

Feature-Specific Anomaly Detection for ADS-B Trajectory Data Using LSTM Autoencoders

Tiffany Kuhn^{*} and Manuel Waltert

Centre for Aviation, Zurich University of Applied Sciences, Winterthur, Switzerland

*Corresponding author: tiffany.kuhn@zhaw.ch

Abstract

Automatic Dependent Surveillance-Broadcast (ADS-B) trajectory data is critical for aviation research, yet data quality is frequently impaired by sensor noise, transmission errors, or spoofing. This paper proposes an unsupervised, feature-specific anomaly detection framework comprising three Long Short-Term Memory (LSTM) autoencoders targeting position, speed, and vertical motion. Using dual reconstruction error thresholds controlled by a scaling parameter r , the framework flags anomalies and assists in error localization across trajectory dimensions without requiring labelled training data. Evaluated on OpenSky trajectories from Swiss airspace, the models successfully detect injected anomalies across various scales and align reasonably well with traditional heuristic filters, while allowing the false-positive rate on normal flight data to be tuned down to one percent or below.

Keywords: ADS-B Trajectory; Anomaly Detection; Machine Learning; Autoencoder

Abbreviations: ADS-B: Automatic Dependent Surveillance-Broadcast, LSTM: Long Short-Term Memory

1. Introduction

ADS-B trajectory data is widely used in aviation research for trajectory prediction, conflict detection, and traffic-flow analysis [1]. Further applications include airport and route optimization [2], as well as sustainability research through improved emissions estimates [3]. All of these applications rely on accurate and consistent trajectories, and poor data quality impairs their performance [4]. Because ADS-B messages are broadcast without authentication and are received by heterogeneous ground stations, recorded trajectories can contain erroneous segments. These errors can arise from data acquisition and encoding issues as well as deliberate spoofing, making data-quality assessment a prerequisite for downstream use [5, 6].

In this work, we use the term *normal trajectory* to denote kinematically plausible aircraft motion, characterised by smooth, physically consistent position, speed, and vertical motion profiles as typically observed in recorded flights. Conversely, an *anomalous trajectory* contains one or more deviations from this behaviour, such as position jumps, noise, frozen or drifting values, or implausible speed and vertical rate offsets and spikes. The distinction is made at the trajectory level, even when deviations affect only individual segments. We consider such data-quality anomalies broadly, without distinguishing their origin, and aim at general-purpose data-quality assessment rather than at a specific route, aircraft type, or attack scenario.

A range of classical signal-processing and machine-learning methods have been proposed for detecting anomalies in aircraft trajectory data. Classical approaches use threshold rules, clustering, and Kalman-based filtering [7]. These methods depend on hand-crafted rules and parameters, which makes them laborious to maintain for large-scale datasets, where manual inspection is impractical.

Machine-learning approaches, particularly encoder-decoder architectures, instead learn normal behaviour from data and flag deviations via reconstruction error [8]. Habler and Shabtai [9] applied an LSTM encoder-decoder to specific flight routes, reporting a 4.3 % false alarm rate with limited transferability, while other studies use variational and generative LSTM variants to detect attacks like spoofing [10, 11]. However, these models primarily target route-specific or security-related anomalies rather than general-purpose data-quality assessment. They provide little insight into which trajectory characteristics drive an anomaly and lack quantitative scores for automated screening, a gap critical for offline screening, where false alarms cause unnecessary data loss. Furthermore, standard preprocessing steps can distort trajectory characteristics and remove rare but genuine events [1].

We propose an unsupervised framework of three feature-specific LSTM autoencoders for position, speed, and vertical motion, trained on flights that pass a quality filter and therefore define “normal”. Our study investigates (i) whether such autoencoders distinguish normal from anomalous trajectory segments, (ii) to what extent a flag can be attributed to the feature group in which the error originates, and (iii) whether feature-specific reconstruction errors can serve as interpretable anomaly scores for automated screening.

2. Methods

2.1 Data and Preprocessing

We used OpenSky [5] ADS-B trajectories from 24 randomly selected days in 2024 within Swiss airspace. The data was resampled to 1 Hz in preprocessing. A rule-based filtering pipeline was applied to determine which flights were retained for analysis. Flights were excluded if they fell outside the study scope (e.g., general aviation and gliders identified by callsign prefixes), did not meet minimum thresholds for duration, speed, or altitude, contained temporal gaps, or violated plausibility and consistency criteria. The latter included discrepancies in position-derived distances, implausible changes in track or speed, and threshold exceedances in derived consistency features.

Flights rejected by the plausibility and consistency checks were retained separately to evaluate how far the models flag the same data that these rules exclude. Of the 87 515 initial flights, 9924 passed the filtering pipeline. These were split by flight ID into training, validation, and test sets (70 %, 10 % and 20 %, respectively) before segmentation to prevent data leakage from overlapping sequences across the sets. Each set was then segmented into overlapping 60 s sequences with a stride of 15 s, yielding 279 347, 39 882, and 78 033 sequences, respectively. All features were scaled using parameters estimated exclusively from the training set.

2.2 Feature Groups and Models

We developed three separate LSTM autoencoders, each operating on a distinct feature group: *Position*, *Speed*, and *Vertical*. This separation allows detected anomalies to be attributed to the corresponding aspect of aircraft motion. Besides directly observed quantities, the groups include derived consistency measures that can reveal discrepancies between signals even when individual measurements appear plausible. All models were implemented in Keras and TensorFlow.

For each group, we compared several encoder–decoder LSTM architectures varying in capacity, latent dimension, loss function, activation function and smoothing (selected architectures: Table 1). One model per group was selected using normal data only, based on feature-wise mean squared error (MSE) and coefficient of determination (R^2) on the validation set, the training–validation error gap, and the false-positive rate (FPR) on normal test sequences. Results are averaged over six training seeds.

Table 1. Input features and architectures of the three LSTM autoencoders. Architectures are specified as encoder layer widths / latent dimension and activation function / decoder layer widths.

Model	Features	Architecture	Loss
Position	dx (m), dy (m), turn intensity ($^\circ$)	96, 48 / 24 (tanh) / 96, 192	MSE
Speed	groundspeed (kt), Δ groundspeed (kt s^{-1}), gs_diff (kt)	128, 64 / 32 (ReLU) / 128, 256	Huber
Vertical	altitude (ft), vertical rate (ft min^{-1}), vs_diff (ft min^{-1})	128, 64 / 32 (ReLU) / 128, 256	Huber

2.3 Anomaly Scoring and Thresholds

Anomalies are identified from reconstruction errors, with larger deviations between the observed and reconstructed features indicating potentially anomalous behaviour. For each sequence of $t = 1, \dots, T$ ($T = 60$) time steps, let $x_{t,f}$ denote the scaled value of feature f at time step t and $\hat{x}_{t,f}$ its reconstruction. The squared reconstruction error $e_{t,f}$ and the mean squared error MSE_f over the sequence are defined as

$$e_{t,f} = (x_{t,f} - \hat{x}_{t,f})^2, \quad MSE_f = \frac{1}{T} \sum_{t=1}^T e_{t,f}. \quad (1)$$

We apply two complementary detection criteria to capture both persistent deviations and short-lived anomalies. The sequence-level criterion uses MSE_f to identify deviations extending over multiple time steps. However, an anomaly lasting only k steps contributes proportionally to k/T of the sequence mean and may therefore remain undetected. The step-level criterion addresses this limitation by identifying sequences containing at least two time steps with unusually large reconstruction error. For each feature, the thresholds are determined from normal validation data: τ_f is the 99% quantile of the sequence-level MSE_f , while θ_f is the 99.9% quantile of $e_{t,f}$ across all validation time steps. A feature is flagged as anomalous if

$$MSE_f > r \cdot \tau_f \quad \text{or} \quad \sum_{t=1}^T \mathbf{1}[e_{t,f} > r \cdot \theta_f] \geq 2, \quad (2)$$

where $r \geq 1$ jointly scales both thresholds. The default setting is $r = 1$, while larger values reduce sensitivity and the frequency of false alarms. A sequence is flagged by a model if at least one of its features satisfies either criterion. The overall anomaly decision is positive if any of the three models flags the sequence.

2.4 Evaluation

We evaluate the autoencoders using two complementary approaches. First, synthetic anomalies of known magnitude and duration are injected into held-out normal flights to assess detection sensitivity under controlled conditions. An injected anomaly is considered detected if it causes the corresponding model to flag a sequence that was not flagged in the unmodified flight. Second, the models are applied to flights rejected by the plausibility and consistency checks during data preparation. We measure the proportion of rule violations detected by the autoencoders to assess their agreement with conventional rule-based filtering. Since these rules were designed for data cleaning rather than establishing ground-truth anomaly labels, this evaluation provides a comparison with heuristic filtering rather than an independent measure of detection accuracy.

3. Results and Discussion

We first assess the flagging rates on normal test data, followed by the detection of injected anomalies and rule violations in excluded flights. All results are averaged over six training seeds and reported for threshold scaling factors $r \in \{1, 1.5, 2\}$.

3.1 Baseline Performance on Normal Data

At $r = 1$, the three models flag between 2.9 % and 4.1 % of normal test sequences. Increasing the threshold scaling factor to $r = 1.5$ reduces these rates to 0.7 % to 1.6 %, while at $r = 2$ they remain at or below 1.0 % (Table 2). The rates at $r = 1$ exceed the nominal 1 % validation quantile because a sequence is flagged if any feature-level detection criterion is satisfied, increasing the overall probability of an alarm.

Table 2. Share of normal test sequences flagged (%), mean over six seeds.

Model	$r = 1$	$r = 1.5$	$r = 2$
Position	4.1	0.7	0.2
Speed	2.9	1.6	1.0
Vertical	3.3	1.4	0.7

3.2 Detection of Injected Anomalies

We injected jumps, spikes, offsets, noise, drift, and freezes of increasing size and duration into normal flights. Zero-magnitude control injections produced no additional detections, indicating that the injection procedure itself did not introduce detectable artifacts. Table 3 gives, for $r = 1$ and $r = 2$, the smallest tested size at which at least half of the events and at least 95 % of the events are found. Ranges cover the tested durations.

Table 3. Injected anomalies: smallest tested size detected in at least 50 % and 95 % of the events. Ranges cover the tested durations. Sizes refer to errors injected into the raw signal before smoothing.

Model	Anomaly	$r = 1$		$r = 2$	
		50 %	95 %	50 %	95 %
Position	jump, temporary [m]	25-50	75-250	50-75	100-250
	jump, persistent [m]	50	250	75	250
	noise, σ [m]	5	10	10	30
	drift [m/s]	15	50	20	50
Speed	noise, σ [kt]	1	1	1	1.5
	offset / spike [kt]	2-5	2-5	2-5	5
	drift [kt/s]	0.1	0.1	0.1	0.25
Vertical	altitude noise, σ [ft]	100	100	100	200
	altitude spike [ft]	100	100-200	100-200	200-300
	altitude step [ft]	100	150	150	200
	rate spike [ft/min]	300	300-400	300-400	400-500
	rate offset [ft/min]	300	300-400	400	500

The low detection limits for speed reflect the smoothness of the preprocessed groundspeed: the model reconstructs normal sequences almost exactly, so the validation-based thresholds are narrow and even small deviations exceed them.

Increasing r raises detection limits but preserves sensitivity to large errors: the largest tested size of every anomaly type except a frozen vertical rate is still detected in at least 95 % of the events. Frozen positions (99 % to 100 %), frozen altitudes (98 % to 100 %) and vertical-rate sign flips (100 %) are almost always detected, as are frozen speeds that change by at least 10 kt during the freeze. A frozen vertical rate is the weakest case (59 % to 76 % on average, 62 % to 86 % at best, depending on r), because constant climb rates also occur in normal flight. Slow position drift needs 15 m s^{-1} to 20 m s^{-1} for a detection rate of 50 % and 50 m s^{-1} for 95 %.

Anomalies injected into one signal never raised an alarm in the model of an unrelated signal with one exception: position errors also change the position-derived speed used by `gs_diff`. At $r = 1$ the speed model therefore flags frozen positions (100 %) and 250 m jumps (97.5 % to 99.3 %; persistent jumps 79 %), but only 16 % to 35 % of 50 m jumps, depending on the duration. A position fault can thus appear in both models, whereas speed and vertical faults are flagged only by their own model.

3.3 Detection of Anomalies in Excluded Flights

We next examine whether the autoencoders identify violations of the rule-based criteria used during data preparation. Since these criteria were designed for data cleaning rather than ground-truth anomaly labelling, the resulting detection rates measure agreement with heuristic filtering rather than detection accuracy.

To understand this agreement, we distinguish two rule categories: Plausibility rules (`dist_diff`, `gs_jump`, `vertical_rate`) mark gross errors far outside normal operation. In contrast, the consistency limits `gs_diff` and `vs_diff` act on smoothed features that are also model inputs, meaning the training data end at these limits by construction. Table 4 shows, for the rule belonging to each model, the share of violations that lie in at least one flagged sequence.

Plausibility violations are found almost completely (`gs_jump` and `vertical_rate`: 99 % to 100 %; `dist_diff`: 82 % to 96 %). The consistency limits are detected less often, as expected: a value just above the limit lies at the edge of the range the model was trained on and is hardly unusual to it. Most `gs_diff` violations exceed the limit only marginally (median 52 kt), persist for only one or two rows and occur near the start or end of a flight (92 %), so they are averaged out over the sequence. Consistently, `vs_diff` detection increases with the size of the violation (71 % to 78 % at $r = 1$ for violations below and above 3000 ft min^{-1} , respectively), whereas the few large `gs_diff` violations show no clear trend.

Table 4. Excluded data: share of violations of the model's own rule that are detected [%], with the rule limit and the number of violations. Top: plausibility rules; bottom: consistency limits.

Model	Rule	Limit	Violations	$r = 1$	$r = 1.5$	$r = 2$
Position	<code>dist_diff</code>	>50 m	27 581	96.2	88.6	81.7
Speed	<code>gs_jump</code>	>30 m s^{-2}	70	100.0	100.0	100.0
Vertical	<code>vertical_rate</code>	>6000 ft min^{-1}	772	100.0	99.7	99.4
Speed	<code>gs_diff</code>	$\geq 45 \text{ kt}$	715	60.2	41.6	32.6
Vertical	<code>vs_diff</code>	$\geq 2000 \text{ ft min}^{-1}$	7308	72.5	58.4	48.3

These excluded-flight results are not used to estimate specificity, since violations of one rule cannot be assumed to represent normal observations for the other models. Instead, false-alarm behaviour is assessed on the normal test sequences in Section 3.1, while the injected anomalies provide a controlled assessment of detection sensitivity.

3.4 Limitations

Synthetic anomalies were injected individually into flights and represent a limited range of error patterns. No manually labelled real anomalies were available, and the excluded flights provide only an imperfect substitute because the exclusion rules were designed for data cleaning. In particular, `gs_diff` and `vs_diff` serve both as model inputs and exclusion criteria, meaning that their detection rates primarily reflect model behaviour near the boundaries of the training data rather than detection accuracy. Small, short-lived deviations in absolute quantities such as altitude remain difficult to detect because of their natural variability across flight phases. Furthermore, the evaluation is restricted to Swiss airspace, leaving generalisability to other airspaces and route structures untested. Finally, some sequences flagged as anomalous may contain genuine but unlabelled anomalies, making it impossible to distinguish these detections from false alarms.

4. Conclusion and Outlook

We presented an unsupervised approach to ADS-B trajectory anomaly detection using three LSTM autoencoders for position, speed, and vertical motion. Detection thresholds were derived from normal validation data, eliminating the need for labelled anomalies during training and calibration. The models detected a wide range of injected anomalies, including position jumps, altitude errors, and speed deviations, while allowing sensitivity to be adjusted through a single threshold scaling factor r . At $r = 1$, approximately 3 % to 4 % of presumed-normal test sequences were flagged. Increasing r reduced these rates to at most 1 %, while maintaining high detection rates for larger anomalies. The separate feature groups also supported anomaly attribution, although position errors could trigger both the position and speed models. Evaluation on excluded flights showed strong agreement with plausibility rules but lower detection rates for consistency-limit violations, highlighting the difficulty of identifying deviations close to the boundaries of normal training data.

Future work should evaluate the approach using manually labelled real anomalies and trajectories from other airspaces to establish detection accuracy and generalisability. Further improvements could include combining model outputs to identify the likely source of an anomaly, adapting thresholds to different flight phases, and developing segment-level filtering to remove affected trajectory portions without discarding entire flights.

Author contributions

- Tiffany Kuhn: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Visualization, Writing (Original Draft)
- Manuel Waltert: Conceptualization, Supervision, Project Administration, Resources, Writing (Review and Editing)

Reproducibility Statement

The code, the trained models and the scripts to reproduce the results are available at <https://github.com/kuhfzhaw/adsb-anomaly-detection> (version 1.0.0, CC BY 4.0). The OpenSky Network data used in this study is not redistributed; the repository describes how to retrieve it from the OpenSky Network.

References

- [1] Ruolan Ren, Jingcheng Zhong, Dizhi Guo, Ruixin Wang, and Christophe Hurter. “Review of ADS-B data usage with the focus on data cleaning”. In: *Journal of Open Aviation Science* 4.2 (2026). DOI: [10.59490/joas.2026.8467](https://doi.org/10.59490/joas.2026.8467).

- [2] Michael Schultz, Judith Rosenow, and Xavier Olive. "Data-driven airport management enabled by operational milestones derived from ADS-B messages". In: *Journal of Air Transport Management* 99, 102164 (2022). DOI: [10.1016/j.jairtraman.2021.102164](https://doi.org/10.1016/j.jairtraman.2021.102164). 198
- [3] Junzi Sun, Xavier Olive, Martin Strohmeier, and Vincent Lenders. "OpenSky report 2025: Improving crowdsourced flight trajectories with ADS-C data". In: *2025 integrated communications, navigation and surveillance conference (ICNS)*. 2025, pp. 1–8. DOI: [10.1109/icns65417.2025.10976963](https://doi.org/10.1109/icns65417.2025.10976963). 199
200
201
202
203
204
- [4] Zhanji Yang, Xiaolei Kang, Yuanhao Gong, and Jiansheng Wang. "Aircraft trajectory prediction and aviation safety in ADS-B failure conditions based on neural network". In: *Scientific Reports* 13, 19677 (2023). DOI: [10.1038/s41598-023-46914-2](https://doi.org/10.1038/s41598-023-46914-2). 205
206
207
- [5] 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. 208
209
210
211
- [6] Antoine Chevrot, Alexandre Vernotte, Pierre Bernabe, Aymeric Cretin, Fabien Peureux, and Bruno Legeard. "Improved testing of AI-based anomaly detection systems using synthetic surveillance data". In: *Proceedings* 59.1, 9 (2020). DOI: [10.3390/proceedings2020059009](https://doi.org/10.3390/proceedings2020059009). 212
213
214
- [7] Xavier Olive, Jan Krummen, Benoit Fiquet, and Richard Alligier. "Filtering techniques for ADS-B trajectory preprocessing". In: *Journal of Open Aviation Science* 2.2 (2025). DOI: [10.59490/joas.2024.7882](https://doi.org/10.59490/joas.2024.7882). 215
216
217
- [8] Oleksandr Provotar, Y. Linder, and M. Veres. "Unsupervised anomaly detection in time series using LSTM-based autoencoders". In: *2019 IEEE international conference on advanced trends in information theory (ATIT)*. 2019, pp. 513–517. DOI: [10.1109/atit49449.2019.9030505](https://doi.org/10.1109/atit49449.2019.9030505). 218
219
220
- [9] Edan Habler and Asaf Shabtai. "Using LSTM encoder-decoder algorithm for detecting anomalous ADS-B messages". In: *Computers & Security* 78 (2018), pp. 155–173. DOI: [10.1016/j.cose.2018.07.004](https://doi.org/10.1016/j.cose.2018.07.004). 221
222
223
- [10] Xinying Guo, Chunhua Zhu, Jing Yang, and Yan Xiao. "An anomaly detection model for ADS-B systems using a LSTM-based variational autoencoder". In: *2021 IEEE 3rd international conference on civil aviation safety and information technology (ICCASIT)*. 2021, pp. 1005–1009. DOI: [10.1109/iccasit53235.2021.9633677](https://doi.org/10.1109/iccasit53235.2021.9633677). 224
225
226
227
- [11] Xianyu Lu, Qiming Mai, Mengzhou Gao, and Guang Cheng. "A method for detecting abnormal ADS-B data using the VAE-GAN-LSTM algorithm". In: *2024 4th international conference on computer science, electronic information engineering and intelligent control technology (CEI)*. 2024, pp. 118–121. DOI: [10.1109/cei63587.2024.10871280](https://doi.org/10.1109/cei63587.2024.10871280). 228
229
230
231