House-Price Prediction

Submitted by:
Twinkle Patel

ACKNOWLEDGMENT

I would like to express my sincere thanks to FlipRobo Technologies company for supporting me throughout the internship and giving me the chance to explore the depth of Data Science by providing numerous projects like this, there are multiple people, YouTubers, organizations who guided me in this wonderful journey and few articles, blogs which helped me develop my models in this project. I would like to thank following people and company for the inspiration and help,

- FlipRobo Technologies
- DataTrained Team
- Arjav Patel

INTRODUCTION

• Business Problem Framing

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 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.

• Review of Literature

Linear Regression is evaluated for its ability to predict house prices for the company which is trying to get into the market and the final model in which gradient regressor gives the best accuracy.

- Motivation for the Problem Undertaken

This housing project was a highly motivated project as it includes the real time problem for The real estate company which is using the machine learning model for the prediction of house prices based on various factors. And The better the model the better of chance of profit for the business.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

The below image shows the Statistics analysis of the variable Sale Price

```
In [9]:   1 train['SalePrice'].describe()

Out[9]: count      1168.000000
        mean     181477.005993
        std       79105.586863
        min       34900.000000
        25%      130375.000000
        50%      163995.000000
        75%      215000.000000
        max      755000.000000
        Name: SalePrice, dtype: float64
```

The correlation of Sale price with all the other variables is given below..

```
SalePrice        1.000000
Skewed_SP        0.945730
OverallQual      0.789185
GrLivArea        0.707300
GarageCars       0.628329
GarageArea       0.619000
TotalBsmtSF      0.595042
1stFlrSF         0.587642
FullBath         0.554988
TotRmsAbvGrd     0.528363
YearBuilt        0.514408
YearRemodAdd     0.507831
GarageYrBlt      0.474346
MasVnrArea       0.466386
Fireplaces       0.459611
BsmtFinSF1       0.362874
LotFrontage      0.341294
OpenPorchSF      0.339500
2ndFlrSF         0.330386
WoodDeckSF       0.315444
HalfBath         0.295592
LotArea          0.249499
BsmtUnfSF        0.215724
BsmtFullBath     0.212924
BedroomAbvGr     0.158281
PoolArea         0.103280
ScreenPorch      0.100284
MoSold           0.072764
3SsnPorch        0.060119
BsmtFinSF2      -0.010151
BsmtHalfBath    -0.011109
MiscVal         -0.013071
Id              -0.023897
LowQualFinSF    -0.032381
YrSold          -0.045508
MSSubClass      -0.060775
OverallCond     -0.065642
EnclosedPorch   -0.115004
KitchenAbvGr    -0.132108
Name: SalePrice, dtype: float64
```

● Data Sources and their formats

The dataset contains 1460 entries each having 81 variables.
The Dataset contains Null values. You need to treat them using the domain knowledge and your own understanding. Extensive EDA has to be performed to gain relationships of important variables and prices. Data contains numerical as well as categorical variables. You need to handle them accordingly.

In [3]:

```python
# head() shows the first 5 rows of the data
train.head()
```

Out[3]:

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

5 rows × 81 columns

In [4]:

```python
test.head()
```

Out[4]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	ScreenPorch	PoolArea	PoolQC	Fence	MiscFeatur
0	337	20	RL	86.0	14157	Pave	NaN	IR1	HLS	AllPub	...	0	0	NaN	NaN	Na
1	1018	120	RL	NaN	5814	Pave	NaN	IR1	Lvl	AllPub	...	0	0	NaN	NaN	Na
2	929	20	RL	NaN	11838	Pave	NaN	Reg	Lvl	AllPub	...	0	0	NaN	NaN	Na
3	1148	70	RL	75.0	12000	Pave	NaN	Reg	Bnk	AllPub	...	0	0	NaN	NaN	Na
4	1227	60	RL	86.0	14598	Pave	NaN	IR1	Lvl	AllPub	...	0	0	NaN	NaN	Na

5 rows × 80 columns

There are 1460 entries in the train data set and 1459 entries in test data set. The data contains some NaN values too.

In [14]:

```python
numerical_features = train.select_dtypes(include=[np.number])
numerical_features.dtypes
```

Out[14]:

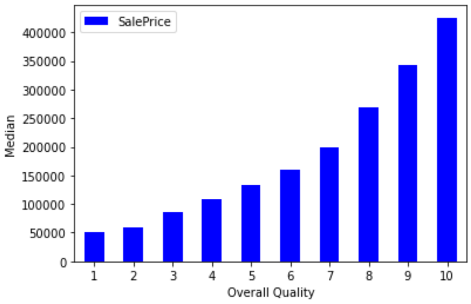
```
Id                 int64
MSSubClass         int64
LotFrontage      float64
LotArea            int64
OverallQual        int64
OverallCond        int64
YearBuilt          int64
YearRemodAdd       int64
MasVnrArea       float64
BsmtFinSF1         int64
BsmtFinSF2         int64
BsmtUnfSF          int64
TotalBsmtSF        int64
1stFlrSF           int64
2ndFlrSF           int64
LowQualFinSF       int64
GrLivArea          int64
BsmtFullBath       int64
BsmtHalfBath       int64
FullBath           int64
HalfBath           int64
BedroomAbvGr       int64
KitchenAbvGr       int64
TotRmsAbvGrd       int64
Fireplaces         int64
GarageYrBlt      float64
GarageCars         int64
GarageArea         int64
WoodDeckSF         int64
OpenPorchSF        int64
EnclosedPorch      int64
3SsnPorch          int64
ScreenPorch        int64
PoolArea           int64
MiscVal            int64
MoSold             int64
YrSold             int64
SalePrice          int64
Skewed_SP        float64
dtype: object
```

● Data Pre-processing

● Firstly, We treated the skewness using Log transformation. After that, We imputed the missing values and encoded the categorical values using One hot encoding Finally, We trained the model on the train set and tested the model on the test set .and Applied hyperparameters for improving the performance.

● Data Inputs- Logic- Output Relationships

In [19]:
```
quality_pivot.plot(kind='bar',color='blue')
plt.xlabel('Overall Quality')
plt.ylabel('Median')
plt.xticks(rotation=0)
plt.show()
```

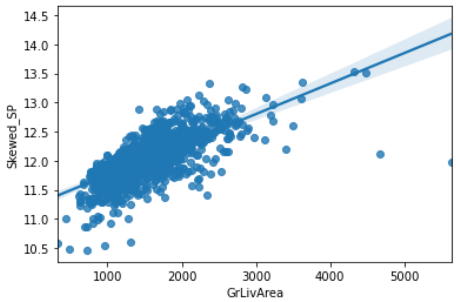


SalePrice varies directly with the Overall quality

SalePrice varies directly with the Overall quality

In [20]:
```
sns.regplot(x='GrLivArea',y='Skewed_SP',data=train)
```

Out[20]: <AxesSubplot:xlabel='GrLivArea', ylabel='Skewed_SP'>



SalePrice increases as the GrLivArea increases. We will also get rid of the outliers which severely affect the prediction of the survival rate.

SalePrice increases as the GrLivArea increases. So, We will also get rid of the outliers which severely affect the visualized of the survival rate.

- ## Hardware and Software Requirements and Tools Used

Hardware: 16GB RAM, 64-bit, i5 processor.

Software: Excel, Jupyter Notebook, python , google,CSV file

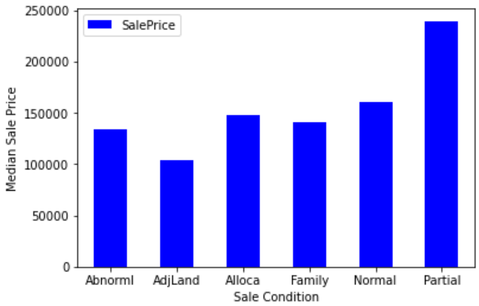
Libraries, Used:-

In [1]:
```python

import pandas as pd
from pandas import Series,DataFrame

import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline

from sklearn import preprocessing

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

Model/s Development and Evaluation

- ## Identification of possible problem-solving approaches (methods)

In [38]:
```python
# Feature Engineering
condition_pivot = train.pivot_table(index='SaleCondition',
                                    values='SalePrice', aggfunc=np.median)
condition_pivot.plot(kind='bar', color='blue')
plt.xlabel('Sale Condition')
plt.ylabel('Median Sale Price')
plt.xticks(rotation=0)
plt.show()
```



The Sale price is highly affected by sale conditions.

- ## Testing of Identified Approaches (Algorithms)

Linear Regression
Ridge Regressor

- # Run and Evaluate selected models

```
In [45]:
from sklearn.model_selection import train_test_split

```

```
In [46]:
X_train, X_test, y_train, y_test = train_test_split(
                                    X, y, random_state=42, test_size=0.4)
```

```
In [47]:
from sklearn import linear_model
from sklearn import ensemble


lr = ensemble.GradientBoostingRegressor()

```

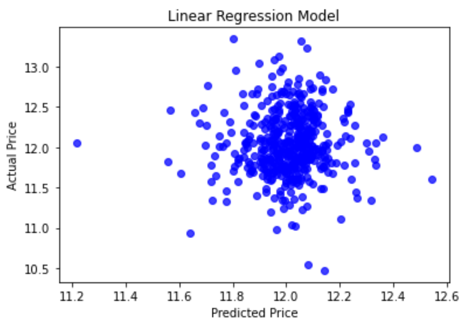
```
In [48]:
model = lr.fit(X_train, y_train)
```

```
In [49]:
print ("R^2 is: \n", model.score(X_test, y_test))

R^2 is:
 -0.1087569185973809
```

- # Key Metrics for success in solving problem under consideration

```
In [51]:
from sklearn.metrics import mean_squared_error
print ('RMSE is: \n', mean_squared_error(y_test, predictions))

RMSE is:
 0.1772903258754215
```

```
In [52]:
actual_values = y_test
plt.scatter(predictions, actual_values, alpha=.75,
            color='b') #alpha helps to show overlapping data
plt.xlabel('Predicted Price')
plt.ylabel('Actual Price')
plt.title('Linear Regression Model')
#pltrandom_state=None.show()

Out[52]: Text(0.5, 1.0, 'Linear Regression Model')
```



The R2 score and the RMSE is given above

- # Interpretation of the Results

The above visualization model and matrices found that the Gradient boost regressor performed the best 99% R2 score, with the least root mean square error which we were able to achieve from the data provided.

CONCLUSION

- ## Key Findings and Conclusions of the Study

From the abov visualization and model building we analyzed that Gradint boost regresor performed better when this type of dataset was given and based on the model performance it can be used to visualize the house price of the house based on numerous factors.

Based on the final model the Real estate company can make decisions and there is a higher possibility that the decisions will be profitable.