OPTIMISING FLIGHT BOOKING DESICIONS THROUGH MACHINE LEARNING PRICE PREDICTIONS:

1)INTRODUCTION:

1.1)OVERVIEW:

Flight booking price prediction is the process of forecasting the cost of airline tickets for future travel. This technology is valuable for both travelers and airlines, as it helps travelers make informed decisions and airlines optimize their pricing strategies. Here's an overview of flight booking price prediction

1.2)PURPOSE:

The purpose of flight booking price prediction serves several key objectives for both travelers and airlines:

**1. Informed Decision-Making for Travelers:**

* **Cost Optimization:** Travelers can plan their trips more efficiently by knowing when and where to book flights at the most favorable prices.
* **Budget Planning:** It helps travelers budget for their trips by providing estimates of flight costs in advance.

**2. Enhanced Booking Experience:**

* **Recommendations:** Airlines and travel agencies can offer personalized recommendations to travelers based on their historical preferences and current market conditions.
* **Transparency:** Transparent pricing information builds trust with customers and improves their booking experience.

**3. Revenue Optimization for Airlines:**

* **Dynamic Pricing:** Airlines can adjust ticket prices in real-time based on demand, optimizing revenue by filling seats at the highest possible prices.
* **Forecasting Demand:** Accurate predictions allow airlines to anticipate demand trends and allocate resources more effectively.

**4. Competitive Advantage:**

* **Market Insights:** Airlines can gain insights into their competitors' pricing strategies and adjust their own pricing accordingly.
* **Customer Loyalty:** Providing competitive prices can help airlines attract and retain customers.

**5. Inventory Management:**

* **Seat Allocation:** Predictive models assist airlines in allocating seats across different booking classes, ensuring a balanced distribution of passengers.
* **Overbooking Prevention:** Accurate demand forecasts reduce the likelihood of overbooking flights.

**6. Seasonal and Economic Trends:**

* **Adaptation:** Airlines can adapt to seasonal variations and economic factors by adjusting pricing strategies proactively.

**7. Risk Mitigation:**

* **Hedging:** Airlines can use price prediction models to manage fuel and currency risk by making financial decisions based on anticipated revenue.

2)LITERATURE SURVEY:

2.1)EXISTING APPROACHES: booking price prediction is a complex problem that has attracted significant research and development efforts. Several approaches and techniques have been employed to predict flight ticket prices accurately. Here are some of the existing approaches for flight booking prediction:

1. **Historical Pricing Analysis:**
   * Analyzing historical ticket price data to identify trends, seasonality, and patterns.
   * Basic statistical methods like moving averages and exponential smoothing can be applied to historical data.
2. **Regression Models:**
   * Linear Regression: Using linear regression models to establish relationships between ticket prices and factors such as departure date, destination, booking class, and lead time.
   * Polynomial Regression: Employing higher-degree polynomial regression to capture nonlinear relationships.
3. **Time Series Forecasting:**
   * ARIMA (AutoRegressive Integrated Moving Average): A popular time series forecasting method to model and predict price fluctuations over time.
   * Seasonal decomposition and exponential smoothing techniques can also be applied.
4. **Machine Learning Algorithms:**
   * Decision Trees: Decision trees can be used to model price prediction by splitting data into branches based on relevant features.
   * Random Forests: Ensemble techniques like random forests can improve prediction accuracy by aggregating the results of multiple decision trees.
   * Gradient Boosting: Algorithms like XGBoost and LightGBM have been successful in flight price prediction tasks.
   * Neural Networks: Deep learning models, including recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), can capture complex patterns in the data.
5. **Time-Windowed Approaches:**
   * Rolling Forecast: Continuously updating the model with new data as it becomes available to improve prediction accuracy.
6. **Feature Engineering:**
   * Creating and selecting relevant features, such as day of the week, time of the day, holidays, and special events, to enhance model performance.
   * Incorporating external data sources like weather, economic indicators, and airline-specific factors.
7. **Market Basket Analysis:**
   * Analyzing the co-occurrence of flight routes, destinations, or classes to provide recommendations and bundle deals to customers.
8. **Demand Forecasting:**
   * Combining flight booking price prediction with demand forecasting to optimize pricing strategies and capacity allocation.
9. **Dynamic Pricing Strategies:**
   * Implementing dynamic pricing algorithms that adjust ticket prices in real-time based on demand and inventory levels.
10. **Reinforcement Learning:**
    * Applying reinforcement learning techniques to optimize pricing decisions over time, learning from past interactions.
11. **Data Mining and Big Data Analytics:**
    * Leveraging big data techniques to process vast amounts of historical data and extract meaningful insights for prediction.
12. **Hybrid Approaches:**
    * Combining multiple prediction models or techniques to improve accuracy and robustness.

2.2)PROPOSED SOLUTION:

A proposed solution for flight booking price prediction involves combining various techniques and approaches to create an accurate and robust model. Here's a high-level outline of a potential solution:

**1. Data Collection and Preprocessing:**

* Gather historical flight booking data, including ticket prices, routes, departure and arrival airports, booking class, departure dates, and relevant external factors (e.g., weather, economic indicators).
* Clean and preprocess the data, handling missing values, outliers, and converting categorical variables into numerical form.
* Feature engineering: Create relevant features such as day of the week, time of the day, holidays, special events, and seasonality.

**2. Exploratory Data Analysis (EDA):**

* Conduct EDA to understand the data distribution, identify correlations, and uncover patterns or anomalies.
* Visualize key trends, such as price fluctuations over time, seasonal variations, and differences across routes and airlines.

**3. Model Selection:**

* Experiment with various machine learning and statistical models to find the best-suited approach for flight price prediction.
* Consider using regression models (linear, polynomial), time series forecasting (ARIMA, exponential smoothing), machine learning algorithms (random forests, gradient boosting), and neural networks (RNNs, LSTMs).

**4. Model Training:**

* Split the dataset into training and validation sets to train and evaluate the model's performance.
* Implement cross-validation techniques to assess the model's robustness and avoid overfitting.
* Tune hyperparameters to optimize the model's accuracy.

**5. Feature Importance Analysis:**

* Analyze feature importance to understand which factors have the most significant impact on flight prices.
* Remove less relevant features if necessary to simplify the model.

**6. Time Windowed Approach:**

* Implement a rolling forecast mechanism where the model continuously updates using new data as it becomes available.
* Periodically retrain the model to adapt to changing market conditions.

**7. External Data Integration:**

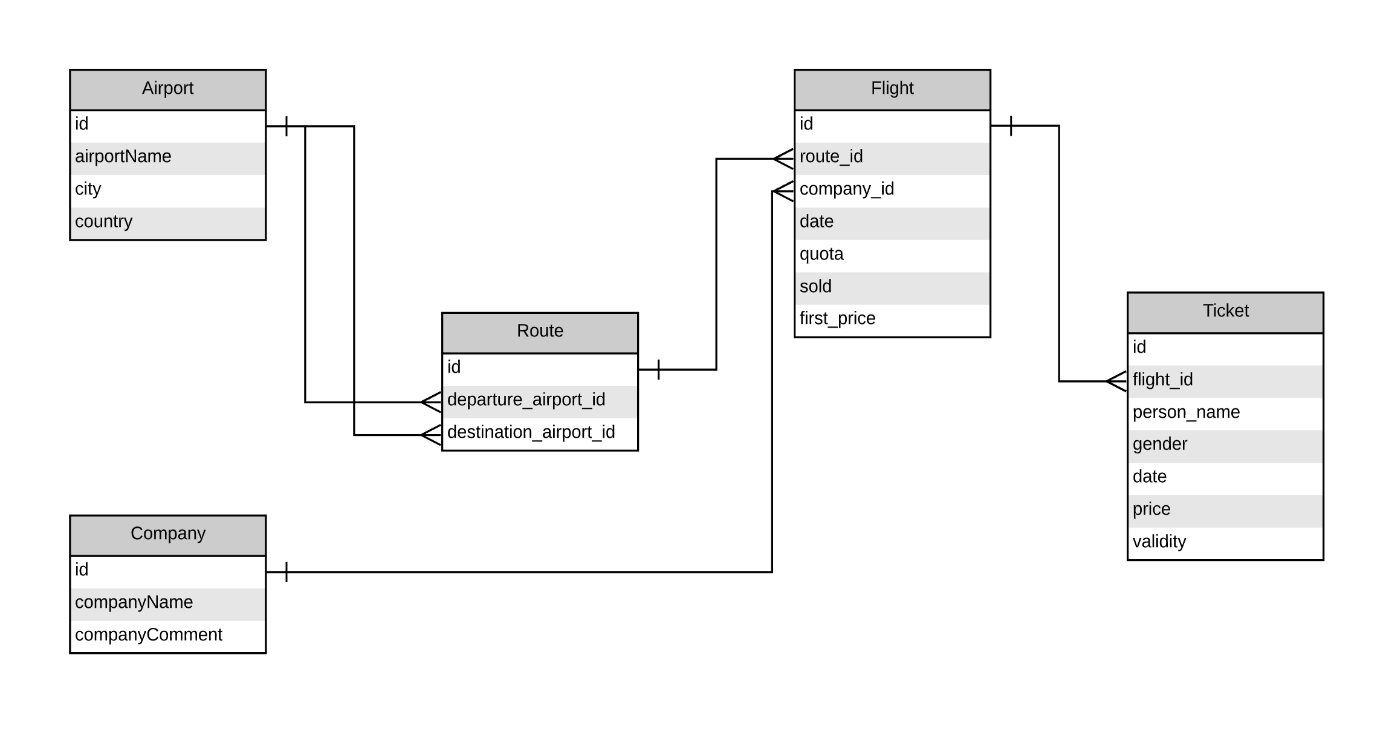
* Incorporate external data sources such as weather forecasts, economic indicators, and airline-specific factors to improve prediction accuracy.

**8. Dynamic Pricing Integration:**

* If applicable, integrate dynamic pricing algorithms that adjust ticket prices in real-time based on demand and inventory level

3)THEORETICAL ANALYSIS:

3.1)BLOCK DIAGRAM:



3.2)HARDWARE/SOFTWARE DESIGNING:

**1. Define Requirements:**

* Understand the specific goals and requirements of your flight booking prediction system. This may include predicting flight ticket prices, availability, or demand.

**2. Data Collection:**

* Gather historical flight booking data, including ticket prices, booking times, seat availability, and other relevant information. This data will serve as the foundation for your prediction models.

**3. Data Preprocessing:**

* Clean and preprocess the collected data. This includes handling missing values, data normalization, and feature engineering to create meaningful input features for your prediction models.

**4. Hardware Infrastructure:**

* Determine the hardware infrastructure you'll need to handle the data processing and prediction tasks. This typically includes servers or cloud resources with sufficient computational power and storage capacity.

**5. Software Development:**

* Develop the software components of your flight booking prediction system.
  + **Machine Learning Models:** Build predictive models using techniques like regression, time series forecasting, or machine learning algorithms. You can use libraries like Scikit-Learn, TensorFlow, or PyTorch for model development.
  + **Feature Engineering:** Create features that capture relevant information, such as historical booking patterns, flight routes, and seasonal trends.
  + **API Integration:** Integrate with external APIs provided by airlines or travel agencies to fetch real-time data on flight availability and prices.
  + **Database:** Set up a database to store and retrieve historical and real-time data efficiently.

**6. Model Training and Evaluation:**

* Train your prediction models on the preprocessed data. Use appropriate evaluation metrics to assess model performance, such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE).

**7. Real-Time Predictions:**

* Implement a real-time prediction mechanism that takes user inputs (e.g., departure date, destination, number of passengers) and provides flight booking predictions based on the trained models.

**8. User Interface:**

* Develop a user-friendly interface (web or mobile app) that allows users to interact with your prediction system. This interface should take user inputs, make predictions, and display results in an understandable format.

**9. Deployment:**

* Deploy your system on the chosen hardware infrastructure, ensuring it can handle concurrent user requests and maintain responsiveness.

**10. Monitoring and Maintenance:**

* Continuously monitor the system's performance, data quality, and model accuracy. Implement regular updates and improvements based on new data and user feedback.

**11. Scalability:**

* Design the system to be scalable so that it can handle increased demand as your user base grows.

**12. Security and Privacy:**

* Implement security measures to protect user data and ensure the privacy of sensitive information.

**13. Compliance:**

* Ensure compliance with relevant regulations, such as GDPR or data protection laws, depending on your target audience and geographic reach.

**14. Feedback Loop:**

* Collect user feedback and usage data to iteratively improve your flight booking prediction system over time.

ABOUT THE DATA SET:

1. **Airline:** So this column will have all the types of airlines like Indigo, Jet Airways, Air India, and many more.
2. **Date\_of\_Journey:** This column will let us know about the date on which the passenger’s journey will start.
3. **Source:** This column holds the name of the place from where the passenger’s journey will start.
4. **Destination:** This column holds the name of the place to where passengers wanted to travel.
5. **Route:** Here we can know about that what is the route through which passengers have opted to travel from his/her source to their destination.
6. **Arrival\_Time:** Arrival time is when the passenger will reach his/her destinationP.
7. **Duration:**Duration is the whole period that a flight will take to complete its journey from source to destination.
8. **Total\_Stops:** This will let us know in how many places flights will stop there for the flight in the whole journey.
9. **Additional\_Info:** In this column, we will get information about food, kind of food, and other amenities.
10. **Price:** Price of the flight for a complete journey including all the expenses before onboarding.

PROJECT FLOW:

1. **EDA:** Learn the complete process of EDA
2. **Data analysis:** Learn to withdraw some insights from the dataset both mathematically and visualize it.
3. **Data visualization:** Visualising the data to get better insight from it.
4. **Feature engineering:** We will also see what kind of stuff we can do in the feature engineering part.

## **1.EDA:**

train\_df.columns

**Output:**

Index(['Airline', 'Date\_of\_Journey', 'Source', 'Destination', 'Route',

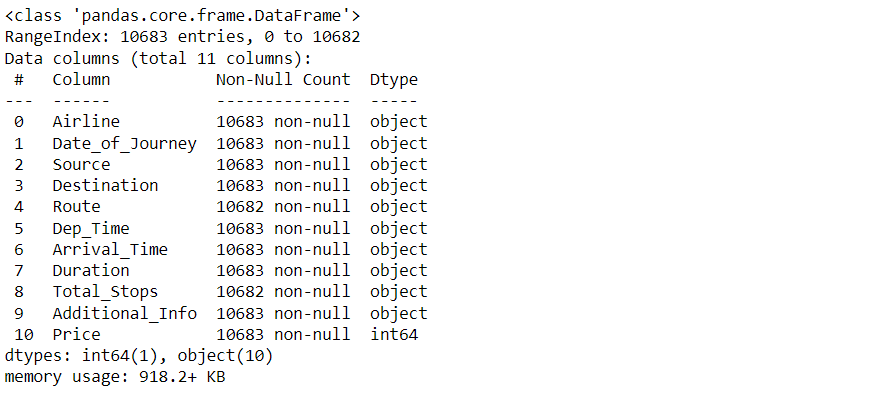
'Dep\_Time', 'Arrival\_Time', 'Duration', 'Total\_Stops',

'Additional\_Info', 'Price'],

dtype='object')

train\_df.info()

**Output:**



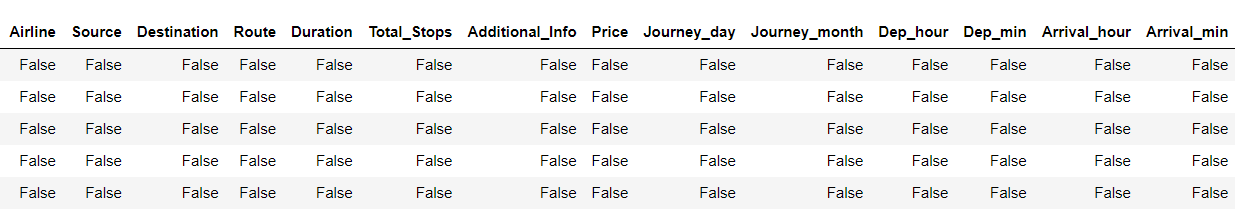
train\_df.describe()

**Output:**



train\_df.isnull().head()

**Output:**



**2.Data analysis:**

train\_df.isnull().sum()

**Output:**

Airline 0

Date\_of\_Journey 0

Source 0

Destination 0

Route 1

Dep\_Time 0

Arrival\_Time 0

Duration 0

Total\_Stops 1

Additional\_Info 0

Price 0

dtype: int64

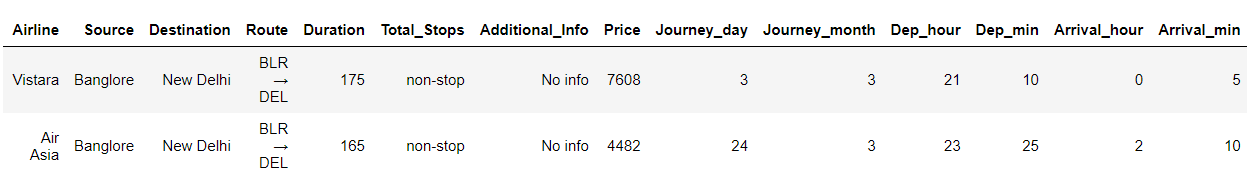
**Dropping NAN values**

train\_df.dropna(inplace = True)

**Duplicate values**

train\_df[train\_df.duplicated()].head()

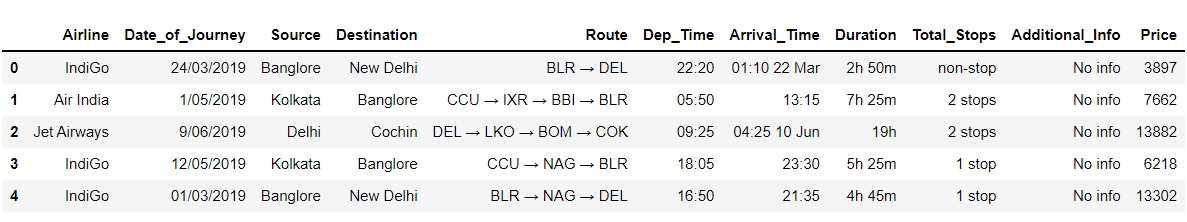
**Output:**



train\_df.drop\_duplicates(keep='first',inplace=True)

train\_df.head()

**Output:**



train\_df.shape

**Output:**

(10462, 11)

train\_df["Additional\_Info"].value\_counts()

**Output:**

No info 8182

In-flight meal not included 1926

No check-in baggage included 318

1 Long layover 19

Change airports 7

Business class 4

No Info 3

1 Short layover 1

2 Long layover 1

Red-eye flight 1

Name: Additional\_Info, dtype: int64

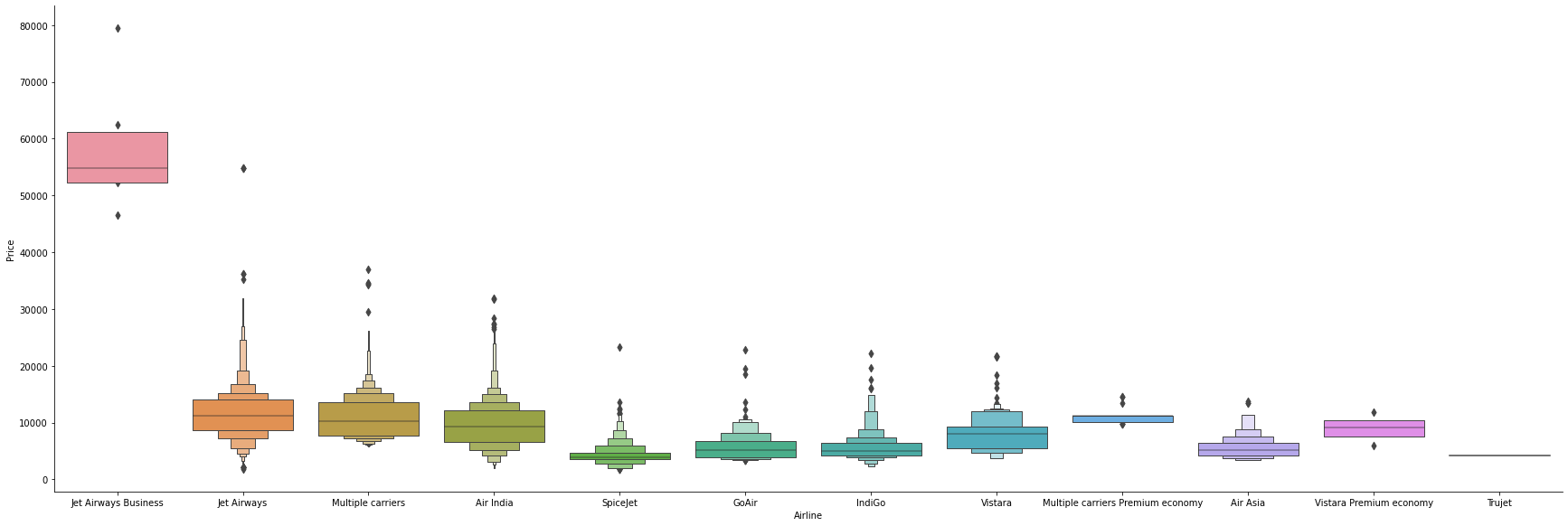
3.Data Visulization:

**Plotting Price vs Airline plot**

sns.catplot(y = "Price", x = "Airline", data = train\_df.sort\_values("Price", ascending = False), kind="boxen", height = 8, aspect = 3)

plt.show()

**Output:**



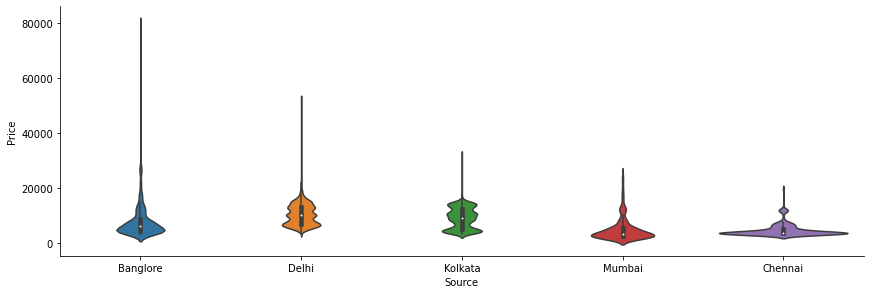
.

**Plotting Violin plot for Price vs Source**

sns.catplot(y = "Price", x = "Source", data = train\_df.sort\_values("Price", ascending = False), kind="violin", height = 4, aspect = 3)

plt.show()

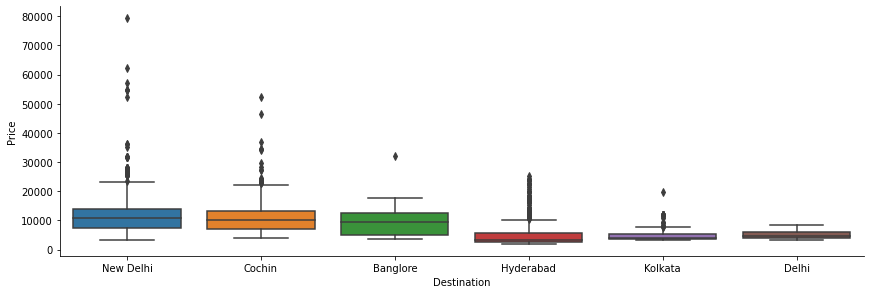
**Output:**



sns.catplot(y = "Price", x = "Destination", data = train\_df.sort\_values("Price", ascending = False), kind="box", height = 4, aspect = 3)

plt.show()

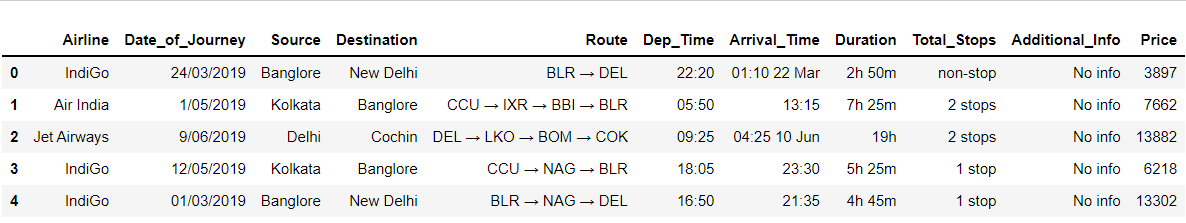
**Output:**



4.Feature Engineering:

train\_df.head()

**Output:**



**Here first we are dividing the features and labels and then converting the hours in minutes.**

train\_df['Duration'] = train\_df['Duration'].str.replace("h", '\*60').str.replace(' ','+').str.replace('m','\*1').apply(eval)

test\_df['Duration'] = test\_df['Duration'].str.replace("h", '\*60').str.replace(' ','+').str.replace('m','\*1').apply(eval)

**Date\_of\_Journey:** Here we are organizing the format of the date of journey in our dataset for better preprocessing in the model stage.

train\_df["Journey\_day"] = train\_df['Date\_of\_Journey'].str.split('/').str[0].astype(int)

train\_df["Journey\_month"] = train\_df['Date\_of\_Journey'].str.split('/').str[1].astype(int)

train\_df.drop(["Date\_of\_Journey"], axis = 1, inplace = True)

**Dep\_Time:** Here we are converting departure time into hours and minutes

train\_df["Dep\_hour"] = pd.to\_datetime(train\_df["Dep\_Time"]).dt.hour

train\_df["Dep\_min"] = pd.to\_datetime(train\_df["Dep\_Time"]).dt.minute

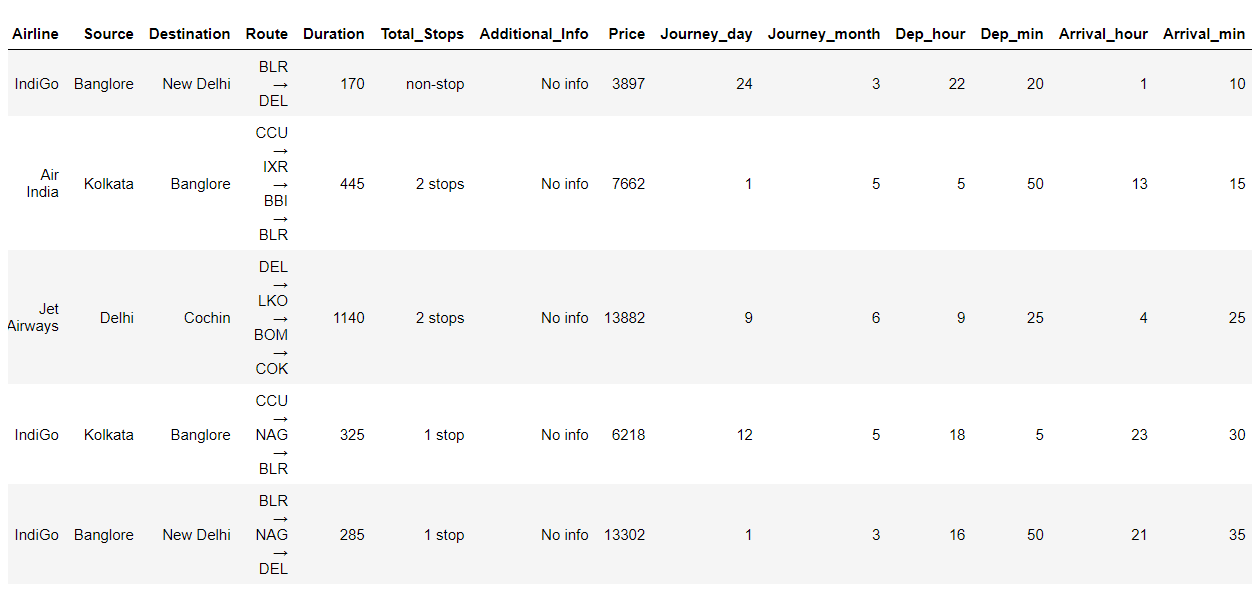
train\_df.drop(["Dep\_Time"], axis = 1, inplace = True)

**Arrival\_Time:**Similarly we are converting the arrival time into hours and minutes.

**Now after final preprocessing let’s see our dataset**

train\_df.head()

**Output:**



**Plotting Bar chart for Months (Duration) vs Number of Flights**

plt.figure(figsize = (10, 5))

plt.title('Count of flights month wise')

ax=sns.countplot(x = 'Journey\_month', data = train\_df)

plt.xlabel('Month')

plt.ylabel('Count of flights')

for p in ax.patches:

ax.annotate(int(p.get\_height()), (p.get\_x()+0.25, p.get\_height()+1), va='bottom', color= 'black')

**Output:**

**Plotting Bar chart for Types of Airline vs Number of Flights**

plt.figure(figsize = (20,5))

plt.title('Count of flights with different Airlines')

ax=sns.countplot(x = 'Airline', data =train\_df)

plt.xlabel('Airline')

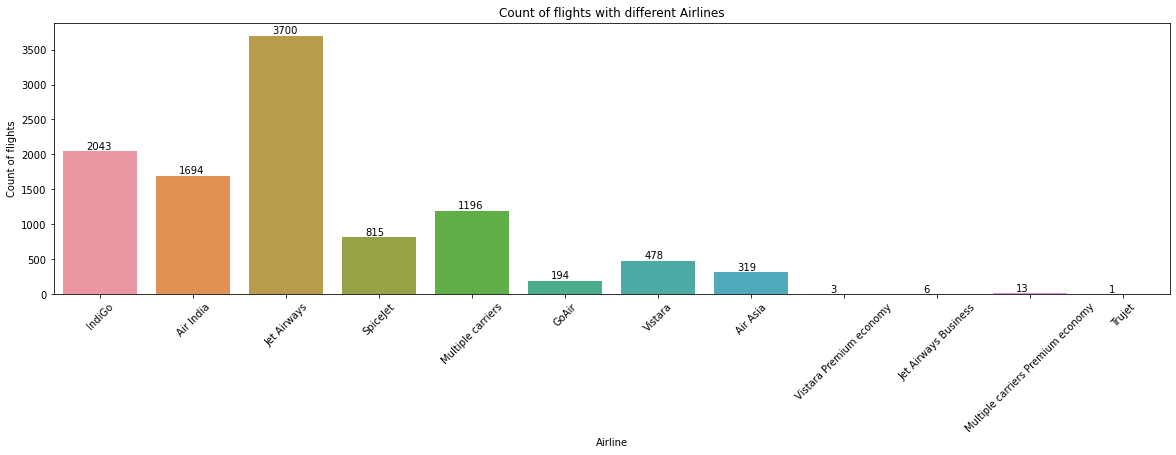
plt.ylabel('Count of flights')

plt.xticks(rotation = 45)

for p in ax.patches:

ax.annotate(int(p.get\_height()), (p.get\_x()+0.25, p.get\_height()+1), va='bottom', color= 'black')

**Output:**



**Plotting Ticket Prices VS Airlines**

plt.figure(figsize = (15,4))

plt.title('Price VS Airlines')

plt.scatter(train\_df['Airline'], train\_df['Price'])

plt.xticks

plt.xlabel('Airline')

plt.ylabel('Price of ticket')

plt.xticks(rotation = 90)

**Output:**

## **Correlation between all Features**

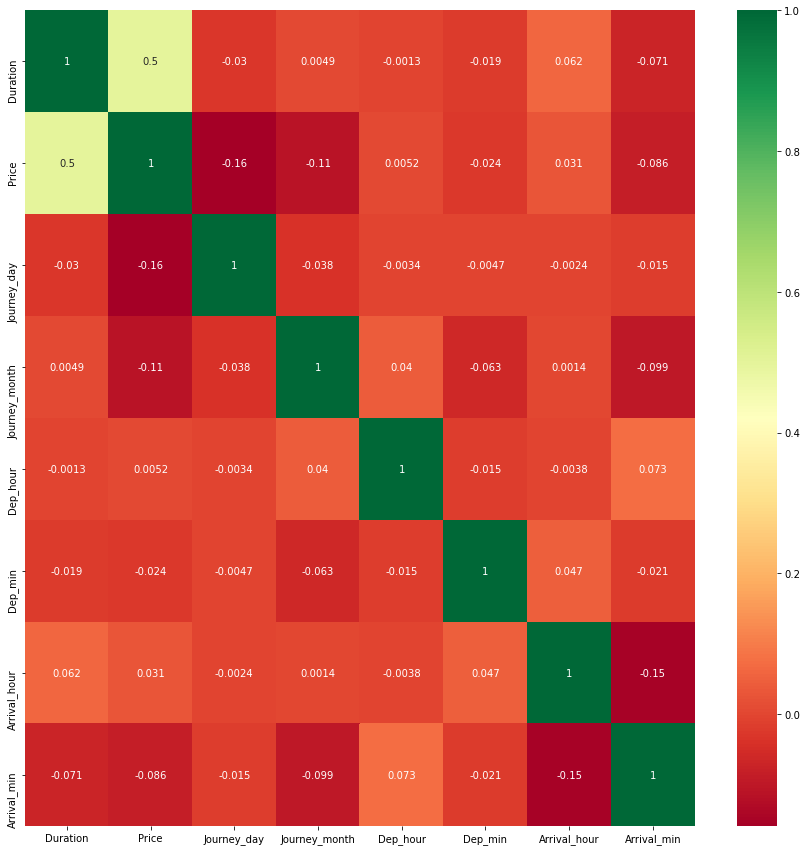
**Plotting Correlation**

plt.figure(figsize = (15,15))

sns.heatmap(train\_df.corr(), annot = True, cmap = "RdYlGn")

plt.show()

**Output:**



**Dropping the Price column as it is of no use**

data = train\_df.drop(["Price"], axis=1)

**Dealing with Categorical Data and Numerical Data**

train\_categorical\_data = data.select\_dtypes(exclude=['int64', 'float','int32'])

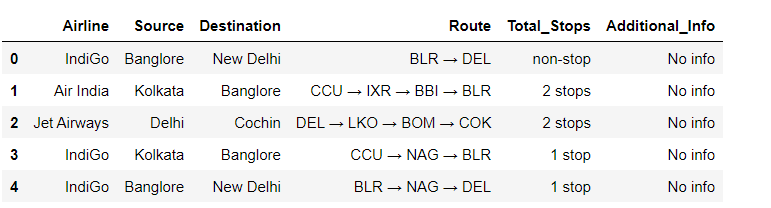
train\_numerical\_data = data.select\_dtypes(include=['int64', 'float','int32'])

test\_categorical\_data = test\_df.select\_dtypes(exclude=['int64', 'float','int32','int32'])

test\_numerical\_data = test\_df.select\_dtypes(include=['int64', 'float','int32'])

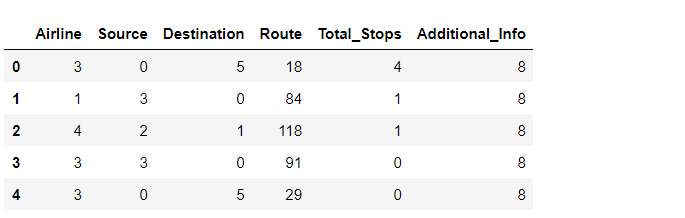
train\_categorical\_data.head()

**Output:**



**Label Encode and Hot Encode for Categorical Columns**

**Output:**



**Concatenating both Categorical Data and Numerical Data**

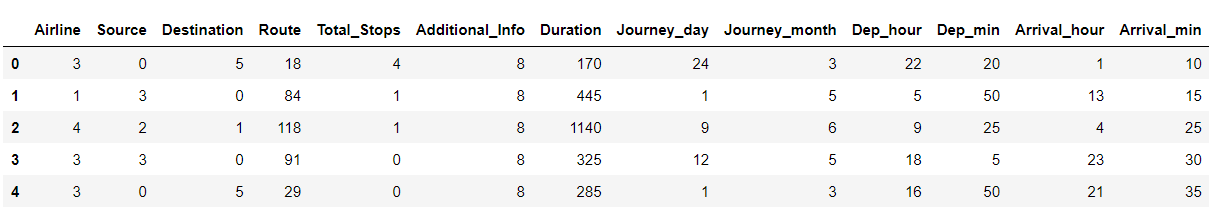
X = pd.concat([train\_categorical\_data, train\_numerical\_data], axis=1)

y = train\_df['Price']

test\_set = pd.concat([test\_categorical\_data, test\_numerical\_data], axis=1)

X.head()

**Output:**



y.head()

**Output:**

0 3897

1 7662

2 13882

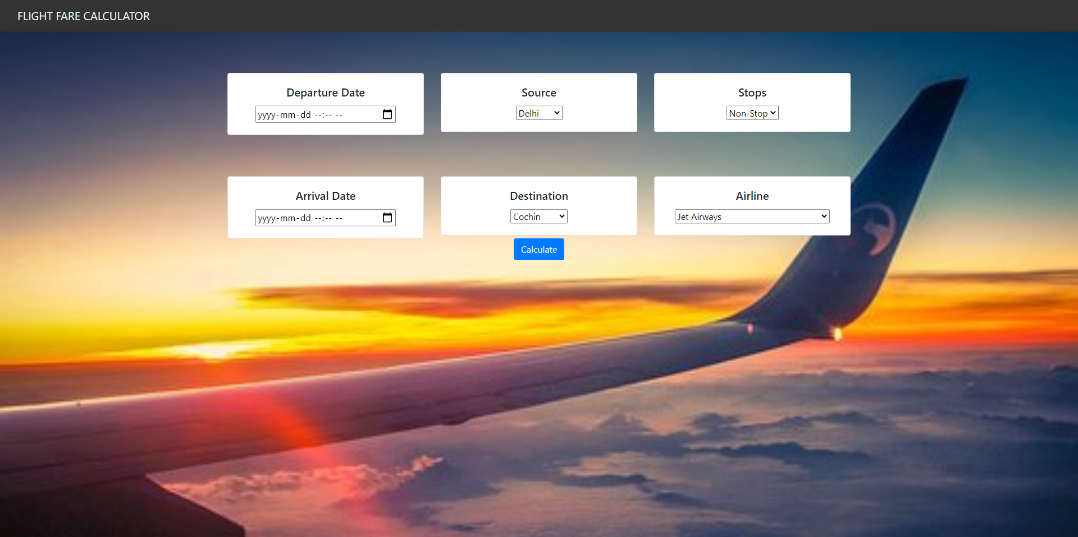
3 6218

4 13302

Name: Price, d

Data type: int64

4.RESULT:



5.ADVANTAGES:

1. **Cost Savings**: Predictive algorithms can help travelers find the best deals on flights by analyzing historical price trends and predicting when fares are likely to be the lowest. This can result in significant cost savings for travelers.
2. **Optimal Timing**: Predictive models can suggest the optimal timing for booking flights to get the best prices. This helps travelers avoid booking too early and overpaying or booking too late and missing out on good deals.
3. **Flexible Planning**: Travelers can make more flexible plans when they have reliable predictions about when and where flights will be available at the best prices. This can lead to more spontaneous travel and better travel experiences.
4. **Improved Airline Operations**: Airlines can use predictive analytics to optimize flight schedules, pricing strategies, and resource allocation. This can lead to better operational efficiency and cost savings for airlines.

DISADVANTAGES:

1. **Data Reliance**: Flight booking predictions heavily rely on historical data, and if the data is incomplete, inaccurate, or not up-to-date, it can lead to unreliable predictions.
2. **Unforeseen Events**: Predictive models may struggle to account for unforeseen events like natural disasters, political unrest, or pandemics, which can significantly impact flight availability and prices.

6.APPLICATIONS:

1. **Flight Search and Booking**: The primary function of flight booking applications is to allow users to search for available flights, compare prices, and book tickets for their desired travel dates and destinations.
2. **Price Comparison**: Flight booking apps often provide tools to compare prices across different airlines, travel dates, and booking classes. This helps travelers find the best deals and save money.
3. **Seat Selection**: Many flight booking apps allow travelers to choose their seats on the aircraft, including options for extra legroom, window seats, or aisle seats.
4. **Flight Alerts**: Users can set up flight alerts to receive notifications when ticket prices drop for specific routes or when preferred airlines have promotions.

7.CONCLUSION:

So as we saw that we have done a complete EDA process, getting

insights, feature engineering, and data visualization as well so after all these steps one can go for the prediction using machine learning model-making steps.

8.FUTURE SCOPE:

**Temporal Features:**

* + **Booking Date:** The date and time when the booking was made.
  + **Flight Date:** The departure date and time of the flight.
  + **Seasonality:** Consider factors like holidays, weekends, and special events.

**Geographical Features:**

* + **Origin and Destination:** The airports or cities involved in the flight.
  + **Route:** The specific flight route taken.
  + **Distance:** The distance between the origin and destination airports.
  + **Weather Conditions:** Current and forecasted weather at the departure and arrival locations.