Programming with R

Group Project

Analysis of Airbnb Listings in Hawaii (Dec, 2019)

Participants:

Name	SNR	ANR
Adrian Konya	2037123	u242159
Fernando Catala	2048042	u270800
Georgios Satyridis	2046944	u836891
Konstantinos Soiledis	2037001	u226393
Miftahul Ridwan	2040778	u989303

Introduction

This research paper puts into practice acquired skills in R by analyzing publicly available data of Airbnb listings in Hawaii in 2019. We considered that the dataset provides us with interesting and easily relatable variables to experiment with, at a reasonable dimension (23745 x 106). Having explored it, we thought of key research questions about topics that drew our attention.

We began with a data preprocessing in order to choose our preferred variables to work with, also performing data cleaning and treating missing values. Then, we did some EDA by plotting several interactions among variables, which we considered relevant towards our questions.

Regarding machine learning analysis, since we were required to use knn and logistic regression methods, we thought that adding random forest would contribute to feature selection.

Research Questions

- What are the determinants of Airbnb price in Hawaii?
- Can we predict which Airbnb host is a a *superhost*? (e.g. in response time, response rate, review score)
  - Are the predictions improved with embedded methods?

Data Preprocessing

The dataset used in this analysis, 'Hawaii Airbnb', was obtained from Kaggle. Hawaii Airbnb consists of total 23.745 observations and 106 variables.

The pre-processing part begins with feature selection and each further step within the data wrangling process was stored in new variables depending on its purpose. The cleaning process was completed in the following order: loading the data, cleaning data and lastly imputing missing data. When loading the data, all empty spaces were set to NA. In addition, a custom function was made during data cleaning in order to remove the unnecessary elements such as; symbols, letters Also. column names were converted from lowercase to uppercase and underscores were substituted by dots.

From initial 106 variables meaningless columns were removed and appropriate columns were selected and converted to suitable data types (integer, numeric, factor). Lastly, during cleaning, columns in range of 3 - 1476 NA's were deleted as imputation would not be useful on them.

For columns which required imputation due to couple of thousand NA's, Mice package was used as it's the most common and powerful packages for imputation, which takes care of uncertainty in missing values by assuming Missing at Random.

Exploratory Data Analysis

We thought of 1) which variables would be most relevant to answer our questions, 2) most relevant interactions between them, and 3) how to visualize these with the appropriate chart. A summary of most relevant observations follows here below:

- Maui and Oahu concentrate ~2/3 of listings *(Fig. 1)*
- Predominant property types are Condominium, Apartment, and House *(Fig. 2)*
- Shared rooms require a median of 1 minimum nights, while Entire home/apt which requires 3 nights *(Fig. 8)*
- Median guests accommodated is quite large and even at 4 in all markets *(Fig. 9)*
- Response rates and review score ratings are high, over 90% *(Figs. 11 & 12)*
- Oahu and The Big Island appear more flexible regarding cancellation policy *(Fig. 13)*
- Response time is quick at 70% within an hour *(Fig. 14)*
- Superhosts achieve slightly higher response rates and review score ratings than other users, and also respond slightly quicker, but anyhow regular users' figures are excellent *(Figs. 14, 15 & 16)*
- Maui and Kauai are pricier than Oahu and The Big Island *(Fig. 17)*
- Median prices among property types vary between ~$40 and ~$400 *(Fig. 18)*
- Price increases mostly linearly with the number of guests accommodated *(Fig. 21)*

Model 1: Random Forest

Motive:

We developed an embedded method with random forests. The idea is to exploit the ability of the algorithm to incrementally divide the dataset to subsets, based on features that provide the most information gain (Dubey, 2018).

Method:

Ranger package was used in a function with 3 inputs. The first input is the dataset, the second input is the feature which will be used for classification. Finally, there is the "eval_metric" input that takes two values, 1 and 2. 1 is Accuracy and 2 is Kappa that handles imbalanced classes ("Cohen's kappa," 2019).

At first, the function splits the dataset, and then creates a matrix to store the results. Then a copy of the train dataset is used. Later, in the while loop, we fit the model and store evaluation metric. After that, we incrementally subtract the feature with the least variable importance. The hyperparameters used are defaults, but we set "mtry" to 1. With a value of 1 we measure each variable individually. The last step of the function is to return the variables with the best evaluation metric, in a "paste" format, ready to be run by the later algorithms along with the matrix.

Model 2: Logistic Regression

We fitted a logistic regression model on the original dataset to predict whether a host on Airbnb is a superhost and we compared the results with the ones obtained from the logistic regression model using the features extracted from our model of random forests.

Based on our results, (Appendix B, Table 1) the first model achieved an accuracy of 73.93% with 81.59% recall and 77.5% precision. The second model produced similar results, specifically an accuracy of 73.53%, 81.7% recall and 77.15% precision. It is obvious that the difference in the performance of both models is minimal and that they predict, with adequate accuracy, if the host is indeed a superhost, however it has to be stated that they tend to misclassify in a few cases.

It is worth noting the most important variables in our prediction *(Fig. 22 & Fig. 23)*. In both models, the response rate of the host, the score of the apartment's cleanliness and the strict cancellation policy, play a pivotal role in predicting the dependent variable. We could say that all of these variables would generally have a big impact in determining who is a superhost, since these hosts have to go above and beyond in order to reach that status on the platform[1].

Model 3: k-Nearest Neighbours

Before we deploy our k-NN model, we investigate the distribution of the price and found out that it is heavily skewed below USD 400 per night (*Fig 24*). Therefore, we decided to discretize the price into price categories to make it relatively equal in distribution, namely Low, Medium - Low, Medium - High, and High (*Fig 25*). We then follow the idea in Logistic Regression to compare the results from the model with all variables are being included and from the model with variables obtained using features extraction technique.

Based on our results (Appendix B, Table 4.1), the embedded model produces slightly higher accuracy (61,34 %) than the baseline model (59,93%). The baseline model and the embedded model predict high price category better than any other price category with around 80% accuracy in both models (Appendix B. Table 4.2 and 4.3). Significant improvement happens in predicting low price category.

There are seemingly consistent results for predictors across categories of price in both baseline and embedded models with features such as Accommodates, Bathrooms, Bedrooms, Beds, and Cancellation Policy being the most important features. Although it is also worth noting that in both baseline and embedded model, they seem to lose the predictive power in Medium - High price category (*Fig 26 and Fig 27*).

[1] Superhosts have 4.8 or above overall rating and their response rate is at least 90%.
https://www.airbnb.com/superhost

Conclusion

Overall, our machine learning algorithms yielded satisfactory results towards predicting superhost status and price. The logistic regression performs quite well with an accuracy of ~74%. On the other hand, the knn algorithm's performance lags somewhat behind at ~60% accuracy, however it still fulfills its duty. Whilst it increases the prediction accuracy in k-NN model, the embedded method contributes largely in improving prediction accuracy for Low price category. On the other hand, on logistic regression, embedded method slightly decreased accuracy.

Also, our procedures allowed us to capture many relevant insights regarding the data, from features of the property listings, of the hosts, the drivers of price, the variables with more predictive power towards superhost status, among others. Besides, we learned that most relevant variables to predict superhost status were host response rate, review score cleanliness and reviews per month.

APPENDIX A

Fig 01: Nr. of Listings per Market

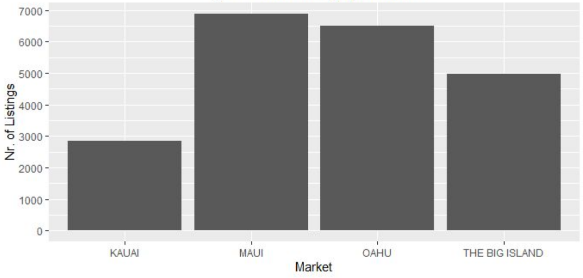


Fig 02: Nr. of Listings per Property Typee

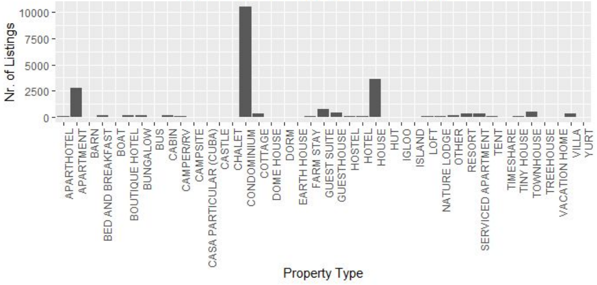


Fig 03: Nr. of Listings per Main Property Type

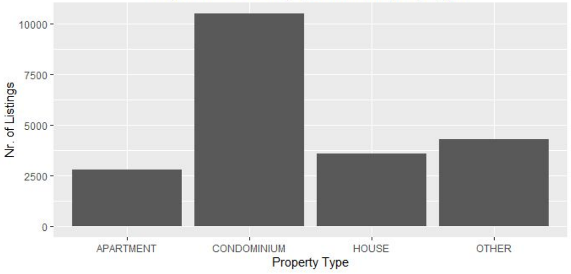


Fig 04: Mix of Main Property Types per Market

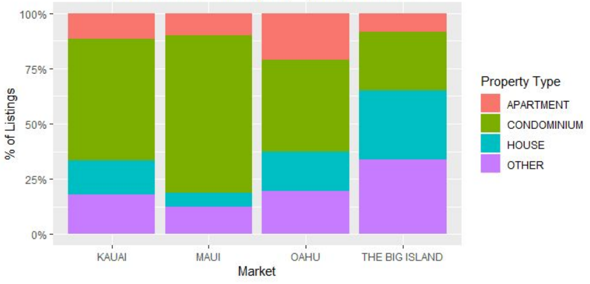


Fig 05: Nr. of Listings per Room Type

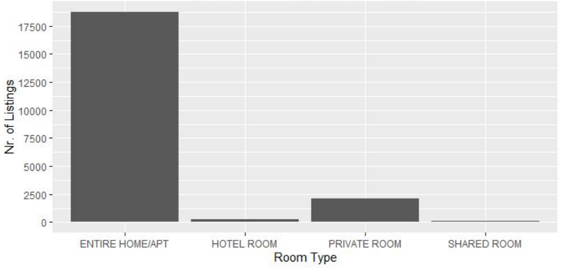


Fig 06: Median Minimum Nights per Market

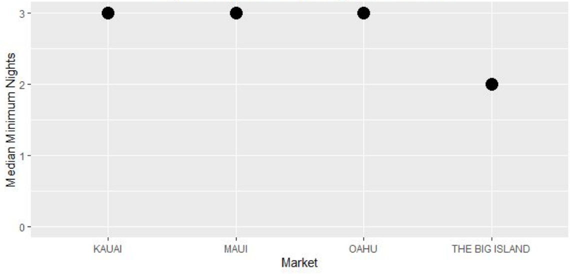


Fig 07: Median Minimum Nights per Main Property Type

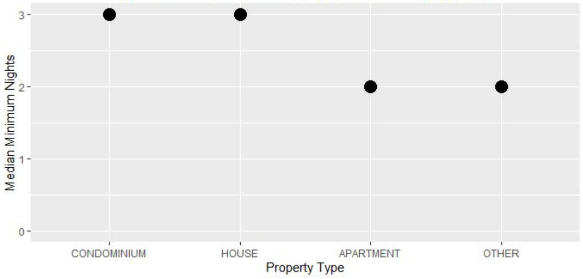


Fig 08: Median Minimum Nights per Room Type

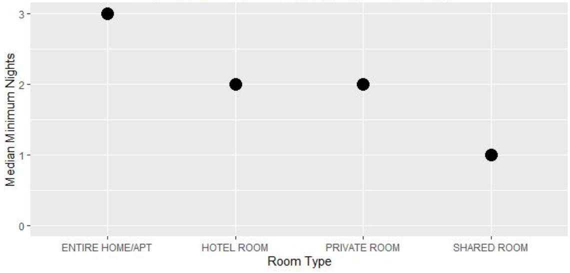


Fig 09: Median Nr. of Guests Accommodated per Market

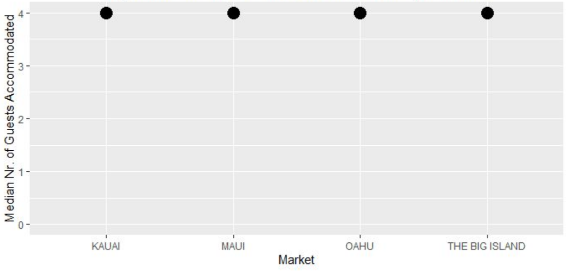


Fig 10: Median Nr. of Guests Accommodated per Main Property Type

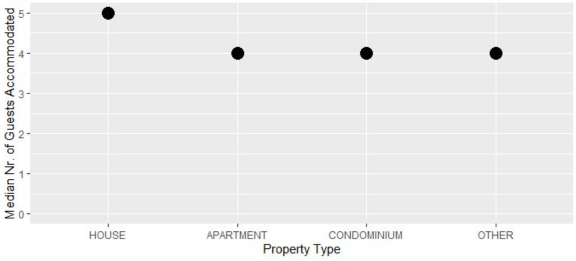


Fig 11: Mean Response Rate per Market

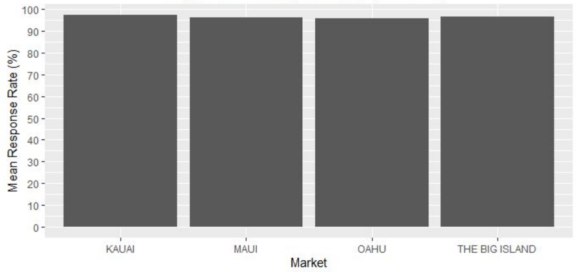


Fig 12: Mean Review Scores Rating per Market

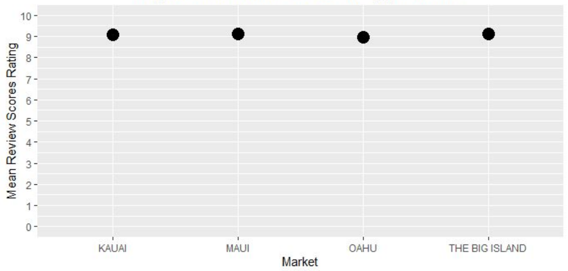


Fig 13: Mix of Cancellation Policy Types per Market

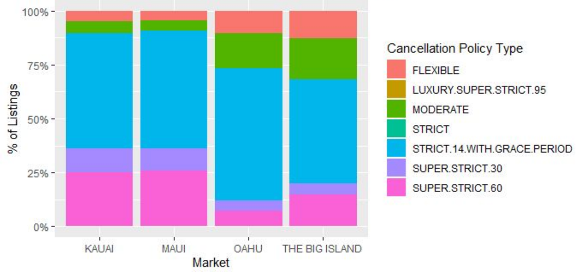


Fig 14: Mix of Response Times per Superthost Status

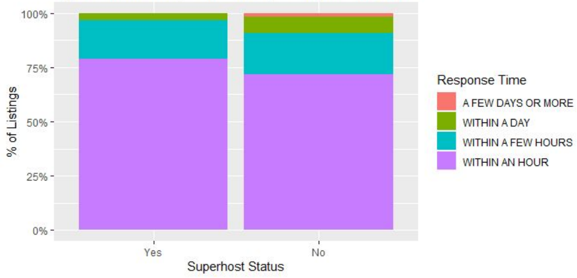


Fig 15: Mean Response Rate per Superhost Status

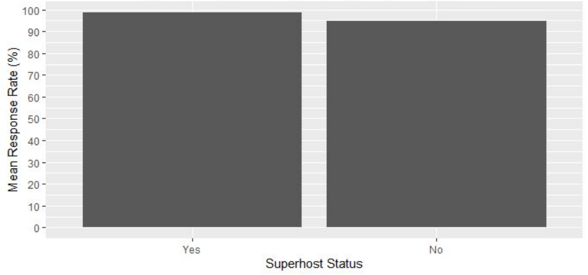


Fig 16: Mean Review Scores Rating per Superhost Status

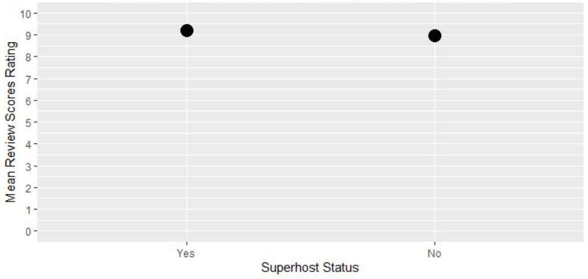


Fig 17: Median Price per Market

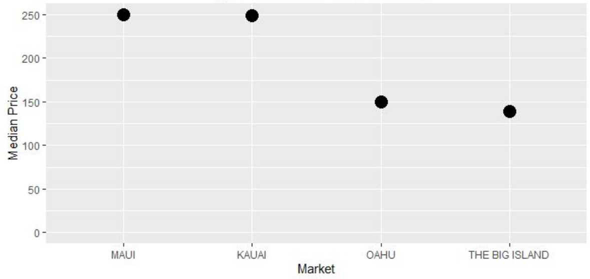


Fig 18: Median Price per Property Type

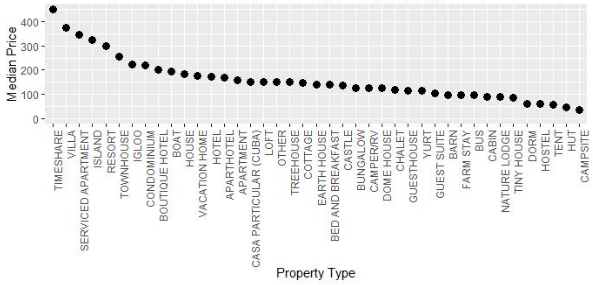


Fig 19: Median Price per Main Property Type

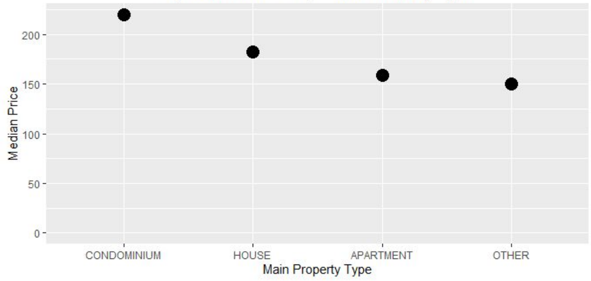


Fig 20: Median Price per Room Type

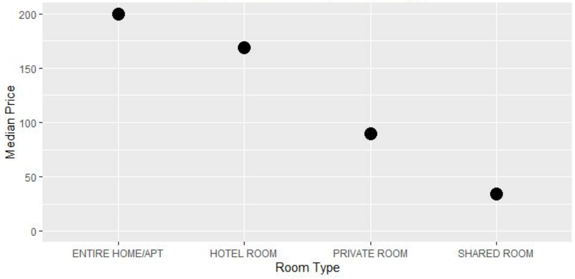


Fig 21: Median Price per Nr. of Guests Accommodated

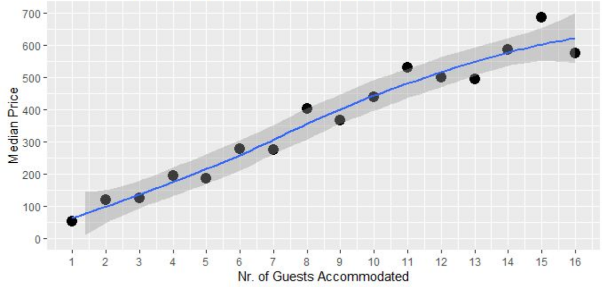


Fig 22. Variable importance - Baseline model

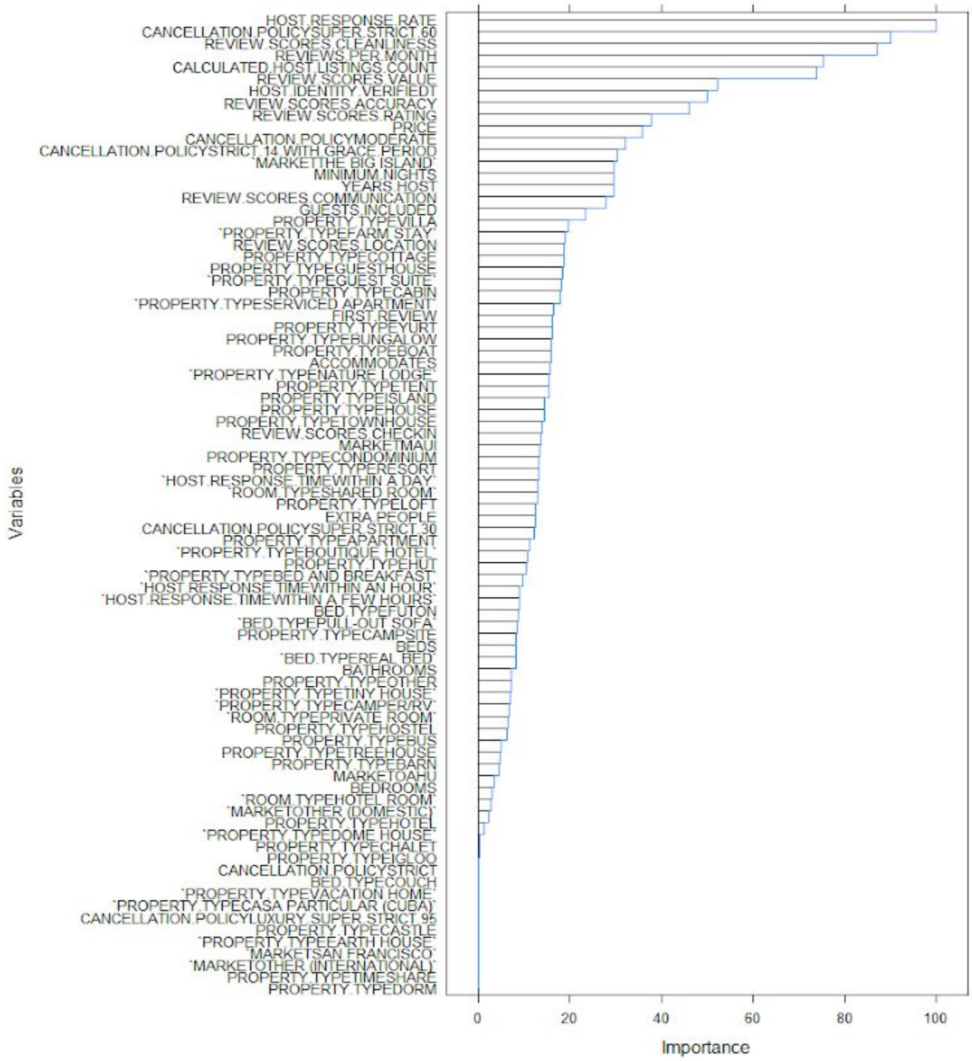


Fig 23. Variable importance - Embedded method model

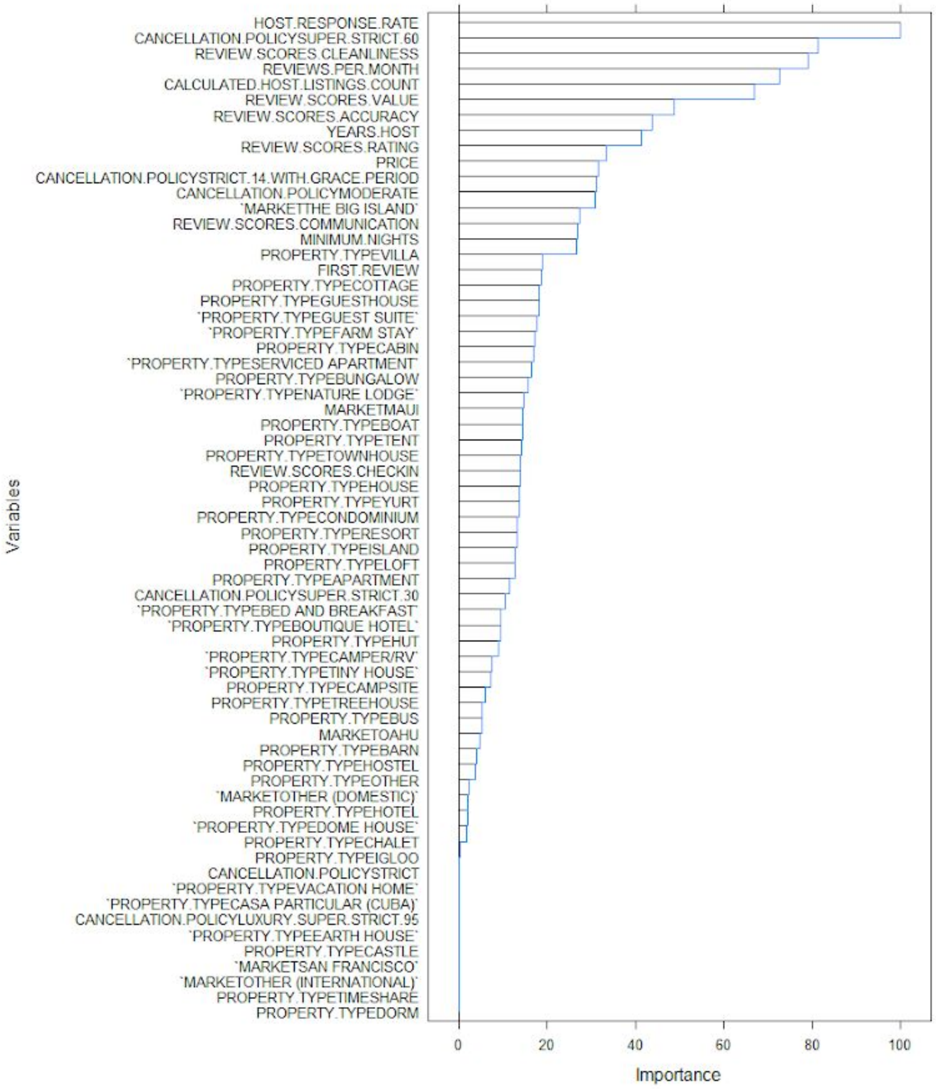


Fig 24. Histogram of Price

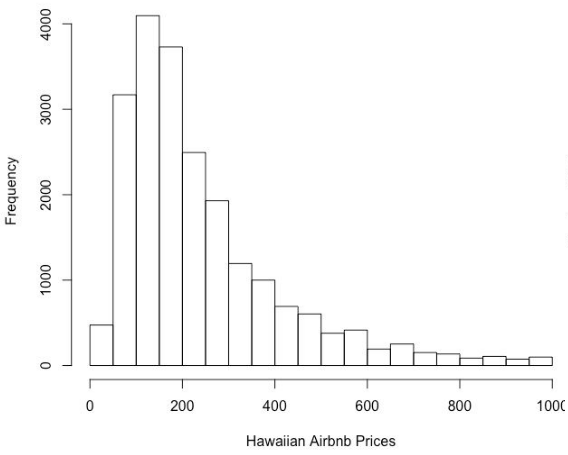


Fig 25. Number of Listing per Price Category

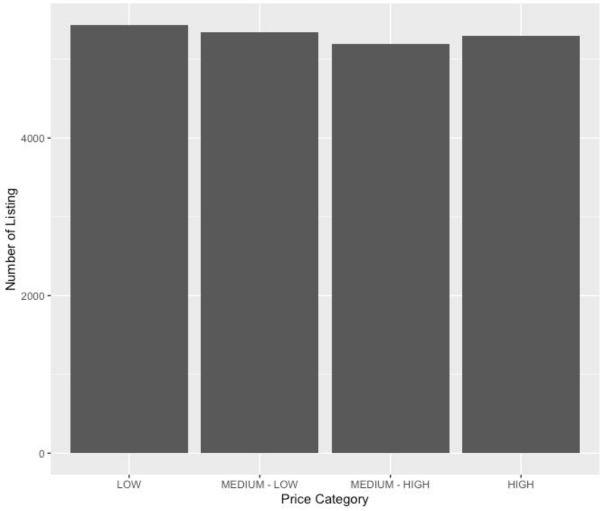


Fig 26. Variable Importance in Predicting Price using baseline k-NN model

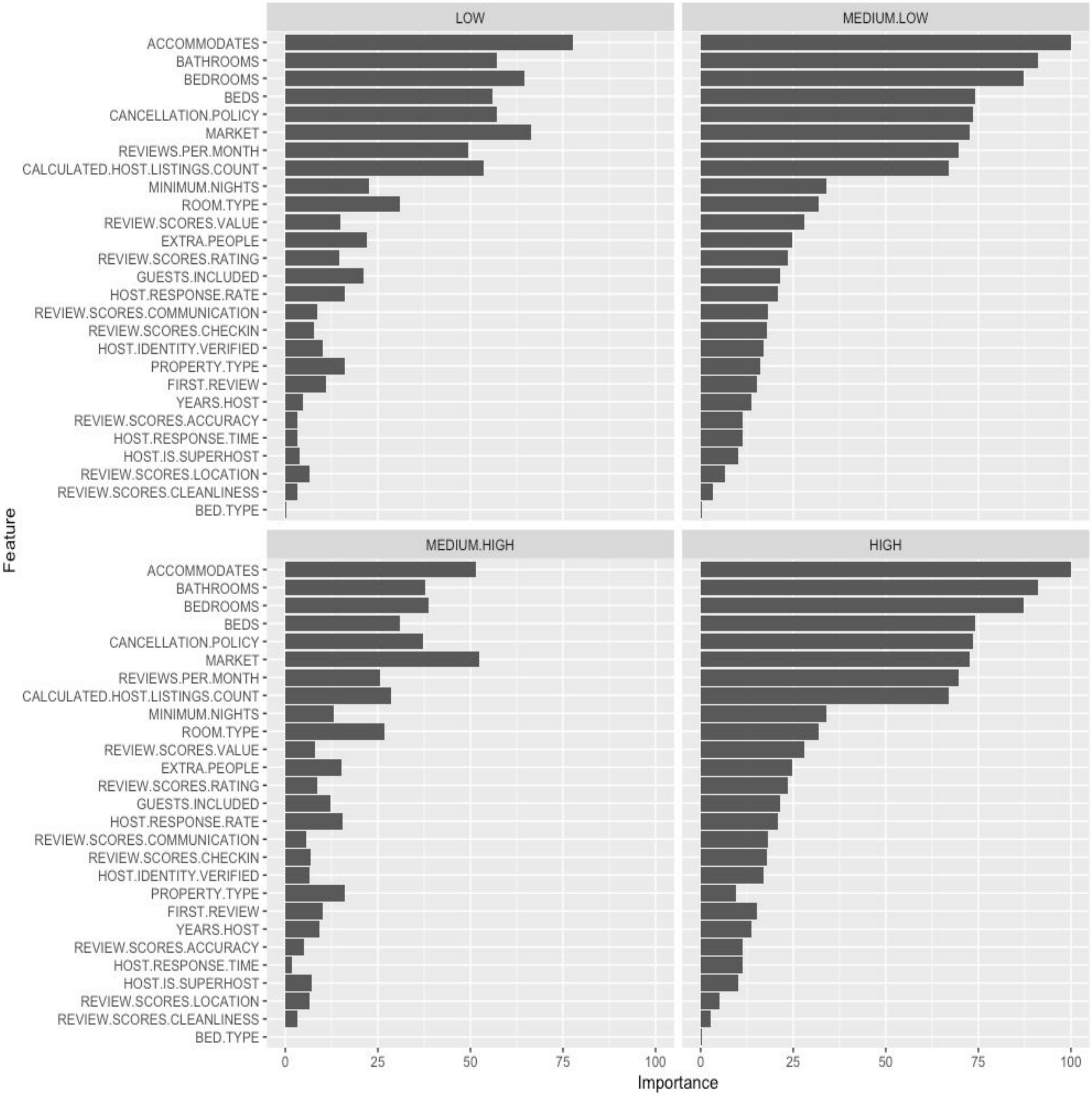
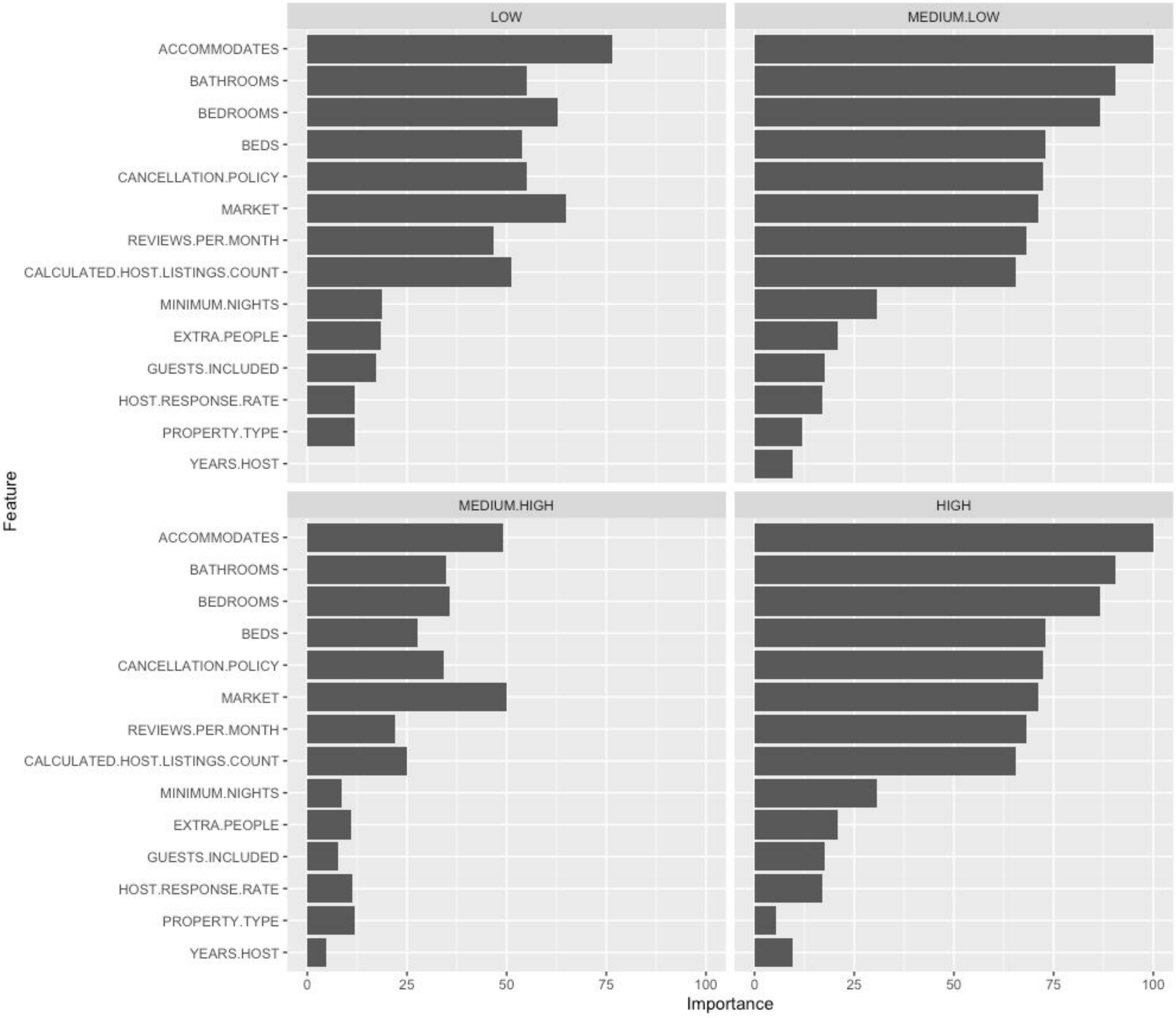


Fig 27. Variable Importance in Predicting Price using embedded k-NN model



APPENDIX B

Table 1

Confusion Matrix - Logistic Regression

Metric	Baseline model	Embedded model
Accuracy	0.7394	0.7354
95% CI	(0.7259, 0.7529)	(0.7218, 0.7486)
No Information Rate	0.6193	0.6193
P-Value [Acc > NIR]	< 2.2e-16	< 2.2e-16
Kappa	0.438	0.429
Mcnemar's Test P-Value	3.414e-05	2.03e-05
Sensitivity	0.8159	0.8137
Specificity	0.6148	0.6080
Pos Pred Value	0.7751	0.7715
Neg Pred Value	0.6725	0.6673
Prevalence	0.6193	0.6193
Detection Rate	0.5053	0.5039
Detection Prevalence	0.6519	0.6531
Balanced Accuracy	0.7154	0.7108

Table 2

Logistic Regression Top Coefficients - Baseline Model

Coefficients	**Estimate**	**Std.Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	-1.814551e+01	2.779417e+00	-6.52852902	6.641874e-11
HOST RESPONSE RATE	7.479864e-02	4.201294e-03	17.80371310	6.613499e-71
REVIEW SCORES CLEANLINESS	5.387065e-01	3.480233e-02	15.47903719	4.806119e-54
REVIEWS PER MONTH	2.003226e-01	1.493162e-02	13.41600267	4.872975e-41
CALCULATED HOST LISTINGS COUNT	-5.117755e-03	3.888483e-04	-13.16131595	1.465116e-39
CANCELLATION POLICY LUXURY SUPER STRICT.95	1.455280e+01	5.354112e+02	0.02718061	9.783157e-01

Table 3

Logistic Regression Top Coefficients - Embedded Method Model

Coefficients	**Estimate**	**Std.Error**	**Z Value**	**Pr(>|z|)**
(Intercept)	-2.334971e+01	7.440347e-01	-31.3825611	3.499963
HOST RESPONSE RATE	7.900726e-02	3.955554e-03	19.97375590	9.317639e-89
REVIEW SCORES CLEANLINESS	5.449216e-01	3.454946e-02	15.77221983	4.832115e-56
REVIEWS PER MONTH	2.113862e-01	1.458724e-02	14.49117559	1.377646e-47
CALCULATED HOST LISTINGS COUNT	-5.084751e-03	3.799467e-04	-13.38280059	7.622038e-41
CANCELLATION POLICY LUXURY SUPER STRICT.95	1.473722e+01	5.354112e+02	0.02752505	9.780410e-01

Table 4

Output Statistics - k-Nearest Neighbours

4.1 Overall Statistics

Metrics	Baseline Model	Embedded Model
Accuracy	0.5994	0.6135
95% CI	(0.5827, 0.6159)	(0.5969, 0.6299)
No Information Rate	0.2569	0.2569
P-Value [Acc > NIR]	< 2.2E-16	< 2.2E-16
Kappa	0.4657	0.4845
Mcnemar's Test P-Value	0.0002012	0.0003761

4.2 Statistics by Class - Baseline Model

Statistics	Baseline Model			
	LOW	MEDIUM.LOW	MEDIUM. HIGH	HIGH
Sensitivity	0.692	0.489	0.544	0.669
Specificity	0.879	0.822	0.837	0.928
Pos Pred Value	0.665	0.478	0.518	0.756
Neg Pred Value	0.892	0.828	0.85	0.894
Precision	0.665	0.478	0.518	0.756
Recall	0.692	0.489	0.544	0.669
F1	0.678	0.483	0.531	0.71
Prevalence	0.257	0.25	0.244	0.249

4.3 Statistics by Class - Embedded Model

Statistics	Embedded Model			
	LOW	MEDIUM.LOW	MEDIUM. HIGH	HIGH
Sensitivity	0.714	0.508	0.544	0.684
Specificity	0.883	0.834	0.84	0.928
Pos Pred Value	0.678	0.505	0.524	0.759
Neg Pred Value	0.899	0.835	0.851	0.899
Precision	0.678	0.505	0.524	0.759
Recall	0.714	0.508	0.544	0.684
F1	0.696	0.506	0.534	0.72
Prevalence	0.257	0.25	0.244	0.249