Nightly Price Prediction Model for Airbnb

REPORT PREPARED BY:
GROUP 53

Table of Contents

Appendices

1 Project Background

1.1 Introduction:

Airbnb operates an online marketplace for renting and leasing. To recommend a lodge with an appropriate price according to the conditions restricted by its users, the python is used to build different models to predict prices, and the report is written to justify the outputs.

This report does exploratory data analysis by analyzing its scatter plot, heatmap and boxplot. Besides, a new feature is created about the location for predicting the price better.

In the model building section, after having tried different models including Linear regression model (OLS, Ridge regression, Lasso, Elastic Net), Decision Tree, Random Forest, XGboosting and Model Stacking in Kaggle competition, Model Stack ranks the highest on the Public Leaderboard and becomes final model.

2 Data Processing

2.1 Data cleaning

Data cleaning include all process that concert raw data into useful information. In this report, missing value imputation and outlier removing has been adopted.

2.2 Missing value imputation

For both training and test dataset, the following data categories have missing values and the number of missing values has been listed:

	training dataset	training dataset
review_scores_rating	491	484
review_scores_accuracy	494	486
review_scores_checkin	493	485
review_scores_communication	491	485
review_scores_location	493	486
review_scores_value	493	487
review_scores_cleanliness	491	485
reviews_per_month	430	414
host_response_time	935	976
host_response_rate	935	976
host_acceptance_rate	713	718
security_deposit	707	670
cleaning_fee	523	501
bedrooms	2	484
beds	12	486

Since the amount of the missing value are large (amount to 976 the highest), simply drop the missing columns or rows might be statistically inaccurate and lead to inconsistency (Aguate, Crossa and Balzarini, 2019). Therefore, imputation method could be applied in this case to assist data analysing (Little and Rubin, 2002).

In this case, different imputation method will be applied to different type of data.

Firstly, since data in review_scores_accuracy, review_scores_checkin, review_scores_communication, review_scores_location, review_scores_value, review_scores_cleanliness, and reviews_per_month are all numerical and the nominal average numbers possess the highest level of ability to represent populations, the mean imputation method will be applied in this case (Little and Rubin, 2002). Hence, the missing value in these columns will be filled by the means of them.

review_scores_rating	float64	Numerical
review_scores_accuracy	float64	Numerical
review_scores_checkin	float64	Numerical
review_scores_communication	float64	Numerical
review_scores_location	float64	Numerical
review_scores_value	float64	Numerical
review_scores_cleanliness	float64	Numerical
reviews_per_month	float64	Numerical

Secondly, although host_response_time, host_response_rate, and host_acceptance_rate has text and numerical records, these 3 columns could be considered as categorical data for simplicity. For instance, for host_response_time, the record that has texts all indicates the same idea, which is “responsed”. Hence, it is reasonable to assume that the blank cell could indicates there is no response. Hence, the missing value in host_response_time will be filled with the text of “no response”.
Same logic will be applied to host_response_rate and host_acceptance_rate.
Also, data in both host_response_rate and host_acceptance_rate columns are numerical data. However, it could be perceived that the missing value in host_response_rate and host_acceptance_rate are highly overlapped with that of host_response_time. Hence, same logic will be applied to host_response_rate and host_acceptance_rate. For host_response_rate, cells with a number would indicates there is a response, otherwise there is no response. The missing value will be filled by “0”. For host_acceptance_rate, cells with a number would indicates the host has accept the order, otherwise there is no acceptance. The missing value will be filled by “0”.

host_response_time	object	Catagorical
host_response_rate	float64	Numerical
host_acceptance_rate	float64	Numerical

Moreover, data in both security_deposit and cleaning_fee are all integers and with large range. The highest value in security_deposit is 7021 and the lowest is 0. The highest value in cleaning_fee is 500 and the lowest is 0. Hence, using mean imputation might cancel out the effect of outlier and generate inaccurate result, while the median imputation could generate a better performance (Edgar and Caroline, 2004).

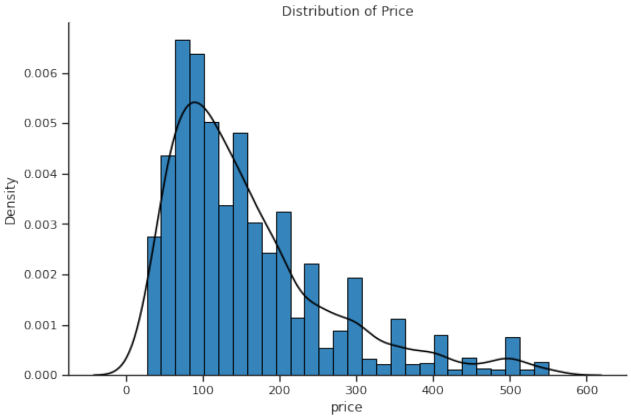
security_deposit	float64	Numerical
cleaning_fee	float64	Numerical

Lastly, bedrooms and beds are numerical data, but mean or median imputation might not be applicable since the average of 2.5 beds is not reasonable. Also, according to common sense, it is rare to have 3 beds but 1 bedroom. Hence, for simplicity, in this case, the number of bedrooms and beds will be assumed to be matched. Therefore, the missing value of bedrooms will be the same value as number of bathrooms, vice versa.

bedrooms	float64	Numerical
beds	float64	Numerical

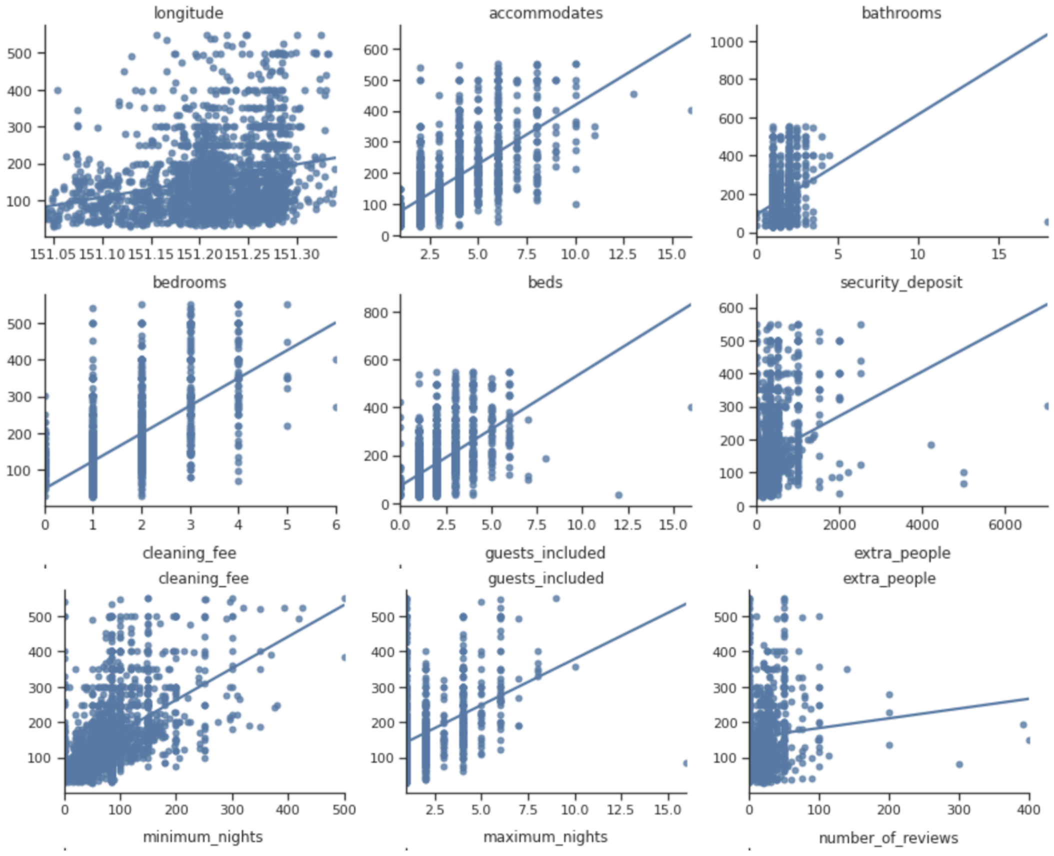
3 Data Visualisation

3.1 Exploratory Data Analysis: Histogram



From the histogram above, the distribution of price is positively skewed. Most of the prices are concentrated at [30,220] interval. Moreover, the maximum price can reach up to 550 dollars per day, which is roughly 18 times as large as the minimum price.

3.2 Exploratory Data Analysis: Linearity graph

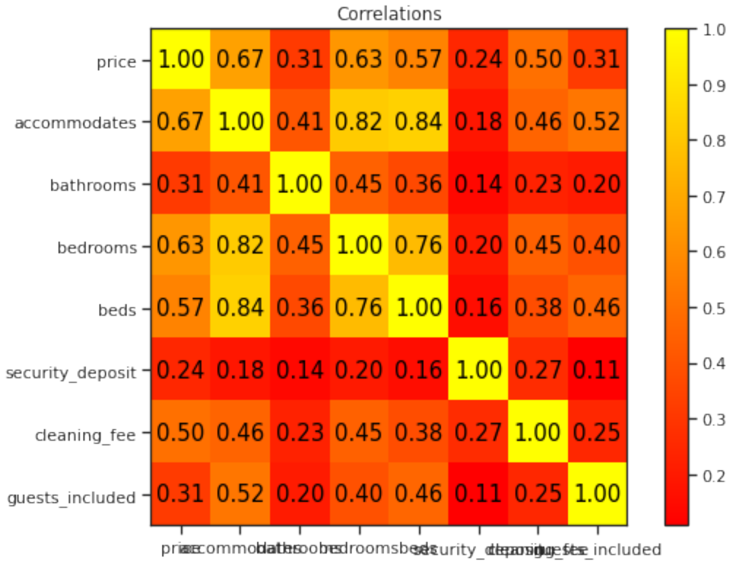


At the next stage, we have plotted a series of graphs that can clearly show the linearity between the price and each of the predictors. In this process, we found that the number of accommodates, bathrooms, bedrooms, beds, guests included, the amount of security deposit, as well as the amount of cleaning fee have a relatively strong linear relationship with the price, which means that they are very likely to be the key predictors determining the price of residence.

Category of data	Boundary value
host_listings_count	200
accommodates	12
bathrooms	10
beds	10
security_deposit	4000
guests_included	12
extra_people	200
minimum_nights	150
number_of_reviews	400
reviews_per_month	10

Besides, we found that the number of accommodates, bathrooms, bedrooms, beds, guests included, the amount of security deposit, as well as the amount of cleaning fee have a relatively strong linear relationship with the price, which means that they are very likely to be the key predictors determining the price of the residence.

3.3 Exploratory Data Analysis: Correlation heat map

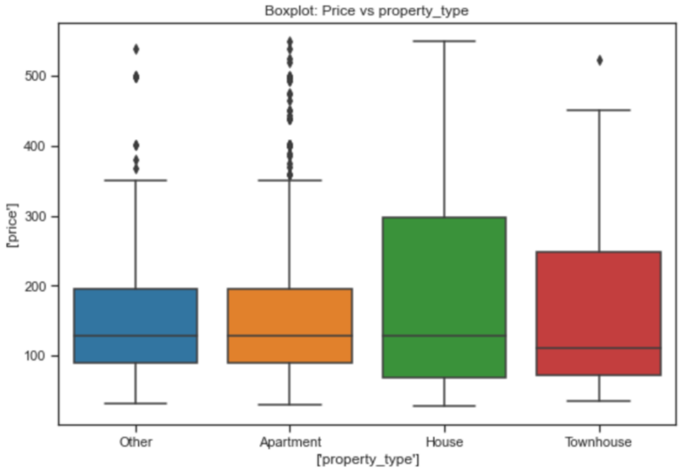


From the correlation heat map above, it is noticeable that the correlation between the price and the number of accommodates is the largest amongst the group, which is 0.67. Moreover, the number of bedrooms and the number of beds also have a relatively strong correlation with the price, which are 0.63 and 0.57 respectively. Besides, the correlation values of the number of bathrooms and the number of guests included are the same, which are both 0.31. Finally, the amount of security deposit has the least correlation with the price amongst the 8 predictors, which is only 0.24.

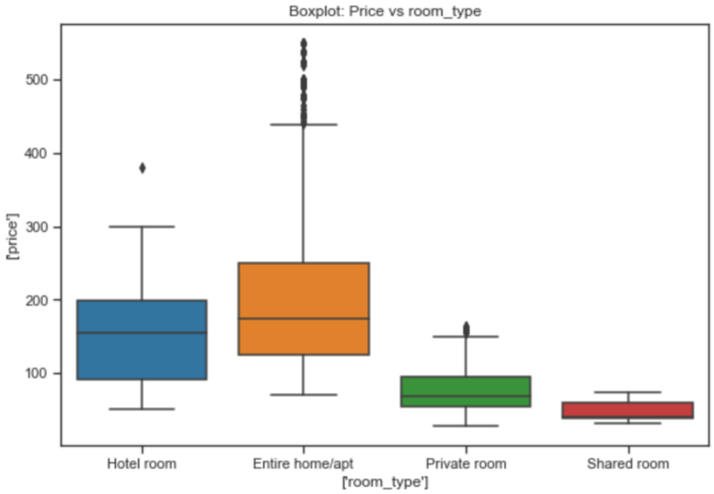
3.4 Exploratory Data Analysis: Boxplot

The boxplots illustrate the relationship between the price and a variable with a comparable small number of distinct values.

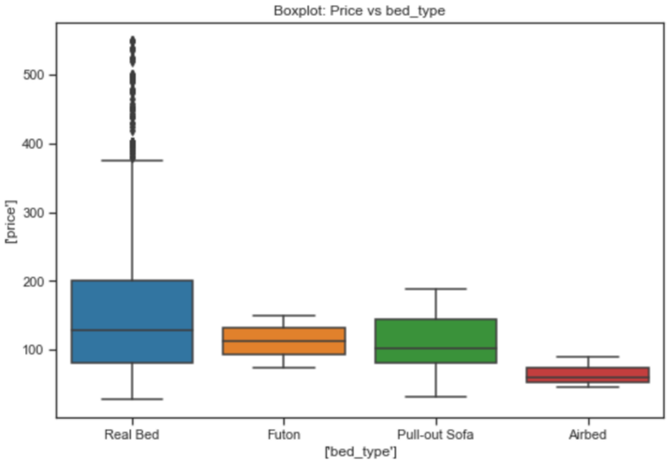
These are a few boxplots illustrating clear relationships.



For the property type, all property types show a similar median, but the variance of house's price is the largest, followed by Townhouse. Besides, apartment and other property type have a similar price range.



For the room type, the Entire home indicates the highest price, followed by hotel room, private room and shared room respectively.



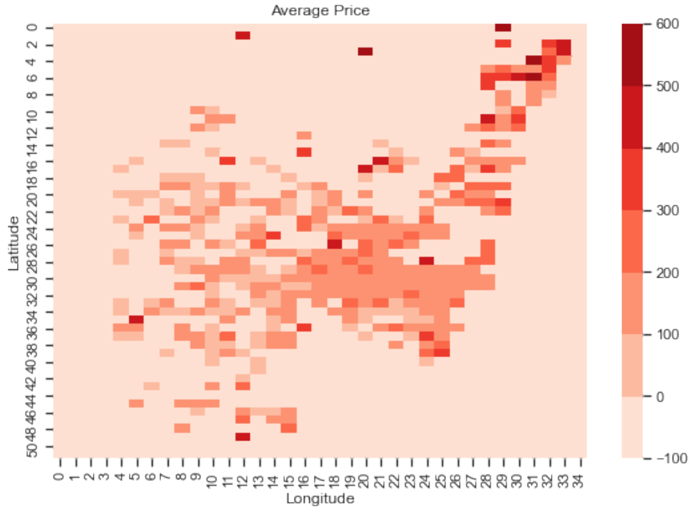
For the bed type, the real bed has the highest variance and many outliers, other bed type demonstrates the opposite information.

In addition, the correlations between numerical features and prices are the same with that shown in scatter plots above.

3.5 Exploratory Data Analysis: Location

The region with all the listings is divided into equal small areas where average price is calculated accordingly to better observe the pricing patterns. The plot below visualizes the geographical distribution of listings along with the average price. High prices are presented as dark red color and the color becomes lighter as prices decline. The patterns of pricing by region can be observed in the plot.

High prices appear to be concentrated in the northern and north-eastern part of the city and the price decreases going south. The listings in the west are relatively cheaper than those in the other regions.



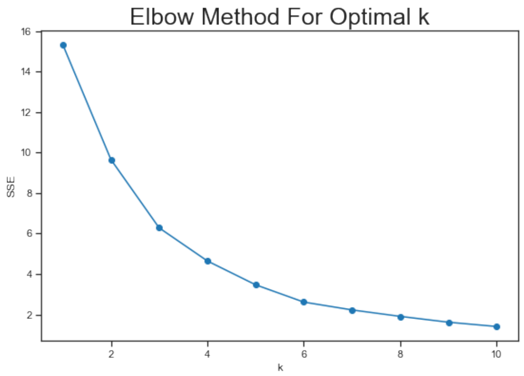
3.6 Dummy variable creation

We transform a series of categorical variables into dummy variables for the purpose of model building. For instance, there are 4 property types in our dataset, which are apartment, house, townhouse and others. We set apartment as the baseline, and then set the other 3 property types as dummy variables, which will only display 0 or 1 in the record.

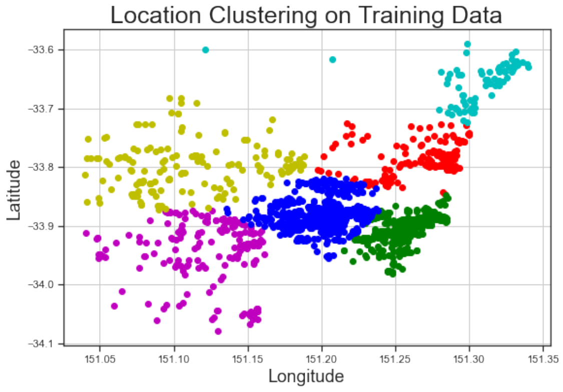
4 Feature Engineering

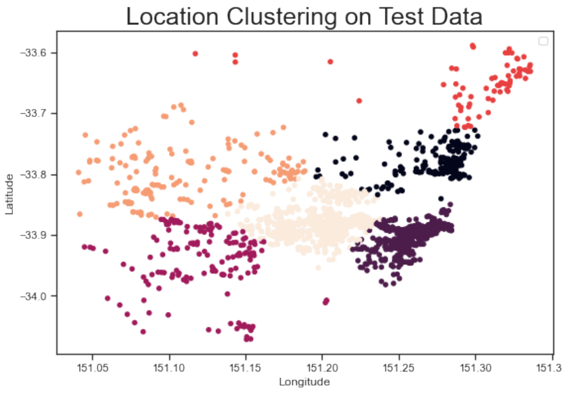
4.1 Feature engineering: Location

Given that the average price may vary by region, the features of longitude and latitude are processed using K-means clustering algorithm to generate several regions. K-means is a type of unsupervised learning, clustering unlabelled data into k clusters (Wikipedia, 2021). This algorithm groups the data by minimizing the sum of squared distances between the data points and their respective closest centroid. Since the number of clusters will affect the clustering result, the elbow method is applied to identify the optimal k value (Yuan & Yang, 2019). For each k varying from 1 to 10, the sum of squared distances between the sample points in each cluster and the centroid of the cluster (SSE) is calculated. Below is the plot of SSE for each k. The elbow in the plot is at k = 6 indicating the optimal number of clusters, since adding another cluster doesn’t decrease SSE much.



Through the package of K-means, a new categorical variable indicating the cluster of each data point is named "location_label" and merged into the original table for both training set and test set for model building. The clustering result is visualized by plotting the data colored by different labels.





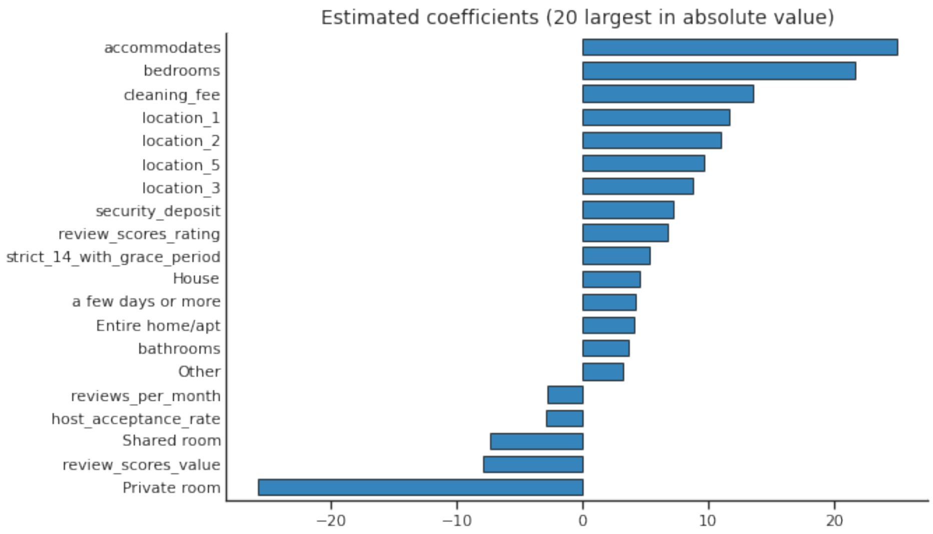
5 Model Building

5.1 Lasso

According to the testing result generated by 30% of the testing data, lasso has the lowest RMSE amongst the four linear models (ols, ridge, lasso, elastic net). Therefore, lasso is selected to be analyzed in this methodology section.

$$\underset{\beta_0,\,\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

Lasso is a regularisation method for the linear regression model. It estimates the coefficients of predictors by minimizing the value of the function shown above. The model complexity penalty term of lasso is the product of tuning parameter lambda and the sum of the estimated coefficients in absolute value. It tends to shrink some of the coefficients towards zero when the lambda is sufficiently large, which can perform a variable selection process and make the model more interpretable.



From the graph above, the magnitude of the estimated coefficient for private room is the largest amongst the group in our model. Then the size of accommodates value ranks the second, followed by bedrooms and cleaning_fee.

```
Estimated coefficients of several key predictors in Lasso
accommodates: 24.92
bedrooms: 21.65
cleaning fee: 13.50
private room: -25.80
```

Here are some estimated coefficients of several predictors in our model. For instance, other things equal, the price of residence will increase by 24.92 dollars if one extra person is allowed to use. Moreover, we can also acknowledge that the price of residence will go up by 21.65 dollars if one more bedroom is required, holding other factors unchanged. Besides, one dollar increase in cleaning fee required is associated with a 13.50 dollars increase in residence's price if other things are equal. Furthermore, the estimated coefficient of private room indicates that the price of residence will decrease by 25.80 dollars if one additional private room is provided, keeping other factors unchanged.

5.2 Regression Tree

Regression tree method is also considered to make predictions. Under this method, the model is obtained by recursively splitting the data space into non-overlapping regions and fitting a simple prediction model in each region.

The splits are selected by minimising the prediction error measured by the residual sum of squares between the observed and predicted values. The function can be written as:

$$\min_{R_1,\ldots,R_M} \sum_{m=1}^{M} \sum_{i:\,\boldsymbol{x}_i \in R_m} (y_i - \widehat{c}_m)^2$$

R_m are the selected regions. c_m is the average of the responses corresponding to the region R_m, which is also the predicted value for each observation in R_m.

Since it is computationally infeasible to solve this problem, recursive binary splitting is applied as an alternative method. The function can be defined as:

$$\min_{j,s} \left\{ \sum_{i:\,\boldsymbol{x}_i \in R_1(j,s)} (y_i - \widehat{c}_1)^2 \; + \sum_{i:\,\boldsymbol{x}_i \in R_2(j,s)} (y_i - \widehat{c}_2)^2 \right\}$$

Having found the best split, the data are divided into the two resulting regions and the splitting process is repeated on each of them until it reaches the minimum node size we have set.

In python we use the DecisionTreeRegressor function to build the regression tree. Given that a large tree will overfit the data leading to poor model generalisation, the tuning parameters including 'max_depth' and 'min_samples_leaf' are selected to control the complexity of the tree structure. The optimal values for these tuning parameters can be identified through the cross-validation under GridSearchCV function.

As a result, the rmse of the regression tree is 69.29 across the validation dataset. The model can be displayed as a tree diagram (Appendix).

The first line of each node represents the splitting condition. "Samples" indicates the number of observations contained in the node. "Value" shows the average of the responses in each node. The predicted value for an observation is the value in the terminal node. For example, an airbnb listing which only permits one person to live in (accommodates <= 3.5, accommodates <= 1.5), is not the entire home or apartment (Entire home/apt <= 0.5) and has review scores for location less than 9.36 (review_scores_location <= 9.36) will charge 50.053 dollars, as shown on the leftmost side of the tree diagram.

The key advantage of the regression tree is that it is simple and highly interpretable. However, the main issues of this method are that the regression tree is inherently unstable and has high variance due to the hierarchical nature. In addition, trees will lead to non-smooth prediction functions and decision boundaries, which can reduce the performance of the model.

5.3 Model Stacking

Model stacking is an aggregated method that using the predicted results from numerous machine learning algorithms as the 3 inputs to predict the final outcome. Each of the prediction, which predicted from other models, will be considered as a feature to the model stacking and improve the predictive accuracy.

Model stacking will generally split models into level1 and level 2. Different levels of model are used to predict different set of data.

In this assignment, the fundamental model that has been adopted as the input in the model stacking are ols, lasso, enet, ridge, tree_search, xbst, and CV_rfr.

This indicates that the level 1 models are:

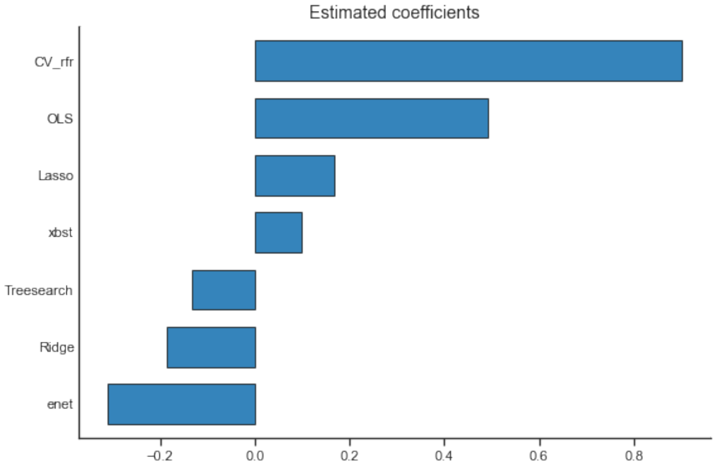
1	OLS
2	Lasso
3	Elastic Net Regression
4	Ridge
5	Decision tree
6	XGBoost
7	Random Forest Regressor

The meta-regressor is the metamodel that we incorporate all the fundamental model in. In this assignment, the meta-regressor, which is the level 2 model, is “Linear Regression Model”.

Afterward, the second layer model has been developed in the form of Linear Regression and the prediction of housing price based on test predictors has been captured as following:

	Id	Predicted price on test predictors
0	0	78.528369
1	1	193.564413
2	2	160.799695
3	3	94.088395
4	4	114.480034
...	...	...

Eventually, to visualize the second layer Linear Regression model, its coefficient can be illustrated by the following figure.



It could be perceived that there are 7 coefficient (consistent to the 7 fundamental models). The coefficient of Random Forest Regressor model is the highest (approximately 0.9). This implies that Random Forest Regressor might has the greatest positive effect on the final pricing prediction when using model stacking. Moreover, XGBoost has the lowest absolute value of coefficient (approximately 0.1). This indicates that XGBoost model might contribute the least when predicting the housing price. Furthermore, Elastic Net Regression has the lowest coefficient number (approximately -0.3). This indicates that there is a negative relationship between the predicted housing price from model stacking and the Elastic Net Regression feature.

5.4 Additional models

Random forest regression is an extension from the decision tree model by constructing several decision trees and generating the mean as the prediction of the trees. This model would be considered more powerful and accurate with its account for non-linear relationships, however, there is the tendency to overfit.

XGBoost is an algorithm that widely applied in machine learning. It can assist the generating of the best parameters based on the tuning parameters and the building of the model. In this assignment, XGBoost has applied to determine the importance of each feature and to calculate the RMSE of the created model toward the training set.

6 Kaggle Validation and Comparison

6.1 Validation and comparison

After participating in Kaggle competition, the validation scores of these five models including Lasso, Decision Tree, RandomForestRegressor, XGboosting and Model Stacking are 69.11719, 69.28673, 65.74750, 70.20734 and 65.61789 respectively.

Lasso is a kind of linear regression with regularization, which prevents overfitting compared with simple linear regression but still cannot express non-linearity ("Algorithm Selection", 2021). To express non-linear regression relationships, the remaining four models are more widely used.

For the regression tree, it creates branches like if-else conditions (Mwiti, 2021). Like the example listed above in the section of regression tree, it can explicitly express the two different outputs with one condition changing. However, Lasso may not express this reversal of correlation directly. The shortcoming of regression tree is that it can create limitless branches, resulting in overfitting.

The Random Forest Regression is an ensemble technique of decision trees, averaging the individual output of each tree to get the final predictions. It can improve prediction accuracy and prevent overfitting ("Algorithm Selection", 2021).

As for XGboosting, its full name is Extreme Gradient Boosting Algorithm, and it is also an ensemble algorithm. XGboosting can be called weak learners because it considers the error in the previous step, then makes corrections (Dwivedi, 2021). A major difference from Random Forest is that its output is a weighted average (Dwivedi, 2021).

The model stacking model behaves the best among the five models in Kaggle competition. The reason is just that the meta-regressor predicts better than a single model (Hansen, 2020).

7 Conclusion

In conclusion, EDA generally indicates the relationships between the features in the listing and prices. Then, through the description, interpretation and comparable of models, the benefits and shortcomings of the statistical learning models have been stated clearly. After that, Kaggle competition marks decide that Model Stacking is the most accurate one for prediction. However, there are still some pitfalls for the analysis, for example, another method to quantify location has been considered, which is to set a center point and calculate the distance from the center point, but it is unsure whether a point is appropriate.

References

Aguate, F., Crossa, J. and Balzarini, M. (2019). Effect of Missing Values on Variance Component Estimates in Multienvironment Trials. *Crop Science*, 59(2), pp.508-517. Retrieved from https://acsess-onlinelibrary-wiley-com.ezproxy.library.sydney.edu.au/doi/pdfdirect/10.2135/cropsci2018.03.0209

Algorithm Selection in Machine Learning - Data Science Primer. EliteDataScience. (2021). Retrieved 2021, from https://elitedatascience.com/algorithm-selection.

Banks, D., McMorris, F., Arabie, P. and Gaul, W. (2004). *Classification, Clustering, and Data Mining Applications |SpringerLink*. Link.springer.com. Retrieved from https://link.springer.com/content/pdf/10.1007%2F978-3-642-17103-1.pdf

Dwivedi, R. (2021). *Random Forest Vs XGBoost Tree Based Algorithms*. Analytics India Magazine. Retrieved 2 June 2021, from https://analyticsindiamag.com/random-forest-vs-xgboost-comparing-tree-based-algorithms-with-codes/.

Hansen, C. (2020). *Stack machine learning models: Get better results*. IBM Developer. Retrieved from https://developer.ibm.com/technologies/artificial-intelligence/articles/stack-machine-learning-models-get-better-results/.

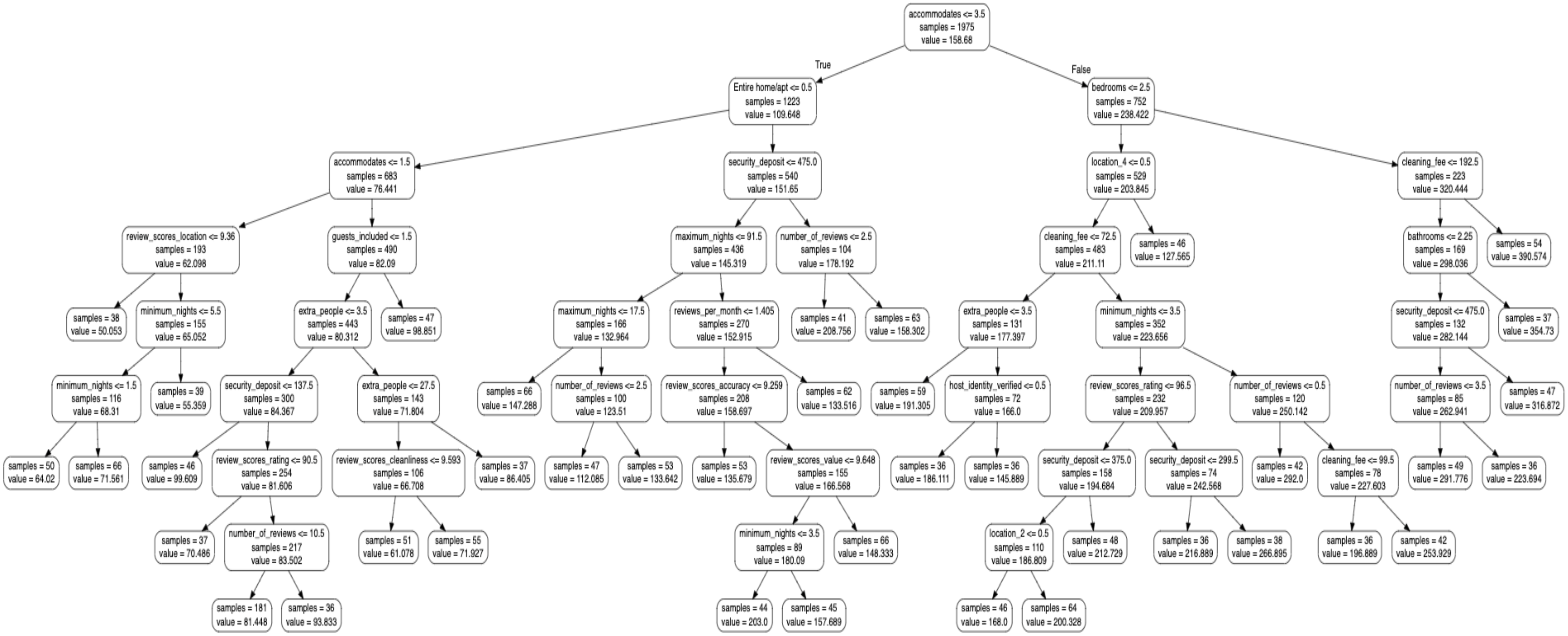
Little, R and Rubin, D. (2002). *Statistical Analysis with Missing Data. Hoboken*, New Jersey: John Wiley & Sons. Retrieved from https://ebookcentral-proquest-com.ezproxy.library.sydney.edu.au/lib/usyd/reader.action?docID=1775204

Mwiti, D. (2021). *Random Forest Regression: When Does It Fail and Why? - neptune.ai*. neptune.ai. Retrieved from https://neptune.ai/blog/random-forest-regression-when-does-it-fail-and-why.

Smith, M. and Martinez, T. (2011). *Improving Classification Accuracy by Identifying and Removing Instances that Should Be Misclassified*. Retrieved from https://ieeexplore-ieee-org.ezproxy.library.sydney.edu.au/stamp/stamp.jsp?tp=&arnumber=6033571

Yuan, C., & Yang, H. (2019). Research on K-Value Selection Method of K-Means Clustering Algorithm. *J*, *2*(2), 226-235. Wikipedia. (2021). k-means clustering. Retrieved from https://en.wikipedia.org/wiki/K-means_clustering

Appendix


accommodates <= 3.5 samples = 1975 value = 158.68
True
False
Entire home/apt <= 0.5 samples = 1223 value = 109.648
bedrooms <= 2.5 samples = 752 value = 238.422
accommodates <= 1.5 samples = 683 value = 76.441
security_deposit <= 475.0 samples = 540 value = 151.65
location_4 <= 0.5 samples = 529 value = 203.845
cleaning_fee <= 192.5 samples = 223 value = 320.444
review_scores_location <= 9.36 samples = 193 value = 62.098
guests_included <= 1.5 samples = 490 value = 82.09
maximum_nights <= 91.5 samples = 436 value = 145.319
number_of_reviews <= 2.5 samples = 104 value = 178.192
cleaning_fee <= 72.5 samples = 483 value = 211.11
samples = 46 value = 127.565
bathrooms <= 2.25 samples = 169 value = 298.036
samples = 54 value = 390.574
samples = 38 value = 50.053
minimum_nights <= 5.5 samples = 155 value = 65.052
extra_people <= 3.5 samples = 443 value = 80.312
samples = 47 value = 98.851
maximum_nights <= 17.5 samples = 166 value = 132.964
reviews_per_month <= 1.405 samples = 270 value = 152.915
samples = 41 value = 208.756
samples = 63 value = 158.302
extra_people <= 3.5 samples = 131 value = 177.397
minimum_nights <= 3.5 samples = 352 value = 223.656
security_deposit <= 475.0 samples = 132 value = 282.144
samples = 37 value = 354.73
minimum_nights <= 1.5 samples = 116 value = 68.31
samples = 39 value = 55.359
security_deposit <= 137.5 samples = 300 value = 84.367
extra_people <= 27.5 samples = 143 value = 71.804
samples = 66 value = 147.288
number_of_reviews <= 2.5 samples = 100 value = 123.51
review_scores_accuracy <= 9.259 samples = 208 value = 158.697
samples = 62 value = 133.516
samples = 59 value = 191.305
host_identity_verified <= 0.5 samples = 72 value = 166.0
review_scores_rating <= 96.5 samples = 232 value = 209.957
number_of_reviews <= 0.5 samples = 120 value = 250.142
number_of_reviews <= 3.5 samples = 85 value = 262.941
samples = 47 value = 316.872
samples = 50 value = 64.02
samples = 66 value = 71.561
samples = 46 value = 99.609
review_scores_rating <= 90.5 samples = 254 value = 81.606
review_scores_cleanliness <= 9.593 samples = 106 value = 66.708
samples = 37 value = 86.405
samples = 47 value = 112.085
samples = 53 value = 133.642
samples = 53 value = 135.679
review_scores_value <= 9.648 samples = 155 value = 166.568
samples = 36 value = 186.111
samples = 36 value = 145.889
security_deposit <= 375.0 samples = 158 value = 194.684
security_deposit <= 299.5 samples = 74 value = 242.568
samples = 42 value = 292.0
cleaning_fee <= 99.5 samples = 78 value = 227.603
samples = 49 value = 291.776
samples = 36 value = 223.694
samples = 37 value = 70.486
number_of_reviews <= 10.5 samples = 217 value = 83.502
samples = 51 value = 61.078
samples = 55 value = 71.927
minimum_nights <= 3.5 samples = 89 value = 180.09
samples = 66 value = 148.333
location_2 <= 0.5 samples = 110 value = 186.809
samples = 48 value = 212.729
samples = 36 value = 216.889
samples = 38 value = 266.895
samples = 36 value = 196.889
samples = 42 value = 253.929
samples = 181 value = 81.448
samples = 36 value = 93.833
samples = 44 value = 203.0
samples = 45 value = 157.689
samples = 46 value = 168.0
samples = 64 value = 200.328