**✈️ Airline Passenger Satisfaction Prediction Report**

**1 Introduction**

The purpose of this project is to analyze **airline passenger reviews and booking data** to understand satisfaction trends and predict whether a passenger is **Satisfied** or **Dissatisfied** based on their flight experience.

This analysis combines **text sentiment analysis** (using the VADER model from NLTK) with structured flight and traveler information, building a **machine learning model** capable of predicting satisfaction outcomes.

**2️ Data Overview**

**Datasets Used**

Four main datasets were utilized in this project:

1. **AirlineScrappedReview\_Cleaned.csv** — textual passenger reviews, ratings, traveler type, class, and route information. // This the one mainly used in the prediction process
2. **Customer\_comment.csv** — customer feedback and sentiment categories.
3. **Passanger\_booking\_data.csv** — booking behavior, trip type, and flight hour details.
4. **Survey data\_Inflight Satisfaction Score.csv** — inflight experience survey responses.

**Key Columns Used**

* **Traveller\_Type** – identifies the traveler (e.g., Couple Leisure, Business).
* **Class** – flight class (Economy, Business, or First Class).
* **Verified** – whether the review is verified by the airline (“Trip Verified”).
* Route – A detail about the flight
* **Sentiment\_Score** – numerical measure of emotional tone from text. Drived from review\_content
* And our target was the satisfaction (Derived from the Ratings column)

**3️ Preprocessing & Sentiment Analysis**

**3.1 Handling Missing Values**

The Route column initially had **766 missing values**.  
These were replaced with the **mode** (most frequent route), ensuring consistent data distribution while preventing data loss.

Missing Route values before: 766

Missing Route values after: 0

This approach preserves dataset integrity without biasing the model.

**3.2 Sentiment Scoring with VADER**

To convert unstructured review text into usable numerical data, **VADER (Valence Aware Dictionary for Sentiment Reasoning)** was applied.

Each passenger review was analyzed to produce a **compound sentiment score** (ranging from −1 to +1), representing emotional polarity, and categorized as follows:

* **Positive** → score > 0.05
* **Neutral** → −0.05 ≤ score ≤ 0.05
* **Negative** → score < −0.05

**Example Output:**

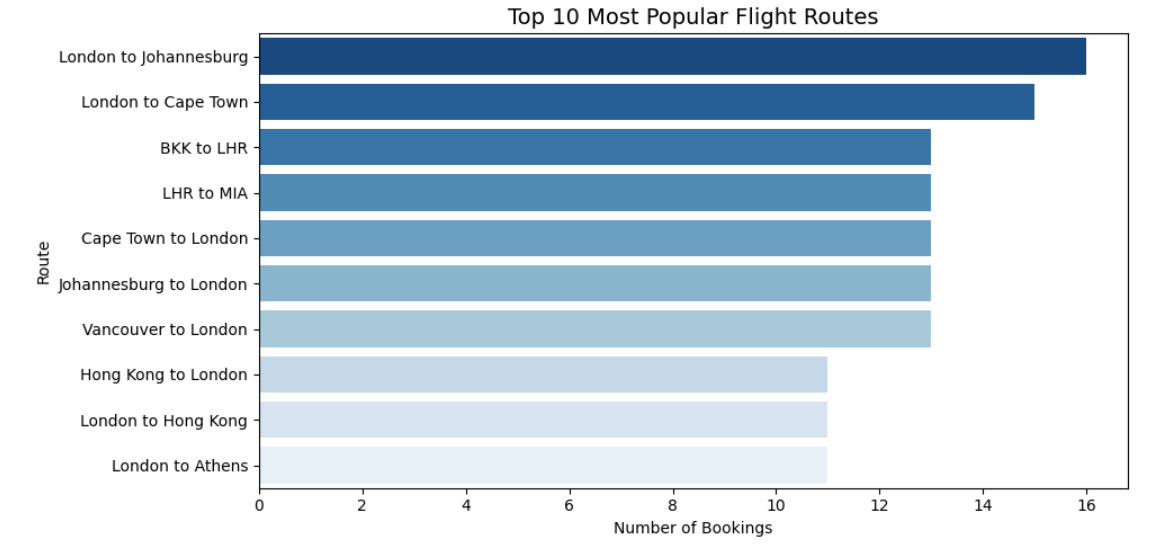
| **Review** | **Sentiment\_Score** | **Sentiment\_Label** |
| --- | --- | --- |
| “The airline lost my luggage...” | −0.799 | Negative |
| “First time flying with BA business class...” | 0.965 | Positive |

This process transformed free-text opinions into quantifiable variables suitable for machine learning.

**4️ Exploratory Data Analysis (EDA)**

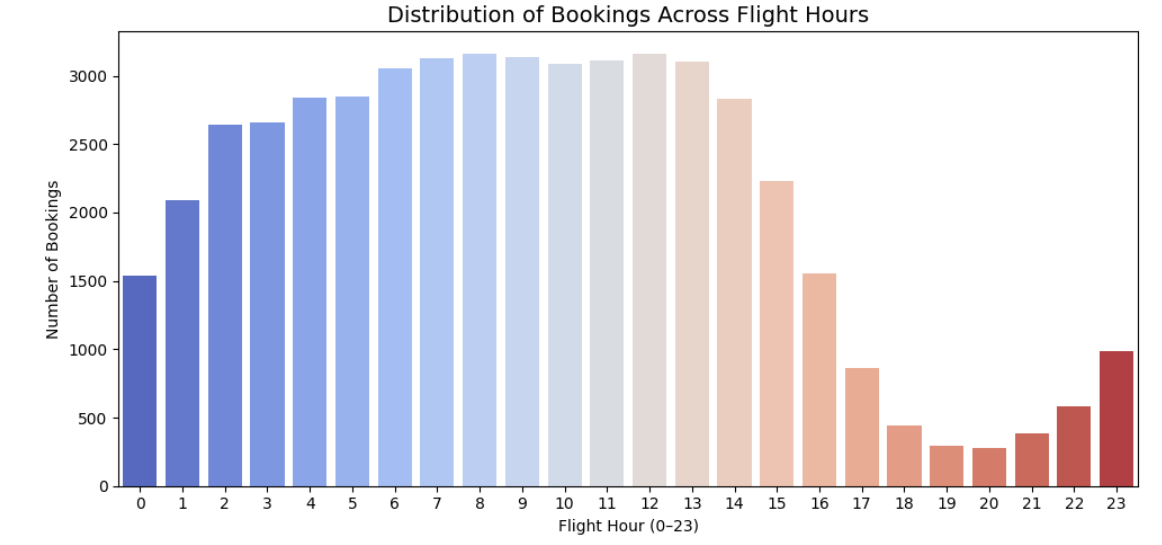
**4.1 Top 10 Most Popular Flight Routes**

A bar chart visualized the ten most frequent routes.  
**Interpretation:**  
High passenger volumes were observed between **London–Cape Town** and **London-Johannesburg**, highlighting key operational hubs and potential focal points for customer satisfaction monitoring.



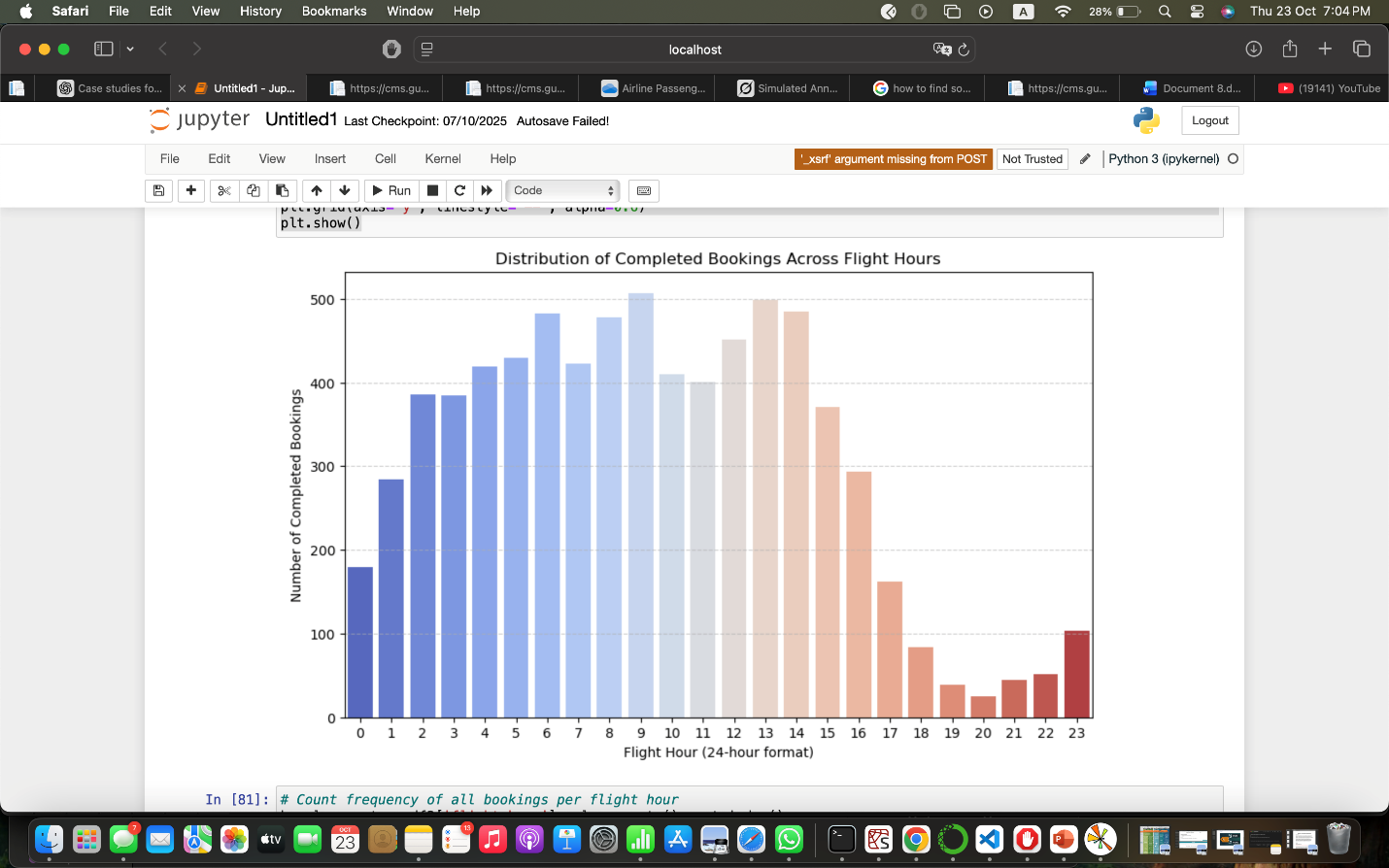
**4.2 Distribution of Bookings Across Flight Hours**

A histogram of flight booking times showed **peak activity between 06:00–09:00** and **low activity between** **18:00–21:00**.



This is a graph for all the completed Bookings

And the following graph is for the Completed bookings only

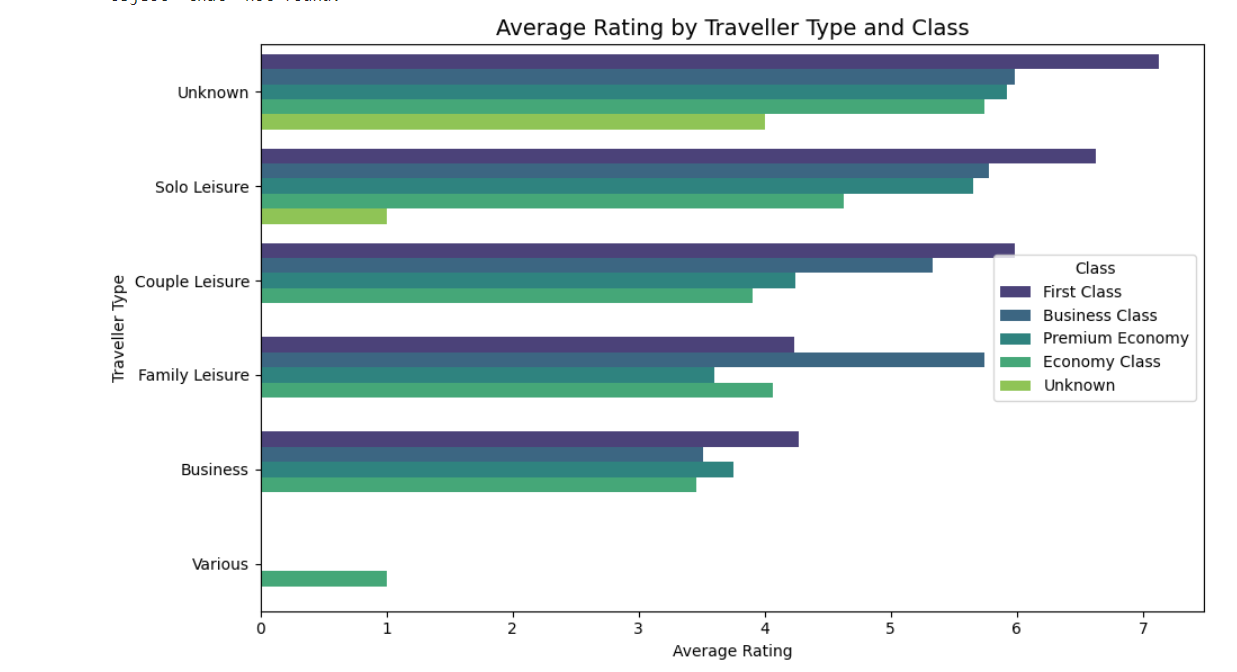


**4.3 Average Rating by Traveller Type and Class**

| **Traveller\_Type / Class** | **Avg. Rating** |
| --- | --- |
| Unknown / First Class | 7.12 |
| Solo Leisure / First Class | 6.63 |
| Business / Economy | 3.46 |
| Solo Leisure / Unknown  Various / Economy | 1.00  1.00 |

**Interpretation:**

* Passengers in **First Class** consistently reported higher satisfaction.
* **Economy Class**, particularly for **business travelers and various**, had lower ratings due to limited comfort, delays, or service issues.
* Leisure travelers showed greater tolerance, aligning with travel purpose expectations.



**5️ Feature Engineering**

A **binary target variable** (Satisfaction) was created based on the passenger’s rating:

df["Satisfaction"] = df["Rating"].apply(lambda x: 1 if x >= 5 else 0)

* **1 → Satisfied**
* **0 → Dissatisfied**

**Features Used for Prediction:**

* Traveller\_Type (categorical) :Different traveler types tend to have different expectations and satisfaction drivers.

*1.Business travelers* may prioritize punctuality, Wi-Fi, and seat comfort.

*2.Leisure travelers* often focus on in-flight service, food, and entertainment.  
 Analyzing traveler type helps the model capture how satisfaction patterns vary across these groups.

* Class (categorical)

1.The travel class directly affects the service level and comfort provided to passengers.

2.Higher-tier classes generally offer better seating, meals, and personal space, which strongly correlate with higher satisfaction ratings.  
 Including this feature allows the model to distinguish between passengers experiencing different service standards.

* Verified (categorical)

1.Verified reviews are typically more authentic and less biased.

2.Verified passengers often provide reliable feedback that aligns closely with their actual travel experience.

3.Non-verified reviews might include exaggerated or unreliable opinions.  
Adding this feature helps the model account for the credibility of each rating.

* Route(categorical)

The route can influence satisfaction levels due to differences in **flight duration, regional service quality, aircraft type, and airport facilities**.

**Target**

* Sentiment\_Score (numeric) (Target)

Categorical features were **label-encoded**, and continuous values were standardized with **StandardScaler** for consistent feature scaling.

**6️ Model Selection**

**Why Logistic Regression?**

**Logistic Regression** was selected because:

* The target variable is **binary** (Satisfied vs. Dissatisfied).
* It outputs **probability-based predictions**, ideal for interpretation.
* It performs well on small to medium datasets.
* Coefficients reveal how individual features influence satisfaction.

While other models like Random Forests or SVMs can capture nonlinear patterns, **this project prioritizes interpretability and transparency** over marginal accuracy gains.

**7️ Model Training and Evaluation**

The data was split into **80% training** and **20% testing** sets.  
After scaling features, the **Logistic Regression** model was trained using:

LogisticRegression(max\_iter=1000)

**Performance Metrics**

| **Metric** | **Score** |
| --- | --- |
| Accuracy | **0.7888** |
| Precision | **0.7493** |
| Recall | **0.8314** |
| F1-Score | **0.7882** |

**Interpretation:**

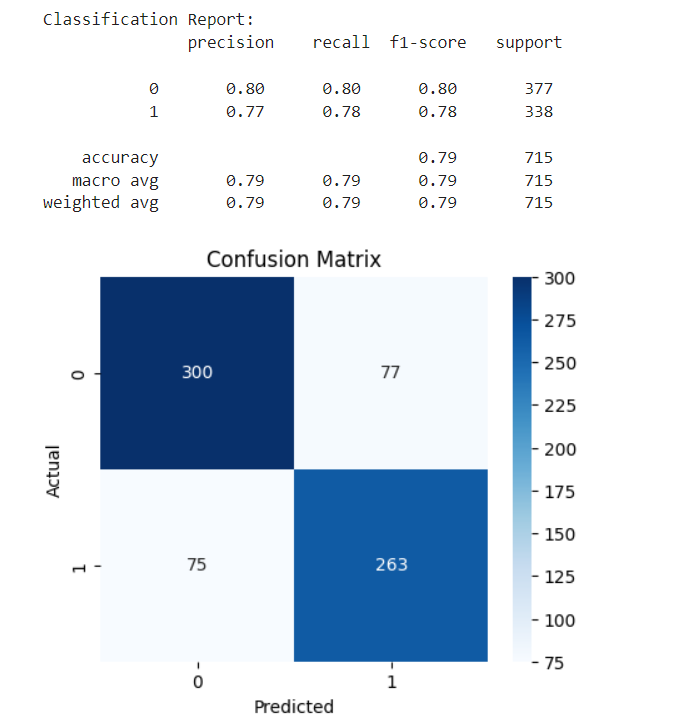
* The model achieved ~79% accuracy, indicating strong predictive ability.
* High recall for “Satisfied” passengers shows good sensitivity in identifying positive experiences.
* Balanced precision and recall confirm that the model generalizes well without overfitting.

**Comparison with Random Forest**

A **Random Forest Classifier** was trained for comparison.

| **Metric** | **Logistic Regression** | **Random Forest** |
| --- | --- | --- |
| Accuracy | 0.7888 | 0.7874 |
| Precision | 0.7493 | **0.7735** |
| Recall | **0.8314** | 0.7781 |
| F1-Score | **0.7882** | 0.7758 |

**Observation:**  
While both models performed similarly, Logistic Regression achieved **better recall and interpretability**, making it the preferred choice.



**8️ Model Explainability**

**8.1 SHAP (SHapley Additive exPlanations)**

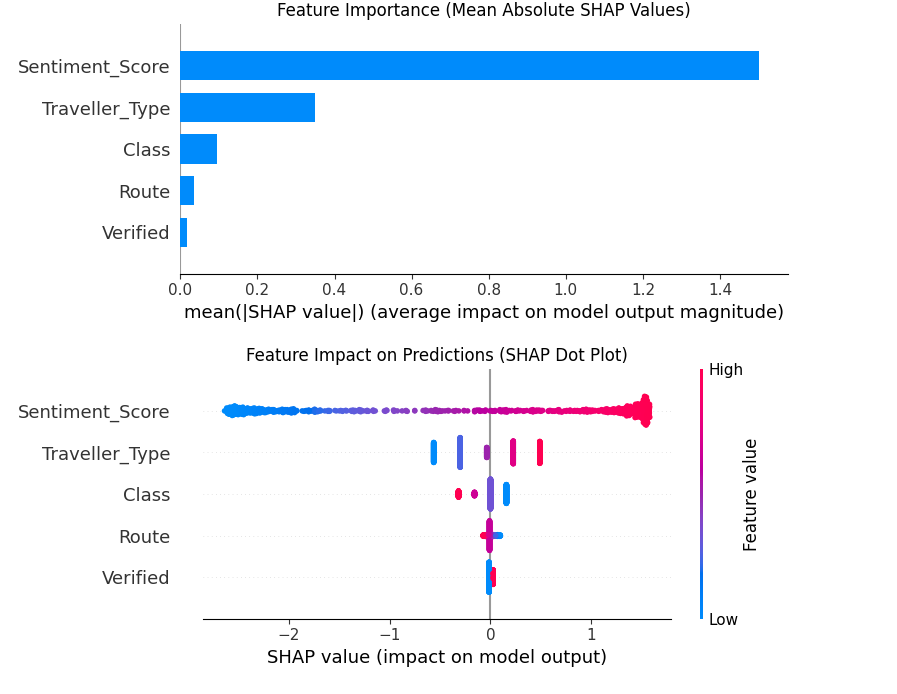
SHAP was used to interpret **feature importance** and **directional influence**.

**Findings:**

* **Sentiment\_Score** had the strongest positive effect — emotional tone dominates satisfaction perception.
* **Traveller\_Type** then **Class** followed, indicating service tier and travel purpose are strong predictors.
* **Followed** by **Route** had minor yet observable effects, showing certain routes have higher or lower satisfaction averages.
* Lastly, **Verified** reviews had the least effect and showed higher satisfaction, reflecting credibility and authenticity.

**Interpretation of Plots:**

* **Bar Plot:** Ranked mean SHAP values show global importance.
* **Dot Plot:** Shows how high or low feature values push predictions toward “Satisfied” or “Dissatisfied.”

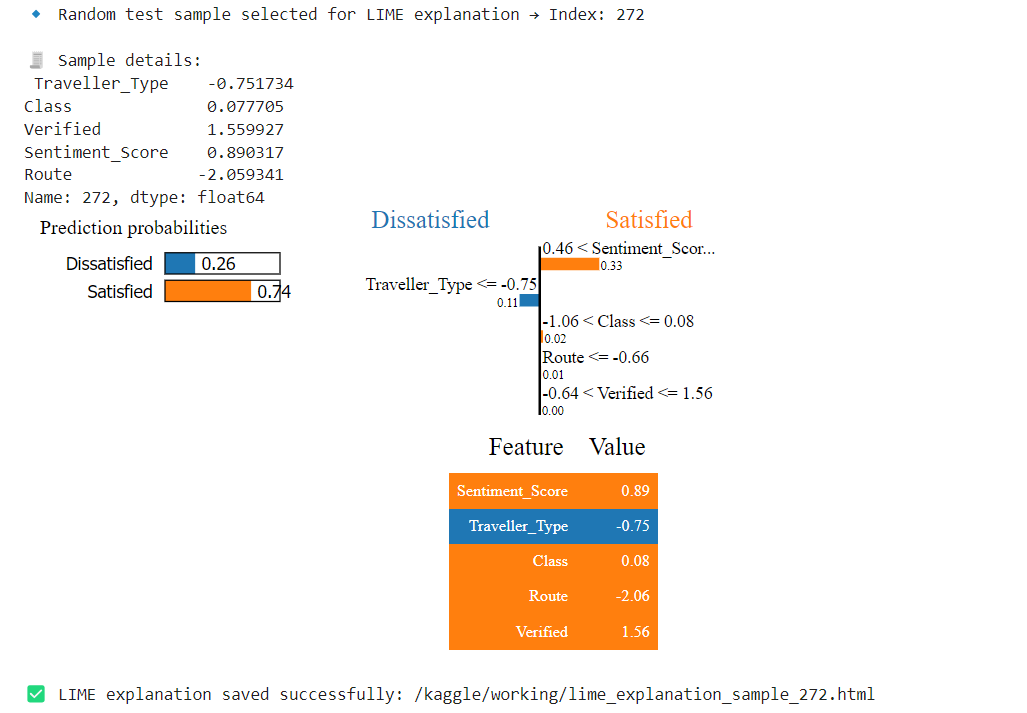


**8.2 LIME (Local Interpretable Model-Agnostic Explanations)**

LIME explains **individual predictions** by perturbing input values and observing prediction changes.

This LIME explanation corresponds to **sample index 272** from the test data, which the model predicted as **“Satisfied” with 74% confidence**. The **Sentiment\_Score (0.89)** had the strongest positive influence on the prediction — reviews with high sentiment scores strongly indicate satisfaction. The **Traveller\_Type (−0.75)** also supported the prediction slightly, meaning this encoded traveler type (likely a leisure or family traveler) correlated with positive experiences. Meanwhile, **Class (0.08)** and **Verified (1.56)** had mild positive contributions, suggesting that verified reviews and mid-to-high class travelers tend to be more satisfied. The **Route (−2.06)** contributed negatively, implying certain routes are linked to lower satisfaction. Overall, the model relied primarily on emotional tone and travel characteristics to classify this passenger as satisfied.

This shows LIME’s strength in providing **case-specific interpretability**, valuable for customer insight tools.



**9️ Inference Function**

Inference function:

def inference\_function(traveller\_type, flight\_class, verified, sentiment\_score, route="London to Amman"):

    """

    Takes raw user input and returns model prediction ('Satisfied' or 'Dissatisfied').

    """

    # 1️⃣ Create a DataFrame with the same structure as training data

    sample = pd.DataFrame({

        "Traveller\_Type": [traveller\_type],

        "Class": [flight\_class],

        "Verified": [verified],

        "Sentiment\_Score": [sentiment\_score],

        "Route": [route]  # Added to match training features

    })

    # 2️⃣ Apply the same label encoding used during training

    for col in ["Traveller\_Type", "Class", "Verified", "Route"]:

        if col in label\_encoders:

            sample[col] = label\_encoders[col].transform(sample[col])

    # 3️⃣ Scale the sample using the same scaler as training

    sample\_scaled = scaler.transform(sample)

    # 4️⃣ Predict

    prediction = model.predict(sample\_scaled)

    prediction\_proba = model.predict\_proba(sample\_scaled)

    # 5️⃣ Decode

    label = "Satisfied" if prediction[0] == 1 else "Dissatisfied"

    confidence = np.max(prediction\_proba)

    print(f"Prediction: {label} (Confidence: {confidence:.2f})")

    return label, confidence

**Example Output:**

Prediction: Satisfied (Confidence: 0.70)

('Satisfied', 0.6975232803830266)

This enables **real-time prediction**, suitable for deployment in airline dashboards or APIs, supporting customer experience monitoring and personalized service feedback.

**🔟 Conclusion**

This project successfully demonstrated an **end-to-end machine learning pipeline** integrating text sentiment analysis with structured airline data to predict passenger satisfaction.

**Key Findings**

* **Sentiment\_Score** is the most influential predictor — emotional tone reflects passenger happiness accurately.
* **Class** and **Traveller\_Type** strongly influence satisfaction, aligning with travel expectations.
* Logistic Regression achieved **~79% accuracy** and offered **transparent feature contributions**, making it ideal for explainable AI use cases.

**Future Work**

* Integrate more behavioral data (e.g., delays, cancellations, loyalty status).
* Explore multilingual sentiment models for global airlines.
* Compare interpretability vs. accuracy using tree-based or neural network models.
* Deploy as a live **customer satisfaction monitoring system**.

✅ **Final Model:** Logistic Regression  
✅ **Accuracy:** 0.7888  
✅ **Chosen for:** Explainability, interpretability, and stable predictive performance