

EVALUATION OF METHODS FOR ANOMALY DETECTION IN THE APPROACH PHASE OF OPERATIONAL FLIGHT DATA

R. Hähnel[1]

[1]German Aerospace Center, Institute of Flight Systems, Lilienthalplatz 7, 38108 Braunschweig, Germany

Abstract

Commercial operators of large passenger aircraft are required to establish a Flight Data Monitoring (FDM) system. Many airlines utilize Quick Access Recorder (QAR) data from every aircraft of their fleet as the foundation for their FDM. One such data set has been made available to the German Aerospace Center (DLR). The data are anonymized and unlabeled, meaning there is no available context information outside the collected parameters. The set contains roughly 130,000 flights and consists of heterogeneous, multivariate time series of different lengths and data types, creating constraints for the methods usable for its evaluation.
This paper investigates multiple methods to find operational anomalies that may or may not be safety critical within the set of flight data. An anomaly in this context refers to a rarely observed, significant deviation from the norm that suggests underlying operational disruptions. To keep the scope of work manageable, the analysis is focused on the approach phase of the flights, which is considered as highly critical from an accident prevention standpoint as a large share of fatal accidents occur during this phase. A data processing workflow has been built that serves to process the raw binary QAR data and outputs a set of human-readable flight samples that can be used by further algorithms.
Various authors have proposed methods of different complexities for anomaly detection (AD) in similar sets of operational flight data. A selection of these methods, including clustering-, forecasting- and reconstruction-based algorithms, is integrated into the processing workflow. Observations show that it can quickly become challenging to differentiate "correct" from "incorrect" AD behavior. Thus, this paper explores ways to benchmark detection algorithms with minimal amounts of labeling data required, for example by using missed approaches (MAs) as a proxy for anomalous flights or by evaluating the agreement of multiple methods.
The overarching goal is to build a better understanding of how safety critical situations can be identified in QAR data and explore options to detect safety critical events in historical data faster and more reliably than is currently possible with the state of the art Exceedance Detection (ED) method.

Keywords: Flight Data, Anomaly Detection, Flight Data Monitoring, Flight Safety, Quick Access Recorder

1. Introduction

Maintaining safe operations is a fundamental task of any aircraft operator. One way to improve safety is to analyze past critical situations, identify problematic trends that could lead to accidents in the future, and derive appropriate measures (e.g., through additional pilot training) in order to intervene before they become critical. This is part of the so called Flight Data Monitoring (FDM) system, which the European Union Aviation Safety Agency (EASA) mandates any commercial operator to establish [1]. The foundation for these analyses lies in discovering anomalous operations within historical data, usually collected using a Quick Access Recorder (QAR) [2]. This anomaly detection (AD) task is complicated due to the nature of the data set being heterogeneous, sequential, highly variant, and very extensive. This paper uses the term "anomaly" to describe unusual approaches that may or may not be safety-critical or inefficient. This definition is purposefully kept vague as it most closely resembles the problem description under Flight Operations Quality Assurance (FOQA), where all conspicuous operations are analyzed.

The state of the art approach to detect anomalous operations used by airlines is called exceedance detection (ED). ED works by defining a set of thresholds, representing safety critical conditions, for particular flight data parameters and marking any sample for further inspection that exceeds these boundaries [2, 3]. The defined checks for this method are simple and only inspect variables individually, making for a facile filter reliant on a deeper analysis by experts. For this reason, the spectrum of anomalous cases that can be detected with ED is limited. As the thresholds are set with respect to conditions that are established as dangerous, only incidents similar to previously experienced ones can be detected. In short, the method has a limited ability to generalize. In recent years, machine learning methods have been employed for many pattern recognition and classification tasks, exactly because of their ability to generalize beyond the data used for training.
This paper contributes on the one hand an evaluation of machine learning based AD methods, discussing in which ways they can improve on ED, and on the other hand, an exploration of methods to use unlabeled data for this evaluation.

2. Data Processing and Exploration

The German Aerospace Center (DLR) has access to anonymized Quick Access Recorder (QAR) raw data recorded during regular airline operations. The data available contain roughly 130,000 flights for several Boeing 737 aircraft.

2.1 Data Processing Workflow

The workflow shown in Figure 1 has been set up to read in the raw data, preprocess them, and output cleaned, individual flights, which are used for anomaly detection.

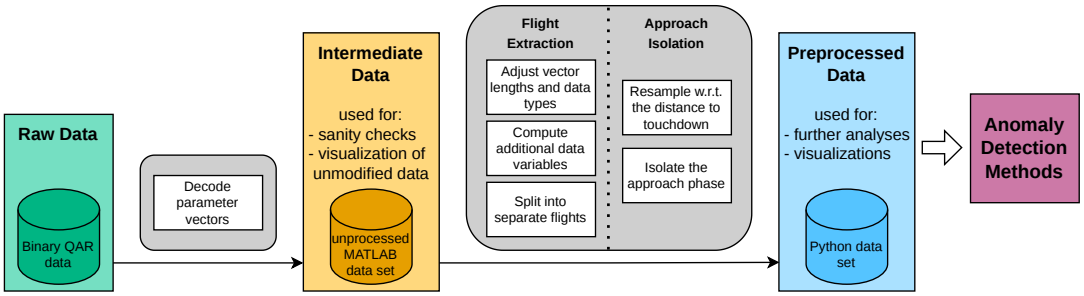


Figure 1 – Schematic of the data processing workflow

All time series signals are sampled to a common rate of 1 Hz, adjusted to be of numeric data types, and QAR files containing multiple flights are split into single flights. Finally, the flights are sampled at a uniform distance to touchdown vector, providing a comparable context and more accurately reflecting pilot management of the flight than using a time vector. This paper focuses on the last 20 nautical miles (NM) before landing, emphasizing the approach phase, which is critical for flight safety due to a significant proportion of fatal accidents occurring during this stage [4]. As a side effect, this preprocessing produces time series data of equal lengths, which makes all evaluation algorithms easier to implement.
Additional characteristic data variables are computed for every flight that are derived from the existing signals. The most important is an estimate of the altitude above ground level (alt AGL), which is calculated from the barometric altitude available in data. The barometric altitude is corrected by considering temperature and pressure prior to landing together with the QNH setting selected by the aircraft's crew. A caveat of this computation is that it loses accuracy the greater the distance to touchdown. For further information on the exact process refer to [5].

2.2 Benchmarking Data

As the data stem from a purely operational context, no labels of anomalous or safety-critical flights are available. A subset of samples was manually checked for unusual behavior and labeled as "nominal" or "anomalous" to obtain a "benchmarking set", which is used to compare the performance of different AD methods. The labeling was performed by comparing every flight in the set to all other

ones, taking note of significant deviations from the mean in multiple data parameters. The most important variables in this regard were related to energy management such as the altitude above ground, the air- and groundspeed, as well as control inputs and aircraft configuration like engine N1 and flaps settings.
This benchmarking set is intended to serve as a baseline for the performance evaluation. It consists of 975 samples in total, from which 47 are labeled as "anomalous", making for a class prevalence of 5 % of the "anomaly" class. Any samples of this set are excluded from later model training sets.

2.3 Missed Approaches

To obtain additional benchmarking data, the dataset was scanned for missed approaches (MAs), defined here as approaches deliberately interrupted before touchdown. The decision to initiate a MA suggests that the approach was perceived as anomalous by the operating crew, making MAs a potential surrogate for labeled anomalies. However, MAs arise from heterogeneous causes and are not limited to unstable or unsafe approaches. For instance, air traffic control may instruct a go-around despite a nominal trajectory. In such cases, the pre-maneuver data may not exhibit detectable deviations.
MA samples were identified by scanning all flights for positive rates of climb below 1000 m above ground level. The cases, for which this condition was met were subsequently inspected manually, yielding 364 confirmed MAs out of roughly 130,000 flights ($\approx 0.25\,\%$). Although this method is not exhaustive, it provides a sufficiently large set of MA samples.
To prepare this set for further evaluation, additional preprocessing was applied. A “decision point” was estimated for each MA, marking the onset of deviation from the nominal approach, where the MA was initiated. The distance to touchdown is a key variable for the workflow used in this paper. However, missed approaches do not result in an immediate landing so a “pseudo touchdown” was defined as a replacement. In order to estimate where the aircraft would have landed, if the approach had not been interrupted, the trajectory from the decision point is extrapolated along a 3 ° glide slope to ground level. This way, the distance to touchdown parameter can be used to normalize the MA samples the same way that all other flights were normalized before.
The AD algorithms have access to the entirety of the data for any flight. This poses a challenge as the final steps of a missed approach past the decision point deviate drastically from those of nominal trajectories. This enables a trivial detection, encouraging the AD methods to simply consider the very end of the data to make a classification.
To address this issue, data beyond the decision point were replaced with segments from a highly similar flight drawn from an independent data set. The "donor sample" was determined using a nearest neighbor search at the decision point and the splice itself was masked by fading between the two samples. A study showed that the resultant samples are smooth for less noisy parameters like altitude or groundspeed, but can show atypical artifacts for some parameters (e.g., pitch angle).
The resulting data set was manually checked for the detectability of the MA. The aim was to eliminate undetectable or non-representative cases, in which no significant precursors to a MA were present, or the MA occurred too early. A precursor could be a momentary strong deviation from the nominal trajectory (e.g., caused by gust) or a gradual trend toward abnormal conditions (e.g., an incorrect sink rate leading the aircraft off the vertical glide path).

3. Methods of Anomaly Detection for Multivariate Time Series Data

The standard approach to AD is framing it as a binary classification problem with an imbalance in class prevalence. Existing methods for this task can be categorized into forecasting, reconstruction, encoding, distance, distribution- and isolation tree methods [6]; however, other groupings are present in the literature. Particularly distance- [7], forecasting- [8] and reconstruction-based AD methods [3] have been used by researchers to analyze operational flight data and are compared to the state of the art threshold-based AD. The alternative methods were chosen because they represent different categories of machine learning, with clustering being a fully unsupervised method and both forecasting- and reconstruction based AD using supervised, neural network based methods. The comparison between forecasting- and reconstruction is particularly interesting as the methods share a very similar structure but work on different principles.

3.1 Temporal Context

The three dimensional data structure comprises sample, time, and feature dimensions. In preprocessing, the time dimension was normalized to a common distance to touchdown for all flights, and all other features are sampled with respect to this vector (see subsection 2.1); thus, the term "time" can be misleading, which is why the distance is used instead. At a remaining distance index s, all aircraft are approximately equidistant from touchdown. The actual time vector is not considered further as part of this paper. When time or time-steps are mentioned in later chapters, it is in reference to the second dimension of the data set matrix, which captures the causal and temporal context of the data.

Analyzing each approach step individually ignores causal dependencies, yielding anomaly scores that lack historical context. To incorporate this context, standard approaches include feature engineering, dimensionality reduction (e.g., Principal Components Analysis - PCA [9]), or specialized distance functions such as Dynamic Time Warping (DTW) [10] and Shape Based Distance (SBD) [11]. This paper prioritizes either PCA or feature engineering depending on the employed method.

3.2 Threshold-based AD

FOQA uses ED as the standard approach, which flags flights when predefined operational limits are exceeded. These limits are typically set by the operator and commonly concern airspeed, rate of descent, and aircraft configuration.

3.3 Clustering-based AD

Clustering-based AD relies on a distance metric, such as the Euclidean distance from Equation 1

$$d_e(\vec{p},\vec{q}) = \sqrt{(p_1-q_1)^2+(p_2-q_2)^2+\ldots+(p_k-q_k)^2} \tag{1}$$

and assumes that anomalous points lie far from dense nominal regions. This idea can be operationalized through clustering, where intra-group similarity is maximized (cohesion), while inter-group similarity is minimized (separation). Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is particularly useful here because it explicitly identifies low-density points as noise [12]. This is why DBSCAN was previously used for FOQA in ClusterAD [7].

The input to a DBSCAN analysis is a two dimensional matrix, in which each row is a feature vector representing one flight. Thus, DBSCAN does not directly consider temporal context. Instead, the time axis is flattened into the one dimensional feature vector.

This resulting vector of input features for every sample is then quite long, which poses a challenge as distance-based methods are vulnerable to the curse of dimensionality [13]. To make a clustering analysis productive, the extensive input vector is reduced using a PCA step, from which only the top principal components that explain 95 % of the variance in the data are kept. Importantly, because PCA finds directions of global co-variation across the time steps, it is expected to preserve meaningful temporal structure even after the dimensionality reduction.

Next DBSCAN assigns each sample either to a cluster or to the noise set. For benchmarking, every sample needs to be assigned a scalar anomaly score. As DBSCAN does not natively produce one, the distance to the closest core point is used as a substitute. A core point is a sample that has at least m points in a radius of ε around it, meaning it is located inside a dense, representative region of the data. Samples located far from any such region therefore receive a high score, which serves as the anomaly indicator.

3.4 Forecasting-based AD

The third approach forecasts future data with a neural network model and evaluates the residuals across all features and steps. A forecasting-based detector therefore consists of a predictor and an analyzer. The forecasting architecture is visualized in Figure 2 with the training data split into subsets ($T1$, $T2$, and $T3$) used to fit the respective models.

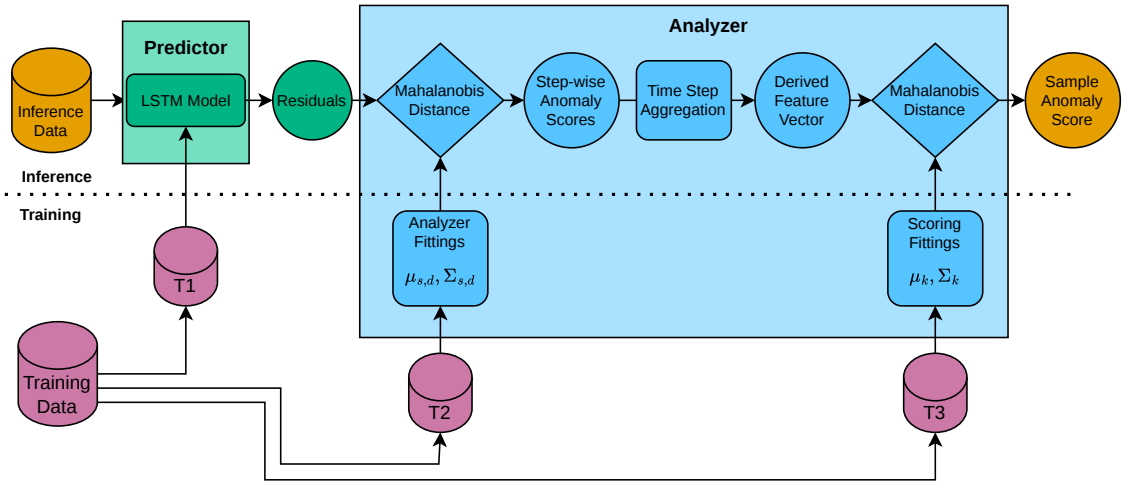


Figure 2 – Overview of the Architecture for the Forecasting Anomaly Detector

The predictor is an LSTM network [14] trained on nominal data, which uses a context window (window size) to forecast the following time steps (prediction horizon). The expectation is that the residual distribution changes when anomalous data are presented during inference.
The analyzer receives the residuals and outputs one scalar anomaly score per sample. First, a step-wise score is computed using the Mahalanobis distance to the mean residual distribution. The Mahalanobis distance d_m, from Equation 2, is a measure for the difference of a multivariate sample vector $\vec{x}$ to a distribution (characterized by a mean vector $\vec{\mu}$ and a covariance matrix Σ):

$$d_m(\vec{x}, \vec{Q}) = \sqrt{(\vec{x} - \vec{\mu})^T \, \Sigma^{-1} \, (\vec{x} - \vec{\mu})}. \tag{2}$$

Unlike the Euclidean distance, the Mahalanobis distance accounts for correlations between features, making it better suited to multivariate data.
The analysis assigns an anomaly score to every time step of every flight. These step-wise scores are then aggregated into a compact set of derived features: the maximum, mean, and standard deviation of the scores, the number of steps exceeding a preset threshold, and the length of the longest consecutive run above that threshold. The threshold used was determined by analyzing the distributions of anomaly scores and placed at a value corresponding to double the standard deviation. Together, these features are designed to capture qualitatively different anomaly patterns like isolated spikes versus sustained deviations. A second Mahalanobis distance is finally applied to map this derived feature vector to a single scalar anomaly score for the flight as a whole.

3.5 Reconstruction-based AD

Reconstruction-based AD follows the same residual-based logic as forecasting AD and uses a neural network as well, which in this case is an autoencoder (AE) instead of a predictor. The autoencoder is trained only on nominal data and consists of an encoder that compresses the input into a low-dimensional latent representation and a decoder that reconstructs the original signal. The bottleneck forces the model to learn the regular structure of the training data, which is expected to reconstruct well for nominal samples, but poorly for outliers [15]. At inference time, the model reconstructs the input time series and the difference to the original input yields residuals for every time step and feature.
The following analyzer is the same as in the forecasting-based detector (see subsection 3.4).

3.6 Feature Optimization and Model Hyperparameter Tuning

To determine the maximum capability of the AD approaches, an effort is made to find highly performant configurations of model hyperparameters. The hyperparameter with the greatest effect on performance for all methods is the choice of in- and if applicable output (target) features. To narrow

down the feature selection, a search is conducted by training models and evaluating their performance on the benchmarking set. The results are shown in Table 1.

Method	Input Features	Output Features
Clustering	altitude above ground, groundspeed, engine N1	—
Forecasting	altitude, groundspeed, pitch, angle of attack, engine N1, elevator position	altitude above ground, distance to touchdown
Reconstruction	altitude above ground, groundspeed, pitch, distance to touchdown, engine N1	—

Table 1 – Data features provided to every method.

Similarly, the hyperparameters for every individual approach are tuned through empirical testing. For the clustering method the neighborhood radius (ε) is chosen through a k-nearest neighbors analysis of the dataset, placing the value for ε so that the top 5 % most isolated samples are excluded and the min. points parameter (m) is chosen to be between 10 % and 20 % of the number of samples in the data set. The hyperparameters for the LSTM and the AE are compiled in Table 2.

Hyperparameter	Value
Hidden	128
Layers	2
Window Size	30
Prediction Horizon	5

(a) Hyperparameters of the forecasting LSTM

Hyperparameter	Value
Hidden Layer 1	384
Hidden Layer 2	192
Latent Layer	16

(b) Hyperparameters of the reconstruction AE

Table 2 – Chosen hyperparameter values for the forecasting and reconstruction models.

The neural networks used for forecasting- and reconstruction AD are not particularly extensive as can be seen by the limited number and sizes of their respective hidden layers.

4. Result Evaluation

This chapter evaluates the performance of the proposed AD methods on the manually labeled benchmarking set, the missed approach set, and unlabeled data. Both labeled data sets exhibit known limitations (see subsection 2.2, subsection 2.3), and model selection was partially based on benchmarking performance, introducing optimistic bias. Reported results on this data set should therefore be interpreted accordingly.

4.1 Evaluation Methods

Binary classification performance is assessed using precision-recall (PR) curves. Each sample is assigned an anomaly score and compared against ground truth labels while sweeping a detection threshold from a maximum to a minimum. The similar and often used Receiver Operating Characteristic (ROC) metrics are omitted in the evaluation due to their limited interpretability under strong class imbalance.

Precision and recall are defined through the relationships of true positive (TP), false positive (FP), and false negative (FN) classifications as

$$P = \frac{TP}{TP+FP}, \quad R = \frac{TP}{TP+FN}. \tag{3}$$

The PR curve captures the trade-off between both metrics across thresholds, from conservative detection (high precision, low recall) to exhaustive detection (high recall, low precision). The area under the PR curve is used to quantify its overall quality in a scalar score (PR-AUC).

In an operational context it might be desirable to maximize recall, which is only viable if precision remains high and sufficient resources are available to evaluate the large number of labeled samples. For operators with limited resources, it may instead be preferable to prioritize high precision to minimize operational costs. This trade-off is quantified using the F_β-measure:

$$F_\beta = (1+\beta^2)\frac{PR}{\beta^2 P + R}. \tag{4}$$

Usually, only the score F_1 ($\beta = 1$) is provided in machine learning evaluations, which is the harmonic mean of precision and recall, both weighted equally. To better represent the scope of demands of FOQA, results are provided for $\beta \in \{0.5, 1, 2\}$, corresponding to precision-oriented, balanced, and recall-oriented regimes. The maximum F_β value along the PR curve is used for evaluation, reflecting optimal threshold selection under different operational priorities.

4.2 Benchmarking Set Performance

Figure 3 shows the PR curves for all detection methods evaluating the benchmarking dataset.

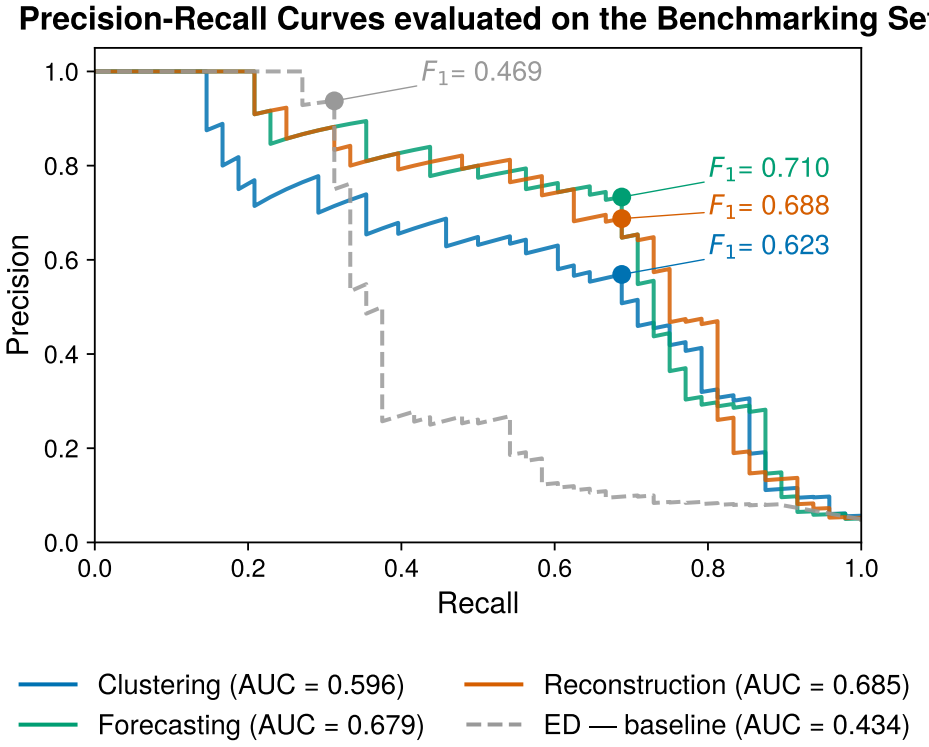


Figure 3 – Precision Recall (PR) Curves obtained by evaluating a benchmark data set using different AD methods.

The PR-AUC is reasonable for all methods, indicating an ability to distinguish between anomalous and normal samples. The PR-AUC for reconstruction and forecasting is very high with values ≥ 0.68, which should be seen critically as the same benchmarking set was previously used to pick the best models for both methods. Due to this bias, the obtained value should be regarded more as an upper limit of the achievable performance.

A number of nominal samples in the benchmarking set are assigned a large anomaly score by every method, leading to a drop and following fluctuation in precision around a recall value of 0.2. This indicates that the algorithms do not find a clear border, but instead a smoother transition from anomalous to nominal samples. ED performs better in this regard, reaching up to 0.3 recall with very high precision before the curve sharply drops.

As shown in Figure 4, forecasting and reconstruction perform similarly across all metrics. The performance of the clustering and the ED methods is clearly lower. Furthermore, the score of the ED method fluctuates significantly with respect to different β values.

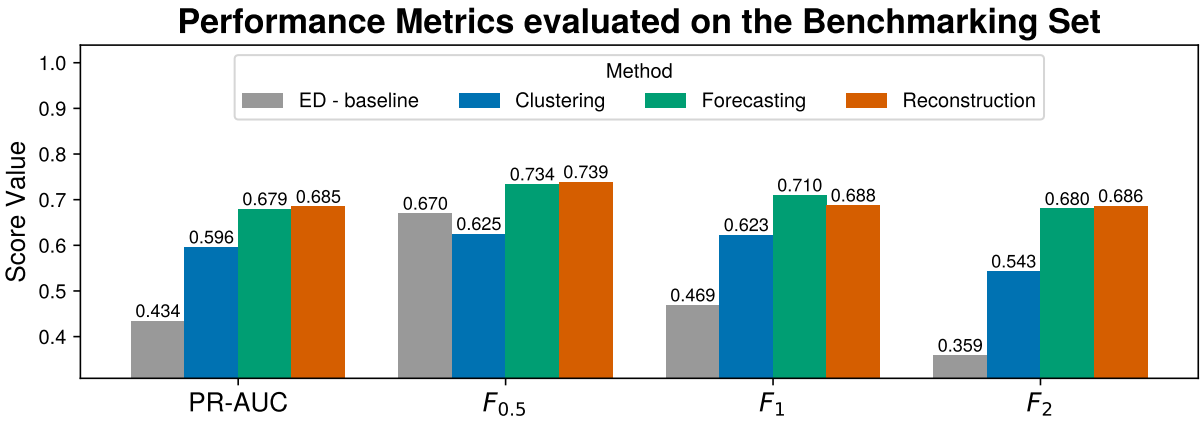


Figure 4 – PR-AUC and F_β performance metrics of anomaly detectors evaluating a benchmarking data set.

Table 3 shows the precision and recall values, at which the respective F_β scores in Figure 4 are achieved. The maxima for $\beta = 0.5$ and $\beta = 1$ for ED are achieved at the same point in the PR-curve. The method, achieving almost perfect precision with a recall of 0.3 highlights how effective this approach can be even though it is conceptually simple. Nevertheless, the highest overall F-scores are reached by the more complex reconstruction and forecasting methods.

	$\beta = 0.5$			$\beta = 1$			$\beta = 2$		
Method	F_β	P	R	F_β	P	R	F_β	P	R
ED - baseline	0.670	0.938	0.312	0.469	0.938	0.312	0.359	0.268	0.542
Clustering	0.625	0.643	0.562	0.623	0.569	0.688	0.543	0.413	0.792
Forecasting	0.734	0.794	0.562	0.710	0.733	0.688	0.680	0.654	0.708
Reconstruction	0.739	0.812	0.542	0.688	0.688	0.688	0.686	0.648	0.729

Table 3 – Performance comparison of AD methods at thresholds maximizing F_β (P and R are the achieved precision and recall values at those thresholds).

4.3 Missed Approach Set Performance

The same detectors are now applied to the missed approach set, which is less structured than the benchmarking set and roughly five times larger. The PR curves in Figure 5 show significantly different results compared to those of the previous benchmarking experiment.

The PR-AUC values are significantly lower than for the benchmarking set evaluation. For the missed approaches, the clustering method achieves the highest score, the forecasting method the lowest. These results indicate that the AD methods have some capability of detecting instabilities in the approach, that could lead to a MA.

The curves of the clustering- and the reconstruction based methods indicate a drop at recall values around 0.4 and 0.3 after a slower decline for the respective methods. This hints at a split in the data, suggesting that roughly 30 % of the MA samples might be easier to detect than the rest. However, it is unclear whether both methods detect the same samples at these lower recall values. The reason for the split might be related either to the severity of the approach instability before the MA maneuver is initiated or to the timing of the decision.

The shape of the PR-curve for the forecasting method is unusual. At very low recall levels (R < 0.1), precision remains low, indicating that many nominal samples receive higher anomaly scores than true MA samples. Precision increases only at lower decision thresholds, suggesting that the model does not reliably rank the most anomalous samples at the top of the score distribution. This is unexpected

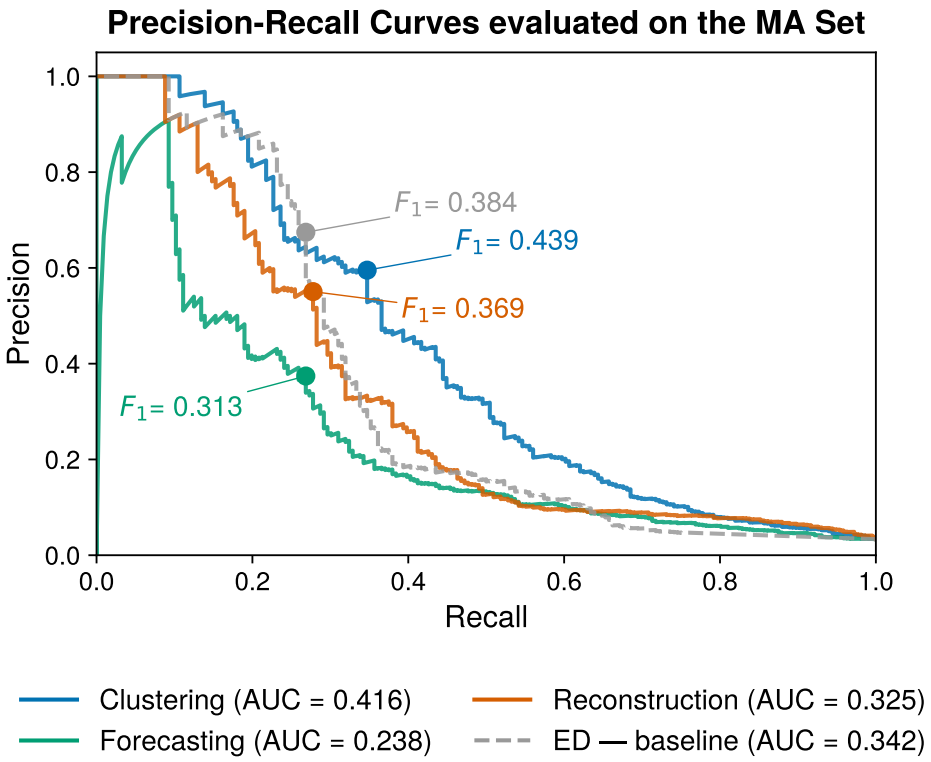


Figure 5 – Precision Recall (PR) Curves obtained by evaluating the set of missed approaches using different AD methods.

behavior and suggests that the forecasting model poorly generalizes to this dataset or perhaps even overfit to the training data.
The shape of the ED method curve is similar to the shape of the previous benchmarking experiment, keeping the same area of high precision up to recall values of 0.3.
Looking at the scalar performance metrics shown in Figure 6, the results match the PR analysis. Compared to the metrics of the benchmarking data set, the performance of the forecasting- and reconstruction methods decreases significantly and that of the clustering method is affected to a lesser extent. ED achieves the best performance score at $\beta = 0.5$, closely followed by the clustering method. For larger values of β, the clustering method achieves the highest score. The forecasting method has the lowest scores across every metric.

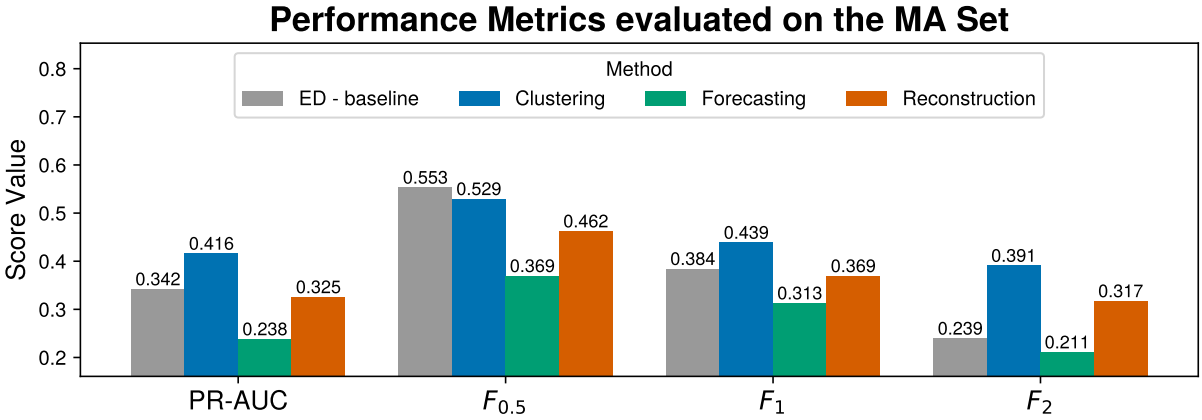


Figure 6 – PR-AUC and F_β performance metrics of anomaly detectors evaluating a set of missed approaches.

Table 4 shows the values for precision and recall at the three operating points. At the precision-

favored point $\beta = 0.5$, both the ED and clustering methods are able to detect roughly 20 % of missed approaches with a precision of 80 %.

	$\beta = 0.5$			$\beta = 1$			$\beta = 2$		
Method	F_β	P	R	F_β	P	R	F_β	P	R
ED - baseline	0.553	0.847	0.231	0.384	0.674	0.269	0.239	0.155	0.523
Clustering	0.529	0.825	0.218	0.439	0.595	0.347	0.391	0.319	0.505
Forecasting	0.369	0.500	0.181	0.313	0.374	0.269	0.211	0.133	0.500
Reconstruction	0.462	0.677	0.204	0.369	0.550	0.278	0.317	0.260	0.407

Table 4 – Performance comparison of AD methods at thresholds maximizing F_β (P and R are the achieved precision and recall values at those thresholds).

4.4 Method Agreement

In a next step, all methods are compared with each other by evaluating a test data set of around 50,000 flights without any labeling. The intent is that truly anomalous samples should produce a signal that can be detected by any method, and that a high disagreement of one individual method could signal a tendency towards false positive results.

In order to ensure comparable conditions across all methods, the AD thresholds are selected such that each method labels the upper quantile of the test dataset as anomalous. Based on the observations made in subsection 2.2, the top 5 % of data were to be considered as anomalous. For forecasting-based AD, reconstruction-based AD, and ED, the threshold is then applied directly to the anomaly scores produced by the respective methods, such that this top percentage of scores is classified as anomalous.

For clustering-based AD, however, the procedure is less straightforward, since anomaly classification is determined by the outcome of the DBSCAN analysis rather than by a continuous anomaly score. Consequently, the number of samples classified as anomalous can only be adjusted indirectly through the selection of the ε hyperparameter. Suitable ε values, which lead to the algorithm labeling roughly 5 % of the data as anomalous, were determined using a k-nearest-neighbor analysis. Nevertheless, the resulting anomaly proportion cannot be matched exactly. While the other methods classify exactly 95 % of the data as anomalous, the proportion obtained with the clustering-based method will always deviate slightly.

as the needs to be fit to the data set, which is done by evaluating a k-nearest neighbor analysis. The result is that the quantiles are not exact for the clustering method.

The absolute quantities of every intersection between the AD methods at a target quantile of 95 % are shown as an UpSet plot in Figure 7. The UpSet plot consists of three subplots: a horizontal bar chart showing the absolute number of samples marked as anomalous for every method (bottom left), and a combination of an intersection visualization (bottom right) with a vertical bar chart, showing the size of the corresponding intersection (top right). A column in the bottom right visualization resembles an intersection. A filled dot in a row for a given method indicates that this method is present in the intersection together with all other filled dots in the same column. The dots are linked, for the case that multiple AD methods are included in the intersection.

Only on 270 out of the 50,000 samples all four methods agree. The largest group by far consists of samples only marked by ED. Reconstruction-based AD has the greatest overlap with the other methods, and classifies only 657 samples that do not match the results of the other methods. The largest agreement between two methods is achieved by the neural-network based forecasting- and reconstruction methods (455). However, by considering all intersections between two or more AD methods, reconstruction and clustering agree on the largest number of samples (1268) compared to 1251 for forecasting- and reconstruction AD although the difference is minimal.

The fact that ED flags a completely different set of flights could be because the method models anomalous conditions fundamentally differently or perhaps because it monitors different features than

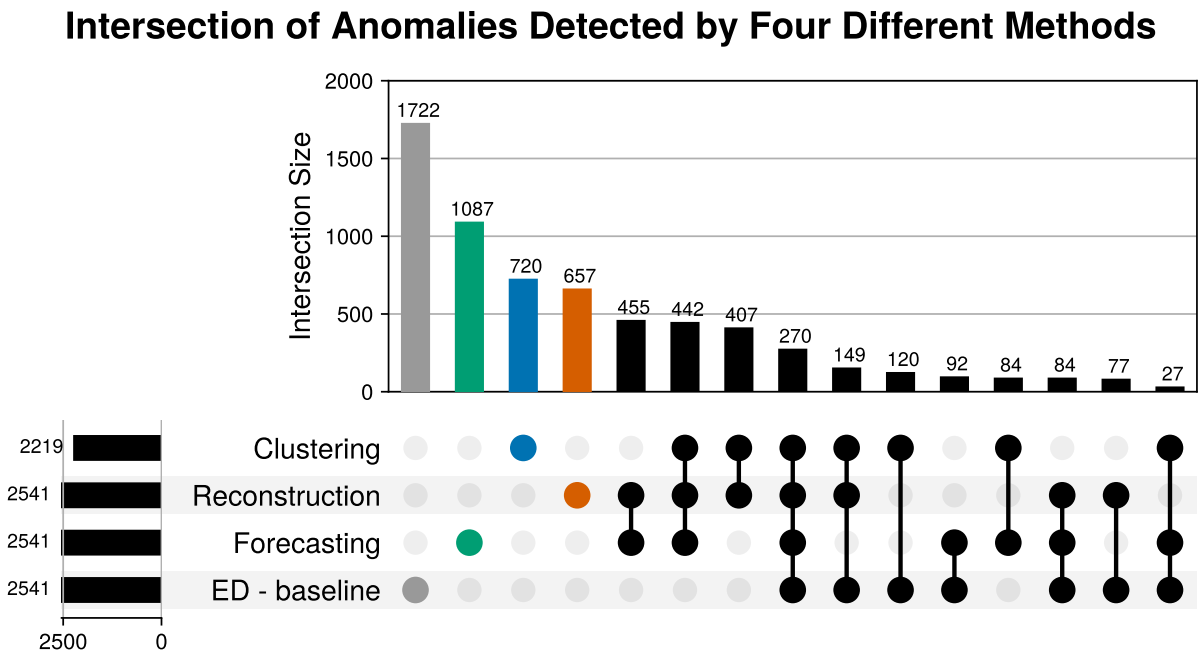


Figure 7 – UpSet plot visualizing the intersection sizes between the anomaly classifications made by four AD methods.

the other three methods. Unlike the other methods, ED is not tuned with respect to the here provided data sets. Instead, the thresholds are placed manually based on observations made over the course of other, regular operations, implicitly incorporating historical data, that none of the other algorithms saw.

The intersection of missed approaches on which all AD methods can agree is small but present, indicating the existence of a number of highly anomalous flights. The disagreement occurs with samples that are in the boundary between nominal and anomalous, where all the methods are individually sensitive to different factors.

The agreement is further formularized using the Jaccard-coefficient in Equation 5, a measure of the intersection between two sets divided by the union of these two sets.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{5}$$

The Jaccard-coefficient ranges from 0 (no overlap) to 1 (complete overlap). The pairwise Jaccard-coefficients are shown in Table 5.

In line with the results from subsection 4.2, the reconstruction and forecasting methods have a large overlap across all quantiles, although even larger values are achieved between reconstruction- and clustering-based AD. On the other hand, ED has a consistently small overlap with all other methods. The values for all Jaccard-coefficients are relatively low, which is expected as every AD method has a different representation of nominal and anomalous conditions.

Quantile	Method	ED	Clustering	Forecasting
Top 10 %	Clustering	0.135	—	
	Forecasting	0.148	0.262	—
	Reconstruction	0.141	0.357	0.383
Top 5 %	Clustering	0.135	—	
	Forecasting	0.103	0.209	—
	Reconstruction	0.129	0.363	0.327
Top 2 %	Clustering	0.129	—	
	Forecasting	0.061	0.109	—
	Reconstruction	0.126	0.306	0.209

Table 5 – Jaccard coefficients between detection methods per quantile labeled as anomalous (Note: Due to the symmetrical nature of the table, a column for the reconstruction method and a row for ED were omitted, as these data were redundant with other cells of the table).

4.5 Explainability

Especially for safety critical systems it is paramount that the employed algorithms deliver explainable and deterministic results. For FOQA it should be clear, why a sample was labeled so that post-processing is streamlined. ED can precisely indicate when a parameter has crossed a threshold, making investigations of any event quick. The neural network based methods lay the foundation for a similar capability, as intermittent residuals are available in a time- and feature-discrete manner. However, implementing an evaluation method that correctly interprets these residuals into accurate violation warnings is an extensive task. The challenge with clustering is similarly complex. Fundamentally, it is possible to determine the distance of a sample to the centroid with respect to every parameter and identify the dominating outlier features. But implementing such a method and validating its functionality is not simple.

5. Discussion

A consistent observation across multiple analyses showed the similarity between the two neural network based AD methods. This is not surprising, as the architectures share many commonalities. At the same time, the reconstruction-based approach performed marginally better in benchmarking and significantly better in detecting MAs. A potential reason for this might be related to difficulties during setup and training of the respective models. Developing a capable forecasting model was difficult and possibly led to overfitting on the training data that was not recognized during previous evaluations. At the same time, the process for the reconstruction AE was much smoother.

The clustering approach showed the smallest difference in performance between the benchmarking and the MA data sets. This could again be due to the setup methodology, as the forecasting- and reconstruction models were chosen with respect to their performance on the benchmarking set, while no such evaluation was done for the clustering model. Furthermore, the clustering implementation is fully unsupervised, which means that the method should inherently be more transferable to other datasets than deep learning-based methods.

The advantages of ED were clearly highlighted to be great precision at low recall. Surpassing the state of the art method with regards to performance was shown to be possible, but much more difficult than anticipated. Beyond performance alone, the usability of a fully-explainable, threshold-based AD method in a safety critical system like FOQA should be mentioned here. With ED it is possible to immediately identify what went wrong in an approach and why exactly it was labeled a certain way. Should a new method be used for FDM, a similar capability needs to be implemented for it to be a serious competitor to ED. The proposed neural network based methods provide the foundation for such a capability, as they abstract all samples to a continuous score as part of their architectures. An anomaly score could yield a kind of "severity ranking", enabling a move towards an algorithm, that similarly to ED now does not just mark flights as anomalous but also ranks them based on how

critical they are. However, with the current methods, the correlation between the assigned score and the actual severity has yet to be proven.
The overlap analysis showed that the machine learning methods are able to identify different samples than ED is, highlighting the short-comings of the state of the art approach. The fact that all three of the more complex methods agreed on a significant amount of samples that were not detected by ED indicates the presence of actual anomalous flights that are potentially safety critical but difficult to detect using current methods.
Beyond the detection performance, the differences in setup- and run-time between the methods are notable. The forecasting-based method required the most effort to set up and optimize. Developing a proficient prediction model was a lengthy and iterative process. The model used in the end is much more complex and computationally expensive than the model used for reconstruction-based AD, leading to longer training times, in turn lengthening the setup and optimization process. Similarly, its complexity leads to slower evaluation. In processing the set of 50,000 flights, the forecasting method took roughly 25 minutes to evaluate, the clustering-based method took around 3 minutes, and both reconstruction-based AD and ED took less than 1 minute (the intent is to provide a rough, qualitative comparison, so no statement about the exact machine is made).
The clustering AD method with the implementation used in this paper requires some manual hyper-parameter tuning for every new supplied data set, whereas the other methods can be applied to new data relative easily.

6. Conclusion and Outlook

This paper showed a selection of machine learning methods that can be used to detect anomalous flights in a fully unlabeled data set. For this purpose, multiple evaluation methods were used that were tailored to working with limited labeling information. The results showed that improving on the state of the art ED method is possible but not trivial. At the same time, it was shown that there is room for improvement that could lead to a safer and more comprehensive FOQA system. ED is limited to predefined conditions and cannot identify novel risk patterns. In contrast, the proposed methods generalize beyond known scenarios and justify a controlled reduction in precision in exchange for improved detection of previously unseen anomalies. However, before machine learning can be used in such a system, it needs to be made as explainable as possible. A next step might be to determine, in how far the anomaly scores assigned by the employed AD methods actually correlate with the severity of a critical event. This way, evaluation algorithms could be developed that show, which parameter led to an anomaly labeling.

7. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

References

[1] European Union Aviation Safety Agency. Commission regulation (eu) no 965/2012. https://eur-lex.europa.eu/eli/reg/2012/965/oj, October 2012. Accessed: 2026-18-05.

[2] Federal Aviation Administration. Ac 120-82: Flight operational quality assurance. Advisory Circular AC 120-82, U.S. Department of Transportation, Federal Aviation Administration, April 2004. Accessed: 2025-12-03.

[3] Milad Memarzadeh, Bryan Matthews, and Ilya Avrekh. Unsupervised anomaly detection in flight data using convolutional variational auto-encoder. *Aerospace*, 7(8), 2020.

[4] Airbus. A statistical analysis of commercial aviation accidents 1958–2024. https://accidentstats.airbus.com/wp-content/uploads/2025/02/20241325_A-Statistical-analysis-of-commercial-aviation-accidents-2025-links.pdf, 2025. Accessed: 2025-12-03.

[5] Tim Großkopf. Analyse der Fahrwerksextraktion von Verkehrsflugzeugen im Anflug auf verschiedene Flughäfen in Bezug auf den Abstand zur Schwelle sowie der Höhen- und Geschwindigkeitsprofile. Bachelor's thesis, Hochschule Bremen, Fakultät Natur und Technik, 2024.

[6] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proc. VLDB Endow.*, 15(9):1779–1797, May 2022.

[7] Lishuai Li, Santanu Das, R. John Hansman, Rafael Palacios, and Ashok N. Srivastava. Analysis of flight data using clustering techniques for detecting abnormal operations. In *AIAA Conference Publication*, Palo Alto, CA; Cambridge, MA; Madrid, Spain; Hong Kong, 2015.

[8] Anvardh Nanduri and Lance Sherry. Anomaly detection in aircraft data using recurrent neural networks (rnn). In *Integrated Communications Navigation and Surveillance (ICNS) 2016*, pages 5C2–1–5C2–8, 2016.

[9] Ian T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. Springer-Verlag, New York, 2002.

[10] Hiroaki. Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978.

[11] John Paparrizos and Luis Gravano. k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, SIGMOD '15, page 1855–1870, New York, NY, USA, 2015. Association for Computing Machinery.

[12] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231. AAAI Press, 1996.

[13] Lance Parsons, Ehtesham Haque, and Huan Liu. Subspace clustering for high dimensional data: a review. *SIGKDD Explor. Newsl.*, 6(1):90–105, June 2004.

[14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[15] Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu Aggarwal, and Mahsa Salehi. Deep learning for time series anomaly detection: A survey. *ACM Computing Surveys*, 57(1):1–42, 2024.