House Rent Prediction

Naman Pande
Department of computer science
PES University
Bengaluru,India
pandenaman007@gmail.com

Chetana Ninganagouda Patil
Department of computer science
PES University
Bengaluru,India

chetananpatil2002@gmail.com

***Abstract*—We are working on house rent prediction dataset. In this dataset we are going to find out the city which has more rent. By using linear regression and gradient boosting regressor.**

For this we used EDA process and data visualization, linear regression, gradient boosting regressor. We used python programming language.

I. Introduction

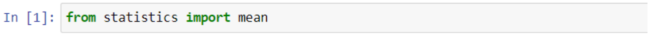
The dataset we used for this prediction is from magic brick.com .By considering the dataset we are going to predict the house value based on area locality which area has maximum rent or which has low rent . By using numpy, pandas, seaborn library we are going to visualize the plots between dependent and independent vsrisbles. By using linear regression we are going to split data into training and testing data. And we going to calculate the linear regression score to know the percentage of correct prediction. By using Gradient boosting regressor we are going to get difference between the actual and predicted dataset values.

Our dataset contains:

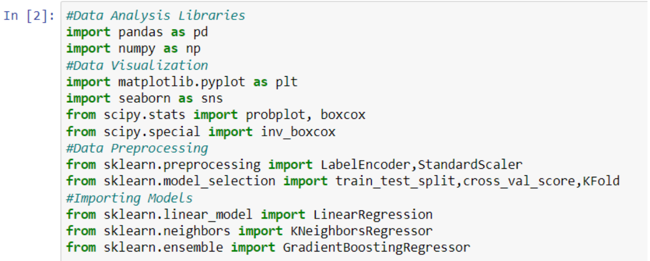
- Posted on
- BHK
- Rent
- Size
- Floor
- Area type
- Area locality
- City
- Furnishing status
- Tenant preferred
- Bathroom
- Point of contact

II. EDA+Data visualisation

A. Importing required python libraries

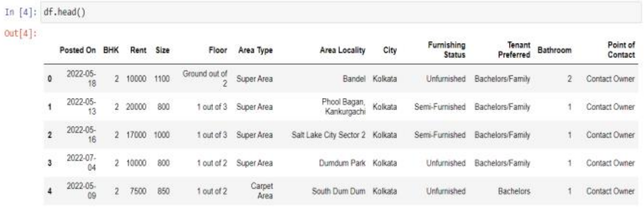


```
In [1]: from statistics import mean
```



```
In [2]: #Data Analysis Libraries
        import pandas as pd
        import numpy as np
        #Data Visualization
        import matplotlib.pyplot as plt
        import seaborn as sns
        from scipy.stats import probplot, boxcox
        from scipy.special import inv_boxcox
        #Data Preprocessing
        from sklearn.preprocessing import LabelEncoder,StandardScaler
        from sklearn.model_selection import train_test_split,cross_val_score,KFold
        #Importing Models
        from sklearn.linear_model import LinearRegression
        from sklearn.neighbors import KNeighborsRegressor
        from sklearn.ensemble import GradientBoostingRegressor
```

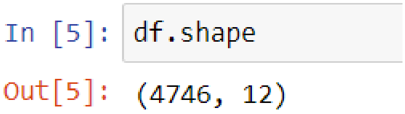
B. Reading .csv file



```
In [4]: df.head()
Out[4]:
```

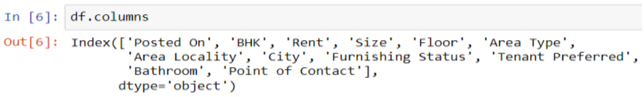
	Posted On	BHK	Rent	Size	Floor	Area Type	Area Locality	City	Furnishing Status	Tenant Preferred	Bathroom	Point of Contact
0	2022-05-18	2	10000	1100	Ground out of 2	Super Area	Bandel	Kolkata	Unfurnished	Bachelors/Family	2	Contact Owner
1	2022-05-13	2	20000	800	1 out of 3	Super Area	Phool Bagan, Kankurgachi	Kolkata	Semi-Furnished	Bachelors/Family	1	Contact Owner
2	2022-05-16	2	17000	1000	1 out of 3	Super Area	Salt Lake City Sector 2	Kolkata	Semi-Furnished	Bachelors/Family	1	Contact Owner
3	2022-07-04	2	10000	800	1 out of 2	Super Area	Dumdum Park	Kolkata	Unfurnished	Bachelors/Family	1	Contact Owner
4	2022-05-09	2	7500	850	1 out of 2	Carpet Area	South Dum Dum	Kolkata	Unfurnished	Bachelors	1	Contact Owner

C. Getting number of rows, columns

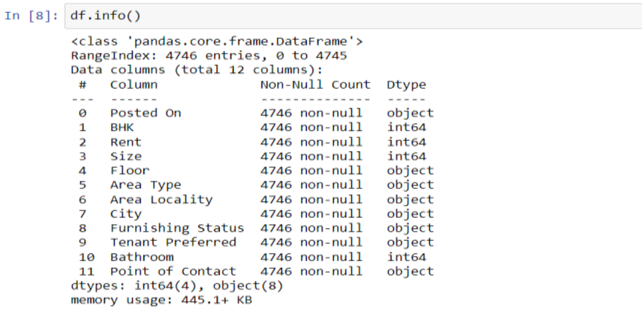


```
In [5]: df.shape

Out[5]: (4746, 12)
```

D. Getting column names and their info

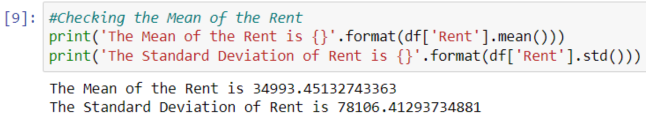


```
In [6]: df.columns

Out[6]: Index(['Posted On', 'BHK', 'Rent', 'Size', 'Floor', 'Area Type',
               'Area Locality', 'City', 'Furnishing Status', 'Tenant Preferred',
               'Bathroom', 'Point of Contact'],
              dtype='object')
```



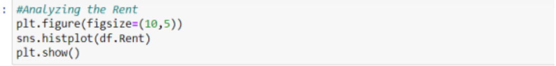
```
In [8]: df.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 4746 entries, 0 to 4745
        Data columns (total 12 columns):
         #   Column             Non-Null Count  Dtype 
        ---  ------             --------------  ----- 
         0   Posted On          4746 non-null   object
         1   BHK                4746 non-null   int64 
         2   Rent               4746 non-null   int64 
         3   Size               4746 non-null   int64 
         4   Floor              4746 non-null   object
         5   Area Type          4746 non-null   object
         6   Area Locality      4746 non-null   object
         7   City               4746 non-null   object
         8   Furnishing Status  4746 non-null   object
         9   Tenant Preferred   4746 non-null   object
         10  Bathroom           4746 non-null   int64 
         11  Point of Contact   4746 non-null   object
        dtypes: int64(4), object(8)
        memory usage: 445.1+ KB
```

E. Finding mean and standard deviation of rent

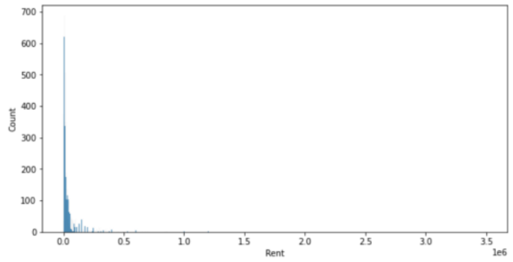


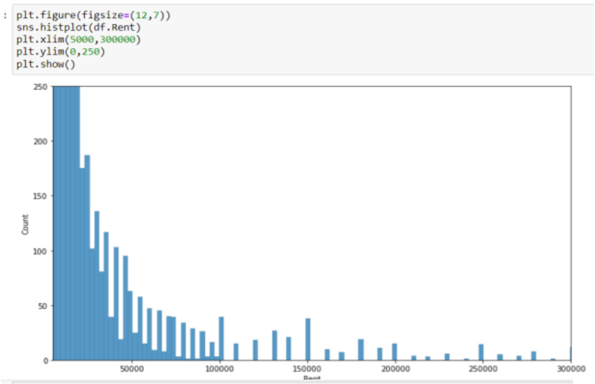
```
[9]: #Checking the Mean of the Rent
     print('The Mean of the Rent is {}'.format(df['Rent'].mean()))
     print('The Standard Deviation of Rent is {}'.format(df['Rent'].std()))

     The Mean of the Rent is 34993.45132743363
     The Standard Deviation of Rent is 78106.41293734881
```

F. For analysing rent we plotted charts using histogram, boxplot to know about outliers.

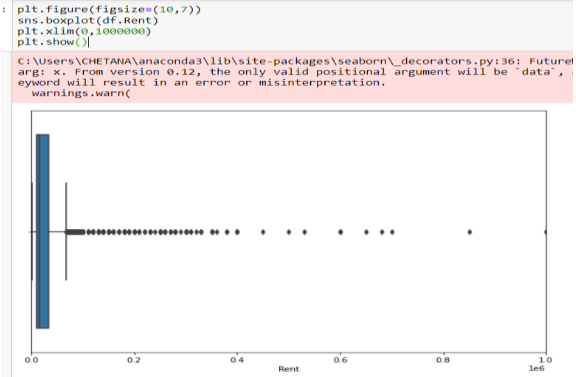


```
: #Analyzing the Rent
  plt.figure(figsize=(10,5))
  sns.histplot(df.Rent)
  plt.show()
```





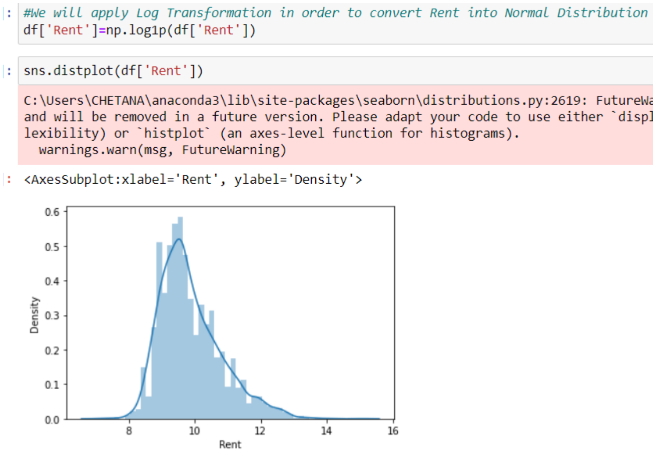
```
: plt.figure(figsize=(12,7))
  sns.histplot(df.Rent)
  plt.xlim(5000,300000)
  plt.ylim(0,250)
  plt.show()
```



```
: plt.figure(figsize=(10,7))
  sns.boxplot(df.Rent)
  plt.xlim(0,1000000)
  plt.show()
```

```
C:\Users\CHETANA\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Future
arg: x. From version 0.12, the only valid positional argument will be `data`, 
eyword will result in an error or misinterpretation.
  warnings.warn(
```

G. We applied log and square transformation to get normal distribution of data.



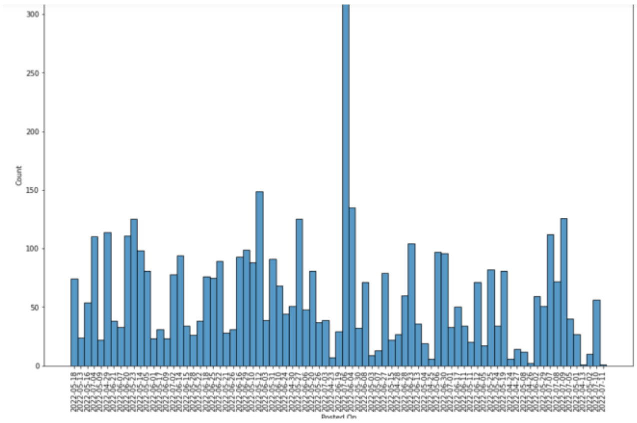
```
: #We will apply Log Transformation in order to convert Rent into Normal Distribution
  df['Rent']=np.log1p(df['Rent'])
```

```
: sns.distplot(df['Rent'])
```

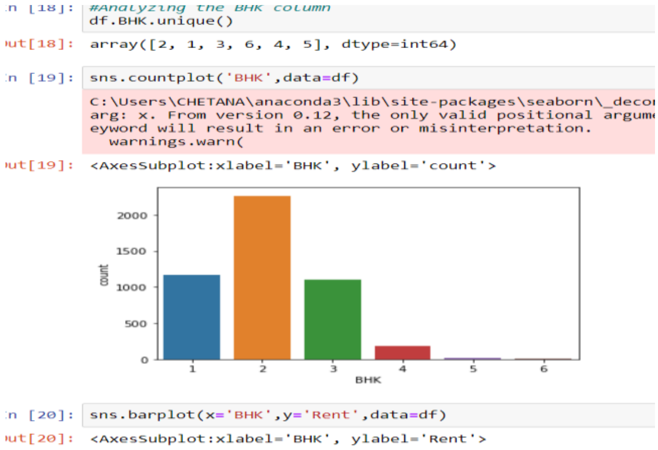
```
C:\Users\CHETANA\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWa
and will be removed in a future version. Please adapt your code to use either `displ
lexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
: <AxesSubplot:xlabel='Rent', ylabel='Density'>
```

H. To get detail about Posted on column we plotted histogram.



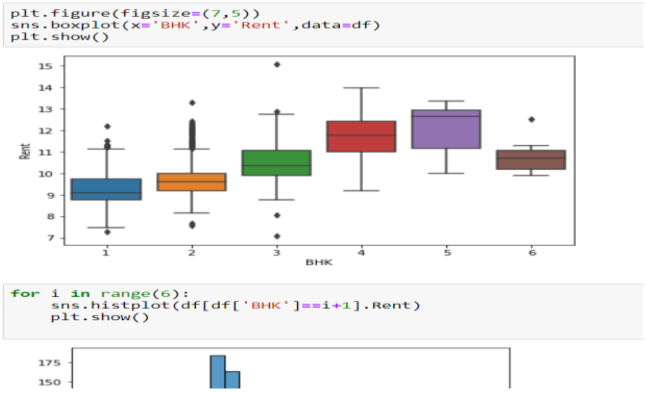
I. Details of BHK by plotting countplot.



```
In [18]: #Analyzing the BHK column
         df.BHK.unique()

Out[18]: array([2, 1, 3, 6, 4, 5], dtype=int64)

In [19]: sns.countplot('BHK',data=df)

         C:\Users\CHETANA\anaconda3\lib\site-packages\seaborn\_decor
         arg: x. From version 0.12, the only valid positional argume
         eyword will result in an error or misinterpretation.
           warnings.warn(

Out[19]: <AxesSubplot:xlabel='BHK', ylabel='count'>
```

```
In [20]: sns.barplot(x='BHK',y='Rent',data=df)

Out[20]: <AxesSubplot:xlabel='BHK', ylabel='Rent'>
```

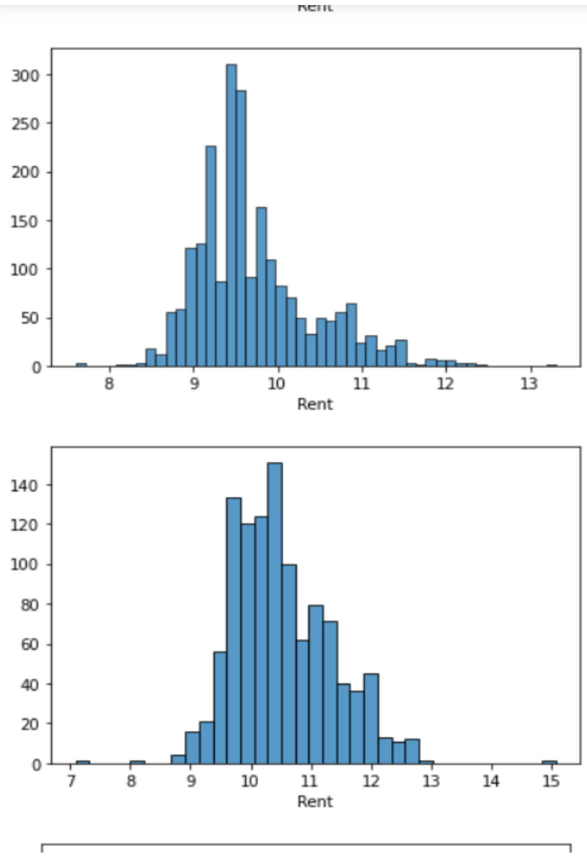
J. Boxplot with respect to BHK and rent



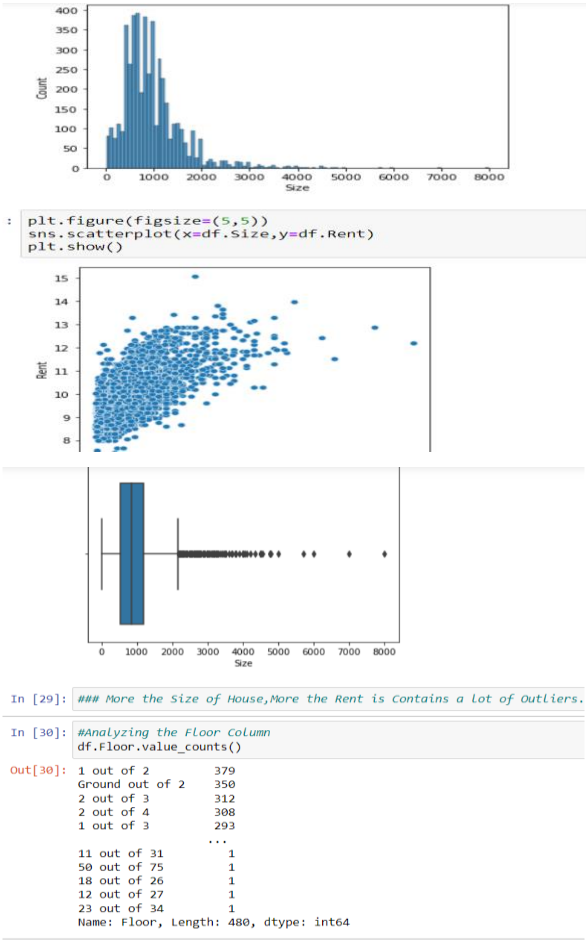
```
plt.figure(figsize=(7,5))
sns.boxplot(x='BHK',y='Rent',data=df)
plt.show()
```

```
for i in range(6):
    sns.histplot(df[df['BHK']==i+1].Rent)
    plt.show()
```

K. Histogram of BHK



L. Scatterplot between size and rent.

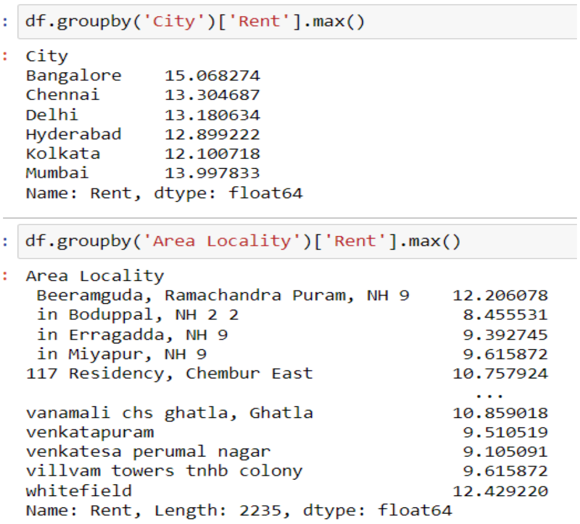


```
: plt.figure(figsize=(5,5))
  sns.scatterplot(x=df.Size,y=df.Rent)
  plt.show()
```

```
In [29]: ### More the Size of House,More the Rent is Contains a lot of Outliers.
```

```
In [30]: #Analyzing the Floor Column
         df.Floor.value_counts()

Out[30]: 1 out of 2          379
         Ground out of 2     350
         2 out of 3          312
         2 out of 4          308
         1 out of 3          293
                            ...
         11 out of 31          1
         50 out of 75          1
         18 out of 26          1
         12 out of 27          1
         23 out of 34          1
         Name: Floor, Length: 480, dtype: int64
```

M. finding which area has maximum rent

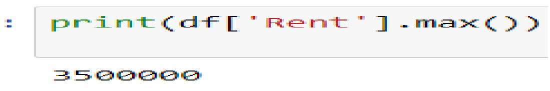


```
: df.groupby('City')['Rent'].max()

: City
  Bangalore    15.068274
  Chennai      13.304687
  Delhi        13.180634
  Hyderabad    12.899222
  Kolkata      12.100718
  Mumbai       13.997833
  Name: Rent, dtype: float64
```

```
: df.groupby('Area Locality')['Rent'].max()

: Area Locality
   Beeramguda, Ramachandra Puram, NH 9    12.206078
   in Boduppal, NH 2 2                     8.455531
   in Erragadda, NH 9                      9.392745
   in Miyapur, NH 9                        9.615872
  117 Residency, Chembur East             10.757924
                                            ...
  vanamali chs ghatla, Ghatla             10.859018
  venkatapuram                             9.510519
  venkatesa perumal nagar                  9.105091
  villvam towers tnhb colony               9.615872
  whitefield                              12.429220
  Name: Rent, Length: 2235, dtype: float64
```

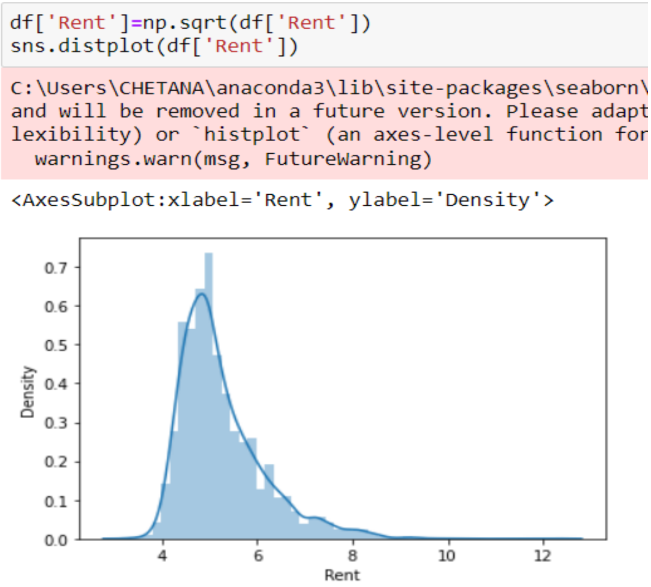
Bengaluru city has maximum rent.

Whitefield is having highest rent under area locality.



```
: print(df['Rent'].max())

  3500000
```

N. After applying squaring transformation to get normal distribution.

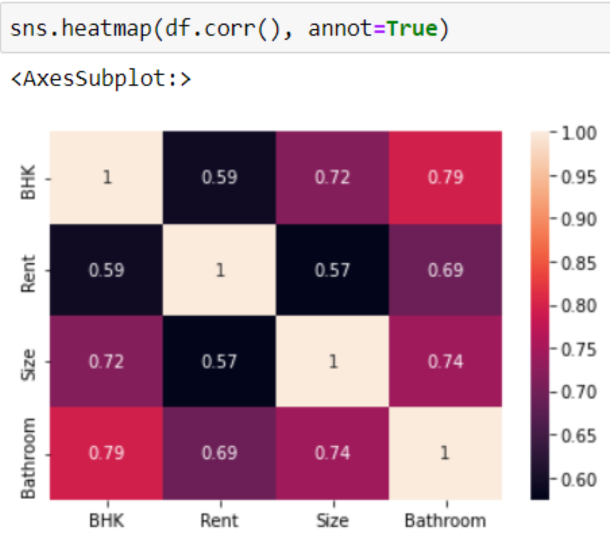


```
df['Rent']=np.sqrt(df['Rent'])
sns.distplot(df['Rent'])
```

```
C:\Users\CHETANA\anaconda3\lib\site-packages\seaborn\
and will be removed in a future version. Please adapt
lexibility) or `histplot` (an axes-level function for
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='Rent', ylabel='Density'>
```

O. Heatmap to know the correlation between the variables.

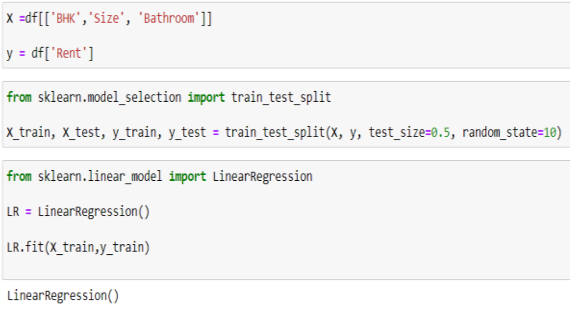


```
sns.heatmap(df.corr(), annot=True)
```

```
<AxesSubplot:>
```

	BHK	Rent	Size	Bathroom
BHK	1	0.59	0.72	0.79
Rent	0.59	1	0.57	0.69
Size	0.72	0.57	1	0.74
Bathroom	0.79	0.69	0.74	1

P. Splitting dataset into training, testing dataset.



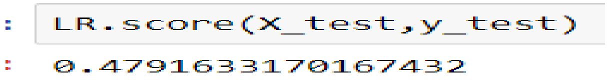
```
X =df[['BHK','Size', 'Bathroom']]

y = df['Rent']
```

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5, random_state=10)
```

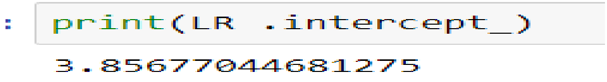
```
from sklearn.linear_model import LinearRegression

LR = LinearRegression()

LR.fit(X_train,y_train)
```

```
LinearRegression()
```

Q. Getting score and intercept of linear regression.

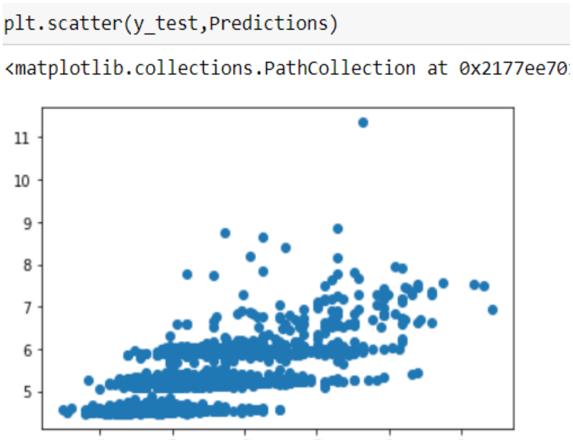


```
: LR.score(X_test,y_test)

: 0.4791633170167432
```



```
: print(LR .intercept_)

  3.85677044681275
```

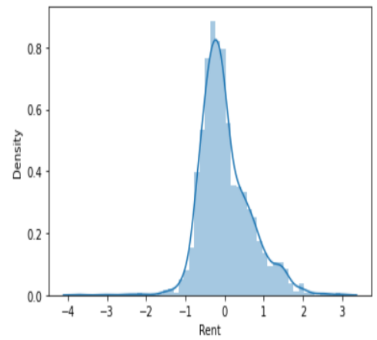
R. Scatterplot of rent

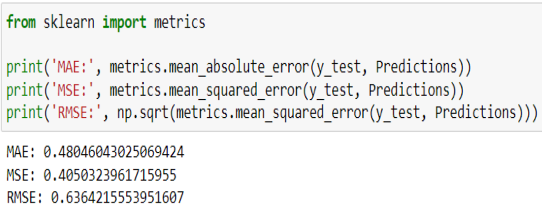


```
plt.scatter(y_test,Predictions)
```

```
<matplotlib.collections.PathCollection at 0x2177ee70
```

S. Calculating MAE,MSE, RMSE.

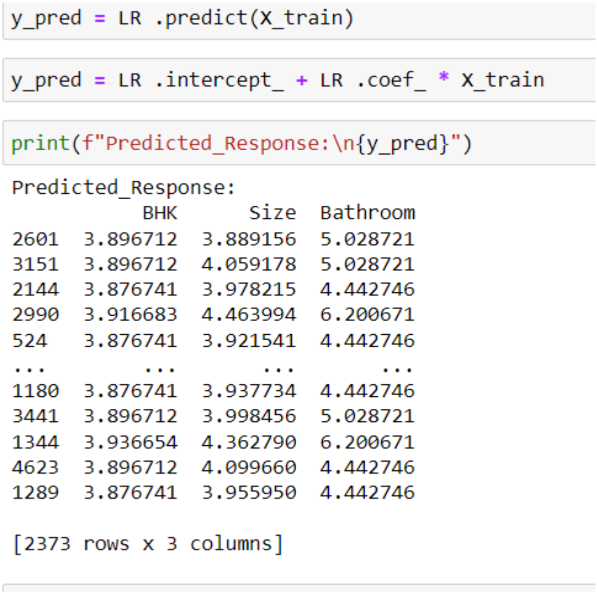




```
from sklearn import metrics

print('MAE:', metrics.mean_absolute_error(y_test, Predictions))
print('MSE:', metrics.mean_squared_error(y_test, Predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, Predictions)))
```

```
MAE: 0.48046043025069424
MSE: 0.4050323961715955
RMSE: 0.6364215553951607
```

T. Predicted response



```
y_pred = LR .predict(X_train)
```

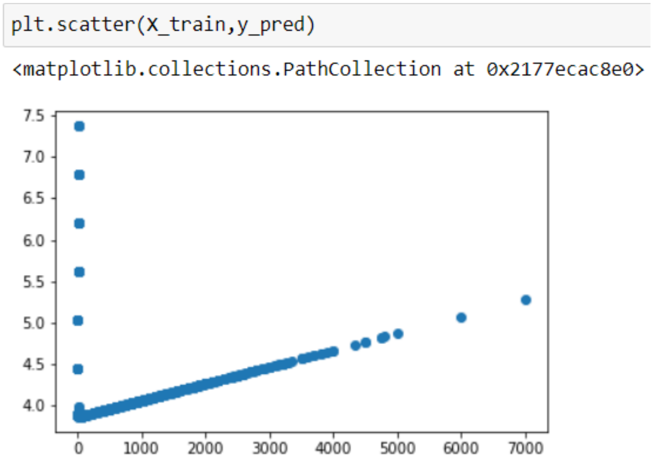
```
y_pred = LR .intercept_ + LR .coef_ * X_train
```

```
print(f"Predicted_Response:\n{y_pred}")
```

```
Predicted_Response:
           BHK      Size  Bathroom
2601  3.896712  3.889156  5.028721
3151  3.896712  4.059178  5.028721
2144  3.876741  3.978215  4.442746
2990  3.916683  4.463994  6.200671
524   3.876741  3.921541  4.442746
...        ...       ...       ...
1180  3.876741  3.937734  4.442746
3441  3.896712  3.998456  5.028721
1344  3.936654  4.362790  6.200671
4623  3.896712  4.099660  4.442746
1289  3.876741  3.955950  4.442746

[2373 rows x 3 columns]
```

□

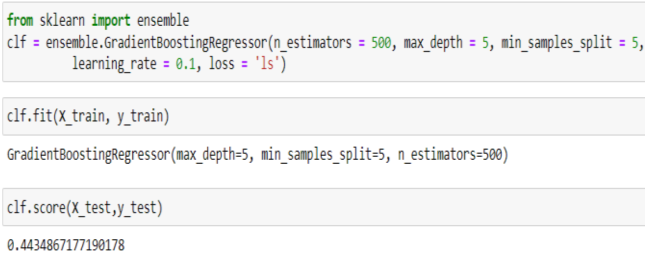
U. Scatterplot between actual data, predicted data.



```
plt.scatter(X_train,y_pred)
```

```
<matplotlib.collections.PathCollection at 0x2177ecac8e0>
```

V. by using gradient boosting regressor finding difference between actual, predicted data.



```
from sklearn import ensemble
clf = ensemble.GradientBoostingRegressor(n_estimators = 500, max_depth = 5, min_samples_split = 5,
          learning_rate = 0.1, loss = 'ls')
```

```
clf.fit(X_train, y_train)
```

```
GradientBoostingRegressor(max_depth=5, min_samples_split=5, n_estimators=500)
```

```
clf.score(X_test,y_test)
```

```
0.4434867177190178
```

Resource used for dataset

- https://www.magicbricks.com/

ACKNOWLEDGMENT

We would like to express our gratitude to Dr. Prajwala TR maám .Thank you for your guidance during literature survey and while making our project and for your teaching throughout the Data analytics elective of 5th semester .

Github link:

https://github.com/chetananpatil/Data-analytics-project