Project Report on **Housing Price Prediction**

Submitted By

Ramesh Pyru

ACKNOWLEDGEMENTS

I would like to thank FlipRobo team for their direction, assistance, and guidance. In particular Ms. Khushboo Garg (SME Flip Robo), her recommendations and suggestions have been invaluable for the project.

I also wish to thank “Data trained” which provided the opportunity for internship at FlipRobo.

Special thanks to my family & friends who helped me in many ways.

References use in this project:

1. SCIKIT Learn Library Documentation
2. Blogs from towardsdatascience, Analytics Vidya, Medium
3. Andrew Ng Notes on Machine Learning (GitHub)

TABLE OF CONTENTS

Introduction

1.1 Background

US-based housing company →“Surprise Housing" planning to enter into Australian market needs data Analytics for getting profits on sale of houses. Company is looking at prospective properties to buy houses to enter the market.

1.2 Problem Definition

Need Machine Learning Model in order to predict the actual value of the prospective properties
And decide whether to invest in property or not !!
We need to construct a realistic model to precisely predict the price of real estate property with great potential for enhancements considering the below pointers

- Which variables are important to predict the sale price of house?
- How do these feature variables describe the price of the house?

1.3 Stages of Project

- Data Collection
- Data Cleaning--->Missing Values imputation--->Handling Outliers
  - 2.a. Univariate Analysis(Graphical representation-->Histograms (Bar Charts) , Box Plots, Bar &Group charts, Stem and Leaf Plots)
  - 2.b.Bivariate Analysis(Graphical representation-->Pai Plots, Bar Plot,CrossTab,Scatter Plots)
  - 2.c. Multivariate Analysis(Graphical representation-->HeatMap,LinePlot,PairPlot)
- Choosing the Right Statistical Methods
- Visualizing and Analyzing Results
- Building the Machine Learning Model
- Test the model with Test Data available

The below figure depicts the different steps for EDA analysis

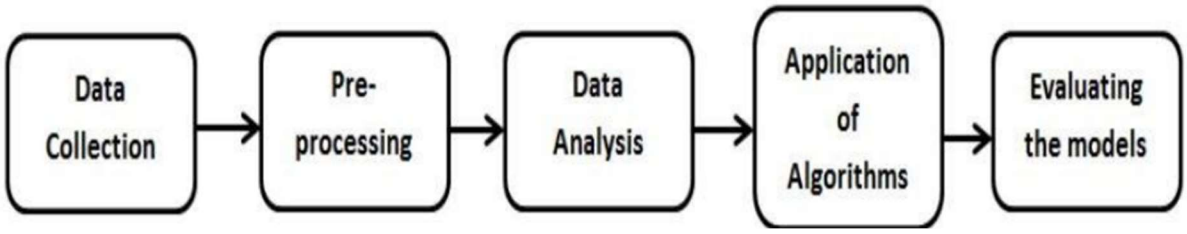


Fig :- High Level Stages

EDA Approach & Outcome

Exploratory Data Analysis (EDA) is an approach to analyze the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations.
Steps involved

1) Importing a dataset
2) Understanding the big picture
3) Preparation
4) Understanding of variables
5) Study of the relationships between variables
6) Brainstorming

Data Preprocessing Done

Data set provided by Flip Robo was in the format of CSV (Comma Separated Values). The dimension of data is 1168 rows and 81 columns. There are 2 data sets that are given. One is training data and one is testing data.

1) Train file will be used for training the model, i.e., the model will learn from this file. It contains all the independent variables and the target variable. Size of training set: 1168 records.

2) Test file contains all the independent variables, but not the target variable. We will apply the model to predict the target variable for the test data. Size of test set: 292 records.

Data pre-processing is an integral step in Machine Learning as the quality of data and the useful information that can be derived from it

- Loading the training dataset as a dataframe
- Checked the number of rows and columns present in our training dataset
- Checked for missing data and the number of rows with null values
- Verified the percentage of missing data in each column and decided to discard the one's that have more than 50% of null values
- Dropped all the unwanted columns and duplicate data present in our dataframe
- Separated categorical column names and numeric column names in separate list variables for ease in visualization
- Checked the unique values information in each column to get a gist for categorical data
- Performed imputation to fill missing data using mean on numeric data and mode for categorical data columns
- Used pie plot, count plot, scatter plot and the others
- With the help of ordinal encoding technique converted all object datatype columns to numeric datatype
- Thoroughly checked for outliers and skewness information
- With the help of heatmap, correlation bar graph was able to understand the Feature vs Label relativity and insights on multicollinearity amongst the feature columns

- Separated feature and label data to ensure feature scaling is performed avoiding any kind of biasness
- Checked for the best random state to be used on our Regression Machine Learning model pertaining to the feature importance details
- Finally created a regression model function along with evaluation metrics to pass through various model formats

Data Inputs- Logic- Output Relationships

When we loaded the training dataset, we had to go through various data preprocessing steps to understand what was given to us and what we were expected to predict for the project. When it comes to logical part the domain expertise of understanding how real estate works and how we are supposed to cater to the customers came in handy to train the model with the modified input data. In Data Science community there is a saying “Garbage In Garbage Out” therefore we had to be very cautious and spent almost 80% of our project building time in understanding each and every aspect of the data how they were related to each other as well as our target label.

With the objective of predicting housing sale prices accurately we had to make sure that a model was built that understood the customer priorities trending in the market imposing those norms when a relevant price tag was generated. I tried my best to retain as much data possible that was collected but I feel discarding columns that had lots of missing data was good. I did not want to impute data and then cause a biasness in the machine learning model from values that did not come from real people

Assumptions

The assumption part for me was relying strictly on the data provided to me and taking into consideration that the separate training and testing datasets were obtained from real people surveyed for their preferences and how reasonable a price for a house with various features inclining to them were.

Hardware and Software Requirements and Tools Used

Hardware Used:

1. RAM: 16 GB
2. CPU: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
3. GPU: Intel® Iris® Xe Graphics

Software Used:

1. Programming language: Python
2. Distribution: Anaconda Navigator
3. Browser based language shell: Jupyter Notebook

Libraries/Packages Used:

- Pandas, NumPy, matplotlib, seaborn, scikit-learn

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

I have used both statistical and analytical approaches to solve the problem which mainly includes the pre-processing of the data and EDA to check the correlation of independent and dependent features. Also, before building the model, I made sure that the input data is cleaned and scaled before it was fed into the machine learning models.

For this project we need to predict the sale price of houses, means our target column is continuous so this is a regression problem. I have used various regression algorithms and tested for the prediction. By doing various evaluations I have selected Gradient Boosting Regressor as best suitable algorithm for our final model as it is giving good r2-score and least difference in r2-score and CV-score among all the algorithms used. Other regression algorithms are also giving me good accuracy but some are over-fitting and some are with under-fitting the results which may be because of less amount of data.

In order to get good performance as well as accuracy and to check my model from over-fitting and under-fitting I have made use of the K-Fold cross validation and then hyper parameter tuned the final model. Once I was able to get my desired final model I ensured to save that model before I loaded the testing data and started performing the data pre-processing as the training dataset and obtaining the predicted sale price values out of the Regression Machine Learning Model.

Testing of Identified Approaches (Algorithms)

The algorithms used on training and test data are as follows:

1) Linear Regression Model
2) SVR--Support Vector Regression Model
3) Decision Tree Regression Model
4) Random Forest Regression Model
5) KNN--K Nearest Neighbor’s Regression Model
6) Gradient Boosting Regression Model
7) Ada Boost Regression Model
8) SGD Regressor Model
9) Extra Trees Regression Model
10) Elastic Net
11) XGBRegressor
12) Ridge Regularization Regression Model
13) Lasso Regularization Regression Model

Key Metrics for success in solving problem under consideration

The key metrics used → r2_score, cross_val_score, MAE, MSE and RMSE.
We tried to find out the best parameters and also to increase our scores by using Hyperparameter Tuning and used GridSearchCV method.

1. **Cross Validation:**

- Dataset is split into K "folds" of equal size.
- Each fold acts as the testing set 1 time, and acts as the training set K-1 times.
- Average testing performance is used as the estimate of out-of-sample performance.
- Also known as cross-validated performance.

Benefits of k-fold cross-validation:

- More reliable estimate of out-of-sample performance than train/test split.
- Reduce the variance of a single trial of a train/test split.

Hence, with the benefits of k-fold cross-validation, we're able to use the average testing accuracy as a benchmark to decide which is the most optimal set of parameters for the learning algorithm.

- If we do not use a cross-validation set and we run grid-search, we would have different sets of optimal parameters due to the fact that without a cross-validation set, the estimate of out-of-sample performance would have a high variance.
- In summary, without k-fold cross-validation the risk is higher that grid search will select hyper-parameter value combinations that perform very well on a specific train-test split but poorly otherwise.

2. R2 Score:

It is a statistical measure that represents the goodness of fit of a regression model. The ideal value for r-square is 1. The closer the value of r-square to 1, the better is the model fitted.

3. Root Mean Squared Error (RMSE):

A metric that tells us how far apart the predicted values are from the observed values in a dataset, on average. The lower the RMSE, the better a model fits a dataset. MSE is a risk function, corresponding to the expected value of the squared error loss. RMSE is the Root Mean Squared Error.

4. Mean Absolute Error (MAE):

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It’s the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

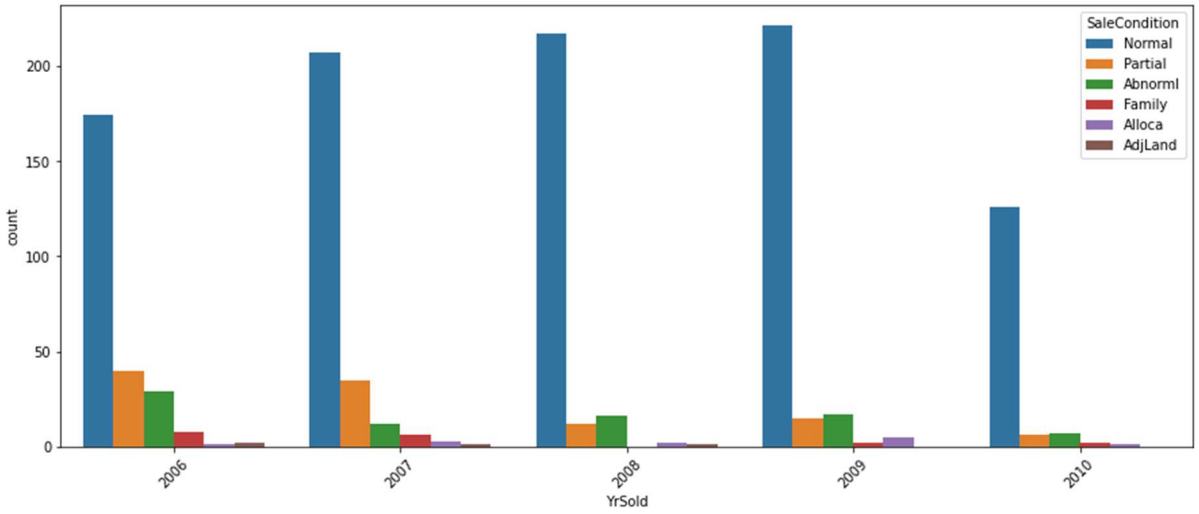
The lower value of MAE, MSE, and RMSE implies higher accuracy of a regression model. However, a higher value of R square is considered desirable.

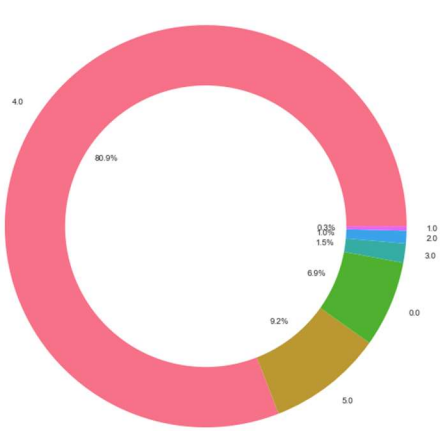
5. Hyperparameter Tuning:

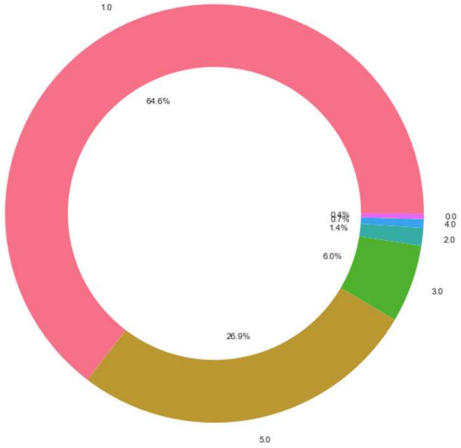
Hyperparameters will define the architecture of the model, and the best part about these is that you get a choice to select these for your model. You must select from a specific list of hyperparameters for a given model as it varies from model to model.This selection procedure for hyperparameter is known as Hyperparameter Tuning. We can do tuning by using GridSearchCV.

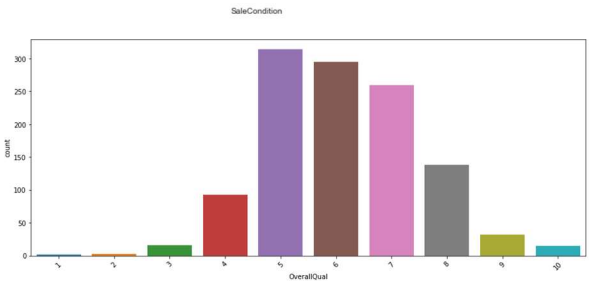
- In essence, the grid search technique allows one to define a grid of parameters that will be searched using K-fold cross-validation.
- Importantly, the grid search technique exhaustively tries every combination of the provided hyper-parameter values in order to find the best model.
- One can then find the highest cross-validation accuracy that matches with the corresponding parameters that optimizes the learning algorithm

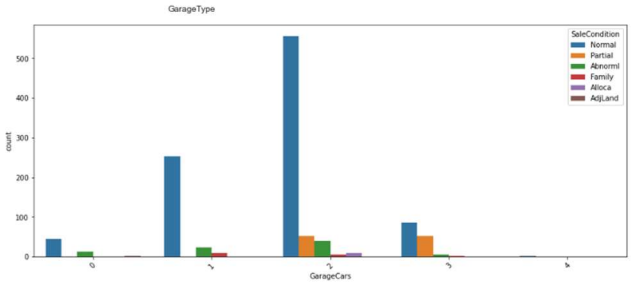
Visualizations(Univariate Analysis)

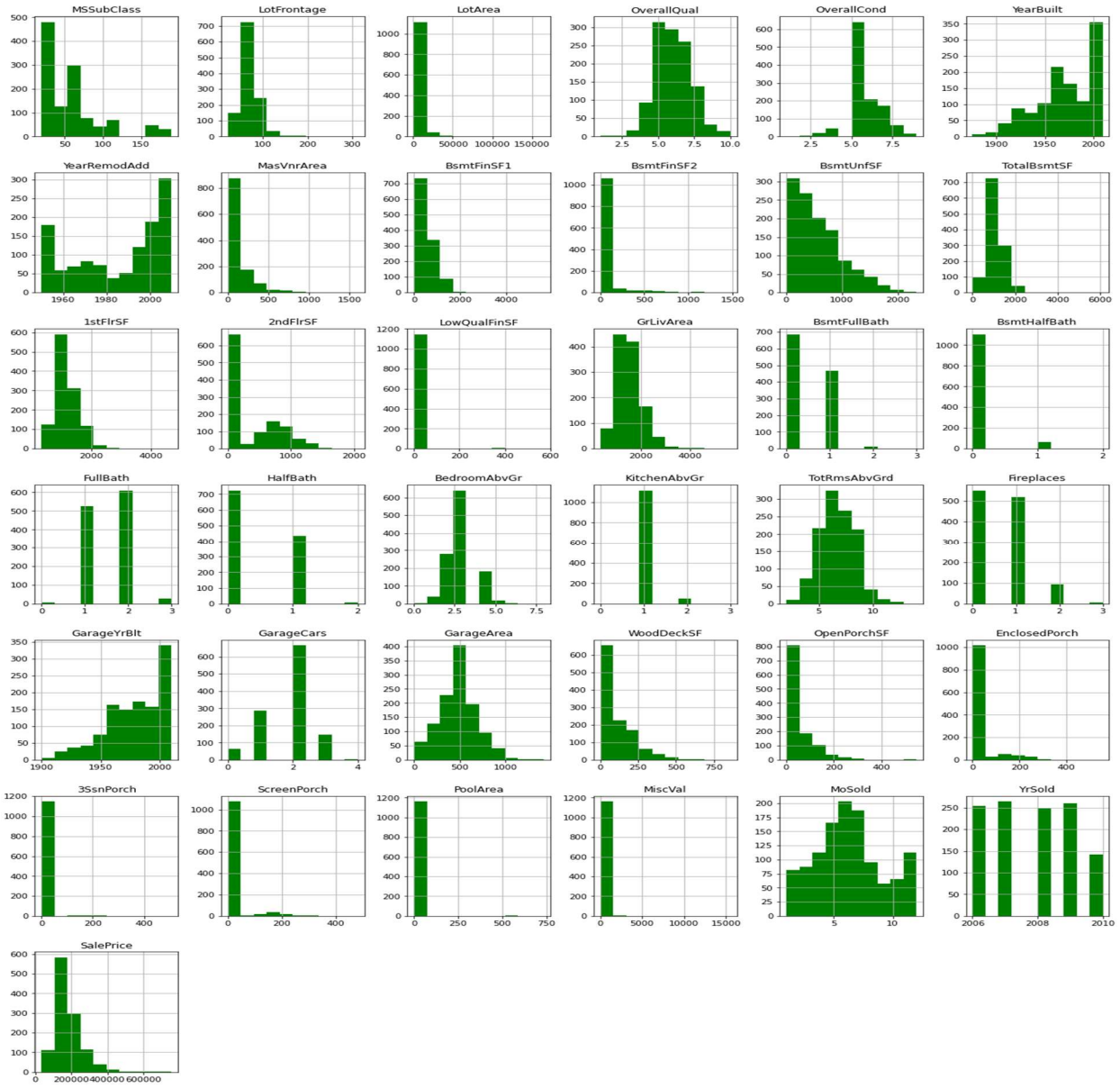










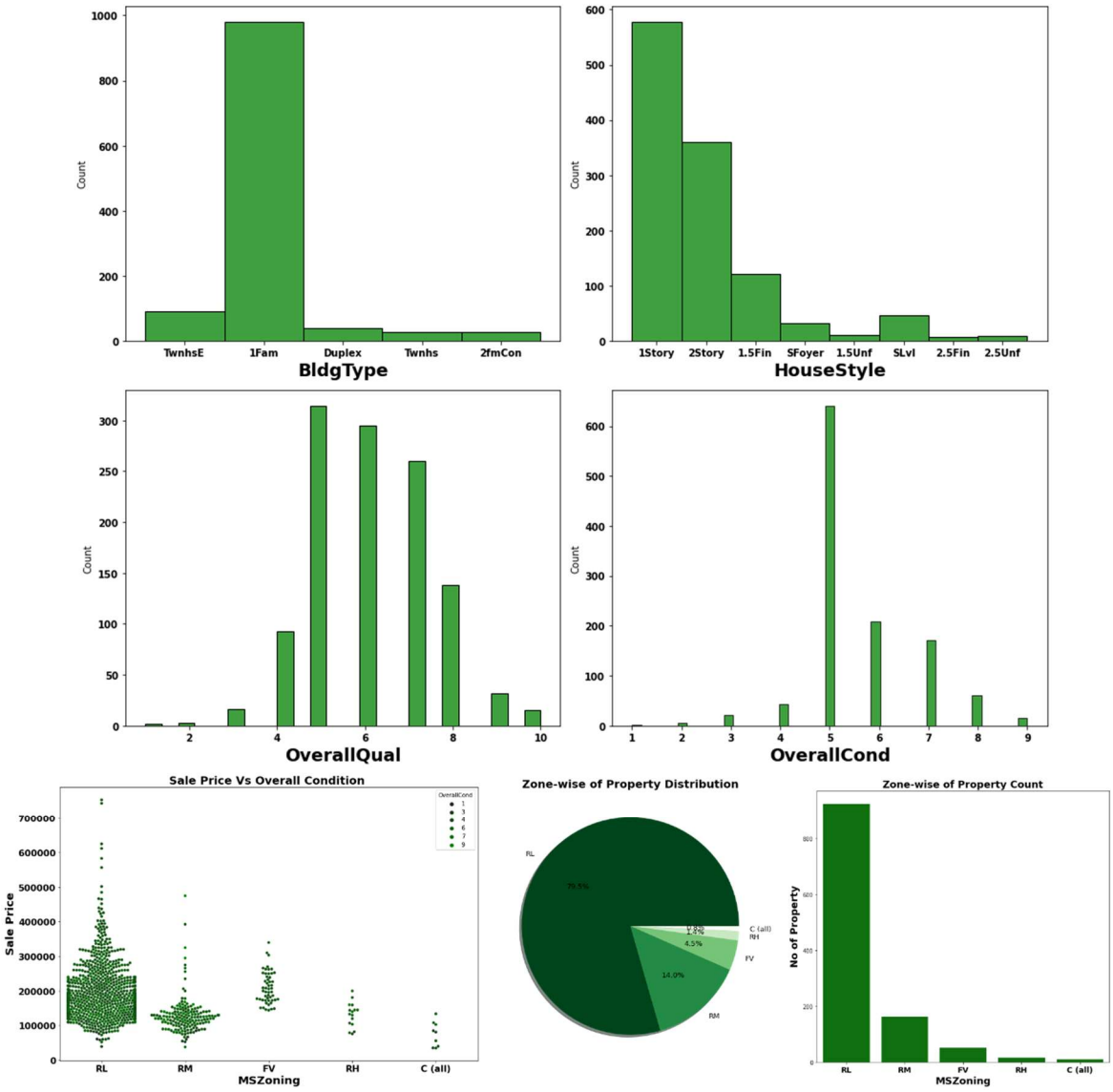


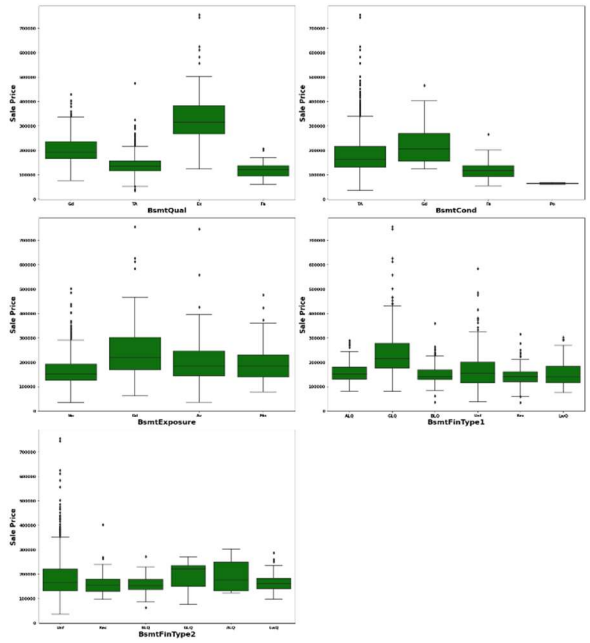
Observations from Univariate Analysis→

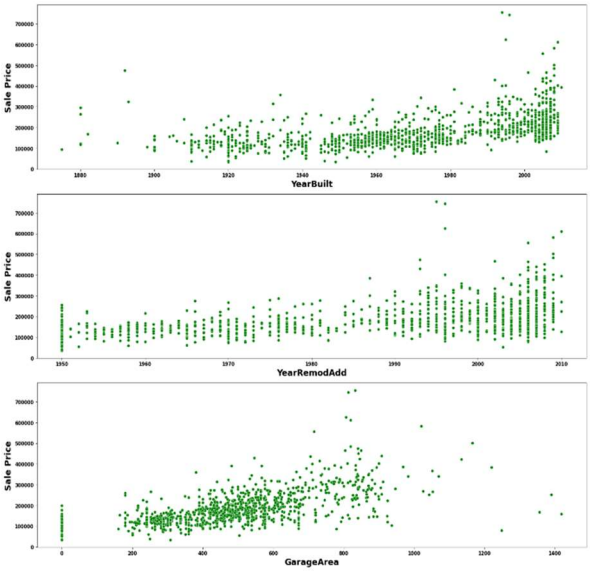
1. For most of the Features Data is not distributed normally, either right or left skewed
2. Lot Frontage: • Data ranges from 21 to 313 with mean value 70.99
3. Year Built: • Data ranges from 1875 to 2010 with mean value 1971
4. YearRemodAdd: • Data ranges from 1950 to 2010 with mean value 1985
5. MassVnrArea: Data ranges from 0 to 1600 with mean value 102.31
6. BsmtFinSF1: • Data ranges from 0 to 5644 with mean value 444.73.
7. BsmtFinSF2: Data ranges from 0 to 1474 with mean value 46.65 •
8. BsmtUnfSF: Data ranges from 0 to 2336 with mean value 569.72
9. YrSold:• Data ranges from 2006 to 2010 with mean value 2007.
10. Sale Price: • Data ranges from 34900 to 755000 with mean value 181477.01

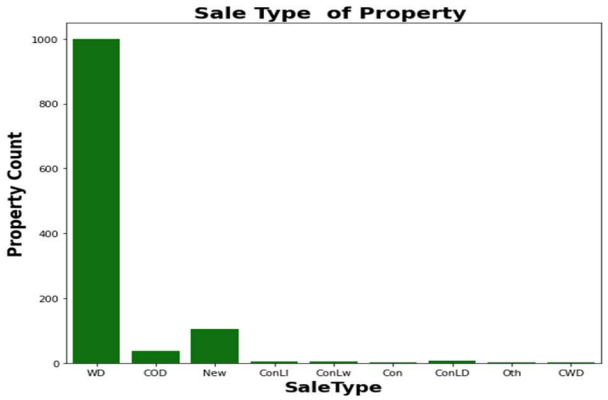
11. Most of the records are for Normally Distributed.
12. OverallQual: • Most of the records are for 5, 6, 7, 8, & 4.
13. Sale Type: • Most of the records are for WD.
14. Sale Condition: • Most of the records are for Normal.
15. Maximum standard deviation of 8957.44 is observed in Lot Area column.
16. Maximum Sale Price of a house observed is 755000 and minimum is 34900.
17. In the columns Full Bath, BedroomAbvGr, Fireplaces, Garage cars, Garage Area, YrSold Median is greater than mean so the columns are negatively skewed.

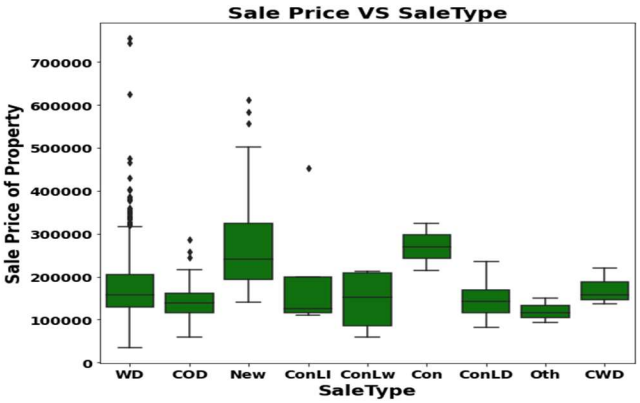
Visualizations(Multivariate Analysis)

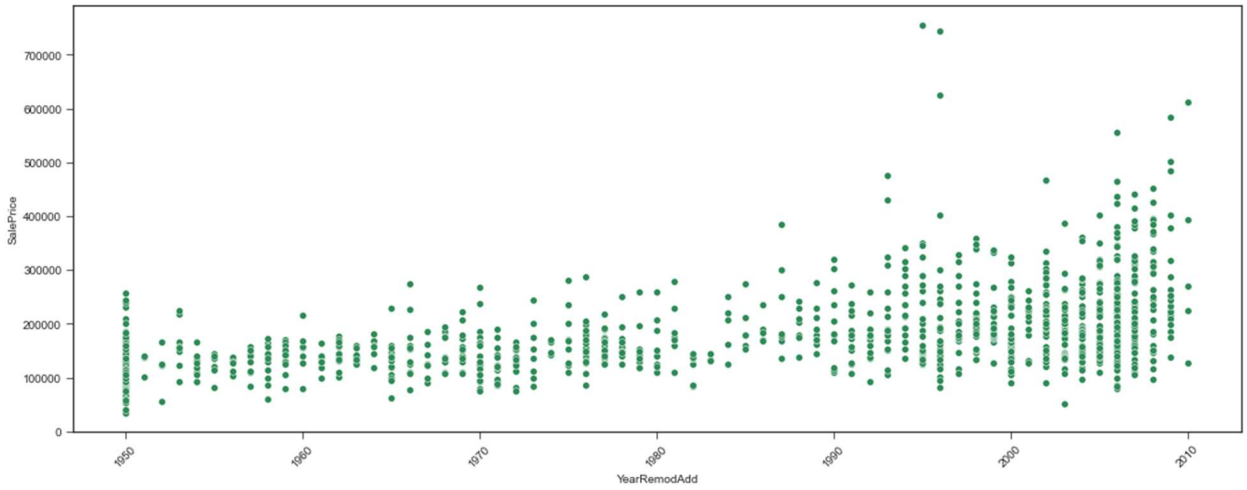


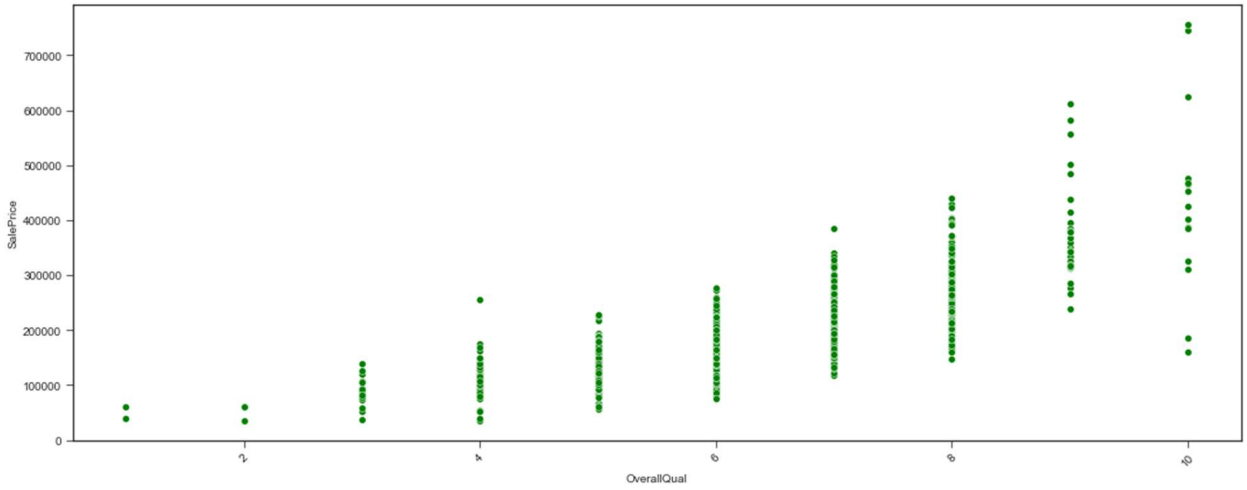

Sale Price
BsmtQual
BsmtCond
BsmtExposure
BsmtFinType1
BsmtFinType2

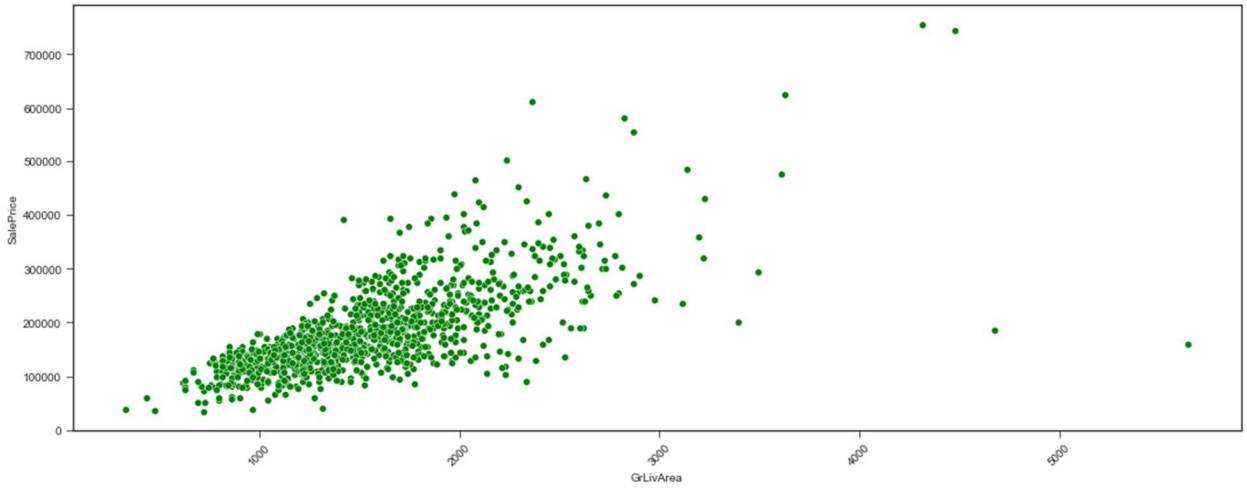

Sale Price
YearBuilt
YearRemodAdd
GarageArea

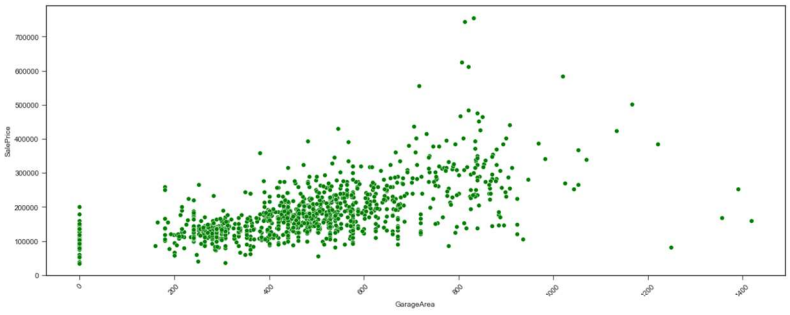

Sale Type of Property
Property Count
SaleType
WD
COD
New
ConLI
ConLw
Con
ConLD
Oth
CWD

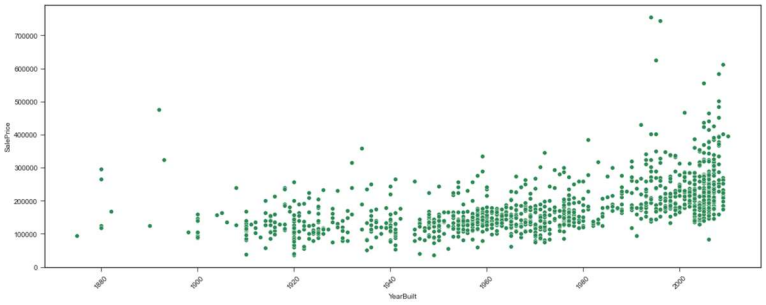

Sale Price VS SaleType
Sale Price of Property
SaleType
WD
COD
New
ConLI
ConLw
Con
ConLD
Oth
CWD


SalePrice
YearRemodAdd


SalePrice
OverallQual

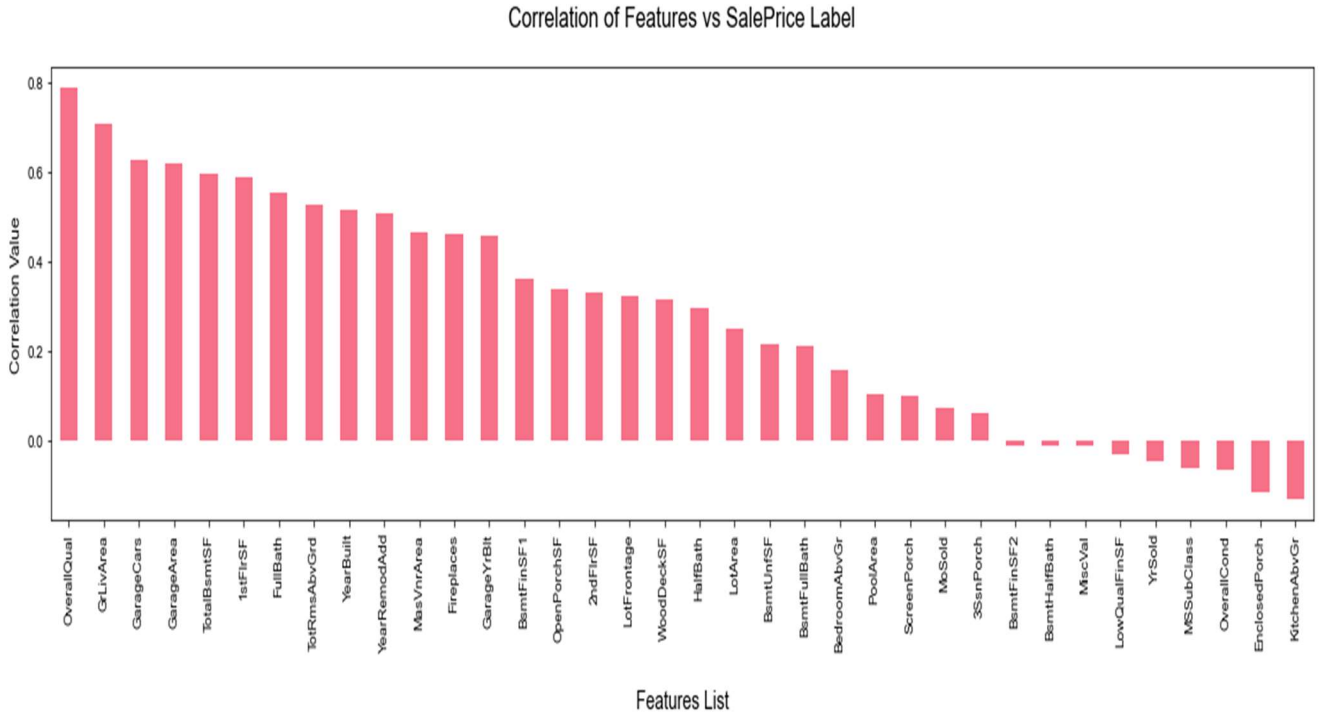

SalePrice
GrLivArea


SalePrice
GarageArea


SalePrice
YearBuilt

Observations→

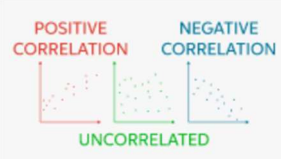
1. 79.5% of House properties belongs to Low Density Residential Area followed by 14 % of properties belong to Medium Density Residential Area.
2. Very Few properties (0.8%) belong to Commercial zone.
3. Let's explore Zone relation with respect to Sale Price
4. Most of property for sale have overall condition rating of either 5 or 6.
5. We already know of 80% of housing data belongs to Low density Residential Area and Now we can see in Swarm Plot that Sale Price inside RL Zone is much higher than another remaining zone. Cheapest properties are available in Commercial zone.
6. Overall Condition Rating may helpful to buyer in taking decision of Buying property but not in determination of House Price.
7. There is No Significant relationship found between Sale price & Lot area. Here we get Important Observation that -
8. As Overall Quality of House Increase the Sale Price of House also Increases
9. In terms of Average Sale price house properties belonging to Floating Village Residential Zone are costlier than rest Effect of Land characteristics on Sale Price Lot Shape Description :- Lot Shape: General shape of property
10. More than 75% of house properties come with overall Quality Rating varies between 5 to 6.



SUMMARY (How we got here!!)

EDA

- Data divided with Datatypes & Top Features
- Univariate Analysis for Features is performed for better insights
- Multivariate analysis is performed across features for Sale Price

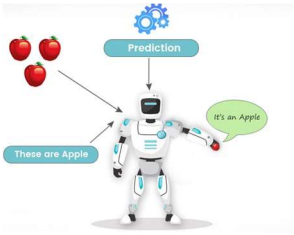


Correlation

- Correlation of Categorical Features
- Multicollinearity is done for finding VIF

Analysis

- Count plot, Pie Chart ,Scatter plots provided insights
- Insights from combination of Multi features are derived



ML Model

- Trained with Multiple Regression models
- **GradientBoostingRegression** model is selected based on the metrics R2Score and having Lesser MAE & RMSE

CONCLUSION

- Housing Price Prediction –helps one to understand the business of real estate. How the price is changing across the Properties.
- Prediction of Sale Price – This helps to predict the future revenues based on inputs from the past and different types of factors related to real estate & property related aspects. This is best done using predictive data analytics to calculate the future price values of houses which helps in segregating houses, identifying the ones with high future value, and investing more resources on them.
- Deployment of ML models – The Machine learning models can also predict the houses depending upon the needs of the buyers and recommend them, so customers can make final decisions as per the needs

	Model	Model R2 Score	Cross Validation R2 Score	Difference in R2 Score	MAE	MSE	RMSE
10	GradientBoostingRegressor()	88.857860	83.671081	0.051868	17496.184121	8.166835e+08	28577.674809
8	RandomForestRegressor()	88.410675	83.929636	0.044810	18216.054103	8.494608e+08	29145.510261
9	ExtraTreesRegressor()	88.200551	82.490560	0.057100	18239.910128	8.648622e+08	29408.540142
12	XGBRegressor()	87.542989	81.446330	0.060967	19300.201656	9.130594e+08	30216.872824
7	ElasticNet()	83.121136	78.904449	0.042167	21694.196661	1.237167e+09	35173.387363
6	Ridge()	83.107514	76.008362	0.070992	22916.022186	1.238166e+09	35187.578659
5	Lasso()	83.095267	75.977739	0.071175	22930.783239	1.239063e+09	35200.331582
0	LinearRegression()	83.092810	75.978425	0.071144	22934.642568	1.239243e+09	35202.889009
3	SGDRegressor()	82.876858	75.751067	0.071258	23231.814865	1.255072e+09	35426.996063
11	AdaBoostRegressor()	81.205542	78.714643	0.024909	25151.824085	1.377574e+09	37115.685951
2	KNeighborsRegressor()	78.365553	72.504415	0.058611	25857.382051	1.585736e+09	39821.305404
1	DecisionTreeRegressor()	80.347582	69.360010	0.109876	26221.666667	1.440460e+09	37953.391179
4	SVR()	-8.018133	-5.896549	0.021216	61801.419124	7.917387e+09	88979.697819

GradientBoostingRegressor is the best model(But with Hyperparameter's rather default) from this study based on r2 score ,MAE & RMSE metrics which helps to predict house prices

DIRECTIONS OF FUTURE WORK

- Many columns are with same entries in more than 80% of rows which lead to reduction in our model performance
- Some additional feature can be added to data which enable us to perform ANN(Artificial Neural network) model
- More features& Data can be available to help in best recommendation.