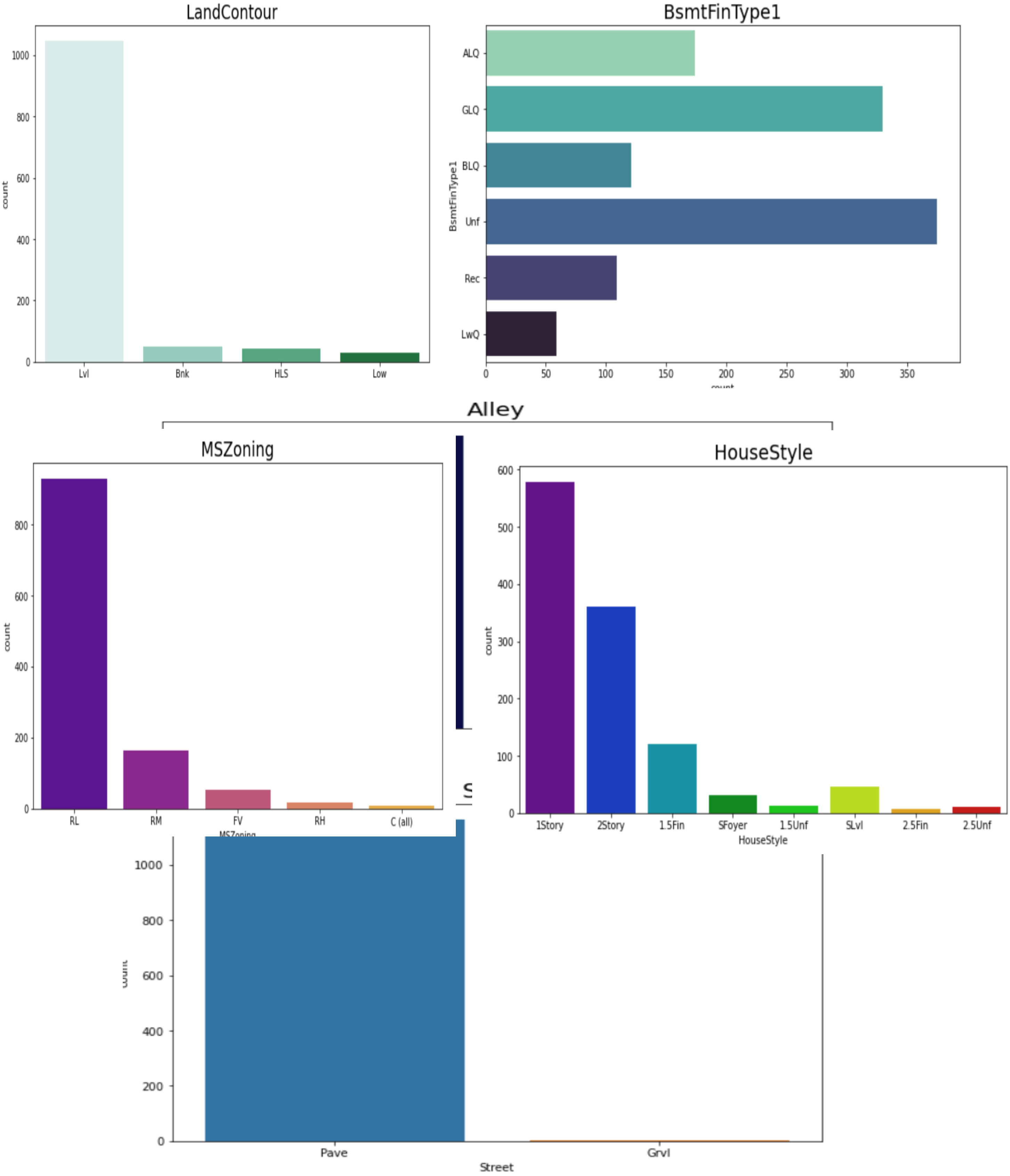
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.22,random_state=14)
```

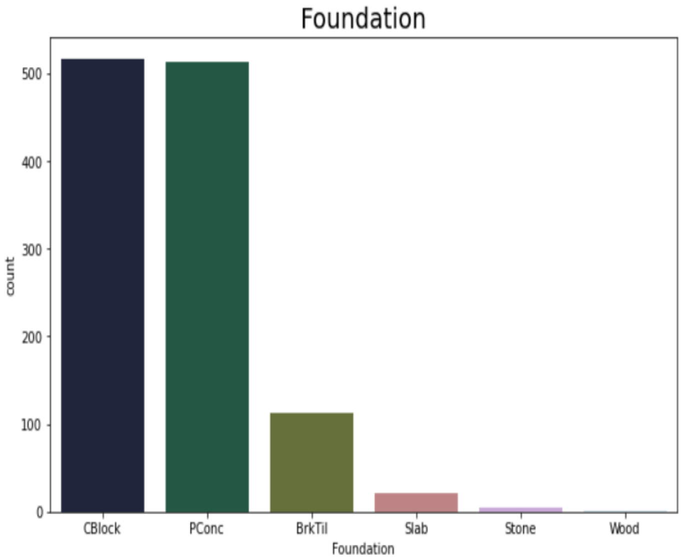
```
xgbs=xg.XGBRegressor()
xgbs.fit(x_train,y_train)
pred=xgbs.predict(x_test)
print("r2 score:",r2_score(y_test,pred))
print("Mean absolute error:",mean_absolute_error(y_test,pred))
print("Mean squared error:",mean_squared_error(y_test,pred))
print("Root mean squared error:",np.sqrt(mean_squared_error(y_test,pred)))
```

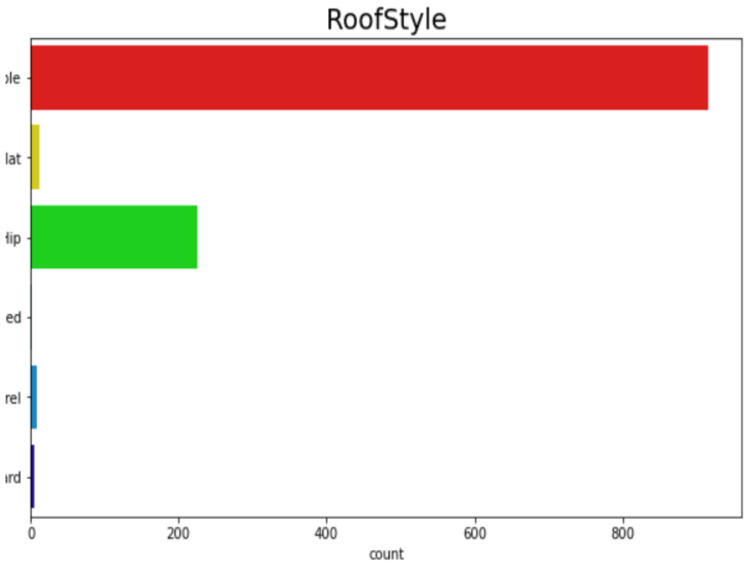
```
r2 score: 0.881273667485789
Mean absolute error: 17808.204046699437
Mean squared error: 599725527.6361005
Root mean squared error: 24489.294143280255
```

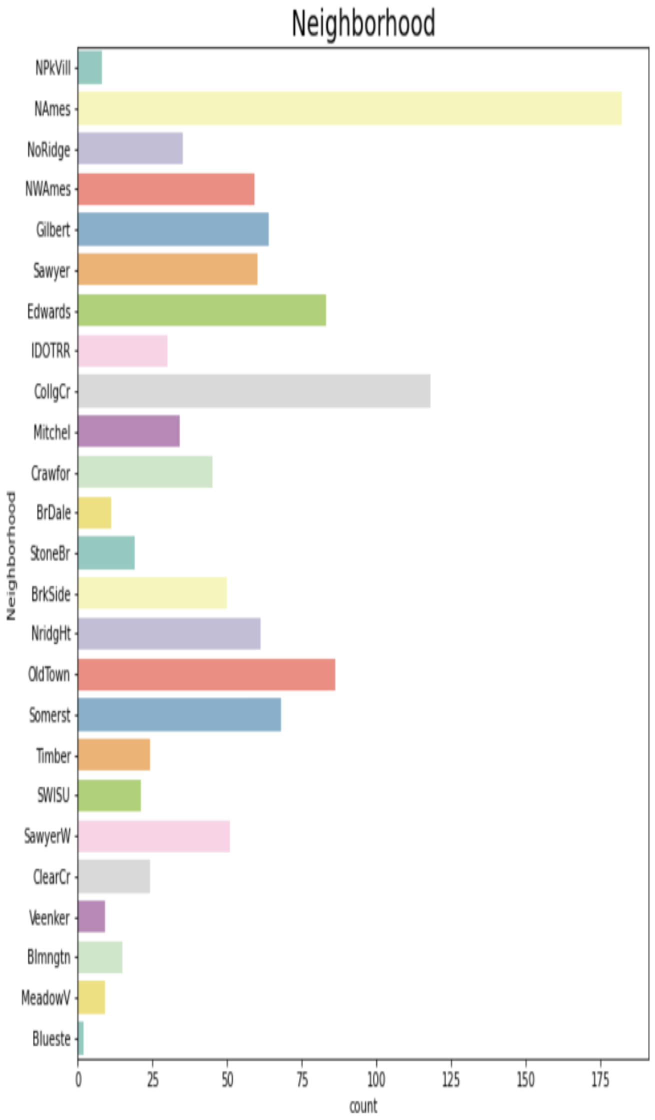
Visualizations

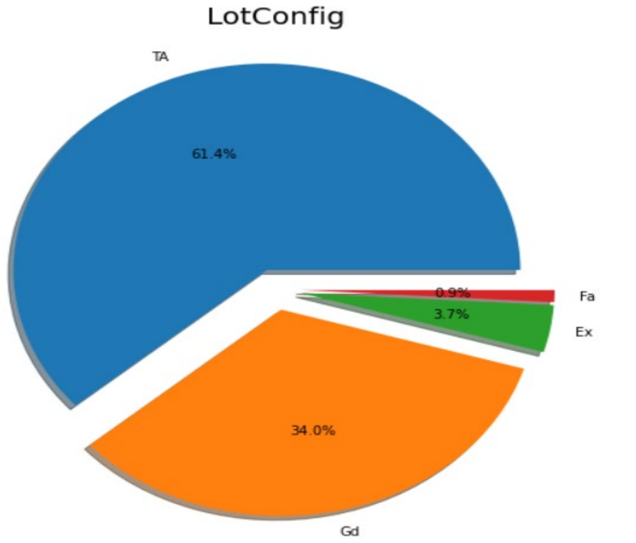


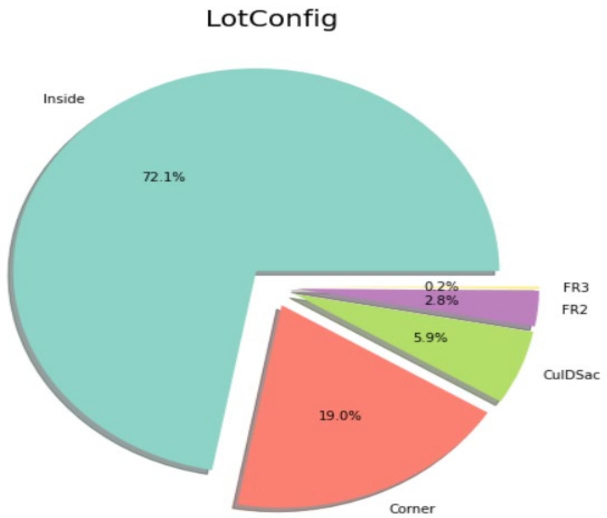

LandContour
count
0
200
400
600
800
1000
Lvl
Bnk
HLS
Low
BsmtFinType1
BsmtFinType1
ALQ
GLQ
BLQ
Unf
Rec
LwQ
0
50
100
150
200
250
300
350
count
Alley
MSZoning
count
0
200
400
600
800
RL
RM
FV
RH
C (all)
MSZoning
HouseStyle
count
0
100
200
300
400
500
600
1Story
2Story
1.5Fin
SFoyer
1.5Unf
SLvl
2.5Fin
2.5Unf
HouseStyle
S
ount
0
200
400
600
800
1000
Pave
Grvl
Street

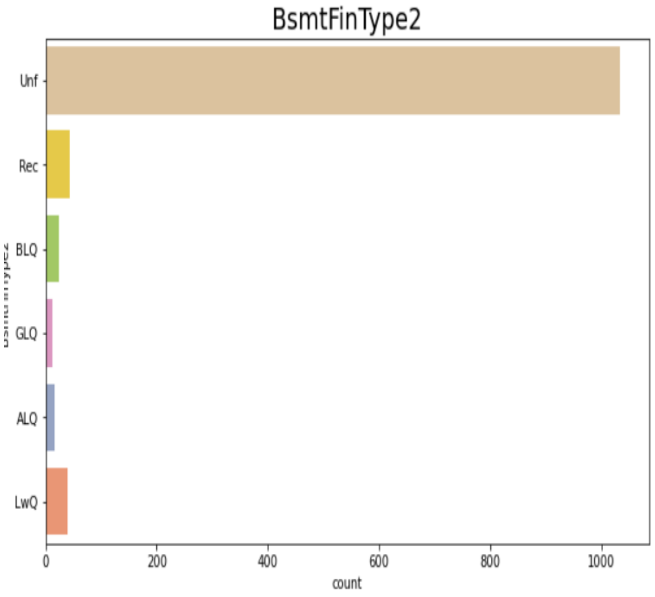

Foundation
500
400
300
200
100
0
count
CBlock
PConc
BrkTil
Slab
Stone
Wood
Foundation

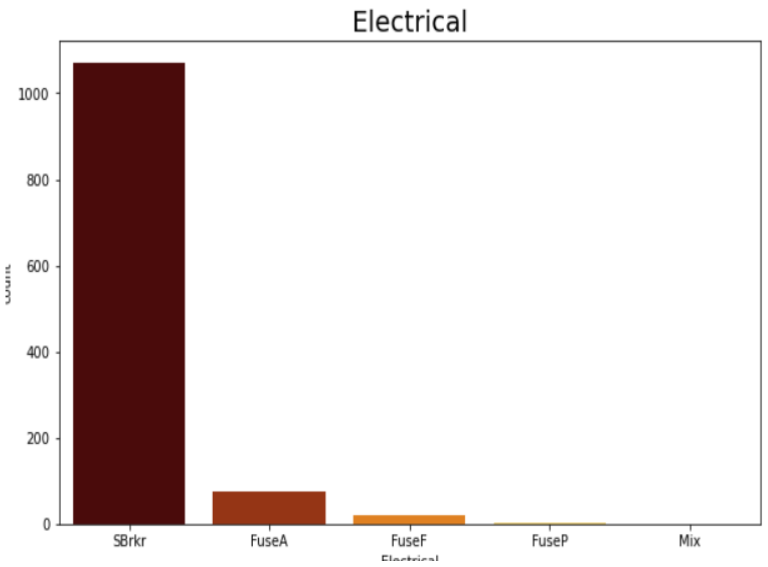

RoofStyle
ble
lat
lip
ed
rel
ard
0
200
400
600
800
count

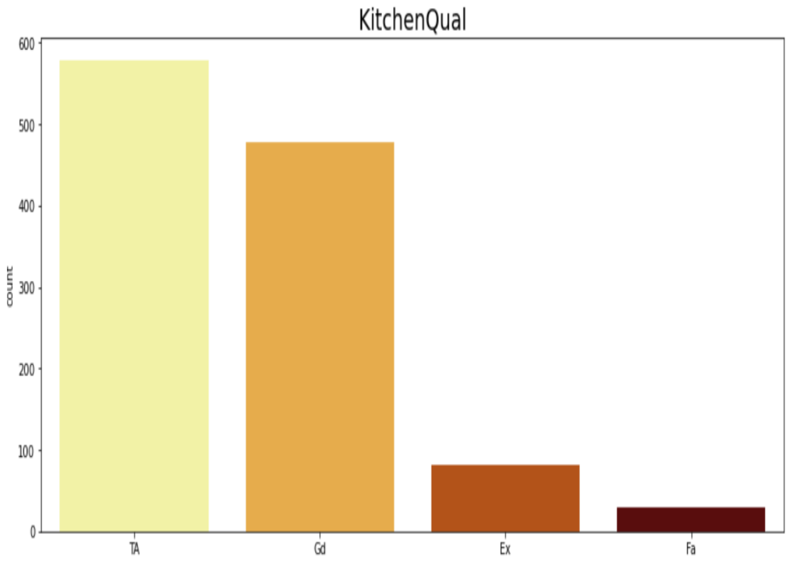

Neighborhood
NPkVill
NAmes
NoRidge
NWAmes
Gilbert
Sawyer
Edwards
IDOTRR
CollgCr
Mitchel
Crawfor
BrDale
StoneBr
BrkSide
NridgHt
OldTown
Somerst
Timber
SWISU
SawyerW
ClearCr
Veenker
Blmngtn
MeadowV
Blueste
Neighborhood
0
25
50
75
100
125
150
175
count

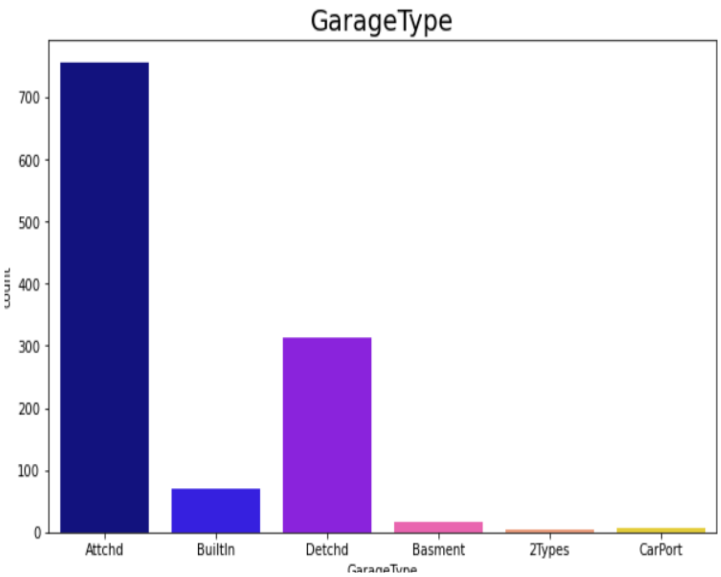

LotConfig
TA
61.4%
0.9%
Fa
3.7%
Ex
34.0%
Gd

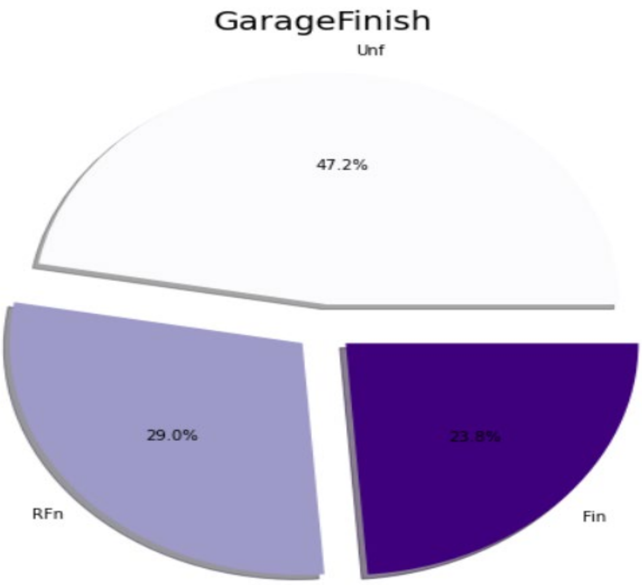

LotConfig
Inside
72.1%
0.2%
FR3
2.8%
FR2
5.9%
CulDSac
19.0%
Corner

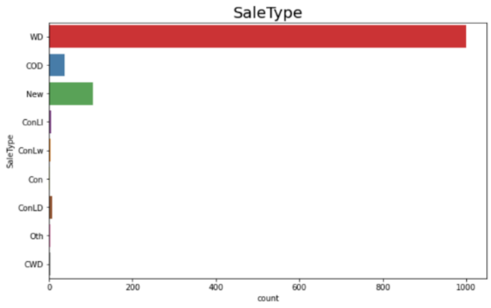

BsmtFinType2
Unf
Rec
BLQ
GLQ
ALQ
LwQ
0
200
400
600
800
1000
count


Electrical
1000
800
600
400
200
0
SBrkr
FuseA
FuseF
FuseP
Mix









1) In the MSSUbClass column the STORY 1946 & NEWER ALL STYLES type house(20) is present highest number of times in the column, The counting of 2-STORY 1946 & NEWER house(60) is also high, rest all the house types are less present and having less number of counts.

2) If we see the MSZoning column, we will find that most of the houses has been built in ths Residential Low Density(RL) Zoning and very less houses are build in the Agriculture, Commercial, Floating Village Residential, Industrial, Residential High,Density, Residential Low Density Park and Residential Medium Density MSzoning.

3) Pave type road is being used in the street and gravel type alley is being used to access the property.

4) In our data plenty of the house's landcontour are Lvl(Near Flat/Level) and bnk(Banked), HLS(Hillside) and Low(Depression) type landcounter are very less.

5) Lot configuration is the one of the most important factor for deciding the house price.Due to less avilibility of frontage on 2 and 3 side property and high supply of insight and corner lot the houses are built inside in most of the cases and very less houses are built in FR2 and FR3 configuration.

6) Around 570 houses are one story, having same housestyle, 370 houses are One and one-half story building wherin 2nd levelis finished reflects different housestyle, approximately 30 houses are Split Foyers, 20 are One and one-half story wherein 2nd level is finished. 40 around houses are Split Level house. Rest of the housestyles which are very less in numbers are Two and one-half storys wherein 2nd level is finished in some and unfinished in others.

7) More than 800 houses have gable roof, 200 around have hip roof and very small number has got flat roofs.

8) Approximately 1130 houses are bulit of using Standard (Composite) Shingle, and 10 of Gravel & Ta, less than 10 are made up of using wood Shakes and Wood Shingles.

9) Condition of the material of 1000 houses is Average/Typical, of 180 houses is good and approximately 30 around houses are in fine condition.

10) 500 houses's foundation is made up of Cinder Block and another 500 around houses' foundation is made up of Poured Contrete. Foundation of 100 houses are build of Brick & Tile. A few numbers are there which are made up of slab, stones and wood.

11) 60 basements are with good living qurаters, 110 are with average rest room, more than 350 are unfinished, approxamtely 120 are with Below Average Living Quarters, around 320 are with Good Living Quarters, and 160 aroud are with Average Living Quarters.

12) More than 1000 basement areas are Unfinshed. 50 around numbers of basements have Average Rec Room and Low Quality. 50 around basements have Good Living Quarters, and Average Living Quarters, also little number of basemnets have Below Average Living Quarters.

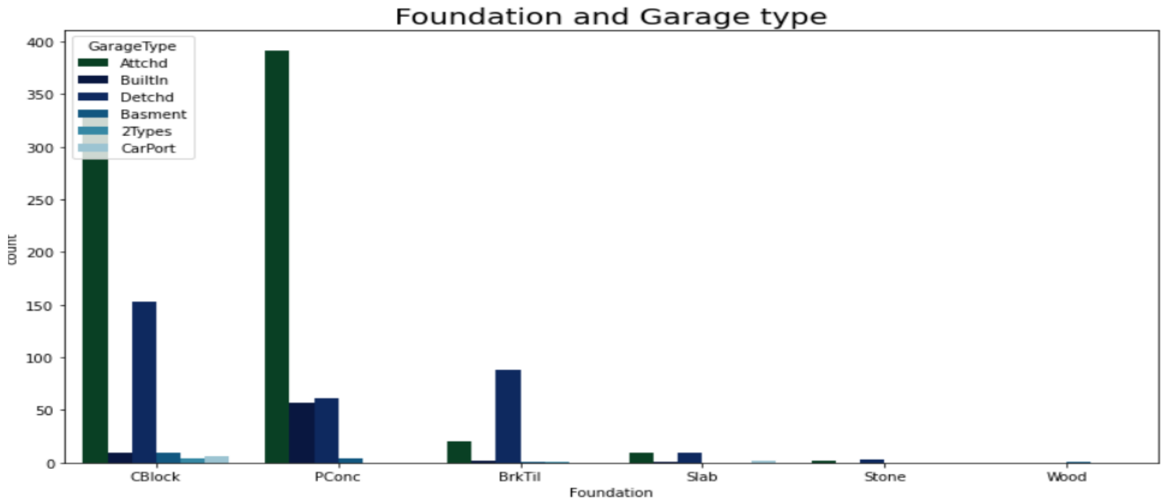
13) Approxately 1150 numbers have Gas forced warm air furnace, while the houses that haveGas hot water or steam heat are negligible.

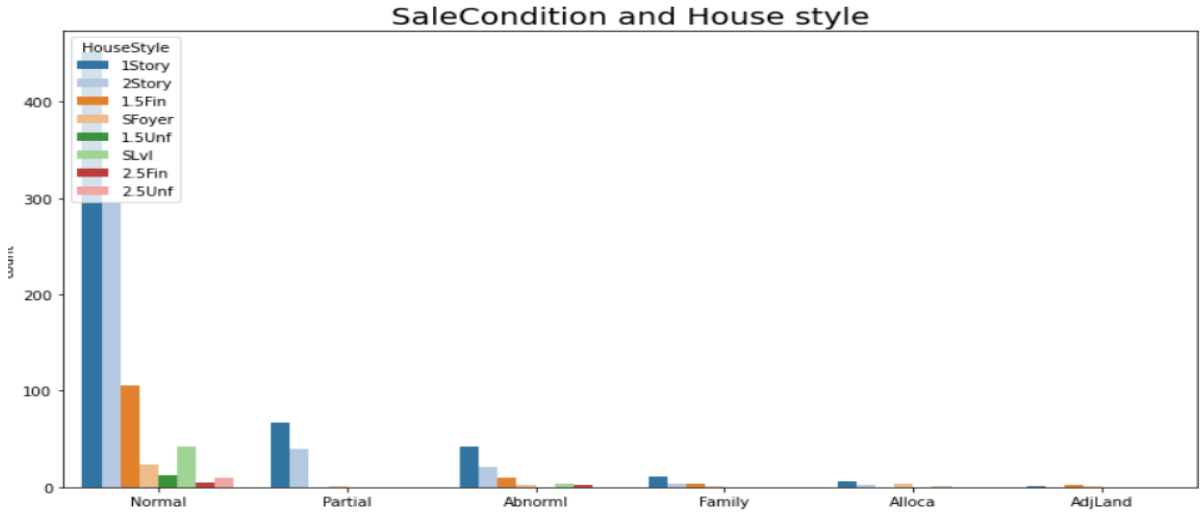
14) More than 1000 houses have Standard Circuit Breakers & Romex. The houses that have Fuse Box over 60 AMP and all Romex wiring (Average) and 60 AMP Fuse Box and mostly Romex wiring (Fair) are very little in number.

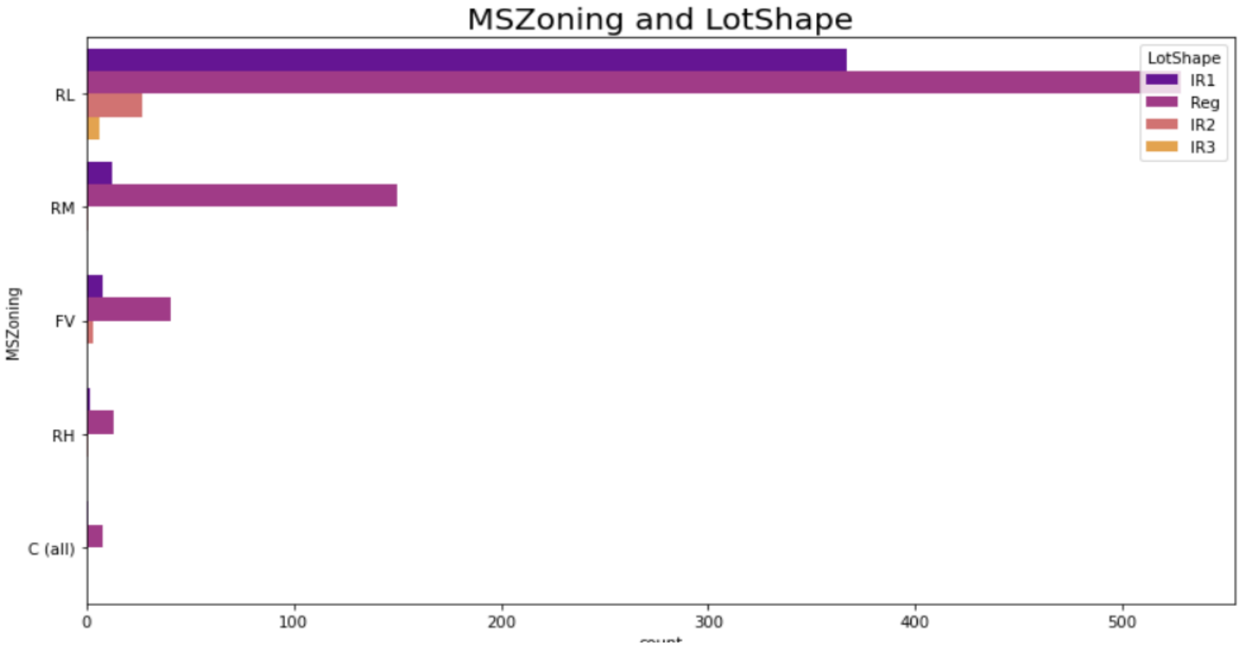
15) In most of the houses (around 750) garage is attched and in 300 aroud houses garage is detched.

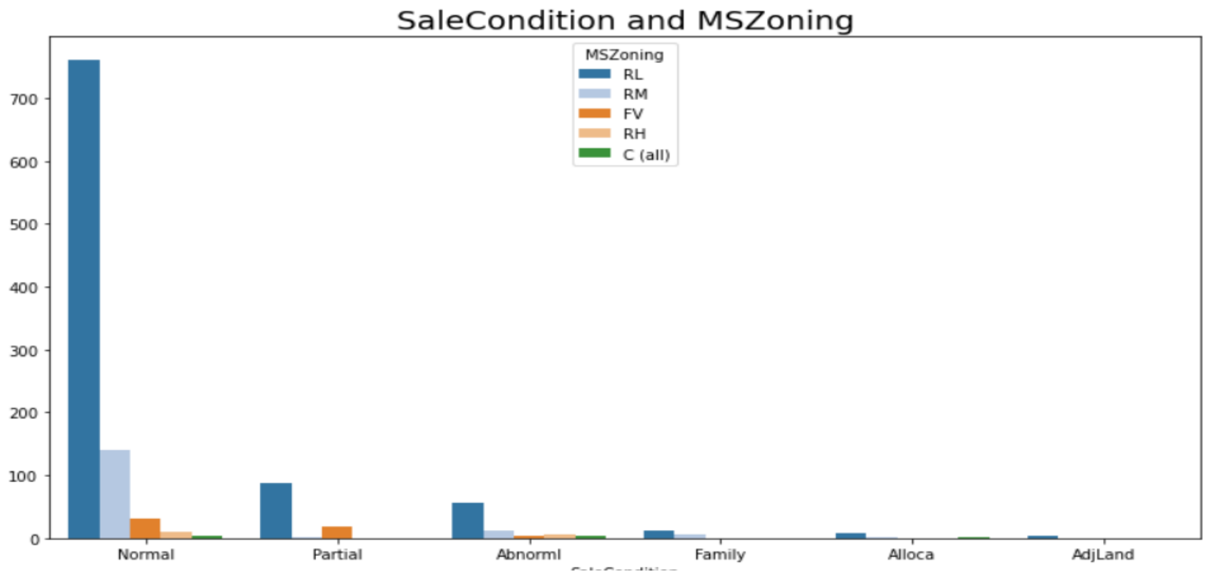
16) Around 570 houses' kicthens are typical average. 460 are good. 90 has got Excellent quality and very less are left with fair quality.

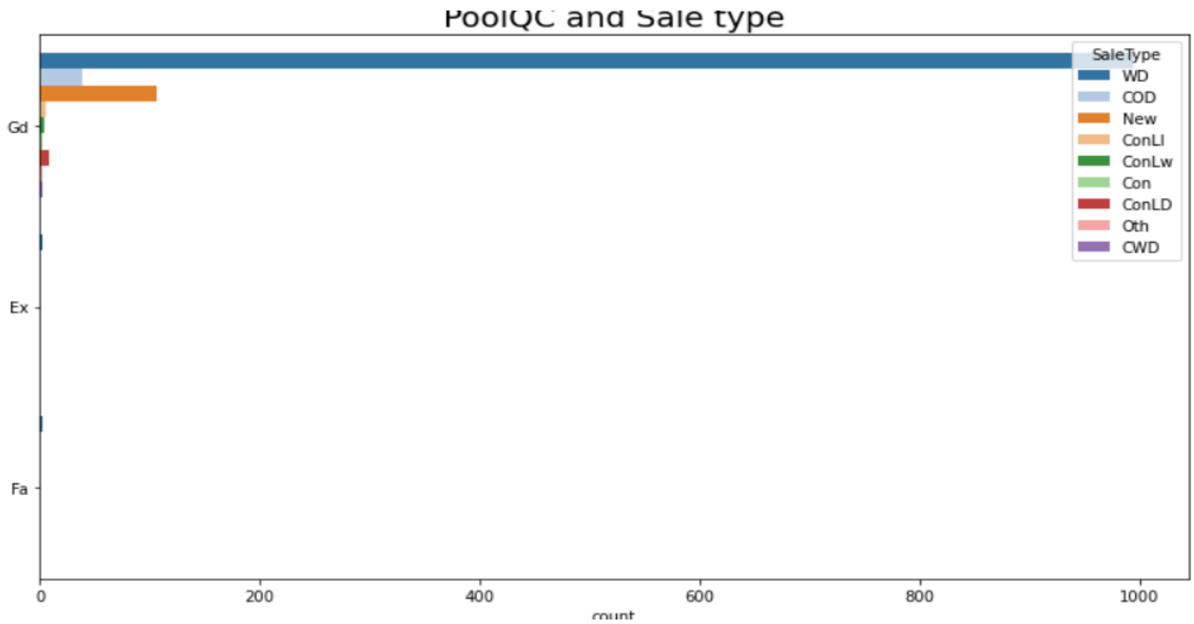
17) Warranty Deed - Conventional is very high in numbers, the Homes that just constructed and sold, are less in numbers. Around 30 houses are being sold though Court Officer Deed/Estate, while other sale types are negligible in numbers.

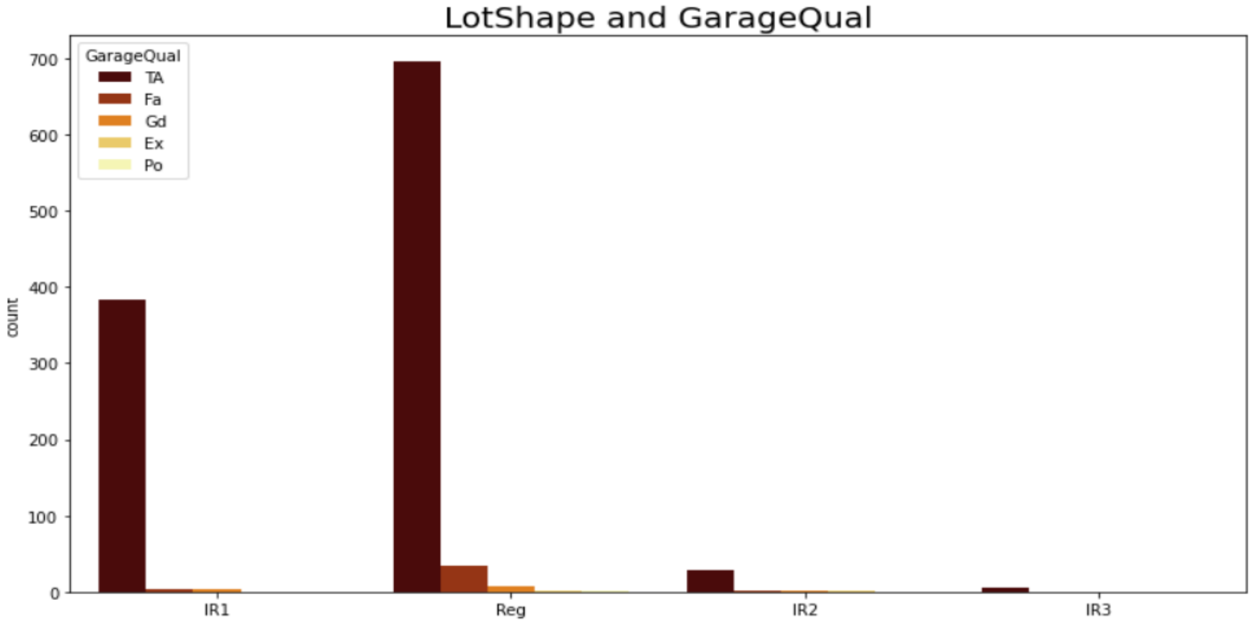

Foundation and Garage type
GarageType
Attchd
BuiltIn
Detchd
Basment
2Types
CarPort
count
0
50
100
150
200
250
300
350
400
CBlock
PConc
BrkTil
Slab
Stone
Wood
Foundation

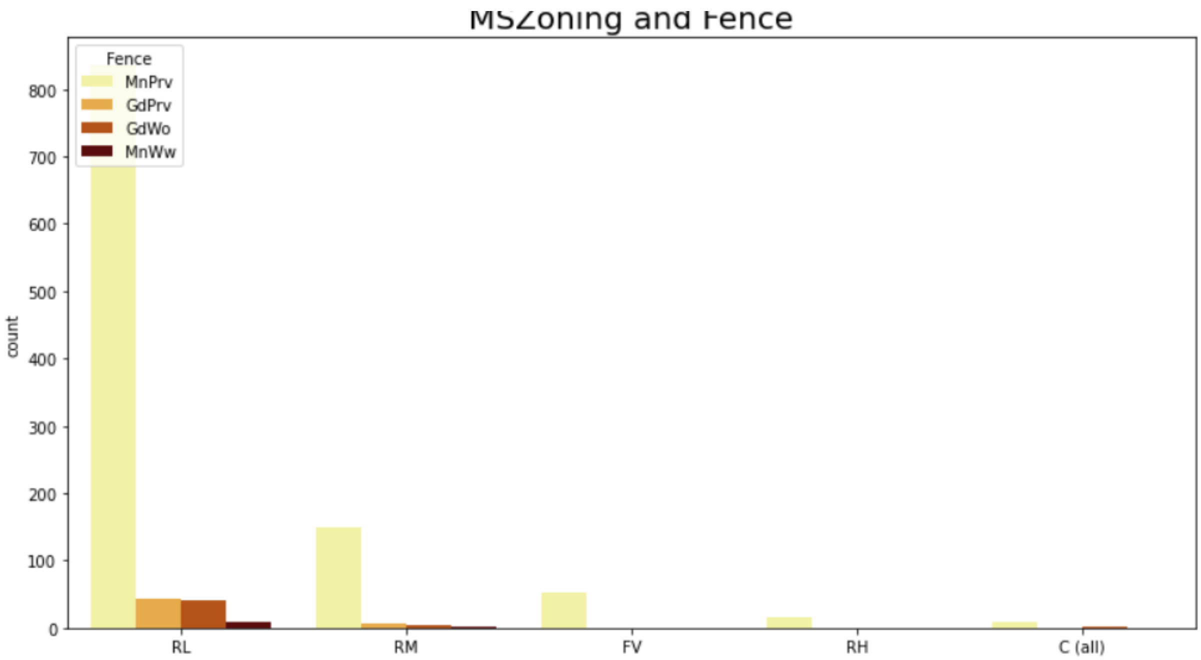

SaleCondition and House style
HouseStyle
1Story
2Story
1.5Fin
SFoyer
1.5Unf
SLvl
2.5Fin
2.5Unf
count
0
100
200
300
400
Normal
Partial
Abnorml
Family
Alloca
AdjLand

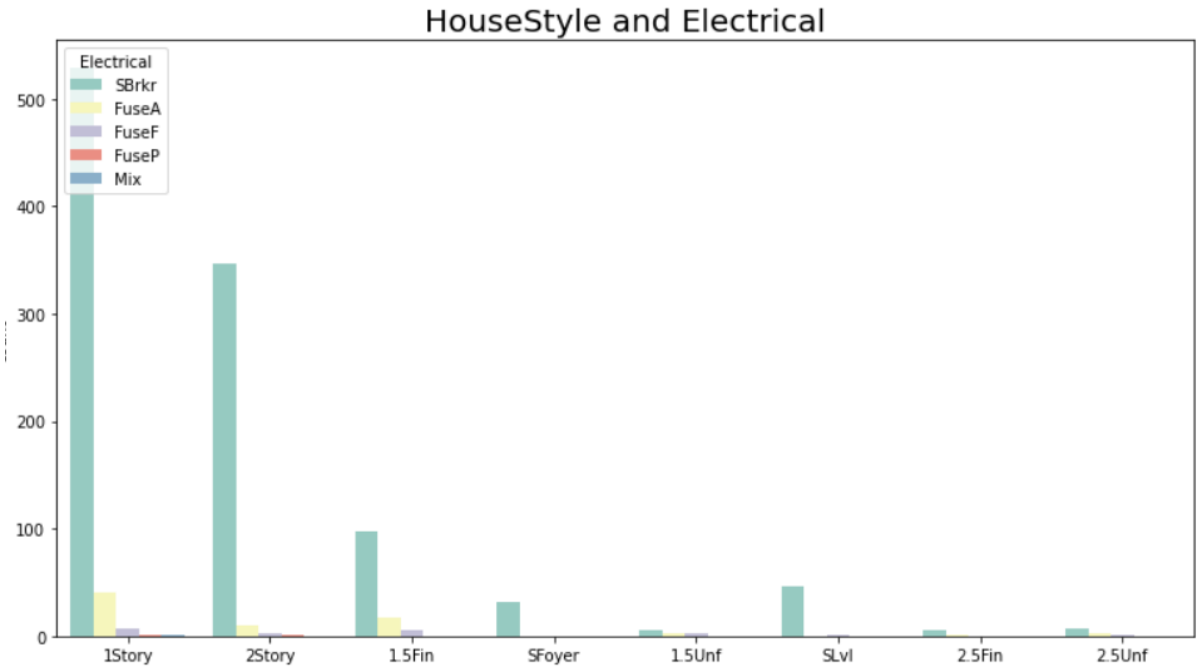

MSZoning and LotShape
LotShape
IR1
Reg
IR2
IR3
MSZoning
RL
RM
FV
RH
C (all)
0
100
200
300
400
500
count

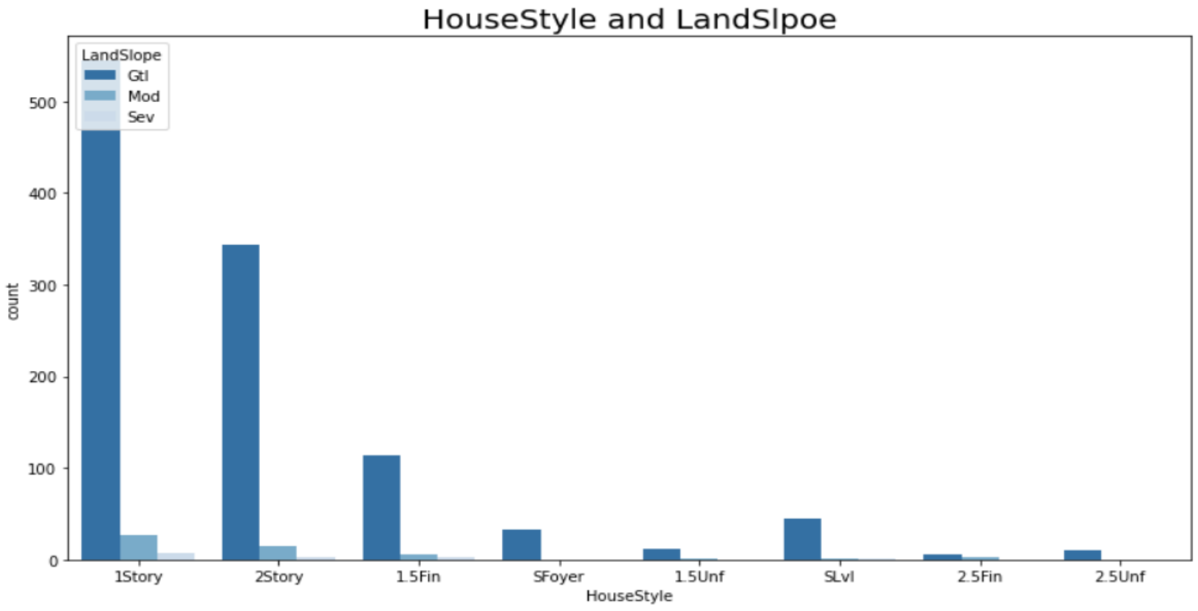

SaleCondition and MSZoning
MSZoning
RL
RM
FV
RH
C (all)
0
100
200
300
400
500
600
700
Normal
Partial
Abnorml
Family
Alloca
AdjLand
SaleCondition


PoolQC and Sale type
SaleType
WD
COD
New
ConLI
ConLw
Con
ConLD
Oth
CWD
Gd
Ex
Fa
0
200
400
600
800
1000
count


LotShape and GarageQual
GarageQual
TA
Fa
Gd
Ex
Po
count
0
100
200
300
400
500
600
700
IR1
Reg
IR2
IR3


MSZoning and Fence


HouseStyle and Electrical


HouseStyle and LandSlpoe

1) HouseStyle and LandSlpoe- Gentle slope land is the common most slope, present in all the house style.

2) Foundation and Garage type- In the cinder block and poured contrete type fundation attched garage is being used in high amount and then detched garage.In the Brick & Tile type foundation detched garage is being used in high amount and other garage is uesd in very less amount.

3) MSZoning and LotShape- in the Residential Low Density Zone the lot shape is regular type in most of the cases and then Slightly irregular lot shape has been built, Moderately Irregular and Irregular lot shape is used in few number of houses only. In the conclusion regular type lot shape is found in all the available Mszoning.

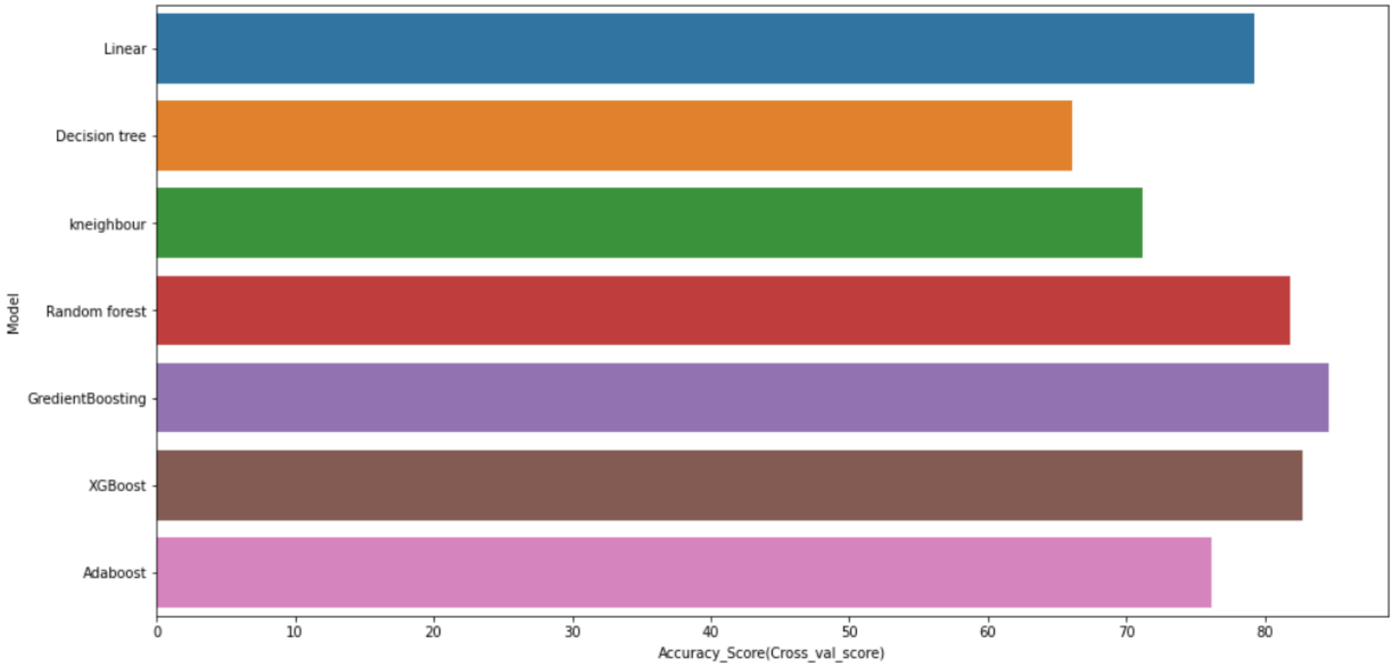
4) PoolQC and Sale type- When good pool quality is good, house sale type is Warranty Deed-Conventional(WD) and higher in numbers, rest all the sale type is very less.

5) LotShape and GarageQual- Typical/Average garage quality is higher in all type of lot shape, In the Moderately Irregular, Irregular lot shape only Typical/Average lot shape is avilable and in the regular lot shape, garage quality fair and good is avilabe but in the negligible amount and again Typical/Average garage quality is higher in regular lot shape.

6) MSZoning and Fence- In the Residential Low Density Zone, Minimum Privacy fence is avilable in huge amount and very less availability of good privacy, good wood and minimum wood type fencing And alos we can say that in all the zoning mimimum privacy type fence is present.

CONCLUSION

From the above results :



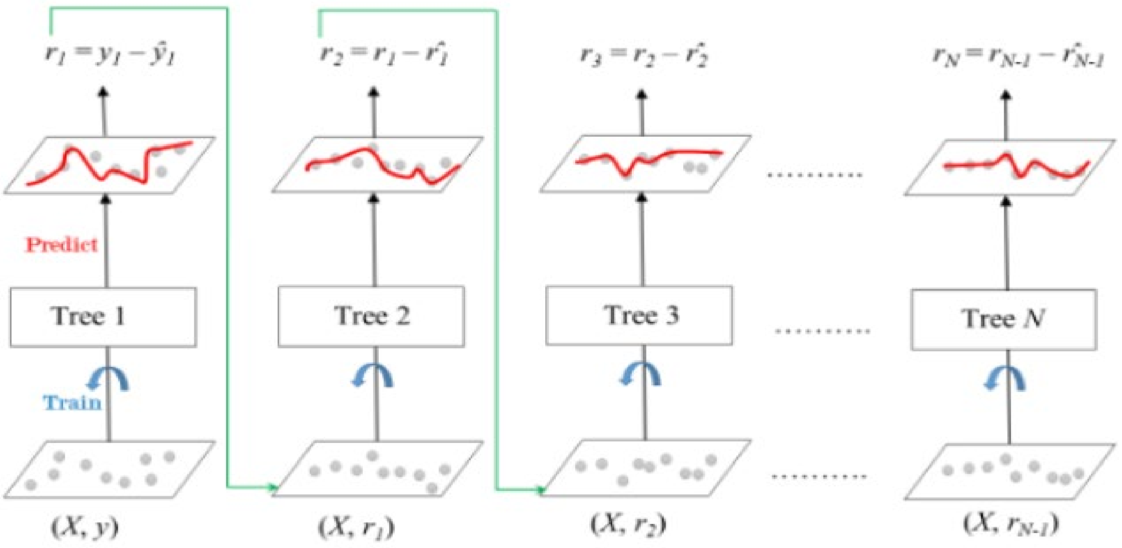
Best model–

Model name: Gradient Boosting Regressor
r2 score: 0.8634
Mean absolute error: 17808.20
Mean squared error: 599725527.63
Root mean squared error: 24489.29

We got 86.34 % accuracy, which can be considers as a good accuracy for predicting the house price.

About the Gradient boosting regressor

Gradient Boosting Regression is an analytical technique that is designed to explore the relationship between two or more variables (X, and Y). Its analytical output identifies important factors (X_i) impacting the dependent variable (y) and the nature of the relationship between each of these factors and the dependent variable.
Gradient Boosting Regression is limited to predicting numeric output so the dependent variable has to be numeric in nature. The minimum sample size is 20 cases per independent variable



Gradient Boosted Trees for Regression

In this paper, we built serveral regression models to predict the price of some house given some of the house features. We eveluated and compared each model to determine the one with highest performance. We also looked at how some models rank the features according to their importance. In this paper, we followed the data science process starting with getting the data, then cleaning and preprocessing the data, followed by exploring the data and building models, then evaluating the results and communicating them with visualizations.

As a recommendation, we advise to use this model (or a version of it trained with more recent data) by people who want to buy a house in the area covered by the dataset to have an idea about the actual price. The model can be used also with datasets that cover different cities and areas provided that they contain the same features. We also suggest that people take into consideration the features that were deemed as most important as seen in the previous section; this might help them estimate the house price better.