```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
<SparkContext master=yarn appName=livy-session-0>
```

Out[22]: Click here to toggle on/off the raw code.

Image 1. Title.

Table of Contents

Abstract

This project embarked on a creation of a model that would. predict the occupancy rates of Airbnb listings by leveraging machine learning techniques on preprocessed and feature-engineered data derived from two datasets: listings and reviews. Initially, feature derivation was performed to create actionable features such as average reviews per month, host experience, and average price per night. This was followed by filtering and preprocessing the raw data to eliminate extreme and impossible values, ensuring data integrity. Subsequently, outlier analysis was conducted to refine the dataset further.

A robust machine learning model was then trained to predict the occupancy rates, significantly outperforming the baseline model. The model's predictions were further analyzed to determine feature importance, offering valuable insights into the contributions of each feature to the occupancy rate. This analysis provides actionable intelligence for optimizing Airbnb listings, ultimately aiding hosts in improving their occupancy rates. The results underscore the efficacy of using advanced data processing and machine learning techniques in understanding and predicting occupancy trends in the short-term rental market.

Problem statement

Ever since it's inception, Airbnb has seen exponential growth in it's business, with the number of listings going from less than 200,000 in 2016, to over 1 million in 2023 in the United States alone. The platform has become a lucrative venture for many property owners, giving them a chance to gain substantial passive income by renting out their living spaces. However, with this growth have come problems. Market saturation over the years has led to much lower occupancy rates for many, and with such a vast variety of potential reasons, it is getting tougher for the owners to stay competitive in today's market.[1]

Given this context, the aim of this project is to develop a machine learning model that predicts Airbnb occupancy rates by leveraging data from Airbnb listings and reviews. By identifying key features that influence occupancy, the model can provide actionable insights for hosts to optimize their listings, improve guest satisfaction, and ultimately increase their occupancy rates. This study also aims to highlight the most important factors contributing to occupancy variations, helping hosts to make data-driven decisions and enhance their rental strategies, in turn helping the company generate more revenue.

Motivation

Airbnb has become a powerhouse in the travel and accommodation industry, generating substantial revenue in 2023 of over 9 billion dollars. Even in 2024, the company surpassed it's previous year Q1 revenue and is on target to reach a new high for the financial year. [2] The company's impressive financial performance showcases its potential as a lucrative venture, both for the company and for property owners who list their spaces on the platform. With such a strong revenue base, it is clear that there are already significant opportunities for income generation.

Given this context, the authors saw an opportunity to tackle a prevalent problem within Airbnb's ecosystem—predicting and optimizing occupancy rates. By addressing this challenge, we aim to help property owners maximize their earnings and, in turn, contribute to Airbnb's overall profitability. Understanding the factors that influence occupancy rates and leveraging advanced machine learning techniques to predict these rates can provide actionable insights for hosts. These insights will enable them to make data-driven decisions, ultimately enhance their revenue.

In essence, the motivation stems from the realization that improving occupancy rates not only benefits individual property owners but also strengthens Airbnb's market position, making it a win-win scenario. By solving this critical problem, the authors aim to contribute to the continued financial success and growth of Airbnb and its community of hosts.

Data Source

Inside Airbnb Dataset

The datasets were pulled from the Asian Institute of Management's (AIM) public data directory. While the `insideairbnb` repository has multiple datasets, only two were used for this study. The `reviews` and the `listings` data for countries in europe and the USA were employed. The dataset is also openly available on the inside airbnb website.

The Features per dataset and samples are as below:

LISTINGS DATA

Feature Name	Data Type
id	Integer
name	String
host_id	Integer
host_name	String
neighbourhood_group	String
neighbourhood	String
latitude	Float
longitude	Float
room_type	String
price	Float
minimum_nights	Integer
number_of_reviews	Integer
last_review	Date
reviews_per_month	Float
calculated_host_listings_count	Integer
availability_365	Integer
number_of_reviews_ltm	Integer
license	String

Table 1. Data description of the listings dataset.

A sample of the dataset has been provided below:

Out[9]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood
0	2384	Hyde Park- Walk to UChicago or Theological Semi...	2613	Rebecca	None	Hyde Park
1	4505	One great apartment, 332 great reviews, 1 bad ...	5775	Craig & Kathleen	None	South Lawndale
2	6715	Lincoln Park Oasis - Unit 2 ONLY	15365	Reem	None	Lincoln Park
3	7126	Tiny Studio Apartment 94 Walk Score	17928	Sarah	None	West Town
4	9811	Barbara's Hideaway - Old Town	33004	At Home Inn	None	Lincoln Park

DataFrame 1. Sample of listings dataset

REVIEWS DATA

Feature Name	Data Type
listing_id	Integer
id	Integer
date	Date
reviewer_id	Integer
reviewer_name	String
comments	String

Table 2. Data description of the reviews dataset.

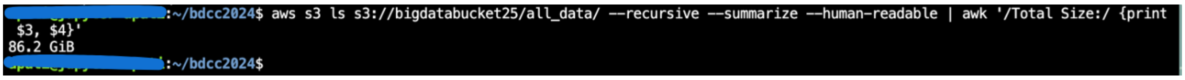
A sample of the dataset has been provided below:

Out[10]:

	listing_id	id	date	reviewer_id	reviewer_name	comments
0	2384	25218143	2015-01-09	14385014	Ivan	it's a wonderful trip experience. I didn't exc...
1	2384	28475392	2015-03-24	16241178	Namhaitou	This is my first trip using Airbnb. I was a li...
2	2384	30273263	2015-04-19	26101401	Patrick	The reservation was canceled 80 days before ar...
3	2384	30974202	2015-04-30	26247321	Cristina	Sólo puedo decir cosas buenas de Rebecca. La h...
4	2384	31363208	2015-05-04	31293837	SuJung	Rebecca was an absolutely wonderful host.

DataFrame 2. Sample of reviews dataset.

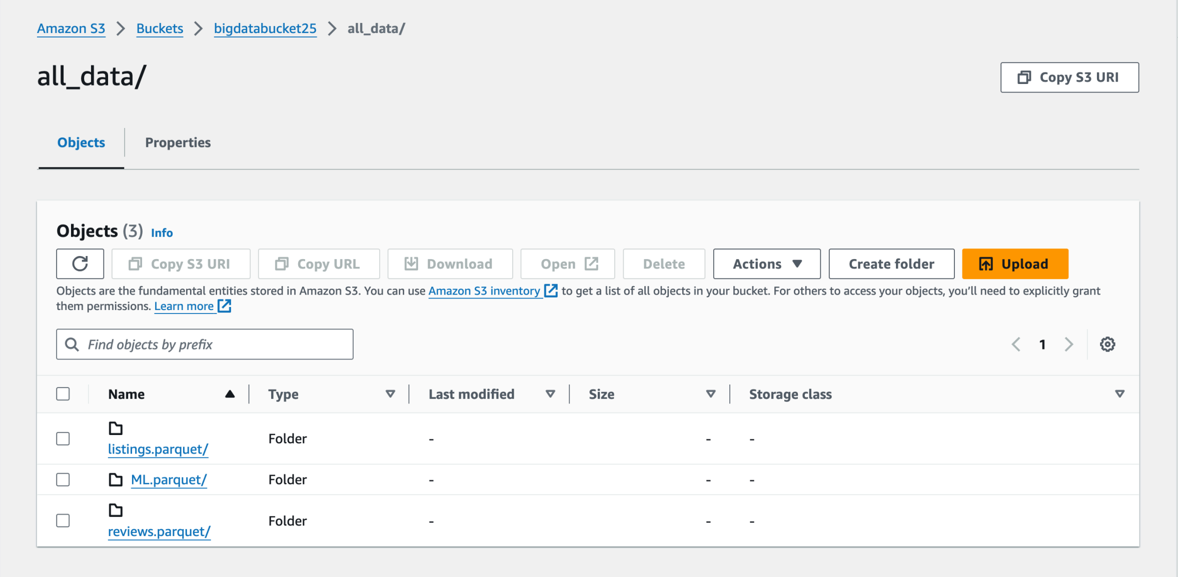
The study uses 3 parquet files as the data processed, hence in order to get the size out, the three were placed in a folder and their size was outputed. The total processed data amount to approximately 86GB, which is above the required 50GB for the study mandates.



```
[redacted]:~/bdcc2024$ aws s3 ls s3://bigdatabucket25/all_data/ --recursive --summarize --human-readable | awk '/Total Size:/ {print
 $3, $4}'
86.2 GiB
[redacted]:~/bdcc2024$
```

Image 2. Data size.

Proof of them being in the bucket has been provided below:



Amazon S3 > Buckets > bigdatabucket25 > all_data/

all_data/

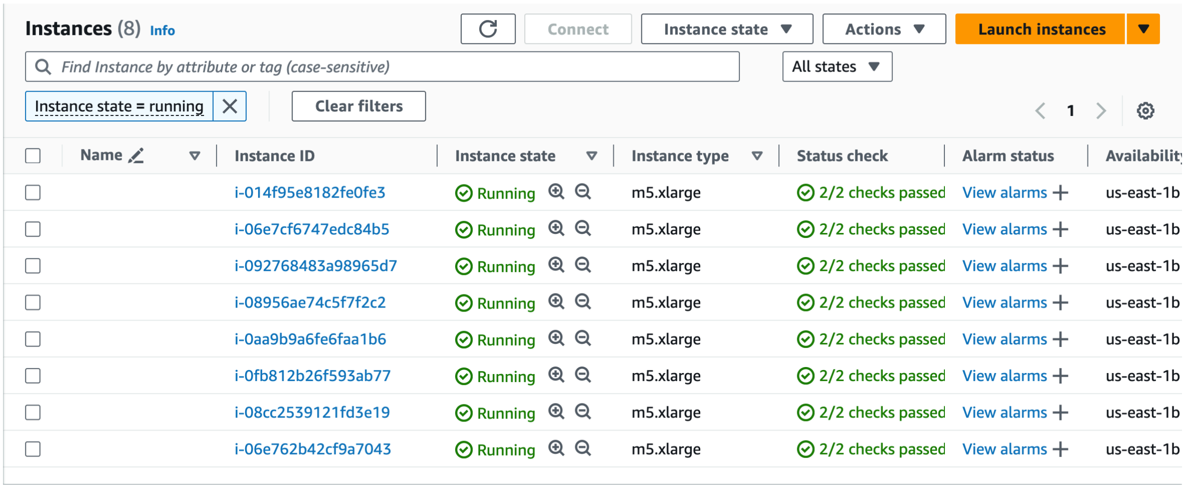
Objects (3) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	listings.parquet/	Folder	-	-	-
□	ML.parquet/	Folder	-	-	-
□	reviews.parquet/	Folder	-	-	-

Image 3. Proof of file.

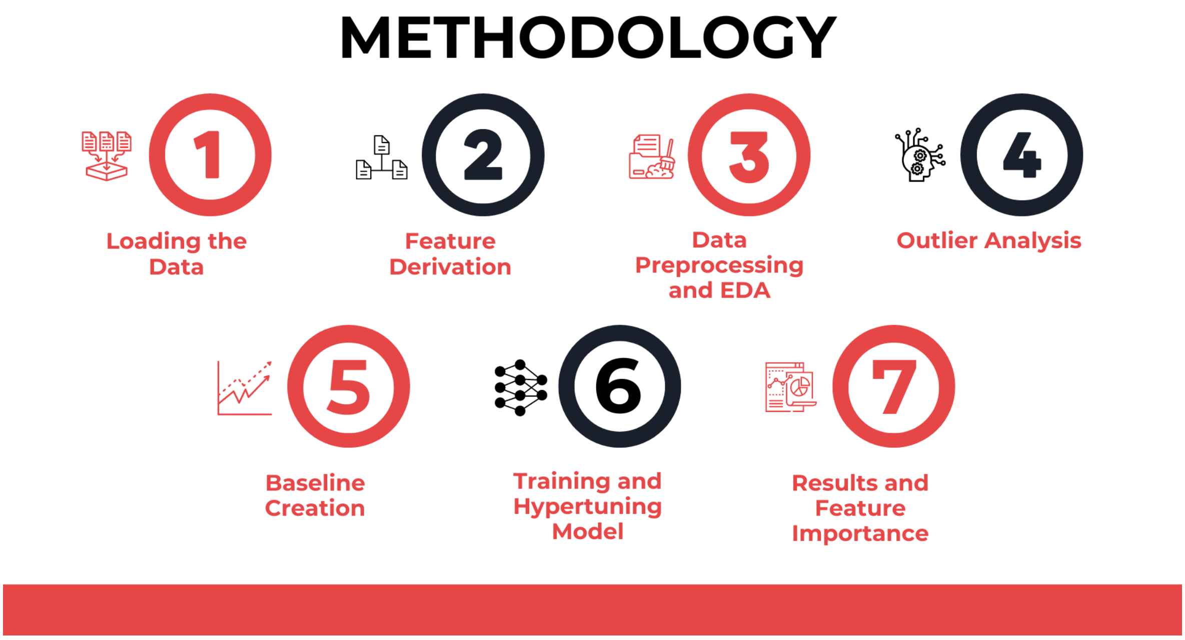
Image of instance types



□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabilit
□		i-014f95e8182fe0fe3	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-06e7cf6747edc84b5	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-092768483a98965d7	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-08956ae74c5f7f2c2	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-0aa9b9a6fe6faa1b6	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-0fb812b26f593ab77	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-08cc2539121fd3e19	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b
□		i-06e762b42cf9a7043	Running	m5.xlarge	2/2 checks passed	View alarms +	us-east-1b

Image 4. instances tab.

Methodology



METHODOLOGY
1
Loading the Data
2
Feature Derivation
3
Data Preprocessing and EDA
4
Outlier Analysis
5
Baseline Creation
6
Training and Hypertuning Model
7
Results and Feature Importance

Image 5. Methodology.

Feature Derivation

Initially, it was clear that many of the available features were either irrelevant or insufficient on their own, requiring additional context or external data to become actionable. This realization led to a feature derivation process, where existing features were transformed and combined to create more insightful and predictive variables for the ML model.

The derived features are as follows:

Feature	Description	Data Type	Formula
`listing_id`	Unique identifier for an Airbnb listing. Serves as the key feature for merging datasets.	double	N/A
`avg_reviews_per_month`	Average number of reviews per month for a listing across all active periods.	double	`mean('reviews_per_mon`
`host_experience`	Calculated as total reviews divided by total listings for an ID, indicating host engagement and guest feedback.	double	`col('total_reviews') /` `col('total_host_listin`
`average_nights`	Average minimum number of nights a guest must stay, based on host settings.	double	`col('total_minimum_nig` `col('total_host_listin`
`avg_price`	Average price per listing ID.	double	`mean('price')`
`avg_price_per_night`	Average of price divided by minimum number of nights, providing nightly charge insights.	double	`col('price') /` `col('minimum_nights')`
`occupancy`	Target variable derived from `availability_365`, indicating occupancy rate over a 365-day period.	double	`1 - (col('availabilit` `365)`
`avg_len_reviews`	Average length of reviews for each listing, offering insight into the typical amount of feedback from guests.	double	`F.mean(F.length('comm`

Table 3. Data description of the derived features from listings and reviews data.

Note - The dataframes were combined further down the pipeline to ensure reduction in run time.

```
VBox()
```

```
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…

VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```

Data Exploration and Preprocessing

During the initial exploratory data analysis (EDA), several graphs and visualizations revealed some unexpected and unusual patterns. On further investigation, the authors identified that certain features contained extreme or impossible values, which needed to be addressed to ensure data quality and model accuracy.

For example, the `avg_price` feature exhibited some negative values, which are clearly unrealistic for a pricing attribute. Such anomalies necessitated a thorough cleaning and preprocessing step to remove outliers and erroneous data points.

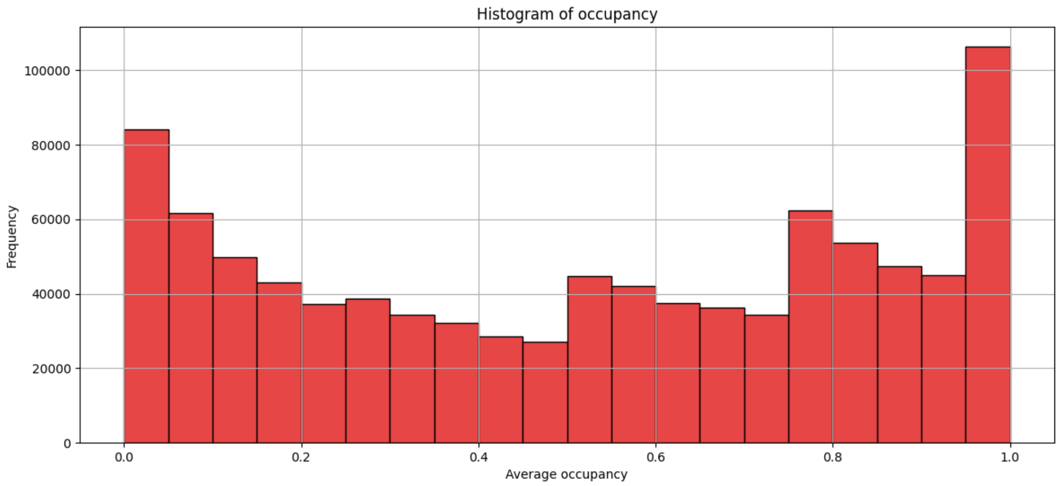
Key steps taken during EDA and preprocessing included:

- **Filtering Extreme Values:** Features with extreme values that were beyond reasonable limits were also scrutinized. This included setting logical bounds for features like `average_nights` to ensure they fell within a plausible range.
- **Handling Missing Data:** Missing values were addressed through imputation methods or by removing records with significant missing information.

Through these steps, the data was cleaned and preprocessed to better reflect realistic scenarios and improve the robustness of the predictive model. Following the filtering, proper EDA was performed

```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…

VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```



Visual 1. occupancy rating spread.

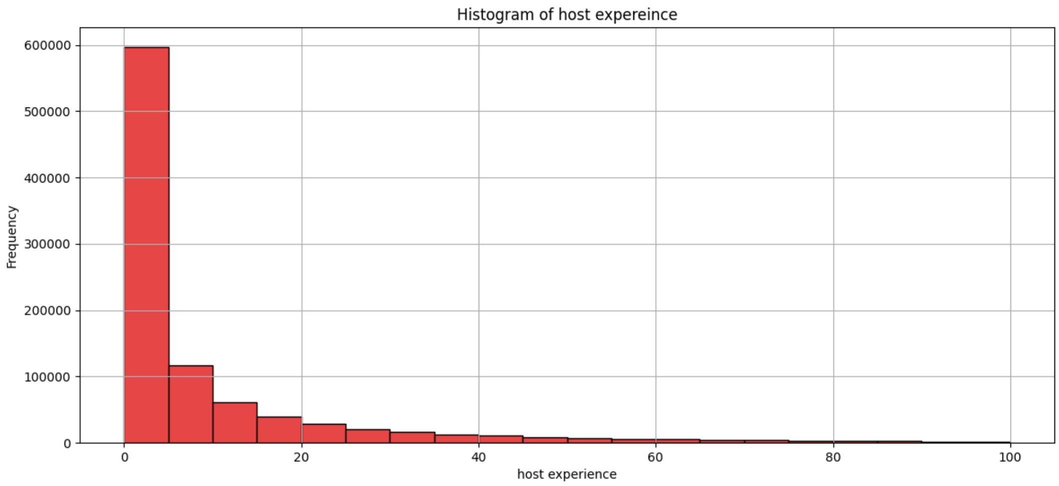
The figure above illustrates the distribution of occupancy ratings. Ideally, the majority of the ratings would cluster around the center of the graph, indicating a balanced spread. However, in this case, the highest frequencies are observed at the extreme ends of the graph.

There are two potential reasons for occupancy ratings to be on the extreme ends:

1. The data has been imputed incorrectly
2. These are ***fake listings*** - such listings are made in order to attract customers, take thier money and never actually offer them a place to live[3]

This unusual pattern provided further motivation to conduct a comprehensive outlier analysis on the occupancy feature to better understand and address these anomalies, which will be showcased in the next segment.

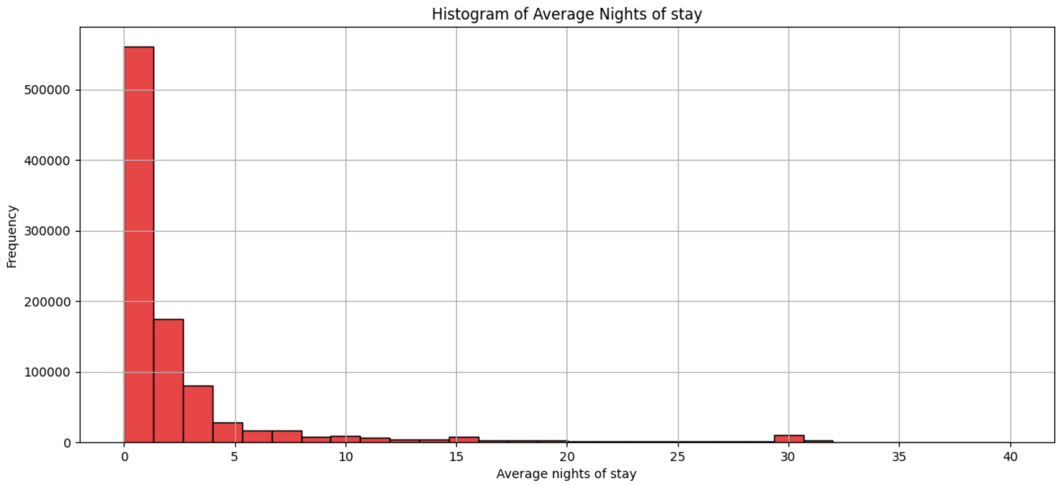
```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```



visual 2. host experience rating spread.

Unlike the previous visual, the host experience does not show any anmalites, indicating no need for outlier analysis.

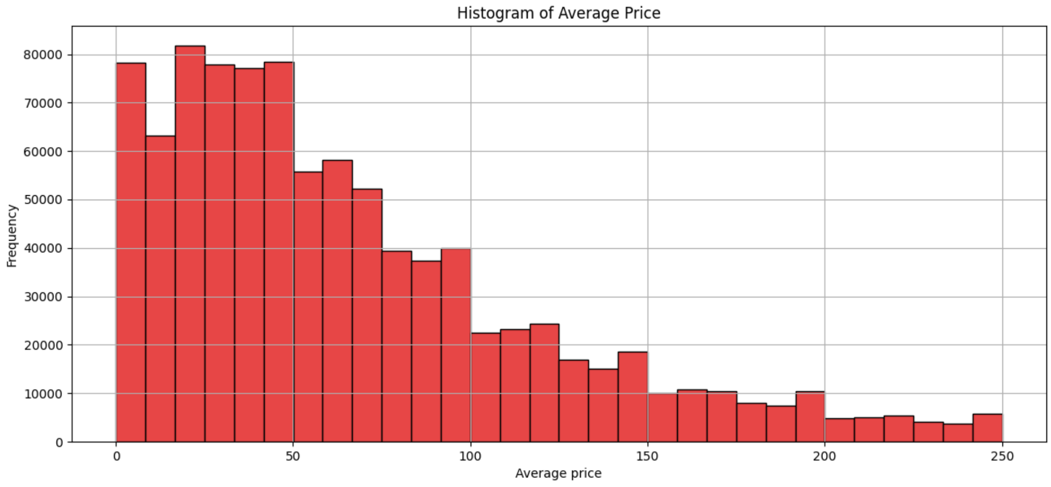
```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```



visual 3. Average nights of stay spread.

The histogram of the average nights follows a right skewed behaviour and does not display anything out the ordinary.

```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```



visual 4. Average price spread.

Similar to the visual of average nights, the average price too follows right skewed behaviour. Such behaviour is expected and since there aren't any anomalies being noticed, outlier analysis need not be done for this feature.

Following the insights gained from EDA, the authors then moved to conduct outlier analysis.

Outlier Analysis

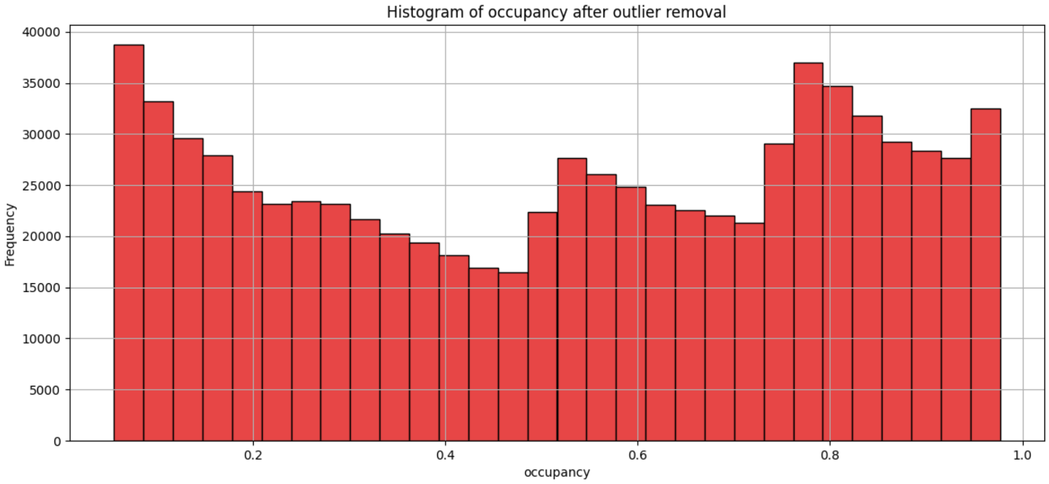
Outliers in the data can not only skew the model's prediction power but also hinder it's ability to extract feature importance. To ensure a more robust and accurate model, outlier removal was pivotal in the study's pipeline.

After delving into the insights extracted from the occupancy histogram, the authors proceeded with outlier analysis and subsequent removal. Initially experimenting with a `Gaussian Mixture Model`, they encountered challenges due to the absence of distinct clusters. Consequently, they turned to statistical methods for analysis. Calculating the standard deviation and mean for the features, they then computed `z-scores` to identify outliers. Leveraging this information, outliers were systematically eliminated, leading to the creation of a refined dataframe devoid of such anomalies. The threshold was computed by multiplying the standard deviation by a factor of 4

Once the outlier removal was completed, both the dataframes were joined using a `broadcast` join

```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```

```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```



visual 5. Occupancy rating spread post outlier removal.

Comparing `Visual 5` and `Visual 1`, it is evident that the application of the `z-score` method for outlier removal has effectively eliminated extreme outliers. This process has resulted in cleaner data, which is more suitable for model development.

Setting a Baseline

To establish a benchmark for evaluating the model results, a baseline was created. This baseline serves as a point of comparison to understand the effectiveness and accuracy of the predictive model. The baseline was constructed by calculating the mean of the target column, which represents the average value of the target variable across all data points. This mean value was then used as the prediction for all listings, regardless of their individual features or characteristics.

By adopting the mean of the target column as the predicted rating for each listing, a simplistic yet effective baseline model was formulated. This approach assumes that the best prediction one could make without any additional information is the average value. While rudimentary, this method provides a useful benchmark against which more sophisticated models can be measured.

Subsequently, the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were calculated for this baseline model. These metrics quantify the prediction errors by comparing the baseline predictions to the actual target values. RMSE provides a measure of the average magnitude of the errors, giving higher weight to larger errors, while MAE offers a straightforward average of absolute errors, treating all errors equally.

The resulting RMSE and MAE values from the baseline model serve as crucial benchmarks. They provide a clear indication of the minimum performance level that any developed model should surpass.

The results are as below:

Baseline	MAE	RMSE
Value	0.32	0.41

Table 6. Baseline mae and rmse

```
VBox()
FloatProgress(value=0.0, bar_style='info', description='Progress:', layout
=Layout(height='25px', width='50%'),…
```

Training and Hypertuning Model

Once the data was fully aggregated and filtered, the final feature columns were selected, and a subsequent Parquet file was created to mitigate memory and run-time issues. This parquet file contains the joined version of the listings and reviews dataset. The parquet files contains two columns, one of the vector assembled features column and the other of the target. The columns used for this were the features and target for the model training, they are as below:

features	occupancy
[19.43299711815562, 7.191483516483516, 4.14328...	0.089463
[20.180174563591024, 84.0592261904762, 80.2, 4...	0.653082
[29.503496503496503, 10.115857142857143, 22.86...	0.330137
[14.528338927237034, 65.0, 98.68125, 8.5187755...	0.059603
[23.654019873532068, 12.631608422939069, 4.865...	0.194082

Table 7. Dataframe for ML model training

The features used in the final model were:

Feature Name
avg_len_reviews
avg_price_per_night
host_experience
avg_reviews_per_month
average_nights

Table 8. Features for model training

This Parquet file was then reloaded, providing a streamlined and efficient dataset for further analysis. A CrossValidator was then established to fine-tune the machine learning model. The components of the CrossValidator included:

1. Model - `RandomForest Regressor` model with the target column as `occupancy`
2. evaluator = `RegressionEvaluator` with metrics `rmse` and `mae`
3. parameter grid - `max depth` of 2 - 10 and `number of trees` 30-50
4. Number of folds - `3` to ensure that the model does not overfit

The use of a CrossValidator ensures that the model is trained and validated across multiple subsets of the data, providing a comprehensive assessment of its performance and aiding in the selection of the best hyperparameters for optimal prediction accuracy. With the cross validator in place, the data was split into a 70-30 train/test split and subsequent model training was conducted.

The results of the best model were:

Hypertuned RF model	MAE	RMSE
Value	0.23	0.27

Table 9. Best model results

Feature importances

After finalizing the model training phase and selecting the optimal model, the researchers proceeded to extract the feature importances. This critical step aimed to unveil the underlying factors influencing the model's predictions, thereby facilitating a deeper comprehension of the decision-making process. By deciphering what precisely drives the model's predictions, valuable insights were gleaned, enabling the identification of actionable strategies to assist hosts effectively.

While storing the parquet file, the columns were vectorised in this order:

1. Average price per night
2. Average len of reviews
3. Minimum nights stay
4. Average reviews per month
5. Host experience

The feature importance was then mapped to the number on the index of the result.

The ***top 3*** features were:

Feature	MAE
Host experience	33%
Average price per night	18%
Minimum nights stay	19%

Table 9. Top three predictors

Mapping the features back to their formula, it can be notcied that all of these features are actionable and are aspects hosts can alter to change their occupancy

Conclusion

Model	MAE	RMSE
Baseline	0.32	0.41
Our model	0.23	0.27

Table 10. Model Comparison

1. Comparing the two above, the authors successfully developed a model capable of predicting occupancy ratings with significantly greater accuracy than the baseline. Despite the lack on actionable features, features were derived and by rigorous forms of data pre processing and outlier removal, a robust Random Forest model was created.
2. By utilizing the model, the authors were able to extract feature importances, providing valuable insights into the primary factors influencing occupancy ratings. This analysis revealed potential strategies for assisting hosts, demonstrating the practical applications and benefits of turning this model into a real-world product.

Recommendations

1. **Expand Feature Set**: The study was conducted using a limited set of features. The authors believe that incorporating more relevant features, as well as features that complement the existing ones, could not only enhance model performance but also provide deeper insights into the factors influencing occupancy rates. A richer feature set could capture more nuances and improve predictive accuracy.
2. **Region-Wise Analysis**: The authors recommend conducting region-specific analyses. While this study encompassed listings from across the globe, it is evident that Airbnb properties vary significantly by location. Different regions have unique characteristics, such as types of accommodations, amenities, tourist attractions, and visitor demographics. By segregating the data into distinct regions, more accurate and region-specific ML models can be developed, offering personalized insights that could greatly benefit hosts by catering to their unique regional dynamics.

3. **Segregate by Room Types**: The study did not account for variations in room types due to encoding errors. The authors suggest conducting separate analyses based on room types. Recognizing the differences between various room categories could yield more precise results and actionable insights. Understanding the specific dynamics of each room type can help tailor strategies to optimize occupancy and enhance overall performance.

By addressing these recommendations, future research can build on the current study's findings to create more robust models and provide more targeted insights, ultimately helping Airbnb hosts maximize their occupancy rates and profitability.

References

1. “Navigating the Airbnb Bust: A Comprehensive Guide to Seizing Opportunities! | IGMS.” IGMS, 29 Sept. 2023, www.igms.com/airbnb-bust/.

2. “Airbnb Revenue 2018-2023 | ABNB.” Macrotrends.net, 2018, www.macrotrends.net/stocks/charts/ABNB/airbnb/revenue.

3. Chloë Nannestad. “Airbnb Scams: The 5 Most Common Ones and How to Avoid Them.” Reader’s Digest, 24 Apr. 2021, www.rd.com/article/airbnb-scams/.

Disclosure

1. Chatgpt was used to aid in writing this report
2. The project was done in AWS but report creation was done in jojie