# HOUSE PRICE PREDICTION USING VARIOUS REGRESSION

**Submitted by:** **NEHA CHAND**

# ACKNOWLEDGMENT

The dataset is provided by a US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The company is looking at prospective properties to buy houses to enter the market. The data and the information are given below.

Data Description.txt: This contains the description of data.

train.csv: This contains the dataset on which you will be working upon

Housing Use case: This contains the problem statement and business

test.csv: Predict the output for these data with your best fit model.

# INTRODUCTION

- Business Problem Framing

We are required to model the price of houses with the available independent variables. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market.

- Conceptual Background of the Domain Problem

A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia.

The company is looking at prospective properties to buy houses to enter the market. We are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:

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

- ## Review of Literature

Houses are one of the necessary needs 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 modeling, Market mix modeling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. The company is looking at prospective properties to buy houses to enter the market. So we need to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not.

- ## Motivation for the Problem Undertaken

To buy a house or rent a house is very necessary and important of Each and every person life around the world and that's why housing and real estate markets are increases day by day and contribute a very handy help in increasing the world's economy. The objective behind this project is that to get the knowledge about this domain. That is to know how they work, what are the factors which are necessary for buying and renting the house. And how the real estate market work. It is a very large market and many companies work in this domain. Data science comes with a very important tool to solve the problems in the domain to help the market focusing and strategies for house sales and the purchases. To get the knowledge of the real estate market and come up with the solutions of the problem that is the main focusing of this project. Recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies.

# Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

In this project, as we analyses the dataset we get to know that:

- The data is not normally distributed as the data is not complete and some features were not included, which made the right or left skewed data somewhere.
- Hence we use a log function to make the feature Normally Distributed.
- And also we use the box cox method to remove some outliers which are very-very far from the normal distribution.
- To see the inter relation of features with the target column we use the correlation method, and then visualize with the help of Heatmap to see the correlation between features and with the target column.
- As the dataset is in the numerical and categorical form, and for the better understanding of the data, I first divided the numerical data into discrete and continuous features on the basis of less than 25 unique values are discrete features and other remaining numerical columns are continuous features.
- And then I done the Feature selection process where I convert the categorical column into the numerical with the help of one hot encoding and used the statistical libraries like skew for checking the skewness and norm for the normal distribution and box-cox for the outliers removal.
- Here I use the z-score with the threshold value of 3 to check the outliers present in the dataset.
- And finally the dataset get divided into x(without target column) and y(only target column present) for the model training.

- Data Sources and their formats

The data was provided by the US-based housing company named Surprise Housing which wants to enter in the Australian market. Thedata they provide was in the form of csv (Comma -separated values)
, also the provide the dataset into parts which are Train dataset for the data analyzing , model training and Test dataset to do the final testing after the model is trained and predict the output. Also the data description was given which is in the form of csv. And the Problem statement in the form pdf.

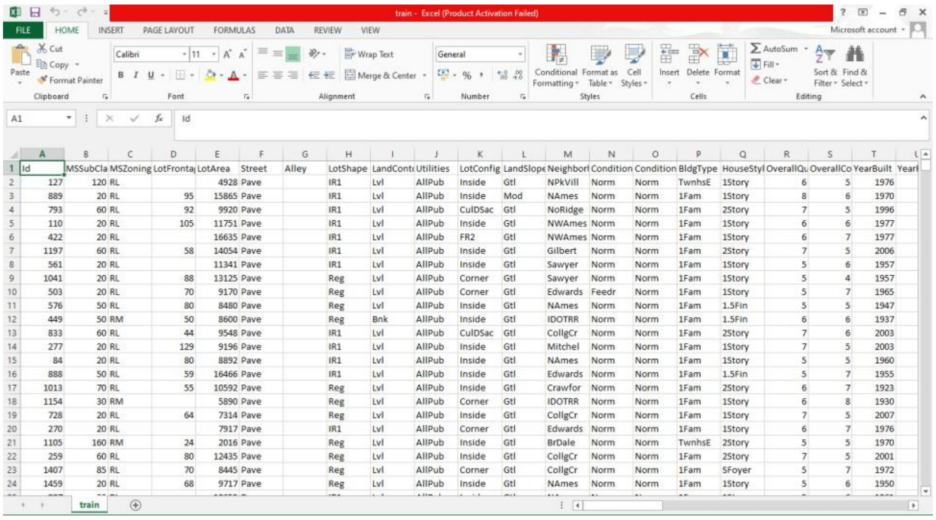
Data Description.txt: This contains the description of data.

train.csv: This contains the dataset on which you will be working upon

Housing Use case: This contains the problem statement and business goal.

test.csv: Predict the output for these data with your best fit model.

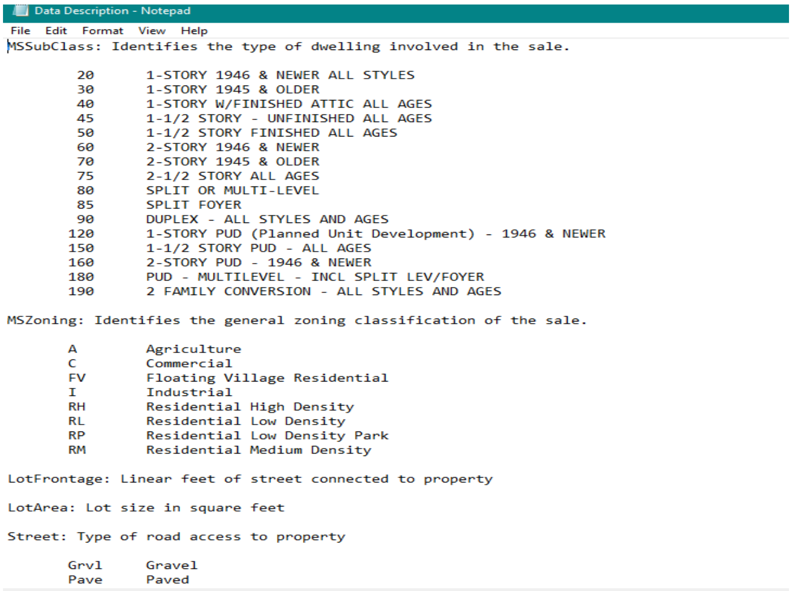
1. Train.csv:



| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Id | MSSubCla | MSZoning | LotFronta | LotArea | Street | Alley | LotShape | LandCont | Utilities | LotConfig | LandSlope | Neighbor | Condition | Condition | BldgType | HouseStyl | OverallQu | OverallCo | YearBuilt | Yearl |
| 2 | 127 | 120 | RL | | 4928 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | NPkVill | Norm | Norm | TwnhsE | 1Story | 6 | 5 | 1976 | |
| 3 | 889 | 20 | RL | 95 | 15865 | Pave | | IR1 | Lvl | AllPub | Inside | Mod | NAmes | Norm | Norm | 1Fam | 1Story | 8 | 6 | 1970 | |
| 4 | 793 | 60 | RL | 92 | 9920 | Pave | | IR1 | Lvl | AllPub | CulDSac | Gtl | NoRidge | Norm | Norm | 1Fam | 2Story | 7 | 5 | 1996 | |
| 5 | 110 | 20 | RL | 105 | 11751 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | NWAmes | Norm | Norm | 1Fam | 1Story | 6 | 6 | 1977 | |
| 6 | 422 | 20 | RL | | 16635 | Pave | | IR1 | Lvl | AllPub | FR2 | Gtl | NWAmes | Norm | Norm | 1Fam | 1Story | 6 | 7 | 1977 | |
| 7 | 1197 | 60 | RL | 58 | 14054 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | Gilbert | Norm | Norm | 1Fam | 2Story | 7 | 5 | 2006 | |
| 8 | 561 | 20 | RL | | 11341 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | Sawyer | Norm | Norm | 1Fam | 1Story | 5 | 6 | 1957 | |
| 9 | 1041 | 20 | RL | 88 | 13125 | Pave | | Reg | Lvl | AllPub | Corner | Gtl | Sawyer | Norm | Norm | 1Fam | 1Story | 5 | 4 | 1957 | |
| 10 | 503 | 20 | RL | 70 | 9170 | Pave | | Reg | Lvl | AllPub | Corner | Gtl | Edwards | Feedr | Norm | 1Fam | 1Story | 5 | 7 | 1965 | |
| 11 | 576 | 50 | RL | 80 | 8480 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | NAmes | Norm | Norm | 1Fam | 1.5Fin | 5 | 5 | 1947 | |
| 12 | 449 | 50 | RM | 50 | 8600 | Pave | | Reg | Bnk | AllPub | Inside | Gtl | IDOTRR | Norm | Norm | 1Fam | 1.5Fin | 6 | 6 | 1937 | |
| 13 | 833 | 60 | RL | 44 | 9548 | Pave | | IR1 | Lvl | AllPub | CulDSac | Gtl | CollgCr | Norm | Norm | 1Fam | 2Story | 7 | 6 | 2003 | |
| 14 | 277 | 20 | RL | 129 | 9196 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | Mitchel | Norm | Norm | 1Fam | 1Story | 7 | 5 | 2003 | |
| 15 | 84 | 20 | RL | 80 | 8892 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | NAmes | Norm | Norm | 1Fam | 1Story | 5 | 5 | 1960 | |
| 16 | 888 | 50 | RL | 59 | 16466 | Pave | | IR1 | Lvl | AllPub | Inside | Gtl | Edwards | Norm | Norm | 1Fam | 1.5Fin | 5 | 7 | 1955 | |
| 17 | 1013 | 70 | RL | 55 | 10592 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | Crawfor | Norm | Norm | 1Fam | 2Story | 6 | 7 | 1923 | |
| 18 | 1154 | 30 | RM | | 5890 | Pave | | Reg | Lvl | AllPub | Corner | Gtl | IDOTRR | Norm | Norm | 1Fam | 1Story | 6 | 8 | 1930 | |
| 19 | 728 | 20 | RL | 64 | 7314 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | CollgCr | Norm | Norm | 1Fam | 1Story | 7 | 5 | 2007 | |
| 20 | 270 | 20 | RL | | 7917 | Pave | | IR1 | Lvl | AllPub | Corner | Gtl | Edwards | Norm | Norm | 1Fam | 1Story | 6 | 7 | 1976 | |
| 21 | 1105 | 160 | RM | 24 | 2016 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | BrDale | Norm | Norm | TwnhsE | 2Story | 5 | 5 | 1970 | |
| 22 | 259 | 60 | RL | 80 | 12435 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | CollgCr | Norm | Norm | 1Fam | 2Story | 7 | 5 | 2001 | |
| 23 | 1407 | 85 | RL | 70 | 8445 | Pave | | Reg | Lvl | AllPub | Corner | Gtl | CollgCr | Norm | Norm | 1Fam | SFoyer | 5 | 7 | 1972 | |
| 24 | 1459 | 20 | RL | 68 | 9717 | Pave | | Reg | Lvl | AllPub | Inside | Gtl | NAmes | Norm | Norm | 1Fam | 1Story | 5 | 6 | 1950 | |

Above is the screenshot of the train dataset which is in the form of csv.

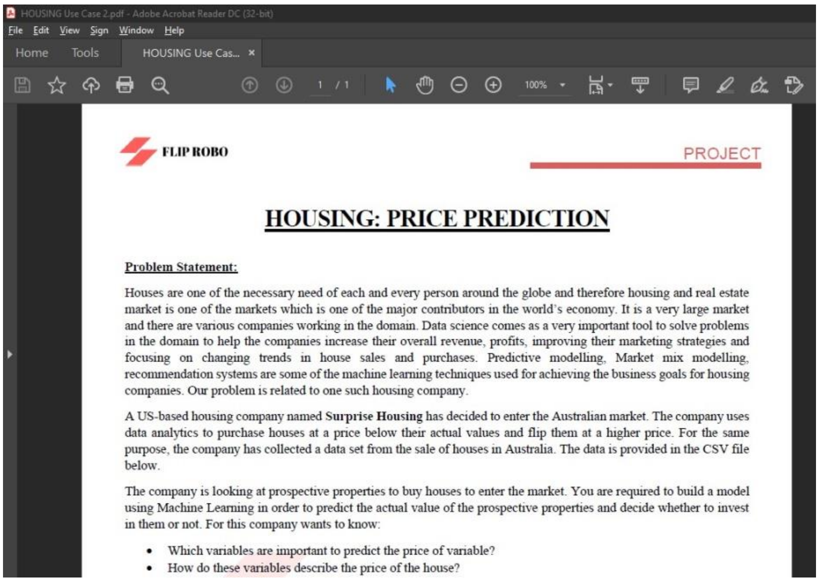
## 2. Data Description.txt:



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

        20      1-STORY 1946 & NEWER ALL STYLES
        30      1-STORY 1945 & OLDER
        40      1-STORY W/FINISHED ATTIC ALL AGES
        45      1-1/2 STORY - UNFINISHED ALL AGES
        50      1-1/2 STORY FINISHED ALL AGES
        60      2-STORY 1946 & NEWER
        70      2-STORY 1945 & OLDER
        75      2-1/2 STORY ALL AGES
        80      SPLIT OR MULTI-LEVEL
        85      SPLIT FOYER
        90      DUPLEX - ALL STYLES AND AGES
       120      1-STORY PUD (Planned Unit Development) - 1946 & NEWER
       150      1-1/2 STORY PUD - ALL AGES
       160      2-STORY PUD - 1946 & NEWER
       180      PUD - MULTILEVEL - INCL SPLIT LEV/FOYER
       190      2 FAMILY CONVERSION - ALL STYLES AND AGES

MSZoning: Identifies the general zoning classification of the sale.

       A        Agriculture
       C        Commercial
       FV       Floating Village Residential
       I        Industrial
       RH       Residential High Density
       RL       Residential Low Density
       RP       Residential Low Density Park
       RM       Residential Medium Density

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access to property

       Grvl     Gravel
       Pave     Paved
```

Above image is the screenshot of the data description which is the format of txt file.

## 3. Housing Use case.pdf :


HOUSING Use Case 2.pdf - Adobe Acrobat Reader DC (32-bit)
File Edit View Sign Window Help
Home
Tools
HOUSING Use Cas... ×
1 / 1
100%
FLIP ROBO
PROJECT

HOUSING: PRICE PREDICTION

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. The data is provided in the CSV file below.

The company is looking at prospective properties to buy houses to enter the market. You are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:

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

This fig is the screenshot of problem statement which is given in the form of pdf.

- Data Preprocessing Done

In the Data Preprocessing, at first we drop the unnecessary variables or the features which are not related to the further processes.

After that, I analyse the Temporal Date time Variables, and then check whether there is a relation between year the house is sold and the sales price. Which gives us that the "Other year field" value near to "year sold" then price is high(Means it’s a new house), if difference is large(If its 140 year old house) then price is less and values distributed with all different prices for same year difference in starting.

After that for temporal handling I replace the null value with median. As we have four Date Time variables i.e. ['YearBuilt', 'YearRemodAdd', 'GarageYrBlt', 'YrSold']. I store them in a Variable which is Year feature for better understanding. Then I convert the Temporal variables like year field and converting it to number by subtracting each year by year sold.

After that I fill the All the NA values in the features (having object datatype) with the “None”.

Few Missing values have the valid value:

- PoolQC : data description says NA means "No Pool". That make sense, given the huge ratio of missing value (+99%) and majority of houses have no Pool at all in general.
- MiscFeature : data description says NA means "no misc feature"
- Alley : data description says NA means "no alley access"
- Fence : data description says NA means "no fence"
- FireplaceQu : data description says NA means "no fireplace"

- GarageType, GarageFinish, GarageQual and GarageCond : Replacing missing data with None/Missing value
- GarageYrBlt, GarageArea and GarageCars : Replacing missing data with 0 (Since No garage = no cars in such garage.)
- BsmtFinSF1, BsmtFinSF2, BsmtUnfSF, TotalBsmtSF, BsmtFullBath and BsmtHalfBath : missing values are likely zero for having no basement
- BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1 and BsmtFinType2 : For all these categorical basement-related features, NaN means that there is no basement.
- MasVnrArea and MasVnrType : NA most likely means no masonry veneer for these houses. We can fill 0 for the area and None for the type.
- MSZoning (The general zoning classification) : 'RL' is by far the most common value. So we can fill in missing values with 'RL'.

Then we do the One hot encoding and convert the categorical features into the numerical features and then we remove the skewness from the data set and after that we remove some outliers also. So then we splitting our dataset into two variables and then do model training and prediction.

## • Data Inputs- Logic- Output Relationships

By the visualization of the data we see the relationship between the features and also with the correlation tells about the all features related to the Target variable. By the correlation method we can see the relationship and effects of features on the target variable. And also we see the correlation between the features.

By correlation we came to see that some features(Overall quality and Above grade living area) are positively highly correlated with the target column(Sale Price). That means if the value of dependent feature increases, Sale Price also increases.

- ## State the set of assumptions (if any) related to the problem under consideration

  Here the assumptions I take were :

  - I first store the four DateTime variable column into One variable and after that I subtract the year it build from the Year it sold. Which give a proper outcome.
  - Here Also first I split the numerical features into two category first is Discrete and the second is Continuous, Discrete features are those which have unique values less than 25 and above than that were continuous.
  - Also there were na values were present in the features which means there is not that particular thing available i.e if there is ‘no pool’ available in the property it shows as NA , so I fill that value with the None, which makes the data lot more easier to understand.
  - I use the Log transformation method to solve the problem where the data is not normally distributed and its make it normally distributed.

- ## Hardware and Software Requirements and Tools Used

  The Project is done on the Window 10, here I use the Software Anaconda platform (Python 3.8.5 64 bit) and the code is written on the Jupyter Notebook where I run different python libraries for the better understanding. The libraries were:

  - Pandas – Pandas library is used for the uploading the file and then to create the dataframe from that file and manipulation of dataframe  and use some in-built operations .

  - Numpy – Numpy is used for the mathematical operations in the dataframe. Here Numpy is used to calculate the mean , median, mode and also the various mathematical ops were used in the dataset, which helps in the better understanding of the different features.

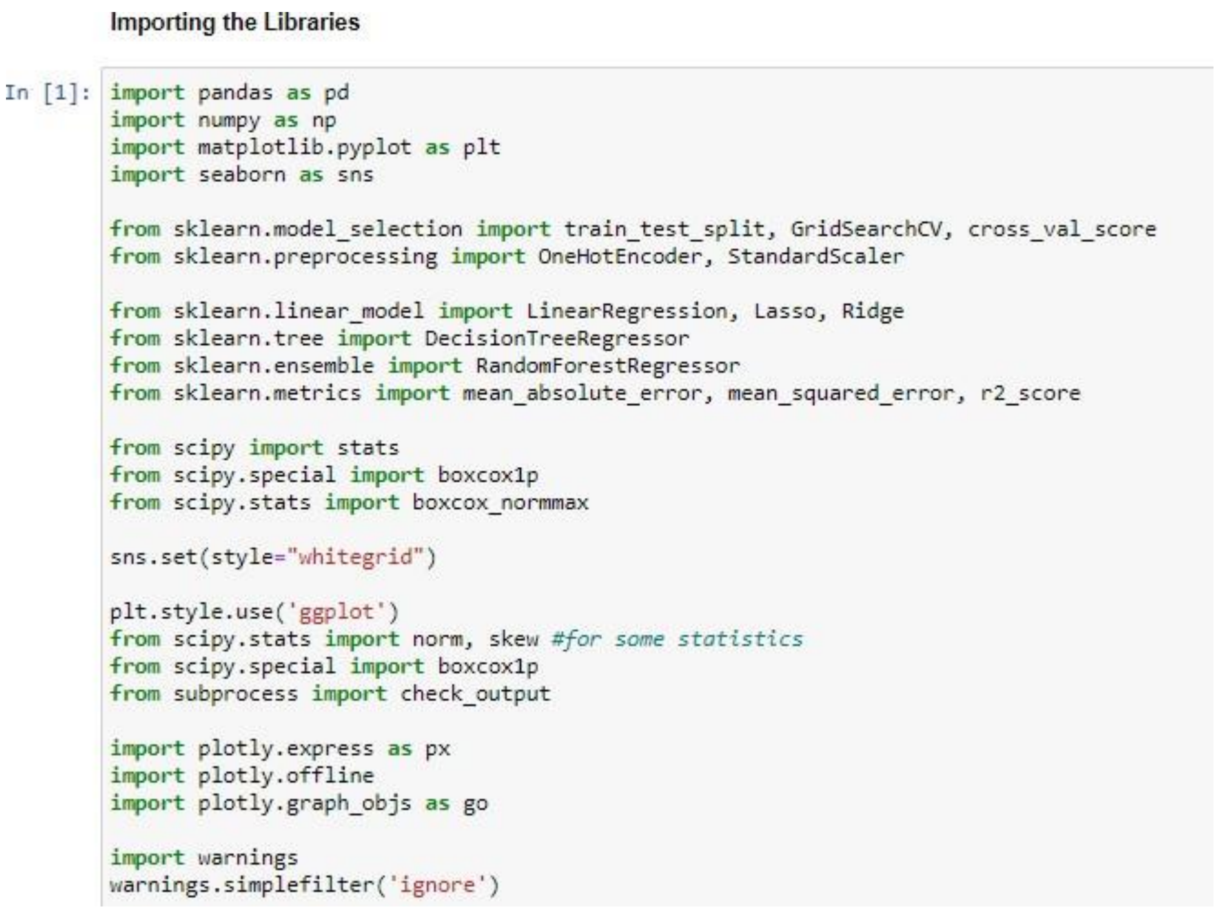
- Matplotlib – Matplotlib.pyplot library is used for the visualization for the dataset. And it helps to understand the data with help of graphs . And also it helps to label the graph , x-axis or y-axis. It also helps to select the size of the graph.

- Seaborn – Seaborn again is used for the visualization and it helps us to visualize the data more efficiently and clearly and understand it properly . Here I used the Heatmap which helps to understand to see the correlation between features and also Here I use the various graphs like histogram, Bargraph, Scatterplots.

- Plotly –The Plotly Python library is an interactive open-source library. This can be a very helpful tool for data visualization and understanding the data simply and easily. Plotly graph objects are a high-level interface to Plotly which are easy to use. It can plot various types of graphs and charts like scatter plots, line charts, bar charts, box plots, histograms, pie charts, etc. Here it is used to check the missing values % and represent it graphically. The bar graph is used for understandable and meaningful.

- Scipy – This library is used for the statistical formula and here I used it for calculate the skew for the checking of skewness and visualize it graphically also, norm is used for the normalization and here I used with the visualization for the better understanding of the data and also the boxcox1p is used for the removal of outliers.

- Sci-kit learn – Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy. Scikit-learn library is very important in the machine learning process. Here I use the library for model selection such as Linear regression , Lasso, Ridge , DecisionTree , some ensemble techniques such as RandomForest Regressor, metrices like Mean absolute error,root mean squared error , r2score, Preproccessing tools like Onehot encoder, StandardScaler and model selection like GridsearchCV , train test split and crossvalidation value.



Importing the Libraries

In [1]:

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

from sklearn.model_selection import train_test_split, GridSearchCV, cross_val_score
from sklearn.preprocessing import OneHotEncoder, StandardScaler

from sklearn.linear_model import LinearRegression, Lasso, Ridge
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

from scipy import stats
from scipy.special import boxcox1p
from scipy.stats import boxcox_normmax

sns.set(style="whitegrid")

plt.style.use('ggplot')
from scipy.stats import norm, skew #for some statistics
from scipy.special import boxcox1p
from subprocess import check_output

import plotly.express as px
import plotly.offline
import plotly.graph_objs as go

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

# Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  Here we first check the data is supervised or unsupervised by checking the target column and then when we see the output it shows it's a regression problem.

  - First, I import the necessary libraries, such as pandas , numpy , seaborn , matplotlib, scipy and sklearn.
  - After that with the help of pandas I create the dataframe from the given set of data.
  - Then I check the information the dataframe which shows that 1168 rows and 81 features and also the datatype float64(3), int64(35), object(43) were present.
  - Then I check the missing values in the dataframe and calculate the missing value % and visualize it with the help of plotly.
  - This gives that Pool quality and Misc. features have the highest number of missing values.
  - Then I check the correlation of features and found that Overall Quality and Grade above living are positively correlated with the Sales Price that means if OverallQuality increases SalePrice increase.
  - After that I groupby the missing value containing features and the target column to see the relation between them which tell about the null and not value present in the dataset graphically.
  - As Our dataset is containing both Object datatype and Integer, float datatype. And for the better understanding of the data, split the dataset into 2 category : Numerical and Categorical features.
  - The numerical features were further splitted into the discrete features on the basis of columns which have unique

value less than 25 and other than that were Continuous features.

- ➢ As four columns were Years , so I store that into a single variable.
- ➢ Barplot were used to represent the discrete features for the better understanding of dataset.
- ➢ Histogram were used to represent the continuous features and it shows that some features were not normally distributed i.e. skewness is present.
- ➢ Then I use the log transformation method for reducing the skewness and visualize with the histogram.
- ➢ After that I check the outliers with the help of zscore and boxplot for the graphical representation and remove them with the help of boxcox.
- ➢ After that I fill the missing values with the None.
- ➢ Then I check the Target column, which shows that column is not normally distributed. And with the help of log function I remove the skewness and then store the value in the dataset.
- ➢ After that I check the GrlivArea and with the help of boxcox I remove the right skewness from that column and make it normally distributed.
- ➢ After that I used the Year column and subtract the yearbuild from the year sold. So this represent the number of years.
- ➢ Then I use the Onehot Encoder and convert the categorical columns into the numerical columns.
- ➢ After that I remove the skewness by using boxcox_normax.
- ➢ And then split the dataset in x and y where x contains all the dependent columns and y contains only target column.
- ➢ After that the model training is done and select the best model which have less RMSE and used for the prediction.
- ➢ Random forest regressor is then used for the testing and for the prediction.

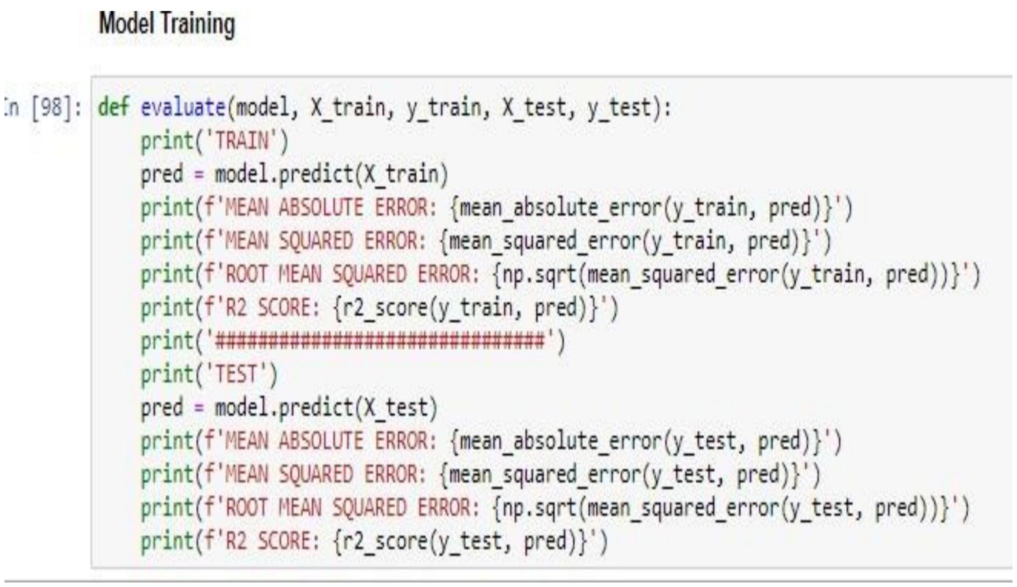
- Testing of Identified Approaches (Algorithms)

Here we use the following Algorithms used for training & Testing:

1. Linear Regression
2. Lasso
3. Ridge
4. Decision Tree Regressor
5. Random forest Regressor

- Run and Evaluate selected models

The algorithms used were :



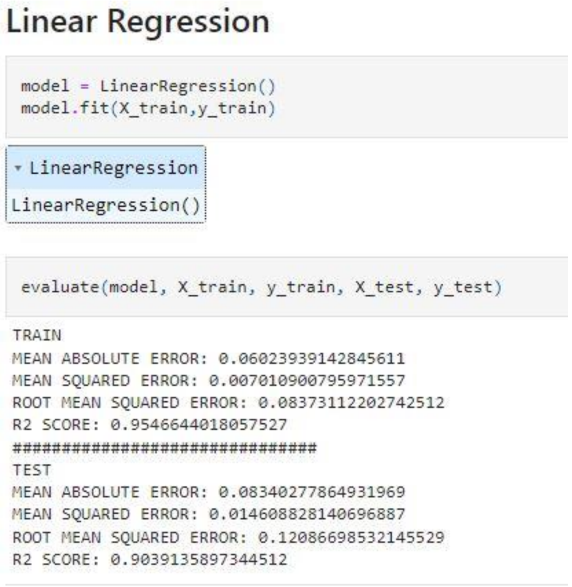
Model Training

```
In [98]: def evaluate(model, X_train, y_train, X_test, y_test):
             print('TRAIN')
             pred = model.predict(X_train)
             print(f'MEAN ABSOLUTE ERROR: {mean_absolute_error(y_train, pred)}')
             print(f'MEAN SQUARED ERROR: {mean_squared_error(y_train, pred)}')
             print(f'ROOT MEAN SQUARED ERROR: {np.sqrt(mean_squared_error(y_train, pred))}')
             print(f'R2 SCORE: {r2_score(y_train, pred)}')
             print('###############################')
             print('TEST')
             pred = model.predict(X_test)
             print(f'MEAN ABSOLUTE ERROR: {mean_absolute_error(y_test, pred)}')
             print(f'MEAN SQUARED ERROR: {mean_squared_error(y_test, pred)}')
             print(f'ROOT MEAN SQUARED ERROR: {np.sqrt(mean_squared_error(y_test, pred))}')
             print(f'R2 SCORE: {r2_score(y_test, pred)}')
```

Above is the function used and after that the model were run on it.

1. **Linear Regression :**

   Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.



## Linear Regression

```
model = LinearRegression()
model.fit(X_train,y_train)
```

```
evaluate(model, X_train, y_train, X_test, y_test)
```

```
TRAIN
MEAN ABSOLUTE ERROR: 0.06023939142845611
MEAN SQUARED ERROR: 0.007010900795971557
ROOT MEAN SQUARED ERROR: 0.08373112202742512
R2 SCORE: 0.9546644018057527
###############################
TEST
MEAN ABSOLUTE ERROR: 0.08340277864931969
MEAN SQUARED ERROR: 0.014608828140696887
ROOT MEAN SQUARED ERROR: 0.12086698532145529
R2 SCORE: 0.9039135897344512
```

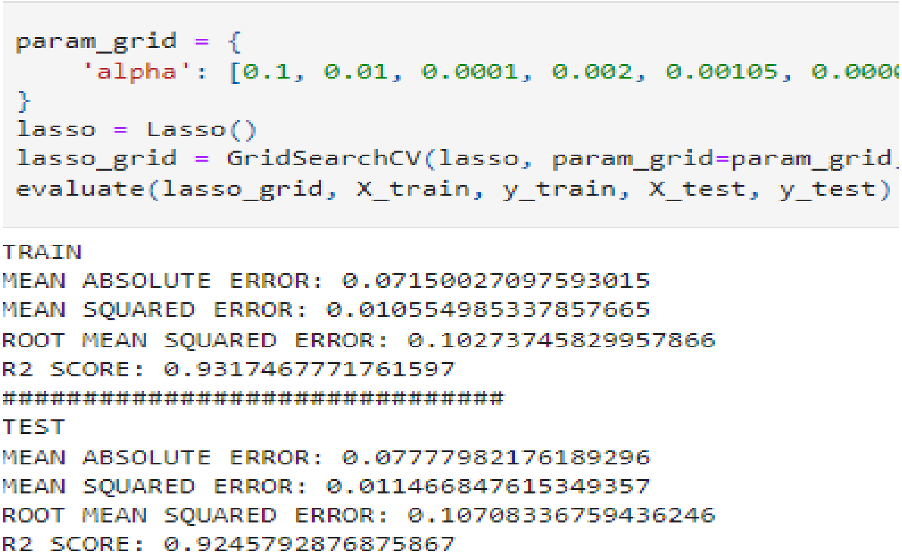
## 2. Lasso :

Lasso regression is a regularization technique. It is used over regression methods for a more accurate prediction. This model uses shrinkage. Shrinkage is where data values are shrunk towards a central point as the mean.

The lasso procedure encourages simple, sparse models (i.e. models with fewer parameters). This particular type of regression is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination.

Lasso Regression uses L1 regularization.

It is used when we have more number of features because it automatically performs feature selection.

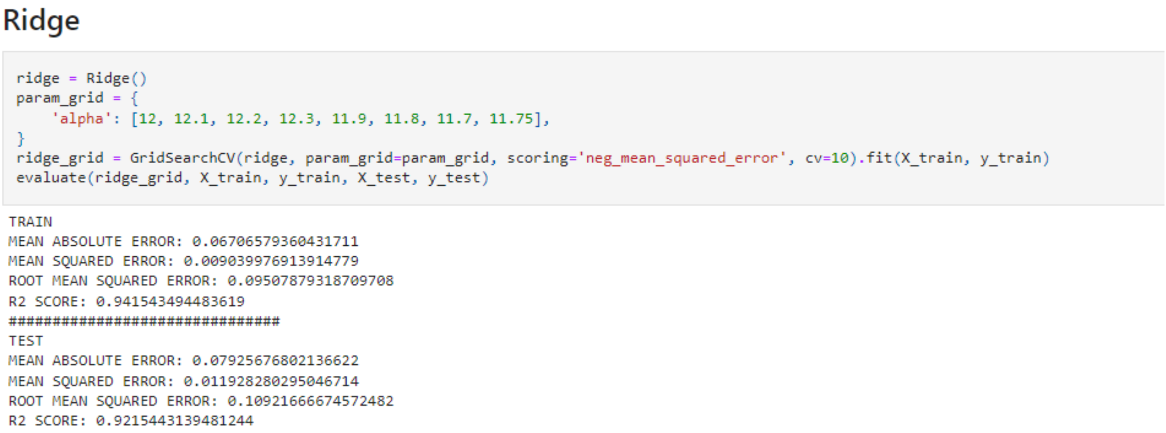


```
param_grid = {
    'alpha': [0.1, 0.01, 0.0001, 0.002, 0.00105, 0.000
}
lasso = Lasso()
lasso_grid = GridSearchCV(lasso, param_grid=param_grid
evaluate(lasso_grid, X_train, y_train, X_test, y_test)
```

```
TRAIN
MEAN ABSOLUTE ERROR: 0.07150027097593015
MEAN SQUARED ERROR: 0.010554985337857665
ROOT MEAN SQUARED ERROR: 0.10273745829957866
R2 SCORE: 0.9317467771761597
################################
TEST
MEAN ABSOLUTE ERROR: 0.07777982176189296
MEAN SQUARED ERROR: 0.011466847615349357
ROOT MEAN SQUARED ERROR: 0.10708336759436246
R2 SCORE: 0.9245792876875867
```

### 3. Ridge:

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.



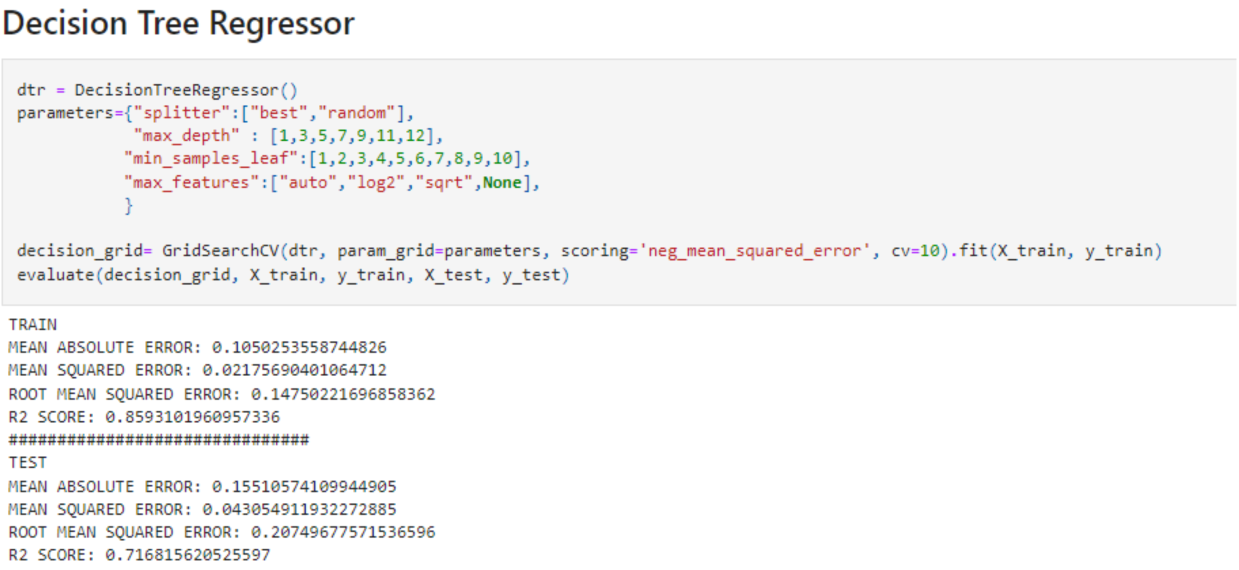
Ridge

```
ridge = Ridge()
param_grid = {
    'alpha': [12, 12.1, 12.2, 12.3, 11.9, 11.8, 11.7, 11.75],
}
ridge_grid = GridSearchCV(ridge, param_grid=param_grid, scoring='neg_mean_squared_error', cv=10).fit(X_train, y_train)
evaluate(ridge_grid, X_train, y_train, X_test, y_test)
```

```
TRAIN
MEAN ABSOLUTE ERROR: 0.06706579360431711
MEAN SQUARED ERROR: 0.009039976913914779
ROOT MEAN SQUARED ERROR: 0.09507879318709708
R2 SCORE: 0.941543494483619
################################
TEST
MEAN ABSOLUTE ERROR: 0.07925676802136622
MEAN SQUARED ERROR: 0.011928280295046714
ROOT MEAN SQUARED ERROR: 0.10921666674572482
R2 SCORE: 0.9215443139481244
```

### 4. Decision Tree Regressor:

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. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy), each representing values for the attribute tested. Leaf node (e.g., Hours Played) represents a decision on the numerical target. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.



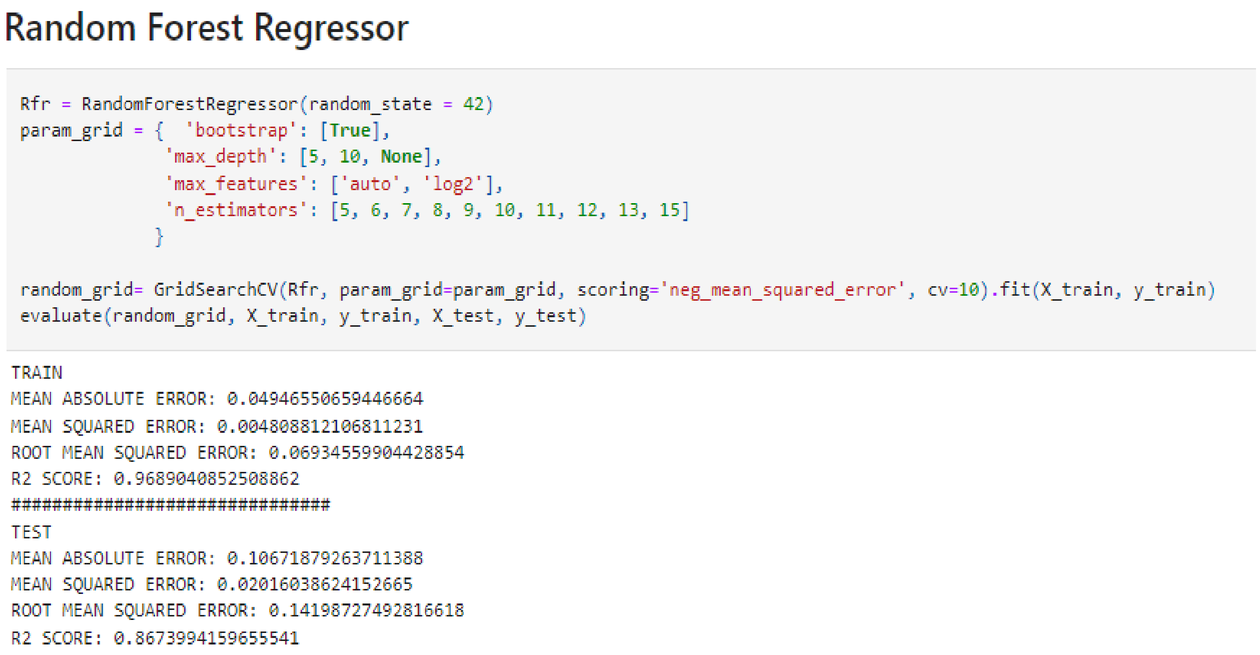
## Decision Tree Regressor

```
dtr = DecisionTreeRegressor()
parameters={"splitter":["best","random"],
            "max_depth" : [1,3,5,7,9,11,12],
           "min_samples_leaf":[1,2,3,4,5,6,7,8,9,10],
           "max_features":["auto","log2","sqrt",None],
           }

decision_grid= GridSearchCV(dtr, param_grid=parameters, scoring='neg_mean_squared_error', cv=10).fit(X_train, y_train)
evaluate(decision_grid, X_train, y_train, X_test, y_test)
```

```
TRAIN
MEAN ABSOLUTE ERROR: 0.1050253558744826
MEAN SQUARED ERROR: 0.02175690401064712
ROOT MEAN SQUARED ERROR: 0.14750221696858362
R2 SCORE: 0.8593101960957336
###############################
TEST
MEAN ABSOLUTE ERROR: 0.15510574109944905
MEAN SQUARED ERROR: 0.043054911932272885
ROOT MEAN SQUARED ERROR: 0.20749677571536596
R2 SCORE: 0.716815620525597
```

### 5. Random Forest Regressor:

Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. 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.



## Random Forest Regressor

```
Rfr = RandomForestRegressor(random_state = 42)
param_grid = {  'bootstrap': [True],
              'max_depth': [5, 10, None],
              'max_features': ['auto', 'log2'],
              'n_estimators': [5, 6, 7, 8, 9, 10, 11, 12, 13, 15]
             }

random_grid= GridSearchCV(Rfr, param_grid=param_grid, scoring='neg_mean_squared_error', cv=10).fit(X_train, y_train)
evaluate(random_grid, X_train, y_train, X_test, y_test)
```

```
TRAIN
MEAN ABSOLUTE ERROR: 0.04946550659446664
MEAN SQUARED ERROR: 0.004808812106811231
ROOT MEAN SQUARED ERROR: 0.06934559904428854
R2 SCORE: 0.9689040852508862
###############################
TEST
MEAN ABSOLUTE ERROR: 0.10671879263711388
MEAN SQUARED ERROR: 0.02016038624152665
ROOT MEAN SQUARED ERROR: 0.14198727492816618
R2 SCORE: 0.8673994159655541
```

- Key Metrics for success in solving problem under consideration

  The key metrices used  along with the project are :

1. **Mean absolute error : (MAE)** represents the difference between the original and predicted values extracted by averaged the absolute difference over the data set.

2. **Mean squared error**: (**MSE**) represents the difference between the original and predicted values extracted by squared the average difference over the data set.

3. **Root Mean squared error**: (**RMSE**) is the error rate by the square root of MSE.

4. **R2 score : R-squared (Coefficient of determination)** represents

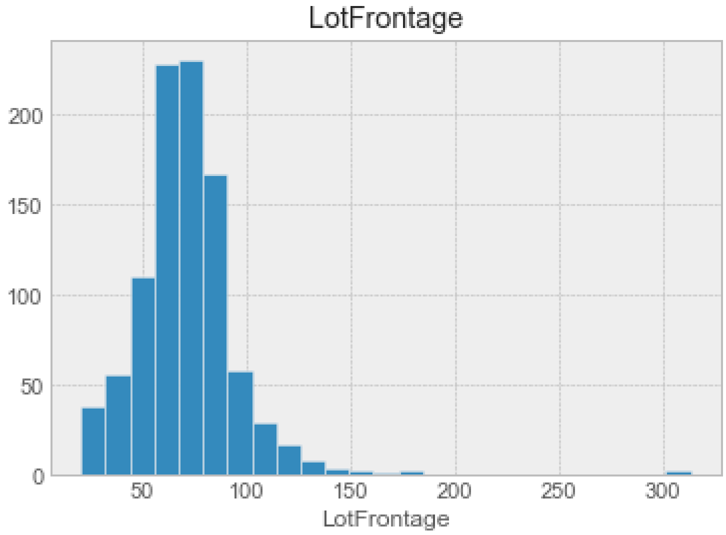
the coefficient of how well the values fit compared to the original values. The value from 0 to 1 interpreted as percentages.The higher the value is, the better the model is.

- Visualizations

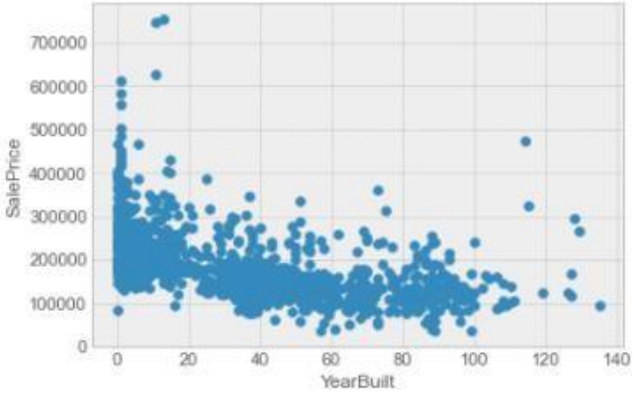
If different platforms were used, mention that as well.

Here we use the following graph:

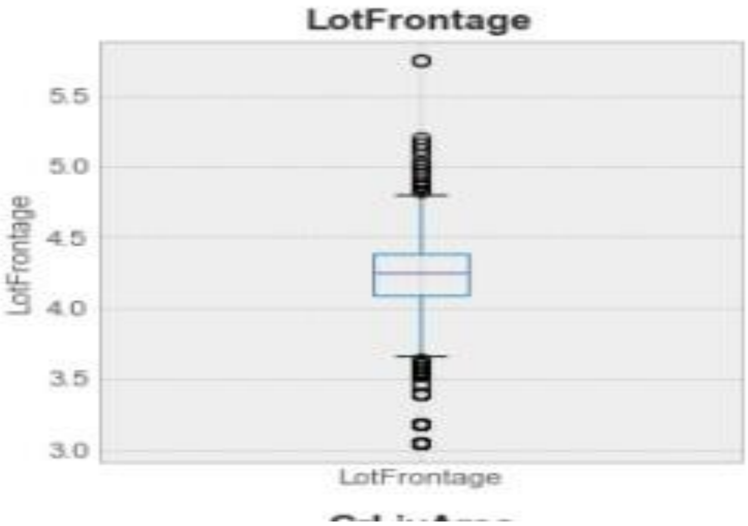
1. **Histogram** for checking the distribution of data of single column. A histogram is basically used to represent data provided in a form of some groups. It is accurate method for the graphical representation of numerical data distribution. It is a type of bar plot where X-axis represents the bin ranges while Y-axis gives information about frequency.



2. **Scatterplot** to see the distribution of data among two columns. Scatter plots are used to observe relationship between variables and uses dots to represent the relationship between them. The scatter() method in the matplotlib library is used to draw a scatter plot. Scatter plots are widely used to represent relation among variables and how change in one affects the other

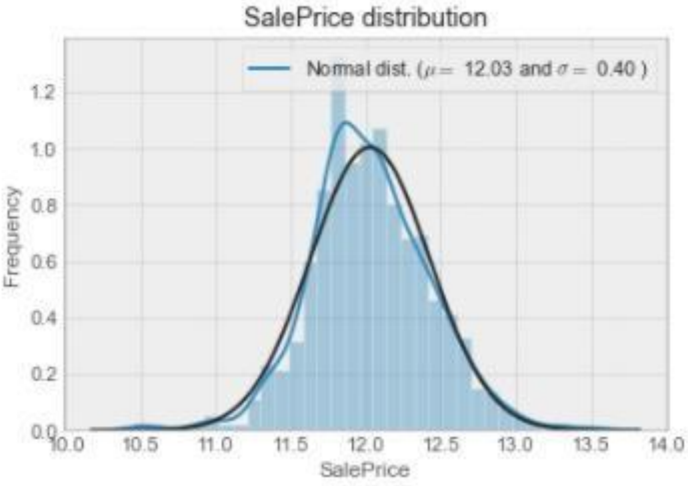


3. **Box plot** : A Box Plot is also known as Whisker plot is created to display the summary of the set of data values having properties like minimum, first quartile, median, third quartile and maximum. In the box plot, a box is created from the first quartile to the third quartile, a vertical line is also there which goes through the box at the median. Here x-axis denotes the data to be plotted while the y-axis shows the frequency distribution.

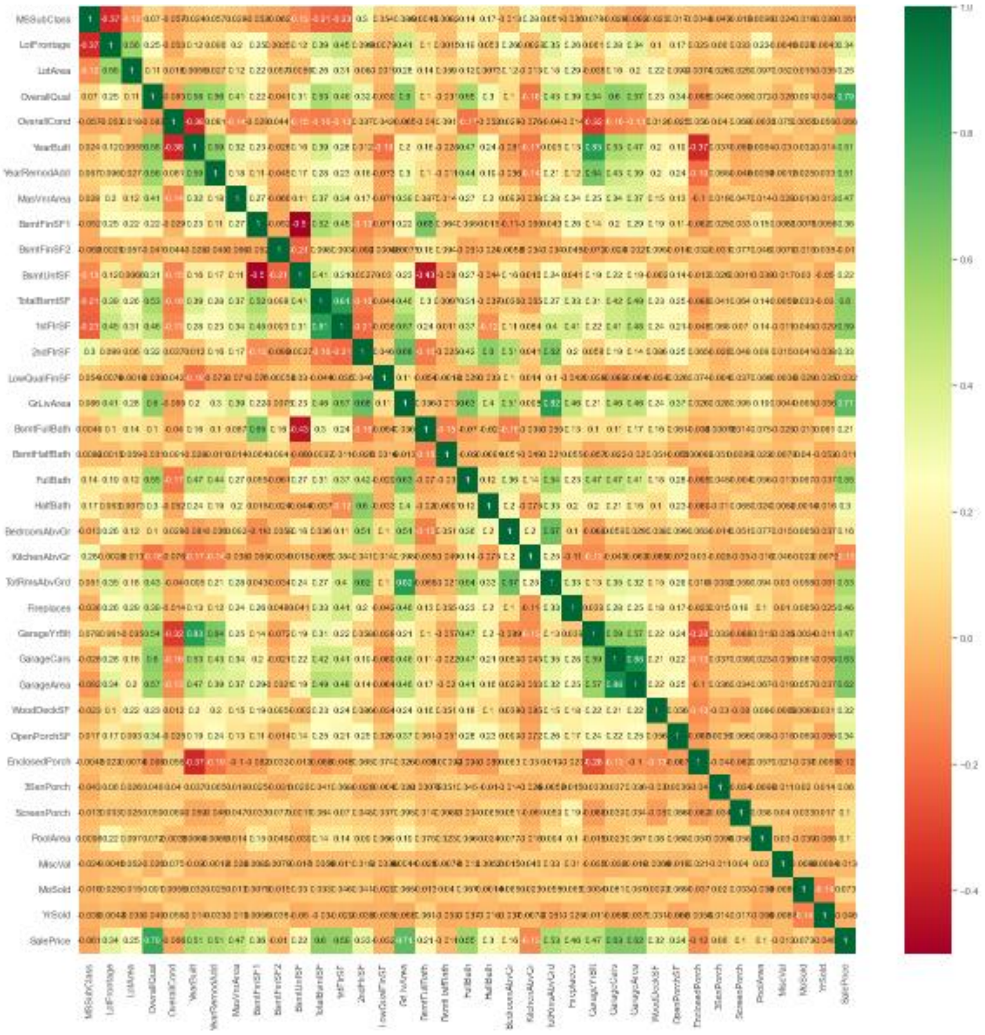


4. **Distribution plot** : Python Seaborn module contains various functions to plot the data and depict the data variations. The seaborn.distplot() function is used to plot the distplot. The distplot represents the univariate distribution of data i.e. data distribution of a variable against the density distribution.

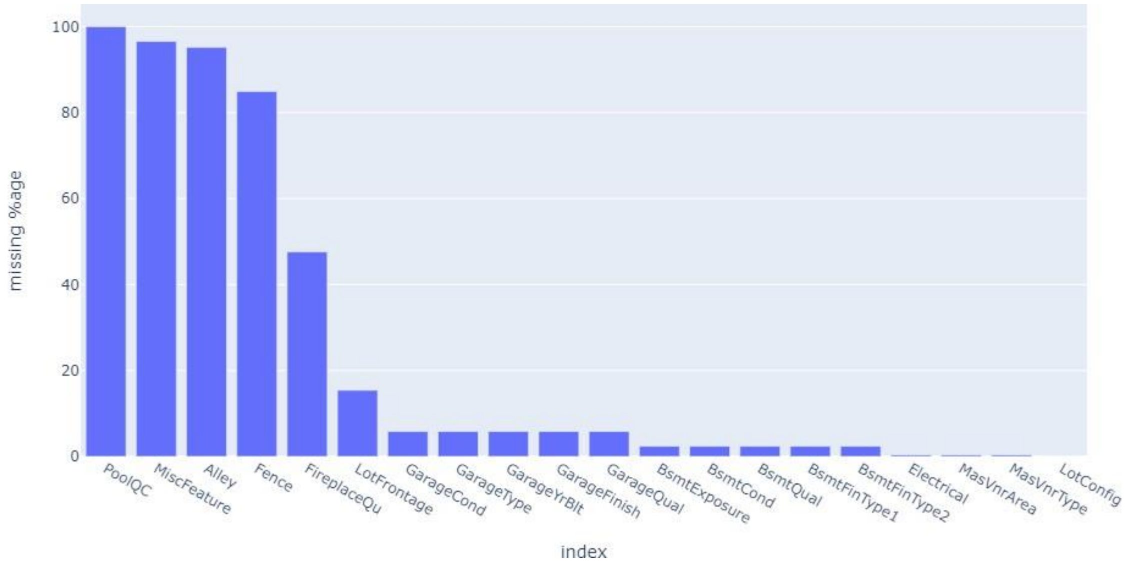
```
mu = 12.03 and sigma = 0.40
```



5. **Heatmap** is used to see the correlation of columns.



6. **Bar Plot :** A bar plot or bar chart is a graph that represents the category of data with rectangular bars with lengths and heights that is proportional to the values which they represent. The bar plots can be plotted horizontally or vertically. A bar chart describes the comparisons between the discrete categories. One of the axis of the plot represents the specific categories being compared, while the other axis represents the measured values corresponding to those categories.



- Interpretation of the Results

The data is giving in two sets i.e train dataset and the test dataset. Then I check the missing values and represent through the plotly and draw the bargraph for the better clearification. After that I check the describe method check the mean, median , Standard deviation, max values which shows that the there are outliers present and the data is not normally distributed as the data is not complete and some features were not included, which made the right or left skewed data somewhere. Hence we use a log function to make the feature Normally Distributed. And also we use the box cox method to remove some outliers which are very-very far from the normal distribution. To see the inter relation of features with the target column we use the correlation method, and then visualize with the help of **Heatmap** to

see the correlation between features and with the target column.
As the dataset is in the numerical and categorical form, and for the better understanding of the data, I first divided the numerical data into discrete and continuous features on the basis of less than 25 unique values are discrete features and other remaining numerical columns are continuous features. Then I visualize the data numerical columns, **Barplot** for the discrete Variables and **Histogram** for continuous variables, and Histogram shows that the some columns are skewed.

And then to check the relationship between the Target column i.e sales price and other columns I used the **scatter plots** for the better understanding and meaningfulness.

And then I did the Feature selection process where I convert the categorical column into the numerical with the help of one hot encoding and used the statistical libraries like skew for checking the skewness and norm for the normal distribution and box-cox for the outliers removal. And to check the outliers I used the **Boxplot** which shows the outliers present in some columns.

Here I use the z-score with the threshold value of 3 to check the outliers present in the dataset.

And finally the dataset get divided into x(without target column) and y(only target column present) for the model training.

# CONCLUSION

- Key Findings and Conclusions of the Study

The prediction for House price we find out that the Sales Price is highly dependent on the

1. OverallQual(Overall material and finish quality) and
2. GrLivArea( Above grade (ground) living area square feet)

- That Means if the Overall Quality of the Material and Finishing of the Home is good , it's Sale Price will be high and Similarly, if the Ground living area square feet is high, Price will increase accordingily.

- Other than that The SalesPrice depends on the 6 other factors which are :

-TotalBsmtSF: Total square feet of basement area

-1stFlrSF: First Floor square feet

-FullBath: Full bathrooms above grade

-TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)

-YearBuilt: Original construction date

-YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)

- PoolQc(PoolQuality), MiscFeature(Miscellaneous feature not covered in other categories) and Alley have the highest number of missing values, as many homes have no pools. Which means the houses which have pools have a Higher price in the market.

- Other than that some columns are right skewed in the dataset.

- In this project we use different models like Linear Regression, Lasso, Ridge, DecisionTree Regression and Random Forest Regression.

- Here we select the RandomForestRegressor model for our final model training and testing as it gives the very less root mean squared error value and also its R2 score is highest among the all models we choose.

- Learning Outcomes of the Study in respect of Data Science

In this project , I came to know about the real estate market working, which factors are important for the sell and purchasing of the house and with the help of the visualization tool like matplotlib, seaborn and plotly , it get easier for the understanding.

The study shows different-different regression algorithms when predicting house prices for Australia .The results were good for the data due to it being  with features and having strong correlation.

Hence, the data needs more features to be added preferably with a strong correlation with the house price. However, Random Forest gave the best RMSE score, got the best R2 score overall. The final results of this study showed that Random Forest makes better prediction compared to other used algorithms.

Kitchen above grade, and Enclosed porch have a weak negative influence on house prices, whereas 3SsnPorch and Month it sold have a weak positive influence.

The results answer the research questions as follows:

- *Which variables are important to predict the price of variable?*
  The Overall quality of the house, material used and the ground above living Area are very important as they were highly correlated with the Sales price. That means if the Overall quality increases house price also increases.

- *How do these variables describe the price of the house?*

  Houses are very important for the person's life and to sell and purchase the house is most important task . The first thing came into mind is the overall quality of the material, which are used for its making and the finishing of the house. The most beautiful is the finishing the more price it get. That's why these features like Overall quality, ground above living area are very important for describing the price for the house. Other than that are :
  - Total square feet of basement area ,First Floor square feet ,Full bathrooms above grade , Total rooms above grade (does not include bathrooms) , Original construction date , Remodel date (same as construction date if no remodeling or additions) .All of these are important.

## Limitations of this work and Scope for Future Work

Future work on this study could be divided into five main areas to improve the result even further. Which can be done by:

- The used pre-processing methods do help in the prediction R2score. However, experimenting with different combinations of pre-processing methods to achieve less Root Mean squared error and Good R2 score.

- Make use of the available features and if they could be combined as binning features has shown that the data got improved.

- Training the datasets with different regression methods such as Lasso and Ridge. In order to expand the comparison and check the performance.

- The correlation has shown the association in the data. Thus, attempting to enhance the data is required to make rich with features that vary and can provide a strong correlation relationship.

- The factors that have been studied in this study has a weak correlation with the sale price. Hence, by adding more factors to the local dataset that affect the house price, such as GDP, average income, and the population. In order to increase the number of factors that have an impact on house prices. This could also lead toa better finding for question 1 and 2.

-----------------------------------Thank You-----------------------------------