PFA HOUSING PROJECT

Name of Data Scientist: Vaibhav Tayade

Contact details: vaibhav_t29@rediffmail.com

Project submitted as a part of internship projects

Problem Statement:

Houses are one of the necessary need of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. It is a very large market and there are various companies working in the domain. Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one such housing company. A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia.

Introduction:

Housing market is very unpredictable market but on long run it has always given good returns. There are many factors which affects the Housing markets. How many years old the property is, Utilities around the property, Built Quality of the property, road access to the property, zoning classification of the property etc. Here we are trying to understand how these factors affects the selling price of the property. We have to build a machine learning model which can predict the selling price of property.

Content of the dataset:

MSSubClass: Identifies the type of dwelling involved in the sale.

MSZoning: Identifies the general zoning classification of the sale.

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access to property

Alley: Type of alley access to property

LotShape: General shape of property

LandContour: Flatness of the property

Utilities: Type of utilities available

LotConfig: Lot configuration

LandSlope: Slope of property

Neighborhood: Physical locations within Ames city limits

Condition1: Proximity to various conditions

Condition2: Proximity to various conditions (if more than one is present)

BldgType: Type of dwelling

HouseStyle: Style of dwelling

OverallQual: Rates the overall material and finish of the house

OverallCond: Rates the overall condition of the house

YearBuilt: Original construction date

YearRemodAdd: Remodel date

RoofStyle: Type of roof

RoofMatl: Roof material

Exterior1st: Exterior covering on house

Exterior2nd: Exterior covering on house

MasVnrType: Masonry veneer type

MasVnrArea: Masonry veneer area in square feet

ExterQual: Evaluates the quality of the material on the exterior

ExterCond: Evaluates the present condition of the material on the exterior

Foundation: Type of foundation

BsmtQual: Evaluates the height of the basement

BsmtCond: Evaluates the general condition of the basement

BsmtExposure: Refers to walkout or garden level walls

BsmtFinType1: Rating of basement finished area

BsmtFinSF1: Type 1 finished square feet

BsmtFinType2: Rating of basement finished area

BsmtFinSF2: Type 2 finished square feet

BsmtUnfSF: Unfinished square feet of basement area

TotalBsmtSF: Total square feet of basement area

Heating: Type of heating

HeatingQC: Heating quality and condition

CentralAir: Central air conditioning

Electrical: Electrical system

1stFlrSF: First Floor square feet

2ndFlrSF: Second floor square feet

LowQualFinSF: Low quality finished square feet

GrLivArea: Above grade (ground) living area square feet

BsmtFullBath: Basement full bathrooms

BsmtHalfBath: Basement half bathrooms

FullBath: Full bathrooms above grade

HalfBath: Half baths above grade

Bedroom: Bedrooms above grade

Kitchen: Kitchens above grade

KitchenQual: Kitchen quality

TotRmsAbvGrd: Total rooms above grade

Functional: Home functionality

Fireplaces: Number of fireplaces

FireplaceQu: Fireplace quality

GarageType: Garage location

GarageYrBlt: Year garage was built

GarageFinish: Interior finish of the garage

GarageCars: Size of garage in car capacity

GarageArea: Size of garage in square feet

GarageQual: Garage quality

GarageCond: Garage condition

PavedDrive: Paved driveway

WoodDeckSF: Wood deck area in square feet

OpenPorchSF: Open porch area in square feet

EnclosedPorch: Enclosed porch area in square feet

3SsnPorch: Three season porch area in square feet

ScreenPorch: Screen porch area in square feet

PoolArea: Pool area in square feet

PoolQC: Pool quality

Fence: Fence quality

MiscFeature: Miscellaneous feature not covered in other categories

MiscVal: $Value of miscellaneous feature

MoSold: Month Sold (MM)

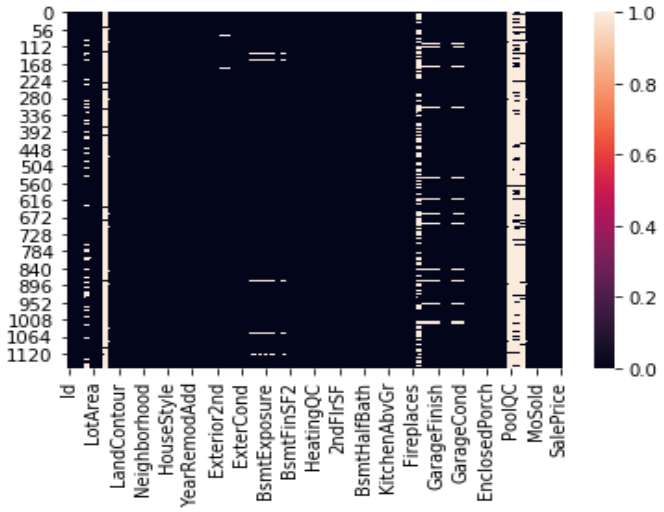
YrSold: Year Sold (YYYY)

SaleType: Type of sale

SaleCondition: Condition of sale

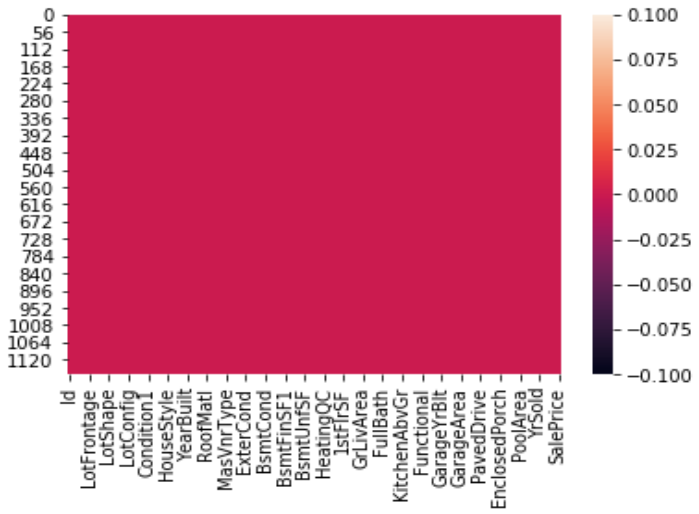
CHECKING FOR NULL VALUES:

We have observed there are many null values present in the dataset by plotting heatmap.



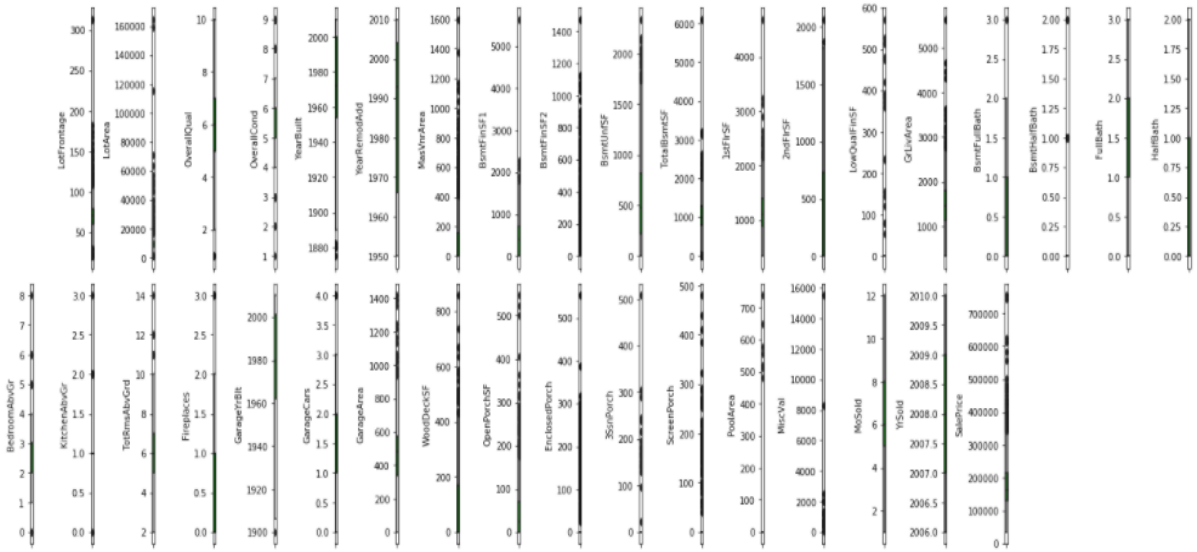
We can handle replace this null values with mean, median or mode values of that particular depending of the data available in that column and also depending of distplot of that particular. We have replaced

null values as shown in the python file. Now we can ensure that there are no Null Values in the dataset by again plotting heatmap.



We have also dropped few columns which seems to be very much irrelevant with the dataset or which contains huge null values like ‘Alley’, ‘FireplaceQu’, ‘PoolQC’, ‘Fence’, ‘MiscFeature’, ‘Id’ columns.

Checking Outliers and Skewness:



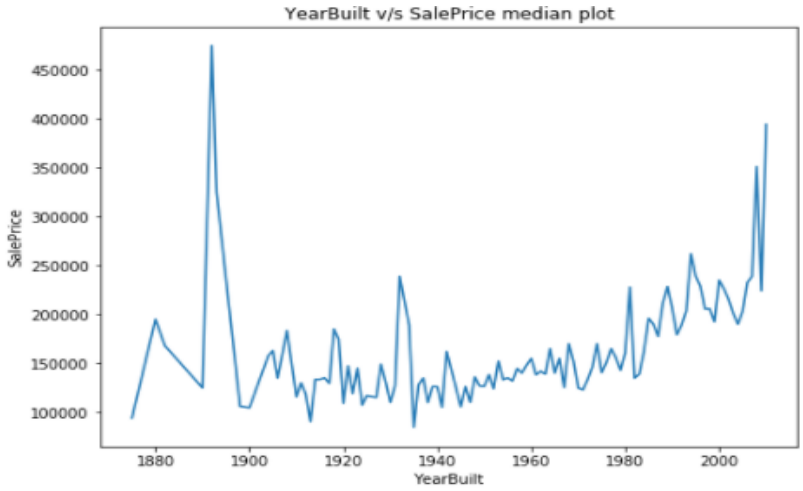
By plotting box plot of all numerical columns, we get to know that almost all columns contains **outliers** in them.

```
Out[117]: MSSubClass       1.422019
          LotFrontage      2.710383
          LotArea         10.659285
          OverallQual      0.175082
          OverallCond      0.580714
          YearBuilt       -0.579204
          YearRemodAdd    -0.495864
          MasVnrArea       2.834658
          BsmtFinSF1       1.871606
          BsmtFinSF2       4.365829
          BsmtUnfSF        0.909057
          TotalBsmtSF      1.744591
          1stFlrSF         1.513707
          2ndFlrSF         0.823479
          LowQualFinSF     8.666142
          GrLivArea        1.449952
          BsmtFullBath     0.627106
          BsmtHalfBath     4.264403
          FullBath         0.057809
          HalfBath         0.656492
          BedroomAbvGr     0.243855
          KitchenAbvGr     4.365259
          TotRmsAbvGrd     0.644657
          Fireplaces       0.671966
          GarageYrBlt     -0.674913
          GarageCars      -0.358556
          GarageArea       0.189665
          WoodDeckSF       1.504929
          OpenPorchSF      2.410840
          EnclosedPorch    3.043610
          3SsnPorch        9.770611
          ScreenPorch      4.105741
          PoolArea        13.243711
          MiscVal         23.065943
          MoSold           0.220979
          YrSold           0.115765
          SalePrice        1.953878
          dtype: float64
```

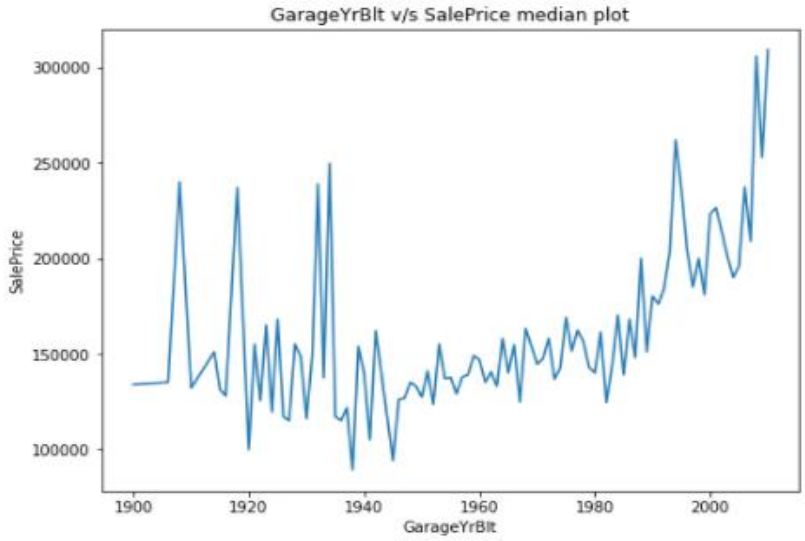
We can observe that almost all columns have presence of skewness in the dataset.

We have also checked skewness by plotting distplot of all columns individually (check ipynb file).

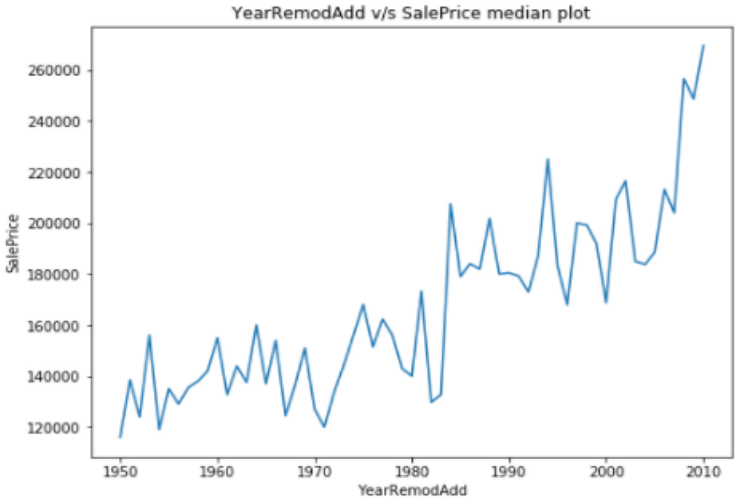
BIVARIATE ANALYSIS:



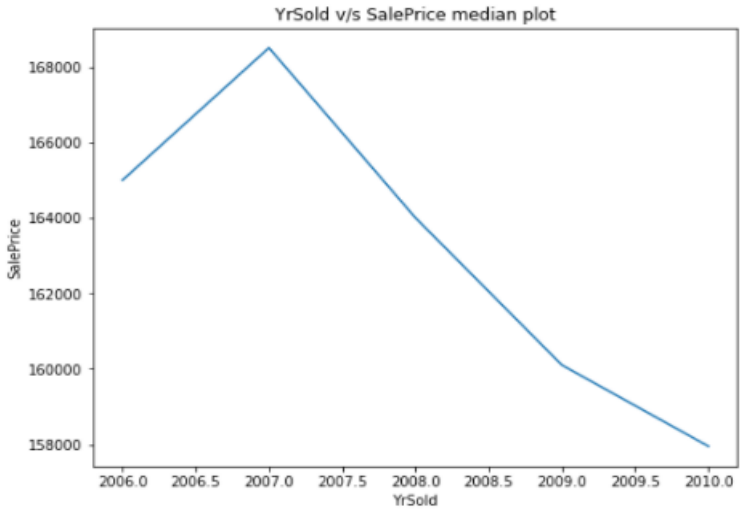
Observation: We can see that from the above median plot, most of sale price ranges from 100000 to 150000 of all yearbuilt range. We can also see that from 1885 to 1990 yearbuilit range, we see huge sale price growth. we can see also see sudden growth in sales price from 1980 to 2000 and above years.



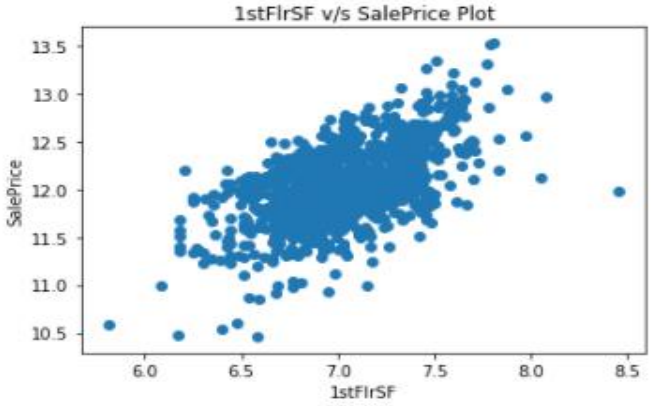
Observation: We can see that from the above median plot, most of sale price ranges from 100000 to 250000 of all GarageYrBuilt range. We can see also see sudden growth in sales price from 1980 to 2000 and above years.



Observation: We see slow and steady growth in salesprice as years range from 1950 to 2010. 1950 YearRemodAdd has least SalePrice and 2010 YearRemodAdd has maximum sale price.

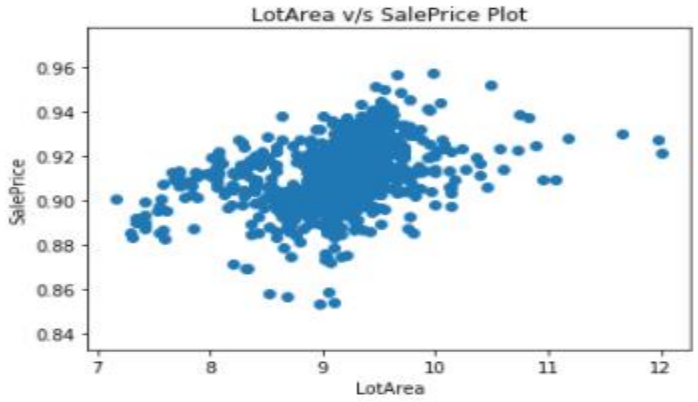


Observation: 2007 yearsold property is having maximum sale price of more than 168000. and curve is deceasing as we go from 2007 to 2010 years range

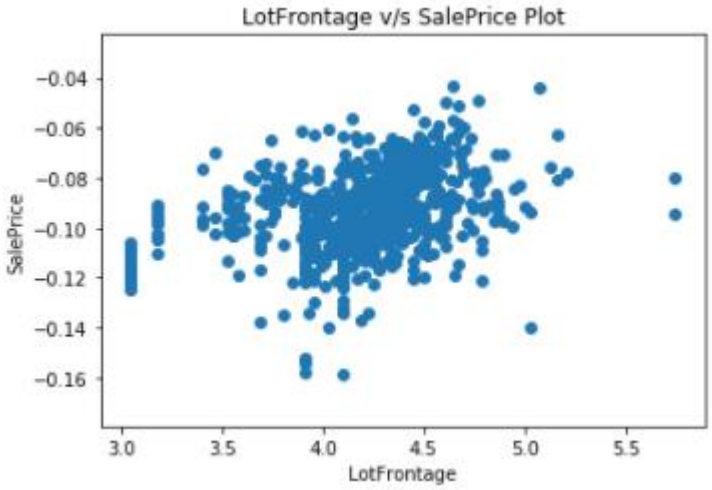


Observation: Most of the Sales Price values range from 11.0 to 13.0 and Most of the 1stFlrSF values range from 6.2 to 7.8. Most of the dataset are linearly spreaded.

Observation: Most of the Sales Price values range from 2.40 to 2.58 and Most of the GrLivArea values range from 6.5 to 8. Most of the dataset are linearly correlated.



Observation: Most of the Sales Price values range from 0.87 to 0.95 and Most of the LotArea values range from 7.5 to 10.5 Most of the dataset are widespreaded.



Observation: Most of the Sales Price values range from -0.13 to -0.06 and Most of the LotFrontage values range from 3.5 to 5. Most of the dataset are widespreaded across.

OUTLIERS AND SKEWNESS:

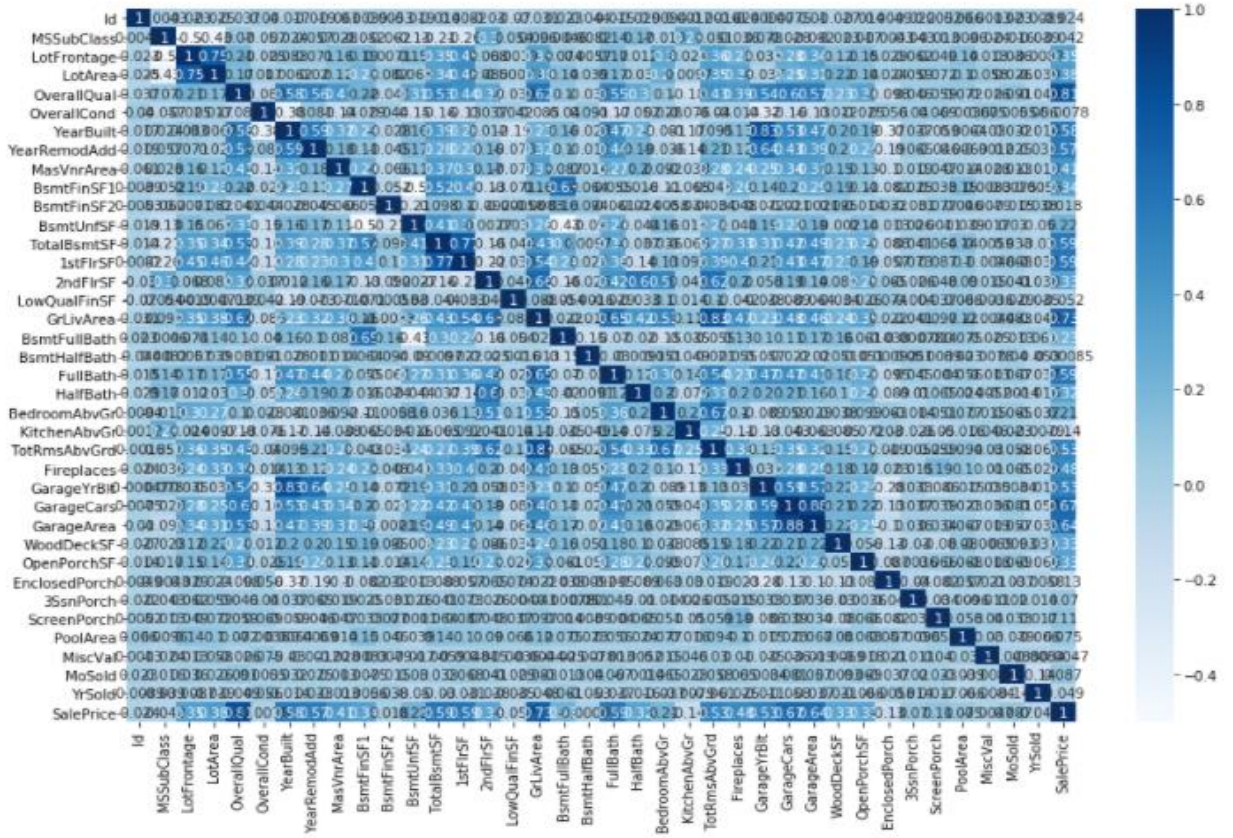
REMOVING OUTLIERS AND SKEWNESS: Here in this dataset, we have used z-score method of removing outliers. A **z-score** (also called a standard score) gives you an idea of how far from the mean a data point is. But more technically it's a measure of how many standard deviations below or above the population mean a raw score is. A z-score can be placed on a normal distribution curve. We can choose a threshold value, Here, in this problem we have choosen 3 as a threshold value above which all are outliers.

We tried to remove outliers and skewness with the help of z-score method, but we observed that out dataset is full of it and it may play an important role in the dataset, it might be the intrinsic property of the dataset, so we kept outliers and skewness as it is.

LABEL ENCODING THE FILE:

Before proceeding further, we need to confirm that all columns are numeric in nature, but we found out that there are columns which are of 'object' data type in nature. We need to convert it into numeric data types, we can do that using **Label Encoding technique** for converting this column in to numeric. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

CHECKING CORRELATION MATRIX:



Observation: We can see that there are various columns which are highly correlated with the Sales price, like LotArea, OverallQual, YearBuilt, YearRemodAdd, MasVnrArea, GrLivArea, GarageCars, GarageArea etc.

SCALING THE DATASET: The most important step in machine learning model making is to ensure all data columns are in same scale of values, we can bring them in same scale by scaling them using Standard Scaler scaling library. Standard Scaler removes the mean and scales each feature/variable to unit variance. This operation is performed feature-wise in an independent way. StandardScaler can be influenced by outliers (if they exist in the dataset, but we have already removed outliers) since it involves the estimation of the empirical mean and standard deviation of each feature.

TRAIN-TEST DATASET BUILING: The train-test split is a technique for evaluating the performance of a machine learning algorithm. It can be used for classification or regression problems and can be used for any supervised learning algorithm. The procedure involves taking a dataset and dividing it into two subsets Here we have kept 20% of dataset for testing the model while 80% of dataset for training the model.

BUILDING MACHINE LEARNING MODELS:

Machine learning model making: Here in our problem we will try making machine learning model with seven different types of algorithms i.e. Linear Regression, Lasso Method Regression, Ridge Method Regression, ElasticNet Regression, Decision Tree Regressor, Random Forest Regressor, Ada Boost Regressor, Support Vector Regressor methods. We will try to find out each model's Accuracy score, Mean Absolute error, Mean Squared Error and Root Mean Squared Error as well. Accuracy score will give us the percentage accuracy of the model in predicting the test datasets. Mean Absolute Error is nothing but the magnitude of difference between the prediction of an observation and the true value of that observation. Mean Squared Error is nothing but the average of the square of the difference between the original values and the predicted values. Root mean square error or root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions. It shows how far predictions fall from measured true values using Euclidean distance.

LinearRegression: Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x).

The term linear model implies that the model is specified as a linear combination of features. Based on training data, the learning process computes one weight for each feature to form a model that can predict or estimate the target value.

Our LinearRegression Model has shown accuracy score of 86.29%

MEAN ABSOLUTE ERROR: 24022.45683066386

MEAN SQUARED ERROR: 2150878999.415809

ROOT MEAN SQUARED ERROR: 46377.57000335193

r2 Score of Linear Regression model: 0.6917192367083568

Lasso regression: In statistics and machine learning, lasso (least absolute shrinkage and selection operator) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the resulting statistical model.

Our Lasso regression model has shown accuracy of 86.29%

Ridge regression: Ridge regression is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values to be far away from the actual values.

Our Ridge Regression model has shown accuracy of 86.29%

ElasticNet Regression: Elastic net is a popular type of regularized linear regression that combines two popular penalties, specifically the L1 and L2 penalty functions. Elastic Net is an extension of linear regression that adds regularization penalties to the loss function during training.

Our ElasticNet Regression model has shown accuracy of 86.29%

MEAN ABSOLUTE ERROR: 23910.36203251475

MEAN SQUARED ERROR: 2139220275.6950967

ROOT MEAN SQUARED ERROR: 46251.70565173891

Decision Tree Regression: Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes

Our Decision Tree Regression model has shown accuracy of 100%

MEAN ABSOLUTE ERROR: 30060.94017094017

MEAN SQUARED ERROR: 2636923818.863248

ROOT MEAN SQUARED ERROR: 51350.986542258834

Random Forest Regression: Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model.

Our Random Forest Regression model has shown accuracy of 97.93%

MEAN ABSOLUTE ERROR: 19627.468632478634

MEAN SQUARED ERROR: 1469426120.7861297

ROOT MEAN SQUARED ERROR: 38333.094328349354

Support Vector Regression: SVR is built based on the concept of Support Vector Machine or SVM. It is one among the popular Machine Learning models that can be used in classification problems or assigning classes when the data is not linearly separable.

Our Support Vector Regression model has shown accuracy of 10.13%

MEAN ABSOLUTE ERROR: 51617.76948360711

MEAN SQUARED ERROR: 6287866307.431715

ROOT MEAN SQUARED ERROR: 79296.0674146689

Ada Boost Regression: AdaBoost algorithm, short for Adaptive Boosting, is a Boosting technique used as an Ensemble Method in Machine Learning. It is called Adaptive Boosting as the weights are re-assigned to each instance, with higher weights assigned to incorrectly classified instances. An AdaBoost regressor is a meta-estimator that begins by fitting a regressor on the original dataset and then fits additional copies of the regressor on the same dataset but where the weights of instances are adjusted according to the error of the current prediction.

Our Ada Boost Regression model has shown accuracy of 88.12%

MEAN ABSOLUTE ERROR: 25910.75141344698

MEAN SQUARED ERROR: 1637974198.5510192

ROOT MEAN SQUARED ERROR: 40471.89393333378

Observation:

After checking all models accuracy score and cross validation score and also their errors, we found out that Ada Boost Regressor seems to be the best model, as it has good accuracy score and least error comparatively. The difference between the accuracy score and cross validation score is also minimal.

Hyper Parameter Tuning Model:

Hyperparameter tuning is basically choosing a set of optimal hyperparameters for a learning algorithm which is mostly done with the help of grid search cross validation method. Grid search cv is basically the most used hyperparameter tuning method. With this technique, we simply build a model for each possible combination of all of the hyperparameter values provided, evaluating each model, and selecting the architecture which produces the best results.

After Hyper parameter tuning the model we found out that below parameters are best parameters in order to make final model.

learning_rate: **2**, loss: **exponential**, n_estimators: **150**, random_state: **42**

Final Model Making:

Our final model is made using best parameters obtained in hyper parameter tuning, we found out the accuracy of **88.43%** earlier it was **88.12%** which is slightly increased by **0.31%.**

So we can now finally say that our model is having accuracy of **88.43%.**

SAVING THE MODEL:

Our final model is saved in pkl format with **'Vaibhav_PFA_Housing_project_Model.pkl'** file.

CONCLUSION:

1) Ada Boost Regression model is the best fit model for PFA Housing Project Sales Prices Prediction.
2) Our model is having accuracy of **88.43%.**
3) **LotArea, OverallQual, YearBuilt, YearRemodAdd, MasVnrArea, GrLivArea, GarageCars, GarageArea** are highly correlated with the sales prices.
4) We can also see sudden growth in sales price from **1980 to 2000** and above years with compared to YearsBuilt.
5) In this dataset, Outliers cannot be removed because they are holding intrinsic property of the dataset.

Thank You