FLIP ROBO

HOUSING - PRICE PREDICTION

Submitted by:

Akshay Shingavi

ACKNOWLEDGMENT

I want to sincerely thank my Data Trained Academy and Flip Robo Technologies Bangalore supervisors for allowing me to work on this project. Their recommendations and guidance have enabled me to effectively complete this job.

References

1. Flip Robo Technologies provided the entire necessary dataset and information.
2. https://www.sciencedirect.com/science/article/pii/S1877050920316318
3. https://www.kaggle.com

INTRODUCTION

A. Business Problem Framing

Every individual on earth needs a home, thus the housing and real estate sectors are among those that have a significant impact on global economic growth. There are several firms operating in the industry, and the market is fairly sizable. In order to assist businesses, enhance their total income and profitability, improve their marketing methods, and pay attention to shifting patterns in home sales and purchases, data science is an essential tool. For housing organizations to achieve their commercial objectives, machine learning approaches including predictive modelling, market mix modelling, and recommendation systems are applied. One of these housing companies is the source of our issue.

B. Conceptual Background of the Domain Problem

The real estate market is one of the most competitive in terms of pricing, and the same tends to vary significantly based on a variety of factors. Predicting property price is a crucial component in decision-making for both buyers and investors in supporting budget allocation, finding property finding stratagems, and determining appropriate policies

C. Review of Literature

The variables influencing land prices must be researched and their effects on price must be modelled. It is necessary to analyse historical data. It may be concluded that creating a straightforward linear mathematical connection for this time-series data is not found to be practical for predicting. Establishing a non-linear model that can accurately match the data characteristics to analyses and estimate future trends became necessary as a result. The study and forecasting of land prices using mathematical modelling and other scientific methodologies is an immediate urgent necessity for decision-making by all those concerned since the real estate sector is rapidly increasing.

D. Motivation for the Problem Undertaken

I have to use the provided independent variables to model the cost of homes. The management will then utilise this model to determine exactly how the prices fluctuate depending on the factors. As a result, they may influence the company's strategy and concentrate on areas that will provide large profits.
Additionally, using the model will help management better understand how prices change in a new market. A key driver for forecasting home prices is the correlation between housing costs and the economy.

Analytical Problem Framing

A. Mathematical/ Analytical Modelling of the Problem

There are two datasets for this specific problem: a train dataset and a test dataset. Using the train dataset, I created a model that forecast sale prices for six test datasets. By examining the target column, I discovered that the data in the SalePrice column were continuous. Since this was a regression problem, I had to design the model using all regression procedures. Additionally, I saw some unused data in some of the columns, such as more than 80% null values and more than 85% zero values, therefore I eliminated those columns.

B. Data Sources and their formats

The data was gathered in csv (comma separated values) format for my internship firm, Flip Robo Technologies.

Additionally, I had two datasets: a train dataset and a test dataset. Using the train dataset, I created a model and predicted the sale price for the test dataset. My test dataset had 292 rows and 80 columns without the goal, while my train dataset had 1168 rows and 81 columns with the target. I have object, float, and integer data types in this specific dataset. I could combine these two datasets and conduct my research, but I have not done so due to a problem with data leaking. When I load my datasets into my Python, they seem to me like this.

C. Data Pre-processing Done

- I started by importing the necessary libraries and the two datasets, both of which were in CSV format.
- After that, I performed all the statistical analysis, including evaluating the shape, nuance, value counts, and information.
- While reviewing the information in the datasets, I discovered some columns that had more than 80% null values; as these columns will cause skewness in the datasets, I made the decision to remove them.
- Then, while examining the value counts, I discovered several columns with more than 85% zero values. Since this causes skewness in the model and increases the possibility of model bias, I have eliminated such columns.

D. Data Inputs- Logic- Output Relationships

I used EDA to analyse the relationship between characteristics and the objective. I used a variety of plots, including bar plots, reg plots, scatter plots, line plots, swarm plots, strip plots, violin plots, and many more. Moreover, it was discovered that certain of the columns,

including OverallQual, TotalRmsAbvGrd, Full Bath, Garage Cars, etc., had a significant positive linear relationship with the label.
I used a heat map and a bar plot to examine the relationship between the target and the features. Where I found the labels' and characteristics' positive and negative association.

E. Hardware and Software Requirements and Tools Used

Hardware required: -
1. Processor — core i5 or i7 and above
2. RAM — 8 GB or above
3. SSD — 250GB or above

Software/s required: -
1.Anaconda
2.Python

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

```
# Importing Liabraries

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

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

For Model Building using Regressor we will require below libraries

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.ensemble import BaggingRegressor
from sklearn.linear_model import LinearRegression
from sklearn.neighbors import KNeighborsRegressor as KNN
from sklearn.linear_model import SGDRegressor
from sklearn.metrics import classification_report
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.model_selection import cross_val_score
from sklearn.ensemble import BaggingRegressor
from sklearn import metrics
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
```

Model/s Development and Evaluation

A. Identification of possible problem-solving approaches (methods)

I have substituted null values using the imputation procedure. I have employed the percentile approach to eliminate outliers. And I applied the Yeo-Johnson approach to eliminate skewness. I must use ordinal encoding to encode the category columns. To determine the relationship between dependent and independent characteristics, use the Pearson's correlation coefficient. Additionally, I've utilised standardisation. Model construction using all regression techniques is the next step.

B. Testing of Identified Approaches (Algorithms)

This specific scenario was a regression challenge since SalePrice was my objective and a continuous column. Additionally, I built my model using all regression procedures. I discovered Random Forest Classifier to be the best model with the least amount of variation by examining the difference between the r2 score and cross validation score. Additionally, we must test several models in order to find the optimal one, and cross validation must be used to minimise overfitting's confusion. The list of regression methods I utilised in my project is shown below.

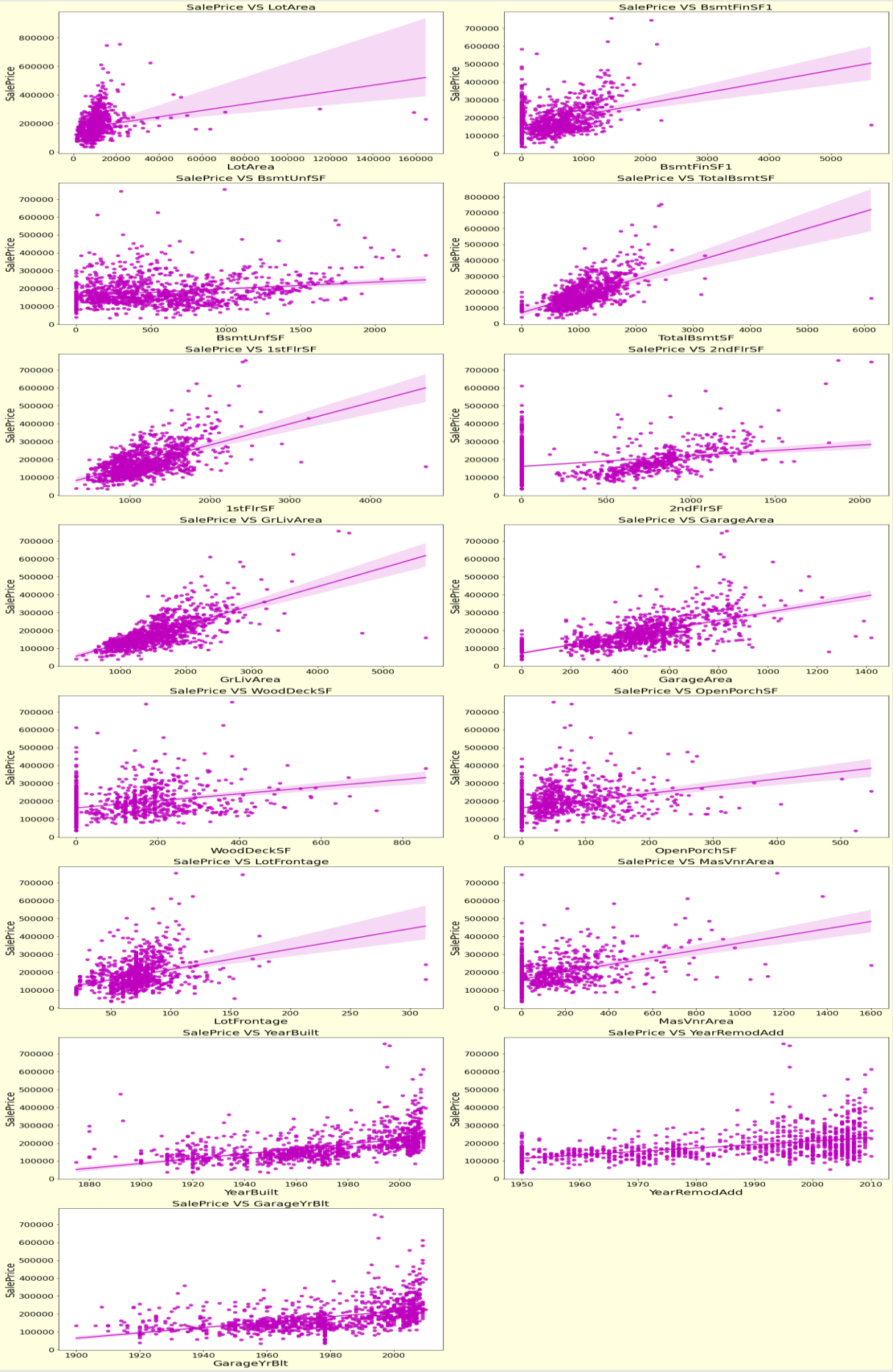
1. RandomForestRegressor
2. Bagging Regressor
3. GradientBoostingRegressor
4. DecisionTreeRegressor

C. Key Metrics for success in solving problem under consideration

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 continuous features.

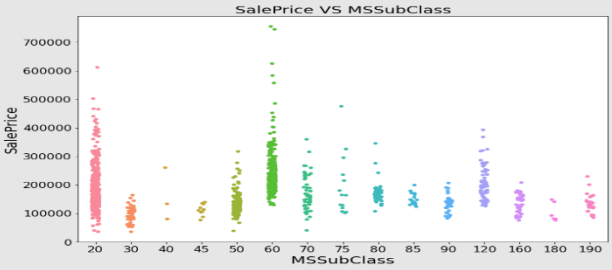
D. Visualizations

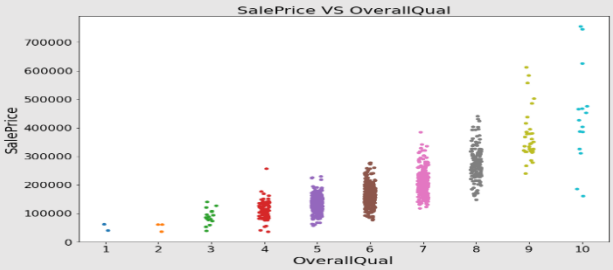
Univariate Analysis Graphical
Multivariate Analysis Graphical

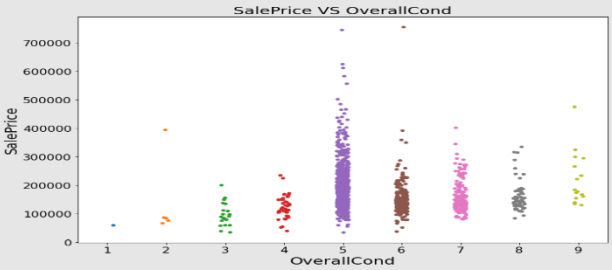

SalePrice VS LotArea
SalePrice
LotArea
SalePrice VS BsmtFinSF1
SalePrice
BsmtFinSF1
SalePrice VS BsmtUnfSF
SalePrice
BsmtUnfSF
SalePrice VS TotalBsmtSF
SalePrice
TotalBsmtSF
SalePrice VS 1stFlrSF
SalePrice
1stFlrSF
SalePrice VS 2ndFlrSF
SalePrice
2ndFlrSF
SalePrice VS GrLivArea
SalePrice
GrLivArea
SalePrice VS GarageArea
SalePrice
GarageArea
SalePrice VS WoodDeckSF
SalePrice
WoodDeckSF
SalePrice VS OpenPorchSF
SalePrice
OpenPorchSF
SalePrice VS LotFrontage
SalePrice
LotFrontage
SalePrice VS MasVnrArea
SalePrice
MasVnrArea
SalePrice VS YearBuilt
SalePrice
YearBuilt
SalePrice VS YearRemodAdd
SalePrice
YearRemodAdd
SalePrice VS GarageYrBlt
SalePrice
GarageYrBlt

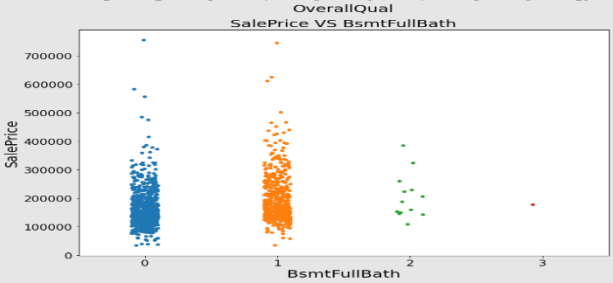
Observation -

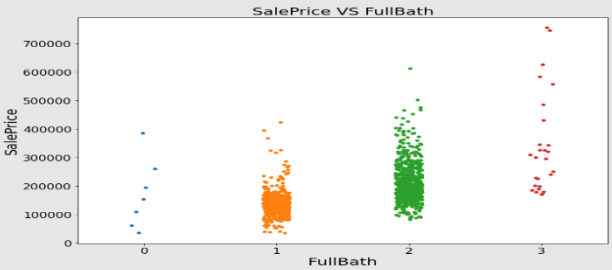
- As the number of linear feet of street frontage on a property increases, sales are declining and the sale price is fluctuating between 0 and 3 lakhs.
- Sales are declining and the sale price is between 0 and 4 lakhs as lot size (lot area) increases.
- Sales are declining and the sale price is between 0 and 4 lakhs as the Masonry Veneer Area (MasVnrArea) grows.
- Sales are declining as Type 1 completed square feet (BsmtFinSF1) rise, and the sale price ranges from 0 to 4 lakhs.
- As unfinished basement square feet (BsmtUnfSF) increase, sales decline and the sale price ranges from 0 to 4 lakhs. There are also some anomalies.
- Sales are declining as total basement square feet (TotalBsmtSF) increases, and the sale price ranges from 0 to 4 lakhs.
- Sales are declining as First Floor square feet (FirstFlrSF) increases, and the sale price ranges from 0 to 4 lakhs.
- As second floor square feet (second floor SF) increases, sales rise by 500–1000 and the sale price ranges from 0–4 lakhs.
- As above-ground living space (GrLivArea) increases, sales decline and the sale price ranges from 0 to 4 lakhs.
- Sales are increasing and the sale price ranges from 0 to 4 lakhs as garage area (measured in square feet) increases.
- As the square footage of wood decks (WoodDeckSF) increases, sales decline and the sale price ranges from 0 to 4 lakhs.
- As open porch square footage (OpenPorchSF) increases, sales decline and the sale price ranges from 0 to 4 lakhs.
- Sales are declining as Year Since Built increases, and the sale price is high for recently constructed buildings, ranging from 0 to 4 lakhs.
- Sales are declining and the sale price is between 1-4 lakhs as of the Year SinceRemodAdded (same as construction date if no remodelling or additions) date, which is increasing.
- Sales are declining and the sale price is between 0 and 4 lakhs as garage age (the year it was built) increases.

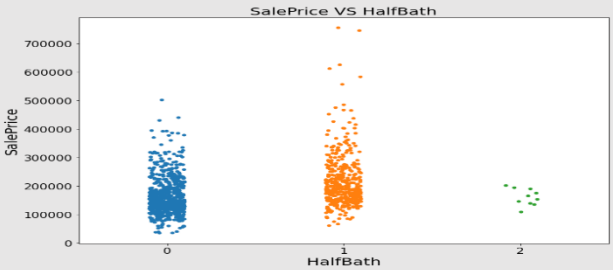

SalePrice VS MSSubClass
SalePrice
MSSubClass

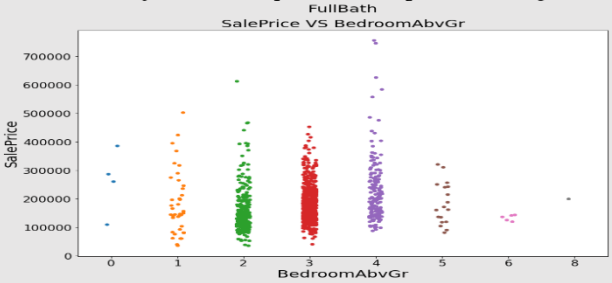

SalePrice VS OverallQual
SalePrice
OverallQual

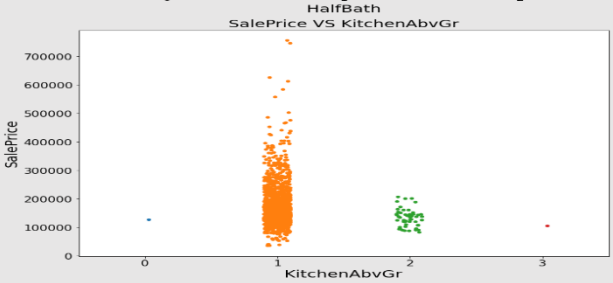

SalePrice VS OverallCond
SalePrice
OverallCond

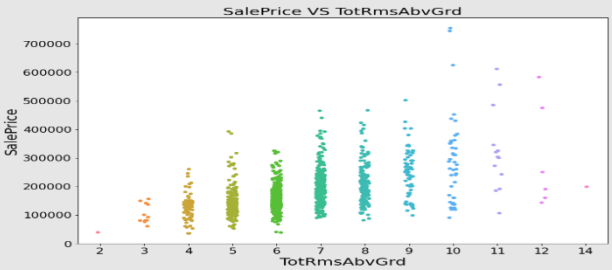

SalePrice VS BsmtFullBath
SalePrice
BsmtFullBath

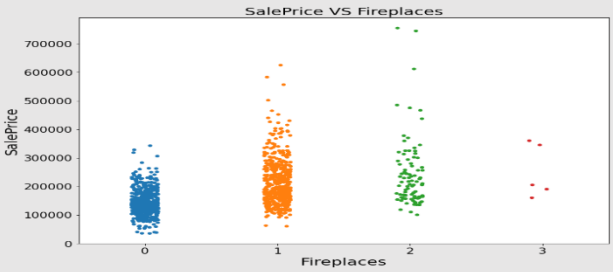

SalePrice VS FullBath
SalePrice
FullBath

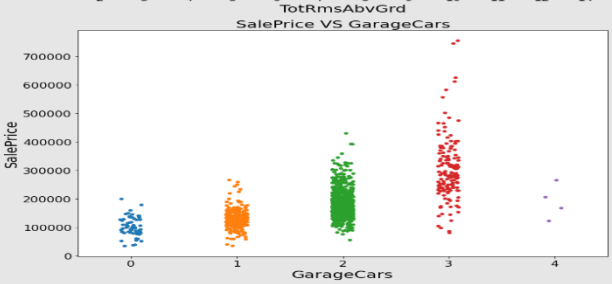

SalePrice VS HalfBath
SalePrice
HalfBath

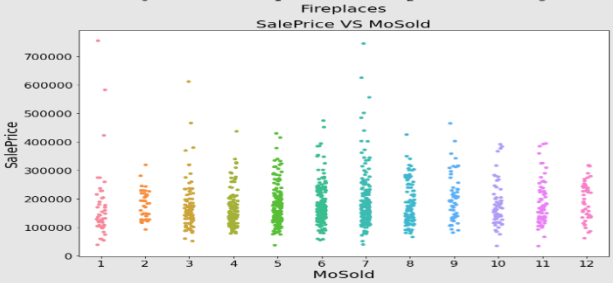

SalePrice VS BedroomAbvGr
SalePrice
BedroomAbvGr

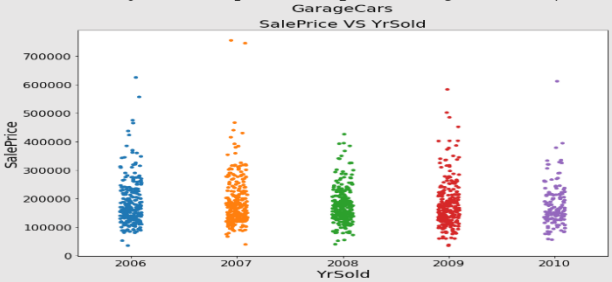

SalePrice VS KitchenAbvGr
SalePrice
KitchenAbvGr


SalePrice VS TotRmsAbvGrd
SalePrice
TotRmsAbvGrd


SalePrice VS Fireplaces
SalePrice
Fireplaces


SalePrice VS GarageCars
SalePrice
GarageCars


SalePrice VS MoSold
SalePrice
MoSold


SalePrice VS YrSold
SalePrice
YrSold

Observations

- The sales are strong and the sale price is high for dwellings that are 1-STORY 1946 & NEWER ALL STYLES (20) and 2-STORY 1946 & NEWER (60) in the MSSuubClass.
- As rates rise, sales and the sale price both rise linearly along with the overall quality of the home's construction (OverallQual).
- The sales are strong and the sale price is similarly high for homes in 5(Average) overall condition.
- Both sales and selling prices are high for basement full bathrooms with a count of 0 and 1, or BsmtFullBath.
- The sale price and sales volume for 0 Basement half bathrooms (BsmtHalfBath) are both high.
- Both sales and selling prices are high for full bathrooms above grade (1 and 2).
- Sales as well as SalePrice are high for homes with 0 and 1 Half bathrooms above grade (HalfBath).
- The sales in addition to SalePrice are high for 2, 3, and 4 bedroom above grade (not Included in basement bedrooms) (BedroomAbvGr).
- For one Kitchen Above Grade, both sales and sale prices are high.
- Both the sales volume and the sale price are high for properties with 4 to 9 total rooms over grade (excluding bathrooms).
- Sales in addition to SalePrice are high for 0 and one fireplaces (Fireplaces).
- Sales are high for garages with a car capacity of 1 and 2, and the sale price is high for garages with a car capacity of 3.
- For Month Sold (MoSold), the sales are good with SalePrice from April to August.
- The salePrice & sales are same for all Years Sold.


SalePrice VS MSZoning
SalePrice VS Street
SalePrice VS LotShape
SalePrice VS LandContour
SalePrice VS LotConfig
SalePrice VS LandSlope
SalePrice VS Condition1
SalePrice VS Condition2
SalePrice VS BldgType
SalePrice VS HouseStyle
SalePrice VS RoofStyle
SalePrice VS RoofMatl
SalePrice VS Exterior1st
SalePrice VS Exterior2nd
SalePrice VS MasVnrType
SalePrice VS ExterQual
SalePrice VS ExterCond
SalePrice VS Foundation
SalePrice VS BsmtQual
SalePrice VS BsmtCond
SalePrice VS BsmtExposure
SalePrice VS BsmtFinType1
SalePrice VS BsmtFinType2
SalePrice VS Heating
SalePrice VS HeatingQC
SalePrice VS CentralAir
SalePrice VS Electrical
SalePrice VS KitchenQual
SalePrice VS Functional
SalePrice VS GarageType
SalePrice VS GarageFinish
SalePrice VS GarageQual
SalePrice VS GarageCond
SalePrice VS PavedDrive
SalePrice VS SaleType
SalePrice VS SaleCondition
SalePrice VS Neighborhood

Observation

The sale price is high for the zoning classifications of Floating Village Residential (FV) and Residential Low Density (RL) properties.

The sale price is high for property access through a paved road (Street).

The SalePrice is high for properties that have a lot shape that is somewhat irregular (IR1), moderately irregular (IR2), or irregular (IR3).

The sale price is high for properties with a hillside that has a significant side-to-side slope (HLS) and flat terrain (LandContour).

The sale price is high for lots that are in cul-de-sacs (CulDSac).

All sorts of property's slope, including gentle slope (Gtl), moderate slope (Mod), and severe slope (Sev), have high sale prices.

The SalePrice is High for Northridge (NoRidge) areas inside the boundaries of the city of Ames.

The maximum SalePrice is for properties located within 200 feet of the North-South Railroad (RRNn), next to positive off-site features (PosA), and close to positive off-site features (PosN) such as parks, greenbelts, etc.

For Near positive off-site feature, such as a park or greenbelt, and for adjacent positive off-site feature (PosA) (PosN) If more than one condition is present, proximity to Condition 2 has a maximum sale price.

The sale price is high for single-family detached homes (1Fam) and townhouse end units (TwnhsE) as building types.

For two-story and two-and-a-half-story homes, the sale price is high since the second level is finished (2.5Fin).

The SalePrice is high for roof styles with shed-style roofs.

The SalePrice is high for Wood Shingles (WdShngl) Roof Material (RoofMat1).

The sale price is high for exterior home coverings made of cement board (CemntBd), imitation stucco (ImStucc), and stone (Exterior1st).

The maximum SalePrice is for Cement Board (CemntBd), Imitation Stucco (ImStucc), and other Exterior covering of the home (if more than one material).

The SalePrice is high for the Stone Masonry veneer type (MasvnrType).

The sale price is high for external material that is of Excellent (Ex) grade.

The sale price is high for materials that are currently in Excellent (Ex) condition on the exterior (ExteCond).

The sale price is high for foundations made of poured concrete (PConc).

The sale price is expensive for the basement's Excellent (100+ inches) (Ex) height (BsmtQual).

The sale price is expensive for the basement's good (Gd) overall condition (BsmtCond).

The highest sale price is for walkout or garden level walls with Good Exposure (Gd) (BsmtExposure).

The maximum SalePrice is for Good Living Quarters (GLQ) of basement completed area (BmtFinType1).

The maximum SalePrice is for the finished basement area of Good Living Quarters (GLQ) and Average Living Quarters (ALQ) (if there are numerous types).

Gas forced warm air furnaces (GasA) and gas hot water or steam heaters (GasW) have high sale prices for the kind of heating (Heating).

Excellent (Ex) heating quality and condition (HeatingQC) command a premium sale price.

The sale price is expensive for buildings with central air conditioning.

The SalePrice is Maximum for Romex (Sbrkr) and Standard Circuit Breakers of Electrical Systems.

Kitchen quality that is Excellent (Ex) is of a high SalePrice.

For Normal Capability(Typ) kind of Home functionality, the SalePrice is high (assume typical unless reductions are justified).

The highest sale price is for an Excellent-Exceptional Masonry Fireplace (Ex) of Fireplace quality (FireplaceQual).

The maximum SalePrice applies to Built-In (Garage section of house - often has space above garage) Garage locations.

The SalePrice is expensive for a garage that has been completely completed on the inside (GarageFinish).

The sale price is high for Excellent (Ex) Garage quality (GarageQual).

The sale price is high for typical/average (TA) and good (Gd) garage conditions (GarageCond).

The sale price is high since the driveway is paved.

The home that was recently built and sold (New) and the contract that required a 15% down payment under normal conditions (Con) had the highest sale price.

The SalePrice is maximal if the home was incomplete when it was last evaluated (related to new homes) (Partial) SalesCondition.

E. Run and evaluate selected models

I have evaluated using the following metrics Mean absolute error, which shows the size of the discrepancy between an observation's real value and its forecast.
One of the most used metrics for assessing the accuracy of forecasts is root mean square deviation, which I have utilised in this case. I utilised the r2 score, which indicates the precision of our model.

Since the objective variable in this problem, SalePrice, is continuous in nature, I may infer that it is a regression-type problem, and to estimate the selling price of the property, I have employed the following regression techniques.
Regression Algorithms
1. RandomForestRegressor
2. Bagging Regressor
3. GradientBoostingRegressor
4. DecisionTreeRegressor

1. RandomForestRegressor

```
RF = RandomForestRegressor()
RF.fit(X_train,Y_train)
pred=RF.predict(X_test)
R2_score = r2_score(Y_test,pred)*100
print('R2_score:',R2_score)
print ("=========================================")
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)))
print ("=========================================")
scores = cross_val_score(RF, X, Y, cv = 10).mean()*100
print("\nCross validation score :", scores)
```

```
R2_score: 90.07788812889656
=========================================
mean_squared_error: 0.10216006032695142
mean_absolute_error: 0.24000479281161544
root_mean_squared_error: 0.31962487438707177
=========================================

Cross validation score : 86.24714129740931
```

Created a Random Forest Regressor model and got accuracy of approximately 86.24%. (Before Cross Validation R2 Score = 90.07%)

2. Bagging Regressor

```
BR = BaggingRegressor()
BR.fit(X_train,Y_train)
pred2=BR.predict(X_test)
R2_score = r2_score(Y_test,pred2)*100
print('\nR2_score:',R2_score)
print ("==========================================")
print('mean_squared_error:',metrics.mean_squared_error(Y_test,pred2))
print('mean_absolute_error:',metrics.mean_absolute_error(Y_test,pred2))
print('root_mean_squared_error:',np.sqrt(metrics.mean_squared_error(Y_test,pred2)))
print ("==========================================")
scores = cross_val_score(BR, X, Y, cv = 10).mean()*100
print("\nCross validation score :", scores)
```

```
R2_score: 87.1618453917562
==========================================
mean_squared_error: 0.13218422310723857
mean_absolute_error: 0.2734382760416678
root_mean_squared_error: 0.36357148280254126
==========================================

Cross validation score : 84.22423712409885
```

Created a Bagging Regressor model and got accuracy of approximately 84.22%. (Before Cross Validation R2 Score = 87.16%)

3. GradientBoostingRegressor

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

```
R2_score: 91.80248364825731
==========================================
mean_squared_error: 0.08440327784089693
mean_absolute_error: 0.22058450558578713
root_mean_squared_error: 0.2905224222687415
==========================================

Cross validation score : 88.01538712073855
```

Created a Gradient Boosting Regressor model and got accuracy of approximately 91.80%. (Before Cross Validation R2 Score = 88.01%)

4. DecisionTreeRegressor

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

```
R2_score: 72.75080843268387
=========================================
mean_squared_error: 0.28056315938998694
mean_absolute_error: 0.38921596637864503
root_mean_squared_error: 0.529682130517905
=========================================

Cross validation score : 68.32628401556391
```

Created a Decision Tree Regressor model and got accuracy of approximately 68.32%. (Before Cross Validation R2 Score = 72.75%)

Hyper Parameter Tunning

```
# Importing GridSearchCV for Hyper parameter tunning for best model
from sklearn.model_selection import GridSearchCV
```

```
RF = RandomForestRegressor()

param ={
       'n_estimators':[100,200],
      'criterion':['mse','mae'],
     'min_samples_split':[2],
    'min_samples_leaf':[1],
}
```

```
RF_grid=GridSearchCV(RandomForestRegressor(),param,cv=4,scoring='accuracy',n_jobs=-1,verbose=2)
```

```
RF_grid.fit(X_train,Y_train)
RF_grid_PRED=RF_grid.best_estimator_.predict(X_test)
```

```
Fitting 4 folds for each of 4 candidates, totalling 16 fits
```

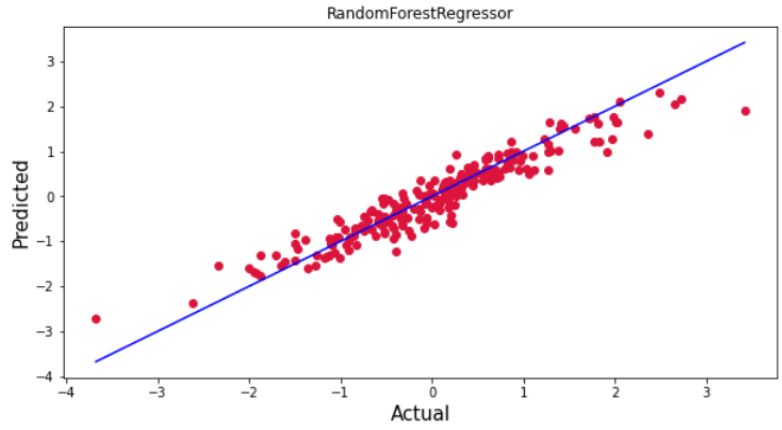
```
RF_grid.best_params_
```

```
{'criterion': 'mse',
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'n_estimators': 100}
```

```
Best_mod=RandomForestRegressor(criterion='mae',min_samples_leaf=2,min_samples_split=2,n_estimators=100)
Best_mod.fit(X_train,Y_train)
pred = Best_mod.predict(X_test)
print('R2_Score:',r2_score(Y_test,pred)*100)
print('mean_squared_error:',metrics.mean_squared_error(Y_test,pred))
print('mean_absolute_error:',metrics.mean_absolute_error(Y_test,pred))
print("RMSE value:",np.sqrt(metrics.mean_squared_error(Y_test, pred)))
```

```
R2_Score: 89.26987638592394
mean_squared_error: 0.1104795118186618
mean_absolute_error: 0.24682372956937923
RMSE value: 0.33238458420730316
```

```
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("RandomForestRegressor")
plt.show()
```



I selected all of the Random Forest Regressor settings, and after fine-tuning the model with the optimal parameters, I increased model accuracy from 86% to 89%. Additionally, mse and rmse readings have decreased, indicating that error has decreased.

Saving Model Predictions

```
#Making dataframe for predicted SalePrice
House_Price_Predictions=pd.DataFrame()
House_Price_Predictions["SalePrice"]=Predicted_Sale_Price
House_Price_Predictions.head(10)
```

	SalePrice
0	-0.491607
1	1.275771
2	1.125489
3	0.463405
4	0.670376
5	0.808376
6	-0.500187
7	0.037161
8	-0.388150
9	-0.877809

```
House_Price_Predictions.to_csv("House_Price_Predictions.csv",index=False)
```

I have saved my model and the image of the left is showing prediction form test data.

CONCLUSION

A. Key Findings and Conclusions of the Study

To anticipate the price of homes, we applied machine learning techniques in our project report. The method for analysing the dataset and determining the connection between the characteristics has been described in detail. As a result, we may choose traits that are independent and not associated with one another. A csv file containing forecasted home prices was created after these feature sets were fed into five algorithms. As a result, we used several performance measures to determine each model's performance and then compared them using these criteria. Then, we additionally stored the dataframe for the test dataset's anticipated prices.

B. Learning Outcomes of the Study in respect of Data Science

Given that it comprises a variety of data types, I found the dataset to be fairly intriguing to work with. It is now feasible to evaluate social data that couldn't previously be recorded, processed, and analysed because to advancements in computing technology. Property research may make advantage of new machine learning analytical approaches. By using a graphical representation, the power of visualisation has enabled us to better grasp the meaning behind the data. One of the most crucial phases is data cleaning, which involves replacing missing values with their corresponding means, medians, and modes as well as null and zero values. In this study, five machine learning algorithms are used to estimate house prices. The techniques are then compared.

C. Limitations of this work and Scope for Future Work

One of the main future plans is to expand the estate database to include other cities, which will enable users to investigate more estates and make informed decisions.
For anybody looking to purchase a home in the region represented by the dataset, I suggest using this model to get a sense of the actual cost. As long as the datasets cover the same cities and regions, the model may

also be applied to datasets covering other locations. I also recommend that individuals evaluate the characteristics that were identified in this study as being the most crucial, since doing so may improve their ability to estimate the price of a home.

Thank you.