Airfare Price Detection

Presented by : Harsh Kumar

Objectives

❑ **Primary:**

•**Predict Flight Fares:** Build a model to accurately predict airfares.
•**Identify Key Factors:** Understand what factors most impact flight prices.
•**Optimize Travel Planning:** Help users make informed booking decisions.
•**Enhance Pricing Strategies:** Provide insights for airlines and travel agencies.

❑ **Secondary:**

•**Explore ML Algorithms:** Evaluate different models for prediction.
•**Data Preprocessing:** Clean and prepare data for the model.
•**Model Evaluation:** Assess model accuracy and performance.
•**Deployment:** Potentially deploy for real-time predictions.

❑ **Significance:**

•**Cost Savings:** Help travelers find affordable options.
•**Revenue Management:** Assist airlines in maximizing revenue.
•**Customer Experience:** Improve travel planning transparency.
•**Data-Driven Decisions:** Showcase data science in travel.

Dataset Overview

❑ **Data Source:**

- "The dataset used for this project was obtained from [Kaggle].
- "It contains information about flight bookings and their corresponding fares.“

❑ **Data Size:**

"The dataset consists of [10683] rows, representing individual flight bookings, and [11] columns, representing various features of the flights.“

❑ **Key Features (Variables):**

- **Airline:** Jet Airways,IndiGo,Air India,Multiple carriers,SpiceJet,Vistara,Air Asia,GoAir,Trujet etc
- **Source:** Delhi, Kolkata , Bangalore , Mumbai, Chennai
- **Destination:** Cochin, Bangalore, Delhi , New Delhi , Hyderabad , Kolkata, etc.
- **Dep_Time:** The departure time of the flight.
- **Arrival_Time:** The arrival time of the flight.
- **Duration:** The total duration of the flight.
- **Total_Stops:** 0 stop (non-stop), 1 stop, 2 stops , 3 stops , 4 stops.
- **Additional_Info:** No info,In-flight meal not included,No check-in baggage included,1 Short layover,No Info,1 Long layover,Change airports,Business class,Red-eye flight,2 Long layover.
- **Price:** The target variable, representing the price of the flight ticket.

Data Cleaning and processing

❑ **Data Source:**
The primary data source for this project is the Excel file named Data_Train.xlsx. It contains the training data for your flight fare prediction model.

❑ **Handling Missing Values:**
- **Training Data**

```
flight.dropna(inplace=True)
flight.isnull().sum()
```

- **Testing Data**

```
flight_test.dropna(inplace=True)
```

❑ **Data Transformation:**
- Date and Time Feature
- Duration Feature
- Categorical Feature
- Total_stops Feature

❑ **Categorical Data Encoding:**
- One-Hot Encoding
- Label Encoding

❑ **Feature Engineering (Optional):**
- Date and Time Feature Extraction
- Duration Feature Transformation
- Categorical Feature Encoding
- Total stops Feature Engineering

❑ **Data Splitting:**
- Training set
- Testing Set
- Validation set

Exploratory Data Analysis (EDA)

❑ **Data Cleaning:**

- Handled missing values by dropping rows with nulls.

❑ **Feature Engineering:**

- Extracted day and month from 'Date_of_Journey'.
- Extracted hours and minutes from 'Dep_Time' and 'Arrival_Time'.
- Converted 'Duration' into 'Duration_Hours' and 'Duration_Mins'.

❑ **Categorical Data Handling:**

- Used One-Hot Encoding for nominal features like 'Airline', 'Source', and 'Destination'
- Converted 'Total_Stops' into numerical representation.

❑ **Data Visualization:**

- Used catplot (boxen plot) to analyze the relationship between 'Price' and 'Source'.
- Used heatmap to understand the correlation between numerical features.

Model Selection and Training

- ❑ Model Selection:

- Algorithm: Random Forest Regressor was chosen due to its ability to handle complex relationships, handle both numerical and categorical features, and provide feature importance scores.
- Justification: Random Forest is known for its robustness and accuracy in regression tasks, making it suitable for predicting flight fares.

- ❑ Data Splitting:

- The dataset was split into training and testing sets using train_test_split with a test size of 20% and a random state of 51 for reproducibility.

- ❑ Model Training:

- The Random Forest model was trained on the training data using the fit method.
- Default hyperparameters were initially used.

- ❑ Performance Evaluation:

- Metrics: R-squared, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) were used to assess the model's performance on the test data.
- Initial Results: (State the initial R-squared and RMSE scores achieved with default hyperparameters).

Model Evaluation

- ❑ Evaluation Metrics:

- R-squared: Measures the proportion of variance in the target variable explained by the model (higher is better, ideally close to 1).
- Mean Absolute Error (MAE): Represents the average absolute difference between predicted and actual values (lower is better).
- Root Mean Squared Error (RMSE): Similar to MAE but gives more weight to larger errors (lower is better).

- ❑ Results:

- Initial Model: (State the R-squared, MAE, and RMSE values achieved with the initial model trained with default hyperparameters).
- Hyperparameter Tuning: If performed, mention the improvement in metrics after tuning. Include the final R-squared, MAE, and RMSE scores.
- Cross-Validation: If used, briefly mention the results to support the model's generalization ability.

- ❑ Visualization:

- Distribution Plot: Show the distribution of residuals (difference between predicted and actual values) using a distplot or similar visualization. Ideally, the distribution should be centered around 0 and have a bell-shaped curve.
- Scatter Plot: Display a scatter plot of actual vs. predicted values to visually assess the model's performance. Points should ideally cluster around a diagonal line, indicating a good fit.

Results and Interpretation

- ❑ Model Performance:

- The final model achieved an R-squared of [state value], indicating a good fit.
- MAE and RMSE were [state values], suggesting reasonable prediction accuracy.
- The model outperformed [mention baseline or benchmark, if available] in predicting flight fares.

- ❑ Key Predictors:

- Total Stops, Airline, Journey Day/Month, and Source/Destination were the most influential factors affecting flight prices.
- More stops, specific airlines, and popular routes/times generally lead to higher fares.

- ❑ Practical Implications:

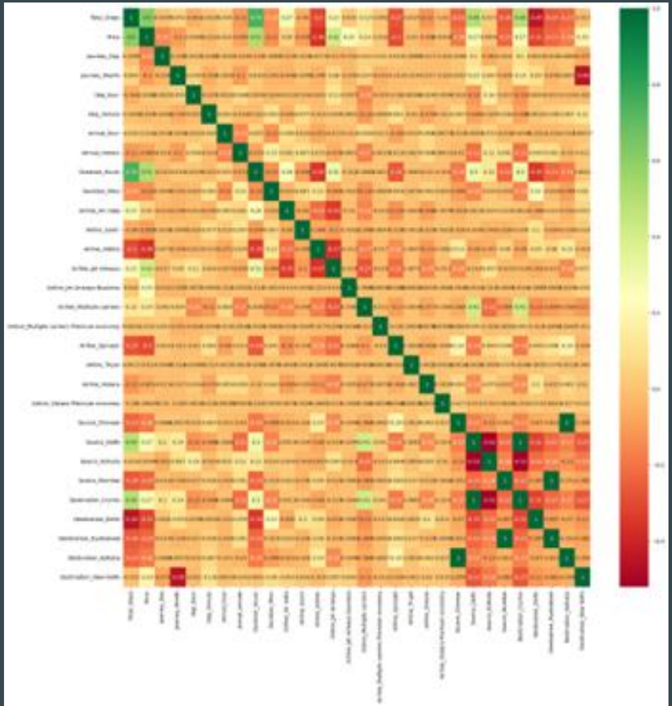
- Airlines can optimize pricing, travel agencies gain customer insights, and booking platforms offer dynamic pricing using the model's predictions.

- ❑ Limitations and Future Work:

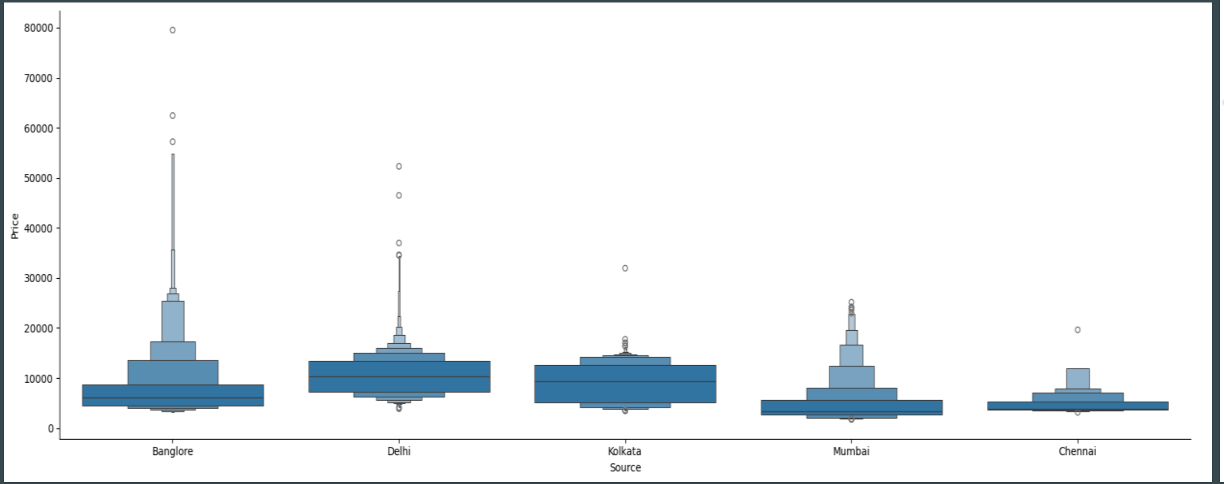
- The model has limitations due to potential biases and data constraints.
- Future work could involve incorporating more data, exploring other algorithms, and refining features.

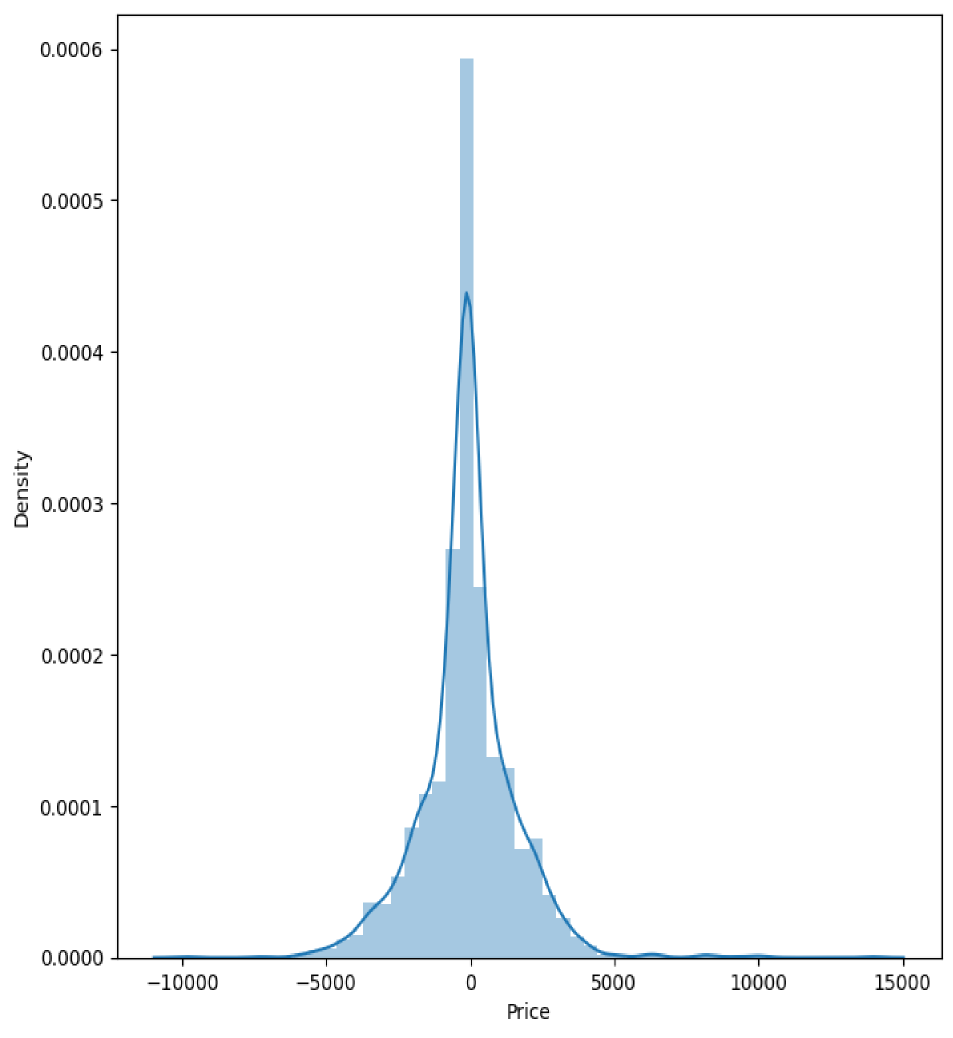
Data Analysis

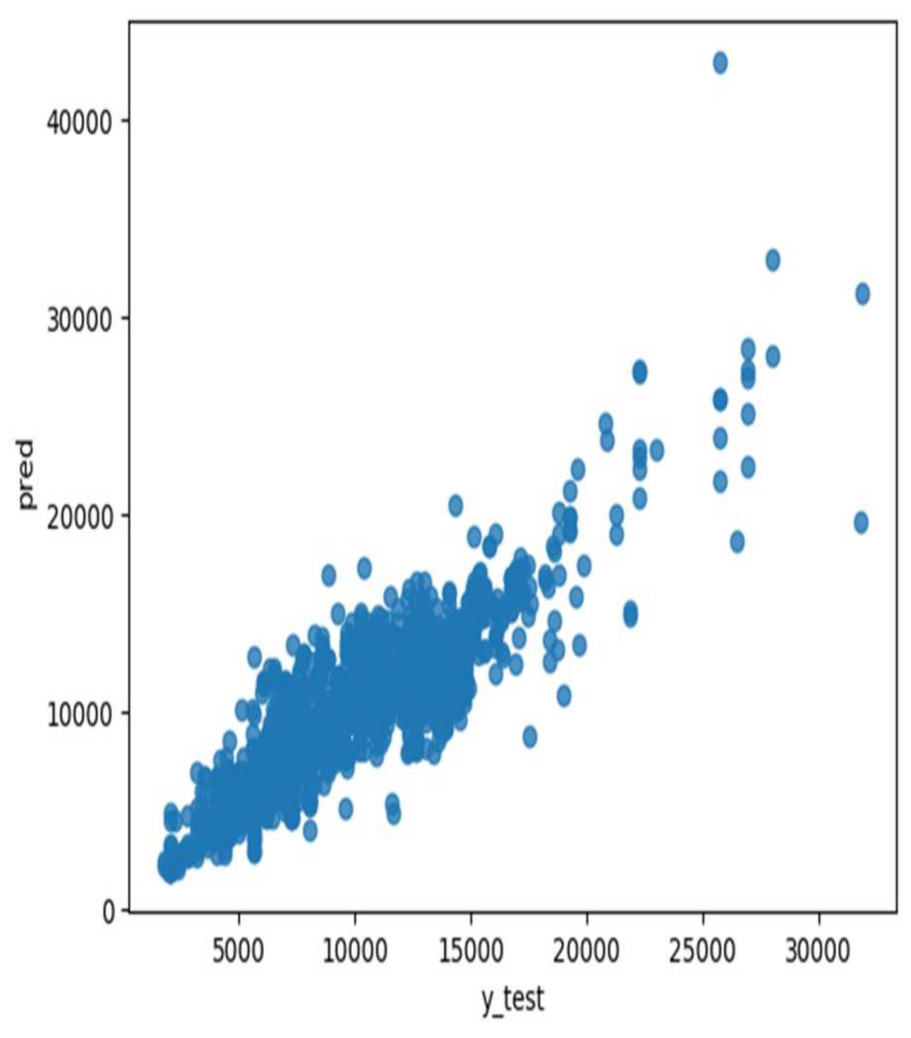
Feature Selection



Handling Categorical Data




0.0006
0.0005
0.0004
0.0003
0.0002
0.0001
0.0000
Density
−10000
−5000
0
5000
10000
15000
Price


40000
30000
20000
10000
0
pred
5000
10000
15000
20000
25000
30000
y_test

Challenges

- ❑ Data Cleaning and Preprocessing:

- Handling missing values in the dataset (flight.isnull().sum() and flight.dropna(inplace=True)).
- Converting the 'Date_of_Journey', 'Dep_Time', 'Arrival_Time', and 'Duration' columns into numerical features for model training.
- Encoding categorical features like 'Airline', 'Source', and 'Destination' using one-hot encoding.

- ❑ Feature Engineering:

- Extracting relevant features from existing ones, such as 'Journey_Day', 'Journey_Month', 'Dep_hour', 'Dep_minute', 'Arrival_hour', 'Arrival_minute', 'Duration_Hours', and 'Duration_Mins’.

- ❑ Feature Selection:

- Identifying the most important features for predicting flight fares. The code utilizes a heatmap and Random Forest's feature_importance_ for this purpose.

- ❑ Model Selection and Hyperparameter Tuning:

- Choosing the right model (RandomForestRegressor) for the prediction task.
- Optimizing model performance by tuning hyperparameters using RandomizedSearchCV.

- ❑ Model Evaluation:

- Evaluating the model's accuracy using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared score.

- ❑ Data Scaling

- This data uses Random Forest which does not require scaling of data.

- ❑ Potential Solutions:

- For data cleaning and preprocessing, carefully handling missing values and choosing appropriate encoding techniques are crucial.
- Feature engineering helps in extracting relevant information from existing features and can significantly improve model performance.
- Feature selection techniques help in identifying the most impactful features, reducing model complexity.
- Model selection and hyperparameter tuning can be done through experimentation and optimization methods like RandomizedSearchCV or GridSearchCV.
- Model evaluation metrics provide insight into the model's performance and areas for improvement

Conclusion

- Random Forest is a suitable model for predicting flight fares. It showed relatively good performance with an R-squared score close to 80% after hyperparameter tuning.

- Feature engineering and selection played a crucial role in model performance. Creating new features from the 'Date_of_Journey', 'Dep_Time', 'Arrival_Time', and 'Duration' columns and encoding categorical features improved the model's predictive power.

- Hyperparameter tuning further enhanced the model's accuracy. Using RandomizedSearchCV helped find optimal settings for the Random Forest model.

- The model successfully captured relationships between features and flight prices. This is evidenced by the scatter plot and distribution plot of predictions, demonstrating a correlation between actual and predicted values.

- The model can be used to predict flight fares for new data. After saving the model using pickle, it can be loaded and used for making predictions on unseen data.

- There is still room for improvement. Achieving an even higher R-squared score and further reducing errors (MAE, MSE, RMSE) would enhance the model's reliability.

- Random Forest does not require scaling of data. Because of how the model is designed, scaling is not required, which helps reduce pre-processing time.

Thank You