Airbnb Listings Analysis

Datasets

The datasets for this project were obtained from official Airbnb website.

Link for datasets : http://insideairbnb.com/chicago/

We have two datasets in this project:

- Listings.csv
- Reviews.csv

Description

Listings.csv

This dataset contains 7,747 rows and 18 columns.

Each row in the file represents a single listing and includes information such as the listing ID, name, host ID, host name, neighbourhood group, neighborhood, latitude, longitude, room type, price, minimum nights, number of reviews, last review, reviews per month,calculated host listings, availability, number of reviews ltm, and licenses.

Reviews.csv

This dataset contains 3,45,939 rows and 6 columns.

Each row in the file represents a single review and includes information such as the unique ID of the listing being reviewed, the date the review was submitted, the ID of the reviewer, the name of the reviewer, and the text of the review itself.

Problem Statement

The problem is to provide insights and recommendations to Airbnb hosts in Chicago on how to optimize their listing performance and maximize their revenue. Specifically, the analysis aims to answer the following questions:

1. What are the key factors that influence the price of an Airbnb listing in Chicago? How can hosts adjust their pricing strategy to maximize their revenue?

2. What are the most popular neighbourhoods among Airbnb guests in Chicago?

3. Is there a correlation between the sentiment expressed in reviews and the price of a listing, and can we make predictions about the sentiment label of a listing based solely on its price and the number of reviews it has received?

4. How are Airbnb listings distributed based on their price and popularity(number of reviews)?

Research Questions

The research problem is to identify the key factors that influence the price of an Airbnb listing, and how hosts can adjust their pricing strategy accordingly.

1) Predicting the price for Airbnb rentals based on listing attributes, and sentiment analysis.

2) Predicting the sentiment label of a listing solely based on its price and the number of reviews it has received.

3) How are Airbnb listings distributed based on their price and popularity?

Data Cleaning

Dataset 1 - Listings.csv

- After taking a look at the dataset, we observed that the neighbourhood_group has only NaN values. So we dropped the neighbourhood_group column.
- Also, last_review, reviews_per_month and license columns have some missing values. But these values cannot be replaced by any other values, so we dropped the rows that have NaN values.
- Lastly, there were some inconsistent values in the dataset such as some listings have price $0.00. So, we removed those rows that have price = 0.00 to maintain consistency.
- After cleaning, we have a total of 5888 records from the original 7747; thus, we've retained a reasonable amount of records.

Dataset 2 - Reviews.csv

- After taking a look at the dataset, we observed that the comments column has some missing values, 92 missing values to be precise. These missing comments cannot be replaced by any other values. Hence, the rows with missing values has been dropped.
- After cleaning, we have a total of 345847 records from the original 345939; thus, we've retained a reasonable amount of records.

Exploratory Data Analysis

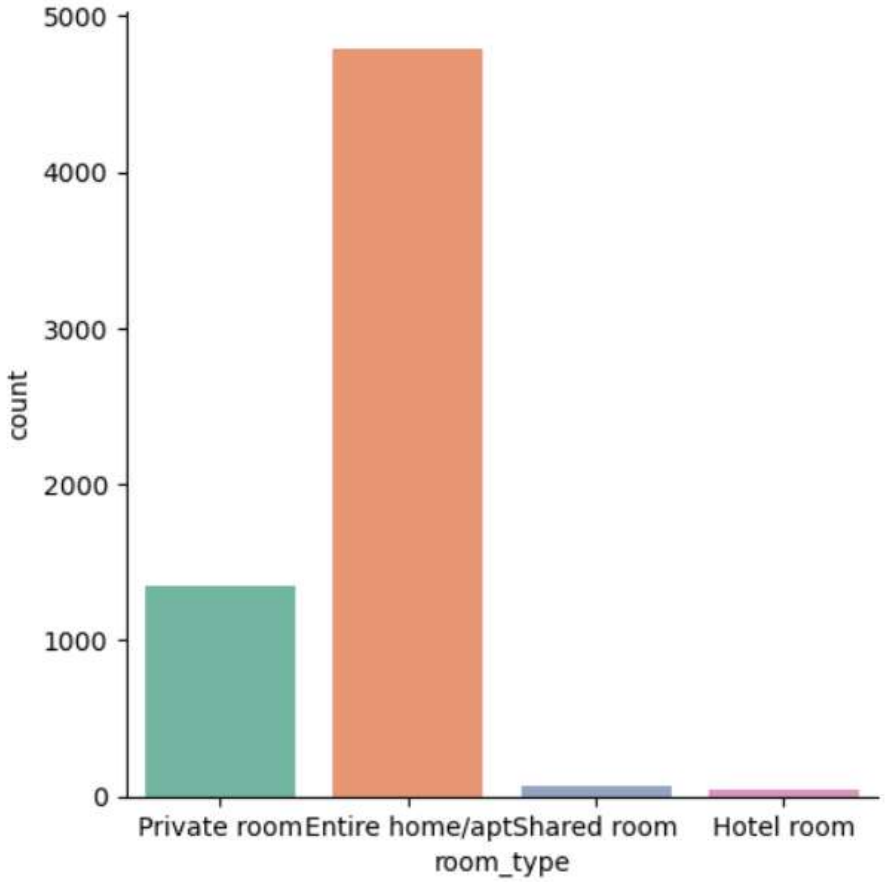
The main problem was broken into 3 sub-problems each targeting a different aspect of the dataset. The three sub-problems are:

- What are the features of a listing that influence its price?
- What are the most popular neighbourhoods among Airbnb guests in the area?
- Does review's sentiment affect the price?

By analysing the dataset with respect to each sub-problem, we can gain useful statistical insights about the data.

Sub-Problem 1 : What are the features of a listing that influence its price?

Our first sub-problem was to focus on the physical features and facilities of the property itself. We wanted to see if there were any common features among the highly priced listings. We mainly focused on the listing's room type.



Based on the countplot, it is evident that the majority of listings are for entire home/apt. Private rooms come in second, followed by shared rooms, and hotel rooms have the lowest count. This provides a glimpse into the types of listings that are available and their respective frequencies.

The above heatmap shows a color-coded representation of prices, where lighter colors indicate lower prices and darker colors indicate higher prices. Analysis of the heatmap reveals that shared rooms have the lightest color, indicating the lowest price point, followed by private rooms with a slightly darker color. Hotel rooms have a darker color than private rooms, and entire houses have the darkest color, indicating that they are the most expensive.

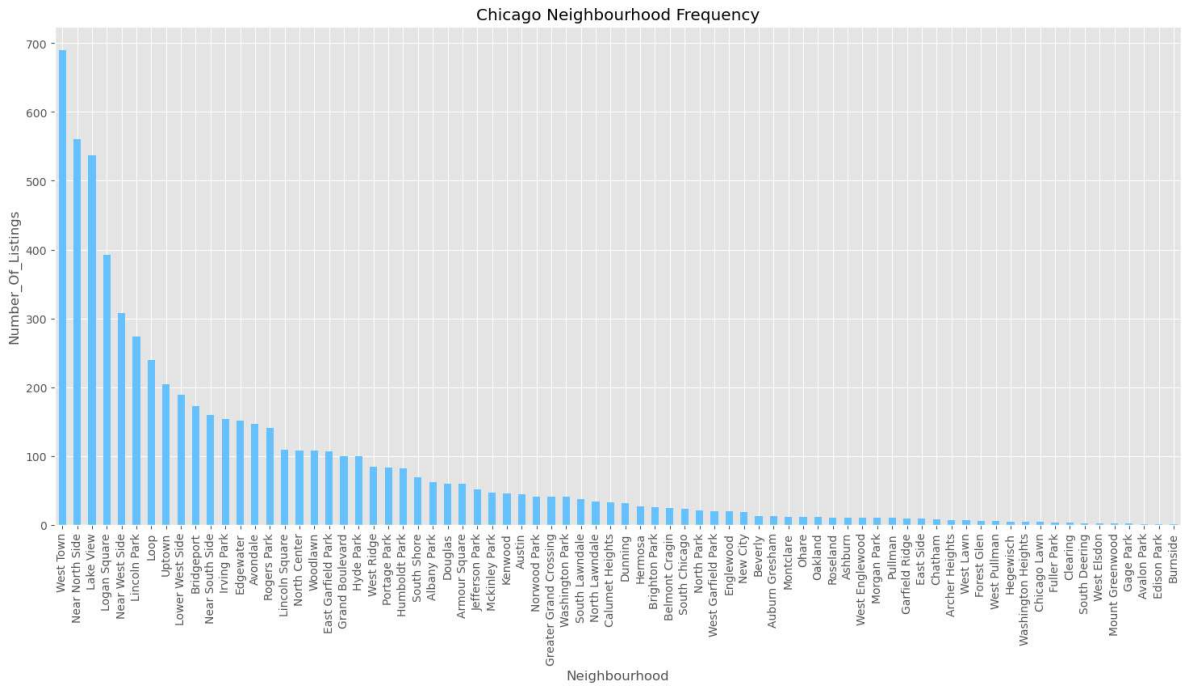
Overall, this observation highlights the significance of the room type in determining the final price of a listing.

Sub-problem 2 : What are the most popular neighbourhoods among Airbnb guests in the area?

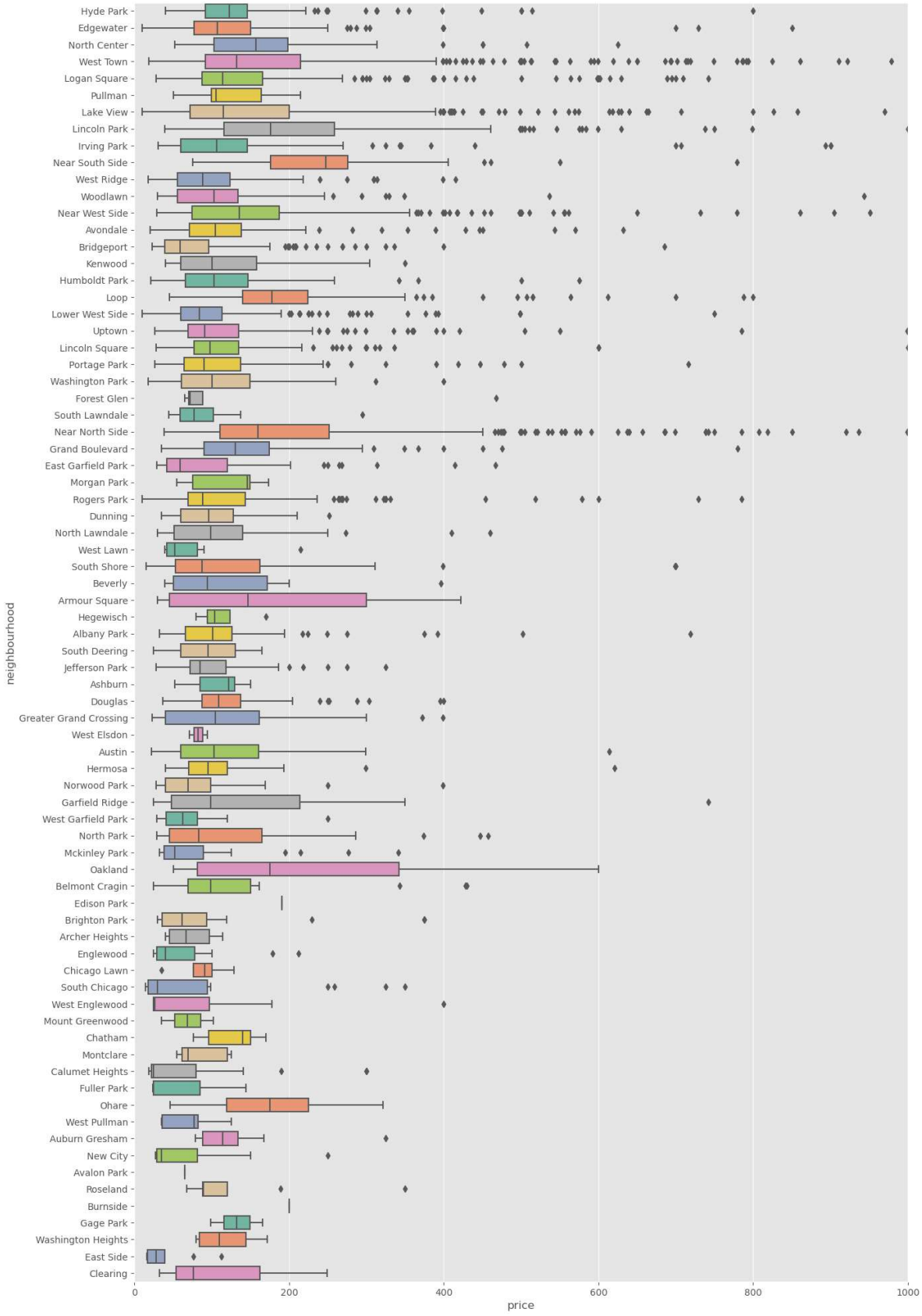
The number of listings for each neighbourhood and its median price.

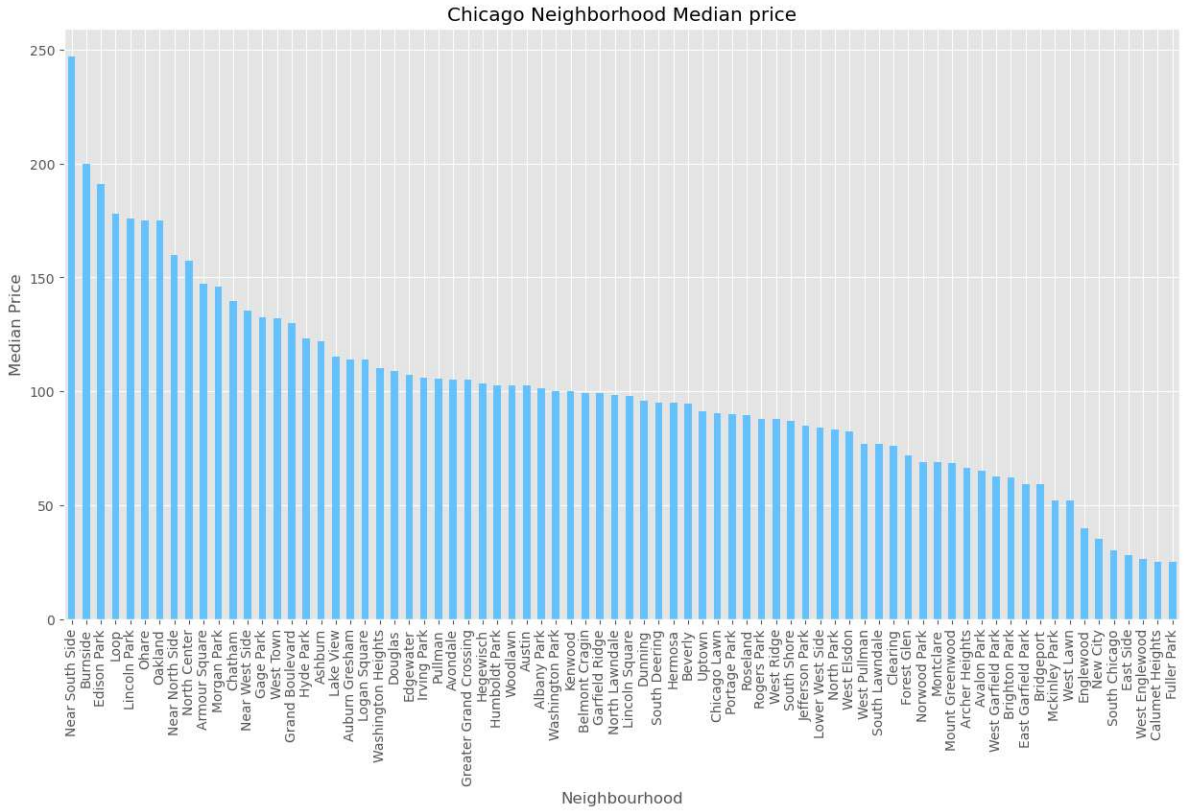
	neighbourhood	Number_Of_Listings	Median_Price
0	West Town	690	132.0
1	Near North Side	560	160.0
2	Lake View	537	115.0
3	Logan Square	392	114.0
4	Near West Side	308	135.5

Visualizing the number of listings for each neighbourhood



We can see that most of the listings appear in 'West Town','Near North Side','Lake View','Logan Square' etc. This gives us a good insight into the potential neighbourhoods where there are high number of listings. Our next step would be to analyze it with the price.

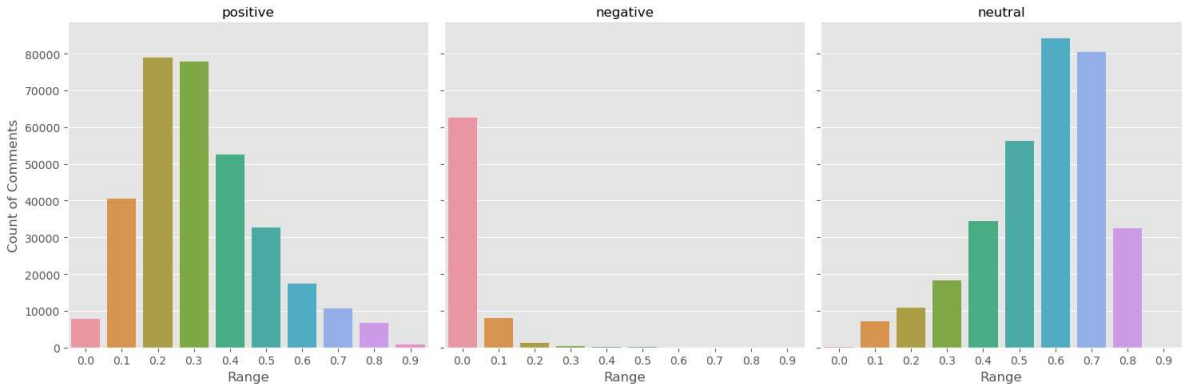

Hyde Park
Edgewater
North Center
West Town
Logan Square
Pullman
Lake View
Lincoln Park
Irving Park
Near South Side
West Ridge
Woodlawn
Near West Side
Avondale
Bridgeport
Kenwood
Humboldt Park
Loop
Lower West Side
Uptown
Lincoln Square
Portage Park
Washington Park
Forest Glen
South Lawndale
Near North Side
Grand Boulevard
East Garfield Park
Morgan Park
Rogers Park
Dunning
North Lawndale
West Lawn
South Shore
Beverly
Armour Square
Hegewisch
Albany Park
South Deering
Jefferson Park
Ashburn
Douglas
Greater Grand Crossing
West Elsdon
Austin
Hermosa
Norwood Park
Garfield Ridge
West Garfield Park
North Park
Mckinley Park
Oakland
Belmont Cragin
Edison Park
Brighton Park
Archer Heights
Englewood
Chicago Lawn
South Chicago
West Englewood
Mount Greenwood
Chatham
Montclare
Calumet Heights
Fuller Park
Ohare
West Pullman
Auburn Gresham
New City
Avalon Park
Roseland
Burnside
Gage Park
Washington Heights
East Side
Clearing
neighbourhood
0
200
400
600
800
1000
price

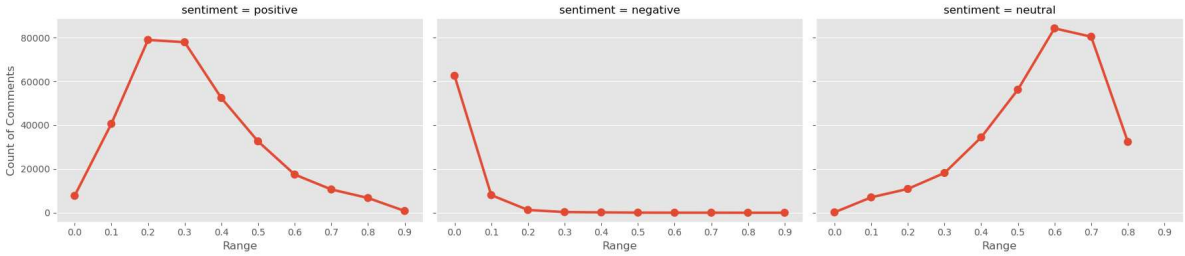


Through an examination of the number of listings and their corresponding prices across different neighborhoods, we can gain a better understanding of which neighborhoods have a high concentration of expensive listings. From the analysis performed thus far, it is apparent that certain neighborhoods have a higher overall price level than others. However, it is worth noting that some of these expensive neighborhoods may not necessarily have as many listings as other neighborhoods with similar price ranges.

Sub-problem 3 : Does review's sentiment affect the price?

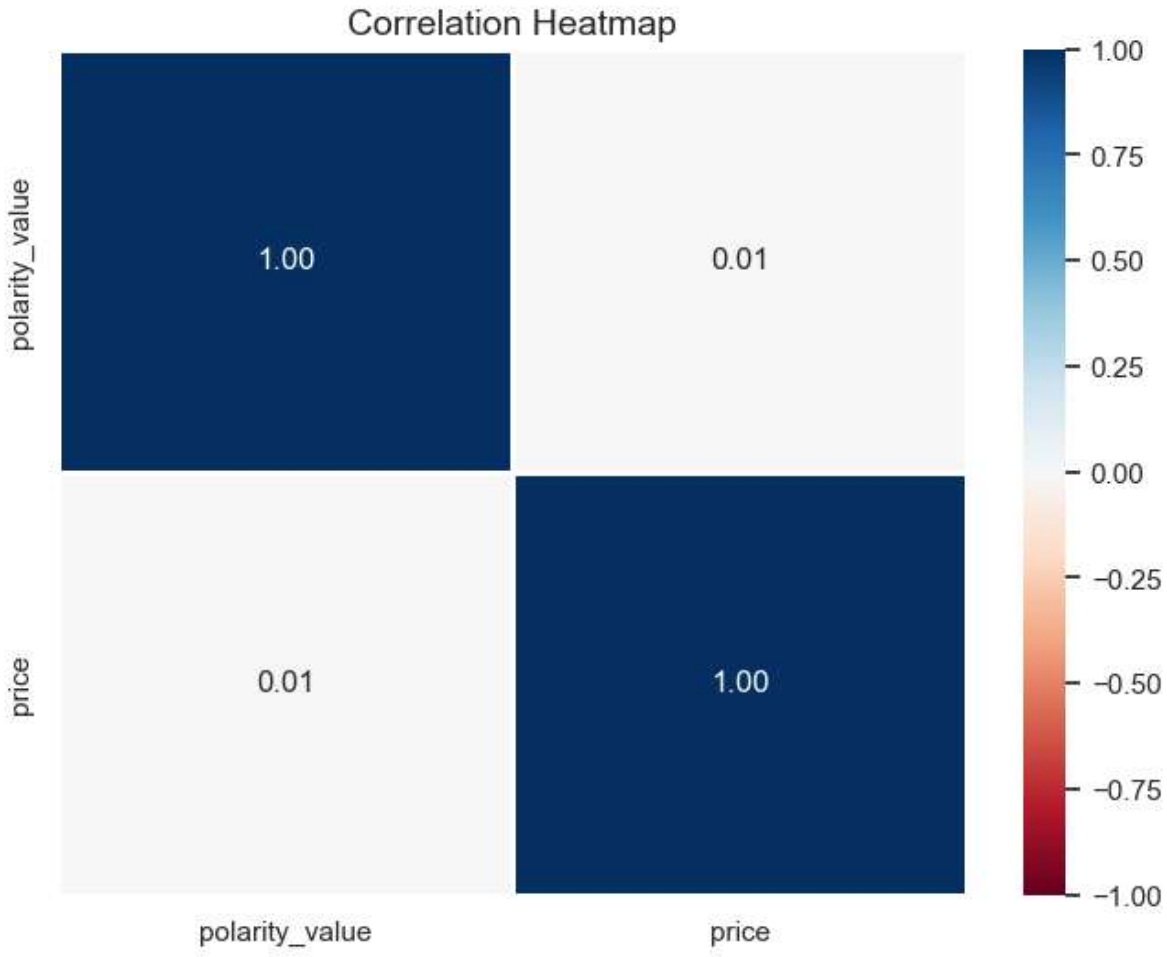
To compute review's sentiment, we have performed Sentiment Analysis.





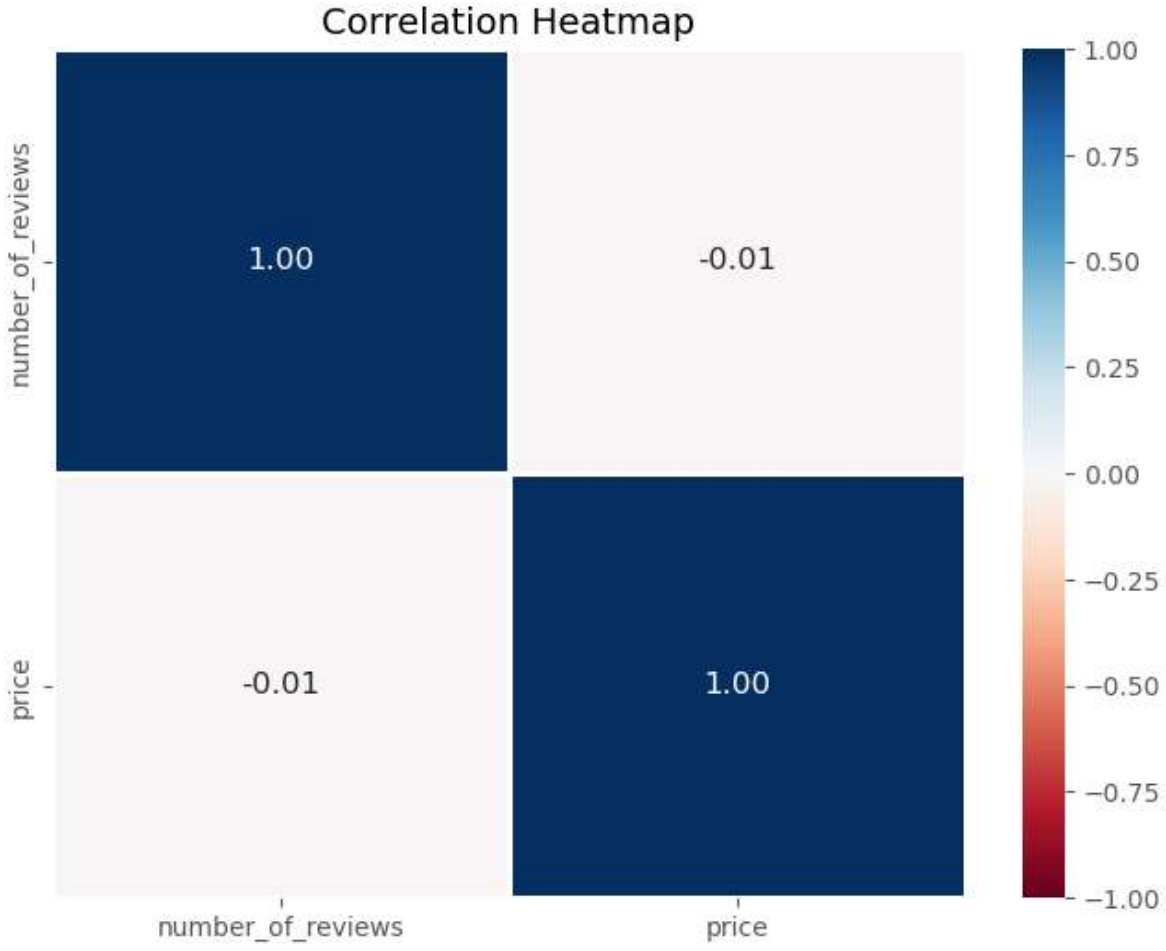
After analyzing the graphs above, we can draw three conclusions. Firstly, the majority of reviews do not contain much negativity, with only a few reviews expressing even a slight degree of negativity. In fact, most reviews exhibit no negativity, as evidenced by their classification as having 0.0 negative sentiment. Secondly, a significant number of reviews convey a reasonable level of positivity. However, it is also apparent that the majority of reviews are classified as neutral, indicating a lack of strong sentiment in either direction. This observation raises some ambiguity when attempting to infer sentiment with respect to price, as the bulk of reviews fall within the neutral category. Thus, we can conclude that most reviews are written with a neutral sentiment, albeit with a slight bias toward positivity.

Analyzing if the polarity_value of a listing affects its price



Analyzing if number of reviews of a listing affects its price.


Reviews based on Price
Number of reviews
Price
3000
2500
2000
1500
1000
500
0
0
2000
4000
6000
8000
10000


Correlation Heatmap
number_of_reviews
price
1.00
-0.01
-0.01
1.00
number_of_reviews
price
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

The above correlation matrix reveals a weak negative correlation between the number of reviews a listing receives and its price. Based on this observation, we can conclude that the number of reviews a listing has does not significantly impact its price.

Data

Dataset 1 - Listings.csv

Original Data : https://github.com/CS418/group-project-team-twenty/blob/main/data/listings.csv

Cleaned Data : https://github.com/CS418/group-project-team-twenty/blob/main/listings_cleaned.csv

Dataset 2 - Reviews.csv

Original Data : https://github.com/CS418/group-project-team-twenty/blob/main/data/reviews.csv

Cleaned Data : https://github.com/CS418/group-project-team-twenty/blob/main/reviews_cleaned.csv

ML/Stats

Question 1 - Predicting the price for Airbnb rentals based on listing attributes, and sentiment analysis.

Data Preparation

The following are performed on the data to ensure it fits into the different regression models:

- Encoding the categorical variables so that it can be fit into the regression models
- Separating the data into predictor and response variables
- Separating the data into training and testing sets (Training Sets: Testing Sets = 80% : 20%)

Regression Models

Regression models are employed to estimate a predicted value by considering independent variables, and they are primarily utilized for identifying the correlation between variables, as well as for prediction and forecasting.

Here, we use regression models to help predict the price based on the significant predictor variables identified in Exploratory Analysis.

Predictor Variables: room_type, number_of_reviews, polarity_value

Response Variable: price

The following regression models are carried out:

Linear Regression

Lasso Regression

Random Forest Regression

Question 2 - Predicting the sentiment label of a listing solely on its price and the number of reviews it has received.

During Exploratory Data Analysis, Sentiment Analysis is performed on reviews dataset and hence, polarity_value is obtained which is further used for classification.

Classification models are machine learning algorithms used to predict the categorical class of a given input based on a set of features or variables. The goal is to learn from a labeled dataset and then use that learning to predict the correct class label for new, unseen data.

Features : number_of_reviews, price

Target : sentiment_label

The following classification models will be carried out:

- Logistic Regression
- Decision Trees
- Random Forest Classifier

Question 3 - How are Airbnb listings distributed based on their price and popularity?

Clustering can be applied to the Airbnb dataset for various purposes, such as segmenting listings based on their features or identifying patterns in the data.

Here, we are using K means clustering to group Airbnb listings based on features like price, and number of reviews. This can help identify distinct types of listings, such budget, and popular rentals.

Results

Regression

Model 1 : Linear Regression

Linear Regression is a supervised learning algorithm that predicts a dependent variable (in this case, price) based on independent variables (the identified predictors). The algorithm attempts

to establish a linear relationship between the variables to make price predictions based on the linear line.

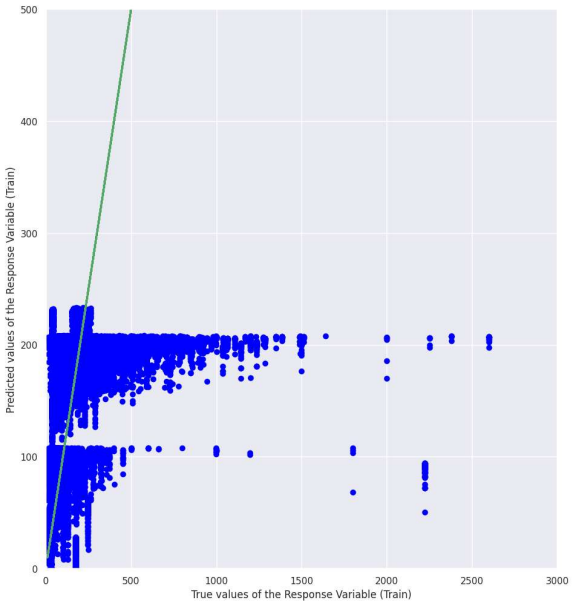
After fitting the Linear Regression on the train datasets, the coefficients of the Linear Regression line obtained are:

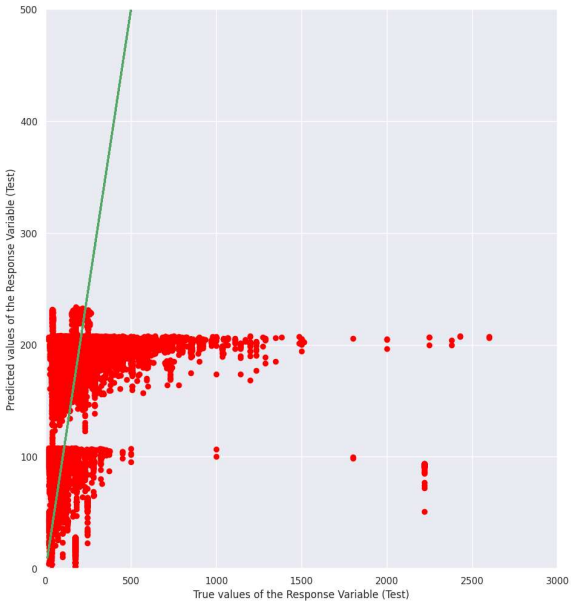
```
Intercept of Regression        : b =  [167.13752972]
```

	Predictors	Coefficients
0	polarity_value	6.358223
1	number_of_reviews	-20.866946
2	room_type_Entire_home_apt	19.691653
3	room_type_Hotel_room	10.855403
4	room_type_Private_room	-22.639729
5	room_type_Shared_room	-9.499813

A positive coefficient indicates that as the predictor variable increases, the response variable also increases. A negative coefficient indicates that as the predictor variable increases, the response variable decreases.

We then used the model to predict Airbnb rental prices for both the training and testing sets. We plotted the predicted values against the true values for both sets.





Points that lie on or near the diagonal line means that the values predicted by the Linear Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

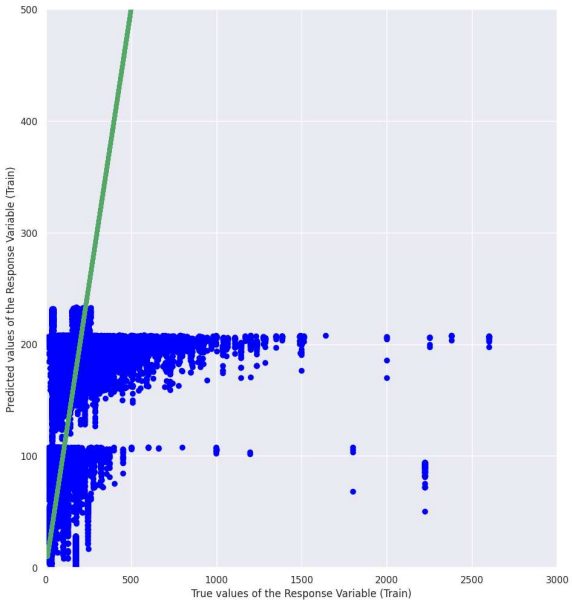
The Linear Regression model provides an insight into the relationship between the predictors and the Airbnb rental prices. However, the results should be interpreted cautiously, as the model may not capture all the complexities of the data, and other factors may be influencing

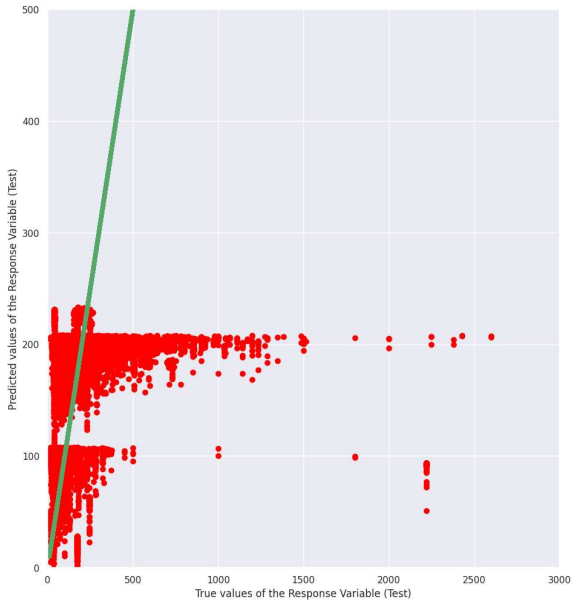
the prices. To obtain a more accurate prediction and deeper understanding, it is recommended to consider additional features, employ more advanced techniques, and use more sophisticated models.

Model 2: Lasso Regression

Lasso Regression is an improved version of linear regression that can be applied to both Regression and Classification problems. It can also be utilized for feature selection, where certain predictors are eliminated after a specified lambda threshold is reached. In this analysis, we used Lasso Regression to predict Airbnb rental prices using the same predictors as in the Linear Regression model.

We initialized the Lasso Regression model and fitted it with the training data, then we iteratively tested different λ values to determine the optimal fit. We stored the predictions and R2 scores for each λ value and identified the highest R2 value. Using the best fit Lasso Regression Model's predictions, we plotted the predicted values against the true values for both the training and testing sets.





Points that lie on or near the diagonal line means that the values predicted by the Lasso Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

Result: The highest R2 value obtained from the Lasso Regression model is: 0.0011440097502187774

The Lasso Regression model has the advantage of feature selection and can provide insights into the relationship between predictors and the response variable. However, the low R2 value indicates that the model may not capture all the complexities of the data, and other factors may be influencing the prices. To obtain a more accurate prediction and deeper understanding, it is

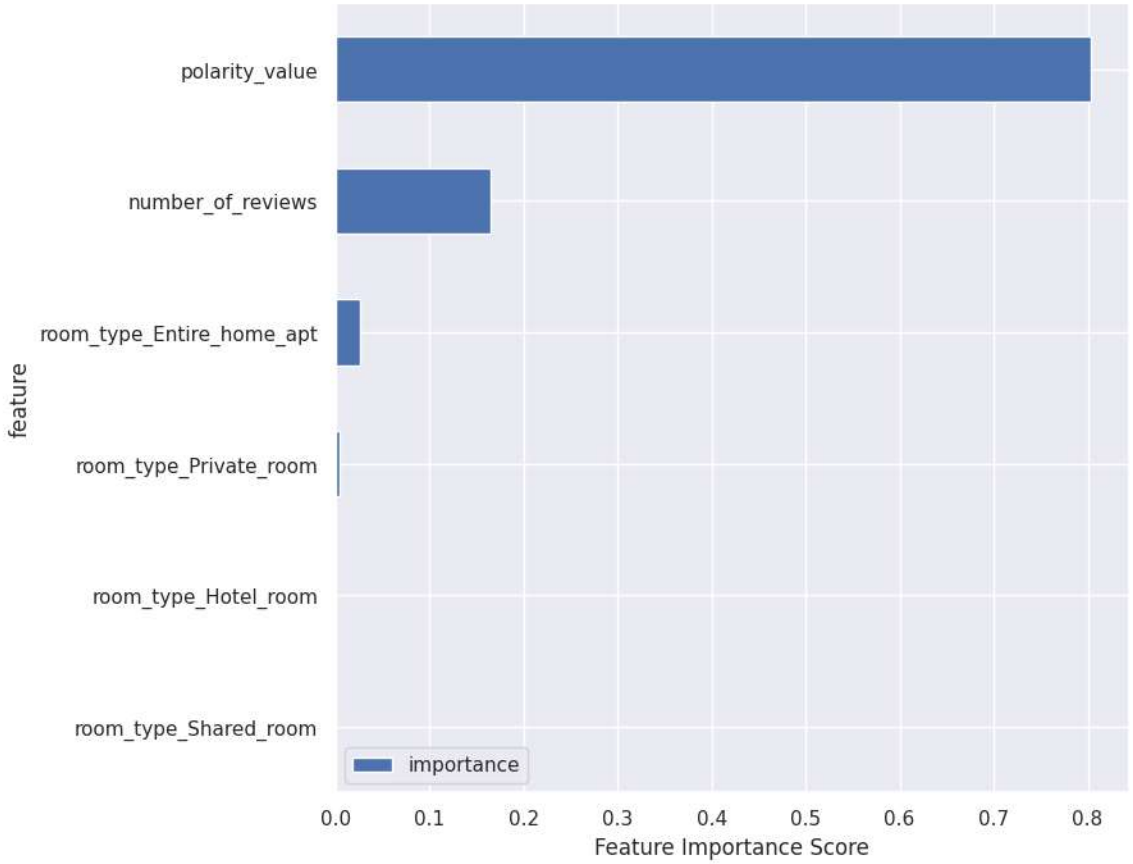
recommended to consider additional features, employ more advanced techniques, and use more sophisticated models.

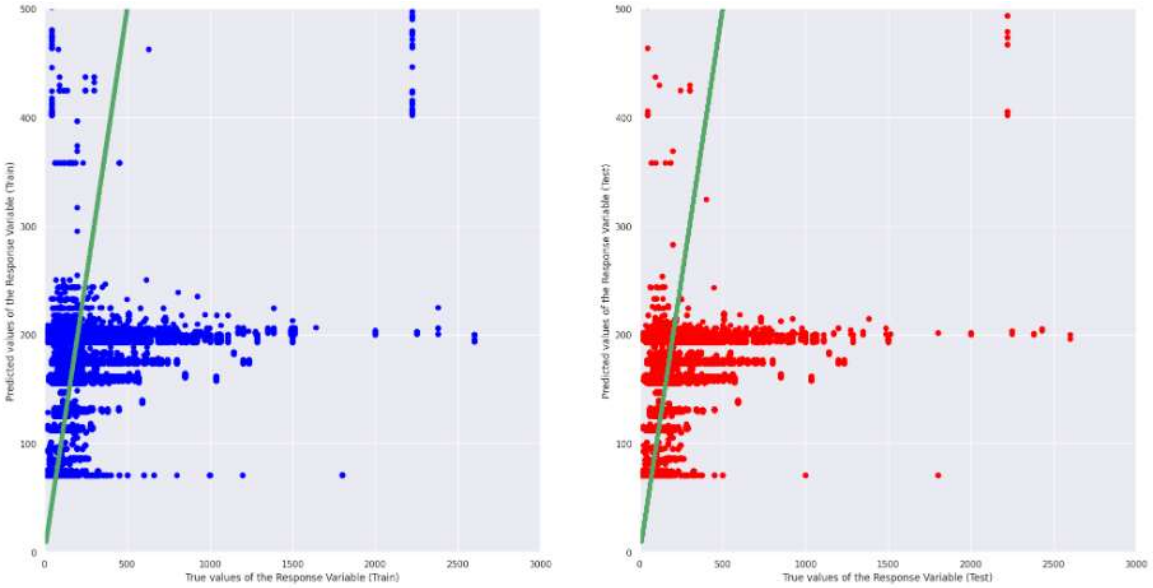
Model 3: Random Forest Regression

The Random Forest algorithm is an ensemble technique that can be used for both Regression and Classification tasks. It accomplishes this by utilizing multiple decision trees and a technique called Bootstrap Aggregation. Instead of relying on a single decision tree for its predictions, the algorithm combines the predictions of multiple decision trees to increase accuracy and robustness. In this analysis, we used the RandomForestRegressor to predict Airbnb rental prices using the same predictors as in the previous models.

We created and fitted the Random Forest Regression model with the training data, then predicted the prices for the training and testing sets.

We also obtained the feature importance scores and plotted those scores.





Note: Points that lie on or near the diagonal line means that the values predicted by the Random Forest Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

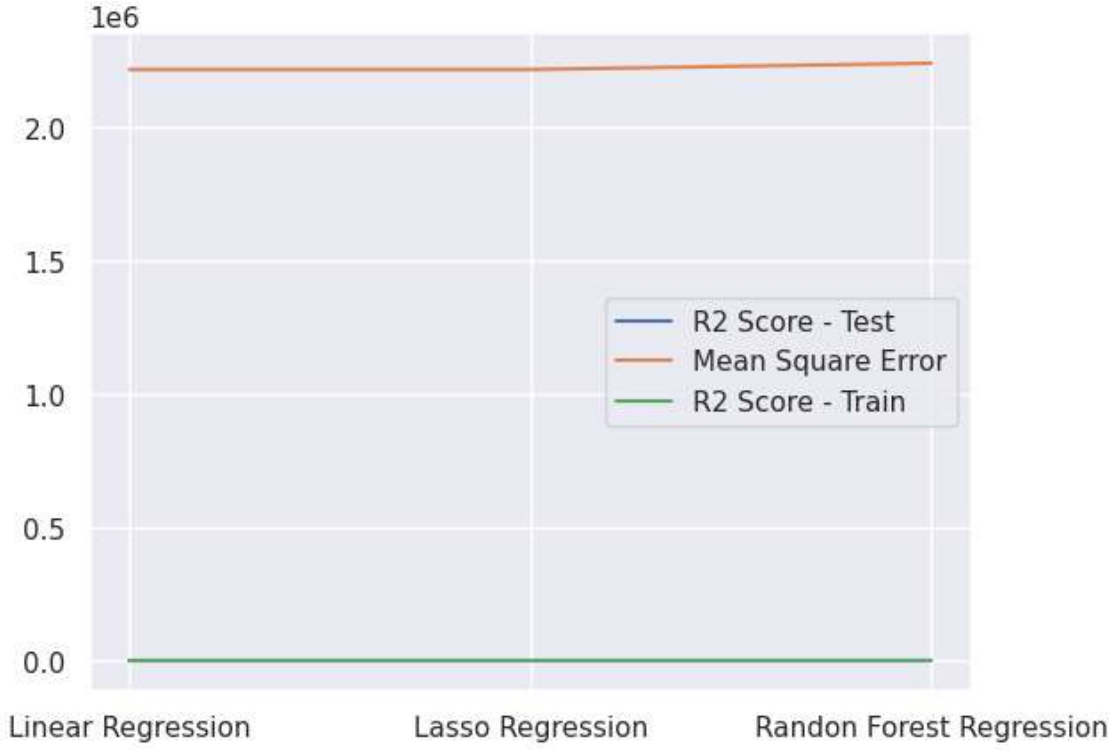
From the above bar graph, we can infer that polarity_value is the most important feature in price prediction.

We then plotted the predicted values against the true values for both sets.

Points that lie on or near the diagonal line means that the values predicted by the Random Forest Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

Evaluation of Models

Validation of model performance is done using Train/Test Set Split in which the data set is split into 80% : 20%.



The Most Important Feature of a Listing

After evaluating several models, we have determined that Random Forest Regression performs the best at predicting prices compared to other models. We proceed to analyze which feature is most important for this task by using a technique called TreeInterpreter.

This method decomposes the Random Forest prediction into a sum of contributions from each feature, where a positive contribution indicates a positive impact and a negative contribution indicates a negative impact.

(Prediction = Bias + Feature1 x Contribution1 + ... + FeatureN x ContributionN.)

To ensure the reliability of these feature contributions, we verify that the prediction of price is accurate compared to the actual value.

In the aforementioned example, the predicted value for instance 1235 closely aligns with its true value, indicating a good prediction. Upon analyzing the feature contributions using TreeInterpreter, we observe that having certain amenities such as a room_type_Entire_home_apt is influential in determining the price of the property.

However, the feature 'number_of_reviews' has a negative contribution for this particular instance, despite previous analysis indicating its significance as a predictor. This is because the instance has a bedroom value of 0. Nonetheless, 'number_of_reviews' remains an important feature for property listings, despite its negative contribution in this instance.

Conclusion

By comparing the R^2 values for the train sets of various regression models, we can determine which model is most accurate in predicting the price of a property based on the variables of room_type, polarity_value and number_of_reviews. In this analysis, the Random Forest Regression model demonstrated the highest accuracy, with the the highest R^2 value.

Furthermore, visualizing the True Values vs. Predicted Values graphs for each model provides a rough indication of their predictive performance. In this case, the graph for the Random Forest Regression model shows a higher concentration of points near the diagonal line, indicating a better fit between the true and predicted values. Therefore, we can confirm that the conclusion that the Random Forest Regression model is the most accurate is valid.

Answering the research problem

Based on the analysis conducted, we can confidently identify the factors that contribute to a property's high price. Aspiring AirBnb hosts in Chicago looking to maximize the price of their listing should prioritize the room type. Conversely, travelers looking to minimize their expenses may want to avoid listings with these features.

Other conclusions drawn are:

Entire properties listed instead of just a single room fetch the highest prices. The reviews a listing gets (quality or quantity) does not have much of an impact in its price. The polarity_value influences the price of a listing, which means that the review's sentiment positively affects the price of a listing.

Classification

Model 1: Logistic Regression

Logistic regression is a statistical model used to analyze the relationship between a binary dependent variable (such as yes/no, true/false, etc.) and one or more independent variables, by estimating probabilities using a logistic function. It is commonly used for classification tasks where the outcome variable is a discrete binary value.

Methodology

We used the logistic regression model to predict the sentiment labels (positive or negative) in the given dataset. The relevant columns were selected, and the dataset was split into training and testing sets. The logistic regression model was then created, fitted to the training data, and used to make predictions on the test data. Finally, performance metrics such as the confusion matrix, classification report, and accuracy score were calculated to assess the model's performance.

RESULT : The results of the Logistic Regression model show that it performs quite well in predicting the sentiment labels (positive or negative) using the number of reviews and price as

features, with an overall accuracy of 97.85%. However, the classification report indicates a significant difference in performance between the two classes. The model has a high precision, recall, and f1-score for the positive class, but these metrics are much lower (nearly zero) for the negative class. This could be due to the imbalance between the two classes in the dataset.

Model 2: Decision Trees

A decision tree classifier can help identify the most important features that affect the sentiment of the listings. By analyzing the structure of the decision tree, you can gain insights into how the features interact with each other and how they contribute to the prediction of the sentiment label. This information can be valuable for hosts in understanding what factors might lead to positive or negative sentiment and for guests in identifying potential issues that might impact their experience.

In this analysis, we use a decision tree to predict the sentiment label ('positive' or 'negative') based on the features such as 'number_of_reviews' and 'price'. The algorithm will learn the best splits of the data based on these features to accurately predict the sentiment label.

Methodology

We created a decision tree model and fitted it to the training data. We then used the model to make predictions on the test data and calculated performance metrics, such as the confusion matrix, classification report, and accuracy score, to assess the model's performance.

RESULT : The results of the Decision Tree Classifier show that the model performs quite well in predicting the sentiment labels (positive or negative) using the number of reviews and price as features, with an overall accuracy of 97.84%.

However, it's important to note that the classification report indicates a significant difference in performance between the two classes. The model has a high precision, recall, and f1-score for the positive class, but these metrics are much lower for the negative class. This could be due to the imbalance between the two classes in the dataset.

Model 3: Random Forest Classifier

Random Forest Classifier is an ensemble learning method used for both classification and regression tasks. It works by constructing multiple decision trees during the training phase and outputting the class that is the mode of the classes (classification) or the mean prediction (regression) of the individual trees.

In this analysis, we use the Random Forest Classifier to predict the sentiment label ('positive' or 'negative') based on features like 'number_of_reviews' and 'price'. The algorithm will learn multiple decision trees and combine their predictions to improve the overall accuracy and stability of the model.

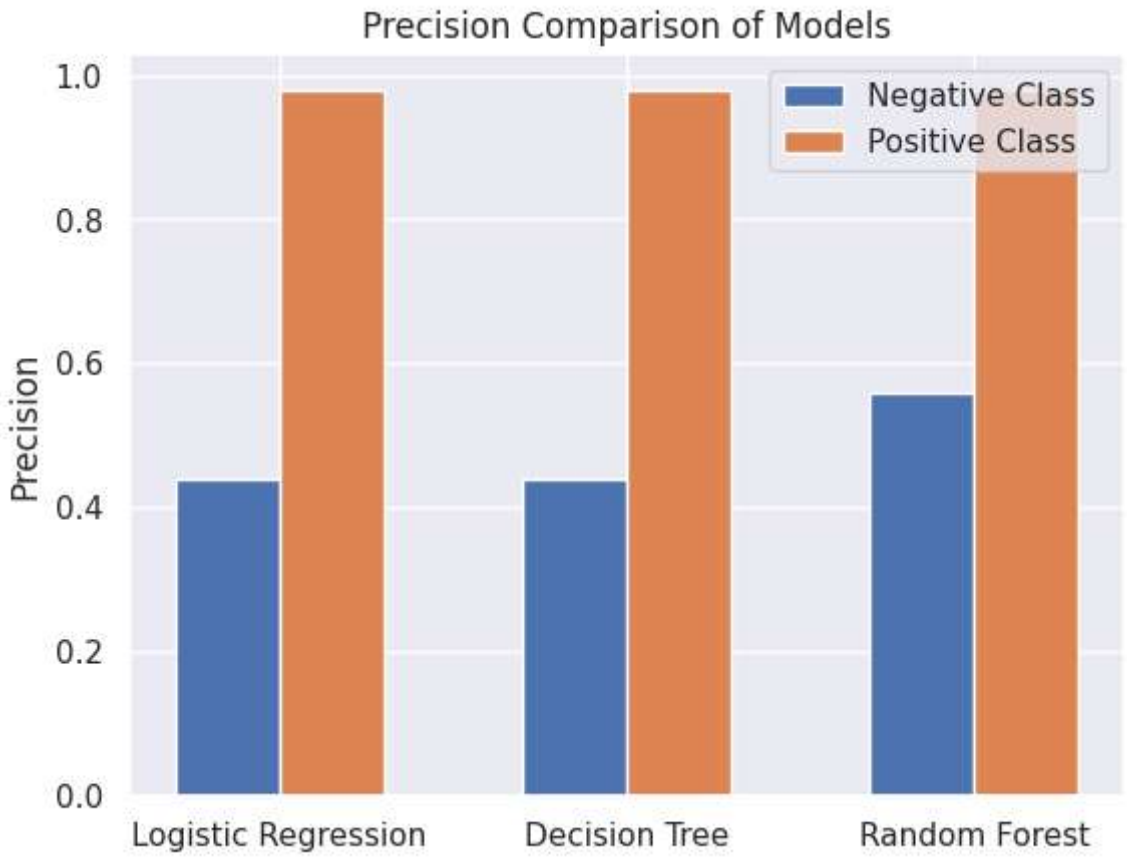
Methodology

We created a random forest model and fitted it to the training data. We then used the model to make predictions on the test data and calculated performance metrics, such as the confusion matrix, classification report, and accuracy score, to assess the model's performance.

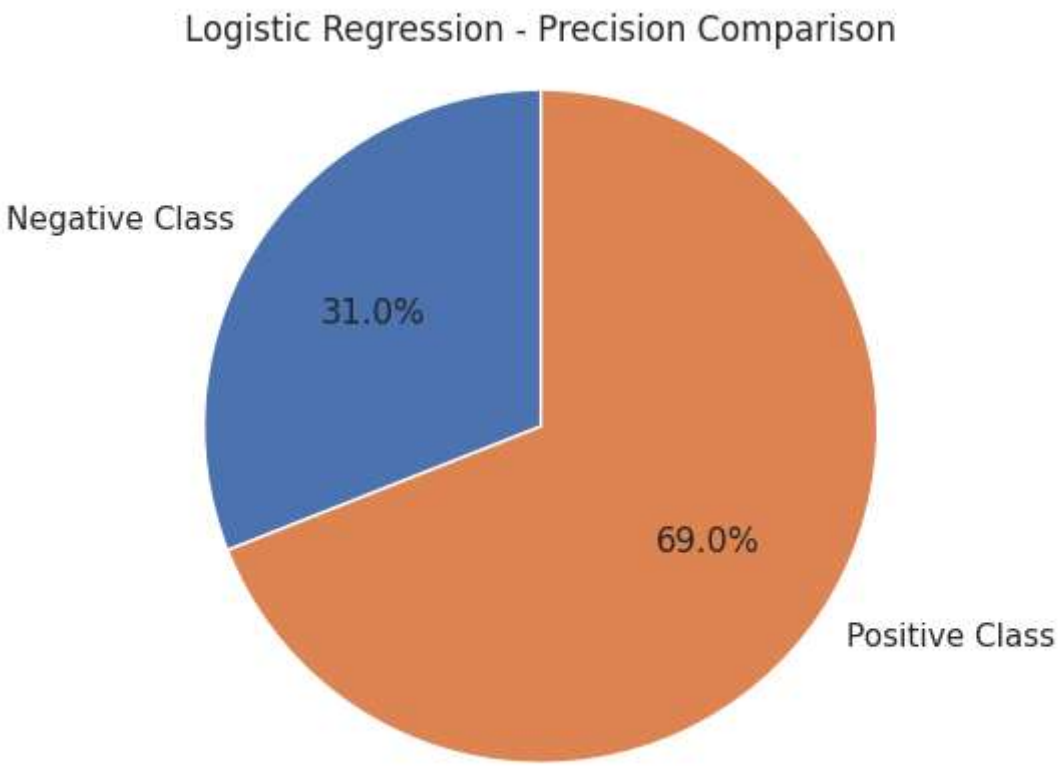
RESULT : The results of the Random Forest Classifier show that the model performs quite well in predicting the sentiment labels (positive or negative) using the number of reviews and price as features, with an overall accuracy of 97.85%.

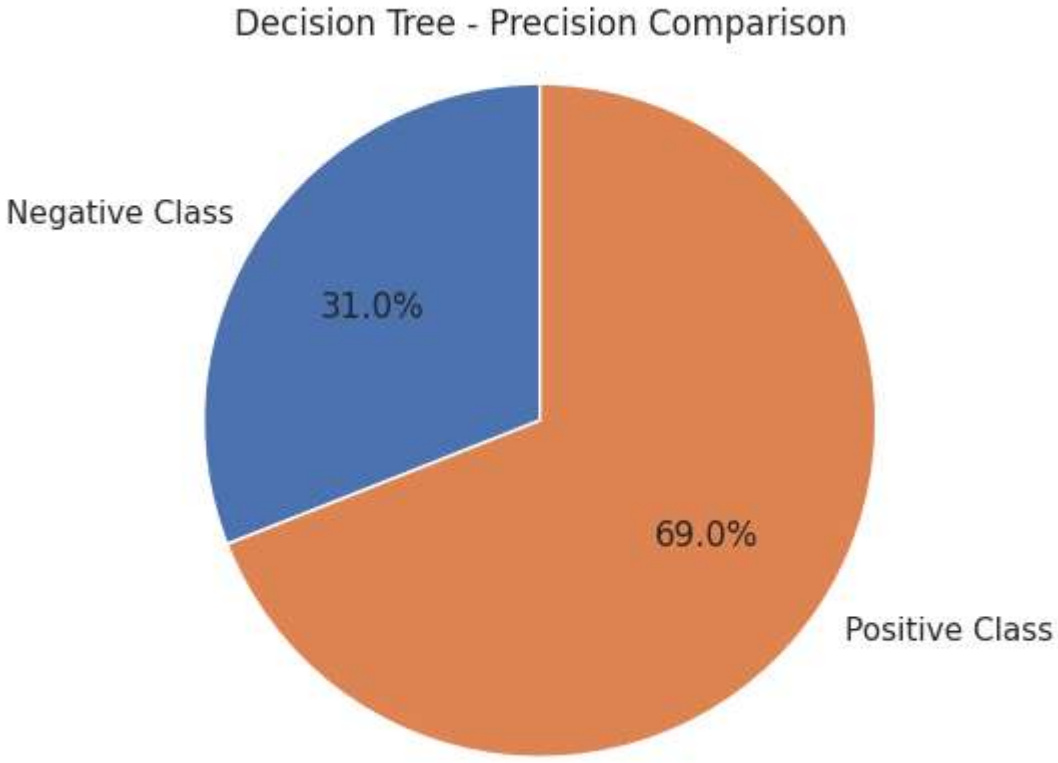
VISUALISATION

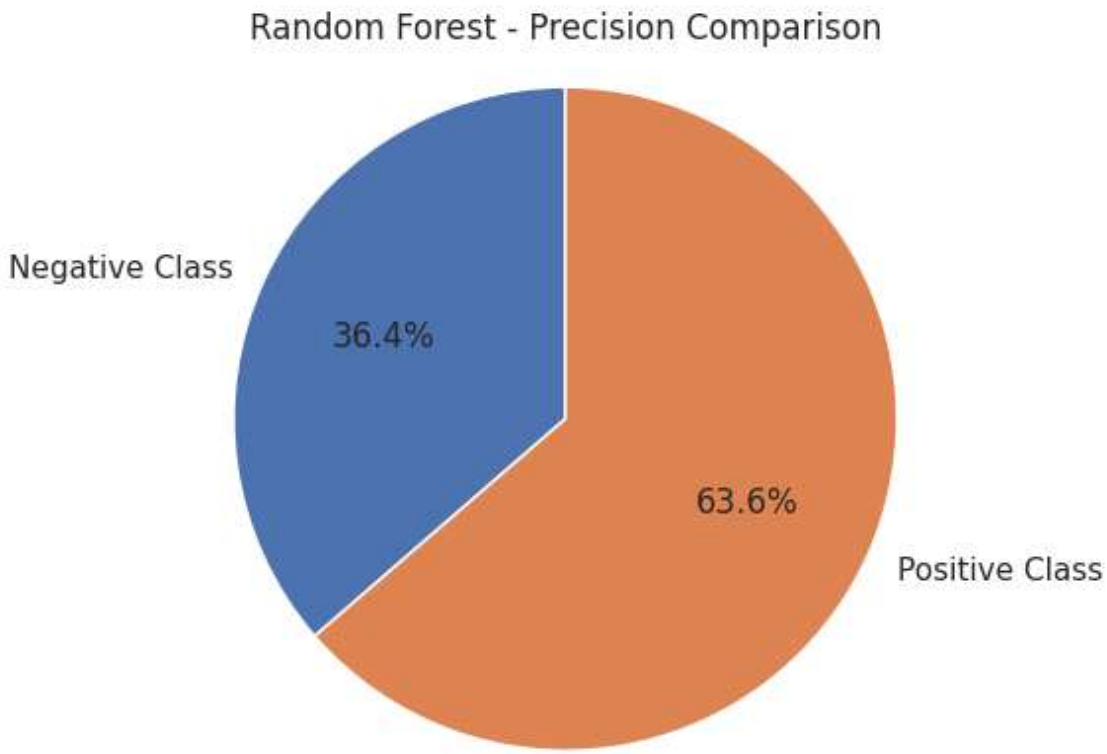
The resulting bar plot shows the precision values for the negative and positive classes in the Logistic Regression, Decision Tree, and Random Forest models. From the visualization, you can observe that the precision for the positive class is quite high (around 0.98) across all three models. However, the precision for the negative class is very low (almost 0) for the Logistic Regression and Random Forest models, while it is slightly better (around 0.44) for the Decision Tree model. This indicates that the Decision Tree model performs better in predicting the negative class compared to the other two models.



The resulting pie charts show the precision values for the negative and positive classes in the Logistic Regression, Decision Tree, and Random Forest models.

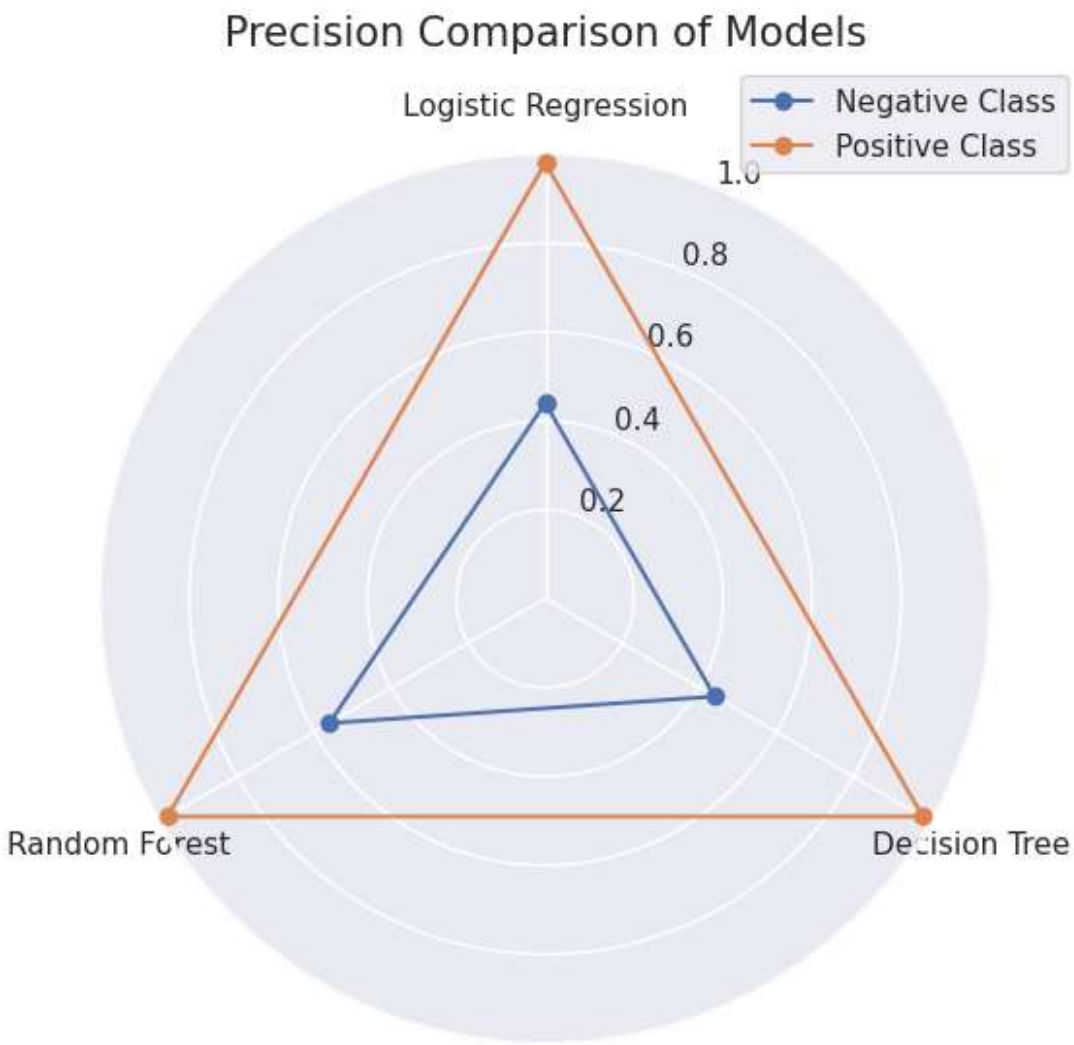

Logistic Regression - Precision Comparison
Negative Class
31.0%
69.0%
Positive Class

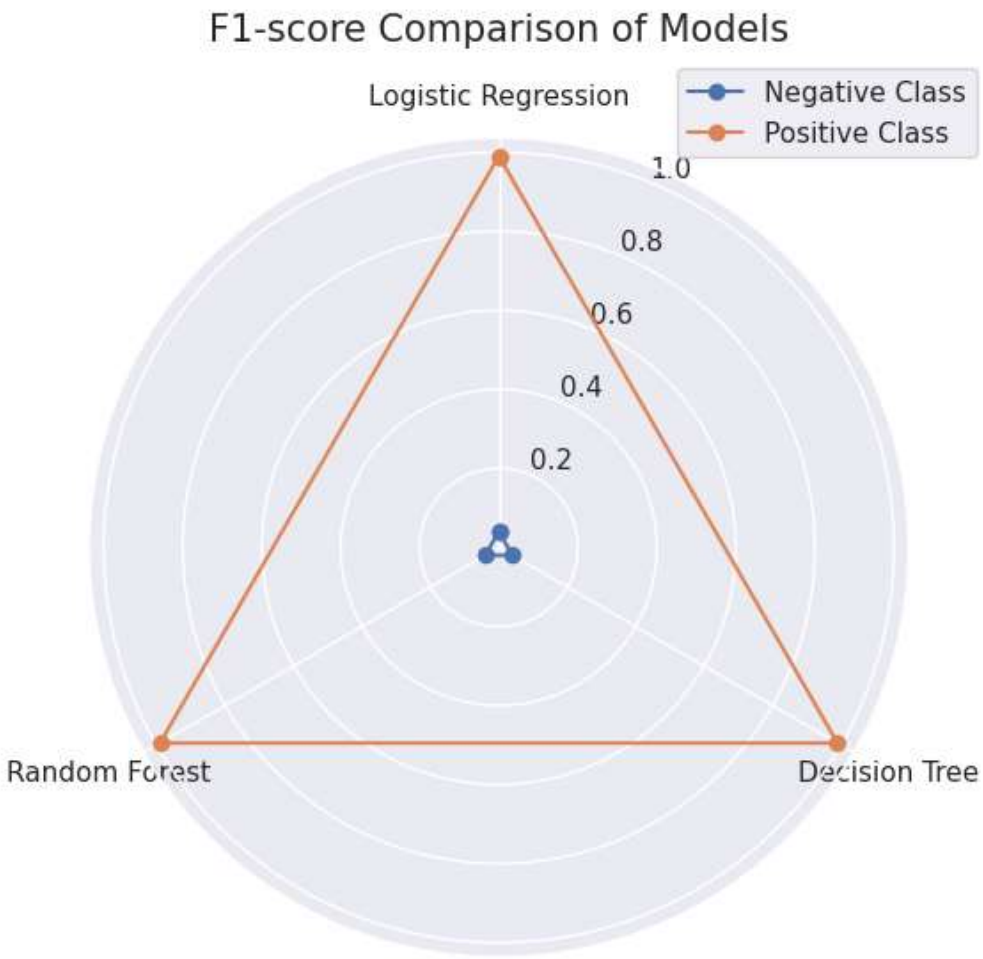

Decision Tree - Precision Comparison
Negative Class
31.0%
69.0%
Positive Class

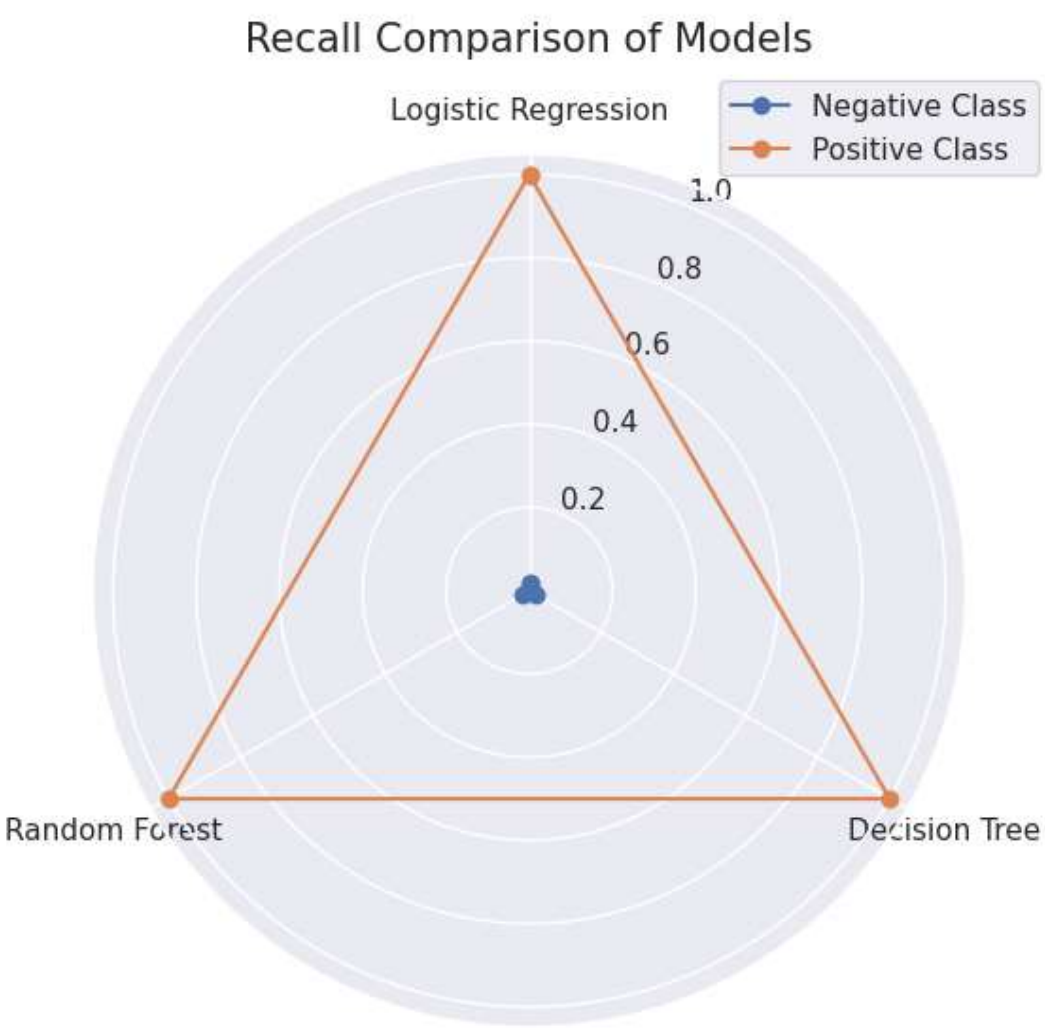


The resulting radar charts show the precision, recall, and f1-score values for the negative and positive classes in the Logistic Regression, Decision Tree, and Random Forest models.

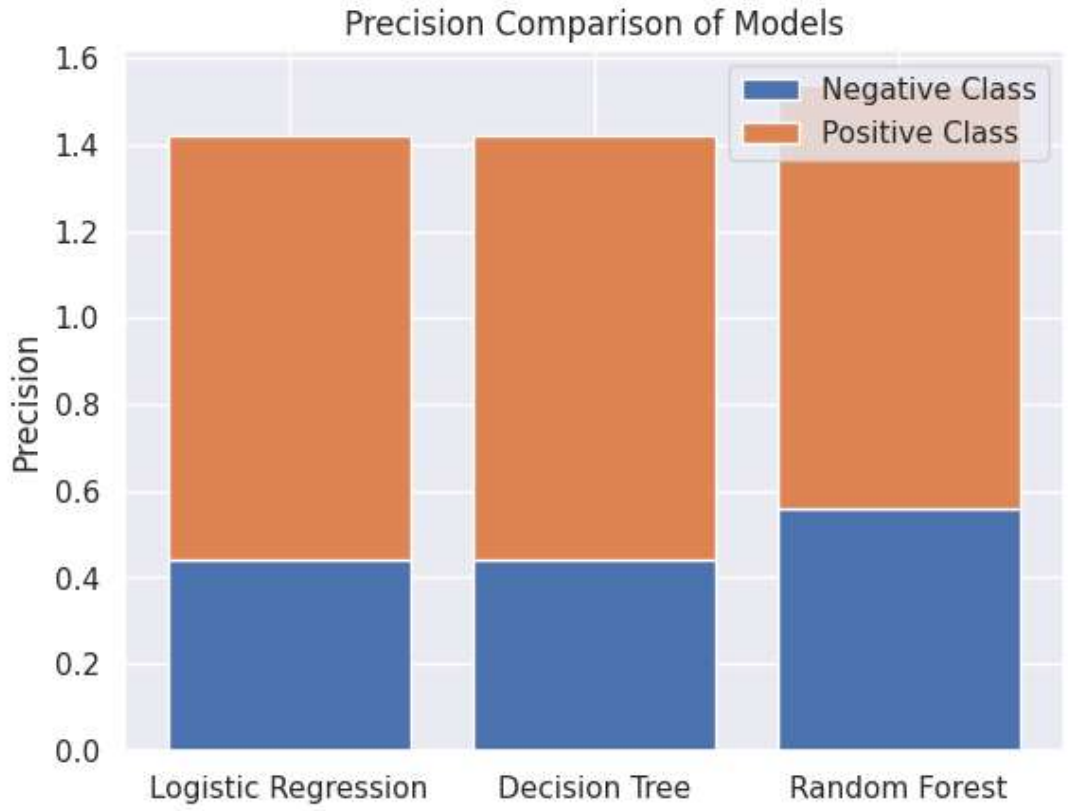
From the visualization, you can observe that the metrics for the positive class are quite high across all three models. However, the metrics for the negative class are very low for the Logistic Regression and Random Forest models, while they are slightly better for the Decision Tree model.


Precision Comparison of Models
Logistic Regression
Negative Class
Positive Class
1.0
0.8
0.6
0.4
0.2
Random Forest
Decision Tree

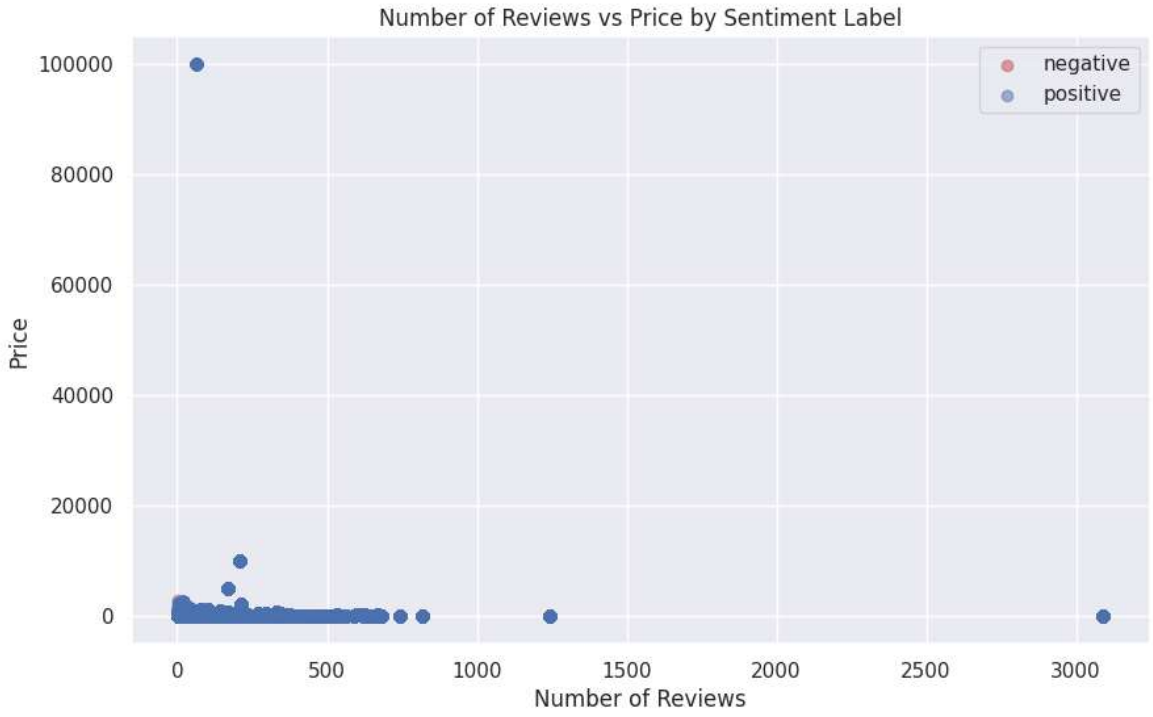

F1-score Comparison of Models
Negative Class
Positive Class
Logistic Regression
1.0
0.8
0.6
0.4
0.2
Random Forest
Decision Tree



The resulting stacked bar charts show the precision, recall, and f1-score values for the negative and positive classes in the Logistic Regression, Decision Tree, and Random Forest models.



Scatter plot comparing the number of reviews to the price for each sentiment label (negative and positive). The negative sentiment data points are plotted in red, while the positive sentiment data points are plotted in blue. From the visualization, you can analyze the relationship between the number of reviews, price, and the sentiment label of Airbnb listings.



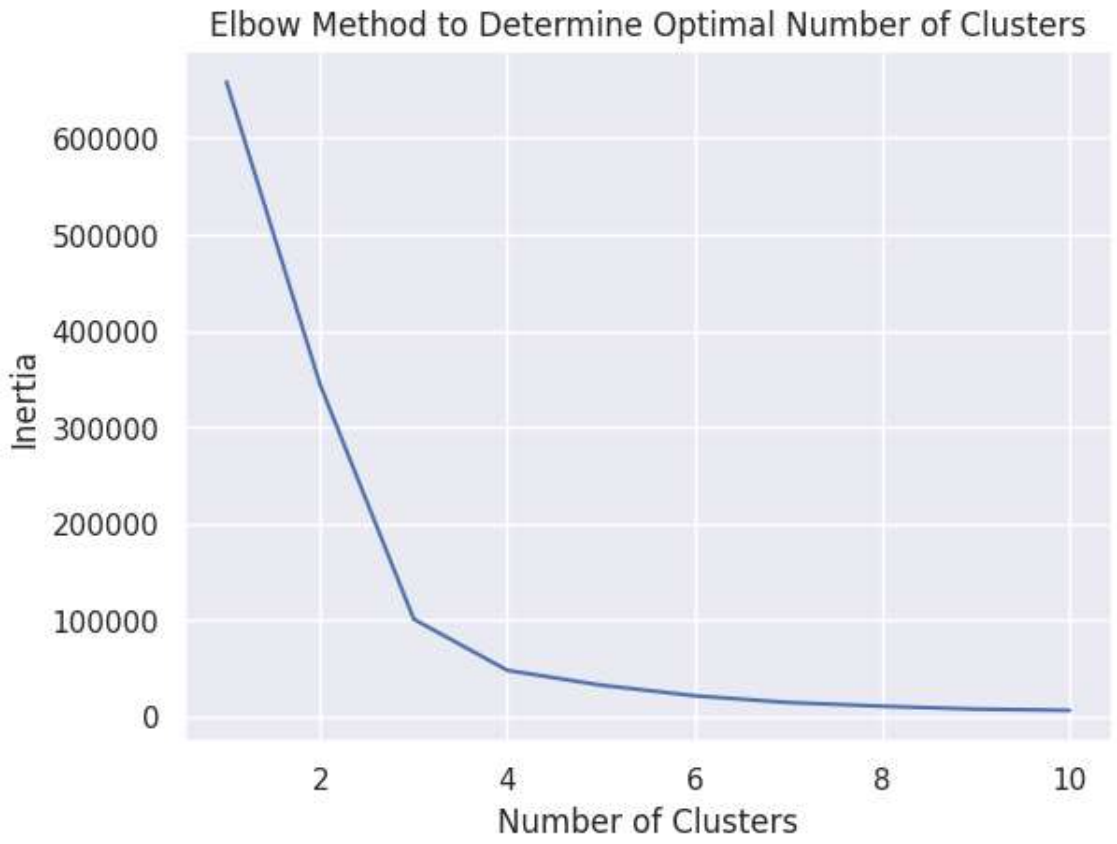
First, the optimal number of clusters is determined using the elbow method. In this case, the optimal number of clusters is set to 3. Then, the K-means clustering algorithm is applied to the normalized features, and the clusters are visualized.

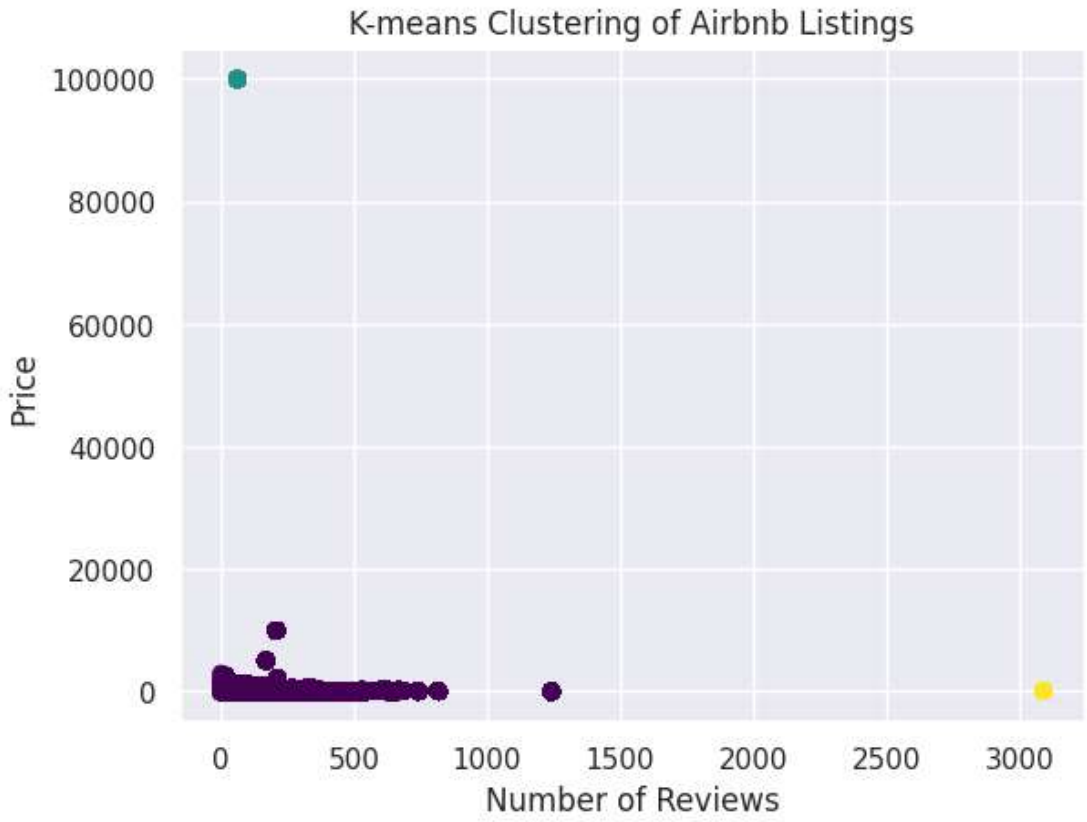
Here's a description of the results:

Elbow Method: The plot of inertia (sum of squared distances between data points and their corresponding cluster centroids) against the number of clusters shows an "elbow" point at 3 clusters. This suggests that 3 is the optimal number of clusters for this dataset.

K-means Clustering: The K-means clustering algorithm is applied with the optimal number of clusters (3). The resulting clusters are added to the original DataFrame as a new column named 'cluster'.

Visualization: The clusters are visualized as a scatter plot of 'number_of_reviews' against 'price' with the data points colored according to their cluster label. The plot reveals three distinct clusters, which could represent different types of Airbnb listings, such as budget and popular rentals.





In conclusion, the K-means clustering algorithm has successfully grouped the Airbnb listings into three clusters based on their price and number of reviews. This can help hosts and guests better understand the types of listings available and identify potential patterns or trends in the data.

Answering the research questions

Predicting the sentiment label of a listing solely based on its price and the number of reviews it has received.

We built three different supervised learning models (Logistic Regression, Decision Tree, and Random Forest Classifier) to predict the sentiment label ('positive' or 'negative') of a listing based on its price and the number of reviews it has received. All three models show a high accuracy in predicting the positive class, but struggle with the negative class due to the imbalanced nature of the dataset. Among the three models, the Random Forest Classifier seems to perform better for the negative class, but there is still room for improvement. Based on these models, it's possible to predict the sentiment label of a listing using only price and the number of reviews, but the prediction quality for negative sentiment labels is limited.

How are Airbnb listings distributed based on their price and popularity?

We applied K-means clustering on the dataset to understand the distribution of Airbnb listings based on their price and popularity (number of reviews). The elbow method suggested that the

optimal number of clusters is 3. The resulting clusters, visualized as a scatter plot, show three distinct groups of Airbnb listings based on price and popularity.

Cluster 1: Listings with a low to moderate number of reviews and low to moderate prices. This cluster could represent budget or average-priced listings that have a moderate level of popularity.

Cluster 2: Listings with a high number of reviews and low to moderate prices. This cluster might represent popular and affordable listings that receive a high number of bookings and reviews.

Cluster 3: Listings with a low to moderate number of reviews but higher prices. This cluster could represent premium or luxury listings that have fewer bookings and reviews due to their higher price.

In conclusion, the analysis shows that Airbnb listings can be grouped into three clusters based on their price and popularity. These clusters can help hosts and guests better understand the types of listings available and make informed decisions.

References

- https://www.datacamp.com/tutorial/techniques-to-handle-missing-data-values
- https://towardsdatascience.com/how-to-handle-missing-data-8646b18db0d4
- https://www.datacamp.com/blog/classification-machine-learning
- https://www.datacamp.com/tutorial/tutorial-lasso-ridge-regression
- https://towardsdatascience.com/random-forest-regression-5f605132d19d
- https://towardsdatascience.com/decision-tree-classifier-explained-in-real-life□picking-a-vacation-destination-6226b2b60575