



Detection and identification of significant events in historical aircraft trajectory data

Xavier Olive, Luis Basora

► To cite this version:

Xavier Olive, Luis Basora. Detection and identification of significant events in historical aircraft trajectory data. Transportation research. Part C, Emerging technologies, 2020, 119, pp.102737. <10.1016/j.trc.2020.102737>. <hal-02972531>

HAL Id: hal-02972531

<https://hal.science/hal-02972531v1>

Submitted on 20 Oct 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Detection and identification of significant events in historical aircraft trajectory data

Xavier Olive^a, Luis Basora^a

^aONERA DTIS, Université de Toulouse, 2 avenue Édouard Belin, 31055 Toulouse cedex 4, France

Abstract

A large amount of data is produced every day by stakeholders of the Air Traffic Management (ATM) system, in particular airline operators, airports, and air navigation service provider (ANSP). Most data is kept private for many reasons, including commercial and security concerns. More than data, shared information is precious, as it leverages intelligent decision-making support tools designed to smooth daily operations.

We present a framework to detect, identify and characterise anomalies in past aircraft trajectory data. It is based on an open source of ADS-B based aircraft trajectories, and extracted information can benefit a wide range of stakeholders: Air Traffic Control (ATC) training centres could play more realistic simulations; ANSP may improve capacity indicators; academics improve safety models and risk estimations; and commercial stakeholders, like airlines and airports, may use such information to improve short-term predictions and optimise their operations.

The technique is based on autoencoding artificial neural networks applied on flows of trajectories, which provide a useful reading grid associating cluster analysis with quantified level of abnormality. In particular, we find that the highest anomaly scores correspond to poor weather conditions, whereas anomalies with a lower score relate to ATC tactical actions.

Keywords:

ADS-B, trajectory data analytics, air traffic flows, anomaly detection, air traffic management, machine learning, autoencoders

*Corresponding author

Email address: xavier.olive@onera.fr (Xavier Olive)

1. Introduction

The identification of significant events in historical aircraft trajectory data falls within the scope of knowledge discovery and information extraction. It focuses on getting value out of the huge amount of data produced every day by stakeholders of the air traffic management (ATM) system. It is an arduous task as trajectories flown every day by commercial, military, private and other aircraft are subject to a large amount of impacting factors commonly modelled as *uncertainty* and causing deviation from the plan in terms of predicted times at given milestones along trajectories from gate to gate.

Sources of uncertainty in air traffic management are manifold. Weather-related uncertainties cause deviation from the plan due to a strong tailwind, deviations due to thunderstorm activities (also referred to as CB, for *cumulonimbus*, in the following) or disruptions when airport operations are interrupted. Air Traffic Control (ATC) staffing, late maintenance issues, airport operations including late boarding causing late startup time also introduce uncertainties in the whole aviation system.

Some of these sources of uncertainties can be predicted, quantified or logged but as the information produced by each stakeholder is not necessarily made available to all stakeholders, a systematic analysis of correlations between events and of their impact in the air traffic network is a considerable challenge.

This contribution presents a successful Machine Learning (ML) approach for anomaly detection and identification of significant events in past trajectory data. The method builds on top of open sources of aircraft trajectory data, as received by a network of receivers of Automatic Dependent Surveillance–Broadcast (ADS-B), stored on and requested from The OpenSky Network [1]. Our approach is based on trajectory data only, although external sources of not necessarily open data were used to help assessing and interpreting detected situations.

Detected significant events on a wide range of study cases include strong deviations (weather disruptions, regulations) and ATC actions, such as deconfliction and sequencing actions. The proposed method complements well with clustering techniques to identify air traffic flows and highlights trajectories within flows as presenting a degree of dissimilarity important enough to justify further analysis.

Detecting and identifying significant events is a valuable asset to address many applications, including safety analyses, preparation of ATC training simulations with realistic situations and feeding collaborative decision making (CDM) tools with operational information coming from the observation of past data in order to improve their acceptability. In particular, identifying traffic situations which could have led to a separation loss, analysing context and proximate events are valuable inputs for collision risk models (CRM) [2], resulting in estimations of mid-air collision risks. Locating hot spots based on identified situations could also

40 help assess the complexity of traffic inside a given sector when it is in operation.

41 The evaluation of our methodology is performed on several months of data
42 at various scales: first with a focus on routes between city pairs, i.e. trajectories
43 flying from city A to city B; then looking from a sector point of view, coupled
44 with periods when the sector was operationally deployed according to the Sector
45 Configuration Plans (SCP). Finally, we have a closer look at Terminal Manoeuvring
46 Areas (TMA) and have a look at internal layers of autoencoder neural networks
47 to understand what kind of representation is learned before being able to detect
48 significant events.

49 The paper is organised as follows. Section 2 reviews the state of the art with
50 respect to trajectory clustering, anomaly detection and their application to safety
51 analyses. Section 3 presents the formalism, based on an extensive use of a special
52 kind of artificial neural networks called autoencoders. Section 4 presents various
53 use cases and analyses detected and labelled situations. Section 5 provides insight
54 on the representation that is produced by our neural networks through the learning
55 process.

56 **2. Background and literature review**

57 Detection and identification of significant events in large amounts of historical
58 aircraft trajectory data is an active area of research. What a significant event means
59 depends ultimately on the specific use case addressed. In general, it encompasses
60 operational situations or pattern behaviours in air traffic which are worth identifying
61 for online tracking or post-operational analysis because of their potential
62 impact on the safety or efficiency of air traffic operations.

63 The methodology described in this paper builds on two previous research efforts
64 [3, 4] based on trajectory clustering and anomaly detection techniques. The
65 most relevant research on these two areas will be introduced here in the corresponding
66 subsections.

67 Trajectory clustering is useful to identify the air traffic flows within an airspace.
68 The major and recurrent flows correspond to the standard modes of operation in an
69 airspace, such as Instrument Landing Approaches (ILS) procedures in TMA or
70 the main ATS routes linking the defined entering and exiting points in an en-route
71 sector. Once the standard modes of operation are determined, it is possible to
72 detect in a more precise way (by using other techniques) specific anomalous traffic
73 patterns. At the trajectory level, concrete examples of these significant events will
74 be given as part of the literature review, but we are generally referring to flight
75 trajectories or flight interactions within or among the main flows with a certain
76 degree of unexpected behaviour. At the flow level, although it is not the focus of

77 our approach, non-nominal flow configurations in a certain airspace at certain time
78 could also be considered as examples significant events.

79 *2.1. Trajectory clustering and flow identification*

80 Clustering is an unsupervised data analysis technique widely used to group
81 similar entities into clusters according to a similarity, or distance, function. Multi-
82 ple clustering algorithms exist in the literature to cluster point-based data such as
83 k-means [5], BIRCH [6], OPTICS [7], DBSCAN [8] or H-DBSCAN [9]. When
84 clustering is applied to trajectories, it requires a proper distance function to be
85 defined between trajectory pairs, which is challenging because of the functional
86 nature of trajectories. The most commonly used approach is to simply sample the
87 trajectory so as to obtain a n -dimensional vector of points for the use of point-based
88 clustering algorithms and distances such as the Euclidean one.

89 When trajectories are represented in the form of high-dimensional vectors, it is
90 good practice to reduce their dimensionality before clustering by projecting trajec-
91 tory data in high-dimensional space to a space with fewer dimensions. Numerous
92 linear or non-linear data transformation techniques exist to reduce the dimension-
93 ality. Principal Component Analysis (PCA) is a widely used linear transformation
94 technique to project data into a lower-dimensionality space so that the variance
95 in the low-dimensional space representation is maximised. Kernel PCA is a non-
96 linear variant of PCA which is based on kernel methods. Autoencoders are a cate-
97 gory of neural networks capable of learning non-linear reduction functions as well
98 as their inverse functions to transform data into the low dimension and back into
99 the original representation.

100 Trajectory clustering methods based on Euclidean distance do not always pro-
101 duce satisfying results, especially when applied to trajectories with different lengths.
102 Fortunately, more specific distances for trajectory and time series exist in the lit-
103 erature [10]. For instance, warping-based distances such Dynamic Time Warping
104 (DTW) [11], Longest Common Subsequence (LCSS) [12], Edit Distance on Real
105 Sequences (EDR) [13], Edit Distance with Real Penalty (ERP) [13] find an opti-
106 mal way of aligning the time dimension of trajectories to achieve a perfect match
107 between them. Other distances exist to better take into account the geometry of
108 the trajectories and their shape. The most well known shape-based distances are
109 Hausdorff [14] and Fréchet [15], but they are not appropriate to compare trajec-
110 tories as a whole. More recently, a more promising shape-based called Symmetrized
111 Segment-Path Distance (SSPD) distance has been proposed [10, 16] which takes
112 into consideration several trajectory aspects: the total length, the variation and the
113 physical distance between two trajectories.

114 A significant number of clustering methods exist in the literature for flow iden-
115 tification, where the goal is to determine the set of clusters that best fit the opera-

116 tional air traffic flows within an airspace. These methods associate each cluster to
117 an air traffic flow which is a traffic pattern with both temporal and spatial charac-
118 teristics. The exact definition of what a flow is ultimately depends on the specific
119 application context. Also, it is worth noticing that a majority of the research have
120 focused primarily on studying the spatial dimension of flows.

121 Some flow identification methods are developed for analysis of air traffic op-
122 erations in the terminal area. For instance, Eckstein [17] combines PCA with
123 k-means to evaluate the performance of individual flights in TMA procedures.
124 Gariel et al. [18] propose a framework based on PCA, DBSCAN and k-means for
125 airspace monitoring and complexity assessment also in the TMA. Rehm [19] and
126 Enriquez [20] apply hierarchical and spectral clustering techniques to identify air
127 traffic flow patterns from and to an airport. Murça et al. [21] present a framework
128 based on DBSCAN and other algorithms to identify and characterise air traffic
129 flow patterns in New York transition/terminal airspace for daily assessment of the
130 tactical air traffic operations.

131 More recently, Murça et al. [22] present a framework based on DBSCAN and
132 other machine learning techniques to characterise traffic flow patterns and compare
133 the terminal airspace design as well as the operational and performance differences
134 in three multi-airport systems (New York, Hong Kong and São Paulo). Olive [23]
135 propose a specific clustering technique for identifying converging flows in the ter-
136 minal area of Toulouse which helps understand how approaches are managed. The
137 analysis of outliers provides elements to understand and assess specific situations
138 calling for more in-depth safety analyses.

139 Other research focus on flow identification within en-route airspace. Sabhnani
140 et al. [24] present a specific clustering algorithm to identify flow patterns and criti-
141 cal points for en-route airspace redesign. Basora et al. [25] propose a framework for
142 the analysis of air traffic flows based on trajectory clustering with the HDBSCAN
143 algorithm [25] and the SSPD distance. Finally, flow identification have been per-
144 form on a larger scale to characterise air traffic route networks [26, 27, 28, 29].

145 2.2. *Anomaly detection*

146 Anomaly detection refers to the problem of finding patterns in data that do not
147 conform to expected behaviour [30]. Anomaly detection techniques can be used
148 to detect significant events in flight data, as these usually correspond to unusual
149 operational situations and so presenting a certain degree of anomaly. Go-around
150 operations, runway excursions, conflict resolution manoeuvres and traffic rerouting
151 are just a few examples of significant events that could be identified by applying
152 anomaly detection methods.

153 A recent survey [31] reviews some of the main anomaly detection techniques
154 and their application to aviation data. The review covers an exhaustive number of

155 use cases and methods for the identification of significant events which is too large
156 to be reproduced here again. Instead, we will focus only on the most relevant and
157 recent research work in the field.

158 MKAD [32] can be considered as one of the first methods designed to effec-
159 tively detect operationally significant anomalies with heterogeneous sequences of
160 both discrete and continuous variables. Based on kernel functions and OC-SVM,
161 MKAD can identify operational situations in flight data such as anomalous flight
162 approaches. More recently, Janakiraman and Nielsen [33] propose an unsupervised
163 anomaly detection approach based on extreme learning machines as an alternative
164 to MKAD for the identification of safety risks in very large aviation datasets.

165 Approaches based on clustering algorithms have been widely used for anomaly
166 detection. For instance, Li et al. [34] propose a cluster-based anomaly detection
167 method based on DBSCAN (ClusterAD) to detect anomalies in a airline dataset
168 for 365 B777 take-off and approach operations. Following up this research, Li et
169 al. [35] present a method based also on DBSCAN called ClusterAD – Flight to
170 identify abnormal flights during take-off or approach as a whole.

171 More recently, Li et al. develop ClusterAD – DataSample [36] which is a
172 Gaussian Mixture Model (GMM) based method capable of instantaneously detect-
173 ing abnormal data samples during a flight rather than abnormal flights as a whole
174 during a specific flight phase. Puranik and Mavris [37] present a generic methodol-
175 ogy based on DBSCAN and SVM for identifying anomalies from general-aviation
176 in the approach and landing phase. Olive et al. [38] propose a method based on
177 functional principal component analysis, HDBSCAN – GLOSH [39] to identify
178 atypical approaches and landings both in post-operational analysis and on-line.

179 Deshmukh et al. [40] propose a temporal logic based anomaly detection al-
180 gorithm (TempAD) applicable to trajectories in the terminal airspace. The algo-
181 rithm, based on a temporal-logic learning approach [41, 42, 43], can learn human-
182 readable mathematical expressions from data which facilitates the feedback and
183 interaction with operational experts. The method uses DBSCAN as a preprocess-
184 ing step to identify the clusters with similar trajectories on which the detection of
185 anomalies with TempAD becomes more effective.

186 2.3. *Significant events and safety analysis*

187 In the two previous subsections, we have presented some techniques to auto-
188 matically discover significant events from a dataset of flight data. In this subsec-
189 tion, we will focus on a few specific examples of what kind of significant events
190 can be identified and how they can be used for safety analysis.

191 It is worth noticing that significant events automatically discovered by an algo-
192 rithm do not necessarily present a safety risk (or not in isolation anyway), which is

193 the reason why these events need to be further analysed by a safety expert. How-
194 ever, the frequent occurrence of certain significant events like deconfliction mea-
195 sures within an area of an airspace can create hotspots leading to high complexity
196 and potentially higher safety risk. Therefore, not only the identification of individ-
197 ual events but also their frequency and geographical density are important from a
198 safety point of view.

199 An efficient method for identification of significant events from flight data can
200 be beneficial for a collision risk model (CRM) to improve safety level estimation.
201 For instance, Garcia [2] and Saéz et al. [44] develop a CRM for Eurocontrol to as-
202 sess European en-route airspace safety risk. The detection of what they call proxi-
203 mate events (situations where the aircraft involved may evolve towards a collision
204 due to a separation minima infringement) is performed based on either a track seg-
205 mentation technique or the simulation of the traffic evolution via a time step and
206 look-ahead time parameters [45]. Our methodology could be useful in such con-
207 text by providing an alternative mean of detecting changes in trajectories associated
208 with aircraft manoeuvres or flight interactions corresponding to potential conflicts.

209 However, the major focus of the past research has been put on identifying sig-
210 nificant events for an individual flight during a specific flight phase, especially
211 during the approach, landing and take-off phases because of their inherent higher
212 safety risks. For instance, MKAD [32] can identify significant events during the
213 approach phase such as go-around operations, unusually high airspeed flights at
214 low altitude, flights impacted by gusty winds and abnormal approaches. Clus-
215 terAD [34] can identify operational situations including high/low energy approaches,
216 unusual pitch excursions, abnormal flap settings and high wind conditions. Clus-
217 terAD – DataSample [36] is able to detect several cases of unstable approaches,
218 strong tailwind and late localizer intercept, just to mention a few examples. Un-
219 fortunately, these approaches work with on-board recorded data whose access is
220 restricted and needs authorisation from the airlines.

221 **3. Methodology**

222 *3.1. Autoencoders*

223 Autoencoders are artificial neural networks consisting of two stages: encoding
224 and decoding. A single-layer autoencoder (Figure 1) is a kind of neural network
225 consisting of only one hidden layer. Autoencoders aim at finding a common feature
226 basis from the input data. They reduce dimensionality by setting the number of
227 extracted features to be less than the number of inputs. Autoencoder models are
228 usually trained by backpropagation in an unsupervised manner. The underlying
229 optimization problem aims to minimize the distance between the reconstructed
230 results and the original inputs.

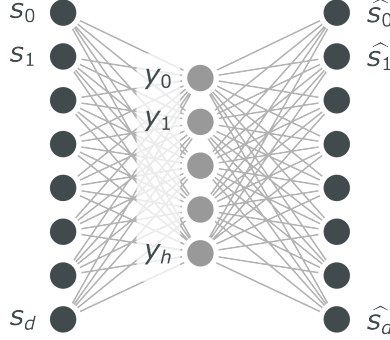


Figure 1: Autoencoder neural network architecture with one layer

231 The encoding function of an autoencoder (such as the one depicted in Figure 1)
 232 maps the input data $s \in \mathbb{R}^d$ to a hidden representation $y \in \mathbb{R}^h = e(s) = g(w \cdot s +$
 233 $b)$ where $w \in \mathbb{R}^{d \times h}$ and $b \in \mathbb{R}^h$ are respectively the weight matrix and the bias
 234 vector and $g(\cdot)$ is a non linear activation function such as the sigmoid or hyperbolic
 235 tangent functions. The decoding function maps the hidden representation back to
 236 the original input space according to $\hat{s} = d(y) = g(w' \cdot y + b')$, $g(\cdot)$ being most of
 237 the time the same activation function.

238 The objective of the autoencoder model is to minimise the error of the recon-
 239 structed result:

$$(w, b, w', b') = \operatorname{argmin} \ell(s, d(e(s))) \quad (1)$$

240 where $\ell(u, v)$ is a loss function determined according to the input range, typically
 241 the mean squared error (MSE) loss:

$$\ell(u, v) = \frac{1}{n} \sum ||u_i - v_i||^2 \quad (2)$$

242 Depending on the volume and complexity of the considered data set, autoen-
 243 coders' layers may be stacked (see Figure 2) in an attempt to better grasp the struc-
 244 ture of the underlying traffic. Other loss functions, such as cross-entropy loss, or
 245 custom loss functions with regularisation terms (see Section 3.3) may be designed
 246 for a better robustness and stability of the results across executions of the method.

247 3.2. Distribution of reconstruction errors and evidence of convergence

248 Autoencoders are trained to project, or compress, data onto a latent space of a
 249 smaller dimension, then to regenerate the original data in the original space based
 250 on a representation in a smaller dimension. The rationale behind the use of autoen-
 251 coders for anomaly detection is that these artificial neural networks specialise on
 252 finding a representation of the data with few parameters, just enough to reconstruct

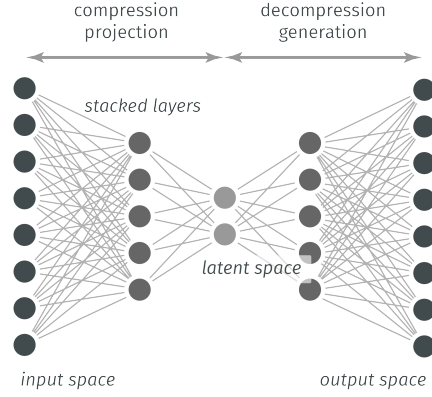


Figure 2: Autoencoder neural network architecture with stacked layers

the original data in most cases. They are trained to minimise a loss function, and learn to reconstruct the original data, at the risk of not being able to reconstruct anomalous samples.

Figure 3 plots the evolution of the MSE loss together with the distribution of reconstruction errors across all the samples contained in a dataset. Details about the specificity of the dataset are not relevant at this point: we will introduce them in depth in Section 4.

After one iteration, i.e. with random weights on the edges of the neural network, reconstruction errors present an initial distribution which starts moving towards lower reconstruction errors after 10 or 100 iterations. After 1000 iterations, although the convergence is obviously not complete, we see that most samples distribute close to 0 and few samples reconstruct with higher residual errors.

Within the unsupervised learning paradigm, after the loss function converges, we have no way to check the accuracy of our network, the authors recommend to plot this distribution as a rule of thumb to verify the convergence of the training process and the relevance of the resulting autoencoder.

3.3. A new term of regularisation

In some of the datasets introduced in Section 4, we found that several runs of the training phase of our autoencoders would distribute along two variation modes. Figure 4 displays such an example of distribution when variation modes of unbalanced weights exist in the original dataset: these modes may subsist in the distribution of reconstruction errors in the form of a distribution with two "hills".

In order to limit a premature optimisation of the autoencoder which would learn to favour one mode over the other (the one with the more samples) in some runs of

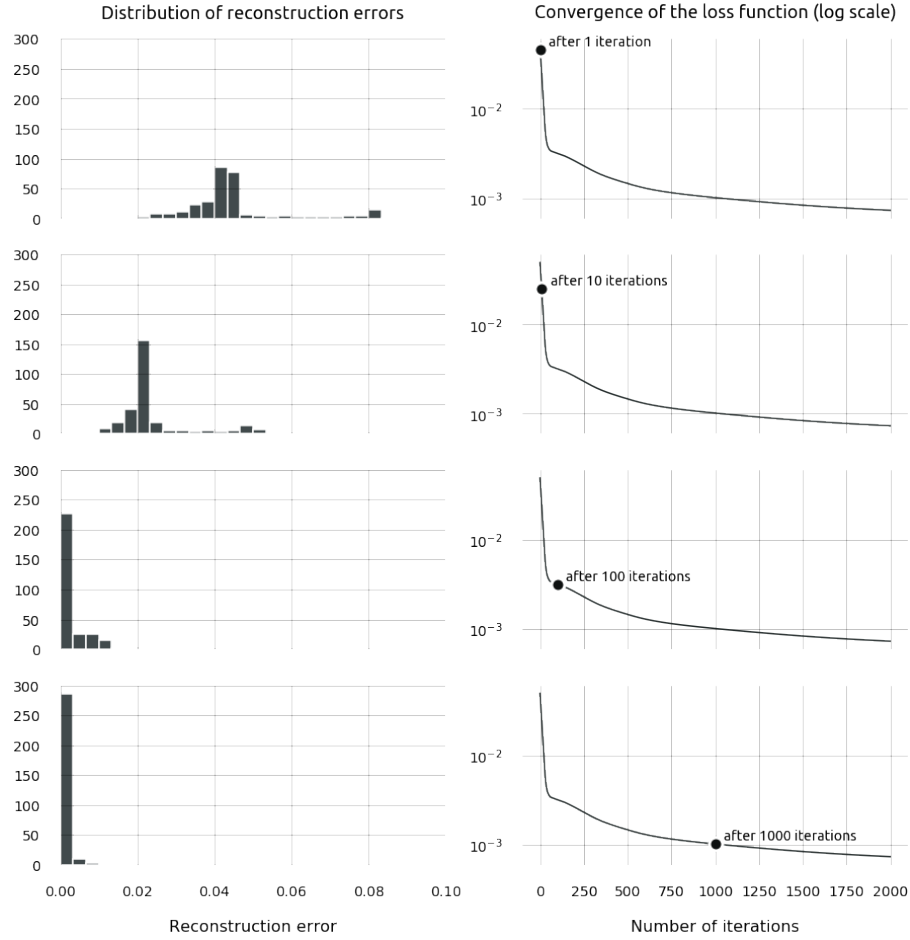


Figure 3: Distribution of reconstruction errors during the training of the autoencoder. The sample dataset used to generate this figure is presented in Section 4.

our algorithm, we improved the training of our network with a regularisation term added to the loss $\ell(u, v)$.

A regularisation term is a penalty term added to the loss that is commonly used to prevent overfitting. Among neural networks, L1- (resp. L2-) regularisations penalize the loss after each iteration with the sum of the absolute (resp. squared) weights of the neural network. In our case, since we expect a distribution of reconstruction errors that would fit an exponential law, we propose a regularisation term based on a measure of distance between distributions.

After each iteration, we fit an exponential law to our distribution of reconstruction errors. The best fit to an exponential distribution can be written based on the mean of all $\rho_i = \|u_i - v_i\|^2$ samples, which have already been computed in form of the MSE loss $\ell(u, v)$. Therefore, the best fit for the probability density function becomes:

$$f : x \mapsto \frac{1}{\ell(u, v)} \cdot e^{-\frac{x}{\ell(u, v)}} \quad (3)$$

Then, we compute the distance between the distribution of reconstruction errors and the fitted exponential probability density function. For each $t_j \in [0, \max(\rho_i)]$ equally sampled with $j \in [1, m]$, we evaluate the difference:

$$\delta_j = \left(\frac{1}{n} \sum_i K_{\rho_i}(t_j) \right) - f(t_j) \quad (4)$$

with K_{ρ_i} a Gaussian Kernel function:

$$K_{\rho_i}(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2} \left(\frac{x - \rho_i}{\sigma} \right)^2} \quad (5)$$

Finally, we sum all the δ_j as a regularisation term to the original square loss. For the specific example of Figure 4, with a bandwidth term $\sigma = 0.02$, we found $\lambda = 10^{-4}$ to be particularly efficient as it helped converging to a better MSE loss than the one obtained without regularisation.

$$\ell^*(u, v) = \ell(u, v) + \lambda \sum_{j=1}^m \delta_j \quad (6)$$

We performed several training runs with the same dataset and parameters. We observed that the regularisation term help balance bias and variance resulting in more stable and predictable results which hopefully should prevent model overfitting and help staying away from multimodal distributions.

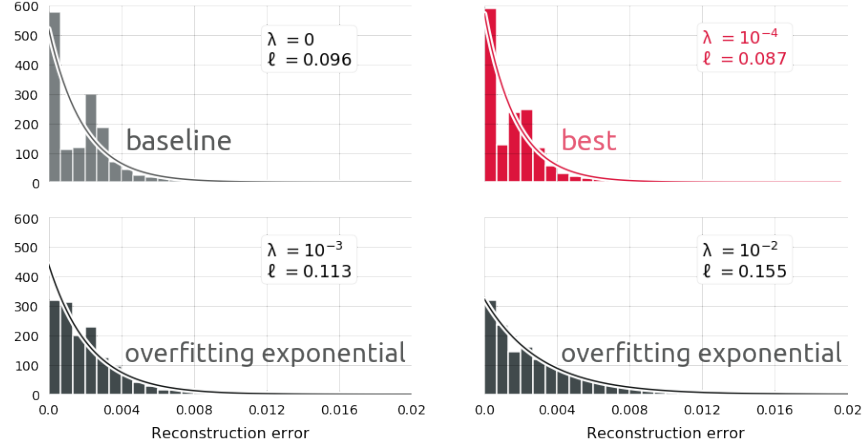


Figure 4: Distribution of p_i , suggesting two modes of variations, and their fitted exponential law: the regularisation term aims at fitting the distribution under the curve while the MSE loss aims at pushing the bins toward zero. When the regularisation is prevalent (overfitting on the exponential distribution), all the bins fit under the curve but samples reconstruct poorly with no added value for anomaly detection.

The differentiation of the regularised loss $\ell^*(u, v)$, necessary to implement the gradient descent and backpropagation during the training period has been delegated to the autograd module of PyTorch [46]. All terms presented in this subsection can be written with torch functions which provide all that is needed for backpropagation.

4. Application to the identification and detection of significant events

4.1. Description of the use cases

We considered in this paper a series of particular use cases. The first proof of concept of this method has been made on a set of city pairs [3], before being extended to flows in an airspace [4]. In this paper, we present a novel use case in the Terminal Manoeuvring Area (TMA) around Zurich airport (LSZH) which we will explore further in Section 5.

Automatic Dependent Surveillance–Broadcast (ADS-B) is a cooperative surveillance technology which provides situational awareness in the air traffic management system. Aircraft determine their position via satellite, inertial and radio navigation and periodically emit it (roughly one sample per second) with other relevant parameters to ground stations and other equipped aircraft. Signals are broadcast at 1090 MHz: a decent ADS-B receiver antenna can receive messages from cruising

320 aircraft located up to 400 km far away, while the range is much lower for aircraft
321 flying in low altitude or on ground.

322 The data used for this study is collected by the OpenSky Network [1], a net-
323 work of ADS-B receivers, which offer querying capabilities on their database for
324 academics. Recorded data contains timestamps (added on the receiver side, with
325 many receivers equipped with a GPS nanosecond precision clock), transponder
326 unique 24-bit identifiers (icao24), space-filled 8 character callsigns, latitude, lon-
327 gitude (in degrees, 5 digit precision), (barometric) altitude (in ft, w.r.t. standard
328 atmosphere), GPS altitude (in ft), ground speeds (in kts), true track angle (in de-
329 grees), vertical speed (in kts). Based on sensors located on the landing gear, differ-
330 ent messages with similar information are sent when the aircraft is on the ground,
331 resulting in a boolean flag in the OpenSky records.

332 Aircraft compute their position within uncertainty margins they are able to es-
333 timate and broadcast. These information are not provided decoded in the OpenSky
334 Network database but could be processed [47] from the raw messages on an as-
335 need basis. However, we kept the uncertainty analysis out of the scope of this
336 paper and chose to manually filter irrelevant data as part of the preprocessing step.

337 The *city-pair* data set consists of one full year of 3536 trajectories flying from
338 Paris–Orly (LFPO) to Toulouse–Blagnac (LFB0) airports between January and De-
339 cember 2017. This data set has been requested based on a set of 28 callsigns
340 commonly attributed to trajectories serving this route.

341 The *airspace* data set consists of seven months of trajectories flying through
342 the LFBBPT airspace of the French Bordeaux Area Control Centre (ACC) between
343 January 1st and August 6th 2017. The data set is limited to 14,461 trajectories
344 crossing the airspace during the time intervals when the sector was operationally
345 deployed according to the Sector Configurations Plans (SCP), also known as *open-*
346 *ing schemes*. The goal of using the SCP is for the traffic under analysis to be repre-
347 sentative of operational situations with a level of workload deemed acceptable by
348 controllers.

349 The *landing* data set consists of 19,480 trajectories landing at Zurich airport
350 (LSZH) between October 1st and November 30rd 2019. We relied on The OpenSky
351 Network [1] database to properly label trajectories landing at LSZH.

352 All datasets have been requested and preprocessed with the help of the Python
353 traffic library [48] which downloads OpenSky data, converts the data to struc-
354 tures wrapping pandas data frame and provides a specialised semantics for aircraft
355 trajectories (e.g., intersection, resampling, filtering, and more). In particular, it
356 iterates over trajectories based on contiguous timestamps of data reports from a
357 given icao24 identifier: all trajectories are then assigned a unique identifier and
358 resampled to one sample per second.

359 Figure 5 shows a preview of the data contained in the pandas data frame un-

	timestamp	altitude	callsign	geoaltitude	groundspeed	icao24	latitude	longitude	squawk	track	vertical_rate	flight_id
0	2017-03-14 20:34:57+00:00	24725.0	AF526KB	25550.0	429.786785	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
1	2017-03-14 20:34:58+00:00	24750.0	AF526KB	25550.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
2	2017-03-14 20:34:59+00:00	24750.0	AF526KB	25550.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
3	2017-03-14 20:35:00+00:00	24800.0	AF526KB	25600.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
4	2017-03-14 20:35:01+00:00	24825.0	AF526KB	25650.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117

Figure 5: Preview on the trajectory data associated to one trajectory.

derlying the trajectory structures. The data for the three use case presented in this paper is now provided as generic imports in the `traffic` library, triggering a download from a corresponding figshare repository [49] if the data is not in the cache directory of the user:

```
from traffic.data.datasets import (
    # -- The city pair use case --
    # -> 3536 trajectories between Paris--Only to Toulouse--Blagnac (2017)
    paris_toulouse_2017,
    # -- The airspace use case --
    # -> 14,461 trajectories crossing the LFBPT sector (1st semester 2017)
    airspace_bordeaux_2017,
    # -- The landing use case --
    # -> 19,480 trajectories landing at Zurich airport (Oct./Nov. 2019)
    landing_zurich_2019
)
```

4.2. City pairs

The provided dataset includes full trajectories from Paris–Only to Toulouse–Blagnac airports. Trajectories from the OpenSky Network are subject to errors happening in different steps of the acquisition chain: errors before the emission of data (imprecision in the localisation, quantification artefacts) and errors in the receiving and decoding of the data by a pool of heterogeneous feeders. A first set of basic filtering and resampling methods provided in the `traffic` library has been applied to (partly) sanitise all data sets before publication.

Then, the second step consists in defining a bounding box containing the set of trajectories. We chose in this section to consider roughly drawn bounding boxes including most trajectories before they enter the TMA of LFB0 airport (here the bottom-left and top-right coordinates of the learning box are resp. (44°17' N, 0°57' E) and (45°32' N, 2°45' E)). We also chose to include in the bounding box the first navigational beacon (i.e., a set of geographical coordinates) of the standard arrival (STAR) procedures.

The next step in the data preparation consists in resampling the trajectories in the selected bounding box. Since autoencoders have a fixed number of inputs, we

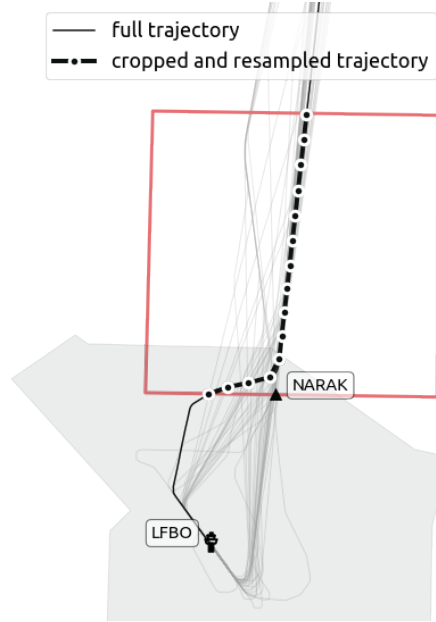


Figure 6: Data preparation: each trajectory is cropped and resampled with 150 samples.

381 resample each subset of trajectory cropped to our learning box so as to get only d
 382 equally distributed samples. Figure 6 plots a resampling with $d = 15$ for illustra-
 383 tions' sake but we chose a larger number of samples $d = 150$ for our experiments.
 384 This choice is arbitrary and we found that other values of d in the same order of
 385 magnitude have no significant impact on the results.

386 Features are chosen among all data provided in the ADS-B specifications: lat-
 387 itude, longitude, GPS and barometric altitude, track angle, ground speed, vertical
 388 speed. Different airspeeds (CAS, IAS, TAS, etc.) are sent by aircraft upon request
 389 on DF 20 and 21 but for the sake of clarity, we chose to keep these features out
 390 of our dataset for future work beyond the scope of this paper. Controller's actions
 391 are most often expressed in terms of altitude ("*climb to flight level 310*"), track
 392 angle ("*turn left heading 210*", "*direct to NARAK*"), and speed ("*reduce speed to*
 393 *160 kts*"); since speeds are expressed in IAS and not in ground speed, we focused
 394 on track angles and altitude profiles.

395 For the sake of clarity, we focus here on an analysis on normalised track an-
 396 gles. The input dimension, i.e. the number of neurons on both input and output
 397 layers of our autoencoder, has been set to $d = 150$. The embedding dimension,
 398 i.e. the number of neurons on the hidden layer has been set to a lower value of
 399 64. All neurons are defined with a sigmoid activation function. The loss function

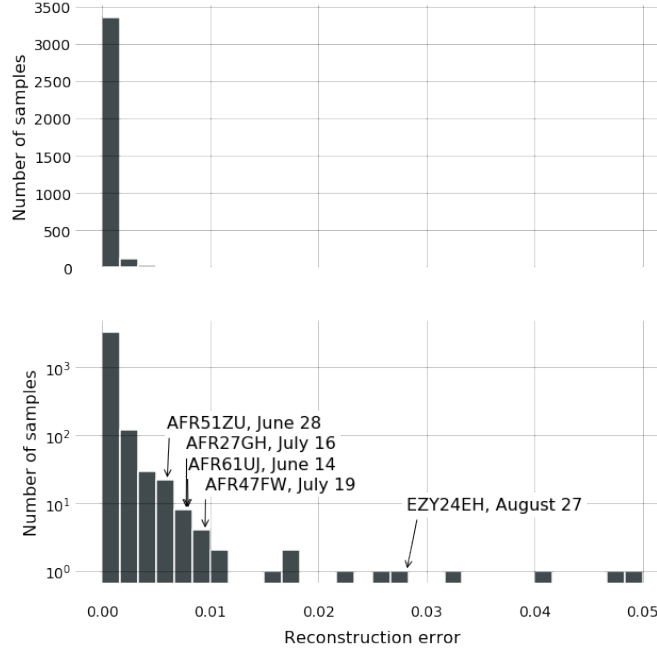


Figure 7: Distribution of reconstruction errors on the Toulouse scenario. The bottom distribution replicates the top distribution on a semi-log y-axis.

used is the mean squared error as defined in (2), which compares vectors of scaled track angles (through min-max normalisation) with their *reconstructions*, i.e. their images through the autoencoder.

As a result of our training process, we get a *reconstruction error*, i.e. a measure of the difference between a given trajectory and its autoencoded representation. Figure. 7 plots the distribution of these reconstruction errors. The model is trained to minimise the sum of all reconstruction errors, so the distribution is centred around zero.

The same distribution is plotted on a logarithmic y-axis to emphasise the few specific trajectories with higher reconstruction errors: we study in the following section the contextual situations associated to such specific trajectories pointed on the distribution. We focused first on trajectories with the highest reconstruction errors, which should be representative of the most unusual trajectories; then on a few situations with lower reconstruction errors, closer to the tail of the bell-shape distribution.

The analysis of the context was made after analysing traffic around anomalous detected trajectories. ADS-B may be an incomplete source of data for a thorough

analysis of the situation since some aircraft are still not properly equipped. In that case, flight plans and trajectories inferred by multilateration give a hint but situations analysed in this paper have been selected so as to be representative of the wide panel of detected situations and to be explainable with ADS-B tracks openly accessible. A perfect analysis would have also involved confirmation of our hypotheses with radio recordings; unfortunately, none were available for the chosen scenarios.

In general, we found that very high reconstruction errors are associated to less common situations; explanations are more to be found in the METAR history or regulation history. Conversely, reconstruction errors closer to the tail of the bell-shaped distribution, are more prone to yielding nominal situations which may be explained by ATC orders issued for a deconfliction or sequencing purpose.

4.2.1. Highest values of reconstruction errors relate to less common situations

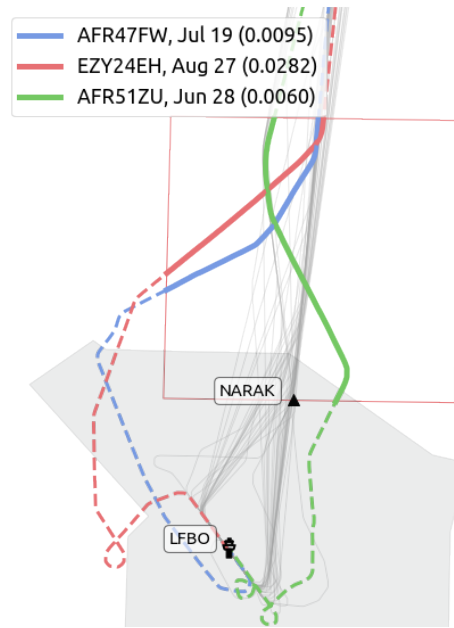


Figure 8: Situations with high reconstruction errors on the city-pair scenario

Flight AFR51ZU on June 28th (Figure 8) yields a high (although not the topmost) reconstruction score and its peculiar route on final approach called the authors' attention. This day was marked with a lot of delays because of weather. METAR on that day is particularly explicit: thunderstorm (TS), presence of cumulonimbus (CB) and a gusting wind forecast (20G35KT). Local news reported that traffic was

435 interrupted on that day in Toulouse–Blagnac when AFR51ZU was in approach. The
 436 aircraft was not rerouted, but the interruption may explain the unusual path and the
 437 two loops during final approach.

438 LFBO 281600Z AUTO 20006KT 180V240 9999 TS FEW033/// SCT047/// BKN060///
 439 ///CB 20/14 Q1004 TEMPO 28020G35KT 2000 TSRA=
 440

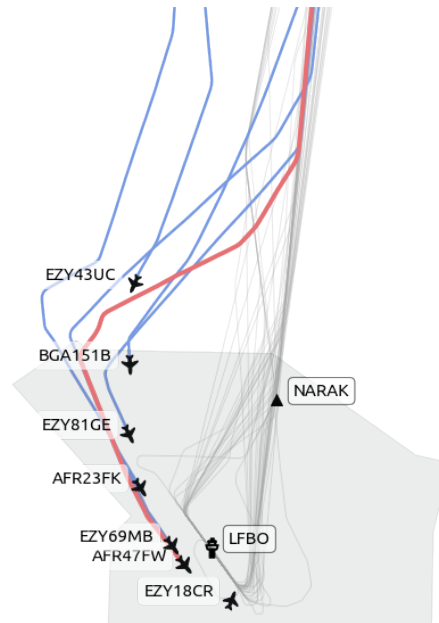


Figure 9: Weather impact on STAR procedures on the city-pair scenario

441 Traffic around Toulouse airport on July 19th was also impacted by poor weather
 442 and cumulonimbus probably located on the path of the STAR procedure starting
 443 from NARAK. All aircraft (including AFR47FW, see Figure 9) coming from the North-
 444 East were deviated to the West prior to entering the TMA, so as to be sequenced
 445 on the usual STAR procedures applicable to aircraft coming from the North-West.

446 LFBO 192100Z AUTO 02004KT 350V050 9999 TS BKN024/// OVC031/// ///CB 23/19
 447 Q1014 TEMPO 1500 TSRAGR BKN010 BKN040CB BECMG 14010KT=
 448

449 QFU refers to the magnetic heading of the runway in use. ATC may change the
 450 QFU at any time depending on the weather conditions. On August 27th, EZY24EH
 451 took a very peculiar route (Figure 8) as other aircraft were landing from the south
 452 on QFU32. After the last flight has landed (16:51), a first aircraft landed at 17:08
 453 on QFU14 and EZY24EH was on hold before landing 3 minutes later.

454 As we validated our approach on different city pairs, including flights from

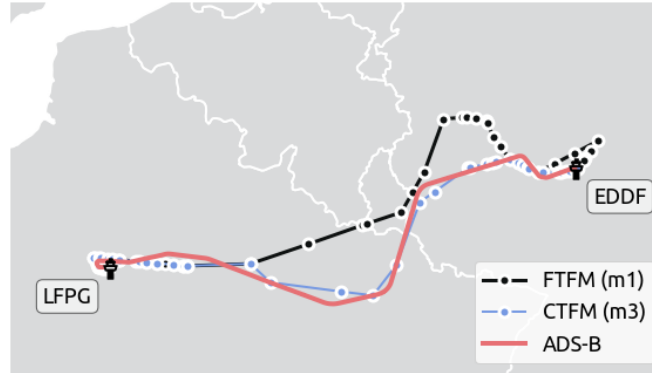


Figure 10: Last filed flight plan and trajectory of DLH2F on June 2nd, 2017, detected as outlier on one year of traffic between LFPG and EDDF.

Paris–Charles-de-Gaulle (LFPG) to Frankfurt (EDFF), we found that regulations set by ATC may also result in very high reconstruction errors. Historical records contain many regulations in place between Paris and Frankfurt (Reims and Langen ACC) in early evening on June 2nd, 2017, because of cumulonimbus present in the area. In particular, DLH2F was impacted by a 15 minute delay before departure because of a regulation filed by Frankfurt arrivals. Figure 10 plots the last filed flight plan (Filed Tactical Flight Model, FTFM) and the Current Tactical Flight Model (CTFM), refined version of the FTFM based on live positions for DLH2F. Such a pattern suggests that DLH2F adapted its route to avoid further regulations in the area.

4.3. High values of reconstruction errors relate to tactical ATC actions

In contrast, we found more classical and conventional ATC situations in the belly of the reconstruction error distributions.

On Figure 11, AFR27GH has a higher reconstruction error, probably because its trajectory is pushed quite to the East of the NARAK beacon, a less usual pattern for trajectories on the LFP0 to LFB0 route. As we investigate closer into the situation, it appears that AFR27GH was flying behind EZY81GE before being instructed to turn left. As EZY7431 arrived from Lyon (to the North-East), AFR27GH was sequenced behind with an appropriate ATC order. RYR3YM arrives next and is sequenced behind AFR27GH in a similar manner.

On Figure 12, DLH4J flies into Frankfurt area. The density of traffic converging on this IAF at this time of the day would probably not explain this shift in trajectory. The explanation could come from TAP571 which took off a bit earlier from runway

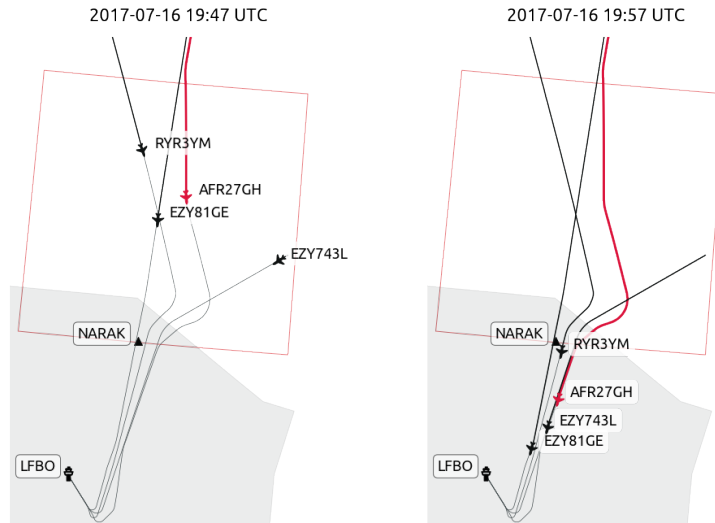


Figure 11: Aircraft are sequenced for landing before entering Toulouse TMA; AFR27GH is vectored behind EYZ81GE and EYZ743L.

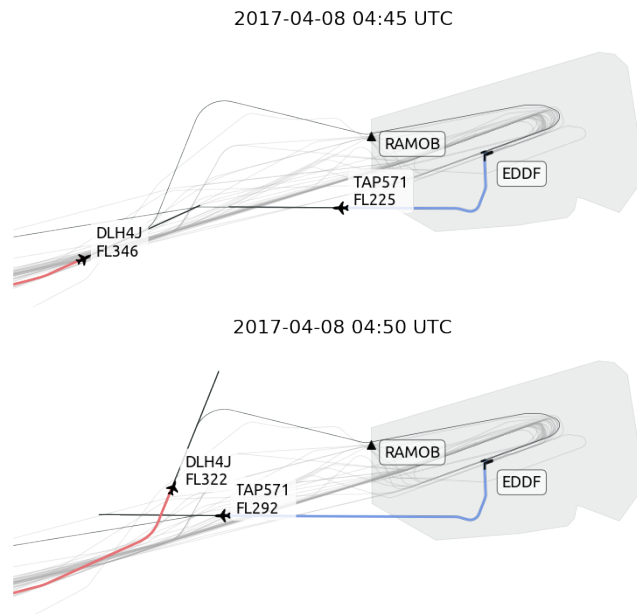


Figure 12: Deconfliction between DLH4J scheduled for landing and TAP571 taking off from Frankfurt airport, runway 18.

18. Without a turn left order to DLH4J, her climb path could have crossed DLH4J's trajectory, probably causing a potential loss of separation.

4.4. Airspaces

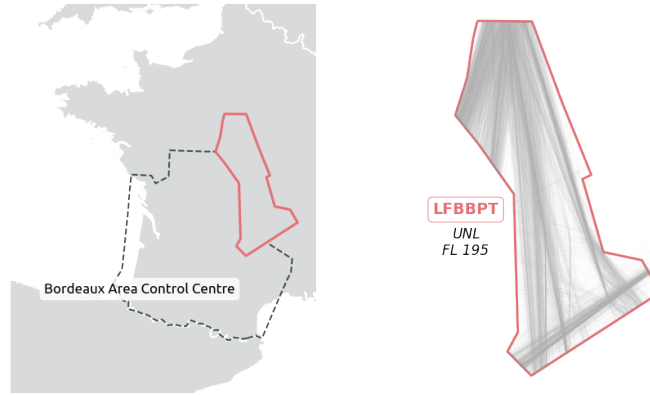


Figure 13: Description of the airspace dataset over the LFBFBPT sector of the Bordeaux Area Control Centre

The provided dataset includes seven months of trajectories from the LFBFBPT sector operated by Bordeaux Area Control Center (ACC) (see Figure 13) for periods when the sector was operationally deployed according to the Sector Configuration Plans (SCP), also known as opening schemes.

The challenge of this section is to apply our anomaly detection method to identified flows of trajectories entering and leaving a sector rather than from city-pair trajectories. Identified clusters may result from the aggregation of sparsely distributed trajectories.

The flow identification method applied in this paper for the use case concerning the en-route airspace [50, 4] is based on a progressive clustering technique originally developed by Andrienko et al. [51, 52]. This method is based on the DBSCAN algorithm, which requires two main parameters determining the size n and the density ε of clusters. For the first application of DBSCAN, we have set ε to 0.4 and n to 1% of the total number of trajectories. For the refinement of the clusters, DBSCAN has been executed with ε set to 0.5 and n set to 1% of the total number of trajectories in the cluster where it is applied. The minimum number of trajectories for a cluster to be formed has been established to 2% of the traffic in the sector.

The resulting cluster centroids representing the flows are displayed in Figure 14. A total number of nine flows have been identified with a percentage of outliers reaching 26.4% of the traffic. We have checked how well the generated clustering centroids match the ATS Route Network (ARN) also published on eAIP.

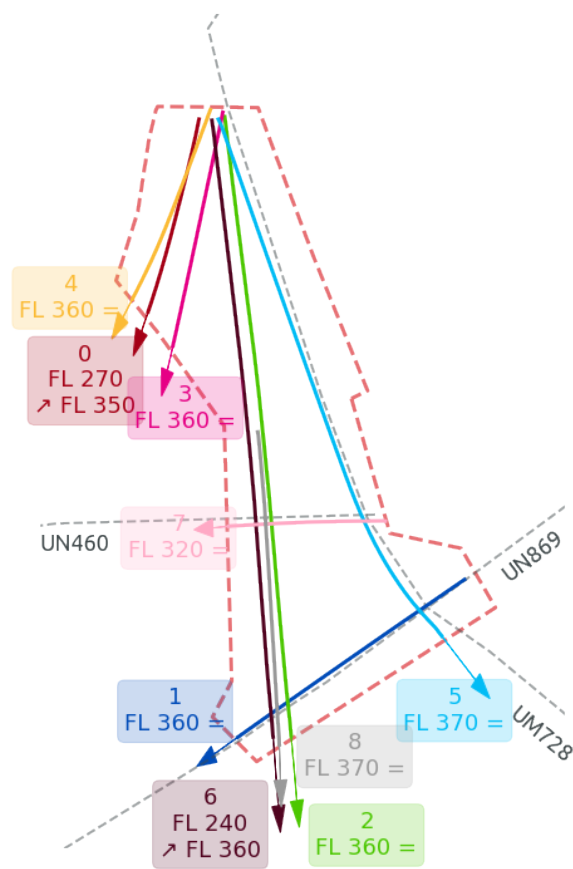


Figure 14: The clustering method applied to our dataset trimmed to LFBBPT sector resulted in the nine following clusters. The altitude below the identifier of the cluster reflects the altitude profile of the centroid.

Some clusters fit well to sections of the published air routes, e.g. UN869 (cluster 1), UM728 (cluster 5) or UN460 (cluster 7) but a similar match is less evident with clusters 0, 3 and 4. Cluster 0 is an evolving flow with flights taking off from Paris area, so it seems reasonable to have it separated from clusters 3 and 4 which are both stable flows with a centroid at FL360. The reason for separating clusters 3 and 4 is less obvious, but certainly due to the fact that the exit points of these two clusters are separated by a less dense area, which can be observed in Figure 13.

All trajectories within a cluster are then considered independently. We applied to each cluster the anomaly detection technique presented in Section 3.1 with a different autoencoder network architecture, adapted to the resampling of our trajectories at 50 points per trajectories, working only with normalised track angles so as to focus on lateral resolutions of potential conflicts.

The MSE loss converged properly for each cluster, although the distribution of reconstruction errors of clusters with more sparsely distributed trajectories (e.g. cluster 4) lead to distribution profiles suggesting two modes, as reflected in Figure 4. The regularisation term presented in Section 3.3 lead to better results and validated the idea behind regularisation which consists in penalising our criterion (the MSE loss) hoping we can avoid overfitting and converge toward more robust solutions.

4.4.1. Weather related events

callsign	date and time of entry in LFBBPT	cluster	rank
TRA47R	2017-06-20 18:21:00Z	2	6
RYR9TG	2017-06-20 19:09:28Z	4	1
TOM84T	2017-06-20 19:14:51Z	4	3
RYR79EY	2017-06-20 18:35:46Z	4	7
SAA235	2017-06-20 19:07:15Z	5	9
DAH1007	2017-06-20 18:33:34Z	6	8
DLH68F	2017-06-20 18:25:17Z	7	8
VLG83TJ	2017-06-20 18:21:00Z	8	2
AFR88DM	2017-07-08 19:55:01Z	0	1
VLG8248	2017-07-08 20:09:53Z	0	6
AEA1008	2017-07-08 19:39:50Z	0	9
AAF221	2017-07-08 20:08:52Z	0	10
FIN611	2017-07-08 19:42:22Z	2	8

Table 1: Most significant trajectories/days in LFBBPT grouped by date.

We found in Section 4.2 with the city-pair data set that once the autoencoder has converged, the few samples with the highest reconstruction errors were associated with exceptional events (mostly weather related) whereas reconstruction errors located in the "belly" of the distribution were matching tactical ATC operations. In

order to validate this assumption with the airspace data set, we selected the top 10 trajectories with the highest reconstruction errors for each cluster, for a total of 90 trajectories. Table 1 shows that two days were particularly represented in that subset of trajectories. The last column (rank) reflects the position of the sample in the distribution: 1 stands for the highest reconstruction error in the cluster, 2 for the second highest, etc.

A first look at the METAR history in airfields located around the LFBFBPT sector (LFLX, Châteauroux, to the North-West of LFBFBPT and LFLC, Clermont-Ferrand in the Southern part of the sector) reflects locations of cumulonimbus (CB) and tower cumulus (TCU) consistent with the location of anomalous trajectories (Figure 15): CB impacted the whole sector (hence clusters 2, 4, 5, 6, 7 and 8) on June 20th but only the Northern part of the sector (mostly clusters 0) on July 8th.

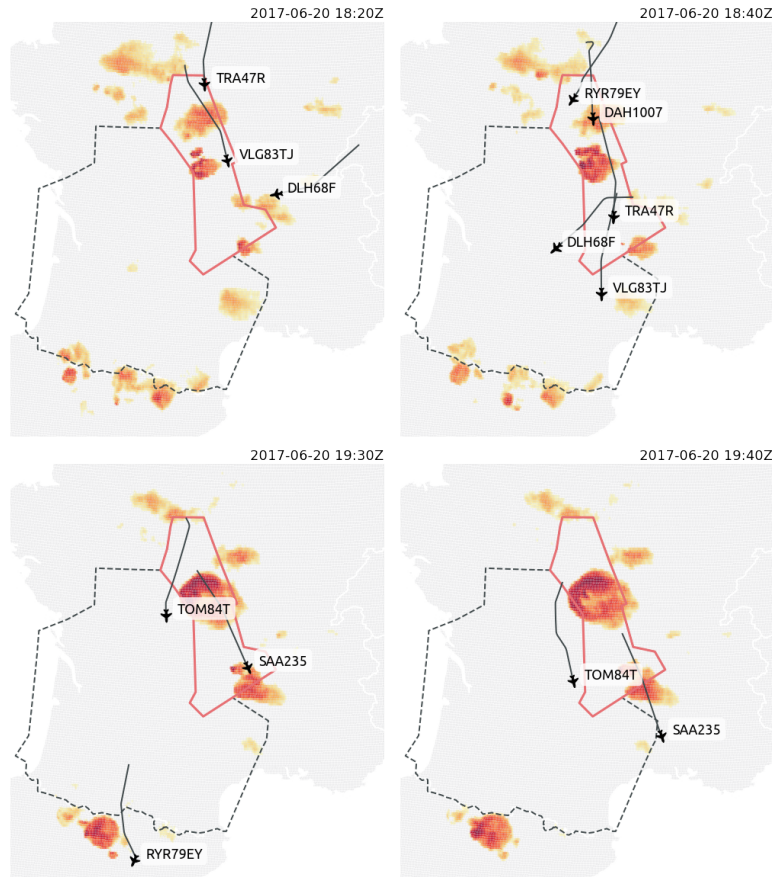


Figure 15: Trajectories of aircraft flying to avoid thunderstorm activities.

Figure 15 plots positions, trajectories for the past 15 minutes of aircraft flagged as anomalous by our method on June 20th 2017. The location of cumulonimbus is estimated from Thermal IR data from the Spinning Enhanced Visible and InfraRed Imager (SEVIRI), collected by the Meteosat Second Generation series of satellite.

The sector was implemented during the last hours of daylight, which may suggest that pilots avoid thunderstorms areas based on their visual perception in the early hours of operation, and relied on their on board weather radar in the later hours. Another explanation for aircraft seeming to fly through CBs may rely in the anvil shape of such cloud developments: satellite images only see the top of the clouds so it may be safe to fly below the cap and still avoid the vertical development of the cloud. Based on this kind of heatmap, future safety studies could assess how pilots manage to avoid thunderstorm areas based on the information of on board weather radars.

4.4.2. Detection of tactical ATC actions

Isolating deconfliction ATC orders in regular traffic is a difficult task because most flights are executed without much deviation from their original intention. Reconstruction errors help isolate flights calling for further analysis. We considered hereafter only a subset of our trajectories, namely trajectories with a reconstruction error higher than a given threshold. We defined the threshold based on the fitted exponential distribution f defined in (3) and illustrated in Figure 16 with the set of samples $\{x_i\}$ s.t.

$$f(x_i) \leq \frac{1}{5} \cdot f(0) \text{ i.e. } x_i \geq \log(5) \cdot \ell_{cluster}(u, v) \quad (7)$$

In an attempt to automatise the process, we computed the closest point of approach (CPA) for all pairs of trajectories which fly at the same moment in LFBPT and which belong to our subset of trajectories defined in (7). For the CPA computation, we used the distance between two trajectories based on the cylindrical norm defined in [53]:

$$d_{CPA} = \min_t \left(\max \left(\frac{d_{lat}(t)}{5 \text{ nm}}, \frac{d_{vert}(t)}{1,000 \text{ ft}} \right) \right) \quad (8)$$

where d_{lat} is the distance between the two WGS84 coordinates and d_{vert} the difference of altitudes. 5 nm and 1000 ft are respectively the lateral and vertical separation minima required between aircraft flying within Reduced Vertical Separation Minima (RVSM) airspace [54]. Since we observe traffic which has allegedly been deconflicted by ATC, all pairs of trajectories should be separated by a distance $d_{CPA} \geq 1$. However, we assume that an action of deconfliction is likely to involve pairs of trajectories with a d_{CPA} relatively small.

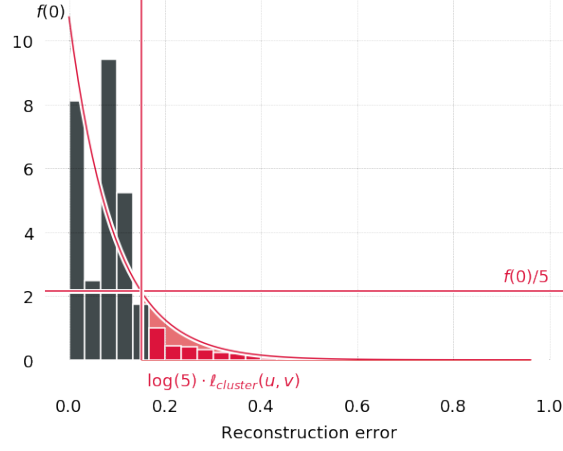


Figure 16: Selection of the sample trajectories with the higher reconstruction scores (example of cluster 4). For the purpose of the study, we chose a threshold of $1/5 = 20\%$ although this could be reconsidered in the future.

In the following we focus on pairs of trajectories respecting the following conditions:

1. each trajectory has a reconstruction error higher than $\log(5) \cdot \ell_{cluster}(u, v)$ (see Figure 16);
2. their $d_{CPA} \leq 2$, i.e. the lateral and vertical distance at the CPA should be smaller than 10 nm and 2000 ft respectively. In addition, we impose a constraint on the vertical distance to be smaller than 1500 ft in order to focus only on aircraft flying at adjacent flight levels.

Each trajectory being associated with a cluster, we build the density matrix as shown in Figure 17 which reads as follows: the darker the colour at position (i, j) with $i \geq j$, the more trajectories from cluster i and j are possibly subject to a deconfliction order from the ATC. A first consistency cross-check with the map on Figure 14 seems convincing: cluster 0 interacts with clusters 3 and 4 (mostly cluster 3); cluster 1 interacts with clusters 2, 6 and 8, albeit less with cluster 5. Cluster 7 flies at a relatively lower altitude and only interacts with cluster 6 (trajectories climbing) but not with clusters 2 and 8 (constant altitudes).

We focus in the following on specific situations in converging flows, then on pairs of trajectories in the same flow which may be impacted by the same factors.

Figure 18 reflects two situations involving cluster 1 and one of clusters 2, 6 and 8, containing mainly deconfliction situations for aircraft flying at the same level. The first situation of Figure 18 involves TVF021Z (cluster 6) and TAP933A (cluster 1). The dashed line projects the situation as if no order had been given five

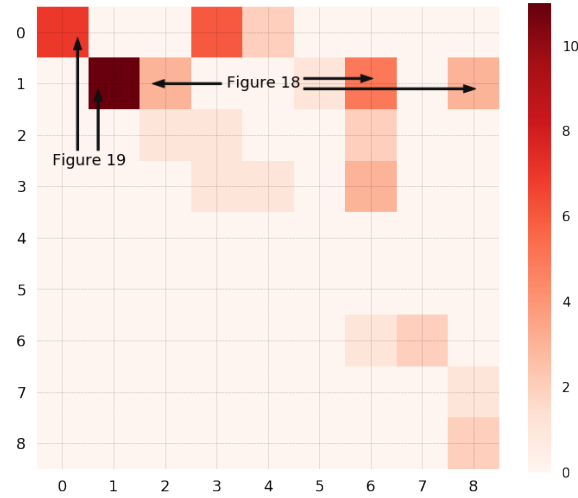


Figure 17: Density matrix (upper triangle only): the darker the colour in (i, j) , the more trajectories from clusters i and j are possibly subject to a deconfliction order from the ATC.

minutes before the closest point of approach with a possible conflict situation at the intersection of both routes. On the right hand side, ATC orders given to IBK7VY and IBK6113 seem to have anticipated the situation earlier with a probable deconfliction order given ten minutes before the closest point of approach. For such situations of converging routes at the same flight level, a future direction for improvement could be to automatically detect the level of anticipation of the deconfliction by looking backward from the CPA.

The density matrix on Figure 17 reflects a high number of potential deconfliction situations between pairs of trajectories from cluster 0 and from cluster 1. Figure 19 looks into those situations. On the left hand side, AEA1038 takes off from Paris–Charles-de-Gaulle and RAM781S from Paris–Orly. They both belong to cluster 0 and will probably be vectored on the same route. Having similar climb profiles (both aircraft are B738), special attention is paid to their lateral separation, probably leading to these peculiar trajectories. On the right hand side, TAP817 flies Milan–Porto while CES709 flies Shanghai–Madrid. When both aircraft join route UN869, a special attention is paid to their separation (see the level-off at FL360) before they are probably separated by being placed on lateral offsets from UN869.

4.5. Terminal Manoeuvring Areas

The provided data set includes 19,480 trajectories landing at Zurich airport between October 1st and November 30th 2019. Here, we applied a stacked autoencoder on 14,399 trajectories landing on runway 14 which is the runway the most

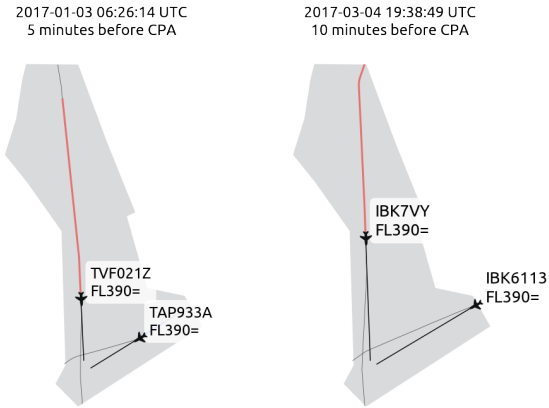


Figure 18: Deconfliction actions between cluster 1 and cluster 6 (left, on January 3rd), resp. cluster 1 and cluster 2 (right, on March 4th).

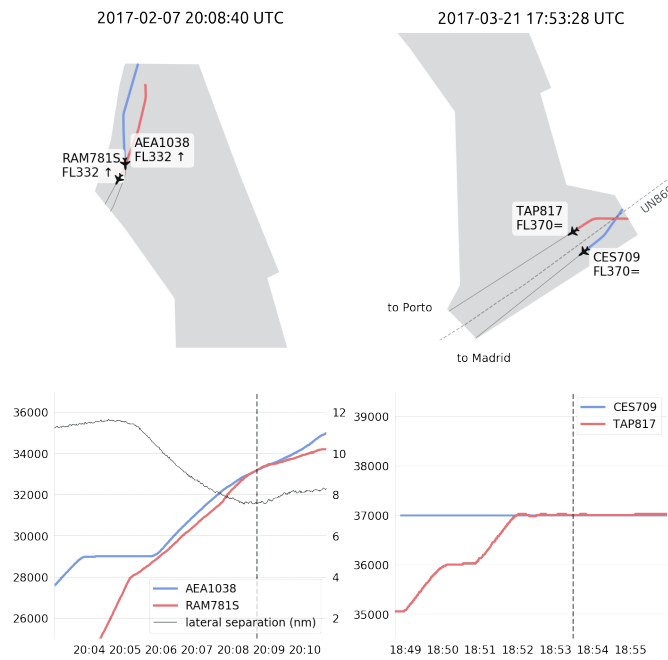


Figure 19: Deconfliction situations between trajectories in the same clusters.

614 often in operation. All trajectories are selected from the moment they enter the area
 615 within 40 nautical radius around the airport until the last point before the runway
 616 threshold.



Figure 20: All trajectories landing at Zurich airport are cropped between a 40 nautical miles distance to the airport and the runway threshold.

617 The challenge here is to be able to detect anomalies in a sample of converg-
 618 ing flows, in contrast with Sections 4.2 and 4.4. The autoencoder architecture we
 619 selected reduced 100 samples of normalised track angles to 48, 24 then 6 neurons
 620 per layer (encoding part) before reconstructing the samples through layers of 24,
 621 48 and 100 neurons each. Figure 21 plots a distribution of reconstruction errors
 622 which looks reasonable in spite of a low density in the bin containing the best
 623 reconstructed samples.

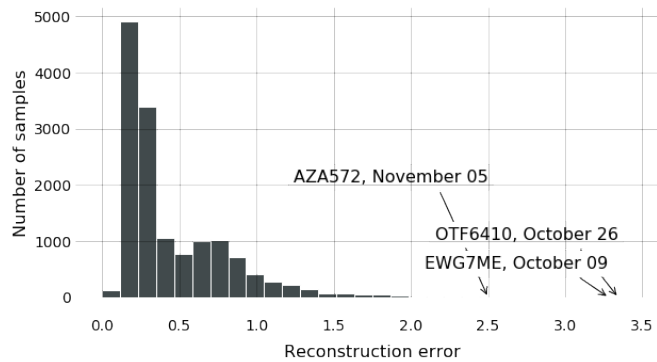


Figure 21: Distribution of the reconstruction errors on the landing dataset. Highlighted samples are plotted on Figure 22.

Figure 22 plots abnormal trajectories and gives hints to interpret why they are so poorly reconstructed. On one hand, Flight EWG7ME had three landing attempts (the first one aborted at an altitude of 2000 ft), and Flight OTF6410 stacked six holding patterns before being allowed to proceed further and align runway 14. However, the autoencoder did not only fail at recovering holding patterns in trajectories: Figure 23 explains why Flight AZA572's trajectory is atypical and non representative of aircraft landing from the South-East incoming flow. The right part of the Figure shows how landing operations were disrupted during that time frame and could provide hints why the aircraft was instructed that way.

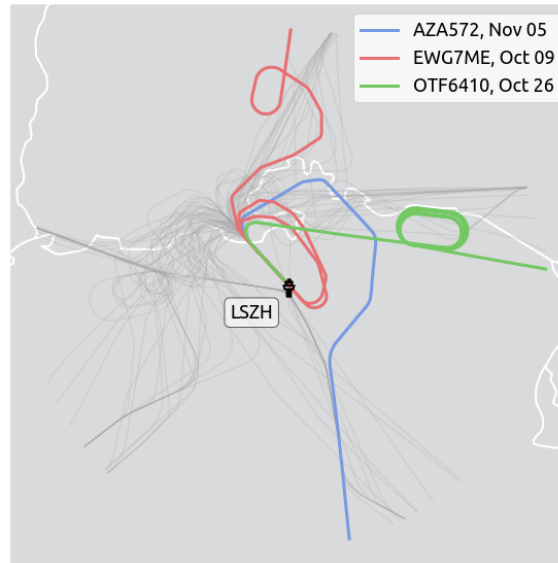


Figure 22: These three trajectories have very high reconstruction errors: Flight EWG7ME had three landing attempts, Flight OTF6410 stacked six holding patterns; the particular case of Flight AZA572 is further detailed on Figure 23

5. Information extraction on the latent space

This reconstructing approach may not be the most relevant to detect ATC tactical instructions in a Terminal Manoeuvring Area when pilots are in constant contact with ATC. Yet it remains a powerful tool to detect unusual patterns.

As explained in details in Section 3.1 and Figure 2, the first part of the autoencoder is a projection operator. Samples are projected onto a smaller dimension space before a generation operator attempts at reconstructing the original samples. In spite of a resulting poorer reconstruction ability, we added a layer to the network

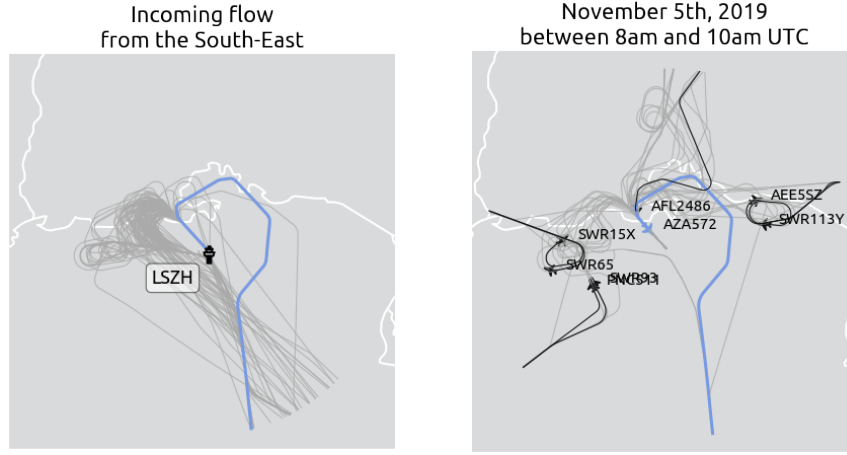


Figure 23: AZA572 is atypical with respect to the usual approach from the South-East (left). Indeed, traffic was particularly disrupted during that time frame (right).

used in Section 4.5 in order to project our samples further onto a two-dimensional space in order to plot the distribution of projected samples on Figure 24. Colours are assigned to all trajectories based on the bearing (interval) from the point they enter the 40 nautical mile radius with respect to the airport. Each colour can be associated to an incoming flow into the TMA of Zurich airport. Figure 24 shows how the network learned to organise all trajectories on the two dimensional space based on similar features.

Figure 25 plots the same distribution for the subset of self-intersecting trajectories, to be related to holding patterns. Three pairs of trajectories that are close to each other in their 2D representation are represented: the top right map displays two trajectories stacking two holding pattern before getting the clearance to land on runway 14. They are both located on the left part of the "cluster" matching the flow of self-intersecting trajectories landing from the West-North-West of Zurich. The two other pairs of trajectories on the bottom right of the same cluster also display similar features when plotted on a map.

Looking at the latent space of autoencoders trained to reconstruct trajectories is particularly enlightening with respect to their ability to extract information from large amounts of data which comes as a side effect of their first intended use.

6. Conclusions

We presented in this paper a simple and successful framework to detect, identify and characterise anomalies in past aircraft trajectory data. The method is based

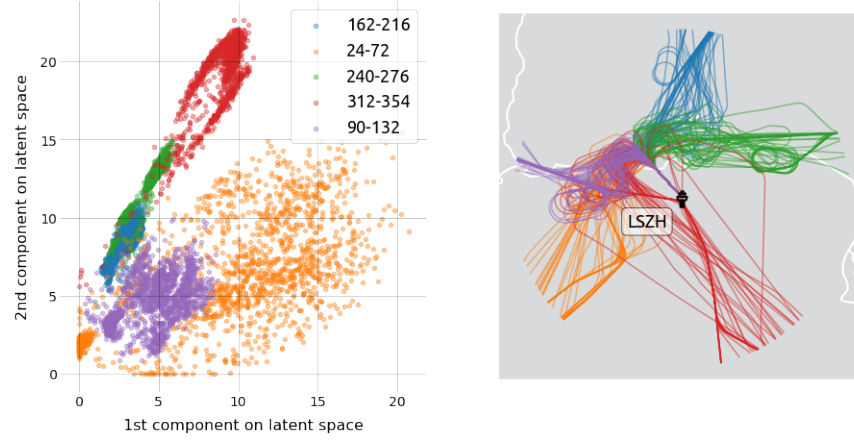


Figure 24: As autoencoders learn to reconstruct samples, they project the data on a small dimension space. Even when forced onto a latent space with two dimensions, they tend to organise the data by similarity. Here colours are associated to an incoming flow.

on unsupervised learning methods, namely autoencoders and clustering algorithms. We challenged this technique on three data sets very different in nature albeit covering a wide range of situations commonly encountered by air traffic controllers.

We introduced autoencoding neural network architectures and explained how a resulting reconstruction error can give hints about the nature of detected anomalies. Most anomalies are the result of ATC tactical actions, and the higher the reconstruction error, the more disruptive the situation. We found critical situations resulting from the impact of poor weather conditions, regulation measures and capacity issues.

This method has a wide range of applications in air transportation studies. Future works should leverage the information extracted from such massive amounts of data in order to produce predictive models in terms of ATC tactical actions (from an ANSP point of view) or estimated take-off and landing times (from an airline or airport point of view). Statistical studies on large amount of detected unusual situations would also improve the quality of safety analyses and collision risk models.

Lastly, unsupervised machine learning being about manipulating unlabelled data sets and extracting information, the question remains as to how to measure the quality of detection models such as the one introduced in this paper. Finding anomalies in past trajectory data is akin to looking for needles in a haystack but the variety of situations presented in this contribution attests the knowledge discovery power unleashed by this approach. In addition to advocating open data sets which are provided for reproducibility and comparison concerns, the authors will pursue

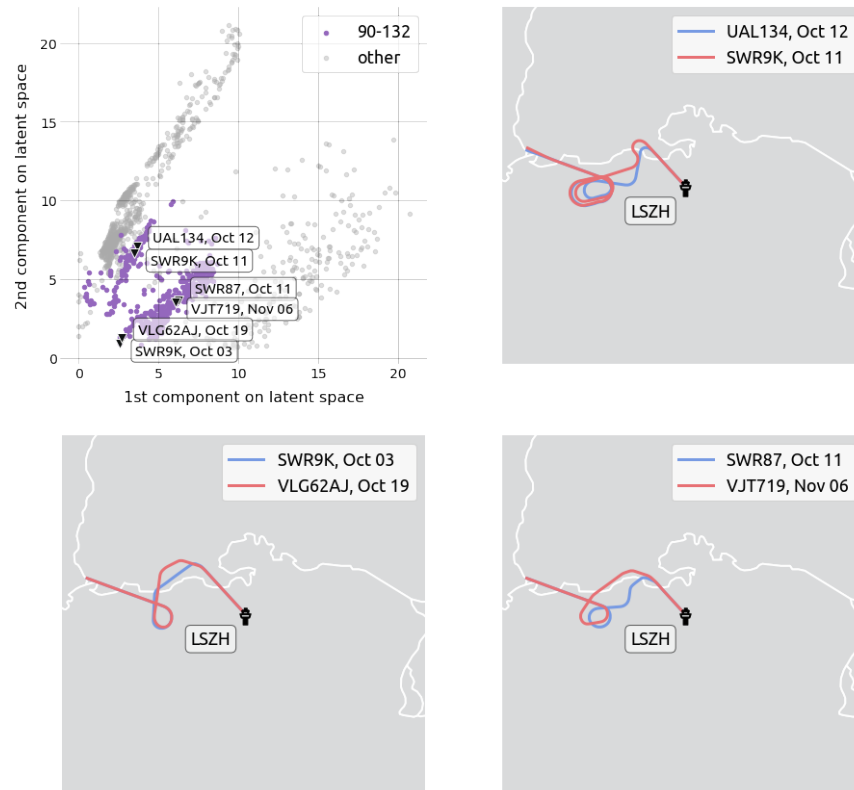


Figure 25: The scatter plot displays the latent space on trajectories containing one or several loops. Samples that are close to each other reveal similar patterns on the map. In particular, trajectories stacking two holding patterns are plotted on the top right and correspond to the left part of the purple cluster in the scatter plot.

684 their research efforts in determining a convincing set of metrics measuring the
685 relevance and accuracy of such frameworks.

686 Acknowledgments

687 In no particular order, the authors would like to thank Julie Saint-Lot from
688 ENAC for her invaluable help in providing an ATC feedback from the first days of
689 this research. Richard Alligier from ENAC found a bug in the code we published
690 with our first implementation for the regularisation term in Section 3.3 and imme-
691 diately offered to fix it. Finally, Simon Proud from Oxford University provided
692 the weather data which helped understanding the most disrupted situations in our
693 datasets.

694 References

- 695 [1] M. Schäfer, M. Strohmeier, V. Lenders, I. Martinovic, M. Wilhelm, Bringing
696 up OpenSky: A large-scale ADS-B sensor network for research, in: Proceed-
697 ings of the 13th international symposium on Information processing in sensor
698 networks, pp. 83–94.
- 699 [2] E. J. Garcia Gonzalez, Development of a 3-dimensional mathematical colli-
700 sion risk model based on recorded aircraft trajectories to estimate the safety
701 level in high density en-route airspaces, Ph.D. thesis, Aeronauticos, 2013.
- 702 [3] X. Olive, J. Grignard, T. Dubot, J. Saint-Lot, Detecting Controllers’ Actions
703 in Past Mode S Data by Autoencoder-Based Anomaly Detection, in: Pro-
704 ceedings of the 8th SESAR Innovation Days.
- 705 [4] X. Olive, L. Basora, Identifying Anomalies in past en-route Trajectories with
706 Clustering and Anomaly Detection Methods, in: Proceedings of the 13th
707 USA/Europe Air Traffic Management Research and Development Seminar.
- 708 [5] S. Lloyd, Least squares quantization in PCM, IEEE transactions on informa-
709 tion theory 28 (1982) 129–137.
- 710 [6] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering
711 method for very large databases, in: ACM Sigmod Record, volume 25.
- 712 [7] M. Ankerst, M. M. Breunig, H.-P. Kriegel, J. Sander, OPTICS: ordering
713 points to identify the clustering structure, in: ACM Sigmod record, vol-
714 ume 28, ACM, pp. 49–60.

- 715 [8] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., A density-based algorithm
716 for discovering clusters in large spatial databases with noise, in: KDD, vol-
717 ume 96, pp. 226–231.
- 718 [9] R. J. Campello, D. Moulavi, J. Sander, Density-based clustering based on
719 hierarchical density estimates, in: Pacific-Asia Conference on Knowledge
720 Discovery and Data Mining, Springer, pp. 160–172.
- 721 [10] P. Besse, B. Guillouet, J.-M. Loubes, R. François, Review and perspective for
722 distance based trajectory clustering, arXiv preprint arXiv:1508.04904 (2015).
- 723 [11] D. J. Berndt, J. Clifford, Using dynamic time warping to find patterns in time
724 series, in: KDD workshop, volume 10, pp. 359–370.
- 725 [12] M. Vlachos, G. Kollios, D. Gunopulos, Discovering similar multidimensional
726 trajectories, in: Proceedings 18th international conference on data engineer-
727 ing, IEEE, pp. 673–684.
- 728 [13] L. Chen, M. T. Özsu, V. Oria, Robust and fast similarity search for moving
729 object trajectories, in: Proceedings of the SIGMOD international conference
730 on Management of data, pp. 491–502.
- 731 [14] F. Hausdorff, Grundzuge der Mengenlehre, volume 61, American Mathemat-
732 ical Society, 1978.
- 733 [15] M. Fréchet, Sur quelques points du calcul fonctionnel, Rendiconti del Circolo
734 Matematico di Palermo (1884-1940) 22 (1906) 1–72.
- 735 [16] B. Guillouet, Apprentissage statistique: application au trafic routier à partir
736 de données structurées et aux données massives, Ph.D. thesis, Université de
737 Toulouse, Université Toulouse III – Paul Sabatier, 2016.
- 738 [17] A. Eckstein, Automated flight track taxonomy for measuring benefits from
739 performance based navigation, in: 2009 Integrated Communications, Navi-
740 gation and Surveillance Conference, IEEE, pp. 1–12.
- 741 [18] M. Gariel, A. N. Srivastava, E. Feron, Trajectory clustering and an applica-
742 tion to airspace monitoring, IEEE Transactions on Intelligent Transportation
743 Systems 12 (2011) 1511–1524.
- 744 [19] F. Rehm, Clustering of flight tracks, in: AIAA Infotech@ Aerospace 2010,
745 2010, p. 3412.

- 746 [20] M. Enriquez, Identifying temporally persistent flows in the terminal airspace
747 via spectral clustering, in: Proceedings of the 10th USA/Europe Air Traffic
748 Management Research and Development Seminar.
- 749 [21] M. C. R. Murça, R. DeLaura, R. Hansman, R. Jordan, T. Reynolds, H. Bal-
750 akrishnan, Trajectory clustering and classification for characterization of air
751 traffic flows, AIAA Aviation (2016).
- 752 [22] M. C. R. Murça, R. J. Hansman, L. Li, P. Ren, Flight trajectory data analytics
753 for characterization of air traffic flows: A comparative analysis of terminal
754 area operations between New York, Hong Kong and Sao Paulo, Transporta-
755 tion Research Part C: Emerging Technologies 97 (2018) 324–347.
- 756 [23] X. Olive, J. Morio, Trajectory clustering of air traffic flows around airports,
757 Aerospace Science and Technology 84 (2019) 776–781.
- 758 [24] G. Sabhnani, A. Yousefi, I. Kostitsyna, J. Mitchell, V. Polishchuk, D. Kier-
759 stead, Algorithmic traffic abstraction and its application to nextgen generic
760 airspace, in: Proceedings of the 10th AIAA Aviation Technology, Integration,
761 and Operations (ATIO) Conference, p. 9335.
- 762 [25] L. Basora, J. Morio, C. Mailhot, A trajectory clustering framework to analyse
763 air traffic flows, in: Proceedings of the 7th SESAR Innovation Days.
- 764 [26] A. Marzuoli, M. Gariel, A. Vela, E. Feron, Data-based modeling and opti-
765 mization of en route traffic, Journal of Guidance, Control, and Dynamics 37
766 (2014) 1930–1945.
- 767 [27] H. Arneson, A. Bombelli, A. Segarra-Torné, E. Tse, Analysis of convective
768 weather impact on pre-departure routing of flights from Fort Worth Center to
769 New York Center (2017).
- 770 [28] A. Bombelli, A. Segarra Torne, E. Trumbauer, K. D. Mease, Automated
771 route clustering for air traffic modeling, in: AIAA Modeling and Simulation
772 Technologies Conference, p. 1318.
- 773 [29] P. Ren, L. Li, Characterizing air traffic networks via large-scale aircraft track-
774 ing data: A comparison between china and the us networks, Journal of Air
775 Transport Management 67 (2018) 181–196.
- 776 [30] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM
777 Computing Surveys 41 (2009) 1–58.

- 778 [31] L. Basora, X. Olive, T. Dubot, Recent advances in anomaly detection methods
779 applied to aviation, *Aerospace* 6 (2019) 117.
- 780 [32] S. Das, B. L. Matthews, A. N. Srivastava, N. C. Oza, Multiple kernel learn-
781 ing for heterogeneous anomaly detection: algorithm and aviation safety case
782 study, in: *Proceedings of the 16th ACM SIGKDD international conference*
783 *on Knowledge discovery and data mining*, ACM, pp. 47–56.
- 784 [33] V. M. Janakiraman, D. Nielsen, Anomaly detection in aviation data using ex-
785 treme learning machines, in: *2016 International Joint Conference on Neural*
786 *Networks (IJCNN)*, IEEE, Vancouver, BC, Canada, 2016, pp. 1993–2000.
- 787 [34] L. Li, M. Gariel, R. J. Hansman, R. Palacios, Anomaly detection in onboard-
788 recorded flight data using cluster analysis, in: *Proceedings of the 30th Digital*
789 *Avionics Systems Conference*, IEEE, pp. 4A4–1.
- 790 [35] L. Li, S. Das, R. John Hansman, R. Palacios, A. N. Srivastava, Analysis
791 of flight data using clustering techniques for detecting abnormal operations,
792 *Journal of Aerospace information systems* 12 (2015) 587–598.
- 793 [36] L. Li, R. J. Hansman, R. Palacios, R. Welsch, Anomaly detection via a Gaus-
794 sian Mixture Model for flight operation and safety monitoring, *Transportation*
795 *Research Part C: Emerging Technologies* 64 (2016) 45–57.
- 796 [37] T. G. Puranik, D. N. Mavris, Anomaly detection in general-aviation oper-
797 ations using energy metrics and flight-data records, *Journal of Aerospace*
798 *Information Systems* (2017) 22–36.
- 799 [38] X. Olive, P. Bieber, Quantitative Assessments of Runway Excursion Precur-
800 sors using Mode S data, in: *Proceedings of the 8th International Conference*
801 *for Research in Air Transportation*.
- 802 [39] R. J. Campello, D. Moulavi, A. Zimek, J. Sander, Hierarchical density esti-
803 mates for data clustering, visualization, and outlier detection, *ACM Transac-*
804 *tions on Knowledge Discovery from Data (TKDD)* 10 (2015) 5.
- 805 [40] R. Deshmukh, I. Hwang, Anomaly detection using temporal logic based
806 learning for terminal airspace operations, in: *AIAA Scitech 2019 Forum*, p.
807 0682.
- 808 [41] A. Jones, Z. Kong, C. Belta, Anomaly detection in cyber-physical systems:
809 A formal methods approach, in: *53rd IEEE Conference on Decision and*
810 *Control*, IEEE, pp. 848–853.

- 811 [42] Z. Kong, A. Jones, A. Medina Ayala, E. Aydin Gol, C. Belta, Temporal logic
812 inference for classification and prediction from data, in: Proceedings of the
813 17th International Conference on Hybrid Systems: Computation and Control,
814 HSCC '14, ACM, New York, NY, USA, 2014, pp. 273–282.
- 815 [43] Z. Kong, A. Jones, C. Belta, Temporal logics for learning and detection of
816 anomalous behavior, IEEE Transactions on Automatic Control 62 (2016)
817 1210–1222.
- 818 [44] F. J. Sáez Nieto, R. Arnaldo Valdés, E. J. García González, G. McAuley, M. I.
819 Izquierdo, Development of a three-dimensional collision risk model tool to
820 assess safety in high density en-route airspaces, Proceedings of the Institu-
821 tion of Mechanical Engineers, Part G: Journal of Aerospace Engineering 224
822 (2010) 1119–1129.
- 823 [45] E. G. Gonzalez, F. S. Nieto, I. Izquierdo, Identification and analysis of prox-
824 imate events in high density enroute airspaces, in: Proceedings of the 7th
825 USA/Europe ATM R&D Seminar.
- 826 [46] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen,
827 Z. Lin, N. Gimsheine, L. Antiga, A. Desmaison, A. Kpf, E. Yang, Z. DeVito,
828 M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chin-
829 tala, PyTorch: An Imperative Style, High-Performance Deep Learning Li-
830 brary, arXiv:1912.01703 [cs, stat] (2019). ArXiv: 1912.01703.
- 831 [47] J. Sun, H. Vũ, J. Ellerbroek, J. M. Hoekstra, pyModeS: Decoding Mode-S
832 surveillance data for open air transportation research, IEEE Transactions on
833 Intelligent Transportation Systems (2019).
- 834 [48] X. Olive, traffic, a toolbox for processing and analysing air traffic data, Jour-
835 nal of Open Source Software 4 (2019) 1518.
- 836 [49] X. Olive, L. Basora, Reference data sets for detection and identification of
837 significant events in historical aircraft trajectory data., 2019.
- 838 [50] L. Basora, V. Courchelle, J. Bedouet, T. Dubot, Occupancy peak estimation
839 from sector geometry and traffic flow data, Proceedings of the 8th SESAR
840 Innovation Days (2018).
- 841 [51] G. Andrienko, N. Andrienko, G. Fuchs, J. M. C. Garcia, Clustering trajec-
842 tories by relevant parts for air traffic analysis, IEEE transactions on visualiza-
843 tion and computer graphics 24 (2018).

- 844 [52] S. Rinzivillo, D. Pedreschi, M. Nanni, F. Giannotti, N. Andrienko, G. An-
845 drienko, Visually driven analysis of movement data by progressive clustering,
846 Information Visualization 7 (2008).
- 847 [53] C. A. Munoz, A. J. Narkawicz, Time of closest approach in three-dimensional
848 airspace, Technical Report, NASA, 2010.
- 849 [54] A. ICAO, unified framework for collision risk modelling in support of the
850 manual on airspace planning methodology with further applications, circ
851 319-an/181 ed, International Civil Aviation Organization, Montreal, Canada
852 (2008).