National College of Ireland

Optimizing Airbnb Rental Prices: A Machine Learning Approach Using Listing Attributes, Review Ratings and Amenities

MSc Research Project
Data Analytics

Muhammad Wasit Shahbaz
Student ID: X23257741

School of Computing
National College of Ireland

Supervisor: Jorge Basilio

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Muhammad Wasit Shahbaz

Student ID: X23257741

Programme: MSc Data Analytics **Year:** 2024

Module: Research Project

Supervisor: Jorge Basilio

Submission Due Date: 11th August 2025

Project Title: Optimizing Airbnb Rental Prices: A Machine Learning Approach Using Listing Attributes, Review Ratings and Amenities

Word Count: 10659 **Page Count:** 24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Muhammad Wasit

Date: 10th August 2025

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Optimizing Airbnb Rental Prices: A Machine Learning Approach Using Listing Attributes, Review Ratings and Amenities

Muhammad Wasit Shahbaz
X23257741

Abstract

This study demonstrates how Airbnb listing price can be optimized with machine learning approach by using listing features like review scores, generic features (bed, bedroom, accommodates) and amenities. To conduct the study various datasets from major cities have been used (Amsterdam, Berlin, Athens, and Los Angeles). Including advance ensemble models like LightGBM, XGBoost, and CatBoost, seven various models are applied on the dataset. Out of these models LightGBM performed the best with an R2 score of 0.782. The research not only predicts the price but also demonstrates the labelling of the lists as underpriced, overpriced and fairly priced, which provides an insight to optimize the price of the listings. Through feature importance and SHAP analysis, it was revealed that factors such as the number of bedrooms and number of bathrooms, review scores of listings, and amenities like the availability of a dishwasher or an elevator can also significantly influence pricing. This study offers the practical tools needed for hosts to improve revenue while also ensuring fairness. Limitations include the lack of seasonal dynamics and textual reviews. This project contributes a scalable data driven pricing model to enhance and improve the decision making in the peer-to-peer rental market.

1 Introduction

1.1 Background

Airbnb has emerged as one of the leading global marketplaces for short-term accommodations due to digital transformation of the hospitality industry. Airbnb has completely transformed people's access to hospitality services as compared to traditional hotels by allowing individuals to rent out private properties. Although this approach offers more flexibility and greater market participation, it also presents numerous challenges, particularly issues related to price variability. Unlike standard hotel prices, Airbnb allows hosts to set prices as they choose to fit, without any proper market insight or use of pricing tools, often leading to listings that exhibit similar characteristics in having significant price discrepancies. This issue reflects inefficiency, ultimately undermining both the host revenue and customer trust. Therefore, it is vital to understand the factors that influence Airbnb pricing, especially in competitive urban markets. Existing research has mainly focused on attributes like location, room type, and availability. However, with the increasing influence of online reviews and ratings is the customer's decision-making process, it is also important to understand how these reviews affect pricing. Factors such as cleanliness, communication, accuracy, and overall experience greatly reflect how guests perceive the listing's value and quality, these are vital things that influence a guest's willingness to pay the listed amount. Despite their significance, these quality indicators have not been properly explored in price modelling literature. Most studies emphasize price prediction but with limited attention to price optimization, such as identifying listings that are mispriced and recommending strategies to improve revenue and fairness. This study seeks to address this research gap by employing a machine learning framework that integrates listing level attributes and review

scores to model and optimize the rental prices of Airbnb listings. Specifically, this research used Airbnb datasets from four different urban markets, Amsterdam, Berlin, Athens, and Los Angeles. Machine learning models like Linear Regression, Ridge Regression, ElasticNet, Random Forest, XGBoost, LightGBM and CatBoost are applied to predict prices. By comparing actual prices to predicted prices, this study identifies underpriced and overpriced listings, leading to the enablement of the development of data-driven strategies for price adjustment. This approach not only offers technical insights but also practical value for hosts seeking to improve profit in a competitive market.

1.2 Research Problem and Objectives

One of the major problems addressed in this research is the absence of how the price factor mechanisms of Airbnb listings work. Which factors are involved in the price variations of the Airbnb listing? Due to a lack of knowledge and expertise, hosts often list their property at a price that is either below or above the market value. This can result in reduced sales from overpricing or minimal revenue from underpricing. Because guests will always prefer low priced listing with almost similar specifications, it creates a downward bias for hosts who unintentionally set a lower price as it reduces their potential revenue. This issue motivates me for my central research question:

"How do listing attributes and review ratings influence Airbnb rental prices, and how can a machine learning model be applied to predict and optimize the listing price to improve revenue and fairness?"

To support my research, the following sub-questions are listed below:

1. How does the review rating influence the prices of rental properties and overall revenue?
2. Is it possible to identify the underpriced or overpriced listings by comparing the actual price with model predicted prices and estimated revenue?
3. What amenities and review scores play a key role in justifying the rental prices?
4. Which amenities are important for premium rental properties?

1.3 Research Scope and Assumptions

This study has used the dataset of Airbnb listings which is publicly available on the Inside Airbnb platform. Datasets include listings from Amsterdam (10,168 listings, last updated on 09 June 2025), Berlin (14,187 listings, last updated on 20 June 2025), Athens (15,632 listings, last updated on 24 June 2025) and Los Angeles (45,421 listings, last updated on 17 June 2025). These datasets collectively include a broad range of variables, such as property type, room configuration, location, availability, and guest-generated review scores. The absence of comprehensive time series data, like peak and off-peak prices, is a significant dataset limitation. Furthermore, there aren't many premium listings in the dataset. Therefore, this study focuses mostly on guest review indicators and listing-level attributes, assuming that these factors are proxies for market appeal and listing quality.

1.4 Structure of the Report

The report is structured as follows: a detailed review of related previous research is in Section 2, followed by the research methodology in Section 3. Section 4 explains the system design, and Section 5 explains a detailed practical implementation of the whole machine learning pipeline. The results and evaluation metrics of each model, along with the model comparisons, are presented in Section 6. Lastly, Section 7 summarizes key findings, research contributions, limitations, and future directions.

2 Related Work

Rental platforms such as Airbnb are gaining popularity mainly because they make booking accommodation easier and offer a more user-friendly interface compared to the traditional process of booking hotels. This is one of the main reasons for its growing academic focus. Many studies have been conducted to predict the rental prices of Airbnb listings with the use of basic statistics and machine learning models. However, they did not consider the dual objective of rental prices, predictions and optimization of the listings' prices. The optimization of rental prices would not only allow underpriced listings to increase their revenue but would also allow overpriced listings to increase their sales, therefore increasing the overall revenue stream.

2.1 Traditional Statistical Approaches

Multiple Linear Regression (MLR) and Ordinary Least Square (OLS) are known as traditional statistical models which are widely used for predicting rental prices. One of the major reasons is because they require less computational power and consume less time and effort for training. These models are utilized for exploratory analysis and in policy making contexts. Samwel (2022) used OLS on a dataset of 30,478 Airbnb listings and achieved an R^2 score of 0.547. Similarly, Ogundunmade et al. (2023) collected 2465 listings data to predict price by using web scraping. They utilized two models, OLS and XGBoost, in which OLS achieved an R^2 score of 0.551. This low variance in both studies underpins that the OLS is unable to identify the price-driven factors. Whereas Mao (2024) applied MLR with enhanced feature selection by using Variance Inflation Factor (VIF) on a rental listing of Guangzhou. Two main features that influenced prices is location and apartment size were found in this paper whereas elevator presence and rent type show limited importance. The model comes out with an R^2 score of 0.464. For initial analysis and for feature selection these models are effective, but these models fail when handling data having non-linear relationships, multicollinearity and homoscedasticity. In this project to overcome these problems, I am using Ridge Regression (L2 regularized) and ElasticNet regression (a hybrid of L1 and L2 penalties), to test if these models give better accuracy than basic linear models.

2.2 Tree-Based and Ensemble Learning Techniques

Ensemble learning models, especially those which are based on decision trees, are more versatile and have emerged as the best model for predicting rental price, as they can handle nonlinear relationships and heterogeneous data types. In real estate price prediction, models like Random Forest, Gradient Boosting Machines (GBM), XGBoost, LightGBM, and CatBoost are widely preferred as they are robust, scalable and offer higher accuracy (Pastukh and Khomyshyn, 2025). Wang (2023) trained four machine learning models to evaluate the Boston Airbnb dataset of 3585 rows. The Gradient Boost regression model comes out to be the best performing model with an R^2 score of 0.719 on the training set and 0.659 on the testing dataset. In comparison with Linear Regression and K-Nearest Neighbors, this model significantly outperformed other posed models. Similar results have been noticed in research where Gradient Boost regression performed the best on dataset of houses in Dublin from Property Services Regulatory Authority (PSRA) with an R^2 score of 0.752 (Mirg and Latifi 2022). These results explain that the model can learn from structured variables like review scores, availability, etc. Moreover, Hu, Huang and Li (2022) utilized a random forest model and achieved high accuracy with an R^2 score of 0.915 in Wuhan dataset and 0.834 in Shanghai dataset. Similarly, Maheshwari et al. (2024) also demonstrated the random forest model strength in capturing complex and nonlinear relationships. These findings support my research directly, suggest using advanced and more powerful ensemble machine learning models like Random Forest, XGBoost, LightGBM and CatBoost. Due to mixed structured

and semi-structured data these models will suit the best in predicting underpriced and overpriced listings along with the new price suggestions.

2.3 Deep Learning and Neural Networks

Due to high dimensional, complex and hierarchical relationships among housing and Airbnb dataset, deep neural networks (DNNs) have gained much attention in predicting the price of the listings, especially when the dataset consists of both numeric and unstructured text data in it. Thakur et al. (2022) constructed a deep neural framework by using an Airbnb listing of 13,720 rows from Rio de Janeiro. After normalizing the input features, a setup of two hidden layers with ReLU activations were trained by using backpropagation. The model predictions come out with an R^2 score of 0.744 which outperformed previous studies. Yang (2021) trained a three-layer feedforward DNN model in comparison with XGBoost using Airbnb listing from Beijing. DNN underperformed with a testing R^2 score of 0.501 while XGBoost R^2 score comes out to be 0.654. These results might have several factors which cause DNN to not perform well, such as poor tuning of the neural architecture (e.g. layer width, depth, learning rate etc.), limited volume of the training dataset, noise of features like amenities, description and textual metadata. If these factors are handled carefully DNN can support high dimensions of data and perform well on tens or hundreds of thousands of records and with both numeric and textual data. In this research, I am not using deep neural networks (DNNs) because the combined dataset from Amsterdam, Athens, Berlin, and Los Angeles does not have enough volume and many text fields like description and amenities have missing values or have inconsistent format data. Instead of using DNN, I have used TF-IDF vectorizer, which works well with models like XGBoost and CatBoost. In the future, I hope to gather and analyze user reviews, booking trends, and listing images alongside other comprehensive information with the goal of text analysis to use deep learning techniques.

2.4 Preprocessing and Feature Engineering

To handle and clean raw data, Data preprocessing and feature engineering are crucial steps before modelling. Especially housing and Airbnb rental data as these datasets often contain mixed features categorial, numeric and unstructured textual features along with the missing values and outliers in them. Various techniques have been used in prior research for robust data preprocessing before model training, for example Sayyad et al. (2023) used real estate data from India and applied data preprocessing techniques to improve model accuracy, techniques like removing outlier, normalization of feature distributions with minmax scaling is used to enhance model accuracy and stability. For vectorization, Zhu, Li and Xie (2020) applied TF-IDF vectorization on the listing titles, sentiment scoring from more advanced natural language processing, and witnessed a rise in accuracy. It was eventually observed that XGBoost's R^2 reached 0.618. This research implements a multi-stage preprocessing pipeline designed to handle mixed data types across listings. Various techniques like handling missing values with filling with median and mode, standard scaler is used for normalizing the columns distribution, label encoding to convert categorical column into numeric, multiple text cleaning techniques like removing stop words, extra white space, special character, lowering the case, etc. and applied TF-IDF vectorization to capture semantic patterns based on word frequency and importance from major textual column.

2.5 Geo-Spatial and Location-Based Feature Engineering

Location is one of the most important features in predicting the price of real estate. Akalin and Alptekin (2024) found that the model accuracy improved from an R^2 score of 0.403 to 0.491 by adding location-based features to predict the Airbnb prices in Istanbul. Whereas

Peng, Li and Qin (2020) trained machine learning models on 9.9 million data points from Inside Airbnb collected across 10 global cities along with listing coordinates. They have used longitude and latitude for modelling and clustering techniques for spectral clustering of the listings. XGBoost was optimized using GridSearchCV, and it outperformed all other models. For this project, listing coordinates are used to measure the effect of location on pricing of Airbnb rentals. Latitude and longitude coordinates are used as numeric inputs and computed the distance to city centers using geodesic distance measures. In addition, city level average price per bedroom is calculated.

2.6 Hyperparameter Tuning for Enhanced Model Performance

Hyperparameter tuning is used to improve model accuracy and to help the model to understand the data and its relationships. Various sets of instructions are given to the model in the form of parameters. Malik, Hassouna and Togher (2023) used RandomizedSearchCV for hyper tuning the logistic regression. The final model achieved 59.18% accuracy, precision of 97.1%, recall of 59.3%, and F1-score of 0.74. A similar approach is employed by Ghosh et al. (2024) where random forest is used along with Grid Search Optimization for hyperparameter tuning. The model achieved an R^2 score of 0.885. For ensemble models such as Random Forest, XGBoost, LightGBM, and CatBoost, I conduct hyperparameter tuning using RandomizedSearchCV. The reason for using these models is because datasets of Airbnb and rental house often have a mix of categorical, numeric and unstructured textual features, along with missing or noisy data and these models work good with such data. With the help of propriate tunning the models are capable of achieving high accuracy and precision in predicting the Airbnb prices.

2.7 Hybrid and Ensemble Modelling

Due to their application in real estate and rental price forecasting, ensemble modelling techniques have become more prominent in contexts relating to prediction accuracy and generalization of the model. Ensemble modelling is useful in heterogeneous markets where no single model is optimal in every scenario, as it aggregates two or more learners (e.g. regression models, neural networks) in order to produce better predictions (Murel and Kavlakoglu, 2024). Chan (2024) worked with "USA Housing Listings" dataset which has 384,977 records with 22 features and included the rent price, square footage, number of beds and baths, location (latitude and longitude), parking and laundry options, and pet-friendly status. After pre-processing, the California and Texas subsets with the 22,946 and 19,156 rows respectively, were used. For California, the housing price prediction used a stacked ensemble model with Ridge Regression, Random Forest, and LightGBM. This combination produced the highest R^2 of 0.885 across all models on the Californian dataset. For Texas, a region with more homogeneous housing data, the study found that a Random Forest model alone outperformed all other models at 0.7992 R^2. This result explains the benefits of stacking models which can improve model accuracy but as well as acquire more resources. My current pipeline's modular design is informed by these findings, implementing a set of models including XGBoost, LightGBM, CatBoost, Ridge, and Random Forest. Although formal stacking has not yet been implemented, modular model wrappers along with consistent feature alignment amongst base learners in the architecture ensure that this capability is supported.

2.8 Outlier, Multicollinearity and Correlation

To achieve the maximum accuracy from any predictive model is to ensure that data is cleaned, preprocessed before modelling and does not have any outlier or multicollinearity

between independent variables. Many studies have shown that removing outliers improves the accuracy of the regression model. For example, in a study Tukey's method was used (Yang 2021) while another Study used Z-score filtering to remove values that were too high or too low for model training and prediction (Hussain et al. 2024). One of the issues that multicollinearity causes is the loss of model interpretability and performance by increasing the variance of the coefficients. This problem was identified by Mao (2024), VIF analysis is applied to remove multicollinear independent variable from the data. To achieve a high level of accuracy and predictions, I identified outliers by distribution plots and descriptive statistics and then removed by using inter quartile range method (IQR). And for skewness, log transformation is performed by using StandardScaler which centers numeric variables.

2.9 Model Weight and Class Imbalance

During model training one of the most problematic and difficult tasks is to handle class imbalance problems. Even worse in a case where one class is represented much less than others. To overcome this problem, Bakirarar and Elhan (2023) showed how model performance could be improved by applying class weighting on imbalanced datasets. They conducted a comprehensive analysis of various class weighting approaches like inverse class frequency, square root scaling, and sample-based approaches on Random Forest and SVM models. They showed that G-mean and balanced accuracy improved greatly when the minority class was properly weighted. I encountered a similar class imbalance problem. Upon checking the price distribution of the listing, I encountered that the frequency of expensive listing in the dataset is low. This causes the model to fail in predicting the higher price listing (expensive listing). This type of imbalance can often bias models towards the majority class and hence poorly predict the performance for the minority (expensive listing,) class. Applying the same principles, I set the model weights to emphasize the expensive listings while training. With this approach, the model was able to appropriately capture the features of the less representative class and enhance its classification performance.

2.10 Systematic Review of Machine Learning Models

Salam et al. (2022) present an overview and perspective on the application of machine learning in real estate. They conducted a systematic review of twenty-two papers published between 2009 and 2020 to understand the prevailing machine learning techniques utilized for predicting the prices and rents of real estate properties. While not relying on a single empirical dataset, the review synthesizes important meta-level insights from numerous research studies with varying attributes of real estate, such as property location, size, type, number of rooms, and distance from essential services. The results emphasize Random Forest as the most popular model used in 10 studies, with Decision Trees, Linear Regression, and Support Vector Machines following in usage at 9, 8, and 6 studies respectively. Random Forest is especially well-known for reliably capturing complex non-linear relationships and dealing with high-dimensional data, making it a go-to model for price and rent predictions. Additionally, the review comments on the application of ensemble methods, supervised regression and classification, as well as unsupervised clustering, which further evidences the real estate analytics movement towards hybrid and data-driven models. These observations help in choosing the machine learning model in this research. I used Random Forest, XGBoost, and other ensemble models, because of their demonstrated effectiveness and expansion capabilities. Salam et al. (2022) also underscore the fact that dissemination area, real estate category, real estate square footage and proximity to urban amenities are among the most important determinants of value change. This review consolidates the application of rigorous ensemble machine learning algorithms through analyzing trends and feature

selection because they create reliable and understandable real estate pricing frameworks systems as outlined in the literature.

3 Research Methodology

In this section a detailed, quantitative and data driven methodology are explained which aims to predict Airbnb listing price along with its optimization. The design of this research project follows a structured pipeline from data collection to price optimizations and suggestions. The pipeline follows a sequence of processes like data collection, data preprocessing, exploratory analysis, feature engineering, data modelling, model development, evaluation and prize optimization. The methodology is constructed to ensure scalability and reliability across multiple cities and generalizability of findings.

3.1 Data Collection

This study used data from Inside Airbnb, which is an open-data repository that contains Airbnb listings and offers datasets for various cities around the world. To conduct this research four metropolitan areas were selected as shown in Figure 1. Amsterdam with 10,168 listings (June 9, 2025), Berlin with 14,187 listings (June 20, 2025), Athens with 15,632 listings (June 24, 2025) and Los Angeles with 45,421 (June 17, 2025). The datasets contain data for one year from the date of update according to the source website Inside Airbnb. Each dataset contains tens of thousands of listings and up to 80 attributes including a wide range of prices, property type and room type, amenities, availability, and rated reviews which holds attributes like cleanliness, accuracy, communication, and overall satisfaction.

```
Amsterdam, North Holland, The Netherlands
(10168, 80)
Athens, Attica, Greece
(15632, 80)
Berlin, Berlin, Germany
(14187, 80)
Los Angeles, California, United States
(45421, 80)
```

Figure 1: Dimensions of used Datasets

3.2 Data Preprocessing

Data Preprocessing is a crucial step as it converts the raw data into structured data and ensures data consistency by handling missing values. It is important to handle both numerical and object type fields according to requirements. All standard techniques are used to handle the data. The columns that show excessive missing data are discarded. A few missing attributes are filled with median and mode. Data types are standardized and price plus review score fields are converted into numerical formats. For textual data, text data cleaning process is performed for better interpretation of the textual columns. To handle categorical or object type features, label encoding is used to convert such features into numeric format for analysis. For mapping these encodings are saved in the Json file for decoding purposes. For Outlier detection, a price distribution graph was used and through inter quartile range method listings that showed high and unreasonable prices are removed as these factors could seriously impact model training. These steps are covered in detail in Section 5.

3.3 Exploratory Data Analysis (EDA)

After cleaning process, data undergoes an important step, Exploratory Data Analysis (EDA). This step helps in understanding the different patterns, relationships and correlations between different features. Various processes are involved to help identify the important and relevant features which are responsible for variation in target variable (i.e. Price). The early

exploration step was taken in data preprocessing as it is standard procedure to remove outlier if any available. The use of histograms and box plots allowed the analysis of the distribution of prices and potential outliers. Furthermore, correlation heatmaps, Pearson and Spearman correlation statistical test is performed to show how strongly prices relate with other features. Variance inflation factor is calculated to check the multicollinearity between independent variables. To study the high multicollinearity variables, distribution graphs are generated and comparison box plot of price with highly correlated feature are also created to analyse them deeply. The EDA was done using Python libraries like matplotlib and seaborn for visualization, therefore revealing patterns clearly and interactively as well as visually.

3.4 Feature Engineering

After visualization, few columns needed to be engineered for better interpretation. Three columns are generated like average bedroom price per city, distance from city centre in meters and amenities count. As these numeric columns will be more helpful for model to accurately predict the price of the listing. The data is grouped by city then average bedroom price is calculated with the help of price and bedroom column. Length function is used to count the amenities. Furthermore, another column is created as average distance from city centre. Haversine distance function is used to measure the distance of the listing from the centre as it may have significance for predicting the price of the Airbnb listing. The distance is measured is meters. To generate these columns Python libraries like Math and NumPy are used. All these new engineered features are added back in the main dataset for further data modelling.

3.5 Data Modelling

After exploration, a set of key steps are taken before training the model, which includes feature engineering, removal of redundant features and vectorization of textual column. With feature engineering, various new attributes are introduced, as explained in Section 3.4. The redundant features and features with high multicollinearity are removed by using insight from VIF score during exploration process. Additionally, some attributes that were minimal correlated are eliminated with the help of Pearson and Spearman correlation statistical test. After which informative features were extracted from unstructured data by transforming text features with the use of TF-IDF vectorization. These processes guaranteed that the dataset was perfect in order to carry out the subsequent steps of model training and evaluation.

3.6 Machine Learning Models

This section provides the reason and an overview of the choice of models. In total, seven models are applied in the scope of this research. Linear models like Linear Regression, Ridge and ElasticNet are used for their effectiveness in stating baseline prediction and assessing feature performance. Regularization parameters such as alpha and L1 ratio were set to avoid overfitting. Advanced ensemble approaches such as Random Forest, XGBoost, LightGBM and CatBoost were used to capture non-linear relationships to increase model accuracy. Except linear regression, all the models are hyperparameter tuned to gain maximum performance and accuracy. For that purpose, randomized search cv grid is used to identify the best performing model with respect to 3-fold cross validation R^2 score. For instance, LightGBM is trained with 300 estimators, 70 leaves and a max depth of -1. Whereas XGBoost was trained using GPU hist tree method for fast training. Sample weights are applied during training process on listing having price of more than 280$ mark to improve the accuracy of expensive listing (an imbalanced class). By using both linear and tree-based models, this research delivers a comprehensive analysis of the listing and states predictive

power of the models implemented. Finally, a goal to identify the best model for a particular problem statement.

3.7 Model Evaluation

A range of quantitative metrics is created to check errors and accuracy. Whereas visual assessment like feature importance and actual vs predicted scattered plot are generated to check the model precision across data. Evaluation metric is calculated, which includes the Mean Absolute Error (MAE) which states the average error in actual units, the Root Mean Square Error (RMSE) which states the average magnitude of the prediction error, R^2 score which states accuracy of the model and lastly cross-validation R^2 score which helps to check model consistency. Training and evaluation of models were conducted using GPU-based 7-fold cross validation. The metrics captured from all models including RMSE, MAE, R^2 and cross-validated R^2 were sufficient for reliable evaluation of their predictive performance. After that best performing model undergoes feature importance analysis to evaluate the interpretability of the model. For visualization, a bar plot is generated. Along with that Shap analysis is also conducted to analyse the directions of the feature for price variation and to identify which review score and amenities are responsible for driving the price upward.

3.8 Analysis of Price Optimization

To provide actionable suggestions, price optimization techniques are used with the help of model outputs. The predicted price with a threshold is compared with the actual price to classify the listing as underpriced, overpriced and fairly priced. Objectives-based revenue optimization with the price prediction model was run to determine how minimal changes to listing prices impacted revenue. In this analysis, listings were priced closer to predictive values and the difference in predicted revenue was analyzed creating a feedback loop for recommending pricing changes. This analysis and output results will help the host to change their price according to the market conditions to maximize the revenue.

3.9 Ethical Considerations

Despite the fact that the datasets are publicly accessible, many ethical principles were observed in order to protect the modelling results and the data's impact. All host names, profile links, and listing URLs containing personal identifiable information were removed to protect user privacy for each step of the analysis. To encourage fair and balanced insights, unrealistic, unaffordable, or exploitative prices that were intentionally developed from the analysis were avoided by removing them from the data. Attention was also placed on bias detection to determine that some neighborhoods, property types, and certain price ranges were not unfairly modeled in favor of or against. These measures not only increase fairness and transparency but also increase accountability in regard to the project.

4 Design Specification

This section explains the structural and functional design of the research project and all the major processes like data flow, model design, tools and key decisions. All components of this research architecture aim to provide a machine learning pipeline which can be used across multiple Airbnb datasets from various cities with few adjustments. This design ensures the modularity, reproducibility and scalability of the overall pipeline. The goal is to build a system that not only predict the rental price of the Airbnb listing but as well as identify the underpriced and overpriced listing and recommend adjustments that can help in generating maximum revenue.

4.1 System Architecture

The system is designed as a reusable and modular pipeline with eight core modules that process the data step by step in a sequence to predict and optimize the Airbnb pricing as shown in Figure 2. It starts with the first step as import data which gathers the listing from a CSV file and combines then to use as a single dataset for the cleaning and preprocessing step using pandas. The second step is data preprocessing, taking the data and continuing with the cleaning process like handling missing values, uneven data types, removal of outliers, textual column cleaning etc. The third step visualization comes in which it creates useful plots and graphs to analyze the data. After this process another module featured engineering kicks in which introduced new features like average bedroom price for cites, distance from city center and amenities count. Then a crucial step takes place in data modelling, which models the data for training. This step includes textual column vectorization, removing high variance features and splitting the data into 80/20 ratio for training and tests. Now the model development process starts, which is responsible for training of the models (Linear Regression, Ridge, ElasticNet, Random Forest, XGBoost, LightGBM, CatBoost). These models are hyper tuned by RandomizedsearchCV. Once model is trained, model evaluation step evaluates the model performance using R^2, RMSE, and MAE. Finally, the Optimization & Insights module identifies underpriced and overpriced listings and estimates potential revenue gains, providing hosts with data-driven pricing recommendations.

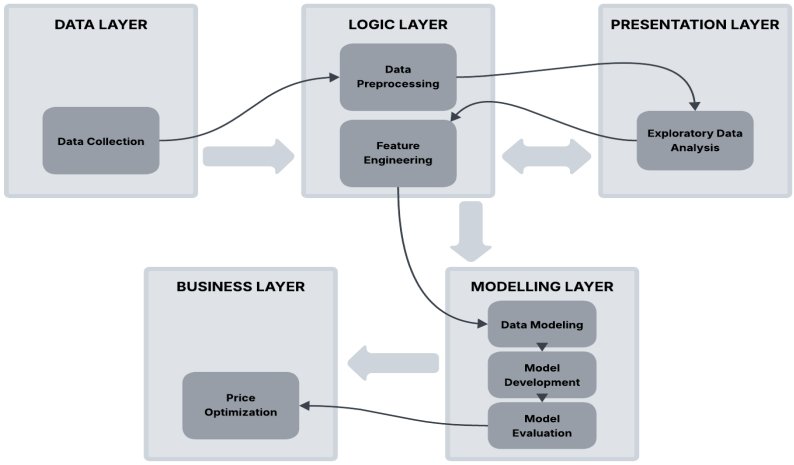


Figure 2: System Architecture Flow Chart

4.2 Data Flow Design

The data flows through the system containing eight core modules as discussed above while ensuring consistency and transparency. The following is the data flow chart in Figure 3:

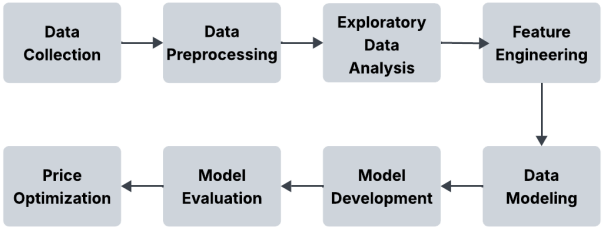


Figure 3: Data Flow Chart

4.3 Tools and Technologies

To conduct this research several tools and technologies have been used throughout the process. These tools have been used in many previous researchers for their reliability, and flexibility. The tools are listed in table 1 below along with their purpose in this research.

Table 1: Tool and Technologies used for Research along with their purpose of use

Library / Module	Purpose
VS code	Primary development environment.
Python 3.10+	Programming language is used for data processing, modelling, and analysis.
Jupyter Notebook	Interactive development environment used for writing and testing code.
warnings	Suppresses unwanted warning messages
Json	To save encoding in json format.
time	Provides time-related functions (e.g., delays, timing execution)
re	Enables regular expression operations for text parsing
numpy (np)	Supports numerical computations and array operations
pandas (pd)	Handles data manipulation and tabular data structures
matplotlib.pyplot (plt)	Used for static data visualization
seaborn (sns)	Enhances visualizations with statistical plotting
cupy (cp)	Enables GPU-accelerated numerical computing (alternative to NumPy)
RandomizedSearchCV	Performs hyperparameter tuning via randomized search
KFold	Enables cross-validation splits
StandardScaler	Normalizes features by removing the mean and scaling to unit variance
LabelEncoder	Encodes categorical labels into numerical format
TfidfVectorizer	Converts text data into TF-IDF feature vectors
Metrics	Provides performance metrics like RMSE, MAE, R^2
Permutation_importance	Computes feature importance through permutation
os	Provides functions for interacting with the operating system
ast	Parses strings into Python abstract syntax trees (used for parsing lists)
scipy.stats	Provides statistical functions, e.g., z-score, skew
outliers_influence.VIF	Computes Variance Inflation Factor for multicollinearity analysis
statsmodels.api (sm)	Used for advanced statistical modelling
joblib	Used to serialize and save trained models
Shap	Used to explain the output of machine learning model

4.4 Output & Deliverables

The system provides various outputs to help the hosts and researchers understand the price variation factors involved in Airbnb listing and its price optimization. Models are trained and evaluated using accuracy and cross-validation scores to predict prices and suggest optimization strategies. A feature importance plot is generated for the best performing model to identify key factors influencing price variation. To find the direction of those features Shap analysis is performed. After that listings are categorized into three groups: Fairly Priced, Underpriced, and Overpriced, based on a 15% margin on the predicted price from the best performing model. The structured tables of results showcase the current price, predicted

price, pricing flag, and revenue improvement potential. These results serve as practical decision-making tools while documenting the system's overall effectiveness.

5 Implementation

This section will discuss the implementation phase of the research. This section includes all the core modules which are designed and implemented. From loading data to model training and from model training to price optimization every model is explained in detail. Moreover, step by step execution of the processes, data handling, feature engineering, model training etc. are concluded in this section.

5.1 Data Import and Setup

Visual Studio Code (VS code) environment within a Jupyter Notebook by using Python 3.10.0 is used for this research. This setup is suitable and reliable for research like these. Once the environment is prepared, the process begins with loading Airbnb data from four metropolitan cities, named Amsterdam, Berlin, Athens, and Los Angeles. The datasets are retrieved using the Pandas library through CSVfiles. After the data retrieval process is complete and all the data from the respective cities is collected, they are merged into a single dataset with an additional column indicating the city and country associated with the dataset. For base level exploration head functions are used to check if the data is loaded successfully or not, info and describe functions are used to understand the datatypes, basic statistics and potential issues as shown in Figure 4. Along with that null values are examined as shown in Figure 5, from which top 20 missing values plot is generated to highlight them and for better insights as shown in Figure 6. The Seaborn and Matplotlib libraries of Python are used to plot graphs and visualizations. After importing the data, data is transferred to the next module for data preprocessing.

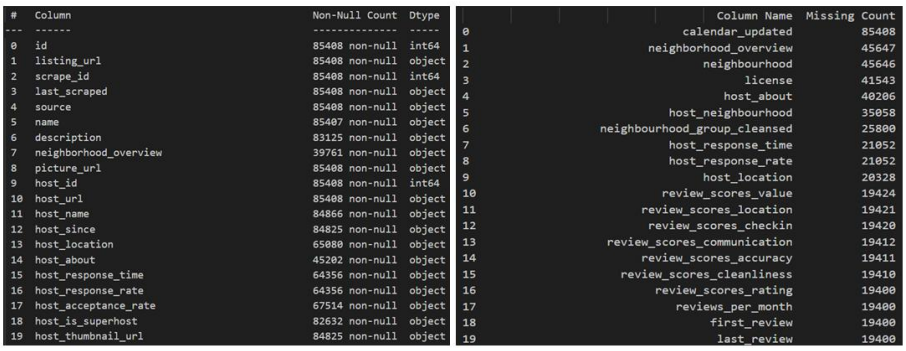
	id	listing_url	scrape_id	last_scraped
0	27886	https://www.airbnb.com/rooms/27886	20250609011745	2025-06-17
1	28871	https://www.airbnb.com/rooms/28871	20250609011745	2025-06-17
2	29051	https://www.airbnb.com/rooms/29051	20250609011745	2025-06-17
3	44391	https://www.airbnb.com/rooms/44391	20250609011745	2025-06-17
4	47061	https://www.airbnb.com/rooms/47061	20250609011745	2025-06-17

	source	name	description
0	city scrape	Romantic, stylish B&B houseboat in canal district	Stylish and romantic houseboat on fantastic hi...
1	city scrape	Comfortable double room	Basic bedroom in the center of Amsterdam.
2	city scrape	Comfortable single / double room	This room can also be rented as a single or a ...
3	previous scrape	Quiet 2-bedroom Amsterdam city centre apartment	Guests greatly appreciate the unique location ...
4	city scrape	Charming apartment in old centre	A beautiful, quiet apartment in the center of ...

	id	scrape_id	host_id	host_listings_count
count	8.540800e+04	8.540800e+04	8.540800e+04	84825.000000
mean	6.274264e+17	2.025062e+13	2.109086e+08	75.194035
std	5.646106e+17	4.189591e+06	2.106633e+08	496.471849
min	2.708000e+03	2.025061e+13	7.670000e+02	0.000000
25%	3.377184e+07	2.025062e+13	2.608205e+07	1.000000
50%	7.459551e+17	2.025062e+13	1.231249e+08	2.000000
75%	1.170225e+18	2.025062e+13	4.010557e+08	11.000000
max	1.449435e+18	2.025062e+13	7.022443e+08	4925.000000

	host_total_listings_count	latitude	longitude	accommodates
count	84825.000000	85408.000000	85408.000000	85408.000000
mean	103.983554	40.018874	-55.766684	3.684631
std	651.652572	7.984447	66.856641	2.471007
min	0.000000	33.338540	-118.917134	1.000000
25%	1.000000	34.056575	-118.346241	2.000000
50%	4.000000	34.201868	-117.998690	3.000000
75%	15.000000	52.357311	13.422007	4.000000
max	9109.000000	52.656110	23.780220	16.000000

Figure 4: Data Head and Descriptive Statistics of Data



#	Column	Non-Null Count	Dtype
0	id	85408 non-null	int64
1	listing_url	85408 non-null	object
2	scrape_id	85408 non-null	int64
3	last_scraped	85408 non-null	object
4	source	85408 non-null	object
5	name	85407 non-null	object
6	description	83125 non-null	object
7	neighborhood_overview	39761 non-null	object
8	picture_url	85408 non-null	object
9	host_id	85408 non-null	int64
10	host_url	85408 non-null	object
11	host_name	84866 non-null	object
12	host_since	84825 non-null	object
13	host_location	65080 non-null	object
14	host_about	45202 non-null	object
15	host_response_time	64356 non-null	object
16	host_response_rate	64356 non-null	object
17	host_acceptance_rate	67514 non-null	object
18	host_is_superhost	82632 non-null	object
19	host_thumbnail_url	84825 non-null	object

	Column Name	Missing Count
0	calendar_updated	85408
1	neighborhood_overview	45647
2	neighbourhood	45646
3	license	41543
4	host_about	40206
5	host_neighbourhood	35058
6	neighbourhood_group_cleansed	25800
7	host_response_time	21052
8	host_response_rate	21052
9	host_location	20328
10	review_scores_value	19424
11	review_scores_location	19421
12	review_scores_checkin	19420
13	review_scores_communication	19412
14	review_scores_accuracy	19411
15	review_scores_cleanliness	19410
16	review_scores_rating	19400
17	reviews_per_month	19400
18	first_review	19400
19	last_review	19400

Figure 5: Information on data types and missing

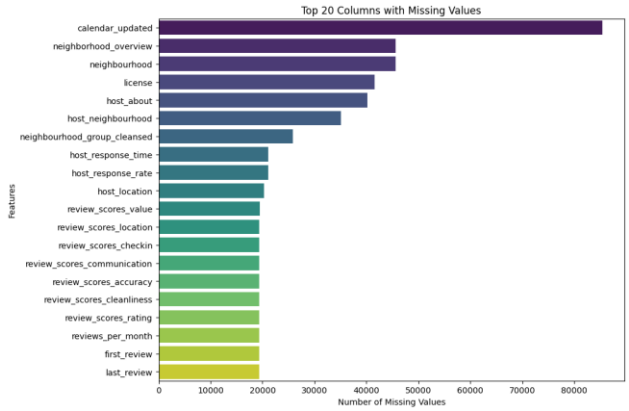


Figure 6: Bar plot of top 20 missing columns

5.1 Data Preprocessing

Data preprocessing is responsible for handling and transforming raw data into well-structured and usable data. A series of steps have been taken; the first step begins by handling missing values. The columns that have more than 40% of missing values and are redundant like id, URLs etc. are removed, whereas rows are removed where crucial features are missing, like estimated revenue. Review scores columns with missing values are treated with median by grouping according to source city. The remaining numeric and object type columns are filled with median and mod, respectively. Price column is converted into float by removing the dollar sign as price was in a string datatype. For other textual columns like name, description about host, a series of text cleaning processes are carried out, including HTML tags removal, converting text in lowercase, eliminating non-alphabetic characters, and removal of stop words and strips commons. A similar approach is used to clean amenities column, where extra white space is removed, special characters and space between the tuples of words are removed for standardization. For textual cleaning two key libraries are used Regis and ast. Label encoder is used for categorical columns like room type etc. to convert them into numeric forms along with that encoding are saved as a Json file. As per standard data preprocessing, a simple outlier identification is performed by using price distribution plot is used (Figure 7). Few listings with very unrealistic prices have been noticed (above 40,000$). To handle outliers, IQR method is used to remove outliers as shown in Figure 7.

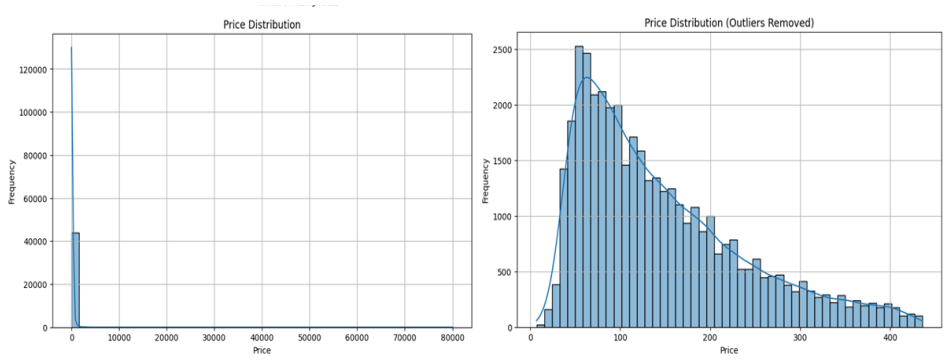


Figure 7: Price Distribution plots before and after outliers

5.2 Exploratory Data Analysis

An exploratory data analysis is conducted to determine the relationship between different features within the data. Multiple graphs and tables are generated to study deeply about features which are related to price. To find the correlation between different attributes a

correlation heatmap is generated (Figure 8). To identify features highly correlated with price, Pearson and Spearman tests are performed using SciPy, along with Variance Inflation Factor (VIF) to detect multicollinearity (Figure 9). High-variance features such as review scores, beds, bedrooms, room type, property type, and accommodates are analyzed through distribution plots (Figure 10), using bar plots. Key price-related features, like beds, bathrooms, bedrooms, and guest capacity, are further examined using box plots (Figure 11). Matplotlib and Seaborn libraries of python are used for Visualizations, and the whole data exploratory module is developed with proper exception handing clause to ensure the traceability and robustness of the data pine line.

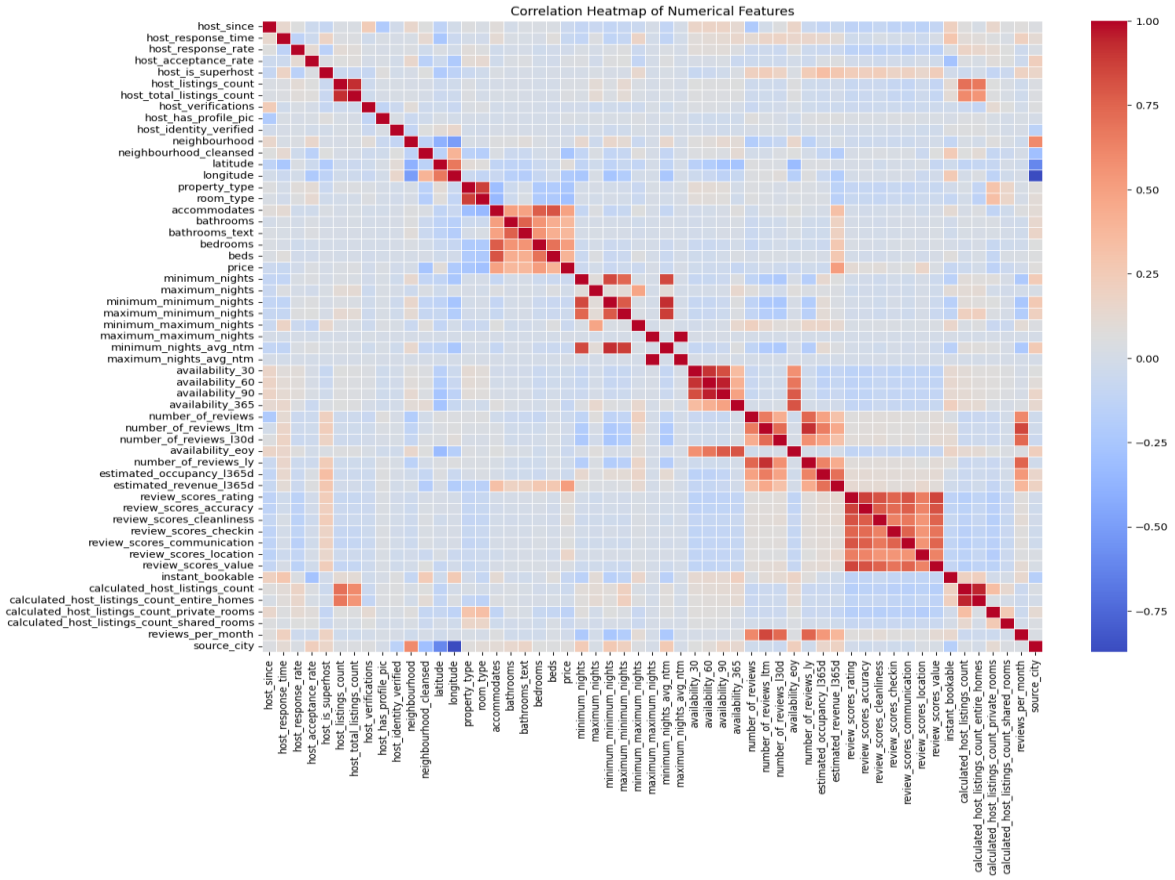


Figure 8: Correlation Heatmap

Feature	Pearson	Spearman
estimated_revenue_l365d	+0.5156	+0.4874
bedrooms	+0.5122	+0.4678
accommodates	+0.4889	+0.4714
beds	+0.3971	+0.3926
bathrooms	+0.3724	+0.3853
bathrooms_text	+0.3719	+0.2258
review_scores_location	+0.1614	+0.2187
review_scores_rating	+0.1099	+0.1573
review_scores_cleanliness	+0.0925	+0.1142
latitude	+0.0794	-0.1087
review_scores_accuracy	+0.0789	+0.1130
source_city	+0.0753	+0.1380
review_scores_communication	+0.0632	+0.0986
review_scores_checkin	+0.0476	+0.0789
host_total_listings_count	+0.0456	-0.0851
host_acceptance_rate	+0.0393	+0.0513
host_listings_count	+0.0350	-0.1089
host_has_profile_pic	+0.0345	+0.0409
review_scores_value	+0.0303	-0.0010
maximum_maximum_nights	+0.0107	-0.0464
maximum_nights_avg_ntm	+0.0107	-0.0518
neighbourhood	+0.0042	+0.0767
host_is_superhost	-0.0012	+0.0122

Feature	VIF
maximum_maximum_nights	387228111.42
maximum_nights_avg_ntm	387227312.35
const	692.53
calculated_host_listings_count	152.18
calculated_host_listings_count_entire_homes	119.56
availability_60	24.16
availability_90	20.25
calculated_host_listings_count_private_rooms	14.45
minimum_maximum_nights	13.77
minimum_nights_avg_ntm	12.84
number_of_reviews_ltm	12.71
host_listings_count	10.03
number_of_reviews_ly	9.42
host_total_listings_count	9.31
longitude	8.76
minimum_minimum_nights	8.04
review_scores_rating	7.71
availability_eoy	7.52
availability_30	6.34
source_city	5.87

Figure 9: Correlation and Multicollinearity Tests

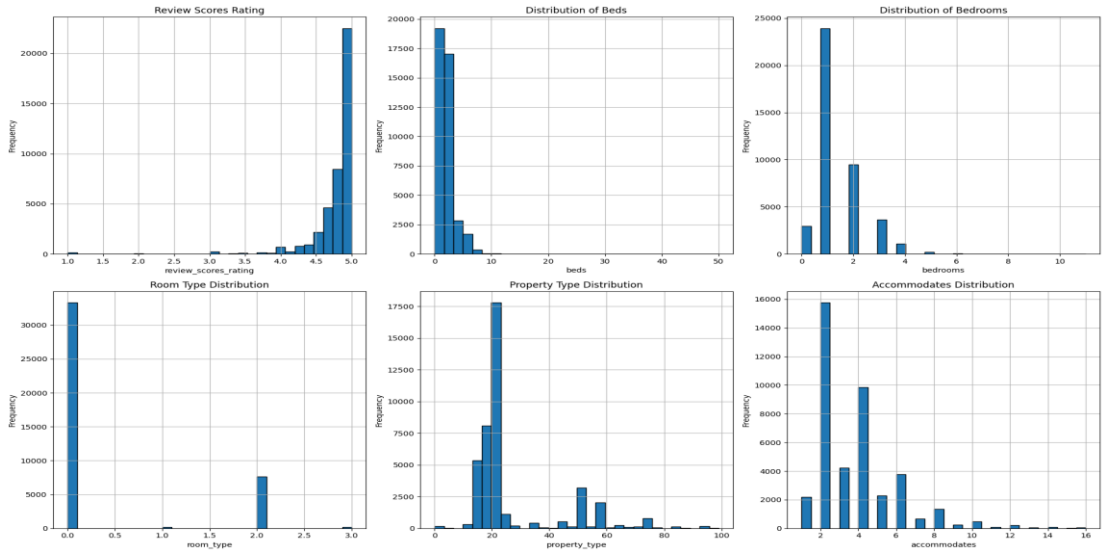


Figure 10: Distribution Graphs of Various Features

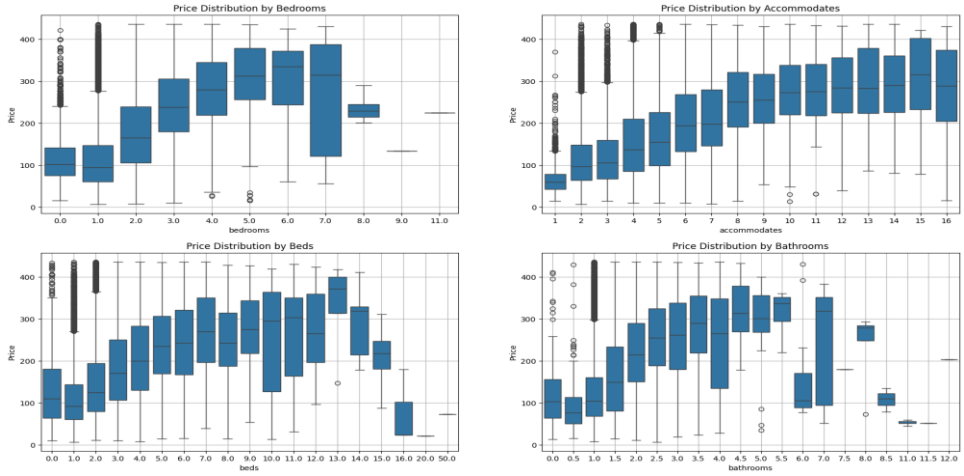


Figure 11: Price Distribution Across Various Features

5.3 Feature Engineering

In this step new features are created from the dataset to improve the model accuracy and its performance. At first, amenities count features is created which is drive from the amenities column that contained a list in string format. The length function is used to count the amenities. An average price of a bedroom for each city is calculated too and to add spatial data, the Haversine distance function was applied to calculate straight-line distance between the listing and city centre of the city from which the listing belongs too. The distance is measured in meters. These additional features will improve model contextual understanding and over all accuracy and precision of the model. An appropriate development paradigm is used for this research with exception handle for the reliability and traceability of the research.

5.4 Data Modelling

After data visualization, data is prepared for modelling. In this step data is transformed to enhance the model's performance and stability. After visualizing the data key plots and advanced statistics like correlation heatmap, distribution graphs, Pearson and Spearman correlation test and Variance Inflation Factor (VIF), columns like latitude, longitude, host total listings count, host listings count, etc. with least correlation and columns with high multicollinearity (VIF score) are removed. These steps help to reduce redundancy and

enhance model performance. After that text fields which were cleaned in data preprocessing step like name, neighborhood overview, description, and amenities undergo vectorization using TF-IDF vectorization. A limit of 200 terms per field is set which captures the important and more frequent terms and transforms them into features. From the price distribution graph and upon checking skewness, it is identified that the price column is positively skewed, a log transformation function is applied to reduce skewness through the Numpy log function. This helps the model to capture linear relationship between the dependent and independent variable and reduce the variance in the coefficient. And in the end for training model, the dataset is split into 80/20 for training and testing with the help of train-test split function from the sklearn library.

5.5 Model Development

In this step a range of different machine learning models are developed and trained to predict the price of Airbnb dataset. For training as discussed in section 2 a combination of linear and ensemble-based models is trained. For linear models' linear regression, ridge regression and elasticnet model are used whereas for ensemble models: Random Forest, XGBoost, LightGBM and CatBoost are used. These ensemble models are well-known for their performance for problems where there is presence of structured data along with the unstructured data. Whereas linear models are for baseline predictions as they are not the best option for such problems but linear models like ElasticNet and Ridge regression are used to test models' performance having hybrid L1 and L2 penalties and L2 regularization respectively. All the models are hyperparameter tuned with the help of Randomized Search CV from Scikit-learn. This function is set up with 30 random sets from the given parameters along with model and evaluate each of the model with 3-fold cross-validation and returns the best performing model with respect to R^2 score. To handle class embalance, especially for expensive listing (price above 280$) a custom sample weight is applied. The listings with prices more than 280$ have given weighting of 2.0. This helps the model capture emb

5.6 Model Evaluation

Once training is complete, the models are evaluated to measure accuracy, generalization, and real-world forecasting capability of Airbnb rental prices. At first the predicted price is converted from log scale to original scale to map them with original price with the help of inverse log function of NumPy library. The test data undergoes with the evaluation metrics consist of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2 Score. These metrics explain how accurate the model for the Airbnb dataset and its capacity to generalize price variation. Moreover, to validate the reliability of the model on Airbnb dataset, a 7-fold GPU-accelerated cross validation R^2 score is calculated on training data with the help a custom function by using Cupy library. This custom function ensures consistent validation across the data while use of GPU speeds up the whole computation. Along with these numeric checks a visual approach is also entertained by plotting scatter plots with the help of seaborn and matplotlib between actual and predicted prices of the listing as shown in Figure 13. A tight grouping along the 45-degree line indicates that predictions are accurate. For the best performing model, a permutation-based feature importance analysis is designed to evaluate the interpretability of the model. This analysis identifies and visualizes the most influential predictors. To visualize the top 20 important features a horizontal bar plot is created with the help of seaborn and matplotlib as shown in Figure 12. Along with that for feature importance direction in price variation Shap analysis is performed to find deep insights about features impact on pricing specially to identify the significant amenities with are responsible for up drifting the price of the Airbnb listing. Two different Shap summary plots are generated, one for all features and the other for only amenities.

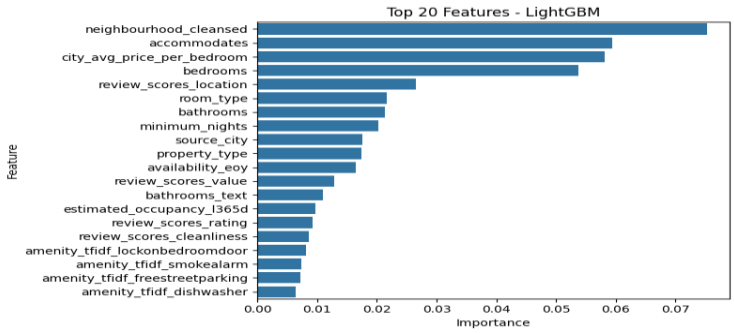


Figure 12: Top 20 Feature Importance Bar Plot for LightGBM

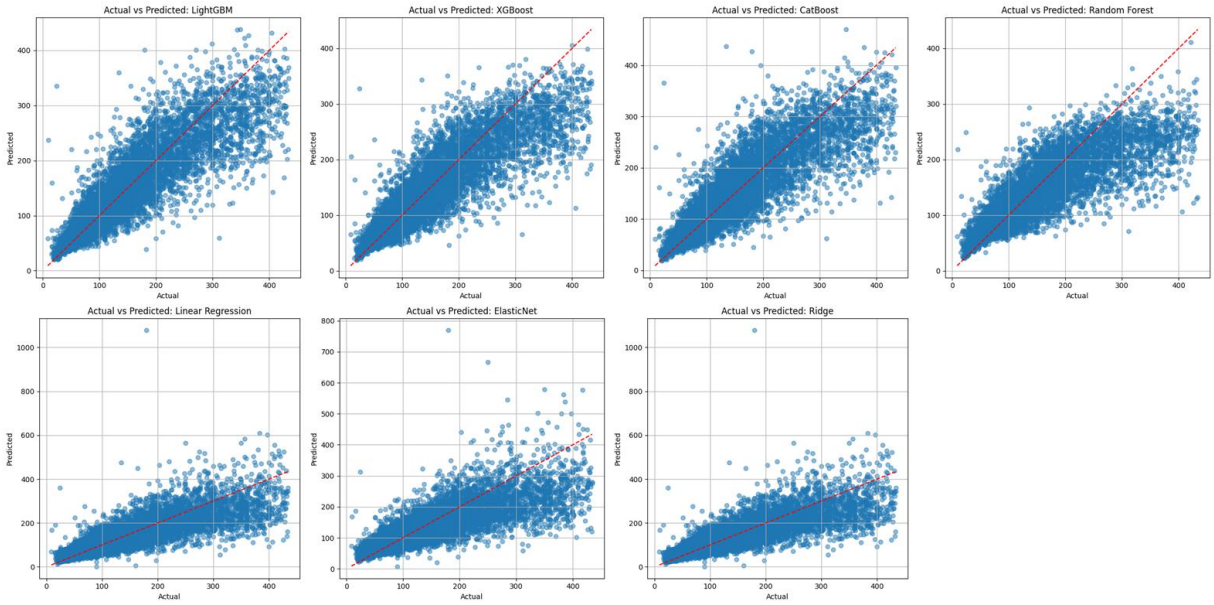


Figure 13: Scatter Plots of Various Machine Learning Models

5.7 Price Optimization Analysis

After evaluating the models, the last step of this pipeline kicks in by using the best performing model to generate actionable insights for the price and revenue optimization of the Airbnb listing. At first the predicted price is calculated by loading the trained model with the help of joblib library. As the predicted prices are in log scale, it is converted back into original form by using log inverse function of NumPy. After that original price and predicted price goes through pricing classification process which categorized the listing as three main categories as underpriced, overpriced and fairly priced as shown in Figure 14. This process uses a threshold of 15% above and below the actual price for categorization. For listings which are identified as underpriced, a revenue opportunity calculation is performed. The potential additional revenue is estimated by multiplying the predicted price by the listing's estimated annual occupancy. This result is stored in a new column called new potential revenue as shown in Figure 15. At last, a summary output is generated which displays the total count of listings in each pricing category as shown in Figure 14. This helps stakeholders to prioritize which listings could benefit the most from pricing adjustments.

	name	price	predicted_price	\
0	romantic stylish bb houseboat canal district	132.0	158.441324	
1	comfortable double room	78.0	94.837355	
2	comfortable single double room	70.0	73.441483	
4	charming apartment old centre	120.0	254.517609	
6	multatuli luxury guest suite location	284.0	281.145194	

	pricing_flag	estimated_revenue_l365d
0	Underpriced	28776.0
1	Underpriced	19890.0
2	Fairly Priced	17850.0
4	Underpriced	4680.0
6	Fairly Priced	72420.0

	Count	count
0	Fairly Priced	26800
1	Overpriced	7367
2	Underpriced	7168

Figure 14: Actual vs Predicted Price and Price Labeling

	name	price	predicted_price	\
	romantic stylish bb houseboat canal district	132.0	158.441324	
	comfortable double room	78.0	94.837355	
	charming apartment old centre	120.0	254.517609	
	cozy central room amsterdam	63.0	76.791773	
	sunny amsterdam east harbor view	123.0	155.373442	

pricing_flag	estimated_revenue_l365d	new_potential_revenue
Underpriced	28776.0	34540.208695
Underpriced	19890.0	24183.525534
Underpriced	4680.0	9926.186735
Underpriced	16065.0	19581.902238
Underpriced	6765.0	8545.539309

Figure 15: Price and Revenue Optimization

6 Evaluation

This section explains the results of the seven machine learning models which were trained. The model includes Linear Regression, ElasticNet Regression, Ridge Regression, XGBoost Regressor, CatBoost Regressor, Random Forest Regressor and LightGBM Regressor. All the models are evaluated by calculating four key performance metrics as shown in table 2.

Table 2: Evaluation Metric and Its Purpose

Abbreviation	Performance Metric	Purpose
RMSE	Root Mean Squared Error	Measures the average magnitude of error. Capture larger error more.
MAE	Mean Absolute Error	Measures average absolute differences between predictions and actual values.

R^2	R^2 Score	Indicates proportion of variance in the target explained by the model.
Mean CV-R^2	Mean Cross-Validation R^2	Average R^2 score across cross-validation folds to assess model generalizability.

LightGBM Regressor:
LightGBM outperformed from all other models used in this research across every metric. It comes out with the highest R^2 score of 0.782, which indicates that the model is able to explain 78% of the variance in the target variable. Whereas RMSE comes out to be the lowest at 42.916 and MAE to be 29.060. These performance metrics made the most precise predictions of the Airbnb listing prices. Furthermore, cross-validation R^2 score of 0.781 confirms the model consistency across the whole listing.

XGBoost Regressor:
XGBoost was the second-best performer in this research across every metric. It comes out with the highest R^2 score of 0.770, which indicates that the model is able to explain 77% of the variance in the target variable. Whereas RMSE comes out to be 44.059 and MAE as 29.623. These performance metrics have slightly underperformed in comparison with the LightGBM model. Furthermore, cross-validation R^2 score of 0.757 confirms a slight overfitting of the model.

CatBoost Regressor:
With an R^2 score of 0.765 for variance explanation on Airbnb listing prices, CatBoost performed competitively. Whereas no such difference is presence in the error metric as RMSE comes out with 44.572 and MAE of 30.825. Surprisingly CatBoost demonstrates better generalization than XGBoost (0.769 vs. 0.757), suggesting it was more stable and generalizable across the dataset. Along with that, CatBoost explained a higher variance on training data than XGBoost's test and training. Despite CatBoost's slight underperformance in comparison to LightGBM, it continued to be a prominent competitor in this analysis.

Random Forest:
Random Forest showed moderate predictions on prices of the Airbnb listings and performed below expectations. The R^2 score comes out to be 0.680 whereas RMSE and MAE to be 52.003 and 34.590 respectively. However, a low cross validation score of 0.640, dropping from 0.680 to 0.640 strongly indicates that the model is overfitting and has weaker generalization. It has outperformed basic statistical models like linear regression and ridge regression. Random Forest worked good in terms of baseline comparisons.

Linear Regression:
Linear Regression came out with a baseline performance for predicting Airbnb listing prices. The evaluation metric came with an R^2 of 0.661, RMSE of 53.538, MAE of 35.709. The cross-validation score comes out to be 0.661, suggesting limitation in complexity capture. The model is useful for quick prototyping or for base line predictions, but it lacks flexibility when the data have well-structured non-linear relationships between them. This model performed weaker than ensemble and boosting models, limiting its usefulness for accurate Airbnb pricing forecasts.

Ridge Regression:
Ridge Regression produced similar results like linear regression. This suggests that the data have nonlinear relationships between them. That's the reason that linear models are not performing well on such data. The model came out with a baseline evaluation metric, where R^2 score was 0.661, RMSE and MAE 53.538 and 35.708 respectively. The cross-validation score was 0.661 suggesting that the addition of L2 regularization did not help in improving the generalizing ability of the model or reduce error.

ElasticNet:

ElasticNet underperformed in all models, recording the lowest R^2 score of 0.627, the highest RMSE (56.111), and MAE (37.983). The cross-validation score of 0.603 indicates overfitting and suggests the model generalization ability is very poor. Even with L1 and L2 regularization, model still failed in understanding the complexity of the Airbnb data. While ElasticNet is more appropriate for high-dimensional or sparse datasets, in this instance, it was the least accurate and consistent, making it the least effective model.

```
=== Model Performance Summary ===
                Model    RMSE     MAE      R²  CV Mean R²
0           LightGBM  42.916  29.060  0.782       0.781
1            XGBoost  44.059  29.623  0.770       0.757
5           CatBoost  44.572  30.825  0.765       0.769
6      Random Forest  52.003  34.590  0.680       0.640
2  Linear Regression  53.538  35.709  0.661       0.661
3              Ridge  53.538  35.708  0.661       0.661
4         ElasticNet  56.111  37.983  0.627       0.603
```

Figure 16: Evaluation Metrics of Various Machine Learning Models

The results are summarized in Figure 16. Overall, ensemble-based models like LightGBM, XGBoost and CatBoost have performed better than all traditional linear models and approaches. LightGBM showcased the best result for this research and is reliable for price prediction and optimizations of the Airbnb listings on the basis of with accuracy, generalization abilities and consistency across the data. Feature importance analysis is performed for LightGBM to understand the importance of the feature in price variation as shown in Figure 12.

SHAP Analysis:

For better insights into features which are influencing the Airbnb listing prices, Shap analysis is performed. It helps in explaining the direction of the feature impact on Airbnb, listing whether it is responsible for lifting the price up or dragging the price lower. Two different summary plots are generated as shown in Figure 17, one for all features and the other for amenities only to identify which amenities are responsible for uplifting the prices of the listing. The summary plots explain a visual interpretation of each feature involved in predicting the price. Each dot of the plot represents a single Airbnb listing, and the horizontal position (negative or positive) indicates how much the feature contributes to explaining the price up or down. The right-hand position explains a positive impact whereas the left-hand side indicates a negative impact of price. Whereas color gradient explains the numeric value of the features. Blue represents a low numeric value and red represents a high numeric value. This layer helps that how different values of the feature impact on the Airbnb listing price as well as explains their presence and absence effect on Airbnb pricing. The outcome result of the summary plot is discussed in table 3.

Table 3: SHAP Summary Interpretation

Features	Interpretation	Summary
Accommodates	Strong positive effect on high values and strong negative effect on low values	More capacity means higher price and less capacity means lower price
Bedrooms	Strong positive effect on high values and strong negative effect on low values	More bedrooms mean higher price less bedrooms low price.
Source City	Shap spread along the axis indicates that every city has its own distinguished price market.	Every city has its own pricing paradigm.
Avg bedroom price/city	Shap spread along the axis indicates that every city has its own distinguished average price per bedroom.	Every city has its own average room price with effected the price accordingly.
Review Score Location	Strong positive effects on high values and low negative effect on low values.	High location score means higher price but minimal effect on low score

Room Type	Strong negative effect on high values and low positive effect on low values.	Shared rooms have low price, private and hotel rooms moderate and entire home and apartment have high prices.
Bathrooms	Strong positive effect on high values and strong negative effect on low values.	More bathrooms mean more price and less bathrooms mean low price.
Minimum Nights	Strong negative effect on high values and low positive effect on low values.	More minimum stay refers to high prices and minor effect on price on low minimum prices.
Dishwasher	Consistent positive effect on presence and consistent negative effect on absence.	The presence of dishwasher means high price comparatively with the listing which don't have it.
Lock on Bedroom Door	Consistent positive effect on absence and strong negative effect on presence.	Presence of door lock indicates low price rather than listing which don't have.
Free Street Parking	Strong negative effects on presence and consistent positive effect on absence.	Absence of free parking tends to have higher prices than having one.
Carbon Monoxide Alarm and Smoke Alarm	Consistent positive effect on presence and consistent negative effect on absence.	Presence of CO alarm means higher price than absence.
TV and Wine Glasses	Slight positive effect on presence and slight negative effect on absence.	Presence of TV means higher price than not having one.
Drying Rack, Crib, Bathtub, Elevator, Dryer, Safety, Balcony, Nespresso	Strong positive effect on presence and minor negative effect on absence.	Such amenities can make the listing premium, but absence does not affect price.
Rice maker, split type ductless system, mini fridge	Strong negative effects on presence and no positive effect on absence.	The presence of such amenities has a negative effect on price.

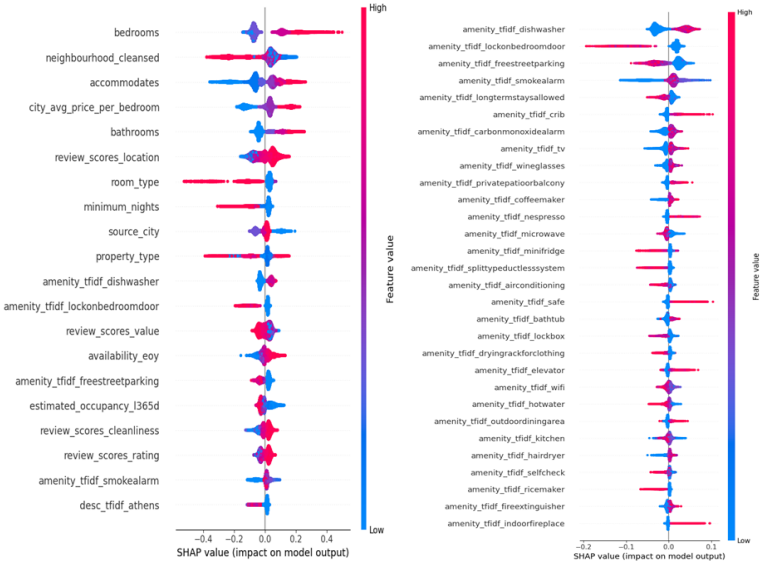


Figure 17: SHAP Summary Plots

Through this analysis hosts and stakeholders get a deep insight into pricing structure of the Airbnb data and take actionable strategic decisions to improve their listing according to the market standards.

7 Conclusion and Future Work

The restatement of core objectives of this research:

1. Which listing features and review ratings are responsible for influencing the Airbnb pricing and how much.
2. Development of a predictive machine learning model which can predict the Airbnb pricing.

3. To identify underpriced and overpriced listings for the price optimization and to maximize the revenue.
4. Provide actionable insights for the host to improve their rental listings.
5. What type of amenities are essential for Airbnb listings and plays an important role in influencing the Airbnb pricing.

The research used dataset from four various global cities (Amsterdam, Berlin, Athens, and Los Angeles). The research undergoes several key processes including data loading and cleaning, exploratory analysis, feature engineering, data modelling and machine leaning modelling and evaluation. In machine learning modelling seven models are used out of which the best performing model is used for price optimization. LightGBM proved to be the best model in predicting the Airbnb pricing in this research. After all, shap analysis is used to interpretate the model prediction and to explain what features and responsible for price inflation and how the different values of the features influence the price.

Success in Addressing the Research Question and Key Findings:

This research has successfully achieved its goals in answering the main core question the research has as well as fulfilled the objectives of this study of Airbnb pricing.

1. The study has identified the key features which are responsible for price variation. Such as number of bedrooms, bathrooms, accommodates, review score for location and amenities like bathtubs, crib, dishwasher, fire alarm and elevator are responsible for major price influence.
2. Amenities like elevators, balcony and Nespresso are the amenities which make the listing premium, and their presence increase the price of the listing.
3. A machine learning model is developed which explains up to 78.2% of the variance in the price of the Airbnb listing (R^2 score of 0.782).
4. It implemented a method to identify the listings as underpriced, overpriced and fairly priced with a threshold of ±15% and suggested the predicted price as new price. And for underprice listing new calculated new potential revenue from price optimization.
5. SHAP analysis identified how each feature impacted the price both negatively and positively. This analysis gave full transparent insights for hosts on how they can improve their listing.

Limitations:

1. Due to lack of time-series data, model assumes the listings have static price and doesn't capture any seasons, events, or any peak and off-peak time periods pricing.
2. The data does not have any type of textual reviews, only numerical review scores were used, limiting emotional and sentiment attributes from customer.
3. The dataset contains much fewer premium listings as compared to normal listings; it makes it harder to train models to predict prices for luxury properties. Somehow, it is catered by adding weights while training.

Future Work:

1. Textual data like reviews, about host, description, amenities, and name can be used as input for deep learning frameworks. In future deep learning models like RNNs can be used to predict the Airbnb listing price. These models could detect finer signals, including sentiment and other textual data, resulting in improvements to prediction and optimization accuracies.
2. Seasonal data can be used for the time series analysis for modelling so it could learn from the historical data on occupancy and pricing to better account for seasonal and event-driven tourism in heavily visited cities.
3. Commercialization through Integration into Platforms The proposed pipeline could be transformed into a SaaS or a plugin for Airbnb hosts that provides real-time pricing evaluation, amenity-based suggestions for improvements, thus unlocking new value for property managers.
4. Fairness and Ethical Pricing Audits More modules could be developed to evaluate discrimination in algorithmic pricing (e.g., by geographical area or demographic indicators) to ensure that suggestions do not perpetuate inequality in a given market.

Summary

In conclusion, the research has met all its objectives and demonstrated and provided comprehensive knowledge that advance ensemble models can predict the Airbnb prices better than any other models. Furthermore, the optimization of revenue enhancement and price fairness equilibrium in Airbnb listings. The provided integration of structured listing data, review scores, amenities, and location-based features of the listing facilitated the construction of a clear, scalable, and efficient modelling pipeline that is advantageous for the Airbnb hosts, the platform, and the research community.

References

Wang, H., 2023. Predicting Airbnb listing price with different models. *Highlights in Science, Engineering and Technology*, *47*, pp.79-86. doi: https://doi.org/10.54097/hset.v47i.8169. [Accessed 03 July 2025]

Hu, C., Huang, R. and Li, H., 2022, December. Prediction and analysis of rental price using random forest machine learning technique take Shanghai and Wuhan for example. In *2022 International Conference on mathematical statistics and economic analysis (MSEA 2022)* (pp. 587-593). Atlantis Press. doi: https://doi.org/10.2991/978-94-6463-042-8_84. [Accessed 09 July 2025]

Samwel, M. (2022) What factors drive the Airbnb listing's prices? *International Business & Economics Studies*, 4(1), pp.26–32. doi: https://doi.org/10.22158/ibes.v4n1p26. [Accessed 02 July 2025]

Mao, X. (2024) Research on the influencing factors of rental house prices. *Transactions on Economics, Business and Management Research*, 10, pp.146–151. doi: https://doi.org/10.62051/cs0pd728. [Accessed 02 July 2025]

Ghosh, J., Maji, K., Mzili, T., Roy, P., Chakraborty, M. and Gupta, S. (2024) Exploring the relationship between rent and flat prices through random forest and grid search. *2024 International Conference on Circuit, Systems and Communication (ICCSC)*, pp.1–5. doi: https://doi.org/10.1109/iccsc62074.2024.10616721.

Thakur, N., Jain, R., Mahajan, A. and Islam, S.M.N. (2022) Deep neural network based data analysis and price prediction framework for Rio de Janeiro Airbnb. *2022 IEEE 7th International conference for Convergence in Technology (I2CT)* pp. 1-7. doi: https://doi.org/10.1109/I2CT54291.2022.9824383. [Accessed 01 July 2025]

Sayyad, S., Saraf, A., Kale, D. and Pardeshi, N. (2023) House price and rent prediction using machine learning. *International Research Journal of Modernization in Engineering, Technology and Science, 5(5), p. 7859.* doi: https://www.doi.org/10.56726/IRJMETS40466. [Accessed 04 July 2025]

Yang, S. (2021) Learning-based Airbnb price prediction model. *2021 2nd International Conference on E-Commerce and Internet Technology (ECIT)*. doi: https://doi.org/10.1109/ecit52743.2021.00068.

Akalin, O. and Alptekin, G.I. (2024) Enhancing Airbnb price predictions with location-based data: a case study of Istanbul. *Annals of Computer Science and Information Systems*, 39, pp.207–212. doi: https://doi.org/10.15439/2024f7603.

Peng, N., Li, K. and Qin, Y., (2020). Leveraging multi-modality data to airbnb price prediction. *2020 2nd International Conference on Economic Management and Model Engineering (ICEMME)* pp. 1066-1071. IEEE. doi: https://doi.org/10.1109/ICEMME51517.2020.00215 [Accessed 07 July 2025].

Zhu, A., Li, R. and Xie, Z., (2020). Machine learning prediction of new york airbnb prices. *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)* (pp. 1-5). IEEE. doi: https://doi.org/10.1109/AI4I49448.2020.00007 [Accessed 07 July 2025].

Malik, A., Hassouna, M. and Togher, M., (2023). Exploring Sustainability in London Airbnb Rentals: A Data-Driven Analysis of Sustainability Keywords Using AI Algorithms. *2023 9th International Conference on Information Technology Trends (ITT)* pp. 84-89. IEEE. doi: https://doi.org/10.1109/ITT59889.2023.10184263 [Accessed 06 July 2025].

Ogundunmade, T.P., Abidoye, M. and Olunfunbi, O.M., (2023). Modelling Residential Housing Rent Price Using Machine Learning Models. *Mod Econ Manag*, *2*. Available at: https://article.innovationforever.com/MEM/20230177.html [Accessed 06 July 2025].

Maheshwari, A., Ranka, M., Malhotra, A., Mishra, R.K. and Basha, M.S.A., (2024), November. Assessing Machine Learning Models for Predictive Analytics in Urban Rental Markets: Rent Price Forecasting in India. *2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS)* pp. 1-9. IEEE. doi: https://doi.org/10.1109/iciics63763.2024.10859514.

Abdul Salam, M.H., Mohd, T., Masrom, S., Johari, N. and Mohamad Saraf, M.H., (2022). Machine learning algorithms on price and rent predictions in real estate: A systematic literature review. Available at: https://ir.uitm.edu.my/id/eprint/65682/1/65682.pdf [Accessed 07 July 2025].

Hussain, M.J., Krishna, P.S., Reddy, J.N., Krishna, J.S.R., Reddy, P.R. and Murali, S., (2024). Developing a Robust Rental Price Prediction System: Insights from Linear Regression, Decision Trees, and Random Forest. *2024 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)* pp. 1-7. IEEE. doi: https://doi.org/10.1109/icses63760.2024.10910581.

Mirg, V., (2022). *Prediction of Property Prices of Dublin Housing Market using Ensemble Learning* (Doctoral dissertation, Dublin, National College of Ireland). Available at: https://norma.ncirl.ie/6229/1/vanimirg.pdf [Accessed 08 July 2025].

Chan, H.R. (2024) Rent price prediction with advanced machine learning methods: a comparison of California and Texas. *Highlights in Science Engineering and Technology*, 85, pp.501–510. doi: https://doi.org/10.54097/84vvv580. [Accessed 05 July 2025]

Bakirarar, B. and Elhan, A.H., (2023). Class weighting technique to deal with imbalanced class problem in machine learning: Methodological research. *Türkiye Klinikleri Biyoistatistik*, *15*(1), pp.19-29. doi: https://doi.org/10.5336/biostatic.2022-93961

Pastukh, O. and Khomyshyn, V., (2025). Using ensemble methods of machine learning to predict real estate prices. *arXiv preprint arXiv:2504.04303*. Available at: https://arxiv.org/abs/2504.04303. [Accessed 12 July 2025]

Murel, J. and Kavlakoglu, E. (2024). 'What is ensemble learning?', *IBM Think*. Available at: https://www.ibm.com/think/topics/ensemble-learning [Accessed 15 July 2025].