

Knowledge-Induced Learning with Adaptive Sampling Variational Autoencoders for Open Set Fault Diagnostics

Working Paper**Author(s):**

Arias Chao, Manuel; Adey, Bryan T.; Fink, Olga

Publication date:

2019-12-28

Permanent link:

<https://doi.org/10.3929/ethz-b-000430653>

Rights / license:

In Copyright - Non-Commercial Use Permitted

Originally published in:

arXiv

Funding acknowledgement:

176878 - Data-Driven Intelligent Predictive Maintenance of Industrial Assets (SNF)

KNOWLEDGE-INDUCED LEARNING WITH ADAPTIVE SAMPLING VARIATIONAL AUTOENCODERS FOR OPEN SET FAULT DIAGNOSTICS

A PREPRINT

Manuel Arias Chao
ETH Zurich
manuel.arias@ethz.ch

Bryan T. Adey
ETH Zurich
adey@ibi.baug.ethz.ch

Olga Fink
ETH Zurich
ofink@ethz.ch

January 1, 2020

ABSTRACT

The recent increase in the availability of system condition monitoring data has led to increases in the use of data-driven approaches for fault diagnostics. The accuracy of the fault detection and classification using these approaches is generally good when abundant labelled data on healthy and faulty system conditions exists and the diagnosis problem is formulated as a supervised learning task, i.e. supervised fault diagnosis. It is, however, relatively common in real situations that only a small fraction of the system condition monitoring data are labeled as healthy and the rest is unlabeled due to the uncertainty of the number and type of faults that may occur. In this case, supervised fault diagnosis performs poorly.

Fault diagnosis with an unknown number and nature of faults is an *open set* learning problem where the knowledge of the faulty system is incomplete during training and the number and extent of the faults, of different types, can evolve during testing. Open set fault diagnostics problems are more difficult than those with abundant labelled data as there are observations that do not have already known system conditions. In this paper, we propose a new method to perform diagnostics for these types of problems.

The open set diagnostics problem is formulated as a semi-supervised learning problem and we demonstrate how it can be solved using a knowledge-induced learning approach with adaptive sampling variational autoencoders (KIL-AdaVAE) in combination with a one-class classifier. The proposed method introduces an additional loss-term into the optimisation objective that regularizes only the part of the training set labelled "healthy" thereby restricting the representation of the healthy system conditions in the latent space of the variational autoencoder. The fault detection and segmentation capability of the proposed method is demonstrated on a simulated case study using the Advanced Geared Turbofan 30000 (AGTF30) dynamical model under real flight conditions and induced faults of 17 fault types. The performance of the method is compared to the different learning strategies (supervised learning, supervised learning with embedding and semi-supervised learning) and deep learning algorithms. The results demonstrate that the proposed method is able to significantly outperform all other tested methods in terms of fault detection and fault segmentation.

Keywords open-set diagnostics, deep learning, variational autoencoders (VAE)

1 Introduction

Faults in complex industrial assets can significantly affect system safety, system availability, system performance and can be expensive. Early fault detection can, therefore, significantly reduce these impacts. Before system failure occurs, it is important to detect fault signatures and distinguish between the possible fault types. The information on the fault types enables maintenance engineers to prepare properly for the maintenance intervention, limiting costs and down time.

The increased availability of condition monitoring data (CM) has triggered the use of data driven approaches for fault detection and diagnostics with the underlying assumption that the relevant information on the health of the system can be learned from past data. The development of data-driven diagnostics models requires a representative dataset covering the possible expected operating conditions and possible fault types. The collection of a representative dataset can take a long time as (a) faults are rare, (b) frequent system health assessments are often expensive or impracticable, and (c) the system operates in different environments and follows different mission profiles resulting in a large range of possible deterioration trajectories. Due to these difficulties and the fact that operators typically require that a fault diagnostics system is available early in the life time of a system, the available datasets are, typically, not fully representative of the system behaviour during the development phase of the diagnostics model. In other words, the available dataset does not contain samples of all possible fault types, and in many cases, it may only consist of data on the healthy system, i.e. without any faults. The high variability of the operating conditions, the evolution of how the system operates over time, the high number of possible system conditions and the occurrence of previously unknown fault types make the development of methods for reliable data-driven fault diagnostics highly challenging.

The most common methods to solve fault diagnosis problems in literature have been developed using supervised learning approaches, i.e. learning the underlying patterns from the previously observed condition monitoring data with known labels from different fault types. Examples of such approaches include Support Vector Machines (SVM) [1] and different types of deep learning algorithms, such as convolutional neural networks (CNN) [2] and Long Short-Term Memory Networks (LSTM) [3]. Supervised learning approaches, however, will perform poorly when applied to newly evolving faults since the underlying patterns will be dissimilar to those learned.

One way of overcoming the need to have a large amount of labelled data (i.e. samples where the condition of the system is known) is to formulate the problem as a semi-supervised learning problem and use both the existing labelled data and data with yet unknown system conditions i.e. unlabeled data. Some examples of the use of semi-supervised approaches in the field of prognostics and health management include self-training [4], graph-based methods [5] and co-training methods [6]. Semi-supervised approaches have been mainly focusing on prognostics problems and there has been relatively few solutions for diagnostics problems. Most of the semi-supervised approaches for diagnostics focus on feature extraction [7, 8] for multi-class classification problems where the training dataset is representative of all the possible fault types (i.e. each fault type is a class). However, to the best of our knowledge, semi-supervised approaches have not yet been used to solve diagnostics problems with fault types that have not been seen in the training dataset.

Several methods have been proposed to identify fault types that have not been seen in the training dataset. We refer to the process of identifying and separating faults of different types as fault segmentation. Clustering approaches directly applied on the CM data [9, 10, 11, 12, 13, 14] are the most widely used fault segmentation approaches. Other approaches applied for fault segmentation include adaptive dictionary learning [15] or causal maps combined with multivariate statistics [16]. Clustering approaches, particularly those based on adaptive density clustering algorithms, have had considerable success in identifying different fault types when the feature space (i.e. either the raw CM data or manually extracted features) is easily separable with respect to the different fault types. However, they do not work well when the feature space is not easily separable, i.e. the dimension of the CM data is high and/or the signals are noisy and correlated. In such cases, it is normally difficult to interpret distances in the feature space and, therefore, to define a valid proximity or similarity metric. This results in some fault types being inseparable, leading to clusters with faults of more than one fault type. Methods using clustering algorithms do, however, perform increasingly better in a more compact and informative representation of the input space. Autoencoders (AE) are typically used to learn a lower dimensional representation of the input space, also referred to as embedding, latent or feature space. This does not, however, guarantee that the latent space provides an improved ability to distinguish between fault types.

One of the possible autoencoders applied to obtain a more informative and compact latent space are Variational Autoencoders (VAE). VAE [17] are generative latent models based on neural networks and have performed very well when used to solve classification and regression problems [17]. One of their strengths is their interpretability and their compact representation of the latent space, which is favourable for fault diagnosis tasks. VAE are, therefore, preferably used instead of traditional AEs due to their improved control of how the latent space is modeled. The prior distribution in VAE can enforce a more favorable latent space representation of the healthy and faulty system conditions.

Several extensions to VAEs have previously been proposed, including the β -VAE [18] that are supposed to favor a disentangled representation in the latent space. While some of the characteristics of β -VAE are advantageous for the task of an open set fault diagnosis, some are disadvantageous since they limit the performance of the fault diagnostics algorithm. Concretely, imposing a close mapping to the prior distribution (originating from large values of β) results in a loss of information about the system data on its representation in the latent space. This phenomenon is well known and also referred to as the posterior collapse [19]. Additionally, it leads to a potential overlap of the representation of different system conditions.

In this paper, we propose to use a semi-supervised learning strategy for fault diagnostics in the situation when the labelled data is only originating from the healthy system conditions, i.e. there is no labelled data associated with faulty system conditions. Additionally, we assume a realistic scenario that the number and nature of the fault types that may occur are not known and the labels not available for training. In the classical problem formulation of semi-supervised learning, the training dataset will contain a sample representation of all the fault types that are expected in the test dataset. However, in our case, at the model development phase, the system will only have experienced normal system conditions and the number and nature of the fault types that may occur are not known. This is a more difficult and realistic, problem than the one where the training dataset is entirely representative of the input space. Increasing the complexity of the task, we assume there is still a need to be able to distinguish between different fault types that may occur at testing even if the only available labeled data available at training are from the healthy system state. This problem setup for fault diagnosis is a special case of an *open set* learning problem [20] where the knowledge of the possible fault types is incomplete at training time, and unknown fault types may occur at testing.

We propose a Knowledge-Induced Learning with Adaptive Sampling Variational Autoencoders (KIL-AdaVAE), an extension of the traditional Variational Autoencoders (VAE), for open set fault diagnostics. The method is developed to address the open set diagnostics problem as a semi-supervised one-class classification problem, where the embedding representations of the known healthy system condition are considered to have the one-class label "healthy". The proposed method involves a) the use of all the information available at the model development phase, including the unlabeled data samples; b) the integration of the available prior knowledge on the healthy conditions in the algorithm design and the learning process through an adjusted loss function and an adjusted sampling algorithm. The formulation as a semi-supervised learning problem enables an improved representation of the signals in the latent space. The adjustment of the sampling procedure of the VAE encoder network enforces informative samples in the latent space. The introduction of an additional loss term that regularizes the healthy data in the training data set restricts the embedding representation of the healthy data.

For segmentation of the fault types, we propose to cluster the faults within the latent feature space of the proposed VAE using a density-based clustering algorithm without any prior knowledge on the type and number of faults.

The proposed method is evaluated on a fault detection and segmentation task in the Advanced Geared Turbofan 30000 (AGTF30) case study using a dataset generated with the dynamical model [21] under real flight conditions and induced faults of 17 different types. The proposed method is compared to other deep learning algorithms with different learning strategies, i.e. a) Supervised one-class learning, b) Supervised one-class learning with embedding, c) Semi-supervised one-class learning with embedding. The results show that the proposed KIL-AdaVAE method is able to outperform others in terms of fault detection accuracy and fault segmentation capability of the unknown fault types. It is also shown that the use of the unlabeled data during training in a semi-supervised way, with the additional regularization, improves the performance of the algorithm.

The remainder of the paper is organized as follows. In Section 2 the problem is formally introduced and the background of the solution strategy is provided. In Section 3 the proposed method is described. In Section 4 the applied case study is introduced and the experiment is explained. In section 5 the results are given. Finally, a summary of the work and outlook are given in Section 6.

2 Background

2.1 Problem Statement

The formulation of the diagnostics problem addressed in this paper is formally introduced in the following. We aim at developing a diagnostics model at time t_a from a multivariate time series of condition monitoring sensors readings $X = [x^{(1)}, \dots, x^{(m)}]^T$, where each observation $x^{(i)} \in R^n$ is a vector of n raw measurements. The corresponding *true* system health condition (i.e. healthy or faulty) is partially known and denoted as $H_s = [h_s^{(1)}, \dots, h_s^{(m)}]^T$ with $h_s^{(i)} \in \{0, 1\}$. Concretely, we consider the situation where certainty about healthy system conditions (i.e. $h_s^{(i)} = 1$) is only available until a past point in time t_b when the system condition was assessed and confirmed as healthy by maintenance engineers, e.g. during inspection. Our partial knowledge of the true health allows, therefore, the definition of two subsets of the available data: a *labeled* dataset $\mathcal{D}_L = \{(x^{(i)}, h_s^{(i)})\}_{i=1}^u$ with $h_s^{(i)} = 1$ corresponding to known healthy system conditions and an *unlabeled sample* $\mathcal{D}_U = \{x^{(i)}\}_{i=u+1}^m$ with unavailable health label (i.e. the system conditions are not associated with either healthy or faulty system condition). During model development, there is therefore only data that has been labelled healthy. The unlabeled dataset is expected to contain data pertaining to both healthy system conditions and faulty system conditions. However, neither the number nor the type of faults are known. The scenario considered in this paper is the one where K fault types are present in \mathcal{D}_U . This represents the common

and realistic situation where records about faults that have occurred in the field are not available at analysis time. A schematic representation of the problem setup is provided in Figure 1.

Given this set-up, the first task is to detect the faulty system conditions within D_U given our available dataset $\mathcal{D} = \mathcal{D}_L \cup D_U$ at time t_a ; which is a transductive learning problem. In other words, this first task involves determining a reliable direct or indirect mapping from the raw data X to the possible system conditions (i.e. \hat{h}_s) on D_U . The generalization capability of the model to detect K_* new faults types is validated in a test dataset $\mathcal{D}_T = \{x_*^{(j)}\}_{j=1}^M$.

The second task is to extend the diagnostics scope to provide an adequate unsupervised segmentation of fault types present in D_U and \mathcal{D}_T . We refer to $\mathbf{V} = \{V_j | j = 1, \dots, C\}$ as the partition of $\{D_U, \mathcal{D}_T\}$ according to the $C = K + K_* + 1$ true system states or condition classes, with $K + K_*$ faulty states and one healthy state.

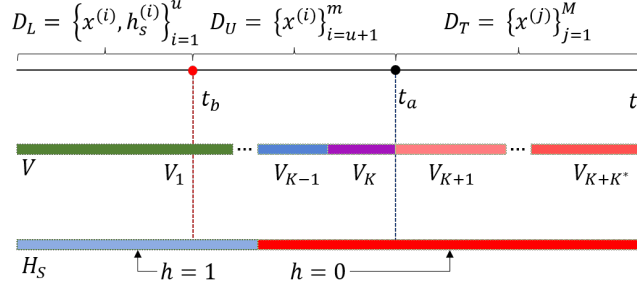


Figure 1: Schematic representation of the problem. Training dataset D has labeled (D_L) and unlabeled data (D_U). The test set (D_T) has only unlabeled samples. The true system conditions any point in time is represented by the H_S bar. Healthy conditions are represented in blue and faulty in red. K fault types are present in D_U and K_* in D_T . The true system states are represented by the V bar. Each fault type and the healthy conditions appears in different color.

2.2 Fault Detection

The fault detection problem has been successfully addressed as a *one-class classification* problem using Hierarchical Extreme Learning Machines (HELM) in [22]. In this case, the task turns to a regression problem that aims at discovering a functional map \mathcal{G} from the healthy system conditions to a target label $\mathbf{T} = \{h_s^{(i)} | x^{(i)} \in S_T\}$ where $S_T \subsetneq D_L$ is a training subset of D_L . We refer to a neural network that discovers the functional map \mathcal{G} as the *one-class network*. The output of the one-class network will deviate from the target value \mathbf{T} when the inner relationship of a new data point $x^{(j)} \in \{D_U, \mathcal{D}_T\}$ does not correspond to the one observed in S_T . Therefore, we consider an unbounded similarity score $s_I(x^{(j)}; \beta)$ of $x^{(j)}$ with respect to the healthy labeled data based on the prediction of $\mathcal{G}(x^{(j)})$ that we define as follows:

$$s_I(x^{(j)}; \xi) = \frac{|\mathbf{T} - \mathcal{G}(x^{(j)})|}{\xi} \quad (1)$$

$$\xi = P_{99.9}(|(\mathbf{T} - \mathcal{G}(S_V))|)1.5 \quad (2)$$

where ξ corresponds to a normalizing threshold given in this case study by the 99.9% percentile of the absolute error of the prediction of \mathcal{G} in a validation set (i.e S_V) extracted from D_L multiplied by a safety margin 1.5. Please note that the percentile and the safety margin are model hyperparameters.

Hence, our fault detection algorithm is given by:

$$\hat{h}_s(x^{(j)}) = \begin{cases} 1 & s_I(x^{(j)}; \xi) < 1 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

To obtain the mapping function \mathcal{G} , we considered three *learning strategies*: supervised learning (SL), supervised learning with embedding (SLE) and semi-supervised learning (SSL) given only one target label $h_s^{(i)} = 1$. The one-class classification problem with these three learning strategies are introduced in the following.

Supervised One-Class Learning (SOCL). A supervised learning strategy aims at discovering a mapping from the input signals (i.e sensor readings) to a target label given a training set [23]. Therefore, we can ignore the unlabeled dataset D_U and treat our fault detection problem as a supervised regression problem. The task is then to find a functional

mapping $\mathcal{G}_s : X_L \mapsto \mathbf{T}$ from the input signals $X_L = \{x^{(i)} \in S_T\}$ to the label class \mathbf{T} . To find \mathcal{G}_{sl} , a discriminative model based on deep feed-forward neural network is considered. Our SL-FF model implements this solution strategy (see Section 2.4 for further details).

Supervised One-Class Learning with Embedding (SOCLE). An alternative strategy to the direct mapping \mathcal{G}_s is to obtain a representation of the raw input data (a.k.a. non-linear embedding) from which a reliable optimal mapping \mathcal{G} can be learned. The task has typically two parts. Firstly, a transformation $E : X_L \mapsto \mathbf{z}_L$ of the input signals to a latent space $\mathbf{z}_L \in R^{m \times d}$ (with $d < n$) is found. The resulting latent space encodes optimal distinctive features of X_L in an unsupervised way (i.e. without having information on the labels). In a second step, a functional mapping $\mathcal{G}_{sle} : E(X_L) \mapsto \mathbf{T}$ from the latent space $\mathbf{z}_L = \{E(x^{(i)}) \mid x^{(i)} \in S_T\}$ to the target label \mathbf{T} is learnt. Generative models based on feed-forward Variational Autoencoders (SLE-VAE and SLE- β -VAE) are implemented to discover the latent representation z_L . For comparison purposes, two discriminative models based on deep feed-forward autoencoders (SLE-AE) and Hierarchical Extreme Learning Machines (SLE-HELM) are also implemented (see Section 2.4 for further details).

Semi-Supervised One-Class Learning (SSOCL). The two learning strategies described above do not make use of the *unlabeled* dataset \mathcal{D}_U available at training time. However, the discovery of the mapping function \mathcal{G} can also be formulated as a semi-supervised learning problem to take advantage of the relevant information that may be present in \mathcal{D}_U . Concretely, semi-supervised learning methods consider both, labeled and unlabeled data during training. Therefore, we define an alternative unsupervised problem $F : X \mapsto \mathbf{z} \mapsto X$ to include all the available sensor data X and obtain a latent representation $\mathbf{z} \in R^{n \times d}$ that encodes features of healthy and potentially faulty states. Similarly as before, in a second step, we find a supervised mapping $\mathcal{G}_{ssl_{M1}} : \mathbf{z}_L \mapsto \mathbf{T}$. Hence, this framework is similar to the M1 model introduced by Kingma et al. in [24]. A variant of this method is to find an optimal mapping $\mathcal{G}_{ssl_{M2}} : X \mapsto \mathbf{z} \mapsto X, \mathbf{T}$ where the latent feature space is also conditioned to provide an optimal mapping for the target label \mathbf{T} . This framework corresponds to the M2 model introduced by Kingma et al. in [24]. However, in this paper, we consider an alternative formulation where we introduce an inductive bias about the labeled data distribution (i.e. healthy data) without explicitly incorporating the target loss. We refer to this option as *knowledge induced learning* (KIL). More details of the proposed method are given in Section 3. Generative models based on feed-forward Variational Autoencoders (SSL-M1-VAE and KIL-VAE) are implemented to discover the latent representation z_L (Section 2.4).

2.3 Fault Segmentation

The latent space z obtained with semi-supervised and supervised with embedding learning methods provide an alternative representation of the input signals. This encoded representation can reveal hidden patterns that make faulty system conditions clearly detectable. Hence, we applied a density-based clustering algorithm to the latent space z to discover unknown system conditions present in the dataset $\{D_U, D_T\}$. Concretely, we used the density-based OPTICS algorithm *Ordering points to identify the clustering structure* to group points $z^{(i)} \in \{D_U, D_T\}$ that are close to each other based on a metric of distance (i.e. Euclidean distance) and a minimum number of points. We selected the OPTICS algorithm because of its ability to detect meaningful clusters in data with varying density. Hence, OPTICS addresses a major weakness of the commonly applied *density-based spatial clustering of applications with noise* (DBSCAN) algorithm [25].

Visualization of the resulting clusters is carried out with a two dimensional representation of the high dimensional latent space (z) using the t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm [26].

2.4 Generative Models

One-class supervised with embedding and semi-supervised learning strategies use autoencoder networks to obtain the latent representation z . Since our proposed method resorts to generative latent models they are introduced in this section.

Generative latent models assume that an observed variable x is generated by some random process involving an unobserved random (i.e. latent) variable z . Hence, latent models define a joint distribution $p(x, z) = p(x|z)p(z)$ between a latent space z , and the input space x [27]. Thus, the underlying generation process resorts to two steps: 1) a value $z^{(i)}$ is generated from some prior distribution $p(z)$ and 2) a value $x^{(i)}$ is generated from some conditional distribution $p(x|z)$. The data generation process is modeled by a complex conditional distribution $p_\theta(x|z)$, which is often parameterized with a neural network. Our proposed method is an extension of the VAEs which are introduced in the following.

Variational Autoencoders. VAEs aim at sampling values of z that are likely to have produced x and compute $p(x)$ from those [28]. VAE models comprise an inference network (encoder) and a generative network (decoder). Contrarily

to discriminative autoencoder models, the latent representation z of the data x is a stochastic variable. Therefore, the encoder and the decoder networks are probabilistic. The inference network $q_\phi(z|x)$ parametrizes the intractable posterior $p(z|x)$ and the generative network $p_\theta(x|z)$ parametrizes the likelihood $p(x|z)$ with parameters θ and ϕ respectively. These parameters are the weights and biases of a neural network. Typically, a simple prior distribution $p(z)$ over the features is assumed (such as Gaussian or uniform).

The natural training objective of a generative model is to maximize the (marginal) likelihood of the data

$$\mathbb{E}_{p(x)}[\log p_\theta(x)] = \mathbb{E}_{p(x)}[\mathbb{E}_{p(z)}[\log p_\theta(x|z)]] \quad (4)$$

However, direct optimization of the likelihood is intractable since $p_\theta(x) = \int_z p_\theta(x|z)p(z)dz$ requires integration [27]. Hence, VAEs consider an approximation of the marginal likelihood denoted Evidence Lower BOUND (ELBO); which is a lower bound of the log likelihood (i.e. $\mathcal{L}_{\text{ELBO}} \leq \mathbb{E}_{p(x)}[\log p_\theta(x)]$)

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{p(x)}[\mathbb{E}_{q_\phi(x|z)}[\log p_\theta(x|z)] - D_{\text{KL}}(q_\phi(z|x)||p(z))] \quad (5)$$

where D_{KL} denotes the Kullback-Leibler (KL) divergence. Hence, the training objective of VAEs is to optimize the lower bound with respect to the variational parameters ϕ and the generative parameters θ

$$\max_{\phi, \theta} \mathbb{E}_{p(x)}[\mathbb{E}_{q_\phi(x|z)}[\log p_\theta(x|z)] - D_{\text{KL}}(q_\phi(z|x)||p(z))] \quad (6)$$

The ELBO objective is the sum of two components. The first term is the expected negative reconstruction error and is equivalent to the training objective of an autoencoder. The second term, i.e. the KL divergence, is a distance measure between two probability distributions (i.e. $D_{\text{KL}} \geq 0$). Hence, the D_{KL} term acts as a regularizer of ϕ trying to keep the approximate posterior $q_\phi(z|x)$ close to the prior $p(z)$.

As default assumption in VAE, the variational approximate posterior $q_\phi(z|x)$ follows a multivariate Gaussian with diagonal covariance (i.e. $q_\phi(z|x) = \mathcal{N}(z; \mu, \sigma^2 \mathbf{I})$). The distribution parameters of the approximate posterior μ and $\log \sigma^2$ are the non-linear embedding of the input x provided by the encoder network with variational parameters ϕ . The encoder output is, therefore, a parametrization of the approximate posterior distribution. Under these assumptions, a valid local reparametrization of z that allows to sample from the assumed Gaussian approximate posterior (i.e. $z^{(i)} \sim q_\phi(z|x^{(i)})$) is:

$$z^{(i)} = \mu^{(i)} + \sigma^{(i)} \odot \epsilon \quad (7)$$

with $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Although an isotropic Gaussian on the approximate posterior and standard Gaussian on the prior are the most common assumptions, the VAE framework is not limited to them. In certain cases, such approximations can lead to poor hidden representations and, therefore, there exists a variety of ways that extend the variational family to enrich the latent representation. One common approach is to scale the D_{KL} divergence term in the ELBO expression to encourage desired properties of the resulting encoder representation. Higgins et al. [18] proposed the β -VAE that introduce a weight to the D_{KL} term

$$\mathcal{L}_{\beta\text{-VAE}} = \mathbb{E}_{p(x)}[\mathbb{E}_{q_\phi(x|z)}[\log p_\theta(x|z)] - \beta D_{\text{KL}}(q_\phi(z|x)||p(z))] \quad (8)$$

The β -VAE offers a trade-off between the information preservation, i.e., how well one can reconstruct x from the z , and the capacity, i.e., how well the z compress information about x . Setting a $\beta > 1$ forces the encoder to better match the factorized unit Gaussian (limiting the capacity of z) and deteriorates the reconstruction of x . The former can better be observed by reformulating the D_{KL} term as follows:

$$\mathbb{E}_{p(x)}[D_{\text{KL}}(q_\theta(z|x)||p(z))] = I(x; z) + D_{\text{KL}}(q(z)||p(z)) \quad (9)$$

For $\beta \geq 1$ the β -VAE penalizes the mutual information between z and x (i.e. $I(x; z)$) but also enforces the so-called aggregated posterior $q(z)$ to factorize to match the prior $p(z)$. This factorization encourages a disentangled representation of the factors of variation in the data x . Hence, it can help to obtain an informative representation of independent factors of variations in the data.

A different strategy is to chose families of distributions for the prior and approximate posterior, $p_\theta(z)$ and $q_\phi(z|x)$ that best suit the nature of the problem at hand. Some examples are location-scale family of distributions (i.e. $z = \text{location} + \text{scale} \odot \epsilon$) such us Laplace, Logistic and composition of those or in general probability density function that are analytically tractable (explicit models).

3 Proposed Method: Knowledge Induced Learning with Adaptive Sampling Variational Autoencoder (KIL-AdaVAE)

Carefully considering the D_{KL} term one can encourage desired properties of the encoder representation that can help the fault detection task. In particular, in a *supervised set-up*, $\beta > 1$ forces the encoded representation of all the factors of variation present in the (healthy) training set to better match the factorized unit Gaussian. This forces an implicit discriminative representation of the healthy system conditions with respect to the faulty. The resulting latent representation can, therefore, be very informative for the detection network. We considered this approach in the SLE- β -VAE model with $\beta = 5$.

However, for large values of β , the latent representation does not convey any information about the input and the reconstruction of the signal deteriorates. Consequently, the resulting representation is also uninformative for downstream regression or classification tasks. This effect is known as the *posterior collapse*. To mitigate this issue and decouple the balance between reconstruction and regularization imposed by the ELBO without altering the training objective, we propose the following simple but yet effective modification of the sampling approach

$$z^{(i)} = \underbrace{\mu^{(i)}}_{\text{location}} + \underbrace{\alpha \log((\sigma^{(i)})^2)}_{\text{scale}} \odot \epsilon \quad (10)$$

The proposed sampling has a functional *scale* that encourages informative samples $z^{(i)}$ while keeping the distribution parameters of the approximate posterior $\mu^{(i)}$ and $\sigma^{(i)}$ close to the prior. This effect is best reasoned with a graphical example. The impact of the functional *scale* in the sampled $z^{(i)}$ is given by the $z^{(i)} - \mu^{(i)}$ term. Figure 2 provides a comparison of the $z^{(i)} - \mu^{(i)}$ terms as a function on $\sigma^{(i)}$ for the proposed sampling (i.e. $z = \mu + \log(\sigma^2) \cdot \epsilon$) and the standard VAE sampling (i.e. $z = \mu + \sigma \cdot \epsilon$). The 95% and 5% percentiles of $z^{(i)} - \mu^{(i)}$ with both sampling are depicted. The contour plot in the background shows the values of D_{KL} as a function of σ for $\mu = 0$. Kullback-Leibler divergence (D_{KL}) is a function of μ and σ and has a minimum at $\mu = 0$ and $\sigma = 1$. The proposed functional scale results in samples $z^{(i)}$ with low variance when $D_{KL} \approx 0$. Concretely, if $\sigma^{(i)} \approx 1$ (i.e. $\log(\sigma^{(i)})^2 \approx 0$) then the generated samples $z^{(i)}$ are maintained close to the mean $\mu^{(i)}$ (i.e. $z^{(i)} \approx \mu^{(i)}$). Since $\sigma^{(i)}$ is matching the prior, the ELBO objective encourages values of $\mu^{(i)}$ that are informative (i.e. low reconstruction error) while being close to $\mu^{(i)} \approx 0$. In contrast, a standard sampling leads to samples $z^{(i)}$ with higher variance in areas of low D_{KL} ; which results in a loss of information about $x^{(i)}$ in the encoded representation $\mu^{(i)}$ and the sample $z^{(i)}$. Besides this descriptive reasoning, experimental evidences demonstrating the impact of the sampling in the reconstruction error (i.e. \mathcal{L}_{ℓ_2}), the optimisation objective (i.e. $\mathcal{L}_{\beta\text{-VAE}}$) and the mutual information between $x^{(i)}$, $\mu^{(i)}$ and $z^{(i)}$ are provided in Section 7.1.

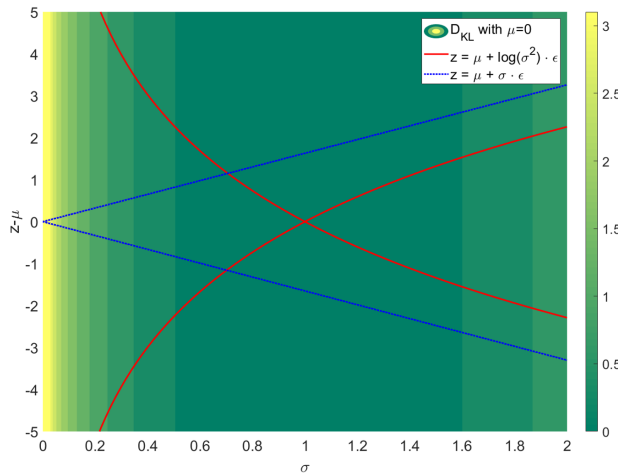


Figure 2: 95% (top) and 5% (bottom) percentiles of the $z - \mu$ term for standard VAE sampling (blue dashed lines) and the proposed adaptive sampling (red continuous lines). The region between these percentiles represent $(z - \mu)$ is the possible sampling region of each method as function of σ . The contour plot shows the Kullback-Leibler divergence (i.e. D_{KL}) is a function σ for $\mu = 0$. The D_{KL} term regularize the reconstruction error by penalizing values of $\mu \neq 0$ and $\sigma \neq 1$. D_{KL} has a minimum at $\mu = 0$ and $\sigma = 1$.

The proposed sampling deviates from an isotropic Gaussian, however, for simplicity, we regularize the optimisation objective with a $D_{KL}(q_\phi(z|x)|p_\theta(z))$ computed under the hypothesis that both $p_\theta(z)$ and $q_\phi(z|x)$ are Gaussian distribu-

tions. Hence, the D_{KL} term can be computed analytically and we can evaluate the impact of the proposed sampling with the same optimisation objective as the standard VAE.

$$-D_{KL}(q_\phi(z|x)||p(z)) = \sum_{j=1}^d (1 - \log(\sigma_j^{(i)})^2 - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2) \quad (11)$$

The SLE-AdaVAE model implements this solution strategy with $\alpha = 1$ and $\beta = 5$.

In a *semi-supervised learning set-up*, we need to follow a different strategy to introduce a similar inductive bias on the distribution of the healthy system condition. Since a mix of healthy and faulty conditions might be present in the training data there is no benefit of forcing $\beta > 1$. Hence, we incorporate an explicit penalty affecting only the labeled class and we optimise the following loss

$$\mathcal{L}_{\text{KIL-AdaVAE}} = \mathcal{L}_{\text{ELBO}} - \gamma D_{KL}(q_\phi(z|x)||p(z))_{S_T} \quad (12)$$

The additional loss-term to the ELBO forces the encoded representation of healthy data (i.e. S_T) to match the factorized unit Gaussian. The weight of this new divergence term relative to the standard D_{KL} in the ELBO is controlled by the hyperparameter γ . The KIL-AdaVAE model implements this solution strategy with $\alpha = 5$ and $\gamma = n$. However, the solution is fairly robust to a large range of values of γ . The entire procedure of the proposed knowledge induced learning with adaptive sampling variational autoencoder is provided in Algorithm 1.

Algorithm 1: Knowledge induced learning with adaptive sampling VAEs.

Input: $\{x^{(i)}\}_{i=1}^m \in \{S_T, D_U\}$ Labeled and unlabeled samples
 $\theta, \phi \leftarrow$ Initialize parameters VAE
 $X_L = \text{Upsample}(\{x^{(i)}\}_{i=1}^u)$ Upsample S_T to size m with replacement
while $i \leq E$ **do**
 $g \leftarrow \nabla_{\theta, \phi} \mathcal{L}_{\text{KIL-AdaVAE}}(x, x_L)$
 $\theta, \phi \leftarrow$ Update parameters using gradient g
end
Input: $\{x^{(i)}, h_s^{(i)}\}_{i=1}^u \in S_T$ Labeled samples and their labels **for** $i \in \{1, \dots, u\}$ **do**
 $\mu^{(i)} \sim q_\phi(z|x^{(i)})$
end
 $\xi \leftarrow \text{MaxMin}(\mu_{S_T})$
 $\mathcal{H} \leftarrow$ Initialize parameters One-Class Network
while $i \leq E_s$ **do**
 $g \leftarrow \nabla_{\mathcal{H}} \mathcal{L}_{\text{One-Class}}$
 $\mathcal{H} \leftarrow$ Update parameters using gradient g
end
Input: $\{x^{(j)}\}_{i=u+1}^m \in D_U$ Unlabeled samples
for $i \in \{1, \dots, m\}$ **do**
 $\mu^{(i)} \sim q_\phi(z|x^{(i)})$
 $\mu_N^{(i)} \leftarrow \text{MaxMin}(\mu^{(i)}, \xi)$
 $s_k(\mu^{(i)}; S_V) = \frac{|1 - \mathcal{G}(\mu^{(i)})|}{F_{99.99}(|(1 - \mathcal{G}(S_V))|)}$
 $\hat{h}_s(x^{(i)}) \leftarrow$ from Equation 3
end
Input: $\{x^{(j)}\}_{j=1}^M \in D_T$ Test samples
for $j \in \{1, \dots, M\}$ **do**
 $\mu^{(j)} \sim q_\phi(z|x^{(j)})$
 $\mu_N^{(j)} \leftarrow \text{MaxMin}(\mu^{(j)}, \xi)$
 $s_k(\mu^{(j)}; S_V) = \frac{|1 - \mathcal{G}(\mu^{(j)})|}{F_{99.99}(|(1 - \mathcal{G}(S_V))|)}$
 $\hat{h}_s(x^{(j)}) \leftarrow$ from Equation 3
end

4 Case Study

4.1 The AGTF30 dataset

A new dataset was designed to evaluate the proposed method. The AGTF30 dataset provides simulated CM data of an advanced gas turbine during flight. The dataset was synthetically generated with the AGTF30 (Advanced Geared Turbofan 30k lbf) dynamical model [21]. Real flight conditions as recorded on board of a commercial jet [29] were taken as input to the AGTF30 model. Figure 3 shows the corresponding flight envelope given by the traces of altitude (Alt), flight Mach number (MN) and power level angle (PLA). Two distinctive stable flying altitudes and a rapid transient maneuver for altitude adaptation were selected. The labeled dataset \mathcal{D}_L (blue) consists of multivariate steady-state responses (i.e. $x^{(i)} \in R^{14}$ of the AGTF30 model during 10.000s of flight at cruise with a healthy system condition (i.e. $h_s = 1$). The unlabeled and test datasets $\{\mathcal{D}_U, \mathcal{D}_T\}$ (orange) contain $C = 17$ concatenated time series of model responses resulting from faulty engine conditions. No additional noise was added to the model response since the input values are already noisy. Each fault corresponds to an individual component fault and has a duration of 200 s (see Table 1). The flight conditions during periods of times in which faults were induced and randomly assigned from three time operation intervals extracted from the cruise envelope (see Table 2). The unlabeled dataset also includes data from 500s of operation when the engine is healthy. The unlabeled and test datasets $\{\mathcal{D}_U, \mathcal{D}_T\}$ are, therefore, a set of $C + 1$ truncated system conditions. The sampling frequency of the simulation is 1Hz. It should be noted that the test set contains failure types not present at training time (e.g. $C=3, 4$ & 17).

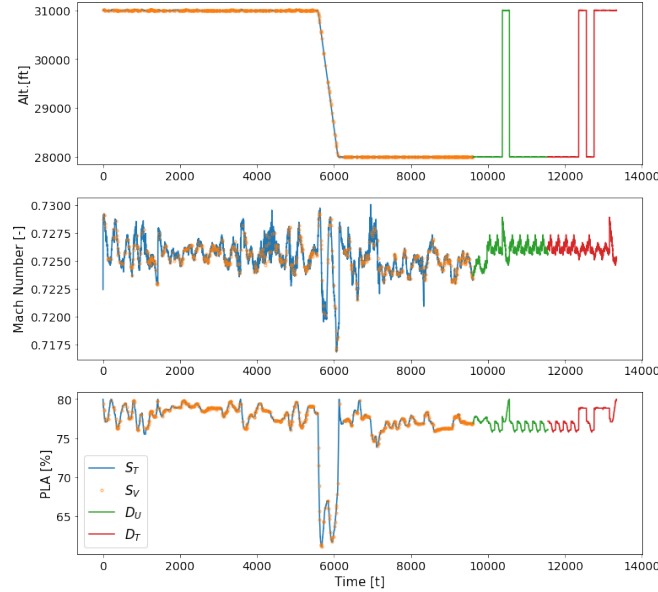


Figure 3: Flight envelope given by the traces of altitude, flight Mach number and power level angle (PLA). Four datasets are shown: S_T (blue), S_V (orange), D_U (green) and D_T (red).

4.2 Pre-processing

The input space X to the autoencoder models is normalized to a range $[-1, 1]$ by min/max-normalization. A validation set S_T comprising 6 % of the labeled data for all the models was chosen for setting the hyperparameters.

4.3 Network architectures

The learning strategies described in Section 2.2 require two overall network architectures. In contrast to the supervised option, the *semi-supervised* and *supervised with embedding* learning architectures require an autoencoder network in addition to the *one-class network*. As shown in Figure 4, the input signals x are reconstructed with the *encoder-decoder* networks; which provide a new representation z of the input signals. The mapping to the target label \mathbf{T} is carried out by the *one-class network* taking as input the transformation of the condition monitoring data to a latent space z .

To evaluate the different methods, we separate the effect of regularization in the form of model and learning strategies choice from other inductive bias in the form of choice of neural network architecture. Therefore, we define a common

Table 1: Overview of Generated Faults

C	Magnitude	Intensity	Flight Condition	Dataset
1	Fan Efficiency	1.0 %	S_1	\mathcal{D}_U
2	Fan Efficiency	1.5 %	S_1	\mathcal{D}_T
3	Fan Capacity	1.0 %	S_1	\mathcal{D}_T
4	LPC Efficiency	1.5 %	S_1	\mathcal{D}_T
5	LPC Capacity	1.5 %	S_1	\mathcal{D}_T
6	LPC Capacity	2.0 %	S_1	\mathcal{D}_U
7	HPC Efficiency	1.0 %	S_2	\mathcal{D}_U
8	HPC Efficiency	1.5 %	S_3	\mathcal{D}_T
9	HPC Efficiency	2.0 %	S_1	\mathcal{D}_U
10	HPC Capacity	1.0 %	S_1	\mathcal{D}_T
11	HPC Capacity	2.0 %	S_2	\mathcal{D}_U
12	HPT Efficiency	1.0 %	S_1	\mathcal{D}_U
13	HPT Efficiency	1.5 %	S_3	\mathcal{D}_T
14	HPT Efficiency	2.0 %	S_1	\mathcal{D}_U
15	HPT Capacity	1.5 %	S_3	\mathcal{D}_T
16	HPT Capacity	2.0 %	S_1	\mathcal{D}_U
17	LPT Capacity	1.0 %	S_2	\mathcal{D}_T

Table 2: Overview of Flight Conditions During Faults

Flight Condition	Alt [ft]	Match [-]	PLA[%]
S_1	28.0k	0.727-0.726	77.2-75.8
S_2	31.0k - 29.6k	0.729-0.722	80.0-66.9
S_3	31.0k - 30.0k	0.727-0.725	78.9-78.7

architecture of the *one-class network* for all our learning strategies. The proposed network architecture uses three fully connected layers. The first hidden layer has 20 neurons, the last hidden layer has 100 neurons. The network ends with a linear output neuron. Hence, in compact notation, we refer to the *one-class network* architecture as $[20, 100, 1]$. *tanh* activation function is used throughout the network. It should be noted that the one-class classification problem formulation is a regression problem and, therefore, the last activation is the identity. Based on the same argument all the autoencoder models use the same encoder architecture with one hidden layer of size 20 neurons and latent space of 8 neurons (i.e. $d = 8$). In compact notation, we refer to the *autoencoder network* architecture as $[n, 20, 8, 20, n]$. The same architecture is also taken for the *SLE-HELM* model.

The supervised learning strategy requires only one network with the architecture $[n, 20, 8, 20, 100, 1]$. Hence, in order to enable a clear comparison of the feature learning capabilities of supervised and semi-supervised models, the structure of the network from the last hidden layer is the same as the one-class in the SLE and SSL. Moreover, we unconventionally refer to the first part of the network from the input to the second last hidden layer as the *equivalent encoder network* and denote the second last hidden layer as the supervised latent space z_s .

4.4 Training Set-up

The optimization of the networks’ weights of all the models except HELM was carried out with mini-batch stochastic gradient descent (SGD) and with the *Adam* algorithm [30]. *Xavier* initializer [31] was used for the weight initializations. The learning rate was set according to Table 3. The batch size for the autonecoders configurations was set to 512 and to 16 for supervised model. Similarly, the number of epochs for autoencoder training was set to 800 and for the supervised models to 300. Therefore, all these methods use the same network architecture and hyperparameters of the optimiser.

4.5 Evaluation Metrics

The performance of the proposed method is evaluated and compared to alternative neural network models on the selected fault diagnostics task: detection of unknown faults (i.e. estimation of h_s), unsupervised segmentation of the fault types (i.e. estimation of \mathbf{V}). To further support the analysis, we also evaluate the capability of different models to find a useful transformation that is informative of the system condition (i.e. representation learning). For each of the three groups, we consider targeted evaluation metrics that are defined in following.

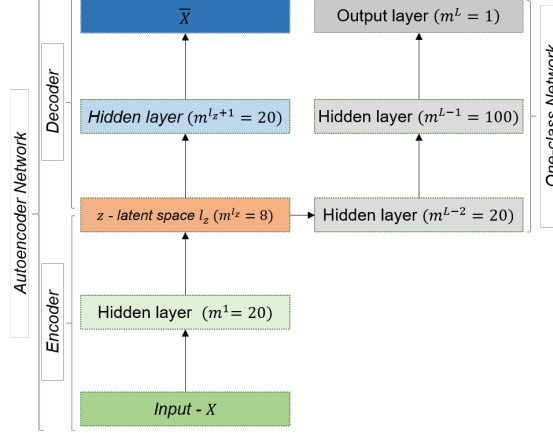


Figure 4: Network architecture for SSL and SLE strategies with an autoencoder (encoder-decoder) and the detection network. VAE models use only μ as the latent representation and therefore $\log(\sigma^2)$ is not considered.

Table 3: Default training parameters

Model	LR	Batch size	Epochs
SL	0.001	16	300
SLE, SSL & KIL	0.005/0.001	512/16	800/300

Metrics for Fault Detection Given the combined dataset $\{D_U, D_T\}$ with true health state h_s and the corresponding estimated health state \hat{h}_s , we evaluate the detection performance of the different neural network models as the accuracy of a binary classification problem. Three metrics for binary classification were considered: accuracy (Acc), true positive rate (TP), false positive rate (FN).

Metrics for Fault Segmentation Since the discovery and classification of fault types from unlabeled data is addressed as a clustering problem, we use the following four standard clustering metrics to evaluate the fault segmentation performance: number of clusters (C), homogeneity (h), completeness (c), and Adjusted Mutual Information (AMI). The corresponding definitions are provided in the appendix (Section 7.2). We select these metrics because of their strong mathematical foundation since they are rooted in information theory, and their ability to analyse non-linear similarities.

Metrics for Representation Learning To provide a deeper understanding of the different model capabilities and to explain their diagnostic performance, we define three auxiliary metrics:

Adjusted Mutual Information Gain (AMIG): The adjusted mutual information (AMI) evaluates the information that an alternative cluster representation of the data $\mathbf{U} = \{U_i | i = 1, \dots, R\}$ shares with the true segmentation of the system conditions $\mathbf{V} = \{V_j | j = 1, \dots, C\}$. AMI takes the value of 1 when the two partitions U and V are identical and 0 when the mutual information between the two partitions equals the value expected due to chance alone. A formal definition of AMI is given in Section 7.2.

In order to measure the gain of information about the true system condition provided by the encoded representation, we evaluate the delta in AMI between the latent space z and the input signals x . Hence, AMIG evaluates the capacity of the encoder to find a useful transformation that is informative of the system conditions and is defined as follows

$$AMIG(x, z, V) = AMI(z, V) - AMI(x, V) \quad (13)$$

Linear Separability Gain (LSG): This metric evaluates the theoretical capacity of the *encoder* transformation to obtain a latent space that is linearly separable in the true system states (i.e V). The linear separability of the latent space (i.e z and z_s) is measured as the accuracy of a linear logistic classifier (g) to predict each of the system states V . It, therefore, uses a supervised classification problem where the true labels (V) are known. The gain in linear separability is then defined as accuracy gain achieved by the *encoder* transformation. Hence, it evaluates if the latent representation is useful for fault segmentation. A standard logistic classifier with a linear kernel is considered. A one-vs-one strategy is assumed for reducing the original multi-class classification problem to multiple binary classification problems. The separability gain is defined as:

$$LSG(z, x; V) = \text{Acc}(g(z), V) - \text{Acc}(g(x), V) \quad (14)$$

Mean Mutual Information (MMI): The mutual information for continuous target variables (i.e. I) is a measure of dependency between two random variables. Hence, $I(x_i, z_j)$ measures how informative each component of input signal (i.e. $x_i \mid i = 1, \dots, n$) is for each of the components of the latent representation (i.e. $x_i \mid i = 1, \dots, n$). In order to have a simple aggregated measure of an overall dependency between x and z , we compute the mean value of mutual information map $\hat{I}(x, z)$ as follows:

$$\hat{I}(x, z) = \sum_{i=1}^n \sum_{j=1}^d I(x_i, z_j) \quad (15)$$

For autoencoder networks, we also compute the mean value of the mutual information map between the sample latent space (z) and each reconstructed signal i.e. $I(z_i, \bar{x}_j)$. Similarly, $I(z_i, x_j)$ is a measure of how informative each component of the latent space (z) is for the components of the reconstructed input signal \bar{x} .

The encoder networks in VAE models provide embedding representations in term of μ and σ . We compute the AMIG, LSG and $\hat{I}(x, z)$ using only μ (i.e. $z=\mu$) since the one class network takes exclusively μ as input.

5 Results

5.1 Fault Detection

Table 4 reports the performance of our ten models on fault detection for the three considered evaluation metrics. The different models show very disparate accuracy values. The proposed method, knowledge induced learning with adaptive sampling variational autoencodes (*KIL-AdaVAE*) is the only model that achieves a perfect accuracy. *KIL-AdaVAE* provides a nearly 1.4% absolute improvement with respect to the next best performing model *SLE-AdaVAE* and 34% absolute improvement with respect to worst performing model *SLE-HELM*. The supervised model *SL-FF* outperforms the embedding counterpart *SLE-AE* with nearly 17% absolute improvement. Within the embedding solutions, generative models based on VAE (i.e. *SLE-VAE*, *SLE- β -VAE* and *SLE-AdaVAE*) outperform the *SL-FF* discriminative model with more than a 7% margin. Hence, these results suggest an advantage of the generative modes for fault detection. Comparing SLE VAE models (i.e. *SLE-VAE* and *SLE-AdaVAE*) and SSL VAE models (i.e. *SSL-M1-VAE* and *SSL-M1-AdaVAE*), one can observe that the use of unlabeled data for training of the autoencoder network does not result in an accuracy improvement. Only when the unlabeled data is combined with the adaptive sampling (i.e. *SSL-M1-AdaVAE*) or with the knowledge induced learning *KIL-VAE*, accuracy results close to the SLE models can be reached. All the models are able to clearly label the healthy class within the datasets $\{D_U, D_T\}$. Hence, the false negative rate is zero.

Table 4: Overview of Fault Detection Results. Mean values of 10 runs

Method	Acc	TN	FN
SL-FF	85.5 \pm 1.6	84.0 \pm 1.8	0.0 \pm 0.0
SLE-AE	68.9 \pm 0.2	65.6 \pm 0.2	0.0 \pm 0.0
SLE-HELM	65.7 \pm 21.3	-	-
SLE-VAE	93.4 \pm 0.8	92.7 \pm 1.6	0.0 \pm 0.0
SLE- β -VAE	93.3 \pm 0.1	92.6 \pm 0.1	0.0 \pm 0.0
SLE-AdaVAE	98.6 \pm 0.1	98.5 \pm 0.1	0.0 \pm 0.0
SSL-M1-VAE	37.8 \pm 2.5	31.1 \pm 2.8	0.0 \pm 0.0
SSL-M1-AdaVAE	93.7 \pm 0.4	93.1 \pm 0.4	0.0 \pm 0.0
KIL-VAE	90.1 \pm 1.3	89.1 \pm 1.5	0.0 \pm 0.0
KIL-AdaVAE	100.0 \pm 0.0	100.0 \pm 0.0	0.0 \pm 0.0

The detection performance of the one-class solutions reported in Table 4 are determined by the capability of the similarity score $s_I(x^{(i)}; \xi)$ to represent a valid and consistent distance to healthy system conditions (i.e. D_L). To demonstrate and verify this behaviour we plot in Figure 5 the similarity score for two HPT efficiency and two HPC capacity faults of different intensities (1% and 2%) at the flight conditions S_1 . We see that for both HPT faults the similarity score is well above the decision threshold $s_I(x^{(i)}; \xi) > 1$ (black horizontal line) and, therefore, the one-class network can clearly discriminate between healthy and faulty system conditions. We also observe that the more severe the fault the higher the detection index; and therefore $s_I(x^{(i)}; \xi)$ also shows the expected consistency. The same behaviour is observed for HPC faults. However, two different scales of the similarity score present for HPT and LPC faults indicate that LPC faults are more difficult to detect since they are closed to the healthy system conditions. In fact only HPC capacity faults with intensity of 2% are clearly detected.

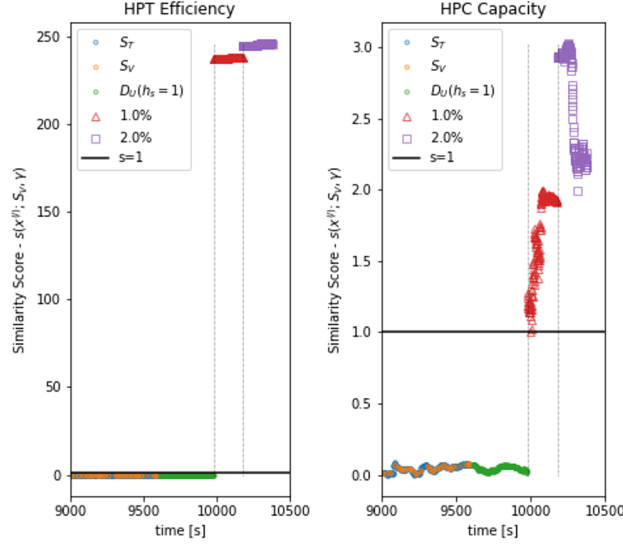


Figure 5: Similarity index for two HPT efficiency and two LPC capacity faults of different intensities with *SSL-VAE* model (1.0% severity is shown with a red triangle and 2.0 % with a purple squares). All the faults occur at flight conditions S_1 . The decision threshold is plotted as horizontal black line ($s = 1$). The onset times of each fault are indicated by the vertical dashed lines. Last samples of the training S_T and validation set S_V (with $t \geq 9000$) are shown in blue and orange

5.2 Fault Segmentation

Table 5 shows the clustering performance of the OPTICS algorithm applied to the input space (x) and to the latent space (z) of our ten models on the combined dataset $\{D_U, D_T\}$. 14 informative clusters are directly identified in the input space. The existing 18 system states (i.e. 17 fault types and the healthy one) are discovered in the latent representation of the proposed model *KIL-AdaVAE*. Within the supervised with embedding solutions, the segmentation of the latent space of all the generative models result in 18 clusters. In contrast, for the discriminative counterpart, *SLE-AE*, only 14 clusters are discovered. Hence, these results also show an advantage of the generative modes for fault segmentation. SSL strategies lead to a reduction on the number of clusters, showing that the use of unlabeled data at training time results in latent space that is not clearly representative of the true system states. Only when the unlabeled data is combined with a knowledge induced learning strategy (i.e KIL models (i.e KIL-VAE and KIL-AdaVAE)), the existing 18 clusters can be distinguished. Hence, only through knowledge induction an accurate fault segmentation can be obtained.

Since the adjusted mutual information (AMI) evaluates how closely the segmentation proposed by the clustering algorithm replicates the true classes, the AMI shows correlation with number of clusters. Hence, with the proposed approach *KIL-AdaVAE*, the cluster assignments obtained by the OPTICS algorithm in the latent space $\mathbf{U} = \{U_j | j = 1, \dots, R\}$ offer the closest segmentation to the true class $\mathbf{V} = \{V_j | j = 1, \dots, C\}$. Furthermore, the proposed clusters result in a nearly perfect *homogeneity (h)* indicating that its clusters contain only data points which are members of a single class and *completeness (c)* since the data points that are members of a given class U_j are elements of the same cluster.

The OPTICS clustering algorithm has two hyperparameters (*min_cluster* size and *min_points*) that have a moderate influence on the results. The minimum cluster size can be easily justified from the knowledge of a relevant time scale of the diagnostics. For instance, if we have an interest in faults conditions that persist for more that 90 seconds then $min_cluster \geq 90$. The minimum *min_points* can be derived from dimensions in the dataset (n) following the general rule $min_points \geq n + 1$. In our case, we used $min_cluster = 90$ and $min_points=15$. However, the number of clusters is fairly robust to the values of these two hyperparameters.

Fault Segmentation Visualization. In order to provide a visualization of the cluster assignments obtained with the OPTICS clustering algorithm we perform a 2-dimensional t-Distributed Stochastic Neighbor Embedding (t-SNE) of the latent space z in the dataset $\{D_U, D_T\}$. Figure 6 shows the resulting normalized 2-D t-SNE representation. The color code corresponds to the cluster number assigned by the OPTICS algorithm (i.e. U_i). t-SNE is able to find a lower dimensional representation that maintains local properties of the high dimensional space. Hence, features that are close

Table 5: Overview Unsupervised Clustering Results with OPTICS.

Method	R	AMI	h	c
Input (x)	14	0.77	0.71	0.84
SL-FF	17 ± 1	0.90	0.89	0.91
SLE-AE	14	0.75	0.69	0.82
SLE-HELM	3	-	-	-
SLE-VAE	18	0.91	0.91	0.90
SLE- β -VAE	18	0.88	0.89	0.88
SLE-AdaVAE	18	0.90	0.91	0.90
SSL-M1-VAE	16 ± 1	0.78	0.73	0.83
SSL-M1-AdaVAE	17	0.88	0.87	0.90
KIL-VAE	18	0.92	0.93	0.92
KIL-AdaVAE	18	0.92	0.93	0.92

in the latent space are also close in the t-SNE representation. Therefore, since 18 clusters are identified in the eight dimensional latent space, nearly all of them are also clearly clustered in the 2-D t-SNE space. The cluster representation of the t-SNE is influenced by the perplexity hyperparameter. We used a perplexity value of 200 (i.e. similar to the size of the fault types) in order to show a very compact representation of the different system states.

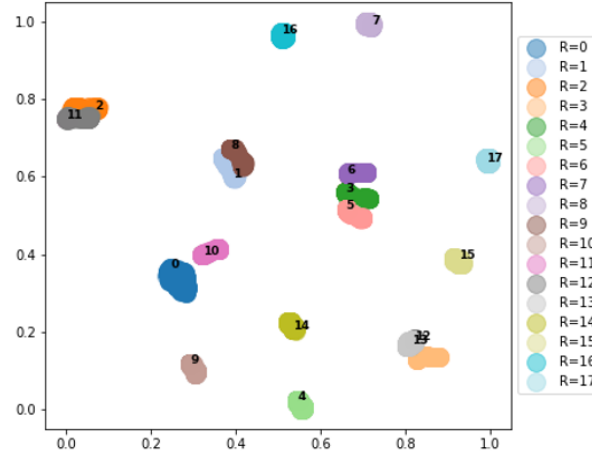


Figure 6: KIL-AdaVAE 2-D t-SNE embedding of the 8-D latent space (z). Color code according to the cluster labels assigned by the OPTICS algorithm (i.e $j = 0, \dots, R$ with $R = 18$). The healthy class is represented with label $R = 0$ (dark blue). The clustering algorithm identifies 18 clusters, therefore, each fault type and the healthy system condition is represented with a different color. Although some of the clusters overlap in the 2-D representation they are separable in the 8-D latent space.

5.3 Representation Learning

The results presented in the previous sections demonstrated that the proposed knowledge induced learning method leads to excellent performance for fault detection and segmentation. Aiming to provide a deeper understanding on how the different latent representations affect the performance on down-stream diagnostics tasks, in this section, the latent representation of each neural network model is analyzed.

First, we evaluate how informative the encoded representation is of the true operation partitions $\mathbf{V} = \{V_j | j = 1, \dots, C\}$ relative to the input space. Figure 7 reports the gain of adjusted mutual information that is obtained with the cluster representation $\mathbf{U} = \{U_i | i = 1, \dots, R\}$ on the latent space (z) with respect to an alternative cluster representation $\mathbf{U}' = \{U'_i | i = 1, \dots, R'\}$ on the input space (x). The adjusted mutual information of the input space is 0.77 and, therefore, the horizontal black line shows the value of AMIG to achieve identical partitions (i.e. AMIG = 23%). The proposed method KIL-AdaVAE provides the maximum increase in mutual information gain within the ten implemented models. Hence, it provides the most informative latent space for fault segmentation.

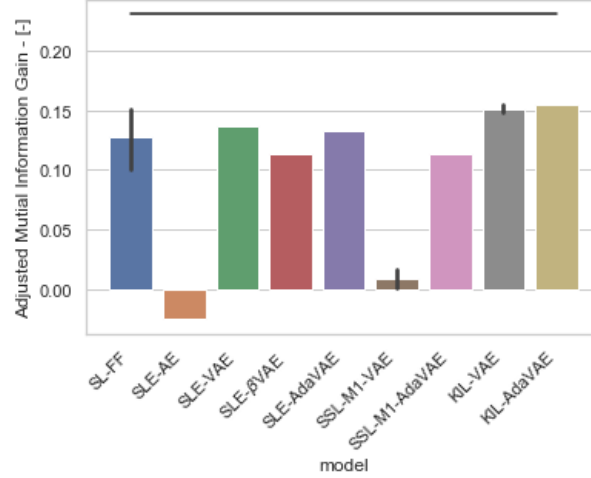


Figure 7: Adjusted mutual information gain AMIG each model. The adjusted mutual information of the input space is 0.77. The horizontal black line shows the value of AMIG to achieve identical partitions to the true system states.

Extending the representation analysis, we also evaluate the theoretical capacity of the *encoder* transformation to obtain a representation of the input space that is linearly separable in the true system states (i.e V). Table 8 shows the gain of linear separability that the latent space provides relative to the input space for each model. A logistic classifier with a one class versus the rest (one-vs-rest) approach results in a 89% accuracy and, therefore, the horizontal black line shows the value of LSG to achieve perfect accuracy (i.e. $LSG = 11\%$). The embedding of the proposed approach results in a 10% gain. Hence, a linear decision boundary can nearly perfectly separate each class versus the rest.

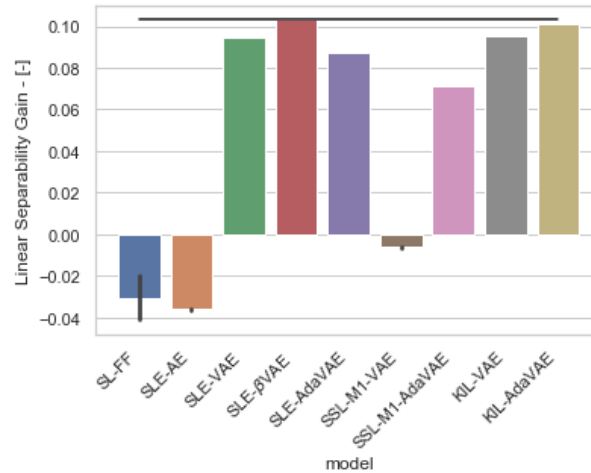


Figure 8: Linear separability gain LSG of the latent space z of each model relative to the input space x . A linear logistic classifier reaches an accuracy of 89% predicting each of the system states V

Information Preservation. The previous analysis shows that the embedding representation of the input signals is very informative of the true class and is linearly separable. We focus now on the analysis of the information preservation through the bottleneck. Figure 9 shows the mutual information map between the component of the input signals ($x_i ; i = 1, \dots, n$) and each of the dimension of the latent spaces $\mu_j ; j = 1, \dots, d$ ($I(x_i, \mu_j)$) for the proposed model *KIL-AdaVAE*. Low mutual information is observed in the latent space for input signals with indexes $[0, 1, 2, 4]$ (i.e. Altitude, flight Mach number, power level angle (PLA) and inlet temperature). In contrast, the latent space shows a clear dependency with the 9 remaining input components. The low mutual information components are the four scenario descriptor or independent variables of the engine process. Hence, the latent space correctly identifies that the

information in signals [0, 1, 2, 4] is redundant as is sensed in other engine measurements. This pattern is characteristics of all the autoencoder networks.

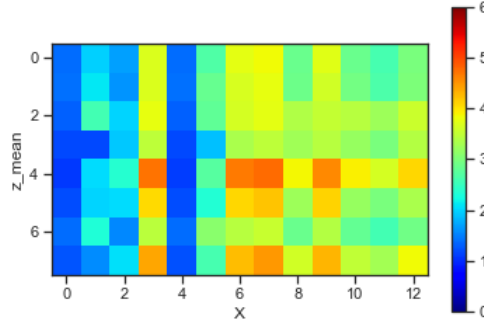


Figure 9: Mutual information map between the input and latent spaces with SSL-AdaVAE model.

Table 6 reports the mean value of the mutual information map between the input space and its embedding representation $\hat{I}(x, z)$. For autoencoder configurations, it also provides the mean value of the mutual information map between the *sample* latent space (z) and each of the reconstructed signals $\hat{I}(z, \bar{x})$. Autoencoder models show similar values of $\hat{I}(x, z)$, which are clearly higher than for the SL-FF model. The latent space of the supervised model is taken at the second hidden layer of the network and at this depth, the information about the input is quickly reduced. Traditional VAE models show nearly zero mutual information between the *sample* latent space (z) and each of the reconstructed signals (\bar{x}) since the sampled representation (z) is clearly dominated by the random Gaussian noise (ϵ). In contrast, the proposed sampling method (i.e. SLE-AdaVAE, SSL-M1-AdaVAE and KIL-AdaVAE) increases the amount of information between the sampled representation and the reconstructed signals. This also corresponds to a less noisy reconstruction signal. The proposed sampling method balances reconstruction and regularization of the ELBO and therefore the values of information on the sampled representation are 40% smaller than traditional autoencoders (SLE-AE and SLE-HELM).

Table 6: Mean Mutual Information between input and latent space $\hat{I}(x, z)$ and between latent space and reconstructed input $\hat{I}(z, \hat{x})$

Method	$\hat{I}(x, z)$	$\hat{I}(z, \bar{x})$
SL-FF	0.49	-
SLE-AE	2.90	3.56
SLE-HELM	3.52	4.51
SLE-VAE	3.04	0.09
SLE- β -VAE	3.00	0.06
SLE-AdaVAE	3.13	1.31
SSL-M1-VAE	3.10	0.10
SSL-M1-AdaVAE	3.06	1.51
KIL-VAE	2.55	0.03
KIL-AdaVAE	2.88	0.69

6 Conclusions

We introduced KIL-AdaVAE, a knowledge induced learning method with adaptive sampling variational autoencoder for open set diagnosis of complex systems.

The performance of the proposed method was evaluated on a synthetic dataset generated with the Advanced Geared Turbofan 30,000 (AGTF30) engine model. The AGTF30 dataset provides simulated condition monitoring data of an advanced gas turbine during real flight conditions under healthy and 17 faulty system conditions. KIL-AdaVAE significantly outperforms other methods in supervised and semi-supervised settings providing perfect detection accuracy and fault segmentation. Through an extensive feature representation analysis we demonstrated that its excellent performance is rooted in the fact that the resulting latent representation of the sensor readings i.e. z is not only very informative about the true system conditions but is also linearly separable with respect to them (healthy system

conditions and all observed fault types). More importantly we show KIL-AdaVAE reaches this discriminative latent representation with only access to the healthy system conditions.

Although the current problem formulation leads to an excellent result, from a theoretical perspective, the proposed adaptive sampling implies a deviation from the Gaussian prior generally assumed in the distribution of the variational approximate posterior that has not been considered in the KL term to keep an analytical close expression. A reformulation of the KL term to fit with the VAE framework remains, therefore, as a future line of research. Although we have proposed a generative method, we have not explored the potential benefits of using synthetically generated data to improve the diagnostics performance. The evaluation of the generative potential of the proposed methodology remains an interesting line of future research. Finally, from the prognostics and health management perspective, the verification of the real potential of the proposed solution in a setting with larger operating conditions variability is subject of further research.

Acknowledgment

This research was funded by the Swiss National Science Foundation (SNSF) Grant no. PP00P2_176878.

References

References

- [1] Zuyu Yin and Jian Hou. Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes. *Neurocomputing*, 174:643–650, jan 2016.
- [2] Luyang Jing, Ming Zhao, Pin Li, and Xiaoqiang Xu. A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox. *Measurement*, 111:1–10, dec 2017.
- [3] Jinhao Lei, Chao Liu, and Dongxiang Jiang. Fault diagnosis of wind turbine based on Long Short-term memory networks. *Renewable Energy*, 133:422–432, apr 2019.
- [4] Andre S Yoon, Taehoon Lee, Yongsub Lim, Deokwoo Jung, Philgyun Kang, Dongwon Kim, Keuntae Park, and Yongjin Choi. Semi-supervised Learning with Deep Generative Models for Asset Failure Prediction. In *KDD17 Workshop on Machine Learning for Prognostics and Health Management*, page 9, Halifax, 2017.
- [5] Ye Zhao, Roy Ball, Jerry Mosesian, Jean-Francois de Palma, and Brad Lehman. Graph-Based Semi-supervised Learning for Fault Detection and Classification in Solar Photovoltaic Arrays. *IEEE Transactions on Power Electronics*, 30(5):2848–2858, may 2015.
- [6] Chao Hu, Byeng D. Youn, and Taejin Kim. Semi-Supervised Learning With Co-Training for Data-Driven Prognostics. In *Volume 2: 31st Computers and Information in Engineering Conference, Parts A and B*, pages 1297–1306. ASME, jan 2011.
- [7] Yang Hu, Piero Baraldi, Francesco Di Maio, and Enrico Zio. A Systematic Semi-Supervised Self-adaptable Fault Diagnostics approach in an evolving environment. *Mechanical Systems and Signal Processing*, 88:413–427, may 2017.
- [8] Li Jiang, Jianping Xuan, and Tielin Shi. Feature extraction based on semi-supervised kernel Marginal Fisher analysis and its application in bearing fault diagnosis. *Mechanical Systems and Signal Processing*, 41(1-2):113–126, dec 2013.
- [9] K.P. Detroja, R.D. Gudi, and S.C. Patwardhan. A possibilistic clustering approach to novel fault detection and isolation. *Journal of Process Control*, 16(10):1055–1073, dec 2006.
- [10] Di Hu, Ali Sarosh, and Yun-Feng Dong. A novel KFCM based fault diagnosis method for unknown faults in satellite reaction wheels. *ISA Transactions*, 51(2):309–316, mar 2012.
- [11] Chaoshun Li and Jianzhong Zhou. Semi-supervised weighted kernel clustering based on gravitational search for fault diagnosis. *ISA Transactions*, 53(5):1534–1543, sep 2014.
- [12] Zhimin Du, Bo Fan, Xinqiao Jin, and Jinlei Chi. Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis. *Building and Environment*, 73:1–11, mar 2014.
- [13] Bruno Sielly Jales Costa, Plamen Parvanov Angelov, and Luiz Affonso Guedes. Fully unsupervised fault detection and identification based on recursive density estimation and self-evolving cloud-based classifier. *Neurocomputing*, 150:289–303, feb 2015.

- [14] Kedong Zhu, Fei Mei, and Jianyong Zheng. Adaptive fault diagnosis of HVCBs based on P-SVDD and P-KFCM. *Neurocomputing*, 240:127–136, may 2017.
- [15] Yanfei Lu, Rui Xie, and Steven Y. Liang. Adaptive online dictionary learning for bearing fault diagnosis. *The International Journal of Advanced Manufacturing Technology*, 101(1-4):195–202, mar 2019.
- [16] Leo H. Chiang, Benben Jiang, Xiaoxiang Zhu, Dexian Huang, and Richard D. Braatz. Diagnosis of multiple and unknown faults using the causal map and multivariate statistics. *Journal of Process Control*, 28:27–39, apr 2015.
- [17] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.
- [18] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *ICLR*, 2017.
- [19] Ali Razavi, Oriol Vinyals, Aäron Van Den Oord, and Ben Poole. Preventing posterior collapse with δ -VAES. In *7th International Conference on Learning Representations, ICLR 2019*, jan 2019.
- [20] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boulton. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1757–1772, 2012.
- [21] Jeffryes W. Chapman and Jonathan S. Litt. Control Design for an Advanced Geared Turbofan Engine. In *53rd AIAA/SAE/ASEE Joint Propulsion Conference*, Reston, Virginia, jul 2017. American Institute of Aeronautics and Astronautics.
- [22] Gabriel Michau, Thomas Palmé, and Olga Fink. Deep Feature Learning Network for Fault Detection and Isolation. *Conference of the PHM Society*, 8(012):1–11, 2017.
- [23] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010.
- [24] Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, volume 4, pages 3581–3589, jun 2014.
- [25] X Ester, M., Kriegel, H. P., Sander, J., & Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Kdd*, 96(34):226–231, 1996.
- [26] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [27] Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Balancing Learning and Inference in Variational Autoencoders. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:5885–5892, jul 2019.
- [28] Carl Doersch. Tutorial on variational autoencoders. *ArXiv*, abs/1606.05908, 2016.
- [29] DASHlink - Flight Data For Tail 687, 2012.
- [30] Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.
- [31] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterton, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.
- [32] Marina Meilă. Comparing clusterings by the variation of information. In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Learning Theory and Kernel Machines*, pages 173–187, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
- [33] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420, Prague, Czech Republic, June 2007. Association for Computational Linguistics.
- [34] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, pages 1–8, New York, New York, USA, 2009. ACM Press.
- [35] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *Journal of Machine Learning Research*, 11:2837–2854, 2010.
- [36] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical Review E*, 69(6):066138, jun 2004.

7 Appendix

7.1 β -VAE Experiments - Avoiding the posterior collapse

Table 7 and Table 8 report the impact of increasing the values β for the models SLE- β -VAE and SLE-AdaVAE in detection accuracy (Acc), mean mutual information $\hat{I}(x, \mu)$, mean mutual information $\hat{I}(z, \bar{x})$, reconstruction loss $\|x - \bar{x}\|_2$ and optimisation loss $\mathcal{L}_{\beta\text{-VAE}}$. As the β increases detection accuracy of SLE- β -VAE model rapidly degrades. As discussed in Section 3 we observe a decrease in the information preserved by the latent representation. This information loss is manifested in the increase of the reconstruction loss ℓ_2 -loss. In contrast, the SLE-AdaVAE is able to maintain very low construction error while keeping the D_{KL} term and hence the total loss small. Therefore, the SLE-AdaVAE model avoids the posterior collapse at high β and as a result is able to maintain a high detection accuracy.

Table 7: Effect of β in β -VAE model

β	Acc	$\hat{I}(x, \mu)$	$\hat{I}(z, \bar{x})$	ℓ_2 -loss	$\mathcal{L}_{\beta\text{-VAE}}$
1	93.4 \pm 1.0	3.05	0.09	0.028	1.42
5	93.2 \pm 0.7	3.00	0.06	0.240	3.73
9	69.3 \pm 3.2	2.83	0.04	0.292	3.80
13	68.6 \pm 6.7	2.77	0.03	0.292	3.80

Table 8: Effect of β in AdaVAE model

β	Acc	$\hat{I}(x, \mu)$	$\hat{I}(z, \bar{x})$	ℓ_2 -loss	$\mathcal{L}_{\beta\text{-VAE}}$
1	93.7 \pm 1.8	3.05	1.34	0.0006	0.022
5	98.6 \pm 0.1	3.13	1.32	0.0008	0.032
9	92.5 \pm 0.9	3.15	1.28	0.0011	0.040
13	96.0 \pm 1.3	3.04	1.19	0.0013	0.046

7.2 Clustering Metrics

In this subsection we define the three clustering metrics considered in section 5.2: adjusted mutual information (AMI), completeness (c) and homogeneity (h). For clarity we introduce first the fundamental concepts from information theory that they build upon. The formulation here presented is a compilation of previous work in [32], [33], [34] and [35].

Mutual Information (MI). The MI is a symmetric measure of dependency between two random variables. Concretely the mutual information can be understood as the decrease of uncertainty that knowing one variable results for the other [34]. MI, therefore, quantifies the information that two random variables share.

The mutual information between two random variables X and Y i.e $\text{MI}(Y, X)$ with discrete-valued join probability distribution $P(Y, X)$ and marginals $P(X)$ and $P(Y)$ over the respective outcome spaces \mathcal{X} and \mathcal{Y} is defined as:

$$\text{MI}(Y, X) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P(y, x) \log \frac{P(y, x)}{P(x)P(y)} \quad (16)$$

Under a valid statistical formulation of the clustering problem, MI can be used to measure the information shared between two clusterings and hence assess its similarity [34]. We then define valid statistical formulation of the clustering problem in the following.

Let's denote as \mathbf{U} and \mathbf{V} two partitions of a dataset $Z = \{z^{(1)}, z^{(2)}, \dots, z^{(M)}\}$ comprising M data point. We refer to $\mathbf{V} = \{V_j | j = 1, \dots, C\}$ as the partition according to the C true classes and to $\mathbf{U} = \{U_i | i = 1, \dots, R\}$ as an alternative partition with R clusters. We consider the case where these partitions are pairwise disjoint (i.e $\cap_i^R U_i = \cap_j^C V_j = \emptyset$) and complete ($\cup_{i=1}^R U_i = \cup_{j=1}^C V_j = Z$). Under this formulation the probability that a data point randomly picked from Z falls into the cluster U_i is given by

$$P(i) = \frac{|U_i|}{M} \quad (17)$$

Thus, $P(i)$ is a discrete random variable taking R values that is uniquely related to \mathbf{U} [32]. Moreover, the uncertainty associated with cluster \mathbf{U} (i.e. with its probability density function) is measured by the entropy $H(\mathbf{U})$. That is,

$$H(\mathbf{U}) = - \sum_{i=1}^R P(i) \log(P(i)) \quad (18)$$

Similarly, $P'(j) = \frac{|V_j|}{M}$ denotes the probability of outcome being in cluster V_j and the uncertainty associated with cluster V is

$$H(\mathbf{V}) = - \sum_{i=1}^C P'(i) \log(P'(i)) \quad (19)$$

The join probability of a data point belonging to both the cluster U_i in \mathbf{U} and cluster V_j in \mathbf{V} is

$$P(i, j) = \frac{|U_i \cap V_j|}{M} \quad (20)$$

Therefore, in this statistical clustering context, we can use equation 16 to evaluate the information that the clusters \mathbf{U} and \mathbf{V} share:

$$\text{MI}(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^R \sum_{j=1}^C P(i, j) \log \frac{P(i, j)}{P(i)P'(j)} \quad (21)$$

Hence, MI measures the decrease of uncertainty that knowing U reduces the uncertainty about the other V .

Based on this definition we see that MI is a non-negative and it is upper bounded by the entropies $H(\mathbf{U})$ and $H(\mathbf{V})$.

$$\text{MI}(\mathbf{U}, \mathbf{V}) \leq (H(\mathbf{U}), H(\mathbf{V})) \quad (22)$$

However, MI is not upper bounded by a fix value and is affected by chance. Therefore, it is generally preferred to used the *adjusted mutual information* where the measure is corrected for randomness.

Adjusted Mutual Information (AMI). The adjusted measure for mutual information is defined as:

$$\text{AMI}(\mathbf{U}, \mathbf{V}) = \frac{\text{MI}(U, V) - \mathbb{E}[\text{MI}(\mathbf{U}, \mathbf{V})]}{\max\{H(\mathbf{U}), H(\mathbf{V})\} - \mathbb{E}[\text{MI}(\mathbf{U}, \mathbf{V})]} \quad (23)$$

AMI takes the value of 1 when two partitions are identical and 0 when the MI between two partitions equals the value expected due to chance alone.

We can also evaluate how closely the partition U replicates the true classes V . Then, we compute the contingency table to see the cluster over cluster overlap between U and V . The information on cluster overlap between U and V is summarized by the entries of the contingency table (i.e $n_{i,j} = |U_i \cap V_j|$) that denote number of data points common to clusters U_i and V_j .

Homogeneity (h). A clustering partition U satisfies homogeneity if all of its clusters contain only data points which are members of a single class [33].

$$h(\mathbf{U}, \mathbf{V}) = \begin{cases} 1 & H(\mathbf{V}, \mathbf{U}) = 0 \\ 1 - \frac{H(\mathbf{V}|\mathbf{U})}{H(\mathbf{U})} & \text{otherwise} \end{cases} \quad (24)$$

where

$$H(\mathbf{V}|\mathbf{U}) = - \sum_{i=1}^R \sum_{j=1}^C \frac{n_{i,j}}{N} \log\left(\frac{n_{i,j}}{\sum_{j=1}^C n_{i,j}}\right) \quad (25)$$

Completeness (c). A clustering partition U satisfies completeness (i.e $c = 1$) if all the data points that are members of a given class are elements of the same cluster [33].

$$c(\mathbf{U}, \mathbf{V}) = \begin{cases} 1 & H(\mathbf{U}, \mathbf{V}) = 0 \\ 1 - \frac{H(\mathbf{U}|\mathbf{V})}{H(\mathbf{V})} & \text{otherwise} \end{cases} \quad (26)$$

$$H(\mathbf{U}|\mathbf{V}) = - \sum_{i=1}^C \sum_{j=1}^R \frac{n_{i,j}}{N} \log\left(\frac{n_{i,j}}{\sum_{j=1}^R n_{i,j}}\right) \quad (27)$$

The mutual information of the continuous target variables is estimated with a non-parametric methods based on entropy estimation from k-nearest neighbors distances following the method in [36]

$$I(Y, X) = \int_X \int_Y p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (28)$$

where $p(x, y)$ is the joint probability density function of X and Y , and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y respectively.