FLIP ROBO

PROJECT REPORT ON :
“HOUSING PROJECT”

SUBMITTED BY :
AMAN KUMAR PATEL

ACKNOWLADGEMENT

I would like to express my special thanks to “FLIP ROBO” team , who has given me this opportunity to deal with this amazing project and it specially thanks to improve my analyzing skills with the work o this project , And I would like to give a very big thanks to “MD. KASHIF” (SME) sir who always helps me a lot . And also I would like to thanks to all the members of FLIP ROBO who always encouraged me and also ready to help me and all of the team member is very co-oprative.

A huge thanks to my academic team “Data trained” who are the reason behind what I am today. Last but not least my parents who have been my backbone in every step of my life. And also thank you for many other persons who has helped me directly or indirectly to complete the project.

Contents:

INTRODUCTION:

1.1 Business Problem Framing:

Houses are one of the necessary need 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. 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?**

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.

Technical Requirements:

- **Data contains 1460 entries each having 81 variables.**
- **Data contains Null values. You need to treat them using the domain knowledge and your own understanding.**
- **Extensive EDA has to be performed to gain relationships of important variable and price.**
- **Data contains numerical as well as categorical variable. You need to handle them accordingly.**
- **You have to build Machine Learning models, apply regularization and determine the optimal values of Hyper Parameters.**
- **You need to find important features which affect the price positively or negatively.**
- **Two datasets are being provided to you (test.csv, train.csv).**

You will train on train.csv dataset and predict on test.csv file. The “Data file.csv” and “Data description.txt” are enclosed with this file

1.2 Conceptual Background of the Domain Problem

In today time the Real Estate market is one of the most competitive terms of pricing houses and the property price is one of the important module in decision making for both buyers and investors. And also the price is decidable with also the location and also the good envoirment and finding property finding new customers and also finding some suitable policies.

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

Some factors Which affect the land price have to be studied and their impact on price has also to be modelled. An analysis of the past data is to be considered. It is inferred that establishing a simple linear mathematical relationship for these time-series data is found not viable for forecasting. Hence it became imperative to establish a non-linear model which can well fit the data characteristic to analyse and forecast future trends. As the real estate is fast developing sector, the analysis and forecast of land prices using mathematical modelling and other scientific techniques is an immediate urgent need for decision making by all those concerned. The increase in population as well as the industrial activity is
attributed to various factors, the most prominent being the recent spurt in the knowledge sector viz. Demand for land started of showing an upward trend and housing and the real estate activity started booming. The need for predicting the trend in land prices was felt by all in the industry viz. the Government, the regulating bodies, lending institutions, the developers and the investors. Therefore, in this project report, we present various important features to use while predicting housing prices with good accuracy. We can use regression models, using various features to have lower Residual Sum of Squares error. While using features in a regression model some feature engineering is required for better prediction.
The primary aim of this report is to use these Machine Learning Techniques and curate them into ML models which can then serve the users. The main objective of a Buyer is to search for their dream house which has all the amenities they need. Furthermore, they look for these houses/Real estates with a price in mind and there is no guarantee that they will get the product for a deserving price and not overpriced. Similarly, A seller looks for a certain number that they can put on the estate as a price tag and this cannot be just a wild guess, lots of research needs to be put to conclude a valuation of a house.

1.4 Motivation for the Problem Undertaken

I have 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. The relationship between house prices and the economy is an important motivating factor for predicting house prices.

2. Analytical Problem Framing

2.1 Mathematical/ Analytical Modeling of the Problem

In this project it is containing two dtaset one of this train dataset and the another one is test dataset . And in the all dataset I found that saleprice is the my target variable and I have also built a model using train dataset with the prediction of dataset. And next point is when I noticed the entries of target columns which was continuous and that means this is a regression problem. Also, I observed some unnecessary entries in most of the columns like in some columns I found more than so much null values and more than huge numbers of zero values so I decided to drop those columns. and as we know If I keep those columns as it is, it will create large skewness in the model. While checking the null values in the datasets I found many columns with nan values and I replaced those nan values with suitable entries like mean for numerical columns and mode for categorical columns. To get better insight on the features I have used ploting like distribution plot, bar plot, reg plot and strip plot. With these ploting I was able to understand the relation between the features in better manner. Also, I found outliers and skewness in the dataset so I removed outliers using percentile method and I removed skewness using yeo-johnson method. I have used all the regression models while building model then tunned the best model and saved the best model. At last I have predicted the sale price fot test dataset using the saved model of train dataset.

2.2 Data Sources and their formats

That whole Data is provided by the my internship company FLIP ROBO. In the from of csv format.
And also , I got two dataset one is the train dataset which is 1168 rows and 81 columns and the second one is test dataset which is having 292 rows and 81 columns . And In all datasets is having float values, integers and string values

I can merge these two datasets and perform my analysis, but I have not done that because of data leakage issue. This is how my datasets looks for me when I import those datasets to my python

2.3 Data Preprocessing Done

- first step was I imported required libraries and I have imported both the train and test datasets which were in csv form.

- after that I did check all the statistical analysis like checking shape, nunique, value counts, info unwanted entries etc..... When I was checking the info of the datasets I found some columns with more than 80% null values, so these columns can create skewness in datasets so I decided to drop those columns.

-While checking for null values I found null values in most of the columns and I have used imputation method to replace those null values (mode for categorical column and mean for numerical columns).

-After that I was looking into the value counts I noticed some columns with more than 85% zero values this also creates skewness in the model and there are chances of getting model bias so I have dropped those columns with more than 85% zero values.

- In Id and Utilities column the unique counts. which means all the entries in Id column have different values is the identity number given for particular asset and all the entries in Utilities column were same so these two column will not help us in model building. So I decided to drop those columns, Next as a part of feature extraction I converted all the year columns to there respective age. Thinking that age will help us more than year.

- And all these steps were performed to both train and test datasets separately and simultaneously.

2.4 Data Inputs-Logic-Output Relationships

- **I have used plots for each pair of categorical features that shows the relationship between the median sale price for the sub categories in each and every categorical feature And for continuous numerical variables I have used reg plot to show the relationship between continuous numerical variable and target variable.**

- **I found that there is a linear relationship between continuous numerical variable and SalePrice**

2.5 Hardware and Software Requirements and Tools Used

Software required :

-- ANACONDA (jupyter notebook)

Library required:

To run the program and to build the model we need some basic libraries as follows:

- import pandas as pd: pandas is a popular Python-based data analysis tool which can be imported using import pandas as pd. It presents a diverse range of utilities, ranging from parsing multiple file formats to converting an entire data table .

- import numpy as np: NumPy is the fundamental package for scientific computing in Python. It is a Python library which provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

- import seaborn as sns: Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

- **Import matplotlib.pyplot as plt: matplotlib.pyplot is a collection of functions . Each pyplot function makes some changing to a figure and it is related to plots: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.**

- **from sklearn.preprocessing import OrdinalEncoder**
- **from sklearn.preprocessing import StandardScaler**
- **from statsmodels.stats.outliers_influence import variance_inflation_factor**
- **from sklearn.ensemble import RandomForestRegressor**
- **from sklearn.tree import DecisionTreeRegressor**
- **from sklearn.ensemble import GradientBoostingRegressor**
- **from sklearn.ensemble import ExtraTreesRegressor**
- **from sklearn.metrics import classification_report**
- **from sklearn.model_selection import cross_val_score**

HardWare Used:

Intel Core i5-2520M

3. Data Analysis and Visualization

3.1 Identification of possible problem-solving approaches (methods)

I found null values and I used imputation method to replace null values. and also see some outliers To remove outliers I have used IQR method. And also get some skewness And to remove skewness I have used yeo-johnson method. And according to dataset it is necessary To encode the categorical columns I have use Ordinal Encoding. Use of Pearson’s correlation coefficient to check the correlation between dependent and independent features. Also I have used standardization. Then followed by model building with all regression algorithms. And give me a good accuracy.

3.2 Testing of Identified Approaches (Algorithms)

Saleprice was my target and it was having a continuous column so this is having regression problem. And I have used all regression algorithms to build my model. By looking into the difference of r2 score and cross validation score I found GRADIENTBOOSTINGREGRESSOR as a best model with least difference. Also to get the best model we have to run through multiple models and to avoid the confusion of overfitting we have go through cross validation. Here the list of algorithms.

- RandomForestRegressor
- ExtraTreesRegressor
- GradientBoostingRegressor
- DecisionTreeRegressor

3.3 Key Metrics for success in solving problem under consideration

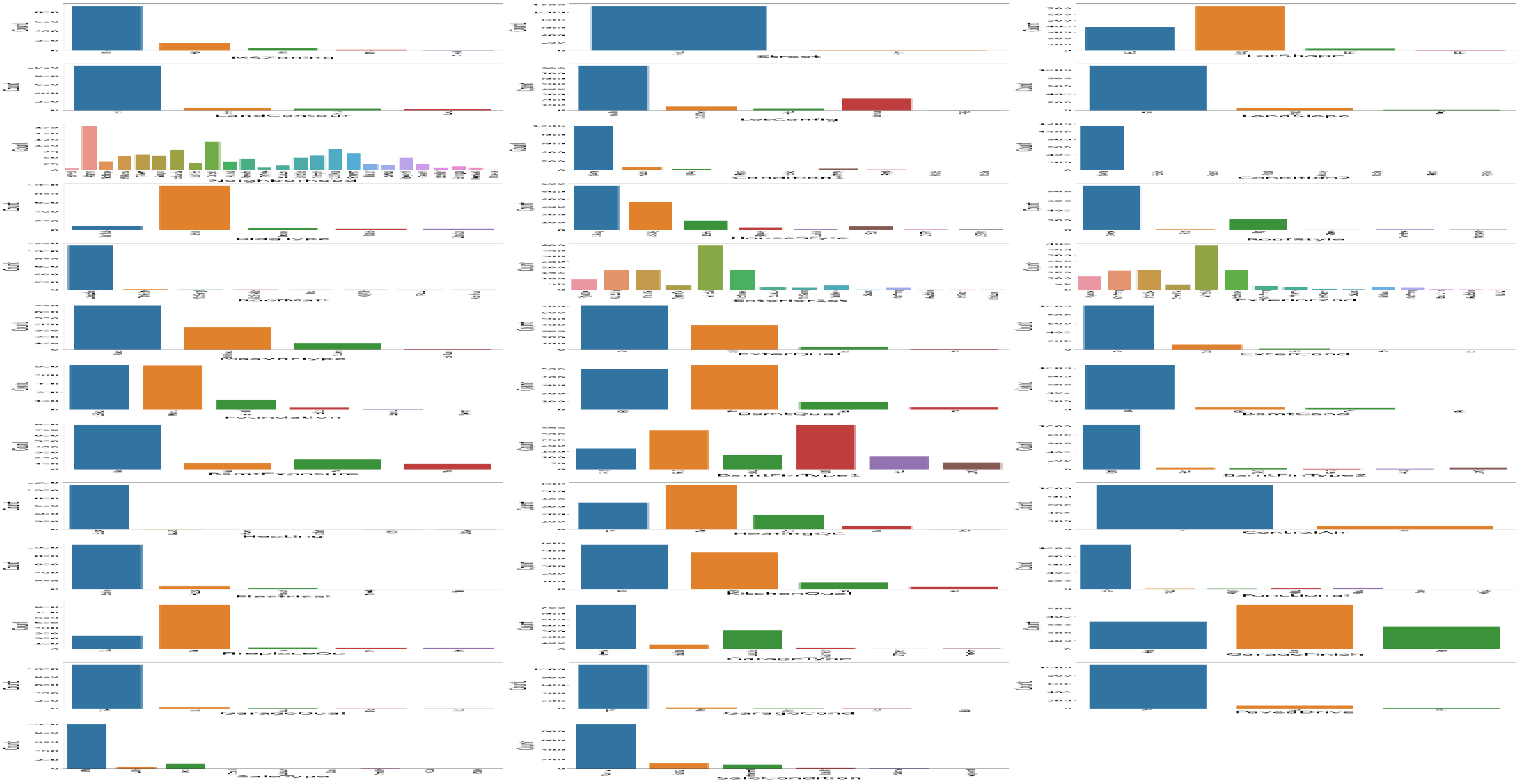
I have used the following metrics:

- I have used mean absolute error which shows the magnitude of difference between the prediction of an observation and the true value of that observation.
- I have used r2 score which tells us about the accuracy of model.
- I have used root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions.

3.4 Visualization

I have used bar plots to see the relation of categorical feature and I have used 2 types of plots for numerical columns one is strip plot for ordinal features and other is reg plot for continuous features

Bar plots for categorial columns.



Obseravtion:

It is found that Residential Low Density zoning has maximum count, for the feature general zoning classification of the sale(MSZoning).
 In Paved streets we can observe maximum count, for the feature Type of road access to property(Street).
Regular shaped property has maximum count, for the feature General shape of property(LotShape). Near Flat/Level property has maximum count, for the feature Flatness of the property(LandContour). Inside lot configured property has maximum count, for the feature Lot configuration(LotConfig). Gentle sloped property has maximum count, for the feature Slope of property(LandSlope).
If the property is located in North Ames then count is good compared to other locations, for the feature Physical locations within Ames city limits(Neighborhood).
If the Proximity to various conditions-1 is normal then count is high for the feature Proximity to various conditions(Condition1).
 If the Proximity to various conditions-2 is normal then count is high for the feature Proximity to various conditions (if more than one is present)(Condition2).
 Single-family Detached dwelling has maximum count for the feature Type of dwelling(BldgType).
 One story dwelling housestyle has maximum count for the feature Style of dwelling(HouseStyle).
 For Gable roof style the count is high for the feature Type of roof(RoofStyle).
 For Standard (Composite) Shingle roof material the count is high for the feature Roof material(RoofMatl).
 For Vinyl Siding exterior-1 covering on house has maximum counts for the feature Exterior covering on house(Exterior1st).
 For Vinyl Siding exterior-2 covering on house has maximum counts for the feature Exterior covering on house (if more than one material)(Exterior2nd). For Masonry veneer type(MasVnrType) None has maximum count. 17.For Typical/Average(TA) quality of the material on the

Typical Functionality has highest count for Home functionality (Assume typical unless deductions are warranted)(Functional).

For good Fireplace quality the count is high for the feature Fireplace quality(FireplaceQu).

If Garage location Attached to home then the count is high, for the feature Garage location(GarageType).

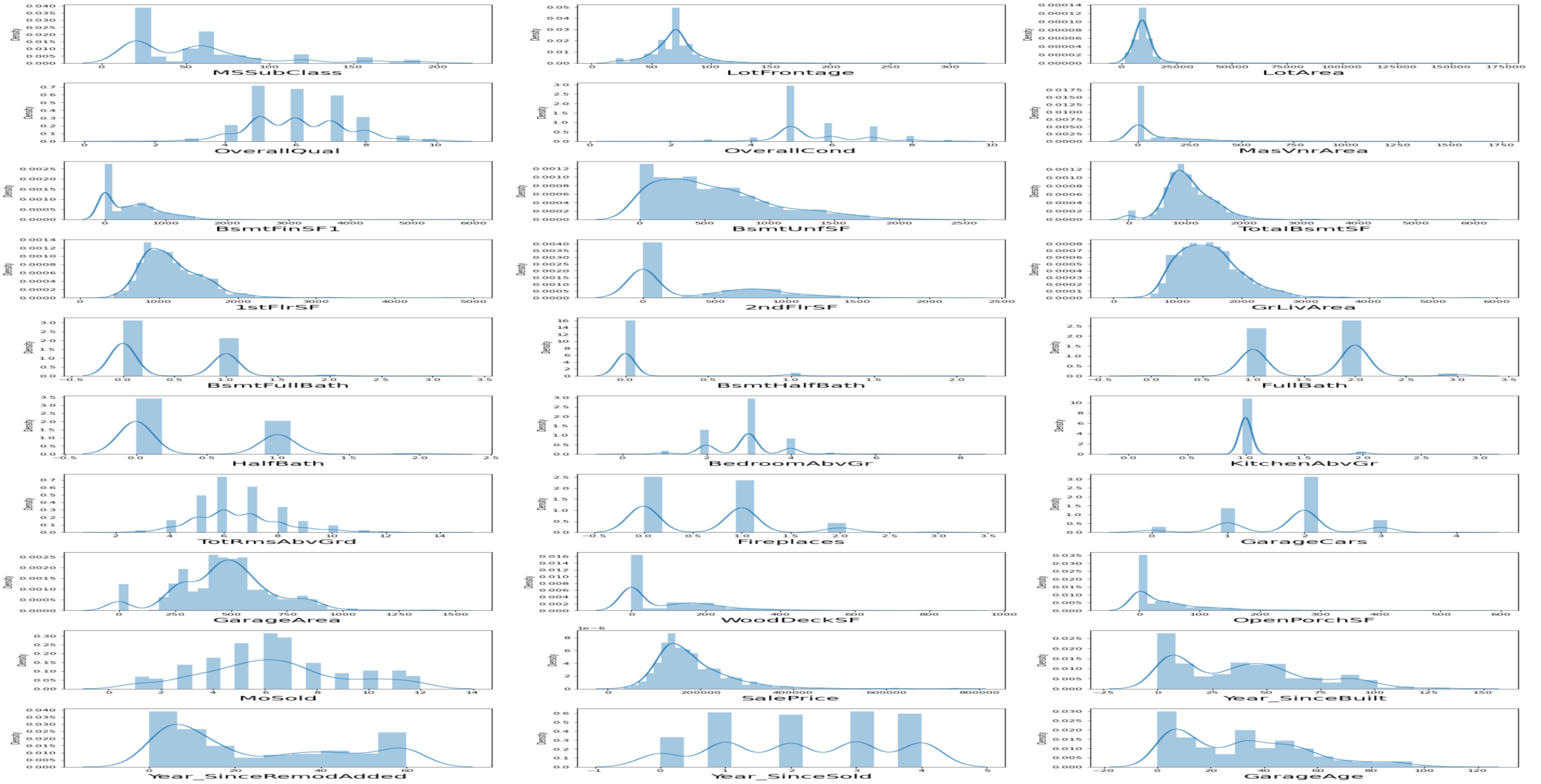
For Unfinished Interior of the garage the count is maximum, for the feature Interior finish of the garage(GarageFinish).

For Typical/Average(TA) Garage quality the count is high, for the feature Garage quality(GarageQual).

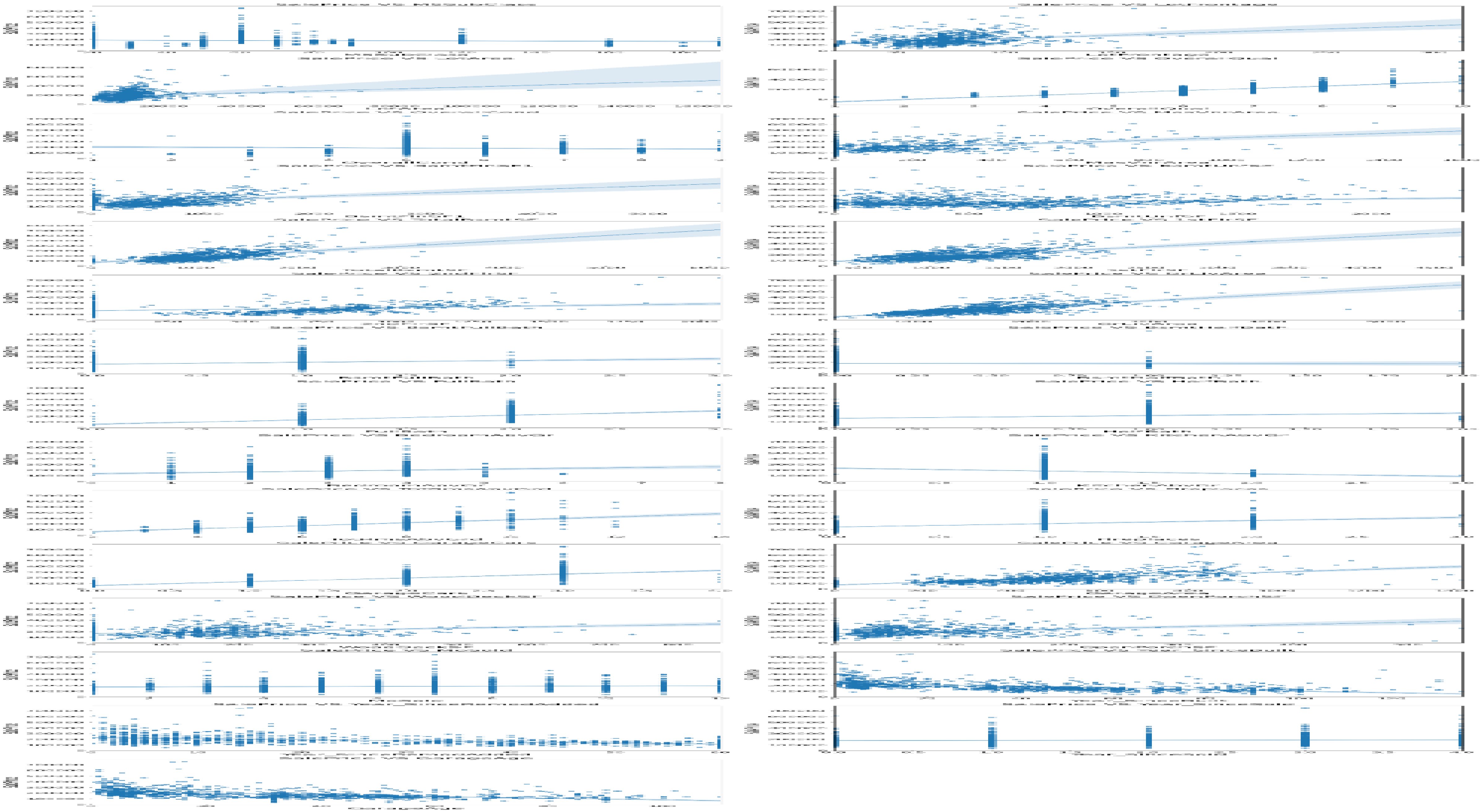
For Typical/Average(TA) Garage condition the count is high, for the feature Garage condition(GarageCond). For Paved driveway the count is maximum, for the feature Paved driveway(PavedDrive).

For Warranty Deed - Conventional type of sales the count is maximum, for the feature Type of sale(SaleType). For Normal sales condition the count is high, for the feature Condition of sale(SaleCondition).

Distribution plots for numerical columns



Reg plots for numerical columns



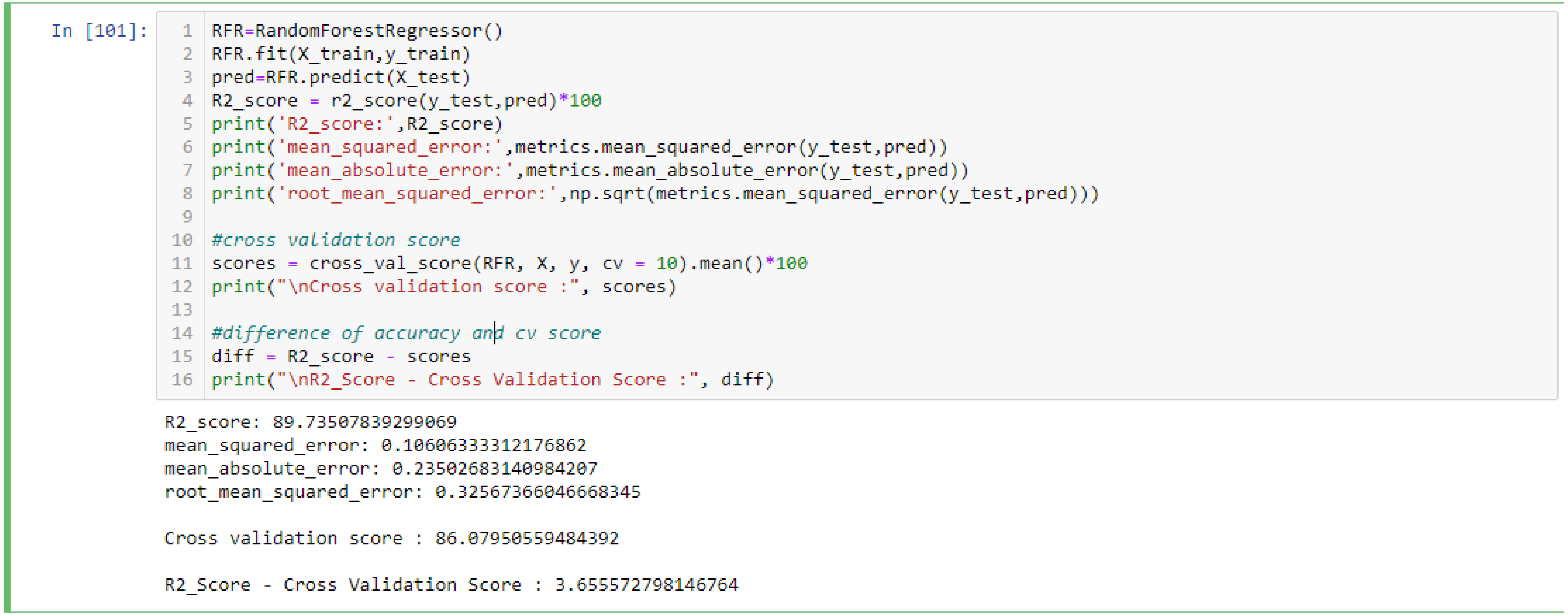
OBSERVATION

1.As Linear feet of street connected to property(LotFrontage) is increseing sales is decreasing and the SalePrice is rangeing between 0-3 lakhs.
2.As Lot size in square feet(LotArea) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
3.As Masonry veneer area in square feet(MasVnrArea) is increasing sales is decreasing and saleprice is rangeing between 0-4 lakhs.
4.As Type 1 finished square feet(BsmtFinSF1) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
5.As Unfinished square feet of basement area(BsmtUnfSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs. There are some outliers also.
6.As Total square feet of basement area(TotalBsmtSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
7.As First Floor square feet(1stFlrSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
8.As Second floor square feet(2ndFlrSF) is increseing sales is increasing in the range 500-1000 and the saleprice is in between 0-4 lakhs.
9.As Above grade (ground) living area square feet(GrLivArea) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
10.As Size of garage in square feet(GarageArea) is increseing sales is increseing and the saleprice is in between 0-4 lakhs.
11.As Wood deck area in square feet(WoodDeckSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
12.As Open porch area in square feet(OpenPorchSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
13.As Year_SinceBuilt is increseing sales is decreasing and the saleprice is high for newly built building and the sales price is in between 0-4 lakhs.

3.5 Run and Evaluate selected models

MODEL BUILDING:

Randomforestregressor:



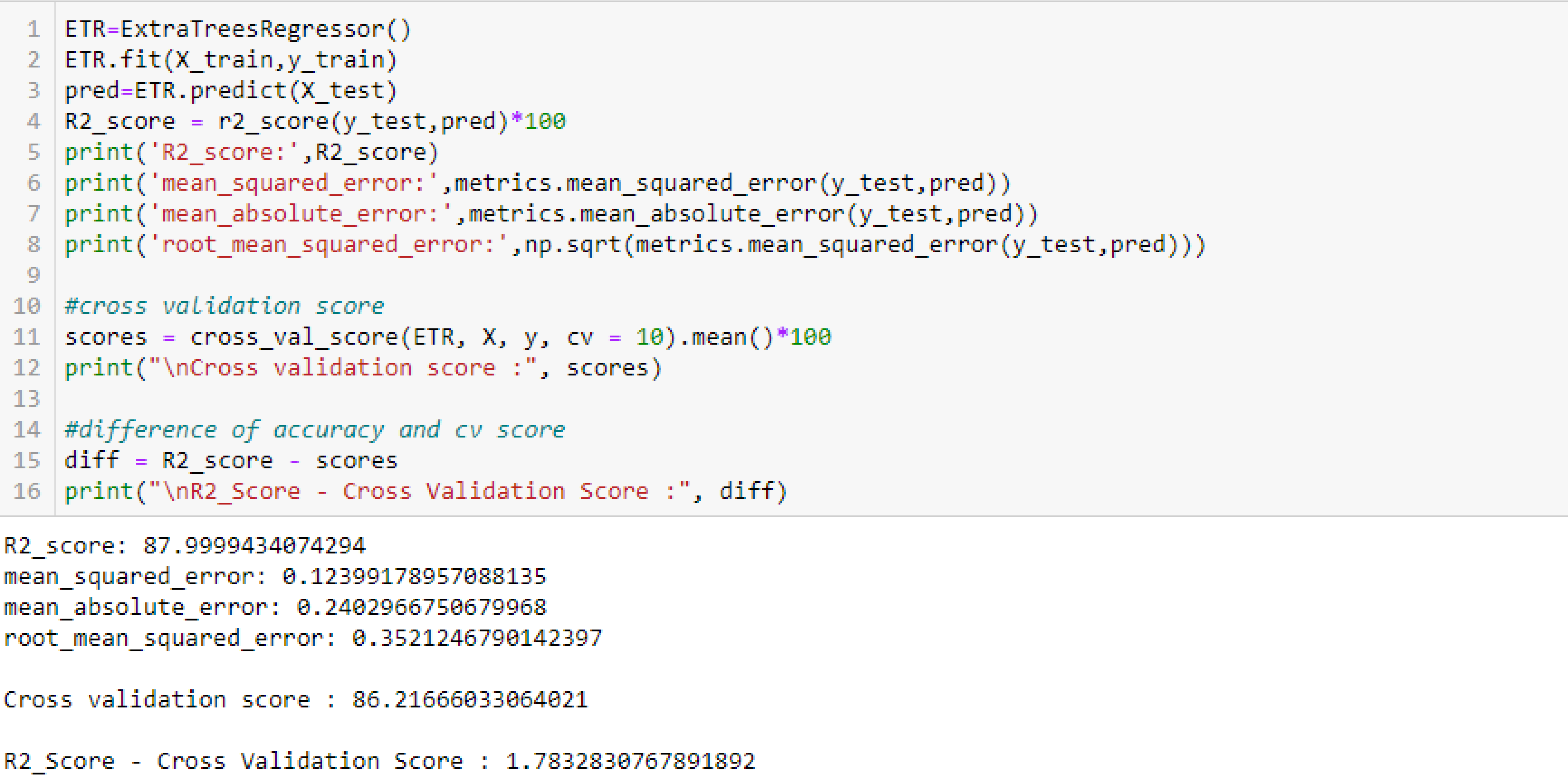
In [101]:

```
RFR=RandomForestRegressor()
RFR.fit(X_train,y_train)
pred=RFR.predict(X_test)
R2_score = r2_score(y_test,pred)*100
print('R2_score:',R2_score)
print('mean_squared_error:',metrics.mean_squared_error(y_test,pred))
print('mean_absolute_error:',metrics.mean_absolute_error(y_test,pred))
print('root_mean_squared_error:',np.sqrt(metrics.mean_squared_error(y_test,pred)))

#cross validation score
scores = cross_val_score(RFR, X, y, cv = 10).mean()*100
print("\nCross validation score :", scores)

#difference of accuracy and cv score
diff = R2_score - scores
print("\nR2_Score - Cross Validation Score :", diff)
```

```
R2_score: 89.73507839299069
mean_squared_error: 0.10606333312176862
mean_absolute_error: 0.23502683140984207
root_mean_squared_error: 0.32567366046668345

Cross validation score : 86.07950559484392

R2_Score - Cross Validation Score : 3.655572798146764
```

RandomForestRegressor is giving me 89% accuracy which is good but I have to use more Algorithms.

Extratreeregressor:

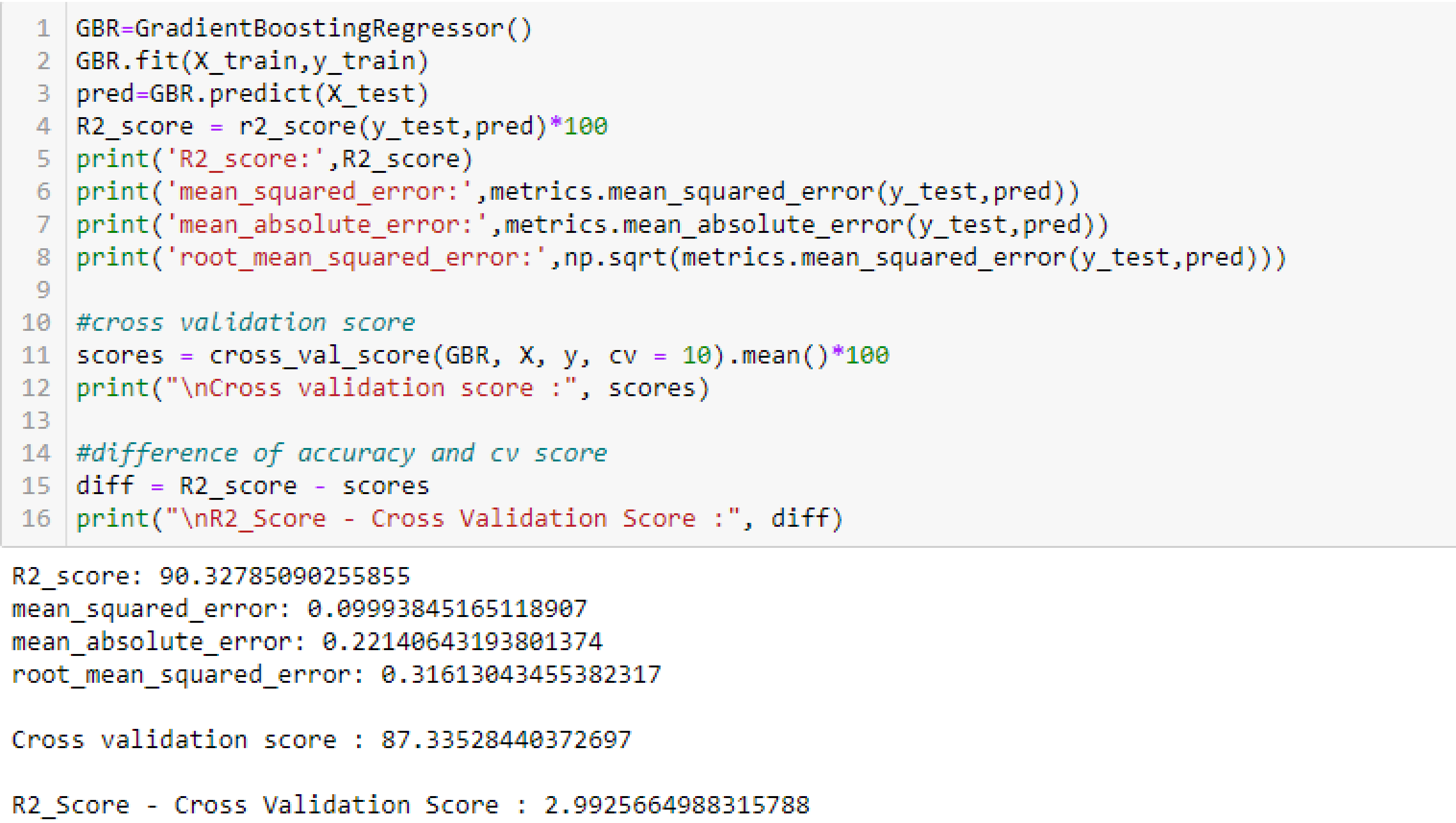


```python
ETR=ExtraTreesRegressor()
ETR.fit(X_train,y_train)
pred=ETR.predict(X_test)
R2_score = r2_score(y_test,pred)*100
print('R2_score:',R2_score)
print('mean_squared_error:',metrics.mean_squared_error(y_test,pred))
print('mean_absolute_error:',metrics.mean_absolute_error(y_test,pred))
print('root_mean_squared_error:',np.sqrt(metrics.mean_squared_error(y_test,pred)))

#cross validation score
scores = cross_val_score(ETR, X, y, cv = 10).mean()*100
print("\nCross validation score :", scores)

#difference of accuracy and cv score
diff = R2_score - scores
print("\nR2_Score - Cross Validation Score :", diff)
```

```
R2_score: 87.9999434074294
mean_squared_error: 0.12399178957088135
mean_absolute_error: 0.2402966750679968
root_mean_squared_error: 0.3521246790142397

Cross validation score : 86.21666033064021

R2_Score - Cross Validation Score : 1.7832830767891892
```

ExtraTreeRegressor is giving me 87% accuracy which is good but I have to use more Algorithms.

GradientBoostingRegressor:

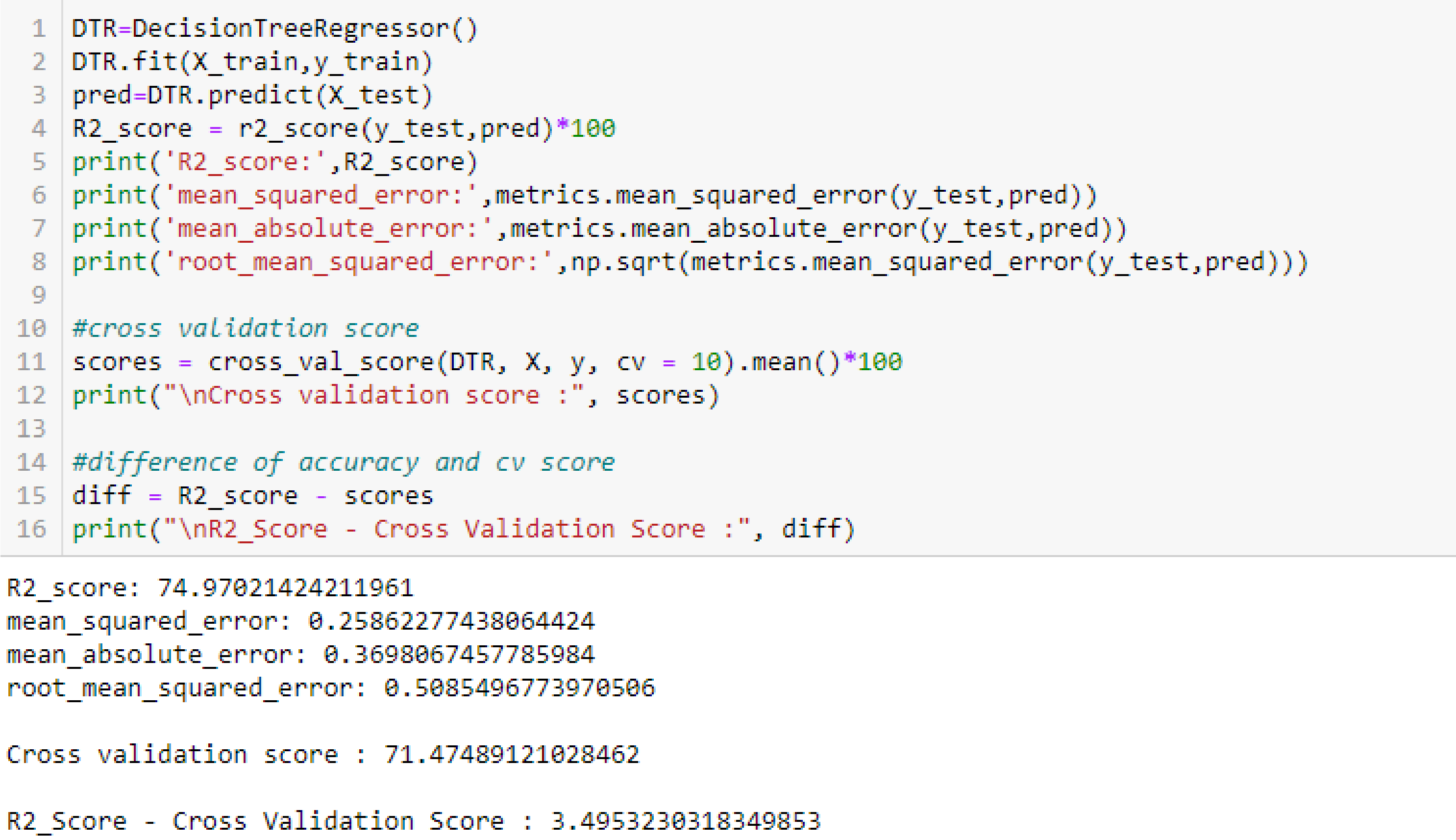


```python
GBR=GradientBoostingRegressor()
GBR.fit(X_train,y_train)
pred=GBR.predict(X_test)
R2_score = r2_score(y_test,pred)*100
print('R2_score:',R2_score)
print('mean_squared_error:',metrics.mean_squared_error(y_test,pred))
print('mean_absolute_error:',metrics.mean_absolute_error(y_test,pred))
print('root_mean_squared_error:',np.sqrt(metrics.mean_squared_error(y_test,pred)))

#cross validation score
scores = cross_val_score(GBR, X, y, cv = 10).mean()*100
print("\nCross validation score :", scores)

#difference of accuracy and cv score
diff = R2_score - scores
print("\nR2_Score - Cross Validation Score :", diff)
```

```
R2_score: 90.32785090255855
mean_squared_error: 0.09993845165118907
mean_absolute_error: 0.22140643193801374
root_mean_squared_error: 0.31613043455382317

Cross validation score : 87.33528440372697

R2_Score - Cross Validation Score : 2.9925664988315788
```

GradientBoositngRegressor is giving me 90.3% accuracy which is good but I have to use more Algorithms.

Decisiontreeregssor:



```
DTR=DecisionTreeRegressor()
DTR.fit(X_train,y_train)
pred=DTR.predict(X_test)
R2_score = r2_score(y_test,pred)*100
print('R2_score:',R2_score)
print('mean_squared_error:',metrics.mean_squared_error(y_test,pred))
print('mean_absolute_error:',metrics.mean_absolute_error(y_test,pred))
print('root_mean_squared_error:',np.sqrt(metrics.mean_squared_error(y_test,pred)))

#cross validation score
scores = cross_val_score(DTR, X, y, cv = 10).mean()*100
print("\nCross validation score :", scores)

#difference of accuracy and cv score
diff = R2_score - scores
print("\nR2_Score - Cross Validation Score :", diff)
```

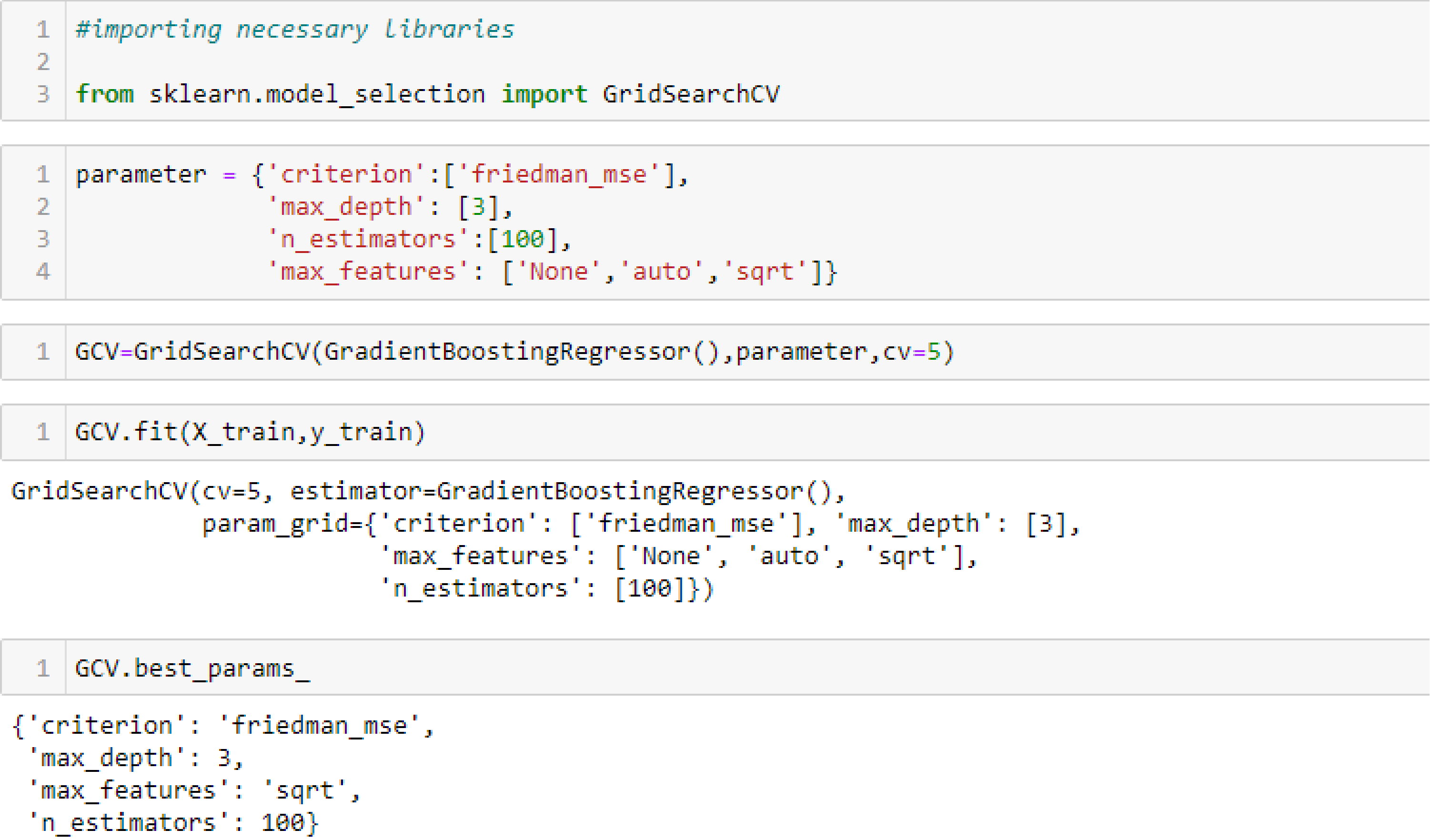
```
R2_score: 74.97021424211961
mean_squared_error: 0.25862277438064424
mean_absolute_error: 0.3698067457785984
root_mean_squared_error: 0.5085496773970506

Cross validation score : 71.47489121028462

R2_Score - Cross Validation Score : 3.4953230318349853
```

DecisionTreeRegressor is giving me 74% accuracy which is good but I have to use more Algorithms.

Now , here the after applying different models, I got the gradient bossting regressor as a best model , because it is giving me best accuracy as compare to other model.

And I will use GBR as a hyper tunning model .

Hyper parameter tunning:



```python
#importing necessary libraries

from sklearn.model_selection import GridSearchCV
```

```python
parameter = {'criterion':['friedman_mse'],
             'max_depth': [3],
             'n_estimators':[100],
             'max_features': ['None','auto','sqrt']}
```

```python
GCV=GridSearchCV(GradientBoostingRegressor(),parameter,cv=5)
```

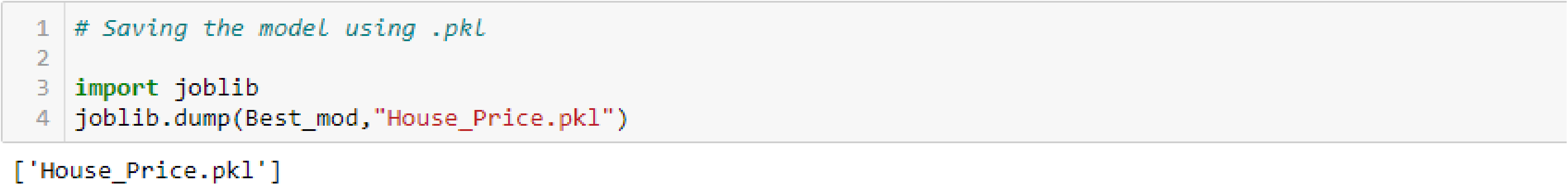
```python
GCV.fit(X_train,y_train)
```

```
GridSearchCV(cv=5, estimator=GradientBoostingRegressor(),
             param_grid={'criterion': ['friedman_mse'], 'max_depth': [3],
                         'max_features': ['None', 'auto', 'sqrt'],
                         'n_estimators': [100]})
```

```python
GCV.best_params_
```

```
{'criterion': 'friedman_mse',
 'max_depth': 3,
 'max_features': 'sqrt',
 'n_estimators': 100}
```

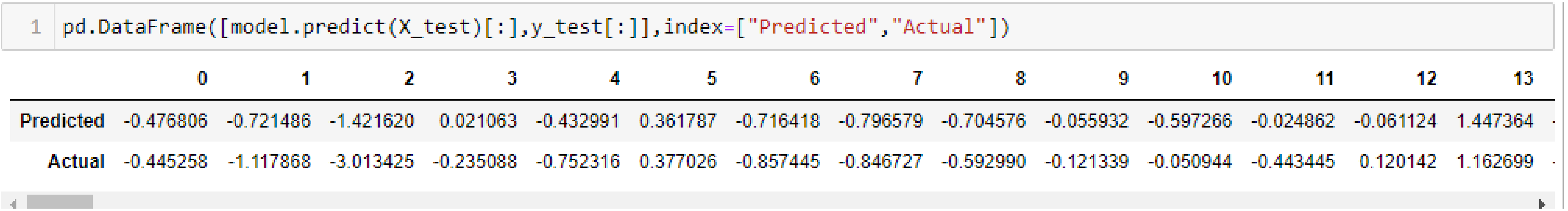
Saving models and predicting sales price:



```python
# Saving the model using .pkl

import joblib
joblib.dump(Best_mod,"House_Price.pkl")
```

```
['House_Price.pkl']
```

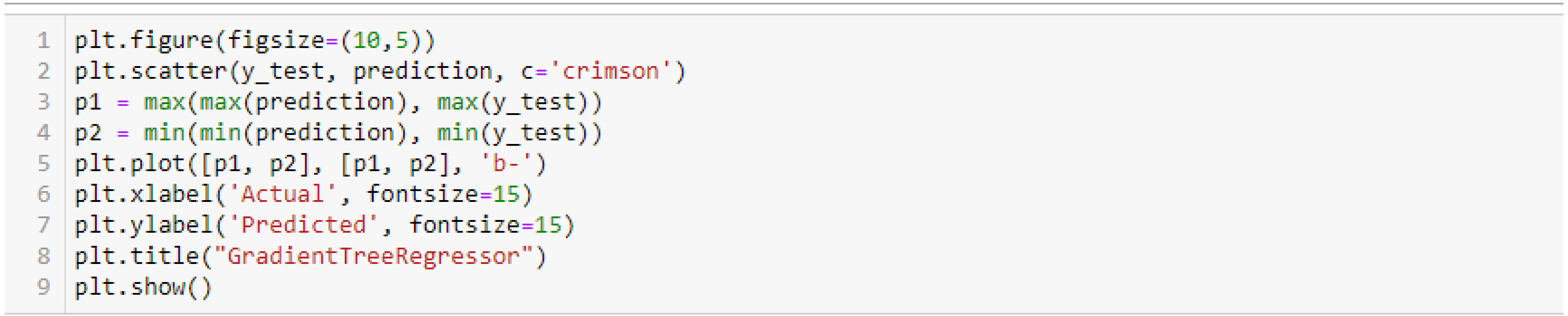
I have saved my best model using .pkl as follows.



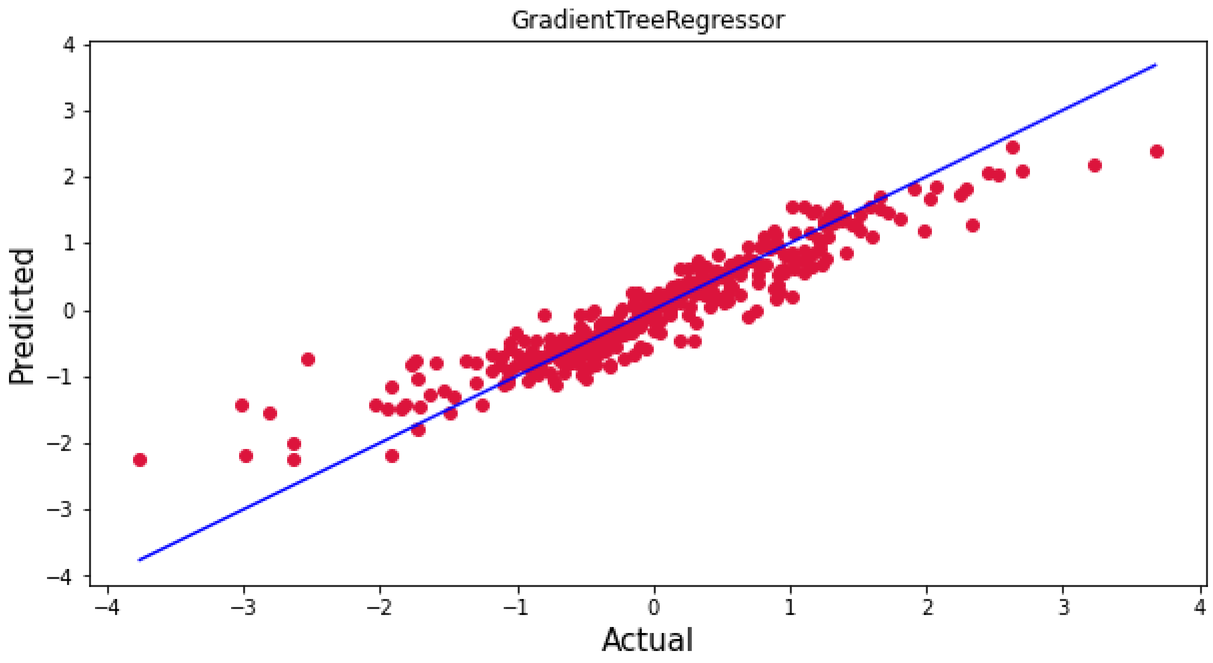
```python
pd.DataFrame([model.predict(X_test)[:],y_test[:]],index=["Predicted","Actual"])
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Predicted	-0.476806	-0.721486	-1.421620	0.021063	-0.432991	0.361787	-0.716418	-0.796579	-0.704576	-0.055932	-0.597266	-0.024862	-0.061124	1.447364
Actual	-0.445258	-1.117868	-3.013425	-0.235088	-0.752316	0.377026	-0.857445	-0.846727	-0.592990	-0.121339	-0.050944	-0.443445	0.120142	1.162699

Here we can see the actual and prediction values are some similar



```python
plt.figure(figsize=(10,5))
plt.scatter(y_test, prediction, c='crimson')
p1 = max(max(prediction), max(y_test))
p2 = min(min(prediction), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('Actual', fontsize=15)
plt.ylabel('Predicted', fontsize=15)
plt.title("GradientTreeRegressor")
plt.show()
```



Plotting Actual vs Predicted, To get better insight. Bule line is the actual line and red dots are the predicted values.

3.6 Interpretation of the Results

dataset was very diffrent as it had separate train and test datasets. We have to work with both datasets seprately.

Firstly, the datasets were having null values and zero entries in maximum columns so we have to be careful while going through the statistical analysis of the datasets.

And proper ploting for proper type of features will help us to get better insight on the data. I found maximum numerical continuous columns were in linear relationship with target column.

I notice a huge amount of outliers and skewness in the data so we have choose proper methods to deal with the outliers and skewness. If we ignore this outliers and skewness we may end up with a bad model which has less accuracy.

Then scaling both train and test dataset has a good impact like it will help the model not to get baised.

We have to use multiple models while building model using train dataset as to get the best model out of it.

And we have to use multiple metrics like mae, mse, rmse and r2_score which will help us to decide the best model.

I found ExtraTreesRegressor as the best model with 89.66% r2_score. Also I have improved the accuracy of the best model by running hyper parameter tunning.

4. Conclusion

4.1 Key Findings and Conclusions of the Study

In this project report, we have used machine learning algorithms to predict the house prices. We have mentioned the step by step procedure to analyze the dataset and finding the correlation between the features. Thus we can select the features which are not correlated to each other and are independent in nature. These feature set were then given as an input to five algorithms and a csv file was generated consisting of predicted house prices. Hence we calculated the performance of each model using different performance metrics and compared them based on these metrics. Then we have also saved the dataframe of predicted prices of test dataset

4.3 Learning Outcomes of the Study in respect of Data Science

I found that the dataset was quite interesting to handle as it contains all types of data in it. Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed and analysed. New analytical techniques of machine learning can be used in property research. The power of visualization has helped us in understanding
the data by graphical representation it has made me to understand what data is trying to say. Data cleaning is one of the most important steps to remove missing value and to replace null value and zero values with there respective mean, median or mode. This study is an exploratory attempt to use five machine learning algorithms in estimating housing prices, and then compare their results.

4.3 Limitations of this work and Scope for Future Work

First draw back is the data leakage when we merge both train and test datasets.
Followed by more number of outliers and skewness these two will reduce our model accuracy.

--Also, we have tried best to deal with outliers, skewness, null values and zero values. So it looks quite good that we have achieved a accuracy of 90.13% even after dealing all these drawbacks.

--Also, this study will not cover all regression algorithms instead, it is focused on the chosen algorithm, starting from the basic regression techniques to the advanced ones.

-- This model doesn't predict future prices of the houses mentioned by the customer. Due to this, the risk in investment in an apartment or an area increases considerably. To minimize this error, customers tend to hire an agent which again increases the cost of the process