DISCOVERING AIRBNB IN MELBOURNE
EXECUTIVE SUMMARY

by Airbnb Tycoons:

GONDOPRASTOWO, Ivan	(3035457642; MStat)
LAM, Hin Tai	(2004062587; MStat)
SO, Kong Lun Alan	(3035456806; MStat)
YUEN, Hoi Chon	(3035456973; MStat)

Introduction

Ever since its inception back in 2008, Airbnb has revolutionized the industries it is operating in, namely travel accommodation and real estate industry. Airbnb is mostly perceived as a substitute for existing accommodations, usually for mid-range hotels.[1] It decentralizes the travel accommodation industry, allowing not only the players such as big hotel chains, hostels and proper accommodation providers, but virtually anyone who has spare rooms in their apartment or house to share. The concept and the platform have helped world travellers, home owners, families and real estate investors alike.

Project Objective

The new opportunity in real estate investment that is ignited by Airbnb platform has challenged our team to use data mining techniques to predict the potential income of a property as Airbnb host. We believe this will be valuable for real estate investors looking to analyze their Airbnb property investments.

In summary, our project objectives are:

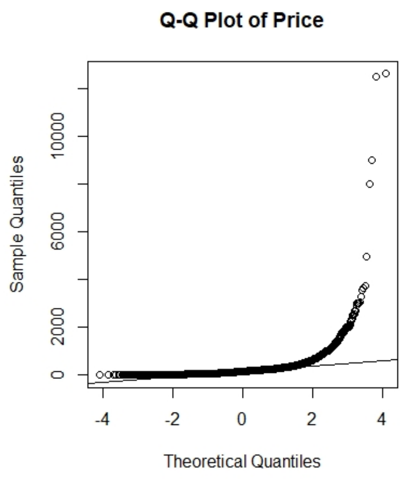
1. Help Airbnb hosts to determine the fair price of their current listing;
2. Provide guidelines to achieve higher occupancy rate to maximize rental income;
3. Seek for optimization on factors affecting occupancy/rental such as reviews, locations; and
4. Provide objective recommendations to travellers to rent Airbnb based on price-performance ratio.

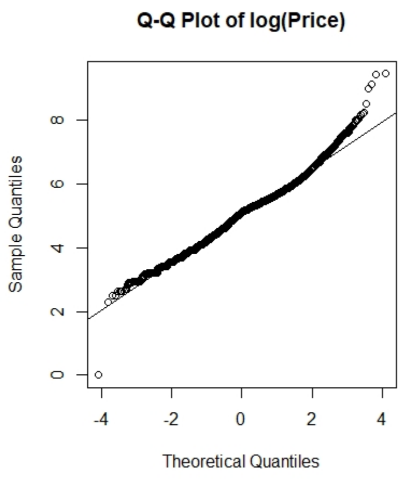
Through the findings of this project, it is anticipated an evidence-based investment decision can be formulated which generates better investment income. Likewise on the traveler side, data-driven objective evaluation of each listing can help traveler pick the best Airbnb for them.

Dataset

The dataset is downloaded from Kaggle “Melbourne Airbnb Open Data” which originally sourced from Inside Airbnb. The data utilizes public information compiled from the Airbnb website including the availability calendar for 365 days in the future, and the reviews for each listing. With the aim of predicting the price (target variable), there are a total 99 features available for model fitting. Features with either all blank value or almost 100% single value are also removed. Finally, 40 fields are retained and among the selected features, missing values were only observed for feature ‘host_total_listings_count' and they are trivial (3 missing out of 22895 observations). Hence, the missing values are simply imputed by median.

Data Transformation





As demonstrated in the qq plot of the target variable price (left), the normality assumption does not hold. A natural remedy is the logarithm transform. After transformation, the qq plot (right) showed a better alignments of sample quantiles to theoretical quantiles.

Model Discussion

Linear Regression

The regression model is a traditional parametric model for continuous target variable. It is usually assumed the target is normally distributed. We can observe that there is a significant improvement in model fitting after taking the logarithm transformation.

The linear regression model has the coefficient of determination (R^2) of 66.14% and 65.71% for the training and testing sets respectively and it serves as the base model to evaluate the performance of other data mining techniques.

Random Forest

Random forest is an ensemble method based on multiple decision trees. The learning for each tree in the ensemble is based on a bootstrap sample (random selection with replacement) of the data. Each splitting is performed on a random subset of features, with the aim of averaging to achieve a better overall prediction performance.[2] This randomness usually gives an edge to random forest in the prediction accuracy when the fitted model is applied to predict from unseen data.

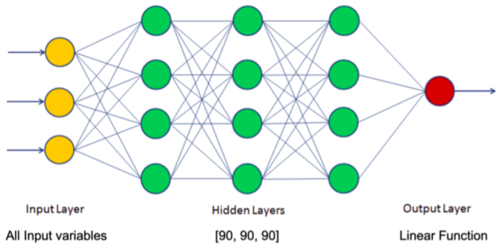
After fitting with a 10-fold cross validation, the random forest model that has achieved the best regression metrics with max depth of 17 and number of tree estimators of 100.

Artificial Neural Network (ANN)

Artificial Neural network is the most discussed machine learning model in recent years thanks to the advancement of computation power. Some consider it as the most powerful machine learning algorithms after Alpha Go, a computer program which is developed based on deep neural network. In fact, given the dataset is large enough, neural network could outclass almost all machine learning models.[3]

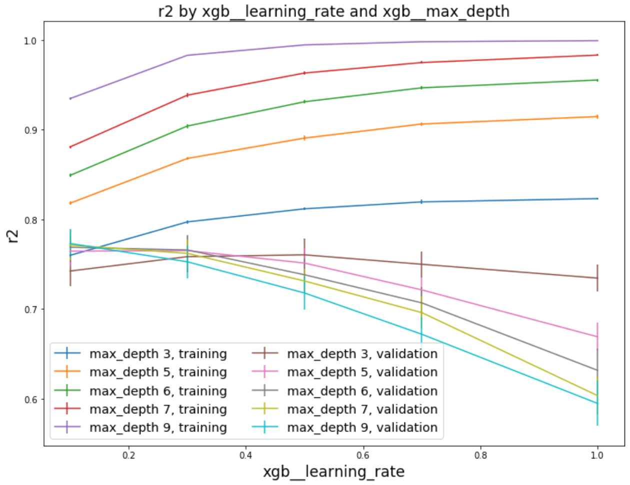
ANN composes of input layer and hidden layers, which are connected by hidden layers. Different combination of neurons, hidden layer, and other hyper-parameters are examined in our model training and cross validation process. The one with lowest mean absolute error (MAE) is finally chosen. To avoid overfitting, L2 regularization is applied in the ANN model.

After extensive computational random search with cross validation, the ANN model with 3 hidden layers, 90 nodes in each hidden layers, 0.0006 learning rate, 0.0005 lambda (for L2 regularization), 50 iterations and 256 batch size is chosen as the locally optimized model.



Extreme Gradient Boosting

The extreme gradient boosting (xgboost)[4] is an implementation of gradient boosted trees. It is similar to random forest and the difference arises from how it is being trained. This model is also designed to handle sparse data. In addition, the model is known to consider cache access patterns, data compression and sharing which aids in its performance and scalability for practical use. The preliminary model of this type yields marginally worse result than random forest, but already giving better result than linear regression, suggesting that this model has the potential to be the model of choice.



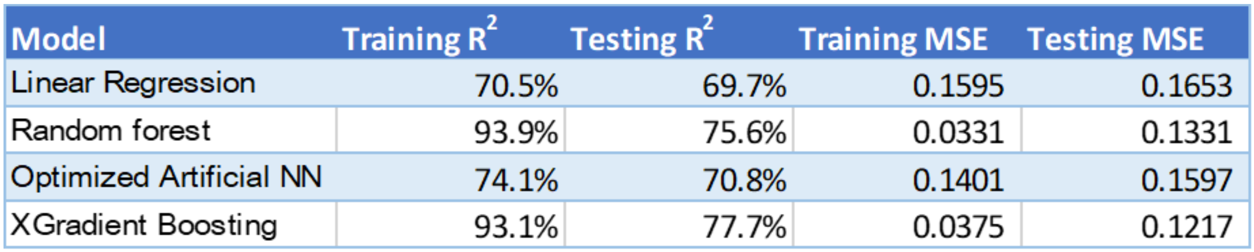
As with other model validations in this project, we split the data into 80% training set and 20% testing set (held-out for final model performance evaluation). We adjust the maximum depth and the learning rate of the model according to the R^2 and MSE by 10-fold cross-validation within the training set. As we can see from the graph on the left hand side, increase in max depth and learning rate in general yields a better performance when training model. However, the performance suggests otherwise in the case with the validation set. This suggests that while it is important to minimize the loss on training data, we must not forget the importance on regularization.

With regularization in mind, we try to pick the minimum max depth required to achieve a high enough R^2 and minimize MSE. We did a second round confirmatory parameter grid search on learning rate and max depth, and found out from the validation R^2 score that the best model has a learning rate of 0.1 and max depth of 9. Though the model with max depth of 9 only gave a marginally higher validation score than the other depths. The difference from varying the learning rate is more significant.

The result as can be seen on model comparison section below, is a model that performs as good in training set, but best in the testing set by R^2 and mean square error (MSE) performance measure.

Model Comparison Summary

In order to conduct a fair comparison, a consistent split of dataset (80% for training and 20% for testing) is applied to various models.

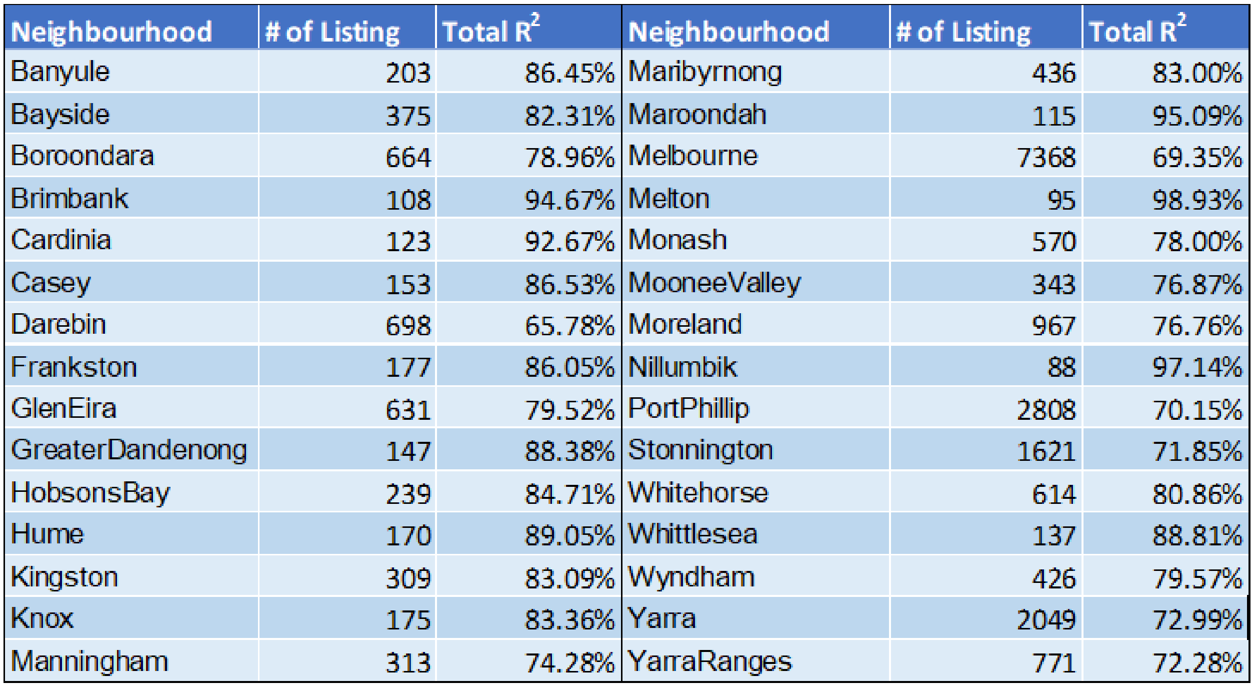


Model	Training R^2	Testing R^2	Training MSE	Testing MSE
Linear Regression	70.5%	69.7%	0.1595	0.1653
Random forest	93.9%	75.6%	0.0331	0.1331
Optimized Artificial NN	74.1%	70.8%	0.1401	0.1597
XGradient Boosting	93.1%	77.7%	0.0375	0.1217

Extreme gradient boosting is best in terms on R^2 and MSE of the testing. The difference between the values from the training set and testing set may indicate the problem of overfitting.

Further Consideration on Price Model

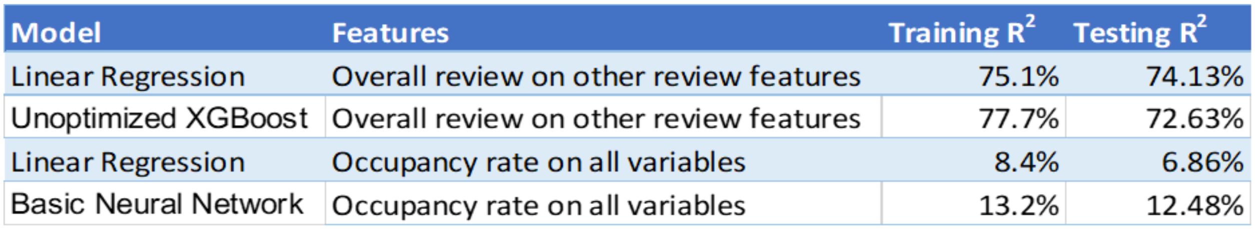
Considering the prediction from other perspective, different neighborhoods might have different behavior. The city of Melbourne is divided into 30 neighbourhoods by administration and a model is set up for each neighbourhood might be appropriate as each should be more homogeneous price distribution. The high R^2 from the fitted regression models reflects the models explain the existing dataset well. However, due to limited observations in each neighbour, the performance of train/test split is subject to high variability.



Neighbourhood	# of Listing	Total R^2	Neighbourhood	# of Listing	Total R^2
Banyule	203	86.45%	Maribyrnong	436	83.00%
Bayside	375	82.31%	Maroondah	115	95.09%
Boroondara	664	78.96%	Melbourne	7368	69.35%
Brimbank	108	94.67%	Melton	95	98.93%
Cardinia	123	92.67%	Monash	570	78.00%
Casey	153	86.53%	MooneeValley	343	76.87%
Darebin	698	65.78%	Moreland	967	76.76%
Frankston	177	86.05%	Nillumbik	88	97.14%
GlenEira	631	79.52%	PortPhillip	2808	70.15%
GreaterDandenong	147	88.38%	Stonnington	1621	71.85%
HobsonsBay	239	84.71%	Whitehorse	614	80.86%
Hume	170	89.05%	Whittlesea	137	88.81%
Kingston	309	83.09%	Wyndham	426	79.57%
Knox	175	83.36%	Yarra	2049	72.99%
Manningham	313	74.28%	YarraRanges	771	72.28%

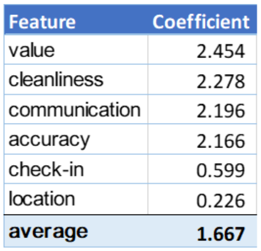
Further Exploration

As mentioned, some exploratory analysis is being performed to be able to predict other useful information on Airbnb investment such as the occupancy rate and the overall review rating component. The summary is presented in the following table.



Model	Features	Training R^2	Testing R^2
Linear Regression	Overall review on other review features	75.1%	74.13%
Unoptimized XGBoost	Overall review on other review features	77.7%	72.63%
Linear Regression	Occupancy rate on all variables	8.4%	6.86%
Basic Neural Network	Occupancy rate on all variables	13.2%	12.48%

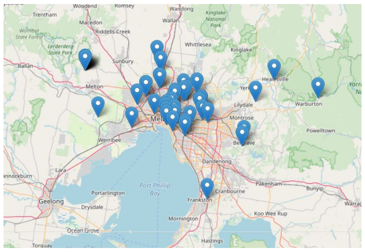
Apart from the price, we try to discover more interesting information from the data, although they might not have been included in our project proposal. Firstly, we discover how guests come up with overall rating given their rating on other fields such as accuracy, cleanliness, check-in experience, guest communication, location and value for money. This is because Airbnb shows the overall score but not the detailed rating, which incentivizes hosts to maximize areas that contribute to the overall rating the most.



Feature	Coefficient
value	2.454
cleanliness	2.278
communication	2.196
accuracy	2.166
check-in	0.599
location	0.226
average	**1.667**

Another interesting question that arisen is the ability to determine occupancy rate from the features available in the data. Unfortunately, we are unable to find a reliable model as R^2 does not exceed even 20% as can be seen above. This indicates that either the data is not sufficient or suitable to predict occupancy rate or further pre-processing on the data is required and analysis to find suitable model to predict occupancy rate.

Lastly, in addition to determination of a fair price for the Airbnb providers, our models can be used to identify great deals for the customers in terms of price–performance ratio. This can be done by finding those Airbnb with listed price far below the predicted price. The top 50 best-valued Airbnb are plotted on the map on the right.



Conclusion

The project aims to derive interesting insights from Airbnb listings and reviews which are useful for investors to start an Airbnb rental investment. We started the study with pre-processing of the data and did the required transformations. We then employ various data mining techniques to come up with extreme gradient boosting as the best model to predict fair price of a listing. Further, different regression models are fitted for various neighborhoods and they can well explain the existing situations . In addition to the prediction of price, we worked on other objectives such as the occupancy rate prediction and how each feature of an experience review affects the overall rating assigned. Lastly, objective Airbnb recommendations in terms of price-performance are recommended to the travellers.

References

[1] Guttentag, D. (2015). Airbnb: disruptive innovation and the rise of an informal tourism accommodation sector. *Curr. Issues Tourism 18(12)*, pp. 1192–1217.

[2] Breiman, L. (2001). Random Forests. *Machine Learning, 45(1)*, pp. 5-32.

[3] Siegelmann, H. T. & Sontag, E. D. (1995). On the computational power of neural nets. *J. Comput. System Sci., 50(1)*, pp. 132-150.

[4] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.