12/2/2024

Presented By: Shahzaib Ali

Machine Learning

Project

# Table of Contents

# 1. Introduction

# 2. Objectives

# 3. Data Preprocessing and Feature Engineering

# 4. Exploratory Data Analysis (EDA)

# 5. Predictive Modeling

# 6. Model Optimization

# 7. Testing and Kaggle Submission

# 8. Conclusion

# 9. Appendices

# Advanced Flight Departure Delay Analysis

## Introduction

🚀 Flight delays pose a significant challenge to the aviation industry, affecting operational efficiency, customer satisfaction, and resource management. This project aims to analyze and predict flight departure delays using machine learning models. The provided datasets include flight details, weather information, and delay records, enabling a comprehensive analysis of factors influencing delays and the development of predictive models. 🚀

## Objectives

1. Analyze and preprocess flight and weather data to uncover patterns influencing delays.

2. Train predictive models for:

- Binary classification: On-time vs. Delayed.

- Multi-class classification: No delay, short, moderate, and long delays.

- Regression: Predicting exact delay durations.

3. Evaluate models using performance metrics.

4. Optimize models using hyperparameter tuning and cross-validation.

5. Test the models on unseen data and submit predictions to Kaggle for evaluation. 🌍

# 1. Data Preprocessing and Feature Engineering

## 1.1 Data Integration

🌌 The project integrates multiple datasets:

- Flight Data: Contains flight schedules and delays.

- Weather Data: Provides weather conditions at airports. 🌌

## 1.2 Data Cleaning and Transformation

🔧 1. Handle Missing Values:

- Missing weather data imputed using mean or forward fill methods.

- Flight records with incomplete key information are discarded.

2. Standardize Time Fields:

- Convert scheduled, actual, and estimated times into a uniform datetime format.

3. Feature Engineering:

- Calculate Delay: Compute departure delays as the difference between actual and scheduled times.

- Merge Weather Data: Incorporate weather features like temperature, humidity, and wind speed.

- Extract Temporal Features:

- Day of the week.

- Hour of the day.

- Month of the year. 🔧

# 2. Exploratory Data Analysis (EDA)

## 2.1 Visualizations

📊 1. Delay Distributions:

- Histograms display delay durations, revealing skewness and common delay ranges.

2. Temporal Analysis:

- Line plots show delay trends across hours, days, and months.

3. Category-Wise Analysis:

- Bar charts compare delays by airline, departure airport, and flight status. 📊

## 2.2 Correlation Analysis

🎯 - Scatter plots and heatmaps illustrate the relationship between weather variables (e.g., wind speed) and delays.

- Correlation coefficients quantify the strength of these relationships. 🎯

# 3. Predictive Modeling

## 3.1 Binary Classification

🔖 - Objective: Classify flights as on-time (delay = 0) or delayed (delay > 0).

- \*\*Models Used\*\*:

1. Logistic Regression.

2. Decision Trees.

3. Random Forests.

4. Support Vector Machines (SVM).

- Evaluation Metrics:

- Accuracy.

- Precision, Recall, and F1-score.

- Confusion matrix. 🔖

## 3.2 Multi-Class Classification

🔰 - Objective: Categorize flights into delay classes:

1. No Delay (0 min).

2. Short Delay (<45 min).

3. Moderate Delay (45–175 min).

4. Long Delay (>175 min).

- Models Used:

1. Random Forest Classifier.

2.Linear Regression.

3. Logistic Regression.

4. Decision Tree Classifier

- Evaluation Metrics:

- Class-wise Precision and Recall.

- Confusion Matrix.

- Overall Accuracy and F1-Score. 🔰

## 3.3 Regression Analysis

📊 - Objective: Predict the exact delay duration.

- Models Used:

1. Linear Regression.

2. Decision Tree Classifier

3. Random Forest Classifier.

- Evaluation Metrics:

- Mean Absolute Error (MAE).

- Root Mean Square Error (RMSE). 📊

# 4. Model Optimization

## 4.1 Hyperparameter Tuning

🔧 - Techniques Used:

1. Grid Search.

2. Random Search.

- Parameters Tuned:

- Number of estimators, learning rates, and max depth for tree-based models.

- Regularization strength for regression models. 🔧

## 4.2 Validation

🏆 - k-Fold Cross-Validation:

- Ensures robust model performance by dividing data into k subsets for training and testing. 🏆

## 4.3 Model Comparison

🔸 - Models are compared based on their performance metrics to select the best performer for each task. 🔸

# 5. Testing and Kaggle Submission

## 5.1 Testing on Unseen Data

🚀 - The test dataset is preprocessed similarly to the training data.

- Predictions are generated using the best-performing models. 🚀

## 5.2 Kaggle Submission Format

🔖 - Regression: Predict exact delay durations.

- Classification: Predict delay categories as strings (e.g., “on-time”, “short delay”).

- CSV File Structure:

- Flight Number.

- Type (departure).

- Status (active, canceled).

- Delay prediction. 🔖

## 6. Conclusion

🌍 This project provides insights into factors influencing flight delays and evaluates predictive models to improve delay management. The analysis highlights temporal patterns, weather impacts, and operational challenges. The predictive models demonstrate the utility of machine learning in real-world applications, with potential for integration into airline systems for proactive decision-making. 🌍

## Appendices

1. Code Snippets: Key Python code for preprocessing, model training, and evaluation. 🔧

2. Visualization Samples: Plots generated during EDA. 📊

3. Dataset Descriptions: Details of integrated datasets and feature engineering steps. 🎯