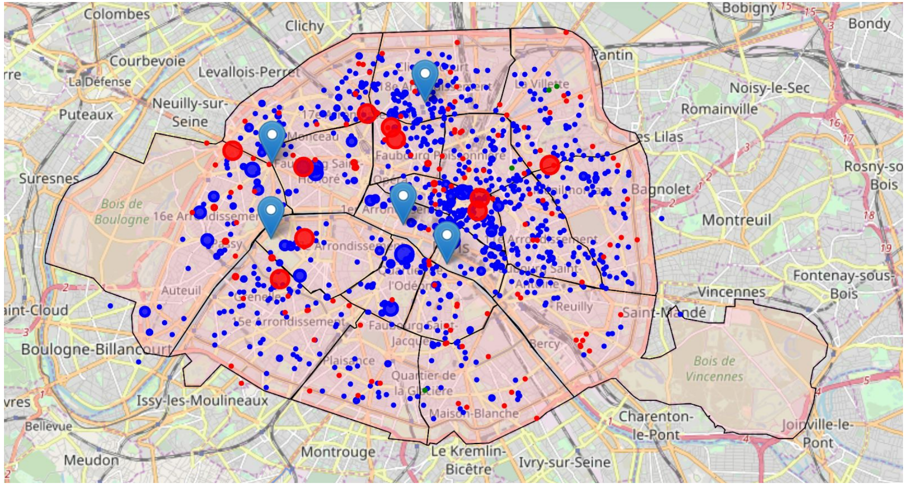
How to get a deal on AirBnB listings in Paris?



Don’t just travel. Travel right

- Who likes to travel? - Well, Who does not?
- Where do people stay while travelling?
    - Family and Friends
    - Hotels
    - AirBnB
- Why do travelers choose AirBnB over hotels?
    - More space
    - More and better amenities (kitchen, washer and dryer, faster wifi, etc.)
    - Less costs
    - More local experiences

The City Of Love - Paris

- One of the most visited European cities (2nd place after London)
- Over 15 million travelers per year
- Over 77,000 listings on AirBnB
- How to get a good deal out of such a variety?

Let’s take a look!

How to get listing data?

- Data source - http://insideairbnb.com/get-the-data.html
- Data sets:
  - **Listings** - over 60K rows - information about each individual listing
  - **Calendar** - over 22MM rows - information about listing availability throughout the year
  - **Reviews** - over 1.1MM rows - information with transcript of all reviews
  - **Neighbourhoods** - geojson file with geographical lines of Paris neighbourhoods

What was rented before?

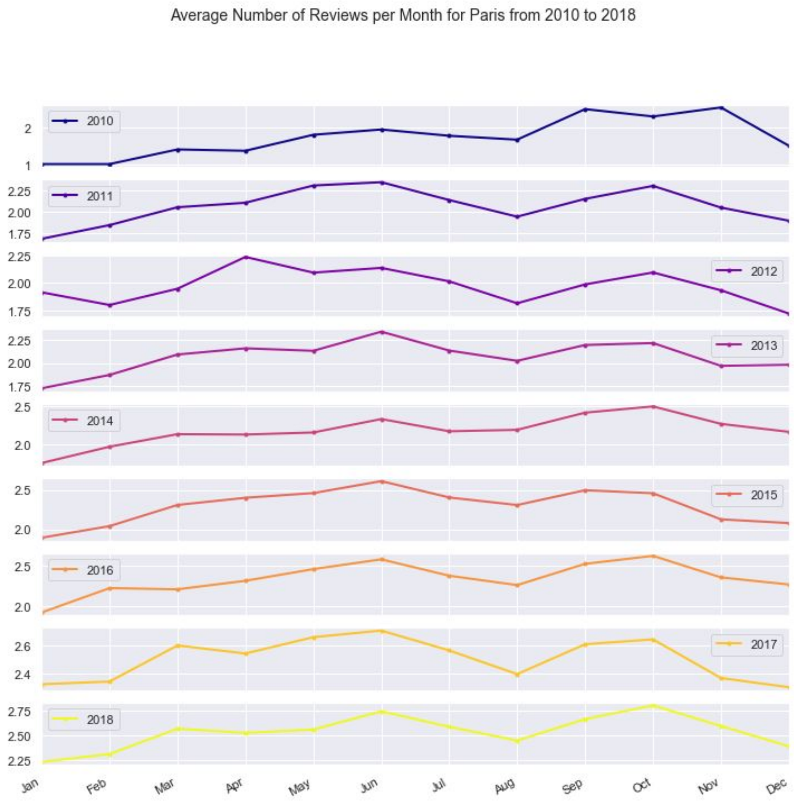
- No official booking data from AirBnB
- San-Francisco model for Occupancy rate =

Average Length of Stay * Reviews per Month / Review Rate

- Average Length of Stay = 5.2 nights for Paris
- Review Rate = 50% (every second guest leaves review)

When do people travel to Paris?

- Lowest demand in winter
- Most travellers in spring and fall
- Demand drop in summer
- August is the slowest summer month



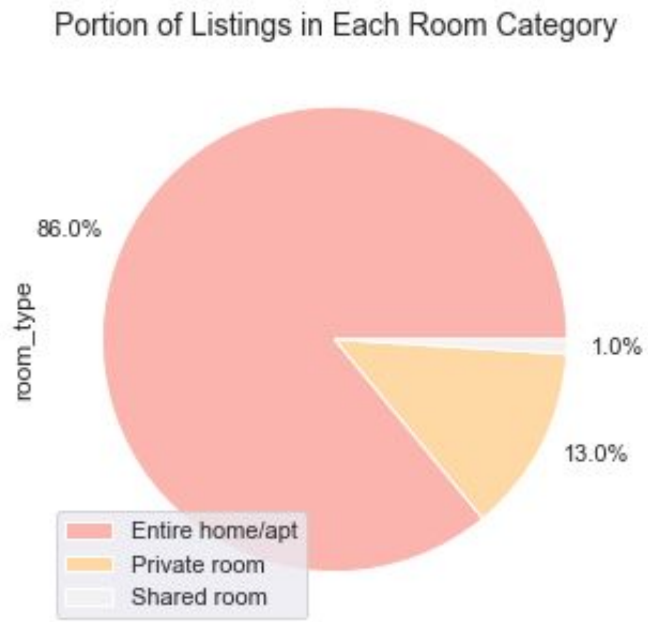
Does supply support demand?

- Most availability in summer
- Hosts travel themselves too
- Increase for winter holiday season

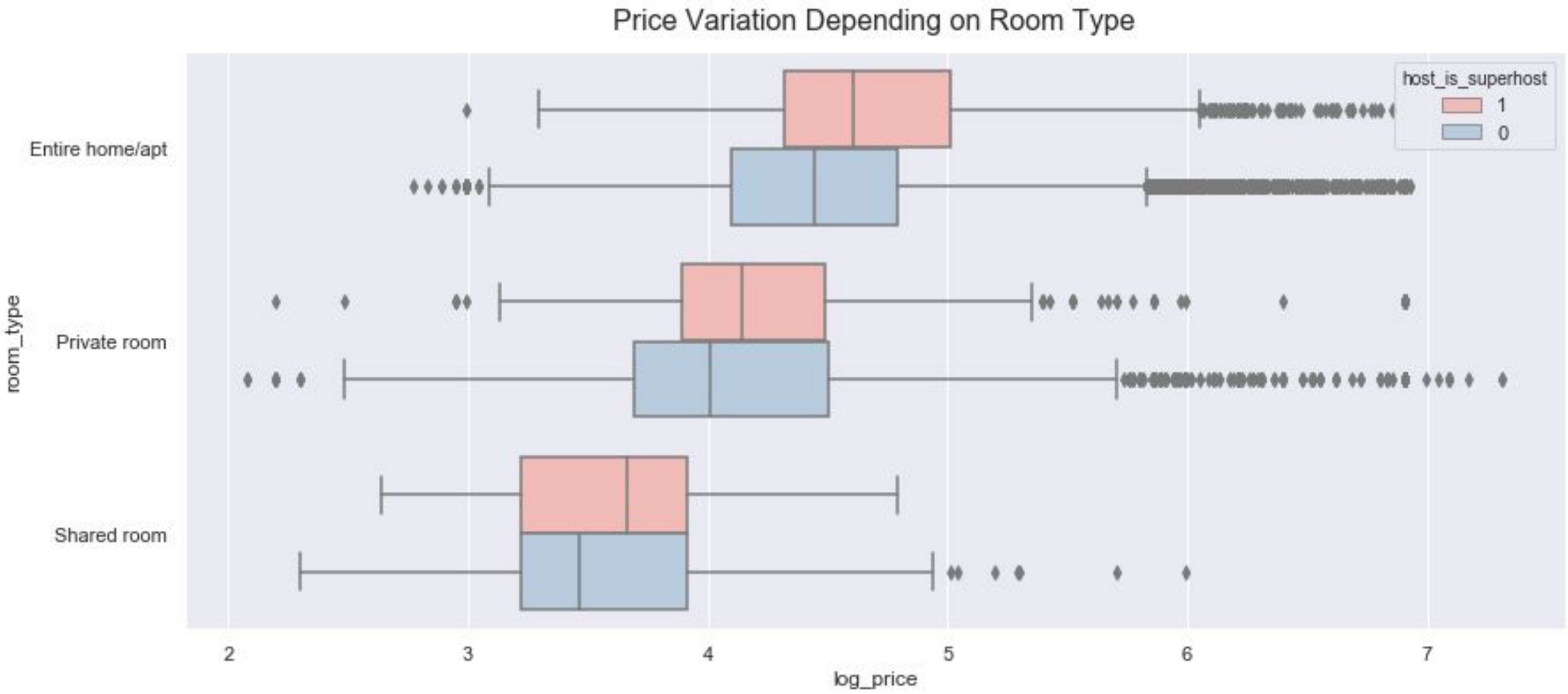
And the price?

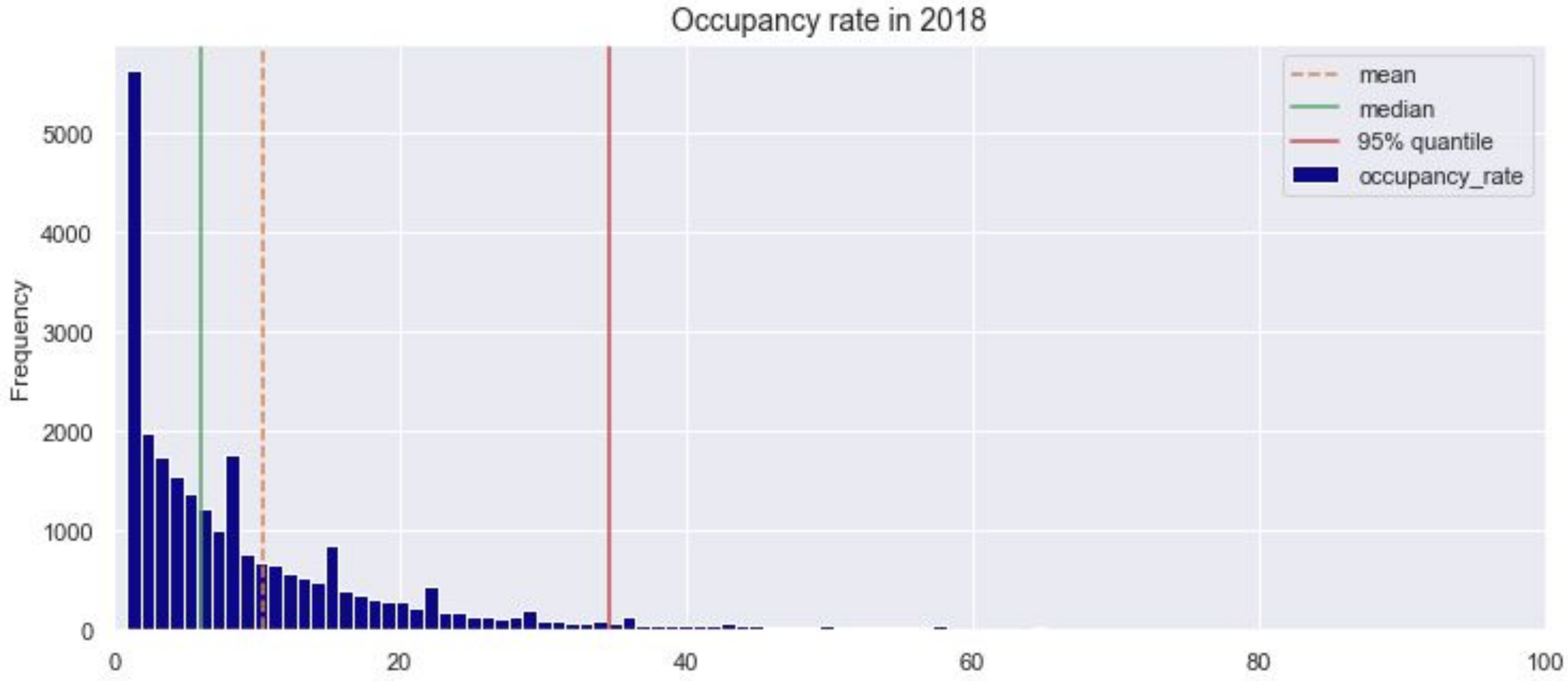
- Clear correlation of price and demand
- Most expensive in May and October
- Cheapest in August - supply is higher than demand

What is being offered?



How much does it cost?



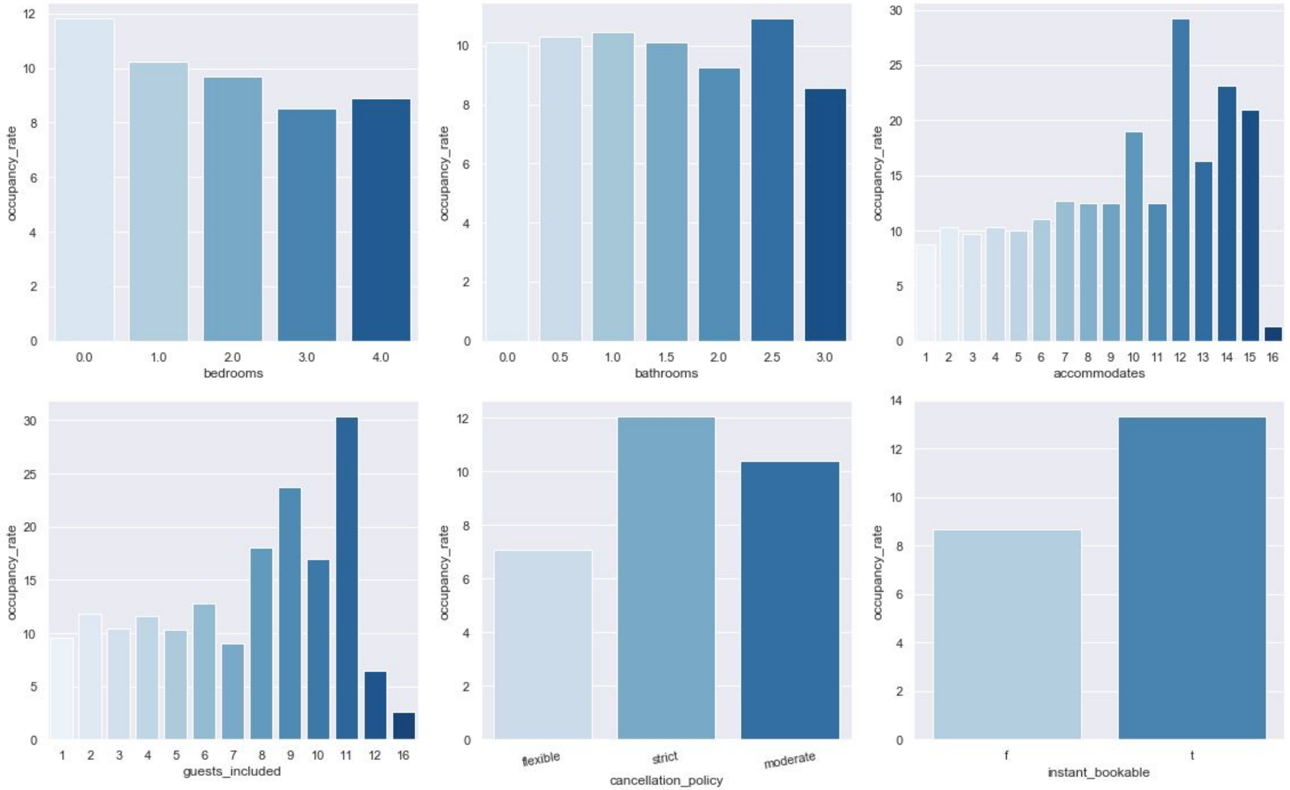


Entire Apartments only - 10% average occupancy rate

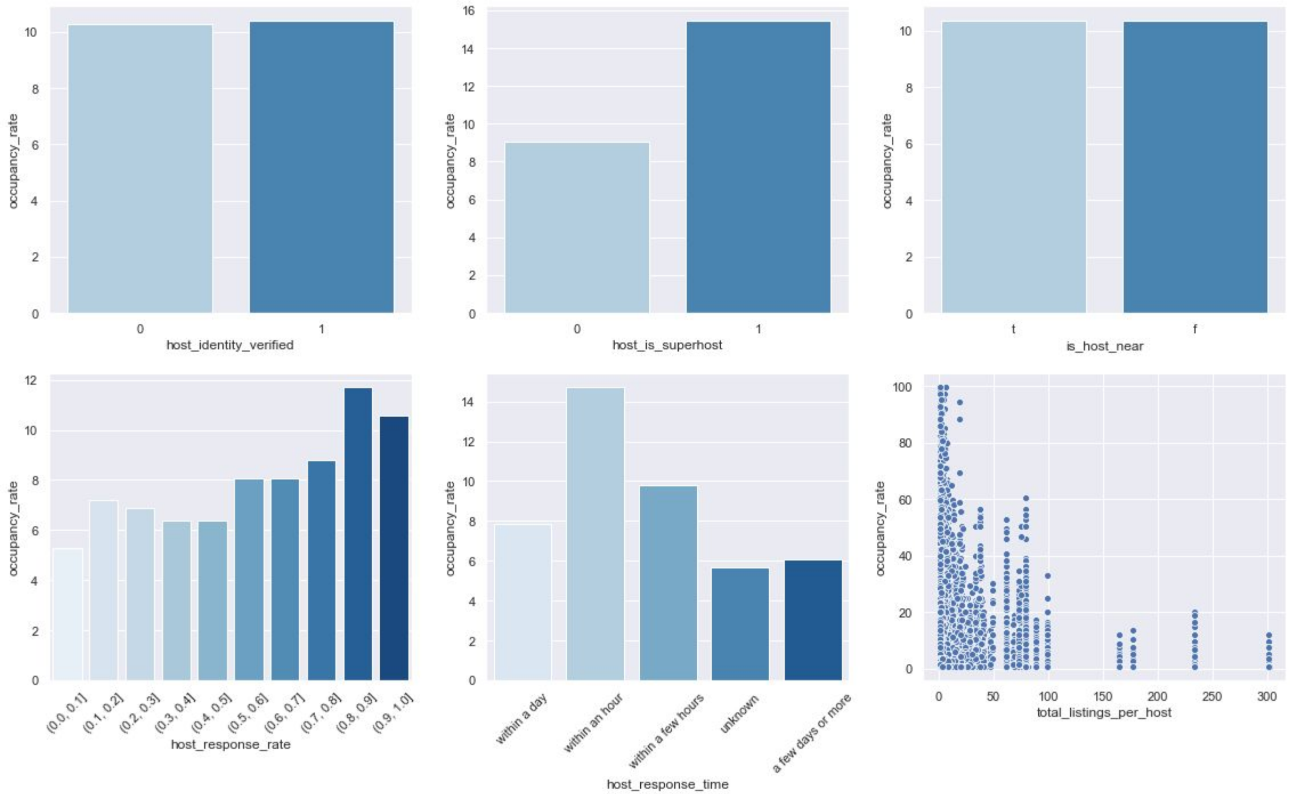
What gets the place rented?

- **Apartment characteristics** (number of bedrooms, bathrooms, instant bookable)
- **Host characteristics** (superhost status, host response rate and time)
- **Opinions about the place** (review ratings of previous guests)

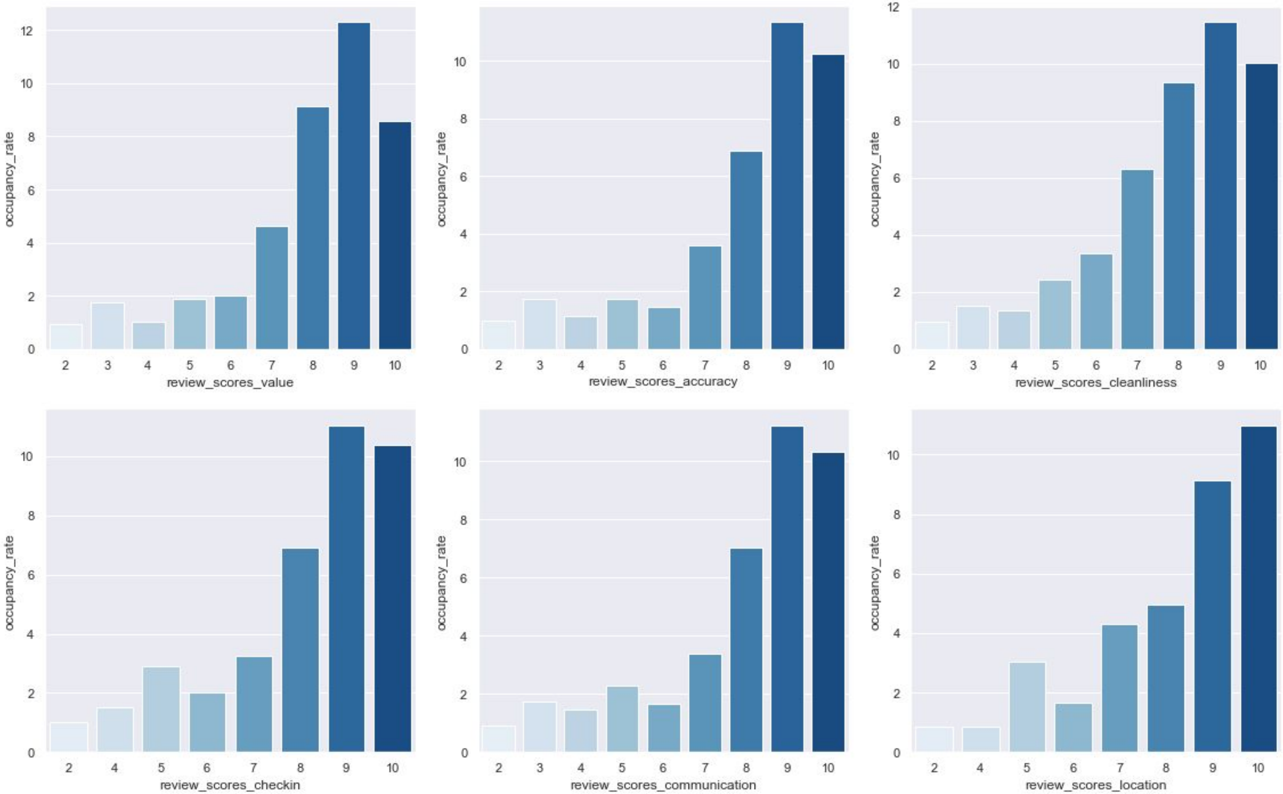
Apartment Characteristics



Host Characteristics



Review Ratings

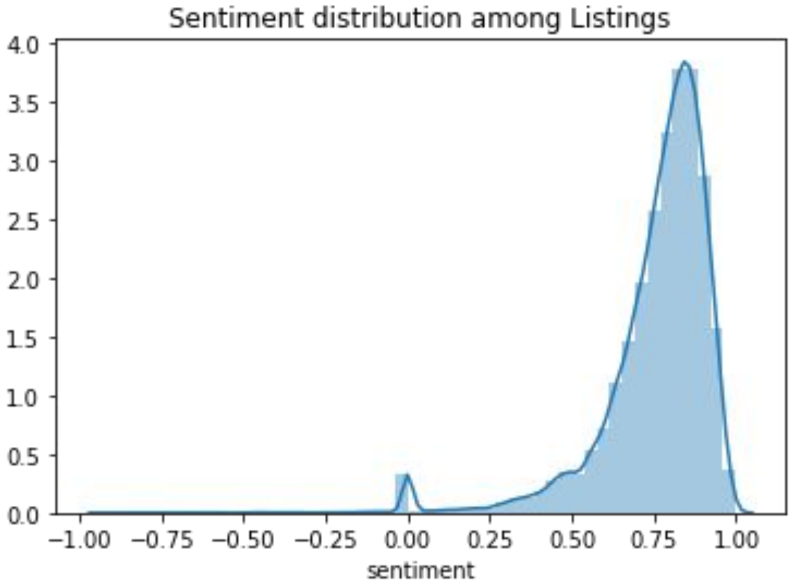


What matters most?

- Possibility to accommodate larger groups
- Ability to rent the place right away without host's approval
- Fast and clear communication with host
- Getting a place from superhost
- Location of the place
- Experiences of previous renters

Sentiment Analysis of Reviews

- Not just review scores
- Actual opinions
- Over 800,000 reviews
- Translate over 300,000 into English with googletrans
- Sentiment score with VaderSentiment
- Average score per listing



What would be a good deal?

- Top 25% of sentiment score
- Below average neighbourhood price
- Top 25% of each score rating:
  - Accuracy
  - Cleanliness
  - Check-in
  - Communication
  - Location
  - Value
  - Overall score
- 1,239 good deals out of 25,991 listings ~ 5%

Can a good deal be predicted?

- Classification task
- Supervised models (pre-labeled data)
- Problem:
  - Imbalanced Data - 95% of majority class
- Decision Metrics
  - Precision = positive predictive value
  - Recall = sensitivity
  - Need balance of both

How to deal with imbalanced data?

Groups	Type of data balancing	F1 Score	Precision Score	Recall Score
Original	Imbalanced data	0.69	0.77	0.63
Under-Sampling	Random Under-Sampling	0.53	0.36	0.98
	NearMiss	0.63	0.52	0.82
	ENN	0.71	0.70	0.72
Over-Sampling	Random Over-Sampling	0.58	0.41	0.98
	SMOTE	0.60	0.44	0.97
	SMOTENC	0.60	0.45	0.93
	ADASYN	0.58	0.41	0.99
Combination	SMOTEENN	0.56	0.39	0.98

What models to use?

- Logistic Regression
- kNN - k-Nearest Neighbours
- SVM - Support Vector Machines
- Naive Bayes
- Decision Tree
- Random Forest
- AdaBoost
- Gradient Boosting

How did each model perform? - Top 10 Results

Classifier Name	F1 score	Precision score	Recall
Random Forest	0.836	0.846	0.827
AdaBoost	0.866	0.834	0.901
Gradient Boosting	0.835	0.831	0.838
SVM	0.717	0.815	0.639
Random Forest ENN	0.850	0.799	0.906
Decision Tree	0.790	0.794	0.787
Decision Tree SMOTENC	0.823	0.791	0.858
AdaBoost ENN	0.849	0.784	0.926
Decision Tree ENN	0.826	0.782	0.875
Gradient Boosting ENN	0.840	0.773	0.920

How to Improve Results?

- Voting classifier - combination of 3 winning models (same precision, slight improvement on recall and f1 score)
- Hyperparameter tuning for Random Forest (3% improvement for precision, recall and f1 score)
- Unsuccessful tries to improve results:
  - Dimension reduction with Principal Component Analysis
  - Feature removals
  - Feature interaction term introduction
  - Hyperparameter tuning on AdaBoost

Random Forest

And the winner is …. with tuned parameters

Precision: 0.87 Recall: 0.85 F1-score: 0.86