Predicting the Likelihood of an Airbnb Reservation to be Cancelled

Anna Apa

Capstone Project 1

Problem Statement

- As part of the sharing/peer economy, Airbnb hosts become the key determinants of the guest’s experience with the company

- This experience can determine whether or not the guest will continue to use the app in the future

- Along with the price and ratings shown in their listings, an Airbnb host’s likelihood of cancelling should also be readily available for prospective guests to see

Objectives:

- Predict likelihood of an Airbnb listing reservation to be cancelled
- Increase transparency and incentivize Airbnb hosts to cancel on guests less

Datasets

- In this project, I focus on the Airbnb market in London, the second largest Airbnb city outside the US
- The following csv files were downloaded from insidearibnb.com:
  - listings.csv
  - reviews.csv
  - neighbourhoods.csv
- I also downloaded the latest London crime report from the “London Datastore” website:
  - crimes.csv

Data Wrangling —reviews.csv

listing_id
id
date
reviewer_id
reviewer_name
comments

- Contains 1,249,466 rows and 6 columns
    - Each row represents a review left on a listing
- Dealing with null values:
    - Drop empty reviews
- 1,249,466 rows → 1,211,982 rows

Getting the Number of Cancellations Made by Each Listing (Target Variable)

reviews.csv

- When a host cancels on a guest, an automated posting is posted on the listing’s page. This posting cannot be deleted.
- There has been two versions of this automated posting:
  - “The host canceled this reservation n days before arrival. This is an automated posting.”
  - “The reservation was canceled n days before arrival. This is an automated posting.”
- We group these automated postings by listing and apply .value_counts to get the number of cancellations made by each listing

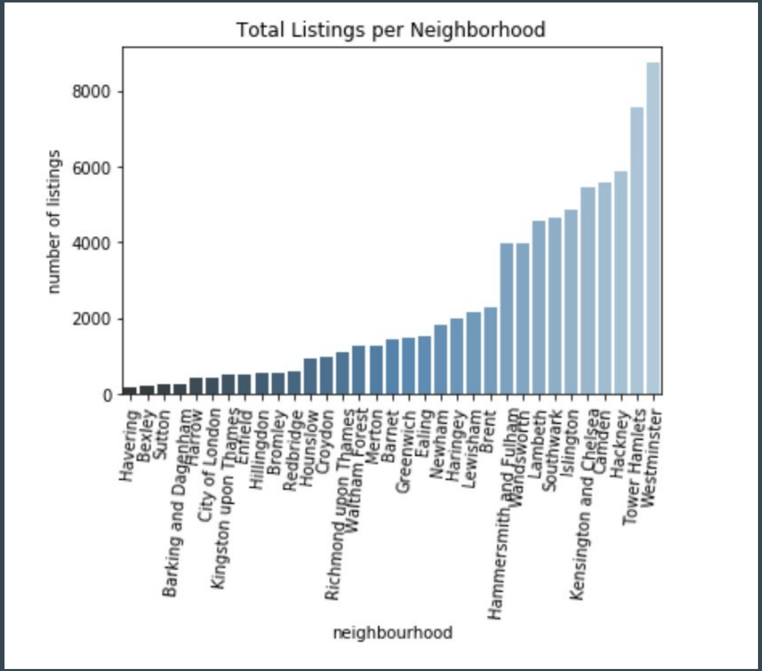
Data Wrangling—listings.csv

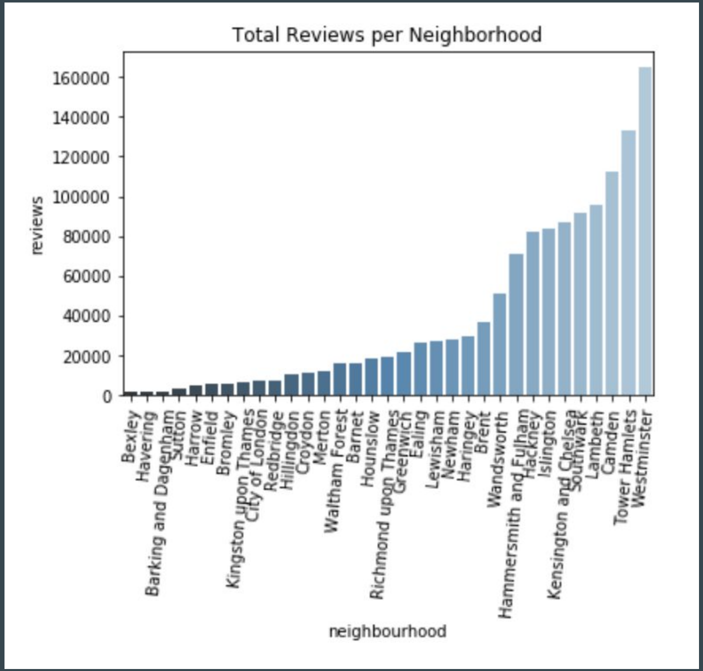
listings.csv

id
name
host_id
host_name
neighbourhood_group
neighbourhood
latitude
longitude
room_type
price
minimum_nights
number_of_reviews
last_review
reviews_per_month
calculated_host_listings_count
availability_365

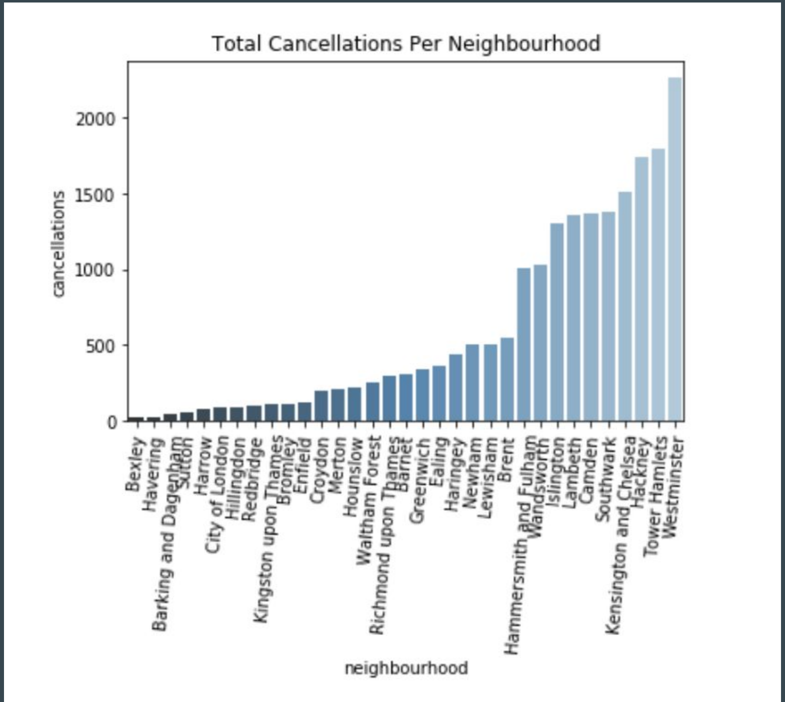
- Contains 80,767 rows and 16 columns
  - Each row represents a unique listing in London
  - Each column represents a feature
- Dealing with null values:
  - Replace missing values with 0
  - Drop inactive listings (‘avaibility_365’==0)
- 80,767 rows → 78,074 rows

Number of Airbnb Listings and Reviews Per Neighbourhood

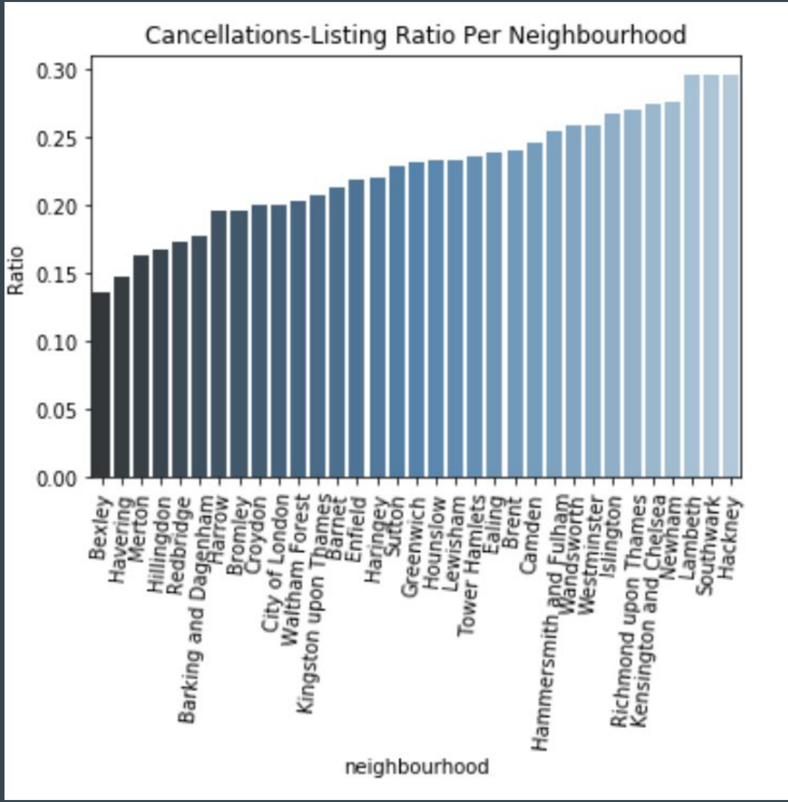




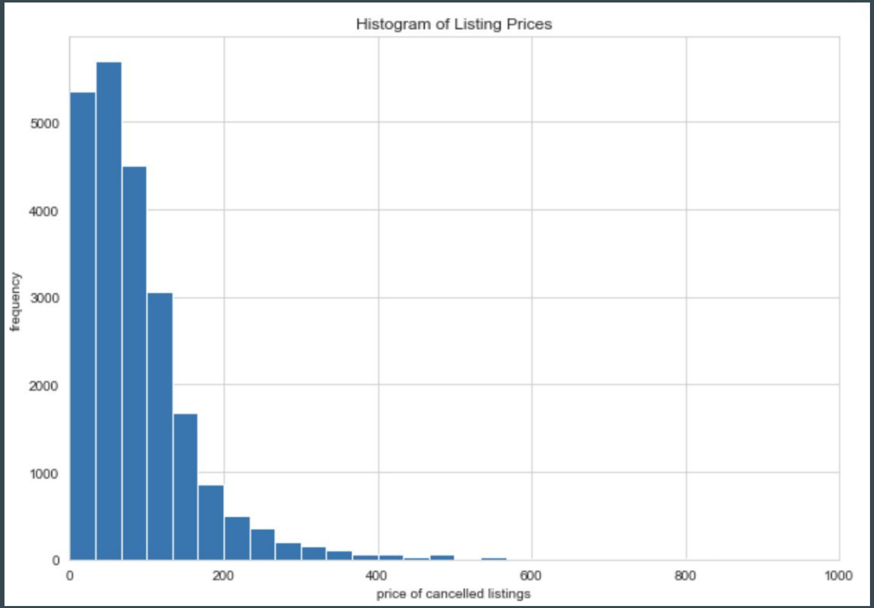
Number of Reservation Cancellations Per Neighbourhood



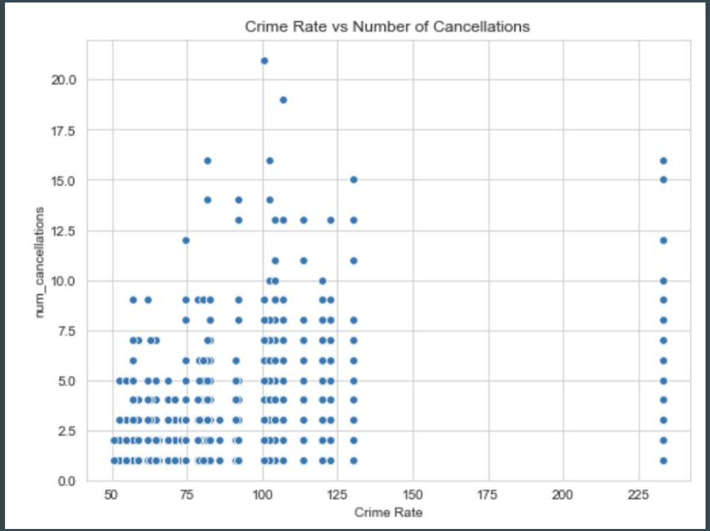
Cancellation-Listing Ratio Per Neighbourhood



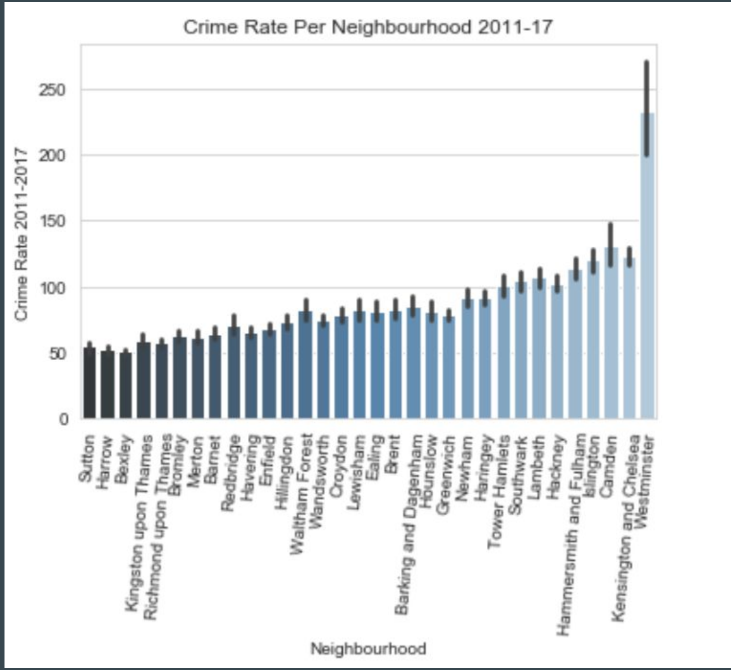
Does Price Correlate with these Cancellations?



Does the Neighborhood Crime Rate Correlate with the Cancellations in the Area?



Which Neighborhood Currently has the Highest Crime Rate?



Crime rates from 2011 (earliest Airbnb review in dataset) - 2017 (latest)

Which Features have a Statistically Significant* Correlation with the Target Variable?

Feature	Observed Correlation	P-Value	Statistically Significant*?
Price	-0.02526	0.00010	**yes**
Number of Reviews	0.21254	0.00000	**yes**
Days Booked (Demand)	-0.0110	0.00220	**yes**
Minimum Nights	-0.00473	0.16280	**no**
Crime Rate	0.01161	0.00090	**yes**

*statistically significant at the 5% level

Does the Number of Cancellations Vary Significantly* Across Neighborhoods and Room Types?

Feature	F-Stat	P-Value	Statistically Significant*?
Neighborhood	4.65329	0.0000000	**yes**
Room Type	11.68846	0.0000084	**yes**

*at the 5% level

Machine Learning

In this study, we try the following estimators and see which one performs best

- Linear Regression
- Ridge Regression
- Lasso Regression
- Decision Trees
- Gradient Boosting
- Random Forest

Machine Learning—continued

Dealing with Categorical Features

- We create dummies for features ‘room_type’ and ‘neighbourhood’

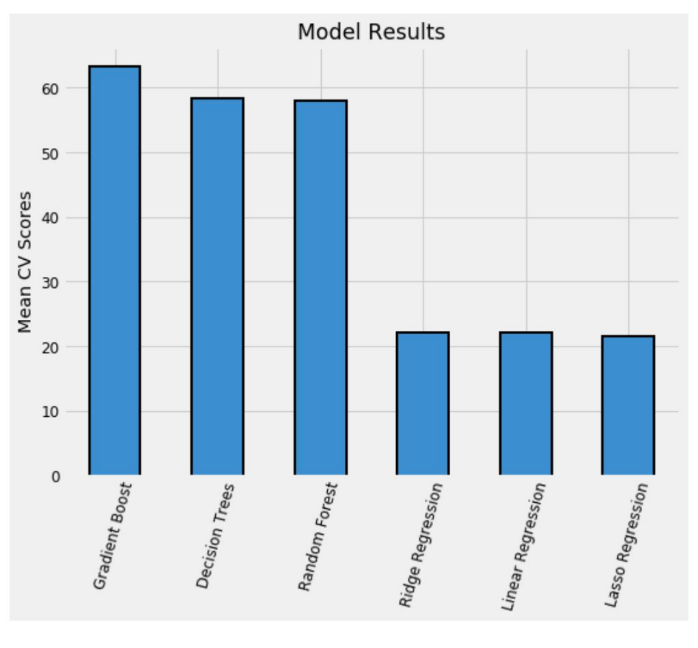
Training Set and Test Set

- We are using 70% of the data as training set and the remaining 30% as the test set

Scoring

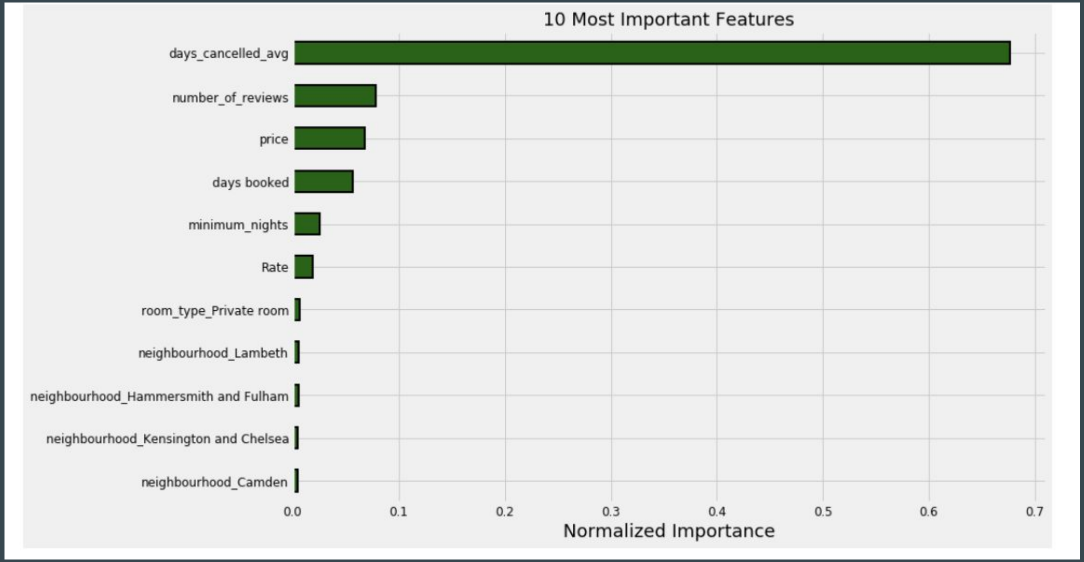
- We are measuring accuracy by getting the average of 5-fold cross validation scores of each model, and use R^2 as the scoring metric

Comparing Model Performance



	cv_mean
Gradient Boost	0.632205
Decision Trees	0.583318
Random Forest	0.580236
Ridge Regression	0.221451
Linear Regression	0.221447
Lasso Regression	0.216374

Feature Selection



Hyperparameter Tuning

Since our Gradient Boosting Regressor model performed best, we can improve the performance further by tuning its parameters.

```
{'best_n': 50, 'best_max_depth': 3, 'best_lr': 0.1}
```

After using these parameters, R^2 increases from 0.6336 to 0.6352

Results

```
Train set score: 64.60%
Test set score: 61.45%
```

Limitations

Model can be improved with more features (data) on each listing

Some information provided by insideairbnb.com were time sensitive--such as the availability of each listing, which only shows the data from the past year.