House Price Predictor

ABHISHEK YADAV
19b090001
Dept. of Mathematics, IIT BOMBAY

INTRODUCTION

In this project, we will build a real estate price prediction model. We will build a model using sklearn and linear regression using banglore home prices dataset from kaggle.com.

During model building, we will use data science concepts such as data load and cleaning, outlier detection and removal, feature engineering, dimensionality reduction, gridsearchcv for hyperparameter tunning, k fold cross validation etc. Technology and tools wise this project includes:

1. Python
2. Numpy and Pandas for data cleaning
3. Matplotlib for data visualization
4. Sklearn for model building

DATASET

Dataset is downloaded from here:
https://www.kaggle.com/amitabhajoy/bengaluru-house-price-data

Features:

1. area_type
2. availability
3. location
4. size
5. society
6. total_sqft
7. bath(no. of bathrooms)
8. balcony

Target Variable: price

The dataset has 13,320 rows.

LIBRARIES USED

1) Pandas: This library is used for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables. This library is also used to loading the datasets from different formats

2) Numpy: NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

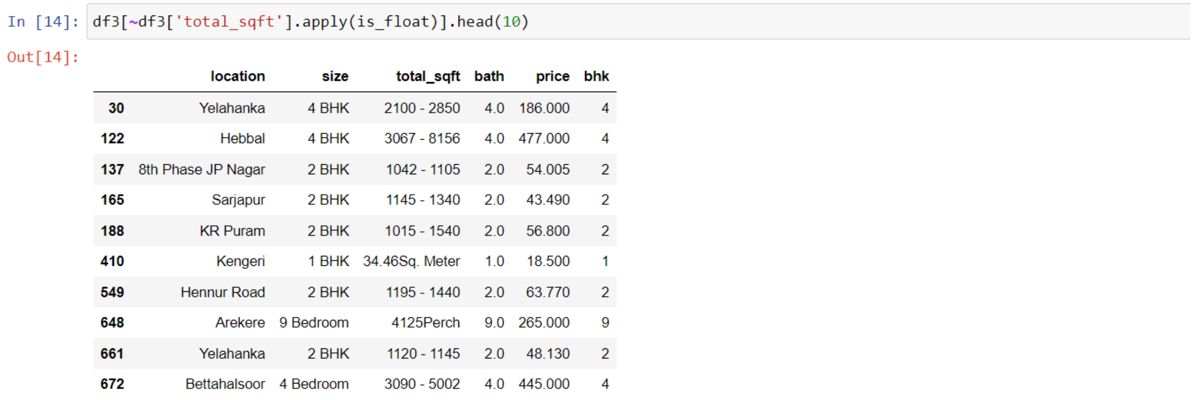
3) Matplotlib: Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

4) Scikit-learn (Sklearn): Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistent interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

DATA CLEANING AND DATA ENGINEERING

- Drop 'area_type','society','balcony' and 'availability' features as they won’t be useful in our model.
- Explore ‘’total_sqft’ feature:

Image below shows that total_sqft can be a range (e.g. 2100-2850). For such case we can just take average of min and max value in the range. There are other cases such as 34.46Sq, we are going to just drop such corner cases.

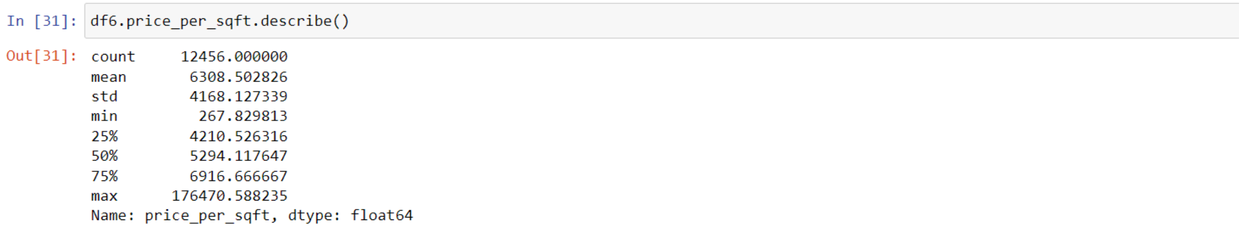


```
In [14]: df3[~df3['total_sqft'].apply(is_float)].head(10)
```

Out[14]:

	location	size	total_sqft	bath	price	bhk
30	Yelahanka	4 BHK	2100 - 2850	4.0	186.000	4
122	Hebbal	4 BHK	3067 - 8156	4.0	477.000	4
137	8th Phase JP Nagar	2 BHK	1042 - 1105	2.0	54.005	2
165	Sarjapur	2 BHK	1145 - 1340	2.0	43.490	2
188	KR Puram	2 BHK	1015 - 1540	2.0	56.800	2
410	Kengeri	1 BHK	34.46Sq. Meter	1.0	18.500	1
549	Hennur Road	2 BHK	1195 - 1440	2.0	63.770	2
648	Arekere	9 Bedroom	4125Perch	9.0	265.000	9
661	Yelahanka	2 BHK	1120 - 1145	2.0	48.130	2
672	Bettahalsoor	4 Bedroom	3090 - 5002	4.0	445.000	4

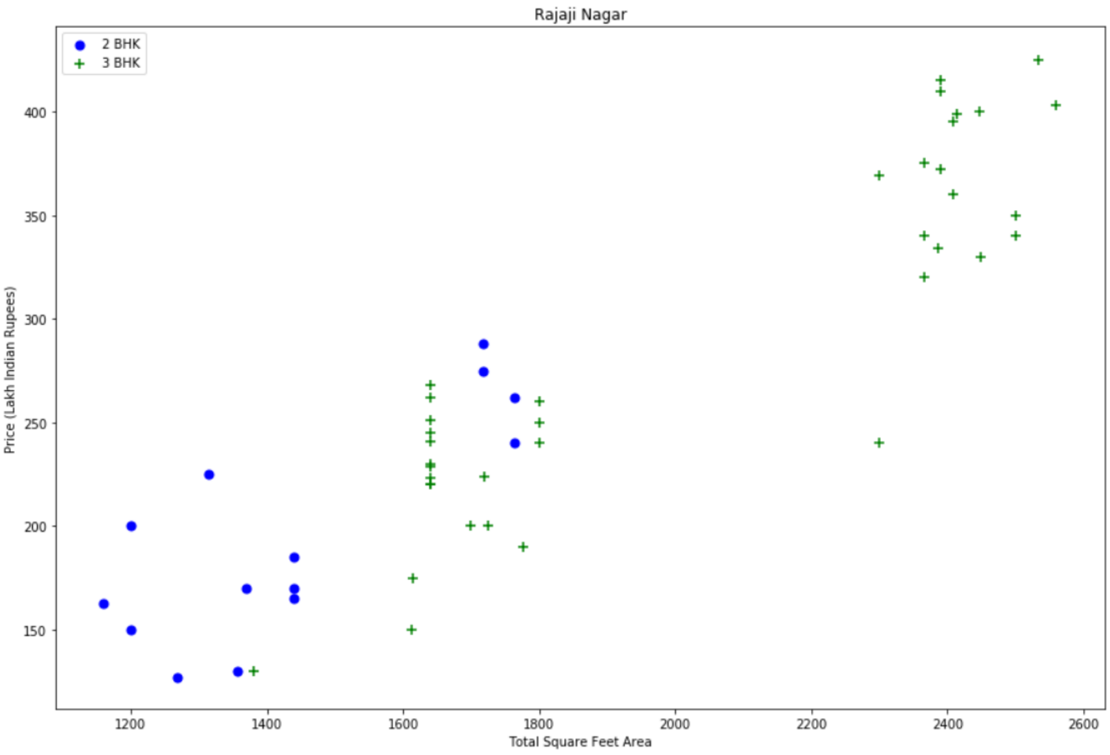
- Added new feature called price per square feet.
- Examine locations which is a categorical variable. We need to apply dimensionality reduction technique here to reduce number of locations.
- Any location having less than 10 data points should be tagged as "other" location. This way number of categories can be reduced by huge amount. Later on when we do one hot encoding, it will help us with having fewer dummy columns
- Outlier Removal Using Business Logic : Normally square ft per bedroom is 300 (i.e. 2 bhk apartment is minimum 600 sqft). If you have for example 400 sqft apartment with 2 bhk than that seems suspicious and can be removed as an outlier. We will remove such outliers by keeping our minimum thresold per bhk to be 300 sqft
- Outlier Removal Using Standard Deviation and Mean:

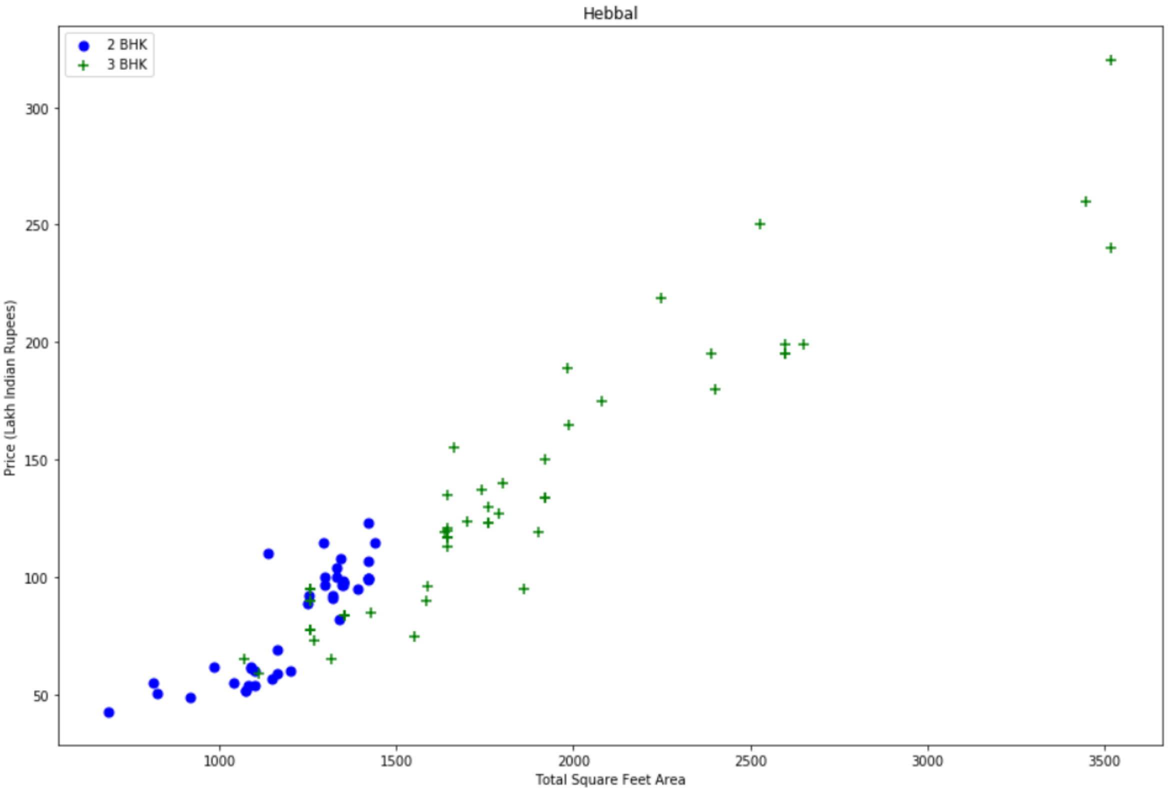


```
In [31]: df6.price_per_sqft.describe()

Out[31]: count     12456.000000
         mean       6308.502826
         std        4168.127339
         min         267.829813
         25%        4210.526316
         50%        5294.117647
         75%        6916.666667
         max      176470.588235
         Name: price_per_sqft, dtype: float64
```

Here we find that min price per sqft is 267 rs/sqft whereas max is 12000000, this shows a wide variation in property prices. We should remove outliers per location using mean and standard deviation

- Let's check if for a given location how does the 2 BHK and 3 BHK property prices look like.


Rajaji Nagar
● 2 BHK
+ 3 BHK
Price (Lakh Indian Rupees)
400
350
300
250
200
150
1200
1400
1600
1800
2000
2200
2400
2600
Total Square Feet Area


Hebbal
● 2 BHK
+ 3 BHK
Price (Lakh Indian Rupees)
300
250
200
150
100
50
1000
1500
2000
2500
3000
3500
Total Square Feet Area

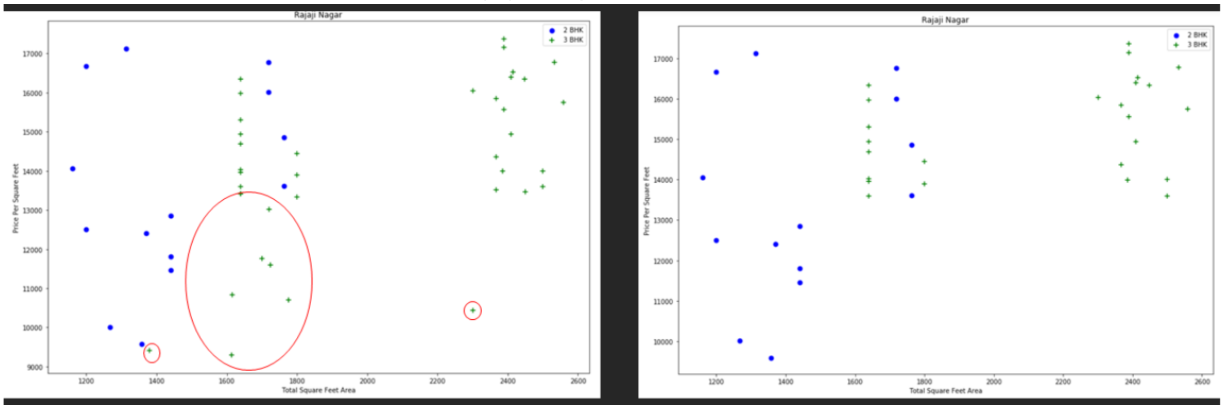
We observe that for same location price of some 3BHK is less than that of 2BHK(with same area).

We should remove properties where for same location, the price of (for example) 3 bedroom apartment is less than 2 bedroom apartment (with same square ft area). What we will do is for a given location, we will build a dictionary of stats per bhk, i.e.

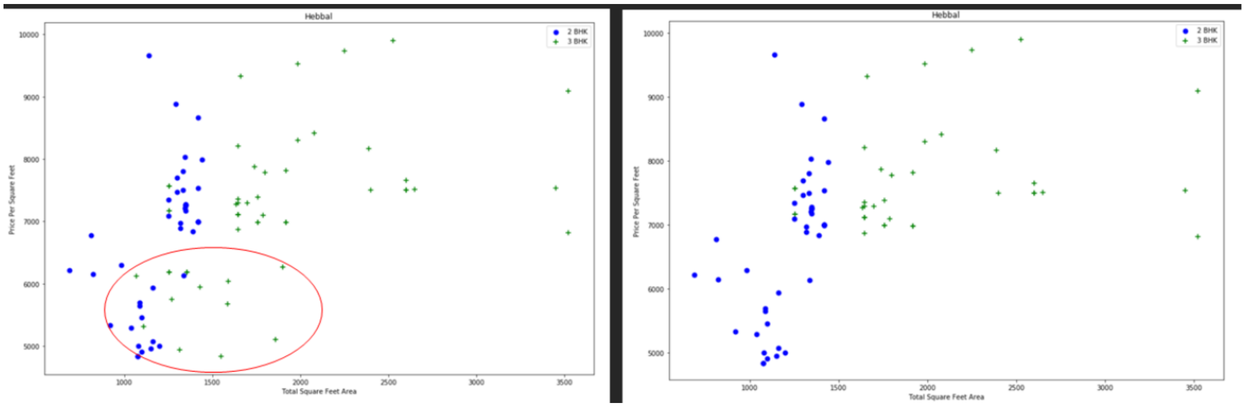
```
    {

        '1' : {

            'mean': 4000,

            'std: 2000,

            'count': 34

        },

        '2' : {

            'mean': 4300,

            'std: 2300,

            'count': 22

        },

  }
```

Now we can remove those 2 BHK apartments whose price_per_sqft is less than mean price_per_sqft of 1 BHK apartment

Before and after outlier removal: Rajaji Nagar



Before and after outlier removal: Hebbal

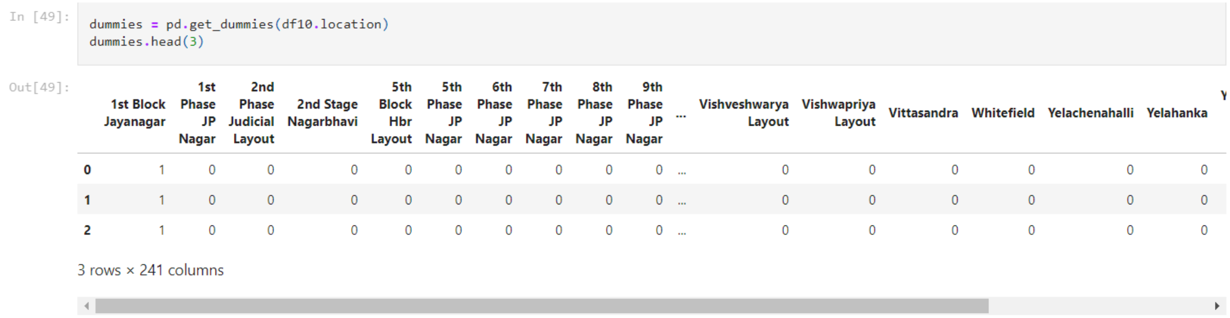


Based on above charts we can see that data points highlighted in red below are outliers and they are being removed.

- Outlier Removal Using Bathrooms Feature : It is unusual to have 2 more bathrooms than number of bedrooms in a home(if you have 4 bedroom home and even if you have bathroom in all 4 rooms plus one guest bathroom, you will have total bath = total bed + 1 max.). Anything above that is an outlier or a data error and will be removed.

ONE HOT ENCODING

Use One Hot Encoding For Location:

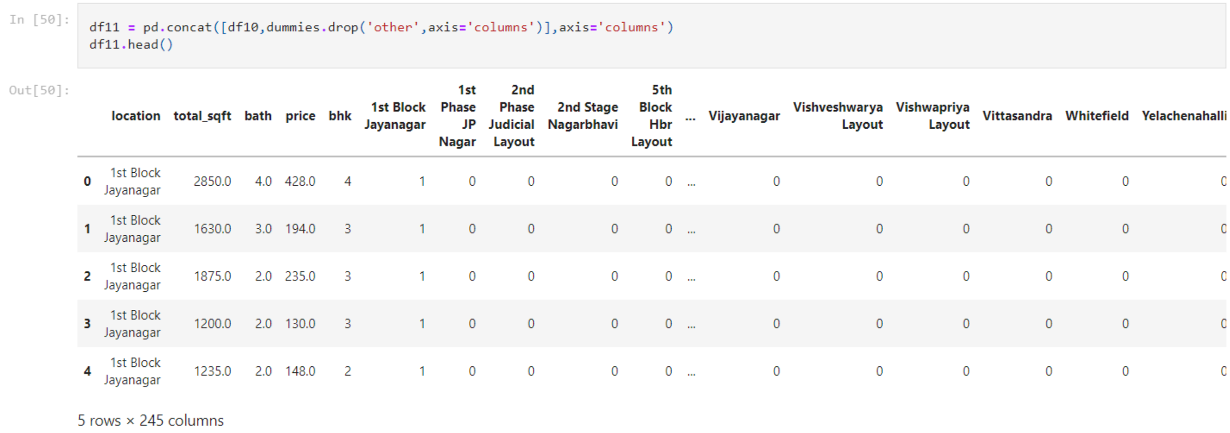


```
In [49]: dummies = pd.get_dummies(df10.location)
         dummies.head(3)
```

Out[49]:

	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	7th Phase JP Nagar	8th Phase JP Nagar	9th Phase JP Nagar	...	Vishveshwarya Layout	Vishwapriya Layout	Vittasandra	Whitefield	Yelachenahalli	Yelahanka	Y
0	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
1	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
2	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	

3 rows × 241 columns



```
In [50]: df11 = pd.concat([df10,dummies.drop('other',axis='columns')],axis='columns')
         df11.head()
```

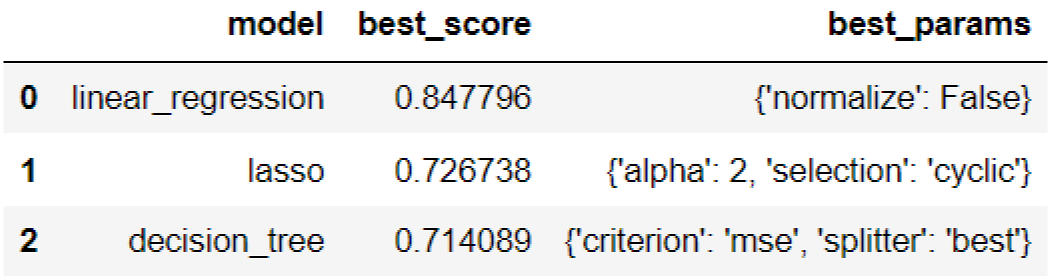
Out[50]:

	location	total_sqft	bath	price	bhk	1st Block Jayanagar	1st Phase JP Nagar	2nd Phase Judicial Layout	2nd Stage Nagarbhavi	5th Block Hbr Layout	...	Vijayanagar	Vishveshwarya Layout	Vishwapriya Layout	Vittasandra	Whitefield	Yelachenahalli	
0	1st Block Jayanagar	2850.0	4.0	428.0	4	1	0	0	0	0	...	0	0	0	0	0	0	
1	1st Block Jayanagar	1630.0	3.0	194.0	3	1	0	0	0	0	...	0	0	0	0	0	0	
2	1st Block Jayanagar	1875.0	2.0	235.0	3	1	0	0	0	0	...	0	0	0	0	0	0	
3	1st Block Jayanagar	1200.0	2.0	130.0	3	1	0	0	0	0	...	0	0	0	0	0	0	
4	1st Block Jayanagar	1235.0	2.0	148.0	2	1	0	0	0	0	...	0	0	0	0	0	0	

5 rows × 245 columns

MODEL

- ·Performed hyperparameter tuning for 3 algorithms(Linear Regression, Lasso and Decision Tree) using GridSearchCV.
- Linear Regression performed best with 0.84 score.



	model	best_score	best_params
0	linear_regression	0.847796	{'normalize': False}
1	lasso	0.726738	{'alpha': 2, 'selection': 'cyclic'}
2	decision_tree	0.714089	{'criterion': 'mse', 'splitter': 'best'}

Based on the above results we can say that LinearRegression gives the best score. Hence we will use that to build the final model.