CUSP-GX 7023 Applied Data Science

Predictive Analytics for Airbnb Prices

Leveraging Data to Forecast Market Trends

Abstract

As one of the most essential forms of accommodation, the price affects tourists' travel, especially for the price-sensitive clients. Our project aims to address two main objectives: predicting the future prices of each Airbnb listing and forecasting listing prices based on their characteristics. We first conducted data cleaning, EDA, and feature engineering to achieve the goal due to the irregular format of variables and miscellaneous variables. Then, we used time series, machine learning and deep learning to predict the price. This will provide valuable information for tourists, helping them understand pricing dynamics, and assist hosts in setting reasonable prices. Finally, we explained the results and provided some suggestions.

Introduction

In our capstone project, we focus on Airbnb pricing dynamics in New York City, aiming to develop a predictive model that accurately forecasts listing prices. By examining various factors such as accommodation type and seasonal fluctuations, we seek to empower hosts with competitive pricing strategies and provide guests with cost-effective accommodations, thus enhancing travel accessibility and affordability.

Our project employs sophisticated data analytics techniques, including time series analysis and machine learning algorithms like XGBoost and Random Forest, to uncover the nuanced interactions dictating pricing. Through this comprehensive approach, we aim to contribute to urban economics research and offer practical insights for market participants, illuminating the complexities of the New York City accommodation market and paving the way for future developments in peer-to-peer accommodation pricing.

Research Questions

As we all know, many factors affect the price of Airbnb listings. On the one hand, the characteristics of the listings, such as the type, location, and amendments of the listings, will determine the price to a large extent. On the other hand, time changes can also cause price fluctuations. To sum up, prices rise and fall with value. Therefore, we aim to address two main questions: (1) How do Airbnb rental prices in NYC vary seasonally, especially during holidays and extended weekends? (2) What are the primary factors influencing Airbnb prices in NYC, as revealed by feature selection? (3) What factors contribute most significantly to the variation in Airbnb room prices across different neighborhoods, and how accurately can these prices be predicted based on neighborhood and room features?（Key Question)

Literature review

This review explores the determinants of Airbnb pricing through various scholarly contributions. A pivotal study titled "Key Factors Affecting the Price of Airbnb Listings: A Geographically Weighted Approach" by Zhang et al. focuses on Airbnb listings in Metro Nashville, Tennessee, identifying key factors like proximity to the convention center, the number of reviews, and review ratings. It was observed that listings closer to the convention center commanded higher prices, indicating a premium on location within the central area compared to other regions. Additionally, listings with a greater number of reviews and higher rating scores tend to adopt varied and often higher pricing strategies.

Another significant piece of research, "The Sharing Economy and Housing Affordability: Evidence from Airbnb" by Barronet al. examines the impact of Airbnb on housing affordability and market dynamics in different zip codes. The study highlights how a 1% increase in Airbnb listings can lead to increases in rents and house prices, especially in areas with a lower share of owner-occupiers. This suggests that Airbnb contributes to a shift in housing supply from long-term rentals to short-term accommodations, affecting affordability.

In the realm of predictive analytics, the paper "Predicting Airbnb Listing Price with Different Models" by Haoqian Wang presents a comparison of various regression models' effectiveness using a Boston dataset. The Gradient Boosting Regression Model outperformed others with the highest R-squared values, indicating a strong predictive capability for Airbnb listing prices. Other models, like Random Forest and

Linear Regression, also showed good performance, providing a basis for selecting appropriate predictive techniques in the research.

Through these studies, we gather insights into how various factors shape Airbnb pricing strategies, from geographical proximity to significant landmarks to user interactions like reviews. This understanding is crucial as we aim to develop a predictive model for hotel prices in NYC, drawing parallels from Airbnb dynamics to provide comprehensive insights for stakeholders navigating the complexities of the urban accommodation market.

Data and Data processing

A. Data Collection and Data Specification

We collected the data from the Airbnb official website. We chose data in January, 2024 for prediction because the key point is the characteristics of each Airbnb house rather than the changes over time; For the Time Series part, we choose data from 2012 to Oct.2023 as the object.

Now, let's see the details of our data. We had 75 variables, including "price" as one dependent variable and 74 other independent variables. There were several kinds of independent variables: the information about the Airbnb house, the information about the host, information related to dates and days and the information about reviews. Among them, there are some highly correlated variables, such as availability_60 and availability_90. We will conduct data cleaning in the next part.

B. Data Cleaning

We first isolated numeric columns to analyze variables impacting Airbnb pricing, such as accommodation size and number of reviews. Then, we managed missing data by removing columns with over 70% missing values and dropping rows where key variables were missing. Numeric missing values were filled appropriately, and categorical fields like 'neighborhood' were standardized to strings to maintain consistency. Rate fields were converted from strings to floats, and boolean fields were mapped from 't'/'f' to true/false. The 'price' column was stripped of non-numeric characters and converted to a float. Date-related fields were transformed into datetime objects to enable new feature creation, such as calculating days since the last review. Next, we used VIF to remove highly collinear variables, thus to destabilize model estimation. Then, we used One-Hot encoding to transform the categorical variables, because the method helps in handling non-numeric data, allowing models to better predict outcomes without assuming ordinal relationships. Also, we conducted the normalization by applying log transformation to the price variable according to its frequency distribution, skewness and kurtosis. This kind of method corrects skewness, enhancing model accuracy as many algorithms assume normally distributed data. After these transformations, a final sweep removed any incomplete rows and irrelevant columns, finalizing the dataset for predictive modeling. This rigorous data cleaning ensures our dataset is primed for accurate analysis and model development.

C. Exploratory Analysis

For our analysis, I do the correlation matrix aimed at predicting Airbnb prices in Figure-1. It appears that variables such as host_acceptance_rate, accommodates, number of bathrooms, number of bedrooms, number of beds, calculated_host_listings_count, and amenities_count may have a significant impact on price. These columns should be given special attention during the feature selection process for the predictive modeling.

In New York City's Airbnb market, our data shows that Manhattan and Brooklyn have significantly more listings than other boroughs, suggesting a strong influence of location on pricing dynamics in Figure-2. This disparity highlights the importance of exploring how geographical factors affect rental prices in these densely listed areas, providing a critical focus for further detailed analysis.

For the bar chart analysis (Figure-3), I aggregated the total bookings for different room types available on Airbnb. The results clearly indicate that "Entire home/apt" and "Private room" categories are the most popular choices among clients. This trend suggests a high demand for these types of accommodations in the Airbnb market. Understanding these preferences is crucial for hosts when considering how to categorize their listings to maximize occupancy and profitability. And the graph shows us the Room_type will also become a significant variable for prediction models.

Methodology

Feature Engineering

I. Ensemble Methods (XGBoost and Random Forest)

Employing techniques like Recursive Feature Elimination (RFE) with tree-based estimators such as XGBoost and Random Forest, we strategically selected features based on their inherent importance derived from the models. This feature selection method leverages the tree models' ability to rank features based on how well they improve the model's performance, typically measured by the reduction in impurity. RFE works by recursively removing the least important features, then refitting the model to assess the impact of removing those features on model performance, using metrics such as R^2 for evaluation.

This approach helps minimize overfitting and ensures that our models are manageable by retaining only the most contributing features. The use of tree-based methods for RFE is particularly advantageous because these models provide a robust mechanism for feature importance evaluation directly from the learned model structure. This importance is usually calculated through either the decrease in node impurity weighted by the probability of reaching that node (for Random Forest) or gain from each feature when used in trees (for XGBoost). The iterative nature of RFE helps in fine-tuning the selection by iteratively evaluating the impact of removing each feature, thus further enhancing the model’s performance and simplicity.

II. PCA

If there're too many features, we are more likely to meet the overfitting problem, which could lead the models to be inaccurate. Our dataset has 224 features, therefore we use PCA for dimensionality reduction. It can map high-dimensional data into low-dimensional Spaces while preserving the most important information in the data.

From the picture (Figure 4), we can see that 130 components can explain 80% of the variance. Actually, we think the result of the number is large. The main reason is that most of the factors we pick are about the nature of the house itself. These factors are trivial and scattered, and none of them has an absolutely dominant impact on prices. In ideal conditions, we can use the fundamental factors, such as local GDP, crime rate etc., but it is hard for us to acquire regional dimension data. Therefore, we still need to keep 107 factors.

Next, we get the name of the most important feature of each principal component. From the picture (Figure 5), we can see that the room type, facilities and review might be the important factors.

We used XGBoost, Random Forest, and PCA to select features based on their median and mean importance scores. This helped minimize overfitting and ensured that our models were not.

Model Building

I. Time Series Analysis

This report employs time series analysis methods to predict Airbnb price trends over the next 180 days, assisting the platform and hosts make more effective pricing strategies and market decisions. The shared accommodation market changes rapidly as the tourism industry recovers and global mobility increases. Accurately predicting lodging prices is crucial for optimizing resource allocation, formulating marketing strategies, and enhancing customer satisfaction. Key literature indicates that time series analysis is an effective tool for predicting market price fluctuations. The SARIMAX model is widely used in economic data forecasting because it handles seasonal and non-seasonal fluctuations simultaneously.The dataset spans from 2012 to the present day, focusing on daily average prices listed on Airbnb.Accurate pricing forecasts can significantly enhance resource distribution, marketing strategy formulation, and consumer satisfaction in the dynamic shared accommodation market.A multiplicative model (Figure 6) was employed to dissect the price variable into trend, seasonality, and residual components, offering a nuanced view of underlying patterns.

II. Ensemble Methods (XGBoost and Random Forest)

For model development, we selected Random Forest and XGBoost due to their robustness in handling non-linear data relationships:

XGBoost is a gradient-boosting framework that uses tree-based learning algorithms. It is renowned for its performance and speed in classification and regression tasks. Random Forest is an ensemble learning method that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees.

Grid Search Cross-Validation: We utilized grid search with cross-validation to optimize model parameters systematically. This method ensured the selection of the most effective hyperparameters.

Hyperparameter Tuning:

A. Random Forest: We tuned parameters such as n_estimators, max_depth, and min_samples_split to control model complexity and improve generalization.
B. XGBoost: Parameters like max_depth and eta were adjusted to optimize training speed and model performance.

III. Neural Network

In recent years, neural networks have become increasingly popular as machine learning models for classification or prediction. Neural Networks mimic the structures of human neurons to process information, receive feedback through gradients, and make decisions at the end of the network like human neurons. Although single-layer perceptrons are also considered neural networks, NN usually have multiple layers and can have complex structures to take in data while processing them (such as Batch Normalizations), transmit previously learned wisdom to the nest layer (such as RNN), convolute nearby data to form new layers (such as CNN), or even skip layers to save time and space and preserve the most critical features (such as ResNet).

As the previously applied machine learning models in this study, NN can also be used to predict Airbnb prices in other areas, as long as they have the same features as inputs. In this case, we use part of the data from the Greater NYC metropolitan area from the Airbnb website to predict the rest of NYC areas' Airbnb prices. Our predictions can reach 82%, with a fairly simple NN architecture and some standard learning rate schedulers, loss functions, optimizers, and hyperparameters combinations.

We use neural networks to take in the features of Airbnb rooms, including room type, review score value, neighborhood, zipcode, minimum nights, maximum nights, etc., to predict the prices of rooms in the same spatial space and time period. In our experiments, we only applied the best model and hyperparameter combination to the test set after we evaluated the validation set accuracy performance.

All training, validation, and test sets are from the January 2024 dataset. Our data is the official inclusive data from Airbnb for the first month of 2024 January. If we use our model to predict the prices of February or March, there wouldn't be much difference between locations. Instead, any discrepancies would be attributed to the temporal differences. However, temporal differences in housing prices, mainly inflation, are not the

subject of this study. We are more interested in figuring out how these geographical and spatial differences in the rooms in an area would affect prices and make predictions based on existing data.

In our experiments, we utilize four different NN architectures and 4 different schedulers, loss functions, and optimizer combinations. Firstly, we use each scheduler, loss function, and optimizer combinations on the most basic architecture (the basic 5-layer). After running 100 epochs on this architecture with 4 different hyperparameter sets, we compare the validation accuracies and pick the best combination as the "winner." Then, we apply the best combination on the other 3 architectures and compare the validation accuracy results with all 7 experiments to determine which combination and architecture works the best. It turns out that the most efficient model and the most suitable hyperparameters are the basic 5-layer with L1 loss, StepLR scheduler, and Adam optimizer. Notably, the differences in those models' performances were not huge and could be due to chances.

Results & Findings

A. Time Series

The SARIMAX model predicts a rising trend in Airbnb prices from November 2023 to April 2024, suggesting market recovery. In Figure7 and Figure 8.

Analysis of Full Data: Time series modeling established from data from 2012 to the present showed that from November 2023 to April 2024, The SARIMAX model predicts a rising trend in Airbnb prices from November 2023 to April 2024, suggesting market recovery. Airbnb's average price is expected to rise from $176.79 to $186.15, reflecting a moderate market recovery trend.

Analysis from 2022 Data: A slight increase in forecasted prices is observed when the model is restricted to post-2022 data. Time series models from 2022 to the present indicated that prices are expected to rise slightly from $185.18 to $185.26. The confidence intervals indicate that, despite potential fluctuations, the expected price changes are not significant, suggesting that the market is gradually recovering from the impact of COVID-19, yet remains relatively stable.

Visual Presentation: Multiple charts showing actual prices versus predicted prices and the seasonal decomposition of price trends.

B. Ensemble Methods (XGBoost and Random Forest)

The results from Table 1 demonstrate that XGBoost and Random Forest models, especially when combined with feature selection techniques like XGBoost and PCA, yield robust predictions. However, PCA models, despite their high training R^2, performed less effectively on test data, suggesting potential overfitting or loss of critical information during dimensionality reduction.

C. Neural Network

In neural networks' training, we ran the models over validation sets during our training, and the differences were minor. The first basic 5-layer model with this set of hyperparameters consistently performed slightly better. The details are shown as Figure 9 & Figure 10. Along the way, we observed that the architecture with 9 models tends to overfit, and the 5-layer with dropouts performed slightly worse than the basic version, which may be attributed to the regularization strategy we put in place.

Discussion

Limitations on this set of experiments for future reference:

In neural network training, as mentioned above, we get our training, validation, and test set from the same month's dataset in neural net training. During our training, we ran the models over validation sets, and the differences were minor and the first basic 5-layer model with SGD parameters (consistently performed slightly better). The details are shown as follows:

Since the dataset is not large, we acknowledge that it has all available Airbnb room information and pricing data we could find online. We can see how the models are doing through validation set accuracy performances, usually averaging around 81%. The model training could perform better if we have a larger available dataset.

However, just because we do not have a large dataset does not mean that such models and our results are not transferable. Since we do not have any new NYC area Airbnb pricing data from January 2024 at hand so far, it wouldn't make much sense to apply it to the following months' data. Nonetheless, for future research, researchers could introduce crime rate, education quality, and environmental data that can be associated and scored with different NYC neighborhoods. As long as these "universal scores" can be determined in neighborhoods in other urban areas, such as Philadelphia or Boston, our models and chosen hyperparameters can still be utilized to prejudice Airbnb prices in a completely different city.

Conclusion & Recommendations

From the dimension of time, we can know that the forecast results of the Time series model support the view that market prices will continue to warm up but also indicate potential market instability. The width of the confidence intervals reveals potential market volatility risks. This study shows that the SARIMAX model effectively predicts seasonal and trend changes in Airbnb prices, providing valuable future perspectives for market participants; From the dimension of each room, we think that the hosts may consider adjusting their rental pricing strategies based on the forecast results, especially during predicted price peaks. The Airbnb platform should strengthen its monitoring of price fluctuations and assist hosts and guests in managing expectations through platform policies. In the future, we encourage researchers to find other "measurable" and "numeric" features that are transferable, such as the "crime rate," "education score," and "vegetation score" associated with certain types of neighborhoods, and so we can use these metrics associated with neighborhoods to do further study and predict the housing prices with more scientific evidence.

Reference

Zhang, Z., Chen, R. J., Han, L. D., & Yang, L. (2017). Key factors affecting the price of Airbnb listings: A geographically weighted approach. *Sustainability*, *9*(9), 1635. https://www.mdpi.com/2071-1050/9/9/1635

Barron, K., Kung, E., & Proserpio, D. (2018). The Sharing Economy and Housing Affordability: EvidKey Factors Affecting the Price of Airbnb Listings: A Geographically Weighted Approach from Airbnb. *EC*, *5*. https://www.semanticscholar.org/paper/The-Sharing-Economy-and-Housing-Affordability%3A-from-Barron-Kung/6a73fc4c634c1a6a01bc520ad831fb647b349219

Wang, H. (2023). Predicting Airbnb Listing Price with Different models. *Highlights in Science, Engineering and Technology*, *47*, 79-86. https://drpress.org/ojs/index.php/HSET/article/view/8169

Inside Airbnb (2024) Inside Airbnb: Get the data http://insideairbnb.com/get-the-data/

NYC Open Data(2023) NYPD Hate Crimes.https://data.cityofnewyork.us/Public-Safety/NYPD-Hate Crimes/bqiq-cu78/about_data

NYC Open Data(2018) Citywide Crime Statistics. https://data.cityofnewyork.us/Public-Safety/Citywide-Crime-Statistics/c5dk-m6ea/about_data

NYC Open Data(2023) DOF Condominium Comparable Rental Income in NYC . https://data.cityofnewyork.us/City-Government/DOF-Condominium-Comparable-Rental-Income-in-NYC/9ck6-2jew/about_data

NYC Open Data(2020) Neighborhood Tabulation Areas (NTAs). https://data.cityofnewyork.us/City-Government/2020-Neighborhood-Tabulation-Areas-NTAs-Tabular/9nt8-h7nd/about_data

Walk Score(2024) Living in New York.https://www.walkscore.com/NY/New_York

Role of members

Siyu Miao: Report (Introduction, Research Question, Conclusion, Perfect Report) Code (Data Cleaning and Exploratory Analysis)

Yuewei Shi: Data Collection and Data Specification, PCA

Yajie Zeng: Time Series Model and Price Predict

Simao Chen (Alice): Data Cleaning, Neural Networks, Results, Discussion, Conclusion and Recommendations

Jiani Lyu：Literature Review, Data Cleaning

Chelsea Liu: Data Preprocessing and Analysis, Feature Engineering, XGBoost, Random Forest

Appendices

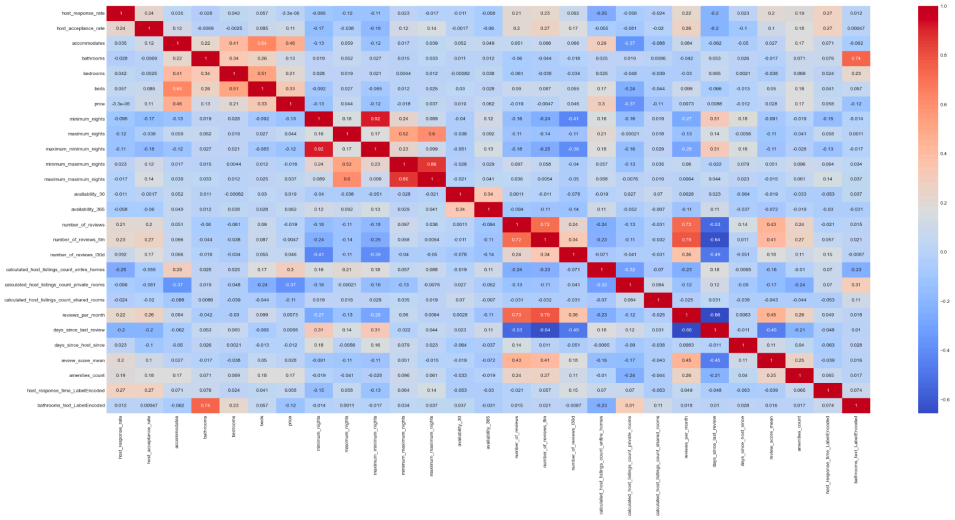

Figure1：correlation matrix

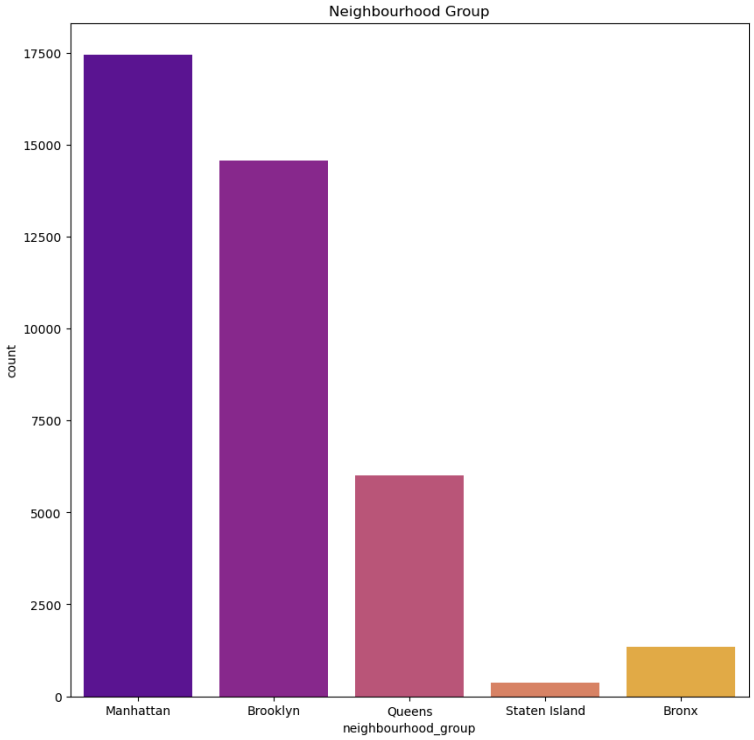

Figure 2：New York City's Airbnb market

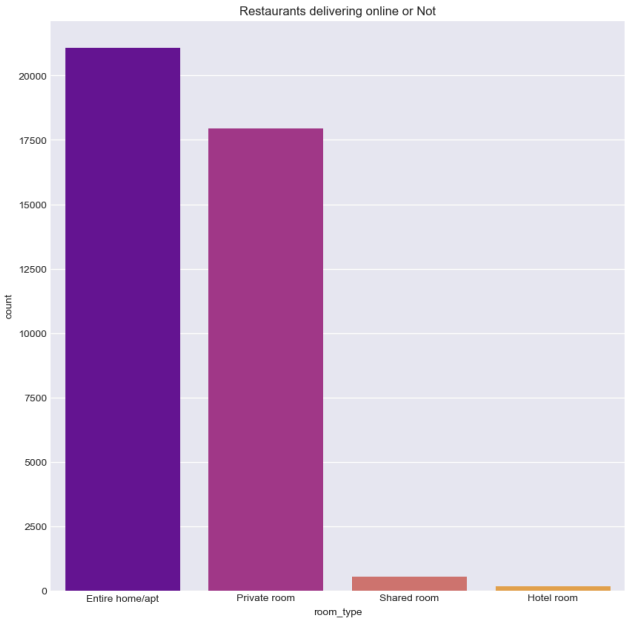

Figure 3: Different room types available on Airbnb

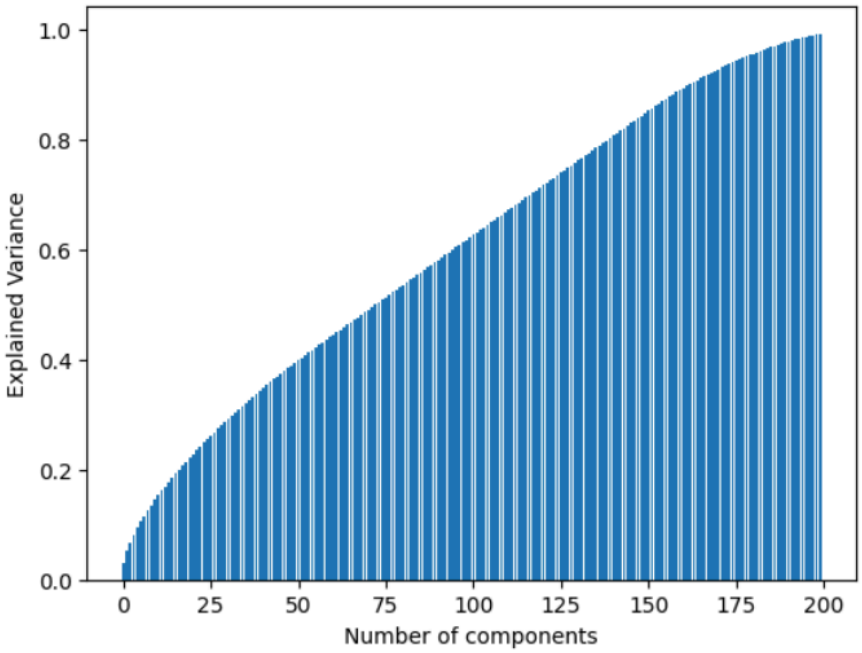

Figure 4: Number of Components

Component	Factor
0	Cheap_room_type
1	reviews_per_month
2	bathrooms_text_labeblEncoded
3	room_type_Shared_room
4	expensive_room_type

Figure 5: Room Type

Feature Selection Method	Model	R² (Training)	MSE (Training)	RMSE (Training)	MAPE (Training)	R² (Testing)	MSE (Testing)	RMSE (Testing)	MAPE (Testing)
None	Random Forest	0.9695	0.0190	0.1378	0.0190	0.7884	0.1369	0.3700	0.0529
None	XGBoost	0.9019	0.0610	0.2471	0.0371	0.7953	0.1325	0.3639	0.0536
Random Forest	Random Forest	0.9705	0.0184	0.1356	0.0188	0.7901	0.1358	0.3686	0.0528
Random Forest	XGBoost	0.9592	0.0254	0.1594	0.0240	0.8068	0.1250	0.3535	0.0519
XGBoost	Random Forest	0.9705	0.0184	0.1355	0.0188	0.7887	0.1367	0.3697	0.0528
XGBoost	XGBoost	0.9461	0.0336	0.1832	0.0276	0.8061	0.1254	0.3542	0.0518
PCA	Random Forest	0.9653	0.0216	0.1468	0.0209	0.7557	0.1581	0.3976	0.0578
PCA	XGBoost	0.9806	0.0120	0.1097	0.0166	0.7528	0.1599	0.3999	0.0586

Table 1: Model Performance Summary

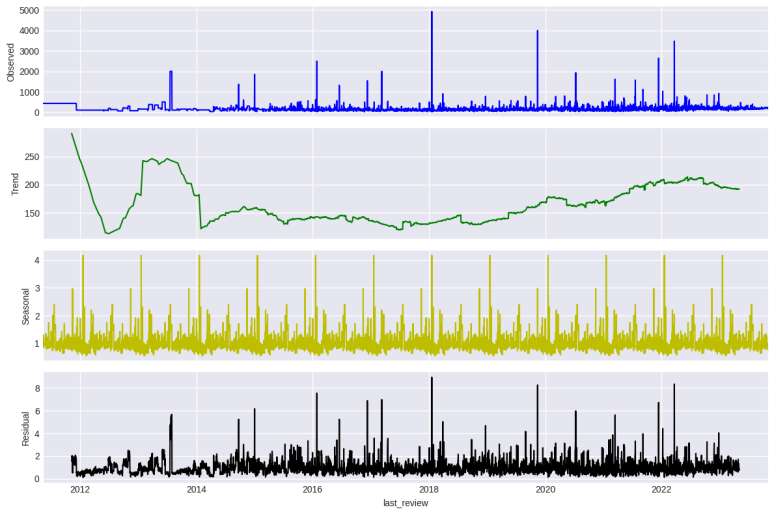

Figure 6: Time Series Trend

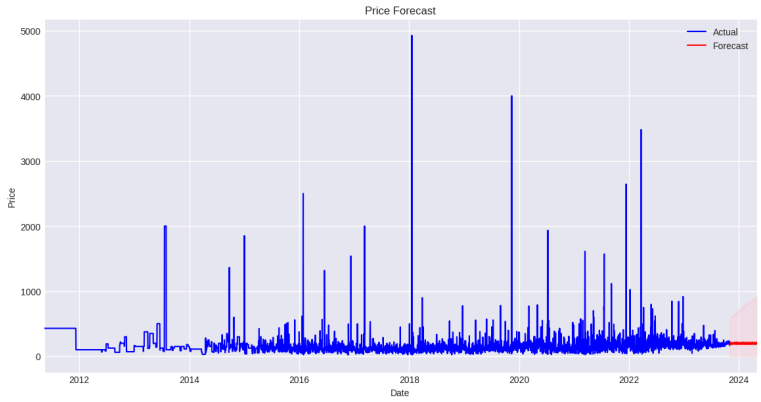

Figure 7: Time Series Predict next 180 days from 2012 to now

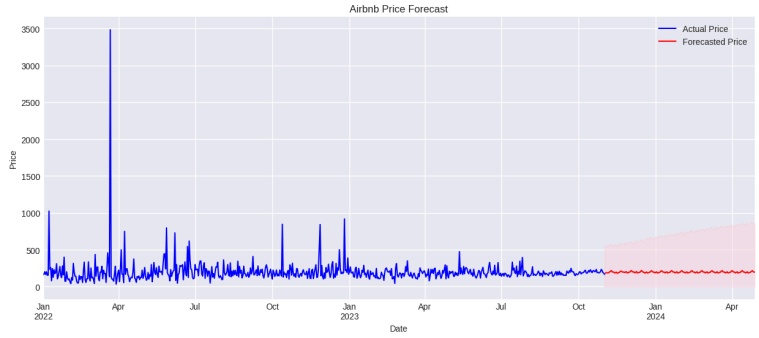

Figure 8: Time Series Predict next 180 days from 2022 to now

feature selection method	model	training				testing			
		R2	MSE	RMSE	MAPE	R2	MSE	RMSE	MAPE
None	Random Forest	0.9695	0.019	0.1378	0.019	0.7884	0.1369	0.37	0.0529
	XGBoost	0.9019	0.061	0.2471	0.0371	0.7953	0.1325	0.3639	0.0536
Random Forest	Random Forest	0.9705	0.0184	0.1356	0.0188	0.7901	0.1358	0.3686	0.0528
	XGBoost	0.9592	0.0254	0.1594	0.024	0.8068	0.125	0.3535	0.0519
XGBoost	Random Forest	0.9705	0.0184	0.1355	0.0188	0.7887	0.1367	0.3697	0.0528
	XGBoost	0.9461	0.0336	0.1832	0.0276	0.8061	0.1254	0.3542	0.0518
PCA	Random Forest	0.9653	0.0216	0.1468	0.0209	0.7557	0.1581	0.3976	0.0578
	XGBoost	0.9806	0.012	0.1097	0.0166	0.7528	0.1599	0.3999	0.0586
Basic 5-layer NN		0.8749	0.0783	0.2798	0.0360	0.7116	0.1886	0.4342	0.0624

Table 2: Feature selection method Model Performance Summary

We use the best loss function, optimizer and lr scheduler combination to model from our previous training and validation results (from experiment 4) to model the test set

- **input_size**: Dimensionality of the input features in the training data. (106)
- **model**: Instance of the `AirbnbPricePredictor` model with input size 106.
- **criterion**: Loss function used for optimization, specifically nn.L1Loss with reduction method set to mean.
- **optimizer**: SGD optimizer used for model parameter optimization with a learning rate of 0.01 and momentum of 0.9.
- **scheduler**: CyclicLR, with max of 0.01 and base_lr of 0.001.
- **epochs**: Total number of training epochs set to 100.
- **threshold**: Convergence threshold set to 0.1, indicating a 10% change in loss between epochs for early stopping or other convergence criteria.

Results with test set:

- **Hit 81.5 test acc around epoch 80, 83.5% val acc around epoch 70, where train acc hits almost 93% around epoch 100** (Currently Best)

Figure 9: neural networks Data description

```python
class AirbnbPricePredictor(nn.Module):
    def __init__(self, input_size):
        super(AirbnbPricePredictor, self).__init__()
        self.fc1 = nn.Linear(input_size, 256)
        self.fc2 = nn.Linear(256, 128)
        self.fc3 = nn.Linear(128, 64)
        self.fc4 = nn.Linear(64, 32)
        self.fc5 = nn.Linear(32, 1)

    def forward(self, x):
        x = F.relu(self.fc1(x))
        x = F.relu(self.fc2(x))
        x = F.relu(self.fc3(x))
        x = F.relu(self.fc4(x))
        x = self.fc5(x)
        return x

    def predict(self, X):
        print(X.shape)
        self.eval()
        with torch.no_grad():
            if isinstance(X, np.ndarray):
                X = torch.from_numpy(X).float()

            print(X.shape)
            if X.dim() == 1:
                X = X.unsqueeze(0)
                print(X.shape)

            output = self(X)
            output = output.squeeze()
            print(output.shape)
            return output.numpy()
```

Figure 10: neural networks' code