HOUSING PRICE PREDICTION MODEL

Submitted by:

BHAVNA PIPLANI

ACKNOWLEDGMENT

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped and guided me in completion of the project.

INTRODUCTION

A. Problem Framing:

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 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?

Description of the domain related concepts that will be useful for better understanding of the project.

Data Cleaning, Data Visualisation using different plotting methods like barplot, countplot, scatterplot, Data pre-processing using LabelEncoder, StandardScalar and Model Training

B. Analytical Problem Framing

a) Mathematical/ Analytical Modelling of the Problem

Statistical modelling is the process of applying statistical analysis to a dataset. A statistical model is a mathematical representation (or mathematical model) of observed data.

When data analysts apply various statistical models to the data they are investigating, they are able to understand and interpret the information more strategically. Rather than sifting through the raw data, this practice allows them to identify relationships between variables, make predictions about future sets of data, and visualize that data so that non-analysts and stakeholders can consume and leverage it.

most common techniques will fall into the following two groups:

- **Supervised learning**, including regression and classification models.
- **Unsupervised learning**, including clustering algorithms and association rules.

Some of the most common regression models include **LinearRegression,Lasso,Ridge,RandomForestRegressor, KNeighborsRegressor,GradientBoostingRegressor, AdaBoostRegressor**.

b) Data Sources and their formats

Data Source file was given in csv format with all the necessary variables for further Data Cleaning, Data pre-processing and Model Training. Data Description can be seen in the following table:

Variable	Definition
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 & NEWI 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
Alley	Type of alley access to property Grvl Gravel Pave Paved NA No alley access
LotShape	General shape of property Reg Regular IR1 Slightly irregular IR2 Moderately Irregular IR3 Irregular
LandContour	Flatness of the property Lvl Near Flat/Level Bnk Banked - Quick and significant rise from street grade to building HLS Hillside - Significant slope from side to side Low Depression
Utilities	Type of utilities available AllPub All public Utilities (E,G,W,& S) NoSewr Electricity, Gas, and Water (Septic Tank) NoSeWa Electricity and Gas Only ELO Electricity only
LotConfig	Lot configuration Inside Inside lot Corner Corner lot CulDSac Cul-de-sac FR2 Frontage on 2 sides of property FR3 Frontage on 3 sides of property
LandSlope	Slope of property Gtl Gentle slope

	Mod Moderate Slope Sev Severe Slope
Neighborhood	Physical locations within Ames city limits Blmngtn Bloomington Heights Blueste Bluestem BrDale Briardale BrkSide Brookside ClearCr Clear Creek CollgCr College Creek Crawfor Crawford Edwards Edwards Gilbert Gilbert IDOTRR Iowa DOT and Rail Road MeadowV Meadow Village Mitchel Mitchell Names North Ames NoRidge Northridge NPkVill Northpark Villa NridgHt Northridge Heights NWAmes Northwest Ames OldTown Old Town SWISU South & West of Iowa State University Sawyer Sawyer SawyerW Sawyer West Somerst Somerset StoneBr Stone Brook Timber Timberland Veenker Veenker
Condition1	Proximity to various conditions Artery Adjacent to arterial street Feedr Adjacent to feeder street Norm Normal RRNn Within 200' of North-South Railroad RRAn Adjacent to North-South Railroad PosN Near positive off-site feature--park, greenbelt, etc. PosA Adjacent to postive off-site feature RRNe Within 200' of East-West Railroad RRAe Adjacent to East-West Railroad

Condition2	Proximity to various conditions (if more than one is present) Artery Adjacent to arterial street Feedr Adjacent to feeder street Norm Normal RRNn Within 200' of North-South Railroad RRAn Adjacent to North-South Railroad PosN Near positive off-site feature--park, greenbelt, etc. PosA Adjacent to postive off-site feature RRNe Within 200' of East-West Railroad RRAe Adjacent to East-West Railroad
BldgType	Type of dwelling 1Fam Single-family Detached 2FmCon. Two-family Conversion; originally built as one-family dwelling Duplx Duplex TwnhsE Townhouse End Unit TwnhsI Townhouse Inside Unit
HouseStyle	Style of dwelling 1Story One story 1.5Fin One and one-half story: 2nd level finished 1.5Unf One and one-half story: 2nd level unfinished 2Story Two story 2.5Fin Two and one-half story: 2nd level finished 2.5Unf Two and one-half story: 2nd level unfinished SFoyer Split Foyer SLvl Split Level
OverallQual	Rates the overall material and finish of the house 10 Very Excellent 9 Excellent 8 Very Good 7 Good 6 Above Average 5 Average 4 Below Average 3 Fair 2 Poor 1 Very Poor

OverallCond	Rates the overall condition of the house 10 Very Excellent 9 Excellent 8 Very Good 7 Good 6 Above Average 5 Average 4 Below Average 3 Fair 2 Poor 1 Very Poor
YearBuilt	Original construction date
YearRemodAdd	Remodel date (same as construction date if no remodeling or additions)
RoofStyle	Type of roof Flat Flat Gable Gable Gambrel Gabrel (Barn) Hip Hip Mansard Mansard Shed Shed
RoofMatl	Roof material ClyTile Clay or Tile CompSh. Standard (Composite) Shingle Membran Membrane Metal Metal Roll Roll Tar&Grv Gravel & Tar WdShake Wood Shakes WdShngl Wood Shingles
Exterior1st	Exterior covering on house AsbShng Asbestos Shingles AsphShn Asphalt Shingles BrkComm Brick Common BrkFace Brick Face CBlock Cinder Block

	CemntBd Cement Board HdBoard Hard Board ImStucc Imitation Stucco MetalSd Metal Siding Other Other Plywood Plywood PreCast PreCast Stone Stone Stucco Stucco VinylSd Vinyl Siding Wd Sdng Wood Siding WdShing Wood Shingles
Exterior2nd	Exterior covering on house (if more than one material) AsbShng Asbestos Shingles AsphShn Asphalt Shingles BrkComm Brick Common BrkFace Brick Face CBlock Cinder Block CemntBd Cement Board HdBoard Hard Board ImStucc Imitation Stucco MetalSd Metal Siding Other Other Plywood Plywood PreCast PreCast Stone Stone Stucco Stucco VinylSd Vinyl Siding Wd Sdng Wood Siding WdShing Wood Shingles
MasVnrType	Masonry veneer type BrkCmn Brick Common BrkFace Brick Face CBlock Cinder Block None None Stone Stone
MasVnrArea	Masonry veneer area in square feet
ExterQual	Evaluates the quality of the material on the exterior

	Ex Excellent Gd Good TA Average/Typical Fa Fair Po Poor
ExterCond	Evaluates the present condition of the material on the exterior Ex Excellent Gd Good TA Average/Typical Fa Fair Po Poor
Foundation	Type of foundation BrkTil Brick & Tile CBlock Cinder Block PConc Poured Contrete Slab Slab Stone Stone Wood Wood
BsmtQual	Evaluates the height of the basement Ex Excellent (100+ inches) Gd Good (90-99 inches) TA Typical (80-89 inches) Fa Fair (70-79 inches) Po Poor (<70 inches NA No Basement
BsmtCond	Evaluates the general condition of the basement Ex Excellent Gd Good TA Typical - slight dampness allowed Fa Fair - dampness or some cracking or settling Po Poor - Severe cracking, settling, or wetness NA No Basement
BsmtExposure	Refers to walkout or garden level walls Gd Good Exposure

	Av Average Exposure (split levels or foyers typically score average or above) Mn Mimimum Exposure No No Exposure NA No Basement
BsmtFinType1	Rating of basement finished area GLQ Good Living Quarters ALQ Average Living Quarters BLQ Below Average Living Quarters Rec Average Rec Room LwQ Low Quality Unf Unfinshed NA No Basement
BsmtFinSF1	Type 1 finished square feet
BsmtFinType2	Rating of basement finished area (if multiple types) GLQ Good Living Quarters ALQ Average Living Quarters BLQ Below Average Living Quarters Rec Average Rec Room LwQ Low Quality Unf Unfinshed NA No Basement
BsmtFinSF2	Type 2 finished square feet
BsmtUnfSF	Unfinished square feet of basement area
TotalBsmtSF	Total square feet of basement area
Heating	Type of heating Floor Floor Furnace GasA Gas forced warm air furnace GasW Gas hot water or steam heat Grav Gravity furnace OthW Hot water or steam heat other than gas Wall Wall furnace
HeatingQC	Heating quality and condition Ex Excellent Gd Good TA Average/Typical

	Fa Fair Po Poor
CentralAir	Central air conditioning N No Y Yes
Electrical	Electrical system SBrkr Standard Circuit Breakers & Romex FuseA Fuse Box over 60 AMP and all Romex wiring (Average) FuseF 60 AMP Fuse Box and mostly Romex wiring (Fair) FuseP 60 AMP Fuse Box and mostly knob & tube wiring (poor) Mix Mixed
1stFlrSF	First Floor square feet
2ndFlrSF	Second floor square feet
LowQualFinSF	Low quality finished square feet (all floors)
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 (does NOT include basement bedrooms)
Kitchen	Kitchens above grade
KitchenQual	Kitchen quality Ex Excellent Gd Good TA Typical/Average Fa Fair Po Poor
TotRmsAbvGrd	Total rooms above grade (does not include bathrooms)
Functional	Home functionality (Assume typical unless deductions are warranted) Typ Typical Functionality Min1 Minor Deductions 1 Min2 Minor Deductions 2 Mod Moderate Deductions

	Maj1 Major Deductions 1 Maj2 Major Deductions 2 Sev Severely Damaged Sal Salvage only
Fireplaces	Number of fireplaces
FireplaceQu	Fireplace quality Ex Excellent - Exceptional Masonry Fireplace Gd Good - Masonry Fireplace in main level TA Average - Prefabricated Fireplace in main living area or Masonry Fireplace in basement Fa Fair - Prefabricated Fireplace in basement Po Poor - Ben Franklin Stove NA No Fireplace
GarageType	Garage location 2Types More than one type of garage Attchd Attached to home Basment Basement Garage BuiltIn Built-In (Garage part of house – typically has room above garage) CarPort Car Port Detchd Detached from home NA No Garage
GarageYrBlt	Year garage was built
GarageFinish	Interior finish of the garage Fin Finished RFn Rough Finished Unf Unfinished NA No Garage
GarageCars	Size of garage in car capacity
GarageArea	Size of garage in square feet
GarageQual	Garage quality Ex Excellent Gd Good TA Typical/Average Fa Fair Po Poor

	NA No Garage
GarageCond	Garage condition Ex Excellent Gd Good TA Typical/Average Fa Fair Po Poor NA No Garage
PavedDrive	Paved driveway Y Paved P Partial Pavement N Dirt/Gravel
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 Ex Excellent Gd Good TA Average/Typical Fa Fair NA No Pool
Fence	Fence quality GdPrv Good Privacy MnPrv Minimum Privacy GdWo Good Wood MnWw Minimum Wood/Wire NA No Fence
MiscFeature	Miscellaneous feature not covered in other categories Elev Elevator Gar2 2nd Garage (if not described in garage section) Othr Other Shed Shed (over 100 SF)

	TenC Tennis Court NA None
MiscVal	$Value of miscellaneous feature
MoSold	Month Sold (MM)
YrSold	Year Sold (YYYY)
SaleType	Type of sale WD Warranty Deed - Conventional CWD Warranty Deed - Cash VWD Warranty Deed - VA Loan New Home just constructed and sold COD Court Officer Deed/Estate Con Contract 15% Down payment regular terms ConLw Contract Low Down payment and low interest ConLI Contract Low Interest ConLD Contract Low Down Oth Other
SaleCondition	Condition of sale Normal Normal Sale Abnorml Abnormal Sale - trade, foreclosure, short sale AdjLand Adjoining Land Purchase Alloca Allocation - two linked properties with separate deeds, typically condo with a garage unit Family Sale between family members Partial Home was not completed when last assessed (associated with New Homes)

c) Explanatory Data Analysis

Hardware and Software Requirements and Tools Used

Anaconda Software – Jupiter Notebook

```python
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.simplefilter("ignore")
```

Importing the dataset from the source file provided.

```python
#importing train dataset
ds=pd.read_csv("housing_train.csv",sep='\t')
ds.head(5)
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC
0	127	120	RL	NaN	4928	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN
1	889	20	RL	95.0	15865	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN
2	793	60	RL	92.0	9920	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN
3	110	20	RL	105.0	11751	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN
4	422	20	RL	NaN	16635	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN

After importing the dataset , checking the datatypes of each column.

```python
ds.dtypes #checking the datatypes of variables
```

```
Id                 int64
MSSubClass         int64
MSZoning          object
LotFrontage      float64
LotArea            int64
                  ...
MoSold             int64
YrSold             int64
SaleType          object
SaleCondition     object
SalePrice          int64
Length: 81, dtype: object
```

Train dataset has 1168 rows and 81 columns

```
ds.shape # checking no of columns and rows
```

```
(1168, 81)
```

After checking the no. of rows and columns count, we will check the null values if any, column count, datatypes of columns.

```
dss=ds.columns[ds.isnull().any()] # extracting all the variables having null values
dss
```

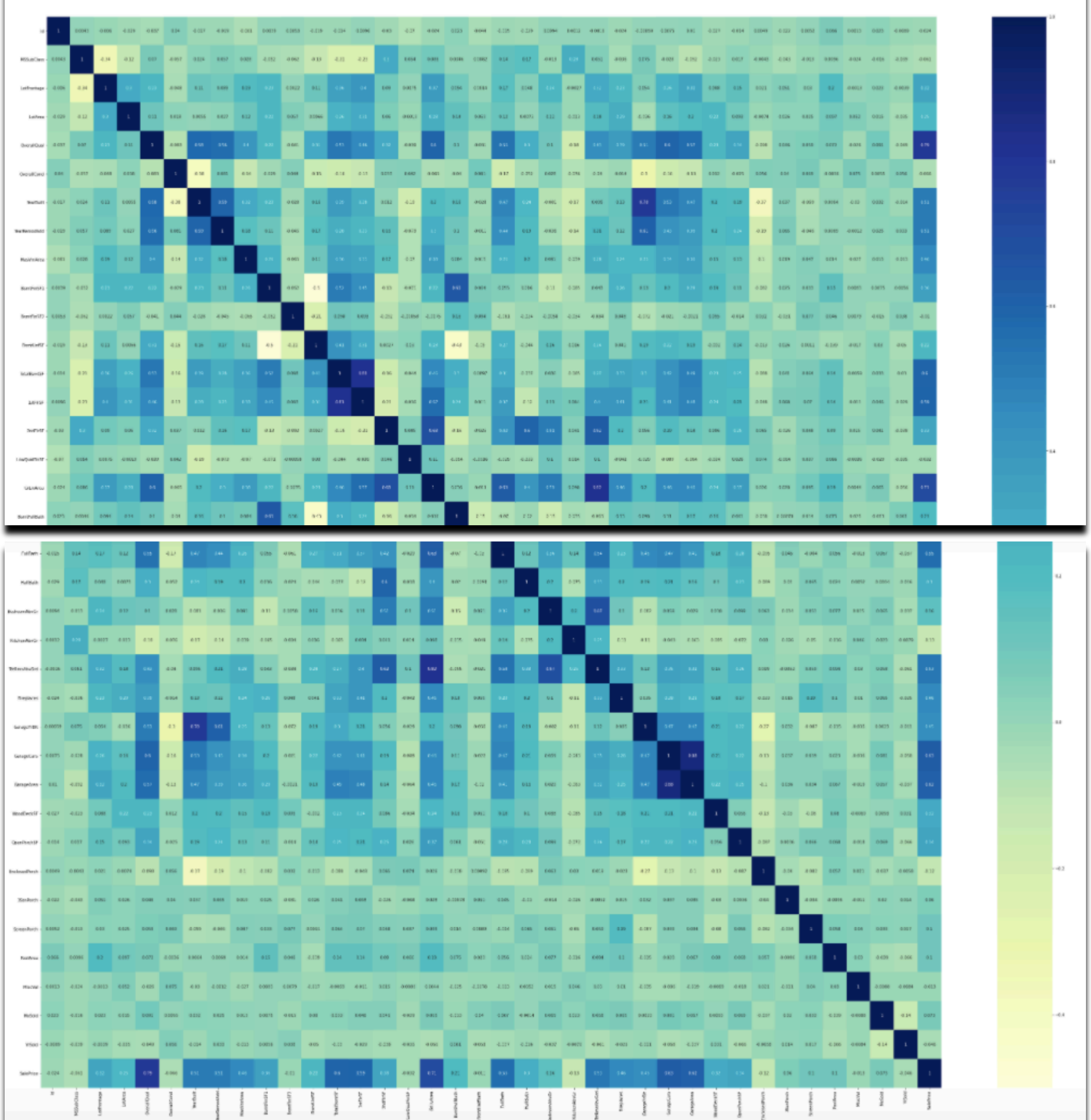
```
Index(['LotFrontage', 'Alley', 'MasVnrType', 'MasVnrArea', 'BsmtQual',
       'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2',
       'FireplaceQu', 'GarageType', 'GarageYrBlt', 'GarageFinish',
       'GarageQual', 'GarageCond', 'PoolQC', 'Fence', 'MiscFeature'],
      dtype='object')
```

The dataset is not clean. Data is missing in many of the features . There are missing or null values in the dataset. Nan values will be replaced by values mentioned in data description provided.

After data cleaning, we will see the correlation of features with the target variable.

```python
plt.figure(figsize=(50,50))
sns.heatmap(ds.corr(),annot=True,cmap='YlGnBu')
```

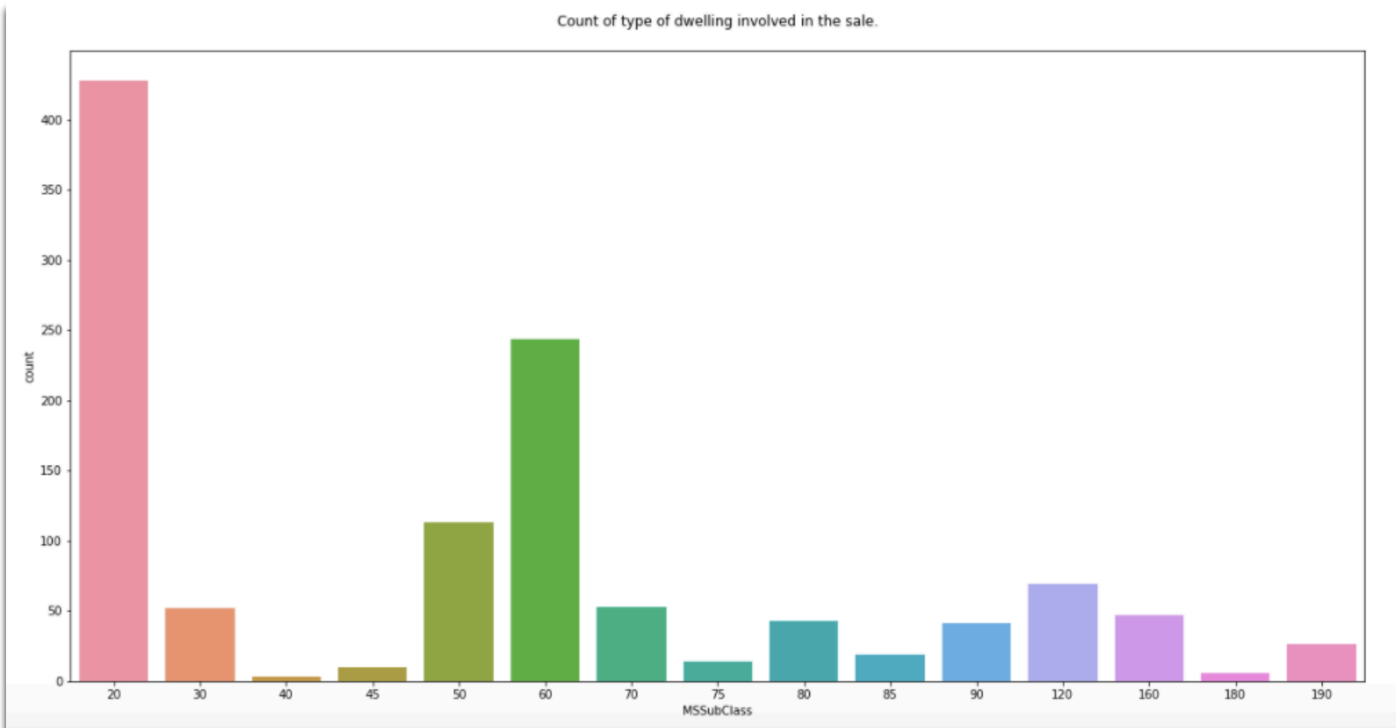
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8562efd60>



key observations here

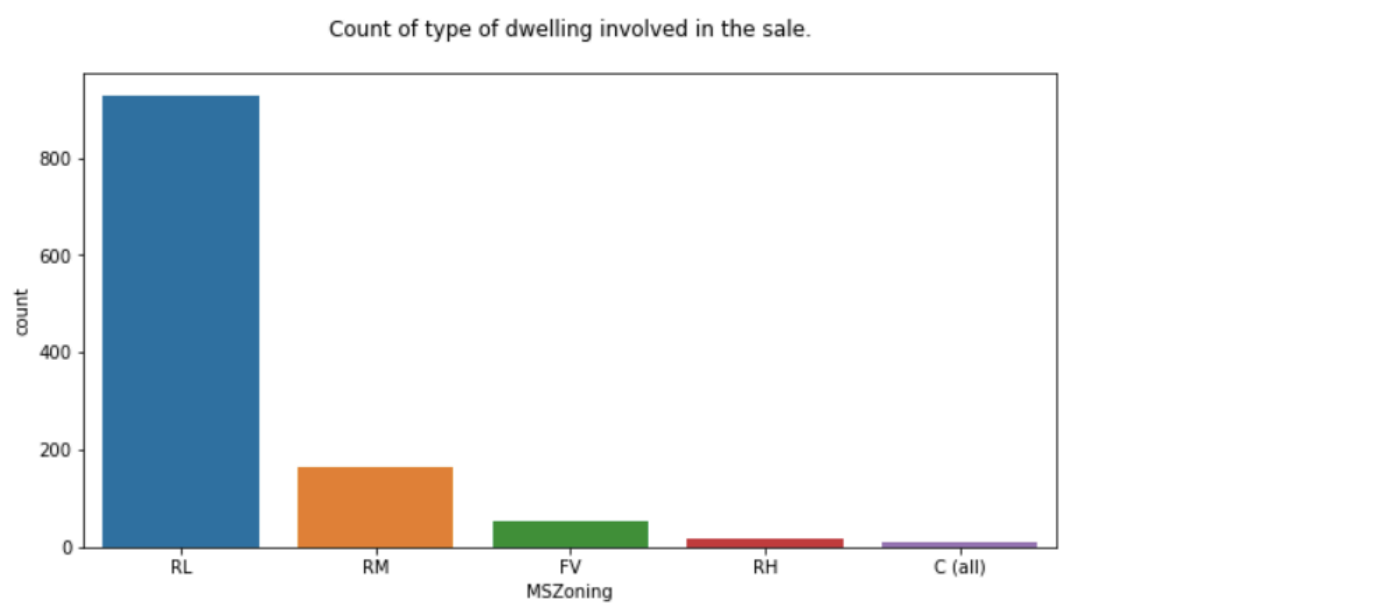
1. OverallQual is highly correlated with target variable ActualPrice.
2. Garagecars,GarageArea are highly correlated with each other.
3. GarageCars,garagearea,TotalBsmtSF, 1FirSF are highly correlated with target variable ActualPrice.

```
plt.figure(figsize=(20,10))
sns.countplot(ds["MSSubClass"])
plt.title("Count of type of dwelling involved in the sale.\n ")
plt.show()
```



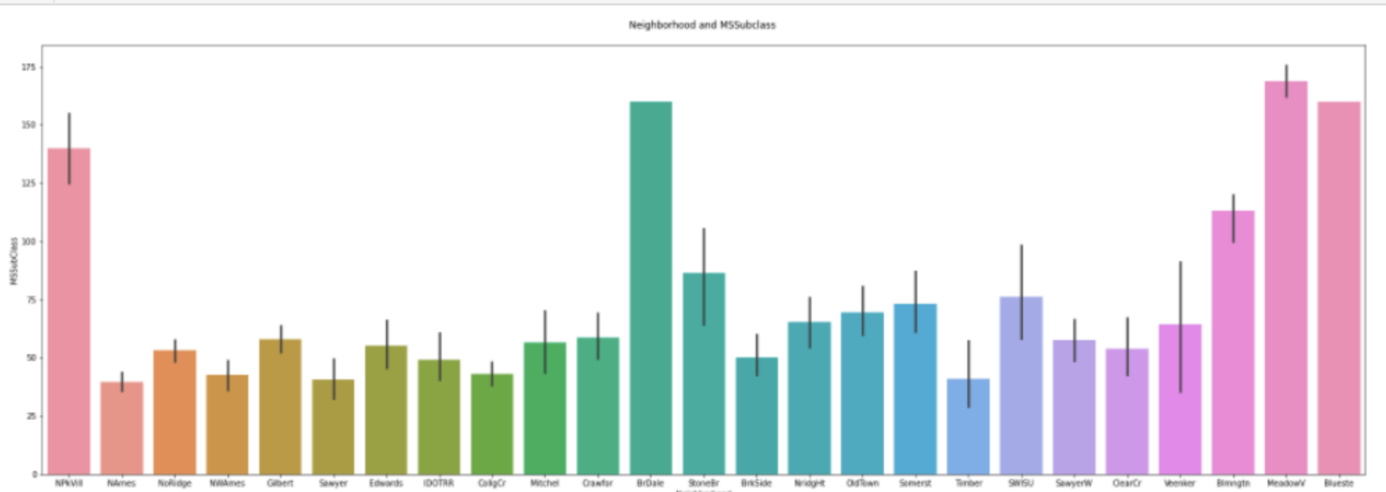
Above countplot shows that
1. 1-STORY 1946 & NEWER ALL STYLES are maximum which are in sale.
2. 1-STORY W/FINISHED ATTIC ALL AGES is least one which is in sale.

```python
plt.figure(figsize=(10,5))
sns.countplot(ds["MSZoning"])
plt.title("Count of type of dwelling involved in the sale.\n ")
plt.show()
```



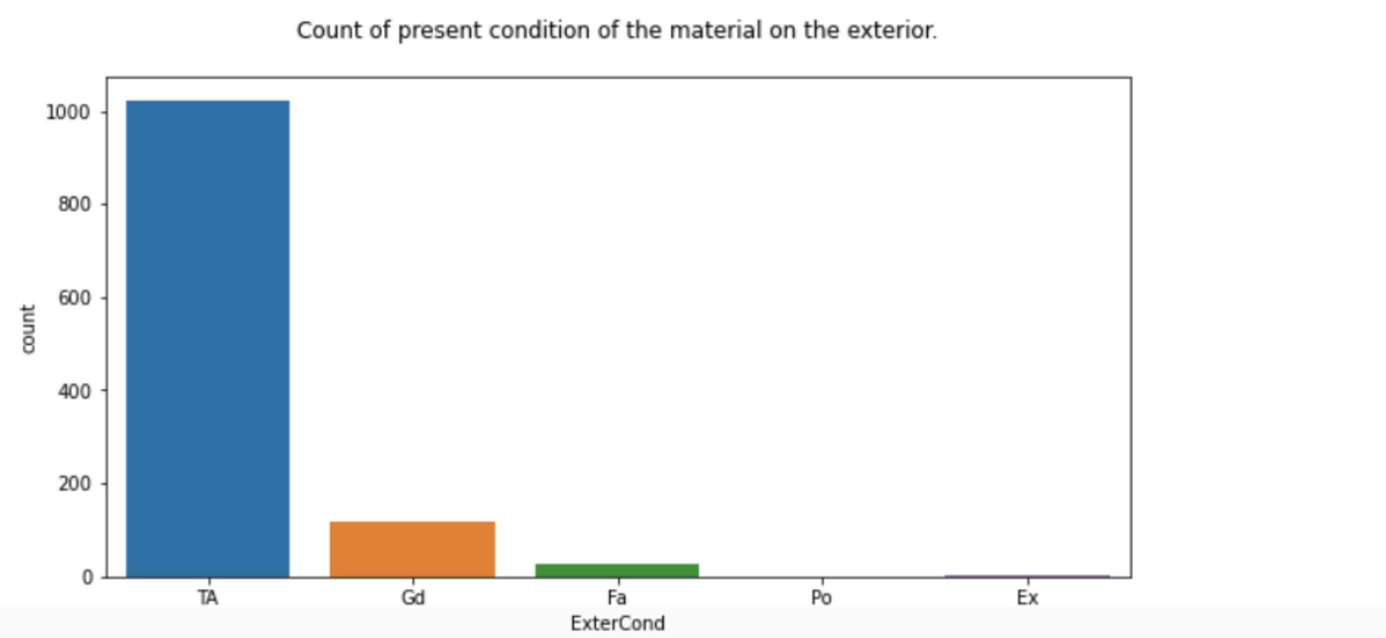
Above countplot shows that Residential Low Density property is maximum on sale. and Commercial is least on sale.

```python
plt.figure(figsize=(30, 10))
sns.barplot(x='Neighborhood',y='MSSubClass',data=ds)
plt.title("Neighborhood and MSSubclass \n")
plt.show()
```



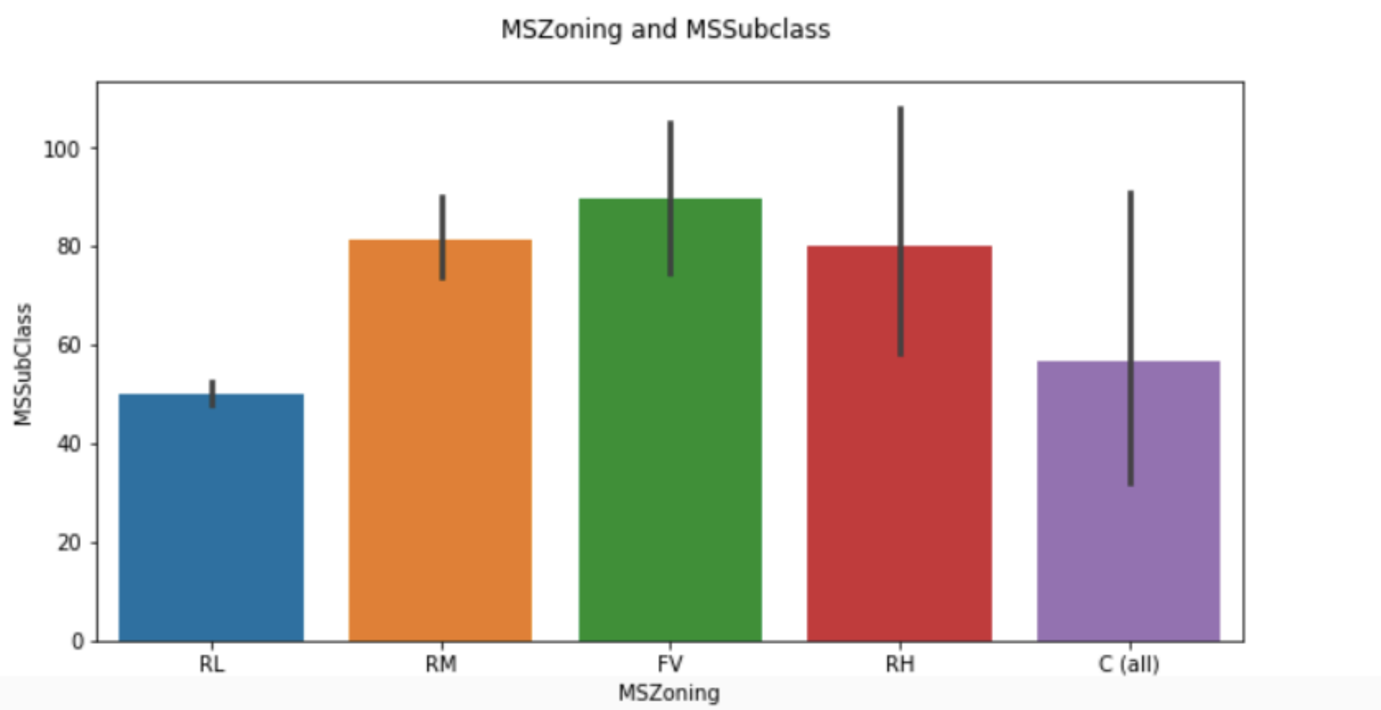
above barplot shows the neighbourhood of all MSSubclass dwelling involved in sale. like "Meadow Village" is in neighbourhood of "2-STORY PUD - 1946 & NEWER"

```
plt.figure(figsize=(10,5))
sns.countplot(ds["ExterCond"])
plt.title("Count of present condition of the material on the exterior.\n ")
plt.show()
```



Above countplot Evaluates the present condition of the material on the exterior is Average/Typical

```
plt.figure(figsize=(10,5))
sns.barplot(x='MSZoning',y='MSSubClass',data=ds)
plt.title("MSZoning and MSSubclass \n")
plt.show()
```



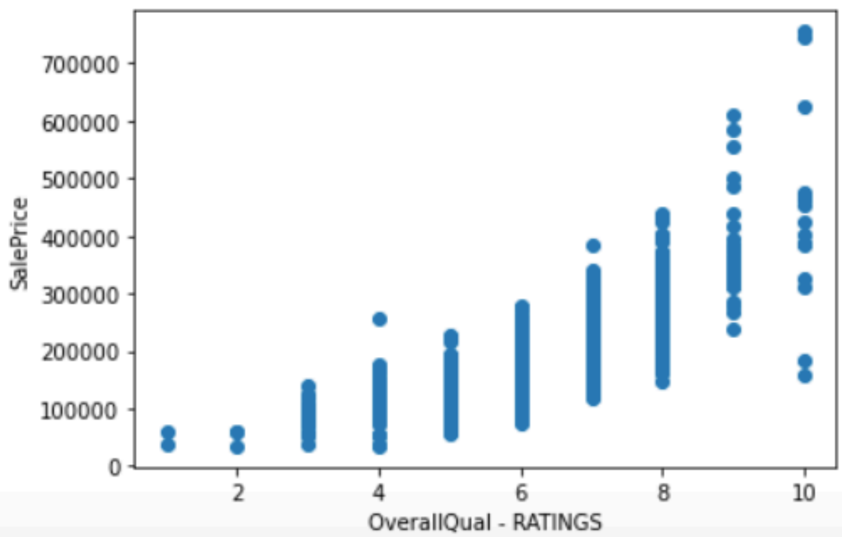
Above countplot shows Maximum dwelling involved in sale are "DUPLEX - ALL STYLES AND AGES" having general zoning classification as "Floating Village Residential"

```
plt.figure(figsize=(10,5))
sns.distplot(ds[ds['MSSubClass']==90]['SalePrice'],kde=False,bins=40)
plt.title('Sale Price of Dwelling having MSSubClass=90  \n DUPLEX - ALL STYLES AND AGES')
```

```
Text(0.5, 1.0, 'Sale Price of Dwelling having MSSubClass=90  \n DUPLEX - ALL STYLES AND AGE
S')
```

Above distplot shows the saleprice of dwelling where MSSubClass=90 i.e DUPLEX - ALL STYLES AND AGES

```python
plt.scatter(ds["OverallQual"],ds["SalePrice"])
plt.xlabel("OverallQual - RATINGS")
plt.ylabel("SalePrice")
plt.show()
```



Above scatterplot shows that Highest rating have highest prices.

d) Pre-Processing Pipeline

Maximum columns have the numerical values other than 18 features. For moving further for model training, we need to transform these nominal values into numerical values by encoding the data.

Label Encoding

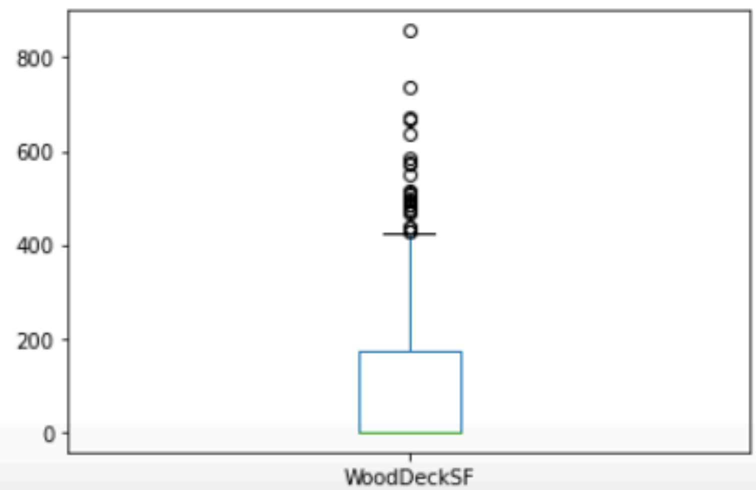
```python
# changing the nominal value to integer for training model
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
list1=['MSZoning','Street', 'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig', 'LandSlope',
       'Neighborhood','Condition1', 'Condition2', 'BldgType', 'HouseStyle', 'RoofStyle','RoofMatl',
       'Exterior1st','Exterior2nd','MasVnrType','ExterQual','ExterCond','Foundation','BsmtQual',
       'BsmtCond', 'BsmtExposure','BsmtFinType1','BsmtFinType2','Heating','HeatingQC','CentralAir',
       'Electrical','KitchenQual','Functional','FireplaceQu','GarageType','GarageFinish','GarageQual',
       'GarageCond','PavedDrive','PoolQC','Fence','MiscFeature','SaleType','SaleCondition']
for val in ds_cat:
    ds[val]=le.fit_transform(ds[val].astype(str))
```

After transforming the nominal variables, the dataset is all numerical.

Handling the outliers¶

```
ds["WoodDeckSF"].plot.box()
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f8bd3f8ed60>



As many type of dwelling may have lotarea, basementfin1 , alley area etc and many may not have . that may be making a huge difference in mean and median and 75% and max values . So, not removing the outliers

We will check the distribution of skewness in the data. If it is there, we will remove it.

```
ds.skew() # checking if the data is skewed
```

```
Id                0.026526
MSSubClass        1.422019
MSZoning         -1.796785
LotFrontage       2.710383
LotArea          10.659285
                    ...
MoSold            0.220979
YrSold            0.115765
SaleType         -3.660513
SaleCondition    -2.671829
SalePrice         1.953878
Length: 81, dtype: float64
```

little skewness is there so we will remove it

```
# seperatng the target variable
ds_x=ds.drop(columns=['SalePrice'])
y_t=pd.DataFrame(ds['SalePrice'])
print(ds_x.shape, y_t.shape)
```

```
(1168, 80) (1168, 1)
```

```
from sklearn.preprocessing import power_transform
```

```
ds_x=power_transform(ds_x,method='yeo-johnson')
```

Once skewness removal has been done, We will do scaling of the dataset. Scaling is a technique to standardize the independent features present in the data in a fixed range. If feature scaling is not done, then a machine learning algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values.

scaling the dataset

```
from sklearn.preprocessing import StandardScaler
```

```
#scaling the dataset
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
scaledX=sc.fit_transform(ds_x)
scaledX.shape
```

```
(1168, 80)
```

Libraries and packages used for model training are listed below

Data Modelling

```
# importing our libraries
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression,Lasso,Ridge,ElasticNet
from sklearn.ensemble import RandomForestRegressor,AdaBoostRegressor,GradientBoostingRegressor
from sklearn.neighbors import KNeighborsRegressor
```

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import cross_val_score
from sklearn.metrics import r2_score,mean_absolute_error,mean_squared_error
```

e) Model Development and Evaluation

Identification of possible problem-solving approaches (methods)

most common techniques will fall into the following two groups:

a. Supervised learning, including regression and classification models.
b. Unsupervised learning, including clustering algorithms and association rules.

For this dataset, I will be using regression model because the output variable is continuous.

Testing of Identified Approaches (Algorithms)

Here, In this project I will be using LinearRegression, Lasso, Ridge, RandomForestRegressor,KNeighborsRegressor,GradientBoostingRegressor, AdaBoostRegressor algorithms

Finding the best random state :

finding the best random state

```
best_rstate=0
accuracy=0
for i in range(30,200):
    x_train,x_test,y_train,y_test=train_test_split(scaledX,y_t,test_size=.22,random_state=i)
    mod=LinearRegression()
    mod.fit(x_train,y_train)
    predlr=mod.predict(x_test)
    tempaccu=r2_score(y_test,predlr)
    if(tempaccu>accuracy):
        accuracy=tempaccu
        best_rstate=i

print("Best Accuracy",accuracy*100, "Random state",best_rstate)
```

```
Best Accuracy 86.00361454154653 Random state 148
```

Using the best random state:

using the best random state

```
x_train,x_test,y_train,y_test=train_test_split(scaledX,y_t,test_size=.22,random_state=148)
```

```
x_train.shape , x_test.shape
```

```
((911, 80), (257, 80))
```

```
y_train.shape , y_test.shape
```

```
((911, 1), (257, 1))
```

Using different algorithms, we will try to find the best model.

Finding the best model

```python
#using algorithms in for loops
model=[LinearRegression(),Lasso(),Ridge(),RandomForestRegressor(),KNeighborsRegressor(),GradientBoostingRegressor()
for m in model:
    m.fit(x_train,y_train)
    y_pred=m.predict(x_test)
    r2score=r2_score(y_test,y_pred)
    cvscore=cross_val_score(LinearRegression(),x_train,y_train,cv=5).mean()
    print(m , "\nAccuracy Score of " ,r2score*100, "Cross Val Score", {cvscore*100})
    print("**********************************************************************\n")
```

```
LinearRegression()
Accuracy Score of  86.00361454154653 Cross Val Score {67.181131669369}
**********************************************************************

Lasso()
Accuracy Score of  86.00664904863633 Cross Val Score {67.181131669369}
**********************************************************************

Ridge()
Accuracy Score of  86.00405781614823 Cross Val Score {67.181131669369}
**********************************************************************

RandomForestRegressor()
Accuracy Score of  90.69848499135716 Cross Val Score {67.181131669369}
**********************************************************************

KNeighborsRegressor()
Accuracy Score of  78.47326833612966 Cross Val Score {67.181131669369}
**********************************************************************

GradientBoostingRegressor()
Accuracy Score of  91.76786443443163 Cross Val Score {67.181131669369}
**********************************************************************
```

Here, **GradientBoostingRegressor** has performed the best with accuracy score round(2) is 0.91

Doing a **GridSearchCV** is a great way to do hyper parameters tuning.

Hyperparameter Tuning

```python
# GradientBoostingRegressor is best performing model so finding its best parameter
from sklearn.model_selection import GridSearchCV
```

```python
GBR=GradientBoostingRegressor()
GBR.fit(x_train,y_train)
y_pred=GBR.predict(x_test)
r2score=r2_score(y_test,y_pred)
cvscore=cross_val_score(GBR,x_train,y_train,cv=5).mean()
print( "\nAccuracy Score of ",GBR ,"is",r2score*100,"and", "Cross Val Score is", {cvscore*100})
print("*****************************************************************************************\n")
search_grid={'n_estimators':[5, 6, 7, 8, 9, 10, 11, 12, 13, 15],'learning_rate':[.001,0.01,.1],'max_depth':[1,2,4]}
search=GridSearchCV(estimator=GBR,param_grid=search_grid,scoring='neg_mean_squared_error',n_jobs=1)
```

```
Accuracy Score of  GradientBoostingRegressor() is 91.88182151052608 and Cross Val Score is {70.03890946820167}
*****************************************************************************************
```

Now, as the model is performing good with the score of 91% , we will save the predicted_model .

Saving the model- Serialization

```python
# saving the prediction model

import pickle
filename="Housingprice.pkl"
pickle.dump(GBR,open(filename,'wb'))
```

```python
# load the model
fitted_model=pickle.load(open("Housingprice.pkl",'rb'))
```

After doing serialization, we will fit the model on test data of 292 rows given.

```python
# predictions over test data (houseprice_test.csv)
predictions=fitted_model.predict(scaled_df)
```

```python
predictions=predictions.astype(int)
```

```python
ds_pred=pd.DataFrame(data=predictions,columns=['SalePrice'])
ds_pred
```

	SalePrice
0	371689
1	212453
2	231748
3	188676
4	218146
...	...
287	246127
288	141279
289	157507
290	159595
291	95635

292 rows × 1 columns

CONCLUSION

Key Findings and Conclusions of the Study

1. OverallQual is highly correlated with target variable SalePrice.
2. Garagecars,GarageArea are highly correlated with each other.
3. GarageCars, garagearea, TotalBsmtSF, 1FirSF are highly correlated with target variable SalePrice.
4. It was found that removing outliers will be loss of more of the data. So, I decided not to remove them.
5. GradientBoostingRegressor was the best fit model.