**CIS 550- Advanced Machine Learning**

**Final Project Summary Report**

Sneha Gajanan Sonkusare(43)

Maria Rabel(35)

Sanjay Dhayalan(10)

**Project Description: Flight Price Prediction Using Random Forest**

**Introduction:**

Air travel has become an essential part of modern life, connecting people across cities and countries for work, leisure, and personal commitments. However, one persistent challenge for travelers is the unpredictability of flight ticket prices. These prices often fluctuate significantly based on a variety of factors, such as demand, competition, seasonality, airline policies, and economic conditions. This variability creates uncertainty for travelers who aim to book tickets at the most economical rates.

The Flight Price Prediction Using Random Forest project addresses this challenge by developing a machine learning model to accurately predict flight ticket prices. Using Random Forest, an ensemble-based machine learning algorithm, this project seeks to analyze historical flight data to uncover patterns and relationships that influence pricing. By understanding these trends, the model can predict future prices, allowing travelers to make cost-effective booking decisions.

The study also provides valuable insights for airlines and travel platforms. Airlines can use such models to forecast demand, optimize pricing strategies, and remain competitive. Similarly, travel platforms can enhance their offerings by integrating predictive tools to improve user experience and build trust through price transparency.

**Importance**

The Flight Price Prediction Using Random Forest project holds significant importance for various stakeholders in the travel ecosystem, including travelers, airlines, and travel platforms. Here’s why this project matters:

**1. For Travelers:**

* **Cost Savings**: By accurately predicting flight ticket prices, travelers can identify the best times to book tickets, avoiding overpaying and securing better deals.
* **Informed Decision-Making**: The project provides insights into how flight prices are influenced by factors such as travel dates, number of stops, and airlines, enabling smarter choices.
* **Trip Planning Optimization**: Travelers can use predictions to align their booking decisions with their budgets and schedules, reducing uncertainty.

**2. For Airlines:**

* **Demand Forecasting**: Understanding price trends helps airlines anticipate peak demand periods and adjust their operational strategies accordingly.
* **Dynamic Pricing Optimization**: Airlines can use predictive models to fine-tune pricing algorithms, ensuring competitiveness and maximizing revenue.
* **Market Insights**: Identifying the factors influencing price changes allows airlines to stay ahead of competitors and refine their pricing strategies.

**3. For Travel Platforms:**

* **Enhanced User Experience**: Platforms can integrate predictive tools to provide users with personalized recommendations and real-time price predictions.
* **Transparency and Trust**: Accurate predictions foster trust among users by offering them clarity on price fluctuations and booking strategies.
* **Increased Engagement**: Predictive features can make travel platforms more engaging and valuable to users, improving customer retention.

**4. Broader Impact:**

* **Encourages Data-Driven Solutions**: The project showcases the power of machine learning in solving complex real-world problems, inspiring further innovation in predictive analytics.
* **Improves Accessibility**: By making airfare trends more predictable, the project helps travelers across different income levels access affordable flights, democratizing air travel.
* **Academic and Research Contributions**: The project serves as a valuable case study for the application of advanced machine learning algorithms like Random Forest in practical scenarios.

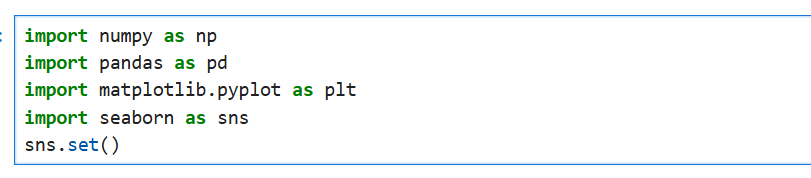
**Problem Statement**

Flight ticket prices are dynamic and influenced by factors like seasonality, demand, airline policies, and route popularity. This unpredictability makes it challenging for travelers to plan trips cost-effectively. The goal of this study is to build an accurate predictive model that provides insights into flight pricing trends.

This project demonstrates the potential of machine learning in solving real-world problems. It empowers travelers with data-driven insights to save money and plan efficiently while showcasing the practical applications of advanced machine learning algorithms like Random Forest in the travel industry.

**Implementation:**

We will implement a Fight price prediction system using Python. Our goal here is to learn how to implement a Fight price prediction system using different Python libraries.



**NumPy**:

A library for efficient numerical computations, supporting multi-dimensional arrays and mathematical operations.

**Pandas**:

A powerful tool for data manipulation and analysis, offering structures like DataFrames for organizing and preprocessing data.

**Matplotlib**:

A versatile library for creating static, animated, and interactive visualizations in Python.

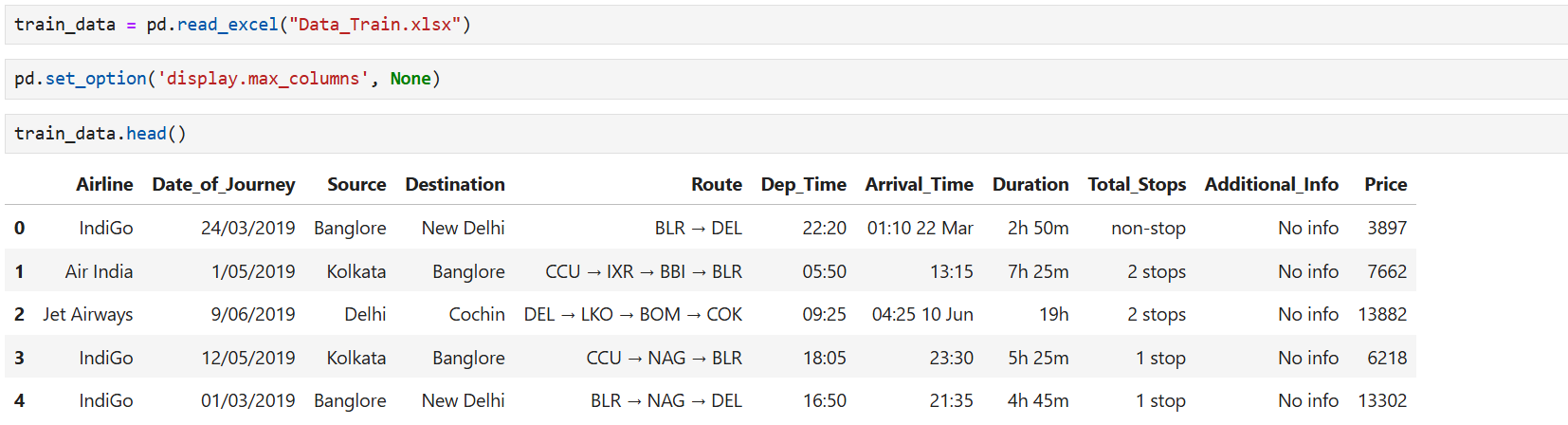
**Seaborn**:

Built on matplotlib, it simplifies creating visually appealing statistical graphics like heatmaps and distribution plots.

**Sns.set()**:

Enhances plot aesthetics by applying Seaborn’s default styling to all visualizations.

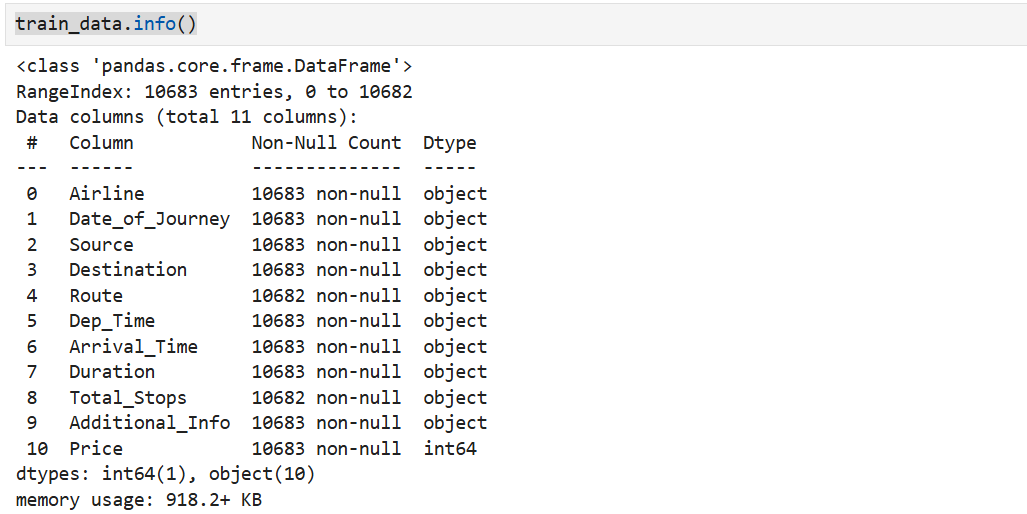
**Part 1: Analyzing the data**



The dataset Data\_Train.xlsx is read into a pandas DataFrame using pd.read\_excel(). This step imports the training data for analysis and model building.

pd.set\_option('display.max\_columns', None) is used to ensure that all columns of the DataFrame are visible when displayed. By default, pandas may truncate the output for DataFrames with many columns.

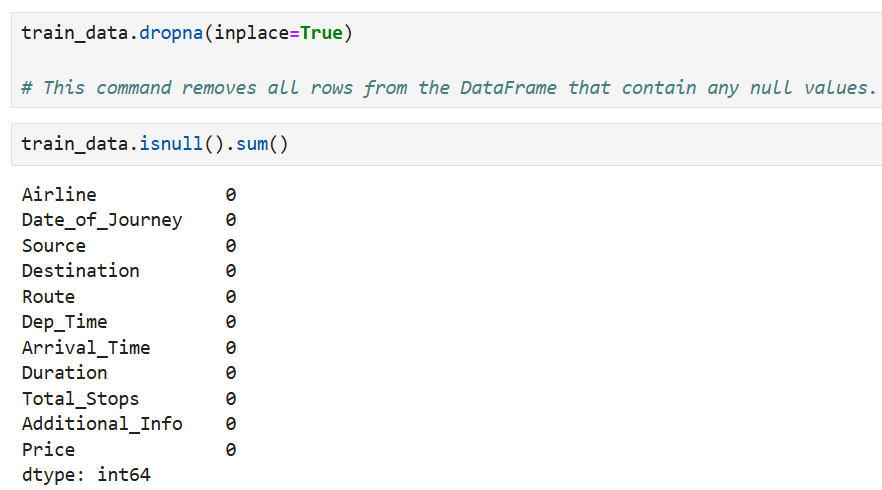
train\_data.head() displays the first five rows of the dataset. This helps in understanding the structure of the data and the type of features available.



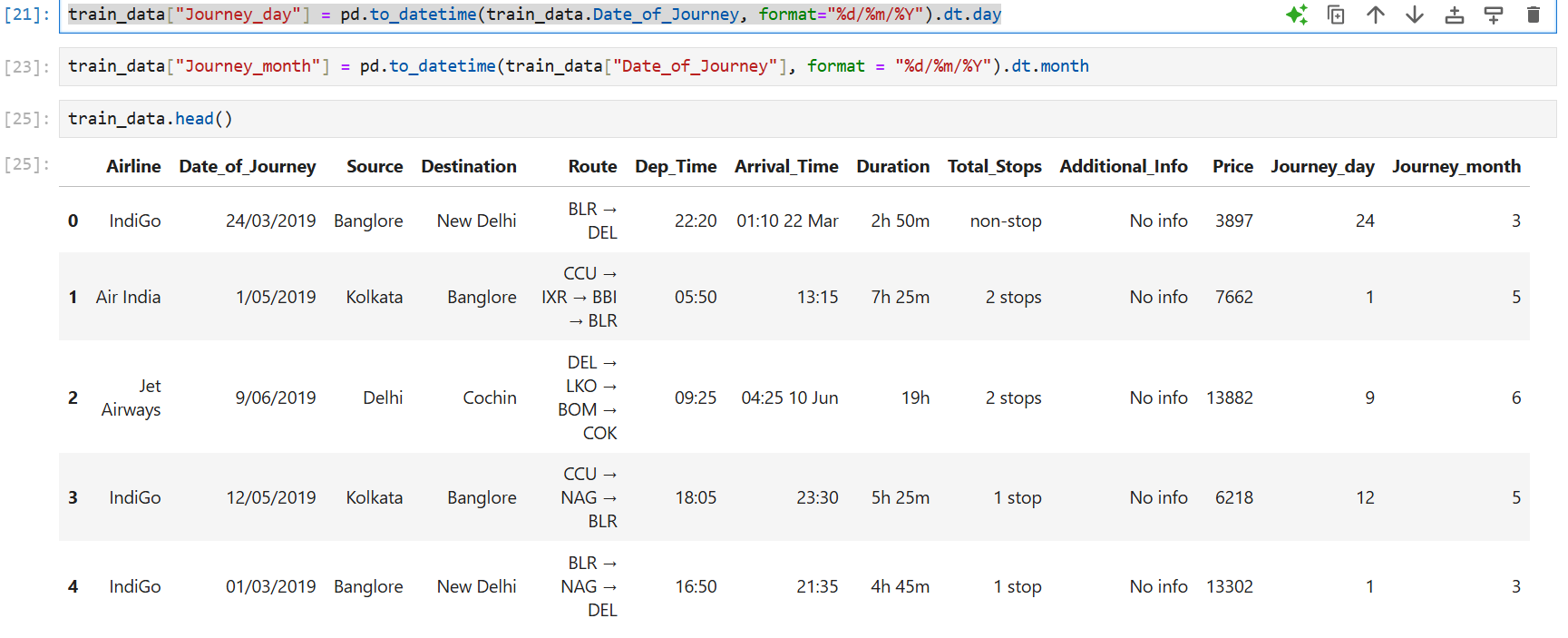
The **train\_data.info()** function summarizes the DataFrame's structure, showing the number of rows, columns, non-null values, data types, and memory usage. It helps quickly assess data quality and identify missing values.

**Part 2- Pre-processing the Data**.

* 1. **Handling the missing values**:

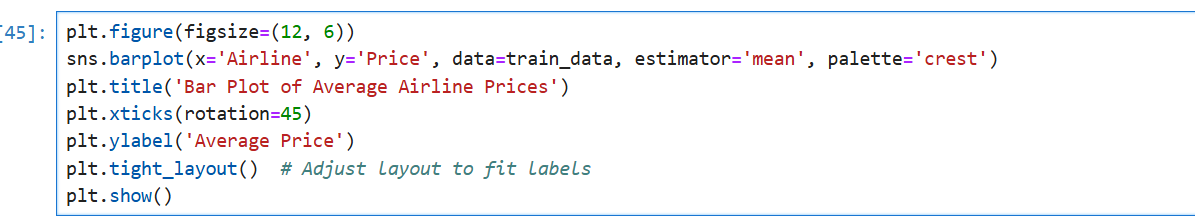


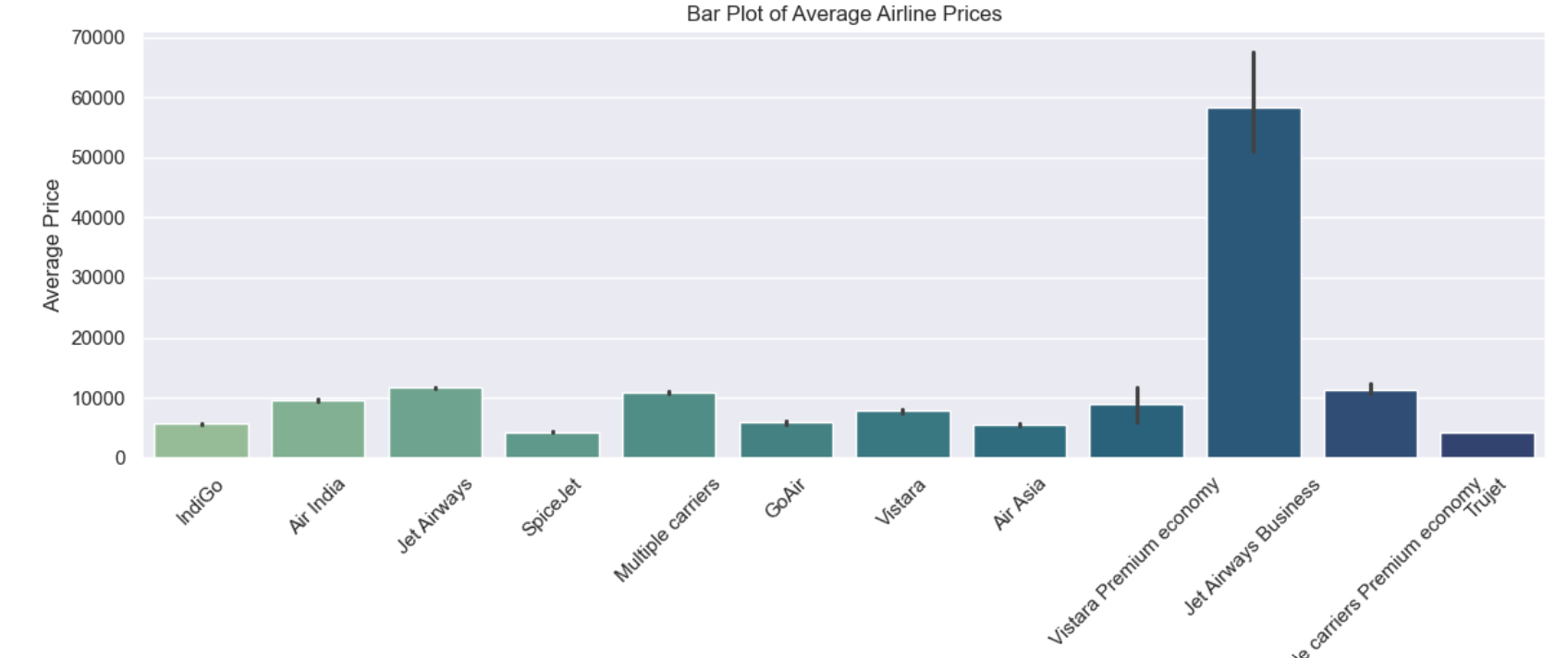
* 1. **Extract Data Feature**



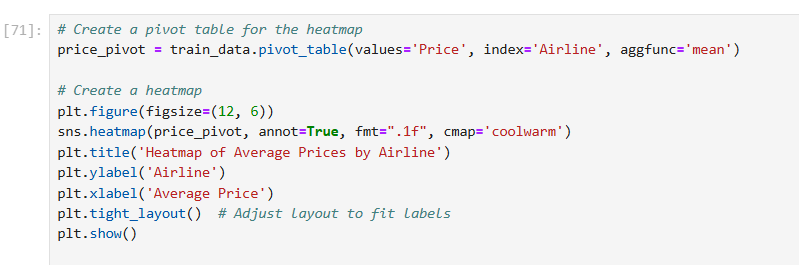
The Date\_of\_Journey column is converted into a datetime object, and new columns Journey\_Day and Journey\_Month are extracted for better analysis.

* 1. **Analyzing the data using bar graph and heat map**





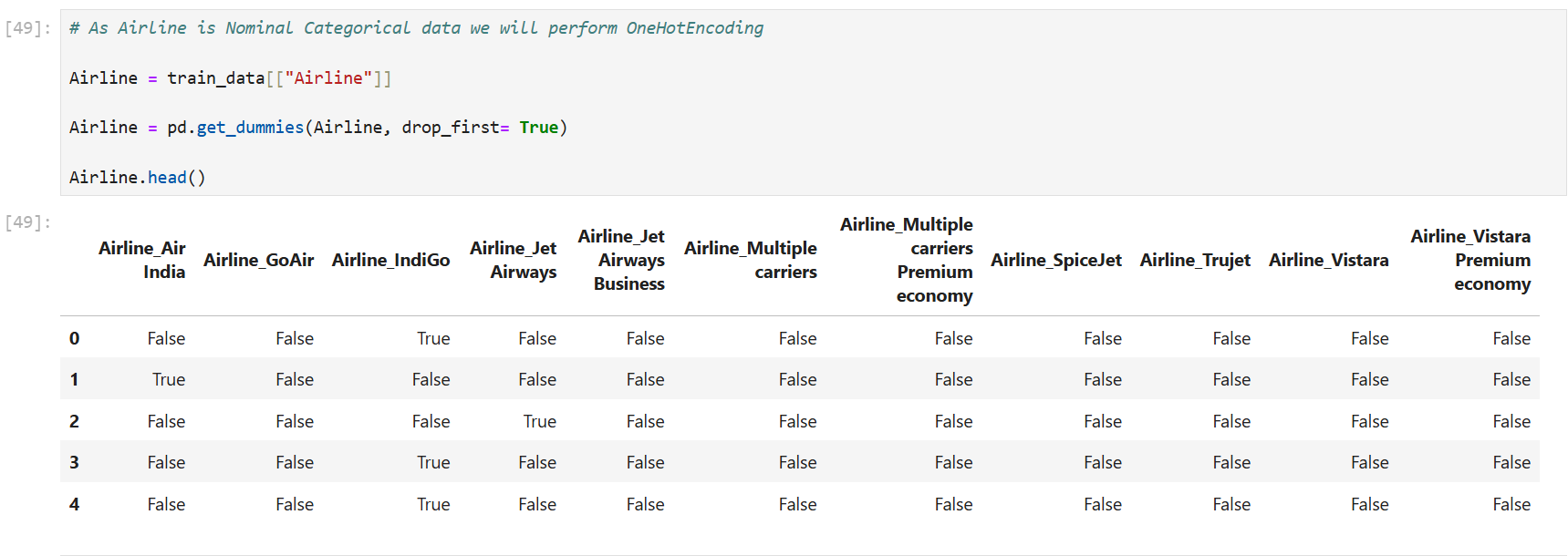
The code plots two visualizations to analyze the relationship between Airline and Price. The bar plot displays the average price for each airline, allowing for a straightforward comparison of ticket prices across airlines. Using one bar per airline, it highlights which airlines offer higher or lower average prices. The heatmap provides a visual table of the same data, where the intensity of colors represents the average prices, and annotations show the exact values. Together, these plots help identify pricing patterns and make it easy to compare prices across different airlines.





The heatmap provides a clear visual representation of the average prices for each airline, with the intensity of the colors indicating the magnitude of the prices. Heatmaps are particularly effective for displaying data in a table-like format, making it easy to spot patterns and differences. By using the annot=True argument, the exact average price values are displayed on the heatmap, providing precise insights. The cmap='coolwarm' color palette further enhances the visual comparison, with warmer colors representing higher prices and cooler colors indicating lower prices, allowing for an immediate understanding of pricing trends across airlines.

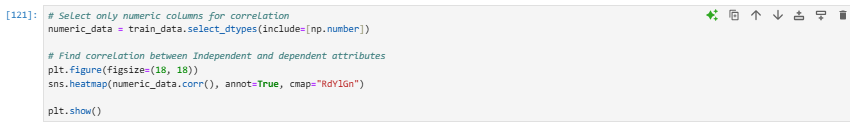
* 1. **Converting Categorical data into Numerical**

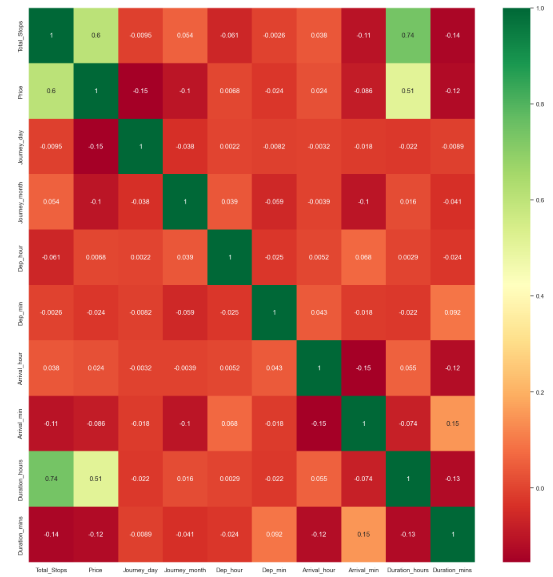




In this step, the Source column, which contains nominal categorical data, is transformed using **one-hot encoding**. This process creates binary columns (Source\_Chennai, Source\_Delhi, etc.) to represent each category in the Source column. The drop\_first=True parameter eliminates one column to avoid redundancy and multicollinearity. This transformation ensures the data is in a numerical format suitable for machine learning models.

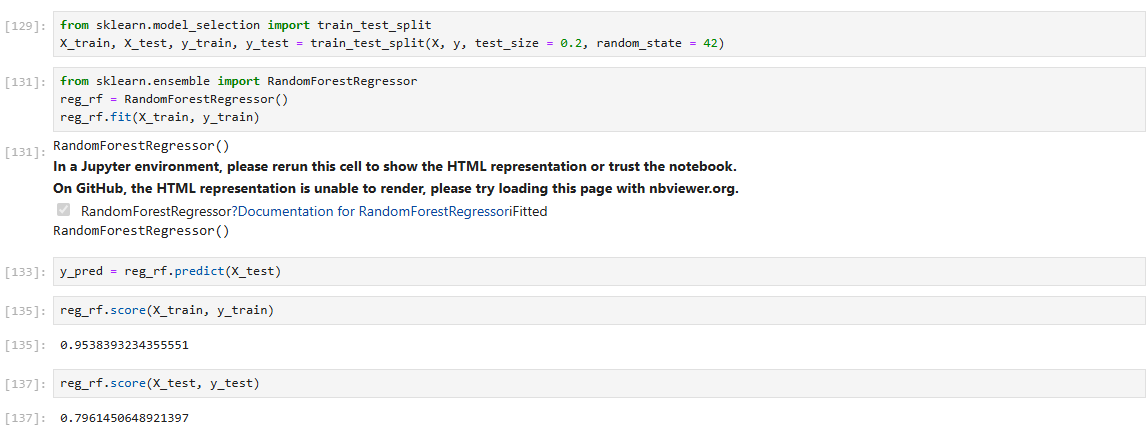
* 1. **Data Analysis**

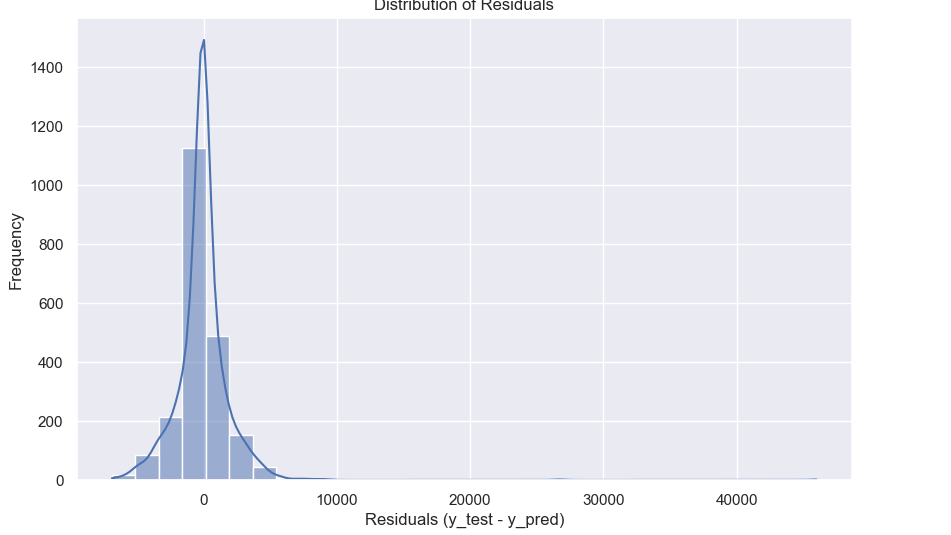




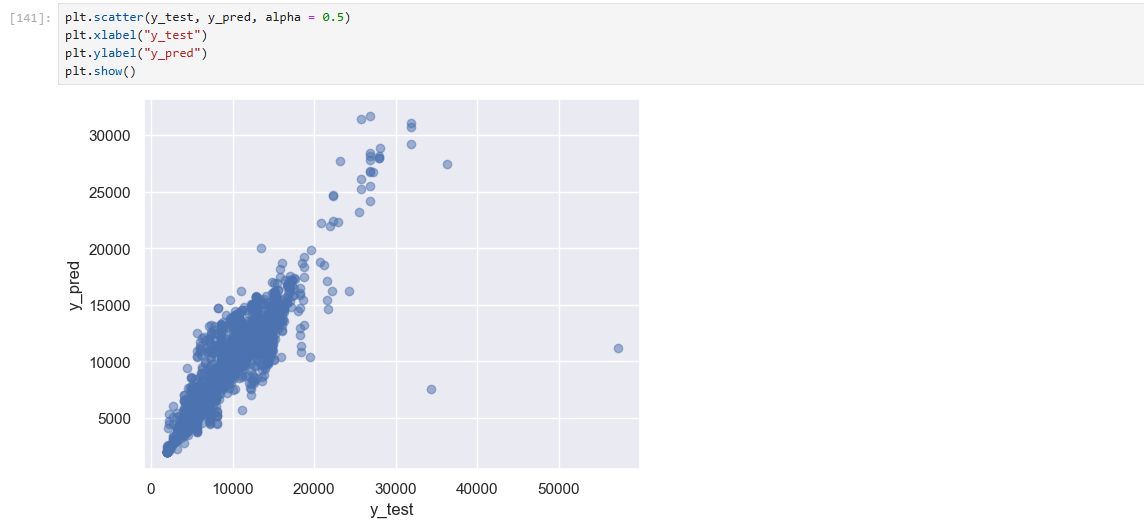
The code performs a correlation analysis on the numerical columns of the dataset. It first selects only the numeric columns and then computes the correlation matrix, which shows the relationships between the variables. The correlation values are visualized in a heatmap, with color intensity representing the strength of the correlations. This analysis helps identify patterns, such as which variables are strongly correlated with each other or with the target variable, aiding in feature election and understanding the data before model building.

**Part 3: Test-Train and Splitting the Data**



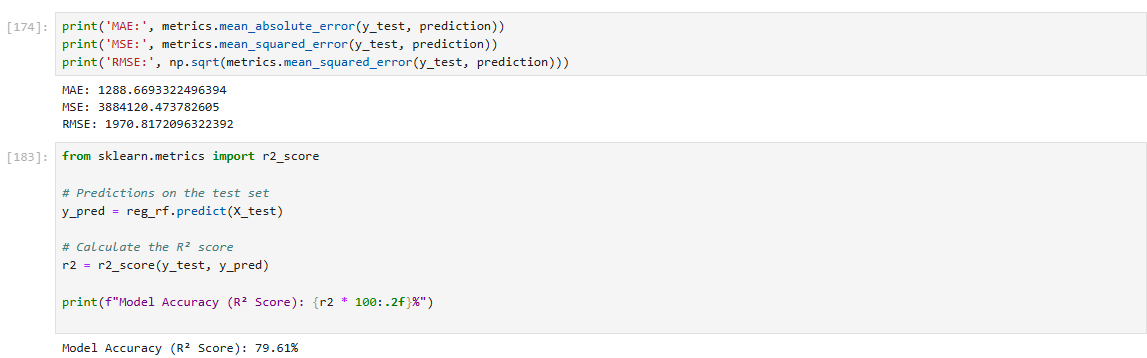


The process of training and evaluating a Random Forest regression model using Python's scikit-learn library. The dataset is split into training and test sets using train\_test\_split. A RandomForestRegressor is then trained (fit) on the training data, and predictions (predict) are made for the test data. The model's performance is evaluated using the score method, yielding R^2 scores of approximately 0.95 on the training data and 0.80 on the test data, indicating good but slightly overfitted performance. The residual distribution (the difference between actual and predicted test values). The residuals are predominantly centered around zero, with a few outliers, which suggests that the model generally performs well but may have difficulty with some predictions. The plot's shape, resembling a normal distribution, is a good indicator of balanced prediction errors.



This is a scatter plot comparing the predicted values (ypred)against the actual test values (ytest). Each point represents a data sample. The plot provides a visual representation of how well the model's predictions align with the actual values. Ideally, if the predictions are accurate, the points should align closely along the diagonal line ypred=ytest ​. In this plot, there seems to be a positive correlation, indicating that the model has learned the pattern to some extent but may have some errors, as the points are not perfectly aligned.

**Part 4- Regression Model Performance Evaluation**



Mean Absolute Error (MAE): Measures the average absolute difference between actual and predicted values. Example: MAE of $1183.93 indicates predictions are, on average, $1183.93 away from actual prices.Mean Squared Error (MSE): Calculates the average of the squared differences between actual and predicted values. Example: MSE of $4,402,946.85 highlights significant deviations.Root Mean Squared Error (RMSE): The square root of MSE, providing average prediction error in original units. Example: RMSE of $2098.32 means the average error is about $2098.32.Normalized RMSE: RMSE divided by the range of the target variable, indicating error as a percentage of the range. Example: Normalized RMSE of 0.0269 means average error is about 2.69% of the target range.R² Score: Explains the variance in actual values by the model's predictions. Example: R² of 0.7984 means the model explains about 79.84% of the variance.

**Conclusion:**

The **project** effectively addresses the challenge of predicting the dynamic and fluctuating nature of flight ticket prices, influenced by factors such as demand, airline policies, seasonality, and route popularity. By leveraging historical flight data, the model analyzes key features like airline, number of stops, flight duration, and date of journey to uncover patterns that influence price changes. The model’s performance, with an **R² score of 0.7984**, demonstrates its ability to explain a significant portion of the variance in flight prices, providing valuable insights for travelers, airlines, and travel platforms. Travelers can benefit from this predictive tool by making smarter decisions about when to book, potentially saving money by avoiding price surges. Airlines can use the model to forecast demand and optimize dynamic pricing strategies, while travel platforms can enhance user experience with real-time price predictions and personalized recommendations. However, the model does have limitations, such as its reliance on the quality and comprehensiveness of the data used for training, as well as potential issues like overfitting and computational costs, especially when scaling with large datasets. Future improvements could include incorporating real-time market data and exploring other machine learning algorithms to further enhance the model's accuracy and adaptability.

Following are the reference for the project:

* 1. <https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction>
  2. <https://www.kaggle.com/datasets/jillanisofttech/flight-price-prediction-dataset>