Capstone 3: Airbnb

By: Pedro Rodriguez

Data

Listing.csv

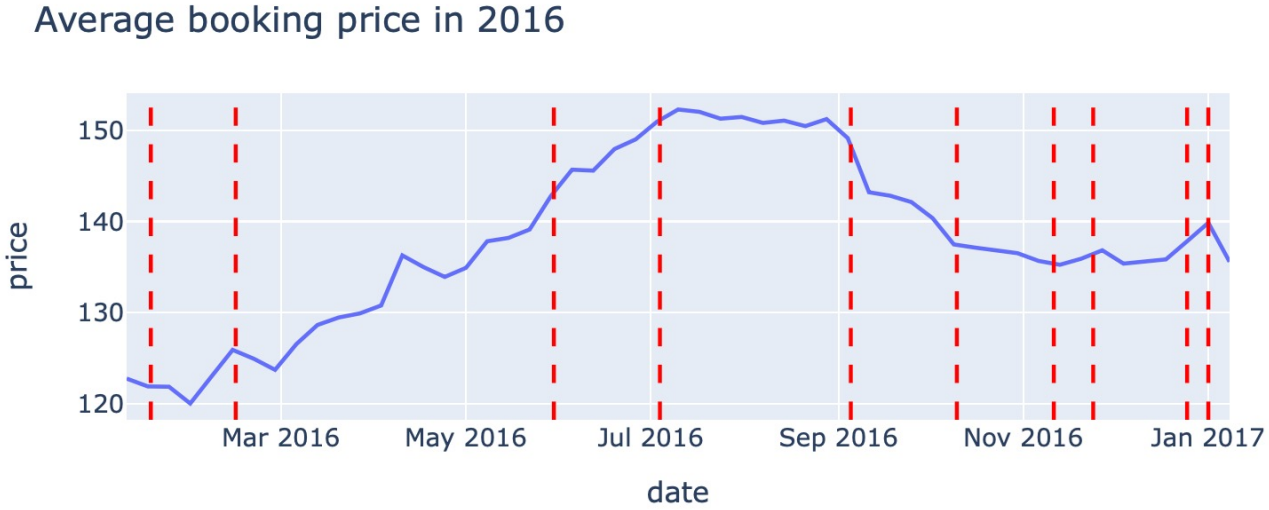
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3817 entries, 0 to 3816
Data columns (total 15 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Unnamed: 0                3817 non-null   int64
 1   id                        3817 non-null   int64
 2   latitude                  3817 non-null   float64
 3   longitude                 3817 non-null   float64
 4   zipcode                   3817 non-null   int64
 5   property_type             3817 non-null   object
 6   room_type                 3817 non-null   object
 7   price                     3817 non-null   float64
 8   accommodates              3817 non-null   int64
 9   cleaning_fee              3817 non-null   float64
 10  availability_30           3817 non-null   int64
 11  bathrooms                 3817 non-null   float64
 12  bedrooms                  3817 non-null   float64
 13  review_scores_rating      3817 non-null   float64
 14  review_scores_cleanliness 3817 non-null   float64
dtypes: float64(8), int64(5), object(2)
memory usage: 447.4+ KB
```

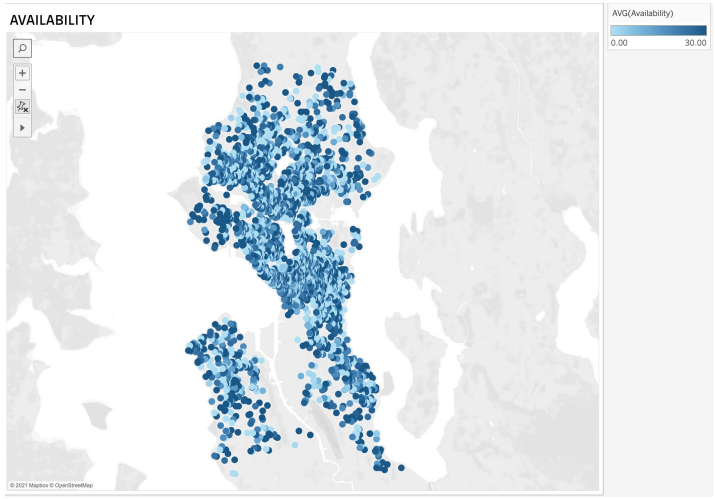
calendar.csv

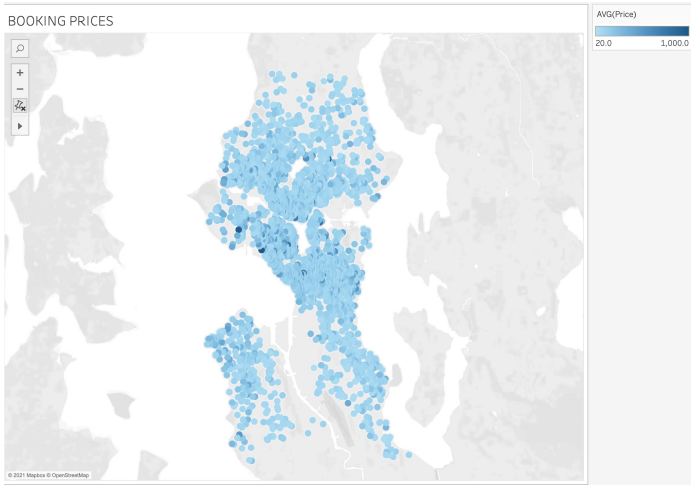
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1393570 entries, 0 to 1393569
Data columns (total 5 columns):
 #   Column      Non-Null Count    Dtype
---  ------      --------------    -----
 0   Unnamed: 0  1393570 non-null  int64
 1   listing_id  1393570 non-null  int64
 2   date        1393570 non-null  object
 3   available   1393570 non-null  int64
 4   price       1393570 non-null  float64
dtypes: float64(1), int64(3), object(1)
memory usage: 53.2+ MB
```

2016


Weekly Availability
available
15k
10k
5k
0
Jan 2016
Mar 2016
May 2016
Jul 2016
Sep 2016
Nov 2016
Jan 2017
date

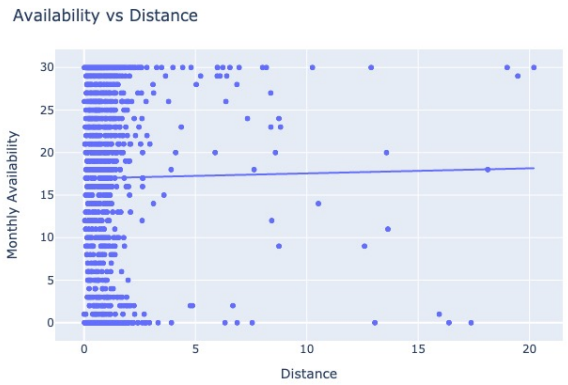

Average booking price in 2016
price
150
140
130
120
Mar 2016
May 2016
Jul 2016
Sep 2016
Nov 2016
Jan 2017
date


AVAILABILITY
AVG(Availability)
0.00
30.00


BOOKING PRICES
AVG(Price)
20.0
1,000.0

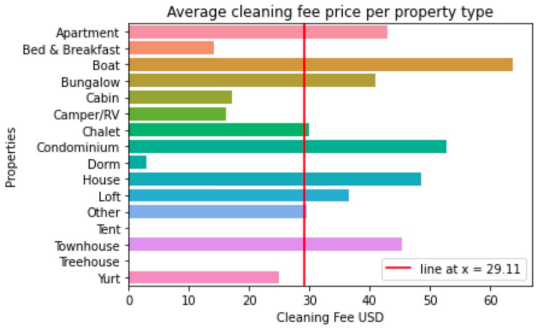
Locations

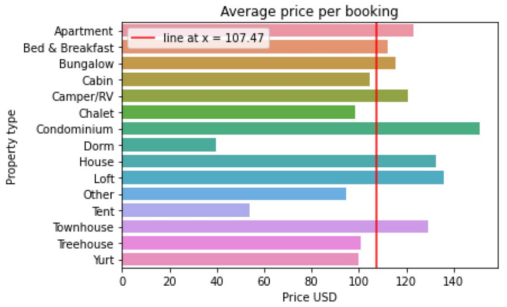

Price vs Distance
Price USD
Distance
0
200
400
600
800
1000
0
5
10
15
20

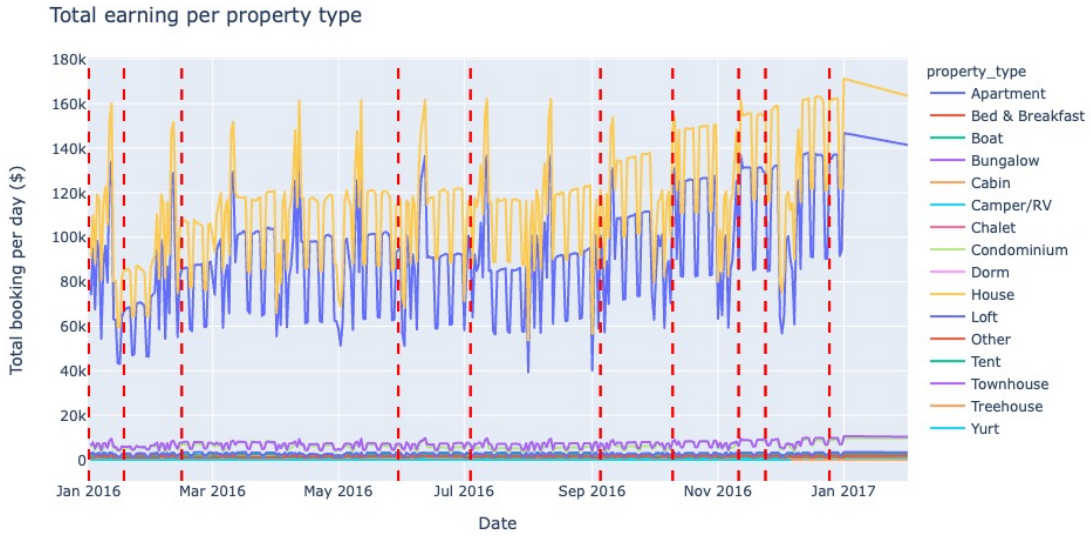

Availability vs Distance
Monthly Availability
Distance
0
5
10
15
20
25
30
0
5
10
15
20

Property Type

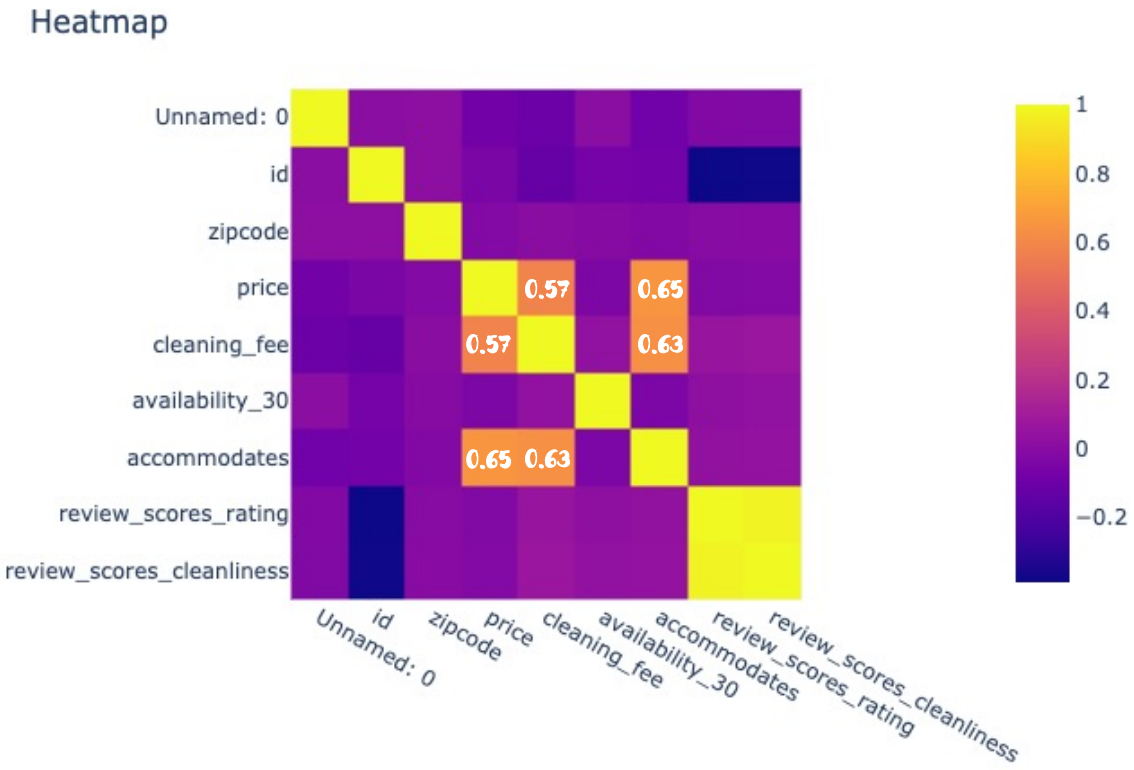
Property	Average Price	Average Cleaning fee
Condominium	$151.11	$54.51
Loft	$135.70	$37.09
House	$132.35	$42.24
Apartment	$122.93	$38.58







Relationships



Relationships

Prediction Model Construction

- Linear Regression
- Random Forest
- Gradient Boost
- Extreme Gradient Boost

```
from sklearn.model_selection import train_test_split

X = house_scaled[['distance', 'accommodates', 'availability_30','cleaning_fee', 'bedrooms',
                  'bathrooms', 'property_type_Apartment', 'property_type_House']].values
y = house_scaled['price'].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)
```

Features	Importance
Bedrooms	0.449162
cleaning_fee	0.152625
distance	0.129531
availability_30	0.084503
bathrooms	0.079779
accommodates	0.077149
property_type_House	0.013932
property_type_Apartment	0.013319

Models Result

- The prediction model doesn't perform well with Apartment and House prices.
- The prediction model predicts better the house's prices.

For House and Apartment's prices

Model	Test Score	MAE	RMSE
Base Linear Regression	0.5228	0.4232	0.6294
Base Random Forest	0.4894	0.4288	0.6511
Base Gradient Boost	0.5194	0.3939	0.6317
Base XGB	0.488	0.4156	0.652

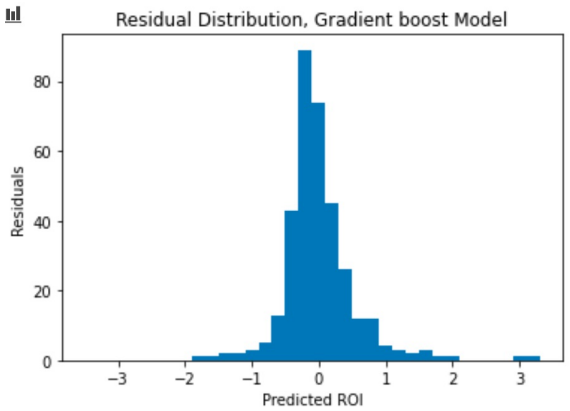
For House's prices

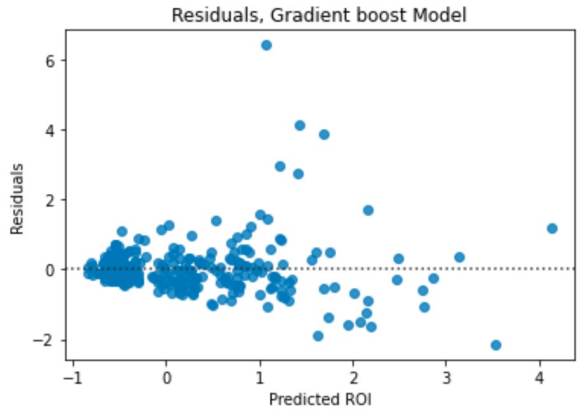
Model	Test Score	MAE	RMSE
Base Linear Regression	0.604	0.4029	0.7332
Base Random Forest	0.5837	0.41	0.7518
Base Gradient Boost	0.5971	0.3998	0.7396
Base XGB	0.5664	0.4363	0.7673

Models tuning

Model	Test Score	MAE	RMSE
Base Linear Regression	0.604	0.4029	0.7332
Base Random Forest	0.5837	0.41	0.7518
Base Gradient Boost	0.5971	0.3998	0.7396
Base XGB	0.5664	0.4363	0.7673

Tune Model	Test Score	MAE	RMSE
Tune Linear Regression	0.604	0.4029	0.7332
Tune Random Forest	0.6141	0.4083	0.701
Tune Gradient Boost	0.6322	0.3861	0.7067
Tune XGB	0.6286	0.3949	0.6876





Q & A