AIRBNB NYC 2019 ANALYSIS REPORT

Using Classical Machine Learning

Prepared by:
Zahra Ballaith
Opeyemi Adeniran
Zaki Alhifzi
Rami Elramadi

Table of Contents

Airbnb NYC 2019 Price Analysis Report

Introduction

Since 2008, Airbnb has revolutionised the short-term rental market by enabling homeowners to list accommodations worldwide. Understanding the factors that influence Airbnb pricing is essential for both hosts seeking to optimise their earnings and the platform itself in enhancing its strategy. This report aims to identify key variables that affect listing prices in New York City using classical machine learning methods and provide insights through data exploration and model-driven analysis. Additionally, we will apply a classical machine learning method using the provided dataset to understand the factors influencing pricing and to segment properties for targeted pricing strategies. This involves training and evaluating regression models (Linear Regression) to predict prices and applying K-Means clustering to group similar properties. The analysis should include data preparation, feature scaling, model training and evaluation, model interpretation, clustering analysis, visualisation of clusters, and analysis of cluster characteristics (Mahfoudi, 2024).

Business Question

What factors influence Airbnb listing prices in NYC, and how can hosts optimise their pricing strategy based on location, room type, and demand trends?

Exploratory Data Analysis (EDA)

1.1 Overview of the Dataset

The dataset used for this analysis is the publicly available **AB_NYC_2019.csv**, which contains Airbnb listing data for New York City in 2019. The dataset includes **48,895 entries** with **16 variables**, such as:

- **price:** nightly listing price (target variable)
- **neighbourhood_group:** the borough of NYC (e.g., Manhattan, Brooklyn)
- **room_type:** type of listing (e.g., entire home, private room)
- **minimum_nights:** minimum stay requirement
- **availability_365:** availability throughout the year
- **reviews_per_month:** average reviews per month
- **calculated_host_listings_count:** number of listings by the same host

1.2 Data Cleaning & Preprocessing

- Columns such as id, name, host_id, host_name, and last_review were dropped as they are either identifiers or not relevant to our price prediction.
- Missing values in reviews_per_month were imputed with 0, assuming no reviews in those cases.

```
# drop 'id', 'host_id', 'name', 'host_name', 'last_review'
df.drop(['id', 'host_id', 'name', 'host_name', 'last_review'],
axis=1, inplace=True)

# fill missing values in 'reviews_per_month' with 0
df['reviews_per_month'].fillna(0, inplace=True)
```

- Outliers were removed: listings with price >= $1000, and minimum_nights > 30 were excluded to focus on short-term, realistic stays.

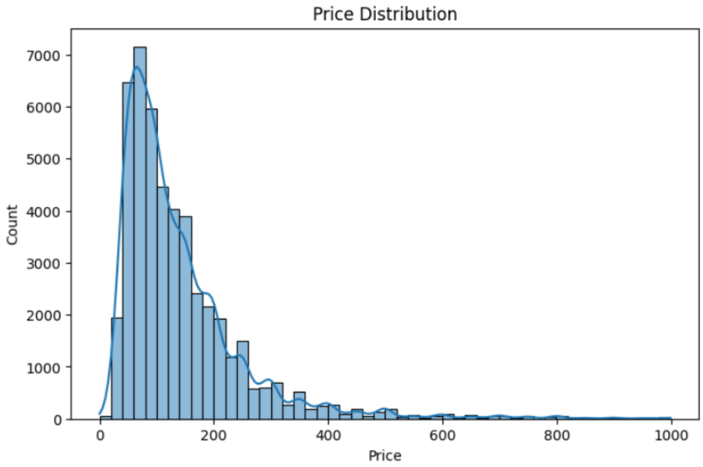
- `#remove the outliners in price & minimum_nights`
- `df = df[df['price'] < 1000]`
- `df = df[df['minimum_nights'] <= 30]`

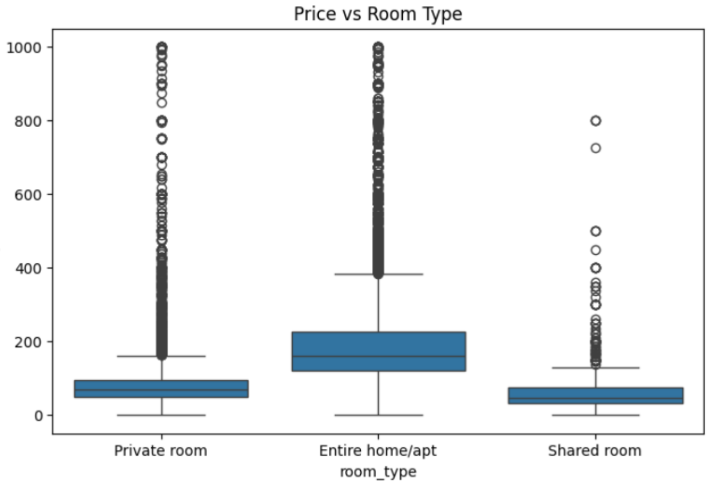
1.3 Descriptive Statistics & Visualisation

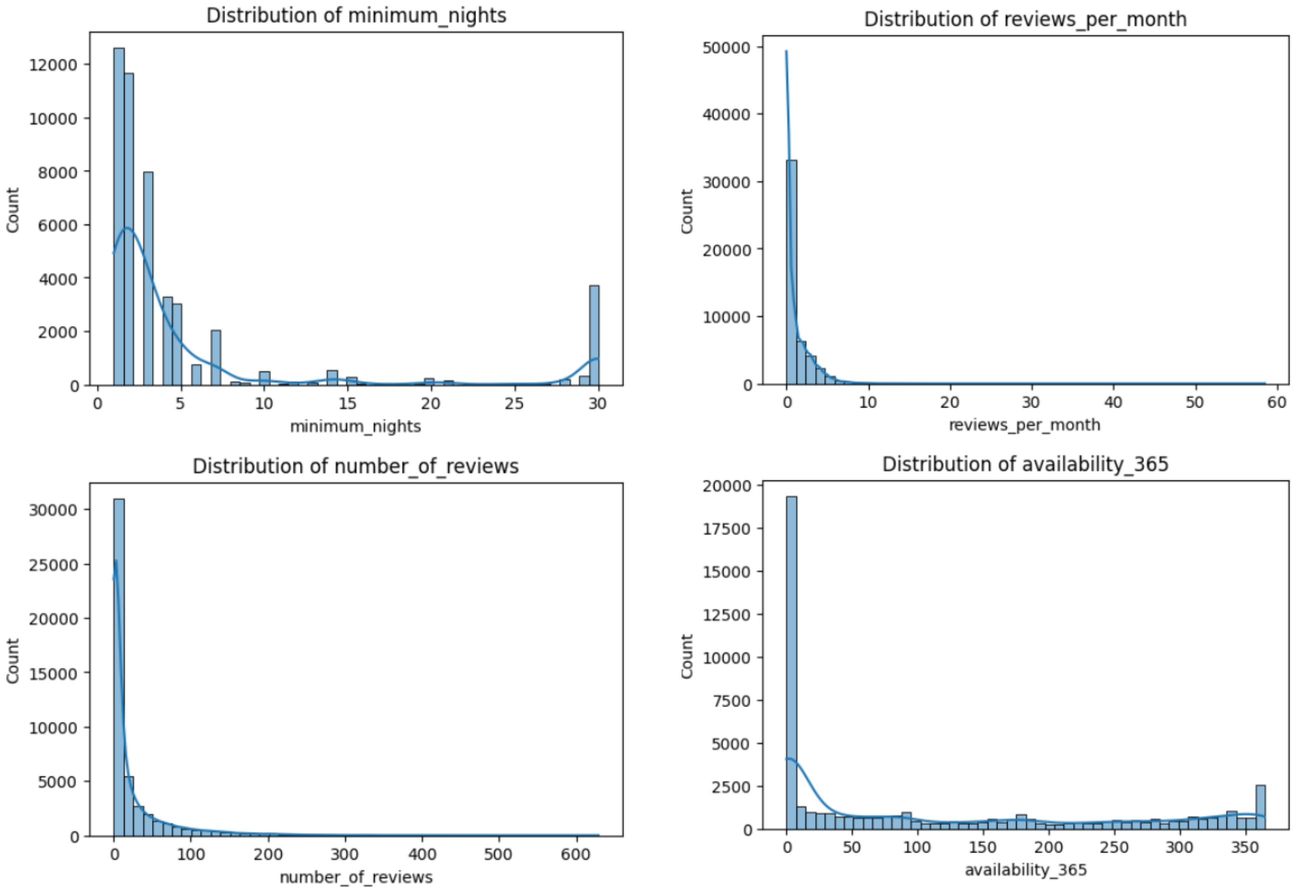
1.3.1 Distribution Analysis

Below is the distribution analysis for numerical variables

- **Price** is right-skewed, with most listings priced below $200 per night.
- **Minimum nights** has a large number of 1-night listings but also contains long-term outliers.
- **Room type** distribution shows that "Entire home/apt" is the most common, followed by "Private room" (Wang, 2024).



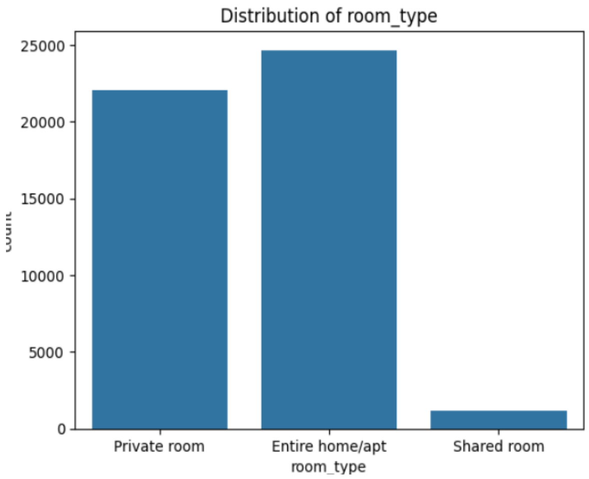


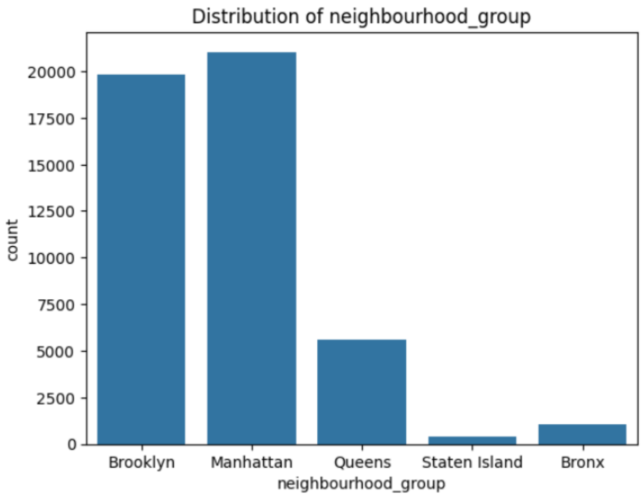


Below are the categorical variables' distributions

```
# Count plots (Categorical Variables Distributions)
for col in ['room_type', 'neighbourhood_group']:
    sns.countplot(x=col, data=df)
    plt.title(f"Distribution of {col}")
    plt.show()
# Check unique values
print(df['room_type'].value_counts())
```

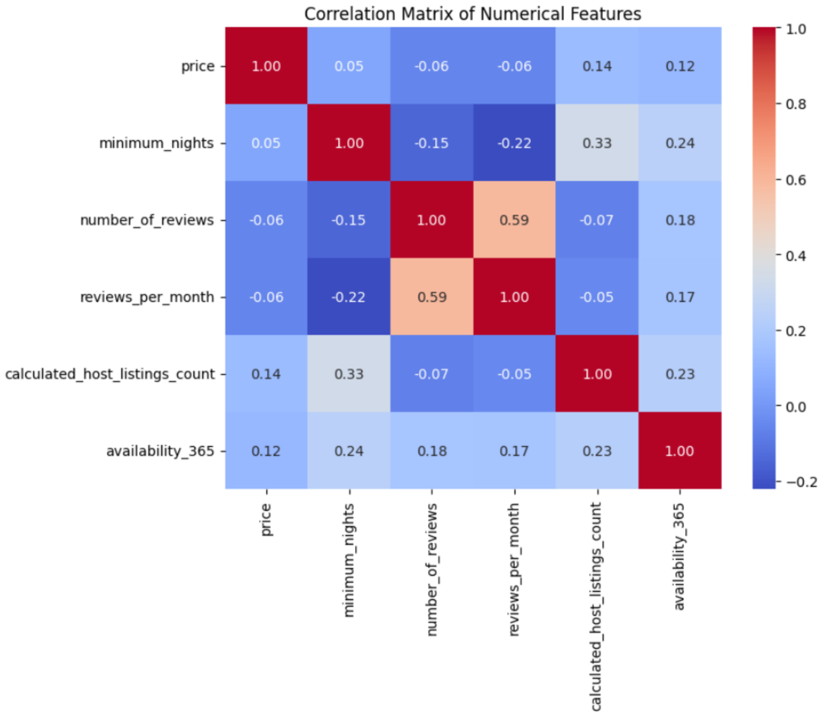
- `print(df['neighbourhood_group'].value_counts())`





1.3.2 Correlation Matrix

- A heatmap of numerical features showed weak linear correlations with price.
- Notably:
  - ➢ reviews_per_month and number_of_reviews had a moderate positive correlation (~0.7)
  - ➢ price had very low correlation with individual features but was best modelled with combinations (Camatti et al., 2024).



- neighbourhood_group and room_type were one-hot encoded for model readiness.
- Example encoded columns: neighbourhood_group_Manhattan, room_type_Private room

The analysis revealed that price is not strongly linearly correlated with any single feature; rather, it appears to be influenced by a combination of variables acting together. Notably, room type plays a critical role, with entire homes and apartments consistently commanding significantly higher prices compared to private or shared rooms. Geographic location also emerged as a pricing factor, listings in Manhattan and Brooklyn are typically more expensive than those located in Queens or the Bronx. Additionally, listings with higher availability and more frequent reviews per month tend to indicate active and popular properties, suggesting that visibility and guest engagement are essential components of a listing's success (ProjectPro, 2024).

Machine Learning Model

2.1 Linear Regression

Feature Preparation

- Independent variables (X) and target (y = price) were selected
- Categorical variables (neighbourhood_group, room_type) were one-hot encoded
- Features were scaled using StandardScaler

Target Variable (y):

The goal of the regression model is to **predict the price** of an Airbnb listing. Therefore:

```
y = df_encoded['price']
```

- ➢ price is a **continuous numeric variable**, which makes it ideal for regression tasks.
- ➢ It represents the nightly cost set by hosts and is influenced by location, property type, availability, reviews, etc

Feature Variables (X):

These are the variables we believed could influence or explain the variation in "price"

```
selected_features = [

    'minimum_nights',

    'number_of_reviews',

    'reviews_per_month',

    'calculated_host_listings_count',

    'availability_365',

    # encoded variables:
```

```
    'neighbourhood_group_Brooklyn',
    'neighbourhood_group_Manhattan',
    'neighbourhood_group_Queens',
    'neighbourhood_group_Staten Island',
    'room_type_Private room',
    'room_type_Shared room']
X = df_encoded[selected_features]
```

Why These Features Were Chosen:

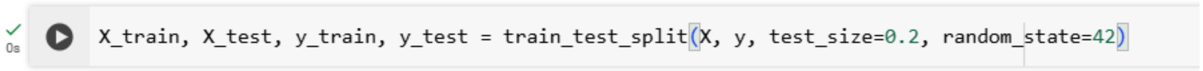
Feature	Reason for Inclusion
minimum_nights	Affects total cost, longer stays might discourage short-term bookings
number_of_reviews	Reflects popularity or booking frequency
reviews_per_month	Indicates listing activity and engagement
calculated_host_listings_count	Hosts with many listings may price differently due to scale
availability_365	Full-year availability suggests potential for more revenue, or high competition
neighbourhood_group	Location is a major price determinant (Manhattan vs. Staten Island, etc.)
room_type	Entire apartments typically cost more than shared/private rooms

***Note**: neighbourhood_group and room_type were originally categorical, so we used **one-hot encoding** to convert them into numerical columns.*

Train-Test Split

Before training the model, we needed to split the dataset into a **training set** and a **testing set**. This allows the model to learn from one part of the data and then be evaluated on unseen data.

```
from sklearn.model_selection import train_test_split
```



```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

x: Our independent variables (features), which include both numerical fields (like reviews_per_month) and one-hot encoded categorical fields (like room_type_Shared room).

y: Our dependent variable, which is the listing price.

We used train_test_split with test_size=0.2 to reserve 20% of the data for testing. This ensures that the model's performance is not just memorised but generalizable to new data.

We initialised a Linear Regression model and fit it to the training data:

```
from sklearn.linear_model import LinearRegression

lr = LinearRegression()

lr.fit(X_train, y_train)
```

This step computes the best-fit line across the feature space that minimises the error between the actual and predicted prices in the training data. The algorithm learns the coefficients for each feature to linearly approximate the price.

Model Prediction and R^2 Score

After training, we predicted prices for the test data:

```
y_pred = lr.predict(X_test)

from sklearn.metrics import r2_score

print("R2 Score:", r2_score(y_test, y_pred))
```

The R^2 score was around (~0.21–0.30), which indicates that the model explains only 21% of the variation in listing prices. This suggests that a simple linear model is insufficient to capture the complexity and non-linearity in the dataset. Multiple refinement strategies were applied to validate the model's behaviour and explore possible performance improvements:

Train-Test Split Adjustments
The default 80/20 split was adjusted in a different ratio in hopes of improving the learning outcome with more training data. However, the R^2 score remained mostly unchanged, suggesting that the model’s limitation lies in feature relationships rather than the amount of training data.

Encoded Feature Format Correction
It was observed that the one-hot encoded variables, such as room_type_Private room and neighbourhood_group_Manhattan, were stored as True/False Boolean types. This was corrected by transforming to binaries; however, this did not affect performance significantly, results still show the same.

To enhance the performance of future pricing models, it is advisable to adopt more advanced machine learning techniques such as Random Forest or XGBoost, which are capable of modeling non-linear patterns more effectively than linear regression. Additionally, feature engineering should be extended to include contextual variables like host tenure, seasonal demand, location-based amenities, and external market trends.

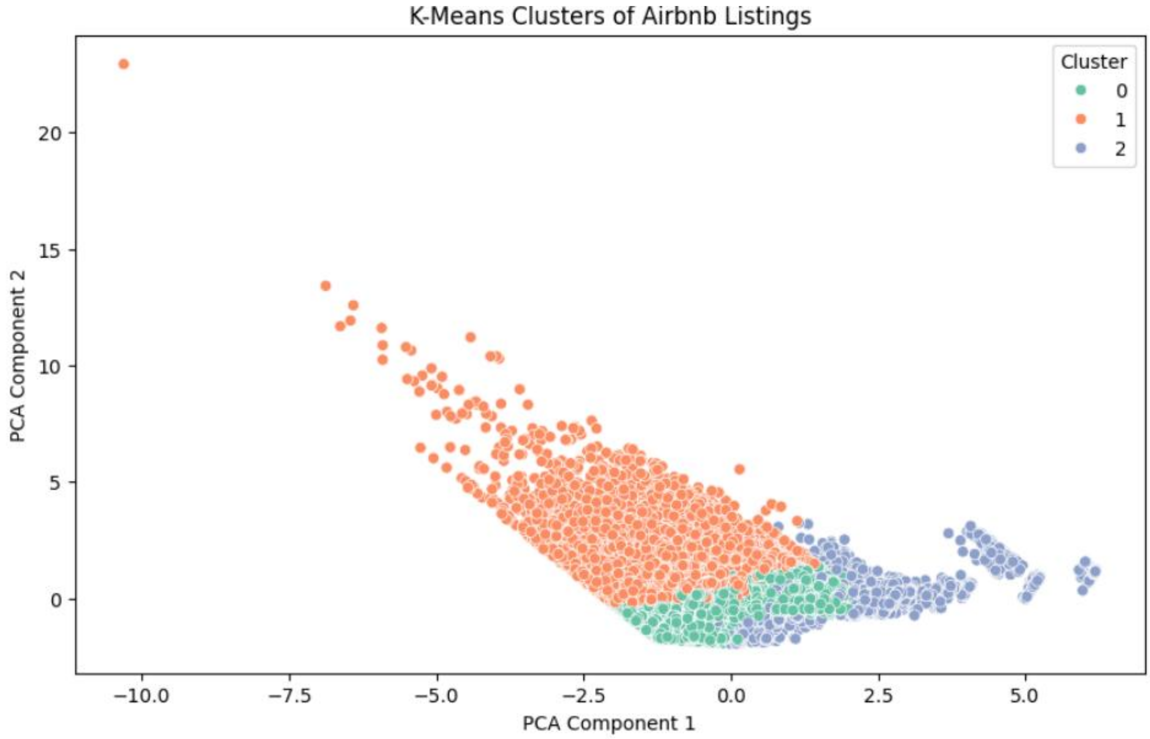
2.2 K-Means Clustering

Clustering was applied to segment Airbnb listings into groups based on their operational and behavioural characteristics, such as availability, reviews, and hosting patterns. Unlike regression, which predicts price, clustering reveals natural groupings in the data. This helps Airbnb:

- Identify distinct market segments (examples: budget listings vs. professional hosts)
- Tailor pricing strategies and marketing efforts for each cluster
- Improve the personalisation of guest recommendations

By grouping similar listings, clustering supports smarter business decisions and a deeper understanding of host and guest behaviour. We have chosen here to apply K-Means which is an unsupervised algorithm that groups similar data points into K clusters. It's helpful for Airbnb because it will reveal pricing tiers or property segments, hosts can also understand which group their listing belongs to, and Airbnb can tailor recommendations, promotions, or fees based on the cluster.

Findings from K-Means Clustering *(below is the visualization of the three clusters resulted)*



➢ **Cluster Distribution:**

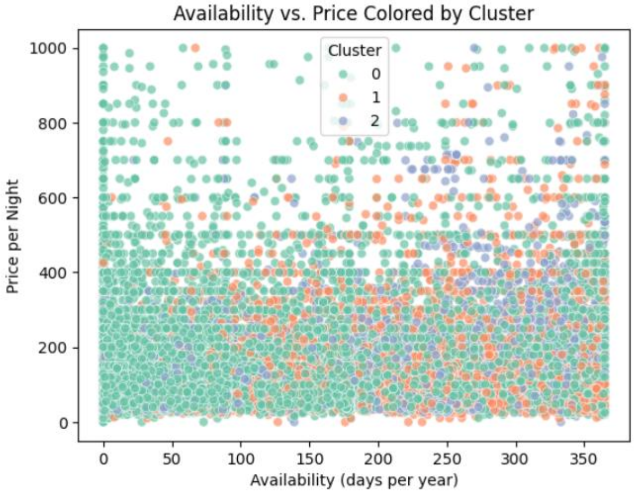
The dataset was grouped into three distinct clusters, with each cluster representing a meaningful segment of listings based on their review count, availability, and other features.

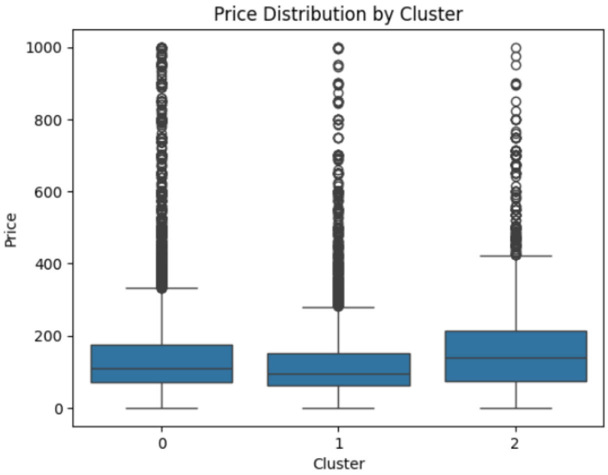
➢ **Cluster Profiles:**

Cluster 0: Listings with lower availability and fewer reviews, likely new or less active hosts.

Cluster 1: Balanced availability and moderate engagement, possibly mid-tier hosts or average performers.

Cluster 2: Listings with high review volume, high availability, suggesting professional or high-demand hosts.





To evaluate and check if our clusters were meaningful, we have used the **Sillhouettee Score**. We observed a score around 0.4, indicating that the clusters are moderately well-separated, a reasonable result for real-world, high-variance data like Airbnb listings.

Airbnb can promote Cluster 0 listings more aggressively to improve visibility. Cluster 2 could be monitored for dynamic pricing adjustments, as demand is already high. The segmentation enables targeted strategies based on operational behaviour rather than just location or price.

Recommendations for Price Strategies

Our analysis found that no single variable linearly determines price, but rather, combinations of factors such as room type, availability, reviews, and location collectively shape pricing outcomes. Based on these insights, we recommend the following business actions:

- Cluster-based host support: Listings with lower engagement (e.g. few reviews or low availability) should receive targeted onboarding and promotional tools to improve performance, while high-performing listings can serve as reference models for successful pricing and engagement strategies.
- Adaptive, data-driven pricing: Instead of fixed pricing logic, Airbnb should introduce cluster-aware dynamic pricing that adjusts based on a listing's operational behavior, offering strategic rate recommendations that account for demand signals, availability, and review trends.
- Search and marketing optimization: Clustering also enables Airbnb to personalize search rankings, marketing campaigns, and seasonal promotions by surfacing listings from underperforming clusters when appropriate or boosting visibility for high-demand areas (McKinsey & Company, 2022).

In short, these findings empower Airbnb and its hosts to make smarter pricing decisions and adapt to demand patterns, room types, and market behaviour, directly addressing the core of the research question and supporting a more balanced, profitable marketplace.

Conclusion

This project utilized classical machine learning techniques to analyse Airbnb listings in New York City, focusing on both price prediction and behavioural segmentation. The application of a Linear Regression model demonstrated limited success, returning a low R^2 score and indicating that listing prices cannot be effectively predicted through simple linear relationships. This result underscores the complexity of Airbnb's pricing mechanisms, which are likely driven by non-linear and multifactorial influences. In contrast, the use of K-Means clustering proved valuable in uncovering natural groupings among listings. These clusters reflected differences in host engagement, availability, and customer interaction, offering deeper operational insights. While regression fell short in pricing accuracy, clustering enabled meaningful segmentation, which can support Airbnb's strategic planning beyond predictive modeling.

References

Camatti, M. et al. (2024). Predicting Airbnb pricing: a comparative analysis of artificial intelligence and traditional approaches. Available at: https://www.researchgate.net/publication/380371567_Predicting_Airbnb_pricing_a_comparative_analysis_of_artificial_intelligence_and_traditional_approaches

Mahfoudi, N. (2024). Is Your Airbnb Pricing Strategy Optimized? Exploring Insights from Data Science. Available at: https://medium.com/@naoufal51/is-your-airbnb-pricing-strategy-optimized-exploring-insights-from-data-science-d54e8f969ecc

ProjectPro (2024) How Data Science increased AirBnB's valuation to $25.5 bn?. Available at: https://www.projectpro.io/article/how-data-science-increased-airbnbs-valuation-to-25-5-bn/199

Wang, S. (2024) Optimizing Airbnb Pricing Strategies: A Data Science Case Study (Part 1: EDA). Available at: https://medium.com/@stellawym/optimizing-airbnb-pricing-strategies-a-data-science-case-study-part-1-c5691b79d005

McKinsey & Company (2022). Personalization: The Next Frontier in Customer Experience. Available at: https://www.mckinsey.com/business-functions/marketing-and-sales