**Project Presentation: Airbnb Price Prediction Model**

Introduction

This project focuses on building a price prediction model for Airbnb listings in Madrid. Our goal was to analyze Airbnb data, engineer features, build a machine learning model, and create a user-friendly web application for predicting listing prices. The entire project covers data preprocessing, exploratory data analysis, feature engineering, model training, evaluation, hyperparameter tuning, and deployment using Streamlit.

Data Overview

We used two main datasets: listings.csv and listings\_detailed.csv. These datasets contain comprehensive information about Airbnb listings, including location, amenities, room type, and reviews. The datasets were merged on the id column to create a unified dataset for analysis.

Data Cleaning and Preprocessing

1. Handling Missing Data:
   * We identified and removed missing values in critical columns like Superhost.
   * Numerical columns with missing values were filled using the median to ensure robust statistics.
2. Handling Categorical Variables:
   * Binary categorical features like Superhost and Available were label-encoded (converted to 0 and 1).
   * Multi-class categorical features, such as Room Type and Economic Class, were one-hot encoded.
3. Feature Engineering:
   * We classified neighborhoods into four economic classes (High Class, Upper-Middle, Lower-Middle, and Low) based on their socioeconomic status.
   * We created binary columns for high-value amenities like Has\_Pool, Has\_Wifi, Has\_Kitchen, and Has\_Elevator to capture their influence on the price.

Exploratory Data Analysis (EDA)

1. Outlier Detection:
   * We used boxplots to identify outliers in the Price column and removed them using the Interquartile Range (IQR) method.
2. Visualizing Relationships:
   * Created a scatter plot to explore the relationship between Capacity (number of guests) and Price.
   * Plotted a correlation heatmap to examine the relationships between numerical features like Price, Capacity, Guest Satisfaction, and Cleanliness Rating.
3. Price Distribution:
   * A histogram of the Price column with a KDE line was used to visualize the price distribution and detect skewness.

Machine Learning Model Building

1. Data Preparation:
   * We selected key numerical features for scaling using StandardScaler to ensure they are on the same scale.
   * The data was split into training and testing sets (80% training, 20% testing).
2. Model Selection and Evaluation:
   * We evaluated several regression models: Linear Regression, Random Forest, Gradient Boosting, Decision Tree, Support Vector Regressor, and XGBoost.
     + Metrics used for evaluation included Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R² (coefficient of determination).
   * Among the models, XGBoost performed the best, showing the lowest RMSE and highest R² score.
3. Hyperparameter Tuning:
   * We applied GridSearchCV to fine-tune the parameters of the Gradient Boosting and XGBoost models.
   * The tuned XGBoost model achieved the best results, with further improvement in RMSE and R² scores.
4. Cross-Validation:
   * We used 5-fold cross-validation to validate the robustness of the Gradient Boosting and XGBoost models, confirming the reliability of the model’s performance.

Model Deployment

We deployed the final XGBoost model using Streamlit, a Python library for building interactive web applications. The app allows users to input listing details and receive an estimated nightly price.

Streamlit App Features

1. User Input:
   * The app collects inputs such as location, capacity, room type, and various amenities from the user via a sidebar interface.
2. Neighborhood Images:
   * The app dynamically displays images of the selected neighborhood to provide additional context for the user.
3. Price Prediction:
   * Using the trained XGBoost model, the app predicts the nightly price based on the user inputs and displays the result.

Key Visualizations in Tableau

We created an interactive Tableau dashboard with the following visualizations:

1. Map of Average Price by Neighborhood:
   * A map view displaying the average listing price across different neighborhoods in Madrid.
2. Price Distribution Histogram:
   * A histogram showing the distribution of prices, with a KDE line for smoother representation.
3. Guest Satisfaction vs. Price Bubble Chart:
   * A bubble chart visualizing the relationship between guest satisfaction scores and listing prices by neighborhood.
4. Room Type Distribution:
   * A pie chart or bar chart showing the frequency of different room types in the dataset.

Results

1. The XGBoost model showed the best performance with the following metrics:
   * RMSE: Low value indicates accurate predictions.
   * MAE: Low value confirms the model’s accuracy in predicting close to actual values.
   * R² Score: High value indicates that the model explains a significant portion of the variance in listing prices.
2. The interactive Streamlit app provides an easy-to-use interface for predicting Airbnb prices, making it accessible for potential hosts and users to estimate their listing price.

Conclusion

This project effectively combined data analysis, machine learning, and web application development. By building a robust predictive model and deploying it through an interactive app, we showcased the practical application of machine learning in the Airbnb rental market. The Tableau dashboard further enhances the project by providing visual insights into key metrics, allowing stakeholders to explore and understand the data more effectively.

Next Steps

* Model Improvement: Further fine-tuning and experimenting with advanced models like LightGBM or CatBoost.
* Additional Features: Integrating more features like seasonal trends or average booking length to improve predictions.
* App Enhancement: Adding more user interactivity, such as dynamic filtering options or additional charts in the Streamlit app.