FLIP ROBO

HOUSING PRICE PREDICTION PROJECT

Submitted by:

JASMINE KAUR

INTERNSHIP 18

ACKNOWLEDGMENT

The internship opportunity at Flip Robo Technologies has given me a great scope of learning and developing my skills in the field of data science. I am very grateful to our SME Khushboo Garg for her valuable and constructive suggestions during the development of this project. Also, I am thankful to the technical team for resolving my queries.

I am very thankful to DT support team for their continuous support and guidance.

INTRODUCTION

- Business Problem Framing

  An important reflection of the picture of any economy is its housing prices. Predicting the prices of houses can be very beneficial for both buyers as well as sellers. With this, the agents can make informed decisions about the property and thus can maximize their profits. On the other hand, customers can utilize this information and optimize the time of buying a particular property in order to minimize their cost. Thus, predicting the right price is very crucial for both sides of the party.

  There are three factors that influence the price of a house:

  1. Physical condition
  2. Concept
  3. Location

  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.

- Conceptual Background of the Domain Problem

  In this project, we need to build a model using machine learning to predict the actual value of the prospective properties and this will help them decide whether to invest or not.

  For this task, the company wants to know:

  1. Which variables are important to predict the price of variable?
  2. How do these variables describe the price of the house?

  Therefore, we are required to predict the price of houses with the available independent variables. This model is then used to understand how the prices vary with the variables. The team can then manipulate the strategy of the firm and can focus on areas that give high returns.

  This machine learning problem is characterized as supervised learning.

  SUPERVISED LEARNING: In supervised learning, we are given a dataset along with the correct output, and thus we are able to find a relationship between input and output.

  REGRESSION PROBLEM: As we dealing with continuous target variable, it means that our goal to map input variable to some continuous function. This is called regression problem.

  The goal in this project is to build an end-to-end solution that is capable of predicting the house prices better and also understands the relationship between house features. Also, how these features are used to predict the house prices.

- Review of Literature

  In recent years, the use of machine learning has gained popularity. It is basically due to its immense capability of dealing with the complex real life problems.

  In this project of predicting the house prices, we have used machine learning models to reach at our goal.

- Motivation for the Problem Undertaken

  Predicting housing prices is a very essential task in order to reap the benefits from the real estate market. This knowledge of knowing the property prices is of great importance for both buyers and sellers.

  Machine learning models will definitely help the housing companies to earn profit. Along with it, these models will help the customers to purchase house as per their requirements and budget. Thus, this project will help in achieving these objectives.

Analytical Problem Framing

- Mathematical/ Analytical Modelling of the Problem

In the dataset, we are given with both input (features) and the output data (target variable), thus it is a supervised machine learning problem. The target variable “Sale Price” is a continuous variable, so it is a regression problem. We have performed regression tasks and it models a relationship between independent variables and the target variable which in turns predict the Sale Price.

The first step in the process of data analysis is exploratory data analysis (EDA). It is performed by summarizing various characteristics of the data, visualization techniques etc. In this step, we try to understand the data and note down the insights related to skewness, missing values, outliers, and multicollinearity. These are done using statistical summary, correlation matrix, pairplot, univariate analysis, bivariate analysis etc.

STATISTICAL SUMMARY:

	count	mean	std	Min	25%	50%	75%	max
MSSubClass	1168	56.76	41.94	20	20	50	70	190
LotFrontage	954	70.98	24.82	21	60	70	80	313
LotArea	1168	10484.74	8957.44	1300	7621.5	9522.5	11515.5	164660
OverallQual	1168	6.10	1.39	1	5	6	7	10
OverallCond	1168	5.59	1.12	1	5	5	6	9
YearBuilt	1168.0	1970.93	30.14	1875	1954	1954	1972	2000
YearRemodAdd	1168.0	1984.75	20.78	20.78	1966	1993	2004	2010
MasVnrArea	1168.0	102.31	182.59	182.59	0	0	160	1600

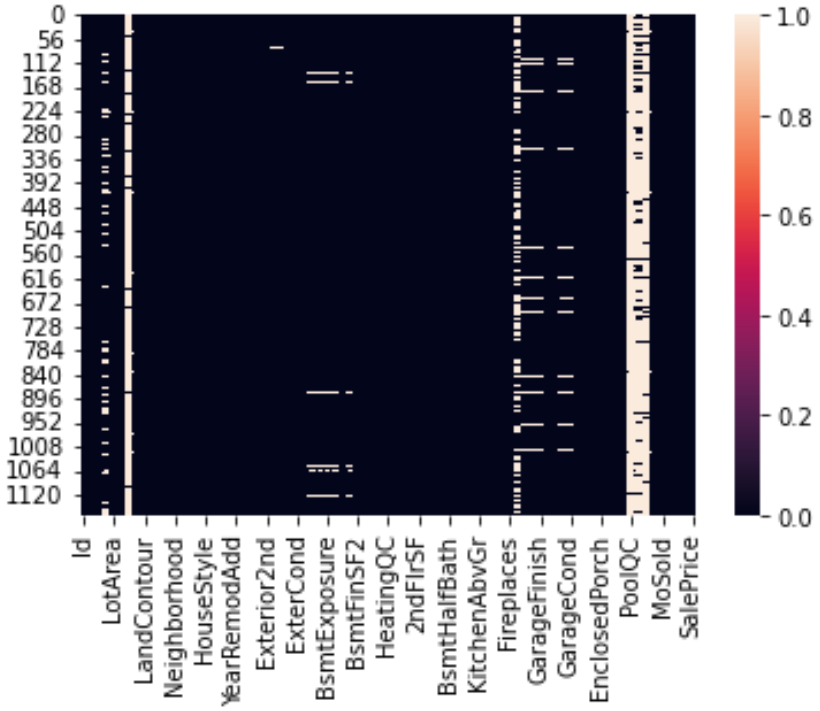
BsmtFinSF1	1168.0		462.66	0	0	385.5	714.5	5644
BsmtFinSF2	1168.0		163.52	0	0	0	0	1474
BsmtUnfSF	1168.0		449.37	0	216	474	816	2336
TotalBsmtSF	1168.0		442.27	0	799	1005.5	1291	6110

OBSERVATION:

1. LotArea has the highest standard deviation
2. Chances of positive correlation in the features: In MSSubclass, LotArea, MasVnrArea, BsmtFinSF1, BsmtFinSF2, BsmtUnfsF, TotalBsmtSF, 1stFlrSF, 2ndFlrSF, LowQualFinSF, GrLivArea, BsmtFullBath, HalfBath, TotRmsAbvGrd, WoodDeckSF, OpenPorchSF, EnclosedPorch, 3SsnPorch, ScreenPorch, PoolArea, Miscval & salePrice.
3. Chances of negative correlation in the features: Full Hd, Bedroom, Fireplace, Garage Car, Garage Area & YrS old.
4. There might be outliers in the following features as there is a huge difference between the 75^{th} percentile and maximum value: MSSubClass, LotFrontage, LotArea, MasVnrArea, BsmtFinSF1, BsmtFinSF2, BsmtUnfSF, TotalBsmtSF, 1stFlrSF, 2ndFlrSF, LowQualFinSF, GrLivArea, BsmtHalfBath, BedroomAbvGr, ToRmsAbvGrd, GarageArea, WoodDeckSF, OpenPorchSF, EnclosedPorch, 3SsnPorch, ScreenPorch, PoolArea, MiscVal & SalePrice

MISSING DATA:-

Missing data, or missing values, occur when **no data value is stored** for the variable in an observation. Missing data are a common occurrence and can have a significant effect on the conclusions that can be drawn from the data.



OBSERVATION:

Following features have majority of missing data:

PoolQC, MiscFeature, Alley, Fence & FireplaceQu.

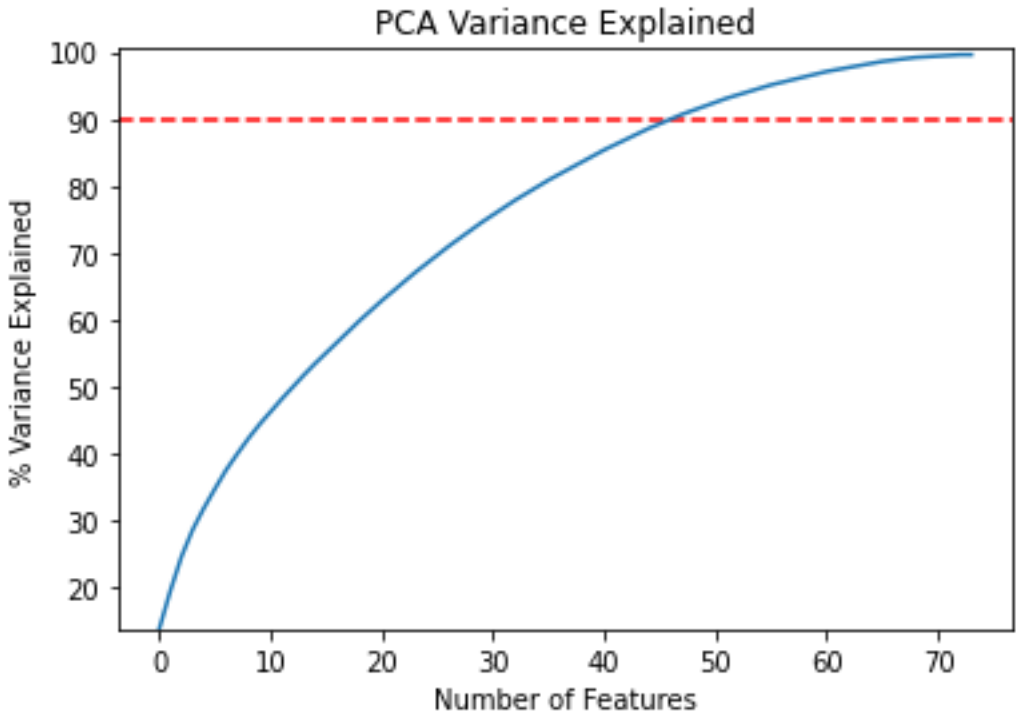
We applied imputation methods to deal with the problem of missing data. We have tackled with the missing values of category and continuous data type separately.

Removing Outliers:

We have applied z score and IQR method to deal with the outliers but it leads to a lot of data loss. Due to this, we have not applied these techniques rather we shift to PCA to deal with it.

PCA:-

Principal Component Analysis is the process of computing the principal components and using them to perform a change on the basis of the data. Due to multicollinearity, we have to apply PCA. The plot of PCA with variance is shown below:



As per PCA and variance, we need 45 features for retaining 90% of the information.

CORRELATION:

There are many correlated values on the columns, so it is a good idea to drop some of the features.

SKEWNESS:

The right range of skewness is between -0.5 and 0.5 as this is considered as fairly symmetrical. But the dataset has features that have skewness outside this range so for that we have applied NumPy mathematical function log transformation.

- Data Sources and their formats

  Dataset:

  1. The sample data is provided in .csv format.
  2. There are 1168 rows and 81 columns in the train dataset.
  3. There are 292 rows and 80 columns.
  4. 18 columns have missing values.
  5. Dataset contains continuous as well as categorical variable.
  6. We have used label encoder for encoding categorical data into numerical format.
  7. We have used standard scaler for pre-processing to bring features to common scale.

- Data Pre-processing Done

  The data pre-processing starts with collecting the data and end with communicating the results and basically it uses data mining technique which is used to transform the raw data in a useful and efficient format.

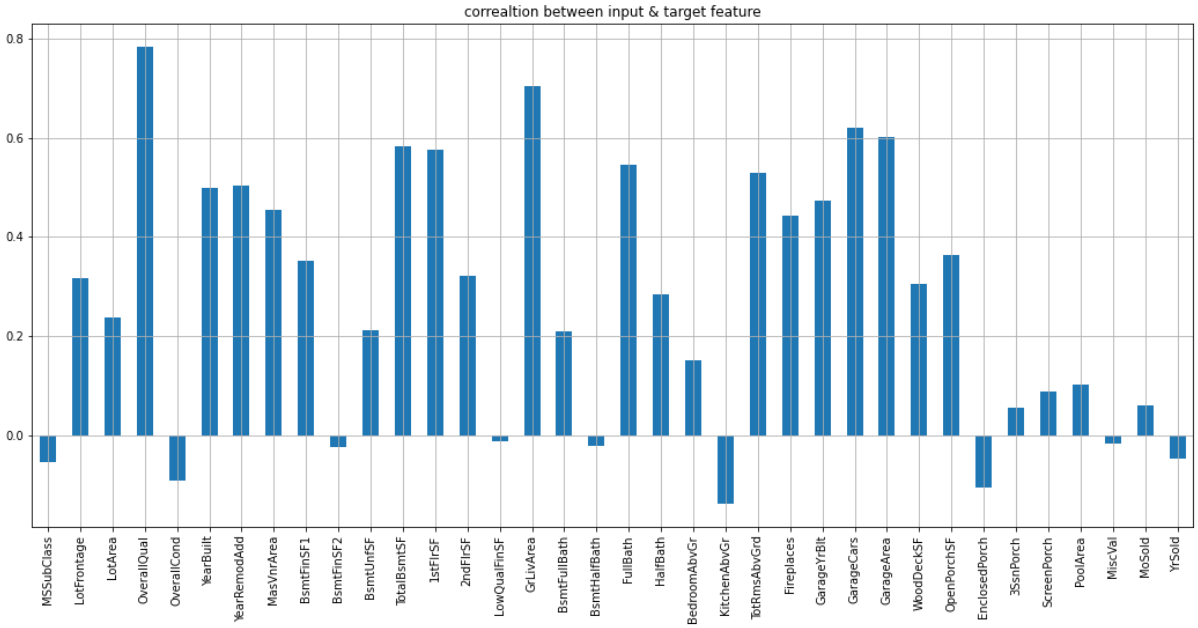
  This involves 4 steps: Cleaning, Formatting, Scaling, and Normalization.

  In the process of machine learning model, following these steps is very important as without these steps we might end up considering that 1% of results that are false.

  The steps performed in this project are:

  1. Missing values found in 18 columns by analysing the empty cells.
  2. Checked unique values in each column to explore dataset more deeply.
  3. The statistical summary and visualization techniques showed that there are outliers present in the data.
  4. The skewness was checked and we found that many features have skewed data. So we used log transformation to deal with this problem.
  5. Feature encoding was done using by observing patterns in the data. We applied different techniques for both continuous as well as categorical data.

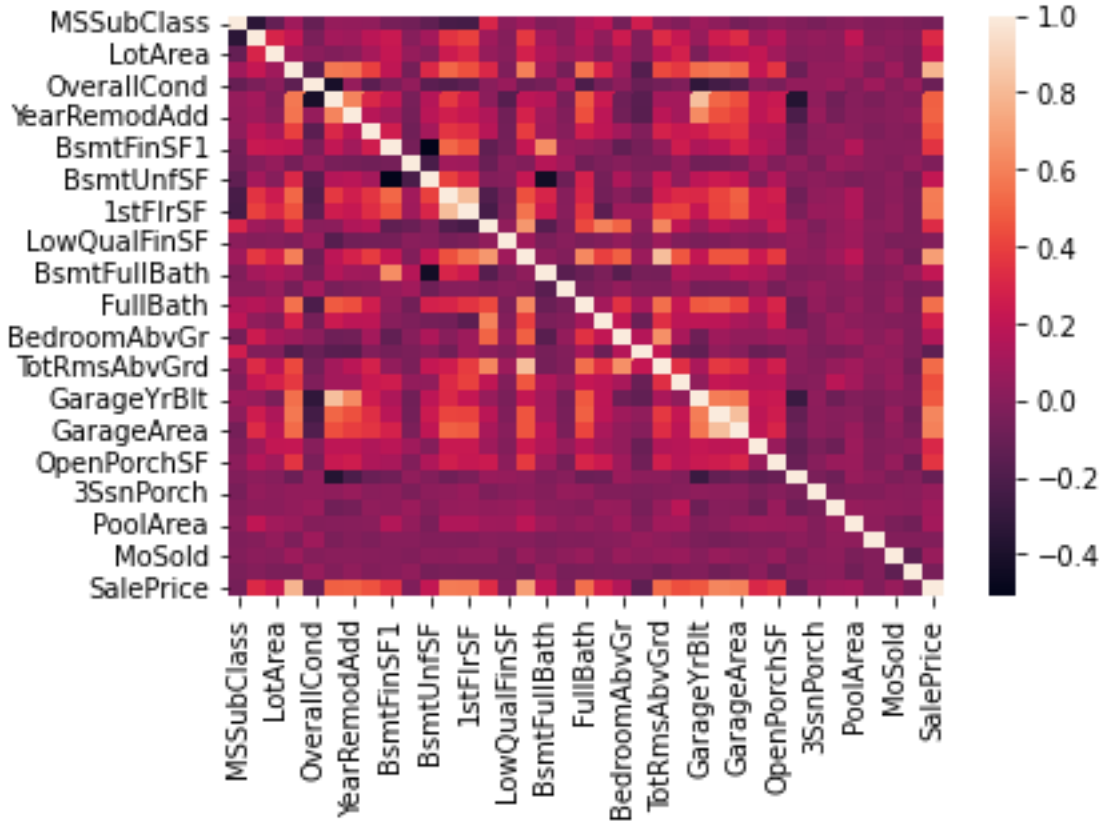
6. To prevent regression biases towards certain features, we dropped some of the irrelevant features.
7. We normalized the data so that all features are evaluated on the same scale.
8. We applied PCA on the dataset for dimension reduction.
9. We took following data cleaning steps:
    - GarageYrBlt is a continuous type feature and it has information about year of manufacture, so we dropped null values from this column.
    - The columns like PoolQC, Misc Feature, Alley and Fence do not have much importance in the dataset, so we have removed these columns.
    - The ID column only contains identification numbers and does not give any insight about the data.
    - The column 'Utilities' have only one unique value and so we removed it from the dataset.
10. Correlation is checked by finding the linear relationship between two variables. With this, we also analysed that there is presence of multi-collinearity in the features.
11. We have checked the relationship between the input variables and the output variable 'Sale Price' and it is shown as below:



OBSERVATION:

1. OverallQual column is most positively correlated with Sale Price column.
2. KitchenAbvGrd column is most negatively correlated with Sale Price column.

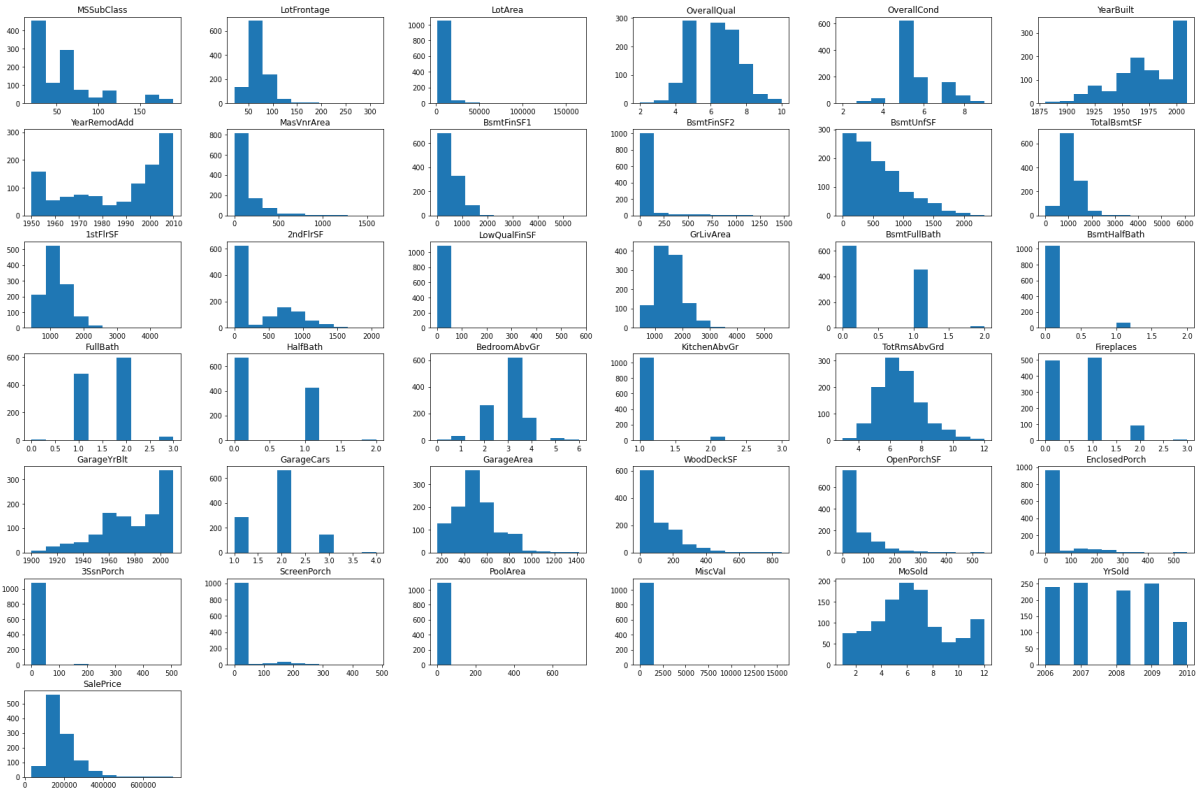
12. We checked for correlation between features. A correlation matrix is a table showing correlation coefficients between variables. The following is the correlation matrix:



OBSERVATION

a) SalePrice is highly positively correlated with the columns OverallQual, YearBuilt, YearRemodAdd, TotalBsmtSF, 1stFlrSF, GrLivArea, FullBath, TotRmsAbvGrd, GarageCars & GarageArea.
b) SalePrice is negatively correlated with OverallCond, KitchenAbvGr, Encloseporch & YrSold columns.
c) Multicollinearity is present between various columns so using Principal Component Analysis (PCA) will be a great choice.

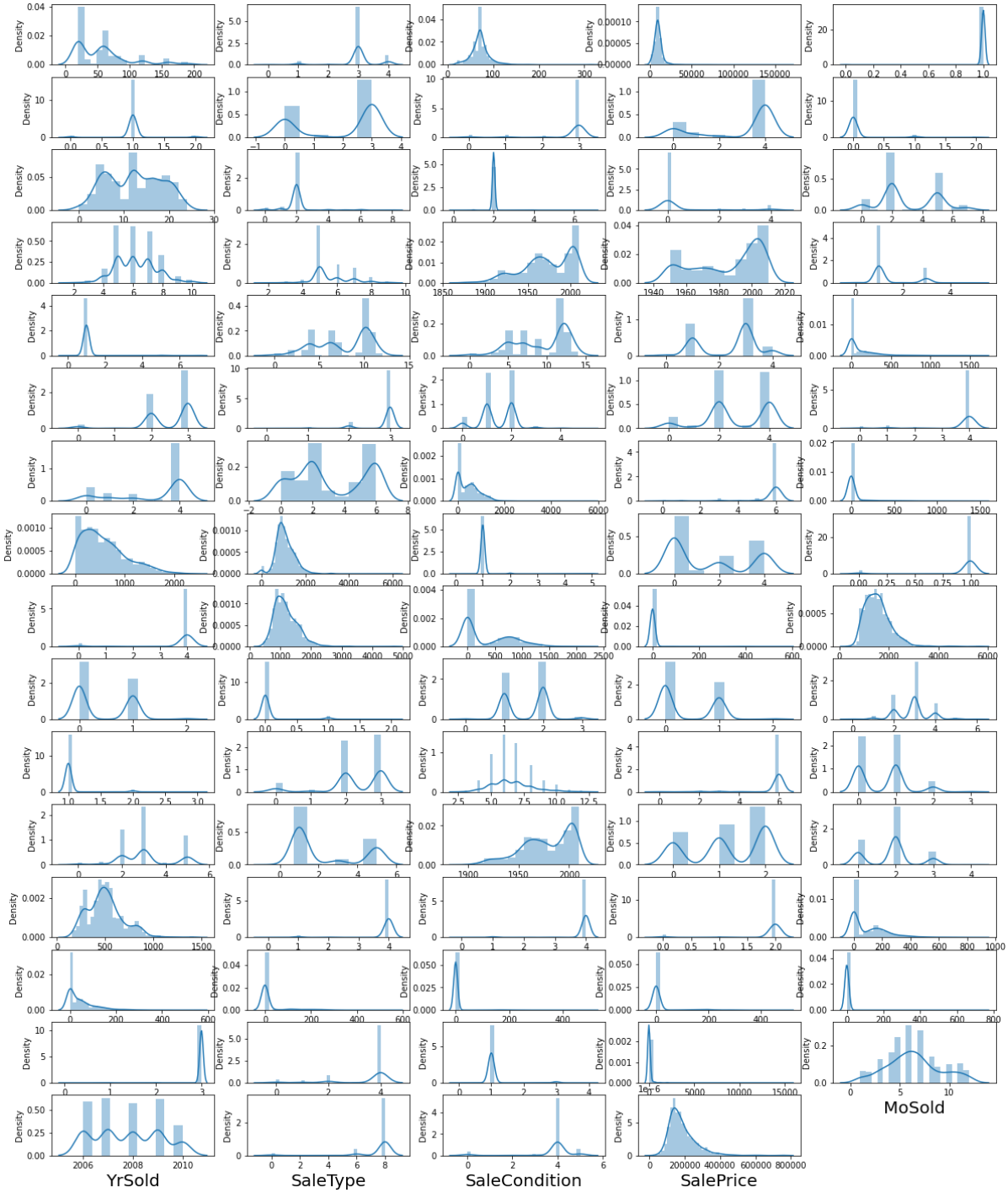
13. We then checked for the histogram to see the distribution of the data. The following is the histograms of all the variables:



OBSERVATION:

Presence of unusual values in above histograms & also distribution is not normal in some columns and these things denote the possibility of potential outliers.

14. We checked for skewness after encoding the categorical features into numerical one. Following is the chart depicting the distribution of all the features.



OBSERVATION:

Skewness is present in various columns. So, we have removed most of skewness using NumPy mathematical functions like log.

- Data Inputs- Logic- Output Relationships

OUTPUT VARIABLE:

In this project, the target variable is sale price. The shape of the target variable is (1104, 1). It is continuous in nature and so it is a regression based problem. Our goal is to predict the price of houses using available independent variables.

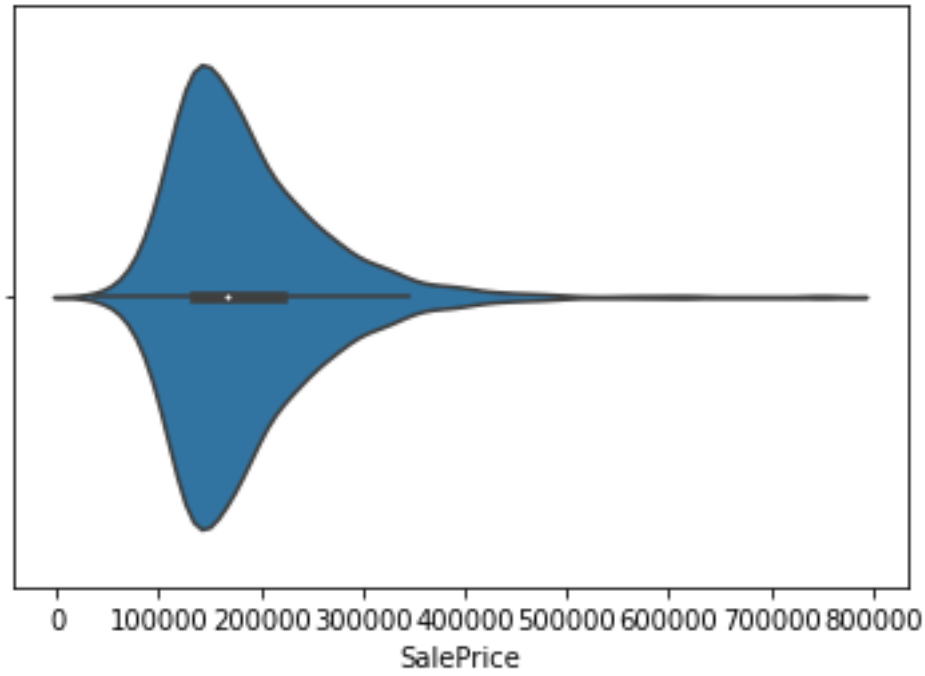
INPUT VARIABLES:

There are 80 features in the dataset which comprises of all the three data types i.e. float, int and object. After performing various techniques and data pre-processing methods, we are left with 45 features to use in order to predict the target variable.

DATA VISUALIZATION:

Visualizing output variable:

Violin plot for target variable and value counts are:

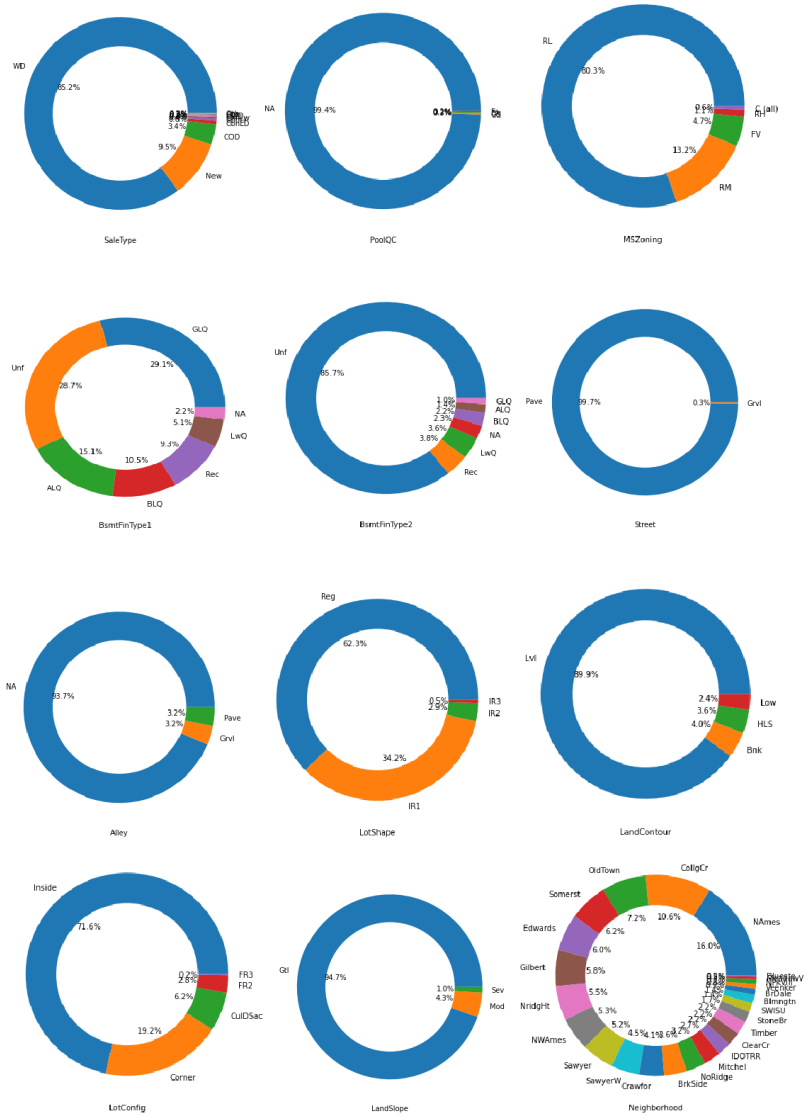


OBSERVATION:

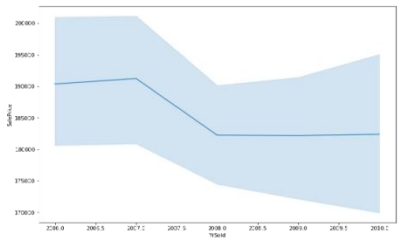
Sale price values are mostly in the range of 140000 and 230000.

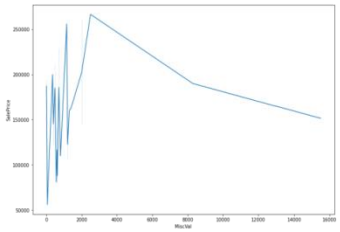
Visualizing input variable:

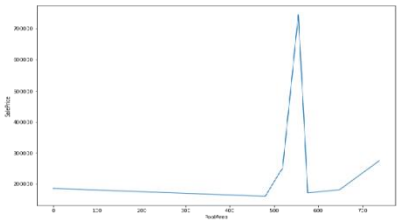
- Count of Categoical variables:

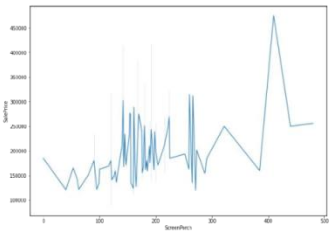


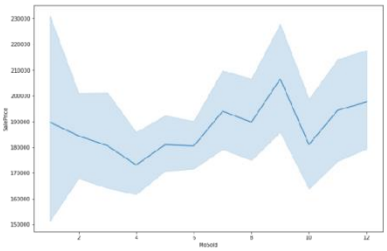
■ Line plot

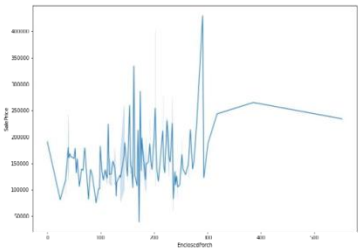

SalePrice
YrSold

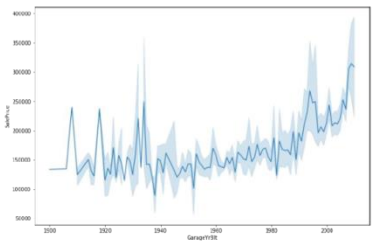

SalePrice
MiscVal

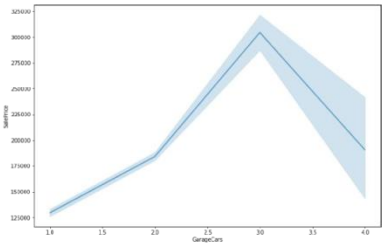

SalePrice
PoolArea


SalePrice
ScreenPorch

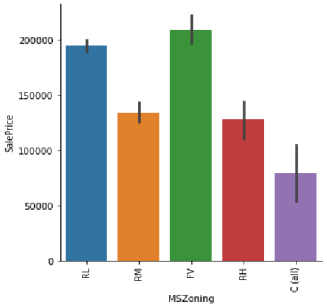

SalePrice
MoSold

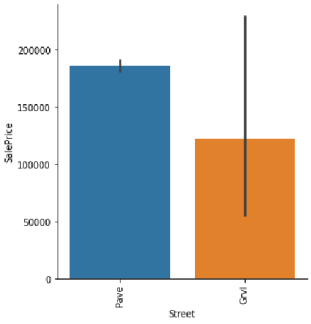

SalePrice
EnclosedPorch

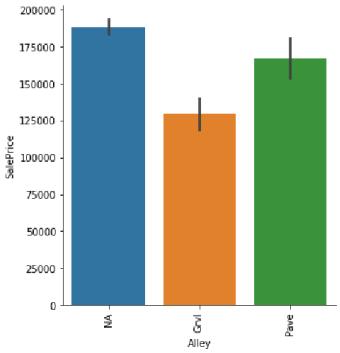

SalePrice
GarageYrBlt

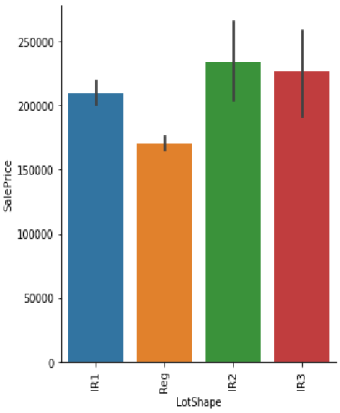

SalePrice
GarageCars

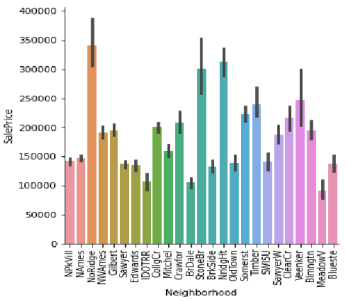
■ Bar plots

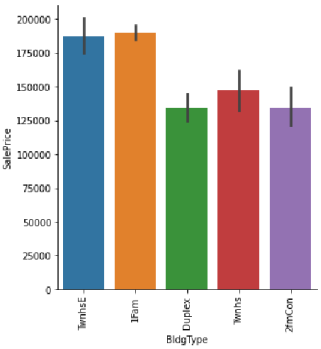

SalePrice
MSZoning
RL
RM
FV
RH
C (all)


SalePrice
Street
Pave
Grvl


SalePrice
Alley
NA
Grvl
Pave


SalePrice
LotShape
IR1
Reg
IR2
IR3


SalePrice
Neighborhood


SalePrice
BldgType
TwnhsE
1Fam
Duplex
Twnhs
2fmCon

- State the set of assumptions (if any) related to the problem under consideration
    - There is presence of multicollinearity between various columns. Thus, PCA is used.
    - We have removed columns with only one category or columns like ID as these are assumed to irrelevant in the process of analysis.

- Hardware and Software Requirements and Tools Used
    - Hardware: 4 GB RAM, Intel 13 processor
    - Software:
    - Anaconda(64 bit)
    - Jupytor notebook
    - Python
    - MS-Office
    - Google Chrome Web Browser
    - Import pandas as pd
    - import numpy as np
    - import seaborn as sns
    - import matplotlib.pyplot as plt
    -
    - from sklearn.preprocessing import StandardScaler
    - from sklearn.model_selection import train_test_split
    - from sklearn.neighbors import KNeighborsRegressor
    - from sklearn.ensemble import RandomForestRegressor
    - from sklearn.linear_model import LinearRegression
    - from sklearn.ensemble import GradientBoostingRegressor

- from sklearn.linear_model import Ridge
- from sklearn.preprocessing import StandardScaler
- from statsmodels.stats.outliers_influence import vari ance_inflation_factor
- from sklearn.metrics import r2_score, mean_square d_error
-
- import warnings
- warnings.filterwarnings('always')
- warnings.filterwarnings('ignore')

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)
  1. As per visualizations, Target Variable “Sale Price” is mostly between 140000 and 230000. Also, it is continuous in nature so we will use different regression algorithms to try and find the features that have the best explanation of the target variable.
  2. Explored input features and their values using pie chart.
  3. Checking missing values and handling them properly.
  4. Checking Summary Statistics to summarize set of observations as- central Tendency, dispersion, skewness, variance, range, deviation etc.
  5. Checking Correlation between target variable and input features using Correlation Matrix & correlation Heatmap using Seaborn.
  6. To check distribution & spread of data we used Histogram.
  7. Checked Scatter Plots between input & output feature for bivariate analysis.
  8. To check highly correlated values with Output variable Sale Price used Marker Plot.
  9. Divided input features into category type & continuous type features.

10. Checked Cat plot for category type & Line plot for continuous type features.
11. Used Scatter matrix for multivariate analysis.
12. Removed irrelevant columns which do not have much impact on dataset.
13. Used Label Encoding to encode categorical data in to numerical format.
14. Used Boxplot for summarizing variations & check outliers.
15. Tried to remove outliers using Z scores but data loss was nearly 60 %. Due to high amount of data loss, we eliminated the Z score implementation process.
16. Divided dataset into input and output sets to explore more briefly.
17. Used Distplot to check distribution of skewness and removed skewness using NumPy mathematical functions like log transformations.
18. Standardization is useful to speed up the learning algorithm and it rescales the features so that they will have the properties of the standard normal distribution with $\mu = 0$ and $\sigma = 1$. We have used Standard Scaler to standardize the data.
19. Multicollinearity is present between various columns so used Principal Component Analysis (PCA) for dimensionality reduction.
20. We will use Coefficient of determination(R2) score as our metric.
21. After Splitting data in to Training & Test Sets, checked scores at best random state after applying different regression algorithms.
22. Used Cross validation to check how accurately a predictive model will perform in practice.
23. To check error of forecasting model, we used Error Metric MSE.
24. To choose set of optimal hyperparameters for learning algorithm, we did Hyperparameter Tuning. We used Grid Search CV to find estimators/neighbours for learning algorithms.

25. Compared all algorithms on basis of scores, plots and errors & finalized the best model.
26. Implementing the best model, calculating scores/errors & also checking predicted values.
27. Saved final model using job lib.

- Testing of Identified Approaches (Algorithms)
  - Linear Regression : Linear regression is a linear approach to modelling the relationship between a scalar response and one or more explanatory variables. Linear Regression fits a linear model with coefficients w = (w1, …) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation
  - Random Forest Regressor : 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. A Random Forest operates by constructing several decision trees during training time and outputting the mean of the classes as the prediction of all the trees
  - Kneighbors Regressor(n_neighbors = 4) : Knearest neighbors is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure (e.g., distance functions). A simple implementation of KNN regression is to calculate the average of the numerical target of the K nearest neighbors

■ Gradient Boosting Regressor : In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

- Run and Evaluate selected models

  The models we used to calculate the r2 score and mse:

```
models = [LinearRegression(), RandomForestRegressor(),
          KNeighborsRegressor(n_neighbors = 4), GradientBoostingRegress
or()]
model_names = ['LR','RF', 'KNN','GB']

r2 = []
mse = []

for model in range(len(models)):
    clf = models[model]
    clf.fit(X_train, y_train)
    pred = clf.predict(X_test)
    r2.append(r2_score(y_test,pred))
    mse.append(mean_squared_error(y_test,pred))


models = {'Modelling Algorithm': model_names, 'r2 score': r2, 'MSE' : m
se}
models_df = pd.DataFrame(models)
models_df
```

RESULT:

	Modelling Algorithm	r2 score	MSE
0	LR	0.796679	1.719831e+09
1	RF	0.792607	1.754272e+09
2	KNN	0.662880	2.851595e+09
3	GB	0.816165	1.555002e+09

- Key Metrics for success in solving problem under consideration
    - R2 score: Coefficient of determination, denoted R2 or r2, is the proportion variation in the dependent variable that is predictable from the independent variables. It is a regression score function and best possible score is 1.0. We have calculated maximum & mean r2 score for models and will select model with best r2 score.
    - Cross Validation score: The result is:

Modeling algorithm	CV score
Linear Regression	0.759852
Random Forest	0.785777
KNearest Neighbors	0.693756
Gradient Boosting	0.812586

Hyperparameter Tuning:

Hyperparameter tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm. A hyperparameter is a parameter whose value is used to control the learning process. We used Grid Search CV to find estimators for learning algorithms and to find best parameters for implementation of final selected model.

- Interpretation of the Results

- The difference between the r2 score and cv score is minimum for the gradient boosting regressor.
- So, we will apply hyperparameter tuning on gradient boosting regressor.
- After applying grid search CV, we got 81.33% r2 score.

CONCLUSION

- Key Findings and Conclusions of the Study
  - By using machine learning algorithm, we predicted the housing prices.
  - We did data cleaning, handled missing values, removed skewness, performed PCA, and applied regression algorithm.
  - The hyperparameter tuning is applied on the best model i.e. gradient boosting.

- Learning Outcomes of the Study in respect of Data Science
  - Data cleaning is quite challenging task and also takes a lot of time in the entire process. While working on this particular project, we faced different issues but after detailed study, research and guidance, we were able to work and solve these issues and moved towards the direction of having clean data.
  - Using Data visualization, we can easily identify outliers, skewness, missing values & correlation etc. Also, we can identify the relation between target & other features using it. In this project we have used Matplotlib & Seaborn library for data visualization.

- Training & Test both Datasets have huge missing values so handled them very carefully.
- Gradient Boosting worked best in terms of r2 score and cross validation. That is why we used it as final model.

- Limitations of this work and Scope for Future Work
  - We could have used more algorithms to have a better r2 score. But due to time constraint, we were able to work only on 4 algorithms.
  - Though 81% is not a bad score, but we will try to work in the direction where we cross 90%.
  - More statistical features could have been used to deal with the problem of outliers and skewness

THANK YOU!