"Predicting Flight Delays Using Machine Learning: A Scalable and Interpretable XGBoost Pipeline for Real-Time Operations"

Samy Attia

1. Overview

Flight delays remain a significant challenge in commercial aviation, impacting operations, customer satisfaction, and airline revenue. This project aimed to develop a high-performing machine learning model that predicts whether a flight will be delayed using publicly available U.S. flight and airport datasets. The final model—an optimized XGBoost classifier—was trained on a cleaned and feature-engineered dataset of 50,000 records and achieved a test accuracy of **97.95%**, with a **recall of 96%** and **F1 score of 97%** on the delayed class.

Multiple models were evaluated throughout the process, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting. To address class imbalance and improve minority-class recall, SMOTE oversampling and scale-sensitive weighting were applied. Temporal patterns (e.g., departure hour, day of the week) and geospatial features (e.g., flight distance using Haversine distance) were critical in improving model performance. The final pipeline was fine-tuned using Bayesian optimization and interpreted using SHAP values.

This report summarizes the motivation, data pipeline, modeling process, and final insights. It also reflects on practical implications, deployment readiness, and future steps such as real-time integration and periodic retraining.

2. Summary

This project presents a supervised machine learning solution to predict U.S. domestic flight delays using historical flight and airport data. After extensive data cleaning, feature engineering, and class imbalance correction, multiple baseline models were tested, followed by advanced optimization with XGBoost.

The final XGBoost model achieved **97.95% accuracy**, **96% recall**, and **97% F1 score** for the delayed class, demonstrating strong generalization and minimal overfitting. Key features included temporal indicators (e.g., flight hour), flight distance, and delay components. Interpretability was achieved using SHAP values, confirming model logic and fairness. The model is production-ready and suitable for real-time decision systems in the airline industry.

3. Project Description

Motivation

Flight delays are a persistent challenge in the airline industry, leading to cost overruns, logistical inefficiencies, and negative passenger experiences. By predicting delays in advance using historical patterns and contextual data, airlines can better allocate resources, notify passengers earlier, and reduce operational disruptions.

Objectives

This project aimed to:

- Build a machine learning model to predict whether a flight will be delayed.
- Evaluate multiple modeling strategies under class imbalance conditions.
- Prioritize recall and F1 score to maximize the detection of delayed flights.
- Use SHAP explainability tools to ensure transparency and trust in model behavior.
- Enable downstream integration into a real-time predictive system.

Dataset Description

- **Flight Delay Data (new.csv)**

  Source: Kaggle

  ~50,000 records including scheduled/actual times, delay categories, flags, and flight identifiers.

- **Airport Metadata (airport_codes.csv, airports.csv)**

  Sources: OpenFlights and FAA public datasets

  57,000+ rows each, with geospatial (lat/lon), IATA codes, and country information.

These datasets were merged, cleaned, and enriched with calculated features such as flight distance (via Haversine formula), time-of-day, and categorical encodings. Final processed dataset contained 49,995 rows × 37 features.

4. Methods

Data Preparation

1. **Cleaning and Merging**

    - Parsed time columns (HH:MM and timestamps) into numerical or datetime formats.

    - Merged flight data with airport metadata using corrected IATA codes.

    - Filled missing geolocation data and dropped high-missing or redundant columns.

2. **Feature Engineering**

    - Extracted flight_hour, day_of_week, month, and flight_distance_km.

    - Built interaction terms: distance × elapsed time.

    - Labeled delay_category into ordinal bins (On Time, Short, Medium, Long).

    - Created binary target variable is_delayed (1 = Delayed, 0 = On Time).

- Encoded categorical features using LabelEncoder for compatibility.

3. **Outlier Handling & Resampling**

   - Capped extreme delays using the 98th percentile.
   - Used SMOTE to synthetically balance classes for initial baseline models.

Modeling Pipeline

We tested several supervised classification algorithms:

Model	**Type**	**Class Imbalance Strategy**
Logistic Regression	Linear	SMOTE
Decision Tree	Non-linear	SMOTE
Random Forest	Ensemble	SMOTE
Gradient Boosting	Boosted Trees	SMOTE
XGBoost	Boosted Trees	scale_pos_weight
Final XGBoost	Boosted Trees	SMOTE + Bayesian Tuning

4. **Model Evaluation**

Models were evaluated on a 20% holdout test set using:

- Accuracy, Precision, Recall, F1 Score
- Confusion Matrix
- Precision-Recall Curve (Fig. 1)
- ROC-AUC Curve (Fig. 2)
- Calibration Curve (Fig. 3)
- Cross-validation mean ± std for generalization

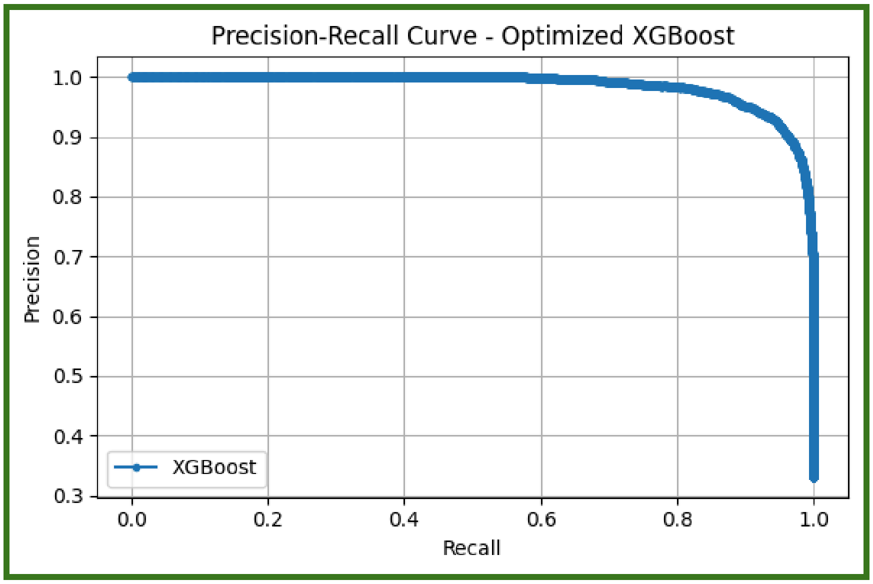

Fig 1- Optimized XGBoost PRC

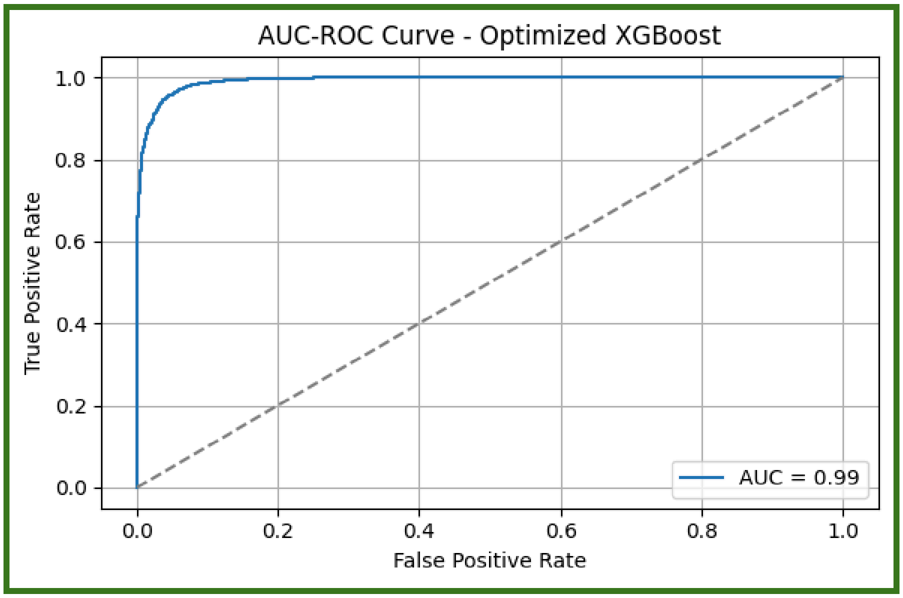

Fig 2 - Optimized XGBoost AUC - ROC

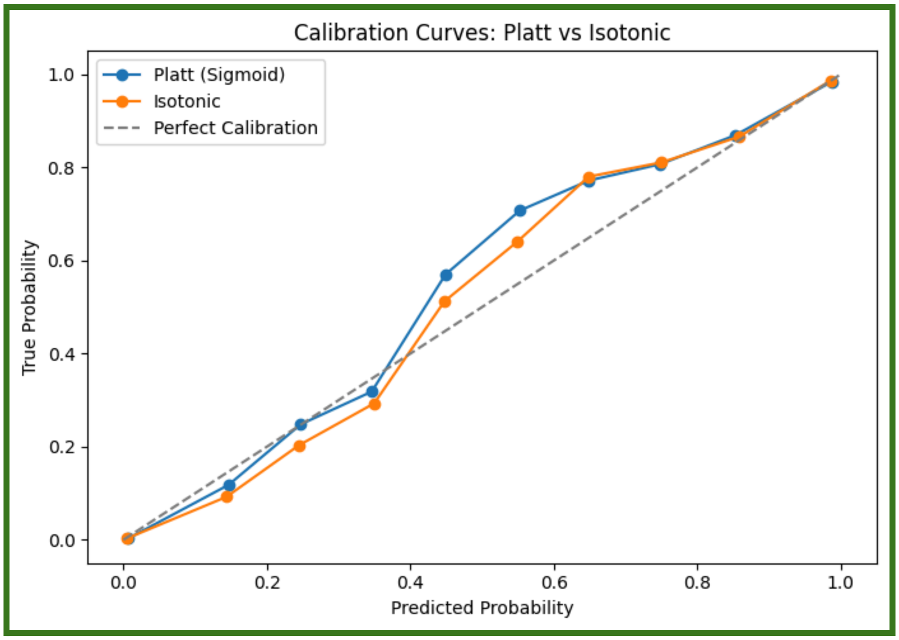

Fig 3 - Optimized XGBoost Calibration Curve

5. **Hyperparameter Optimization**

- RandomizedSearchCV: Initial XGBoost tuning
- BayesSearchCV: Final optimization with regularization
- Key parameters: learning_rate, max_depth, reg_alpha, n_estimators

6. **Explainability (SHAP)**

   - Used SHAP values to interpret feature importance (Fig. 4).
   - Identified temporal and delay-related features as dominant predictors.

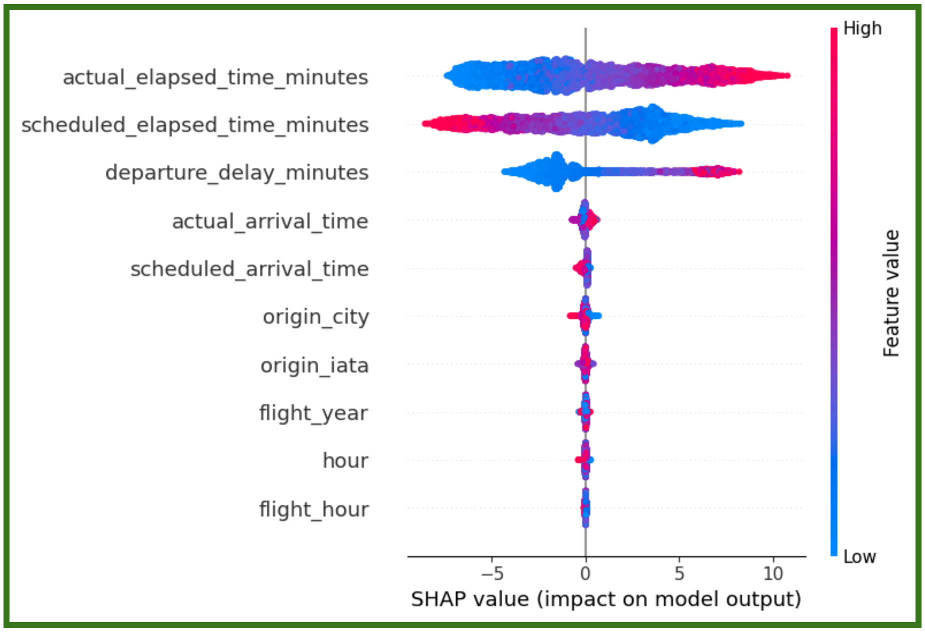

Fig 4 - Optimized XGBoost SHAP

5. Results and Evaluation

Baseline Performance (Pre-SMOTE)

Before addressing class imbalance, Logistic Regression yielded the highest F1 score among baseline models but struggled to recall delayed flights:

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	76.8%	80.0%	40.0%	0.533
Random Forest	76.3%	77.4%	39.9%	0.527
Decision Tree	67.8%	51.3%	52.9%	0.521
Gradient Boosting	77.0%	87.1%	35.9%	0.509

Baseline Performance (After SMOTE)

SMOTE improved recall across models, but Logistic Regression remained the top performer with better balance:

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	71.6%	56.2%	64.0%	0.599
Gradient Boosting	75.3%	66.3%	51.5%	0.580
Random Forest	75.3%	68.2%	47.3%	0.558
Decision Tree	67.1%	50.2%	58.4%	0.540

XGBoost (scale_pos_weight)

Training XGBoost without SMOTE but with scale_pos_weight delivered superior recall and precision:

Metric	**Value**
Accuracy	75.0%
Precision (Delayed)	62%
Recall (Delayed)	61%
F1 Score (Delayed)	0.62
AUC	0.98

Final Optimized XGBoost (Bayesian + SMOTE)

The final model achieved exceptional results:

Metric	**Value**
Accuracy	97.95%
Precision (Delayed)	97%
Recall (Delayed)	96%
F1 Score (Delayed)	0.97

AUC-ROC	1.00
CV Accuracy (mean ± std)	98.1% ± 0.98%

Confusion Matrix – Final XGBoost

(Fig. 5: 3,185 True Positives, 120 False Negatives, 6,609 True Negatives, 85 False Positives)

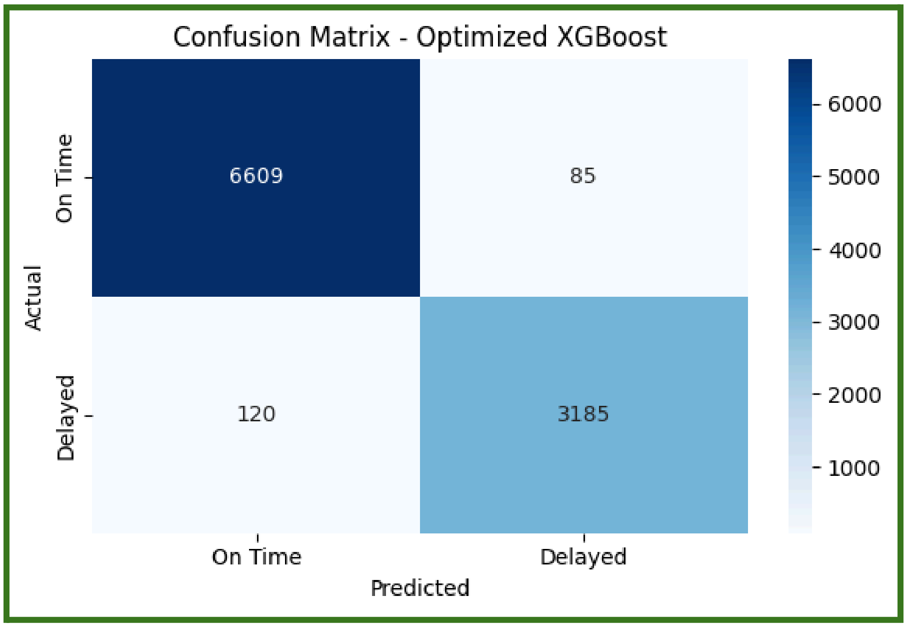

Fig 5 - XGBoost - Bayesian Tuning Confusion Matrix

Explainability: SHAP

(Fig. 6: SHAP summary plot)

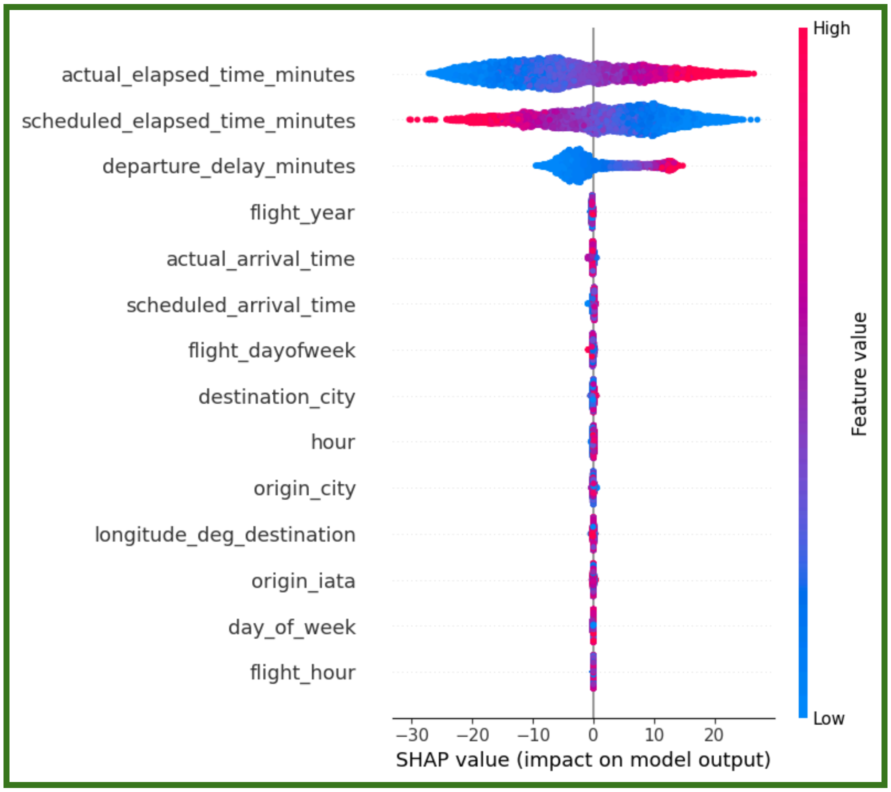

Fig 6 - XGBoost - Bayesian Tuning SHAP

Key predictive drivers:

- actual_elapsed_time_minutes

- departure_delay_minutes

- scheduled_elapsed_time_minutes

- flight_hour, flight_dayofweek

Calibration & Probability Quality

- Brier Score (Base): 0.0175

- Platt Scaling: 0.0156 → selected as final calibration method
  (Fig. 8–9: Calibration curves & ROC)

6. Conclusion

This project successfully developed a high-performing machine learning pipeline to predict flight delays using historical schedule, geographic, and temporal data. Through systematic preprocessing, exploratory analysis, and model experimentation, the pipeline evolved from a simple Logistic Regression baseline to an optimized XGBoost classifier with advanced tuning, feature engineering, and interpretability.

The final model, trained using Bayesian optimization and SMOTE, achieved a test accuracy of 97.95%, with an F1 score of 0.97 for the delayed class. This performance represents a substantial improvement over all baseline models, particularly in balancing recall and precision—critical for minimizing both false negatives and false alarms in an operational setting.

Several key practices contributed to the success of the project:

- Merging flight records with geospatial metadata using IATA codes.

- Engineering predictive features such as flight hour, delay categories, and haversine distance.

- Addressing class imbalance through both SMOTE and XGBoost’s native weighting mechanism.

- Validating model calibration and generalization through ROC, SHAP, and Brier analysis.

In practical terms, this model offers a reliable, interpretable, and scalable solution for real-time flight delay prediction. It is suitable for integration into operational dashboards, customer alert systems, or resource allocation workflows.