Project : Flight Booking Price Prediction

Name : Deepak Kumar
Roll No. : 2401030344
Email ID. : 2401030344@mail.jiit.ac.in
Semester. : 3rd Semester
Course. : Introduction to Python Programming & Machine Learning

Project Overview

This project classifies flight ticket prices into categories — Low, Medium, and High — based on features such as airline, source, destination, departure time, arrival time, class, etc.

It uses Python libraries like Pandas, Matplotlib, Seaborn, and Scikit-learn for data processing, visualization, and model building.

GitHub Repository

I have uploaded the complete project on my GitHub repository.

You can view the source code, cleaned dataset, visualizations, and machine learning models here:

Flight Booking Price Prediction – GitHub Repository

https://github.com/Deepak152-coder/Flight-Booking-Price-Prediction

This repository includes:

- *Cleaned dataset (`Clean_Dataset.csv`)*

- *Final project notebook (`FinalProject.py`)*
- *Visualizations and EDA code*
- *Machine learning model training and evaluation*

Technologies Used

***-> Python 3.13.3** – Programming language used to build the project*
***-> Pandas** – For data loading and preprocessing*
***-> NumPy** – For numerical operations*
***-> Seaborn & Matplotlib** – For data visualization*
***-> Scikit-learn** – For model building, training, and evaluation*
***-> VS Code** – Code editor and development environment*

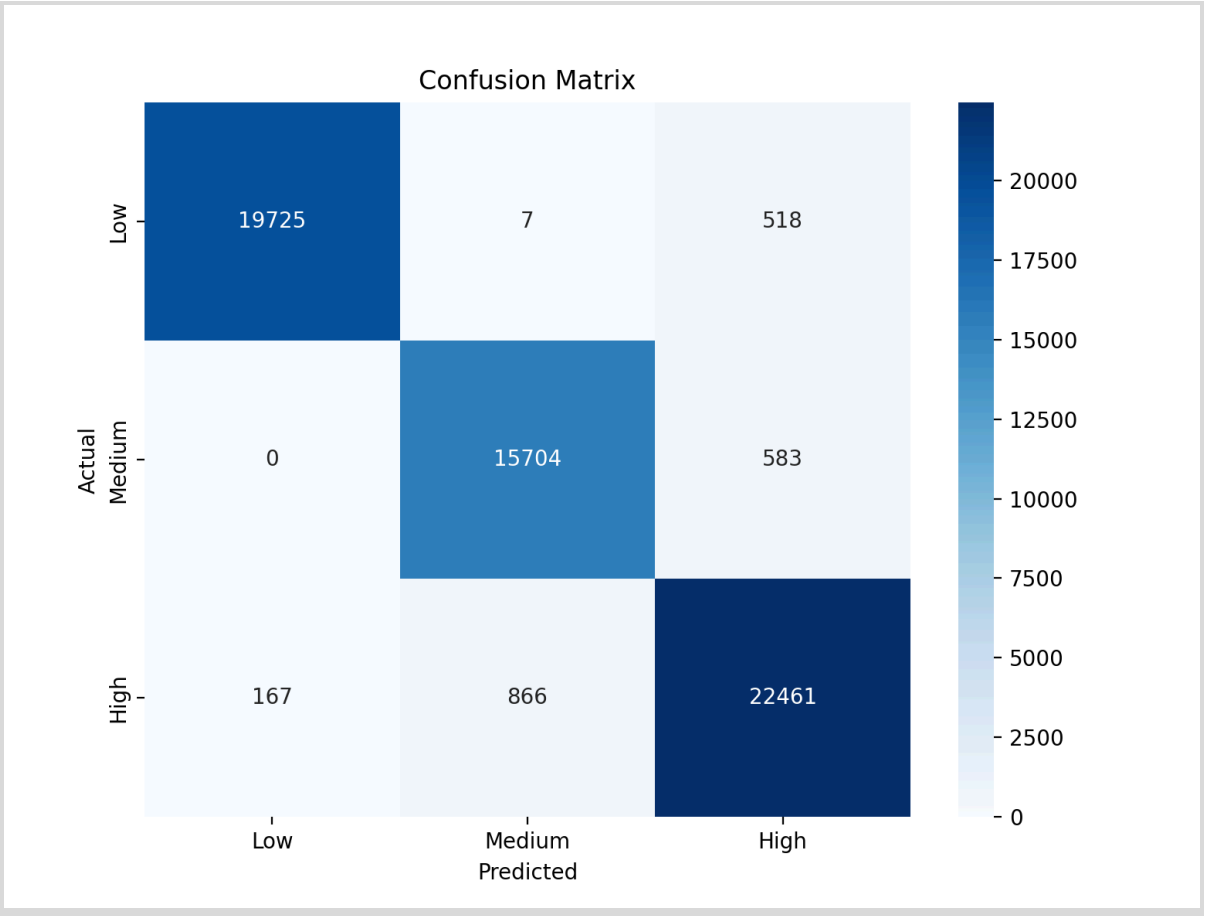
Code:-

```
print("\n" + "-"*60 + " Project: Flight Booking Price Prediction " + "-"*60 + "\n")

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.ensemble import RandomForestClassifier

df = pd.read_csv("Clean_Dataset.csv")
df = df.drop(columns=["Unnamed: 0"], errors='ignore')

price_bins = [0, 5000, 15000, float('inf')]
price_labels = ['Low', 'Medium', 'High']
df['price_category'] = pd.cut(df['price'], bins=price_bins,
labels=price_labels)
```

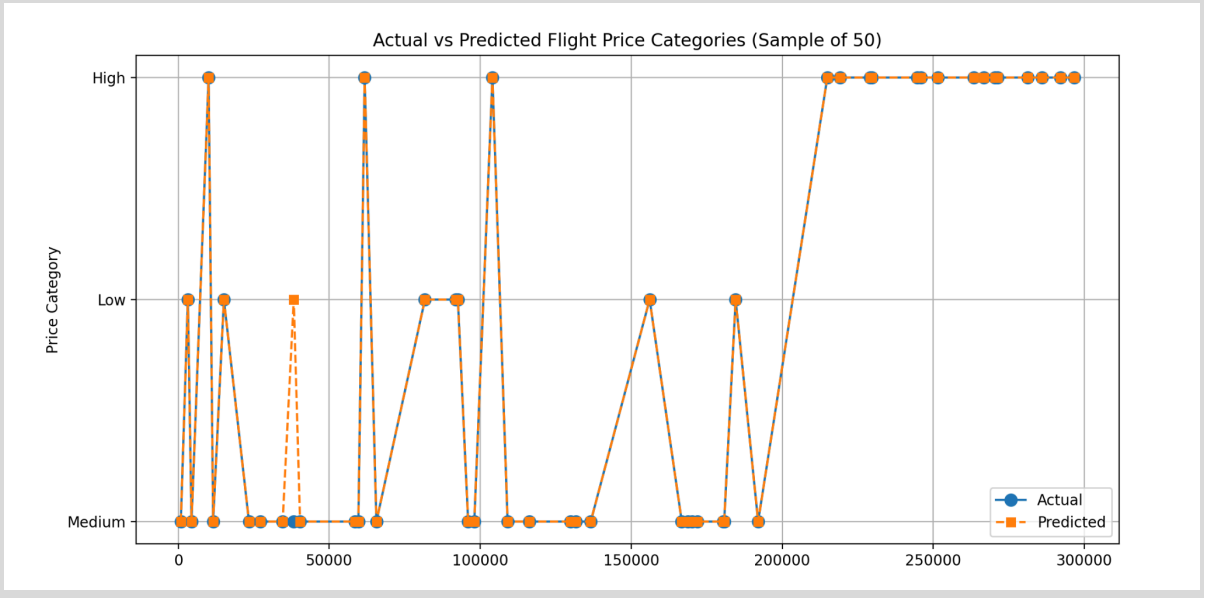
```
le = LabelEncoder()
categorical_cols = df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    if col != 'price_category':
        df[col] = le.fit_transform(df[col].astype(str))

X = df.drop(columns=['price', 'price_category'])
y = df['price_category']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

rfc = RandomForestClassifier(random_state=42)
rfc.fit(X_train, y_train)
y_pred = rfc.predict(X_test)

print("\n" + "-"*45 + " Confusion Matrix " + "-"*45 + "\n")
cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=price_labels, yticklabels=price_labels)
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

print("\n" + "-"*42 + " Classification Report " + "-"*42 + "\n")
print(classification_report(y_test, y_pred, target_names=price_labels))

print("\n" + "-"*40 + " Predicted vs Actual Output " + "-"*40 + "\n")
results = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
sample_results = results.sample(50, random_state=42).sort_index()

plt.figure(figsize=(12, 6))
plt.plot(sample_results['Actual'], 'o-', label='Actual', markersize=8)
plt.plot(sample_results['Predicted'], 's--', label='Predicted', markersize=6)
plt.title('Actual vs Predicted Flight Price Categories (Sample of 50)')
plt.ylabel('Price Category')
plt.legend()
plt.grid(True)
plt.show()

print("\n" + "-"*73 + " End of Project " + "-"*73 + "\n")
```

Output (Screenshots):-

```
---------------------------------------------------------- Project: Flight Booking Price Prediction ----------------------------------------------------------


---------------------------------------------- Confusion Matrix ----------------------------------------------
```




```
---------------------------------------------- Classification Report ----------------------------------------------

              precision    recall  f1-score   support

         Low       0.99      0.97      0.98     20250
      Medium       0.95      0.96      0.96     16287
        High       0.95      0.96      0.95     23494

    accuracy                           0.96     60031
   macro avg       0.96      0.96      0.96     60031
weighted avg       0.96      0.96      0.96     60031


---------------------------------------------- Predicted vs Actual Output ----------------------------------------------
```




```
---------- End of Project ----------
```

Conclusion :-

This project successfully demonstrates how flight ticket prices can be categorized into levels such as Low, Medium, and High using machine learning.

Through data preprocessing, categorical encoding, and model evaluation, we implemented a classification approach—where the Random Forest Classifier delivered strong performance, as shown by the confusion matrix, classification report, and predicted vs actual comparison.

GitHub Repository :-

You can find the complete project and code here:

Flight Booking Price Prediction – GitHub Repository

https://github.com/Deepak152-coder/Flight-Booking-Price-Prediction

Acknowledgements :-

Thanks to:

=> Datasets from Flight Price Prediction

=> Mentors :-

Himani Gupta (himani.gupta@mail.jiit.ac.in)

Ankur Gupta (ankur.gupta@mail.jiit.ac.in)

=> Python open-source community