house price pred

April 6, 2023

```
[1]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     %matplotlib inline
     import matplotlib
     matplotlib.rcParams["figure.figsize"]=(20,10)
```

```
[2]: import warnings
     warnings.filterwarnings("ignore")
```

```
[3]: df=pd.read_csv("Bengaluru_House_Data.csv")
     df.head()
```

```
[3]:               area_type   availability                  location       size  \
     0  Super built-up  Area         19-Dec  Electronic City Phase II      2 BHK
     1            Plot  Area   Ready To Move          Chikka Tirupathi  4 Bedroom
     2        Built-up  Area   Ready To Move               Uttarahalli      3 BHK
     3  Super built-up  Area   Ready To Move        Lingadheeranahalli      3 BHK
     4  Super built-up  Area   Ready To Move                  Kothanur      2 BHK

        society total_sqft  bath  balcony   price
     0  Coomee        1056   2.0      1.0   39.07
     1  Theanmp       2600   5.0      3.0  120.00
     2      NaN       1440   2.0      3.0   62.00
     3  Soiewre       1521   3.0      1.0   95.00
     4      NaN       1200   2.0      1.0   51.00
```

```
[4]: df.shape
```

```
[4]: (13320, 9)
```

1 Data Cleaning

```
[5]: df["area_type"].value_counts()
```

```
[5]: Super built-up  Area    8790
     Built-up  Area          2418
     Plot  Area              2025
     Carpet  Area              87
     Name: area_type, dtype: int64
```

```
[6]: df_copy1=df.copy()
```

```
[7]: df_copy1.
      ↪drop(["availability","society","area_type","balcony"],axis=1,inplace=True)
     df_copy1.head()
```

```
[7]:                   location       size total_sqft  bath   price
     0  Electronic City Phase II      2 BHK       1056   2.0   39.07
     1          Chikka Tirupathi  4 Bedroom       2600   5.0  120.00
     2               Uttarahalli      3 BHK       1440   2.0   62.00
     3        Lingadheeranahalli      3 BHK       1521   3.0   95.00
     4                  Kothanur      2 BHK       1200   2.0   51.00
```

```
[8]: df_copy1.isnull().sum()
```

```
[8]: location       1
     size          16
     total_sqft     0
     bath          73
     price          0
     dtype: int64
```

```
[9]: median_bath=df_copy1["bath"].median()
     df_copy1.fillna(median_bath,inplace=True)
```

```
[10]: df_copy1["bath"].isnull().sum()
```

```
[10]: 0
```

```
[11]: df_copy1["size"].unique()
```

```
[11]: array(['2 BHK', '4 Bedroom', '3 BHK', '4 BHK', '6 Bedroom', '3 Bedroom',
              '1 BHK', '1 RK', '1 Bedroom', '8 Bedroom', '2 Bedroom',
              '7 Bedroom', '5 BHK', '7 BHK', '6 BHK', '5 Bedroom', '11 BHK',
              '9 BHK', 2.0, '9 Bedroom', '27 BHK', '10 Bedroom', '11 Bedroom',
              '10 BHK', '19 BHK', '16 BHK', '43 Bedroom', '14 BHK', '8 BHK',
              '12 Bedroom', '13 BHK', '18 Bedroom'], dtype=object)
```

```
[12]: df_copy1.dropna(inplace=True)
```

```
[13]: df_copy1.isnull().sum()
```

```
[13]: location      0
      size          0
      total_sqft    0
      bath          0
      price         0
      dtype: int64
```

```
[14]: # df_copy1["size_temp"]=df_copy1["size"]
      # df_copy1["size"]=df_copy1["size"].str.split(" ").str[0]
      # df_copy1["size_temp"]=df_copy1["size_temp"].str.split(" ").str[1]
      # df_copy1["size_temp"]=df_copy1["size_temp"].replace("Bedroom","1")
      # df_copy1["size_temp"]=df_copy1["size_temp"].replace("BHK","3")
      # df_copy1["size_temp"]=df_copy1["size_temp"].replace("RK","2")
      # # df_copy1["size_temp"]=df_copy1["size_temp"].replace("np.nan","0")
      # # df_copy1["size_temp"]=df_copy1["size_temp"].astype(int)
      # # df_copy1["size"].astype(int)
      # # df_copy1["size"]=df_copy1["size"]+df_copy1["size_temp"]
      # # df_copy1["size"].head()
```

```
[15]: df_copy1["BHK"]=df_copy1["size"].apply(lambda x:int(x) if type(x)== float else␣
        ↪int(x.split(" ")[0]))
      df_copy1["BHK"]=df_copy1["BHK"].astype(int)
```

```
[16]: df_copy1.head()
```

```
[16]:                    location       size total_sqft  bath   price  BHK
      0  Electronic City Phase II      2 BHK       1056   2.0   39.07    2
      1          Chikka Tirupathi  4 Bedroom       2600   5.0  120.00    4
      2               Uttarahalli      3 BHK       1440   2.0   62.00    3
      3        Lingadheeranahalli      3 BHK       1521   3.0   95.00    3
      4                  Kothanur      2 BHK       1200   2.0   51.00    2
```

```
[17]: df_copy1.BHK.unique()
```

```
[17]: array([ 2,  4,  3,  6,  1,  8,  7,  5, 11,  9, 27, 10, 19, 16, 43, 14, 12,
             13, 18])
```

```
[18]: # there is/are house with 43 bedrooms which don't feel practical
      df_copy1[df_copy1.BHK>20]
```

```
[18]:                       location        size total_sqft  bath  price  BHK
      1718  2Electronic City Phase II      27 BHK       8000  27.0  230.0   27
      4684                 Munnekollal  43 Bedroom       2400  40.0  660.0   43
```

```
[19]: df_copy1["total_sqft"].unique()
```

[19]:
```
array(['1056', '2600', '1440', …, '1133 - 1384', '774', '4689'],
      dtype=object)
```

[20]:
```
def is_float(n):
    try:
        float(x)
    except:
        return False
    return True
```

[21]:
```
(~df_copy1["total_sqft"].apply(is_float)).sum()
```

[21]:
```
13320
```

[22]:
```
def conv_sqft_to_num(n):
    tokens=n.split("-") # for value is in the form of 2000-2200
    if len(tokens)==2:
        return (float(tokens[0]) + float(tokens[1]))/2
    try:
        n=float(n)
        return n
    except:
        return None
```

[23]:
```
a=conv_sqft_to_num("90")
print(a)
```

```
90.0
```

[24]:
```
df_copy1["total_sqft"]=df_copy1["total_sqft"].apply(conv_sqft_to_num)
```

[25]:
```
df_copy1["total_sqft"].unique()
```

[25]:
```
array([1056. , 2600. , 1440. , …, 1258.5,  774. , 4689. ])
```

[26]:
```
df_copy1["total_sqft"].mean()
```

[26]:
```
1559.626693912912
```

[27]:
```
df_copy1.head()
```

[27]:
```
                   location       size  total_sqft  bath   price  BHK
0  Electronic City Phase II      2 BHK      1056.0   2.0   39.07    2
1          Chikka Tirupathi  4 Bedroom      2600.0   5.0  120.00    4
2               Uttarahalli      3 BHK      1440.0   2.0   62.00    3
3        Lingadheeranahalli      3 BHK      1521.0   3.0   95.00    3
4                  Kothanur      2 BHK      1200.0   2.0   51.00    2
```

[28]:
```
df_copy1["price_per_sqft"]=(df_copy1["price"]*100000)/df_copy1["total_sqft"]
```

[29]:
```
len(df_copy1["location"].unique())
```

[29]:
```
1306
```

[30]:
```
(df_copy1["location"].apply(is_float)).sum()
```

[30]:
```
0
```

[31]:
```
df_copy1.location=df_copy1.location.apply(lambda x: str(x).strip())
```

[32]:
```
df_copy1["location"].info()
```

```
<class 'pandas.core.series.Series'>
RangeIndex: 13320 entries, 0 to 13319
Series name: location
Non-Null Count  Dtype
--------------  -----
13320 non-null  object
dtypes: object(1)
memory usage: 104.2+ KB
```

[33]:
```
df_copy1.shape
```

[33]:
```
(13320, 7)
```

[34]:
```
location_stats=df_copy1["location"].value_counts().sort_values(ascending=False)
```

[35]:
```
len(location_stats[location_stats<=10])
```

[35]:
```
1054
```

[36]:
```
location_stats_lessthan_10=location_stats[location_stats<=10]
```

[37]:
```
len(df_copy1["location"].unique())
```

[37]:
```
1295
```

[38]:
```
df_copy1["location"]=df_copy1["location"].apply(lambda x:"others" if x in␣
 ↪location_stats_lessthan_10 else x)
```

[39]:
```
len(df_copy1["location"].unique())
```

[39]:
```
242
```

[40]:
```
df_copy1.shape
```

[40]:
```
(13320, 7)
```

[41]:
```
df_copy1[df_copy1["total_sqft"]/df_copy1["BHK"]<300]
```

[41]:
```
                   location       size  total_sqft  bath  price  BHK  \
9                    others  6 Bedroom      1020.0   6.0  370.0    6
45               HSR Layout  8 Bedroom       600.0   9.0  200.0    8
58            Murugeshpalya  6 Bedroom      1407.0   4.0  150.0    6
68      Devarachikkanahalli  8 Bedroom      1350.0   7.0   85.0    8
70                   others  3 Bedroom       500.0   3.0  100.0    3
…                       …          …           …    …      …    …
13277                others  7 Bedroom      1400.0   7.0  218.0    7
13279                others  6 Bedroom      1200.0   5.0  130.0    6
13281       Margondanahalli  5 Bedroom      1375.0   5.0  125.0    5
13303        Vidyaranyapura  5 Bedroom       774.0   5.0   70.0    5
13311      Ramamurthy Nagar  7 Bedroom      1500.0   9.0  250.0    7

       price_per_sqft
9        36274.509804
45       33333.333333
58       10660.980810
68        6296.296296
70       20000.000000
…                 …
13277    15571.428571
13279    10833.333333
13281     9090.909091
13303     9043.927649
13311    16666.666667

[744 rows x 7 columns]
```

[42]:
```
df_copy2=df_copy1[(df_copy1["total_sqft"]/df_copy1["BHK"])>=300]
df_copy2.head()
```

[42]:
```
                   location       size  total_sqft  bath   price  BHK  \
0  Electronic City Phase II      2 BHK      1056.0   2.0   39.07    2
1          Chikka Tirupathi  4 Bedroom      2600.0   5.0  120.00    4
2               Uttarahalli      3 BHK      1440.0   2.0   62.00    3
3        Lingadheeranahalli      3 BHK      1521.0   3.0   95.00    3
4                  Kothanur      2 BHK      1200.0   2.0   51.00    2

   price_per_sqft
0     3699.810606
1     4615.384615
2     4305.555556
3     6245.890861
```

```
4     4250.000000
```

[43]:
```
df_copy2.shape
```

[43]:
```
(12530, 7)
```

[44]:
```
df_copy2["price_per_sqft"].describe()
```

[44]:
```
count     12530.000000
mean       6303.979357
std        4162.237981
min         267.829813
25%        4210.526316
50%        5294.117647
75%        6916.666667
max      176470.588235
Name: price_per_sqft, dtype: float64
```

[45]:
```
# def remove_outliers(df):
#     for key,sub_df in df.groupby("location"):
#         m=np.mean(sub_df.price_per_sqft)
#         sd=np.std(sub_df.price_per_sqft)
#         df_out=sub_df[(sub_df.price_per_sqft>(m-sd)) & (sub_df.
 ↪price_per_sqft<=(m+sd))]
#         return df_out
```

[46]:
```
# df_copy3=remove_outliers(df_copy2)
```

[47]:
```
# df_copy3.shape
```

[48]:
```
IQR=6916.666667-4210.526316
lower_range=4210.526316-(1.5*IQR)
upper_range=6916.666667+(1.5*IQR)
print(lower_range)
print(upper_range)
```

```
151.3157895000004
10975.8771935
```

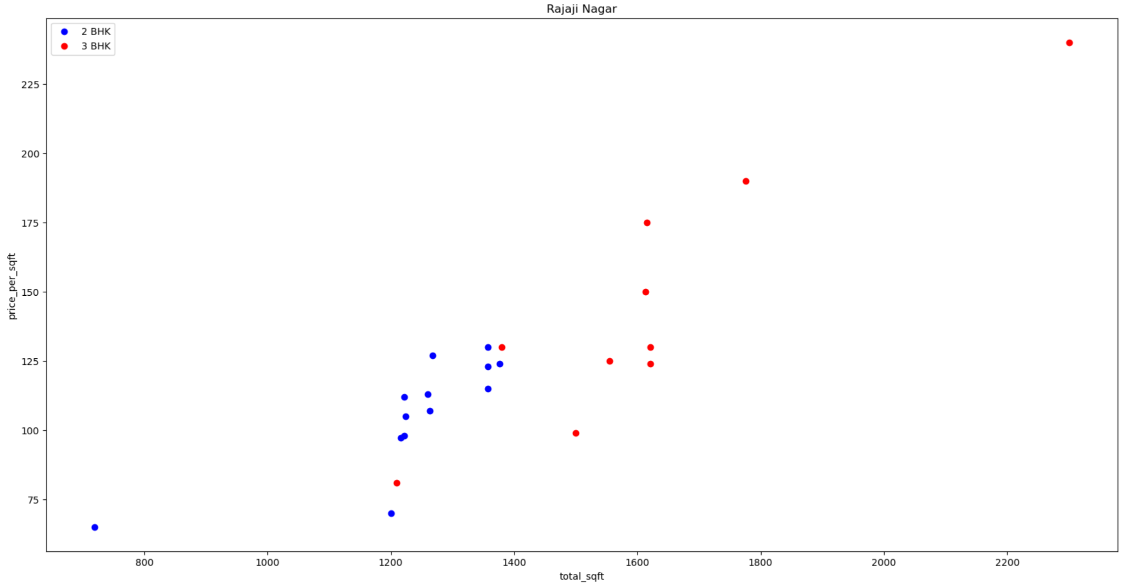
[49]:
```
df_copy2=df_copy2[(df_copy2["price_per_sqft"]>=lower_range) &␣
 ↪(df_copy2["price_per_sqft"]<=upper_range)]
df_copy2.shape
```

[49]:
```
(11523, 7)
```

[50]:
```
def plot_scatter_chart(df,location):
    bhk_2=df[(df.location==location) & (df.BHK==2)]
    bhk_3=df[(df.location==location) & (df.BHK==3)]
    plt.scatter(bhk_2.total_sqft,bhk_2.price,color="blue",label="2 BHK")
    plt.scatter(bhk_3.total_sqft,bhk_3.price,color="red",label="3 BHK")
    plt.xlabel("total_sqft")
    plt.ylabel("price_per_sqft")
    plt.title(location)
    plt.legend()
```

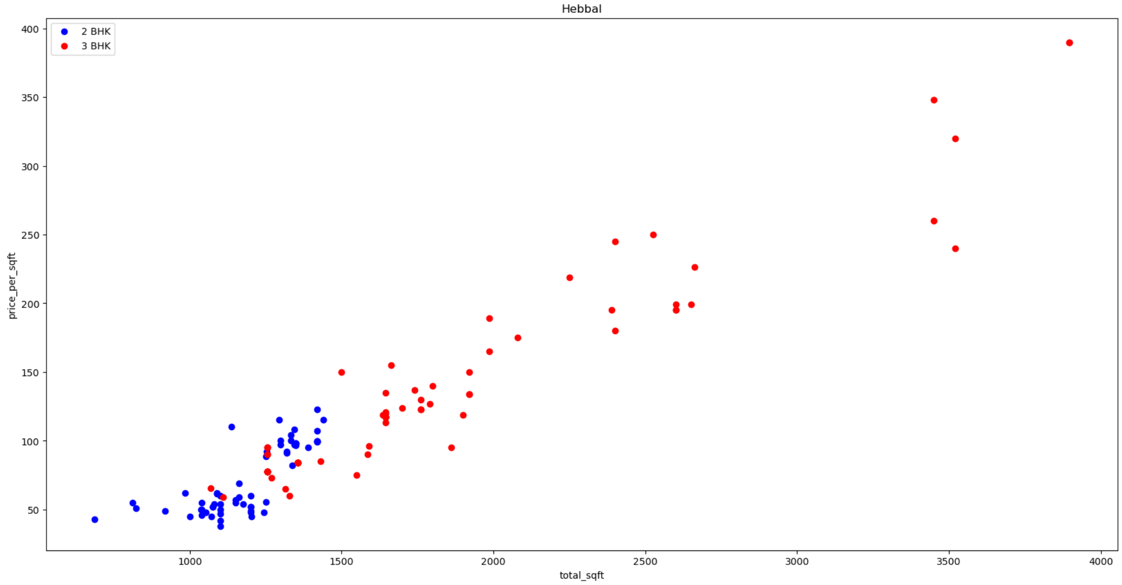
[51]:

```
plot_scatter_chart(df_copy2,"Rajaji Nagar")
```



[52]:

```
plot_scatter_chart(df_copy2,"Hebbal")
```



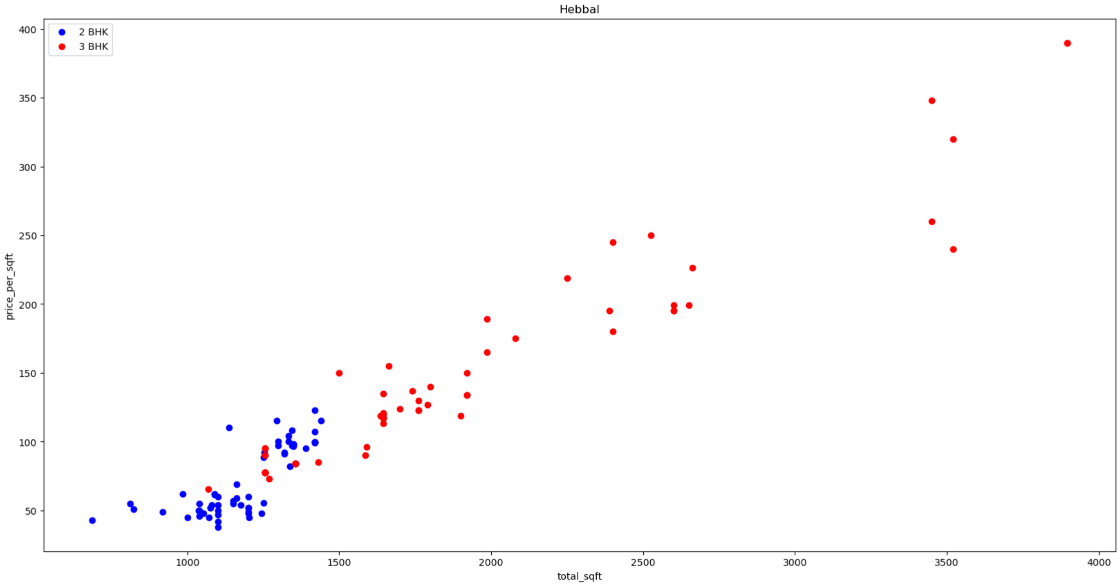
[53]:
```
# for same location there are some house with high price with less bedrooms.

def remove_bhk_outliers(df):
    exclude_indices=np.array([])
    for location,location_df in df.groupby("location"):
        bhk_stats={}
        for bhk,bhk_df in location_df.groupby("BHK"):
            bhk_stats[bhk]={
                "mean":np.mean(df.price_per_sqft),
                "std":np.std(df.price_per_sqft),
                "count":bhk_df.shape[0]
            }
        for bhk,bhk_df in location_df.groupby("BHK"):
            stats=bhk_stats.get(bhk-1)
            if stats and stats["count"]>5:
                exclude_indices=np.append(exclude_indices,bhk_df[bhk_df.
 ↪price_per_sqft<(stats["mean"])].index.values)
    return df.drop(exclude_indices,axis="index")
```

[54]:
```
df_copy3=remove_bhk_outliers(df_copy2)
```

[55]:
```
df_copy3.shape
```

[55]:
```
(7753, 7)
```

[56]:
```
plot_scatter_chart(df_copy3,"Hebbal")
```

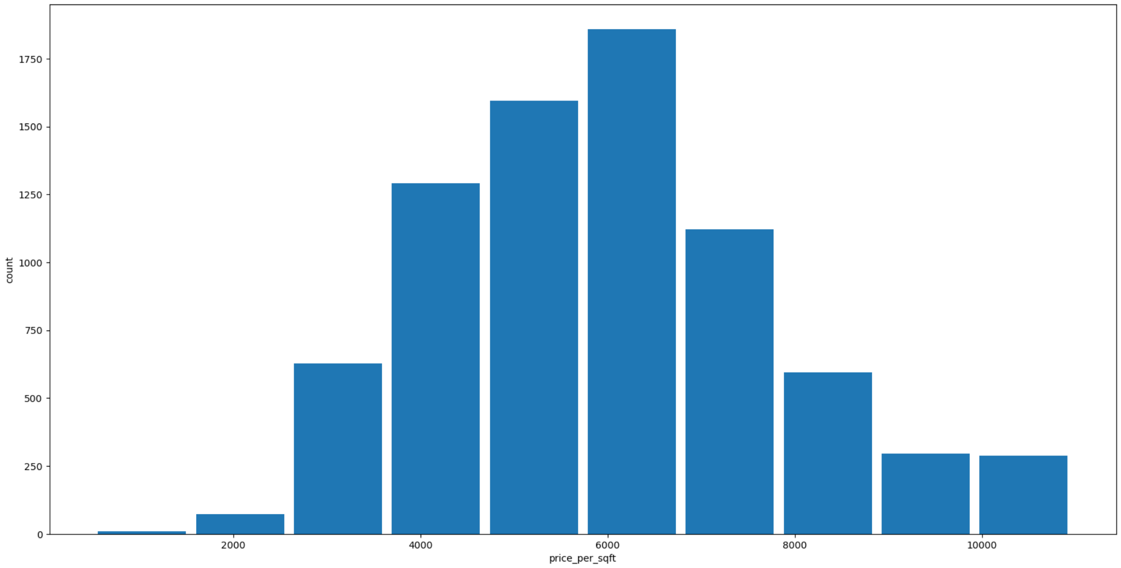


[57]:

```
plt.hist(df_copy3.price_per_sqft,rwidth=0.9)
plt.xlabel("price_per_sqft")
plt.ylabel("count")
```

[57]:

```
Text(0, 0.5, 'count')
```



[58]:

```
df_copy3.bath.unique()
```

[58]:

```
array([ 2.,  3.,  4.,  5.,  1.,  8.,  6.,  7.,  9., 12., 16., 10., 13.])
```

[59]:

```
df_copy3[df_copy3.bath>10]
```

[59]:

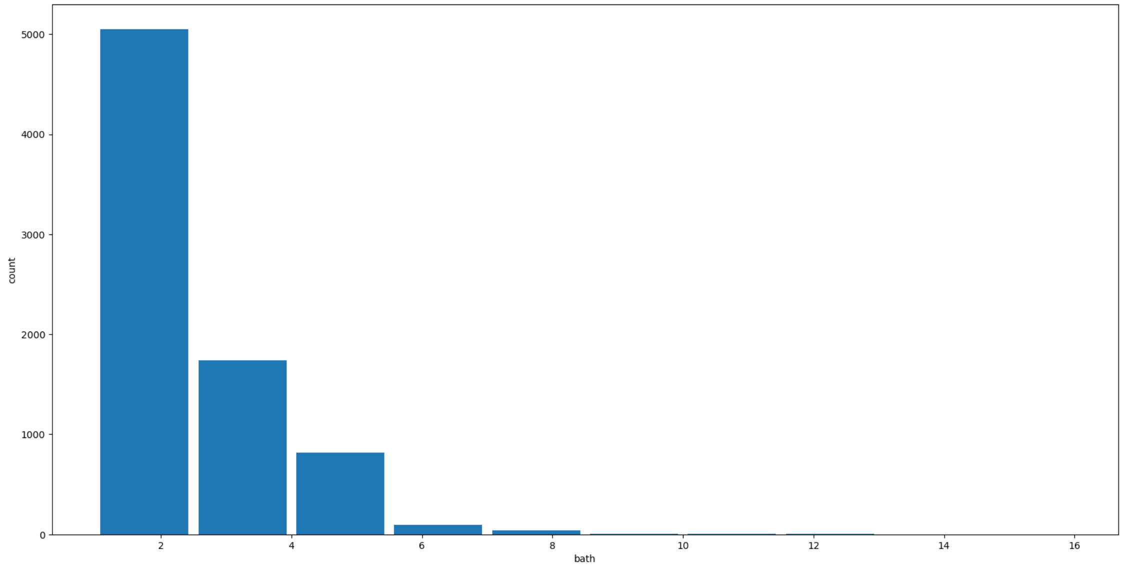
```
             location    size  total_sqft  bath  price  BHK  price_per_sqft
3096           others  10 BHK     12000.0  12.0  525.0   10     4375.000000
3609           others  16 BHK     10000.0  16.0  550.0   16     5500.000000
7979           others  11 BHK      6000.0  12.0  150.0   11     2500.000000
8636  Neeladri Nagar  10 BHK      4000.0  12.0  160.0   10     4000.000000
9935           others  13 BHK      5425.0  13.0  275.0   13     5069.124424
```

[60]:

```
plt.hist(df_copy3.bath,rwidth=0.9)
plt.xlabel("bath")
plt.ylabel("count")
```

[60]:

```
Text(0, 0.5, 'count')
```



[61]:
```
df_copy3=df_copy3[df_copy3.bath<df_copy3.BHK+2]
```

[62]:
```
df_copy3.shape
```

[62]:
```
(7672, 7)
```

[63]:
```
df_copy4=df_copy3.drop(["size","price_per_sqft"],axis=1)
```

[64]:
```
df_copy4.head()
```

[64]:
```
                 location  total_sqft  bath   price  BHK
0   Electronic City Phase II      1056.0   2.0   39.07    2
3         Lingadheeranahalli      1521.0   3.0   95.00    3
4                   Kothanur      1200.0   2.0   51.00    2
6             Old Airport Road      2732.0   4.0  204.00    4
11                Whitefield      2785.0   5.0  295.00    4
```

2 Encoding

[65]:
```
dummies=pd.get_dummies(df_copy4.location)
dummies.head(3)
```

[65]:
```
   1st Block Jayanagar  1st Phase JP Nagar  2nd Phase Judicial Layout  \
0                    0                   0                          0
3                    0                   0                          0
4                    0                   0                          0
```

```
   2nd Stage Nagarbhavi  5th Block Hbr Layout  5th Phase JP Nagar  \
0                     0                     0                   0   
3                     0                     0                   0   
4                     0                     0                   0   

   6th Phase JP Nagar  7th Phase JP Nagar  8th Phase JP Nagar  \
0                   0                   0                   0   
3                   0                   0                   0   
4                   0                   0                   0   

   9th Phase JP Nagar  …  Vishveshwarya Layout  Vishwapriya Layout  \
0                   0  …                     0                   0   
3                   0  …                     0                   0   
4                   0  …                     0                   0   

   Vittasandra  Whitefield  Yelachenahalli  Yelahanka  Yelahanka New Town  \
0            0           0               0          0                   0   
3            0           0               0          0                   0   
4            0           0               0          0                   0   

   Yelenahalli  Yeshwanthpur  others  
0            0             0       0  
3            0             0       0  
4            0             0       0  

[3 rows x 241 columns]
```

[66]:

```
df_copy5=pd.concat([df_copy4,dummies.drop("others",axis=1)],axis=1)
df_copy5.head(3)
```

[66]:

```
                   location  total_sqft  bath  price  BHK  \
0  Electronic City Phase II      1056.0   2.0  39.07    2   
3        Lingadheeranahalli      1521.0   3.0  95.00    3   
4                  Kothanur      1200.0   2.0  51.00    2   

   1st Block Jayanagar  1st Phase JP Nagar  2nd Phase Judicial Layout  \
0                    0                   0                          0   
3                    0                   0                          0   
4                    0                   0                          0   

   2nd Stage Nagarbhavi  5th Block Hbr Layout  …  Vijayanagar  \
0                     0                     0  …            0   
3                     0                     0  …            0   
4                     0                     0  …            0   

   Vishveshwarya Layout  Vishwapriya Layout  Vittasandra  Whitefield  \
0                     0                   0            0           0   
```

```
3                     0                   0            0           0
4                     0                   0            0           0

   Yelachenahalli  Yelahanka  Yelahanka New Town  Yelenahalli  Yeshwanthpur
0               0          0                   0            0             0
3               0          0                   0            0             0
4               0          0                   0            0             0

[3 rows x 245 columns]
```

[67]:

```
df_copy5=df_copy5.drop("location",axis=1)
df_copy5.shape
```

[67]:

```
(7672, 244)
```

3 Model bulding

[68]:

```
x=df_copy5.drop("price",axis=1)
x.head()
```

[68]:

```
    total_sqft  bath  BHK  1st Block Jayanagar  1st Phase JP Nagar   \
0       1056.0   2.0    2                    0                   0
3       1521.0   3.0    3                    0                   0
4       1200.0   2.0    2                    0                   0
6       2732.0   4.0    4                    0                   0
11      2785.0   5.0    4                    0                   0

    2nd Phase Judicial Layout  2nd Stage Nagarbhavi  5th Block Hbr Layout   \
0                           0                     0                      0
3                           0                     0                      0
4                           0                     0                      0
6                           0                     0                      0
11                          0                     0                      0

    5th Phase JP Nagar  6th Phase JP Nagar  …  Vijayanagar  \
0                    0                   0  …            0
3                    0                   0  …            0
4                    0                   0  …            0
6                    0                   0  …            0
11                   0                   0  …            0

    Vishveshwarya Layout  Vishwapriya Layout  Vittasandra  Whitefield   \
0                      0                   0            0           0
3                      0                   0            0           0
4                      0                   0            0           0
6                      0                   0            0           0
```

```
11                     0                    0           0           1

    Yelachenahalli  Yelahanka  Yelahanka New Town  Yelenahalli  Yeshwanthpur
0                0          0                   0            0             0
3                0          0                   0            0             0
4                0          0                   0            0             0
6                0          0                   0            0             0
11               0          0                   0            0             0

[5 rows x 243 columns]
```

[69]:
```
y=df_copy5.price
y.head()
```

[69]:
```
0      39.07
3      95.00
4      51.00
6     204.00
11    295.00
Name: price, dtype: float64
```

[70]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.
 ↪22,random_state=29)
```

[71]:
```
from sklearn.linear_model import LinearRegression
lr_clf=LinearRegression()
lr_clf.fit(x_train,y_train)
lr_clf.score(x_test,y_test)
```

[71]:
```
0.8430104651996677
```

[72]:
```
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score

cv=ShuffleSplit(n_splits=5,test_size=0.22,random_state=12)

cross_val_score(LinearRegression(),x,y,cv=cv)
```

[72]:
```
array([ 8.36901632e-01,  8.63577341e-01, -2.30433018e+14, -8.26731318e+14,
        8.36353524e-01])
```

[73]:
```
from sklearn.model_selection import GridSearchCV

from sklearn.linear_model import Lasso
from sklearn.tree import DecisionTreeRegressor

```

```
def find_best_model_using_gridsearchcv(x,y):
    algos={
        "liner_regression":{
            "model":LinearRegression(),
            "params":{
                "normalize":[True,False]
            }
        },
        "lasso":{
            "model":Lasso(),
            "params":{
                "alpha":[1,2],
                "selection":["random","cyclic"]
            }
        },
        "decision tree":{
            "model":DecisionTreeRegressor(),
            "params":{
                "criterion":["mse","friedman_mse"],
                "splitter":["best","random"]
            }
        }
    }

    scores=[]
    cv=ShuffleSplit(n_splits=5,test_size=0.22,random_state=12,)
    for algo_name, config in algos.items():
        ␣
 ↪gs=GridSearchCV(config['model'],config["params"],cv=cv,return_train_score=False)
        gs.fit(x,y)
        scores.append({
            "model":algo_name,
            "best_score":gs.best_score_,
            "best_params":gs.best_params_
        })
    return pd.DataFrame(scores,columns=["model","best_score","best_params"])
find_best_model_using_gridsearchcv(x,y)
```

```
[73]:              model  best_score  \
      0  liner_regression    0.851008
      1             lasso    0.830676
      2     decision tree    0.826229

                                             best_params
      0                              {'normalize': True}
      1                {'alpha': 2, 'selection': 'cyclic'}
      2  {'criterion': 'friedman_mse', 'splitter': 'best'}
```

[74]:
```
x.columns
```

[74]:
```
Index(['total_sqft', 'bath', 'BHK', '1st Block Jayanagar',
       '1st Phase JP Nagar', '2nd Phase Judicial Layout',
       '2nd Stage Nagarbhavi', '5th Block Hbr Layout', '5th Phase JP Nagar',
       '6th Phase JP Nagar',
       …
       'Vijayanagar', 'Vishveshwarya Layout', 'Vishwapriya Layout',
       'Vittasandra', 'Whitefield', 'Yelachenahalli', 'Yelahanka',
       'Yelahanka New Town', 'Yelenahalli', 'Yeshwanthpur'],
      dtype='object', length=243)
```

[75]:
```
np.where(x.columns=="2nd Stage Nagarbhavi")[0][0]
```

[75]:
```
6
```

[76]:
```
def predict_price(location,sqft,bath,BHK):
    loc_index=np.where(x.columns==location)[0][0]

    a=np.zeros(len(x.columns))
    a[0]=sqft
    a[1]=bath
    a[2]=BHK

    if loc_index>=0:
        a[loc_index]=1
    return lr_clf.predict([a])[0]
```

[77]:
```
predict_price("1st Block Jayanagar",1500,3,3)
```

[77]:
```
115.96596278763438
```

[78]:
```
predict_price("1st Block Jayanagar",1500,4,4)
```

[78]:
```
127.78170187851941
```

[79]:
```
predict_price("Indira Nagar",1500,4,4)
```

[79]:
```
153.1193750858707
```

[80]:
```
predict_price("Indira Nagar",1500,3,3)
```

[80]:
```
141.30363599498568
```

[81]:
```
predict_price("Vijayanagar",2000,4,4)
```

[81]:
```
146.25340547707984
```

[82]:

```
predict_price("Vijayanagar",2000,3,4)
```

[82]:

```
141.02842337329878
```

[83]:

```
import pickle
with open("banglore_home_prices_model.pickle","wb") as f:
    pickle.dump(lr_clf,f)
```

[84]:

```
import json
columns={
    "data_columns":[col.lower() for col in x.columns]
}
with open("columns.json","w") as f:
    f.write(json.dumps(columns))
```

[ ]:

```
```