Bringing Explainability to Autoencoding Neural Networks Encoding Aircraft Trajectories

Zakaria Ezzahed, Antoine Chevrot, Christophe Hurter, Xavier Olive

Bringing Explainability to Autoencoding Neural Networks Encoding Aircraft Trajectories

Zakaria Ezzahed*, Antoine Chevrot†, Christophe Hurter*, Xavier Olive†

*École Nationale de l'Aviation Civile
Université de Toulouse
Toulouse, France

†ONERA – DTIS
Université de Toulouse
Toulouse, France

***Abstract*—Autoencoders, a class of neural networks, have emerged as a valuable tool for anomaly detection and trajectory clustering: they produce a compressed latent space and capture essential features in the data. However, their lack of interpretability poses challenges in the context of ATM, where clear-cut explanations are crucial. In this paper, we investigate this issue by exploring visual methods to enhance the interpretability of autoencoders applied to aircraft trajectory data. We propose techniques to extract meaningful information from the structure of the latent space, and to promote a better understanding of generative models behaviours. We present insights from two simplified and real-world datasets and evaluate the structure of the latent space of autoencoders. Furthermore, we introduce suggestions for more realism in trajectory generation based on Variational Autoencoders (VAE). This study offers valuable recommendations to developers in the field of ATM, fostering improved interpretability and thus safety for generative AI in air traffic management.**

***Keywords* —** Autoencoders, eXplainable Artificial Intelligence, Interpretability

I. Introduction

Air Traffic Management (ATM) plays a critical role in ensuring the safety and efficiency of air travel worldwide. With the rapid increase in aircraft data, there is a growing need for advanced analytical techniques to gain insights into the details of aircraft paths. Anomaly detection and clustering of trajectories have emerged as crucial tasks in understanding and analysing this data.

Autoencoders, a special type of neural networks, have gained popularity for their ability to compress data, identify clusters [1], and detect unusual patterns [2], [3]. These models leverage a bottleneck structure to efficiently capture important information while filtering out noise and irrelevant features. The bottleneck, also called the latent space, serves as a compressed representation of input data, capturing essential features and patterns in a lower-dimensional space [4]. A major limitation of autoencoders is their lack of interpretability, as they can appear as no more than black box models. In the context of ATM, where safety is of importance, there are high expectation for model to provide explanations in Artificial Intelligence (AI) models understandable to end-users.

Different types of explanations have been developed to shed light on the behaviour of machine learning models [5]. For end-users, i.e. Air Traffic Control (ATC) operators, we aim to provide explanations that are detached from the underlying mechanics, making them interpretable without requiring extensive Machine Learning (ML) knowledge. For developers, we aim to provide explanations that establish a clear link between inputs, weights and outputs, clarifying the model's behaviour and highlighting which components are responsible for specific behaviours.

Noteworthy efforts have been made in explaining deep learning models in the field of generative AI for computer vision and natural language processing [6], with a particular focus on autoencoders. Researchers have developed techniques to understand the role of different parts of the model in generating specific outputs, enabling control over various aspects of the generated output [7]. There is a possibility of adapting and applying similar methods to explain models for aircraft trajectory data.

Some explainability methods employ analytical techniques to clarify critical aspects such as the significance of inputs, model weights, and outputs in relation to a model's decision-making process [8], [9]. However, when it comes to trajectory data, using these methods directly is quite challenging. Unlike images or text that humans can easily grasp, trajectory data, represented as raw numbers in tables, is more abstract and harder to understand. In contrast, users often prefer visual representations, such as altitude or speed profiles and 2D/3D trajectories, to comprehend trajectory information [10]. Although such visualizations offer a more intuitive understanding of aircraft trajectories, they may not fully expose all relevant information crucial for interpretability. As a result, our research direction primarily focuses on exploring visual methods for explaining the behaviour of autoencoders applied on aircraft trajectory data.

The existing explainability methods have primarily been applied to computer vision and natural language processing domains [11], with limited adaptation to time series anomaly detection using autoencoders. These efforts have primarily focused on predicting the next point in a time series [12], while our interest lies in classifying, clustering, and detecting anomalies among entire trajectory sequences, which falls under the category of time series classification [13]. The unique nature of aircraft trajectory data presents challenges in extracting relevant information to explain the model's behaviour.

To the best of our knowledge, no previous research has explored methods for explaining models applied to such data. Our approach promotes a more intuitive comprehension of model behaviour, encompassing visual latent space analysis and exploring the intricate relationships between dimensions within the latent space [14], [15], [16]. Through methodologies like disentanglement, which urges the model to decipher interpretable features relative to the input, we aim to yield perceptive interpretations of the latent vectors and their relationship to the input vectors [17].

This study builds upon existing techniques to enhance our understanding of how autoencoders operate with aircraft trajectory data. We utilize both simplified datasets and real-world data to evaluate the performance of autoencoders in detecting flight patterns. In the following, Section II presents the background and related work, emphasizing Air Traffic Management (ATM), autoencoders, and explainability methods, while highlighting the necessity of visual explanations in ATM and XAI. Methodologies, including dataset descriptions and autoencoder architectures, are delineated in Section III. Section IV delves into visual explanations for autoencoder interpretability, proposing several visual methods for

SESAR Innovation Days
27-30 November 2023
Seville

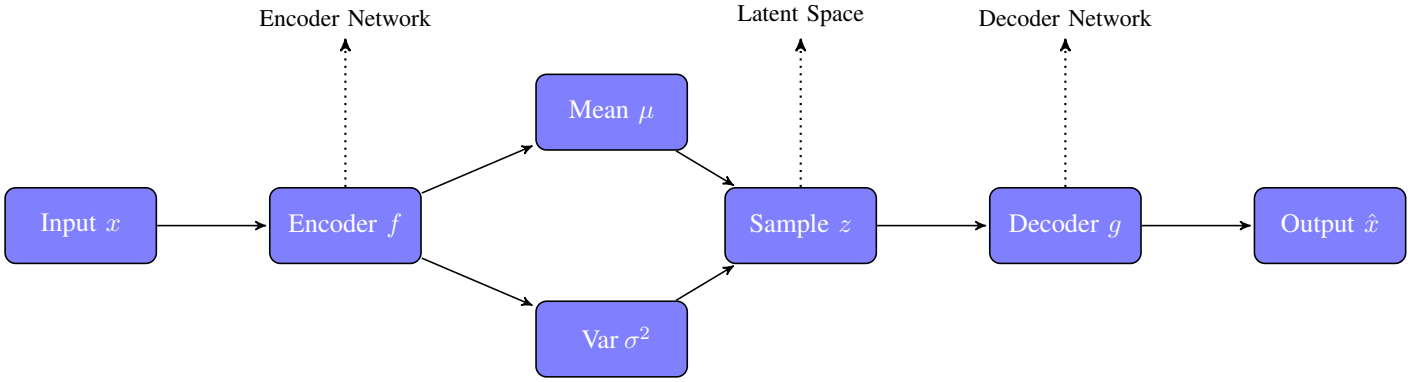

Figure 1: Architecture of a Variational Autoencoder

meaningful latent space insights. Experimental results and evaluations are presented in Section V, assessing visual explanation efficacy. Implications and recommendations specific to the ATM domain are summarized in Section VI, summarizing contributions and suggesting future research directions in enhancing interpretable deep learning models for trajectory data analysis in ATM.

II. Literature Review

The complexity of trajectory data has presented challenges for traditional machine learning methods, leading to the adoption of deep learning approaches to tackle the high-dimensional nature of the data. In the field of Air Traffic Management (ATM), the use of deep learning techniques has become increasingly popular for various applications.

Autoencoders have emerged as powerful and adaptable tools in Air Traffic Management, effectively addressing challenges related to the complexity of trajectory data through clustering and anomaly detection applications. Their widespread use and various modifications have shown promising results in enhancing the understanding and analysis of aircraft trajectories. They have found extensive use in trajectory clustering for different flight phases, including en-route and terminal trajectories.

The primary goal of autoencoders is to project high-dimensional trajectory data (N) into a lower-dimensional latent space (n), where $N \gg n$, allowing the application of conventional clustering and projection algorithms [4]. Notably, Olive et al. [1] employed an autoencoder with a modified error term to create a clustered structure of the data manifold within the latent space. Similarly, Zeng et al. [18] used a Gaussian mixture model after feature extraction with a deep autoencoder to identify the main traffic flow patterns in terminal airspace.

Wang et al.[19] explored the use of autoencoders for extracting learned features from the high-dimensional aviation data. The authors also used autoencoders for deep cleaning the data, reducing the workload on data scientists. Another study by Memarzadeh et al.[20] developed a Robust and Explainable Semi-supervised (RESAD) applied to aviation data. The developed model use weakly labelled datasets to detect anomaly on landing approaches of commercial flights.

The quest for making deep learning more understandable has attracted significant attention in recent years, resulting in various methods for interpreting models. These methods fall into two categories: model-agnostic and model-specific approaches [11]. Model-agnostic techniques, like LIME (Local Interpretable Model-agnostic Explanations) [9] and SHAP (SHapley Additive exPlanations) [21], aim to explain how a model behaves using only inputs and outputs, without concern for the underlying architecture. They provide explanations by highlighting the importance of features with respect to the input. Notably, Antwarg et al. [22] used SHAP in their time series anomaly detection method to clarify the results of their autoencoder. However, applying these techniques directly to trajectory data is challenging due to the inherent complexity of interpreting time series like trajectories.

On the other hand, model-specific methods, such as Layer Wise Propagation (LRP) [8], have been prominent in making models interpretable. LRP assigns relevance scores to neurons based on specific rules, revealing the importance of input features in relation to the output. Although this approach is effective in domains like Computer Vision, applying it to aircraft trajectories faces similar challenges as mentioned earlier.

In the domain of generative artificial intelligence, VAEs [23], depicted in figure 1, have garnered significant interest for understanding their latent space. Particularly, beta-VAE [7], a modified version of VAE with an enforced disentangling of latent vectors through a modified loss function, has gained attention. Disentangling means that two independent features in the input space will have independent linked vectors in the latent space [7]. This advancement has allowed researchers to manipulate and control generative aspects of VAE models from the latent space. Higgins et al. [7] successfully manipulated facial features and 3D chair models using latent variables and provided interpretations of VAE-trained musical tracks where latent variables corresponded to different aspects of the output track.

Krauth et al. [24] delve into the computational methodology for generating flight trajectories within the terminal zone of Zurich Airport. The authors initially employ a VAE for the model training phase and subsequently use the decoder to generate trajectories based on the latent space. To improve the model's ability to handle time-series data, they introduce a Temporal Convolutional Variational Neural Network (TCVAE), which is particularly skilled at capturing temporal characteristics. Additionally, the authors utilize dimensionality reduction techniques like Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbour Embedding (t-SNE) to identify distinct clusters within the latent space. By sampling from the proximity of these clusters, they can generate trajectories that exhibit features closely related to those associated with the identified clusters.

In line with these principles, our research aims to develop interpretable architectures for trajectory clustering and anomaly detection in the field of Air Traffic Management (ATM). Our objective is to make the latent variables in the autoencoder interpretable with respect to the decision-making process of the autoencoder.

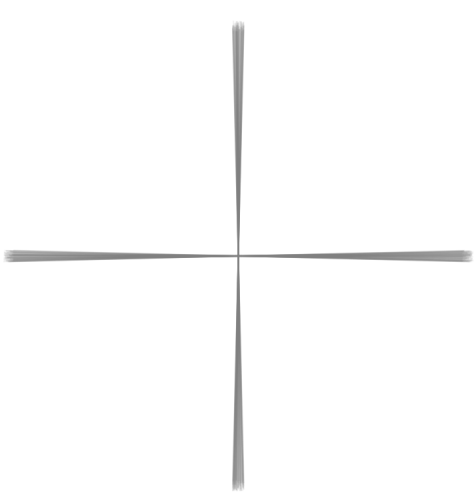

Figure 2: Representation of trajectories forming the toy dataset.

Figure 3: Trajectories landing at Zurich airport, runway 14

III. Methodology

A. Datasets

Our experimentation starts with a straightforward dataset comprising four converging trajectories, each originating from distinct directions and meeting at a central point (Figure 2). An element of controlled randomness is introduced to augment the variance and add a layer of variability to the trajectory data.

Then, we shift our attention to a dataset of two months of trajectories evolving in the Terminal Manoeuvring Area (TMA) before landing in Zurich airport (LSZH). The dataset [25]. The trajectories from the dataset are further preprocessed with the traffic Python library [26]: trajectories are resampled to ensure consistency, resulting in 100 equidistant points along each trajectory. This dataset consists of only normal trajectories. Unusual manoeuvres such as holding patterns and segments of trajectories after a possible go-around are excluded from the dataset. (Figure 3)

B. Autoencoders

Autoencoders are a specific type of neural network used for encoding and decoding tasks, assisting in reducing data dimensionality and extracting a compressed, simpler feature set. The encoder part, crucial to the autoencoder architecture, is represented mathematically by a function f and transforms the input data vector $\mathbf{x}$ into a latent space, referred to as $\mathbf{Z}$.

The decoder part, in contrast, reconstructs the original input data. Given a vector $\mathbf{z}$ from latent space $\mathbf{Z}$, the decoder aims to reverse the function performed by the encoder. The autoencoder learns through unsupervised learning, concentrating on capturing inherent patterns in the input data.

An autoencoder's efficacy is commonly evaluated using a specific loss function, like the Mean Squared Error (MSE). The MSE-based loss function, $\mathcal{L}(\boldsymbol{\theta})$, depends on the model parameters $\boldsymbol{\theta}$ and is described as:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$$

Here, n is the total number of data points. The goal of the loss function is to reduce the difference between the original and reconstructed data vectors $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i$, respectively.

C. Variational Autoencoders

Different from standard autoencoders, Variational Autoencoders (VAEs) have a unique feature: they map input data not to fixed points in the latent space but to a probability distribution covering that space [23].

VAEs have sparked significant academic interest, especially in generative artificial intelligence, because they encode input data as probabilistic distributions, not fixed points, which enhances data sampling capabilities. This makes VAEs quite useful for generative tasks and applications where understanding the data distribution is crucial.

VAEs work in a way that the encoder takes input data, marked as x, and maps it efficiently to a probability distribution within a latent or latent space, marked as z. This mapping is achieved through parameters, symbolized as ϕ, resulting in a mapping, mathematically expressed as $Q_\phi(z|x)$. Contrarily, the decoder takes a point in the latent space, z, and maps it back to the original data space, represented as x, using a different set of parameters, termed θ, resulting in another mapping, denoted as $P_\theta(x|z)$.

Using a method called variational inference, VAEs estimate the distribution of z based on the observed data, x, represented as $Q_\phi(z|x)$. The parameters ϕ and θ for the encoder's mapping $Q_\phi(z|x)$ and the decoder's mapping $P_\theta(x|z)$ are learned during the optimization process, which focuses on maximizing the evidence lower bound (ELBO) to improve data reconstruction and align the latent space distribution $Q_\phi(z|x)$ with the predefined prior $P(z)$, typically chosen to be a standard normal distribution for its simplicity and mathematical tractability.

Upon concluding training, VAEs can generate new data points by sampling from the learned probability distribution in the latent space. This is especially useful in tasks like image generation, as VAEs demonstrate a powerful technique that combines neural networks with probabilistic modelling, excelling not only in data compression but also in generating new, similar data based on what was learned during training. This versatility has proven very valuable in a variety of applications, including trajectory generation [24]

IV. Visual Explanations for Autoencoders

In this section, we talk about the mathematical structure presented in [27] to thoroughly explain the latent space of autoencoders. Our main goal is to assign importance to each part within this many-sided space and understand the features it has learned.

In a kind of autoencoder called variational autoencoders (VAEs), the latent vector Z that describes the latent data shape includes two key parts: the internal component (S) that grabs meaningful information needed for rebuilding data, and the external component (U) which takes care of local noise that doesn't affect the important information. Being able to tell the difference between and ignore this noise is crucial to really understanding the meanings within the latent space.

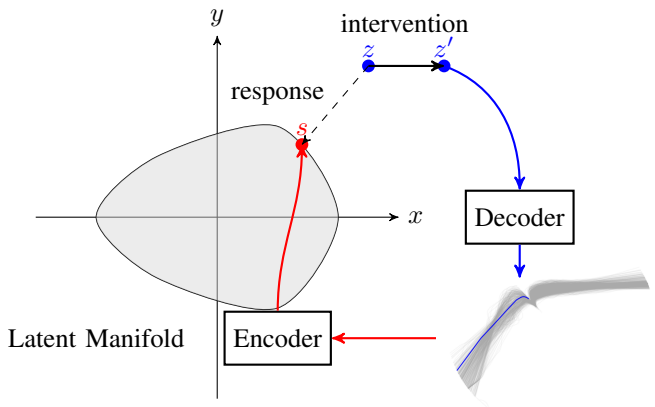

Figure 4: 2D plot of the decoding into a trajectory and re-encoding process in latent space.

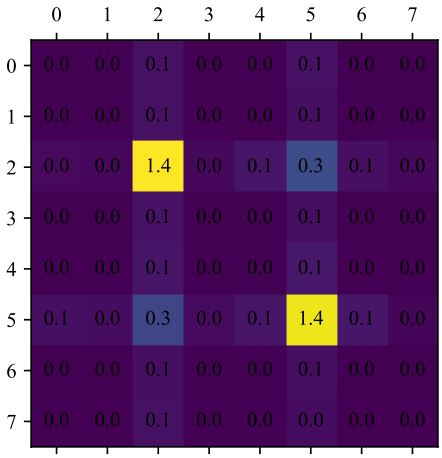

Figure 5: The response matrix, derived from the toy dataset, possesses distinct interpretative elements. The diagonal el-ements reflect response scores, serving as indicators of the extent to which each respective dimension was utilized. Alternatively, the off-diagonal elements represent the causal scores, elucidating the degree of causal interaction between each pair of dimensions.

We want to understand more about the semantics latent in this space. To do this, we think of these changes as interventions, where we carefully change a selected latent variable from z to z' and keep all other variables the same. As shown in figure 4, z' is decoded and then encoded again. This action results in a new latent point s, which is the internal part of z. This way of exploring based on interventions lets us examine the latent space and discover the learned features. The entire process acts as a response.

We introduce the following concepts from [27] to analyse the latent spaces:

- the **latent response matrix** a practical tool based on latent responses. This matrix aims to describe the extent to which latent variables causally affect one another in the learned generative process. Mathematically, each element M_{jk} in the matrix $M \in \mathbb{R}^{d\times d}$ quantifies the degree to which an intervention in latent variable j causes a response in latent variable k.

$$M_{jk}^2 = \frac{1}{2}\mathbb{E}_{z\sim p(Z);\tilde{z}_j\sim p(Z_j)}\left[\left|h_{\phi\theta}^k\left(\Delta(z_j \leftarrow \tilde{z}_j)(z)\right) - h_{\phi\theta}^k(z)\right|^2\right]$$

  The diagonal elements of the matrix can be interpreted as quantifying the extent to which an intervention along a specific latent variable is detectable. Off-diagonal elements show how an intervention on one variable affects another, thereby revealing the causal structure of the latent space.
- the $u(z)$ **function** serves as a mathematical tool to distinguish between a given latent sample z and its associated response s, which is generated by the encoding and decoding operations within the autoencoder framework. The function is defined as $u(z) = s - z \approx h_{\phi\theta}(z) - z$. This function is crucial for constructing divergence and mean curvature plots, which are used to explore the boundaries of the latent space.
- the **divergence plots** are employed to visually explore the latent space. These plots are constructed by computing $u(z)$ across a range of latent samples. Regions marked by pronounced curvature on the divergence plot indicate areas where $|u(z)|$ attains its minimum values, signalling proximity to the underlying data manifold X. These plots are instrumental in identifying regions where $u(z)$ experiences divergence, thus providing insights into potential discontinuities in the latent representation.
- the **mean curvature plot** serves as an invaluable tool for dissecting the latent space's structural intricacies. Computed as an approximate distance function to the data manifold, the mean curvature H is estimated using finite differencing across a grid in the latent space. This plot illuminates regions where the latent space is most likely to converge to the data manifold, indicated by high curvature values. In essence, high curvature corresponds to areas where $u(z)$ is small and locally convergent. The mean curvature plot not only aids in identifying these "high curvature regions" but also offers insights into the nonlinear characteristics of the latent manifold. For instance, regions with notably high curvature values can guide more meaningful interpolations between latent samples, ensuring that the chosen path stays proximal to the data manifold. This dual utility of the mean curvature plot—both as a visual guide and an analytical tool—enhances our understanding of the latent space's topology and its relationship with the underlying data manifold.

The introduced *latent responses* framework provides a systematic approach to probe and understand the latent space of variational autoencoders, which are a cornerstone in generative modelling. The tools developed under this framework allow for interventions in the latent space, thereby enabling a detailed analysis of how data manifolds are embedded and how latent variables interrelate within that space. This deep dive into the latent space not only facilitates a nuanced understanding of the generative processes, enhancing the explainability and interpretability of generative AI models [24], but also opens avenues for identifying inconsistencies and anomalies [3], [2] within the latent space. The ability to separate and analyse semantic information and noise within the latent variables, as well as to identify and quantify causal relationships therein, provides a robust mechanism for detecting anomalies and understanding their origins, thereby contributing to the development of more reliable and introspective generative AI models.

Further, investigating and understanding latent spaces using these tools can be pivotal in seeking more readable designs in artificial intelligence. By examining the latent space and exposing the complex connections between variables, researchers, and practitioners can acquire precious knowledge about how various design choices affect the learned portrayals and creative processes of the model. The capability to conduct interventions in the latent space and measure the connections between latent variables affords a means to

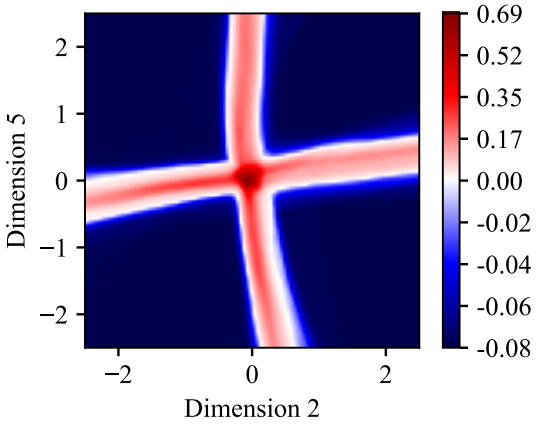

Figure 6: The divergence map allows the delineation of high and low response divergence regions.

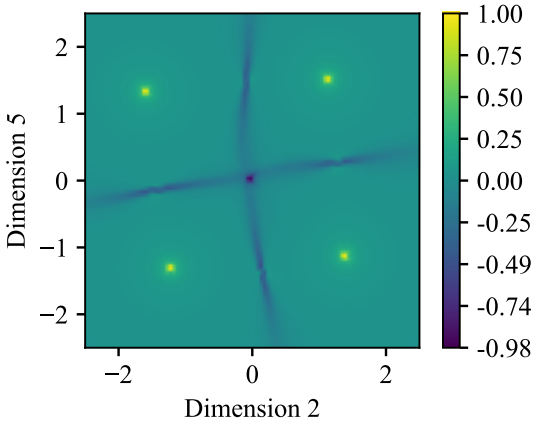

Figure 7: The mean curvature mean shows four high curvature spots corresponding to the four flows of the latent data manifold.

systematically explore the impacts of different architectural elements and training approaches on the model's capacity to learn significant and separated representations. This can, consequently, steer the creation of designs that are naturally more understandable, by lining up the learned latent variables more tightly with intuitive and semantically relevant factors of change in the data. Additionally, the understandings obtained from examining the latent space can also guide the creation of new regularization methods and training goals that expressly promote the learning of readable and causally relevant representations, thus making the models not only more comprehensible but also potentially robust and more adaptable across diverse and changing application domains.

V. Case studies

A. An introductory toy dataset

We commence our investigation by employing a VAE to train on a toy dataset. During this initial phase, the VAE is trained using a value of 2 for beta to ensure a disentangled representation in the latent space. This dataset represents trajectories and employs *longitude* and *latitude* as the sole feature dimensions.

The utilization of a response matrix allows us to concentrate our analytical efforts on dimensions characterized by the highest responses, indicating their capacity to encapsulate the majority of relevant information. Notably, dimensions 2 and 5 demonstrate such prominent responses.

In our examination of the toy dataset, we proceed to visualize the projection of dataset X onto the latent space along the two discerned latent dimensions. This projection yields the formation of four distinct clusters, a result that aligns with our expectations. Each cluster represents a flow in the toy dataset. The model successfully learned and encoded the initial flow of each trajectory on these two plotted dimensions.

Our analysis extends beyond the confines of the latent data manifold, as our objective encompasses the comprehensive exploration of the latent space. To this end, we present a divergence map that vividly delineates four distinct regions characterized by boundaries, indicating areas where the model encounters challenges in terms of reconstruction. Even though the aggregate posterior is highly concentrated at a few points, the negative divergence almost everywhere suggests that the extent of the decoder can handle extends well beyond the posterior, confirming the model's robustness and capability to manage a broader latent space. This is visually evidenced by the divergence plot, which, despite the concentration of the posterior in certain regions, illustrates that most

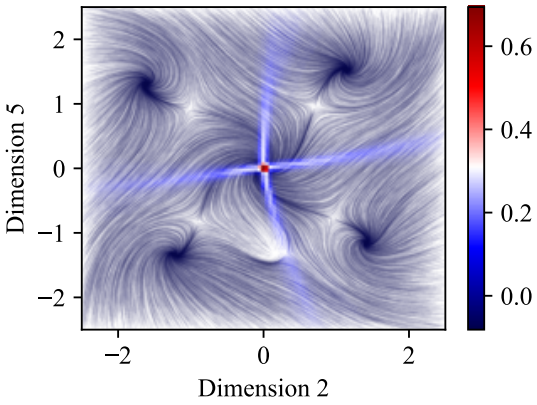

Figure 8: Response matrix on the toy dataset. The sources of vortexes are regions in the latent space where the encoder tends to project the encoded trajectories.

of the latent space, depicted in blue, does indeed converge rather than diverge, aligning with expectations and indicating a robust model. This robustness and convergence throughout the latent space, even in areas not densely populated by the posterior, underscore the model's stability and reliability in handling various data and generating consistent reconstructions, thereby enhancing its utility in practical applications and further research in generative models and anomaly detection.

Furthermore, we use the mean curvature map, highlighting regions of the latent space where the input trajectories are projected. High curvature regions, plotted in yellow on Figure 7, indicated areas of the latent space that are densely populated with meaningful data.

In our endeavour to combine the insights provided by both mean curvature and divergence, utilizing a composite visualization emerged as a pivotal strategy to encapsulate the multifaceted information embedded within the latent space of generative models. Given that both the mean curvature and divergence maps are derived from the same foundational element, the response map, we leveraged its grid of 2D vectors to craft a composite visualization that succinctly conveys the nuanced characteristics of the latent space.

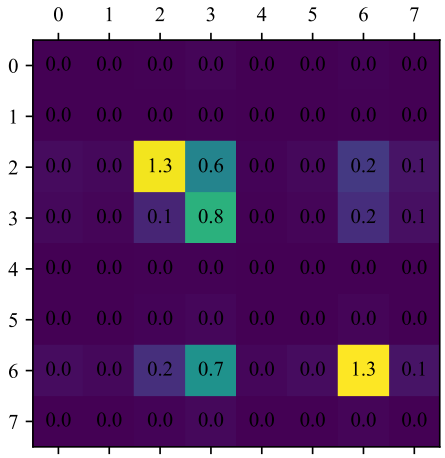

Figure 9: Response matrix on the Zurich dataset. Three dimensions show high responses with noticeable causal scores.

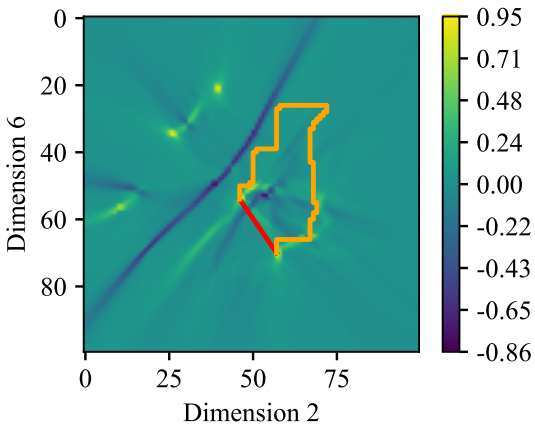

Figure 10: Mean curvature plot on the Zurich dataset. Two paths are shown: A direct path that crosses through a low curvature region and an optimized path that goes through high curvature regions corresponding to more realistic trajectories.

This approach was underpinned by extracting the principal sources of curve and curl from the response map, thereby illuminating the regions of high curvature and divergence within the latent space. Furthermore, we employed a strategic colour-coding scheme within the composite visualization to elucidate the various features of the response map, ensuring that each aspect, from areas of high divergence to regions of pronounced curvature, was distinctly and intuitively represented. Consequently, this composite visualization not only served as a coherent representation of the latent space but also facilitated a more holistic understanding of how the latent variables and their respective manifolds interact and evolve, thereby providing a unified view that seamlessly blends the insights offered by both the mean curvature and divergence maps into a single, comprehensive plot.

B. *Trajectories landing at Zurich airport*

In the investigative exploration of the Zurich dataset, characterized by two flows, a VAE is employed to elucidate the inherent latent structures and interrelationships embedded within the data. In this experiment, we use four input features to describe the trajectories, namely *track angle, ground speed, altitude* and *time*. As illustrated in Figure 9, the response matrix unveils that three dimensions manifest a pronounced latent response, indicating their pivotal role in encapsulating important information about the dataset. The utilization of a greater number of dimensions, in comparison to the toy dataset, is attributed to the relatively augmented complexity of the Zurich dataset. Furthermore, these dimensions register elevated causality scores, thereby underscoring the existence of complex and potentially non-linear relationships within the latent space of the Zurich dataset. Although we set aside most complex procedures from this dataset, there still is a big variation in the trajectories reflexed by the presence of multiple data curvature "peaks" highlighted in yellow on the figure 10. The features extracted within the latent space of this network may represent the different approaches present in the dataset.

Figure 11 presents a composite plot that utilizes colours to visually represent the areas in the latent space where our model encounters challenges regarding accurate and stable reconstructions. A detailed examination of this plot allows us to identify boundaries and regions exhibiting varying degrees of divergence. This analysis provides valuable insights into the stability and robustness of the latent representations learned by our model, as well as its generative capabilities within the context of a Variational Autoencoder (VAE).

In Figure 10, we explore two distinct paths through the latent space, each resulting in interpolated aircraft trajectories. Our focus here is on how these paths adhere to regions with differing mean curvature. The first path, characterized by a straight trajectory, represents the most direct route between two points in the latent space. However, it passes through areas with notably low curvature, suggesting that it traverses regions that lack dense data representation. Consequently, the aircraft trajectories reconstructed along this direct path may exhibit less realism and fail to conform to expected physical and operational constraints. This is because the path does not fully engage with the underlying data manifold within the latent space.

In contrast, the second path is specifically optimized to align with regions of high curvature. As a result, it inherently adheres more closely to the data manifold, navigating through areas in the latent space that are more likely to correspond to realistic and semantically meaningful data. This path's alignment with high curvature points ensures that the interpolated points and subsequently reconstructed aircraft trajectories are more likely to exhibit realism and align with observed data dynamics. Therefore, the disparity in the realism and reliability of the reconstructed trajectories from these two paths underscores the critical importance of considering the intrinsic geometry and curvature of the latent space in the generation and interpolation tasks within generative models.

VI. Conclusion and Future Works

This paper applied a technique introduced in [27] for providing detailed visualizations, with a special focus on exploring its latent space. Divergence plots and multifaceted composite plots outline areas where the model shows an inability to accurately reconstruct trajectories, leading to the appearance of gaps and irregularities within the latent space.

The mean curvature map provides a more practical mechanism for interpolating between trajectories, by optimizing a path across regions with high curvature, thereby ensuring a constant presence in areas where the data manifold is notably prevalent. These optimized paths enable the creation of realistic trajectories, which can prove useful when working on designing new procedures, assessing collision risk probabilities, or other events. This work not only offers insights into the reliability of autoencoders but also advocates an improved methodological approach for exploring and understanding

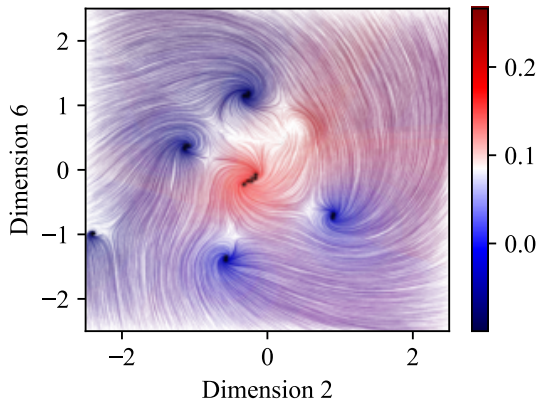

Figure 11: Composite visualization on the Zurich dataset.

their latent spaces, and paving the way for an improved acceptability by operational stakeholders.

The long-term objective of such an approach is to provide explainable ML models for anomaly detection: a good understanding of the structure of the latent space with existing and more advanced tools is expected to help to provide models able to give operational hints about detected anomalies based on the structure of the local neighbourhood in the latent space, thereby bridging the gap toward more explainable AI in the air traffic management domain.

References

[1] X. Olive, L. Basora, B. Viry, and R. Alligier, "Deep Trajectory Clustering with Autoencoders," in *Proceedings of the 9th International Conference on Research in Air Transportation*, 2020.

[2] X. Olive, J. Sun, A. Lafage, and L. Basora, "Detecting Events in Aircraft Trajectories: Rule-Based and Data-Driven Approaches," in *Proceedings of the 8th OpenSky Symposium*, Nov. 2020.

[3] X. Olive and L. Basora, "Identifying Anomalies in past en-route Trajectories with Clustering and Anomaly Detection Methods," in *Proceedings of the 13th USA/Europe Air Traffic Management Research and Development Seminar*, (Vienna, Austria), June 2019.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[5] A. Degas, M. R. Islam, C. Hurter, S. Barua, H. Rahman, M. Poudel, D. Ruscio, M. U. Ahmed, S. Begum, M. A. Rahman, S. Bonelli, G. Cartocci, G. Di Flumeri, G. Borghini, F. Babiloni, and P. Aricó, "A Survey on Artificial Intelligence (AI) and eXplainable AI in Air Traffic Management: Current Trends and Development with Future Research Trajectory," *Applied Sciences*, vol. 12, p. 1295, Jan. 2022.

[6] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in beta-VAE," *arXiv:1804.03599*, 2018.

[7] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proceedings of the International Conference on Learning Representations*, 2016.

[8] A. Binder, G. Montavon, S. Lapuschkin, K.-R. Müller, and W. Samek, "Layer-wise relevance propagation for neural networks with local renormalization layers," in *Proceedings of the 25th International Conference on Artificial Neural Networks*, pp. 63–71, 2016.

[9] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," *arXiv:1602.04938 [cs, stat]*, Aug. 2016.

[10] C. Hurter, B. Tissoires, and S. Conversy, "FromDaDy: Spreading data across views to support iterative exploration of aircraft trajectories," *IEEE TVCG*, vol. 15, no. 6, pp. 1017–1024, 2009.

[11] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, June 2020.

[12] T. Rojat, R. Puget, D. Filliat, J. Del Ser, R. Gelin, and N. Díaz-Rodríguez, "Explainable artificial intelligence (xAI) on timeseries data: A survey," *arXiv:2104.00950*, 2021.

[13] M. Veerappa, M. Anneken, N. Burkart, and M. F. Huber, "Validation of XAI explanations for multivariate time series classification in the maritime domain," *Journal of Computational Science*, vol. 58, p. 101539, Feb. 2022.

[14] T. Sainburg, M. Thielk, and T. Q. Gentner, "Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires," *PLOS Computational Biology*, vol. 16, pp. 1–48, Oct. 2020.

[15] Y. Liu, E. Jun, Q. Li, and J. Heer, "Latent Space Cartography: Visual Analysis of Vector Space Embeddings," *Computer Graphics Forum*, vol. 38, pp. 67–78, June 2019.

[16] X. Liu and J. Wang, "LatentVis: Investigating and Comparing Variational Auto-Encoders via Their Latent Space," in *Proceedings of the 3rd Workshop Advances in Interpretable Machine Learning and Artificial Intelligence*.

[17] E. Mathieu, T. Rainforth, N. Siddharth, and Y. W. Teh, "Disentangling Disentanglement in Variational Autoencoders," *arXiv:1812.02833 [cs, stat]*, June 2019.

[18] W. Zeng, Z. Xu, Z. Cai, X. Chu, and X. Lu, "Aircraft trajectory clustering in terminal airspace based on deep autoencoder and gaussian mixture model," *Aerospace*, vol. 8, no. 9, 2021.

[19] L. Wang, P. Lucic, K. Campbell, and C. Wanke, "Autoencoding features for aviation machine learning problems," 2020.

[20] M. Memarzadeh, B. Matthews, and I. Avrekh, "Unsupervised Anomaly Detection in Flight Data using Convolutional Variational Auto-Encoder," *Aerospace*, vol. 7, no. 8, 2020.

[21] S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *arXiv:1705.07874 [cs, stat]*, Nov. 2017.

[22] L. Antwarg, R. M. Miller, B. Shapira, and L. Rokach, "Explaining anomalies detected by autoencoders using Shapley Additive Explanations," *Expert Systems with Applications*, vol. 186, p. 115736, Dec. 2021.

[23] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv:1312.6114 [cs, stat]*, Dec. 2013.

[24] T. Krauth, A. Lafage, J. Morio, X. Olive, and M. Waltert, "Deep generative modelling of aircraft trajectories in terminal maneuvering areas," *Machine Learning with Applications*, vol. 11, p. 100446, Mar. 2023.

[25] X. Olive and L. Basora, "Reference data sets for detection and identification of significant events in historical aircraft trajectory data.," 12 2019. https://doi.org/10.6084/m9.figshare.11406735.v1.

[26] X. Olive, "traffic, a toolbox for processing and analysing air traffic data," *Journal of Open Source Software*, vol. 4, p. 1518, 2019.

[27] F. Leeb, S. Bauer, M. Besserve, and B. Schölkopf, "Exploring the Latent Space of Autoencoders with Interventional Assays," *arXiv:2106.16091 [cs]*, Jan. 2023.