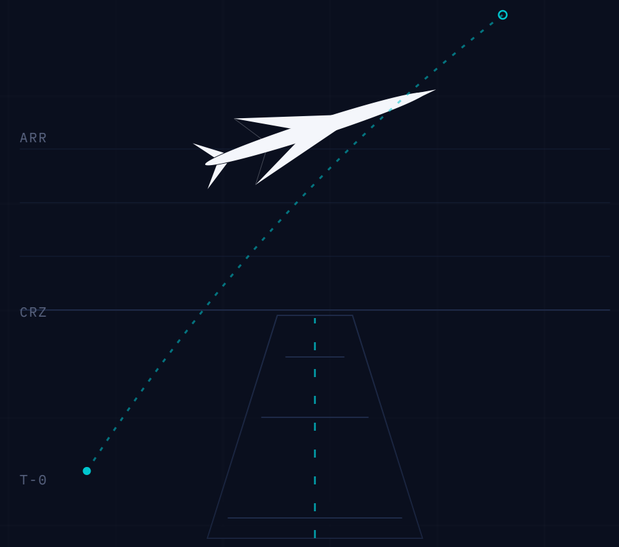
ML PROJECT · END-TERM PRESENTATION

Advanced Predictive Modeling for Flight Delay Mitigation

Predicting Delay Recovery in U.S. Commercial Aviation

FIG · 01 — DEP TRAJECTORY



COORDINATES

N 41°58′ · W 087°54′ → N 33°56′ · W 118°24′

CHAPTER OPEN

01

Problem Statement

Why delay recovery — not just delay prediction — matters.

The Unasked Question in Aviation AI

Most predictive models in aviation ask:
“Will this flight be delayed?”

We ask:
“If a flight departs late, will it recover and arrive on time?”

Flight delays propagate across entire networks — a single delay can cascade through dozens of connected flights. Yet no existing operational tool quantifies a flight's **recovery capability** before it departs.

GOAL

Build a model that predicts **Recovery_Minutes** — the exact minutes of delay a flight will gain back in the air — and outputs a **Resilience Score** for airline dispatchers.

~20%

of all U.S. flights depart late each year

1 → 4

Cascade effect: a single delay can trigger up to four downstream disruptions

+4.5 min

average recovery recouped in-air across our 2024 dataset

From Reactive to Predictive Operations

CARD 01 · OPERATIONS

Airline Operations

Without a recovery score, dispatchers react *after* delays spiral. A pre-departure resilience signal enables proactive gate management and crew reallocation.

CARD 02 · EXPERIENCE

Passenger Experience

Passengers miss connections partly because airlines can't identify unrecoverable delays early. Accurate recovery prediction enables timely passenger re-booking.

CARD 03 · SCALE

Industry Scale

The FAA estimates flight delays cost the U.S. economy **~$33B per year**. Even marginal improvements in recovery prediction reduce systemic costs.

BOTTOM LINE *Our solution turns raw operational data into a deployable* **Resilience Score** *— a first-of-its-kind metric for delay recovery.*

CHAPTER OPEN

02

Literature Survey

What exists, what's missing, and how we go further.

REF · ZHOU, DEC 2025

Integrating Delay-Absorption Capability

METHODOLOGY

Two-stage ML pipeline computing an **“AbsorbScore”** — the probability a flight recovers from an upstream delay. Uses **CatBoost** (Stage 1) + **XGBoost** (Stage 2). Data from BTS + NOAA.

LIMITATION

Trained on a narrow 3-month window (Summer 2023 only) — entirely missing seasonal variation and post-pandemic operational shifts.

GAP WE ADDRESS

Manual weather-to-flight data merging introduced errors at scale. Our dataset (**Aeolus**) pre-matches these sources with zero manual joining.

FIG · 02 — PIPELINE SKETCH

REF · ALBASSAM ET AL.

SMOTE + Random Forest for Delay Prediction

METHODOLOGY

Applied **Synthetic Minority Oversampling Technique (SMOTE)** with Random Forest to address the 80/20 class imbalance between on-time and delayed flights.

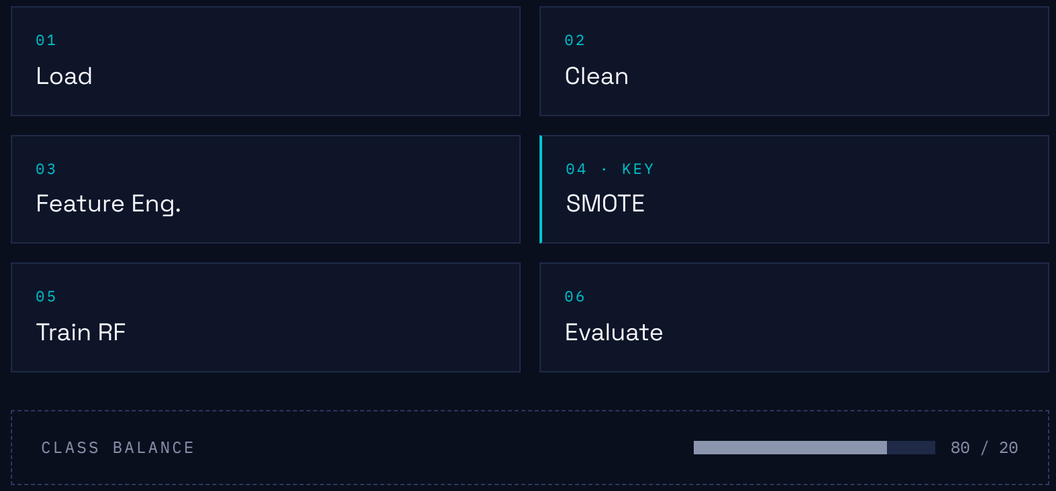
LIMITATION

Synthetic data generation distorts the physical realities of aviation routing — fabricated samples don't reflect true weather-flight interactions.

GAP WE ADDRESS

SMOTE creates exponential RAM overhead on multi-year national datasets. We avoid synthetic generation entirely by using a real, full-year 2024 corpus with **54.6M authentic samples**.

FIG · 03 — ARCHITECTURE BLOCKS



REF · DISPATCHER3 / DE FALCO, 2025

Probabilistic Flight Process Prediction

METHODOLOGY

Monte Carlo-style probabilistic distributions of turnaround times. Combines a regression model with error discretization and a probabilistic classifier to output **uncertainty-aware block time predictions.**

$P(t \mid x) \rightarrow \int \text{samples} \rightarrow \text{distribution}$

LIMITATION

Severe computational overhead at inference — unusable for real-time tactical dashboards. Relies on **proprietary, closed-source European consortium data.**

INFERENCE ~seconds / flight

GAP WE ADDRESS

Our XGBoost pipeline runs at **milliseconds-per-prediction** inference speed. Built on publicly available U.S. BTS data — **fully reproducible**.

INFERENCE < 50 ms / flight

TRADE-OFF

Probabilistic richness vs. operational latency — our model prioritizes deployability.

Where Existing Work Falls Short — And Where We Stand

BENCHMARK	SAMPLES	NO DATA LEAK	NO SYNTHETIC	TIME-SPLIT	FLIGHT-DELAY
[13]	16,679	✗	✗	✗	✗
Tabrizia	3,087	✗	✗	✗	✗
WildTab	54,943	✗	✗	✗	✗
TableShift	840,582	✓	✓	✗	✗
TabRed	7,163,150	✓	✓	✓	✗
Aeolus (Ours)	**54,674,003**	✓	✓	✓	✓

KEY TAKEAWAY

Aeolus is the **only** benchmark satisfying all four integrity criteria *and* covering the flight-delay domain at full national scale.

4 / 4 ✓

CHAPTER OPEN

03

Dataset & Feature Preprocessing

6.28M flights. 34 raw features. A rigorous pipeline.

Our Data Foundation — Aeolus 2024

DATASET PROFILE

SOURCE	Aeolus (Xu et al., Sichuan University / HKUST)
SUBSET	Full-year **2024** U.S. domestic flights
RAW SIZE	6,284,841 rows (post cancelled + diverted removal)
FEATURES	34 initial columns spanning flight ops + weather
ORIGIN	U.S. **BTS** flight records, 1-to-1 matched with **Meteostat** weather observations at origin + destination airports.
ETHICS	Public government + meteorological data. No PII. No proprietary records. **Fully reproducible.**

FEATURE CATEGORY BREAKDOWN

Flight Operations — 10

DEP_DELAY · ARR_DELAY · CRS_ELAPSED · ACT_ELAPSED · TAXI_OUT · TAXI_IN · AIR_TIME …

Temporal — 4

FL_DATE · MONTH · DAY_OF_MONTH · DAY_OF_WEEK

Route — 4

ORIGIN · DEST · OP_CARRIER · OP_CARRIER_FL_NUM

Weather · Origin — 3

O_TEMP · O_PRCP · O_WSPD

Weather · Destination — 3

D_TEMP · D_PRCP · D_WSPD

Geospatial — 4

O_LAT · O_LON · D_LAT · D_LON

Data Quality Audit — Principled Cleaning

MISSING VALUE COUNTS

COLUMN	MISSING ROWS	% OF DATASET
O_TEMP	513	0.008%
O_PRCP	513	0.008%
O_WSPD	513	0.008%
D_TEMP	750	0.012%
D_PRCP	750	0.012%
D_WSPD	750	0.012%

TOTAL AFFECTED

Flights with *any* missing weather data: **1,263** — only **0.02%** of the full dataset.

DECISION REASONING

Since the missing fraction is below our **5% threshold**, we applied **listwise deletion** (dropped affected rows).

Imputation (e.g. regression) was *deliberately avoided* — fabricated weather values would corrupt the physical truth of the prediction.

RESULT

6,283,578

clean rows retained for modeling

ALSO DROPPED

All `CANCELLED=1` and `DIVERTED=1` flights — they do not represent recoverable delay scenarios.

Engineering Features That Reflect Real Aviation Physics

CARD 01 · TARGET

Recovery Target

```
Recovery_Minutes = DEP_DELAY − ARR_DELAY
```

Positive value = delay was recovered in-air. This is our **prediction target**. Average across delayed flights: **+4.5 minutes.**

CARD 02 · BUFFER RATIO

Padding Ratio

```
Padding_Ratio = CRS_Elapsed_Time / Actual_Elapsed_Time
```

Captures the hidden schedule buffer airlines build in. **Ratio > 1** = airline padded extra time for in-air recovery. *A physics-based feature.*

CARD 03 · BUFFER RAW

Available Buffer

```
Available_Buffer = CRS_Elapsed_Time − Actual_Elapsed_Time
```

Raw minutes of slack time. Together with Padding Ratio, gives the model **two complementary views** of recovery capacity.

CARD 04 · TEMPORAL

Cyclical Time Encoding

```
Sch_Dep_Hour_Sin = sin(2π × hour / 24) Sch_Dep_Hour_Cos = cos(2π × hour / 24)
```

Preserves time's cyclic nature: 23:00 and 01:00 are neighbors. Integer hours would make them look far apart.

DISTRIBUTION *Average in-air recovery = +4.5 min. Right-skewed: most flights recover modest delays, few recover large ones.*

Converting Categorical Data Without Exploding Dimensionality

THE PROBLEM

300+ airports + **15+ airlines**. One-hot encoding would add 300+ binary columns, dramatically increasing memory and noise.

SOLUTION · TARGET ENCODING

Each airline/airport is replaced with its **historical mean Recovery_Minutes**, computed from the **training set only** (prevents leakage).

01 Group training flights by carrier (e.g. “UA”)

02 Compute mean Recovery_Minutes per group

03 Replace the string label with that number in all splits

04 Model receives a **meaningful recovery-performance score**, not an arbitrary ID

CARRIER ENCODING VALUES — MIN RECOVERED

15 Features. Zero Leakage. Temporal Split.

FINAL FEATURE SET · 15

Sch_Dep_Hour_Sin	Sch_Dep_Hour_Cos	OP_CARRIER*	ORIGIN*	DEST*
CRS_ELAPSED_TIME	Available_Buffer	Padding_Ratio	DEP_DELAY	O_TEMP
O_PRCP	O_WSPD	D_TEMP	D_PRCP	D_WSPD

* TARGET-ENCODED · ◆ ENGINEERED

FEATURES REMOVED

Actual arrival/departure times (post-takeoff leakage); flight number, tail number (would memorise individual flights, not physics).

3-WAY CHRONOLOGICAL SPLIT · DAY OF MONTH

TRAIN	Days 01 – 18 · **~60%**	748,716
VAL	Days 19 – 24 · **~20%**	271,907
TEST	Days 25 – 31 · **~20%**	272,862

Strict temporal split prevents data leakage. The model never sees "future" flights during training — matching real-world forecasting conditions.

CHAPTER OPEN

04

ML Methodology

A two-stage architecture purpose-built for delay recovery.

Two-Brains XGBoost — A Sequential Hurdle Model

WHY THIS DESIGN

Brain 1 acts as a gatekeeper — filters out flights physically incapable of recovery (no buffer time, severe weather, ultra-short route). Prevents Brain 2 from wasting capacity on impossible cases.

Brain 2 focuses entirely on magnitude of recovery for feasible flights — a cleaner, more focused regression task.

This outperforms a single end-to-end regressor because the two sub-problems have *fundamentally different statistical structures*.

HURDLE MODEL

A statistical pattern: split a hard problem into a binary “does it happen” step and a magnitude step.

Two-Brains Architecture — Powered by XGBoost

The system uses a **sequential modeling approach** — two specialized XGBoost models, each solving a sub-problem the other can't.

BRAIN 1 · CLASSIFIER

The Gatekeeper

Identifies flights that are **mathematically unlikely to recover any delay** and filters them out before they reach Brain 2.

BRAIN 2 · REGRESSOR

The Predictor

For flights that pass the first stage, predicts **exactly how many minutes** of delay will be recovered during the journey.

WHY XGBOOST FOR BOTH STAGES

CARD 01

Gradient Boosting on Tabular Data

XGBoost captures the **complex, non-linear interactions** that drive delay recovery — route length, congestion, scheduling buffers, and weather all interact in ways linear models can't reach.

CARD 02

Robust to Overlapping Features

Aviation datasets contain many correlated variables describing similar operational conditions (Available_Buffer ↔ Padding_Ratio). XGBoost handles this **without manual orthogonalization**.

CARD 03

No Dimensionality Reduction

We deliberately avoided PCA. PCA converts features into abstract components that are hard to interpret. We keep the model **transparent and explainable** — so dispatchers see exactly which operational factors drive each prediction.

Three Architectures Benchmarked — Consistent Winner

BRAIN 1 · GATEKEEPER (CLASSIFICATION)

MODEL	ACCURACY	PRECISION	RECALL	F1
XGBoost	**71.36%**	**73.58%**	**92.98%**	**0.821**
LightGBM	71.17%	72.89%	94.47%	0.823
MLP Neural Net	70.35%	73.12%	91.99%	0.815

BRAIN 2 · PREDICTOR (REGRESSION)

MODEL	MAE (MIN)	RMSE (MIN)
XGBoost	**10.88**	**17.12**
LightGBM	10.89	17.33
MLP Neural Net	10.91	17.33

CALLOUT · BALANCED PERFORMANCE

XGBoost wins on accuracy, F1, and regression error simultaneously — a **consistent winner across both stages**, not a metric-specific outlier.

Bayesian Optimization with Optuna — Pushing the Ceiling

SEARCH SPACE · 20 TRIALS / MODEL

Bayesian search over 20 trials per model. Parameters searched:

n_estimators	500 – 2000
learning_rate	0.005 – 0.05
max_depth	4 – 11
subsample	0.6 – 1.0
colsample_bytree	0.6 – 1.0
scale_pos_weight (B1)	0.5 – 2.0

Optuna uses Tree-structured Parzen Estimator (TPE) — a smarter search than grid or random, finding near-optimal hyperparameters efficiently.

BRAIN 1 · BEST RESULT · TEST SET

71.36% accuracy

Post-Optuna accuracy on the held-out **test set**.

n_estimators=1106 · lr=0.0218 · max_depth=10

BRAIN 2 · BEST RESULT · TEST SET

10.88 min MAE

Post-Optuna MAE on the held-out **test set**.

n_estimators=1894 · lr=0.0119 · max_depth=10

Validation set drove the hyperparameter search; **test-set performance is what we report** *— the only number that reflects real generalization.*

Opening the Black Box — SHAP Feature Attribution

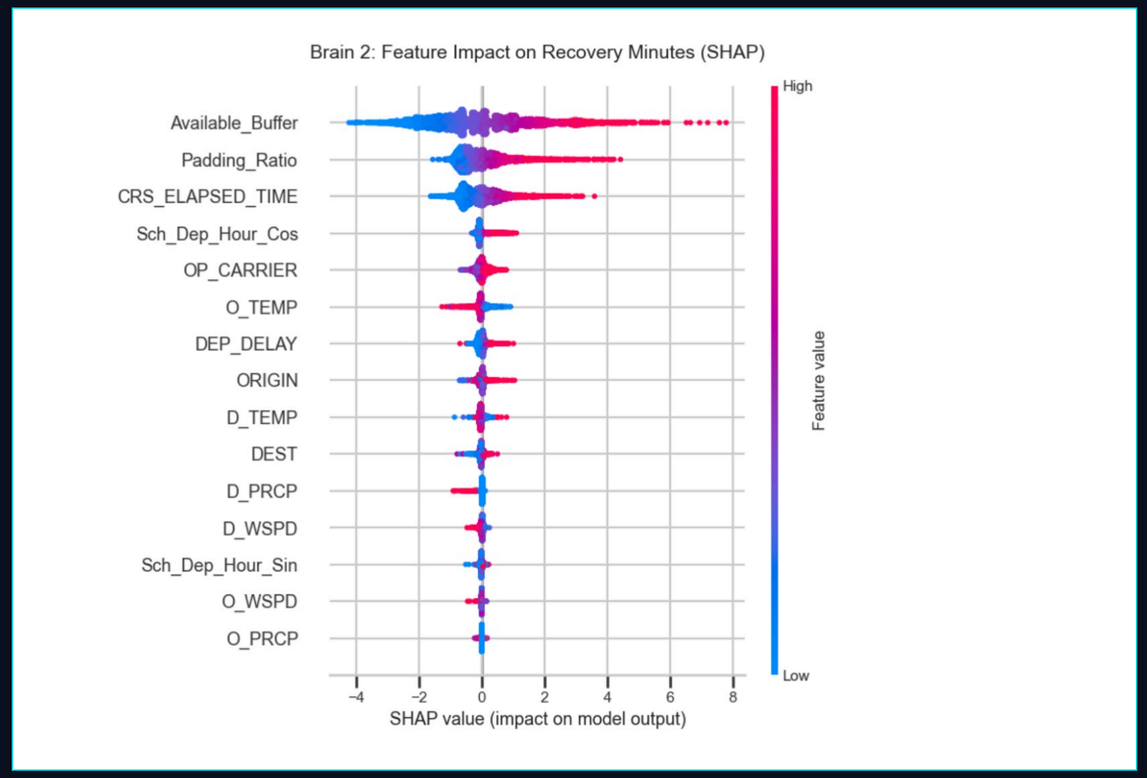
WHAT SHAP DOES

SHAP (SHapley Additive exPlanations) assigns each feature a contribution value for every individual prediction. Unlike feature importances, SHAP values are:

DIRECTIONAL	Does feature X push recovery up or down?
CONSISTENT	Larger impact = larger SHAP value
LOCAL	Per-flight explainability, not just global averages

> **EXAMPLE EXPLANATION**
>
> *A dispatcher sees exactly why Flight AA1234 scored low: its Available_Buffer was 12 min but DEP_DELAY was 47 min —* **physically unrecoverable.**

SHAP SUMMARY · BRAIN 2 · TEST SET



FEATURE VALUE — HIGH — LOW

What Almost Broke the Model — and How We Fixed It

CHALLENGE	#	OUR ENGINEERING FIX
Memory Limits	01	Filtered to **1.29M delayed flights** to ensure the pipeline stays computable on a local laptop — no cluster required.
Time Logic	02	Encoded the 24-hour clock with **sine / cosine waves** so 23:00 and 01:00 sit next to each other on a continuous circle, not at opposite ends of a number line.
Target Leakage	03	Purged all post-takeoff fields and enforced a strict **chronological split** — Days 1–18 train, Days 25–31 test. Model never sees the future during training.
Majority Bias	04	Tuned the decision threshold to **0.55** instead of the default 0.5 — prioritizing **Precision (73.58%)** over a classifier that lazily predicts the majority class.
Overfitting	05	Applied **early stopping** on the validation set and capped tree depth `(max_depth)` at **10** for Brain 1 and **8** for Brain 2.

CHAPTER OPEN

05

Performance Metrics & Deployability

How good is “good enough” — and can this ship?

Brain 1 — Gatekeeper Performance Analysis

WHY PRECISION IS THE NORTH STAR

FALSE POSITIVE — COSTLY

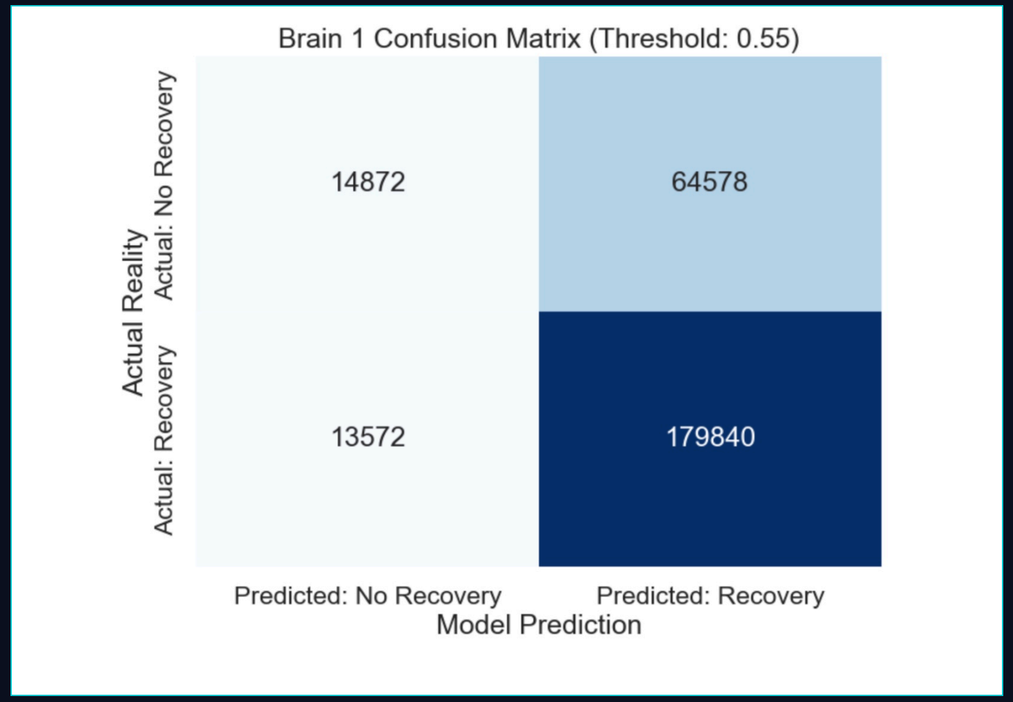
Predicting *“will recover”* when the flight actually **won't**. Dispatchers trust the signal and skip rebooking — then passengers miss connections anyway. **Damages passenger trust.**

FALSE NEGATIVE — LESS HARMFUL

Predicting *“won't recover”* when it actually does. Dispatchers act conservatively — the flight still arrives on time, just with extra precaution.

Therefore: **maximizing Precision is the priority.** Our XGBoost Brain 1 achieves **Precision = 73.58%** — when the model says “recovery,” it's right roughly three out of every four times.

CONFUSION MATRIX · BRAIN 1 · THRESHOLD 0.55



TRUE POS	179,840	TRUE NEG	14,872
FALSE POS	64,578	FALSE NEG	13,572

Brain 2 — Predictor Accuracy in Real-World Terms

10.88

MAE minutes · test set · Optuna-tuned XGBoost

17.12

RMSE minutes · test set

+4.5

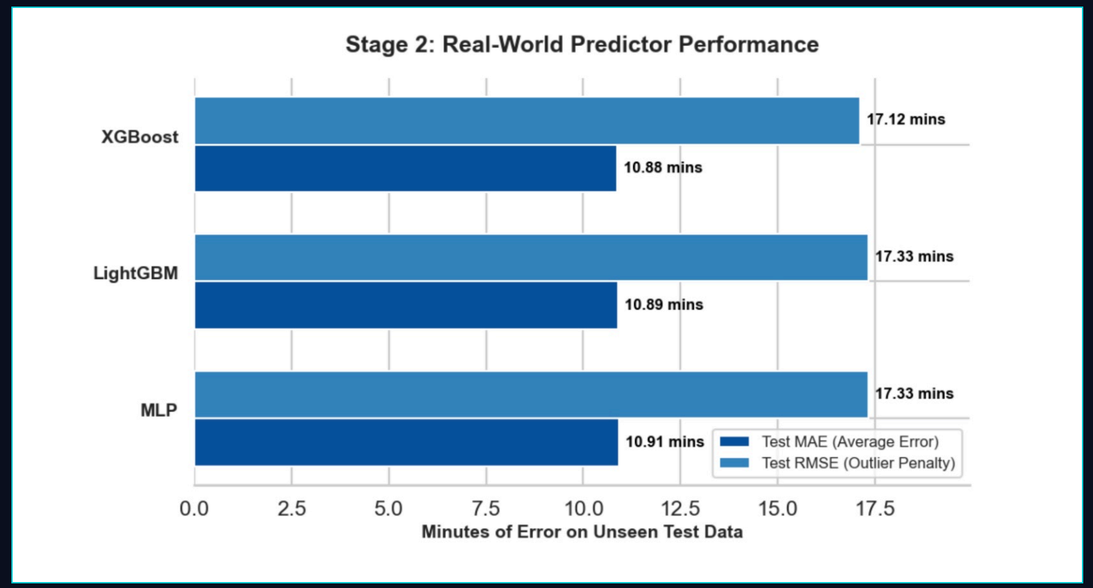
Average recovery the model is forecasting against (minutes)

INTERPRETING THESE NUMBERS

An MAE of **10.88 minutes** means: on average, the model's prediction of *how much delay a flight will recover* is off by ~11 minutes. In the context of delay recovery (most recoveries land in the 5–20 minute range), this is operationally useful.

The validation set drove hyperparameter search; **10.88 min is the held-out test-set MAE** — the figure that actually reflects how the model will behave on flights it has never seen.

STAGE 2 · REAL-WORLD PREDICTOR PERFORMANCE



HELD-OUT TEST SET

XGBOOST WINS BOTH METRICS

Can This Be Deployed — at Plaksha, or in Real Operations?

PANEL 01

Minimum Viable Deployment

Requires only **9 pre-departure data points**: departure delay, scheduled elapsed time, origin/destination codes, carrier, weather at both airports, and scheduled departure hour. **All available before pushback.**

INFERENCE < 50 ms / flight (CPU)

PANEL 02

Plaksha Deployment Scenario

Deployable as a web service (Flask / FastAPI) receiving live BTS-format feeds, returning a **Resilience Score dashboard**.

01 BTS data connector

02 Meteostat weather API call

03 Pre-trained model binary

SETUP EFFORT · 1 – 2 days engineering

PANEL 03

Scaling Challenges

CONCEPT DRIFT
Schedules + delay patterns shift. Periodic retraining (monthly / seasonal).

WEATHER API
Real-time weather must fetch reliably per flight.

CLASS IMBALANCE
On-time vastly outnumbers recoverable delayed flights — threshold calibration matters.

NOVEL AIRPORTS / AIRLINES
Target encoding for new carriers defaults to global mean until data accumulates.

What We Built — End-to-End Summary

PROBLEM

Predicting in-air delay recovery for departing-late flights.

DATA

6.28M real U.S. domestic flights (2024) · Aeolus · 15 physics-grounded features · **zero synthetic** · temporal split.

MODEL

Two-Brains XGBoost · Brain 1 (Gatekeeper) screens impossible recoveries · Brain 2 (Predictor) forecasts exact minutes recovered.

NOVELTY

First model to use a **hurdle architecture** on the Aeolus dataset for recovery prediction. SHAP-transparent. **Deployable at pre-departure.**

KEY RESULTS

METRIC	VALUE
Brain 1 Accuracy · test	**71.36%**
Brain 1 Precision (priority)	**73.58%**
Brain 1 F1 Score	0.821
Brain 2 MAE · test (Optuna)	**10.88 min**
Brain 2 RMSE · test	17.12 min
Dataset size	6,283,578

END-OF-DECK · QUESTIONS

Thank You

PRESENTED BY

Kartik Kaushik · Saksham Bhasin · Vedant Kapoor

Dataset: U.S. BTS + Meteostat (2024) · Models: XGBoost · LightGBM · MLP · Tools: Python · pandas · scikit-learn · SHAP · Optuna

REFERENCES

[01] · ZHOU · 2025

Integrating Delay-Absorption Capability.

arXiv:2512.08197

[02] · ALBASSAM ET AL.

Flight Delay Prediction Using ML and SMOTE.

PMC12685205

[03] · DE FALCO / DISPATCHER3 · 2025

Probabilistic Flight Process Prediction.

ScienceDirect · S0957417425029288

[04] · XU ET AL. · 2025

Aeolus: A Multi-structural Flight Delay Dataset.

arXiv:2510.26616