Airbnb Pricing Analysis with Supervised Learning Models

Zhiwei (Alina) Gu, Xiyi Lin, Yi (Grace) Xie, Judy (Zhiyi) Zhu

Table of Contents

Introduction

Airbnb, Inc. is an American San Francisco-based company, which was founded in 2008. Airbnb, originally known as AirBedandBreakfast.com, operates an online marketplace for short-term and long-term home stays and experiences. This platform has not only revolutionized travel accommodations but has also shaped the way people connect and experience new destinations.

In the dynamic landscape of the rental market, pricing plays a pivotal role in the success of Airbnb listings. Multiple factors need to be considered in this process, including location, size, rating, etc. The complexity of this process poses significant challenges for hosts to decide the price for their listings.

In this research, we employed multiple supervised learning methods to predict the prices of Airbnb accommodations, such as Linear Regression, Neural Network, Tree, and XGBoost. By harnessing the power of data-driven insights, we aim to develop a nuanced understanding of the factors influencing pricing strategies within the Airbnb ecosystem and therefore offer more guidance to Airbnb hosts and users.

Data

1. Dataset Overview

This dataset from Kaggle describes the latest listing activity in New York City, New York as of January 5th, 2024. The raw dataset comprised 22 columns and 20,758 rows, with each row corresponding to an individual Airbnb listing. A thorough inspection revealed no missing values, indicating a well-maintained dataset, albeit with potential for optimization to better suit our analysis goals.

2. Data Cleaning and Preprocessing

Last Review Date

One of the first steps in our preprocessing was to filter listings based on the last_review date. Recognizing the dynamic nature of the Airbnb market, we focused on listings with a last_review date post 1/1/2019. This criterion was applied to exclude potentially inactive listings from our analysis, ensuring that our predictive model would be trained on data representative of current market conditions.

Neighbourhood Grouping

The dataset featured extensive geographical data, including a detailed breakdown into neighbourhoods. However, to avoid the complexity associated with a multitude of categories and to facilitate a more streamlined analysis, we opted to utilize the neighbourhood_group attribute. This decision was also informed by the reduction of geographical details, as we dropped latitude and longitude data, considering the neighbourhood group as a sufficiently precise yet manageable geographical marker for our analysis.

Rating Normalization

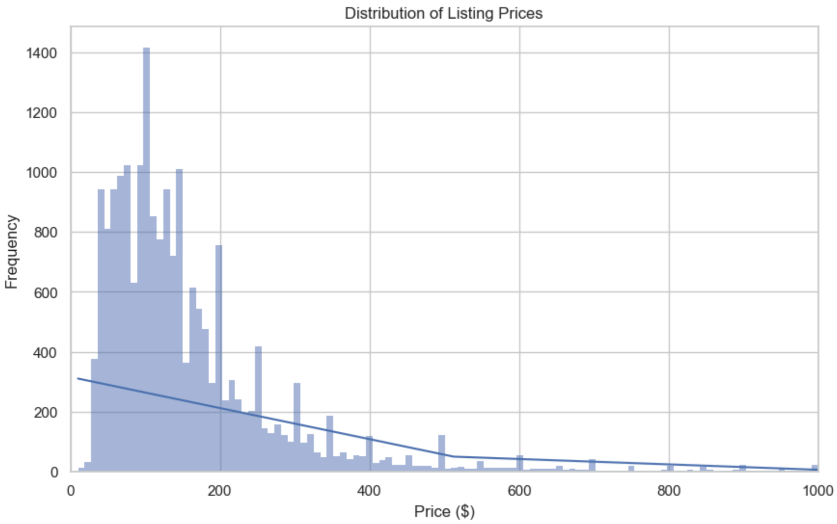
The rating column presented varied classifications, including listings without ratings and newly listed properties. We treated listings marked as “No rating” by assigning them a value of 0, reflecting the absence of qualitative feedback. Conversely, listings labeled as “New” were excluded due to the lack of substantial data to assess their potential market performance accurately.

Handling Specific Attributes

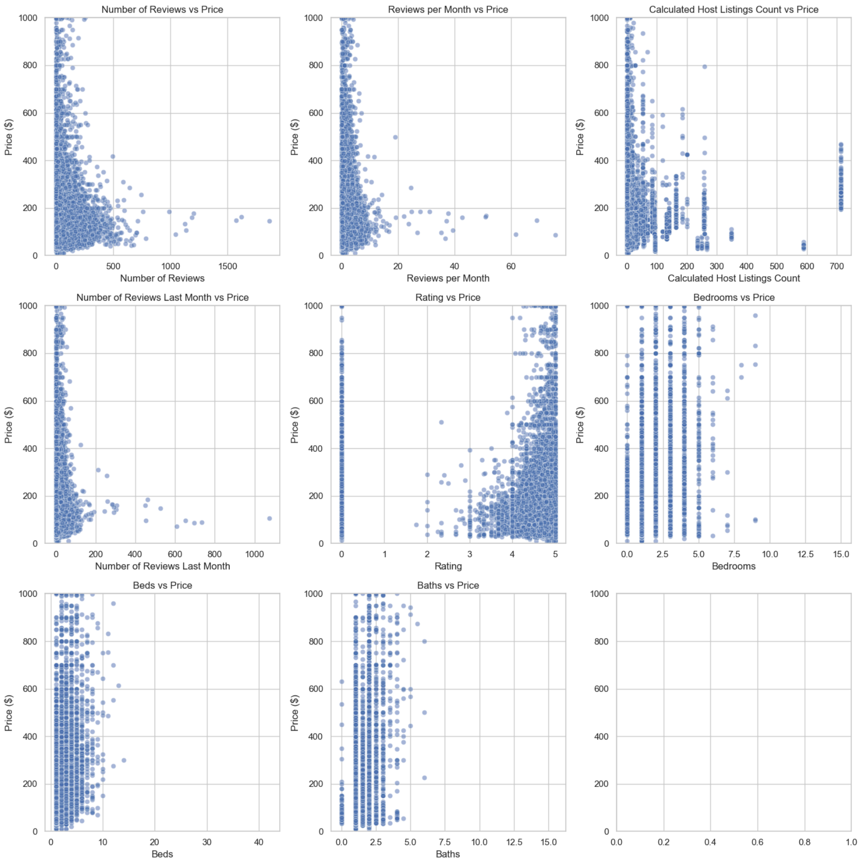
For attributes such as bedrooms and baths, specific considerations were made to align them with a continuous variable framework. Listings identified as “Studio” were assigned a bedroom count of 0, acknowledging the unique nature of studio apartments. Similarly, “No Specified” entries for baths were replaced with 0, ensuring consistency in treating these attributes as continuous variables, which is crucial for the subsequent analytical processes.

3. Exploratory Data Analysis

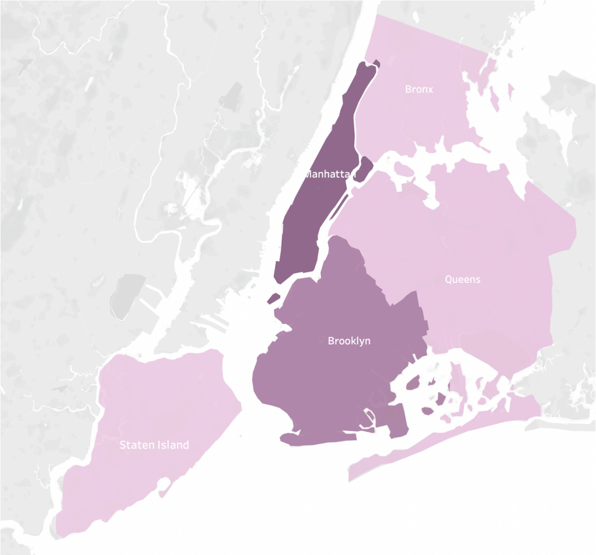
Based on the histogram of distribution of prices, we could notice that the data is highly right skewed, with most of the data between 0 - 200, every few above 600. Such distribution supports our choice of taking log over price in order to smooth the trend and avoid the impact of outliers.



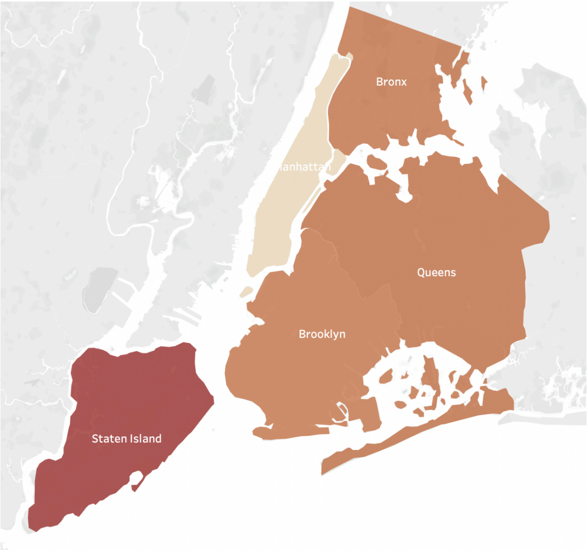
We also visualize the relationship between each continuous variable and price. Most relations have points clustered around low values and correspond to a wide range of prices. For example, in rating, we have a few zeros and most values near 4 and 5. For those listings with ratings 4 to 5, they have a wide range of prices. This represents the relationship between these variables and price are nonlinear.



We also visualize the relationship of price by district and rating by district. Darker color means higher price / higher rating. We could notice that Manhattan has the highest average price but the lowest average rating. Staten Island has one of the lowest prices but the highest average rating.



Average Price By District



Average Rating by District

4. Feature Engineering

In the feature engineering phase of our analysis, we approached the refinement of our dataset with the objective of optimizing the predictive power of our model, specifically targeting the prediction of listing prices. Here, we delineate the strategies employed to engineer meaningful features from the raw data.

Minimum Nights Feature Transformation

The `*minimum_nights*` attribute displayed a broad spectrum of values, inclusive of extreme outliers which could potentially skew the model's performance. To mitigate the impact of these outliers and distill more actionable insights, we categorized the values into three distinct bins: "less than 7" nights, "7 to 30" nights, and "more than 30" nights. This binning strategy was predicated on the notion that the duration of stay could significantly influence pricing dynamics, and the majority of listings were observed to fall within the "less than 7" nights category. This categorization aids in simplifying the model's understanding of how minimum stay requirements might affect listing prices.

Last Review Date Transformation

The `*last_review*` feature was initially presented in a date format. Recognizing the importance of recency in guest reviews for prospective tenants, we transformed this attribute into the `review_days` feature by calculating the number of days between the data collection date (1/5/2024) and the `last_review` date. This transformation provides a quantifiable measure of recency, thereby enabling our model to account for the potential impact of recent feedback on listing prices.

License Feature Simplification

The `*license*` column contained three types of values: "No License", "Exempt", and various specific license numbers. Given the extensive range of license numbers, we streamlined this feature by consolidating any specific license number into a single "With License" category. This simplification was guided by the hypothesis that the presence of a license might distinguish professional listings from personal ones, which could in turn reflect on the pricing structure.

Response Variable Transformation

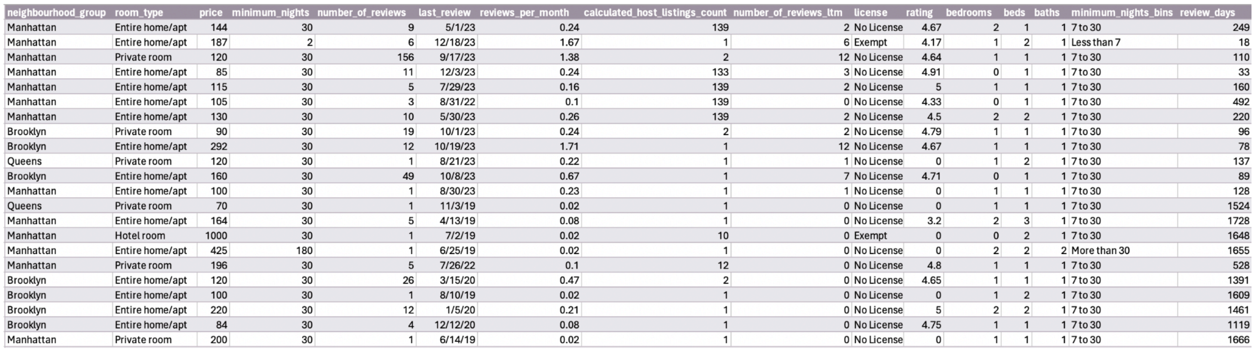
A pivotal step in our preprocessing was the transformation of the price variable. Given the skewed distribution typically observed in real estate pricing data, we applied a logarithmic transformation to normalize this distribution. This approach is a widely acknowledged practice in statistical modeling, particularly useful for stabilizing variance and linearizing relationships between the target variable and predictors. Such normalization is instrumental in enhancing model performance, especially in the context of predicting variables like price, which can exhibit significant outliers.

Data Preparation and Modeling

Upon completing the aforementioned transformations and analysis in Python, we migrated the refined dataset into R for the subsequent modeling phase. Our model aims to predict listing prices (`price`), utilizing a set of predictors including:

`neighbourhood_group`,`room_type`,`number_of_reviews`,`reviews_per_month`, `calculated_host_listings_count`,`number_of_reviews_ltm`, `license`, `rating`, `bedrooms`, `beds`, `baths`, `minimum_nights_bins`, and `review_days`

This comprehensive suite of features was meticulously chosen to encapsulate both the intrinsic characteristics of the listings and the external factors likely to influence their market prices, thereby ensuring a holistic approach to price prediction.



neighbourhood_group	room_type	price	minimum_nights	number_of_reviews	last_review	reviews_per_month	calculated_host_listings_count	number_of_reviews_ltm	license	rating	bedrooms	beds	baths	minimum_nights_bins	review_days
Manhattan	Entire home/apt	144	30	9	5/1/23	0.24	139	2	No License	4.67	2	1	1	7 to 30	249
Manhattan	Entire home/apt	187	2	6	12/18/23	1.67	1	6	Exempt	4.17	1	2	1	Less than 7	18
Manhattan	Private room	120	30	156	9/17/23	1.38	2	12	No License	4.64	1	1	1	7 to 30	110
Manhattan	Entire home/apt	85	30	11	12/3/23	0.24	133	3	No License	4.91	0	1	1	7 to 30	33
Manhattan	Entire home/apt	115	30	5	7/29/23	0.16	139	2	No License	5	1	1	1	7 to 30	160
Manhattan	Entire home/apt	105	30	3	8/31/22	0.1	139	0	No License	4.33	0	1	1	7 to 30	492
Manhattan	Entire home/apt	130	30	10	5/30/23	0.26	139	2	No License	4.5	2	2	1	7 to 30	220
Brooklyn	Private room	90	30	19	10/1/23	0.24	2	2	No License	4.79	1	1	1	7 to 30	96
Brooklyn	Entire home/apt	292	30	12	10/19/23	1.71	1	12	No License	4.67	1	1	1	7 to 30	78
Queens	Private room	120	30	1	8/21/23	0.22	1	1	No License	0	1	2	1	7 to 30	137
Brooklyn	Entire home/apt	160	30	49	10/8/23	0.67	1	7	No License	4.71	0	1	1	7 to 30	89
Manhattan	Entire home/apt	100	30	1	8/30/23	0.23	1	1	No License	0	1	1	1	7 to 30	128
Queens	Private room	70	30	1	11/3/19	0.02	1	0	No License	0	1	1	1	7 to 30	1524
Manhattan	Entire home/apt	164	30	5	4/13/19	0.08	1	0	No License	3.2	2	3	1	7 to 30	1728
Manhattan	Hotel room	1000	30	1	7/2/19	0.02	10	0	Exempt	0	0	2	1	7 to 30	1648
Manhattan	Entire home/apt	425	180	1	6/25/19	0.02	1	0	No License	0	2	2	2	More than 30	1655
Manhattan	Private room	196	30	5	7/26/22	0.1	12	0	No License	4.8	1	1	1	7 to 30	528
Brooklyn	Entire home/apt	120	30	26	3/15/20	0.47	2	0	No License	4.65	1	1	1	7 to 30	1391
Brooklyn	Entire home/apt	100	30	1	8/10/19	0.02	1	0	No License	0	1	2	1	7 to 30	1609
Brooklyn	Entire home/apt	220	30	12	1/5/20	0.21	1	0	No License	5	2	2	1	7 to 30	1461
Brooklyn	Entire home/apt	84	30	4	12/12/20	0.08	1	0	No License	4.75	1	1	1	7 to 30	1119
Manhattan	Private room	200	30	1	6/14/19	0.02	1	0	No License	0	1	1	1	7 to 30	1666

Modeling

1. Linear Regression

In our foundational analysis, we initiated our investigation by constructing a baseline linear regression model, wherein we opted to model the logarithm of the price. The analysis of the model's coefficients revealed that several predictors—namely, "neighborhood", "room type", "minimum nights", "last review", "calculated host listings count", "license condition", "rating", "bedrooms", "beds", "baths", and "minimum night"—exert a statistically significant impact on the dependent variable. This finding underscores the importance of these features in influencing the log-transformed price of listings, thereby validating their inclusion in the model.

Subsequently, we engaged in a meticulous process of model refinement through 10-fold cross-validation, aimed at identifying the optimal lambda value for the Lasso regression model. This step is pivotal as it ensures the model's robustness by preventing overfitting.

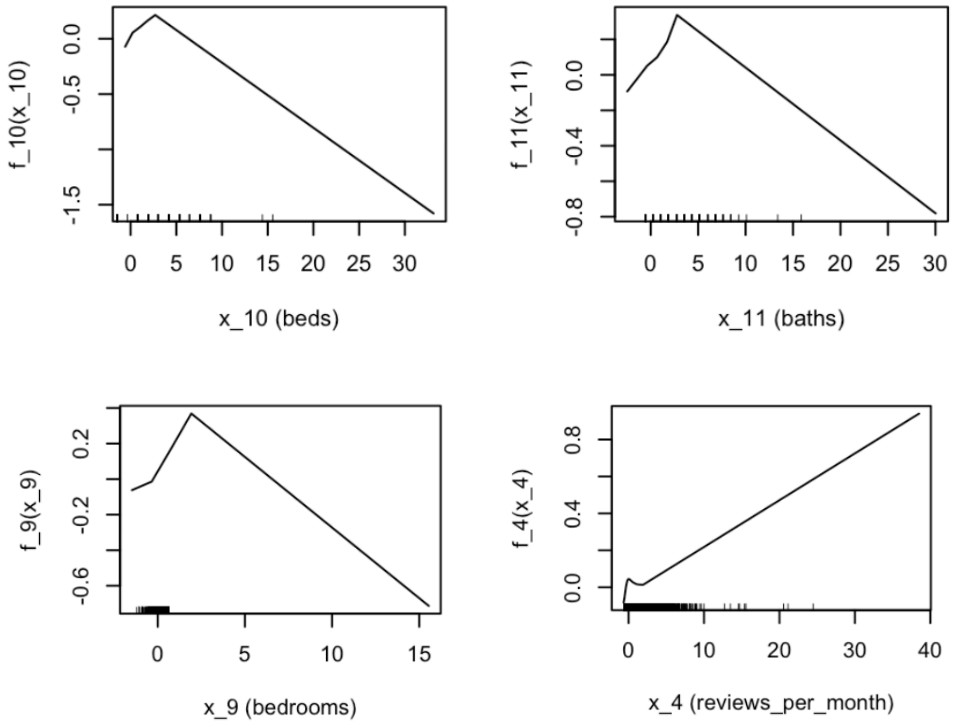
The culmination of this process yielded an average R-squared value of 0.44, indicating that approximately 44% of the variance in the logarithm of the price is accounted for by the model. The mean Root Mean Square Error (RMSE) of the validation model was determined to be 1568.54.

2. Neural Network

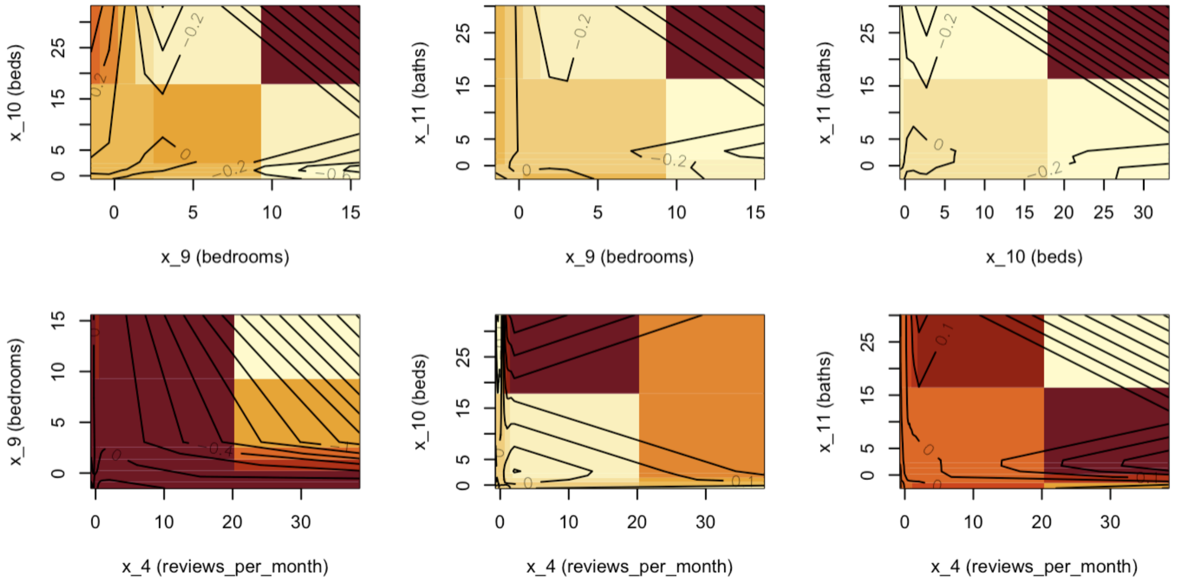
For the Neural Network model, we treated neighbourhood_group, room_type, license, and minimum_nights_bins as categorical variables through factors, and then standardized remaining continuous variables.

We performed 5-fold cross validation with 3 replicates to decide the optimal number of hidden notes and decay. When size is 15 and decay is 0.1, the model reaches the highest cross validation R squared 0.560. RMSE is 0.472. The residuals are mostly around 0 based on the residual plot.

Based on the ALE plots, by looking at the range of effect, the strong predictors are beds, baths, bedrooms, and reviews_per_month. For beds, baths, and bedrooms the effect first increases and then decreases. For reviews_per_month, the effect first increases fast and then slows down. The effects are relatively non-linear for the region with most of the values.



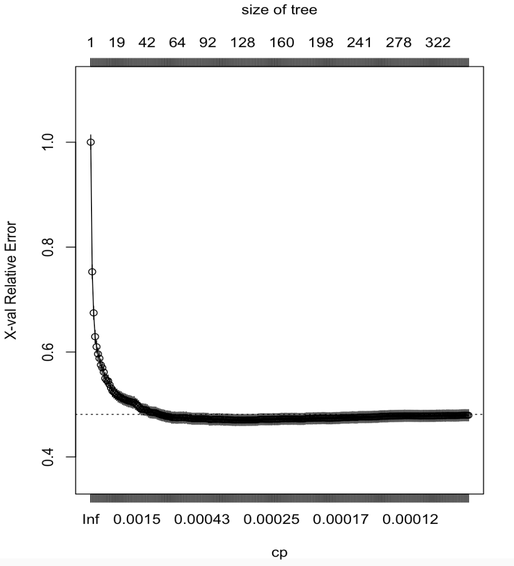
For these predictors, we created 2nd-order interaction ALE plots. When the bedrooms variable is above 9 and the beds variable is below 18, the interaction is relatively significant and shows a reduction in the overall effect on price. When the bedrooms variable is between 2 and 9 and the beds variable is above 18, the interaction is also significant and negative. Similar pattern exists between beds, bedrooms, and baths. For reviews_per_month, the interactions are more significant when baths are high, bedrooms are high, or baths are low.



3. CART

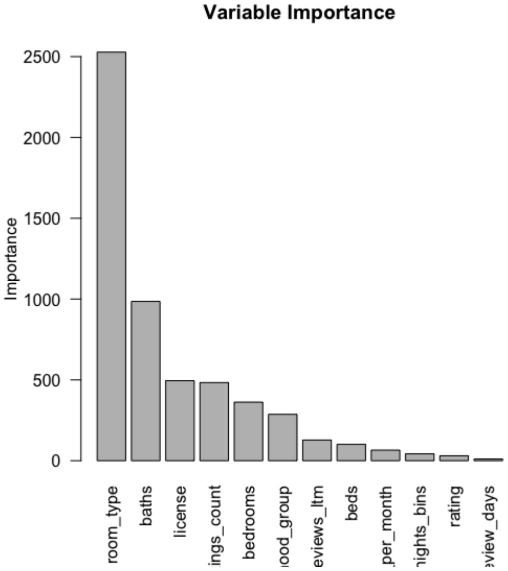
For the tree model, we factorized neighbourhood_group, room_type, license, and minimum_nights_bins to treat them as categorical data , and then standardized remaining continuous variables.

We used a 10 fold cross validation tree model with a minimum bucket of 20 to tree the model. We pruned the tree using the one standard error rule. We then checked the model performance and importance of variables.



Based on the result, the trained cross validation error is 0.456. This shows that on average, the model explains 54.3% of the data. Looking at the cross validation result, the overfitting issue with the model should be neglectable.

By looking at the importance metrics, we identified the most important factors influencing the housing price. Room type, number of baths, whether the landlord has a license, number of host listings, number of bedrooms, residing neighborhood group, number of reviews are the most important factors based on the model.



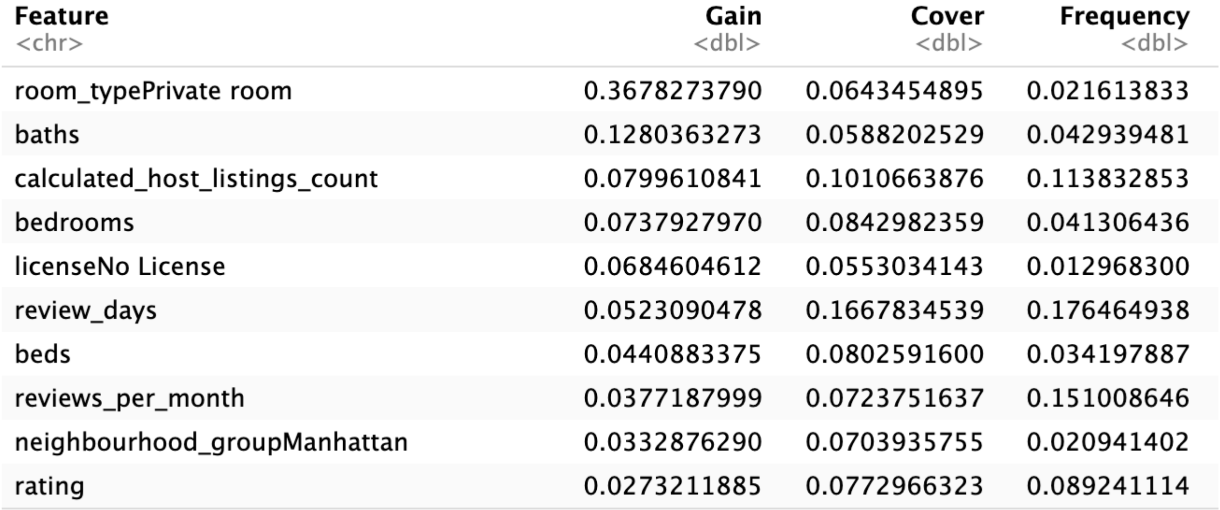
4. XGBoost

The modeling process with XGBoost for predicting Airbnb listing prices was strategically designed to utilize the strengths of the algorithm, notably its capability to handle unstandardized numeric data efficiently. This was particularly relevant for our dataset, where a log-transformation was applied to the price variable to normalize its distribution—a crucial preprocessing step before modeling. The transformation of the cleaned data into an XGBoost DMatrix object facilitated a more expedited training process by optimizing the data structure for XGBoost's algorithm requirements.

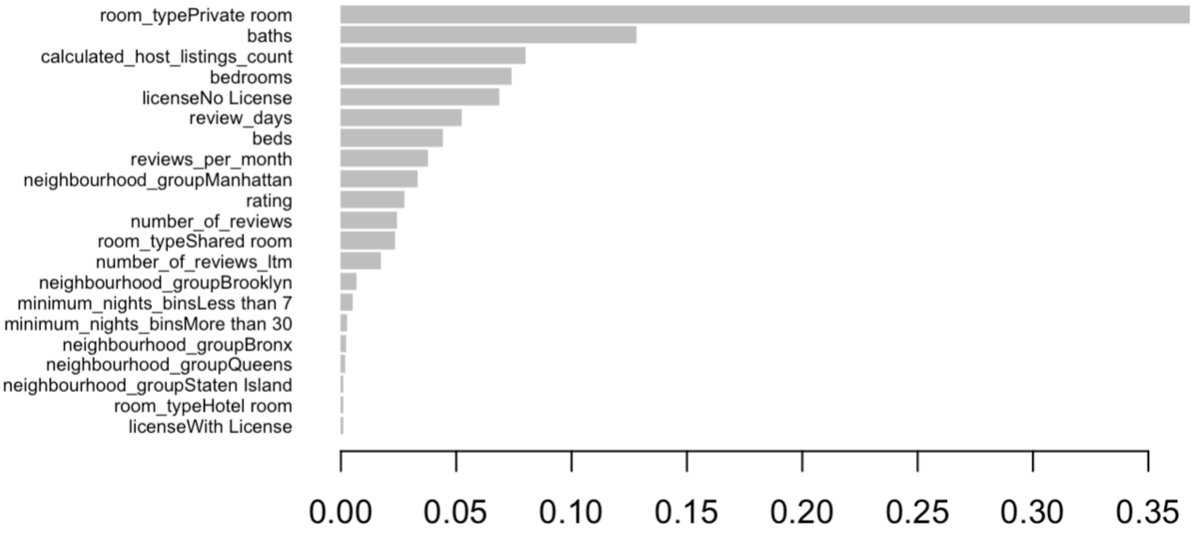
Given the regression nature of our task, we defined our model's objective as "*reg:squarederror*". This choice aligned with our goal to minimize prediction errors, for which we employed evaluation metrics such as the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE). These metrics served as quantitative benchmarks to assess the model's performance throughout the training and hyperparameter tuning phases.

Hyperparameter tuning emerged as a pivotal stage in our modeling process. XGBoost's inherent support for L1 and L2 regularization—akin to Lasso and Ridge regression, respectively—plays a vital role in mitigating overfitting. These regularization mechanisms function by imposing penalties on model complexity, thus favoring simpler, more generalizable models. Beyond regularization, we explored a range of parameters, including `*max_depth*`, `*gamma*`, `*lambda*`, and `*alpha*`, to identify the optimal configuration. Our methodical grid search approach yielded significant improvements, with RMSE and MAE on the test set dropping from 0.464 to 0.356 and from 0.343 to 0.267, respectively. This enhancement in predictive accuracy underscored the efficacy of our tuning efforts. Additionally, we achieved an optimized R-squared value of 75.5%, indicating a strong fit between our model's predictions and the actual data.

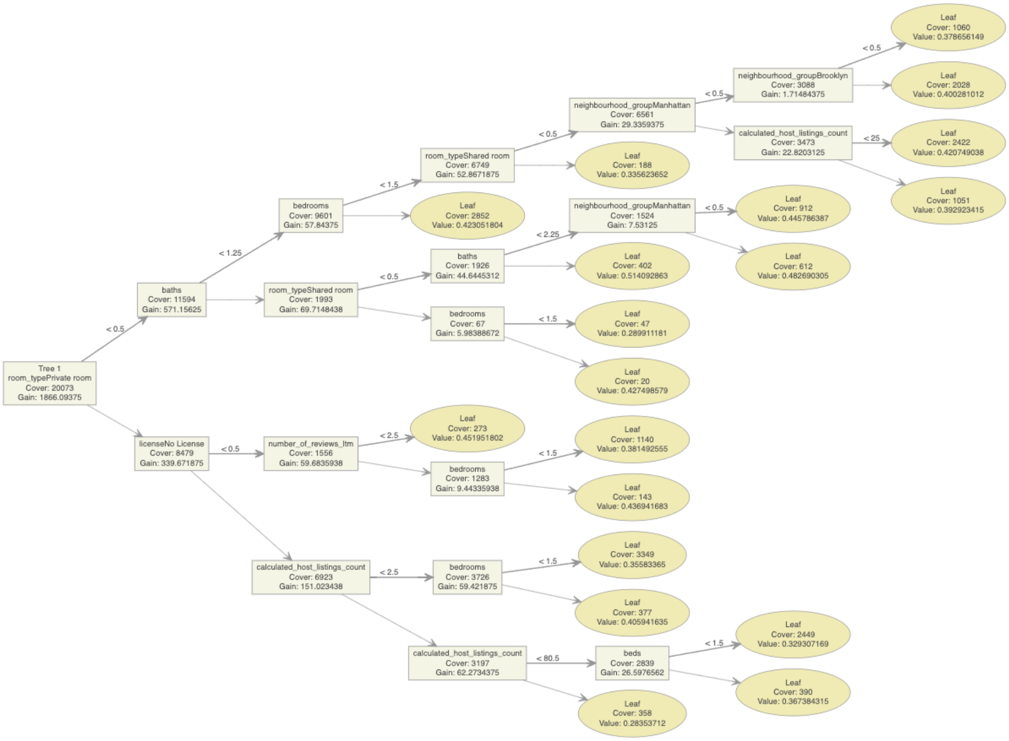
For model interpretation, we leveraged Feature Importance plots to delineate the relative significance of different predictors. The "gain" metric, indicative of a feature's contribution to model improvement, provided profound insights. Notably, `*room_typePrivate room*` emerged as a paramount predictor, reflecting its critical role in determining listing prices. The visualization of feature importance not only highlighted the dominance of certain variables like `*baths*` and `*calculated_host_listings_count*` but also facilitated a comprehensive understanding of their impact on the model's decisions.



Feature	Gain	Cover	Frequency
<chr>	<dbl>	<dbl>	<dbl>
room_typePrivate room	0.3678273790	0.0643454895	0.021613833
baths	0.1280363273	0.0588202529	0.042939481
calculated_host_listings_count	0.0799610841	0.1010663876	0.113832853
bedrooms	0.0737927970	0.0842982359	0.041306436
licenseNo License	0.0684604612	0.0553034143	0.012968300
review_days	0.0523090478	0.1667834539	0.176464938
beds	0.0440883375	0.0802591600	0.034197887
reviews_per_month	0.0377187999	0.0723751637	0.151008646
neighbourhood_groupManhattan	0.0332876290	0.0703935755	0.020941402
rating	0.0273211885	0.0772966323	0.089241114



Further enriching our interpretive analysis, XGBoost's capability to visualize decision trees offered a granular view of the model's reasoning. The visualization exemplified how initial splits were predominantly based on the `*room_typePrivate room*` variable, reaffirming its predictive power. By examining the decision paths and the conditions leading to specific predictions, we gained nuanced insights into the model's operational dynamics. Each decision rule, represented by paths from the root to the leaf nodes, encapsulates a segment of the model's predictive logic, which, when aggregated across the ensemble of trees, constructs the full predictive model.


Tree 1 room_typePrivate room Cover: 20073 Gain: 1866.09375
< 0.5
baths Cover: 11594 Gain: 571.15625
< 1.25
bedrooms Cover: 9601 Gain: 57.84375
< 1.5
room_typeShared room Cover: 6749 Gain: 52.8671875
< 0.5
neighbourhood_groupManhattan Cover: 6561 Gain: 29.3359375
< 0.5
neighbourhood_groupBrooklyn Cover: 3088 Gain: 1.71484375
< 0.5
Leaf Cover: 1060 Value: 0.378656149
Leaf Cover: 2028 Value: 0.400281012
calculated_host_listings_count Cover: 3473 Gain: 22.8203125
< 25
Leaf Cover: 2422 Value: 0.420749038
Leaf Cover: 1051 Value: 0.392923415
Leaf Cover: 188 Value: 0.335623652
Leaf Cover: 2852 Value: 0.423051804
room_typeShared room Cover: 1993 Gain: 69.7148438
< 0.5
baths Cover: 1926 Gain: 44.6445312
< 2.25
neighbourhood_groupManhattan Cover: 1524 Gain: 7.53125
< 0.5
Leaf Cover: 912 Value: 0.445786387
Leaf Cover: 612 Value: 0.482690305
Leaf Cover: 402 Value: 0.514092863
bedrooms Cover: 67 Gain: 5.98388672
< 1.5
Leaf Cover: 47 Value: 0.289911181
Leaf Cover: 20 Value: 0.427498579
licenseNo License Cover: 8479 Gain: 339.671875
< 0.5
number_of_reviews_ltm Cover: 1556 Gain: 59.6835938
< 2.5
Leaf Cover: 273 Value: 0.451951802
bedrooms Cover: 1283 Gain: 9.44335938
< 1.5
Leaf Cover: 1140 Value: 0.381492555
Leaf Cover: 143 Value: 0.436941683
calculated_host_listings_count Cover: 6923 Gain: 151.023438
< 2.5
bedrooms Cover: 3726 Gain: 59.421875
< 1.5
Leaf Cover: 3349 Value: 0.35583365
Leaf Cover: 377 Value: 0.405941635
calculated_host_listings_count Cover: 3197 Gain: 62.2734375
< 80.5
beds Cover: 2839 Gain: 26.5976562
< 1.5
Leaf Cover: 2449 Value: 0.329307169
Leaf Cover: 390 Value: 0.367384315
Leaf Cover: 358 Value: 0.28353712

In conclusion, the detailed analysis and optimization of our XGBoost model not only enhanced its predictive accuracy but also provided valuable insights into the factors influencing Airbnb listing prices. Through rigorous preprocessing, strategic model configuration, and comprehensive interpretation of results, we have developed a robust predictive framework that combines high CV R^2 with meaningful interpretability.

5. Model Comparison

Model Performance:

	RMSE (CV)	R^2 (CV)
Linear Regression	0.533	0.440
Neural Network	0.472	0.560
CART	0.481	0.543
XGBoost	0.356	0.755

Variable Importance:

	Top 5 Significant Features
Linear Regression	Beds, Bedrooms, License, Private room type, Manhattan
Neural Network	Beds, Baths, Bedrooms, Reviews per month, Number of reviews
CART	Room type, Baths, License, Number of host listings, Bedrooms
XGBoost	Private room type, Baths, Number of host listings, Bedrooms, No License

Conclusion

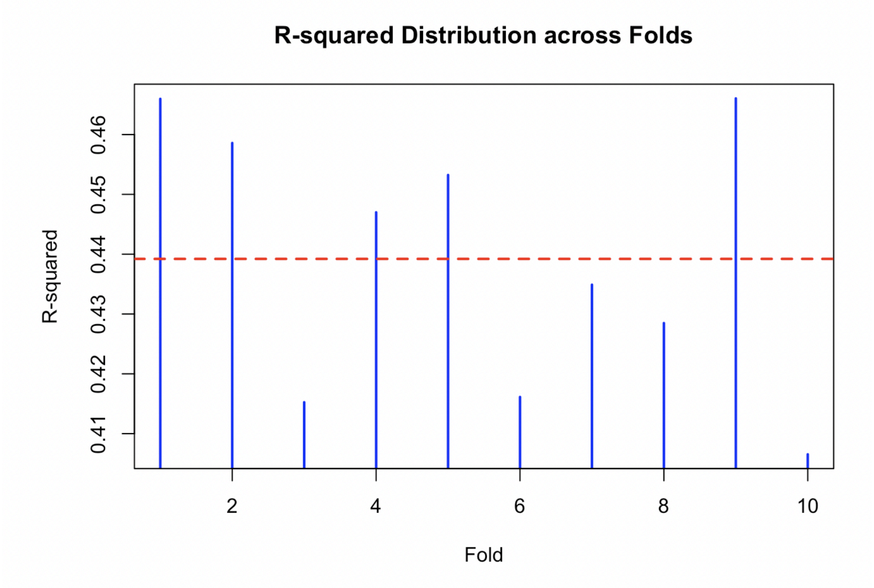
Based on the R^2 (CV) of linear regression, neural network, CART, and XGBoost, the best model is XGBoost. It has R^2 of 0.755 and RMSE of 0.356. Furthermore, a comprehensive examination of important features across four distinct models reveals a consistent pattern: private room type, number of bedrooms, and number of baths have influence on Airbnb prices. These factors significantly shape the overall staying experience on Airbnb, thereby having a substantial impact on pricing. It is justifiable for larger accommodations to command higher prices, given the enhanced amenities and space they offer.

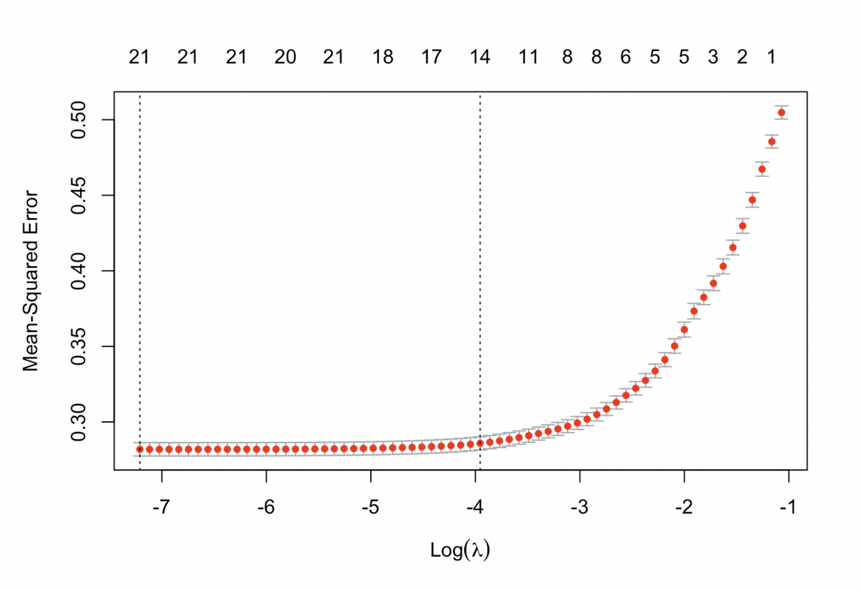
Discussion

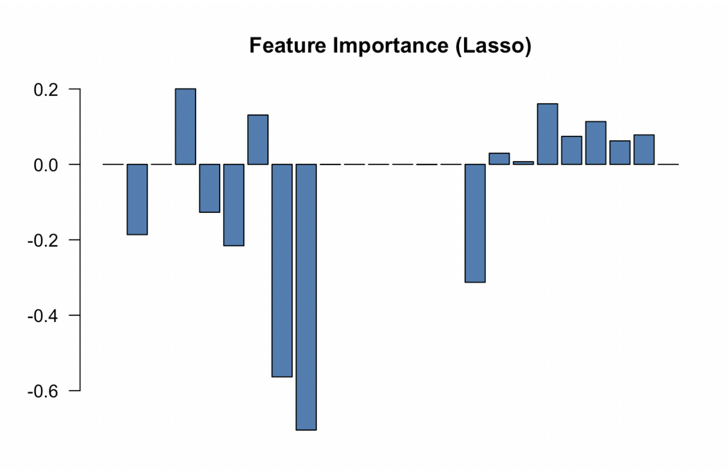
If we can add more detailed neighborhood information, spatial analysis could reveal patterns related to geography. By moving beyond broad district groups and incorporating specific neighborhood data, the model could include more nuanced and layered analysis. This enhancement would provide a granular understanding of how different areas impact Airbnb pricing dynamics.

Also, based on the neighborhood group, we can research more about socioeconomic status and how it might affect our analysis. By looking into the socioeconomic characteristics of each neighborhood, we can gain insights into how these variables contribute to further variations in pricing.

Appendix


R-squared Distribution across Folds
R-squared
0.41
0.42
0.43
0.44
0.45
0.46
2
4
6
8
10
Fold


21 21 21 20 21 18 17 14 11 8 8 6 5 5 3 2 1
Mean-Squared Error
0.30
0.35
0.40
0.45
0.50
-7
-6
-5
-4
-3
-2
-1
Log(λ)


Feature Importance (Lasso)
0.2
0.0
-0.2
-0.4
-0.6