Setting a Competitive First Listing in London!

with Machine Learning

Outline

- **Context and Problem definition:** What makes a listing sand out?
- **Gaining Insights from data:** What the data tells us about existing Listings?
- **Modeling technique**: XGBoost and Genetic Algorithm for feature selection.
- **Modelling first listing Price and an Overall Review Value**:
  - Feature selection
  - Model evaluation and tuning
  - Interpretation and analysis
- **Future Improvements**

Context and Problem definition: What makes a listing sand out?

- Help new hosts to establish their listings successfully in the Airbnb market.

- **Goal**: Maximise the chance of a new listing to appear in the Airbnb search ranking algorithm.

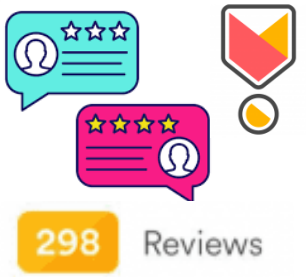
- **Solution**:
  - Build a machine learning model for **predicting** a **competitive price** for the first listing.

  - Build a ML that **predicts a new overall review metric** that reflects the over all consumer response of satisfaction. This metric is **intended to advise the host** how to **improve** their **customers' response**.

Some metrics to improve chances for a listing to be ranked

Price (No1)

Complete and accurate



Quantity and Quality

Verification Level

Host Response

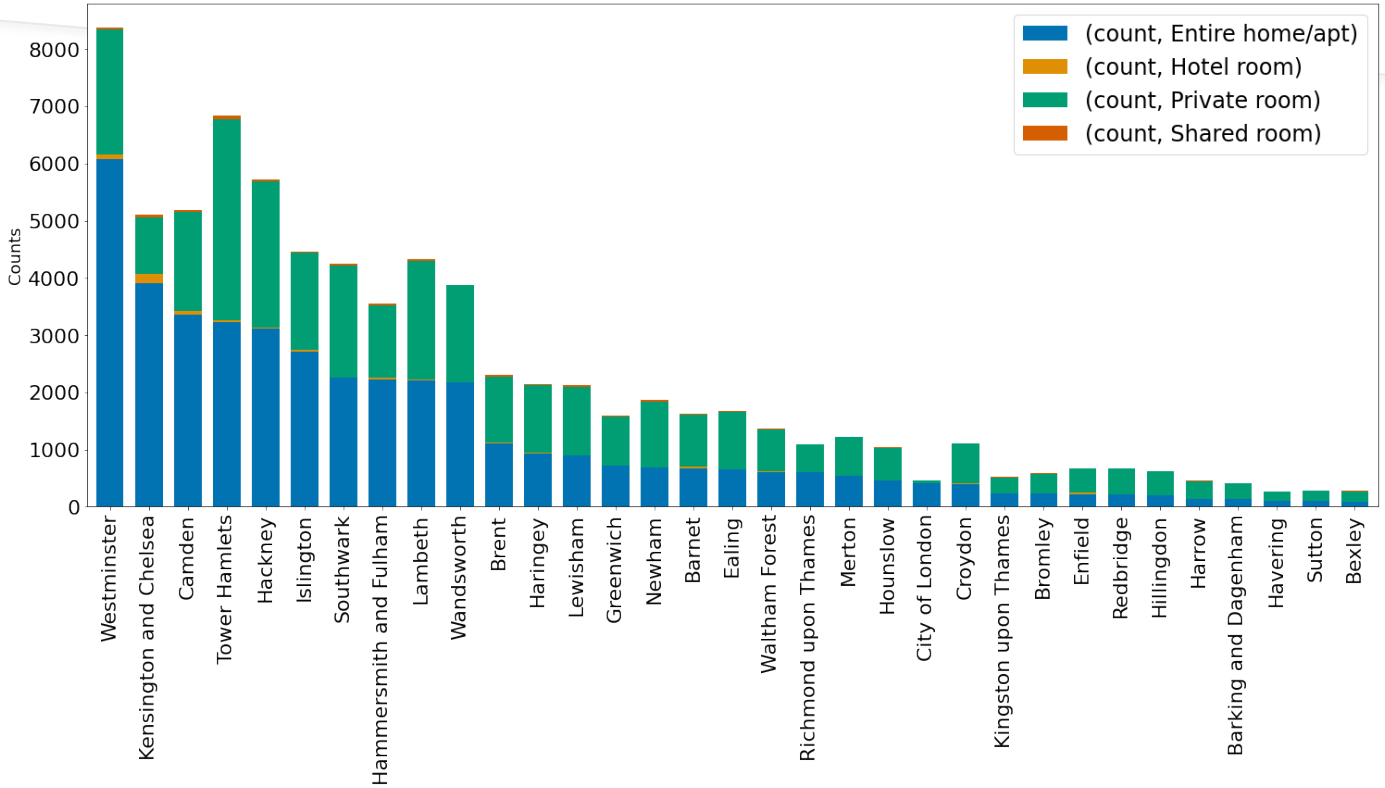
Instant Booking

Nights

Location

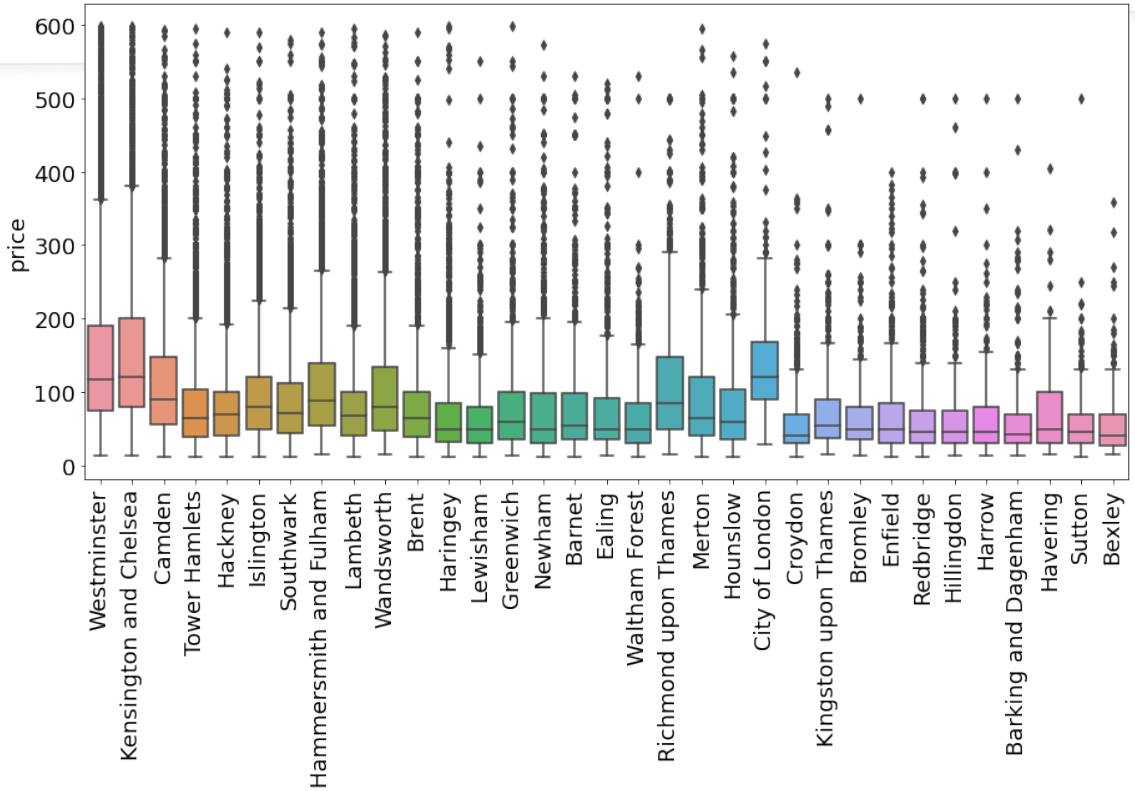
Gaining Insights from data: What the data tells us about existing Listings?

Type of Airbnb in London Boroughs



- Most of the Boroughs closer to Central London are the ones with more properties.
- Mostly there are entire home/apt and private Rooms.
- Higher number Hotel rooms are in Kensington & Chelsea, Westminster, & Camden.

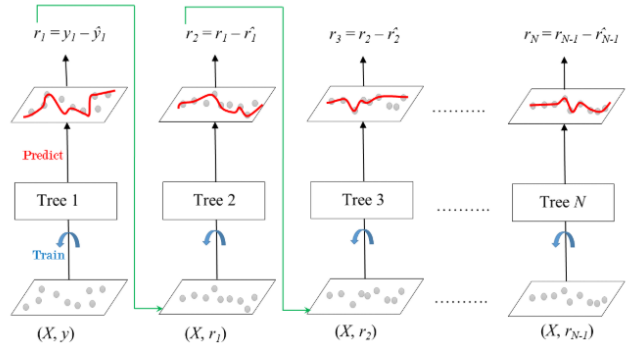
Price Distribution



- The prices in Westminster, Kensington & Chelsea, and City of London tends to be over the median of £75.
- The price distribution is skewed towards higher prices.

Modeling techniques: XGBoost Regression

- This is an **additive sequential tree base ensemble** algorithm: The base learners are classification and regression trees (CART).
- At each iteration, a new predictor is added and trained with respect to the residual errors committed by the preceding predictors.



Advantages:

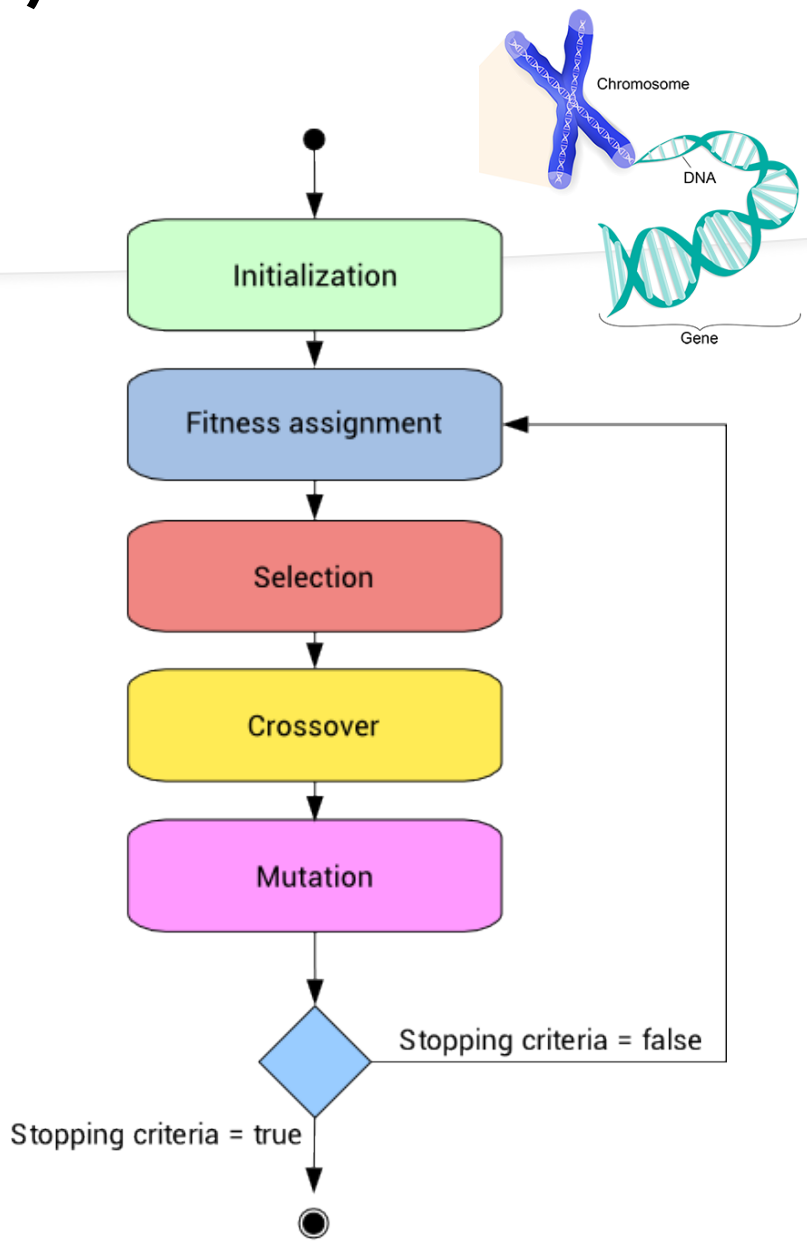
- Boosted tree methods give better model performance.
- No data pre-processing required, can take categorical/numerical data, and handles missing data (not need of mean-imputation)
- Handles large-sized data sets and it is less prone to overfitting.
- Tree-based models are more transparent and interpretable than some other ML techniques.

Disadvantages:

- GBM will continue improving to minimize all errors. This can give more importance to outliers and cause overfitting (CV).
- Computationally expensive since it requires many trees (time and memory exhaustive).
- High flexibility with many parameters that interact and influence the behaviour of the algorithm (large grid search for tuning).
- Less interpretable than non-ML algorithms, but it can be addressed by looking at feature importance, partial dependence plots, SHAP values, etc.

Modeling techniques: Genetic Algorithm (GA) for Feature Selection

- Feature selection is the process of **identifying and removing irrelevant features** that do not contribute to the performance of the model.
- Selecting the right subset of variables that gives the best prediction is a **combinatory and optimisation problem**.
- GA is base on the idea of ***natural selection*** where the fittest individuals or genes are selected over different generations.
- The advantage of GA over other methods is that it allows the **best solution to emerge from best priory solution**.
- It **combines** the different **solutions generation after generation**, extracting the best genes or variables, so it creates **new and more fit individuals**.

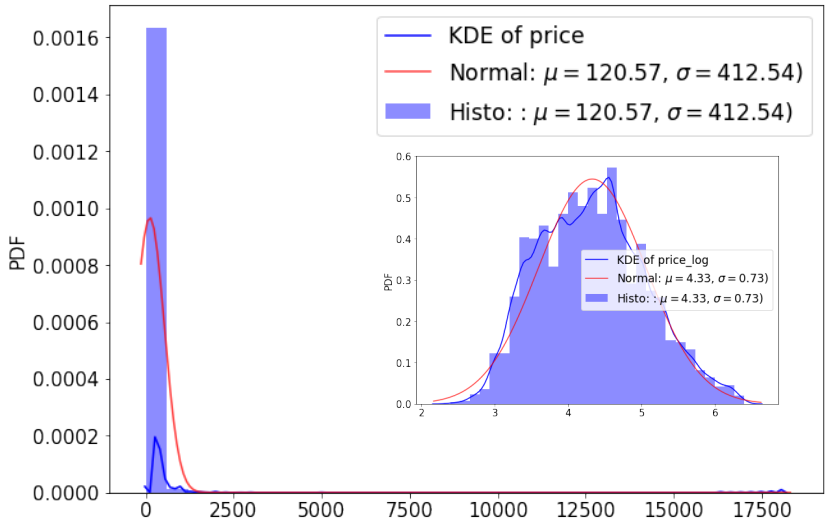


Modelling First Listing Price

Modelling First Listing Price

Data preparation and feature selection

- The model is using features just related to the *property*, the *location*, and *amenities*.
- They were cleaned, transformed some categorical to numerical, and the top 42 more frequent amenities were one-hot encoded.
- The Price has a skewed distribution due to outliers. These were removed (only kept £10 < price < £600) so the final data set for modeling contains 98.7% of the original data: 76,114 out of 77,136.
- A log transformation was performed to the price, so it follows a normal-like shape distribution.
- Data set split 80% Train (60,891) / 20% Test (15,223)
- Our fitness or performance metric is the RMSE, which is used given that it penalises outliers and it is easier to interpret.
- The feature selection process started with 63 variables and, after 3 rounds of 10 generations each, 12 of them survived.



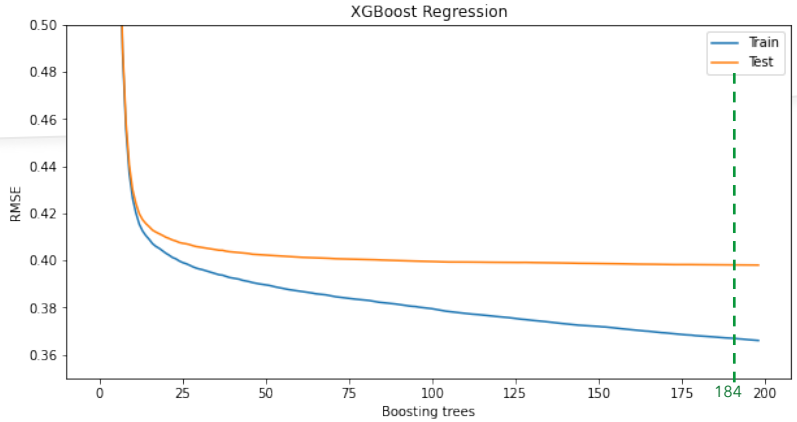
Prices over £600 and less than £10 were removed

Modelling First Listing Price: Model tuning and evaluation

- The model was tuned by a non-exhaustive grid search and cross-validation technique:

Parameters	Description
max_depth, max_child_weight	Constraint the architecture or complexity of the trees
subsample, colsample_bytree	Control the sampling of the data set at each boosting round.
learning_rate, reg_lambda	Controls the gradient descendent steps for learning & regularisation on weights to control overfitting.
num_boost_round	number of boosting trees.

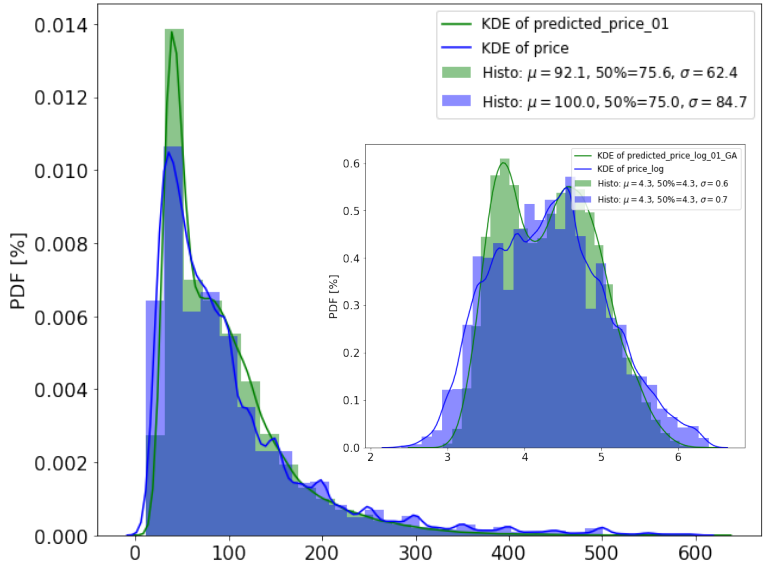
Learning Curve (trade off bias-variance)



Model	MRSE (£)	Relative Uplift w.r.t base %
Base line (mean of price)	2.076	
Model 64 variables	1.492	28.12%
Model 12 variables (GA)	1.507	27.40%
Final model after tuning	1.483	28.56%

	Feature_importance%
room_type_num	71.541931
bedrooms	7.528047
accommodates	6.514672
bathrooms_shared	4.043593
tv	3.436185
dishwasher	2.866000
availability_30	1.106085
longitude	0.873748
latitude	0.860276
hot_water	0.540220
maximum_nights	0.489207
cable_tv	0.200041

[1]



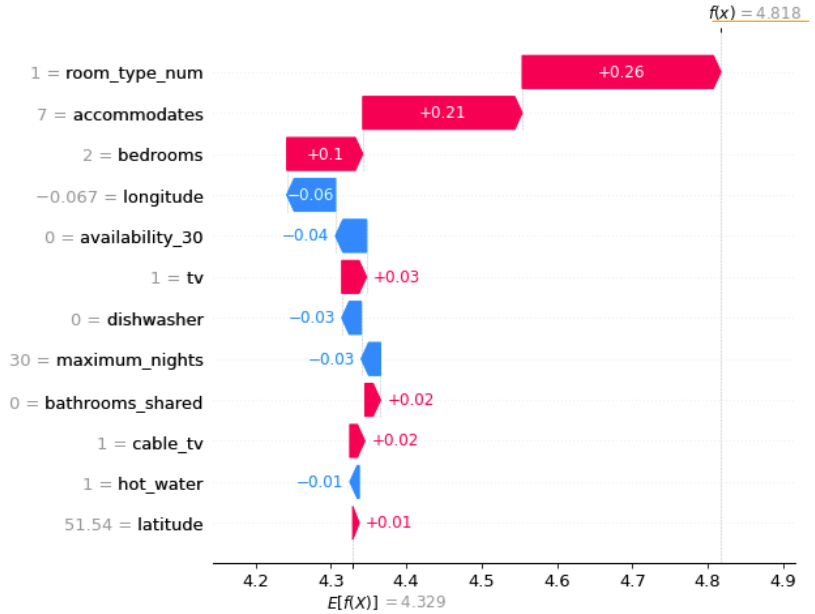
[1] The feature importance measure is Gain. It accounts for the improvement in accuracy brought by a feature to the branches it is on.

Modelling First Listing Price

Model interpretability

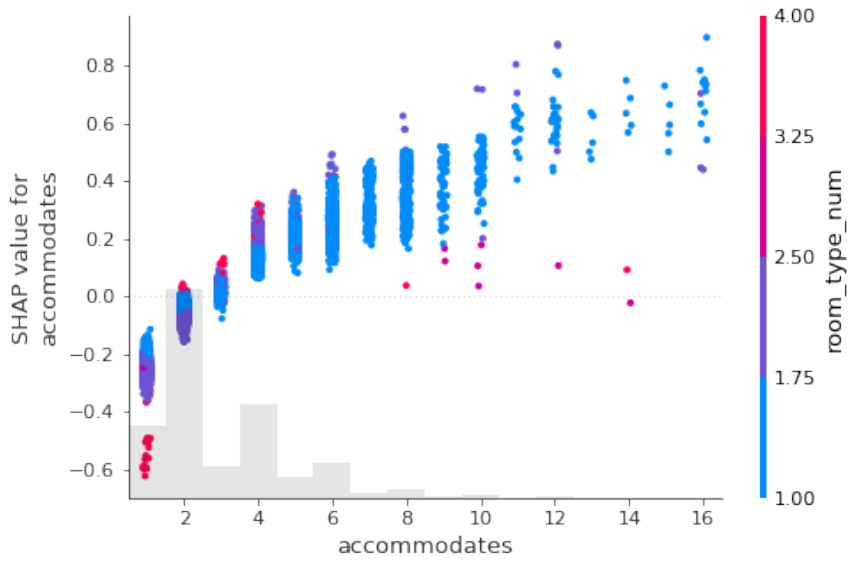
- How each variable or predictor affects the model's predictions? SHAP values break down a prediction to account the impact of each feature.

	price_log	price	predicted_price_log_01_GA	predicted_price_01_GA
2329	4.60517	100.0	4.817716	123.682289



Sum(SHAP values all variables) = pred_for_observation – pred_for_base_line_values

- SHAP values of all features sum up to explain why a given prediction was different from the baseline.



Room type num	Description
4	Hotel Room
3	Shared room
2	Private room
1	Entire home/apt

- There are non-lineal interaction effects between the two variables.
- Same value for *accommodates* have different impact on the model's outcome for different observations.

Modelling Overall Review value

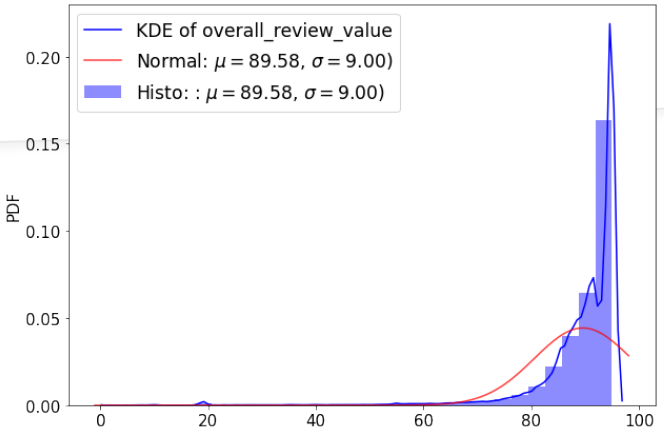
Modelling Overall Review value

- A new metric summarising the review scores was created as weighted average:

Weight %	Feature	Description
30	*score_value*	Emphasising the price and service that the guest gets from the lisitng.
20	*score_accuracy*	Reflects the accuracy of the host in describing the property & services
15	*score_communication*	Response time of host to queries
10	*score_rating*	Reflects the overall guest experience
10	*score_cleanliness*	Reflects the quality of hygiene
10	*score_location*	Reflects the secure/conectiviy/facilities in the area (not controled by host)
5	*score_checkin*	Reflects the satisfaction with the check-in flexibility

* All the variables were transformed to the same scale before computing the new metric.

overall_review_value



Data preparation and Feature selection

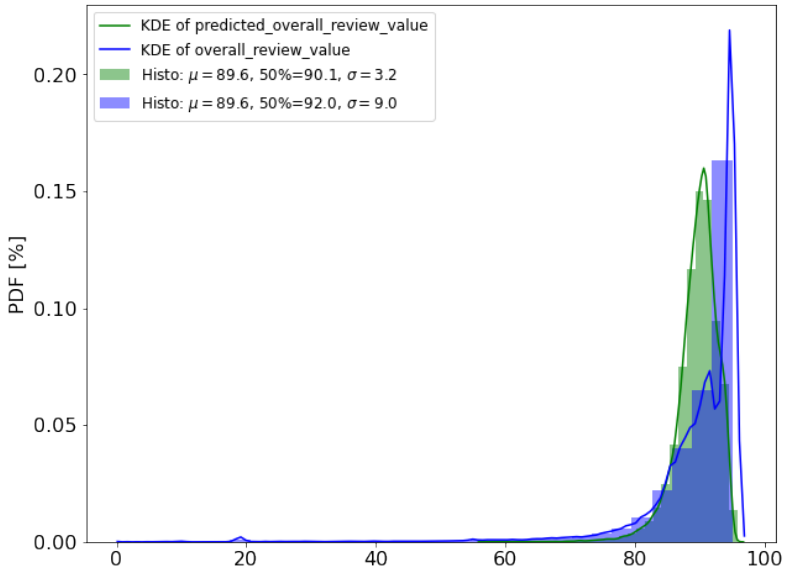
Steps	Description
68% of observations kept	Dropped observations without the 7 metrics for the calculation of *overall_review_value*.
Model requires to use	Predicted price, description, and *neighborhood_overview*.
Text feature processing	1) Cleaned by removing: html tags, stop words, punctuation, digits, etc. 2) Represented by its TF-IDF (Term Frequency-Inverse Document Frequency) 3) Transformed into two components with PCA (Principal Component Analysis)
New features	--> *description_pca_x, description_pca_y.* --> *neighborhood_overview_pca_x, neighborhood_overview_pca_y.*
Feature selection	1) 28 varaibles --> 10 Genetic Algorithm rounds Including variables that reflect features of the host, location, availability, time with reviews. 2) Time constraints --> features selected by importance & Spearman correlation criteria.

Modelling Overall Review value: Model tuning and evaluation

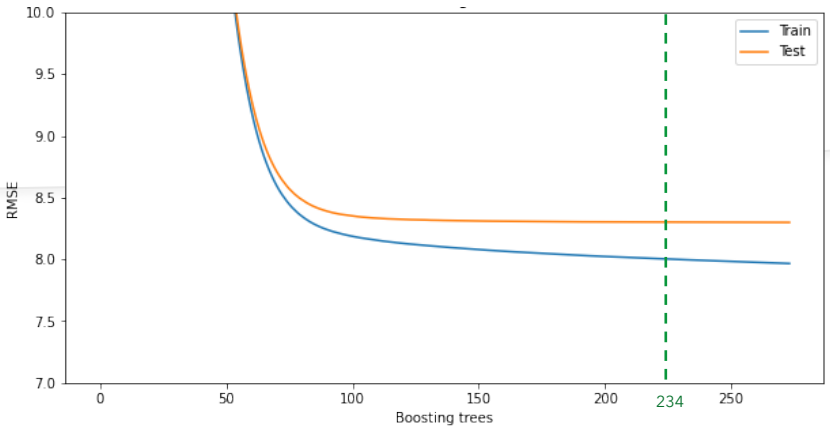
- The model was tuned by a non-exhaustive grid search and cross-validation technique, similar to how it was done for the pricing model.

Model	MRSE	Relative Uplift w.r.t base %
Base line (mean of new metric)	9.230	
Model 28 variables	8.600	6.83%
Model 19 variables (GA)	8.626	6.54%
Final model after tunning	8.575	7.10%

overall_review_value



Learning Curve (trade off bias-variance)

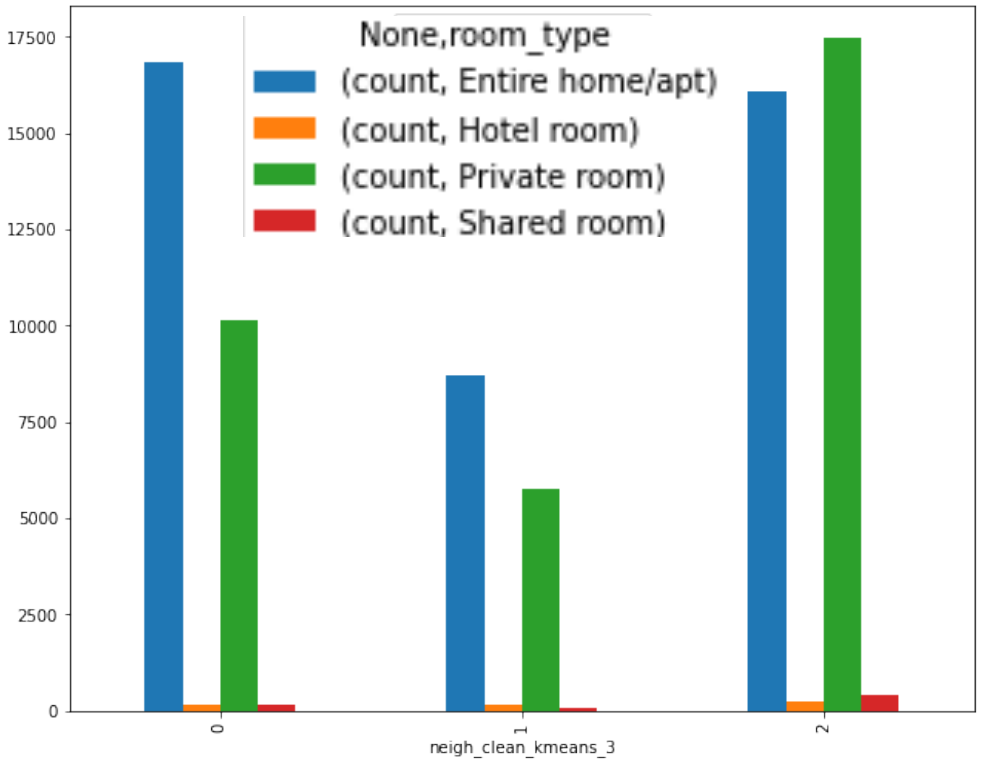


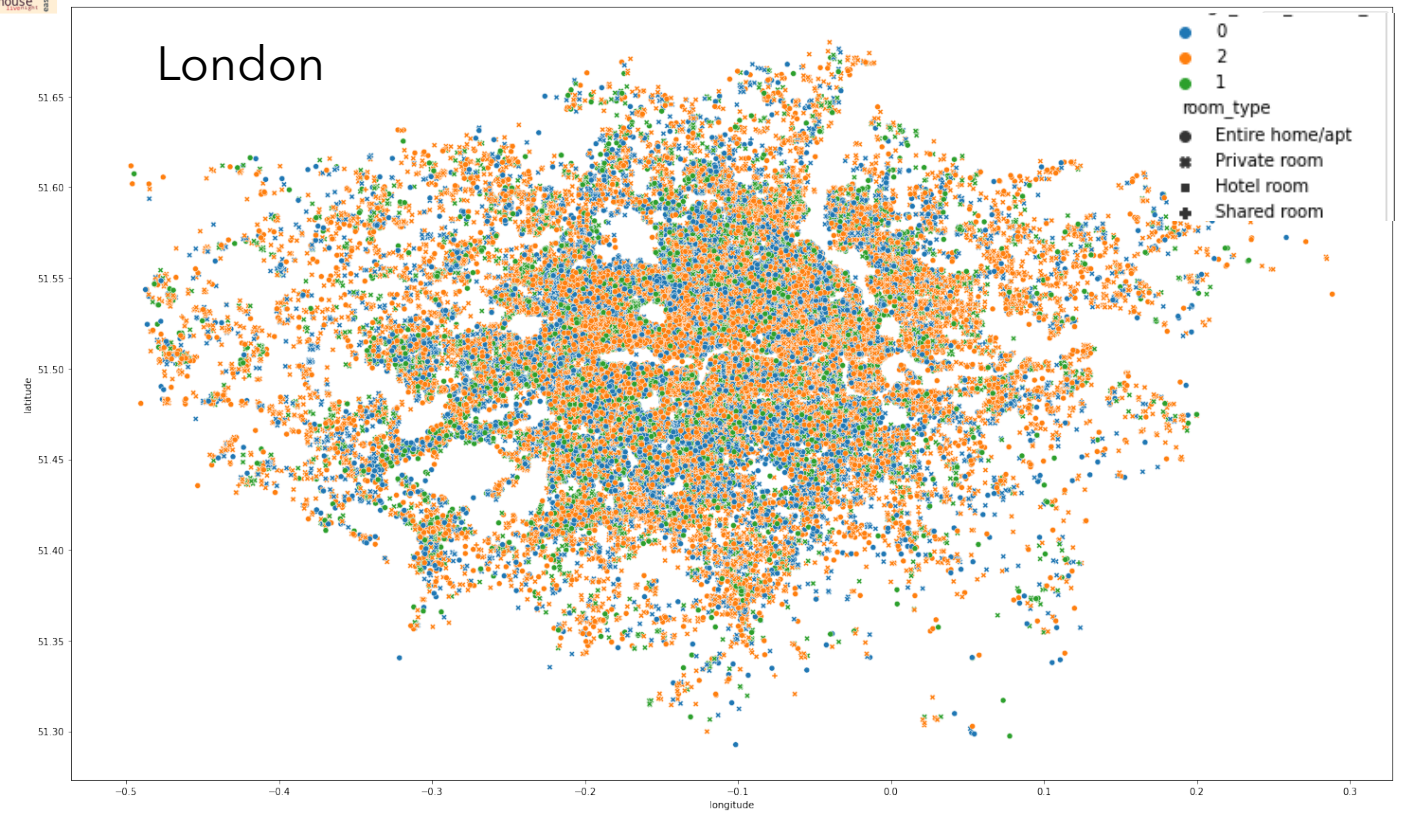
	Feature_importance%
host_is_superhost	22.950920
instant_bookable	13.276325
availability_90	7.056178
availability_60	5.736556
host_response_rate	5.422876
number_of_reviews	5.005566
host_acceptance_rate	4.782241
host_number_verifications	4.637665
reviews_per_month	4.319179
neighborhood_overview_pca_x	3.730767
bedrooms	3.445701
has_availability	3.296118
months_as_host	3.001118
months_since_first_reviews	2.891196
latitude	2.623170
predicted_price_log_01_GA	2.412449
desccription_pca_y	2.084501
desccription_pca_x	1.859632
neighborhood_overview_pca_y	1.467847

Analysis on Neighborhood description

- By using the new metric *Overall Review value f*rom our model, we can take those hosts with best score per area and analyse the most relevant words used to describe their neighborhood.

- It will become a tool to suggest how to write a successful description of the host's neighborhood and property for their fist listing.





Future Improvements

Modelling First Listing Price

- Considering the seasonality of pricing. Acquire data set with reviews and price over the years, since this could approximate demand if there is no booking data.
- This approach will help to have a more precise price for a new listing that wishes to start on a certain time of the year and location.

Modelling First Listing Price

- Improve the cleaning of the text fields of *description and neighborhood_overview* to have a better idea of what makes the listing more attractive given the *room_type* and *location*.
- Get the reviews for the properties and make a sentiment analysis to suggest to the host what makes their guests more comfortable, so they can provide a better service and therefore get better reviews.

Thanks!

Questions