

EDITORIAL

# Aircraft Fuel Burn Estimation: The EUROCONTROL PRC 2025 Data Challenge

Junzi Sun <sup>\*,1,2</sup> Enrico Spinielli <sup>3</sup> and Martin Strohmeier <sup>4,2</sup>

<sup>1</sup>Faculty of Aerospace Engineering, Delft University of Technology, The Netherlands

<sup>2</sup>OpenSky Network, Switzerland

<sup>3</sup>EUROCONTROL, Brussels, Belgium

<sup>4</sup>Cyber-Defence Campus, armasuisse Science and Technology, Switzerland

\*Corresponding author: [j.sun-1@tudelft.nl](mailto:j.sun-1@tudelft.nl)

(Published: 25 March 2026)

## Abstract

The EUROCONTROL Performance Review Commission launched a data challenge in 2025 for machine-learning-based fuel burn prediction, in collaboration with OpenSky Network and TU Delft. This paper describes the dataset built for the challenge. It pairs ACARS fuel telemetry, crowdsourced through airframes.io, with ADS-B trajectory data from the OpenSky Network, augmented with flight list information from EUROCONTROL over the period April to October 2025. Consumer-grade ACARS receivers supply fuel-on-board reports at irregular intervals; ADS-B provides dense kinematic trajectories at sub-second resolution. We fuse these two sources and validate fuel labels against physics-based predictions from TU Delft's OpenAP model to infer ambiguous reporting units and filter erroneous records. The resulting training set is approximately 5 GB and includes real-world noise, coverage gaps, and operational variability. We describe the data collection pipeline, the unit inference methodology, quality assurance procedures, and the structure of the released dataset.

**Keywords:** fuel burn estimation; ACARS; ADS-B; open aviation data; machine learning benchmark

**Abbreviations:** ACARS: Aircraft Communications Addressing and Reporting System, ADS-B: Automatic Dependent Surveillance–Broadcast, FOB: Fuel On Board, PRC: Performance Review Commission, RMSE: Root Mean Square Error, ATM: Air Traffic Management

## 1. Introduction

Aviation accounts for roughly 2–3% of global CO<sub>2</sub> emissions [1], and the share continues to grow. Accurate fuel burn data matters for emissions inventories, trajectory optimization, and performance benchmarking, yet fuel consumption records have long been proprietary. Airlines rarely share them. Researchers who need such data must either negotiate restricted-access agreements or rely on physics-based estimates, which carry their own assumptions.

The EUROCONTROL Performance Review Commission (PRC), established in 1998 to provide independent analysis of European ATM performance [2], has run a series of open data challenges to address this gap. The 2024 edition targeted takeoff weight prediction [3]. The 2025 challenge asked participants to predict fuel consumed during flight intervals, given flight list information, trajectory and aircraft type data. It ran in two phases from October through November 2025. Evaluation was based on RMSE and no aviation domain knowledge was required in order to engage with the broader

data science community.

This paper documents the dataset created for the 2025 challenge. Our main contributions are:

1. A multi-source data fusion methodology that pairs sparse ACARS fuel telemetry with dense ADS-B trajectories.
2. A physics-based unit inference procedure that resolves ambiguous fuel reporting units using OpenAP [4] as a reference model.
3. A publicly released dataset of paired trajectory–fuel records covering seven months of commercial operations, suitable for the competition, academic teaching and ongoing research.
4. A fully open processing pipeline, from raw data ingestion to final dataset generation.

Section 2 reviews the data sources. Section 3 describes the collection pipeline, fuel unit inference, and quality assurance. Section 4 presents the dataset. Section 5 discusses applications and limitations.

## 2. Background and data sources

### 2.1 ACARS

The Aircraft Communications Addressing and Reporting System is a two-way digital datalink between aircraft and ground stations. Aircraft periodically transmit position reports, operational parameters, and, most relevant to this work, fuel state information. Access to ACARS data has historically required commercial subscriptions, but community platforms such as *airframes.io* [5] now crowdsource ACARS reception using consumer-grade radio equipment. Messages arrive in text format with airline-specific structures and variable reporting intervals (typically 5–30 minutes for fuel reports).

The fuel-relevant fields we extracted are fuel on board (FOB), occasionally takeoff weight (TOW), and supplementary parameters such as altitude, true airspeed (TAS), calibrated airspeed (CAS), and Mach number. A recurring challenge is that FOB values are reported without explicit units. Depending on the airline and message format, a reported FOB may be in kilograms or pounds, and may be scaled by a factor of 10, 100, or 1000 for transmission compactness. Section 3.2 describes how we resolve this ambiguity.

### 2.2 ADS-B via the OpenSky Network

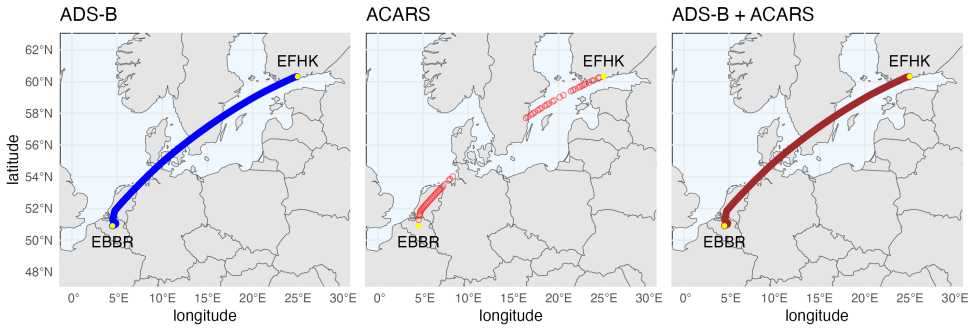
Automatic Dependent Surveillance–Broadcast is a GPS-based surveillance technology in which aircraft broadcast their position, velocity, and identification at high frequency (typically 0.5–2 second intervals) on 1090 MHz. The OpenSky Network [6] operates a global network of over 4,000 active crowdsourced receivers that collect these transmissions and archive them in a Trino database accessible to researchers.

OpenSky’s standard state vector data store provides pre-decoded positions and velocities, but these contain considerable noise and outliers from decoding errors, receiver clock drift, and multipath interference. Rather than using the state vectors directly, we download raw Mode S messages and re-decode them from scratch. We implemented a `rebuild()` method that re-constructs each flight by decoding Compact Position Report (CPR)-encoded positions with *pyModeS* [7, 8], pairing odd and even frames, and applying strict consistency checks (detailed in Section 3.1.2). This re-decoding step eliminates most of the artifacts present in the pre-processed state vectors.

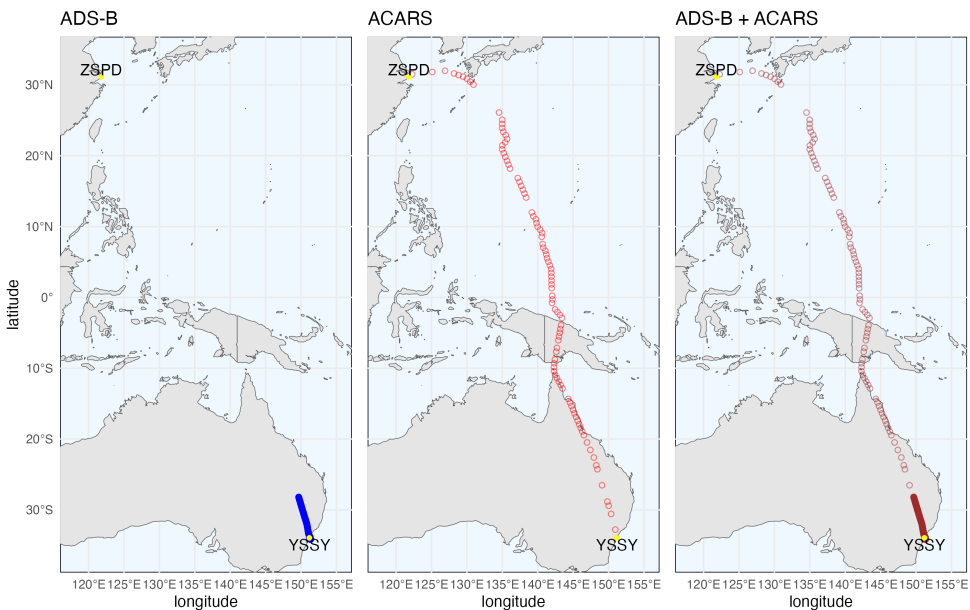
### 2.3 Complementary strengths

ACARS and ADS-B complement each other: ACARS has the fuel labels but sparse timing; ADS-B has dense trajectories but no fuel data. Figures 1 and 2 illustrate the difference. On the European flight

(EFHK–EBBR), both sources cover the full route. On the long-haul (ZSPD–YSSY), ADS-B ground receivers only see the endpoints while ACARS fills the oceanic gap.



**Figure 1.** European flight (EFHK–EBBR) with good ADS-B (blue) and ACARS (red) coverage. Left: fused trajectory. Center: ADS-B only. Right: ACARS only.

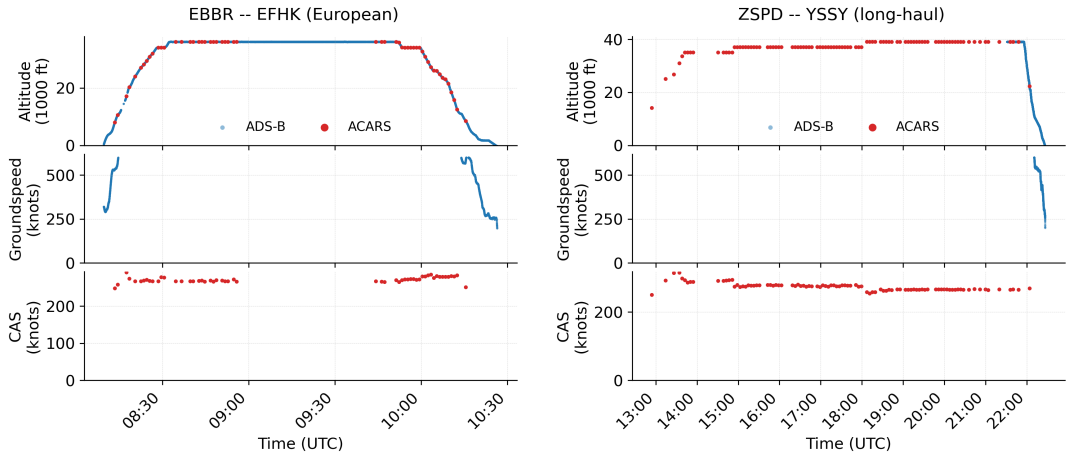


**Figure 2.** Long-haul flight (ZSPD–YSSY) with sparse ADS-B but extensive ACARS coverage. ADS-B ground receivers only cover the departure and arrival areas; ACARS fills the oceanic gap.

The complementarity extends beyond position coverage. Figure 3 shows the altitude, groundspeed, and calibrated airspeed (CAS) profiles for the same two flights. ADS-B provides groundspeed (a ground-referenced measurement), while ACARS reports CAS (an air-referenced measurement). During cruise at 35,000 ft, the European flight shows a groundspeed around 480 knots but a CAS of only 270 knots, the difference being due to the decrease in air density at altitude. For the long-haul flight, ADS-B data is almost entirely absent over the ocean, but ACARS provides both altitude and CAS throughout.

## 2.4 Flight metadata

Flight metadata, including takeoff and landing times, aircraft type codes, and callsign-to-flight matching, was provided by EUROCONTROL. We restrict the dataset to commercial operations with valid



**Figure 3.** Speed profiles for the European flight EFHK–EBBR (left) and long-haul flight ZSPD–YSSY (right). Each panel shows altitude, groundspeed (ADS-B, blue), and CAS (ACARS, red). The European flight has ADS-B coverage during climb and descent but a gap over the Baltic. The long-haul flight has ADS-B only near the destination, while ACARS covers the full route.

ICAO 24-bit addresses and callsigns, where both origin and destination airports fall within the ACARS and ADS-B coverage area, and where at least eight ACARS fuel messages are available for any given flight.

### 3. Methodology

#### 3.1 Data collection pipeline

The dataset construction proceeds in three stages.

##### 3.1.1 Stage 1: ACARS extraction

We query the airframes.io API using the Scrapy framework, filtering for messages containing fuel-related keywords (“FOB”, “TOW”, “RMK/FUEL”, “FUEL IN TANKS”). Scraping runs hourly. Duplicate messages (same aircraft within a 2-minute window) are discarded, and results are stored in a PostgreSQL database with one table per day.

From each message, we extract the FOB value using regular expressions. The FOB pattern requires some care: we use negative lookbehinds to avoid matching “REQD FOB” (a fuel request, not a report) and handle multiple airline-specific formats. Some airlines report fuel as a bare integer (“FOB 177”), others embed it in structured fields (“/FOB 0264/”), and a few use natural-language variants (“RMK/FUEL 123.4”). The patterns used are:

```

FOB\s*(\d+)
/FOB\s+(\d+)/
RMK/FUEL\s+(\[\d.\ ]+)
FUEL IN TANKS\s+(\[\d.\ ]+)

```

TOW, when present, is extracted separately and merged to the same flight by matching on ICAO address and flight number.

### 3.1.2 Stage 2: ADS-B trajectory retrieval

For each flight with at least eight fuel messages, we query the OpenSky Trino database by ICAO 24-bit address and time window (takeoff to landing, rounded to the nearest hour). We retrieve raw Mode S messages for both position and velocity reports.

Position decoding from ADS-B requires pairing odd and even CPR frames. We match each odd frame to the nearest even frame within a 10-second window (and vice versa), then decode using `pyModeS`. To reject decoding errors, we validate each decoded position against two reference points: the position decoded 5 messages earlier and 10 messages earlier. Any position where either reference comparison exceeds  $0.1^\circ$  in latitude or longitude is discarded. Velocity reports are merged to the nearest position by timestamp with a 3-second tolerance.

This two-stage reference validation catches the occasional CPR decoding artifact that a single-reference check would miss, particularly near the equator and the prime meridian where CPR zone boundaries can produce spurious jumps.

### 3.1.3 Stage 3: Data fusion

We parse each ACARS message for altitude (“ALT” or flight level “FL” fields), speeds (“TAS”, “CAS”), Mach number (“MCH” or “M 0. xxx” patterns), and position. Flight levels below 500 are converted to feet by multiplying by 100. Mach values are normalized: a raw value like “853” becomes 0.853.

The parsed ACARS records are concatenated with the ADS-B state vectors into a single time-sorted dataframe. Each record carries a source label (“acars” or “adsb”). ADS-B provides position, altitude, ground speed, track, and vertical rate. ACARS adds Mach, TAS, CAS, and latitude/longitude where available. The output is one Parquet file per flight.

### 3.1.4 Timestamp handling

The two data sources use different timing mechanisms. ADS-B timestamps come from the OpenSky receivers: each Mode S message is tagged with the time it was received at the ground station (`mintime` in OpenSky’s schema), measured in Unix seconds. These are generally reliable but can drift by up to a few seconds depending on the receiver’s clock synchronization.

ACARS timestamps come from the `airframes.io` API and reflect when the message was received and logged by the platform. This introduces two layers of delay relative to the actual onboard measurement: transmission latency from aircraft to ground station, and processing latency within `airframes.io`. The ACARS message text itself does not contain an explicit timestamp for the reported parameters.

When we merge ADS-B and ACARS into a single timeline, we sort all records by their respective timestamps and interleave them. For the fuel unit inference step, we resample the combined trajectory to 5-second intervals and interpolate the OpenAP mass profile at ACARS message timestamps.

For the final dataset, consecutive ACARS timestamps define the fuel intervals, so any timing offset in ACARS propagates directly into the interval boundaries and the associated trajectory segment.

This means the dataset has an inherent timestamp inconsistency: the ACARS-reported fuel state may lag the actual onboard measurement by seconds to minutes, while the ADS-B trajectory timestamp refers to the moment the signal was received on the ground. We do not attempt to correct for this offset because the ACARS transmission delay is unknown and varies by message. Users should be aware that the fuel interval boundaries are approximate, not exact.

## 3.2 Fuel unit inference

### 3.2.1 The problem

ACARS fuel-on-board messages report a numeric value without specifying units. The value might be in kilograms or pounds, and might be scaled by a factor of 10, 100, or 1000. A single message reading “FOB 177” has eight possible interpretations: 177 kg, 1,770 kg, 17,700 kg, 177,000 kg, 177 lbs (80 kg), 1,770 lbs (803 kg), 17,700 lbs (8,028 kg), or 177,000 lbs (80,286 kg). Without knowing which is correct, the fuel data is unusable.

### 3.2.2 Physics-based resolution

We resolve this by comparing ACARS-reported fuel differences against physics-based predictions from OpenAP [4]. The procedure works as follows.

First, we construct a reference fuel consumption profile for the flight. The combined ADS-B/ACARS trajectory is filtered and resampled to 5-second intervals using the traffic library [9]. OpenAP then estimates fuel flow at each point, given the aircraft type, speed, altitude, vertical rate, and an assumed initial mass of  $0.8 \times \text{MTOW}$  (MTOS = Maximum Takeoff Weight). Integrating this fuel flow gives a cumulative mass profile. We interpolate this profile at each ACARS message timestamp to obtain an OpenAP-predicted mass at the time of each fuel report.

Next, we test eight unit hypotheses against this reference. For each consecutive pair of ACARS fuel messages, the raw FOB difference  $\Delta\text{FOB}$  is converted to kilograms under each hypothesis:

- Four hypotheses assume the native unit is kg:  $\Delta\text{FOB} \times k$ , with  $k \in \{1, 10, 100, 1000\}$ .
- Four hypotheses assume pounds:  $\Delta\text{FOB} \times k \times 0.4536$ , with the same scaling factors.

We compute the total absolute error between each hypothesis and the OpenAP-predicted mass change across all message pairs (restricting to pairs where FOB decreases). The hypothesis with the smallest total error is selected.

This works because OpenAP predictions, while not exact, are accurate enough to distinguish between hypotheses that differ by at least a factor of 2.2 (kg vs. lbs) or 10 (scaling factors). Only intervals where the inferred fuel consumption falls between 5% and 500% of the OpenAP prediction are retained.

### 3.2.3 Quality filtering

After unit selection, we discard intervals where FOB increases (no mid-flight refueling) and intervals where the result is non-positive.

## 3.3 Coordinate parsing

ACARS position reports appear in at least six text formats. We apply a hierarchical set of regular expressions, trying the most specific format first and falling back to broader patterns:

1. Named format: LAT [NS] dd.ddd, LON [EW] ddd.ddd
2. DDMMSS: [NS]ddmmss[EW]dddmmss
3. Combined: [NS]ddmm.m[EW]dddmm.m
4. Separated: [NS]ddmm.m [EW]dddmm.m
5. Decimal: [NS] dd.ddd [EW] ddd.ddd
6. Embedded: ddm[NS]dddmm[EW]

Parsed coordinates are range-checked and cross-validated against the ADS-B trajectory where both sources overlap.

### 3.4 Flight interval construction

The ground-truth labels for the challenge consist of fuel intervals: pairs of consecutive ACARS fuel reports that define a time window and the fuel consumed within it.

For each pair of consecutive, time-ordered ACARS fuel messages in a flight, we record the start and end timestamps, the FOB at each endpoint, and the fuel consumed (in kg, after unit correction). Intervals shorter than 5 minutes or longer than 60 minutes are excluded. We require FOB to be monotonically decreasing across consecutive reports within a flight.

## 4. Dataset description

### 4.1 Temporal and geographic coverage

The dataset spans seven months: training data covers April through August 2025 (five months), the ranking evaluation set covers September 2025, and the final evaluation set covers October 2025. Geographic coverage centers on Europe and the North Atlantic corridor, with opportunistic global coverage wherever ACARS and ADS-B reception overlap. The coverage reflects the distribution of community-operated receivers in both networks.

### 4.2 Dataset statistics

Table 1 summarizes the dataset. The training partition contains 11,037 flights and 131,530 fuel intervals, totaling roughly 5 GB in Parquet format. Across all partitions, the dataset covers 15,761 flights with 193,275 fuel intervals and 27 aircraft types.

Table 1. Dataset statistics

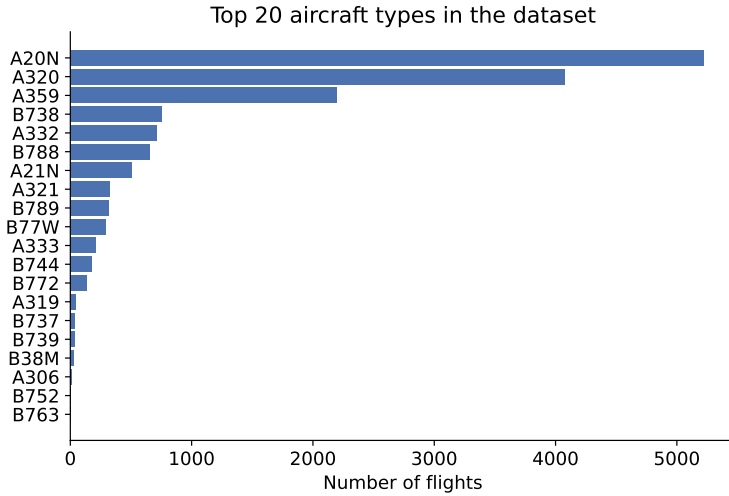
	Training	Ranking	Final
Period	Apr–Aug 2025	Sep 2025	Oct 2025
Flights	11,037	1,888	2,836
Fuel intervals	131,530	24,289	37,456
Unique aircraft types	26	19	21

The aircraft type distribution (Figure 4) is dominated by the A320 family (A20N and A320 together account for 59% of flights), followed by the A350 (14%), B737-800 (5%), and A330 (5%). Wide-body types such as A350, B787, B777, and B744 make up roughly a quarter of the dataset. The average flight duration is 333 minutes (median 251), with an average of 12 fuel intervals per flight. Individual intervals average 8 minutes in length and have a median fuel consumption of 200 kg. Figure 5 shows the distribution of fuel consumed per interval. Most intervals correspond to short cruise segments with low fuel burn; the long tail reflects climb phases and long intervals on wide-body aircraft. Figure 6 shows the daily flight count over the dataset period, reflecting both the data collection ramp-up and seasonal variation.

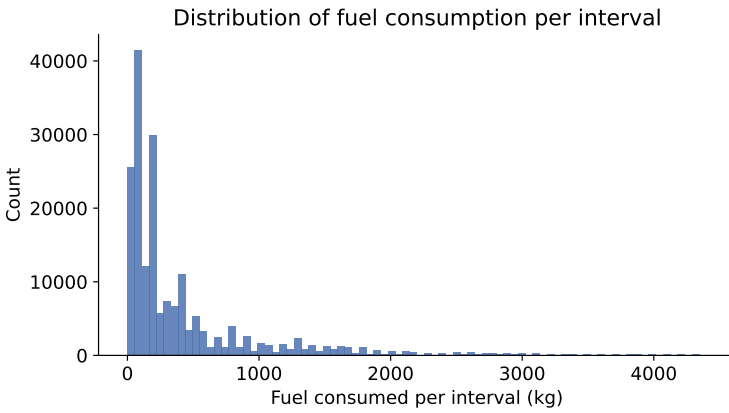
### 4.3 Data format and schema

The dataset for the competition is organized as follows:

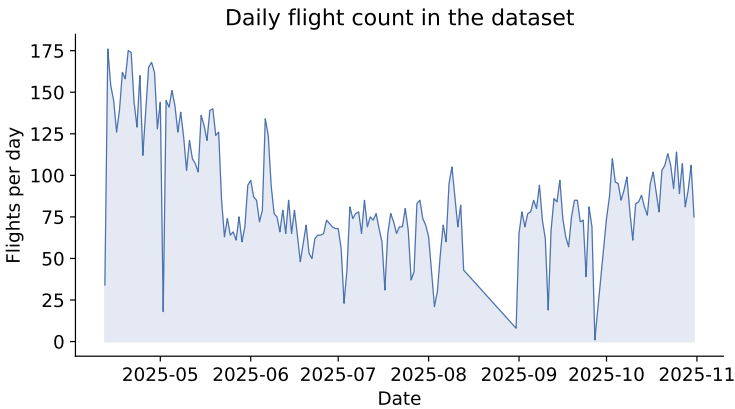
```
prc2025_dataset/
  fuel_train.parquet
  fuel_rank_submit.parquet
  fuel_final_submit.parquet
  flightlist_train.parquet
```



**Figure 4.** Distribution of aircraft types across all dataset partitions. The A320 family dominates, consistent with the European commercial fleet.



**Figure 5.** Distribution of fuel consumed per interval (kg). The 99th percentile is truncated for clarity.



```

flightlist_rank.parquet
flightlist_final.parquet
flights_train/
  prc12345.parquet
  prc12346.parquet
  ...

```

Each fuel interval record contains a flight identifier, start and end timestamps (UTC), the fuel consumed in kilograms (provided for training data, withheld for evaluation sets), and a sequence index. Table 2 describes the trajectory file schema.

**Table 2.** Trajectory file schema

Column	Type	Unit	Description
timestamp	datetime	UTC	Record timestamp
flight_id	string	-	Flight identifier
typecode	string	-	ICAO aircraft type
latitude	float	deg	Geographic latitude
longitude	float	deg	Geographic longitude
altitude	float	m	Barometric altitude
groundspeed	float	m/s	Ground speed
track	float	deg	Track angle
vertical_rate	float	m/s	Vertical speed
mach	float	-	Mach number (from ACARS)
TAS	float	m/s	True airspeed (from ACARS)
CAS	float	m/s	Calibrated airspeed (from ACARS)
source	string	-	“acars” or “adsb”

Flight metadata files include the flight identifier, date, aircraft type code, origin and destination ICAO codes and airport names, and actual takeoff and landing times.

#### 4.4 Data quality

The dataset retains real-world noise. Not all airlines report fuel via ACARS, and ADS-B coverage has oceanic gaps. The unit inference may produce residual errors for aircraft types underrepresented in OpenAP.

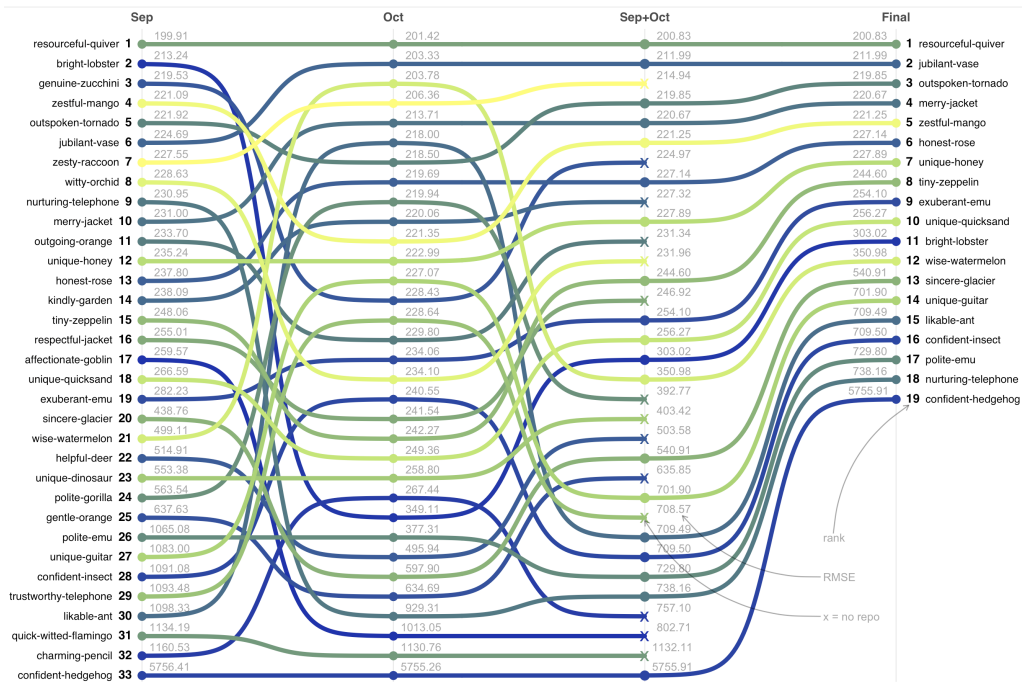
## 5. Discussion

### 5.1 Competition results

The challenge asked participants to predict fuel consumed in each defined interval, given a flight trajectory and aircraft type. Evaluation uses RMSE in kilograms. No baseline models were provided; teams were free to choose any approach.

The competition saw the registration of 179 teams of which 33 submitted till the final stage. Figure 7 shows how team rankings shifted across the final evaluation phase (September data, October data, and the combined score). Several teams that ranked well on one month’s data dropped or rose on

the other, suggesting that generalization across time periods was a real challenge. The winning team achieved a final RMSE of 201 kg on the combined evaluation set.



**Figure 7.** Ranking evolution across the final phase. Teams are ranked by RMSE (lower is better) on September data, October data, the combined set, and the overall final score. Lines connect the same team across columns.

Beyond the competition, the dataset supports aircraft performance model validation (e.g., comparing BADA [10] and OpenAP against observed fuel burn), flight efficiency analysis across operators and routes, trajectory optimization, and anomaly detection in fuel consumption patterns. The open format also makes it suitable for classroom exercises and student projects.

## 5.2 Limitations

The geographic coverage is concentrated in Europe and the North Atlantic, reflecting where community ACARS and ADS-B receivers are deployed. Not all aircraft or airlines report fuel via ACARS, so the dataset has a selection bias toward equipped fleets. Airline identities are anonymized for the competition, which limits operator-specific analysis.

The unit inference and outlier filtering rely on OpenAP as a reference model, but OpenAP is not always accurate. Its fuel flow estimates can deviate substantially from reality for certain engine variants, non-standard operating procedures, or aircraft types with limited performance data. To avoid discarding valid data points, we use a deliberately wide acceptance margin (5% to 500% of the OpenAP prediction). This means some incorrectly labeled intervals pass through the filter. In particular, the kg-versus-lbs ambiguity is hard to resolve for short flights or flights with few ACARS messages, where the cumulative fuel difference between the two hypotheses may be small relative to the OpenAP prediction error. As a result, the dataset still contains some inaccurate fuel labels. Users should be aware of this residual noise when training models or interpreting results.

## 6. Conclusion

This paper described how we built the dataset for the 2025 PRC Data Challenge on fuel burn estimation. The core idea is to pair two crowdsourced data sources that individually are insufficient: ACARS provides fuel measurements but at sparse intervals, and ADS-B provides dense trajectories but no fuel information. Fusing them produces labeled training data for supervised learning without requiring access to airline-internal records.

The main technical challenge was resolving the ambiguous units in ACARS fuel reports. We addressed this with a physics-based hypothesis test using OpenAP, accepting a wide margin to avoid discarding valid data at the cost of some residual label noise. The resulting dataset covers 15,761 flights and 193,275 fuel intervals across 27 aircraft types. Thirty-three teams participated in the competition, with the best achieving an RMSE of 201 kg.

## Acknowledgement

We thank the EUROCONTROL Performance Review Commission for organizing and funding the data challenge. We are grateful to the OpenSky Network community for maintaining the ADS-B reception infrastructure, and to the airframes.io contributors who operate the ACARS reception network. We also thank the competition participants for their feedback during the challenge.

## Author contributions

- Junzi Sun: Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Writing (Original Draft)
- Enrico Spinielli: Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Writing (Review and Editing)
- Martin Strohmeier: Data Curation, Resources, Writing (Review and Editing)

## Open data statement

The dataset is available on Zenodo at <https://doi.org/10.5281/zenodo.19184661> under a Creative Commons Attribution 4.0 International license (CC BY 4.0). The dataset consists of Parquet files totaling approximately 5 GB.

## References

- [1] David S Lee, David W Fahey, Agnieszka Skowron, Myles R Allen, Ulrike Burkhardt, Qi Chen, Sarah J Doherty, Susannah Freeman, Piers M Forster, Jan Fuglestedt, et al. "The contribution of global aviation to anthropogenic climate forcing for 2000 to 2018". In: *Atmospheric Environment* 244 (2021), p. 117834.
- [2] EUROCONTROL Performance Review Commission. *Performance Review Report (PRR 2023)*. Tech. rep. EUROCONTROL, 2024.
- [3] Enrico Spinielli, Junzi Sun, Martin Strohmeier, Xavier Olive, Sam Goens, Rainer Koelle, Katrin Tart, and Irene Fitzgerald. "Aircraft takeoff weight estimation: The EUROCONTROL PRC 2024 Data Challenge". In: *Journal of Open Aviation Science* 3.2 (2025). DOI: [10.59490/joas.2025.8252](https://doi.org/10.59490/joas.2025.8252).
- [4] Junzi Sun, Jacco M Hoekstra, and Joost Ellerbroek. "OpenAP: An open-source aircraft performance model for air transportation studies and simulations". In: *Aerospace* 7.8 (2020), p. 104.
- [5] *Airframes.io: Community ACARS aggregator*. <https://airframes.io>. Accessed: 2025-09-01. 2025.

- [6] Matthias Schäfer, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm. “Bringing up OpenSky: A large-scale ADS-B sensor network for research”. In: *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*. IEEE, 2014, pp. 83–94.
- [7] Junzi Sun, Huy Vũ, Joost Ellerbroek, and Jacco M Hoekstra. “pyModeS: Decoding Mode S surveillance data for open air transportation research”. In: *IEEE Transactions on Intelligent Transportation Systems* 21.7 (2020), pp. 2777–2786.
- [8] ICAO. *Annex 10 to the Convention on International Civil Aviation: Aeronautical Telecommunications, Volume IV – Surveillance and Collision Avoidance Systems*. 2018.
- [9] Xavier Olive. “traffic, a toolbox for processing and analysing air traffic data”. In: *Journal of Open Source Software* 4 (2019), p. 1518. ISSN: 2475-9066. DOI: [10.21105/joss.01518](https://doi.org/10.21105/joss.01518).
- [10] David I A Poll and Ulrich Schumann. “An estimation method for the fuel burn and other performance characteristics of civil transport aircraft in the cruise. Part 1: Fundamental quantities and governing relations for a general atmosphere”. In: *The Aeronautical Journal* 125.1284 (2021), pp. 257–295.