Housing Price Prediction Project

Presented by-
Sankalp Mahapatra
Internship-29

TABLE OF CONTENTS

- INTRODUCTION

  Business Problem Framing

  Business goal
- Motivation for the Problem undertaken
- Exploratory Data Analysis (EDA)
- Data Visualization
- Model/s Development and Evaluation
- Hyperparameter tuning using GridSearchCV
- Conclusion

INTRODUCTION

- Thousands of houses are sold every day. There are some questions every buyer asks himself like: What is the actual price that this house deserves? Am I paying a fair price? Also Is it the location? Is it the overall quality of the house? Is it the size? Could it be sold at a good price in future? All these questions come in to our mind when we decide to purchase a house.

- In this study, a machine learning model is proposed to predict a house price based on data related to the house (its size, the year it was built in, etc.). During the development and evaluation of our model, we will show the code used for each step followed by its output. This will facilitate the reproducibility of our work.

Business Problem Framing

- 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.

- The project endeavors to extensive data analysis and implementation of different machine learning techniques in python for having the best model with most important features of a house on insight of both business value and realistic perspective.

Business goal

- With the help of available independent variables, we need to model the price of the houses. 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.

Motivation for the Problem undertaken

- House prices increase every year, so there is a need for a system to predict house prices in the future. House price prediction can help the developer determine the selling price of a house and can help the customer to arrange the right time to purchase a house.

- The problem statement is related to the 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 company is looking at prospective properties to buy houses to enter the market. It is 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 house?
    - How do these variables describe the price of the house?

- In this section, we evaluate widely used regression technologies like Linear Regression, regularization, bagging and boosting and many more ensemble techniques to predict the house sale price result.

Exploratory Data Analysis (EDA)

These are the different steps I have undertaken to perform the Exploratory Data Analysis:

- Importing the necessary dependencies and libraries.
- Reading the CSV file and converted into data frame.
- Checking the data dimensions for the original dataset.
- Looking for null values and accordingly fill the missing data.
- Checking the summary of the dataset.
- Checking unique values and plotting the value counts of each unique value of each column.
- Checking all the categorical columns in the dataset.

- Visualizing the relation between the features and label and relation between the features i.e. finding multicolinearity by using matplotlib and seaborn.

- Performing encoding using the ordinal encoder on categorical features.

- Checking for co-relation/multi-colinearity in a heat map.

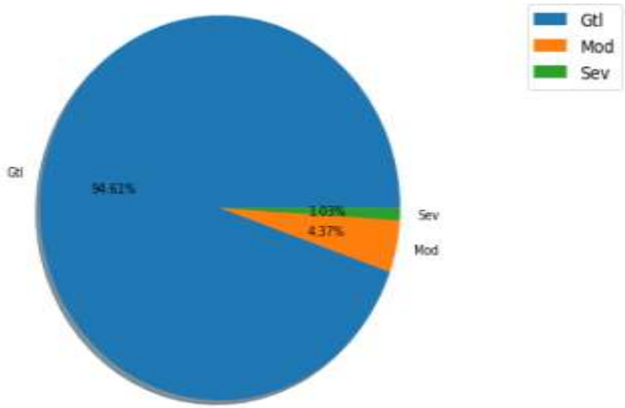
- Checking for Outliers/Skewness using box plot and distribution plot.

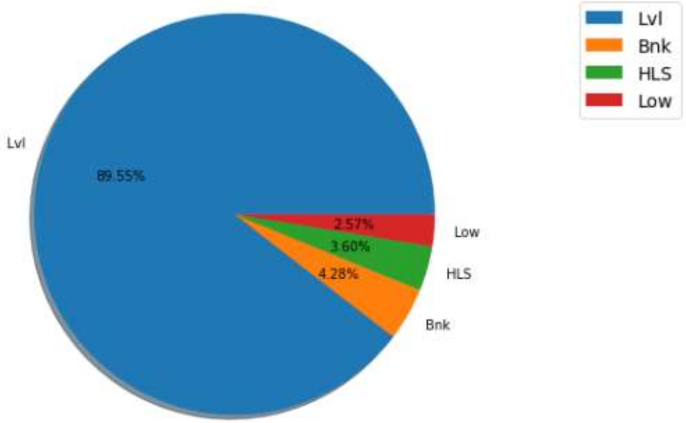
- Checking how the data is distributed in all the features.

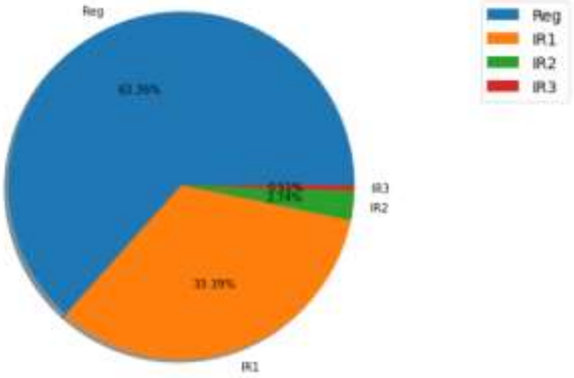
- Perform Scaling using Standard Scaler method.

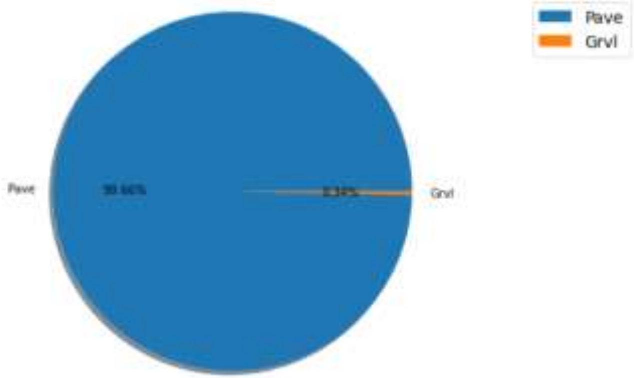
- Used Vif to cross check if Multicolinearity still exists in the dataset.

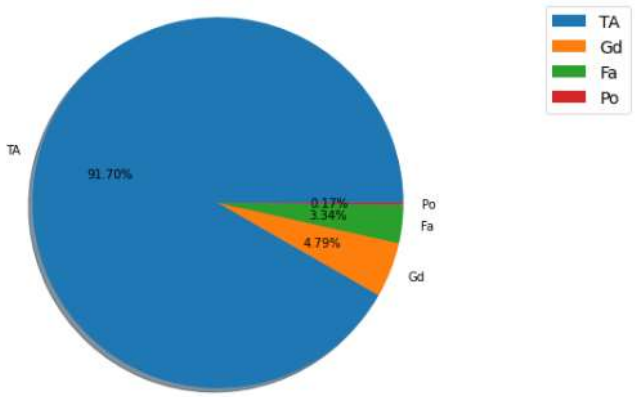
Data Visualization

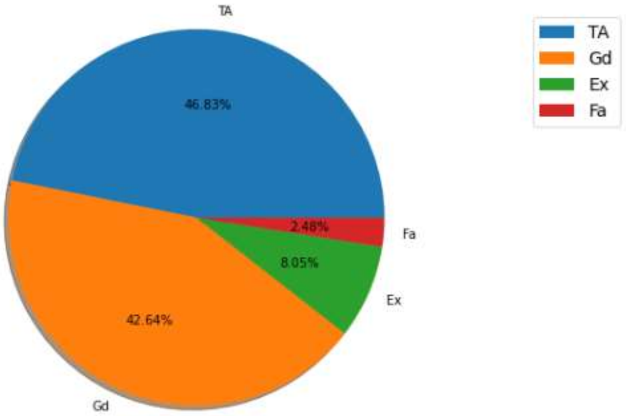


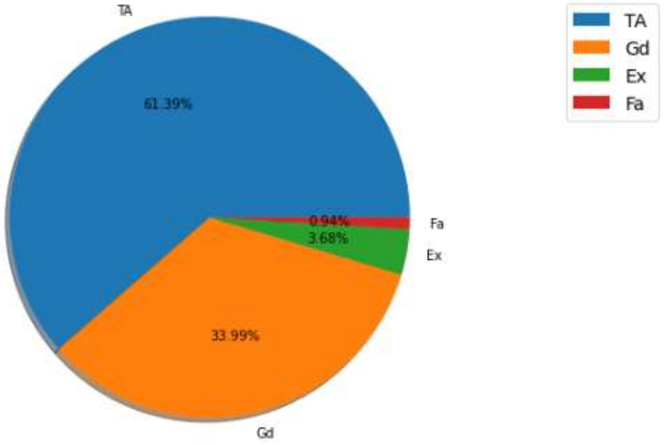


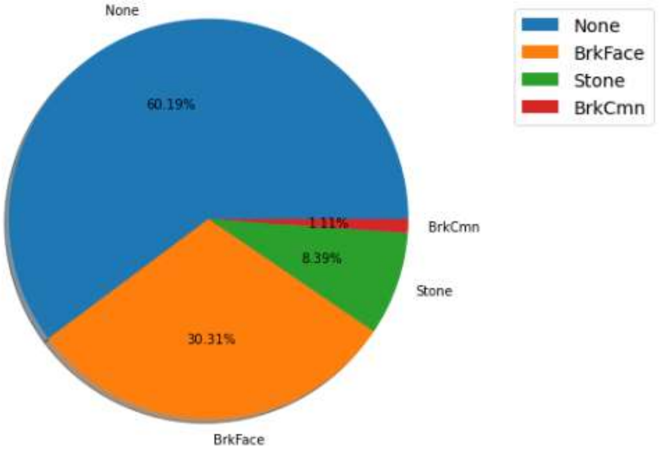


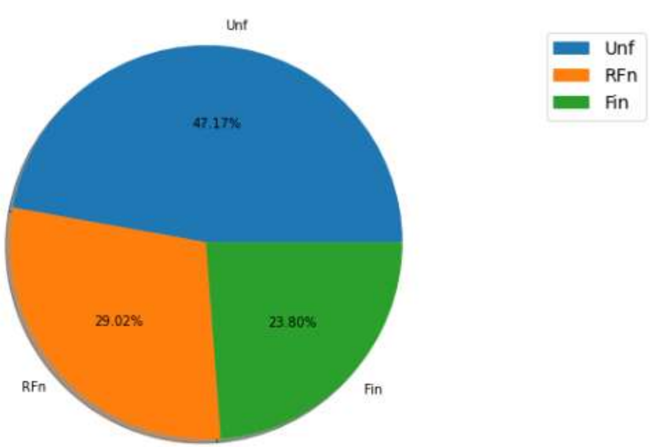


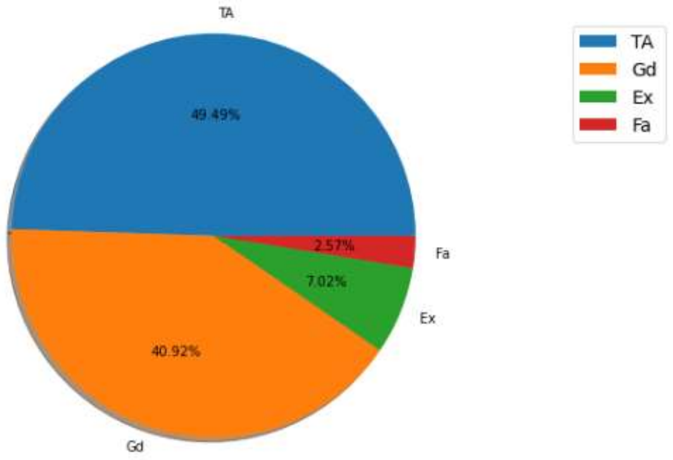

TA
Gd
Fa
Po
TA
91.70%
0.17%
3.34%
4.79%
Po
Fa
Gd

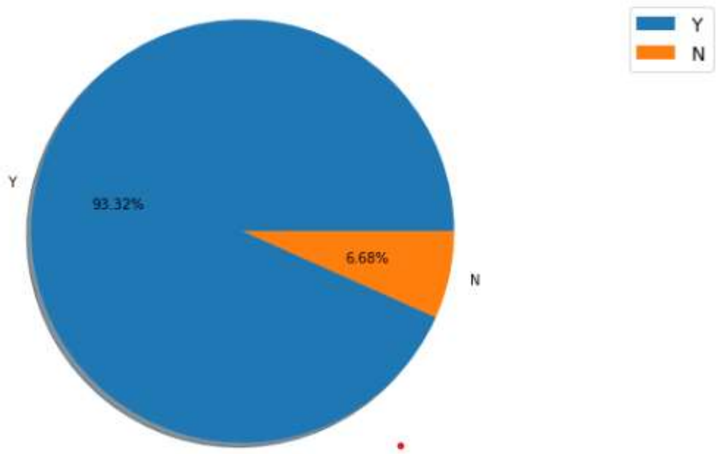

TA
Gd
Ex
Fa
TA
46.83%
2.48%
8.05%
42.64%
Fa
Ex
Gd

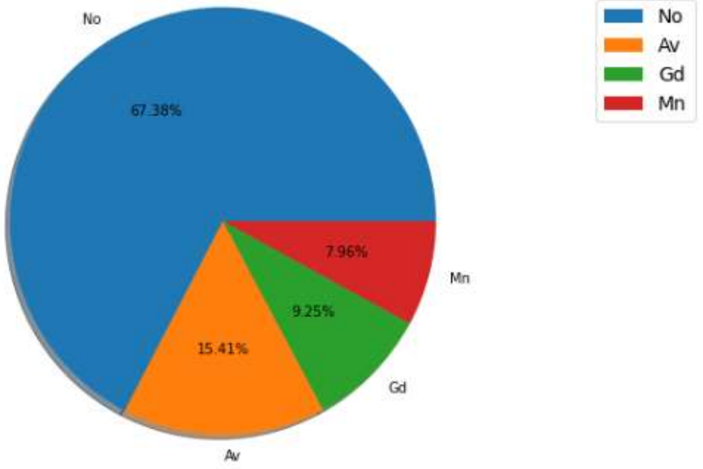

TA
Gd
Ex
Fa
TA
61.39%
0.94%
3.68%
33.99%
Fa
Ex
Gd

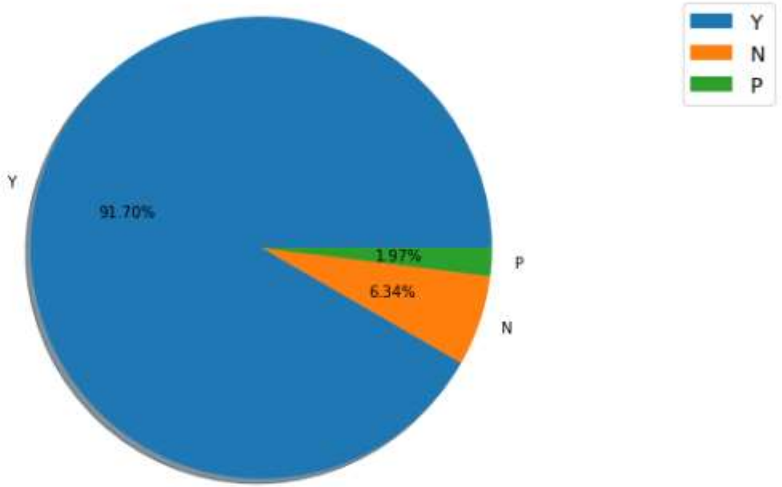

None
BrkFace
Stone
BrkCmn
None
60.19%
1.11%
8.39%
30.31%
BrkCmn
Stone
BrkFace


Unf
47.17%
29.02%
23.80%
RFn
Fin
Unf
RFn
Fin


TA
49.49%
2.57%
Fa
7.02%
Ex
40.92%
Gd
TA
Gd
Ex
Fa


Y
93.32%
6.68%
N
Y
N

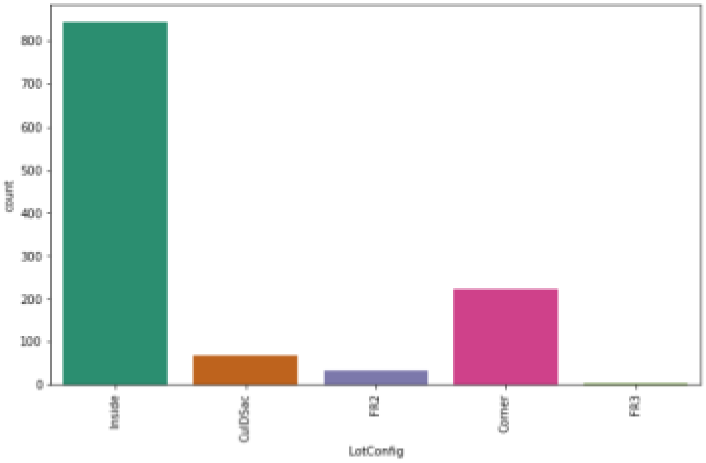

No
67.38%
7.96%
Mn
9.25%
15.41%
Gd
Av
No
Av
Gd
Mn

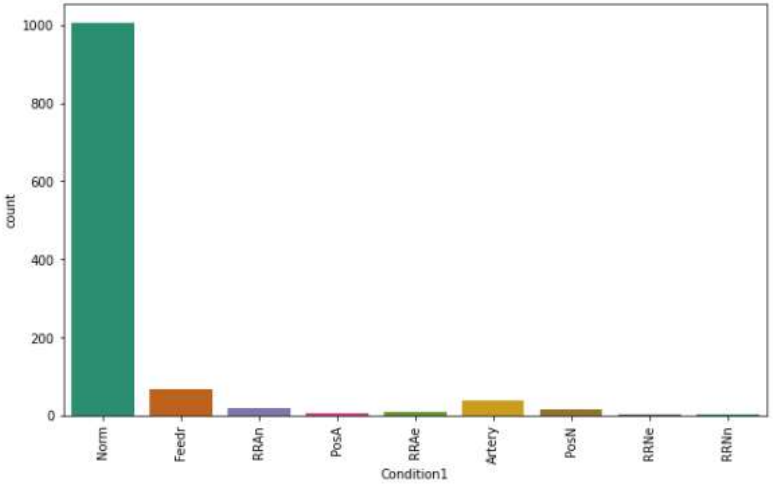

Y
91.70%
1.97%
P
6.34%
N
Y
N
P

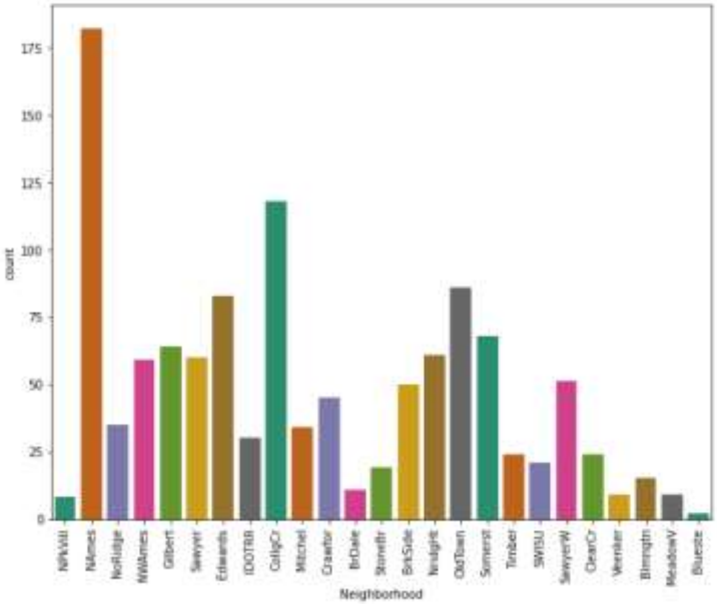
Observation from the Pie plots

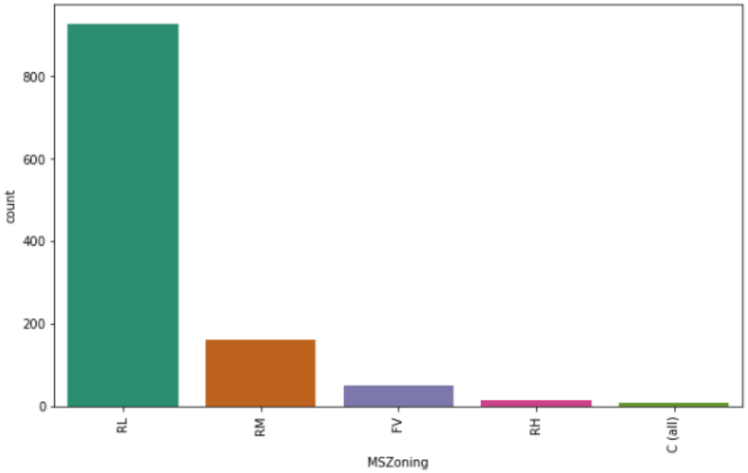
- The count of road access to the property Paved is 1164 which covers around 99.66% of the property where Graved type has count 4 that is only 0.34%.
- The count is high for the property having the shape regular.
- The total number of flatness of the property for level is high which has 89.55%.
- The slope of the property Gentle slope has very high count of 1105 i.e, 94.61%.
- Around 60% of the houses does not have Masonry veneer type and 30% of the houses contains Brick Face type of Masonry veneer.
- Around 61% of the houses evaluates typical/average quality of the material on the exterior, 34% of the houses have good quality of the material on the exterior. Only a few have excellent quality.

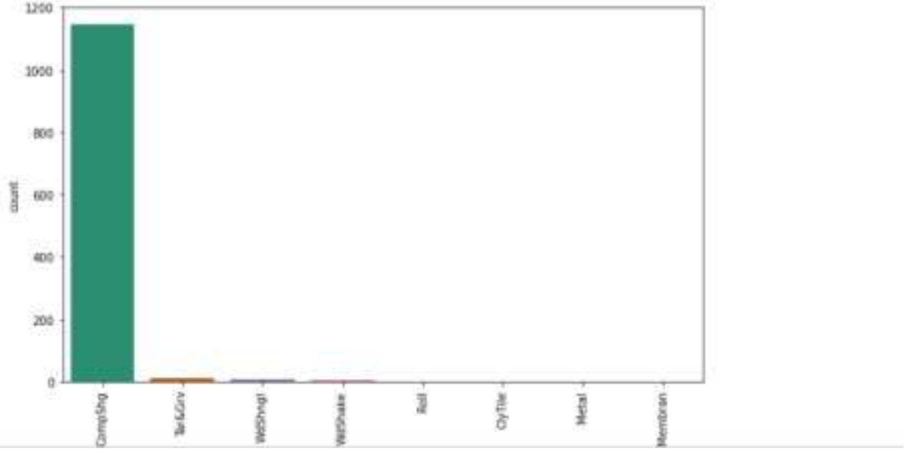
- Most of the houses evaluates typical/average and good quality of height of the basement.
- Around 91% of the houses have typical/average condition of the basement.
- Around 67% of the houses does not contain any walkout or garden level walls.
- 93.32% of the houses have central air conditioning.
- 49% of the houses contains typical/average kitchen quality and 40% of the houses have good kutchen quality. The count for excellent kitchen quality is very low and is around 2%.
- 47% Of the houses have unfinished garage interior, 29% rough finished and only 23% of the houses' interior garage has finished.
- 91.70% of the houses contains the paved drive way.

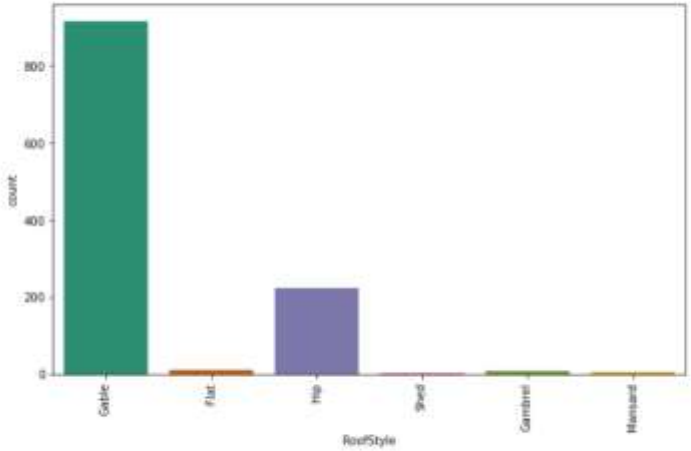

count
800
700
600
500
400
300
200
100
0
Inside
CulDSac
FR2
Corner
FR3
LotConfig

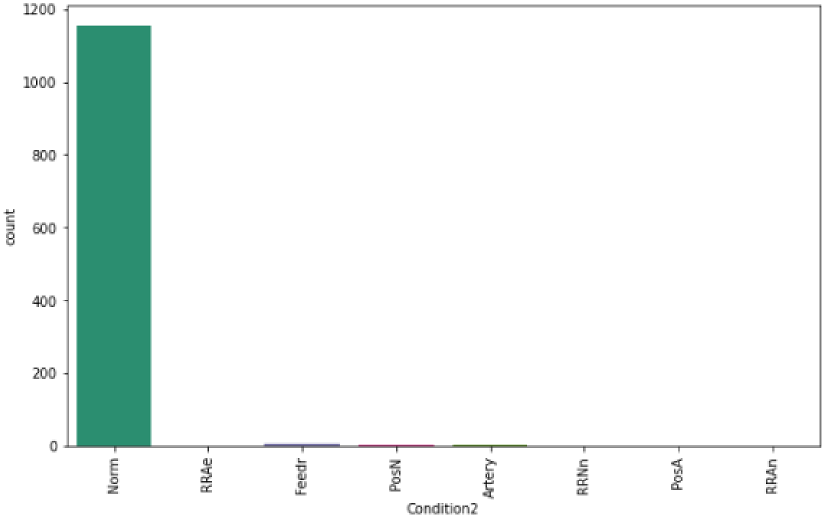

count
1000
800
600
400
200
0
Norm
Feedr
RRAn
PosA
RRAe
Artery
PosN
RRNe
RRNn
Condition1

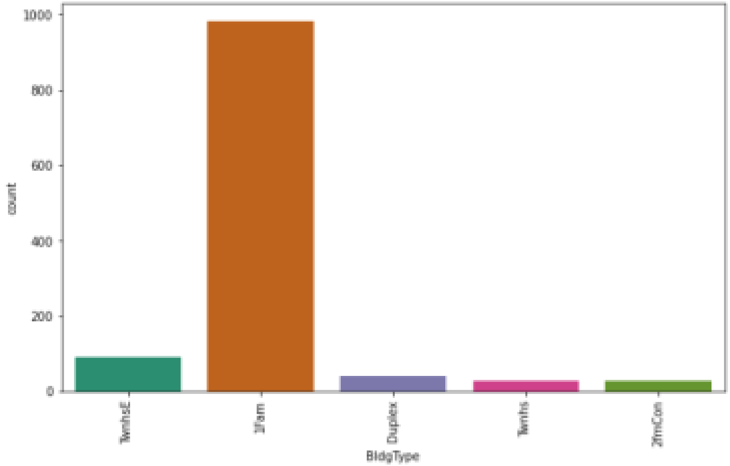

count
175
150
125
100
75
50
25
0
NPkVill
NAmes
NoRidge
NWAmes
Gilbert
Sawyer
Edwards
IDOTRR
CollgCr
Mitchel
Crawfor
BrDale
StoneBr
BrkSide
NridgHt
OldTown
Somerst
Timber
SWISU
SawyerW
ClearCr
Veenker
Blmngtn
MeadowV
Blueste
Neighborhood

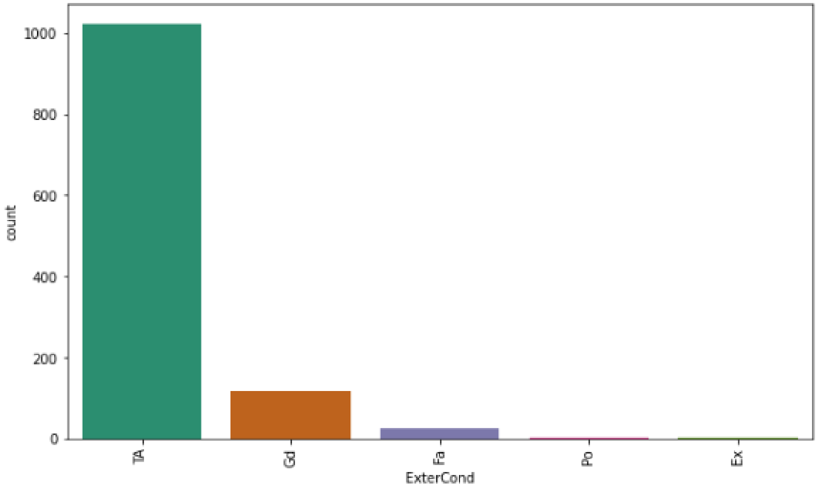

count
800
600
400
200
0
RL
RM
FV
RH
C (all)
MSZoning

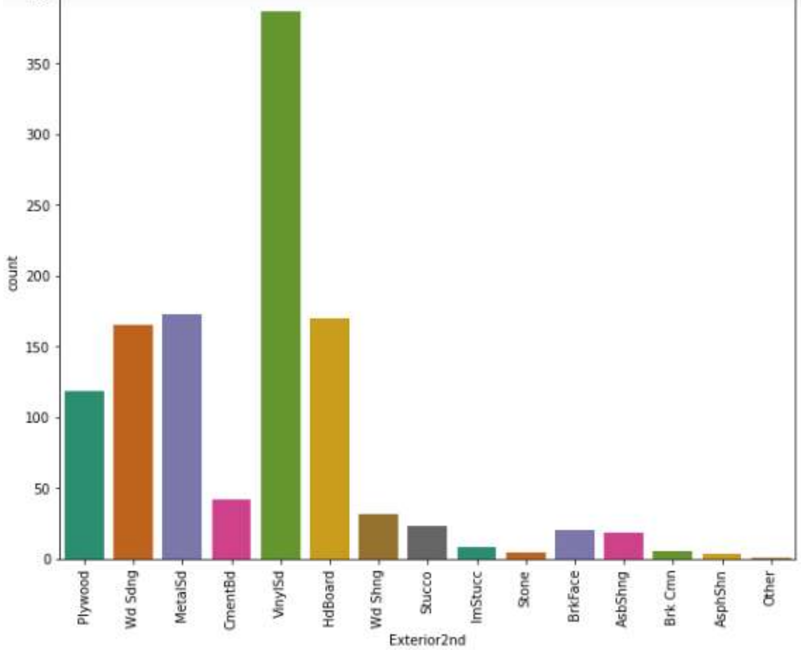

1200
1000
800
600
400
200
0
count
CompShg
Tar&Grv
WdShngl
WdShake
Roll
ClyTile
Metal
Membran

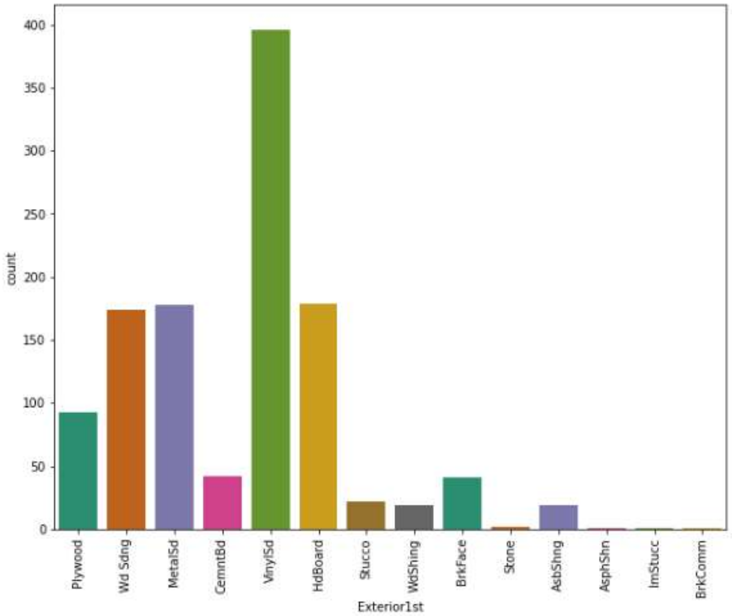

800
600
400
200
0
count
Gable
Flat
Hip
Shed
Gambrel
Mansard
RoofStyle

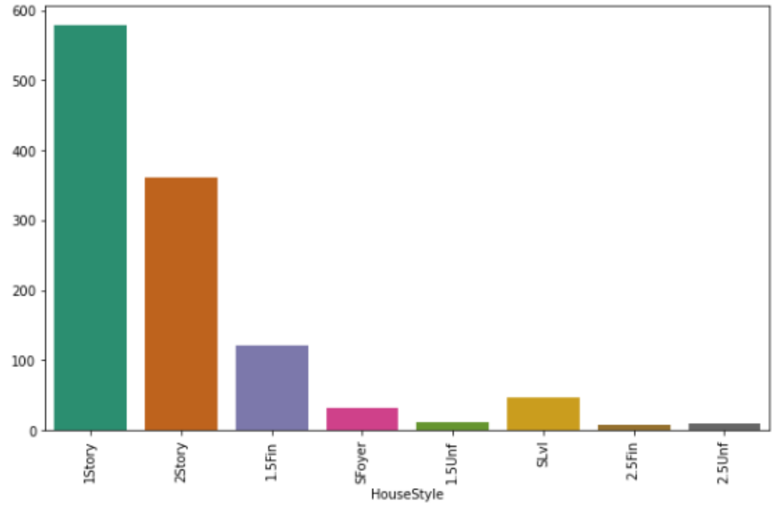

1200
1000
800
600
400
200
0
count
Norm
RRAe
Feedr
PosN
Artery
RRNn
PosA
RRAn
Condition2

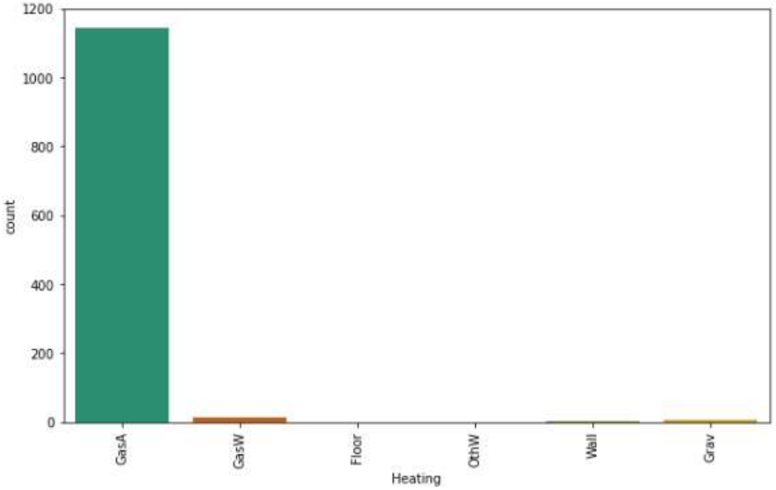

1000
800
600
400
200
0
count
TwnhsE
1Fam
Duplex
Twnhs
2fmCon
BldgType

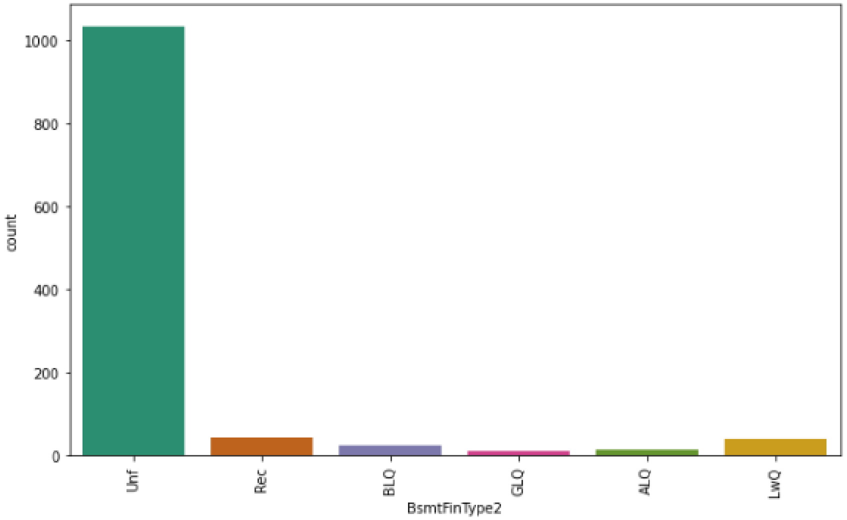

count
1000
800
600
400
200
0
TA
Gd
Fa
Po
Ex
ExterCond

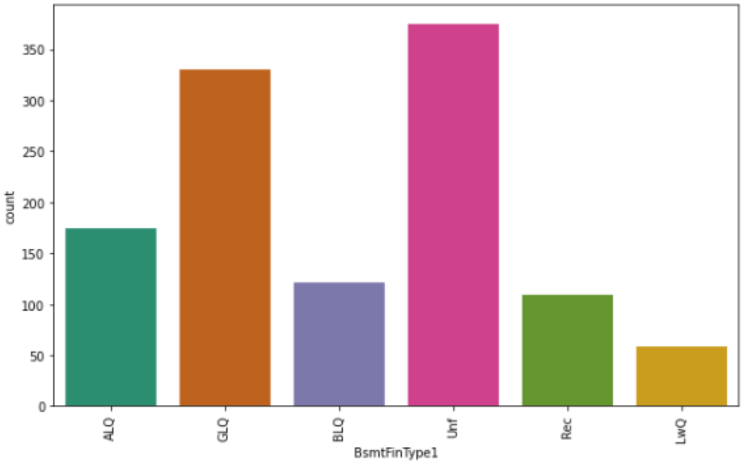

count
350
300
250
200
150
100
50
0
Plywood
Wd Sdng
MetalSd
CmentBd
VinylSd
HdBoard
Wd Shng
Stucco
ImStucc
Stone
BrkFace
AsbShng
Brk Cmn
AsphShn
Other
Exterior2nd

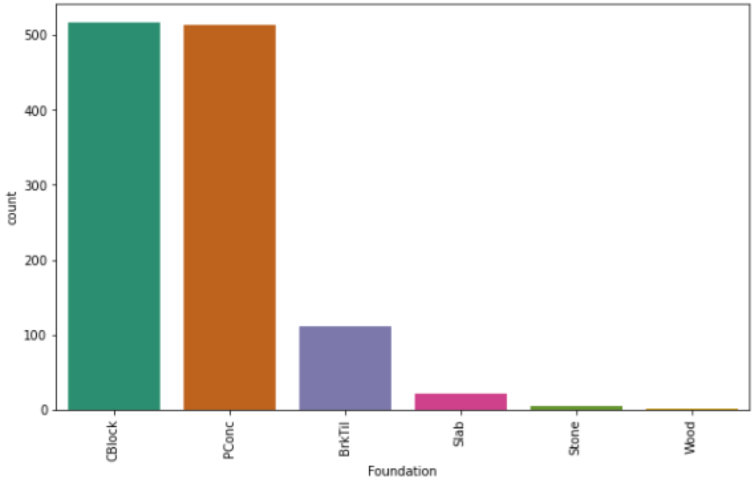

count
400
350
300
250
200
150
100
50
0
Plywood
Wd Sdng
MetalSd
CemntBd
VinylSd
HdBoard
Stucco
WdShing
BrkFace
Stone
AsbShng
AsphShn
ImStucc
BrkComm
Exterior1st

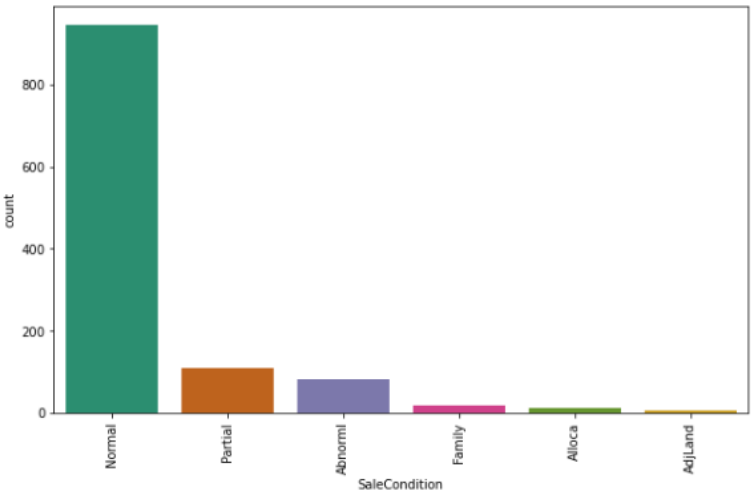

600
500
400
300
200
100
0
1Story
2Story
1.5Fin
SFoyer
1.5Unf
SLvl
2.5Fin
2.5Unf
HouseStyle

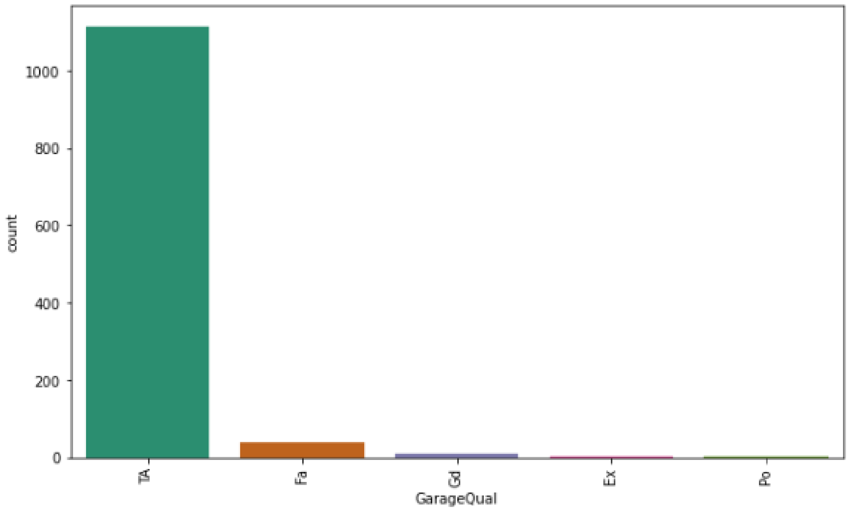

1200
1000
800
600
400
200
0
count
GasA
GasW
Floor
OthW
Wall
Grav
Heating

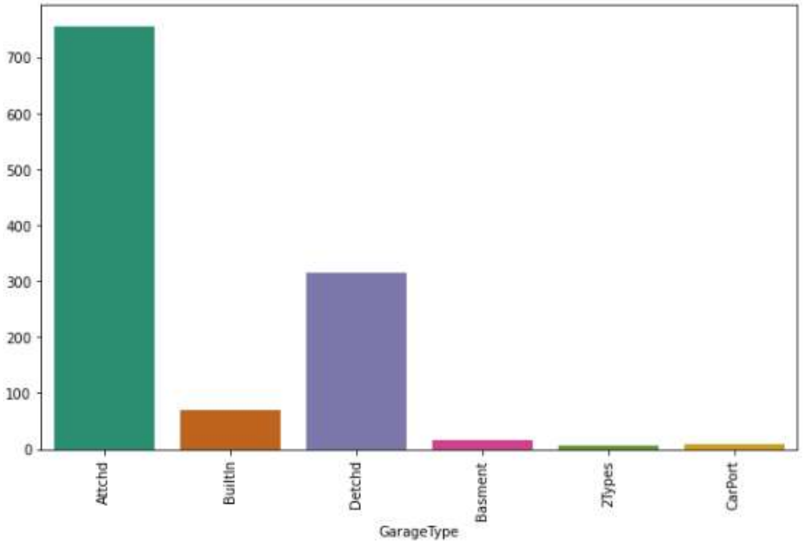

1000
800
600
400
200
0
count
Unf
Rec
BLQ
GLQ
ALQ
LwQ
BsmtFinType2

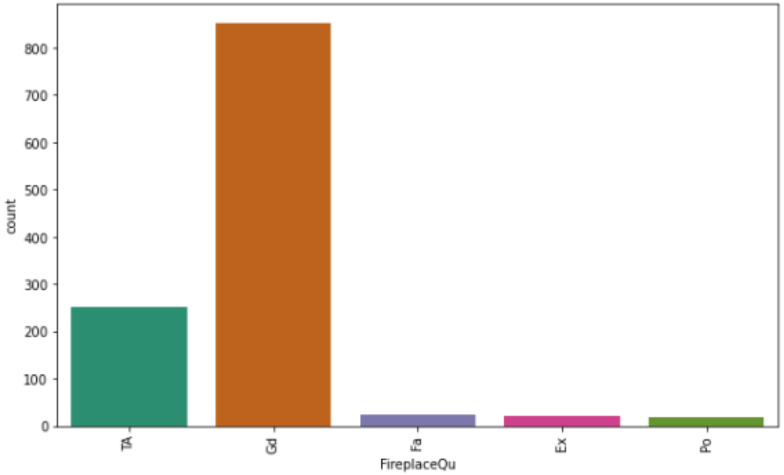

350
300
250
200
150
100
50
0
count
ALQ
GLQ
BLQ
Unf
Rec
LwQ
BsmtFinType1


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


count
800
600
400
200
0
Normal
Partial
Abnorml
Family
Alloca
AdjLand
SaleCondition


count
1000
800
600
400
200
0
TA
Fa
Gd
Ex
Po
GarageQual


700
600
500
400
300
200
100
0
Attchd
BuiltIn
Detchd
Basment
2Types
CarPort
GarageType


count
800
700
600
500
400
300
200
100
0
TA
Gd
Fa
Ex
Po
FireplaceQu

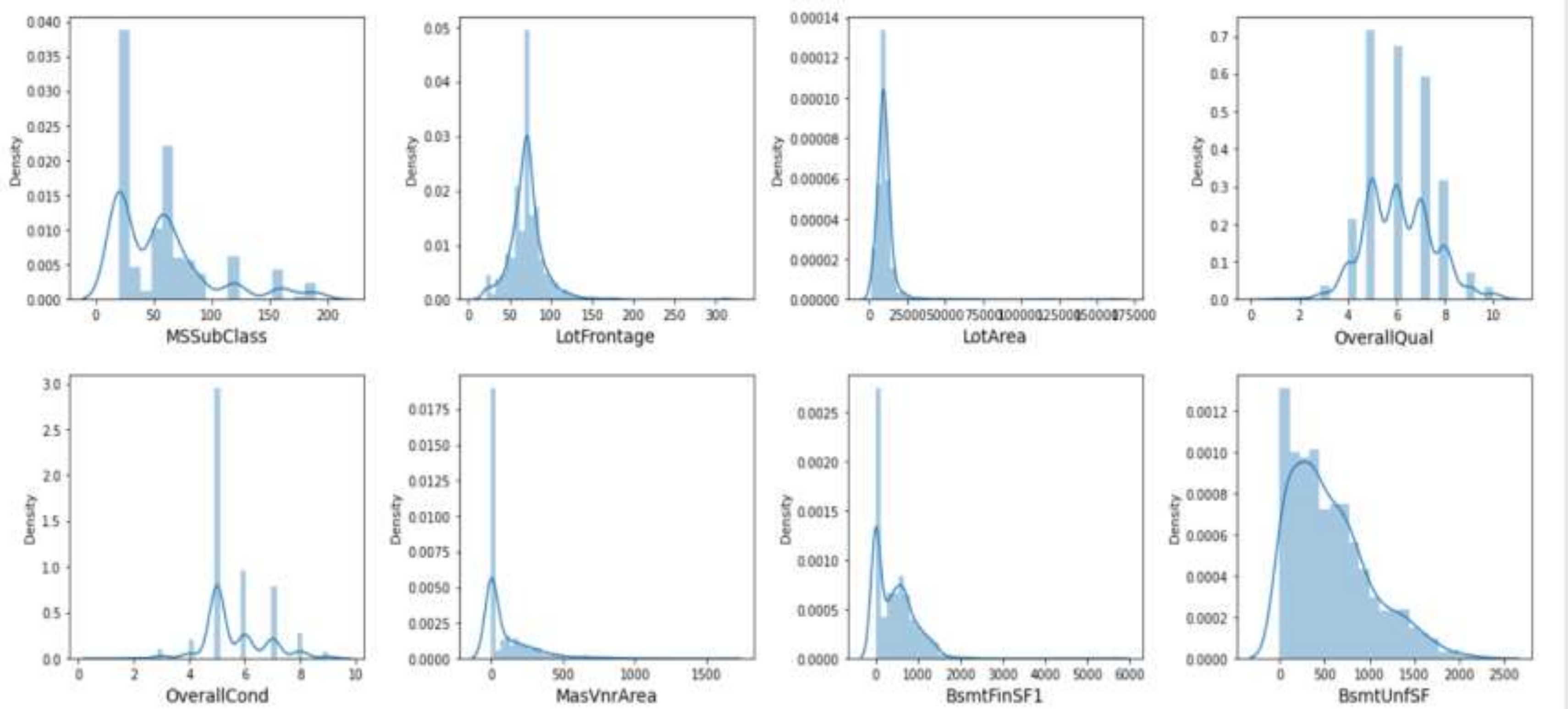
Observations from the count plots

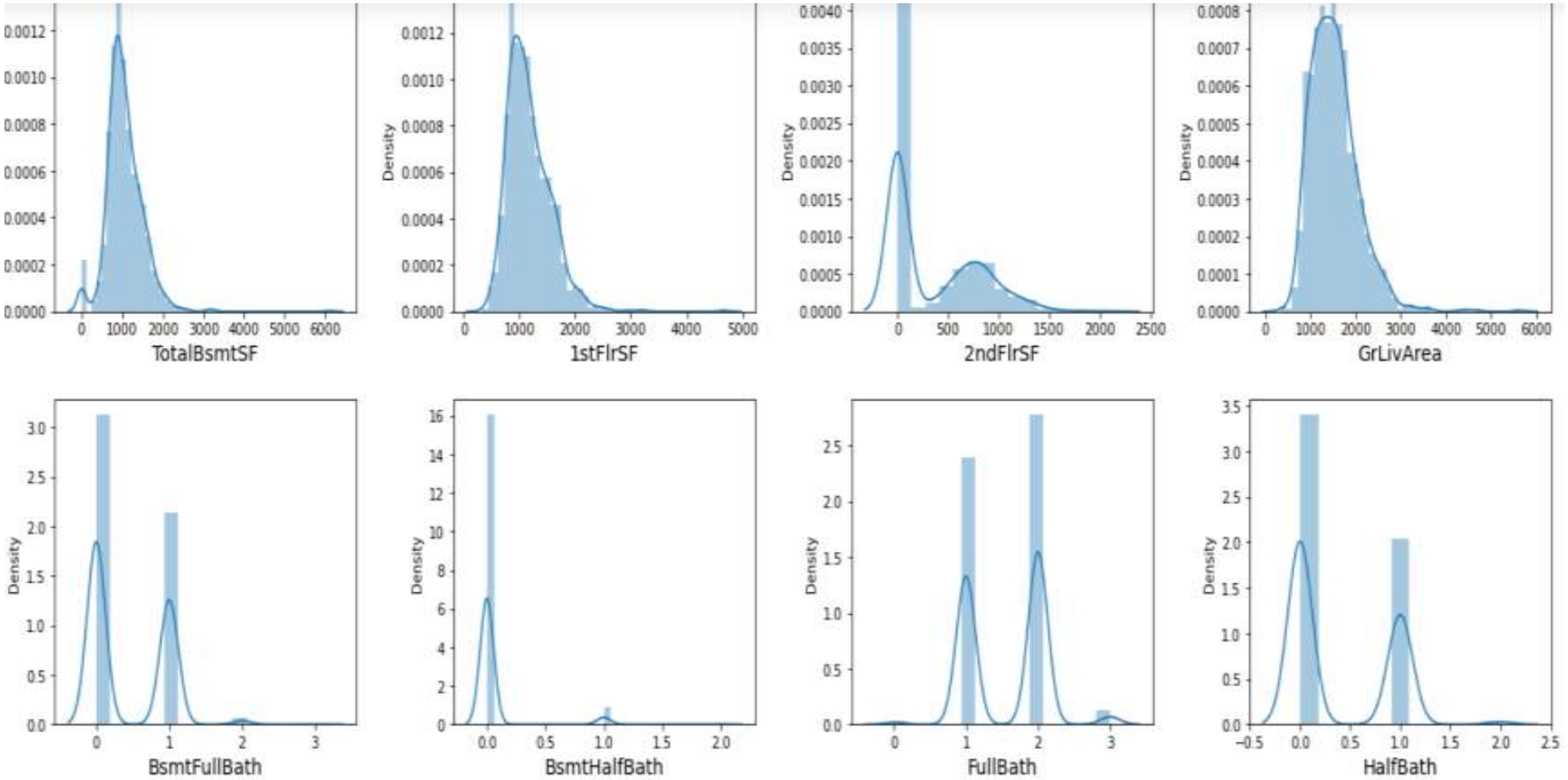
- The houses having Residential Low Density zoning of the sale have high count and commercial zoning sale have very less count compared to others.
- Inside lot configiration has high count and Frontage on 3 sides of property have very less count compared to others.
- The count is high for the North Ames cities floowed by college creek and count is very low for Bluestem.
- The count is high for the Normal proximity condition apart from this all the others have very less count.
- Similar to condition1, in comdition2 also Normal proximity has very high count compared to others.
- Single-family detached dwelling type has very high counts compared to other types have very less count.
- 1 story style of dwelling has high count followed by 2 story and others have very less count.
- The flat type roof has high count and shed has very less count.
- The roof material type Standard (Composite) Shingle has highest count and others have very less counts.

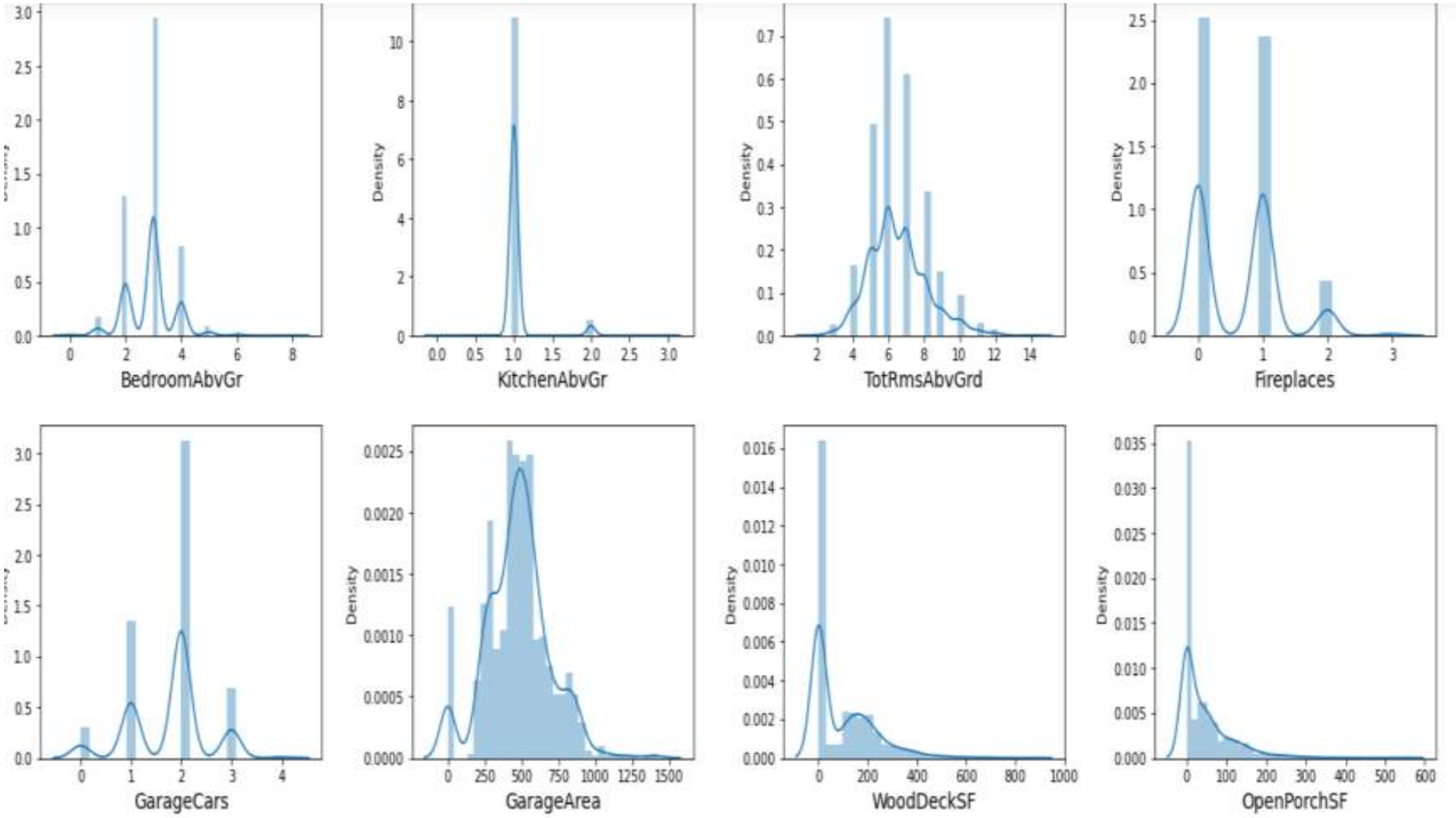
- Most of the houses have Vinyl Siding exterior covering materials followed by hard board also Brick Common, Asphalt Shingles and Imitation Stucco have very count which means there are no more houses with these types.
- Similar to Exterior1st, here also most of the houses have Vinyl Siding exterior covering materials.
- The present condition of the material on the exterior for most of the houses are Average/Typical.
- Most of the houses have Cinder Block and Poured Contrete type of foundation.
- The count is high for the houses having unfinished basement area. Also some houses have Good Living Quarters.
- Similar to BsmtFinType1, here also the count is high for unfinished basements.
- Most of the houses have Gas forced warm air furnace heating type.
- Most of the houses have excellent heating quality and condition also some houses have typical/average HeatingQC and only 1% of the houses have poor heating quality and condition.

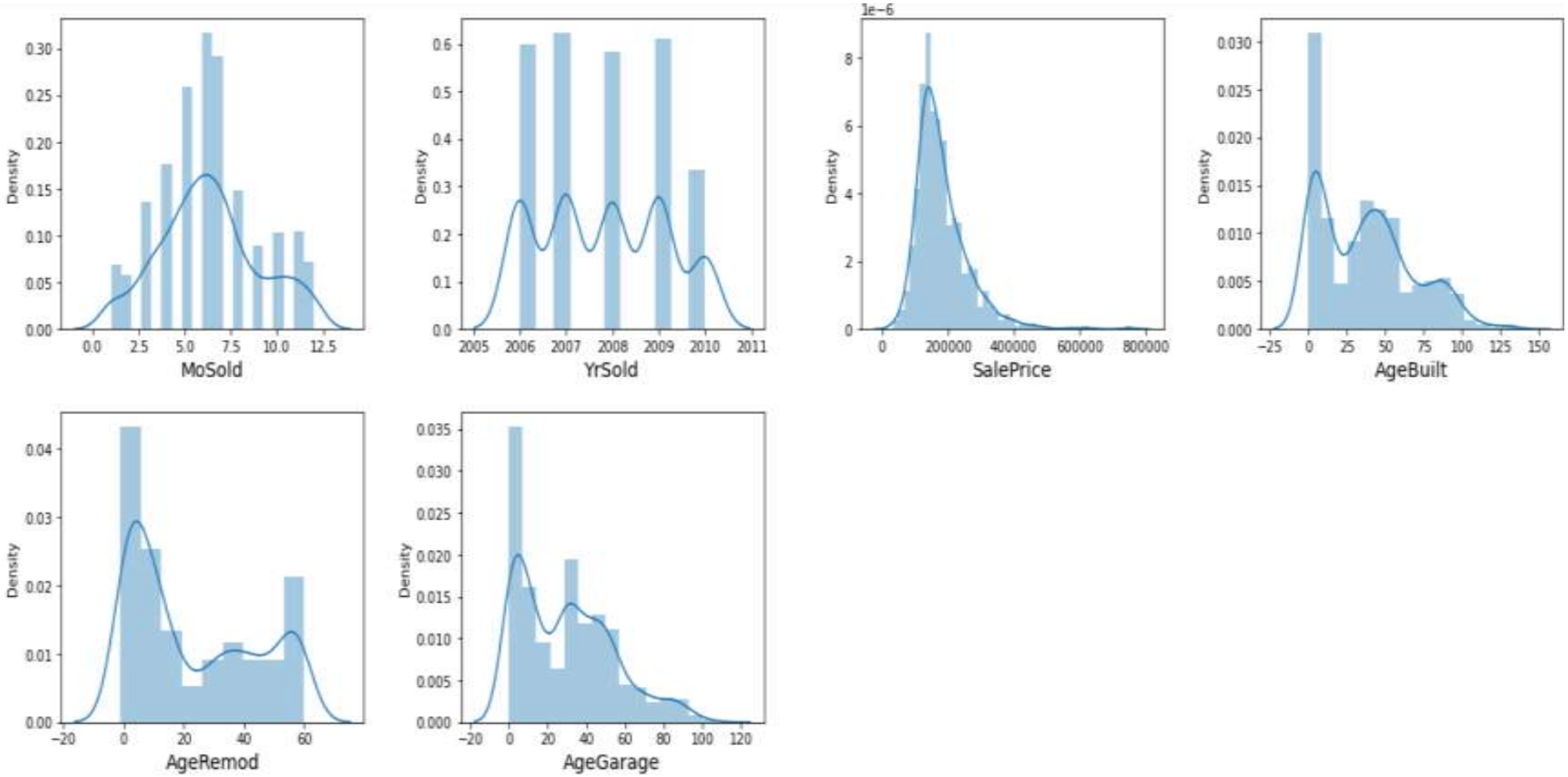
- The electrical system of the type Standard Circuit Breakers & Romex has very high count which means most of the houses have this facility.
- The total number of home functionality of the property for typical functionality have high count compared to others.
- The houses with Masonry Fireplace in main level have good quality compared ot others.
- The garage location attachec to home has high count also the garage locations detached to home have moderate level of counts. Only few houses have car port and more than one type of garage.
- Houses with typical/average garage quality have high count compared to others.
- Houses with typical/average garage condition have high count compared to others.
- Warranty Deed type of sale has high count followed by Home just constructed and sold(new).
- Normal sale has high count also the home which was not completed when last assessed also has average sale condition.

Visualizing the data distribution of all features




Density
0.0000
0.0002
0.0004
0.0006
0.0008
0.0010
0.0012
0
1000
2000
3000
4000
5000
6000
TotalBsmtSF
0
1000
2000
3000
4000
5000
1stFlrSF
0.0005
0.0010
0.0015
0.0020
0.0025
0.0030
0.0035
0.0040
500
1500
2500
2ndFlrSF
0.0001
0.0003
0.0005
0.0007
GrLivArea
0.0
0.5
1.0
1.5
2.0
2.5
3.0
1
2
3
BsmtFullBath
4
6
8
10
12
14
16
BsmtHalfBath
FullBath
3.5
−0.5
HalfBath


Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
2
4
6
8
BedroomAbvGr
Density
0
2
4
6
8
10
0.0
0.5
1.0
1.5
2.0
2.5
3.0
KitchenAbvGr
Density
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
2
4
6
8
10
12
14
TotRmsAbvGrd
Density
0.0
0.5
1.0
1.5
2.0
2.5
0
1
2
3
Fireplaces
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
1
2
3
4
GarageCars
Density
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025
0
250
500
750
1000
1250
1500
GarageArea
Density
0.000
0.002
0.004
0.006
0.008
0.010
0.012
0.014
0.016
0
200
400
600
800
1000
WoodDeckSF
Density
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
0
100
200
300
400
500
600
OpenPorchSF

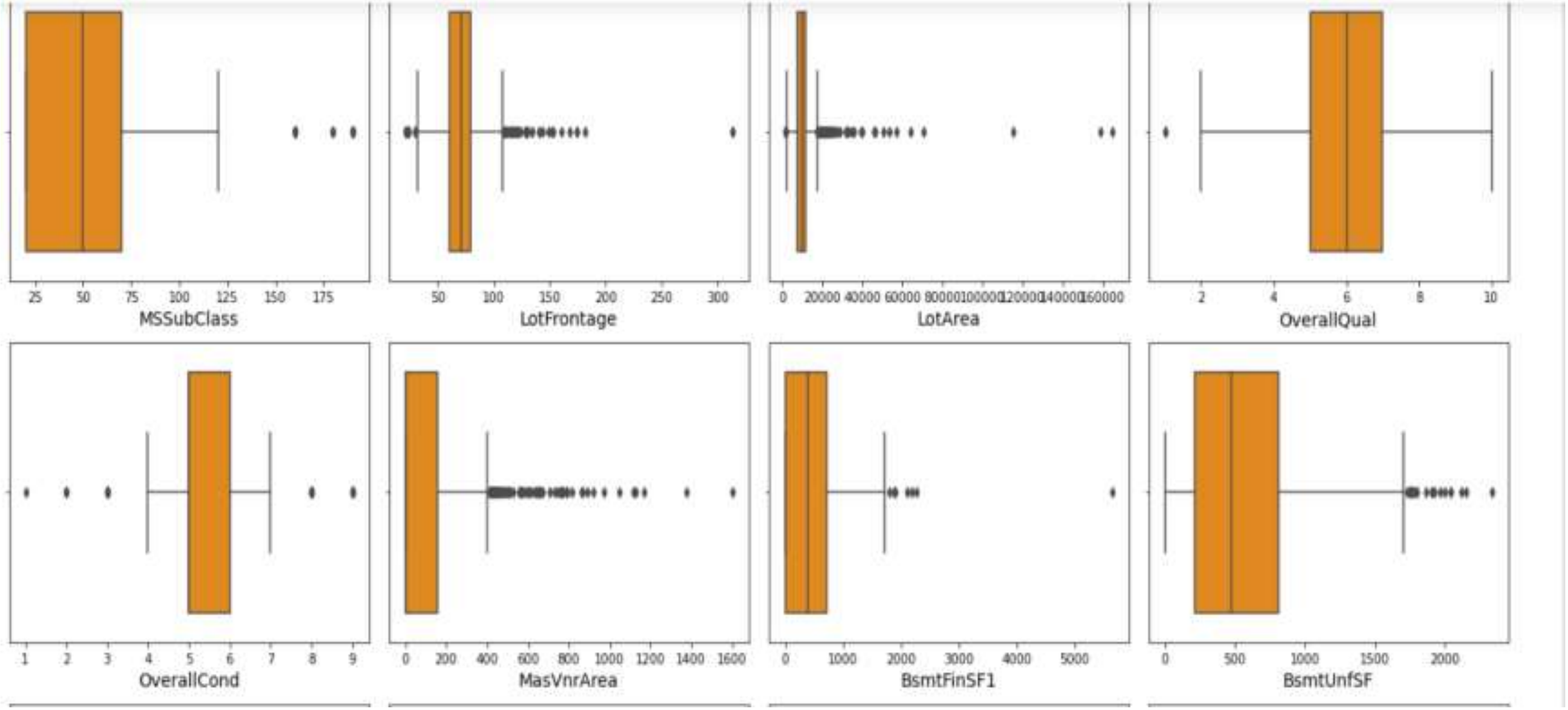

Density
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0.0 2.5 5.0 7.5 10.0 12.5
MoSold
Density
0.6
0.5
0.4
0.3
0.2
0.1
0.0
2005 2006 2007 2008 2009 2010 2011
YrSold
1e−6
Density
8
6
4
2
0
0 200000 400000 600000 800000
SalePrice
Density
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−25 0 25 50 75 100 125 150
AgeBuilt
Density
0.04
0.03
0.02
0.01
0.00
−20 0 20 40 60
AgeRemod
Density
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−20 0 20 40 60 80 100 120
AgeGarage

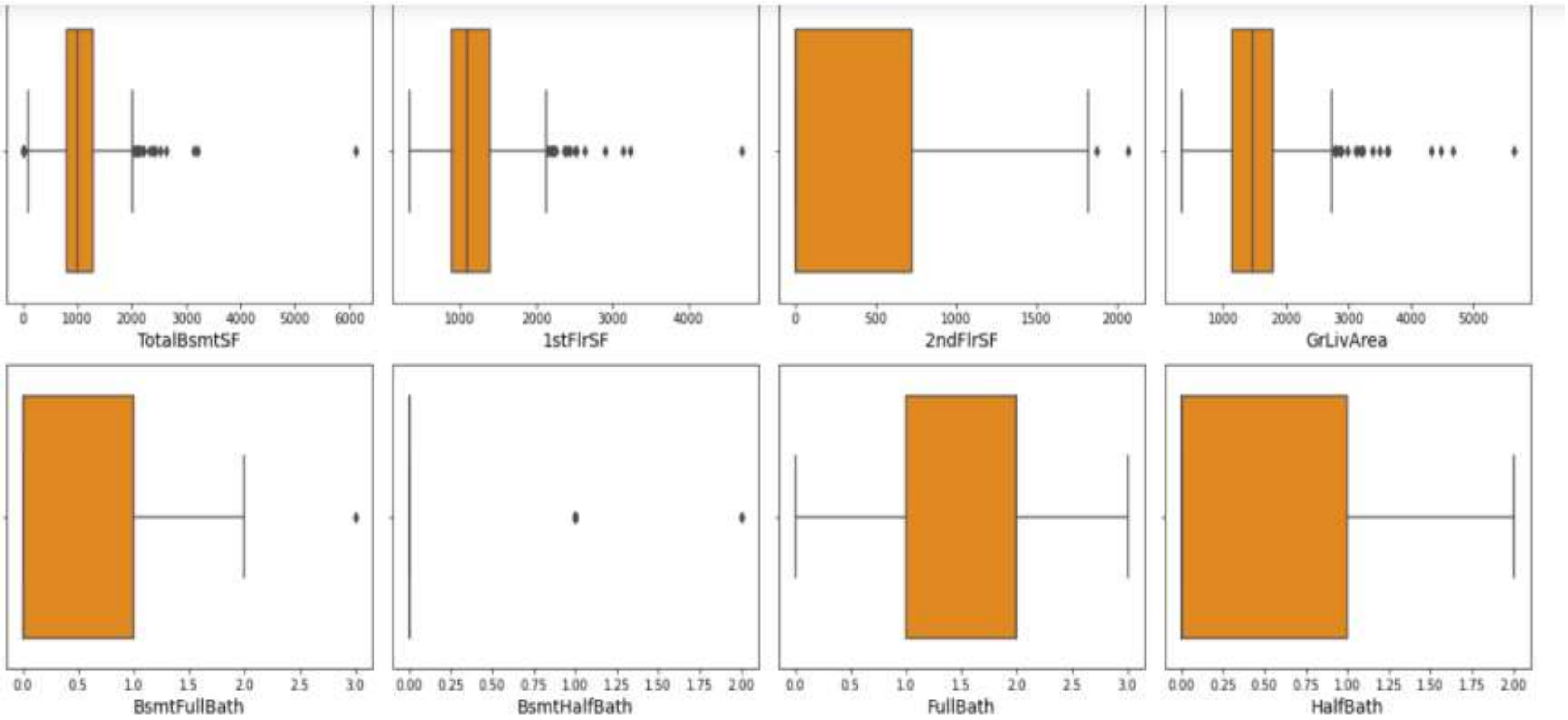
Observations from the distplots:

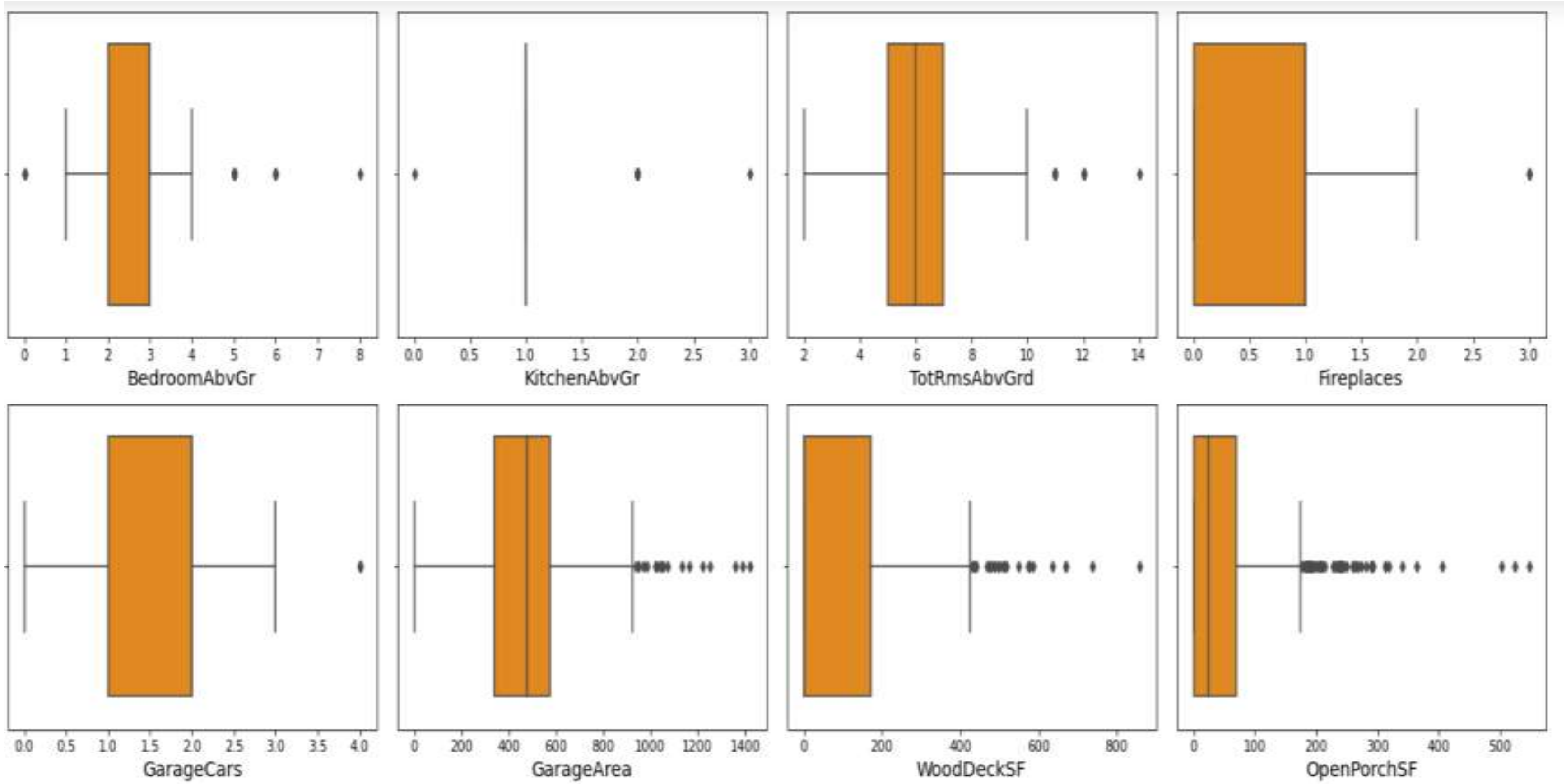
- From the above distribution plot we can observe most of the columns are not normally distributed only some of the columns are almost distributed normally.

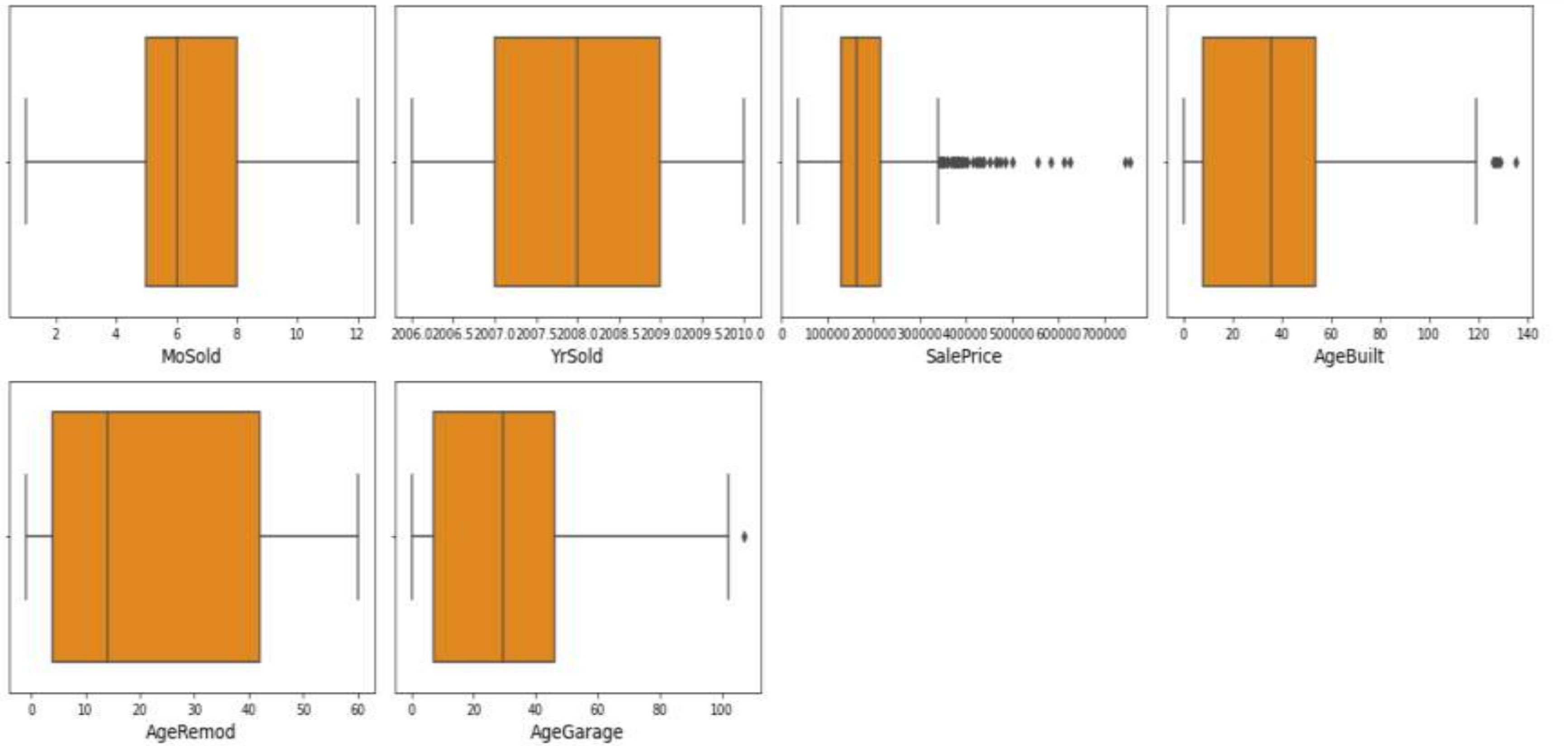
- Almost all the columns have skewness and are skewed to right. We need to remove this skewness before building our machine learning models.

Identifying the outliers by using Boxplots

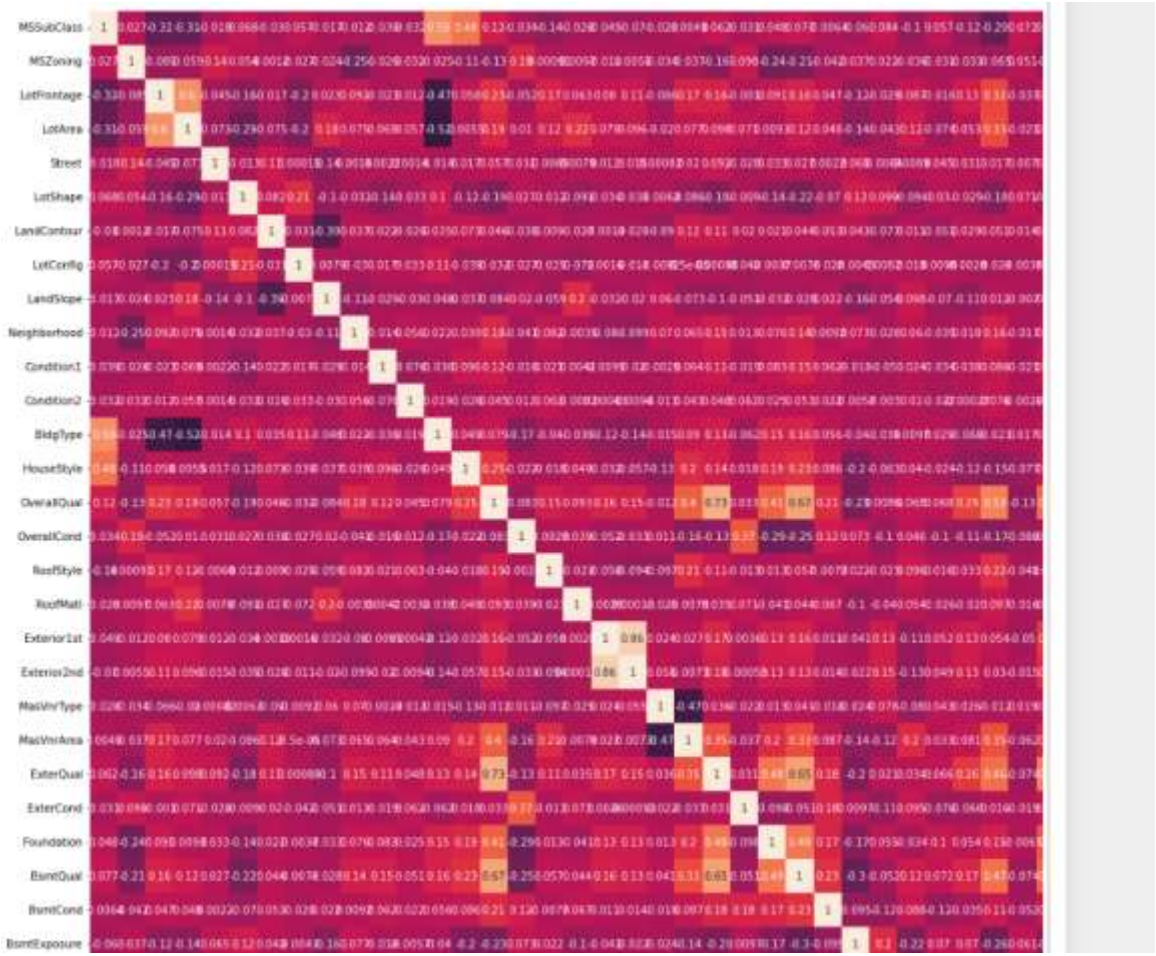



0 1000 2000 3000 4000 5000 6000
TotalBsmtSF
1000 2000 3000 4000
1stFlrSF
0 500 1000 1500 2000
2ndFlrSF
1000 2000 3000 4000 5000
GrLivArea
0.0 0.5 1.0 1.5 2.0 2.5 3.0
BsmtFullBath
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00
BsmtHalfBath
0.0 0.5 1.0 1.5 2.0 2.5 3.0
FullBath
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00
HalfBath

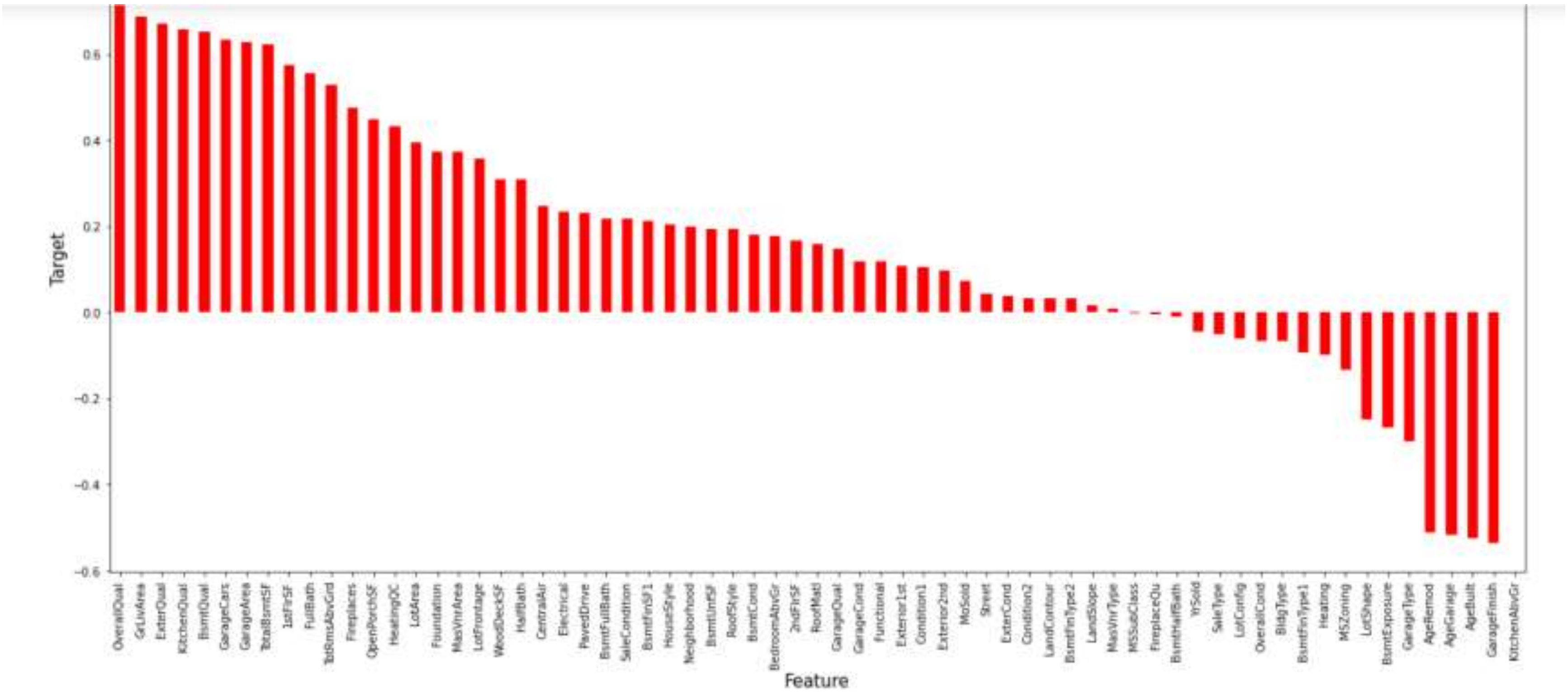

0 1 2 3 4 5 6 7 8
BedroomAbvGr
0.0 0.5 1.0 1.5 2.0 2.5 3.0
KitchenAbvGr
2 4 6 8 10 12 14
TotRmsAbvGrd
0.0 0.5 1.0 1.5 2.0 2.5 3.0
Fireplaces
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0
GarageCars
0 200 400 600 800 1000 1200 1400
GarageArea
0 200 400 600 800
WoodDeckSF
0 100 200 300 400 500
OpenPorchSF


2
4
6
8
10
12
MoSold
2006.0
2006.5
2007.0
2007.5
2008.0
2008.5
2009.0
2009.5
2010.0
YrSold
0
100000
200000
300000
400000
500000
600000
700000
SalePrice
0
20
40
60
80
100
120
140
AgeBuilt
0
10
20
30
40
50
60
AgeRemod
0
20
40
60
80
100
AgeGarage

Checking multicolinearity by using Heatmap

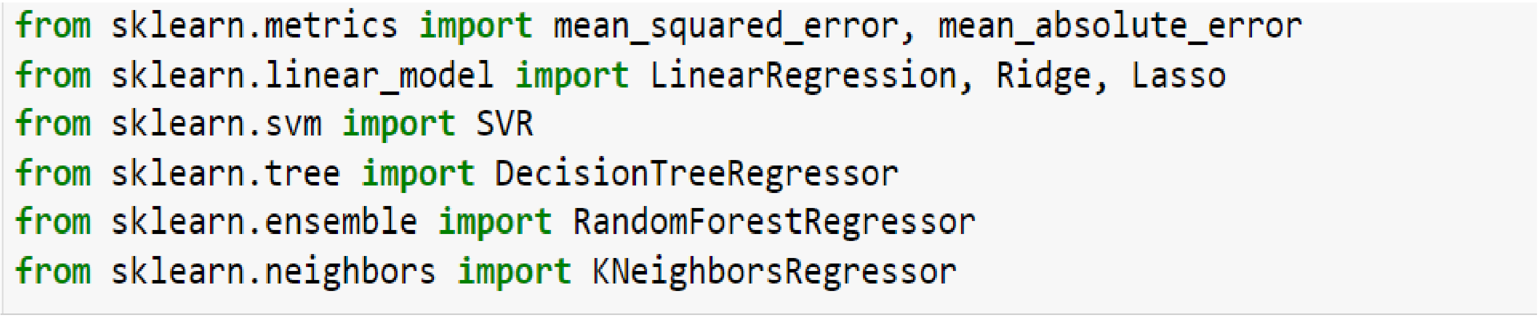


Checking the relationship between the features and the label

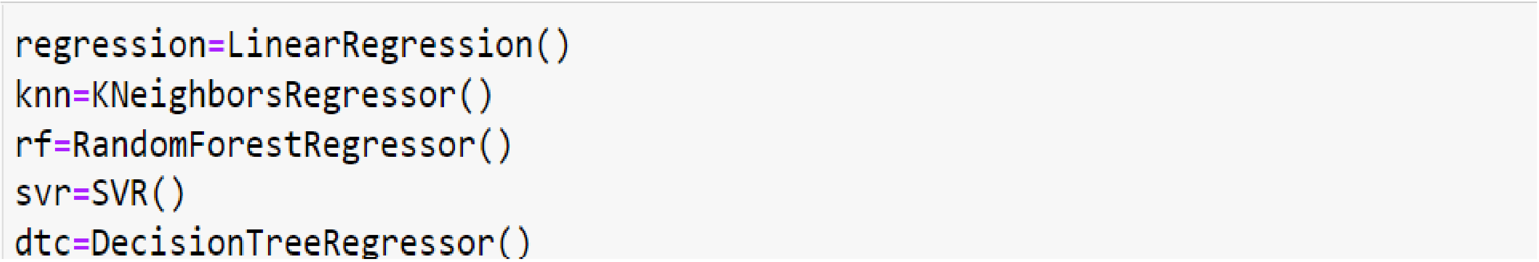


Model Development and Evaluation

- Importing the required libraries and declaring the models.

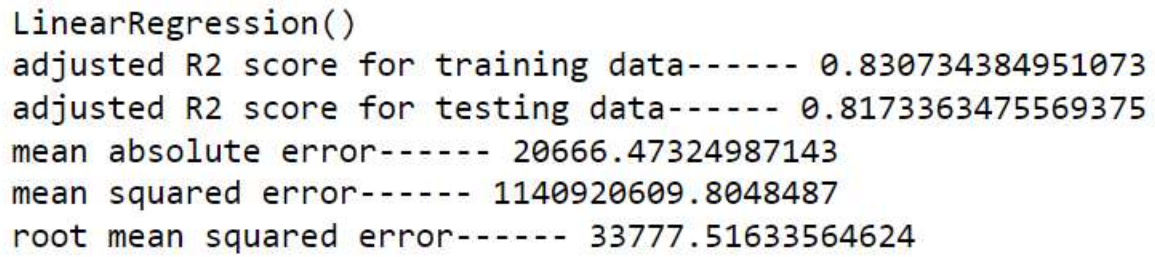


```
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.linear_model import LinearRegression, Ridge, Lasso
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.neighbors import KNeighborsRegressor
```

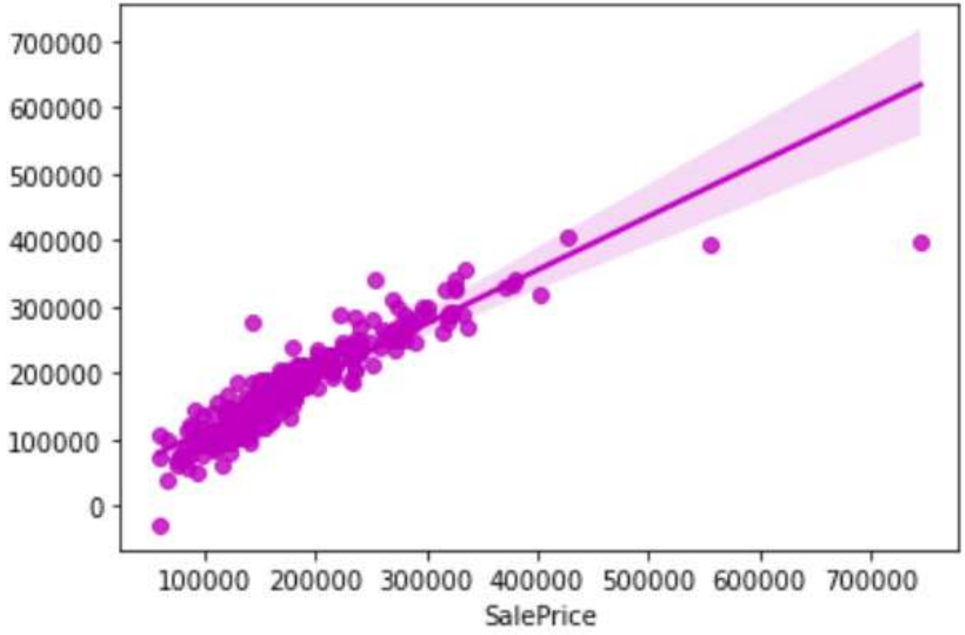


```
regression=LinearRegression()
knn=KNeighborsRegressor()
rf=RandomForestRegressor()
svr=SVR()
dtc=DecisionTreeRegressor()
```

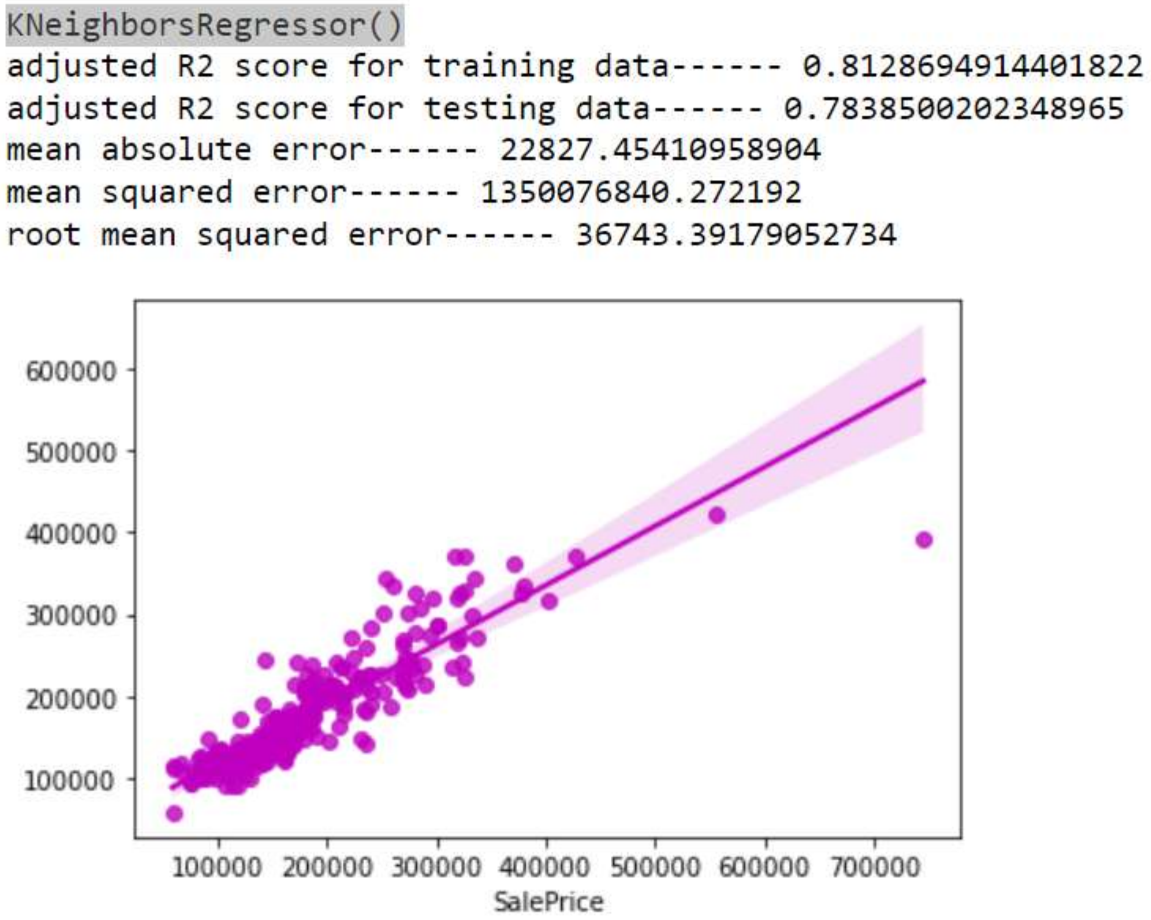
LinearRegression Model



```
LinearRegression()
adjusted R2 score for training data------ 0.830734384951073
adjusted R2 score for testing data------ 0.8173363475569375
mean absolute error------ 20666.47324987143
mean squared error------ 1140920609.8048487
root mean squared error------ 33777.51633564624
```

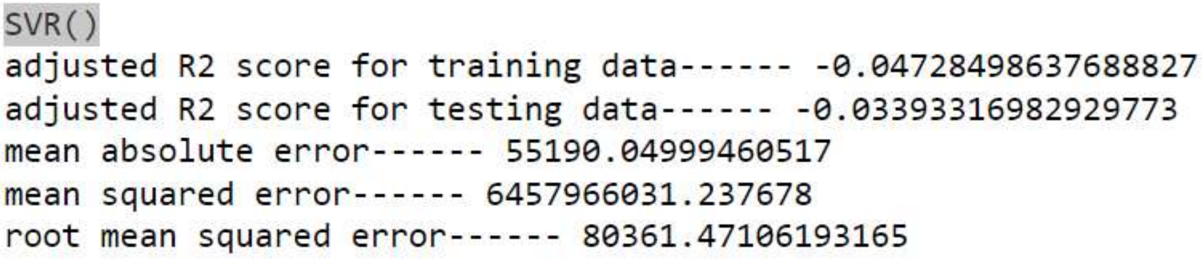


KNeighborsRegressor Model

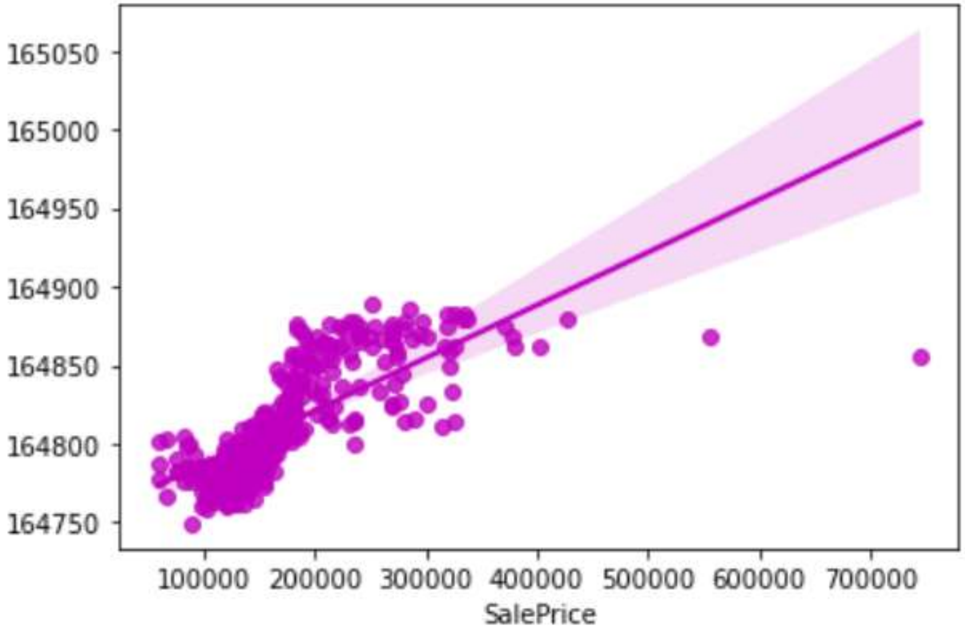


```
KNeighborsRegressor()
adjusted R2 score for training data------ 0.8128694914401822
adjusted R2 score for testing data------ 0.7838500202348965
mean absolute error------ 22827.45410958904
mean squared error------ 1350076840.272192
root mean squared error------ 36743.39179052734
```

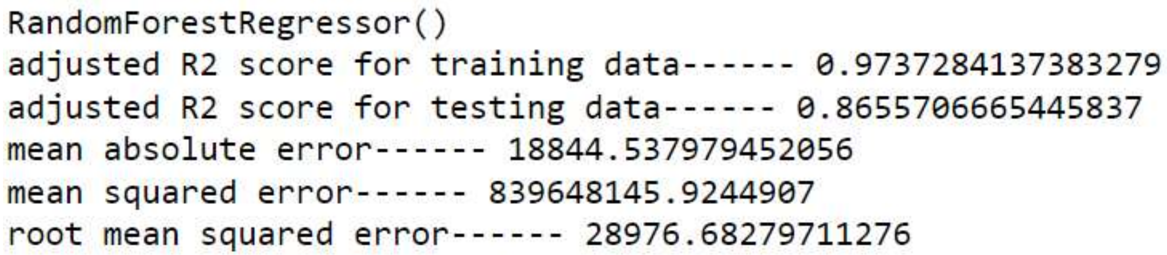
SVR Model



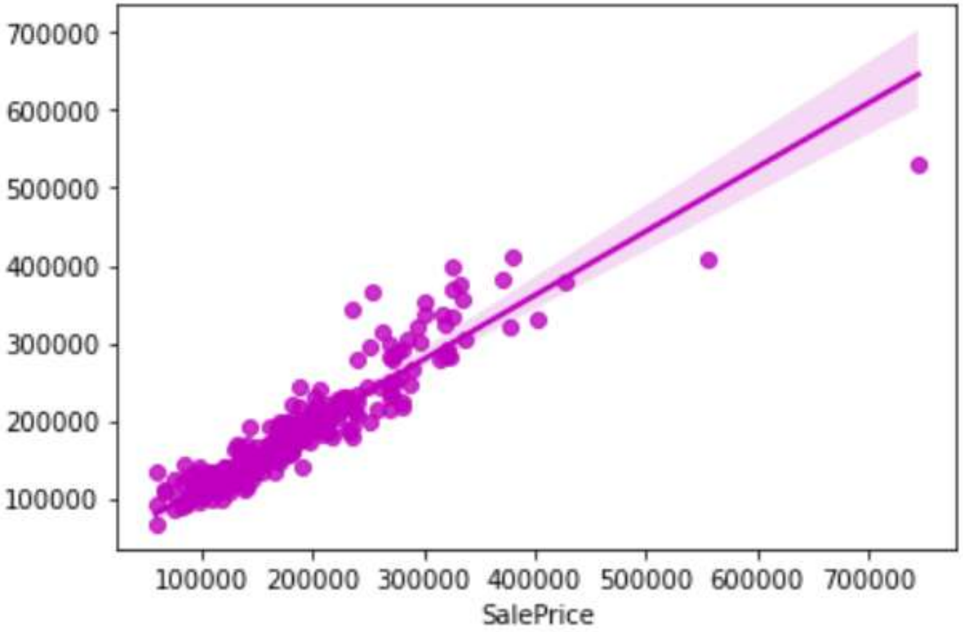
```
SVR()
adjusted R2 score for training data------ -0.04728498637688827
adjusted R2 score for testing data------ -0.03393316982929773
mean absolute error------ 55190.04999460517
mean squared error------ 6457966031.237678
root mean squared error------ 80361.47106193165
```



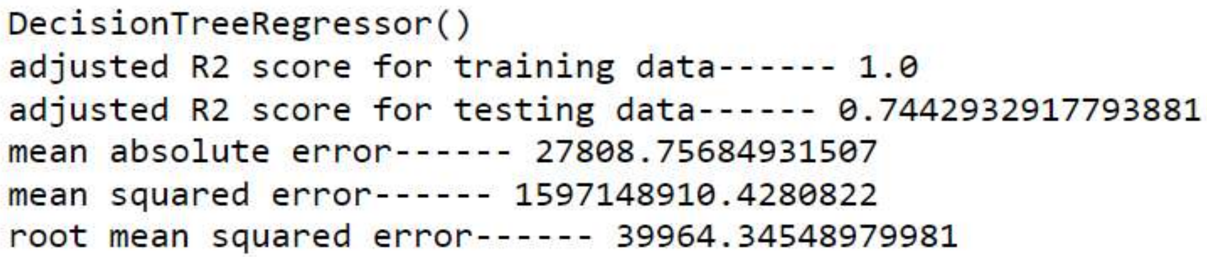
RandomForestRegressor Model



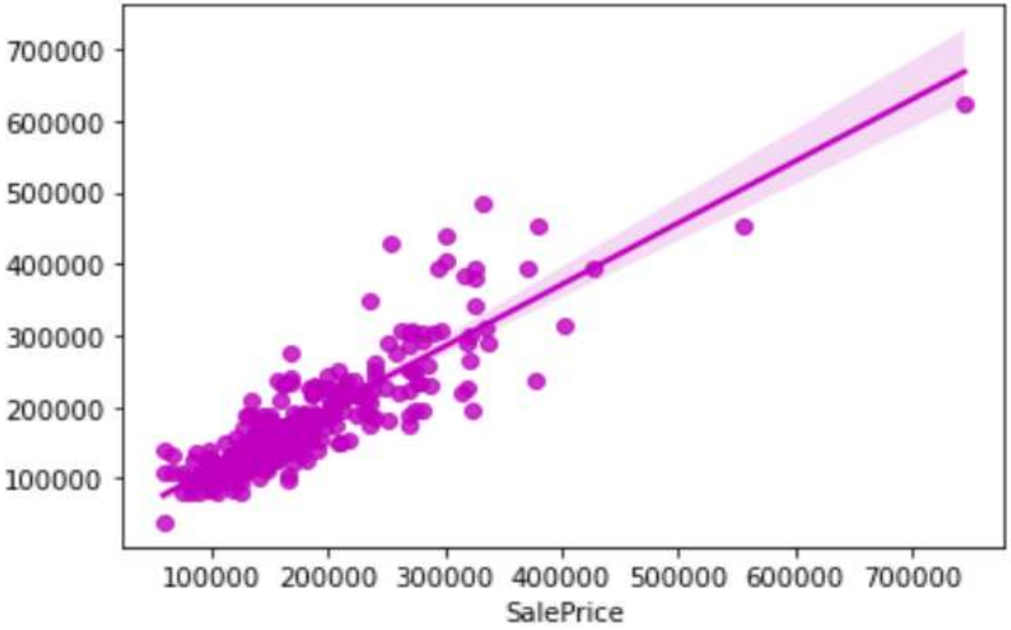
```
RandomForestRegressor()
adjusted R2 score for training data------ 0.9737284137383279
adjusted R2 score for testing data------ 0.8655706665445837
mean absolute error------ 18844.537979452056
mean squared error------ 839648145.9244907
root mean squared error------ 28976.68279711276
```



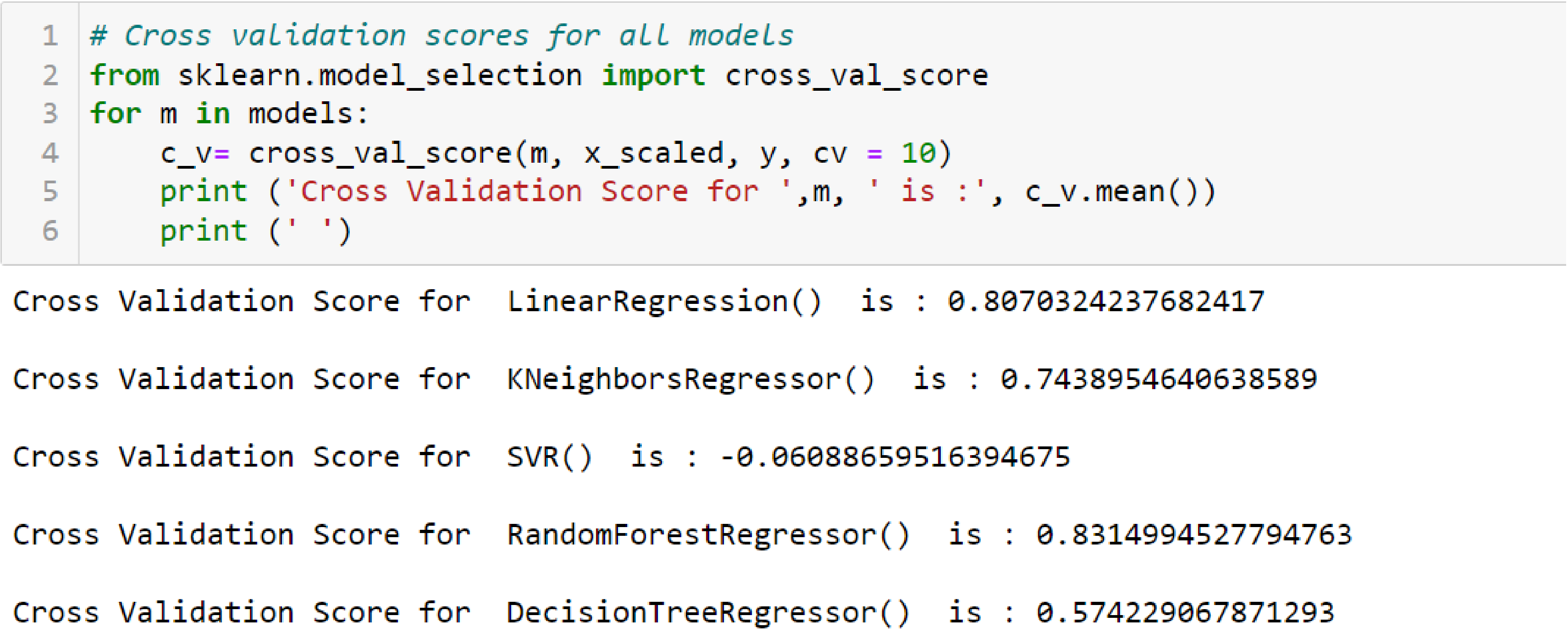
DecisionTreeRegressor Model



```
DecisionTreeRegressor()
adjusted R2 score for training data------ 1.0
adjusted R2 score for testing data------ 0.7442932917793881
mean absolute error------ 27808.75684931507
mean squared error------ 1597148910.4280822
root mean squared error------ 39964.34548979981
```



Cross-Validation Scores for all the Models



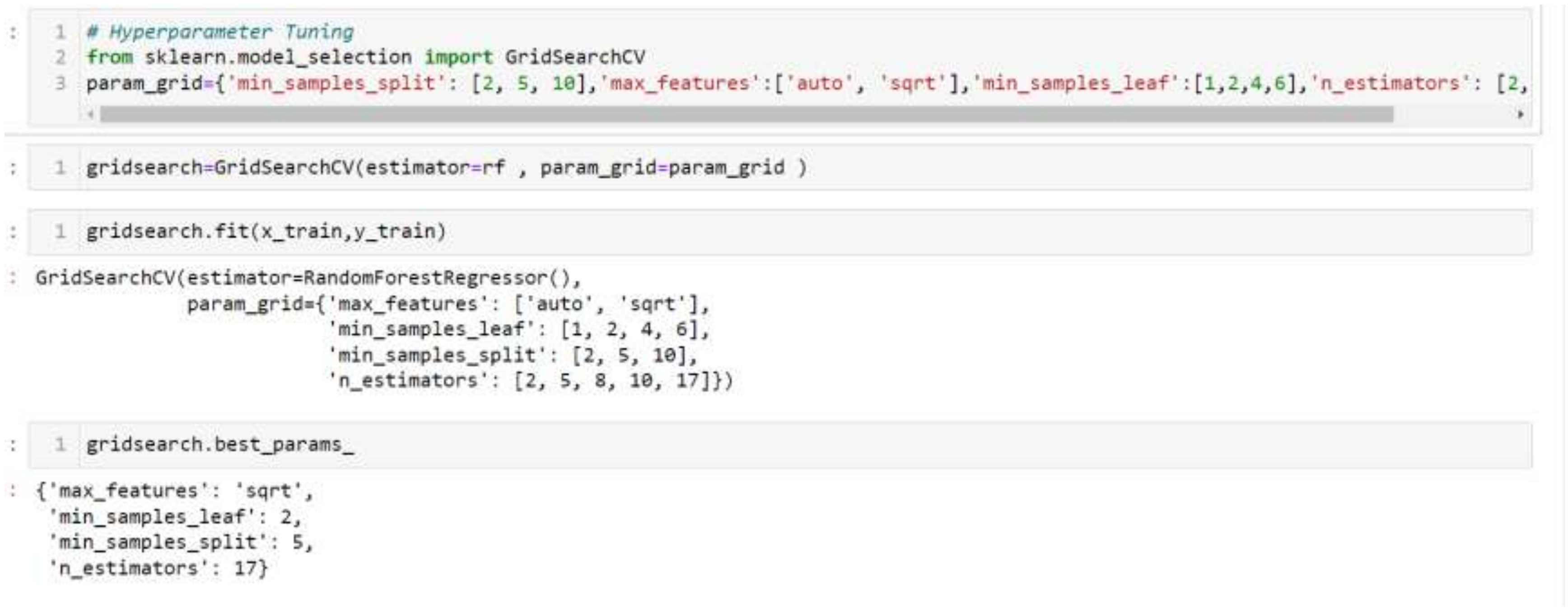
```python
# Cross validation scores for all models
from sklearn.model_selection import cross_val_score
for m in models:
    c_v= cross_val_score(m, x_scaled, y, cv = 10)
    print ('Cross Validation Score for ',m, ' is :', c_v.mean())
    print (' ')
```

```
Cross Validation Score for  LinearRegression()  is : 0.8070324237682417
 
Cross Validation Score for  KNeighborsRegressor()  is : 0.7438954640638589
 
Cross Validation Score for  SVR()  is : -0.06088659516394675
 
Cross Validation Score for  RandomForestRegressor()  is : 0.8314994527794763
 
Cross Validation Score for  DecisionTreeRegressor()  is : 0.574229067871293
```

Choosing the best Model

- After analyzing all the models we have concluded that RandomForestRegressor() model gives the best R2 score and cross validation score. And based on the R2 score we have chosen RandomForestRegressor() as the best model. We will use RandomForestRegressor() model for further analysis.

Hyperparameter Tuning of RandomForestRegressor() model using GridSearchCV



```
# Hyperparameter Tuning
from sklearn.model_selection import GridSearchCV
param_grid={'min_samples_split': [2, 5, 10],'max_features':['auto', 'sqrt'],'min_samples_leaf':[1,2,4,6],'n_estimators': [2,
```

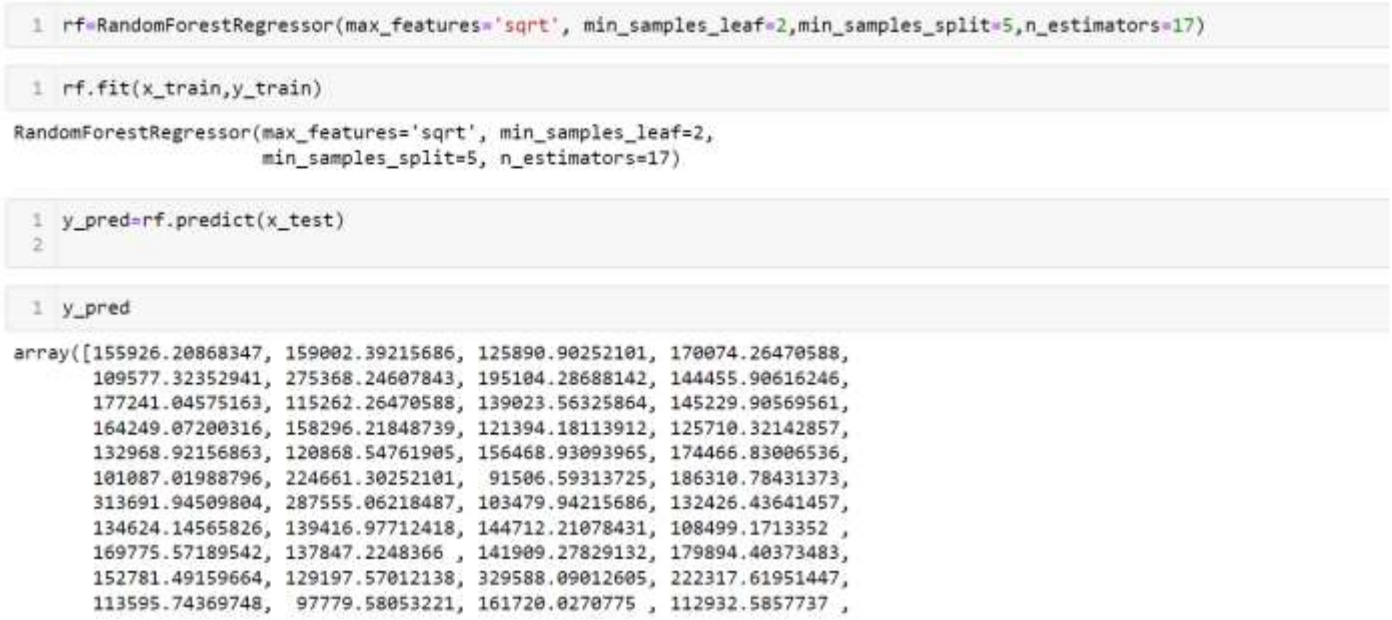
```
gridsearch=GridSearchCV(estimator=rf , param_grid=param_grid )
```

```
gridsearch.fit(x_train,y_train)
```

```
GridSearchCV(estimator=RandomForestRegressor(),
             param_grid={'max_features': ['auto', 'sqrt'],
                         'min_samples_leaf': [1, 2, 4, 6],
                         'min_samples_split': [2, 5, 10],
                         'n_estimators': [2, 5, 8, 10, 17]})
```

```
gridsearch.best_params_
```

```
{'max_features': 'sqrt',
 'min_samples_leaf': 2,
 'min_samples_split': 5,
 'n_estimators': 17}
```



```
rf=RandomForestRegressor(max_features='sqrt', min_samples_leaf=2,min_samples_split=5,n_estimators=17)
```

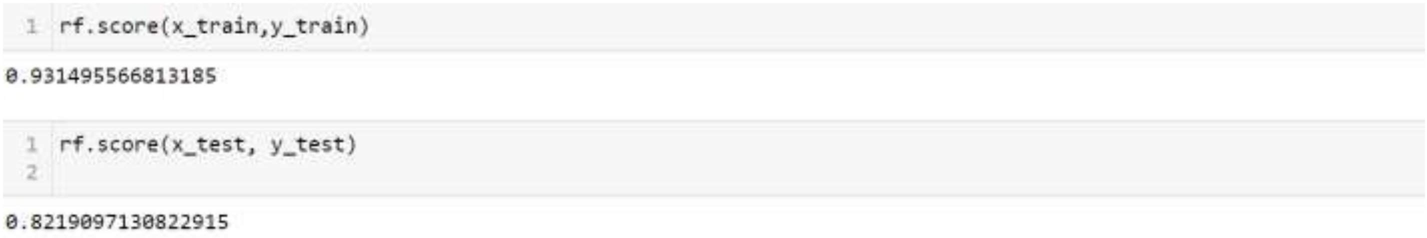
```
rf.fit(x_train,y_train)
```

```
RandomForestRegressor(max_features='sqrt', min_samples_leaf=2,
                      min_samples_split=5, n_estimators=17)
```

```
y_pred=rf.predict(x_test)
```

```
y_pred
```

```
array([155926.20868347, 159002.39215686, 125890.90252101, 170074.26470588,
       109577.32352941, 275368.24607843, 195104.28688142, 144455.90616246,
       177241.04575163, 115262.26470588, 139023.56325864, 145229.90569561,
       164249.07200316, 158296.21848739, 121394.18113912, 125710.32142857,
       132968.92156863, 120868.54761905, 156468.93093965, 174466.83006536,
       101087.01988796, 224661.30252101,  91506.59313725, 186310.78431373,
       313691.94509804, 287555.06218487, 103479.94215686, 132426.43641457,
       134624.14565826, 139416.97712418, 144712.21078431, 108499.1713352 ,
       169775.57189542, 137847.2248366 , 141909.27829132, 179894.40373483,
       152781.49159664, 129197.57012138, 329588.09012605, 222317.61951447,
       113595.74369748,  97779.58053221, 161720.0270775 , 112932.5857737 ,
```



```
rf.score(x_train,y_train)
```

```
0.931495566813185
```

```
rf.score(x_test, y_test)
```

```
0.8219097130822915
```

- After using Hyperparameter tuning the accuracy score for RandomForestRegressor Model has increased.

Conclusions

- The houses which have very excellent overall quality like material and finish of the house have high sale price. Also we have observed from the plot that as the overall quality of the house increases,the sale price also increases. That is there is good linear relation between SalePrice and OverallQual. So, if the seller builds the house according to these types of qualities that will increase the sale price of the house.
- There is a linear relation between the SalePrice and 1stFlrSF. As we have seen as the 1st floor area increases, sales price also increases moderately. So, people like to live in the houses which have only 1-2 floors and the cost of the house also increases in this case.
- Also, we have seen the positive linear relation between the SalePrice and GarageArea. As size of garage area increases, sale price also increases.
- There is positive linear relation between sale price and TotalBsmtSF. As total basement area increases, sale price also increases.
- Using the features which have some relation with target we have built many ML models also seen the increase in accuracy of the best model.