QBUS6810 Group Assignment

Group 96

Semester 2, 2023

SIDs

520398467

520302774

520389591

510665339

520167416

Table of Contents

1 Introduction

Over the past 18 months, there has been a significant surge in the supply of properties across the key real estate markets in Australia. Notably, Airbnb listings have experienced a notable 25% upsurge in comparison to the preceding year, reaching a cumulative total of 166,000 listings as of June 2023 (Khandelwal, 2023). However, for aspiring Airbnb hosts, establishing an optimal pricing strategy has proven to be a challenging undertaking. Several factors necessitate careful consideration, including the time investment associated with property management, initial setup expenses, market demand dynamics, and the decision of whether to designate the property for short-term or long-term rentals. These elements collectively influence the pricing strategies adopted by landlords (REI, 2023).

In response to the pandemic-induced restrictions and the rapid proliferation of industry competitors like Bees Knees vacation rental and property service firms (Bronwyn & Adcock, 2022), Airbnb endeavors to identify listings that can yield higher profits. Thus, our team undertook a comprehensive analysis and modeling of the existing dataset to forecast the nightly property rates along the eastern coast of Australia.

In this report, after evaluating multiple models, we focused on three specific models - Elastic Net, Decision Tree, and Xgboost - for predictive modeling. Subsequent comparisons revealed that the Xgboost model demonstrates superior predictive accuracy. The model exhibits the smallest margin of error when forecasting data, thus indicating its capability to provide more precise estimations for the nightly property prices along the Australian East Coast. Consequently, Xgboost was chosen as the final predictive model.

2 Data Processing

2.1 Data description

The training set data contains a total of 12,500 rows of data and 20 columns of data, including 8 columns of floating-point data, 6 columns of integer data, and 6 columns of string data.

The test set data contains a total of 2500 rows of data and 19 columns of data, including 8 columns of floating-point data, 5 columns of integer data, and 6 columns of string data.

2.2 Data processing

To begin with, we initiate the data import process, meticulously parsing the training and test datasets separately. Upon scrutinizing the training dataset, we note the presence of missing values solely within the 'beds' column, while the test dataset exhibits a complete absence of missing values. Subsequently, we conduct a comprehensive check for null values within the dataset. The findings indicate the existence of null values exclusively within the 'beds' column of the training dataset, whereas the test dataset remains devoid of any null values, aligning precisely with our observations on missing values. Guided by these insights, we opt to eliminate the two rows of data containing null values, thereby deriving the finalized training dataset.

Next is the relevant data processing in feature engineering：

Initially, we generate a dataset named X, encompassing all columns from the dm_train data frame except for the 'price' and 'log price' columns. This step is undertaken to eliminate the target variable utilized for model training, as well as any ancillary variables that could potentially impede the

prediction process. Subsequently, we establish a dataset named y, specifically incorporating the 'price' column from the dm_train data frame. This column serves as the target variable during the model training phase. Finally, the dataset is partitioned into an 80% training set and a 20% validation set. This deliberate division ensures the availability of data for assessing the model's efficacy during the training phase, thereby facilitating an evaluation of its performance.

3 Exploratory Data Analysis

3.1 Response variable

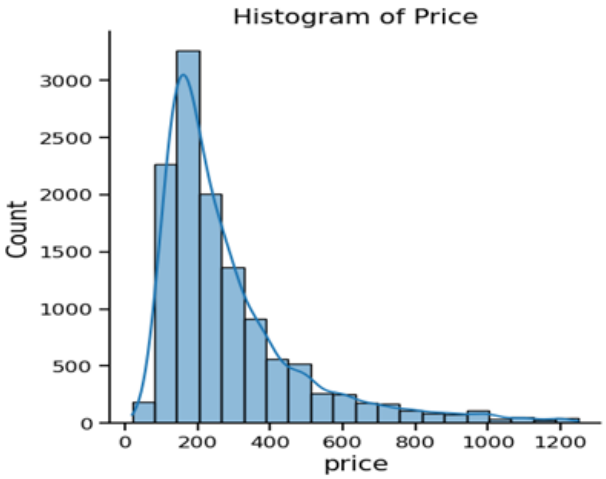

Figure 1

For the response variable ‘price’, Figure 1 shows there are highly right skews, the prices are concentrated in the range of $150 to $250, and due to that, we perform log transformation on price.

3.2 Textual Variable

From the data set, we can see that there are some textual data, which are description, neighborhood_overview, and amenities. In general, the host's literal description of the house is closely related to the price. Therefore, we extract the high-frequency words in the high-priced (price over 1000) house. According to the frequency of words, we can find out some words that may have an impact on the price:

- Description and Neighborhood_overview: After deleting the symbols and numbers contained in the data, we can find that in the data set ‘description’, the word ‘bedroom’ appears the most frequently, and in the data set ‘neighborhood_overview’, the word ‘beach’ appears the most frequently.
- Amenities: Though this data set looks like a list, it actually a string. We need to remove the square brackets, quotation marks, and spaces before further processing the data. Then, we can find out the word ‘kitchen’ appears the most frequently.

Although we found high-frequency words for each variable, it does not show that the higher the frequency of the words, the higher the price of the house. Because we only focused on homes priced above $1,000, these words appear more frequently in high-priced homes and may also appear more frequently in other price ranges, but this should not be the only basis for judging the price standard.

According to our subsequent data exploration and feature engineering, we found that when some high-frequency words were removed, their correlation with price and the results of RMSE performed better. Therefore, after reading the text description in the data, we found the common features in the three variables, for example:

1. Location: the expensive homes are generally clustered in more prosperous or scenic areas, such as Darling Harbour;
2. Coastal: close to the beach, such as Bondi Beach and Manly Beach;
3. Some adjectives words, such as the description, include words like ‘luxury’.

So, we decided to screen some keywords manually. Due to the correlation with price and the results of RMSE, we finally extracted 20 keywords in Description and 54 keywords in Neighborhood_overview and Amenities for later research.

3.3 Location Variable

In previous data explorations, we've seen that location has an existential effect on price. Therefore, we use the three variables neighborhood, latitude, and longitude in the data set to further study the relationship between location and price.

Firstly, we can check the average price of homes in each neighborhood.

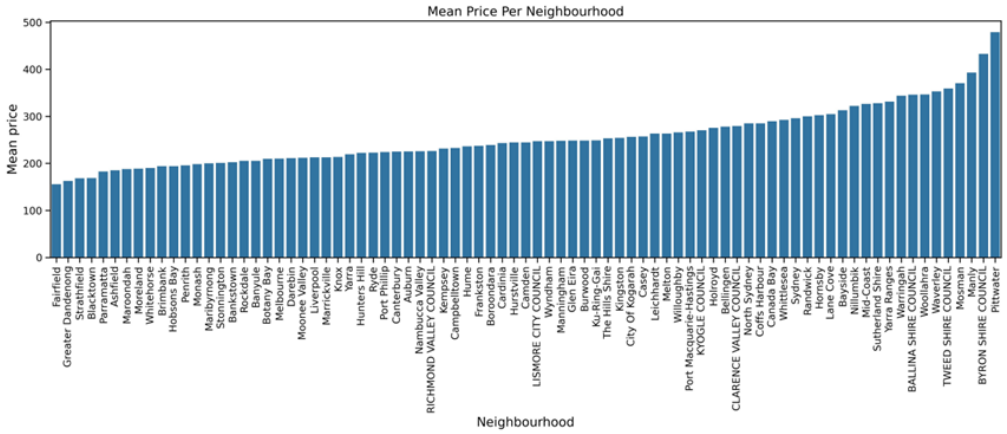

Figure 2

From Figure 2, it is not difficult to find that the neighborhoods close to ports or beaches, such as Pittwater, Byron Shire council, and Manly, are more expensive than others because most Airbnb properties are for travelers, that kind of neighborhood’s tourism is more developed and are more tourists to travel.

Then, Figure 3 shows the data is broadly divided into three categories by latitude and longitude, we can be roughly divided into the Melbourne area, the Gold Coast area, and the Sydney area.

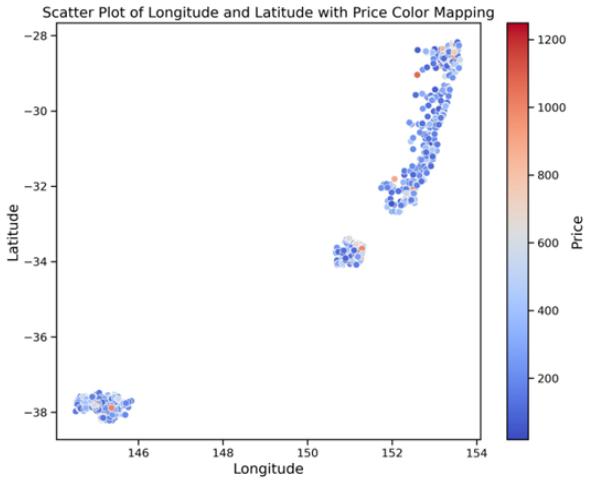

Figure 3

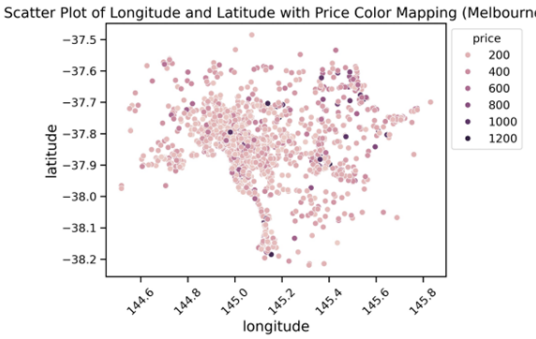

Figure 4

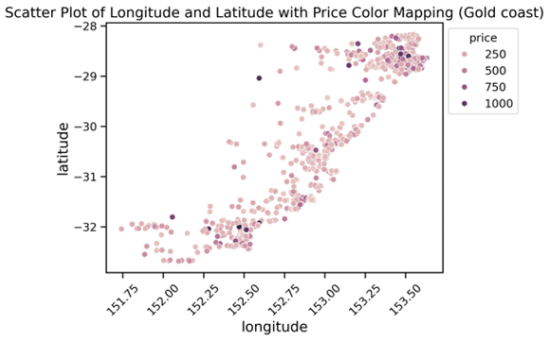

Figure 5

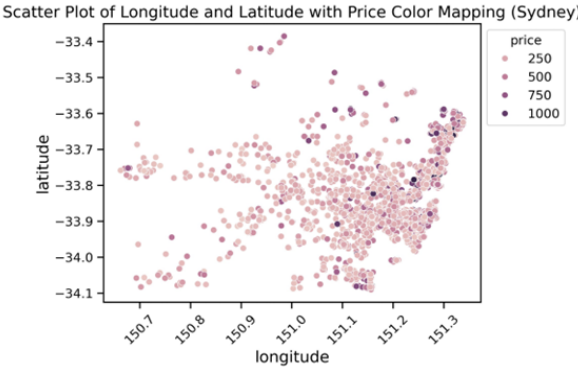

Figure 6

From Figures 4, 5, and 6 we can see that most of the houses are located around the city centre because this area is more convenient for transportation, easy for tourists to travel, and the economy is relatively prosperous and convenient for life. For Figures 5 and 6, as you move towards the coastline, the longitude increases and you can find that the price of the house also increases, this is because the beach is more popular with travelers. Also, it is worth mentioning that in Figure 5, it is obvious that the houses on the Gold Coast are distributed according to the coastline, because the Gold Coast is a tourist city, the homes closer to the coastline, the more popular.

3.4 Other Numerical Variable

We also explored other numerical variable data, such as the rate at which the host accepts booking requests, but did not find a significant linear relationship with prices or log transformation on price. However, there are still some new discoveries. When we explore the relationship of accommodates with log(price), we can see that there is a slight linear relationship (Figure 7). As the number of people living in the house increases, the area and facilities required for the house will also increase, which may cause the price to rise accordingly.

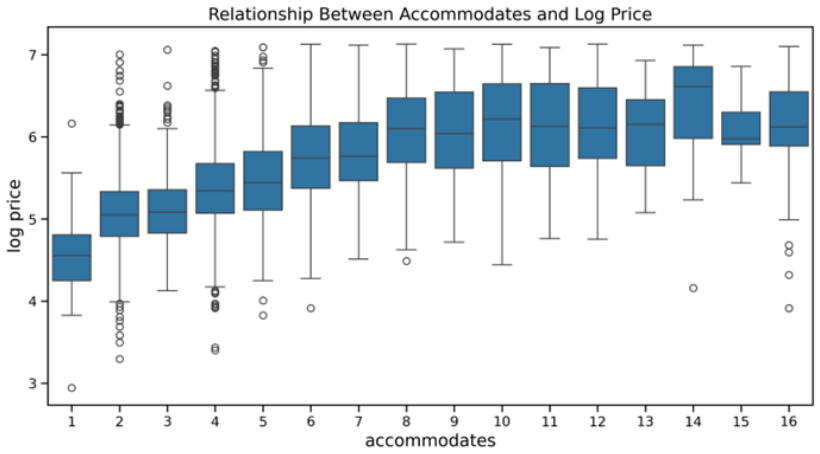

Figure 7

We further studied the variable ‘bedroom’, and found and deleted outliers with more than 12 bedrooms. Figure 8 shows there is a slight relationship between bedrooms and log(price) as well, which means when visitors need more bedrooms, it usually means that more people will stay overnight, and the price will be higher.

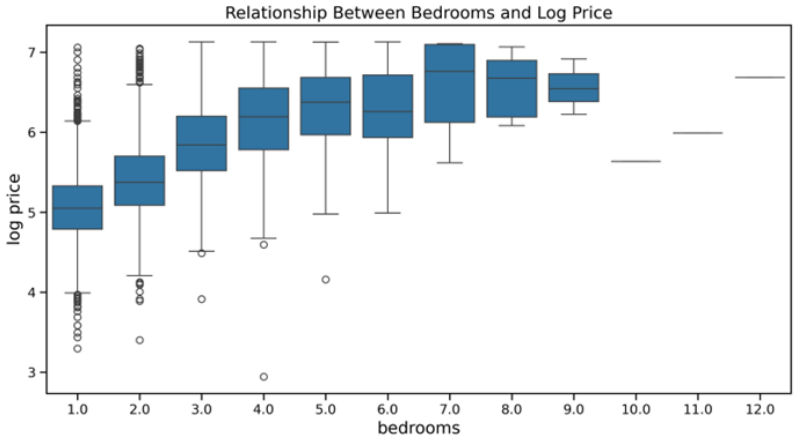

Figure 8

Moreover, we also found a tiny linear relationship between the log(price) and the number of beds (Figure 9), we found and deleted outliers with more than 20 beds. It makes sense that when more beds are customers needed, the prices rise.

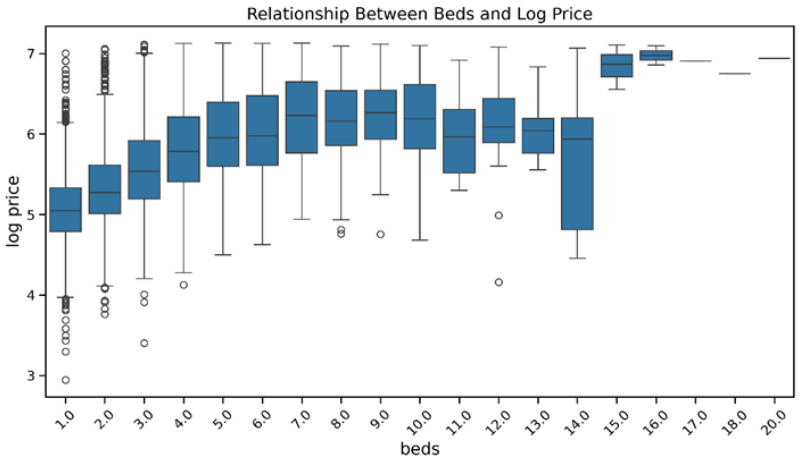

Figure 9

We also studied the variable “minimum night”, but found no effective information, most hosts set the minimum number of nights consumers can book the listing as 0. Then we analyzed consumer reviews but did not find any useful information. Most of the ratings are clustered around 5 points, which may mean that consumers are generally satisfied with their chosen home, and these ratings are not guaranteed to be true and valid.

Finally, Figure 10 shows the positive relationship between price and accommodates, bedrooms and beds are stronger because the correlation is closer to 1, this conforms to our previous research. These variables may be used in our future research.

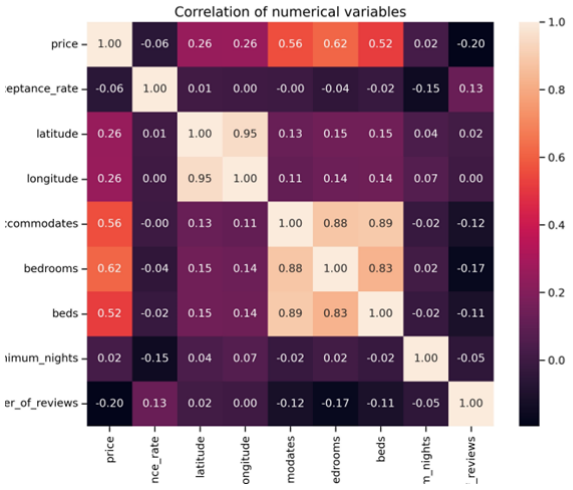

Figure 10

3.5 Other Categorical Variable

To deal with ‘property_type’ and ‘room_type’, we can find that the information in ‘property_type’ contains the information in ‘room_type’, and then we will further process it in feature engineering. Now, we can check out how room type interacts with price. Figure 11 shows that the entire home/apartment is more expensive than a private room, and a hotel room is more expensive than an entire home/apartment when the number of bedrooms is less than 4 (the maximum number of bedrooms of a hotel room is 4), and the private room has the lowest price.

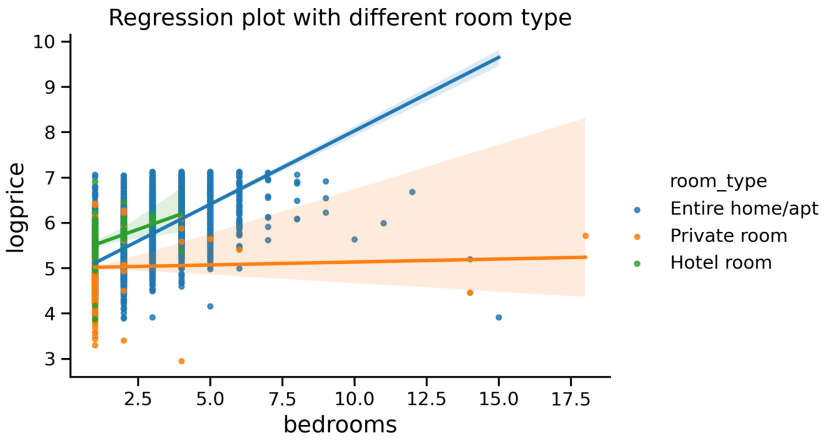

Figure 11

4 Feature Engineering

4.1 Textual Variable

Convert textual data into quantity data - key vocabulary statistics of 'description', 'neighborhood overview', and 'amenities'

We identified 20 high-frequency words based on word frequencies during the exploratory data analysis (EDA) phase, which we believe may significantly impact housing prices. Then, we plan to re-count the number of occurrences of these high-frequency words within the description of each listing in training and test datasets and intend to use them as new analytical features.

Simultaneously, in order to conduct a more in-depth analysis of the factors affecting housing prices, we extracted 54 keywords from listings with higher prices, respectively, involving the surrounding environment description and supporting facilities within the housing listings. Since these key words may potentially affect the price, we will tally the occurrences of these words in the neighborhood overview and amenities descriptions of each listing in both the training and test datasets. The new relevant features will be created through this process and enable a more detailed analysis in the following modeling. Our aim is to capture the key information in the text, providing a comprehensive understanding of how a listing's features influence its price. These new features are expected to furnish additional insights for modeling, enhancing the model's predictive capability for housing prices.

4.2 Location Variable

Combine and transform latitude and longitude data into a comprehensive geographical location feature

Research believes that the price of housing is closely related to its geographical location. The closer to the city center, the higher the price of housing (Perez-Sanchez et al., 2018). Therefore, we comprehensively considered the distribution data of housing listings in EDA, and finally set the CBD coordinate of three major cities of Melbourne, Sydney, and Gold Coast as reference centers. We computed the distance between each housing listing and its respective CBD by combining latitude and longitude information, which has been integrated as a new feature, serving as a valuable addition that

explains the spatial relationship to CBDs influences housing prices, enhancing the model's capacity to capture and interpret such geographical nuances.

Convert the value of latitude to a positive number

In geographical coordinates, the south latitude in latitude is represented by negative values, and the north latitude is shown by positive values (Rahul Awati, 2022), however, since the data will be applied to machine learning, in order to facilitate subsequent data processing and make the data conform to model expectations, we will convert negative values for southern latitudes to positive values to improve data uniformity.

4.3 Numerical Variable

The original data set review scores are divided into three parts. Hence, we hope to summarize and convert them into a comprehensive evaluation feature, in which we sum up the three scores and calculate the average. Then, we divide all review scores into intervals according to the score size and convert this numerical data into ordinal data. For example, scores ranging from 1 to 3 points are classified as 'low,' scores from 3 to 4 points as 'medium,' and scores from 4 to 5 points as 'high'.

4.4 Categorical Variable

Despite the extensive categories within the three categorical variables (property_type, room_type, and neighborhood), not all categories exhibit high frequency. To simplify subsequent modeling, we tend to reclassify less frequently occurring categories. Specifically, listings appearing fewer than 100 times in property_type are grouped under 'Other.' Additionally, given the low frequencies of private rooms and hotel rooms in room_type, both are consolidated into the 'single room' category. While neighborhoods boast numerous variations, for analytical convenience, areas with a frequency lower than 51 are collectively labeled as 'Other.' This approach streamlines the categorical variables, making them more manageable for analysis without altering the overall dataset structure.

4.5 Features with Multiple Variables

Neighborhood quality combined with price

According to Figure 12, we can conduct a deeper study of the distribution and correlation between housing prices and other variables. In order to explore the connection between listing prices and neighbourhood, we plan to create a new set of features that combine these two variables. Specifically, we will divide the neighborhood quality with price level into 'Low', 'Medium', and 'High'. Then, we will conduct a dummy variable transformation on the data, converting the processed categorical and ordinal variables into a numerical format comprehensible to machine learning models. Analyzing the correlation between these newly generated features and prices provides a nuanced insight into the impact of each variable on house prices.

Furthermore, to enhance data smoothness, we also apply logarithmic transformations to the variables, which mitigate the skewness observed in the data, facilitating the model in capturing the intricate relationships between variables.

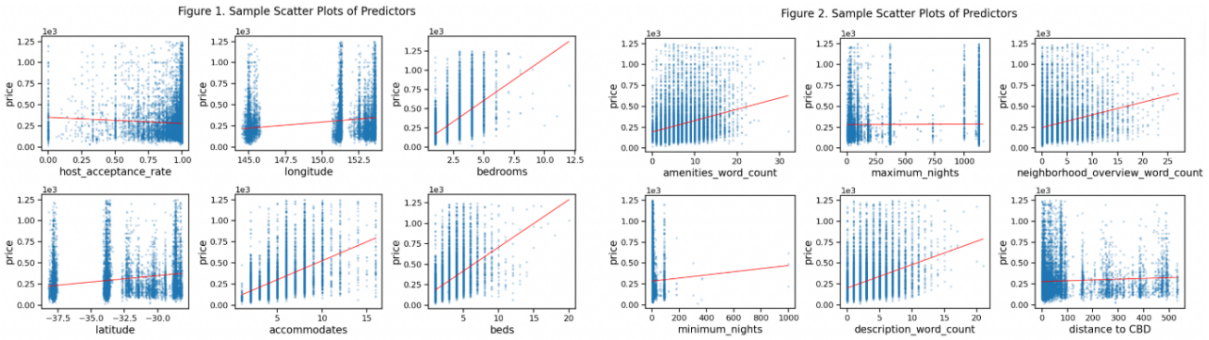

Figure 12

Process the data into a form that is easier to apply to linear models

In order to improve model performance and simplify the model structure in the subsequent process, we will select target variables with high correlation as key parameters of modeling, which helps to improve the predictive ability of the model and reduce the risk of overfitting. Additionally, we will check the skewness and kurtosis of numerical variables within the data and perform logarithmic transformation for those with bad sickness. Moreover, we will standardize the data to eliminate the influence of scale between different features, which enables each feature to have a mean of 0 and a standard deviation of 1, which ensures the models understand and weigh respective features better. Ultimately, we will construct some key interaction effects in advance to enhance the linear model's ability to adapt to nonlinear relationships. For example, we will explore how the number of reviews is related to neighborhood price quality, the number of rooms to neighborhood price quality, and log-transform all these relationships.

5 Methodology

5.1 Model Descriptions

We have selected three distinct models—Elastic Net, Decision Tree, and XGBoost—with the expectation that their unique strengths and focuses will collectively contribute to predicting future Airbnb house prices under constrained conditions.

- The Elastic Net linear regression model combines the lasso and ridge regression methods (CFI Team, 2023), which improves the limitations of lasso and is effective for studying high-dimensional data and features with collinearity. Our data set contains many features, so we hope that this model can effectively identify key features within the data and improve the generalization ability of the model.
- As a powerful nonlinear model, the Decision Tree can effectively capture the complex non-linear relationships between features (IBM.com, 2023). There may be non-linear relationships in the data that are difficult for us to observe and these factors may have an implied impact on housing price predictions, which the Decision Tree can intuitively explain the relationships and present the result directly.
- XGBoost is an ensemble learning method in machine learning, capable of aggregating the output of multiple models based on a single model structure (guest_blog, 2018). It has demonstrated remarkable success in addressing the instability of Decision Trees. Due to the model's capacity to dynamically adjust its complexity during training, we anticipate its efficiency in capturing nonlinear factors and producing high-quality prediction results. This

adaptability is particularly valuable when optimizing computation in the analysis of complex datasets.

5.2 Model 1: Linear Regression Model (ElasticNet Regression)

The first model we decided to use is the ElasticNet Regression model. Because it is a relatively simple model, it can be used as a baseline model that can be benchmarked against more complex models. Also, the model has no limit on the number of variables selected and is suitable for use when multiple features are related to price, easier to understand and explain (Elastic Net, n.d.).

Then, using "ElasticNetCV()" function to fit the logarithmic and standardized train set. ElasticNetCV can search for multiple alpha values and apply the best one (DataTechNotes, n.d.). We will then define the model with alpha values and use our training data for fitting. After that, we can see the l1_ratio equal to 1. From the:

$$\hat{\beta}_{\text{elastic net}}=\text{argmin}\beta\|y-X\beta\|_2^2+\alpha l1\_ratio\|\beta\|_1+0.5\alpha(1-l1\_ratio)\|\beta\|_2^2$$

when l1_ratio =1, which shows the penalty just the L1 penalty, we can think of this model as the Lasso regression model. Now, we can use the validation set to verify the model, and the Root Mean Squared Error is 145.1885. Meanwhile, the Root Mean Square Error of the whole training set (combining the train set and the validation set again) is 144.9244. This result shows that the linear regression model does not predict the performance of this dataset particularly well. The linear regression model is given in Equation 1.

$$\begin{aligned} log(\hat{y}_{price}) = \beta_0 &+ \beta_1 \log(X_{bedrooms}) + \beta_2 \log(X_{dscription\_word\_count}) \\ &+ \beta_3 \log(X_{\text{neighborhood\_overview\_word\_count}}) + \beta_4 \log(X_{accommdates}) \\ &+ \beta_5 log(X_{longitude}) - \beta_6 X_{neighborhood\_price\_qity\_low} \\ &+ \beta_7 X_{neighborhood\_BYRON\ SHIRE\ COUNCIL} + \beta_8 X_{amenities\_word\_count} \\ &+ \beta_9 X_{property\_type\_Entrie\ home} - \beta_{10} X_{number\_of\_reviews} + \cdots \\ &+ \beta_{15} \log(X_{beds}) \end{aligned}$$

And then, we visualize the 15 coefficients for our models (Figure 13), which shows the 'bedrooms' have the greatest influence on predicting the price, which validates our results when we do exploratory data analysis in the front, it was followed by 'description_word_count' and 'neighborhood_overview_word_count', this is also consistent with our results for feature engineering. Also, we can see that there are some variables whose beta coefficients are close to or equal to zero, indicating that these variables have only a minimal effect on the price.

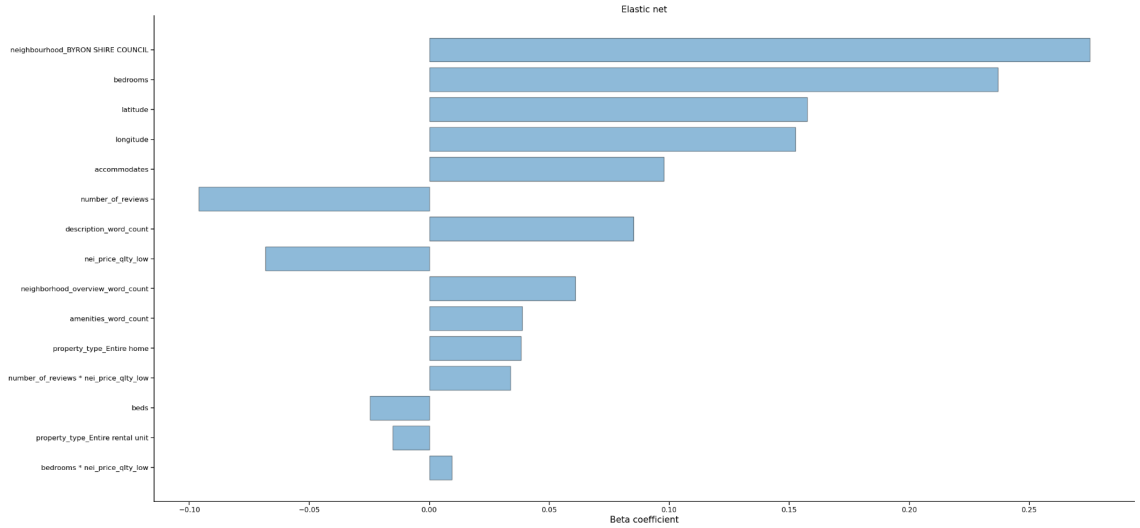

Figure 13

5.3 Model 2: Single Regression Tree Model (Decision Tree Model)

The second model we decided to use is the Decision Tree model because decision trees represent straightforward and easily interpretable models. They are capable of deriving lucid prediction rules that can be effectively communicated to non-specialists. Furthermore, decision trees possess the ability to handle categorical predictors seamlessly without the necessity of generating dummy variables. Their versatility extends to the approximation of intricate nonlinear relationships, encompassing various forms of interactions. Additionally, decision trees serve as the fundamental cornerstone of influential predictive methods, such as random forests and boosted trees, providing a robust framework for the sophisticated models mentioned above (scikit-learn, 2023). Hence, we selected the decision tree model as our secondary modeling approach.

Initially, we constrained each leaf of the decision tree to contain no fewer than 5 samples. Subsequently, we utilized the training data X and y to compute the cost complexity pruning path of the decision tree model, resulting in the extraction of ccp_alphas and impurities along the pruning trajectory. Here, ccp_alphas represents a collection of potential values for the cost complexity parameter (ccp), while impurities denote the corresponding cumulative impurity values. Following this, we employed GridSearchCV for cross-validation, thereby facilitating the exploration of the optimal configuration for hyperparameters. Finally, through the model training process, we successfully obtained the optimal decision tree model, as illustrated in Figure 14 below.

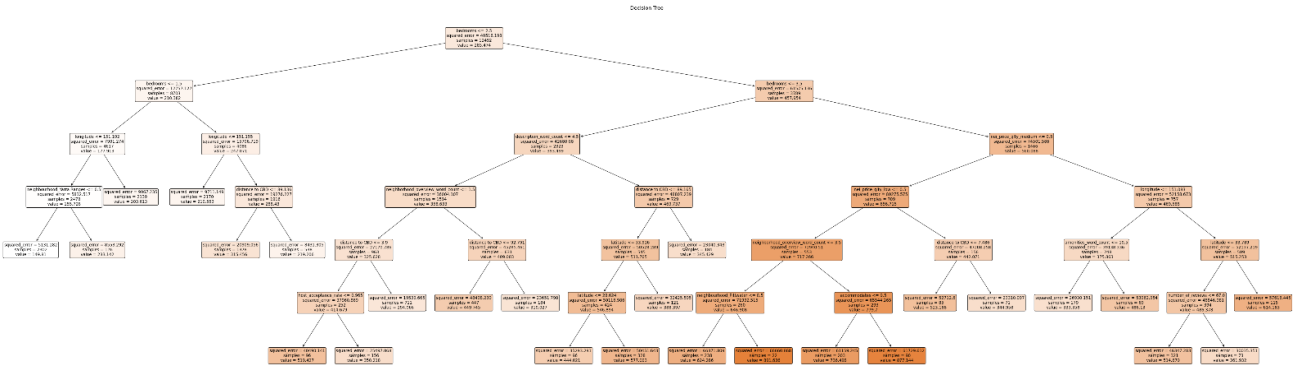

Figure 14

We try to use the best decision tree model to fit train set data and make a forecast using the validation set, and the result of RMSE is 142.9636. Meanwhile, the Root Mean Square Error of the whole training set (combining the train set and the validation set again) is 137.6322. It can be seen that the prediction results are improved compared to the first model, but still not ideal. This is because this decision tree model has a large depth. This will lead to overfitting the data since a small number of examples in each region leads to high variance for estimating the constants.

5.4 Model 3: Advanced Model (XGBoost)

XGBoost is our third selected model, which intends to leverage the model for gradient calculations, iterating through multiple weak learners (Decision Tree models), correcting errors, and ultimately creating a strong learner to enhance overall model performance. Consequently, the model's objective function comprises two key components: the Loss function and Regularization (xgboost.readthedocs.io, 2022).

$$obj(\theta) = L(\theta) + \Omega(\theta)$$

The model employs the gradient descent direction to optimize and adjust prediction results, aiming to minimize the loss function and enhance the model's fitting accuracy. Simultaneously, regularization is incorporated to govern the model's complexity, prevent overfitting, and elevate the overall computational efficiency of the model.

$$\min_{\theta}\left\{\sum_{i=1}^{n} L(y_i, f_{m-1}(x_i) + T(x_i, \theta)) + \alpha|T| + \lambda\sum_{j=1}^{|T|}\gamma_j^2\right\}$$

Furthermore, we obtained the following optimal parameter settings by performing a random search on the training data to find the optimal model's hyperparameter combination within the specified range: learning rate is 0.05, number of trees equals to 2800, and maximum tree depth is 8, the subsample sampling rate is 0.9. Subsequently, we used the validation set to verify the resulting model, and the obtained Root Mean Squared Error (RMSE) was 123.6115. The RMSE of the whole training set (combining the train set and the validation set again) is 52.1624. This result shows that the model can capture the nonlinear characteristics within the data extraordinarily, resulting in high-quality prediction results that are relatively low in residuals from the true value.

Nevertheless, we visualize the importance of the model coefficients (Figure 15). The chart clearly implies that the number of bedrooms, the neighborhood quality level, and the area where the housing is actually located have a prominent impact on housing prices, providing strong support for future price settings and potential profit estimates.

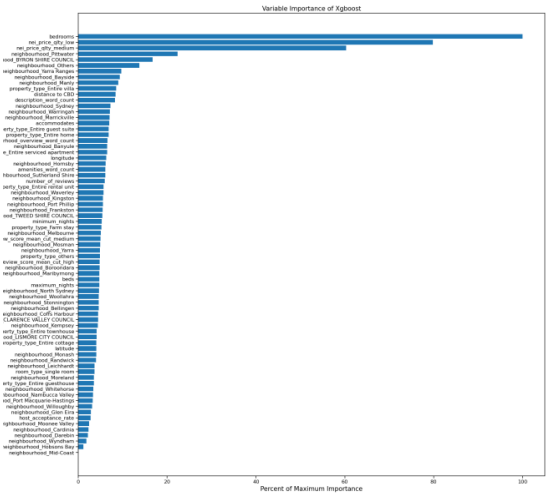

Figure 15

5.5 Model assumptions

The fundamental assumptions underlying the Lasso model are twofold. Firstly, it posits the presence of a linear relationship within the data, indicating a correlation between the dependent variable and the independent variable. Secondly, it assumes that the model coefficients exhibit sparsity, implying the possibility of certain coefficients being reduced to zero, thereby facilitating feature selection. These foundational assumptions empower the Lasso model to accommodate linear data and conduct feature selection, consequently enhancing the model's interpretability and its ability to generalize to new data (Great Learning Team, 2023).

The fundamental premise of the decision tree model is dual in nature. Initially, it assumes the existence of a non-linear association within the data, enabling the partitioning of the data space based on combinations of features, thereby accommodating the modeling of intricate relationships. Secondly, the decision tree presupposes the ability to categorize or predict data through a sequence of binary divisions contingent upon feature values. This partitioning mechanism equips the model to proficiently handle data characterized by multiple categories and continuous attributes, thereby enhancing the model's adaptability and explanatory capacity (Belloc et al., 2022).

The core of the assumptions within the XGBoost model lies in two primary facets. Firstly, it acknowledges the potential ordinal relationship among the encoded integer values of each input variable, allowing the model to comprehend the ordinal interdependencies between variables and more effectively manage the intricacies of processing categorical attributes. Secondly, XGBoost presupposes the likelihood of incomplete data, thus equipping itself with the capability to manage missing values. This adeptness in dealing with missing values empowers the model to adeptly handle the absence of information in real-world datasets, thereby augmenting the model's adaptability and resilience (Nisha & KDnuggets, 2022).

6 Model Validation and Comparisons

6.1 Training and validation scores

Model Name\ RMSE score	Training RMSE	Validation RMSE
Elastic Net	144.9244	145.1885
Decision Tree	137.6322	142.9636
XGBoosting	52.1624	119.2386

6.2 Model comparisons and limitations

From the measurements of our selected models, we obtained the training RMSE (measure the performance on the validation set from the given training dataset) and the validation RMSE (retrain models using the whole training dataset and measure the performance on predicting the data from the test dataset). What should be noticed is that the RMSE score we got is measured using half of the test dataset. Thus it cannot used as the real performance of our models.

When we first took our eye on the regression plot of the predictors in our training dataset, it was obvious that almost half of the numerical predictors have weak or no linear relationships with our response value. Therefore, we were expected to get better results on more complex models compared to the linear regression one. The table above proves our assumption. In the modeling phase, our elastic net model failed to capture the nonlinear characteristics beneath the dataset and performed way worse than other models. Some small improvements appeared when we switched the decision tree model, but it is still not enough. The reason for the mediocre performance on the decision tree may caused by the bias towards the dominant class and the constant value on each assigned node. At last, we obtained our best result on the XGBoosting model.

Among all selected models, the elastic net has the best interpretability and lowest complexity despite having the worst performance. Meanwhile, it should have the lowest computational cost for the trade-off of low prediction accuracy. Compared to the linear model, the decision tree also has high interpretability. By tracing the branches in the decision tree, we could understand how it works and what it takes to provide the result of prediction to us. But deep and complex trees can become less interpretable. When a decision tree has thousands of branches and deep depth, it would be hard to interpret by humans. Finally, XGBoosting should be the ideal model that has reasonable interpretability and complexity. On the one hand, regularization and tree pruning help XGBoosting prevent overfitting and control the complexity. On the other hand, the built-in capability to handle missing values could improve the accuracy of predictions.

Limitations do exist in our selected models. Linear models, like elastic net, cannot present the nonlinearity in the dataset. For a decision tree, it may become hard to interpret when the depth is too deep; it would also miss some important values in modeling. For XGBoosting, the greatest disadvantage is the computational cost. Usually, it would take almost an hour to get the best estimator every time, and our computer cannot do anything when the code is running.

7 Conclusions

In summary, we used various models to predict the Airbnb listing price with the given information for each listing and successfully achieved some predictions using the XGBoosting model.

Furthermore, we have found several factors that have a great influence on the price of listings. For instance, listings with more beds and bedrooms tend to have higher prices, regardless of the location they are. Meanwhile, we discovered that the listings in some areas, like Byron Bay, have higher prices on average than other places. We tried to extract some characteristics from these areas, like the distance to the beach, and apply them to other places to improve the accuracy of the predictions but failed to do so.

Besides the numerical information like the number of beds and bedrooms, the textual information of listings like amenities, descriptions, and neighborhood overview also play a crucial role in determining the listing price. We noticed that some keywords appeared in descriptions and neighborhood overviews related to comfort, luxury, specific locations (Byron Bay, for example), privacy, and security always come with high listing prices.

In order to produce more precise predictions, it is suggested to include more information about the security, the Convenience degree on eating and shopping, and famous Scenic spots nearby.

8 Appendix

8.1 Statement of Contribution

510665339 created the team of Kaggle. All students created Kaggle accounts and joined the team. 520167416 inspected and cleaned the data. 520398467 write the introduction.

All students worked on the exploratory data analysis. Specially, 520167416 did the part of Exploratory Data Analysis, 520302774 did the part of Feature Engineering, 520389591 was responsible for building model 1, 520398467 worked on model 2, and 510665339 worked on model 3. 520389591 was responsible for the final Kaggle submission. 520398467 was responsible for compiling together the final Jupyter notebook for submission.

All students were involved in writing the report. 520167416 collated everyone's EDA and summarized the results in Section 2. 510665339 wrote the methodology section for model 1, 520398467 wrote the methodology section for model 2, and 520389591 wrote the methodology section for model 3. 520302774 wrote Model Validation and Comparisons, Conclusions. All students were responsible for references and Appendix

Briefly described about Generative AI:

We only used Generative AI for making final edits to the report text.

8.2 Meeting Minutes

Meeting Minute 1

Date: 19/10/2023

Present: 510665339,520302774, 520389591, 520398467,520167416

Apologies: None

Agenda

1. Purpose of the assignment
2. Kaggle
3. Data

Meeting notes

- The content and requirements of the assignment were discussed;
- Each team member created a Kaggle account;
- The contents of the dataset were discussed

Actions

- ➢ 510665339(19/10/23): to create a Kaggle team
- ➢ All members (23/10/23): Create Kaggle account and join the team
- ➢ 520167416(24/10/23): To inspect and clean the data
- ➢ 520398467(25/10/23): To write the introduction

Meeting Minute 2

Date: 26/10/2023

Present: 510665339,520302774, 520389591, 520398467,520167416

Apologies: None

Agenda

1. Distribution code
    a. Exploratory Data Analysis
    b. Feature Engineering
    c. Methodology
    d. Kaggle competition

Meeting notes

- Discuss how the data should be further processed;
- what models to choose for prediction;
- discuss the use of different models

Actions

- ➢ 520167416(26/10/23): Doing Exploratory Data Analysis
- ➢ 520302774(28/10/23): Doing Feature Engineering
- ➢ 520389591, 520398467,510665339(30/10/23): Doing three Methodology
- ➢ 520389591, 520398467(3/11/23): optimization model, Kaggle competition

Meeting Minute 3

Date: 4/11/2023

Present: 510665339,520302774, 520389591, 520398467,520167416

Apologies: None

Agenda

1. Distribution report
    a. Data Processing
    b. Exploratory Data Analysis
    c. Feature Engineering
    d. Methodology
    e. Model Validation and Comparisons
    f. Conclusions
    g. References
    h. Appendix

Meeting notes

- Discuss the model results
- Assign each section of the report

Actions

- ➢ 520167416(5/11/23 to 10/11/23): To write the Data Processing, Exploratory Data Analysis
- ➢ 510665339(5/11/23 to 10/11/23): To write the Feature Engineering, Methodology (1)
- ➢ 520398467(5/11/23 to 10/11/23): To write the Methodology (2)
- ➢ 520389591(5/11/23 to 10/11/23): To write the Methodology (3),
- ➢ 520302774(5/11/23 to 10/11/23): To write Model Validation and Comparisons, Conclusions
- ➢ All members (5/11/23 to 10/11/23): References
- ➢ All members (10/11/23): Appendix

8.3 Reference

Belloc, I., Giménez-Nadal, J. I., & Molina, J. A. (2022). The gasoline price and the commuting behavior: Towards sustainable modes of transport (Working Paper 1130). GLO Discussion Paper. https://www.econstor.eu/handle/10419/261321

Bronwyn, A., & Adcock, B. (2022, July 29). Taking stock: How has 10 years of Airbnb changed Australia? The Guardian. https://www.theguardian.com/australia-news/2022/jul/30/taking-stock-how-has-10-years-of-airbnb-changed-australia

CFI Team. (2023). *Elastic Net*. Corporate Finance Institute. https://corporatefinanceinstitute.com/resources/data-science/elastic-net/

DataTechNotes. (n.d.). *ElasticNet Regression Example in Python*. Retrieved November 9, 2023, from https://www.datatechnotes.com/2019/08/elasticnet-regression-example-in-python.html

Elastic Net. (n.d.). Corporate Finance Institute. Retrieved November 9, 2023, from https://corporatefinanceinstitute.com/resources/data-science/elastic-net/

guest_blog. (2018, September 6). Introduction to XGBoost Algorithm in Machine Learning. *Analytics Vidhya*.

https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/

Great Learning Team. (2023, May 30). A Complete understanding of LASSO Regression. Great Learning Blog: Free Resources What Matters to Shape Your Career! https://www.mygreatlearning.com/blog/understanding-of-lasso-regression/

IBM.com. (2023). *What is a Decision Tree*. IBM.Com. https://www.ibm.com/topics/decision-trees

Khandelwal, R. (2023, July 17). The State of Short-Term Rentals in Australia: Insights and Opportunities. PriceLabs. https://hello.pricelabs.co/the-state-of-short-term-rentals-in-australia/

Nisha, A., & KDnuggets. (2022, August 4). What are the Assumptions of XGBoost? KDnuggets. https://www.kdnuggets.com/what-are-the-assumptions-of-xgboost.html

IBM.com. (2023). *What is a Decision Tree*. IBM.Com. https://www.ibm.com/topics/decision-trees

Rahul Awati. (2022, August). *What is latitude and longitude?* WhatIs.Com. https://www.techtarget.com/whatis/definition/latitude-and-longitude

REI, R. E. I. (2023). Pros and Cons of Renting Out Your Investment as an Airbnb [Blog]. Realestateinvestar. https://blog.realestateinvestar.com.au/pros-and-cons-of-renting-out-your-investment-as-an-airbnb

scikit-learn. (2023). 1.10. Decision Trees. Scikit-Learn. https://scikit-learn/stable/modules/tree.html

xgboost.readthedocs.io. (2022). *Introduction to Boosted Trees—Xgboost 2.0.2 documentation*. https://xgboost.readthedocs.io/en/stable/tutorials/model.html