Project Report

Ma412 – Mathematical Tools for Data Science and Decision-Making

Audit, Correction and Robustness Analysis of a Flight Delay Prediction Model

Student:
Simon Lignac

Instructor:
Leila Gharsalli

Promotion 2027 – Aero 4
June 2026
Academic Year 2025–2026

Table of Contents

Part I – Executive Summary

Overview

This project audits, corrects, and extends a faulty Machine Learning pipeline designed to predict flight delays for an airline company. The dataset contains 300 flights described by 13 features (carrier, origin, destination, weather, congestion, previous delay, etc.) with a binary target `Delay` $\in \{0, 1\}$. The class distribution is nearly balanced: 149 flights are delayed (49.67%) and 151 are on time (50.33%), eliminating the need for resampling techniques.

The starter notebook contained 7 methodological flaws, two of them critical (inverted train/test split ratio, evaluation on the training set), which made all reported metrics meaningless. After correction, two models were trained and evaluated on an honest 60-sample test set: a Logistic Regression (LR) and a Random Forest (RF). The RF was selected as the best model for its superior recall (0.7037 vs. 0.6296), which is the critical metric when missed delays are operationally more costly than false alarms.

Key figures at a glance:

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression (corrected)	0.6833	0.6538	0.6296	0.6415
Random Forest	0.6500	0.5938	0.7037	0.6441

The model is moderately robust across preprocessing variants but shows high instability across random splits (F1 std = 0.073 over 10 seeds), a direct consequence of the small dataset size. An operational recommendation is proposed in Part IV.

Part II – Critical Audit and Model Correction

Issues Identified

Seven issues were identified in the starter notebook, classified below by severity. Each flaw was corrected in the pipeline described in the following section.

#	Severity	Issue
1	Critical	`test_size=0.80`: model trained on only 60 samples (20%), tested on 240. Severely undertrained; metrics unreliable.
2	Critical	Evaluation on `X_train`/`y_train`: the reported 93.3% accuracy measures memorisation, not generalisation.
3	Major	`Flight_ID` kept as a feature: a unique identifier that generates 300 spurious binary columns after encoding.
4	Major	`fillna(0)` applied uniformly: for `Prev_Delay_Min`, 0 is a valid value; for `Weather`, it creates a non-existent category.
5	Major	`drop_first=False`: all dummy categories kept, causing perfect multicollinearity (dummy variable trap).
6	Major	No `StandardScaler`: features on very different scales (`Distance_km` up to 4000, `Weekend` $\in \{0, 1\}$) prevent lbfgs from converging, directly causing the `ConvergenceWarning` in the output.
7	Minor	`max_iter=100` too low: a consequence of Issue 6; the optimiser stops before reaching its optimum.

Corrected Pipeline and Results

The corrected Logistic Regression pipeline applies the following steps in order: (1) drop `Flight_ID`; (2) 80/20 train/test split *before* any imputation (prevents data leakage); (3) median imputation for `Prev_Delay_Min` and `Load_Factor_pct`, mode imputation for `Weather` (both fitted on train only); (4) `get_dummies(drop_first=True)`; (5) `StandardScaler` (fitted on train only); (6) `LogisticRegression(max_iter=1000)`.

Evaluated on the test set (60 samples):

	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.6833	0.6538	0.6296	0.6415
Confusion matrix:	TN = 24	FP = 9	FN = 10	TP = 17

A naive baseline (always predict "on time") would yield 50.33% accuracy; the corrected LR achieves 68.33%, a gain of +18 percentage points. However, 10 true delays were missed (false negatives), which motivates testing a model with higher recall.

Part III – Model Improvement and Analysis

Additional Method: Random Forest

A Random Forest (100 trees, `random_state=42`) was selected as the second model for four reasons: (i) it captures non-linear interactions between features (weather × congestion × previous delay) that Logistic Regression cannot model with a single linear boundary; (ii) it is insensitive to feature scale, removing the need for `StandardScaler`; (iii) it provides built-in feature importance; (iv) it has no iterative optimiser that can fail to converge.

Test set results and comparison:

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.6833	0.6538	0.6296	0.6415
Random Forest	0.6500	0.5938	0.7037	0.6441

	TN	FP	FN	TP
Logistic Regression	24	9	10	17
Random Forest	20	13	8	19

The RF catches 2 additional true delays (8 FN vs. 10 FN) at the cost of 4 extra false alarms. Given that false negatives are more operationally costly (see Section D below), the Random Forest is the recommended model.

Per-class analysis confirms this trade-off: the Random Forest achieves an F1-score of 0.66 on the "on time" class versus 0.64 on the "delayed" class, meaning it slightly under-predicts delays. This asymmetry reinforces the case for lowering the decision threshold below 0.5 in an operational context.

Top 3 most influential features (Mean Decrease in Impurity):

1. `Gate_Congestion_Index`: 0.1505 (15.05%)
2. `Prev_Delay_Min`: 0.1102 (11.02%)
3. `Load_Factor_pct`: 0.1012 (10.12%)

Ground-side congestion is the strongest predictor, confirming that operational bottlenecks (gate conflicts, turnaround pressure) drive delays more than distance or carrier identity.

Error Analysis

Of 60 test flights, 21 were misclassified (13 false positives, 8 false negatives). Three concrete examples:

Example 1: False Negative (FL1152, BlueWing, NCE → MAD): Weather = Clear, `Prev_Delay` = 47.0 min, `Congestion` = 43.8, `Turnaround` = 98.0 min, `Crew change` = 1, `Load` = 100%. Predicted on time, actually delayed. The model assigned $P(\text{delayed}) = 0.39$: no single feature was extreme enough to cross the 0.5 threshold, yet the combination of a saturated aircraft (100% load), a crew change, and a prior 47-min delay was sufficient to cause a real delay. The linear combination of signals was under-weighted.

Example 2: False Positive (FL1009, AeroNord, TLS → BCN): Weather = Fog, `Prev_Delay` = 0.0 min, `Congestion` = 70.5, `Turnaround` = 48.5 min, `Load` = 52.4%. Predicted delayed, actually on time. $P(\text{delayed}) = 0.52$: Fog + high congestion pushed the model over the threshold, but the absence of a previous delay and a light load factor allowed the flight to depart on time.

Example 3: False Positive (FL1226, BlueWing, MRS → AMS): Weather = Windy, `Prev_Delay` = 5.0 min, `Congestion` = 56.4, `Turnaround` = 74.9 min, `Crew change` = 1, `Load` = 69.7%. Predicted delayed, actually on time. $P(\text{delayed}) = 0.54$: multiple moderate risk factors coincided without producing an actual delay.

Which error is more costly? False Negatives. A missed delay on a full flight triggers EU Regulation 261/2004 compensation (up to 600 EUR per passenger), crew overtime, cascading delays on subsequent rotations, and reputational damage, costs that are unbounded and difficult to recover. A false alarm costs only a precautionary gate reallocation or crew standby, a bounded and reversible expense of a few hundred euros.

Part IV – Robustness and Decision-Making

Robustness Results

Three robustness tests were conducted on the Random Forest model.

1. Random seed variation (10 seeds: 0, 1, 7, 21, 42, 99, 123, 256, 512, 999):

	Mean	Std	Min	Max
F1-score	0.5736	0.0731	0.4528	0.6567
Accuracy	-	-	0.5167	0.6500

The standard deviation of 0.073 is high: the model's F1-score ranges from 0.45 to 0.66 depending solely on the random split. This instability is a direct consequence of the small dataset (only 60 test samples). A single atypical partition can move metrics by more than 20 percentage points. These estimates should therefore be interpreted as *rough bounds*, not reliable point estimates.

2. Feature removal (random_state = 42):

Configuration	F1-score	Δ vs baseline
All features (baseline)	0.6441	-
Without `Weather`	0.5806	-0.0635
Without `Gate_Congestion_Index`	0.5965	-0.0476
Without `Turnaround_Time_Min`	0.6000	-0.0441
Without `Crew_Change`	0.5862	-0.0579
Without `Prev_Delay_Min`	0.6102	-0.0339
Without top 2 features	0.6333	-0.0108

Removing `Weather` causes the largest single-feature F1 drop (-0.064), followed by `Crew_Change` (-0.058). Notably, removing both top features simultaneously causes only a -0.011 drop, suggesting that remaining features partially compensate. The model relies on a *combination* of signals rather than a single dominant predictor.

3. Preprocessing variation (random_state = 42): Switching from median to mean imputation changes F1 from 0.6441 to 0.6207 (-0.023). With only 15 missing values out of 300 (5%), the imputation strategy has limited influence. The model is robust to this choice.

What-if: 20A simulation replacing 20% of non-Fog test flights with Fog left the predicted delay count unchanged: 32/60 (53.3%) in both cases. This counter-intuitive result reveals that `Weather` alone does not deterministically push predictions across the threshold, the model requires corroborating signals (congestion, previous delay). However, if Fog becomes structurally more frequent due to climate or seasonal shift, the model faces covariate shift: the training distribution no longer matches deployment conditions, and Fog-related patterns learned on few examples may become unreliable. Periodic retraining is required.

Operational Recommendation

Context: The airline operates under budget constraints and cannot monitor every flight with equal intensity.

Recommendation: Deploy the Random Forest with a decision threshold lowered to 0.40 (instead of 0.50). At this threshold, the model flags more flights as potentially delayed, trading some additional false alarms for higher recall, a justified exchange since false negatives cost up to 600 EUR $\times N_{\text{pax}}$ in mandatory compensation, versus a few hundred euros for an unnecessary precaution.

Prioritisation rule: Concentrate preventive resources on flights where:

- `Gate_Congestion_Index` > 60 (top feature, importance 15.05%), *and*
- `Prev_Delay_Min` > 20 min (importance 11.02%), *or*
- `Load_Factor_pct` $> 90\%$ with a crew change scheduled.

These thresholds correspond to the feature ranges most associated with delayed flights in the training set and are directly observable by ground operations staff.

Limitations and monitoring: The model's F1-score on any given 60-sample test set can range from 0.45 to 0.66 due to sampling variance. The current 300-flight dataset is insufficient for production deployment. Before operationalisation, the airline should: (1) collect at least 2,000 flights to stabilise estimates; (2) retrain quarterly to track seasonal covariate shift (especially weather patterns); (3) monitor the false negative rate as the primary KPI, with an alert if it exceeds 20% over any rolling 4-week window.