12/12/2024

PROJECT REPORT

MACHINE LEARNING

Sahal Saeed
22i-0476
AI-A
FAST NUCES

Table of Contents

Phase 1: Data Preprocessing and Feature Engineering

1. Data Integration

Objective: Combine the weather dataset with the flight dataset to enrich the features for analysis and modeling.

Steps:

1. Join the weather dataset and flight dataset on common attributes such as date, time, and location (e.g., airport).

2. Ensure the integrity of the integrated dataset by validating the correctness of merged records.

2. Data Cleaning and Transformation

2.1 Handle Missing Values

- Identify missing values in critical fields (e.g., scheduled time, actual time, weather attributes).

- Apply appropriate imputation techniques:

  - Mean/median for numerical fields.

  - Mode for categorical fields.

  - For time-related fields, interpolate or forward-fill based on temporal patterns.

2.2 Format Time Fields

- Convert time fields (e.g., Scheduled Departure, Actual Departure, Estimated Arrival) into a standard `datetime` format.

- Calculate additional time-based differences such as:

  - Actual Departure - Scheduled Departure (departure delay).

  - Scheduled Arrival - Scheduled Departure (planned flight duration).

3. Feature Engineering

3.1 Calculate Departure Delay

- Compute the `departure_delay` as the difference between `actual_departure_time` and `scheduled_departure_time`.

- Categorize delays for multi-class classification tasks.

3.2 Merge Weather Data

- Extract relevant weather features (e.g., temperature, wind speed, humidity).

- Merge weather data with flight data based on timestamp and location (e.g., airport code).

3.3 Extract Temporal Features

- Derive additional features to capture patterns:

1. *Day of the Week:* Extract Day names (e.g., Monday, Tuesday).
2. *Hour of the Day:* Bin time into hourly intervals.
3. *Month of the Year:* Extract month names.

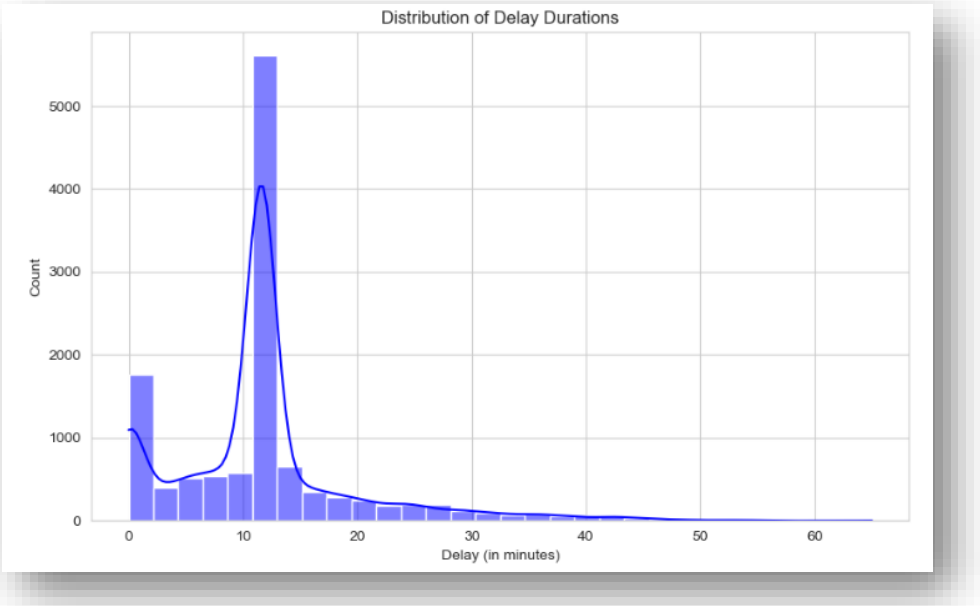
- Optional: Add derived features such as holidays or seasonal indicators.

Phase 2: Exploratory Data Analysis (EDA)

1. Visualizations

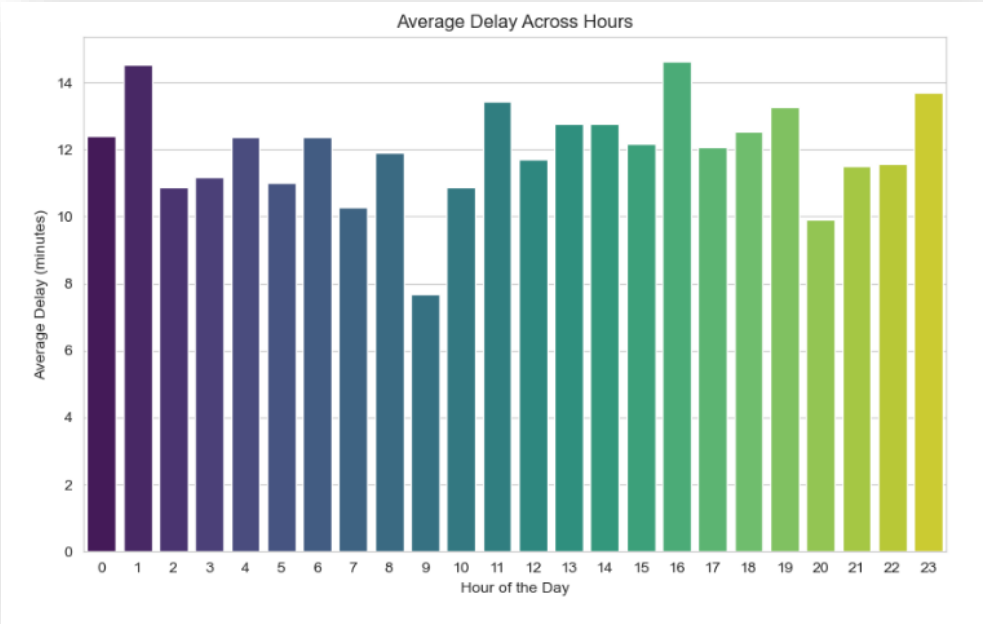
1.1 Delay Distributions

- Plot a histogram to visualize the distribution of delay durations.

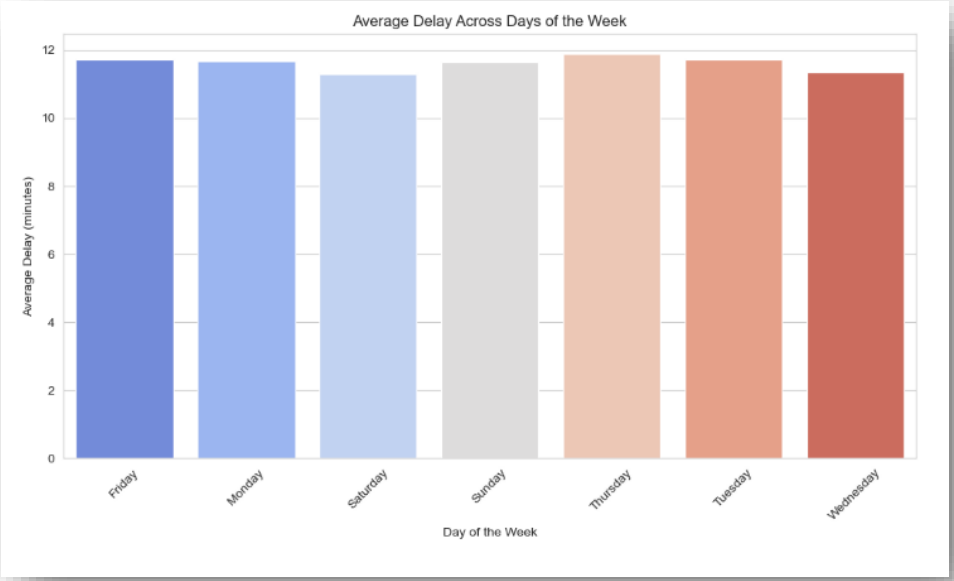


1.2 Temporal Analysis

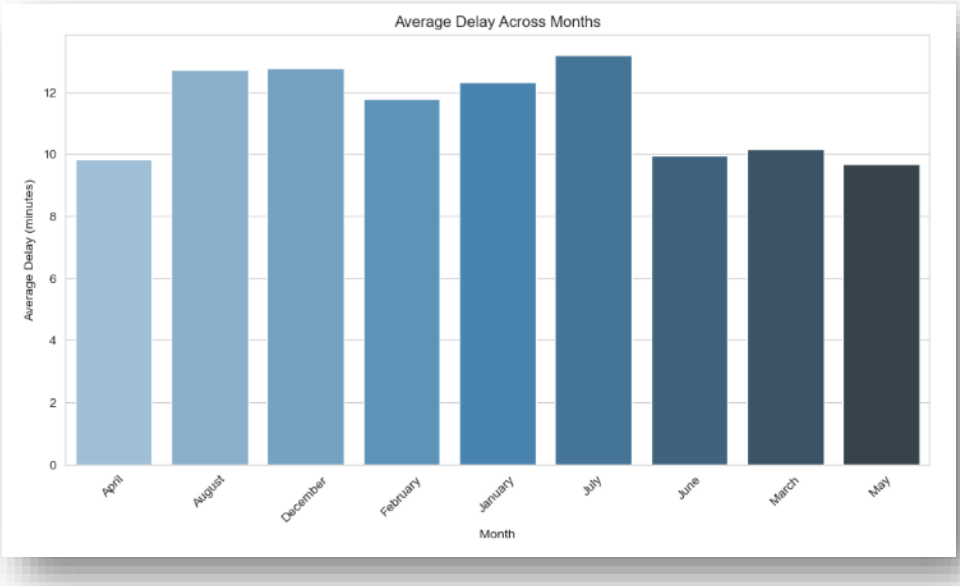
- Use line plots or bar charts to analyze delays across different temporal dimensions:
  - Hour of the day.
  - Day of the week.
  - Month of the year.
- Average delay across hours



Average delay across week



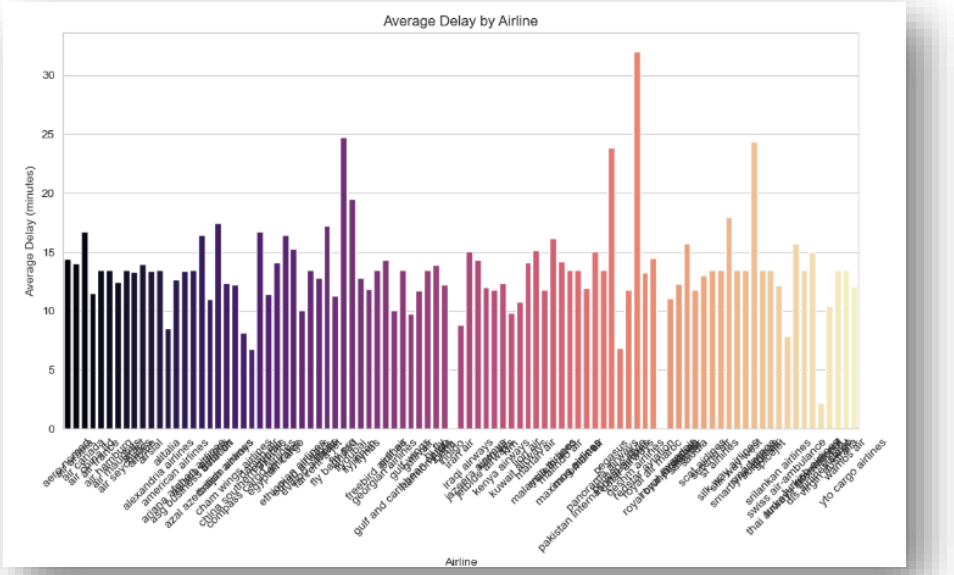
Average delay across month



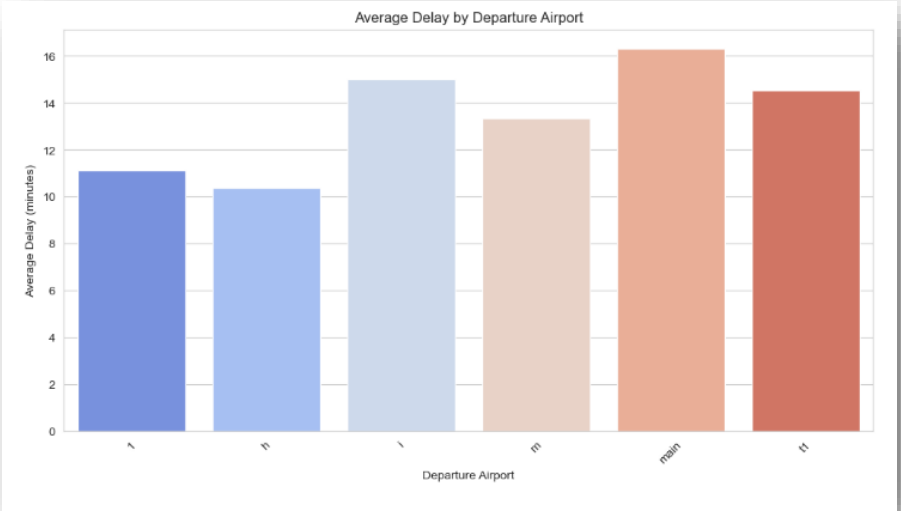
1.3 Category-Wise Analysis

- Group delays by:

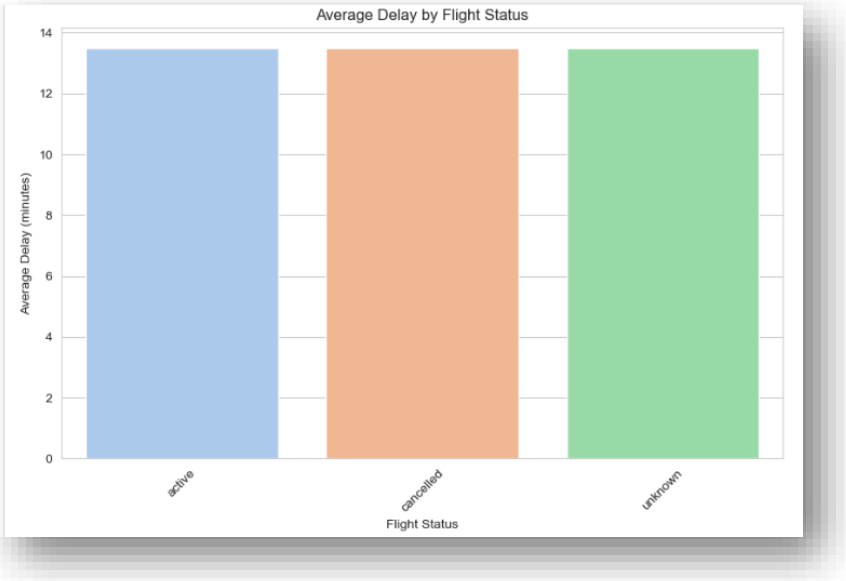
  - Airline.



  - Departure airport.



- Flight status (on-time or delayed).

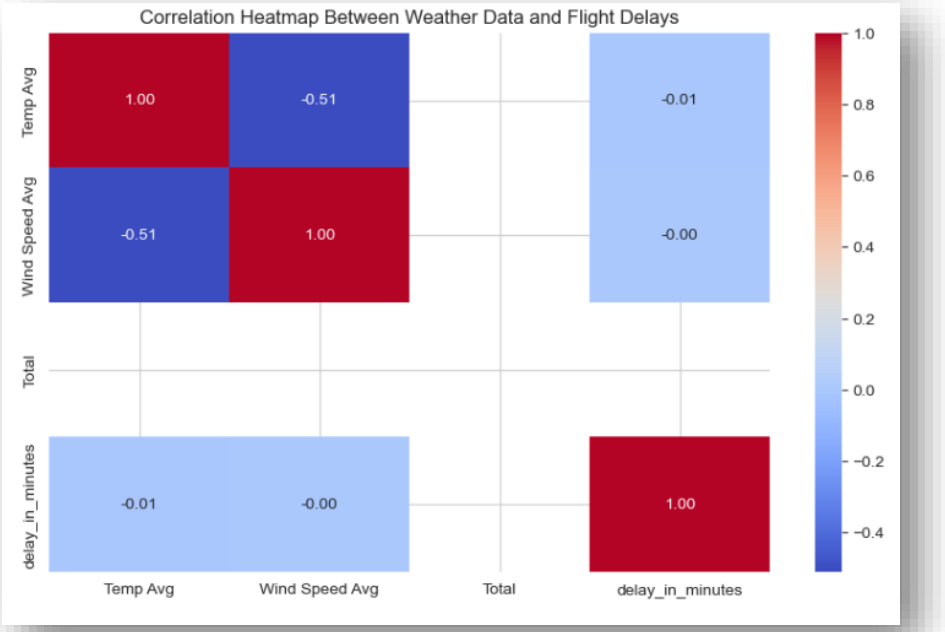


2. Correlation Analysis

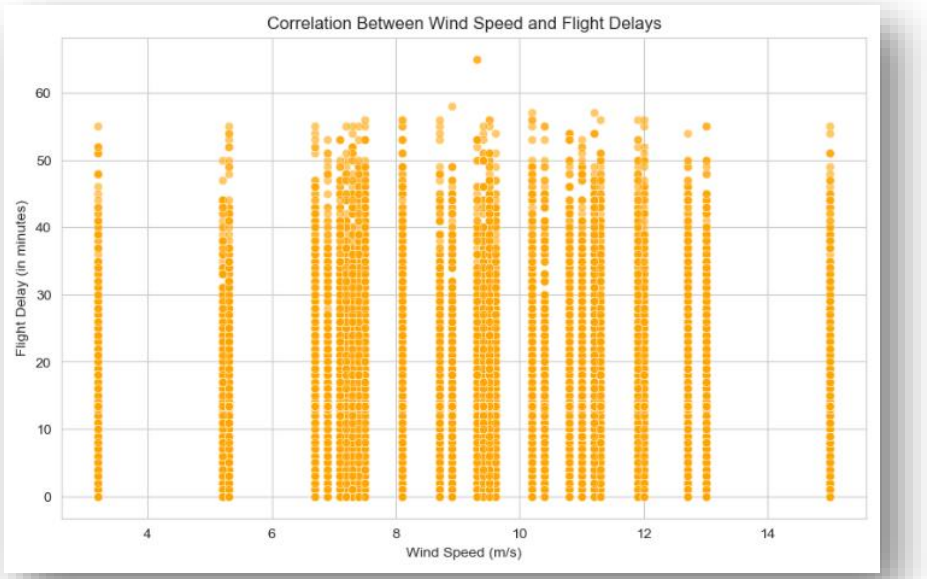
- Analyze the relationship between weather features and flight delays.

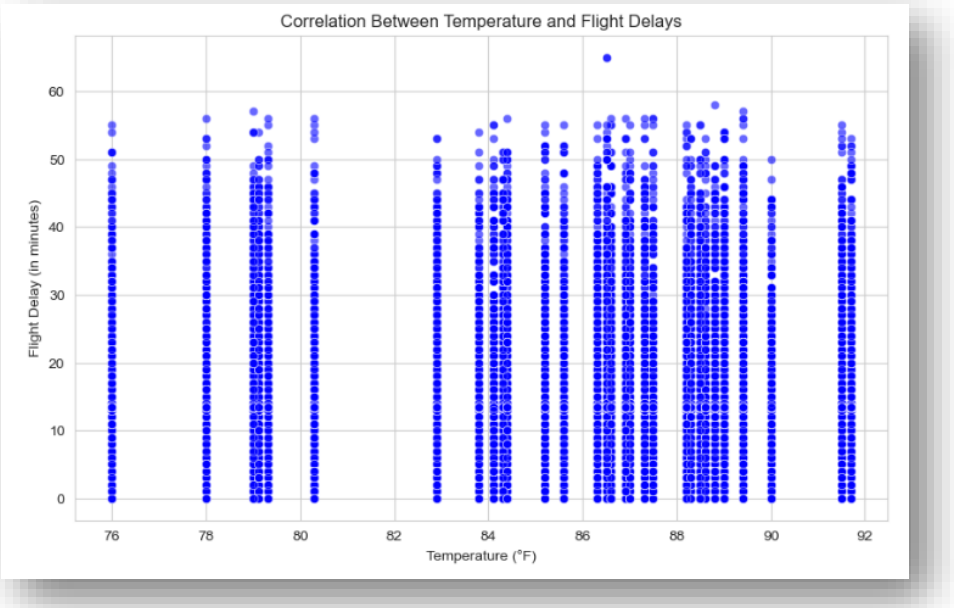
- Use visualizations such as:

1. Heatmaps to show correlations between numerical attributes.



2. Scatter plots for delay duration vs. temperature or wind speed.





3. Comparison

- Compare delay statistics across training and testing datasets to ensure consistency in data distributions.

Phase 3: Analytical and Predictive Tasks

1. Classification Tasks

1.1 Binary Classification

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

Steps:

1. Train a binary classification model using features such as weather, time, and airline data.

2. Evaluate performance:

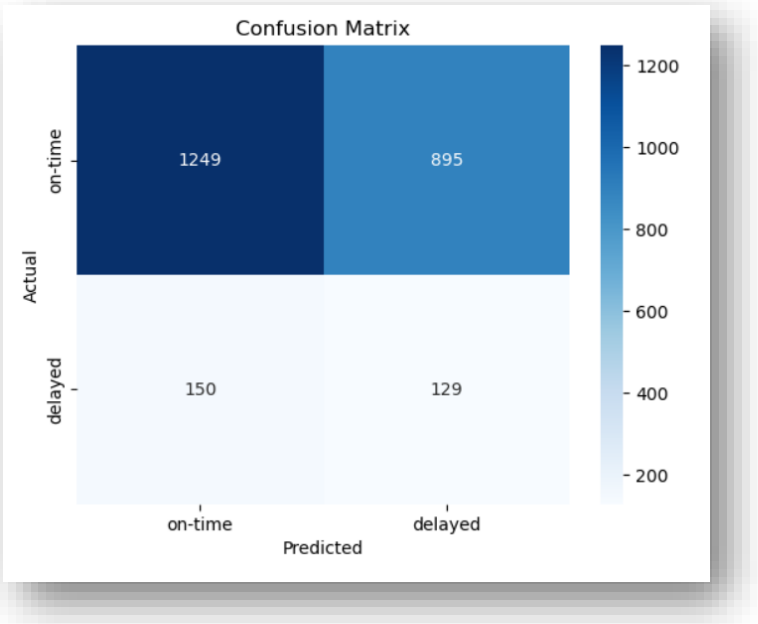
Accuracy: Measure overall correctness.

Precision-Recall: Assess balance between false positives and false negatives.

F1-Score: Evaluate the harmonic mean of precision and recall.

Class-wise Precision-Recall: Break down precision and recall for each class.

Confusion Matrix: Visualize prediction errors.



1.2 Multi-Class Classification

Objective: Categorize flights into:

1. No Delay (0 min).
2. Short Delay (<45 min).
3. Moderate Delay (45–175 min).
4. Long Delay (>175 min).

Steps:

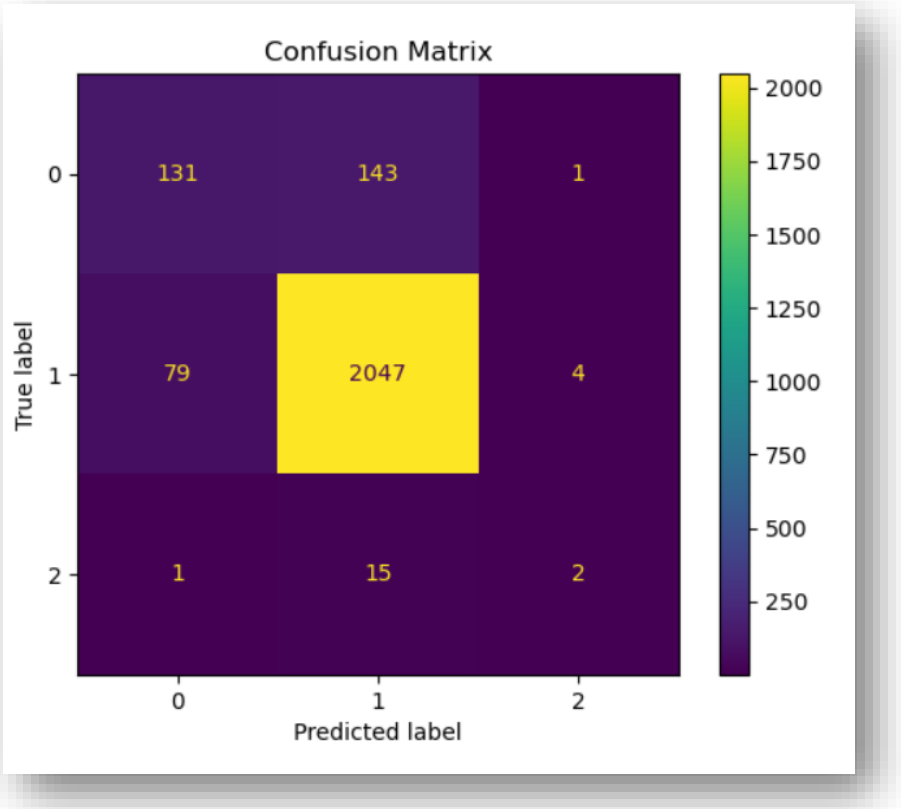
1. Train a multi-class classification model.
2. Evaluate performance:

   Accuracy: Measure classification accuracy.

   Precision-Recall: Evaluate performance per class.

   F1-Score: Assess overall balance between precision and recall.

   Confusion Matrix: Identify specific areas of misclassification.



2. Regression Analysis

Objective: Predict exact delay durations.

Steps:

1. Train regression models (e.g., Linear Regression, Random Forest Regressor, Gradient Boosting).

2. Validate models using cross-validation techniques.

3. Evaluate performance:

Mean Absolute Error (MAE): Average magnitude of errors.

Root Mean Square Error (RMSE): Penalizes larger errors more heavily.

Phase 4: Model Optimization and Evaluation

1. Hyperparameter Tuning

- Use techniques like:

Grid Search: Exhaustive search over a parameter grid.

Random Search: Randomly sample parameter combinations.

2. Validation

- Apply k-fold cross-validation to:

- Reduce overfitting.

- Assess model performance on different subsets of the training data.

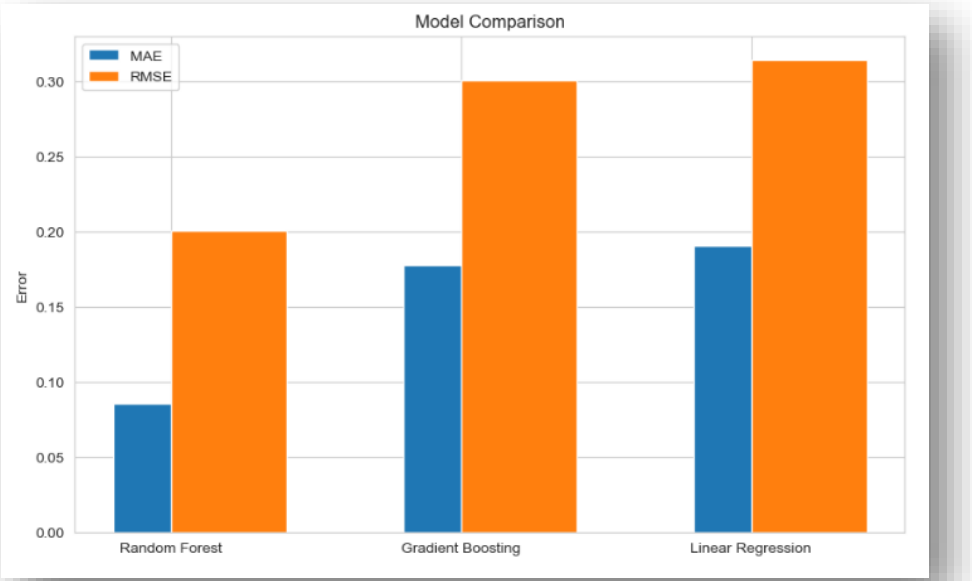
3. Model Comparison

- Compare models based on:

- Performance metrics (e.g., Accuracy, MAE, RMSE).

- Robustness to unseen data.

- Training and inference time.



Phase 5: Model Testing

1. Predictions on Test Dataset

- Use trained models to make predictions on the test dataset.

- Evaluate and save predictions:

Regression: Predict exact delay durations.

Classification: Predict delay categories (No Delay, Short, Moderate, Long) or binary outcomes (on-time/delayed).

2. Submission Format

- For classification tasks:

- Delay column must contain string labels (“on-time” or “delayed”).

- Avoid numerical values like 0 or 1 in the submission.

- For regression tasks:

- Submit predicted delay durations directly as numerical values.