DATA SCIENCE PROJECT

Hoang Bui 02-12-2019

1. Introduction

Airbnb, Inc. is an online marketplace for arranging or offering lodging, primarily homestays, or tourism experiences wiki (https://en.wikipedia.org/wiki/Airbnb). The name comes from ***"air mattress B&B"*** It currently covers more than 81,000 cities and 191 countries worldwide. **Airbnb** haved provided a new personalize way to tourims experience in staying, especially homestays and today, **Airbnb** became one of a kind service that is used and recognized by the whole world.

These millions of listings generate a lot of data - data that can be analyzed and used for security, business decisions, understanding of customers' and providers' (hosts) behavior and performance on the platform, guiding marketing initiatives, implementation of innovative additional services and much more. In this kernel, we will observe Amstedam 2019 dataset, provided by Airbnb, the original source can be found on website (http://insideairbnb.com/get-the-data.html)

There are 3 questions we need to answer through this dataset:

- Describe some insides about the listing in Amsterdam?
- How does current infomation affect to the price of the listing?
- Can we use listing features to predict housing price of the stays?

2. Geting started

2.1 Prerequired

- scikit-learn
- pandas
- seaborn
- scipy

Out[116]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_
0	2818	Quiet Garden View Room & Super Fast WiFi	3159	Daniel	NaN	Oostelijk Havengebied - Indische Buurt	52.36575	4.94142	Private room	59	3
1	20168	Studio with private bathroom in the centre 1	59484	Alexander	NaN	Centrum-Oost	52.36509	4.89354	Private room	80	1
2	25428	Lovely apt in City Centre (w.lift) near Jordaan	56142	Joan	NaN	Centrum-West	52.37297	4.88339	Entire home/apt	125	14
3	27886	Romantic, stylish B&B houseboat in canal district	97647	Flip	NaN	Centrum-West	52.38673	4.89208	Private room	155	2
4	28871	Comfortable double room	124245	Edwin	NaN	Centrum-West	52.36719	4.89092	Private room	75	2

Data Explaination

- **id**: listing ID
- **name**: name of the listing
- **host_id**: host ID
- **host_name**: name of the host
- **neighbourhood_group**: location
- **neighbourhood**: area
- **latitude**: latitude coordinates
- **longitude**: longitude coordinates
- **room_type**: listing space type
- **price**: price in dollars
- **minimum_nights**: amount of nights minimum
- **number_of_reviews**: number of reviews
- **last_review**: latest review
- **reviews_per_month**: number of reviews per month
- **calculated_host_listings_count**: amount of listing per host
- **availability_365**: number of days when listing is available for booking

3. Understand and Analyze data

3.1 Understanding our dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20244 entries, 0 to 20243
Data columns (total 16 columns):
id                                20244 non-null int64
name                              20209 non-null object
host_id                           20244 non-null int64
host_name                         20239 non-null object
neighbourhood_group               0 non-null float64
neighbourhood                     20244 non-null object
latitude                          20244 non-null float64
longitude                         20244 non-null float64
room_type                         20244 non-null object
price                             20244 non-null int64
minimum_nights                    20244 non-null int64
number_of_reviews                 20244 non-null int64
last_review                       17902 non-null object
reviews_per_month                 17902 non-null float64
calculated_host_listings_count    20244 non-null int64
availability_365                  20244 non-null int64
dtypes: float64(4), int64(7), object(5)
memory usage: 2.5+ MB
```

Overviews:

- Dataset contains 20244 rows
- There are some nullable fields: name, host_name, neighbourhood_group, last_review, reviews_per_month
- **neighbourhood** doesn't contain any data

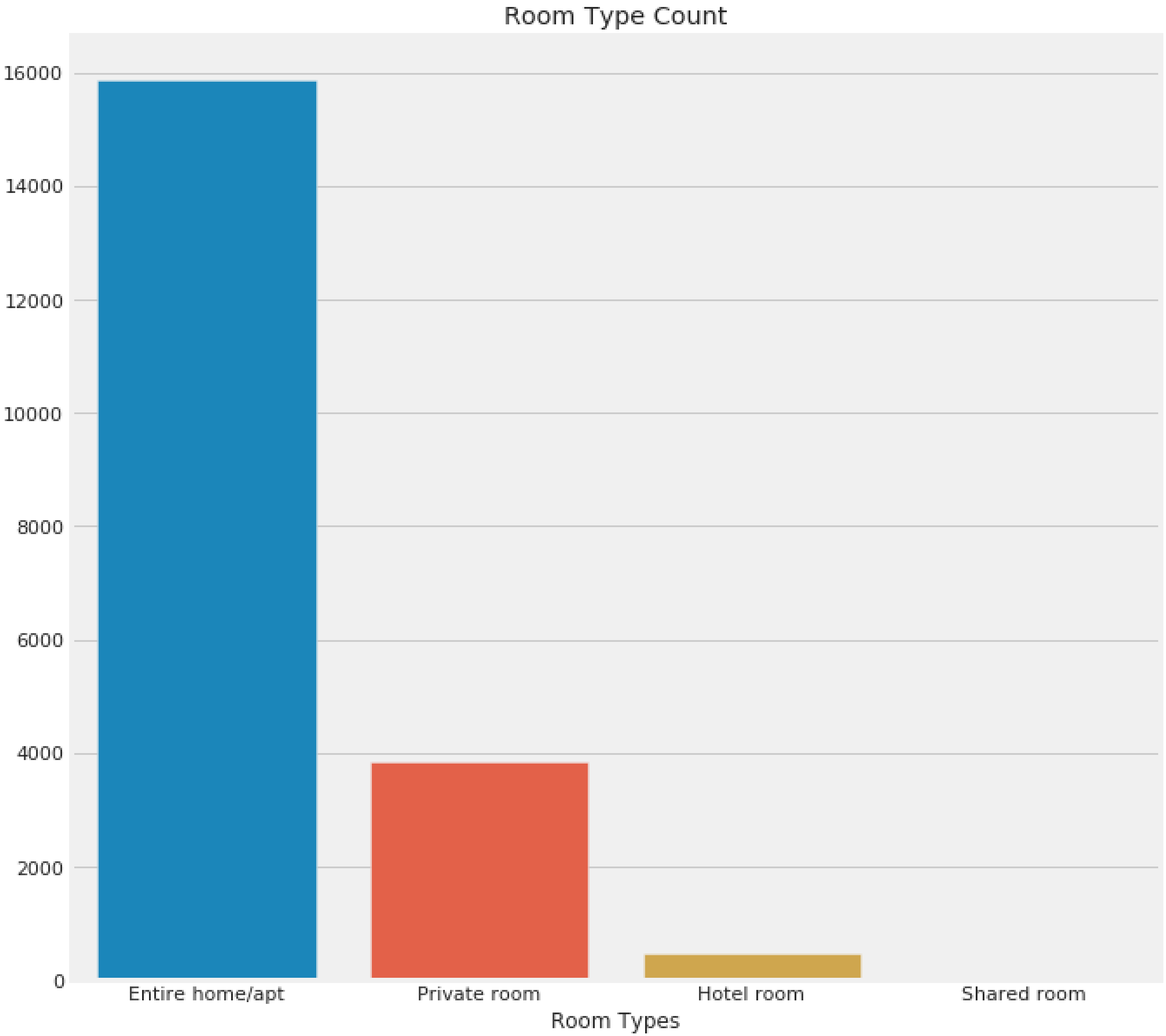
Then we need to do some pre-process steps:

- remove neighbourhood_group column
- fill NAN values of **name**, **host_name**, and zero values of **last_review**, **reviews_per_month**

3.2 Analysis

1. Distribution of room_type

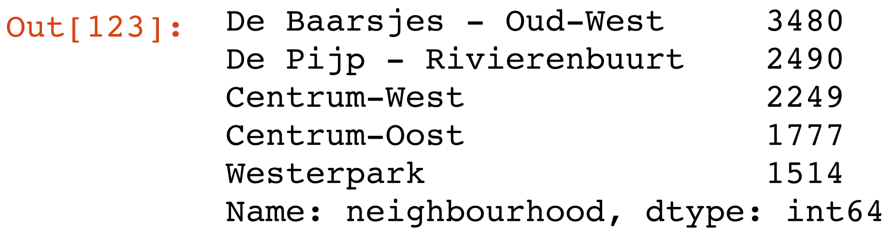
```
Out[121]:  Entire home/apt    15880
           Private room        3850
           Hotel room           461
           Shared room           53
           Name: room_type, dtype: int64
```


Room Type Count
16000
14000
12000
10000
8000
6000
4000
2000
0
Entire home/apt
Private room
Hotel room
Shared room
Room Types

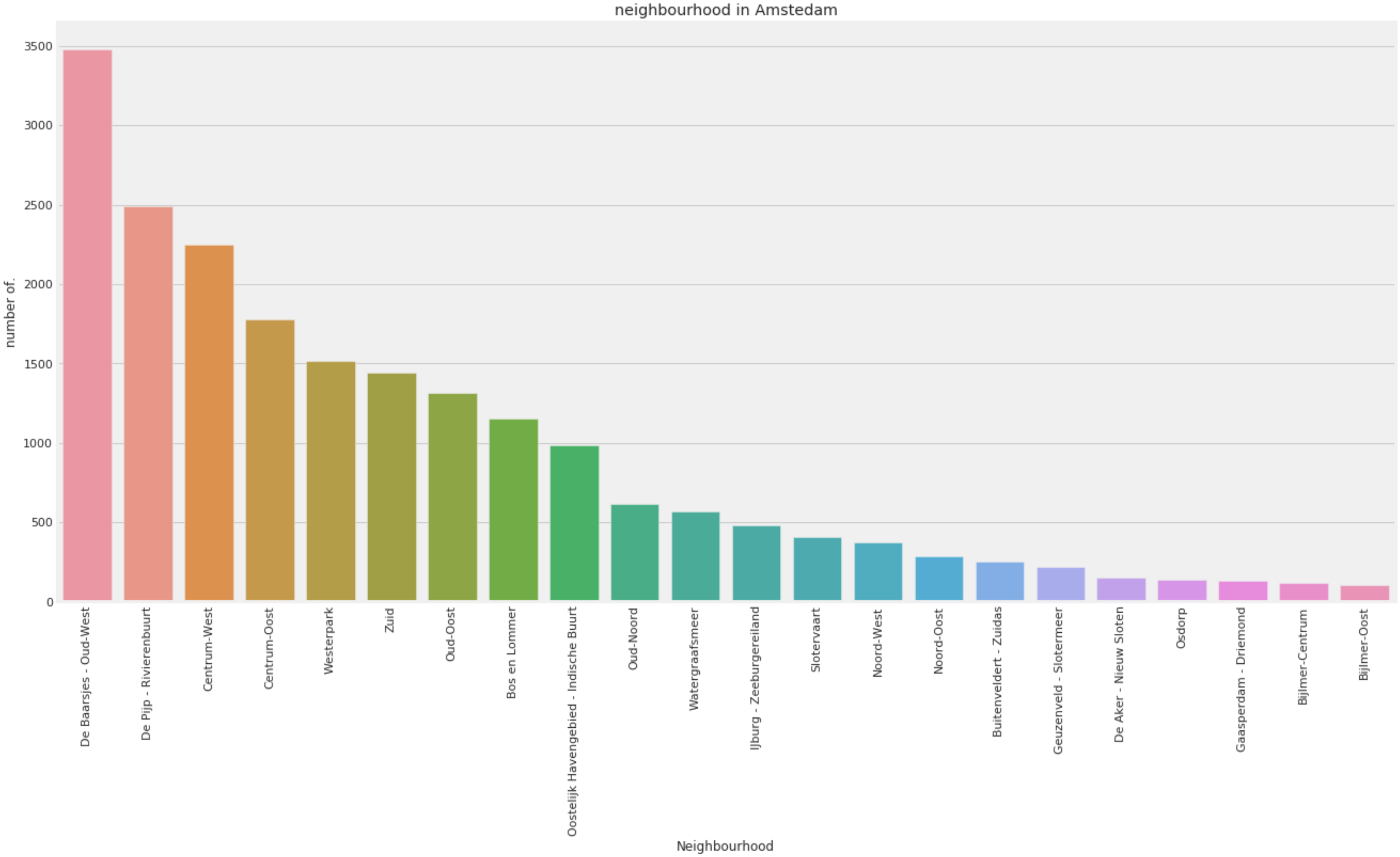
Discussion:

- Almost rooms in Airbnb are home/aptment or private room whilst shared room and hotel room take a minor amount
- It's obviously that all the people who rent an Airbnb prefer an entire home !

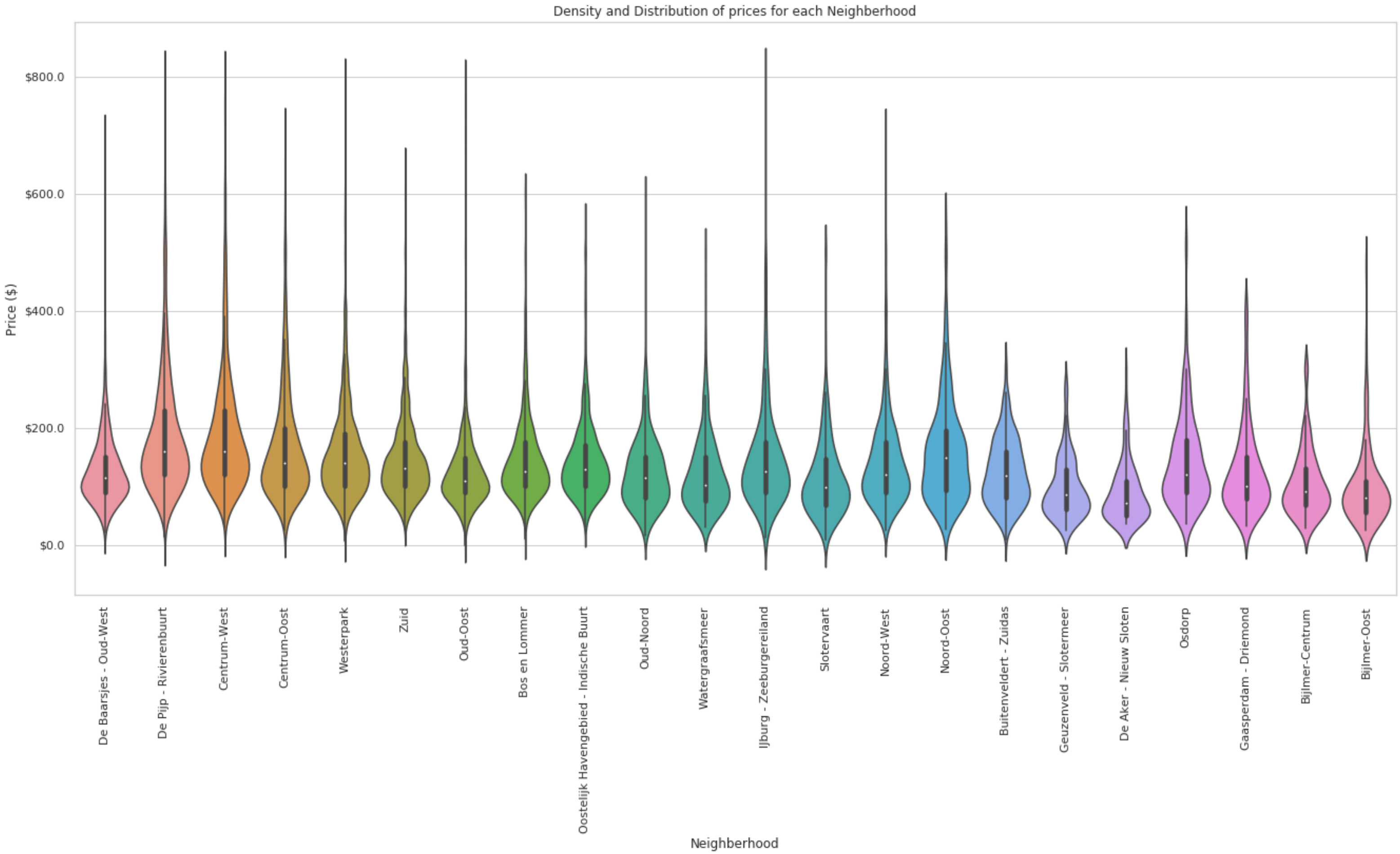
2. Neighborhood in Amstedam



```
Out[123]:  De Baarsjes - Oud-West     3480
           De Pijp - Rivierenbuurt    2490
           Centrum-West               2249
           Centrum-Oost               1777
           Westerpark                 1514
           Name: neighbourhood, dtype: int64
```


neighbourhood in Amstedam
number of.
3500
3000
2500
2000
1500
1000
500
0
De Baarsjes - Oud-West
De Pijp - Rivierenbuurt
Centrum-West
Centrum-Oost
Westerpark
Zuid
Oud-Oost
Bos en Lommer
Oostelijk Havengebied - Indische Buurt
Oud-Noord
Watergraafsmeer
IJburg - Zeeburgereiland
Slotervaart
Noord-West
Noord-Oost
Buitenveldert - Zuidas
Geuzenveld - Slotermeer
De Aker - Nieuw Sloten
Osdorp
Gaasperdam - Driemond
Bijlmer-Centrum
Bijlmer-Oost
Neighbourhood

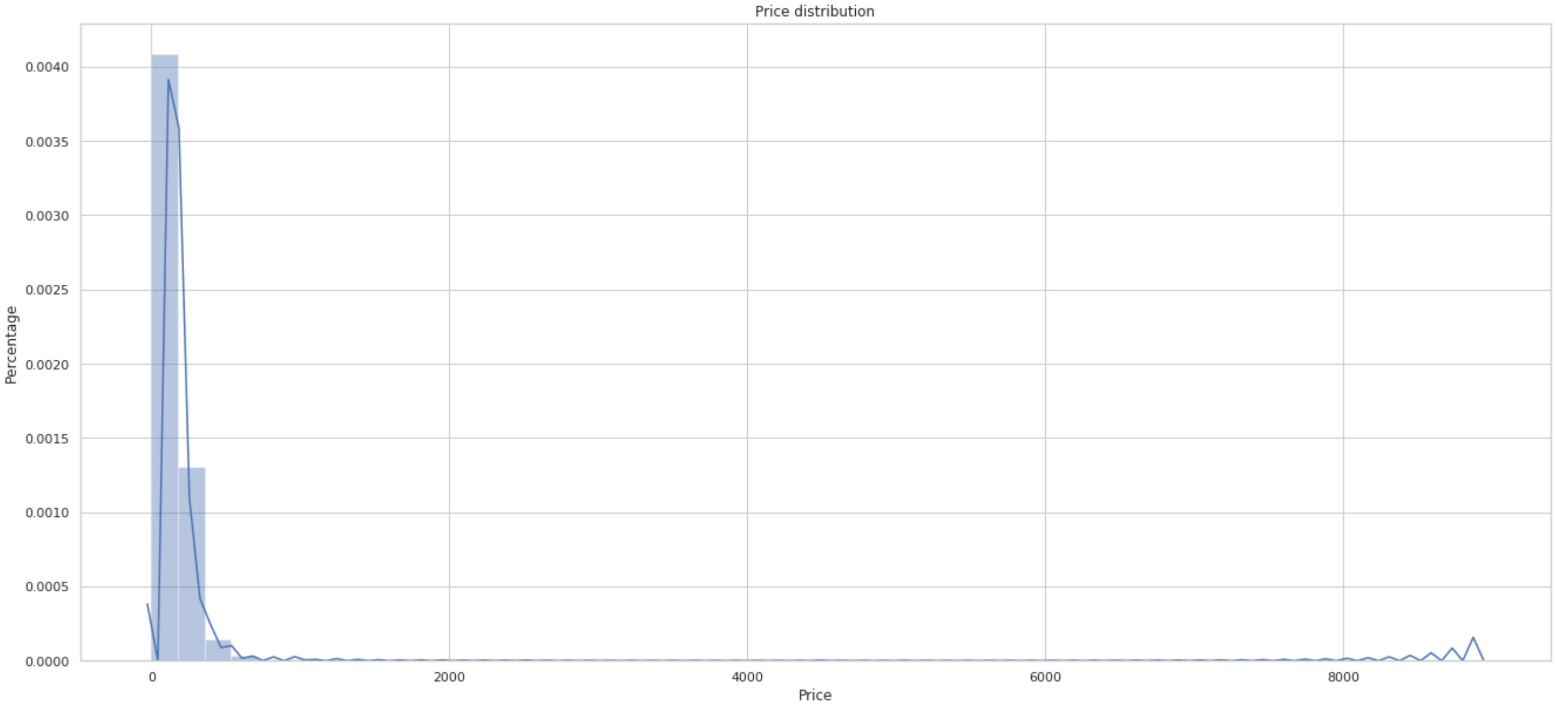
Neighbourhood with price in Amstedam



Discussion:

- Arcording to the **"Neighborhood in Amstedam"** bar chart and **"Density and Distribution of prices for each Neighberhood"** violin chart, we can set up that there are some neighbourhood have higher average listing price such as **Noord-Oost** or **De Pijp - Rivierenbuurt**, whilst some area with fewer number of listing have lower price/night also.
- **30%** of areas take more than **70%** of listing
- If you'd like to pick a crowdy area with most airbnb listing or prefer the lower budget with the lower price/night, let consider **De Baarsjes - Oud-West**

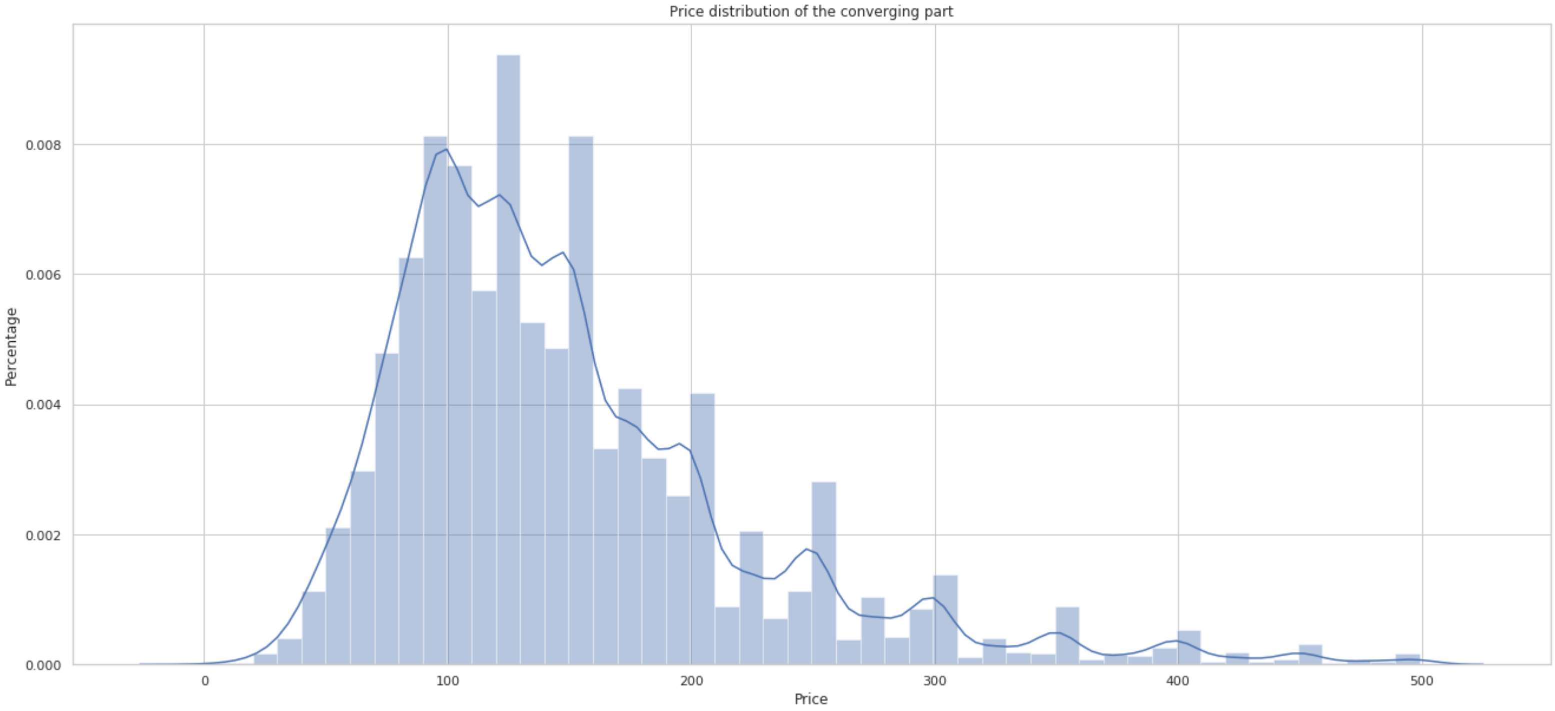
3. Price



Price distribution
Percentage
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
0
2000
4000
6000
8000
Price

Airbnb renting price in Amsterdam is quite converging, but there are some outliers listing have quite expensive than others. We are going to observe the converging and the outlier part

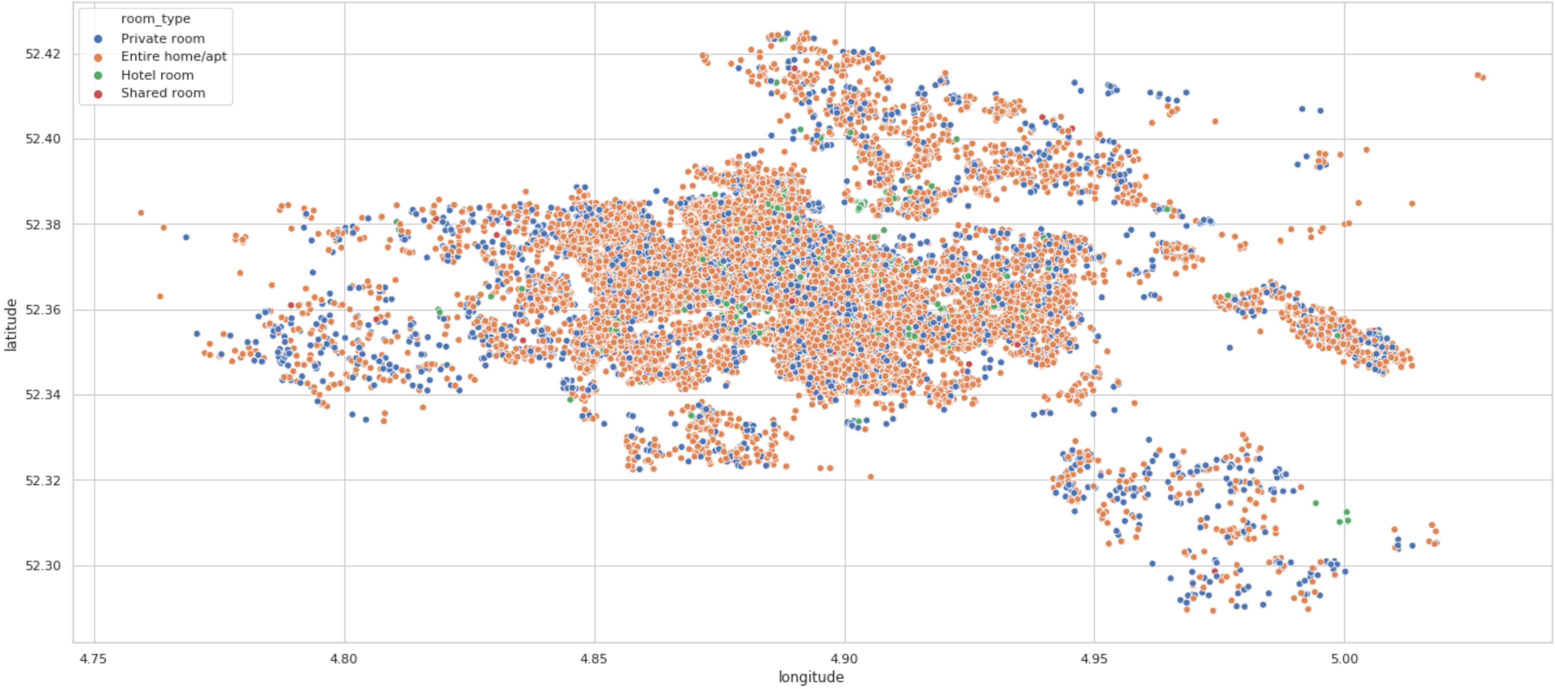
Price distribution of the convering part



Discussion:

- With the budget from 50 to 300 USD per night, you are able to rent almost 95% of house in Airbnb

4. Geomatric



From this **"map"**, we can observe the crowdy in the middle of Amstedam. There are alot of entire home and apartment for you to rent, where can easy to go around the city

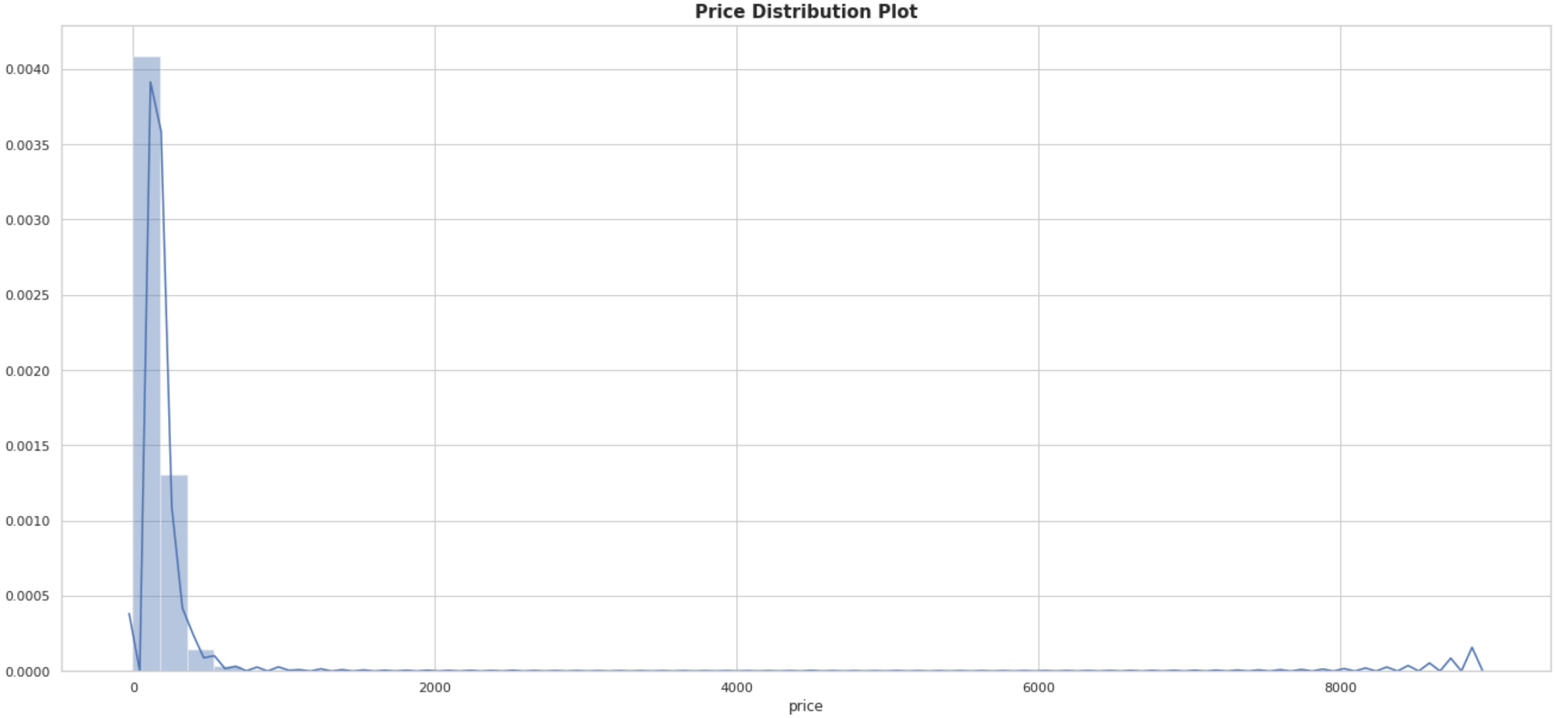
4. Price prediction

Preparation

There are some features we need to normalized before starting training. Let's take a closer look at:

a. price

Without normalize

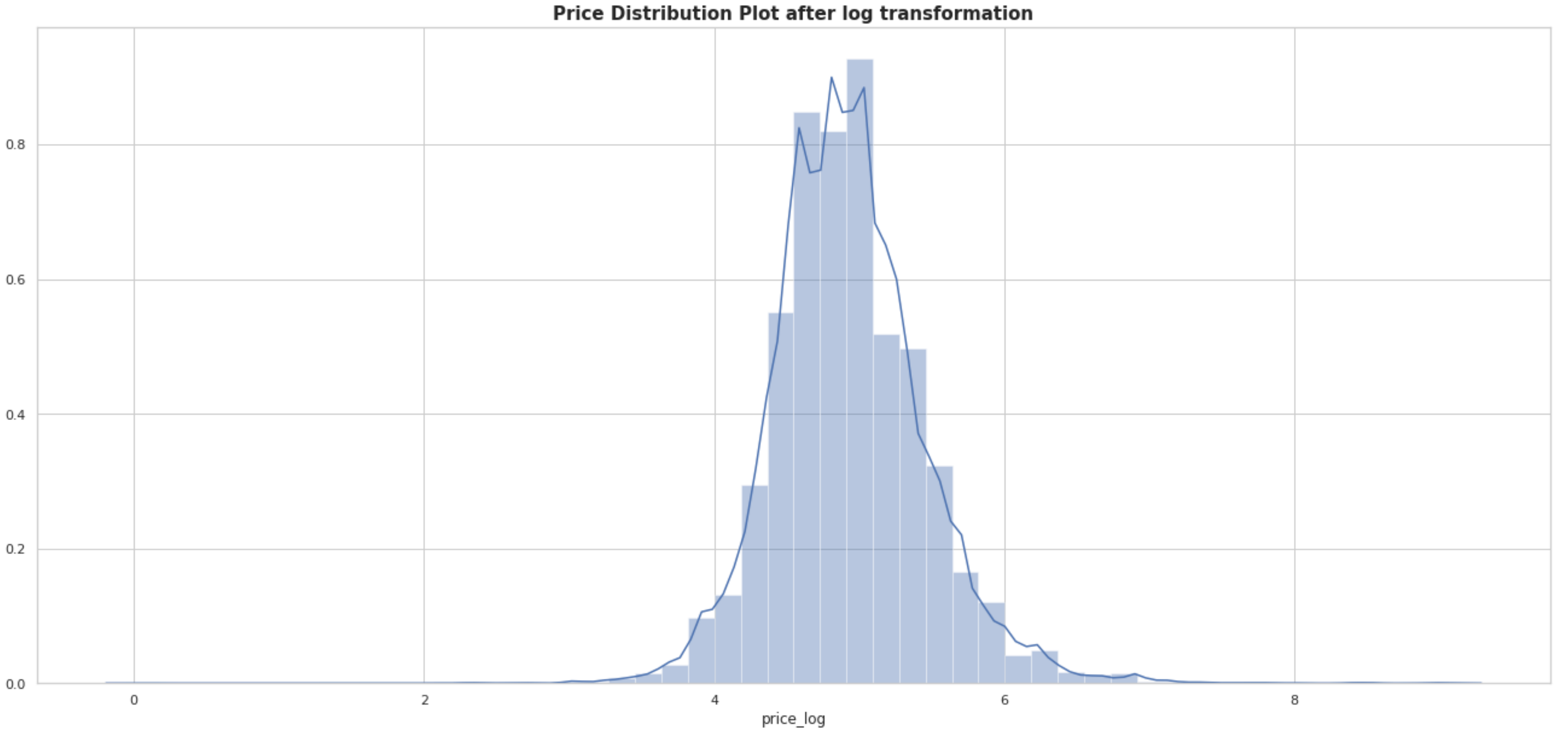


The graph shows that there is a right-skewed distribution on **price**, log transformation will be used to make less skewed. There are some benefits of **log** transformation:

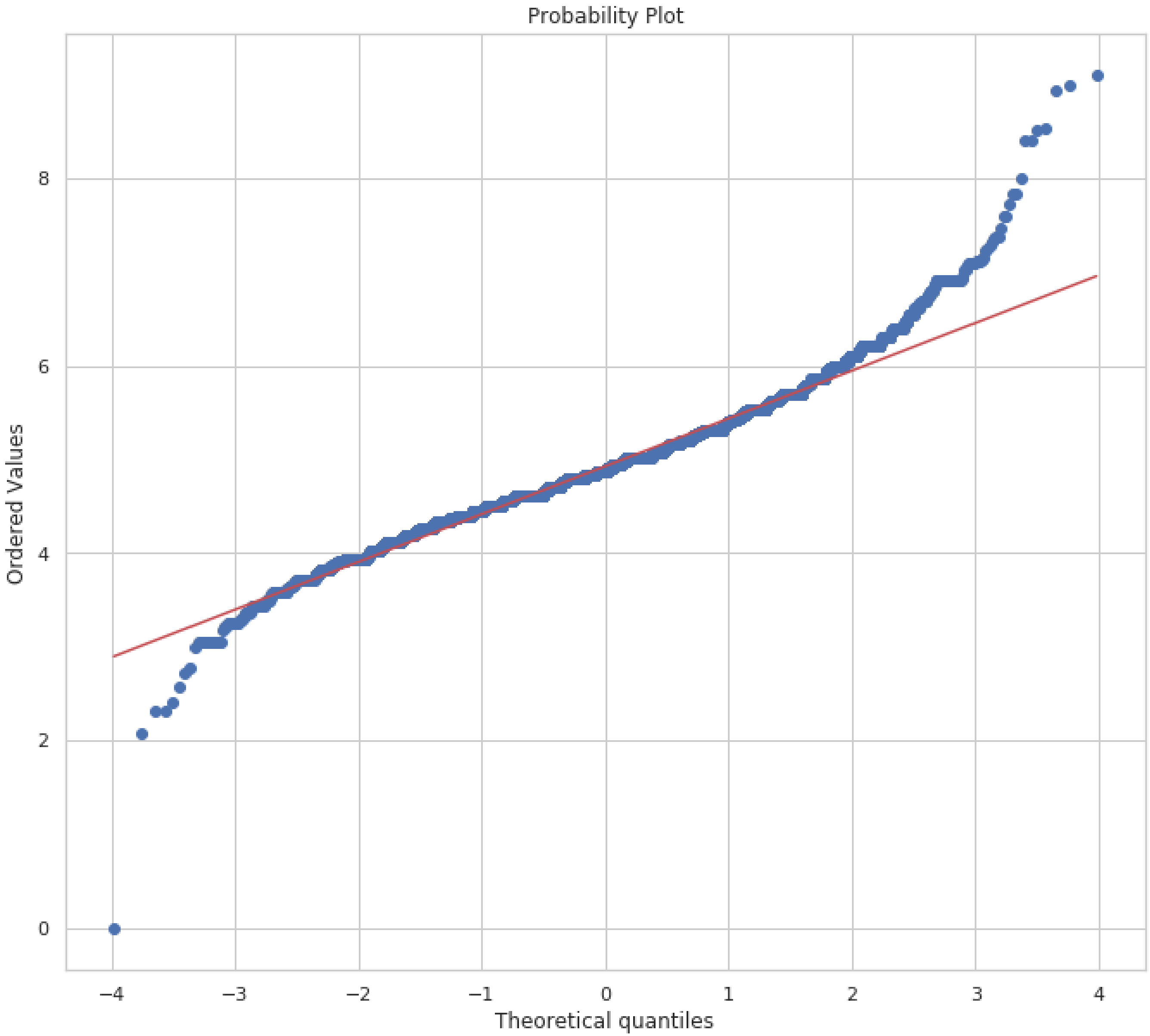
- avoid overfitting
- to normal distribution

Since division by zero is a problem, log+1 transformation would be better.

And plot again:



In below graph shows normality is a reasonable approximation.


Probability Plot
Ordered Values
Theoretical quantiles
0
2
4
6
8
−4
−3
−2
−1
0
1
2
3
4

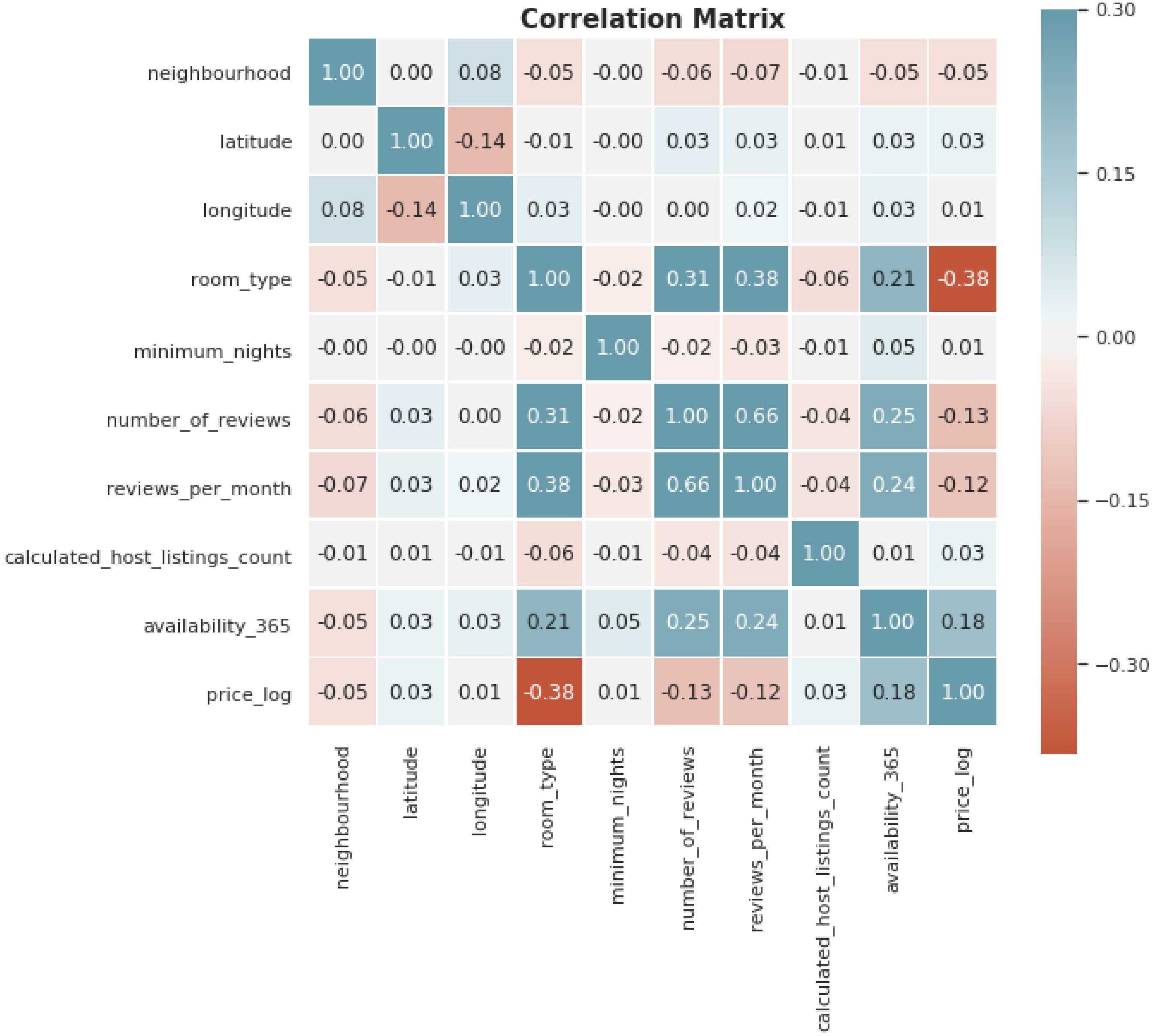
b. Drop non-nomial features

Some non-nomial and origin price won't not be used then will eliminated because it doesn't contain prediction information

4.1 What do features take effects to AirBnb housing prices?

Correlation metrix

Now is the time to look detail on the feature correlations. Correlation matrix will be created:



	neighbourhood	latitude	longitude	room_type	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365	price_log
neighbourhood	1.00	0.00	0.08	-0.05	-0.00	-0.06	-0.07	-0.01	-0.05	-0.05
latitude	0.00	1.00	-0.14	-0.01	-0.00	0.03	0.03	0.01	0.03	0.03
longitude	0.08	-0.14	1.00	0.03	-0.00	0.00	0.02	-0.01	0.03	0.01
room_type	-0.05	-0.01	0.03	1.00	-0.02	0.31	0.38	-0.06	0.21	-0.38
minimum_nights	-0.00	-0.00	-0.00	-0.02	1.00	-0.02	-0.03	-0.01	0.05	0.01
number_of_reviews	-0.06	0.03	0.00	0.31	-0.02	1.00	0.66	-0.04	0.25	-0.13
reviews_per_month	-0.07	0.03	0.02	0.38	-0.03	0.66	1.00	-0.04	0.24	-0.12
calculated_host_listings_count	-0.01	0.01	-0.01	-0.06	-0.01	-0.04	-0.04	1.00	0.01	0.03
availability_365	-0.05	0.03	0.03	0.21	0.05	0.25	0.24	0.01	1.00	0.18
price_log	-0.05	0.03	0.01	-0.38	0.01	-0.13	-0.12	0.03	0.18	1.00

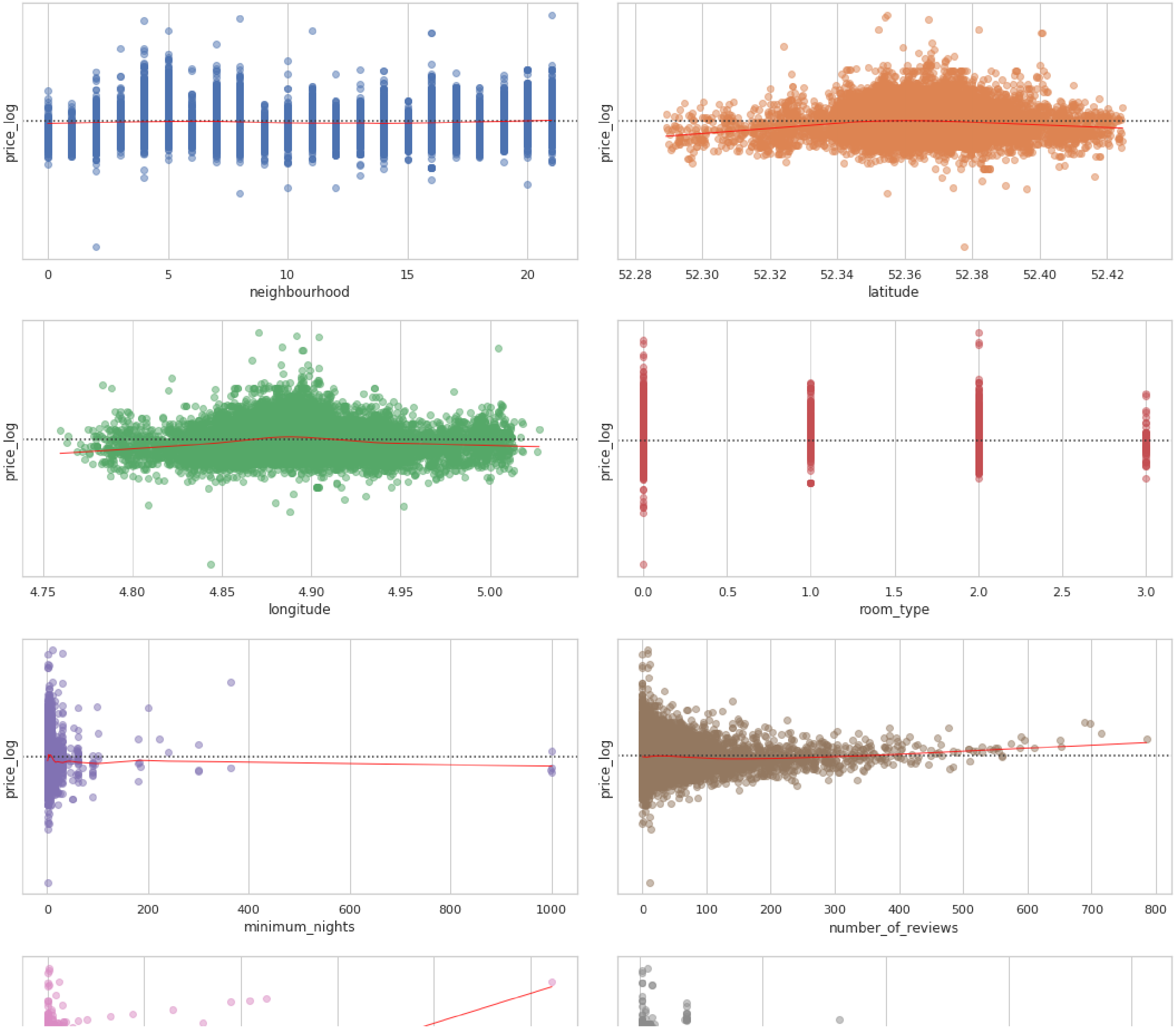
Correlation table shows there is no strong enough relationship between features and price_log. This indicates no feature needed to be taken out of data. This relationship will be found with feature_importances method

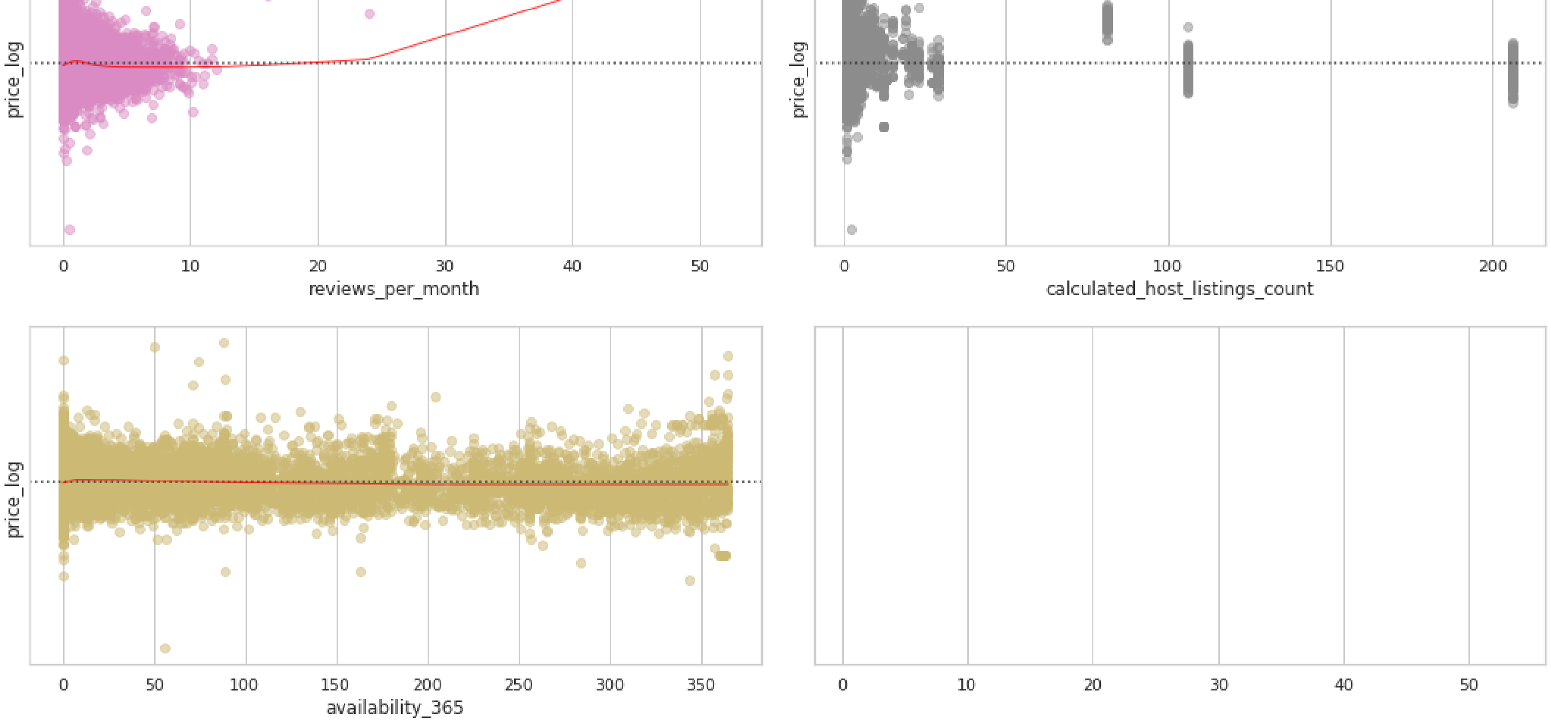
Residual Plots

Residual Plot is strong method to detect outliers, non-linear data and detecting data for regression models. The below charts show the residual plots for each feature with the price.

An ideal Residual Plot, the red line would be horizontal. Based on the below charts, most features are non-linear. On the other hand, there are not many outliers in each feature. This result led to underfitting. Underfitting can occur when input features do not have a strong relationship to target variables or over-regularized. For avoiding underfitting new data features can be added or regularization weight could be reduced.

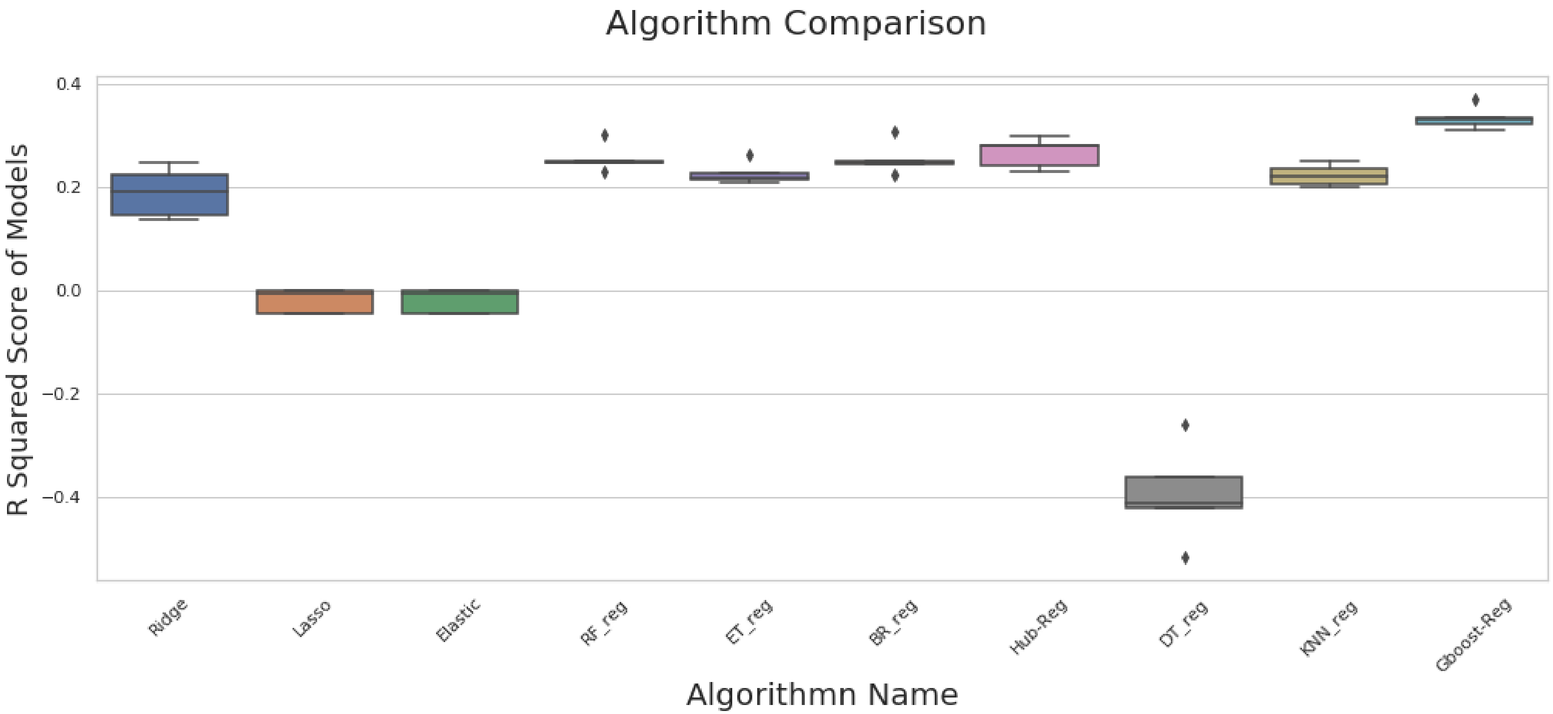
In this kernel, since the input feature data could not be increased, Regularized Linear Models will be used for regularization and polynomial transformation will be made to avoid underfitting.


price_log
0
5
10
15
20
neighbourhood
price_log
52.28
52.30
52.32
52.34
52.36
52.38
52.40
52.42
latitude
price_log
4.75
4.80
4.85
4.90
4.95
5.00
longitude
price_log
0.0
0.5
1.0
1.5
2.0
2.5
3.0
room_type
price_log
0
200
400
600
800
1000
minimum_nights
price_log
0
100
200
300
400
500
600
700
800
number_of_reviews


price_log
0
10
20
30
40
50
reviews_per_month
price_log
0
50
100
150
200
calculated_host_listings_count
price_log
0
50
100
150
200
250
300
350
availability_365
0
10
20
30
40
50

4.2 Build model

R^2 show the relation of variables and target



Some of the top performance models are:

- **Gradient Boosting Regressor**: 0.333503
- **Scaled_BR_reg:** 0.252624

4.3 Training

- Now we start training with the best model found above

- From the above residual plot, the polynomial transformation will be made with a second degree which adding the square of each feature.

```
    from sklearn.preprocessing import PolynomialFeatures
    Poly = PolynomialFeatures(degree=2, interaction_only=True, include_bias=Fals
e)
    X_train = Poly.fit_transform(X_train)
    X_test = Poly.fit_transform(X_test)
```

Finetune best hyperparameters

We use Radomize Search to help us tune the best hyper parameters for BGR model

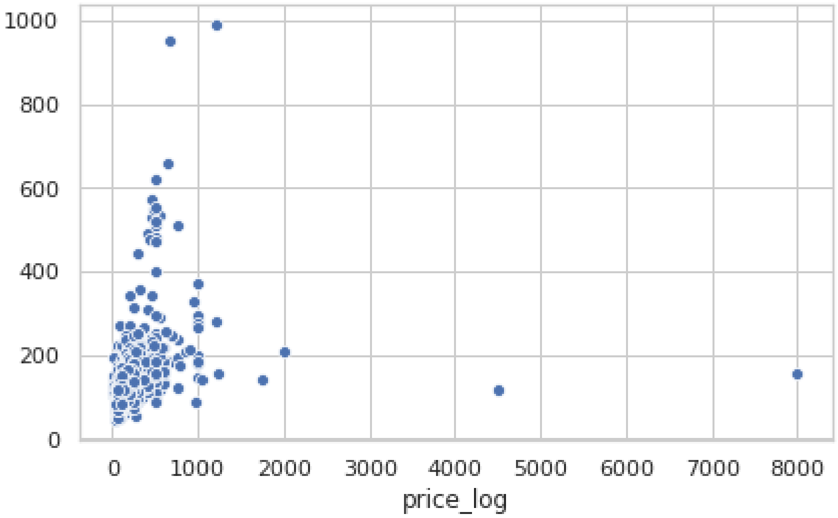
```
Out[160]:  {'n_estimators': 150, 'max_depth': 13, 'loss': 'lad'}
```

Now we train with the full dataset

And the result:

```
MAE:  54.16081010289887
MSE:  29653.278102963508
RMSE:  172.20127207126987
```

```
Out[164]: <matplotlib.axes._subplots.AxesSubplot at 0x128f70898>
```



1000
800
600
400
200
0
0
1000
2000
3000
4000
5000
6000
7000
8000
price_log

5. Conclusion

Summarizing our findings, suggesting other features

- This Airbnb dataset for the 2019 year appeared to be a very rich dataset with a variety of columns that allowed us to do deep data exploration on each significant column presented.
- First, we have been processing and transforming the dataset in order to clean and refine the data with actions, and do a lot of analysis with the data. Further, we started Diagnostic Analysis section to shoe the most used tools of Data Scientists to see what happened and try to understand the past to take advantage in the future using "Matrix Correlation"

- Lastly, we got into price predictive model using the latest stack technology in order to predict the price of Airbnb's over the year. We have used Machine Learning as application of Artificial Intelligence (AI), and we also applied the most optimized and newest algorithms, trying "Gradient Boosted Regressor Model" where we got a positive results coming up with the generalized increase in prices in Amstedam city

- Overall, we discovered a very good number of interesting relationships between features and explained each step of the process. This data analytics is very much mimicked on a higher level on Airbnb Data/Machine Learning team for better business decisions, control over the platform, marketing initiatives, implementation of new features and much more