



HOUSING PRICE PRDICTON PROJECT

Submitted by:

RAHUL SINGH

ACKNOWLEDGMENT

Thanks for giving me the opportunity to work in FlipRobo Technologies as Intern and would like to express my gratitude to Data Trained Institute as well for trained me in Data Science Domain. This helps me to do my projects well and understand the concepts.

Resources used – Seaborn.pydata, matplotlib.org for visualization, Scikit-learn for machine learning algorithms, Blogs for conceptual referring.

INTRODUCTION

- **Business Problem Framing**

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?

- **Conceptual Background of the Domain Problem**

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. The company is looking at prospective properties to buy houses to enter the market.

- **Review of Literature**

I analysed the given data and checked which attributes are performing well to predict the sale price. I dropped some columns which are not useful to train the model and filled null values in the given trained data. Removed skewness and outliers from the data, did an encoding for categorical data to train the model.

- **Motivation for the Problem Undertaken**

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.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem
- Here our target variable is the “Sale Price” and need to predict the prices of the house. As the data is continuous and our problem is Regression.
- Data Sources and their formats

The Data is provided by Flip Robo Technologies, and it has Train and Test Data Set and need to train our data in Train dataset and need to load the Test dataset to make the predictions. Data is having Null values and We need to treat the missing values and can be discussed in pre-processing.

- Data Preprocessing Done
The data set contains null values and handled those null values with scikit-learn imputation pre-processing. Dropped columns 'Id'(unique number), in columns where the data set contains nearly 80% null values 'PoolQC', 'Fence', 'MiscFeature', etc.

- Data Inputs- Logic- Output Relationships

Mainly the material used for the total constructions (exterior, foundation, interior), Garage (quality, basement height) and the age of the house are the main things that decide the house sale price.

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

By looking at the data set, I understood that the main factor determining the price of the house depends on the age and quality of the house .

- Hardware and Software Requirements and Tools Used

Model training was done on Jupiter Notebook.

Kernel Version is Python3.

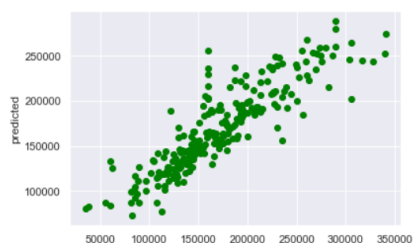
Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)
I used seaborn to analyse the categorical and numerical data to check skewness and outliers. With the function `df.describe()` we can check the descriptive statistics of all numerical columns.
- Testing of Identified Approaches (Algorithms)
RandomForest
KNeighbour
AdaBoost
Gradient Boost
Linear Regression
- Run and Evaluate selected models

```
from sklearn.ensemble import RandomForestRegressor  
  
rfr = RandomForestRegressor()  
rfr.fit(x_train,y_train)  
y_pred = rfr.predict(x_test)  
scr_rfr = cross_val_score(rfr,x,y,cv=5)  
  
print("r2_Score", r2_score(y_test,y_pred))  
print("CV Score", scr_rfr.mean())  
print("MSE",mean_squared_error(y_test,y_pred))  
print("RMSE",np.sqrt(mean_squared_error(y_test,y_pred)))  
print("Train Score", rfr.score(x_train,y_train))  
print("Test Score", rfr.score(x_test,y_test))
```

```
r2_Score 0.7764702961347599  
CV Score 0.7475715208781862  
MSE 770093879.3082483  
RMSE 27750.565387181723  
Train Score 0.9628173825395469  
Test Score 0.7764702961347599
```

```
[71]: plt.scatter(y_test,y_pred, color = 'green') #Scatter Matrix for Actual VS predicted for the model  
plt.xlabel("Actual")  
plt.ylabel("predicted")  
plt.show()
```



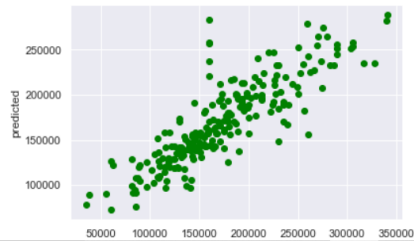
```
1 [72]: from sklearn.neighbors import KNeighborsRegressor
```

```
knr = KNeighborsRegressor(n_neighbors = 5)
knr.fit(x_train,y_train)
y_pred = knr.predict(x_test)
scr_knr = cross_val_score(knr,x,y,cv=5)

print("r2_Score", r2_score(y_test,y_pred))
print("CV Score", scr_knr.mean())
print("MSE",mean_squared_error(y_test,y_pred))
print("RMSE",np.sqrt(mean_squared_error(y_test,y_pred)))
print("Train Score", knr.score(x_train,y_train))
print("Test Score", knr.score(x_test,y_test))
```

```
r2_Score 0.7240238145058926
CV Score 0.5519686312843624
MSE 950779997.5075214
RMSE 30834.72064909169
Train Score 0.7720316309484047
Test Score 0.7240238145058926
```

```
1 [73]: plt.scatter(y_test,y_pred, color = 'green')      #Scatter Matrix for Actual VS predicted for the model
plt.xlabel("Actual")
plt.ylabel("predicted")
plt.show()
```



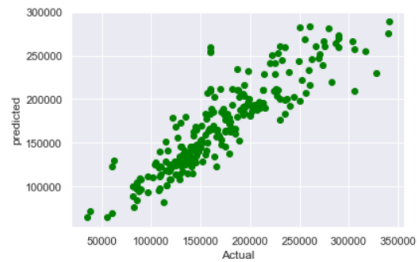
```
1 [74]: from sklearn.ensemble import GradientBoostingRegressor
```

```
gbr = GradientBoostingRegressor()
gbr.fit(x_train, y_train)
y_pred = gbr.predict(x_test)
scr_gbr = cross_val_score(gbr,x,y,cv=5)

print("r2_Score", r2_score(y_test,y_pred))
print("CV Score", scr_gbr.mean())
print("MSE",mean_squared_error(y_test,y_pred))
print("RMSE",np.sqrt(mean_squared_error(y_test,y_pred)))
print("Train Score", gbr.score(x_train,y_train))
print("Test Score", gbr.score(x_test,y_test))
```

```
r2_Score 0.7873990417763039
CV Score 0.7503756296088676
MSE 732442685.8357979
RMSE 27063.678350065387
Train Score 0.9156286309462046
Test Score 0.7873990417763039
```

```
1 [75]: plt.scatter(y_test,y_pred, color = 'green')      #Scatter Matrix for Actual VS predicted for the model
plt.xlabel("Actual")
plt.ylabel("predicted")
plt.show()
```



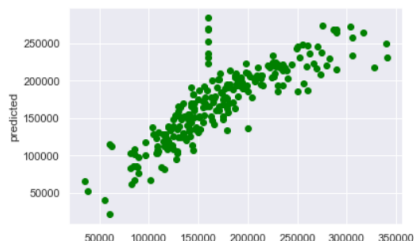
```
from sklearn.linear_model import LinearRegression

lr = LinearRegression()
lr.fit(x_train,y_train)
y_pred = lr.predict(x_test)
scr_lr = cross_val_score(lr,x,y,cv=5)

print("r2_Score", r2_score(y_test,y_pred))
print("CV Score", scr_lr.mean())
print("MSE",mean_squared_error(y_test,y_pred))
print("RMSE",np.sqrt(mean_squared_error(y_test,y_pred)))
print("Train Score", lr.score(x_train,y_train))
print("Test Score", lr.score(x_test,y_test))

r2_Score 0.7194377394441989
CV Score 0.6952036967821108
MSE 966579724.6757176
RMSE 31089.865304882194
Train Score 0.7244934918276908
Test Score 0.7194377394441989
```

```
In [77]: plt.scatter(y_test,y_pred, color = 'green') #Scatter Matrix for Actual VS predicted for the model
plt.xlabel("Actual")
plt.ylabel("predicted")
plt.show()
```



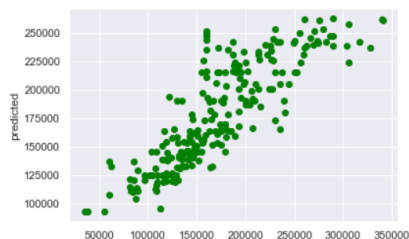
```
from sklearn.ensemble import AdaBoostRegressor

abr = AdaBoostRegressor()
abr.fit(x_train, y_train)
y_pred = abr.predict(x_test)
scr_abr = cross_val_score(abr,x,y,cv=5)

print("r2_Score", r2_score(y_test,y_pred))
print("CV Score", scr_abr.mean())
print("MSE",mean_squared_error(y_test,y_pred))
print("RMSE",np.sqrt(mean_squared_error(y_test,y_pred)))
print("Train Score", abr.score(x_train,y_train))
print("Test Score", abr.score(x_test,y_test))

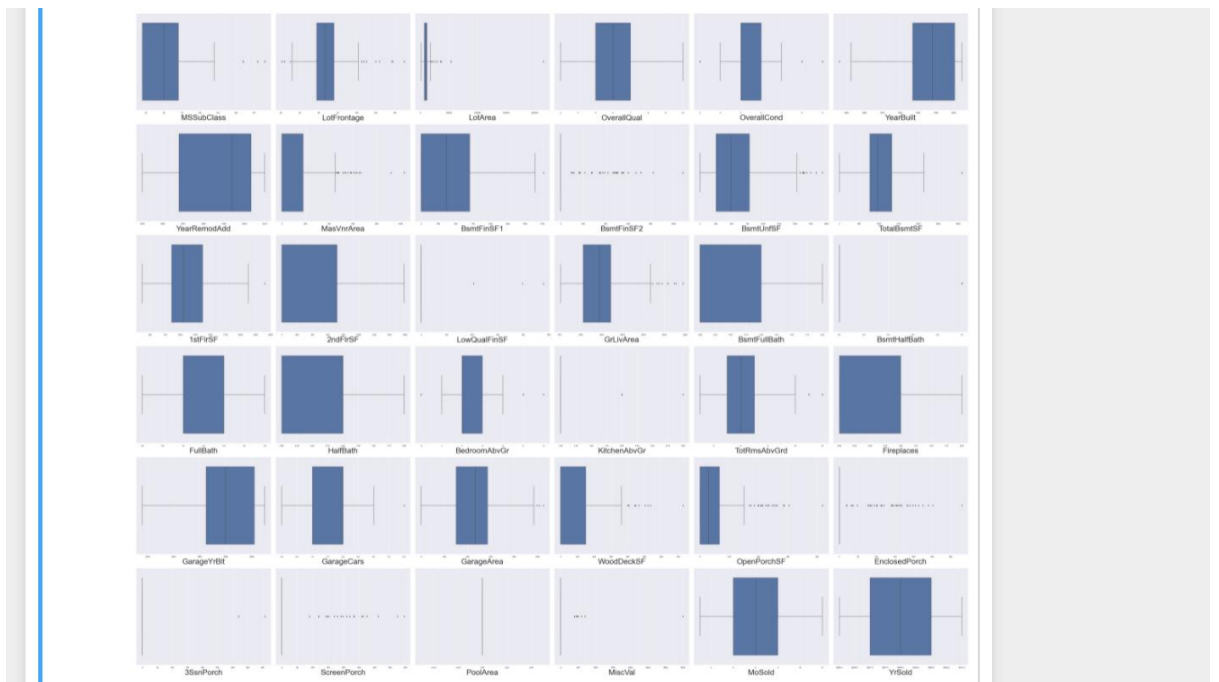
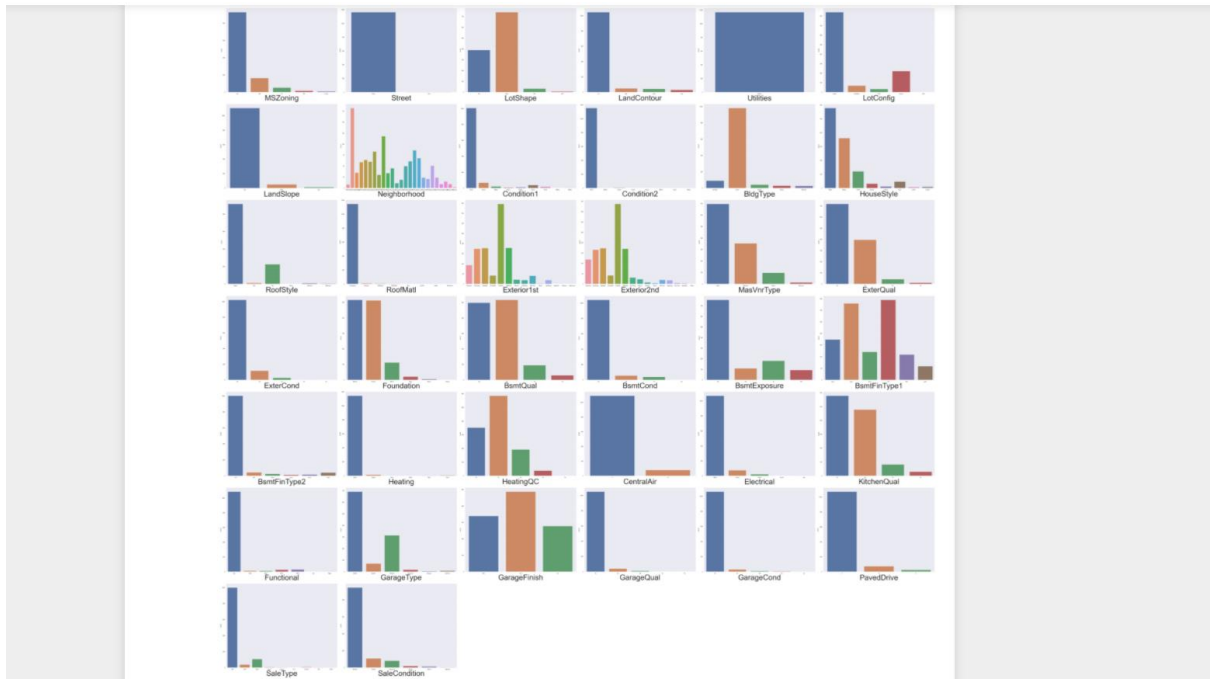
r2_Score 0.6978202777456763
CV Score 0.6862654556204426
MSE 1041055173.1389296
RMSE 32265.38661071535
Train Score 0.7479844365575083
Test Score 0.6978202777456763
```

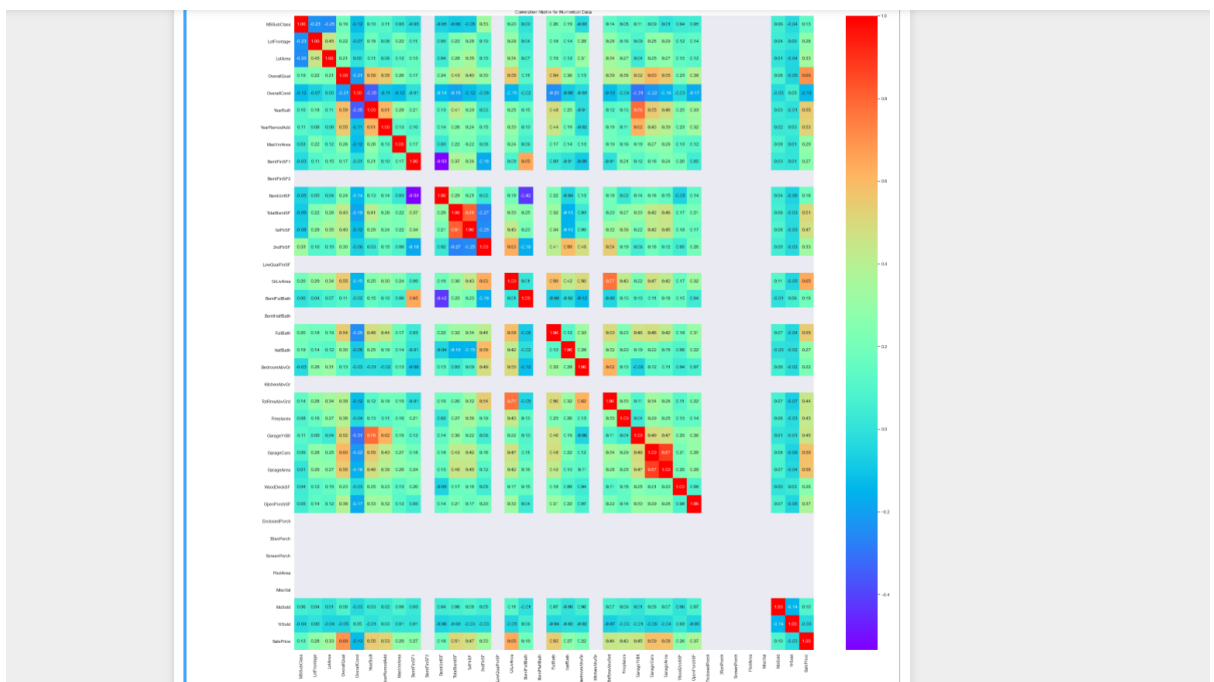
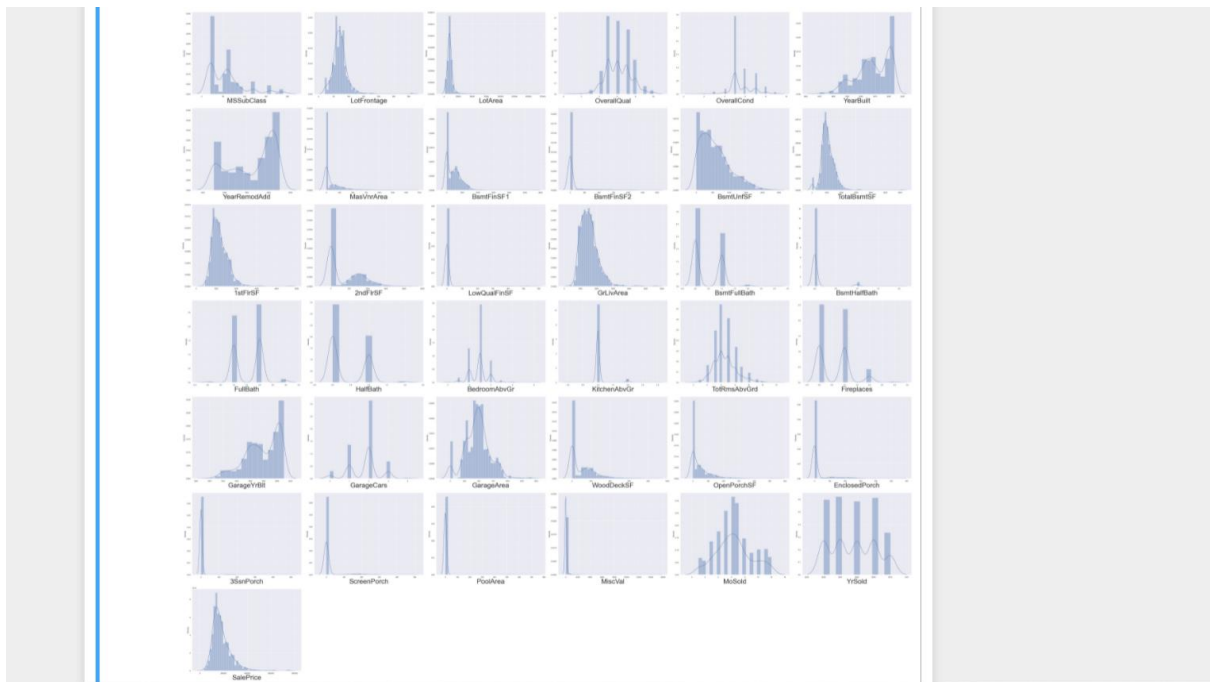
```
In [79]: plt.scatter(y_test,y_pred, color = 'green') #Scatter Matrix for Actual VS predicted for the model
plt.xlabel("Actual")
plt.ylabel("predicted")
plt.show()
```



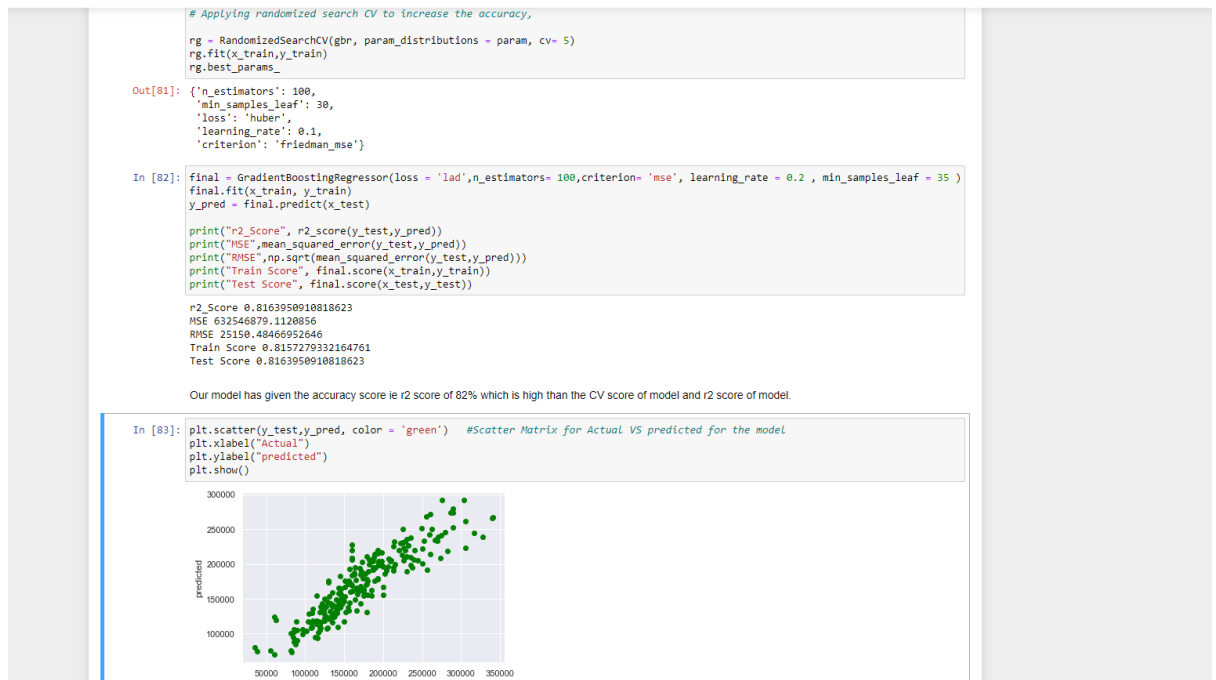
- **Key Metrics for success in solving problem under consideration**
Since this is regression model RMSE (Root mean squared error) value is considered to check how good the model is performing. From the above mentioned algorithms we can see that Gradient Boost is the Best Model.
- **Visualizations**

Box plot and Dist plot





- Interpretation of the Results



CONCLUSION

- **Key Findings and Conclusions of the Study**
- **As this project is about predicting the price of house in US market, The problem is Regression and I have used 5 algorithms to build the model and among all Gradient Boosting algorithm is the best model.**

I have used hyper parameter tuning to increase the accuracy scores as well and price which predicted was slightly varying from the actual price.

We have 2 datasets for training and testing the data and it has more columns but less rows and removed the columns which are less important using feature importance feature engineering.

So, this will help the clients to understand the variables which are important to predict the price of house and helps them to get success in their business.

- **Learning Outcomes of the Study in respect of Data Science**

Random forest and Boosting algorithms work best and Randomized search CV is faster than Grid search CV.

As this dataset has 2 data files and we need to do all pre-processing, EDA to both datasets and train dataset has target variable and test dataset will not have target variable which we need to predict it.

This is kind of different and time taking process but helps me to learn more and most important is that I am getting hands-on experience more on Data Science Concepts.

- **Limitations of this work and Scope for Future Work**

If there is skewness in the given data first we have to apply log then square root transformation taking the threshold value as ± 0.5 , if these both transformation are not working good, then the distribution of the data in that column is not uniform. For such cases we have to build model keeping skewed data.

The world is shaping according to the latest trends and one such trend in the modern world is Data Scientists. It is one of the most sought-after career options for today's youth. From multinational companies to small startups, every organization requires a Data Scientist for proper usage of the huge amount of data it generates and stores. There is a broad scope of Data Science in the present and future scenarios.