FLIP ROBO

Project Name:-

PFA_Housing_Project

Name : Amit Matreja

ACKNOWLEDGMENT

I would like to express my special gratitude to "Flip Robo" team, who has given me this opportunity to deal with a beautiful dataset and it has helped me to improve my analyzation skills. And I want to express my huge gratitude to **Ms. Swati Mahaseth** (SME Flip Robo), she is the person who has helped me to get out of all the difficulties I faced while doing the project. She also encouraged me a lot with her unconditional support I have ended up with a beautiful Project. A huge thanks to my academic team "Data Trained".
Who taught me all the skills required to be job ready and for successful carrier.

Contents:

1. 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.
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. House Price prediction, is important to drive Real Estate efficiency. As earlier, House prices were determined by calculating the acquiring and selling price in a locality. Therefore, the House Price prediction model is very essential in filling the information gap and improve Real Estate efficiency. The aim is to predict the efficient house pricing for real estate customers with respect to their budgets and priorities. By analysing previous market trends and price ranges, and also upcoming developments future prices will be predicted. ... cost of property depending on number of attributes considered.
Now as a data scientist our work is to analyse the dataset and apply our skills towards predicting house price.

1.2 Conceptual Background of the Domain Problem:

The real estate market is one of the most competitive in terms of pricing and same tends to vary significantly based on numerous factors; forecasting property price is an important module in decision making for both the buyers and investors in supporting budget allocation, finding property finding stratagems and determining suitable policies. 5 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?

Why is house price prediction important?

➢ House Price prediction, is important to drive Real Estate efficiency. As earlier, House prices were determined by calculating the acquiring and selling price in a locality. Therefore, the House Price prediction model is very essential in filling the information gap and improve Real Estate efficiency.

There are three factors that influence the price of a house which include physical conditions, concept and location. Hence it becomes one of the prime fields to apply the concepts of machine learning to optimize and predict the prices with high accuracy. Therefore, in this project report we present various important features to use while predicting housing prices with good accuracy. While using features in a regression model some feature engineering is required for better prediction.

1.3 Review of Literature

The factors that 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 6 attributed to various factors, the most prominent being the recent spurt in the knowledge sector viz. Information Technology (IT) and Information technology enabled services. 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:

This particular problem has two datasets one is train data-set and the other is test data-set. I have built model using train data-set and predicted Saleprice for 7 test data-set. By looking into the target column, I came to know that the entries of SalePrice column were continuous and this was a Regression problem so I have to use all regression algorithms while building the model. Also, I observed some unnecessary entries in some of the columns like in some columns I found more than 80% null values and more than 85% zero values so I decided to drop those columns. If I keep those columns as it is, it will create high 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:

The data was collected and provided to me by my internship company – Flip Robo technologies in csv (comma separated values) format. The Company provided me two datasets one is train and other is test. I built model using train dataset and predicted SalePrice for test dataset. My train dataset w as having 1168 rows and 81 columns including target, and my test dataset was having 292 rows and 80 columns excluding target. In this particular datasets I have object, float and integer types of data. I did analysis of both the datasets separately so that we don’ t face any data leakage issue. This is how my datasets looks for me when I import those datasets to my jupyter Notebook.

2.3 Data PreProcessing Done :

- As a first step I have imported required libraries and I have imported both the datasets which were in csv format.
- Then I did all the statistical analysis like checking shape, nunique, value counts, info etc.....
- While checking the info of the datasets I found some columns with more than 80% null values, so these columns will create skewness in datasets so I decided to drop those columns.
- Then while looking into the value counts I found 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.
- 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).
- In Id and Utilities column the unique counts were 1168 and 1 respectively, which means all the entries in Id column are unique and ID is the identity number given for perticular 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 box plot for each pair of categorical features that shows the relation with the median sale price for all the sub categories in each categorical feature.
- And also 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:

While taking up the project we should be familiar with the Hardware and software required for the successful completion of the project. Here we need the following hardware and software.

Hardware used: -

1. Processor — core i5
2. RAM — 8 GB
3. HDD — 1TB

Software/s required: -

1.Anaconda

Libraries required :-

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

```
# importing all required libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt

import warnings
warnings.filterwarnings('ignore')
```

✓ **import pandas as pd:** pandas is a popular Python-based data analysis toolkit 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 into a numpy matrix array. This makes pandas a trusted ally in data science and machine learning.

✓ **import numpy as np:** NumPy is the fundamental package for scientific computing in Python. It is a Python library that 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 that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: 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 xgboost import XGBRegressor
✓ from sklearn.ensemble import GradientBoostingRegressor
✓ from sklearn.ensemble import ExtraTreesRegressor
✓ from sklearn.metrics import classification_report
✓ from sklearn.model_selection import cross_val_score

With this sufficient libraries we can go ahead with our analytical skills.

3.Data Analysis and Visualization

3.1 Identification of possible problem solving approach (methods):

I have used imputation method to replace null values. To remove outliers I have used percentile method. And to remove skewness I have used yeo johnson method. 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.

3.2 Testing of Identified Algorithms:

Since Saleprice was my target and it was a continuous column so this perticular problem was 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 ExtraTreesRegressor 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. Below are the list of regression algorithms I have used in my project.

- XGBRegressor
- DecisionTreeRegressor
- GradientBoostingRegressor
- ExtraTreesRegressor
- KNeighborsRegressor
- RandomForestRegressor

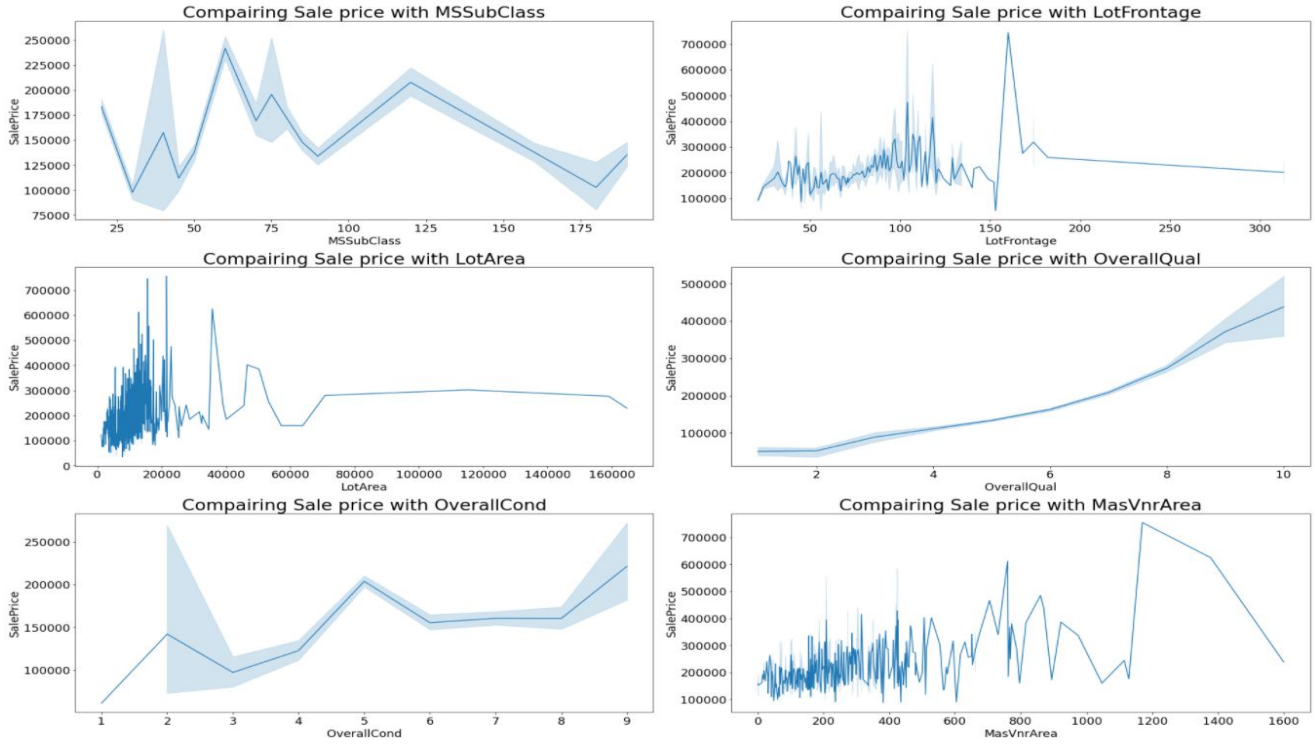
3.3 Key Metrics for success in solving Problem Under consideration:

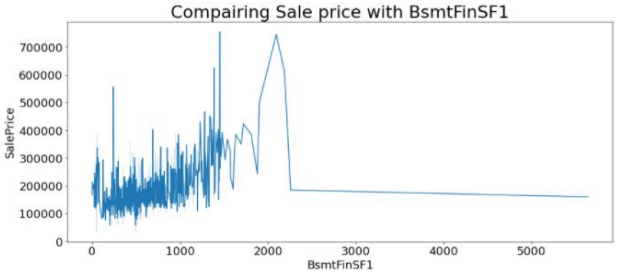
I have used the following metrics for evaluation: • I have used mean absolute error which gives magnitude of difference between the prediction of an observation and the true value of that observation. • I have used root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions. • I have used r2 score which tells us how accurate our model is.

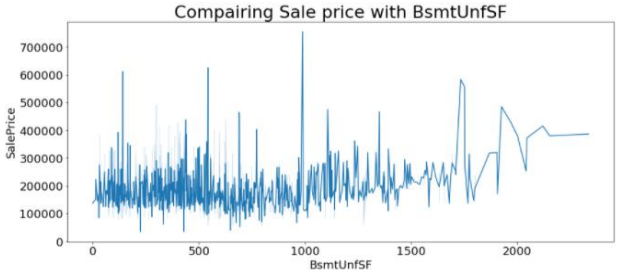
3.4 Visualizations:

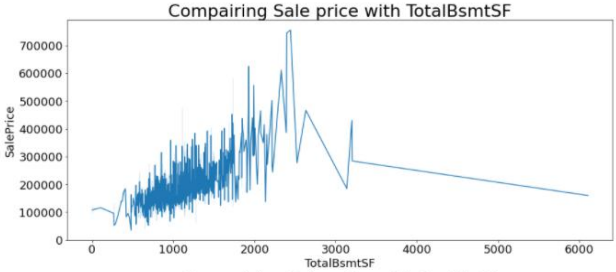
I have used bar plots to see the relation of categorical feature and I have used 2 types of plots for numerical columns one is strip plot for ordinal features and other is reg plot for continuous features.

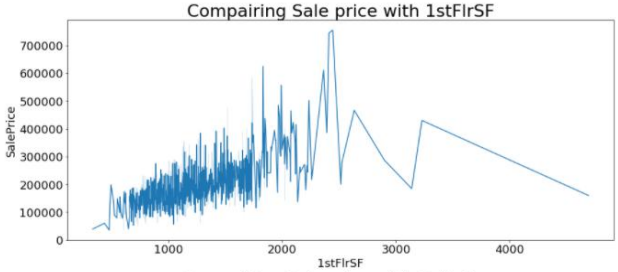
1. Visualization of numerical features with target:

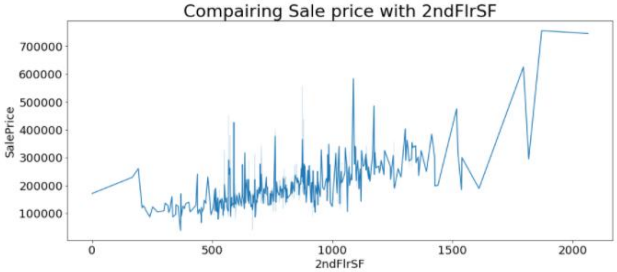


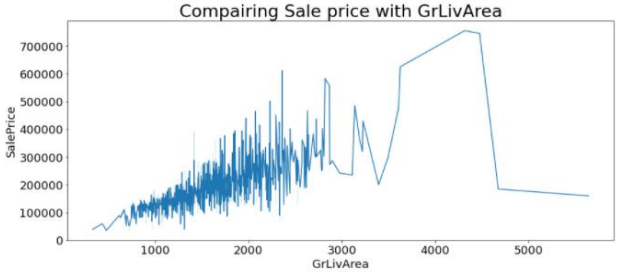

Compairing Sale price with BsmtFinSF1
SalePrice
BsmtFinSF1

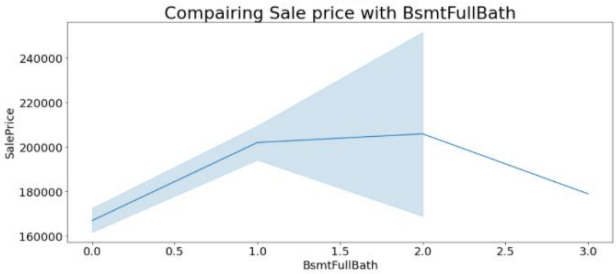

Compairing Sale price with BsmtUnfSF
SalePrice
BsmtUnfSF


Compairing Sale price with TotalBsmtSF
SalePrice
TotalBsmtSF

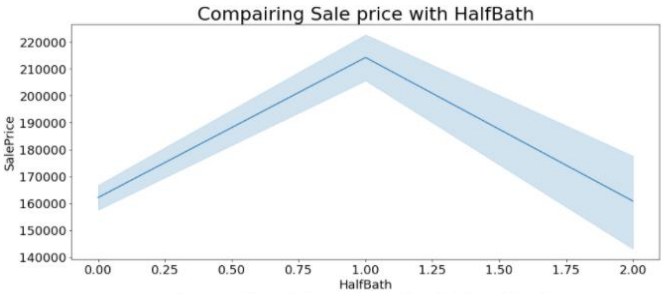

Compairing Sale price with 1stFlrSF
SalePrice
1stFlrSF

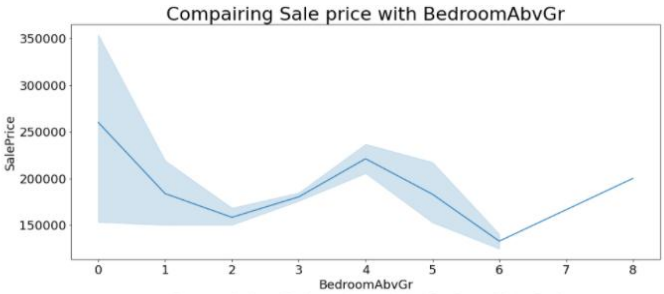

Compairing Sale price with 2ndFlrSF
SalePrice
2ndFlrSF

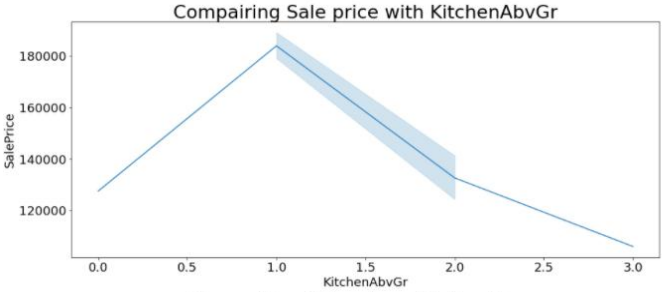

Compairing Sale price with GrLivArea
SalePrice
GrLivArea

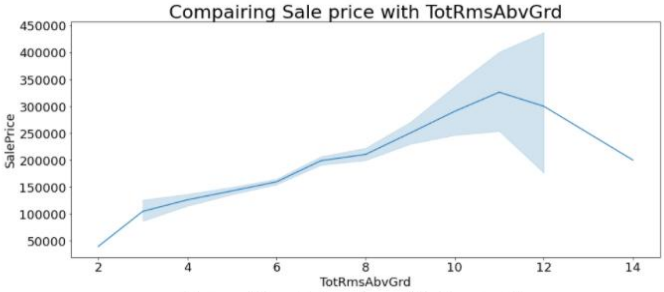

Compairing Sale price with BsmtFullBath
SalePrice
BsmtFullBath

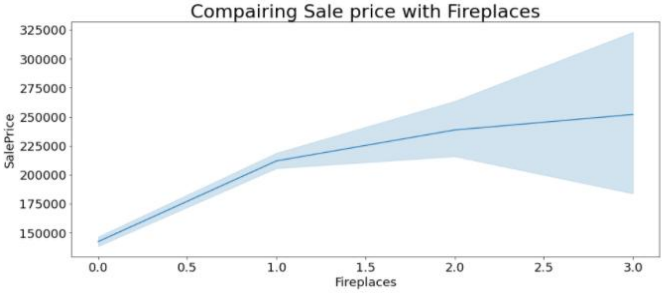

Compairing Sale price with FullBath
SalePrice
FullBath

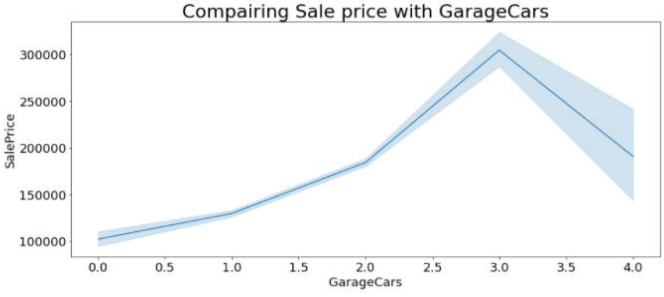

Compairing Sale price with HalfBath
SalePrice
HalfBath

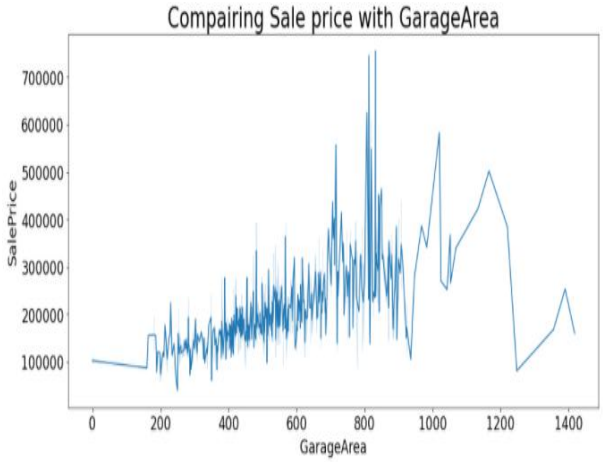

Compairing Sale price with BedroomAbvGr
SalePrice
BedroomAbvGr

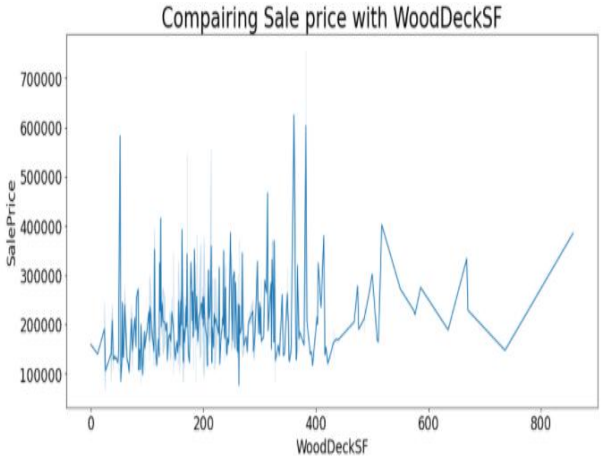

Compairing Sale price with KitchenAbvGr
SalePrice
KitchenAbvGr

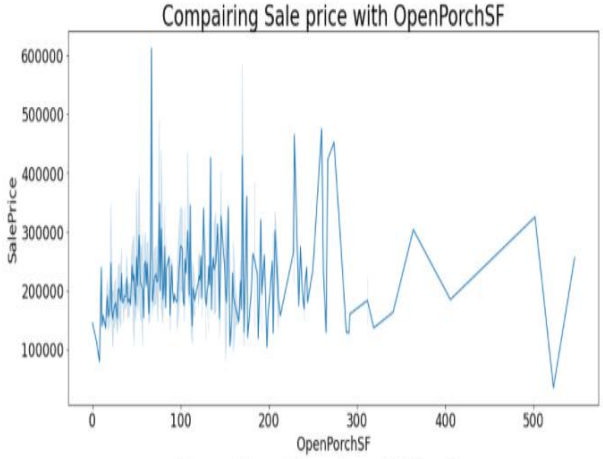

Compairing Sale price with TotRmsAbvGrd
SalePrice
TotRmsAbvGrd


Compairing Sale price with Fireplaces
SalePrice
Fireplaces

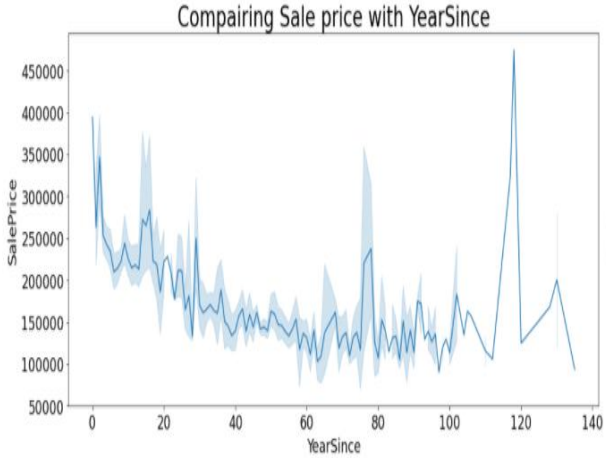

Compairing Sale price with GarageCars
SalePrice
GarageCars


Compairing Sale price with GarageArea
SalePrice
GarageArea

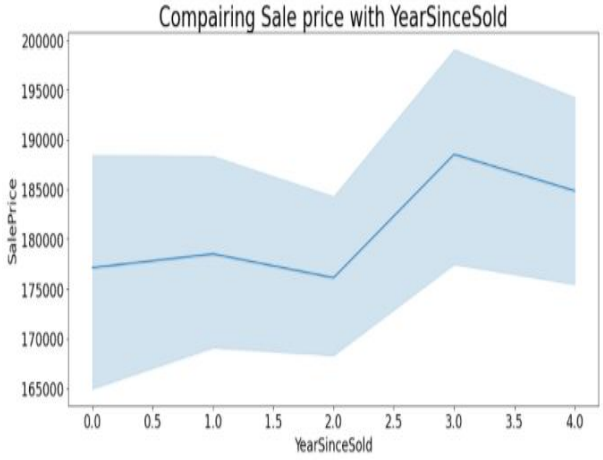

Compairing Sale price with WoodDeckSF
SalePrice
WoodDeckSF

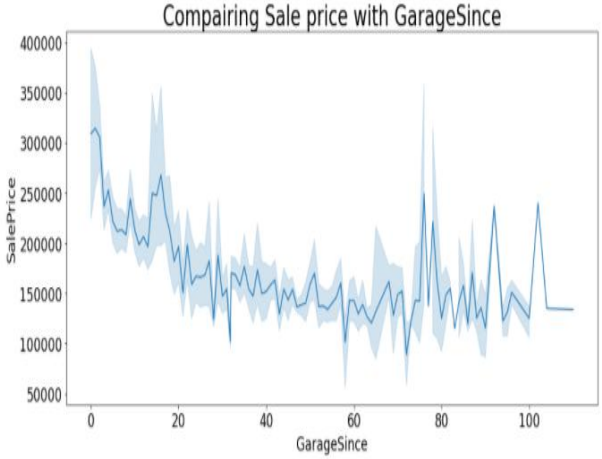

Compairing Sale price with OpenPorchSF
SalePrice
OpenPorchSF


Compairing Sale price with MoSold
SalePrice
MoSold


Compairing Sale price with YearSince
SalePrice
YearSince


Compairing Sale price with YearSinceRemodAdd
SalePrice
YearSinceRemodAdd


Compairing Sale price with YearSinceSold
SalePrice
YearSinceSold

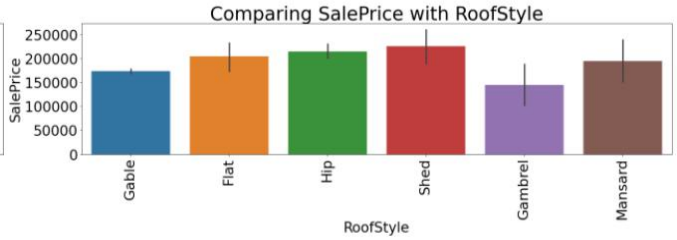

Compairing Sale price with GarageSince
SalePrice
GarageSince

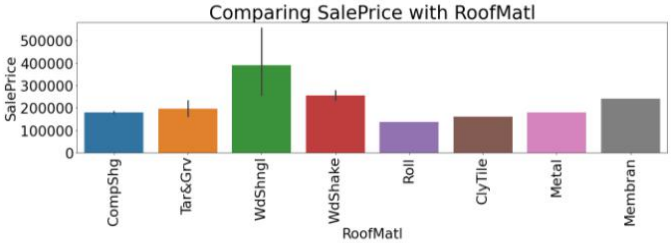
Observations:

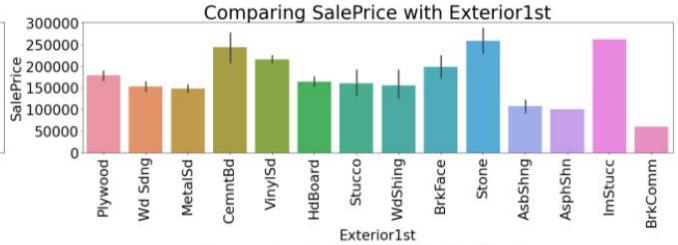
- In 1st graph the sales is good and SalePrice is also high 1-STORY 1946 & NEWER ALL STYLES(20) and 2-STORY 1946 & NEWER(60) types of dwelling(MSSuubClass).
- In 2nd graph in Linear feet of street connected to property(LotFrontage) most of sales are happening below 135 feet but best SalePrice is between 160 to 180 feet.
- In 3rd graph Lot size in square feet(LotArea) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
- In 4th graph Rates the overall material and finish of the house(OverallQual) is increasing linearly sales is also increasing And SalePrice is also increasing linearly.
- In 5th graph (Average) overall condition of the house(OverallCond) the sales is high and SalePrice is also high.
- In 6th graph Masonry veneer area in square feet(MasVnrArea) is increasing sales is decreasing and saleprice is rangeing between 0-4 lakhs.
- In 7th graph Type 1 finished square feet(BsmtFinSF1) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
- In 8th graph 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.
- In 9th graph Total square feet of basement area(TotalBsmtSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
- In 10th graph First Floor square feet(1stFlrSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
- In 11th graph Second floor square feet(2ndFlrSF) is increseing sales is increasing in the range 500-1000 and the saleprice is in between 0-4 lakhs.
- In 12th graph Above grade (ground) living area square feet(GrLivArea) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
- In 13th graph For 0 and 1 Basement full bathrooms(BsmtFullBath) the sales as well as SalePrice is high.
- In 14th graph For 1 and 2 Full bathrooms above grade(FullBath) the sales as well as SalePrice is high.

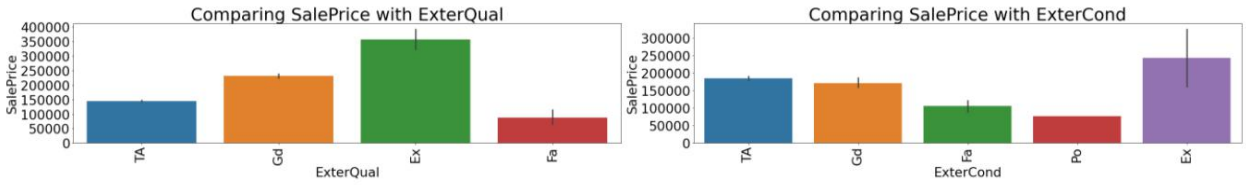
- In 15th graph For 0 and 1 Half baths above grade(HalfBath) the sales as well as SalePrice is high.
- In 16th graph For 2, 3 and 4 Bedrooms above grade (does NOT include basement bedrooms)(BedroomAbvGr) the sales as well as SalePrice is high.
- In 17th graph For 1 Kitchens above grade(KitchenAbvGr) the sales as well as SalePrice is high.
- In 18th graphFor 4-9 Total rooms above grade (does not include bathrooms)(TotRmsAbvGrd) the sales as well as SalePrice is high.
- In 19th graph For 0 and 1 Number of fireplaces(Fireplaces) the sales as well as SalePrice is high.
- In 20th graph For 1 and 2 Size of garage in car capacity(GarageCars) the sales is high and for 3 Size of garage in car capacity(GarageCars) the SalePrice is high.
- In 21st graph As Size of garage in square feet(GarageArea) is increseing sales is increseing and the saleprice is in between 0-4 lakhs.
    - In 22nd graph Wood deck area in square feet(WoodDeckSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
    - In 23rd graph Open porch area in square feet(OpenPorchSF) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.
    - In 24th graph between april to august for Month Sold(MoSold) the sales is good with SalePrice.
    - In 25th graph YearSince is increseing sales is decreasing and the saleprice is high for newly built building and the sales price is in between 0-4 lakhs.
    - In 26th graph Since Remodel date (same as construction date if no remodeling or additions)(YearSinceRemodAdd) is increseing sales is decreasing and the saleprice is in between 1-4 lakhs.
    - In 27th graph For all the YearSinceSold the salePrice and sales both are same.
    - In 28th graph Since Year garage was built(GarageAge) is increseing sales is decreasing and the saleprice is in between 0-4 lakhs.

2. Visualization of categorical columns with target column:





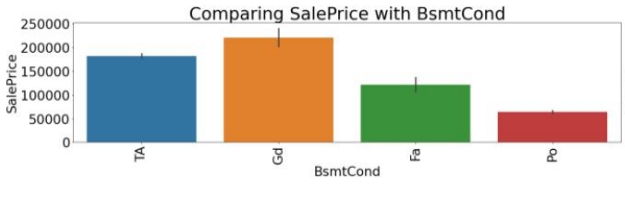



Comparing SalePrice with ExterQual
SalePrice
ExterQual
TA
Gd
Ex
Fa

Comparing SalePrice with ExterCond
SalePrice
ExterCond
TA
Gd
Fa
Po
Ex


Comparing SalePrice with Foundation
SalePrice
Foundation
CBlock
PConc
BrkTil
Slab
Stone
Wood


Comparing SalePrice with BsmtQual
SalePrice
BsmtQual
Gd
TA
Ex
Fa

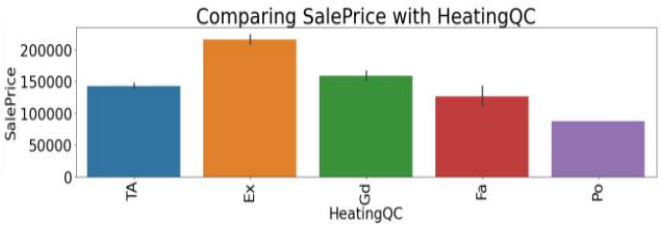

Comparing SalePrice with BsmtCond
SalePrice
BsmtCond
TA
Gd
Fa
Po


Comparing SalePrice with BsmtExposure
SalePrice
BsmtExposure
No
Gd
Av
Mn


Comparing SalePrice with BsmtFinType1
SalePrice
BsmtFinType1
ALQ
GLQ
BLQ
Unf
Rec
LwQ


Comparing SalePrice with BsmtFinType2
SalePrice
BsmtFinType2
Unf
Rec
BLQ
GLQ
ALQ
LwQ


Comparing SalePrice with Heating
SalePrice
Heating
GasA
GasW
Floor
OthW
Wall
Grav

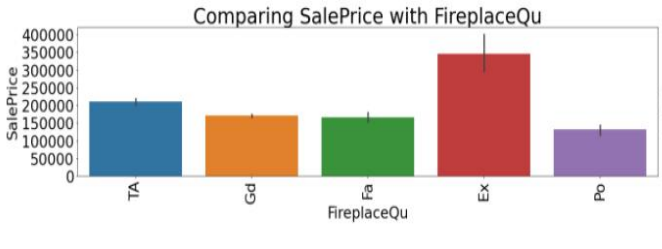

Comparing SalePrice with HeatingQC
SalePrice
HeatingQC
TA
Ex
Gd
Fa
Po

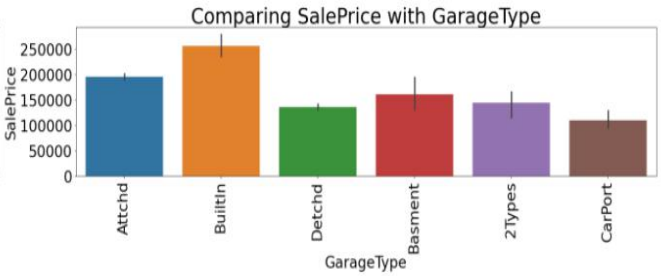

Comparing SalePrice with CentralAir
SalePrice
CentralAir
Y
N

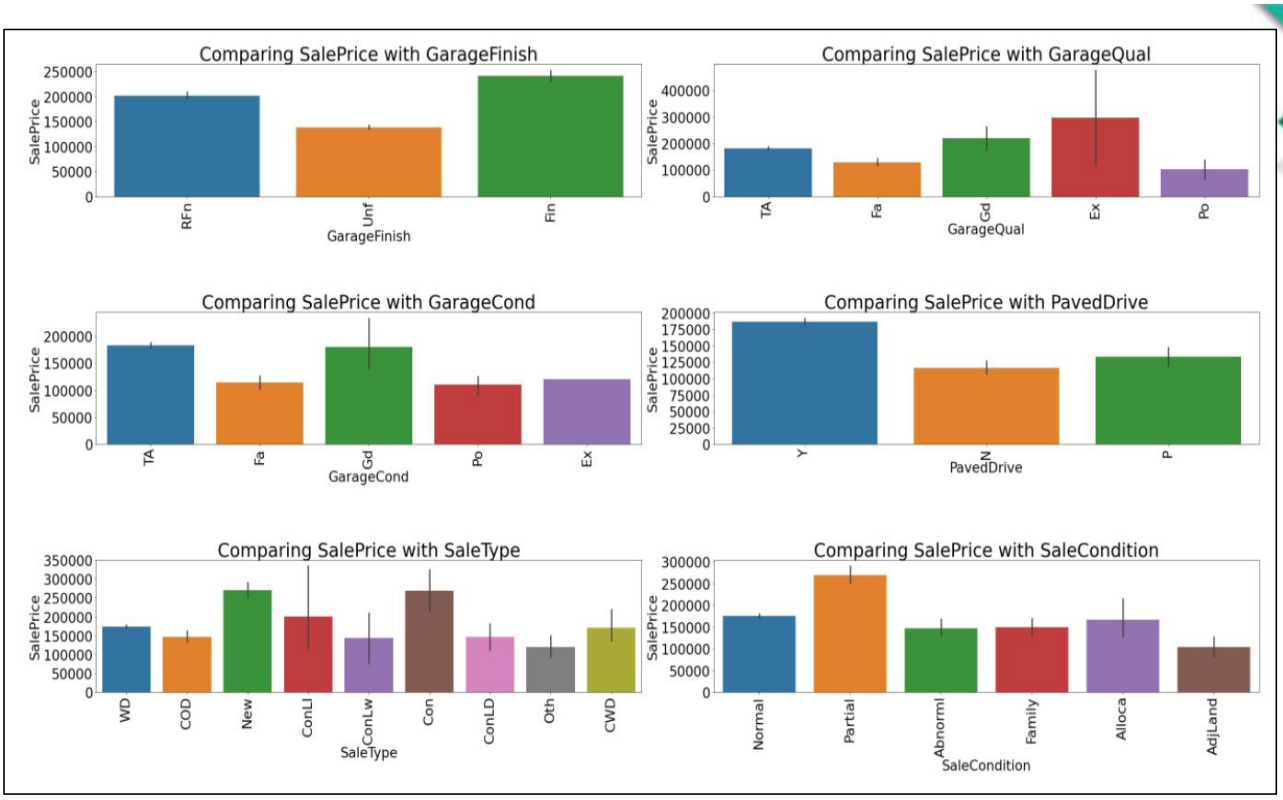

Comparing SalePrice with Electrical
SalePrice
Electrical
SBrkr
FuseA
FuseF
FuseP
Mix


Comparing SalePrice with KitchenQual
SalePrice
KitchenQual
TA
Gd
Ex
Fa


Comparing SalePrice with Functional
SalePrice
Functional
Typ
Mod
Maj1
Min1
Min2
Sev
Maj2


Comparing SalePrice with FireplaceQu
SalePrice
FireplaceQu
TA
Gd
Fa
Ex
Po


Comparing SalePrice with GarageType
SalePrice
GarageType
Attchd
BuiltIn
Detchd
Basment
2Types
CarPort



- IN graph 1 Floating Village Residential(FV) and Residential Low Density(RL) zoning classification of the sale(MSZoning) the saleprice is high.
- In graph 2 paved type of road access to property(Street) the SalePrice is high.
- In graph 3 Slightly irregular(IR1), Moderately Irregular(IR2) and Irregular(IR3) shape of property(LotShape) the SalePrice is high.
- In graph 4 Hillside - Significant slope from side to side(HLS) Flatness of the property(LandContour) the SalePrice is High.
- In graph 5 Cul-de-sac(CulDSac) Lot configuration(LotConfig) the SalePrice is High.
- In graph 6 all types of Slope of property(LandSlope) i.e.,Gentle slope(Gtl), Moderate Slope(Mod) and Severe Slope(Sev) the SalePrice is High.
- In graph 7 Northridge(NoRidge) locations within Ames city limits(Neighborhood) the SalePrice is High.
- In graph 8 Within 200' of North-South Railroad(RRNn), Adjacent to postive off-site feature(PosA) and Near positive off-site feature--park, greenbelt, etc.(PosN) Proximity to various conditions(Condition1) has the maximum SalePrice.
- In graph 9 Adjacent to postive off-site feature(PosA) and Near positive off-site feature--park, greenbelt, etc.(PosN) Proximity to various conditions (if more than one is present)(Condition2) has maximum SalePrice.
- In graph 10 Single-family Detached(1Fam) and Townhouse End Unit(TwnhsE) type of dwelling(BldgType) the SalePrice is high.
- In graph 11 2Story and Two and one-half story: 2nd level finished(2.5Fin) Style of dwelling(HouseStyle) the SalePrice is high.
- In graph 12 Shed Type of roof(RoofStyle) the SalePrice is high.
- In graph 13 Wood Shingles(WdShngl) Roof material(RoofMat1) the SalePrice is high.
- In graph 14 Cement Board(CemntBd), Imitation Stucco(ImStucc) and Stone type of Exterior covering on house(Exterior1st) the SalePrice is high.
- In graph 15 Cement Board(CemntBd), Imitation Stucco(ImStucc) and other Exterior covering on house (if more than one material)(Exterior2) has maximum SalePrice.
- In graph 16 Stone Masonry veneer type(MasvnrType) the SalePrice is high.
- In graph 17 Excellent(Ex) quality of the material on the exterior(ExterQual) the SalePrice is high.
- In graph 18 Excellent(Ex) present condition of the material on the exterior(ExterCond) the SalePrice is high.
- In graph 19 Poured Contrete(PConc) Type of foundation(Foundation) the SalePrice is high.
- In graph 20 Excellent(100+ inches)(Ex) height of the basement(BsmtQual) the SalePrice is high.
- In graph 21 Good(Gd) general condition of the basement(BsmtCond) the SalePrice is high.
- In graph 22 Good Exposure(Gd) of walkout or garden level walls(BsmtExposure) has maximum SalePrice.
- In graph 23 Good Living Quarters(GLQ) of basement finished area(BsmtFinType1) has maximum SalePrice.
- In graph 24 Good Living Quarters(GLQ) and Average Living Quarters(ALQ) of basement finished area (if multiple types)(BsmtFinType2) has maximum SalePrice.
- In graph 25 Gas forced warm air furnace(GasA) and Gas hot water or steam heat(GasW) Type of heating(Heating) has high SalePrice.
- In graph 26 Excellent(Ex) Heating quality and condition(HeatingQC) the SalePriceis high.
- In graph 27 building having Central air conditioning(CentralAir) the SalePrice is high.
- In graph 28 Standard Circuit Breakers & Romex(Sbrkr) of Electrical system(Electrical) the SalePrice is Maximum.
- In graph 29 Excellent(Ex) Kitchen quality(KitchenQual) the SalePrice is high.
- In graph 30 Typical Functionality(Typ) type of Home functionality (Assume typical unless deductions are warranted)(Functional) the SalePrice is high.
- In graph 31 Excellent - Exceptional Masonry Fireplace(Ex) of Fireplace quality(FireplaceQual) has highest SalePrice.
- In graph 32 Built-In (Garage part of house - typically has room above garage)(BuiltIn) Garage location(GarageType) the SalePrice is maximum.
- In graph 33 Completely finished(Fin) Interior of the garage(GarageFinish) the SalePrice is high.
- In graph 34 Excellent(Ex) Garage quality(GarageQual) the SalePrice is high.
- In graph 35 Typical/Average(TA) and Good(Gd) Garage condition(GarageCond) the SalePrice is high.
- In graph 36 having Paved driveway(PavedDrive) the SalePriceis high.
- In graph 37 Home just constructed and sold(New) and Contract 15% Down payment regular terms(Con) of type of sale(SaleType) has highest SalePrice.
- In graph 38 Home was not completed when last assessed (associated with New Homes)(Partial) Condition of sale(SalesCondition) the SalePrice is maximum.

3.5 Run and Evaluate Selected Models:

Model Selection:

1) XGB Regressor

```
xgb=XGBRegressor()
xgb.fit(x_train,y_train)
pred=xgb.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(xgb, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.8755429213543253
mean_squared_error:  499134409.02854735
mean_absolute_error:  15317.156311419025
Cross Validation Score:  0.8617055030502548
```

2) Decision Tree Regressor:

```
dtr=DecisionTreeRegressor()
dtr.fit(x_train,y_train)
pred=dtr.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(dtr, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.6717055713614568
mean_squared_error:  1316622946.7138364
mean_absolute_error:  25881.418238993712
Cross Validation Score:  0.7033554876714299
```

3) Gradient Boosting Regressor:

```
gbr=GradientBoostingRegressor()
gbr.fit(x_train,y_train)
pred=gbr.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(gbr, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.9139543787171535
mean_squared_error:  345085476.8235429
mean_absolute_error:  13966.520738526257
Cross Validation Score:  0.87856772216345
```

4) Extra Trees Regressor:

```
etr=ExtraTreesRegressor()
etr.fit(x_train,y_train)
pred=etr.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(etr, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.903279902652068
mean_squared_error:  387895402.6261918
mean_absolute_error:  14309.488113207546
Cross Validation Score:  0.8569968921908483
```

5) KNeighbors Regressor:

```
knn=KNN()
knn.fit(x_train,y_train)
pred=knn.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(knn, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.8580686079979793
mean_squared_error:  569215043.7759749
mean_absolute_error:  17887.970440251574
Cross Validation Score:  0.793634573970961
```

6) Random Forest Regressor:

```
rfr=RandomForestRegressor()
rfr.fit(x_train,y_train)
pred=rfr.predict(x_test)
R2_Score=r2_score(y_test,pred)
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))

# cross validation score
score= cross_val_score(rfr, x, y, cv=10).mean()
print("Cross Validation Score: ", score)
```

```
r2_score:  0.9001517363672393
mean_squared_error:  400440895.7946937
mean_absolute_error:  14610.544150943393
Cross Validation Score:  0.8595471513462956
```

- Here after looking at the r2 score and cross vaildation score of all the models, least difference is observed in XGBRegressor, so, we will take XGBRegressor as our best model and will proceed for Hyper Parameter Tuning.

Hyper Parameter Tuning for Best Model:

```
# Lets import all required libraries for hyperparameter tuning
from sklearn.model_selection import GridSearchCV
```

```
parameter = {'booster':['gbtree','gblinear'],
             'n_estimators' : [100, 500, 900],
            'learning_rate':[0.05,0.1,0.15,0.20]}
```

```
gcv=GridSearchCV(XGBRegressor(),parameter,cv=5)
```

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

```
GridSearchCV(cv=5,
             estimator=XGBRegressor(base_score=None, booster=None,
                                    colsample_bylevel=None,
                                    colsample_bynode=None,
                                    colsample_bytree=None,
                                    enable_categorical=False, gamma=None,
                                    gpu_id=None, importance_type=None,
                                    interaction_constraints=None,
                                    learning_rate=None, max_delta_step=None,
                                    max_depth=None, min_child_weight=None,
                                    missing=nan, monotone_constraints=None,
                                    n_estimators=100, n_jobs=None,
                                    num_parallel_tree=None, predictor=None,
                                    random_state=None, reg_alpha=None,
                                    reg_lambda=None, scale_pos_weight=None,
                                    subsample=None, tree_method=None,
                                    validate_parameters=None, verbosity=None),
             param_grid={'booster': ['gbtree', 'gblinear'],
                         'learning_rate': [0.05, 0.1, 0.15, 0.2],
                         'n_estimators': [100, 500, 900]})
```

```
gcv.best_params_
```

```
{'booster': 'gblinear', 'learning_rate': 0.15, 'n_estimators': 100}
```

```
b_model=XGBRegressor(booster="gblinear", learning_rate=0.15, n_estimators=100)
b_model.fit(x_train,y_train)
pred=b_model.predict(x_test)
print('R2_Score: ',r2_score(y_test,pred))
print('Mean_Squared_Error: ',metrics.mean_squared_error(y_test,pred))
print('Mean_Absolute_Error: ',metrics.mean_absolute_error(y_test,pred))
```

```
R2_Score:  0.8889651538489723
Mean_Squared_Error:  445304621.6274415
Mean_Absolute_Error:  16648.909939747937
```

- Here we have got score of 88.9% and we managed to improve score by 1.4% after hyper parameter tuning.

Saving the Best model:

```
# lets save the model using .pkl
import joblib
joblib.dump(b_model,"PFA_House_Project.pkl")
```

```
['PFA_House_Project.pkl']
```

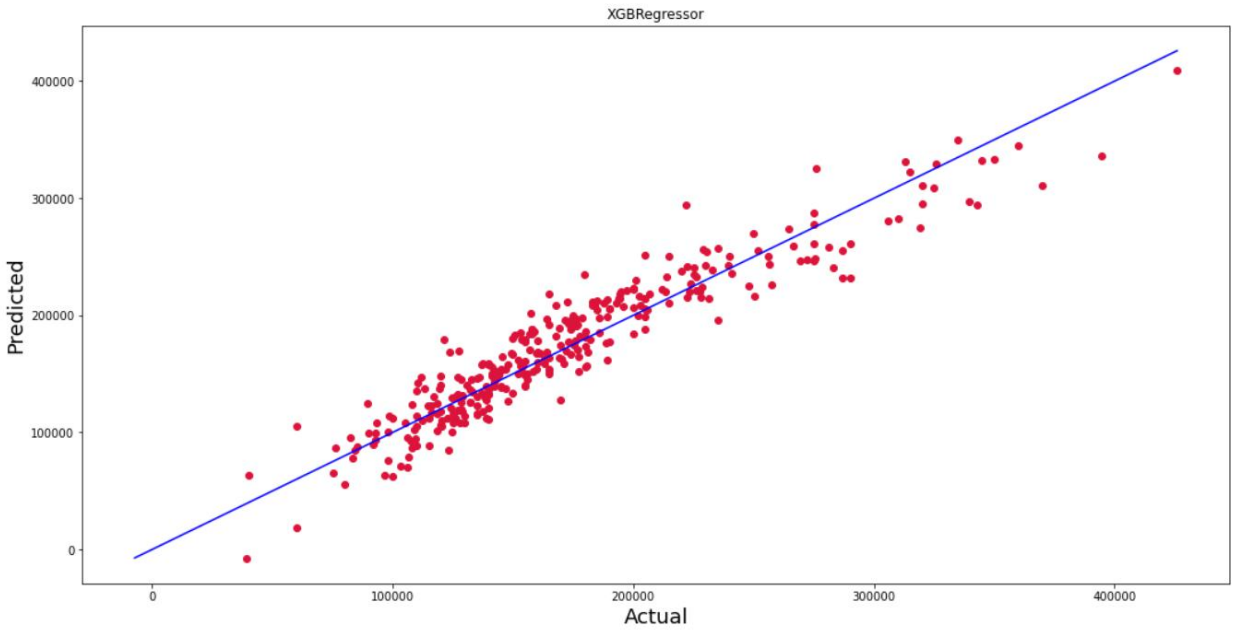
Loading saved model and predicting:

```
# loading saved model
model=joblib.load("PFA_House_Project.pkl")

# predicting
prediction= model.predict(x_test)
prediction
```

```
array([179612.55 ,  84585.375,  77697.63 , 110334.19 , 105388.59 ,
       179892.14 , 222118.78 , 198150.95 , 231810.9  , 139206.42 ,
       107825.164, 179415.58 , 207893.94 , 195951.84 , 145401.44 ,
       216307.11 , 152189.03 ,  55357.21 , 220078.92 , 273985.53 ,
       136911.39 , 217886.5  , 145335.17 ,  99048.15 , 192537.08 ,
       344905.44 , 154432.2  , 139299.02 , 162114.92 , 212298.31 ,
       111394.734, 122944.06 , 329296.06 , 130043.61 , 196647.64 ,
       255153.03 , 230191.08 , 112289.9  , 191233.19 , 196221.17 ,
       152478.69 , 332057.3  , 130941.18 ,  84754.945, 248022.3  ,
        94399.89 , 214307.9  , 105840.805, 168322.67 , 250195.28 ,
       310507.03 , 197715.33 , 277280.94 , 145687.81 , 205890.23 ,
       159390.23 , 179592.62 , 122721.37 , 282754.12 , 174683.05 ,
       215786.42 , 107852.17 , 182191.6  , 115565.   , 153922.2  ,
       139295.75 , 308650.7  , 280358.03 , 163499.94 , 123944.25 ,
       114671.69 , 156548.52 ,  65287.824, 240592.86 ,  70989.3  ,
       207099.36 , 132851.47 ,  86464.6  , 137363.44 , 182537.5  ,
       261588.33 , 169319.62 , 164993.83 , 156373.55 , 241171.19 ,
       242222.77 , 108233.04 , 147266.47 , 336119.6  , 322344.25 ,
       138527.23 , 112171.24 , 221372.61 , 234668.75 , 219819.11 ,
       153805.92 , 219988.69 ,  70719.055, 232781.95 , 157509.22 ,
       131062.81 , 221319.94 , 261054.27 , 214492.4  , 232856.89 ,
       108563.93 , 164884.48 , 214833.23 , 157856.83 , 133427.47 ,
       255114.23 , 325264.56 , 211753.61 , 167197.36 ,  87804.89 ,
        19107.186, 132701.67 , 194977.89 , 105290.27 , 146729.67 ,
        89651.375, 173882.78 , 136049.97 , 242396.38 , 183406.9  ,
        95772.27 , 224239.38 , 112904.74 , 214528.6  , 162919.9  ,
       114127.336, 100459.54 , 250573.39 , 159466.81 , 147242.23 ,
       117516.46 ,  63479.508, 127047.08 , 161486.2  ,  88633.17 ,
       183757.12 , 166421.45 , 112108.8  , 111941.664, 166361.03 ,
       168284.92 , 162931.31 , 208868.44 , 152052.55 , 140709.72 ,
       156075.33 , 150627.83 , 208725.16 ,  92928.26 , 116216.16 ,
       216656.81 , 153445.02 ,  87226.09 , 149609.08 , 184771.69 ,
       152775.69 , 184000.53 , 208494.31 , 256097.38 , 139062.67 ,
       110277.21 , 108327.92 , 197154.98 , 157740.4  , 259668.83 ,
       241860.4  , 178390.5  , 169513.19 , 181142.39 , 149752.64 ,
       287040.9  , 119678.56 , 168194.33 , 185751.56 , 222051.45 ,
       152780.34 , 205546.17 , 125809.16 , 246629.28 , 137653.88 ,
       146857.3  ,  88622.59 , 215322.6  , 274455.56 , 184267.5  ,
       167216.81 , 178651.48 , 123114.11 , 137442.67 , 140773.23 ,
       100430.39 , 297254.78 , 237436.2  , 170350.2  , 163677.78 ,
       224812.3  , 205063.8  , 294266.25 , 200093.45 , 157254.55 ,
        63521.613, 409510.97 , 177752.47 , 189550.1  ,  79354.164,
       174632.34 , 295491.25 ,  95675.94 , 177450.2  , 113791.24 ,
       218727.67 , 293971.1  , 124405.47 , 196056.12 , 192089.7  ,
       131402.03 , 331158.78 , 159178.05 , 177281.56 , 171085.61 ,
       188374.03 , 258042.72 , 125505.734, 210192.02 , 350141.84 ,
       184755.88 , 254089.5  , 112324.48 , 190880.33 , 134673.27 ,
       116895.98 , 149080.89 , 132009.48 , 125059.586, 140092.89 ,
       205775.5  , 117740.59 , 188137.19 , 147753.11 , 142592.23 ,
       195499.02 , 130487.77 , 250599.42 , 227166.28 ,  99380.9  ,
       220416.3  , 211691.6  , 234420.02 , 146466.16 , 132526.92 ,
       333446.06 , 256971.64 , 116365.56 , 208889.34 , 153619.8  ,
        62526.137, 176651.19 , 135768.25 , 243193.53 , 246642.48 ,
       310730.5  , 218203.94 , 130621.68 , 158709.16 , 186451.16 ,
       201796.36 ,  76099.875, 127479.41 , 214680.05 , 205014.17 ,
       223116.95 , 114004.09 , 248799.6  , 149208.08 , 158809.7  ,
       177970.92 , 102287.664, 120519.21 , 210912.   , 198675.4  ,
       238758.8  , 147860.14 , 130083.25 , 187693.83 , 270393.56 ,
       187686.08 , 210836.8  ,  -7227.527, 199291.47 , 150034.12 ,
       235788.67 ,  94070.63 , 105817.47 , 157389.81 , 168613.86 ,
       150267.92 , 160613.73 , 192214.97 , 133818.97 , 101136.06 ,
       128035.48 , 118838.695, 213928.81 , 121381.266, 117774.64 ,
       125558.586, 251127.61 , 200065.   , 179477.4  , 232411.73 ,
       143423.53 , 167334.64 , 226219.23 ], dtype=float32)
```

```
plt.figure(figsize=(18,9))
plt.scatter(y_test,prediction, c='crimson')
price1=max(max(prediction),max(y_test))
price2=min(min(prediction),min(y_test))
plt.plot([price1,price2],[price1,price2],"b-")
plt.xlabel("Actual",fontsize=18)
plt.ylabel("Predicted",fontsize=18)
plt.title("XGBRegressor")
plt.show()
```




```
# lets make dataframe for all predicted SalePrice for test dataset
test_dataset_price=pd.DataFrame()
test_dataset_price['SalePrice']=prediction
test_dataset_price
```

	SalePrice
0	323759.656250
1	224881.140625
2	250301.703125
3	189063.484375
4	228391.390625
5	75278.406250
6	131973.375000
7	301567.187500
8	239647.593750
9	68502.789062
10	138015.062500
11	119972.156250
12	141759.125000
13	110021.742188
14	117586.304688

```
# lets save the predicted sale price
test_dataset_price.to_csv('data_new.csv',index=False)
```

- Successfully saved the predicted SalePrice of test dataset into CSV format.

3.6 Interpretation of the Results:

- Here, 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 plotting 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 chosen proper analytical methods to deal with the outliers and skewness. If we ignore this outliers and skewness we might have spoiled our model which would have predicted wrong/ less accuracy.
- Then we performed scaling on both train and test dataset so that our model do not to get baised.
- We used multiple models while building model using train dataset to get the best model out of it.
- Then we used multiple metrics like mae, mse and r2_score which will help us to decide the best model.
- I found XGBRegressor as the best model with 87.55% r2_score. Also we managed to improve the accuracy of the best model by running hyper parameter tunning to 88.9%.
- At last I have predicted the SalePrice for test dataset using saved model of train dataset. It was good!! that I was able to get the predictions near to actual values.

4 Conclusion:

4.1 Key findings and Conclusions:

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.

4.2 Learning Outcomes of the Study:

We 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 28 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. To conclude, the application of machine learning in property research is still at an early stage. We hope this study has moved a small step ahead in providing some methodological and empirical contributions to property appraisal, and presenting an alternative approach to the valuation of housing prices. Future direction of research may consider incorporating additional property transaction data from a larger geographical location with more features, or analysing other property types beyond housing development.

4.3 Limitations of this work:

1. First draw back is the data leakage when we merge both train and test datasets.
2. Followed by more number of outliers and skewness these two will reduce our model accuracy.
3. Then we have tried our best to deal with outliers, skewness, null values and zero values. So it looks quite good that we have achieved a accuracy of 88.9% even after dealing all these drawbacks.
4. This study did not cover all regression algorithms however, it was focused on the chosen algorithm, starting from the basic regression techniques to the advanced ones which was also good for learning.

4.4 Scope for Future Work:

Here we observed that even after getting very good score of close to 89%. still there was difference in predicted SalePrice and Actual Price. May be we required some more data to be able to make better predictions. For future we hope to get more accurate data to be able to make far better and accurate predictions.

5. References:

1) Referred by FlipRobo Technologies

2) https://www.kaggle.com/anmolkumar/house-price-predicti

on challenge/tasks?taskId=2304


THANK
YOU
for your
TIME AND
ATTENTION