FLIP ROBO

NAME OF THE PROJECT

HOUSING: PRICE PREDICTION

Submitted by:

BISHWAJIT BHATTACHARYA

Problem Statement:

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.

I have some training data which has lots of null value. Which I taken care using SK Learn. As target value is integer, I need to process regression techniques. There is mixed of int, string and float data need to encode that also.

After all this Model will be ready.

Problem Framing

Total 1168 Rows and 81 Columns

But the issue is There is lots of null values and also data set has lots of different data type like float, int and String.

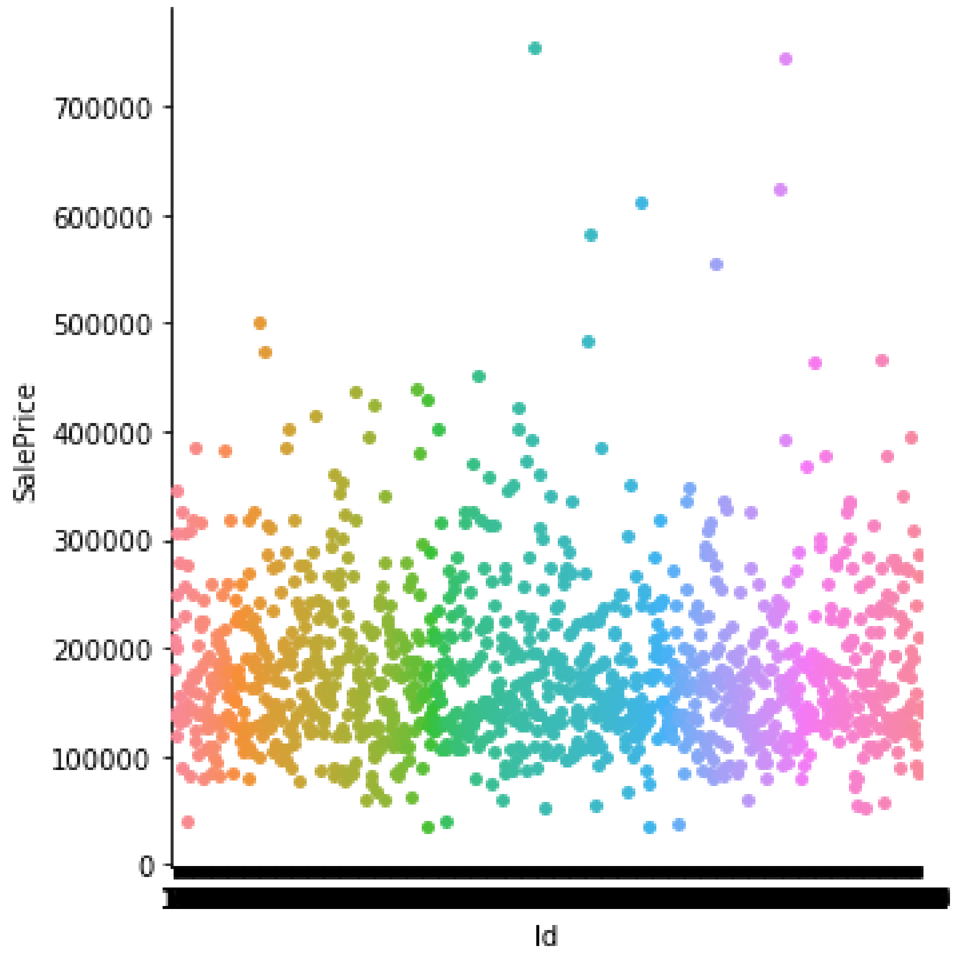
So before going to eda I just remove null value int data set bye using mean() technique.

```
train_data['LotFrontage']=train_data['LotFrontage'].replace(np.NAN,train_data
['LotFrontage'].mean())

train_data['GarageYrBlt']=train_data['GarageYrBlt'].replace(np.NAN,train_data
['GarageYrBlt'].mean())
```

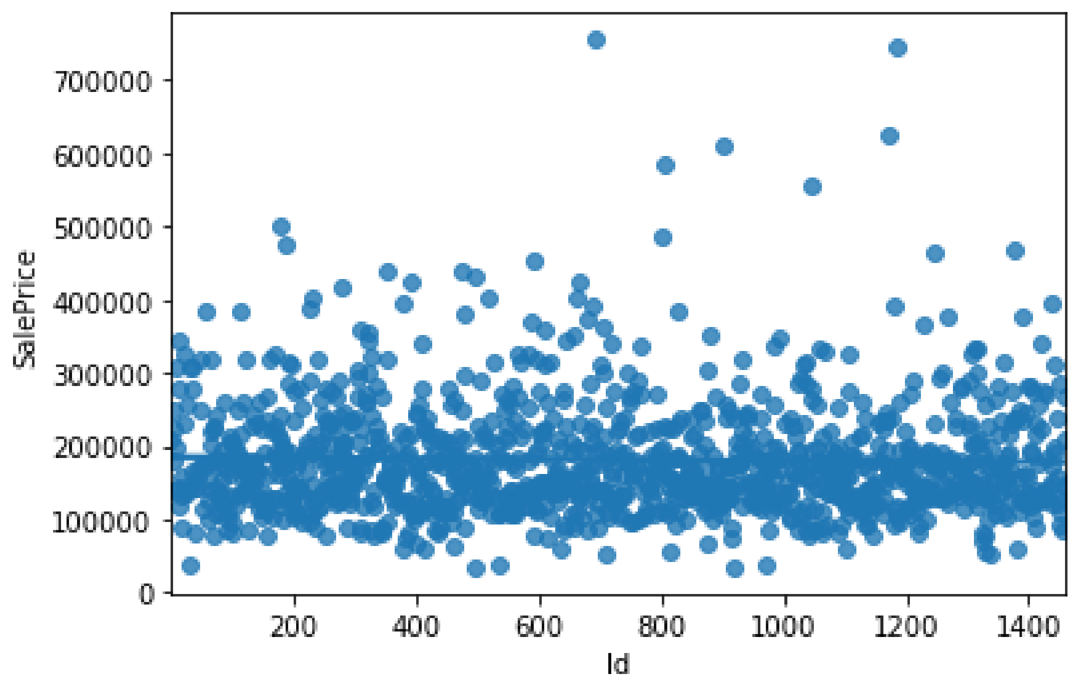
EDA

ID VS SALE PRICE

```
<seaborn.axisgrid.FacetGrid at 0x1ed8f1dbc40>
```



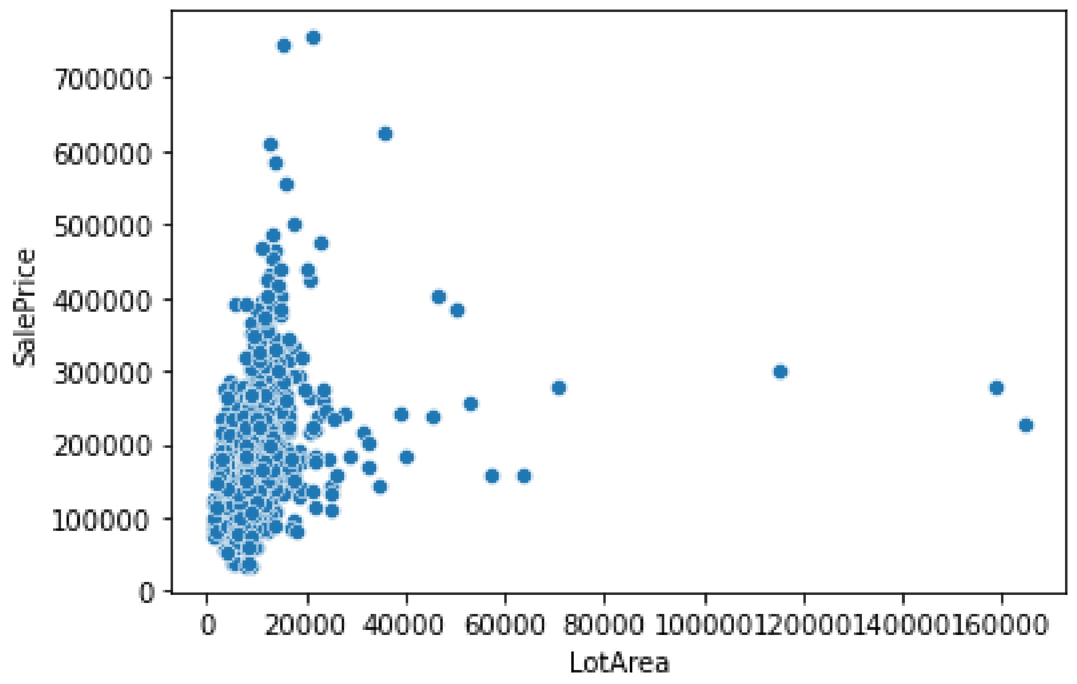
```
AxesSubplot:xlabel='Id', ylabel='SalePrice'>
```



Sale price is depended on id although there is some fluctuation still more value of id is more value of sell price.

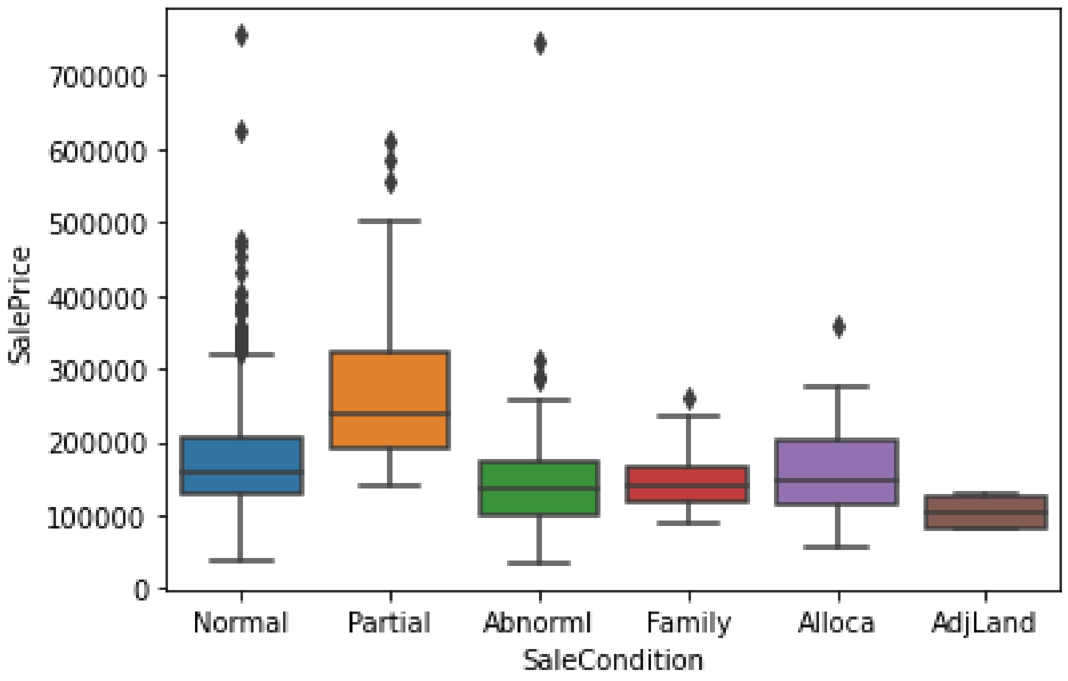
Lot Area Vs Sale Price

```
AxesSubplot:xlabel='LotArea', ylabel='SalePrice'>
```



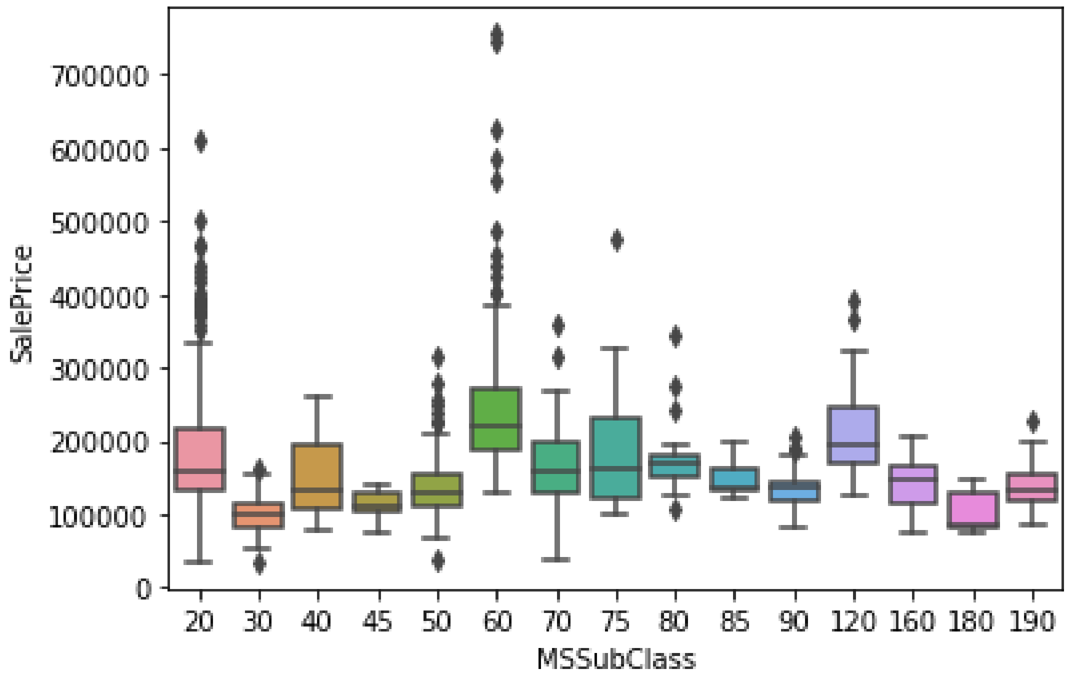
So now I have graph of LotArea vs SalePrice.

Sale Condition vs Sale Price

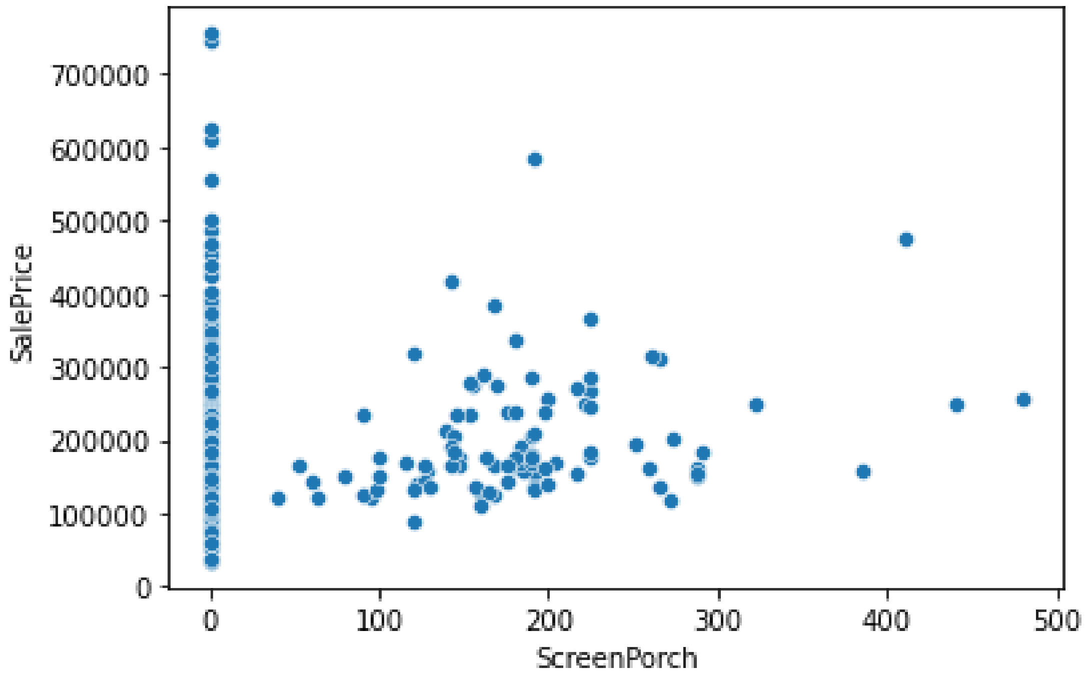


Sale price is lower when house condition is abnormal or it is sailing to family members, or adjust land. Sale price not much fluctuate if the condition is normal.

MsSubClass Vs Sale Price

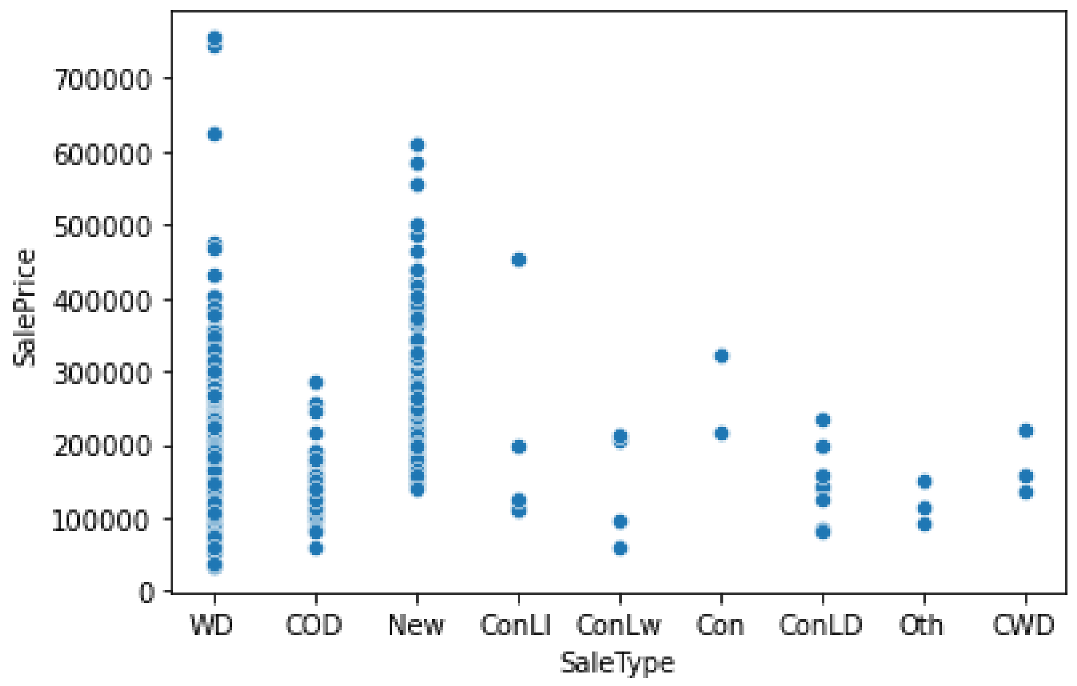


ScreenPorch Vs Sale Price



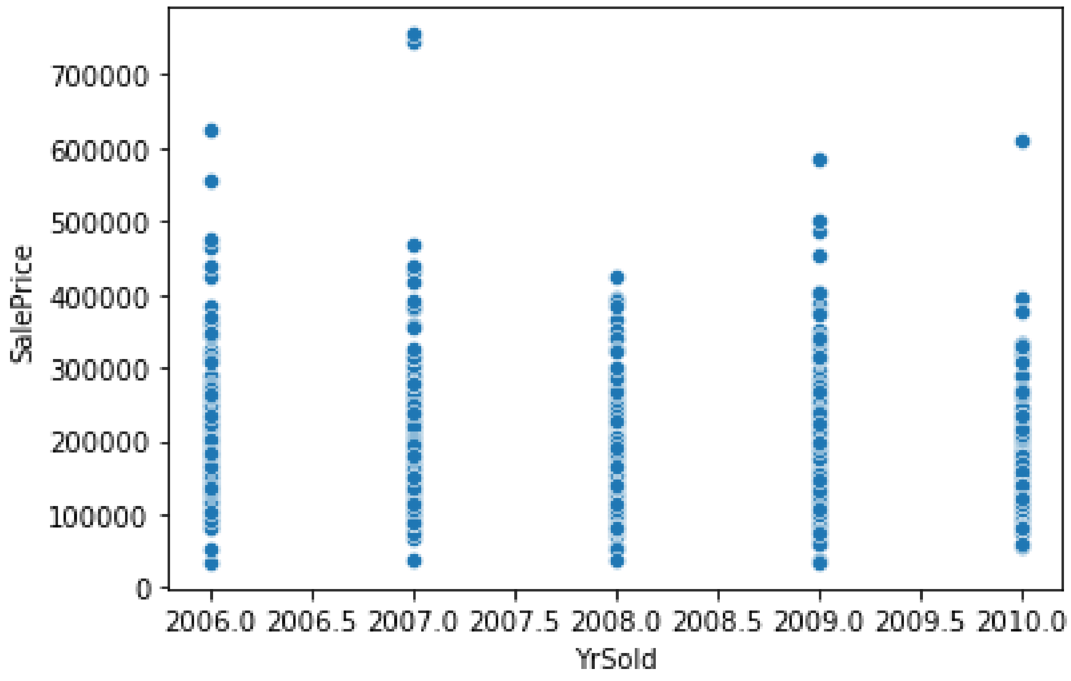
Sale Type Vs Sale Price

```
<AxesSubplot:xlabel='SaleType', ylabel='SalePrice'>
```



Price of Home just constructed and sold is not that much of fluctuate. Sale price of Other is least. Contract Low Down payment and low interest, , Contract Low Down price is lesser. Warranty Deed – Conventional most sale price.

Year sold vs Sale Price



Here is the graph of YrSold vs SalePrice.

As now most of the eda is done, now next step is all the null value should be clear and data type of all should be same.

First step is changing the data type.

Encoding

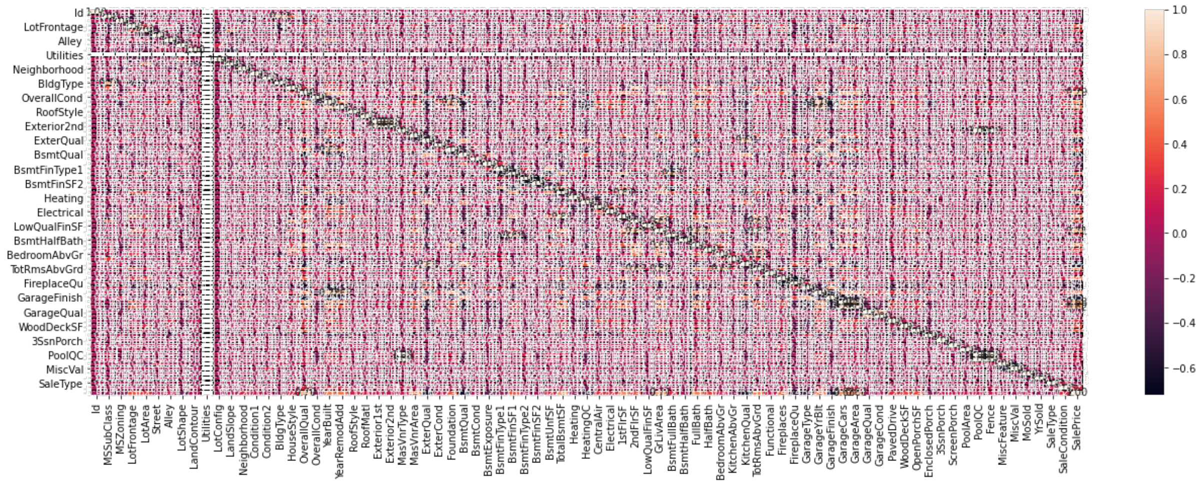
```
import sklearn

from sklearn.preprocessing import LabelEncoder

lencode=LabelEncoder()

train_data['SaleType']=lencode.fit_transform(train_data['SaleType'])
```

Like this.

After changing of datatype can process mean() procedure for change all the null value as all the data type are float or int.

Heat Map

```
<AxesSubplot:>
```



Features and Target

```
x1=train_data.drop("SalePrice",axis=1)
y=train_data['SalePrice']
```

R2 Score from Linear Regression technique is 0.835 83%.

Lasso and Rigid gave 0.834 83.4%

Cv score is 0.74

Mathematical Calculation

Calculation of Linear is
Y=W0+W1X1+W2X2+…..WnXn

$$cost(w) = 1/(2*n)\sum_{i=1}^{i=n}(y_i - \hat{y}_i)^2 + \lambda\sum_{j=1}^{j=D} w_j^2$$

Ridge Regression cost function

$$cost(w) = 1/(2*n)\sum_{i=1}^{i=n}(y_i - \hat{y}_i)^2 + \lambda\sum_{j=1}^{j=D} |w_j|$$

Lasso Regression Cost Function

For Elasticnet we got result of 0.8261

inear regression with combined L1 and L2 priors as regularizer.

Minimizes the objective function:

```
1 / (2 * n_samples) * ||y - Xw||^2_2
+ alpha * l1_ratio * ||w||_1
+ 0.5 * alpha * (1 - l1_ratio) * ||w||^2_2
```

If you are interested in controlling the L1 and L2 penalty separately, keep in mind that this is equivalent to:

```
a * ||w||_1 + 0.5 * b * ||w||_2^2
```
where:

```
alpha = a + b and l1_ratio = a / (a + b)
```
The parameter l1_ratio corresponds to alpha in the glmnet R package while alpha corresponds to the lambda parameter in glmnet. Specifically, l1_ratio = 1 is the lasso penalty. Currently, l1_ratio <= 0.01 is not reliable, unless you supply your own sequence of alpha.

Now I use Random Forest regression technique

And got

```
R2 Score: 84.09689126428151
Cross Val Score: 84.22248296942365
```

R 2 = 1 − sum squared regression (SSR) total sum of squares (SST) , = 1 − ∑ (y i − y i ^) 2 ∑ (y i − y ¯) 2 . The sum squared regression is the sum of the residuals squared, and the total sum of squares is the sum of the distance the data is away from the mean all squared.

Model Framing

```
import pickle
filename = 'housepriceprediction.pkl'
pickle.dump(rf, open(filename, 'wb'))
loaded_model = pickle.load(open('housepriceprediction.pkl', 'rb'))
result = loaded_model.score(x1test, ytest)
print(result)
```

```
0.840968912642815
```

Predicted	204758.25	207920.73	93262.08	....	130793.18	365769.49	355983.1	.....	116197.7
Original	204758.25	207920.73	93262.08	....	130793.18	365769.49	355983.1	.....	116197.7

2 rows × 351 columns

CONCLUSION

So Model is successfully created, selling price for test data can be generated by using this model.

The data set has lots of null value and also data type is different, we cannot null value if the data type is object, so first I changes the data set and then I handle the null value.

But the problem is as I put mean value instead of null, data frame gone fluctuated. Its effect my model result , although I done the overfitting bye using lasso, ridge, and also random forest grid search still my model gave 84% r2 score.

Result should be higher if data set has less null value.

THANK YOU