Anomaly Detection with Variational Autoencoders via Reconstruction Error of the Expected Latent Representation

Anonymous Author(s)
Affiliation
Address
email

Abstract

A common approach to anomaly detection is to model normality and adopt the difference from normality as an anomaly measure. One approach to modeling normality is to introduce latent variables that are inferred from observed variables. Variational autoencoders are one such state of the art approach for incorporating latent variables. In this paper, we leverage the stochastic nature of the latent variables learned by variational autoencoders, as each point in the latent space is sampled from probability distributions parameterized during the learning process. We define the expected latent representation and the reconstruction error of the expected latent representation, which we adopt to improve anomaly detection via variational autoencoders. Results from evaluations on benchmark datasets produce superior results to single sample approximations of the expected reconstruction error, while producing competitive results to comparable anomaly detection techniques.

1 Introduction

Anomalies can be thought of as unusual or unexpected behavior in a system. Anomaly detection is important, as failure to properly identify anomalies can be costly, resulting in a loss of trust, revenue, or life. However, the nature of anomalies are that they are unusual, and typically rare, so there may be many examples of normal behavior but few examples of anomalies.

A common task in anomaly detection is to identify anomalous behavior by its contrast to predefined normal behavior [3]. In this scenario a set of normal examples are used as a training set, and a set of normal and anomalous examples used as a test set. The goal is to learn a model of normality using the training set, expecting to identify anomalous examples in the test set via differences from the normal model, and by extension anomalies in new data. The degree of difference from normality is used as an anomaly measure.

Reconstruction models attempt to reproduce each input at their output, subject to some internal representation constraint that prevents simple duplication. If properly trained, a reconstruction model should reproduce normal data accurately, but struggle to reproduce anomalous data. The error in reconstructing the input can be used as an anomaly measure.

Variational autoencoders (VAEs) are one such approach that incorporates reconstruction error as part of its training objective. Variational autoencoders are stochastic by design and this stochastic process is present during the training, testing, and sampling. Although critical to training and sampling, a stochastic element in testing hinders anomaly detection as it introduces variability in the reconstruction error. Computing the reconstruction error of an example relies on sampling from the approximate posterior for that example, before feeding the sample to the generative model to

Submitted to NeurIPS 2021 Workshop on DGMs and Downstream Applications. Do not distribute.

approximate the expected reconstruction error. Generally, one sample is taken for each example to reduce computational requirements. This sampling process is at odds with the anomaly detection task as it creates blurry models of normality, which can in turn make it difficult to pick out anomalies.

The contribution of this paper is to define the expected latent representation and then the reconstruction error of the expected latent representation as a measure that allows improved anomaly detection in variational autoencoders. We do this by taking advantage of the parameters of the approximate posteriors that define the latent representation of each data example, given a fully trained variational autoencoder. Our approach removes variation in the latent representation that is introduced from the sampling process, helps distinguish between latent representations of different examples with overlapping approximate posteriors, and provides a clear view of what we expect to see in the latent space for a given example. The result is an improvement in anomaly detection via variational autoencoders in comparison to single sample approximations of the expected reconstruction error, while producing competitive results to comparable anomaly detection techniques when evaluated on benchmark datasets.

2 Background

Latent variable models are probabilistic models that attempt to explain observed variables in terms of latent variables. *Latent variables* are hidden variables that are not directly observed but instead inferred from observed variables. Latent variable models make the assumption that given an input $\mathbf{x}$ there is an underlying latent variable $\mathbf{z}$ that can help explain $\mathbf{x}$ or reveal something useful about $\mathbf{x}$ [9]. Applications of latent variable models include dimensionality reduction, clustering, density estimation, and sample generation [9].

Two advantages of latent variable models are:

1. Some phenomena cannot be naturally observed; latent variables are useful for capturing this hidden information.
2. Given prior knowledge about the data, we can leverage it by incorporating it in the model as latent variables or constraints on latent variables.

It is typical to use the joint distribution of the observed latent variable $\mathbf{x}$ and a latent variable $\mathbf{z}$ to define the marginal likelihood of $\mathbf{x}$. Formally, given $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^n$, the marginal distribution over the observed variables is:

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z} \tag{1}$$

$$= \int p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z} \tag{2}$$

where $p_\theta(\mathbf{x})$ is the marginal likelihood of $\mathbf{x}$, $p_\theta(\mathbf{x}, \mathbf{z})$ is the joint distribution of $\mathbf{x}$ and $\mathbf{z}$, and $p_\theta(\mathbf{z})$ is the prior distribution of the latent variable $\mathbf{z}$. However, the solution to this equation is generally intractable and an approximate solution is required.

Variational Autoencoders are a stochastic variational inference and learning algorithm that performs approximate inference in latent variable models where the marginal likelihood and the true posterior density are intractable [7, 8].

To make the intractable problem tractable, variational autoencoders introduce a parametric inference model $q_\phi(\mathbf{z}|\mathbf{x})$, where ϕ are the variational parameters of the inference model [8, 9]. The variational parameters are optimized such that:

$$q_\phi(\mathbf{z}|\mathbf{x}) \approx p_\theta(\mathbf{z}|\mathbf{x}) \tag{3}$$

where $q_\phi(\mathbf{z}|\mathbf{x})$ is the approximate posterior and $p_\theta(\mathbf{z}|\mathbf{x})$ is the true posterior.

The log marginal likelihood of $\mathbf{x}^{(i)}$ can then be written as:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) = D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{(i)})\right) + \mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) \tag{4}$$

where the first term is the Kullback-Leibler divergence (D_{KL}) of the approximate posterior and the true posterior, and the second term is the evidence lower bound of the log marginal likelihood.

Although we cannot solve this equation it its entirety, variational autoencoders estimate the lower bound of the log marginal likelihood $\mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right)$, otherwise known as the variational lower bound or the evidence lower bound (ELBO).

The ELBO can be written as:

$$\mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) = -D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right) + \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})\right] \tag{5}$$

where the first term is the negative D_{KL} of the approximate posterior $q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})$ and the prior $p_{\boldsymbol{\theta}}(\mathbf{z})$. The second term is the expected value of the log probability density of $\mathbf{x}^{(i)}$ under the generative model. This is interpreted as the expected negative reconstruction error from the perspective of autoencoder techniques. The first term can be integrated analytically but the second term requires estimation by sampling:

$$\widetilde{\mathcal{L}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) = -D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right) + \frac{1}{L}\sum_{l=1}^{l}\left(\log p_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)}\right)\right) \tag{6}$$

where $\mathbf{z}^{(i,l)}$ is a sample, and L is the number of samples. In practice only one sample is taken to approximate the expected negative reconstruction error to reduce the computational complexity. The ELBO is formulated such that:

$$\log p_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}\right) \geq \mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) \simeq \widetilde{\mathcal{L}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) \tag{7}$$

where $\log p_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}\right)$ is the log marginal likelihood of input $\mathbf{x}^{(i)}$, ϕ are the variational parameters, and θ are the generative model parameters.

In practice, the prior $p_{\boldsymbol{\theta}}(\mathbf{z})$ is commonly Gaussian as the $D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right)$ has a closed form solution given a Gaussian prior, and sampling of $\mathbf{z}$ can be accomplished via a reparameterization trick.

Variational autoencoders make the assumption that data is generated via a random process involving an unobserved continuous random variable $\mathbf{z}$. The random variable $\mathbf{z}$ is defined as:

$$\mathbf{z} \backsim q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \tag{8}$$

where $\mathbf{z}$ has the probability distribution of the approximate posterior $q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})$. The approximate posterior $q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})$ is commonly defined by a multivariate Gaussian distribution with diagonal covariance:

$$q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)}\mathbf{I}) \tag{9}$$

The Gaussian distribution is then parameterized by an encoder or recognition network:

$$(\boldsymbol{\mu}^{(i)}, \log \boldsymbol{\sigma}^{(i)}) = EncoderNetwork_{\phi}(\mathbf{x}^{(i)}) \tag{10}$$

where the encoder network learns the parameters $\boldsymbol{\mu}^{(i)}$ and $\log \boldsymbol{\sigma}^{(i)}$ for each datapoint $\mathbf{x}^{(i)}$. When parameterized by a neural network, ϕ includes the parameters of the recognition network, such as the weights and biases.

Sampling can be achieved by using the reparameterization trick with a noise variable defined by a Gaussian distribution with $\mathbf{0}$ mean and unit variance $\boldsymbol{I}$, such that:

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}) \tag{11}$$

where z is sampled as follows:

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}. \tag{12}$$

and $\odot$ is the element-wise product.

3 Expected Latent Representation

We take advantage of the parameters learned by the recognition network to define the expected latent representation. Given that $\mathbf{z}$ is defined by a multivariate Gaussian distribution with diagonal covariance we define the expected value of $\mathbf{z}$ for datapoint $\mathbf{x}^{(i)}$ as:

$$\mathbb{E}_{\mathbf{z} \backsim q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})}[\mathbf{z}] = \boldsymbol{\mu}^{(i)} \tag{13}$$

where we have fixed parameters ϕ. We can think of this as the expected latent representation for $\mathbf{x}^{(i)}$ given the fixed weights and biases of the recognition network, where $\boldsymbol{\mu}^{(i)}$ represents the expected value of the latent variable $\mathbf{z}$ for $\mathbf{x}^{(i)}$ or the expected position of the $\mathbf{x}^{(i)}$ in the latent space. Due to indentifiability problems in latent variable models [2] the expected latent representation will be different for different parameters ϕ.

The expected latent representation has several advantages for anomaly detection. First, it summarizes the distribution responsible for the latent representation, providing a clear picture of what to expect in the latent space for a specific input. Second, it avoids extreme values that can be generated from the sampling process as samples may happen to be drawn from the extremes. Third, the expected latent representation is deterministic as $\boldsymbol{\mu}^{(i)}$ is the same for each $\mathbf{x}^{(i)}$ given fixed ϕ, in contrast to $\mathbf{z}$ which is stochastic and generates a different $\mathbf{z}$ each time $\mathbf{x}^{(i)}$ is evaluated. Although sampling is critical during training and useful for generating new examples, it is problematic for anomaly detection since it adds noise to the models of normality, making it harder to pick out anomalies.

3.1 Reconstruction Error of Expected Latent Representation

Anomaly detection via variational autoencoders involves approximating the reconstruction error of an example by taking a sample from the approximate posterior and feeding it to the generative network to estimate the expected reconstruction error. However, once we have a trained network with fixed weights and biases we do not need to use a sample. Instead, we take the expected latent representation $\boldsymbol{\mu}^{(i)}$ and extend it to reconstruction error to compute the reconstruction error of the expected latent representation.

Given a fully trained network with fixed weights and biases, we define the negative reconstruction error of the expected latent representation as:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z} = \boldsymbol{\mu}^{(i)}) \tag{14}$$

or to simplify the notation:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\boldsymbol{\mu}^{(i)}) \tag{15}$$

We use the negative reconstruction error of the expected latent representation to approximate the negative expected reconstruction error:

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})] \approx \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\boldsymbol{\mu}^{(i)}) \tag{16}$$

From another perspective, if we were to take enough samples from the approximate posterior for $\mathbf{x}^{(i)}$, eventually the average value will approach $\boldsymbol{\mu}^{(i)}$, and subsequently the negative reconstruction error of that average value will approach the negative reconstruction error produced by $\boldsymbol{\mu}^{(i)}$.

3.2 Variational Lower Bound via Reconstruction Error of Expected Latent Representation

Given a fully trained network with fixed weights and biases, we approximate the variational lower bound of the marginal likelihood of $\mathbf{x}^{(i)}$ as:

$$\widetilde{\mathcal{L}}^{ED}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) = -D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})\|p_{\boldsymbol{\theta}}(\mathbf{z})\right) + \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\boldsymbol{\mu}^{(i)}) \tag{17}$$

We adopt Equation 17 as an anomaly detection measure, using it to detect anomalies once the variational autoencoder has been fully trained using Equation 6. We refer to this method of training and testing as the Expected Latent Representation Decoder Variational Autoencoder (ED-VAE).

3.3 Additional Benefits of Expected Latent Representation

The expected latent representation has the advantage of mitigating negative effects from overlapping approximate posteriors. Figure 1 provides an illustrative example of why overlapping approximate posteriors can cause difficulties for anomaly detection. We visualize the probability density functions of three different Gaussian distributions denoted as $\mathcal{N}_1$, $\mathcal{N}_2$, and $\mathcal{N}_3$. The amount of overlap between these three distributions is fairly characteristic of the typical behavior of approximate posteriors, depending on the amount of posterior collapse. If we consider that the generative model is fed the

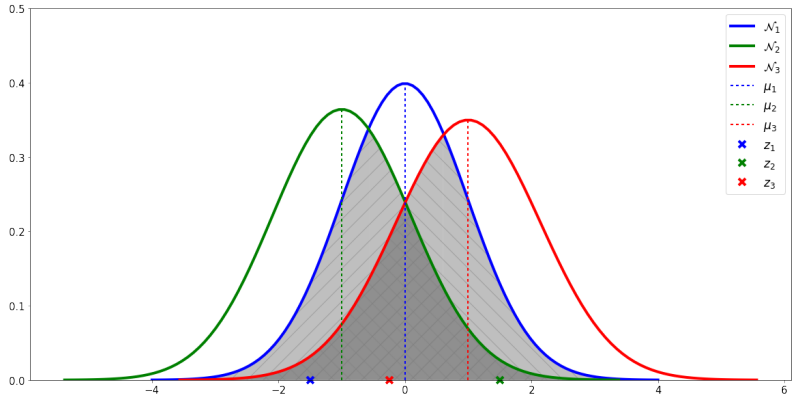

Figure 1: Overlapping Approximate Posteriors

latent representation sampled from these approximate posteriors to reconstruct the original input, then we know the latent representation must contain some sort of discriminative information. However, if we take a random sample from these three distributions we could generate samples like z_1, z_2, and z_3, where the latent representations would cause discrimination issues as they could fall in any order on the x-axis. Instead, we can use the expected latent representation μ_1, μ_2, and μ_3 in place of the random samples, eliminating the discrimination issue.

The expected latent representation also has the advantage of removing variation in the latent representation introduced by the sampling process, changing this from a stochastic process to a deterministic one. This is beneficial for anomaly detection as it has the effect of removing variation from the reconstruction error, providing a clear view of normal and anomalous behavior. Additionally, this provides consistency in evaluating potential anomalies.

The expected latent representation also avoids relying on a single sample that may produce a latent representation at the edges of the distribution defined by the approximate posterior. This avoids extreme values that could adversely impact anomaly detection.

In the end, these three considerations help create a stable representation of normality, making it easier detect differences from the normal model.

4 Experiments

To empirically evaluate our proposed approach we used common benchmark datasets, comparing the performance of ED-VAE to variational autoencoders and comparable models. Additional experiments can be found in the Appendix.

4.1 Experimental Setup

Training was performed on a set of normal examples, while the test set was a mixture of normal and anomalous examples. To create a more realistic evaluation for an anomaly detection scenario, we sampled the anomalies so that a certain ratio $p = 0.2$ of the test set were anomalies. In other words, for each test set 20% percent of the test set are anomalies. Keeping the ratio of anomalies to normal examples the same between datasets provided the added benefit of simplifying comparisons between the respective performance of each dataset as the PR-AUC that represents random performance is constant. In this case, a model demonstrates random performance with PR-AUC of 0.2 where $p = 0.2$.

We evaluated each of our proposed methods on the MNIST [10] and Fashion-MNIST [13] datasets, both common benchmarks for evaluating anomaly detection techniques. For each individual class, we treated that class as normal and the remaining classes as anomalous.

Hyperparameters (e.g. # of latent dimensions, filter size, kernel, stride, and learning rate) were chosen via grid based search using a validation set with normal and anomalous examples. We did

not exhaustively optimize the hyperparameters due to the large number of possible hyperparameter configurations and the heavy compute requirements.

For both the MNIST and Fashion-MNIST dataset the encoder of the VAE included two convolutional layers with a filter size of 32, a 3x3 kernel, a stride of 2, a fully connected layer of size 1568, and 32 latent dimensions. The decoder reversed the operations of the encoder using convolutional transpose layers [5, 14]. All intermediate activation functions were ReLU, the decoder output activation functions were sigmoid, and the activation functions for $\boldsymbol{\mu}$ and $\log \boldsymbol{\sigma}$ were linear. The network structure of the encoder and decoder is fairly rudimentary, to simplify performance comparisons with competing models.

We chose Principal Component Analysis (PCA), Kernal PCA (KPCA), Deep Structured Energy Based Models (DSEBM) [15], Autoencoders (AE), and Variational Autoencoders (VAE) as comparable models. The AE was structured similarly to the VAE, but the sampling layer was replaced with an encoded layer. Each comparable model had the same number of latent or encoded dimensions as the variational autoencoder.

We evaluated each model for 10 repetitions and report the mean and SEM for each class, and the mean for each dataset. The anomalies were re-sampled for each of the repetitions. AUC and PR-AUC were chosen as performance measures and the anomalies were treated as the positive class.

Training was performed to a maximum of 1000 epochs with early stopping and a patience of 20 epochs, using the ELBO as a stopping criteria. We split each training set into training and validation sets (80/20), with the validation set used for the early stopping. We used the Adam optimizer with a learning rate of 0.0001 and a batch size of 128.

We provide the source code for the main experiments at `https://github.com/anon12a/ed-vae/`, which were executed on Google Colab with an approximate compute time of 6 hours per dataset.

4.2 Results

Table 1 reports the AUC for the MNIST and Fashion-MNIST datasets and Table 2 reports the PR-AUC, where ED-VAE outperformed VAE from the perspective of AUC and PR-AUC for the majority of classes.[1] Additionally, ED-VAE outperformed the comparable models for the vast majority of our evaluations based on AUC, and also outperformed the comparable models for the majority of our evaluations based on PR-AUC. When it did not, its performance was only slightly worse or similar. Additional results can be found in the supplementary material of the Appendix.

5 Discussion

Reconstruction error plays an important role in anomaly detection via variational autoencoders; it is sometimes adopted as the sole metric for discovering anomalies. However, computing the reconstruction error of an example relies on sampling from the approximate posterior of a given example before feeding the sample into the encoder to approximate the expected reconstruction error. This sampling process can be at odds with anomaly detection task, as it creates blurry models of normality, which can in turn make it harder to pick out anomalies. Our results strongly suggest that we can improve the anomaly detection process in variational autoencoders by replacing sample-based estimates with the reconstruction error of the expected latent representation.

There are a few important considerations for this approach:

How much the reconstruction error of the expected latent representation improves anomaly detection in variational autoencoder depends on the variance parameter of the approximate posteriors. If the variance is small then the sample will likely be close to the mean, compared to approximate posteriors with large variance where a random sample could be further from the mean. The closer the samples are to the mean, the closer the reconstruction of the expected latent representation is to the expected reconstruction error.

[1] At first glance the PR-AUC (and AUC for the MNIST dataset) of class 5 seems unusual as PCA performed better than ED-VAE in both benchmark datasets. We initially suspected this might be a bug in the code given that it is specifically class 5 for both datasets, but our investigations revealed that this was not the case. This footnote will be removed from the final submission and is only included for the benefit of the reviewers.

Table 1: Performance evaluation of ED-VAE and comparable models based on AUC. ED-VAE almost always outperforms VAE and other comparable models.

dataset	class	pca	kpca	dsebm	ae	vae	ed-vae
MNIST	0	99.2±0.0	99.0±0.0	53.7±7.8	96.9±0.1	99.7±0.0	**99.8±0.0**
	1	99.9±0.0	99.8±0.0	98.2±0.3	98.7±0.0	99.9±0.0	**99.9±0.0**
	2	92.4±0.2	89.7±0.3	51.8±4.2	78.1±0.2	93.0±0.4	**95.3±0.4**
	3	95.2±0.1	93.9±0.2	50.8±1.2	85.8±0.3	94.6±0.4	**95.8±0.3**
	4	94.3±0.1	94.6±0.2	69.3±2.1	88.2±0.3	95.0±0.2	**96.3±0.3**
	5	**97.4±0.1**	95.0±0.1	55.7±0.8	72.5±0.4	95.8±0.2	96.7±0.2
	6	98.4±0.1	97.3±0.1	58.5±4.2	86.8±0.2	98.9±0.1	**99.3±0.1**
	7	97.1±0.1	96.4±0.2	75.7±0.9	91.2±0.3	96.5±0.2	**97.3±0.2**
	8	85.5±0.5	85.4±0.5	46.4±3.2	78.0±0.8	89.5±0.4	**91.1±0.5**
	9	96.4±0.1	95.4±0.1	66.3±1.7	88.0±0.2	97.6±0.1	**98.3±0.1**
	avg	95.6	94.7	62.6	86.4	96.0	**97.0**
Fashion-MNIST	0	89.8±0.2	90.6±0.2	87.3±0.6	89.7±0.4	90.7±0.2	**91.1±0.2**
	1	98.5±0.1	98.2±0.1	78.4±4.2	98.0±0.2	98.6±0.1	**98.7±0.1**
	2	88.7±0.3	89.1±0.2	83.3±0.4	86.7±0.4	88.4±0.3	**89.1±0.2**
	3	91.9±0.2	92.6±0.4	91.3±0.5	90.5±0.5	92.4±0.3	**92.8±0.3**
	4	88.5±0.4	88.5±0.4	87.5±0.6	88.7±0.3	90.0±0.4	**90.9±0.4**
	5	88.6±0.3	88.8±0.3	87.2±0.3	87.6±0.2	89.3±0.3	**89.5±0.2**
	6	81.3±0.4	82.1±0.3	75.6±0.6	77.6±0.5	82.3±0.4	**83.1±0.4**
	7	98.4±0.1	98.4±0.1	95.3±1.8	98.1±0.1	98.2±0.1	**98.5±0.1**
	8	83.7±0.2	84.0±0.3	79.6±1.1	82.1±0.7	83.5±0.4	**84.8±0.4**
	9	97.1±0.2	96.6±0.2	**97.3±0.2**	97.1±0.2	96.3±0.3	97.1±0.2
	avg	90.7	90.9	86.3	89.6	91	**91.6**

Table 2: Performance evaluation of ED-VAE and comparable models based on PR-AUC. ED-VAE outperforms VAE and other comparable models for most classes and helps close the gap between VAE and comparable models when that is not the case.

dataset	class	pca	kpca	dsebm	ae	vae	ed-vae
MNIST	0	96.3±0.1	95.6±0.2	31.9±9.1	88.4±0.4	98.8±0.1	**99.1±0.1**
	1	99.5±0.0	99.2±0.0	93.6±1.0	95.5±0.2	99.6±0.0	**99.6±0.0**
	2	80.9±0.4	74.4±0.5	27.8±3.4	49.9±0.5	84.7±0.7	**88.3±0.7**
	3	80.8±0.4	80.0±0.5	25.4±0.9	61.4±0.8	86.0±0.7	**88.4±0.6**
	4	87.6±0.3	86.4±0.3	44.6±2.2	66.7±0.6	89.5±0.4	**91.2±0.5**
	5	**90.8±0.3**	83.7±0.4	30.1±0.7	44.4±0.5	88.5±0.4	90.0±0.4
	6	94.8±0.1	91.3±0.2	32.1±3.0	54.7±0.4	96.2±0.3	**97.3±0.2**
	7	91.3±0.3	89.5±0.4	58.7±0.9	75.2±0.5	91.3±0.4	**92.5±0.3**
	8	69.9±0.6	68.2±0.6	21.4±2.3	46.1±2.1	80.9±0.6	**84.5±0.5**
	9	87.9±0.2	83.2±0.3	43.5±1.5	62.2±0.6	92.0±0.3	**93.7±0.2**
	avg	88.0	85.1	40.9	64.4	90.8	**92.5**
Fashion-MNIST	0	69.9±0.4	**73.1±0.5**	67.4±1.2	71.2±0.5	72.2±0.5	72.2±0.6
	1	94.0±0.2	93.4±0.2	63.9±5.3	91.9±0.7	94.4±0.2	**94.8±0.2**
	2	70.8±0.4	**75.1±0.3**	62.5±0.7	69.4±1.0	74.2±0.3	74.9±0.4
	3	79.3±0.6	**82.9±0.6**	81.6±0.7	80.2±0.8	82.4±0.6	82.8±0.6
	4	76.2±0.5	76.3±0.6	71.2±1.6	74.0±0.5	78.6±0.6	**80.0±0.5**
	5	**82.5±0.3**	81.1±0.2	76.6±0.5	76.3±0.3	81.6±0.3	81.1±0.2
	6	59.9±0.7	62.4±0.7	47.0±1.5	50.9±0.7	64.0±0.7	**64.5±0.7**
	7	96.3±0.1	96.1±0.1	91.2±2.7	94.7±0.1	95.8±0.2	**96.3±0.1**
	8	61.4±0.4	**65.8±0.4**	46.5±1.8	52.7±1.3	59.3±0.7	60.8±0.7
	9	**93.2±0.4**	91.3±0.4	91.1±0.6	91.8±0.9	91.1±0.4	92.0±0.4
	avg	78.4	79.8	69.9	75.3	79.4	**79.9**

Success of the reconstruction error of the expected latent representation can be impacted by the amount of overlap between approximate posteriors, which occurs frequently with posterior collapse. Posterior collapse occurs when the approximate posterior of a latent variable (or dimensions of a latent variable) closely matches the prior. The closer the approximate posterior is to the prior the more posterior collapse. In the extreme the approximate posterior is identical to the prior, leading to an uninformative latent variable given an uninformative prior. Using the expected latent representation rather than sampling from the approximate posterior reduces the chances of overlap from posterior collapse causing discrimination issues. Although two approximate posteriors with similar means and similar variances can easily overlap for the majority of their probability density functions, their means are a distinguishing factor.

Approximating the expected reconstruction error is a stochastic process. Using the expected latent representation of a datapoint to approximate the expected reconstruction error creates a deterministic process where the reconstruction error will be same for any given datapoint that passes through a fully trained variational autoencoder with fixed weights and biases. This removes variation caused by single sample or multiple sample approximations of the expected reconstruction error where the reconstruction error will be different for each sample. This is a significant advantage for the anomaly detection task as we are generally not interested in approaches that label a datapoint an anomaly or normal depending on variation due to sampling.

An alternative to approximating the expected reconstruction error via a single sample is to sample multiple times per datapoint and compute the average reconstruction error to approximate the expected reconstruction error. This is computationally expensive depending on the number of samples and the number of examples. Sampling has a computational complexity of $n \times l$ where n is the number of examples and l is the number of samples per example. This can become prohibitively expensive: if $n = 10000$ with $l = 1000$ samples per datapoint, it would require 1000000 computations to compute a relatively good approximation of the expected reconstruction error. This becomes even more expensive in complex models with additional layers of latent variables, such as the recently proposed N-VAE [12]. The reconstruction error of the expected latent representation offers a computationally efficient method of approximating the expected reconstruction error in fully trained models.

Similar Results Between ED-VAE and VAE Similar results between ED-VAE and VAE are probably the result of similar reconstruction error. There are several reasons this might happen for a given example. One might be that the sampling process was lucky and the sample taken to approximate the expected reconstruction error was close to the reconstruction error of the expected latent representation. A more likely scenario is that the approximate posteriors for the test examples have variances that approach zero, resulting in expected latent representations that are almost identical to samples taken using the associated approximate posterior. This can occur when there is almost no posterior collapse in the approximate posterior for one or multiple latent dimensions as those dimensions are relatively more important to producing accurate reconstructions, causing reconstruction error to outweigh the regulation effect of $D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})\|p_{\boldsymbol{\theta}}(\mathbf{z})\right)$.

6 Related Work

Several approaches have been proposed for anomaly detection via variational autoencoders. An and Cho [1] propose reconstruction probability for anomaly detection. Soelch et al. [11] briefly evaluate different measurements of likelihood produced via variational autoencoders as measures of normality for on-line anomaly detection in time series data. Choi et al. [4] explore several measures of likelihood produced by variational autoencoders, making comparisons to an ensemble-based out-of-distribution detection technique that they propose where out-of-distribution examples are detected via the Watanabe Akaike Information Criterion.

Our approach differs from these approaches by changing anomaly detection via variational autoencoders from a stochastic process to a deterministic process. This creates stable representations of normality that make it easier to detect differences from the normal model, leading to improved anomaly detection. An advantage of our proposed approach is it can be implemented using any variation of variational autoencoders, as long as the approximate posterior is parameterized during training and testing. Additionally, given the previous statement, our approach can be adopted to any variational autoencoder based anomaly detection approach that uses sampling to compute an anomaly detection measure.

It is also worth noting that although $\boldsymbol{\mu}^{(i)}$ is commonly used by practitioners for downstream tasks unrelated to anomaly detection (e.g. dimensionality reduction followed by clustering), this is done without any discussion of what $\boldsymbol{\mu}^{(i)}$ represents from a theoretical perspective. Thinking of $\boldsymbol{\mu}^{(i)}$ as the expected latent representation of $\boldsymbol{x}^{(i)}$ given fixed parameters ϕ, provides a conceptual framework for adopting the expected latent representation for downstream tasks, while also providing justification for adopting the expected latent representation to improve the anomaly detection task.

7 Conclusions

We defined the concept of the expected latent representation to improve anomaly detection in variational autoencoders, by taking advantage of the parameters of the approximate posteriors that define the latent representation of each datapoint, given a trained variational autoencoder. This removes variation in the latent representation that is introduced from the sampling process. It also helps distinguish latent representations of different examples that may have overlapping approximate posteriors with similar means and variances. This provides a clear view of what we expect to see in the latent space for a given example rather than relying on a single sample that may produce a latent representation at the edges of the distribution defined by the approximate posterior.

Additionally, we proposed a computationally efficient method for approximating the expected reconstruction error given a trained variational autoencoder, as an alternative to the current practice of approximating the expected reconstruction error via single or multiple samples. This is accomplished by extending the concept of the expected latent representation to reconstruction error, by feeding the expected latent representation to the decoder/generative model of the variational autoencoder to produce the reconstruction error of the expected latent representation. This is valuable for anomaly detection as it removes a source of variation from the reconstruction error, allowing for easier discrimination between normal and anomalous examples.

Finally, we performed a comprehensive evaluation of the variational lower bound approximated via the reconstruction error of the expected latent representation, as an anomaly detection measure, and empirically demonstrated that it produced superior results for anomaly detection, when compared to traditional sampling techniques. This strongly suggests that there is value in taking this approach for anomaly detection via variational autoencoders.

References

[1] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. Report, SNU Data Mining Center, Seoul National University, 2015.

[2] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–58, 2009.

[4] Hyunsun Choi, Eric Jang, and Alexander A. Alemi. WAIC, but why? Generative ensembles for robust anomaly detection. *arXiv:1810.01392*, 2018.

[5] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv:1603.07285*, 2016.

[6] Irina Higgins, Loic 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. In *ICLR*, 2017.

[7] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv:1401.4082*, 2014.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013.

[9] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *arXiv:1906.02691*, 2019.

[10] Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[11] Maximilian Soelch, Justin Bayer, Marvin Ludersdorfer, and Patrick van der Smagt. Variational inference for on-line anomaly detection in high-dimensional time series. *arXiv:1602.07109*, 2016.

[12] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[13] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017.

[14] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on Computer Vision and pattern recognition*, pages 2528–2535. IEEE, 2010.

[15] Shuangfei Zhai, Yu Cheng, Weining Lu, and Zhongfei Zhang. Deep structured energy based models for anomaly detection. In *Proceedings of The 33rd International Conference on Machine Learning (ICML)*, 2016.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 4.1 for URL.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.1.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We report the SEM for the main experiments in Section 4.2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.1 and in code accessed via the URL in Section 4.1.
   (b) Did you mention the license of the assets? [Yes] See code accessed via the URL in Section 4.1.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Code for main experiments via the URL in Section 4.1.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

The following is supplementary material that expands on our evaluation of the reconstruction error of the expected latent representation. We evaluated the reconstruction error of the expected latent representation as an anomaly detection measure on its own rather than as part of the variational lower bound approximated via the expected latent representation.

A.1 Multiple Samples

We compared the performance of the reconstruction error of the expected latent representation to single sample and multiple sample approximations of the expected reconstruction error on the MNIST (Figure 2) and the fashion-MNIST dataset (Figure 3). We also included the performance of the best and worst performing sample drawn from the multiple sample approximation. The models were trained using the same procedure laid out previously and were evaluated for 20 repetitions. We report the mean and SEM of the AUC and PR-AUC for each class. 100 samples were drawn for each datapoint $\mathbf{x}^{(i)}$ for the multiple sample approximation as initial tests indicated little or no difference in performance after 100 samples. The reconstruction error of the expected latent representation performed better than the single sample approximation while also performing the same or better than the multiple sample approximation, for the majority of the evaluations. It is worth noting that the worst performing sample from the multiple sample approximation performed as poorly as the single sample approximation, which strongly suggests single sample approximations should not be used for anomaly detection, even though this is a commonly adopted strategy.

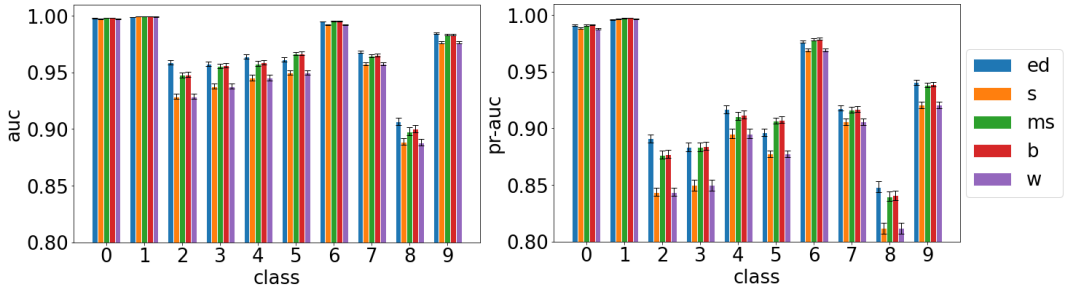

Figure 2: Performance comparison of the reconstruction error of the expected latent representation (ed), single sample approximation (s), multiple sample approximation (ms), best sample (b), and worst sample (w) for the MNIST dataset.

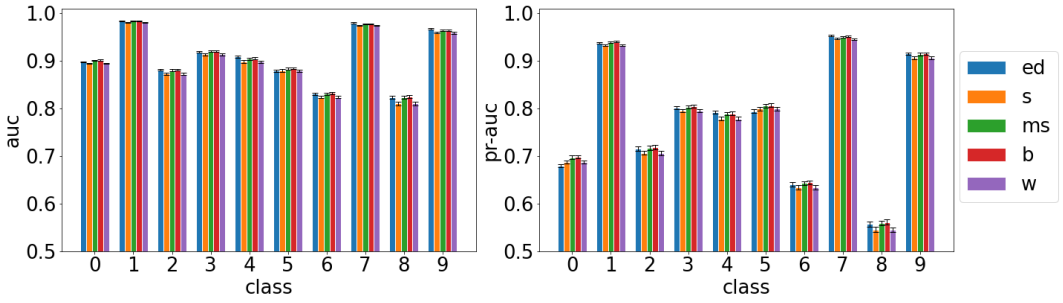

Figure 3: Performance comparison of the reconstruction error of the expected latent representation (ed), single sample approximation (s), multiple sample approximation (ms), best sample (b), and worst sample (w) for the Fashion-MNIST dataset.

A.2 Posterior Collapse

Approaches that are robust to the negative impacts of posterior collapse on anomaly detection are valuable tools for improving anomaly detection with variational autoencoders as posterior collapse is a common problem with variational autoencoders. Although numerous variations of variational autoencoders have been proposed to limit posterior collapse this is still an ongoing area of research.

We evaluated the effect of posterior collapse on the reconstruction error of the expected latent representation and the single sample approximation of the reconstruction error. We artificially forced the posterior to collapse by adopting the objective function of β-VAE [6] (see Equation 18) for training. We also modified Equation 18 to use the reconstruction error of the expected latent representation (see Equation 19), adopting it as an anomaly detection measure once we have a fully trained model.

$$\widetilde{\mathcal{L}}^{\beta}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) = -\beta D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right) + \frac{1}{L}\sum_{l=1}^{l}\left(\log p_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)}\right)\right) \tag{18}$$

$$\widetilde{\mathcal{L}}^{\beta-ED}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) = -\beta D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right) + \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\boldsymbol{\mu}^{(i)}) \tag{19}$$

We evaluated β values from 0 to 1.5 at intervals of 0.1. Equation 6 is equivalent to to Equation 18 when $\beta = 1$, as is the case for Equation 17 and Equation 19. Each model was trained using the same procedure laid out previously and were evaluated for 10 repetitions. We report the mean AUC and PR-AUC for each class for each beta value. Figure 4 and Figure 5 report the results for the MNIST dataset and Figure 6 and Figure 7 report the results for the Fashion-MNIST dataset. Increasing the value of β increases the regularization power of $D_{KL}\left(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)}) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right)$, thus increasing posterior collapse by encouraging the approximate posterior to be closer to the prior. In general, the performance gained from adopting the reconstruction error of the expected latent representation increased up to a point where artificially collapsing the posterior no longer had a meaningful effect. This strongly supports our argument that ED-VAE is more effective when there is more posterior collapse. It also follows that it is more effective when there is more overlap between the approximate posteriors, as there is more overlap with more posterior collapse.

A.3 Reconstruction Error

We compared the performance of the reconstruction error of ED-VAE and VAE on the MNIST and Fashion-MNIST dataset (Table 3). The experimental setup is identical to the experiments that produced the results in Table 1 and Table 2, where we compared ED-VAE to comparable models. The reconstruction error is pulled directly from that evaluation. ED-VAE outperformed VAE for the majority of classes.

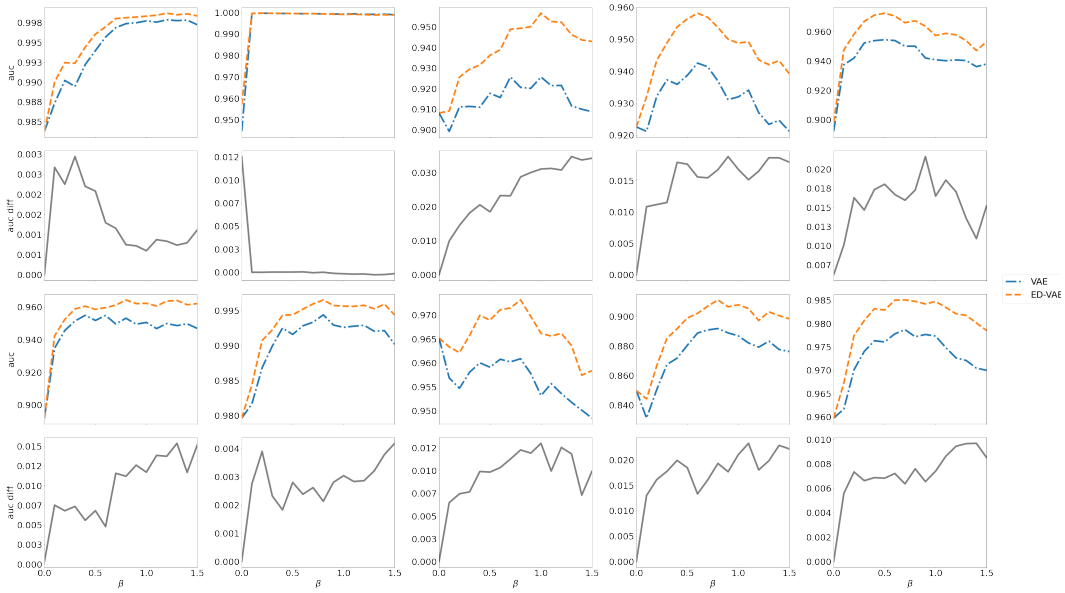

Figure 4: Performance evaluation of the effect of posterior collapse on the reconstruction error of the expected latent representation on the MNIST dataset. Performance is measured via AUC. Classes are ordered from left to right with the Class 0 in the top left. The row immediately below visualizes the difference between the reconstruction error of ED-VAE and VAE. In general, the difference increases between ED-VAE and VAE with ED-VAE outperforming VAE, up until a point where artificially collapsing the posterior no longer has a meaningful effect.

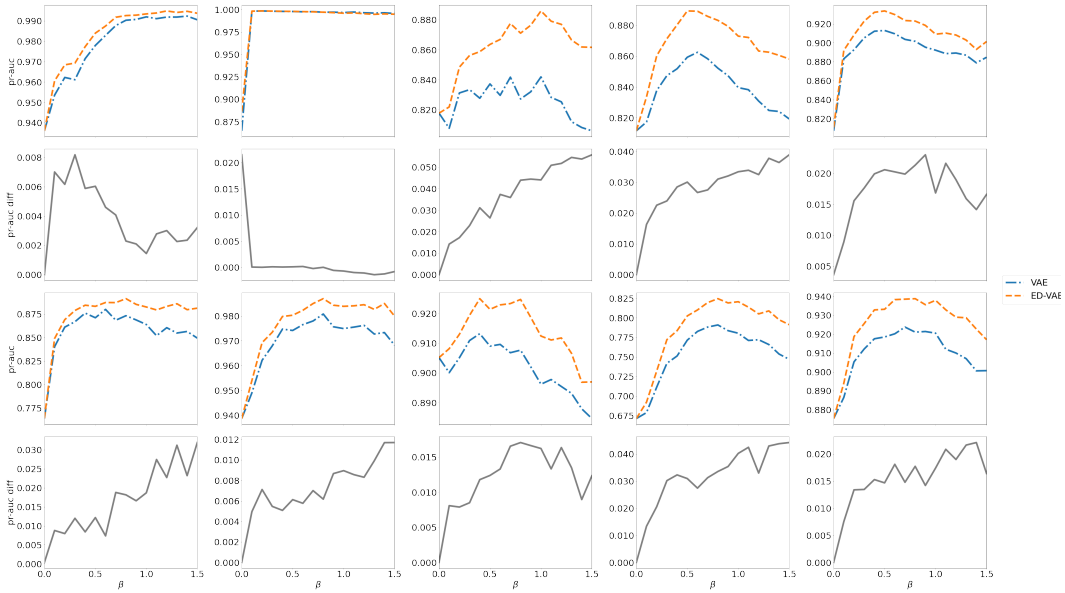

Figure 5: Performance evaluation of the impact of posterior collapse on the reconstruction error of the expected latent representation on the MNIST dataset. Performance is measured via PR-AUC.

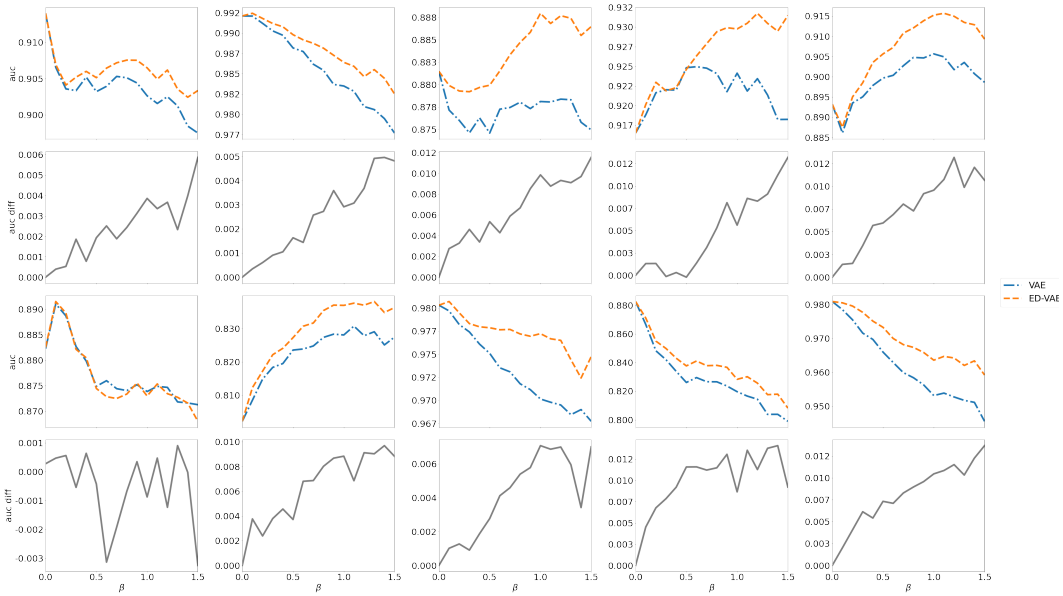

Figure 6: Performance evaluation of the impact of posterior collapse on the reconstruction error of the expected latent representation on the Fashion-MNIST dataset. Performance is measured via AUC.

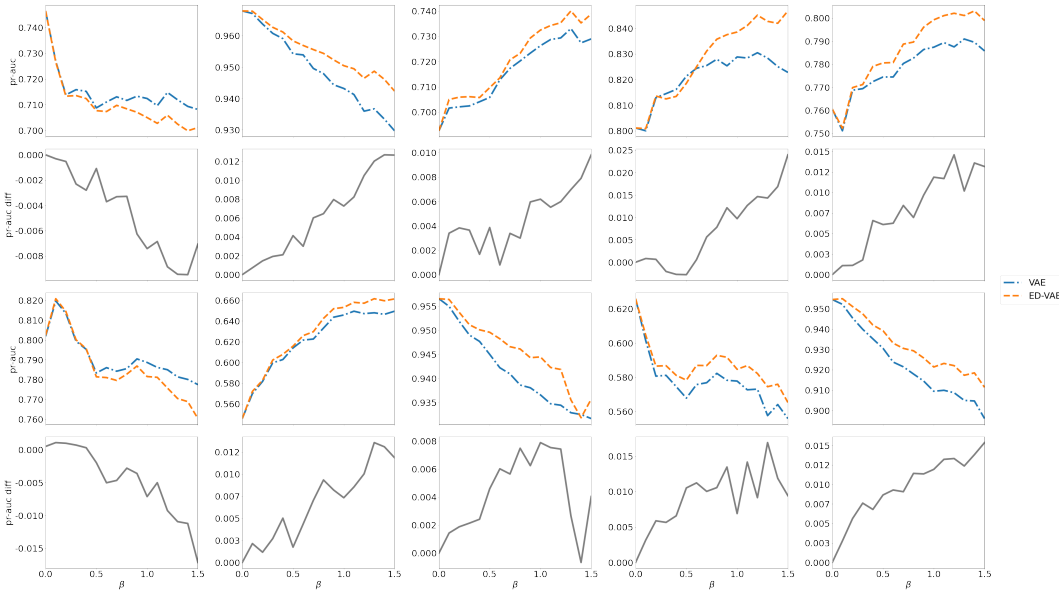

Figure 7: Performance evaluation of the impact of posterior collapse on the reconstruction error of the expected latent representation on the Fashion-MNIST dataset. Performance is measured via PR-AUC.

Table 3: Performance evaluation of the reconstruction error of ED-VAE and VAE where the performance is measured by AUC and PR-AUC. The reconstruction error of the expected latent representation performed better for the majority of the classes for both the MNIST and Fashion-MNIST dataset.

		auc		pr-auc	
dataset	class	vae	ed-vae	vae	ed-vae
MNIST	0	99.7±0.0	**99.8±0.0**	98.8±0.1	**99.1±0.1**
	1	99.9±0.0	**99.9±0.0**	**99.7±0.0**	99.6±0.0
	2	92.7±0.4	**95.6±0.3**	84.2±0.7	**88.6±0.7**
	3	94.2±0.3	**95.7±0.3**	85.8±0.6	**88.5±0.5**
	4	94.7±0.3	**96.4±0.4**	89.8±0.5	**91.7±0.6**
	5	95.9±0.2	**96.9±0.2**	89.4±0.3	**91.0±0.4**
	6	99.1±0.1	**99.4±0.1**	96.7±0.3	**97.7±0.2**
	7	96.0±0.2	**97.0±0.2**	90.9±0.3	**92.1±0.3**
	8	89.5±0.4	**91.1±0.6**	81.8±0.6	**85.6±0.5**
	9	97.8±0.1	**98.6±0.0**	92.9±0.2	**94.6±0.1**
	avg	96.0	**97.0**	91.0	**92.9**
Fashion-MNIST	0	89.6±0.2	**90.1±0.2**	**68.7±0.6**	68.4±0.6
	1	98.2±0.1	**98.5±0.1**	93.7±0.2	**94.3±0.2**
	2	87.8±0.3	**88.6±0.2**	72.2±0.4	**72.8±0.5**
	3	91.6±0.2	**92.1±0.2**	80.1±0.6	**80.6±0.5**
	4	90.1±0.4	**91.2±0.4**	78.1±0.6	**79.3±0.6**
	5	88.5±0.3	**88.5±0.2**	**81.0±0.3**	80.3±0.2
	6	82.0±0.5	**82.7±0.4**	63.1±0.7	**63.5±0.7**
	7	97.7±0.1	**98.1±0.1**	95.0±0.3	**95.7±0.2**
	8	81.3±0.3	**82.7±0.4**	55.0±0.6	**56.5±0.7**
	9	95.9±0.3	**96.9±0.2**	90.5±0.5	**91.5±0.4**
	avg	90.3	**91**	77.7	**78.3**